BACHELORARBEIT

von Martina Unterburger

Sommersemester 2009

SOLAR II

Komplexe Modellierung von beruflichen Allergierisiken unter Berücksichtigung der Problematik fehlender Daten

---

Ludwig-Maximilians-Universität München
Institut für Statistik

In Zusammenarbeit mit
Institut und Poliklinik für Arbeits-, Sozial- und Umweltmedizin des Klinikums der Universität München

Betreuer:
PD Dr. Christian Heumann, Ludwig-Maximilians-Universität München
Prof. Dr. Katja Radon, MSc und Jessica Kellberger, Institut und Poliklinik für Arbeits-, Sozial- und Umweltmedizin des Klinikums der Universität München

BACHELORARBEIT

von Martina Unterburger

Sommersemester 2009

SOLAR II

Komplexe Modellierung von beruflichen Allergierisiken unter Berücksichtigung der Problematik fehlender Daten

---

Ludwig-Maximilians-Universität München
Institut für Statistik

In Zusammenarbeit mit
Institut und Poliklinik für Arbeits-, Sozial- und Umweltmedizin des Klinikums der Universität München

Betreuer:
PD Dr. Christian Heumann, Ludwig-Maximilians-Universität München
Prof. Dr. Katja Radon, MSc und Jessica Kellberger, Institut und Poliklinik für Arbeits-, Sozial- und Umweltmedizin des Klinikums der Universität München

Danksagung

Diese Bachelorarbeit entstand in Zusammenarbeit mit dem Institut und der Poliklinik für Arbeits-, Sozial- und Umweltmedizin des Klinikums der Universität München.
An dieser Stelle möchte ich mich bei den Betreuerinnen Prof. Dr. Katja Radon und Jessica Kellberger für das Ermöglichen der Arbeit und die Bereitstellung der Datensätze bedanken. Sie standen mir stets bei fachlichen und epidemiologischen Fragestellungen mit Rat und Tat zur Seite.
Ganz besonders möchte ich mich auch bei PD Dr. Christian Heumann für seine tatkräftige Unterstützung bedanken. Durch sein Bemühen und seinen Einsatz wurde maßgeblich zum Gelingen der Arbeit beigetragen.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

1.1 Zielsetzung

"Beruflich bedingte Allergien stehen seit Jahren an der Spitze der angezeigten Berufskrankheiten. Aufgrund des gleichzeitig wachsenden Anteils der Atopiker in der Bevölkerung und der schlechten Prognose von Berufsasthma besteht dringender Handlungsbedarf im Bereich der Primärprävention. Voraussetzung für wirksame Präventionsmaßnahmen sind jedoch fundierte Kenntnisse über individuelle und berufliche Risikofaktoren." [RADON et al. 2005]

Um diese fundierten Kenntnisse zu erlangen, wurde im Rahmen dieser Bachelorarbeit die SOLAR-Kohortenstudie mit drei Beobachtungszeitpunkten (ISAAC II, SOLAR und SOLAR II) betrachtet.

Ziel dieser Arbeit war die Modellierung beruflicher Allergierisiken. Dabei sollte unter anderem ein möglicher Zusammenhang zwischen den bisherigen Tätigkeiten und dem Auftreten von Allergischer Rhinitis und Asthma untersucht werden. Aufgrund der teilweise lückenhaften Angaben der Probanden war die Auseinandersetzung mit der Problematik fehlender Daten und der Anwendung verschiedener Imputationsmethoden ein weiteres Ziel dieser Arbeit.

Folgendes Zitat soll die "Selbstverständlichkeit von Imputationsmethoden" auf anderen Forschungsgebieten erläutern:

"If archaeologists threw away every piece of evidence, every tablet, every piece of pottery that was incomplete we would have entire cultures that disappeared from the historical record. (...) It is a ridiculous proposition because we can take all the partial sources, all the information in each fragment, and build them together to reconstruct much of the complete picture without any invention. Careful models for missingness allow us to do the same with our own fragmentary sources of data." [HONAKER und KING 2008]

1.2 Aufbau

Die vorliegende Arbeit ist folgendermaßen aufgebaut:

Kapitel 2 stellt das Studiendesign der Studien ISAAC II, SOLAR und SOLAR II in Bezug auf Aufbau und Ziele der Studien sowie Untersuchungsinstrumente und Probandenauswahl dar.

Die Kodierung der Tätigkeiten anhand des ISCO-Codes und die Zuweisung möglicher Expositionen unter Anwendung der asthmaspezifischen Job-Expose-Matrix wird in Kapitel 3 erläutert.

In Kapitel 4 wird in die Theorie zur Problematik und Behandlung fehlender Daten eingeführt. Dabei werden diverse Analyse- und Imputationsmethoden vorgestellt, sowie deren Vor- und Nachteile diskutiert.

Die Maßnahmen, die in Bezug auf Datenmanagement und Datenbereinigung der vorliegenden Studien notwendig waren, werden in Kapitel 5 dargestellt.

Kapitel 6 soll zunächst einen Überblick über die fehlenden Angaben im vorliegenden Datensatz geben. Weiterhin wird die praktische Umsetzung der Imputationsmethoden erläutert, durch die der Datensatz vervollständigt werden konnte.

Das grundlegende Vorgehen bei der Berechnung der Expositionen wird in Kapitel 7 beschrieben.

Kapitel 8 beginnt mit einer Einführung in die Theorie zur logistischen Regression und führt in die Schritte der Modellwahl ein, die in der vorliegenden Bachelorarbeit verwendet wurden. Anschließend wird auf unterschiedlichen Teildatensätzen das Auftreten von Allergischer Rhinitis und von Asthma modelliert. Die Erstellung und Interpretation dieser logistischen Regressionsmodelle war eines der Hauptziele dieser Arbeit.

Kapitel 9 fasst die wichtigsten Ergebnisse der erstellten logistischen Regressionsmodelle zusammen.

Abschließend werden in Kapitel 10 die Ergebnisse dieser Arbeit diskutiert und ein kurzer Ausblick auf mögliche weitere Schritte gegeben.

2 Studiendesign der zugrundeliegenden Studien

In Rahmen dieser Arbeit wurden folgende drei Studien am gleichen Kollektiv betrachtet:

- **ISAAC II**
  International **S**tudy on **A**sthma and **A**llergies in **C**hildhood
- **SOLAR**
  Studie in **O**st- und Westdeutschland zu beruf**l**ichen **A**llergie**r**isiken - 1. Follow-up
- **SOLAR II**
  Studie in **O**st- und Westdeutschland zu beruf**l**ichen **A**llergie**r**isiken - 2. Follow-up

Der zeitliche Untersuchungsablauf dieser Kohortenstudie ist Abbildung 2.1 zu entnehmen.

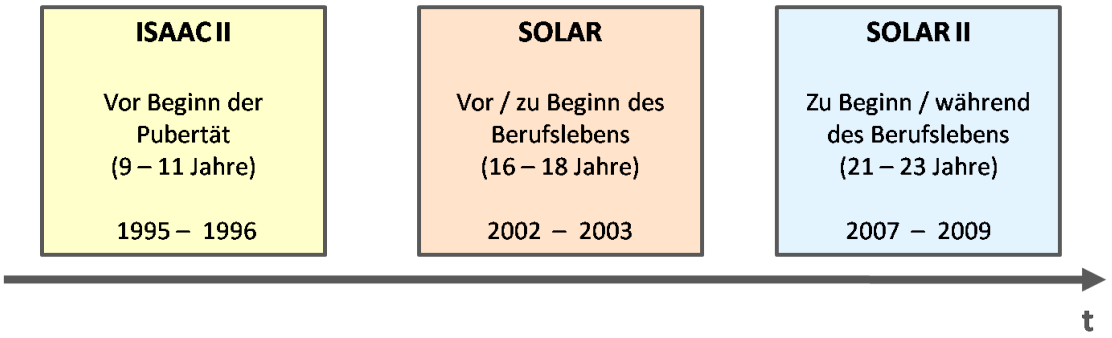

Abbildung 2.1: Zeitlicher Untersuchungsablauf der Studien

Im Folgenden wird auf das Studiendesign der zugrundeliegenden Studien eingegangen, indem jeweils Studienziele, Motivation, Probandenauswahl, Studienaufbau und Untersuchungsinstrumente erläutert werden. Die im folgenden Kapitel dargestellten Studienbeschreibungen basieren im Wesentlichen auf [RADON et al. 2005] und [RADON et al. 2008].

2.1 Basisuntersuchung: ISAAC-Studie

Studienziele und Motivation

Die Studie International Study on Asthma and Allergies in Childhood (ISAAC) hatte sich als Ziel gesetzt, die weltweite Prävalenz von asthmatischen und allergischen Erkrankungen sowie deren Symptome im Kindesalter zu beschreiben. Die Studie wurde in 56 Ländern und 119 Studienzentren weltweit durchgeführt, darunter München und Dresden. Durch die Wahl dieser Studienzentren sowohl in den neuen Bundesländern (Dresden) als auch in den alten Bundesländern (München) ergab sich die Gelegenheit zum Vergleich zweier genetisch vergleichbarer Populationen, die während der letzten 40 Jahre unterschiedlichen Lebensumständen und Umweltfaktoren ausgesetzt waren.
Die Studie bestand aus zwei Phasen. In den Jahren 1994/95 wurde Phase I durchgeführt, indem 6-7 und 13-14-jährige Kinder weltweit untersucht wurden. An dieser ersten Phase der Untersuchung nahmen über 250.000 Kinder im Alter von 6-7 Jahren und über 460.000 Jugendliche im Alter von 13-14 Jahren teil. Die Phase II der ISAAC-Studie wurde 1995/96 durchgeführt, um zusätzlich objektive Marker für Sensibilisierungen und allergische bzw. asthmatische Erkrankungen sowie Allergenexpositionen zu erfassen. Dabei wurden die Eltern von Kindern im Alter zwischen 5 und 7 sowie in der Altersgruppe 9-11 befragt. In Deutschland beteiligten sich drei Studienzentren (München, Dresden und Leipzig), insgesamt waren knapp 30 Studienzentren beteiligt. [RADON et al. 2005]
In die vorliegende Arbeit gingen ausschließlich Informationen aus der zweiten Phase (ISAAC II) ein. Dabei wurde sich auf die Altersgruppe der 9-11-Jährigen, die kurz vor Beginn der Pubertät befragt wurden, und die Studienzentren München und Dresden beschränkt.

Probandenauswahl

Zur Auswahl von potenziellen Teilnehmern für ISAAC II wurden zunächst Grundschulen in München und Dresden zufällig ausgewählt. [1] Daraufhin wurden jeweils die Schüler der 4. Klassenstufen zur Teilnahme an der Studie eingeladen. Durch dieses Auswahlverfahren gelang man zu einer repräsentativen Auswahl von Kindern, die sich durchschnittlich im Alter zwischen 9 und 11 Jahren befanden und wohnhaft in München oder Dresden waren. Von den ursprünglich eingeladenen 7.498 Schülern füllten 6.399 Eltern den Fragebogen aus (Teilnahmequote 85%).

Untersuchungsinstrumente

Die Probandendaten wurden anhand eines ausführlichen und einheitlichen Elternfragebogens erhoben. Dieser Fragebogen bestand in Phase II unter anderem aus Fragen zu soziodemographischen Daten, standardisierten Fragen zur Prävalenz von Asthma und sonstigen Atemwegserkrankungen des Kindes, Fragen zur Gesundheit der Familie und

[1] Schulen für körperlich oder geistig behinderte Kinder wurden nicht in die Studie aufgenommen. Weiterhin waren auch Schulen mit einem Ausländeranteil von über 80% von der Studie ausgeschlossen, da primär Kinder mit deutscher Nationalität untersucht werden sollten. Durch dieses Vorgehen konnten genetische und Lebensstil-Faktoren als Einflussgröße minimiert werden.

Fragen zur Exposition gegenüber Allergenen und Irritanzien im Wohnbereich (z.B. Milben oder Zigarettenrauch). Mit Hilfe dieser Fragen wurden mögliche Einflussfaktoren auf die Allergie- bzw. Asthmaentstehung, aber auch auf die Entwicklung von chronischer Bronchitis im Kindesalter erfasst und mögliche Zusammenhänge untersucht. Zusätzlich wurden bei einem Teil der Teilnehmer in Phase II klinische Untersuchungen durchgeführt (u.a. Hautallergietests, Blutuntersuchungen und Lungenfunktionsmessungen mit Methacholinprovokation).

Studienaufbau

Ursprünglich war die ISAAC-Studie als reine Querschnittsstudie geplant. Da diese Studie allerdings bereits einen entscheidenden Beitrag zur Beschreibung der weltweiten Prävalenz von Asthma und Allergien im Kindesalter leistete, erschien es sinnvoll, diese Studie fortzuführen. Weiterhin sprach für die Fortführung der Studie, dass eine prospektive Kohortenstudie mit Beginn im Kindesalter nötig ist, um zuverlässige Aussagen über allergische Erkrankungen, bronchiale Hyperreaktivität und Atemwegssymptome im Kindesalter zu machen und zusätzlich alle Berufstätigkeiten seit Ausbildungsbeginn und die Gründe für eventuelle Berufswechsel zu erfassen.
“Insbesondere bevölkerungsbezogene Kohortenstudien liefern einen besseren Aufschluss über die Häufigkeit von Berufsasthma, da die Verzerrung durch Selektion aus dem Beruf auf ein Minimum reduziert wird, außerdem ermöglichen sie ein besseres und genaueres Verständnis der Krankheit und ihrer Diagnostik, wobei diese Studien wesentlich teurer und zeitaufwändiger sind.” [RADON et al. 2005]

2.2 1. Follow-up: SOLAR-Studie

Studienziele und Motivation

Aus den oben genannten Gründen wurde in den Jahren 2002 und 2003 die Studie in Ost- und Westdeutschland zu beruflichen Allergierisiken (SOLAR) als Follow-up-Studie von ISAAC II angesetzt. Durch diese longitudinale Beobachtung der Studienteilnehmer eröffnete sich die Möglichkeit, Langzeitprognosen zu ermitteln und spezifische Risikofaktoren zu erforschen, so dass Maßnahmen zur Primärprävention von Allergien und Asthma am Arbeitsplatz entwickelt werden können.
Durchgeführt wurde die Studie durch das Institut und die Poliklinik für Arbeits-, Sozial- und Umweltmedizin des Klinikums der Universität München in Kooperation mit dem Dr. von Haunerschen Kinderspital des Klinikums der Universität München, der Kinderklinik der Carl-Gustav-Carus Universität Dresden, dem Institut für Epidemiologie der Universität Ulm sowie der Justus-Liebig-Universität Giessen.
Ziel der SOLAR-Studie war es, den Einfluss von allergischen und asthmatischen Erkrankungen auf die Berufswahl sowie den Einfluss des Berufs auf das Auftreten allergischer und asthmatischer Erkrankungen während der Pubertät zu untersuchen. Weiterhin sollten die Erkrankungen über den Verlauf der Pubertät bis zum Eintritt ins Berufsleben betrachtet werden. Dadurch konnte zum einen das Auftreten bzw. die Entwicklung von Atemwegsbeschwerden und Allergien in diesem Zeitraum erfasst werden. Zum anderen

konnten auch Berufswünsche sowie Ausbildungsziele zeitnah erfragt werden.

Probandenauswahl

Die Teilnehmer der ISAAC II-Studie aus München und Dresden wurden im Rahmen von SOLAR erneut zur Teilnahme gebeten. Diese Jugendlichen waren im Alter von 16-18 Jahren und konnten vor bzw. zu Beginn ihres Berufslebens befragt werden.
Von den 6.399 Teilnehmern von ISAAC II erklärten sich 85% der Eltern (N=5.438) zu einer erneuten Kontaktaufnahme bereit. Von diesen Personen konnten insgesamt 4.893 Personen im Rahmen der SOLAR-Studie tatsächlich kontaktiert werden (90%). Von den kontaktierten Personen nahmen 3.929 Jugendliche an der SOLAR-Studie teil (80%).
Für die Verknüpfung der erhobenen Daten aus SOLAR mit den bereits vorhandenen Daten aus der ISAAC II-Studie war das Einverständnis der Teilnehmer nötig, um die Daten als Längsschnittdatensatz verwenden zu können. Diese Einverständnis gaben 3.785 Jugendliche (96%).

Untersuchungsinstrumente

Als Untersuchungsinstrument wurde ein Fragebogen mit 121 Fragen entwickelt, die teilweise der ISAAC II-Studie entnommen wurden. Fragen zu folgenden Themen waren darin enthalten: Demographie, Genetik, Atemwegssymptome und - erkrankungen, Wohnung, Haustierkontakt, Rauchverhalten und Exposition gegenüber Passivrauch, Ausbildung, Berufswahl und Arbeitsplatz, Sport, körperliche Entwicklung und Stressfaktoren.

Studienaufbau

Der Zeitpunkt für die Durchführung der SOLAR-Studie wurde ganz bewusst zu Beginn des Berufslebens der Teilnehmer festgesetzt. Durch dieses Vorgehen konnten während eines Berufslebens alle relevanten beruflichen Expositionen erfasst werden, da auch Ferienjobs sowie Tätigkeiten neben der Schule eingingen. Dadurch konnte der Einfluss auf die Entwicklung von Atemwegsbeschwerden und -erkrankungen sowie Hauterkrankungen analysiert werden. Allerdings ist zu beachten, dass die Fallzahlen in den einzelnen Berufsgruppen noch sehr gering waren, da während des Untersuchungszeitraums der Studie erst ein Drittel der Teilnehmer beruflich tätig war. Eine berufliche Tätigkeit wurde dadurch definiert, dass diese Tätigkeit mindestens 8 Stunden pro Woche und mindestens für einen Monat lang ausgeübt wurde. [RADON et al. 2005]

2.3 2. Follow-up: SOLAR II-Studie

Studienziele und Motivation

Die Fragestellungen, die durch die SOLAR-Studie nicht abschließend geklärt werden konnten, sollten nun durch eine zweite Nachfolgestudie - die SOLAR II-Studie - beantwortet werden. Durchgeführt wurde diese Studie erneut durch das Institut und die Poliklinik für Arbeits-, Sozial- und Umweltmedizin des Klinikums der Universität München in Koopera-

tion mit dem Dr. von Haunerschen Kinderspital des Klinikums der Universität München, der Universitätskinderklinik Dresden, der Universität Ulm sowie der Universtiät Giessen. Dabei verfolgte die SOLAR II-Studie folgende Hauptziele:
Zunächst sollte geklärt werden, wie sich allergische Erkrankungen während der Ausbildung und im weiteren Berufsleben entwickeln.
Darauf aufbauend soll ein Instrumentarium zur Optimierung der individuellen Beratung atopischer Jugendlicher bei der Berufswahl entwickelt werden. Durch dieses Instrumentarium soll zur Senkung der Jugendarbeitslosigkeit beigetragen werden. Durch ein optimiertes Instrumentarium soll verhindert werden, dass Jugendlichen, die zu allergischen Erkrankungen neigen, der Zugang zu vielen Berufen unnötigerweise verschlossen bleibt.
Jugendlichen, die zur Entwicklung von allergischen Erkrankungen tendieren ("Atopiker"), wird derzeit pauschal von Berufen abgeraten, bei denen ein erhöhtes Risiko bekannt ist. Unter Jugendlichen liegt der Atopikeranteil etwa bei 40%. Problematisch ist dabei, dass nur einer von sechs Atopikern tatsächlich eine berufsbezogene Atemwegserkrankung entwickelt und somit fünf Jugendlichen unnötigerweise von dem Beruf abgeraten wird.
Aus diesem Grund soll ein Punktesystem ("Risikoscore") entwickelt werden, das das individuelle Risiko einer Person auf berufsbedingte Atemwegserkrankungen oder Berufsallergien vorhersagt. Dabei soll nicht nur der Atopiestatus eingehen, sondern weitere Faktoren berücksichtigt werden, wie beispielsweise atopische Erkrankungen in der Familie, Stillen, Geschwisterzahl, Passivrauchbelastung, Aktivrauch und berufliche Faktoren. Mit Hilfe dieses Risikoscores kann dem Jugendlichen dann eine bessere Empfehlung, mit größerer Vorhersagesicherheit, für oder gegen einen Beruf gegeben werden. [RADON 2008]
Brechen Jugendliche ihre Ausbildung aufgrund von gesundheitlichen Gründen ab, so sind die Ursachen in einem Drittel der Fälle Probleme der Haut und der Atemwege. Wird ein Beruf aufgrund von Asthma oder allergischen Erkrankungen abgebrochen, so erfolgt dieser Berufsabbruch in etwa 88% der Fälle bereits während der Ausbildung. Für die Betroffenen bedeutet das neben gesundheitlichen Gründen oft Umschulungsmaßnahmen oder sogar Arbeitslosigkeit aufgrund mangelnder Berufsalternativen ohne bekanntes Asthma- oder Allergierisiko.
Aus diesen Gründen soll weiterhin das Instrumentarium zur Früherkennung von Anzeichen allergischer Atemwegs- und Hauterkrankungen bereits während der Ausbildung bzw. den ersten Berufsjahren optimiert werden. Dieses Instrumentarium soll zur Verbesserung des Schutzes der Beschäftigten vor berufsbedingten Erkrankungen und zur Sicherung der Erhaltung der Arbeitsfähigkeit beitragen. Weiterhin soll dieses Instrumentarium die frühzeitige Anwendung erforderlicher Arbeitsschutzmaßnahmen unterstützen. [RADON et al. 2008]
Dabei soll eine klare Unterscheidung mit hoher Prognosesicherheit möglich werden, ob bei Jugendlichen mit Symptomen der Atemwege eine Umschulungsmaßnahme dringend notwendig ist, eine Minderung der Exposition zum langfristigen Schutz der Gesundheit ausreicht oder ausschließlich eine intensive Nachbeobachtung zunächst ausreicht. Durch diese Unterscheidung soll die Notwendigkeit von Umschulungsmaßnahmen reduziert werden. Das hat zum einen eine erhebliche Kostenersparnis zur Folge, zum anderen kann die Jugendarbeitslosigkeit reduziert werden.

Probandenauswahl

In der SOLAR II-Studie, die in den Jahren 2007 bis 2009 stattfand, wurden all diejenigen Probanden erneut kontaktiert, die sich während der SOLAR-Studie zur Teilnahme an einer weiteren Untersuchung bereit erklärt hatten (N=3.054, 78%). Von diesen Personen wurden bis zum Beginn dieser Arbeit (31. März 2009) tatsächlich auch 2.910 Jugendliche erreicht (95%). Das Studienteam setzte sich eine Teilnahmequote von 80% als Ziel. Bis 31. März 2009 konnte dieses Ziel noch nicht vollständig erreicht werden, da bis dahin erst 1.966 Jugendliche teilnahmen. Dies entspricht einer Teilnahmequote von 68%.
Durch die Auswahl des Befragungszeitpunktes wurden die Jugendlichen, die zu diesem Zeitpunkt im Alter zwischen 21 und 23 Jahren waren, zu Beginn bzw. während ihres Berufslebens befragt.

Untersuchungsinstrumente

In Anlehnung an die SOLAR-Studie wurde auch hier ein Fragebogen entwickelt. Dieser enthielt 136 Fragen und ging verstärkt auf die Themen Gesundheit, Wohnung, Rauchen, Arbeitssituation, Sport, körperliche Entwicklung und Belastungssituationen ein. Zusätzlich wurden klinische Untersuchungen durchgeführt (u.a. Hautallergietests, Blutuntersuchungen und Lungenfunktionsmessungen).

Studienaufbau

Da während der SOLAR-Studie nur ein geringer Teil der Probanden bereits berufliche Tätigkeiten ausübte, wurde der Untersuchungszeitraum für die SOLAR II-Studie so gewählt, dass der Zeitraum zu Beginn bzw. während des Berufslebens untersucht wurde. Somit befanden sich nun neben Haupt- und Realschülern auch Abiturienten bereits in der Ausbildung oder gingen diversen (Neben-)Tätigkeiten nach. Im Vergleich zur SOLAR-Studie wurde somit ein größerer Teil des Berufs- und Tätigkeitsspektrums und auch der Bevölkerung abgedeckt. [RADON 2008]
Durch die longitudinale Verknüpfung der drei Studien erhielt man weiterhin die Chance, den Verlauf von allergischen Erkrankungen und Atemwegserkrankungen im Kindesalter über die Pubertät bis zum Eintritt ins Berufsleben zu verfolgen.

2.4 Zusammenfassung

Zusammenfassend sollen nochmals die Anzahl der Teilnehmer und die Teilnahmebereitschaft an den verschiedenen Studien anhand Abbildung 2.2 verdeutlicht werden.

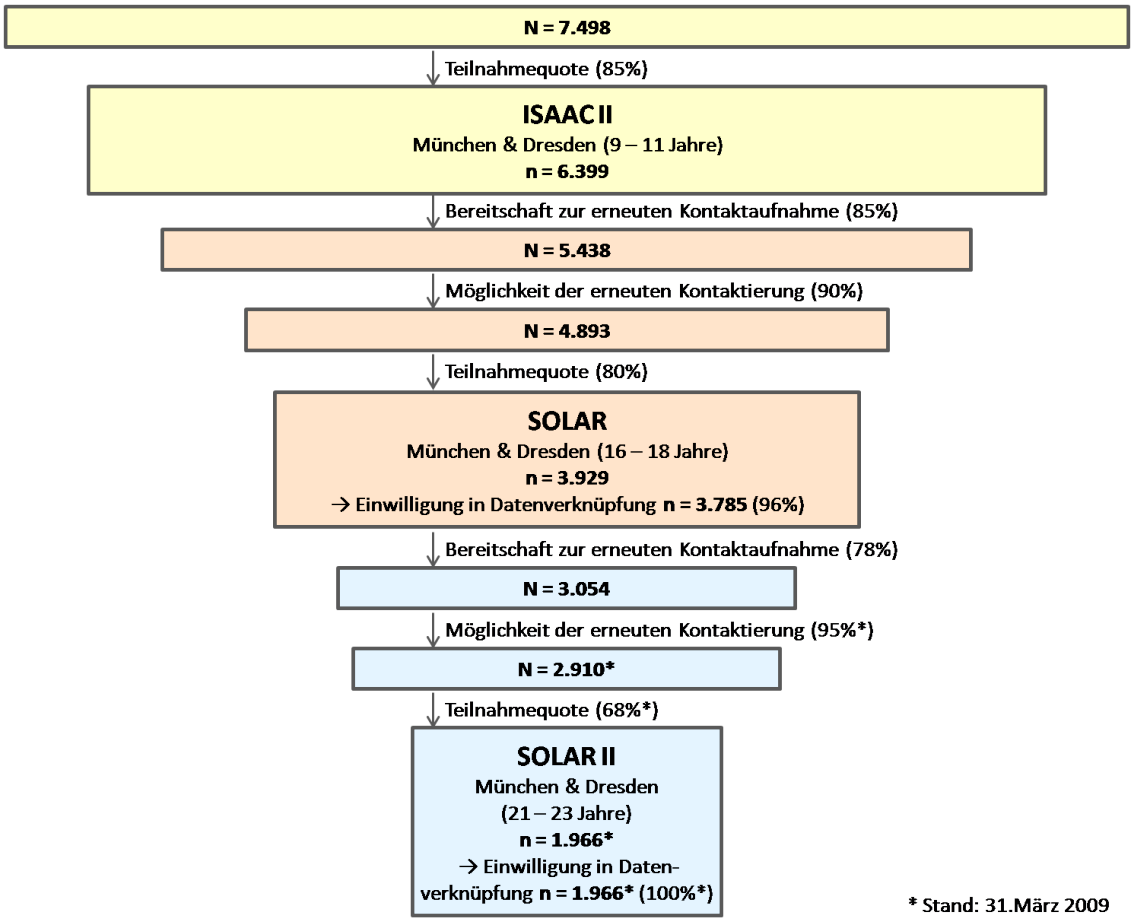

Abbildung 2.2: Übersicht über die Anzahl der Teilnehmer und die Teilnahmebereitschaft im Verlauf der Kohortenstudie

3 Tätigkeitskodierung

In den beiden SOLAR-Studien wurden die Probanden nach Berufen, Berufswünschen, ausgeführten Tätigkeiten, Ferienjobs etc. gefragt. Diese Tätigkeiten wurden zunächst nach dem ISCO-88 Code kodiert und daraufhin in eine Asthma-spezifische Job-Exposure-Matrix überführt, die zur Abschätzung der beruflichen Exposition verwendet wurde. Durch dieses Vorgehen wurde jeder Tätigkeit eine Exposition zugeordnet, die potenziell zu berufsbedingtem Asthma oder einer Allergie führen kann. [RADON et al. 2005]

3.1 Berufssystematik: ISCO-88

Die von den Teilnehmern der Studien in den Fragebögen angegebenen (beruflichen) Tätigkeiten bzw. Berufswünsche wurden nach der Internationalen Standardklassifikation der Berufe (ISCO) der Internationalen Arbeitsorganisation (ILO) in Genf kodiert. Die Vorteile der ISCO-Klassifikation sind die Gliederung nach berufssoziologischen Gesichtspunkten und die internationale Vergleichbarkeit. Es gibt zwei Arten dieser Berufssystematik (ISCO-68 und ISCO-88). In den vorliegenden Studien wurde der neuere und wesentlich verbesserte Schlüssel ISCO-88 zu Grunde gelegt. Der Schlüssel ISCO-88 ist dem älteren Schlüssel im Hinblick auf Systematik, Konsistenz und Anwendung deutlich überlegen. Um die ISCO-Kodierung anwenden zu können, werden Angaben zur Tätigkeit und evtl. zur Berufsbranche benötigt (nicht Titel, Amtsbezeichnung oder Stellung im Beruf). Bei dieser Klassifikation werden Tätigkeiten zu Berufen zusammengefasst. Eng verwandte Berufe werden zu einer Berufsgruppe zusammengefasst, nach den jeweiligen Aufgaben und Pflichten einer Person gruppiert und in eine mit vierstelligen Ziffernfolgen gekennzeichnete Hierarchie eingeordnet. [GEIS 2007]

Die erste Ebene dieser Hierarchie bilden zehn Berufshauptgruppen:

0. Soldaten
1. Angehörige gesetzgebender Körperschaften, leitende Verwaltungsbedienstete und Führungskräfte der Privatwirtschaft
2. Wissenschaftler (=Hochschulabsolventen)
3. Techniker und gleichrangige nichttechnische Berufe (=Fachhochschulabsolventen)
4. Bürokräfte, Kaufmännische Angestellte
5. Dienstleistungsberufe, Verkäufer in Geschäften und auf Märkten
6. Fachkräfte in der Landwirtschaft, Forstwirtschaft und Fischerei
7. Handwerks- und verwandte Berufe

8. Anlagen- und Maschinenbediener sowie Montierer
9. Hilfsarbeitskräfte

Diese Berufshauptgruppen werden jeweils nochmal durch drei weitere Gliederungsebenen genauer spezifiziert:

- Berufsgruppen (sub-major groups)
- Berufsuntergruppen (minor groups)
- Berufsgattungen (unit groups)

Jeder Tätigkeit kann so ein vierstelliger Code zugeordnet werden.
Beispielsweise würde einem Maurer der ISCO-88-Code *7122* zugeordnet werden.

7	*Handwerksberufe und verwandte Berufe*
71	*Mineralgewinnungsberufe und Bauberufe*
712	*Baukonstruktionsberufe und verwandte Berufe*
7122	*Maurer, Bausteinmetzen*

Diese Vercodung erfolgte händisch. Um dabei die Validität zu gewährleisten, wurde die Kodierung von zwei Personen unabhängig voneinander vorgenommen. Stimmte die Kodierung überein, wurde sie übernommen. Gab es Abweichungen zwischen den beiden Vercodungen, so wurde die endgültige Kodierung in einem Expertenschritt festgelegt.

3.2 Job-Exposure-Matrix

Da in den Studien die für Asthma- und Allergieentstehung relevanten beruflichen Expositionen analysiert werden sollten, war es nötig, die Exposition bei einer bestimmten Tätigkeit abzuschätzen. Diese Abschätzung erfolgte mit Hilfe der Asthma-spezifischen Job-Exposure-Matrix, die von Dr. Susan Kennedy (University of British Columbia) entwickelt wurde und die spezifische Expositionen betrachtet, die zu berufsbedingtem Asthma oder einer Allergie führen können. [KENNEDY et al. 2000] Durch diese Matrix können Tätigkeitsangaben genutzt werden, um Probanden einer Risikogruppe im Hinblick auf Asthma und Allergien zuzuordnen. Diese Zuordnung basiert darauf, dass die jeweilige Person einen Job ausübt, in dem sie einem hohen Risiko ausgesetzt ist, berufsbedingtes Asthma zu entwickeln. Die Matrix besteht aus zwei Dimensionen. Zum einen enthält sie die Berufscodes der ISCO-88-Klassifikation (Zeilen), zum anderen enthält sie 22 Expositionsgruppen (Spalten). Aus diesen 22 Gruppen werden vier davon zu einem Block mit niedrigem Asthmarisiko, die restlichen 18 zu einem Block mit hohem Asthmarisiko zusammengefasst. Dieser Block mit hohem Asthmarisiko kann unterteilt werden in hochmolekulare Stoffe (HMW), niedermolekulare Stoffe (LMW), gemischte Stoffe (Mixed) und irritative Spitzenexposition (Irrpeaks).
Die folgende Abbildung 3.1 verdeutlicht in Anlehnung an [RADON et al. 2005] die Einteilung der beruflichen Exposition anhand der Job-Exposure-Matrix grafisch.

Hohes Asthmarisiko				Niedriges Asthmarisiko
HMW Hochmolekulare Stoffe	**LMW Niedermolekulare Stoffe**	**Mixed Gemischte Stoffe**	**Irrpeaks Irritative Spitzenexposition**	**Lowrisk Niedriges Risiko**
- Milbenexposition - Enzymexposition - Latexexposition - Bioaerosole - Tierexposition - Fischexposition - Mehlexposition - Pflanzenexposition - Pharmaka-exposition	- Reaktive Stoffe-Exposition - Isocyanide-Exposition - Reinigungsmittel-exposition - Metallstaub-exposition - Holzstaub-exposition	- Flüssigmetall-exposition - Textilienexposition - Landwirtschafts-exposition	- Irritative Spitzenexposition	- Abgasexposition - Passivrauch-exposition - Irritanzien-exposition - Geringe Antigene-Exposition

Abbildung 3.1: Einteilung der beruflichen Exposition anhand der Job-Exposure-Matrix

In der Matrix enthält jede Zelle eine Klassifikation für das Exposure-Risiko (Ja/Nein). Eine berufliche Tätigkeit wird als exponiert klassifiziert, wenn es bei dieser Tätigkeit sehr wahrscheinlich ist, dass man einer Asthma-relevanten Exposition ausgesetzt ist. Die Tätigkeiten, bei denen man weder einem hohen, noch einem niedrigen Asthmarisiko ausgesetzt ist, werden als nicht exponiert kodiert. Somit wird in dieser Matrix ausschließlich die Unterscheidung getroffen, ob eine Exposition bei der jeweiligen Tätigkeit vorliegt oder nicht. Die Intensität der Exposition in einem Job wird durch diese Matrix nicht beschrieben.
Durch diese Job-Exposure-Matrix kann nun also mit Hilfe des ISCO-Codes jeder beruflichen Tätigkeit eine Exposition zugeordnet werden. Es kann allerdings nicht beurteilt werden, ob ein Proband, der diese Tätigkeit ausübte, auch tatsächlich dieser Exposition ausgesetzt war, da jeder Arbeitsplatz individuell ist. Je konkreter die Angaben zu den Tätigkeiten dabei sind, umso besser kann die tatsächliche Exposition mit Hilfe der Job-Exposure-Matrix abgeschätzt werden.
Beispiel: Eine Person hat als Tätigkeit Bäcker angegeben. Diese Person kann tatsächlich an der Teigherstellung oder dem Zubereiten von Backwaren beteiligt sein und dadurch gegenüber Mehlstaub exponiert sein. Oder sie ist ausschließlich im Verkauf oder der Zulieferung von Backwaren beteiligt und somit wahrscheinlich nur geringfügig oder gar nicht gegenüber Mehlstaub exponiert.
Weiterhin gilt es folgenden Aspekt zu beachten:
"Da die ISCO-Codierung für wirtschaftliche und nicht gesundheitliche Zwecke entworfen wurde, ist bei einem Teil der Expositionen in der JEM eine Experten Re-Evaluation nötig." [RADON et al. 2005] Für diesen Zweck ist in der Matrix eine zusätzliche Spalte enthalten, die bei bestimmten Codes angibt, ob die jeweiligen Angaben zu den Expositionen nochmals überprüft werden sollten. In einem Expertenschritt wurde dann von Frau Prof. Dr. Radon die Richtigkeit des ISCO-Codes und der daraus resultierenden Expositionsangaben überprüft und gegebenenfalls korrigiert.

4 Theorie zur Problematik und Behandlung fehlender Daten

Bei der Analyse von Daten kann es häufig vorkommen, dass fehlende Werte in den Daten vorhanden sind. Bei den Studien, die für diese Arbeit analysiert wurden, traten an diversen Stellen, d.h. sowohl bei den Confoundervariablen als auch bei den Tätigkeitsangaben, fehlende Werte auf . Aus diesem Grund ist es nötig, einige Grundlagen für die Behandlung von fehlenden Daten zu schaffen.
Zunächst interessieren häufig die Gründe für das Fehlen der Werte. Einige Beispiele sollen die Bandbreite der unterschiedlichen Gründe verdeutlichen.

- *Bei einer Meinungsumfrage mittels Telefon oder Fragebogen antworten Personen nicht auf alle gestellten Fragen. Ursache für das Nichtbeantworten der Frage kann zum einen sein, dass die befragte Person tatsächlich keine Meinung bezüglich des konkreten Themas äußern kann oder eine Frage schlichtweg übersehen wurde. Zum anderen kann sich die Person aber auch weigern eine bestimmte Angabe zu machen, wie z.B. zu ihrem Einkommen. Teilweise werden aber auch falsche Antworten gegeben - sowohl versehentlich als auch absichtlich - die im Nachhinein als unsinnige Antworten identifiziert werden und nicht in die Analyse mit einbezogen werden können.*

- *In Laboruntersuchungen können Werte aufgrund von technischen Problemen fehlen, da einzelne Werte beispielsweise durch technische Geräte nicht aufgezeichnet wurden.*

- *Bei einem longitudinalen Studiendesign können Teilnehmer einer früheren Studienphase nicht mehr erneut kontaktiert werden, weil die Adresse nicht mehr gültig ist oder weil sie sich ausdrücklich gegen eine erneute Kontaktierung ausgesprochen haben. Die kompletten Angaben dieser Personen fehlen dann in allen folgenden Phasen dieser Studie.*

In den ersten beiden Beispielen handelt es sich jeweils um item-nonresponse, da ausschließlich Werte bestimmter Variablen fehlen, Angaben zu anderen Variablen jedoch vorliegen. Im letzten Beispiel spricht man von unit-nonresponse, d.h. es liegt überhaupt keine Information über diesen Fall bei einer Studie bzw. Studienphase vor. Die im Folgenden erläuterten Methoden beziehen sich ausschließlich auf den Fall item-nonresponse.
Unabhängig vom konkreten Grund des Fehlens muss eine statistische Auswertung diesen fehlenden Daten mit geeigneten Methoden Rechnung tragen, um unverzerrte Schlüsse ziehen zu können. Da das Verwerfen von fehlenden Daten im harmlosesten Fall einen erheblichen Informations- und Aussageverlust und im schlimmsten Fall starke Verzerrungen zur Folge hat, stellt die Statistik eine ganze Palette von Methoden zur Behandlung von fehlenden Daten bereit. Im Folgenden werden Grundbegriffe und Methoden zur Analyse von Daten mit fehlenden Werten vorgestellt, die im Wesentlichen auf [LITTLE und RUBIN 2002], [TOUTENBURG 2003] und [TOUTENBURG und HEUMANN 2006] zurückgehen.

4.1 Fehlendmechanismen

Zunächst muss man sich bei fehlenden Werten immer die Frage stellen, ob diese Werte rein zufällig fehlen oder ob das Fehlen dieser Werte einer bestimmten Systematik unterliegt, die beispielsweise von anderen Variablen abhängig ist. Aufgrund dieser Fragestellung unterscheidet man bei fehlenden Daten zwischen verschiedenen Mechanismen, die den fehlenden Daten zugrunde liegen. Diese Fehlendmechanismen sind entscheidend, da die Eigenschaften der Methoden für fehlende Daten stark von diesen Mechanismen abhängen. Um die verschiedenen Mechanismen zu erläutern, geht man im Folgenden von einem partiell fehlendem Response Y (*z.B. Einkommen*) und einer vollständig beobachteten Kovariable X (*z.B. Alter*) aus:

- **MCAR: M**issing **c**ompletely **a**t **r**andom
  Die Wahrscheinlichkeit, dass der Response Y_i beobachtet wird, ist weder von der Kovariable X noch vom Response Y abhängig. Die beobachteten Daten sind dann eine echte Zufallsstichprobe aller Daten. Eine Analyse dieser zufälligen Substichprobe würde aufgrund des reduzierten Stichprobenumfangs lediglich zu einem Effizienzverlust führen. Verzerrungen würden somit hier nicht auftreten.
  Beispiel: Die Wahrscheinlichkeit, dass das Einkommen fehlt, ist für alle Individuen gleich groß, unabhängig von deren Alter und deren Einkommen. Das beobachtete Einkommen stellt somit eine echte Zufallsstichprobe aller betrachteten Individuen dar.

- **MAR: M**issing **a**t **r**andom
  Die Wahrscheinlichkeit, dass der Response Y_i beobachtet wird, ist von der Kovariable X, aber nicht vom Response Y abhängig. Die beobachteten Daten sind dann in jeder bezüglich X gebildeten Klasse eine Zufallsstichprobe. Auch hier treten bei der Analyse innerhalb der Klassen keine Verzerrungen auf, einziger negativer Effekt ist der Effizienzverlust.
  Beispiel: Die Wahrscheinlichkeit, dass das Einkommen fehlt, ist je nach Alter des Teilnehmers unterschiedlich, hängt somit also vom Alter ab. Beispielsweise könnte man sich vorstellen, dass jüngere Personen eher bereit sind, ihr Einkommen preiszugeben, als ältere Personen. Diese Wahrscheinlichkeit hängt allerdings nicht vom Einkommen des Probanden selbst ab (bei gleichen Altersstufen). Somit kann man sagen, dass die Angaben innerhalb der Altersstufen zufällig fehlen. Die beobachteten Einkommensdaten können zwar in jeder Altersstufe, aber nicht insgesamt, als Zufallsstichprobe betrachtet werden.

- **MNAR: M**issing **n**ot **a**t **r**andom
  a) Die Wahrscheinlichkeit, dass der Response Y_i beobachtet wird, ist nicht von der Kovariable X abhängig, hängt aber vom Response Y ab.
  b) Die Wahrscheinlichkeit, dass der Response Y_i beobachtet wird, ist sowohl von der Kovariable X , als auch vom Response Y abhängig.
  Die Daten fehlen somit in beiden Fällen nicht zufällig. Analysen, die auf solchen Substichproben basieren, sind in der Regel verzerrt.
  Beispiel: Die Wahrscheinlichkeit, dass das Einkommen fehlt, kann vom Alter abhängen (b) oder nicht (a). Entscheidend ist hier allerdings, dass die Wahrscheinlichkeit,

dass das Einkommen fehlt, vom Einkommen selbst abhängt. Beispielsweise können sich Personen mit sehr hohem Einkommen häufiger weigern, dieses preiszugeben, als Personen anderer Einkommensschichten. Würde man nun das Einkommen auf Basis der erhaltenen Angaben schätzen, so würde man das mittlere Einkommen unterschätzen.

4.2 Analyse von Daten mit fehlenden Werten

Die folgende Übersicht soll einen kurzen Überblick über die Analysemethoden geben. In den folgenden Abschnitten werden die einzelnen Verfahren ausführlich erläutert.
Liegen Daten mit fehlenden Werten vor, so gibt es grundsätzlich zwei Vorgehensweisen der Datenanalyse. Zum einen können alle Fälle mit fehlenden Werten ausgeschlossen werden. Zu dieser Methode gehören folgende Verfahren:

- **Complete Case Analysis**
  In diese Analyse werden ausschließlich Fälle einbezogen, bei denen zu *allen erhobenen Variablen* Werte vorliegen.
- **Available Case Analysis**
  In dieser Analyse werden nur diejenigen Fälle betrachtet, die bei den *jeweils betrachteten Variablen* vollständig sind.

Insbesondere die Complete Case Analysis macht keinen Gebrauch von Fällen mit fehlenden Werten. Deshalb werden im weiteren Verlauf dieses Abschnitts zusätzlich Methoden vorgestellt, die fehlende Werte von Variablen imputieren (d.h. durch neue Werte ersetzen). Diese Methoden können folgendermaßen untergliedert werden:

- **Single Imputation**
  Für jeden fehlenden Wert einer Variable wird *ein einziger Wert imputiert.*
  Gängige Methoden sind:
  - Mean Imputation
  - Imputation durch Ziehen aus der empirischen Verteilung
  - Hot deck Imputation
  - Cold deck Imputation
  - Regression Imputation
- **Multiple Imputation**
  Pro fehlendem Wert einer Variable werden *mehrere Werte imputiert.* Dadurch wird der Unsicherheit bei der Imputation Rechnung getragen.

Um im folgenden Abschnitt die Problematik fehlender Daten auf mehr als zwei Variablen zu verallgemeinern, werden die vorliegenden Daten nun mit Hilfe einer Datenmatrix etwas formaler dargestellt. [TOUTENBURG 2003] In der Datenmatrix D stellen die Spalten standardmäßig die Variablen dar und die Zeilen entsprechen den einzelnen Fällen. Somit werden also p Variablen und n Fälle betrachtet. Einzelne fehlende Beobachtungen werden

durch das Symbol * dargestellt. Alle mit . dargestellten Zellen liegen vollständig vor.

$$D = \begin{pmatrix} d_{11} & d_{12} & d_{13} & d_{14} & . & . & . & d_{1p} \\ d_{21} & d_{22} & * & d_{24} & . & . & . & d_{2p} \\ d_{31} & d_{32} & d_{33} & d_{34} & . & . & . & d_{3p} \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ d_{(n-1)1} & * & * & * & . & . & . & d_{(n-1)p} \\ d_{n1} & d_{n2} & d_{n3} & d_{n4} & . & . & . & d_{np} \end{pmatrix}$$

4.2.1 Ausschluss von Fällen mit fehlenden Werten

Complete Case Analysis - Nutzung der kompletten Fälle

Bei der Complete Case Analysis werden nur die Fälle in die Analyse einbezogen, bei denen zu allen Variablen die Ausprägungen vorliegen. Somit werden aus der Datenmatrix alle unvollständig beobachteten Zeilen gelöscht. In der zuvor eingeführten Datenmatrix D würden die zweite und die vorletzte Zeile gelöscht werden. In die Analyse geht die reduzierte Datenmatrix D^{CC} ein.

$$D^{CC} = \begin{pmatrix} d_{11} & d_{12} & d_{13} & d_{14} & . & . & . & d_{1p} \\ d_{31} & d_{32} & d_{33} & d_{34} & . & . & . & d_{3p} \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ d_{n1} & d_{n2} & d_{n3} & d_{n4} & . & . & . & d_{np} \end{pmatrix}$$

Vorteile dieser Methode ist zunächst ihre Einfachheit, da zur Analyse Standardmethoden (für vollständig beobachtete Daten) ohne weitere Modifikationen angewandt werden können. Weiterhin sind berechnete Statistiken bei diesen Analysen vergleichbar, da alle auf einer einheitlichen Stichprobengröße basieren.
Eine Voraussetzung für die Anwendung der Complete Case Analysis ist, dass der Anteil der Zeilen mit fehlenden Werten hinreichend gering ist. Somit kann diese Methode zufriedenstellende Ergebnisse liefern, wenn nur wenige (zufällig) fehlende Daten vorliegen. Fehlen die Daten allerdings nicht zufällig, unterliegen sie also dem Fehlendmechanismus MNAR, so kann es zu starken Verzerrungen kommen.
Unabhängig davon, ob die Daten zufällig fehlen oder nicht, ist als Nachteil dieser Vorgehensweise der Effizienz- und Informationsverlust zu nennen, der aus dem Löschen von Probandendaten resultiert. Eventuell bleiben in bestimmten Situationen auch nur sehr wenige Fälle übrig. Sobald die Ausprägung einer einzigen Variable fehlt, also sowohl bei einer der Ko- als auch bei der Zielvariable, so wird der komplette Fall aus der Analyse ausgeschlossen. Es ist einleuchtend, dass vor allem bei der Analyse eines großen Datensatzes mit vielen Variablen und vielen fehlenden Werten eine Vielzahl von Fällen und damit eine große Menge an Informationen verworfen wird. Zusätzlich wird man beim Löschen von Probanden bei epidemiologischen und klinischen Studien mit einem ethischen Problem konfroniert. Die Probanden dieser Studien haben ihre Zeit “geopfert”, um an der

Studie teilzunehmen und sich eventuell den Risiken eines neuen Medikaments unterzogen. Folglich ist es dann nicht gerechtfertigt, diese Daten nicht einzubeziehen.
Zieht man desweiteren die Tatsache in Betracht, dass man Aussagen über die gesamte Zielpopulation machen möchte, und nicht nur über die Personen, die Antworten auf alle Fragen geliefert haben, so erscheint die Complete Case Analysis in dieser Hinsicht meist als eher ungeeignet.
Wird eine Regressionsanalyse auf Basis eines Complete-Case-Datensatzes berechnet, so kommt es weiterhin zu Verzerrungen der Schätzer, wenn die Wahrscheinlichkeit für das Fehlen der Kovariablenwerte von der Zielvariable selbst abhängt. Erwartungstreue Schätzungen erhält man bei der Regression nur, wenn die Wahrscheinlichkeit für das Fehlen nicht vom Response abhängt. In diesen Fällen darf die Wahrscheinlichkeit nur von den Kovariablenwerten selbst und den fehlenden Werten abhängen. [TOUTENBURG 2003]

Available Case Analysis - Verwendung aller verfügbaren Daten

Bei der Available Case Analysis werden jeweils alle Fälle in der Analyse benutzt, die bezüglich der jeweils betrachteten Variable vollständig sind. Somit gehen pro Variable jeweils alle gegebenen Antworten mit in die Analyse ein. Ein Informationsverlust liegt somit - zumindest im univariaten Fall - nicht vor.
Führt man allerdings multivariate Analysen durch, so müssen alle darin betrachteten Variablen vollständig sein. Je mehr Variablen (mit fehlenden Werten) in solch eine Analyse einbezogen werden, um so mehr Fälle und somit auch Information geht dabei verloren.
Konzentriert man sich zur Veranschaulichung auf den univariaten Fall und betrachtet beispielsweise die Variable d_{i1}, so würde folgende Datenmatrix $D^{d_{i1}}$ (hier im univariaten Fall: Datenvektor) in die Available Case Analysis eingehen.

$$D^{d_{i1}} = \begin{pmatrix} d_{11} & d_{21} & d_{31} & . & . & d_{(n-1)1} & d_{n1} \end{pmatrix}^T$$

Bei Betrachtung der Variable d_{i3} würde die Datenmatrix $D^{d_{i3}}$ (hier im univariaten Fall: Datenvektor) wie folgt aussehen.

$$D^{d_{i3}} = \begin{pmatrix} d_{13} & d_{33} & . & . & d_{n3} \end{pmatrix}^T$$

Die Datenmatrix $D^{d_{i1}}$ für die Variable d_{i1} enthält Werte zu allen n Fällen. Die Datenmatrix $D^{d_{i3}}$ für die Variable d_{i3} enthält Werte zu (n-2) Fälle, da der zweite und der vorletzte Fall in dieser Variable fehlende Werte aufweisen. Somit liegen bei der Analyse der Variablen unterschiedliche Stichprobenumfänge zugrunde. Dieses Problem darf bei der Available Case Analysis nicht vernachlässigt werden, da die Vergleichbarkeit zwischen verschiedenen Variablen nicht ohne Weiteres gegeben ist.

4.2.2 Imputationsmethoden

Im Hinblick auf die soeben angesprochenen Nachteile der Complete und der Available Case Analysis und der Tatsache, dass diese Methoden keinen Gebrauch von Fällen mit fehlenden Werten machen, werden im folgenden Abschnitt Methoden vorgestellt, um auch Fälle mit fehlenden Werten in die Analyse einbeziehen zu können.
Generell wird bei Imputationsmethoden die lückenhafte Datenmatrix mit Hilfe verschiedener Verfahren aufgefüllt und die daraus resultierenden vollständigen Daten werden mit Hilfe von Standardmethoden analysiert.
Allerdings muss bei Imputationsmethoden stets mit einer Abweichung zwischen dem imputierten Wert und dem (unbekannten, fehlenden) Originalwert gerechnet werden. [TOUTENBURG 2003] Daher ist es wichtig, diese Unsicherheit bei der Ersetzung fehlender Werte adäquat zu berücksichtigen.

Single Imputation

Mean Imputation - Mittelwertsimputation

Für die fehlenden Daten einer Variable wird das arithmetische Mittel von denjenigen Individuen eingesetzt, die bei dieser Variable Werte vorliegen haben. Eine sinnvolle Möglichkeit stellt in diesem Zusammenhang auch das Vorgehen dar, diese Mittelwerte abhängig von bestimmten Klassen oder Gruppen zu bilden und einzusetzen, beispielsweise je Geschlecht oder je Altersgruppe. Unter welchen Umständen diese Methode zu unverzerrten Ergebnissen führt, ist allerdings schwer abzuschätzen.
Problematisch ist weiterhin, dass bei der Mittelwertsimputation, die empirische Verteilung der Daten verzerrt wird. Somit können beispielsweise Varianzschätzer durch Standardmethoden nicht mehr konsistent geschätzt werden. [LITTLE und RUBIN 2002]
Beispiel: Treten bei der Variable Einkommen fehlende Werte auf und führt man dafür eine Mittelwertsimputation durch, so kann man davon ausgehen, dass in diesem Fall Armut und Reichtum unterschätzt werden, da fehlende Daten wohl eher in den unteren und den oberen Einkommensschichten zu vermuten sind.
Diese Methode ist für stetige Variablen geeignet. Liegen kategoriale Daten vor, so kann statt des arithmetischen Mittels auch der Modus zur Imputation verwendet werden. Bei ordinalen Daten verwendet man den Median.

Imputation durch Ziehen aus der empirischen Verteilung

Bei dieser Imputationsmethode werden die imputierten Werte aus der empirischen Verteilung der beobachteten Daten jeder einzelnen Variable gezogen. Durch dieses Verfahren bleiben die Randverteilungen der einzelnen Variablen erhalten, d.h. die Randverteilung der beobachteten Daten einer Variable entspricht der Randverteilung der durch Imputation vervollständigten Daten dieser Variable. Da die Imputation jeder Variablen einzeln, also unabhängig von anderen Variablen, durchgeführt wird, bleiben mögliche Abhängigkeitsstrukturen zwischen den Variablen bei dieser Methode unberücksichtigt. Aus diesem Grund besteht bei dieser Imputationsmethode die Gefahr, die Korrelationsstruktur in den

Daten zu verschleiern oder zu zerstören.

Hot deck Imputation

Bei der Hot deck Imputation werden die bei einem Individuum fehlenden Werte einer Variable durch die beobachteten Werte von "ähnlichen" Individuen (der gleichen Erhebung) ersetzt. Diese Methode ist in der Praxis sehr gebräuchlich. Teilweise werden dabei sehr aufwändige und ausgeklügelte Schemata angewandt, um solche "ähnlichen" Individuen für die Imputation auszuwählen.
Beispiel: Fehlt bei einer Person A die Angabe zu deren Einkommen, so wird eine "ähnliche" Person B (mit vorhandener Angabe zum Einkommen) auf Basis verschiedener Variablen wie beispielsweise Alter, Geschlecht, Familienstand, Beruf, Kinder ausgewählt. Das Einkommen dieser "ähnlichen" Person wird dann verwendet, um den fehlenden Einkommenswert von Person A zu imputieren.[1]
Allerdings müssen diese Schemata, durch die die Ähnlichkeit von Individuen festgelegt werden, ebenfalls auf Plausibilität geprüft werden. Weiterhin ist es nötig, dass die Werte der Variablen, die zum Vergleich der Individuen herangezogen werden, vollständig vorliegen.
Vorteil der Hot deck Imputation ist die Tatsache, dass die imputierten Werte die empirische Verteilung der Daten nicht verzerren, wie dies bei der Mittelwertsimputation der Fall ist. Allerdings sind die Schätzer bei dieser Imputationsmethode meist nur unter der im Allgemeinen unrealistischen Annahme MCAR unverzerrt. [LITTLE und RUBIN 2002]

Cold deck Imputation

Bei der Cold deck Imputation wird ein fehlender Wert einer Variable durch einen konstanten Wert ersetzt, der aus einer externen Quelle stammt, beispielsweise einem Erfahrungswert aus einer früheren Studie oder einer Konstanten aus der Population (*z.B. Durchschnittsalter der männlichen Bevölkerung*). Dies mag zunächst sehr einfach klingen, allerdings wird man in sehr wenigen Fällen solch einen konstanten Wert zur Verfügung haben. Weiterhin hängt die Qualität dieser Imputationsmethode stark von der Wahl der externen Quelle ab, die zur Imputation verwendet wird.

Regression Imputation

Bei der Regression Imputation wird der entsprechende fehlende Wert geschätzt bzw. vorhergesagt. Konkret geht man dabei folgendermaßen vor: Zunächst wird eine Regression auf Basis der vollständigen Fälle durchgeführt. Die daraus resultierenden Regressionsparameter werden im nächsten Schritt auf die unvollständigen Fälle angewandt, für die dann ein Wert prognostiziert werden kann. Dieser vorhergesagte Wert kann dann für den fehlenden Wert eingesetzt werden. Möchte man zusätzlich der Unsicherheit des vorhergesagten Wertes Rechnung tragen, so kann ein Residuum hinzugefügt werden.

[1] Ein ähnliches - allerdings deutlich komplexeres - Verfahren wendet das U.S. Census Bureau in der "Current Population Study" an, um fehlende Einkommensangaben durch Hot deck Imputation zu vervollständigen. [LITTLE und RUBIN 2002]

Beispiel: Der Einfluss des Geschlechts (X) auf das Einkommen (Y) soll untersucht werden. Es interessiert nun also primär die Regressionsbeziehung von Einkommen auf Geschlecht ($y_i = \alpha + \beta\ x_i$). Bei der Variable Geschlecht fehlen allerdings einige Werte. Zunächst berechnet man also eine Hilfsregression von Geschlecht auf Einkommen ($x_i = \gamma + \delta\ y_i$). Diese dient nun zur Ersetzung der fehlenden Werte bei der Variable Geschlecht. Im einfachsten Fall kann man für den fehlenden Wert die Prognose $\widehat{x_i} = \widehat{\gamma} + \widehat{\delta}\ y_i$ einsetzen. Vorteilhaft ist bei dieser Methode, dass die Struktur innerhalb der Variablen ausgenutzt und somit die Korrelationsstruktur erhalten bleibt. Die Güte und Validität der Regression Imputation wird allerdings durch die Ursache des Fehlens (zufälliges Fehlen vs. nichtzufälliges Fehlen) beeinflusst. [TOUTENBURG und HEUMANN 2006]

Probleme der Single Imputation

Hauptproblem bei der Single Imputation ist die Tatsache, dass diese Methoden die Unsicherheit bei der Imputation nicht berücksichtigen (einzige Ausnahme: Regression Imputation mit zusätzlich hinzugefügtem Residuum). Werden Standard-Varianzformeln auf diese vervollständigten Daten angewandt, so unterschätzen diese Formeln systematisch die Varianz der Schätzer. Beispielsweise erhöht sich bei der Mittelwertsimputation zwar der Stichprobenumfang (im Vergleich zur Complete Case Analysis), allerdings nicht die Varianz.
Aus diesem Grund werden Standardfehler systematisch unterschätzt, p-Werte von Tests fallen zu klein und Konfidenzintervalle zu schmall aus. [LITTLE und RUBIN 2002]

Multiple Imputation

Die Idee der multiplen Imputation wurde in den siebziger Jahren von Donald Rubin entwickelt. Dabei wird jeder fehlende Wert durch mehrere (m > 1) plausible Werte ersetzt, um dann m vervollständigte Datensätze zu erhalten. Um solche plausible Imputationswerte für fehlende Werte generieren zu können, muss ein Verteilungsmodell für die kompletten Daten (vollständige und fehlende Daten) zu Grunde gelegt werden, beispielsweise eine multivariate Normalverteilung. Die Imputation erfolgt dann nach folgendem Schema: Ersetzt man den fehlenden Wert durch den ersten plausiblen Wert, so erhält man den ersten vervollständigten Datensatz. Ersetzt man den fehlenden Wert durch den zweiten plausiblen Wert, so erhält man den zweiten vervollständigten Datensatz, und so weiter. Diese m vervollständigten Datensätze enthalten somit die gleichen beobachteten Werte und unterschiedliche imputierte Werte an den ursprünglich fehlenden Stellen. Dann werden Standardmethoden angewandt, um jeden einzelnen der m Datensätze zu analysieren. Die Schätzungen aus den verschiedenen Analysen (Schätzwerte, Varianzen, Standardfehler, etc.) können dann zu einer Schätzung kombiniert werden. Diese kombinierte Schätzung berücksichtigt auch die Unsicherheit, die bei der Vorhersage der fehlenden Daten aus den beobachteten Daten besteht. Das Schema 4.1 verdeutlicht das Vorgehen bei der multiplen Imputation, bei der beispielhaft drei imputierte Datensätze (m = 3) dargestellt sind.

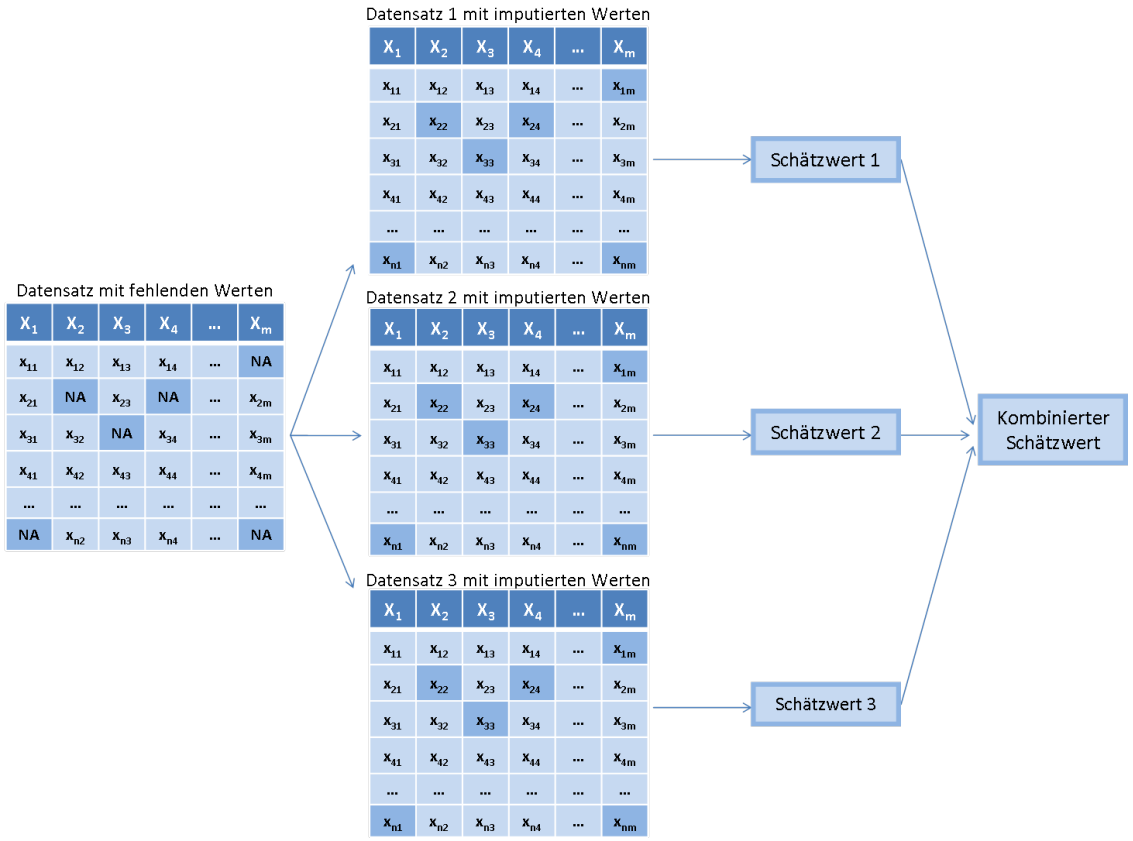

Abbildung 4.1: Vorgehen bei der multiplen Imputation

Zur Kombination der verschiedenen Schätzungen aus den m vervollständigten Datensätzen gibt es klare Regeln. [LITTLE und RUBIN 2002, SCHAFER und OLSEN 2007] Angenommen Q sei der interessierende Parameter und V dessen zugehörige Varianz. Beispielsweise kann Q ein Regressionskoeffizient sein und V die zugehörige Varianz. Somit erhält man aus der Analyse der m vervollständigten Datensätze die Schätzer $\hat{Q}_1$, $\hat{Q}_2$, ..., $\hat{Q}_m$ und die entsprechenden geschätzten Varianzen $\hat{V}_1$, $\hat{V}_2$, ..., $\hat{V}_m$, die alle gleich plausibel sind. Der kombinierte Schätzwert $\hat{Q}$ kann berechnet werden durch

$$\hat{Q} = \frac{1}{m} \sum_{i=1}^{m} \hat{Q}_i. \tag{4.1}$$

Die Varianz des Schätzers besteht aus zwei Komponenten: aus der Varianz innerhalb jedes imputierten Datensatzes (within-imputation variance W) und der Varianz zwischen den m imputierten Datensätzen (between-imputation variance B). Die Varianz innerhalb jedes Datensatzes ist das arithmetische Mittel der geschätzten Varianzen

$$W = \frac{1}{m} \sum_{i=1}^{m} \hat{V}_i. \tag{4.2}$$

Die Varianz zwischen den m Datensätzen entspricht der Stichprobenvarianz der Schätzer.

$$B = \frac{1}{m-1}\sum_{i=1}^{m}(\hat{Q}_i - \hat{Q})^2 \tag{4.3}$$

Bildet man die Summe aus beiden Komponenten (incl. eines Korrekturterms, der die Verzerrung aufgrund von m $< \infty$ korrigiert), so erhält man die geschätzte Gesamtvarianz des Schätzers

$$\hat{V} = W + (1 + \frac{1}{m})B. \tag{4.4}$$

Dadurch, dass die geschätzte Gesamtvarianz bei der multiplen Imputation aus zwei Komponenten besteht und somit auch die Vorhersageunsicherheit bei der Imputation berücksichtigt wird, existiert das Problem der Varianzunterschätzung im Gegensatz zur Single Imputation hier nicht mehr.
Der einzige Nachteil, den die multiple Imputation aufweist, ist der größere Aufwand um die Imputation und die Analyse durchzuführen. In Zeiten von leistungsstarken Rechnern fällt dies kaum noch ins Gewicht.
Zu Beginn dieses Abschnitts wurde angesprochen, dass vor der Imputation ein Verteilungsmodell für die kompletten Daten festgelegt werden muss. Was passiert nun bei Abweichungen von diesem Verteilungsmodell? Viele Studien haben gezeigt, dass multiple Imputation nicht sensibel auf solche Abweichungen reagiert. Beispielsweise ist es tragbar, binäre oder kategoriale Variablen unter der Normalverteilungsannahme zu imputieren und dann die stetigen imputierten Werte zur nächstliegenden Kategorie auf bzw. abzurunden. [SCHAFER und OLSEN 2007]
Eine weitere formale Annahme ist der Fehlendmechanismus MAR. Diese Annahme ermöglicht es, unverzerrte Schätzwerte für die fehlenden Werte auf Basis der beobachteten Werte zu erhalten. Problematisch ist bei dieser Annahme allerdings, dass die MAR-Hypothese nicht auf Basis der vorliegenden Daten getestet werden kann. Man kann folglich nicht mit Sicherheit festlegen, ob MAR vorliegt oder nicht. Allerdings schneiden Methoden, die MAR vorraussetzen, laut [SCHAFER und OLSEN 2007] oftmals besser ab, als sogenannte “Ad-hoc-Prozeduren”, wie beispielsweise Mittelwertsimputation.

5 Datenmanagement

5.1 Datengrundlage

Für diese Arbeit wurden ausschließlich die Datensätze derjenigen Teilnehmer verwendet, die an allen drei Studien (ISAAC II, SOLAR, SOLAR II) teilnahmen und sich mit der longitudinalen Verknüpfung der Daten einverstanden erklärten. Weiterhin mussten zum Zeitpunkt des Beginns dieser Arbeit die Angaben zu den beruflichen Tätigkeiten (aus SOLAR II) bereits gemäß der ISCO-Berufssystematik vollständig kodiert worden sein. Bei den Personen, deren berufliche Tätigkeiten noch nicht vollständig kodiert waren, kann davon ausgegangen werden, dass diese zufällig fehlen und durch das Ausschließen dieser Probanden keine Verzerrung zu erwarten ist.
Fehlten in einer der drei Studien im Fragebogen die zentralen medizinischen Angaben zum Thema Neurodermitis, Allergische Rhinitis oder Asthma, so wurde dieser Fall ebenfalls aus der Analyse ausgeschlossen. Diese Fälle wurden gelöscht, da man sich bei den medizinischen Daten für ein konservatives Vorgehen entschied, und somit nur "sichere" Krankheitsfälle in der Analyse behielt. Eine Imputation der entsprechenden medizinischen Variablen war folglich nicht angemessen.
Zunächst mussten somit aus dem gelieferten Datensatz die Fälle ermittelt werden, auf die obige Bedingungen zutreffen und die anderen Fälle gelöscht werden. Insgesamt konnten von den ursprünglich 1.966 Datensätzen für die in dieser Arbeit durchgeführten Analyse 1.187 Datensätze verwendet werden. Die Grafik 5.1 gibt eine Übersicht über die Datengrundlage.

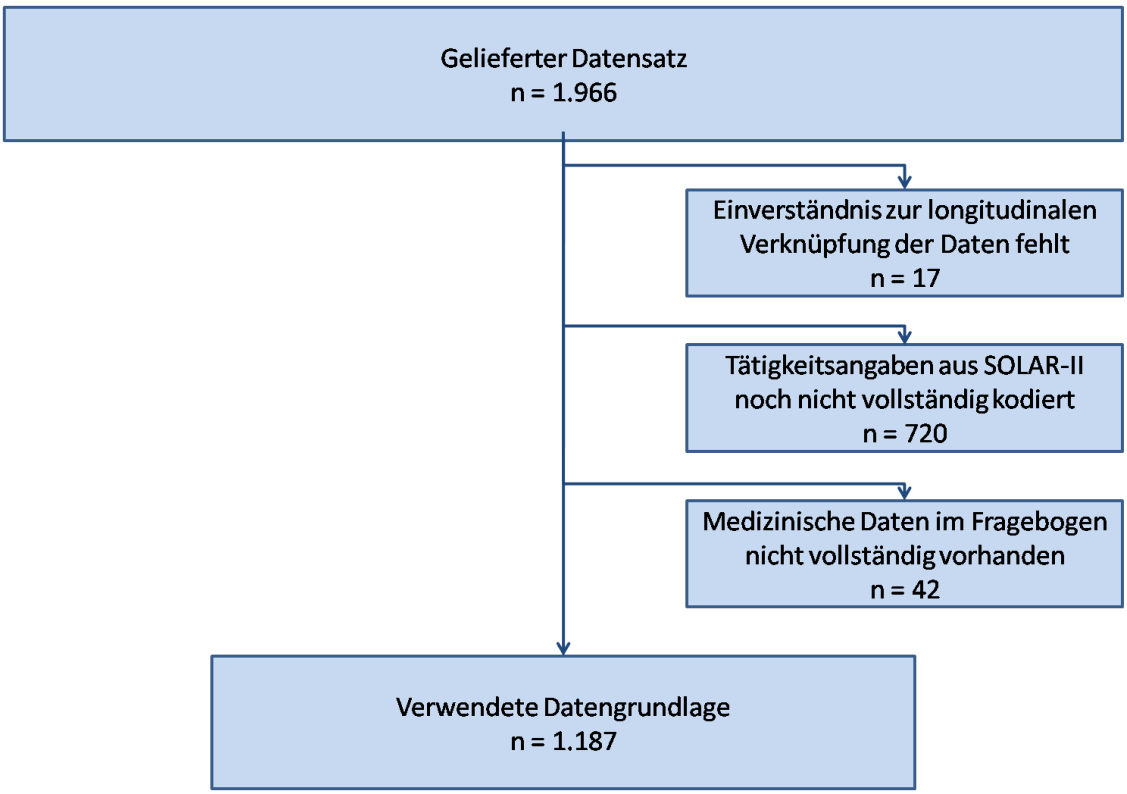

Abbildung 5.1: Datengrundlage der durchgeführten Analysen

5.2 Datenbereinigung

5.2.1 Bereinigung der Confounder- und der Zielvariablen

Nachdem der Datensatz nun ausschließlich die relevanten Fälle enthielt, wurden noch einige nachträglich gelieferte Zusatzinformationen an den Datensatz gemerged. Weiterhin mussten zahlreiche Kodierungen bei den Confounder- und den Zielvariablen vorgenommen werden, um die relevanten Variablen in der benötigten Kodierung zu erhalten. Die Kodierung der Zielvariablen wird im Folgenden kurz dargestellt. Die ausführliche Beschreibung aller nötigen Kodierungen der Confoundervariablen und die Details zur Bildung der Zielvariablen sind dem Anhang zu entnehmen (vgl. Anhang A).
Die Zielvariable “Asthma in SOLAR II” wurde analog zu SOLAR gebildet und zeigt an, ob bei einem Probanden zum Zeitpunkt von SOLAR II Asthma vorlag. Asthma lag vor, wenn diese Person bei sich selbst Asthmasymptome (pfeifendes oder brummendes Geräusch im Brustkorb) innerhalb der letzten 12 Monate beobachtet hatte und gleichzeitig eine Arztdiagnose Asthma oder spastische/asthmatische Bronchitis vorlag (d.h. Asthma wurde bereits mindestens einmal oder spastiche/asthmatische Bronchitis bereits mehrmals von einem Arzt diagnostiziert).
Analog zu SOLAR wurde ebenfalls die Zielvariable “Allergische Rhinitis in SOLAR II” gebildet. Diese Variable zeigt an, ob zum Zeitpunkt von SOLAR II bei einem Probanden Allergische Rhinitis vorlag. Allergische Rhinitis lag bei einer Person vor, wenn bei

dieser Person innerhalb der letzten 12 Monate Nasenprobleme (Niesanfälle oder laufende, verstopfte Nase ohne Erkältung) zusammen mit juckenden, tränenden Augen auftraten und gleichzeitig schon einmal von einem Arzt allergischer Schnupfen diagnostiziert wurde.

5.2.2 Bereinigung der Tätigkeitsangaben aus SOLAR

Die Tätigkeitsangaben aus SOLAR lagen als R-Datensatz vor (pro Proband eine Zeile). Bevor mit diesen Tätigkeitsangaben gearbeitet werden konnte, mussten folgende Korrekturen durchgeführt werden:
Zunächst wurden die relevanten Fragen zu den Tätigkeitsangaben auf Plausibilität geprüft. Somit musste jemand, der mindestens einen Eintrag zu den Tätigkeitsangaben gemacht hatte (Frage 66) auch die vorherigen Frage (Frage 65) bejahen, dass er in diesem Zeitraum gearbeitet habe. Bei den Probanden, bei denen das nicht der Fall war (n = 10), wurde Frage 65 entsprechend korrigiert (vgl. Anhang A zur Variablenkodierung).
Weiterhin wurden die Angaben zum Anfang und Ende der Tätigkeit auf Plausibilität geprüft. Falls das Ende der Tätigkeit vor dessen Anfang lag, wurde geprüft, ob es sich um einen Eingabefehler handelte oder ob der Proband tatsächlich so geantwortet hatte. Zwei Probanden hatten tatsächlich inkonsistent geantwortet. Bei diesen Fällen wurden sowohl Anfangs- als auch Endzeitpunkt auf NA gesetzt, da nicht entschieden werden konnte, ob bzw. welche Angabe richtig war.
Weiterhin wurde geprüft, ob der ISCO-Code überall vorhanden war, d.h. der Code 9999, falls keine Tätigkeit ausgeübt wurde und ein entsprechender 4-stelliger Code, falls eine Tätigkeit angegeben war. Zwei Fälle mussten korrigiert werden, da sie trotz Tätigkeitsangaben den ISCO-Code 9999 erhalten hatten. In einem weiteren Fall wurden zwar Angaben gemacht, dass eine Tätigkeit ausgeführt wurde, allerdings fehlte die Angabe, welche Tätigkeit ausgeübt wurde. In diesem Fall wurde der ISCO-Code auf 94 (unklar) gesetzt, da keine Kodierung möglich war.
Für spätere Berechnungsschritte war es nötig, die Tätigkeitsangaben zeilenweise anzuordnen, so dass pro Tätigkeit eine Zeile vorhanden war, d.h. jeder Proband fünf Zeilen erhielt (für fünf mögliche Tätigkeitsangaben). Dafür wurden zunächst aus dem Ursprungsdatensatz jeweils pro möglicher Tätigkeitsangabe ein Datensatz extrahiert. Die so gewonnenen fünf Datensätze konnten dann nach einer einheitlichen Benennung der Spaltennamen untereinander zusammengefügt werden.
Zusätzlich durfte in späteren Schritten in der Job-Exposure-Matrix ausschließlich der Code 0 oder 1 vorkommen. Bei allen Probanden, bei denen der Code 9 in der Job-Exposure-Matrix auftauchte (d.h. kein Job ausgeführt wurde), wurden die entsprechenden Zellen auf 0 gesetzt.
Weiterhin wurden die Wochenstundenangaben auf Plausibilität geprüft. Zum einen gaben einige Probanden (n = 3) mehr als 60 Wochenstunden an. Da diese Angaben in Absprache mit Frau Prof. Dr. Radon als unplausibel deklariert wurden, wurden in diesen Fällen die Wochenstunden auf NA gesetzt. Diese gelöschten Angaben wurden dann später imputiert. Zum anderen wurden die Angaben, die weniger als 12 Wochenstunden betrugen, nochmals überprüft, ob hier eine mögliche Verwechslung mit Stunden pro Tag vorlag. Bei allen Probanden, die eine Lehre/Ausbildung, Zivildienst, Bundeswehr, Praktikum oder Freiwilliges Soziales Jahr als Tätigkeit angaben (n = 11), wurde davon ausgegangen, dass es sich

bei der Angabe um Stunden pro Tag handelte. Aus diesem Grund wurden die Angaben mit fünf multipliziert, um die Wochenstundenanzahl zu erhalten. Bei diesen Probanden wurden zusätzlich alle weiteren Tätigkeitsangaben (der jeweiligen Studie) kontrolliert und gegebenenfalls analog korrigiert.
Von 159 Probanden lagen 169 Tätigkeitsangaben vor, bei denen ausschließlich die Angaben zum Ende der Tätigkeit fehlte und die restlichen Angaben vollständig waren (d.h. ISCO-Code, Anfang der Tätigkeit und Wochenstunden). Bei diesen Personen wurde davon ausgegangen, dass die Tätigkeit zum Zeitpunkt der Befragung noch ausgeführt wurde. Aus diesem Grund wurde hierfür ein Ersatzende eingesetzt. Als Ersatzende wurde falls Vorhanden das Ausfülldatum, ansonsten das Einscanndatum des Fragebogens verwendet. Somit konnte die Dauer der jeweiligen Tätigkeit und später auch die entsprechende Exposition bis zum Zeitpunkt der Befragung bestimmt und berechnet werden.
Nach dem Einfügen des Ersatzendes musste erneut geprüft werden, ob Anfang und Ende der Tätigkeit weiterhin plausibel waren. In zwei Fällen lag nun der Beginn der Tätigkeit nach dem Ersatzende, d.h. nach dem Ausfülldatum des Fragebogens. Dies bedeutete wiederum, dass die Probanden eine zukünftige Tätigkeit angaben, die erst nach Ausfüllen des Fragebogens begonnen wird. Da für diese Probanden zum Zeitpunkt des Fragebogens in dieser Tätigkeit noch keine Exposition vorlag, wurde bei diesen Tätigkeitsangaben der ISCO-Code auf 97 (Tätigkeit liegt in der Zukunft) gesetzt. In diesen Fällen ergab sich eine Exposition von 0.
In den Daten gab es auch Personen, die zwar eine Tätigkeit angaben, bei denen allerdings nicht festgestellt werden konnte, um welche Tätigkeit es sich genau handelte und daher auch nicht welche Exposition vorlag. Da für diese Personen kein passender ISCO-Code zugeordnet werden konnte, erhielten sie den ISCO-Code 94 (unklar) oder 95 (Hausfrau). Da in diesen Fällen konservativ vorgegangen wurde, nahm man an, dass keine Exposition bei diesen Personen vorlag.
Das konservative Vorgehen wurde gewählt, um nicht durch Imputation "unsichere" Expositionen in die Daten aufzunehmen und dadurch in den Logitmodellen die Trennschärfe und potenzielle Effekte zu verlieren. Eine Unterschätzung der Exposition wurde in diesem Fall in Kauf genommen, um einer Überschätzung vorzubeugen.
Tabelle 5.1 liefert eine Übersicht über die zusätzlich eingeführten ISCO-Codes.

ISCO-Code	**Beschreibung**	**Anzahl der Personen in SOLAR**
94	unklar	9
95	Hausfrau	1
97	Tätigkeit liegt in der Zukunft	2
98	Schüler oder Student	0

Tabelle 5.1: Übersicht über die zusätzlich eingeführten ISCO-Codes

5.2.3 Bereinigung der Tätigkeitangaben aus SOLAR II

Die Tätigkeitangaben aus SOLAR II lagen als Excel-Datei vor, in der die Tätigkeitangaben bereits zeilenweise angeordnet waren, d.h. pro Proband bereits fünf Zeilen angelegt waren. Erste Plausibilitätschecks und Korrekturen wurden bereits in der Excel-Datei durchgeführt.
Zunächst wurde die ISCO-Kodierung überprüft. Drei Fälle waren enthalten, die trotz Tätigkeitsangaben den Code 8888 erhielten, der für "Keine Tätigkeit" stand. Nach Absprache mit Frau Kellberger und teilweise erneuten Telefonaten mit den Probanden wurden die ISCO-Codes (und evtl. zusätzlich die Expositionen) entsprechend korrigiert (Details vgl. Anhang A).
Ein Proband gab statt der erlaubten fünf Tätigkeitsangaben sechs Jobs an. Da es sich dabei bei zwei aufeinander folgenden Angaben um die gleiche Tätigkeit (mit dem gleichen ISCO-Code) handelte, konnten diese beiden Angaben zu einer Tätigkeit mit entsprechend längerer Dauer zusammengefasst werden. Somit hatte auch dieser Proband nur noch fünf Tätigkeitsangaben.
Nachdem diese ersten Korrekturen in Excel vorgenommen wurden, wurde die Datei in R eingelesen und noch einige Zusatzinfos angemerged.
Als ersten Kontrollschritt im R-Datensatz wurde die Plausibilitätsprüfung zu den Tätigkeitsangaben durchgeführt. Wurde mindestens ein Eintrag bei den Tätigkeitsangaben (Frage 93) gemacht, so musste auch die Frage, ob gearbeitet wurde (Frage 92), bejaht worden sein. War dies nicht der Fall (n = 27), wurde die entsprechende Korrektur in einer neuen Variable vorgenommen.
Wie bei SOLAR wurde Anfang und Ende der Tätigkeit auf Plausibilität geprüft, Eingabefehler korrigiert und tatsächlich unplausible Angaben gelöscht (n = 3).
Waren in der Job-Exposure-Matrix fehlende Werte enthalten, da der Proband nicht gearbeitet hat, so wurden diese durch 0 ersetzt.
Analog zu SOLAR ging man auch bei den Wochenstunden vor. Zunächst wurden Angaben, die mehr als 60 Wochenstunden umfassten, gemäß der Vereinbarung mit Frau Prof. Dr. Radon auf NA gesetzt (n = 6). Weiterhin wurden die Angaben, die weniger als 12 Wochenstunden betrugen, auf eine mögliche Verwechslung mit Stunden pro Tag hin überprüft. Bei allen Probanden, die Ausbildung, Zivildienst, Bundeswehr, Praktikum oder Freiwilliges Soziales Jahr als Tätigkeit angaben (n = 18), wurden die Angaben mit fünf multipliziert, um die Wochenstundenanzahl zu erhalten. Bei diesen Probanden wurden zusätzlich alle weiteren Tätigkeitsangaben (der jeweiligen Studie) kontrolliert und gegebenenfalls analog korrigiert.
Bei den Datensätzen, bei denen nur das Ende der Tätigkeit fehlte, wurde als Ende falls Vorhanden das Ausfülldatum oder ansonsten das Einscanndatum des Fragebogens eingesetzt. Somit konnten 300 Datenzeilen von 275 Probanden vervollständigt werden.
Nach dieser Vervollständigung musste erneut die Plausibilitätsprüfung zu Beginn und Ende der Tätigkeit durchgeführt werden d.h. ob das Anfangsdatum der Tätigkeit nach dem Ausfüll- bzw. Einscanndatum des Fragebogens lag. In fünf Fällen lag diese Situation vor und es wurde bei diesen Tätigkeitsangaben wie auch in SOLAR der ISCO-Code auf 97 (Tätigkeit liegt in der Zukunft) gesetzt.
Analog zu SOLAR gab es auch hier den Codes 94 (unklar) für Probanden, deren Tätigkeit nicht kodiert werden konnte. Zusätzlich dazu gab es auch den Code 98 für Schüler und

Studenten, für die kein passender ISCO-Code existiert. Auch bei dieser Personengruppe wurde konservativ vorgegangen und die Exposition auf 0 gesetzt.
Tabelle 5.2 liefert eine Übersicht über die zusätzlich eingeführten ISCO-Codes.

ISCO-Code	**Beschreibung**	**Anzahl der Personen in SOLAR**
94	unklar	6
95	Hausfrau	0
97	Tätigkeit lag in der Zukunft	5
98	Schüler oder Student	6

Tabelle 5.2: Übersicht über die zusätzlich eingeführten ISCO-Codes

5.2.4 Zusammenfassung der Bereinigungsschritte aus SOLAR und SOLAR II

Folgende Tabelle 5.3 liefert eine Übersicht über die Bereinigungsschritte der Tätigkeitsangaben aus SOLAR und SOLAR II.

Bereinigungsschritt	**korrigierte Fälle in SOLAR**	**korrigierte Fälle in SOLAR II**
Plausibilitätsprüfung der Tätigkeitsangaben	10	27
Bereinigung der Zeitangaben	2	3
Korrektur des ISCO-Codes	3	3
Bereinigung der Wochenstunden	14	24
Ersetzen des fehlenden Endes der Tätigkeit	159	275

Tabelle 5.3: Zusammenfassung der Bereinigungsschritte der Tätigkeitsangaben

5.2.5 Auswahl der Personen mit vollständigen Tätigkeitsangaben jeweils für SOLAR und SOLAR II getrennt

Um festzustellen, ob ein Proband insgesamt vollständige Tätigkeitsangaben hatte, musste zunächst jeweils für SOLAR und SOLAR II diese Frage beantwortet werden. Die im folgenden dargestellten Schritte wurden separat auf den Tätigkeitsangaben aus SOLAR und SOLAR II durchgeführt.
Zunächst wurden eine Reihe von Hilfsvariablen gebildet.

- Eine Variable war notwendig, die angab, ob in der jeweiligen Zeile ein Eintrag mit Tätigkeitsangaben vorhanden war.
- Für jeden Probanden wurde dann aus dieser Variable pro Studie eine Summe gebildet, die angab, wie viele Tätigkeitsangaben dieser Proband in der jeweiligen Studie machte.

- Weiterhin wurde eine Hilfsvariable gebildet, die angab, ob die Zeile vollständig ausgefüllt wurde oder imputiert werden musste. Eine Zeile wurde in folgenden Fällen als vollständig kodiert:
  - Die angegebenen Tätigkeitsangaben in dieser Zeile waren vollständig ausgefüllt, d.h. Angaben zu Beginn und Ende der Tätigkeit, Wochenstunden und der ISCO-Code lagen vor.
  - Der ISCO-Code 94, 95, 97 oder 98 trat auf. In diesen Fällen wurde die Exposition (in einem späteren Schritt) auf 0 gesetzt, d.h. hier musste nichts mehr imputiert werden. Beispielsweise hatten hier fehlende Zeitangaben keine Auswirkung, da die Dauer der Tätigkeit in einem späteren Schritt mit der Exposition multipliziert wurde, die hier auf 0 festgelegt wurde.
  - Es wurden weniger als acht Wochenstunden gearbeitet. Bei diesen Probanden wurde die Exposition (in einem späteren Schritt) ebenfalls auf 0 gesetzt, d.h. hier musste ebenfalls nichts mehr imputiert werden, auch wenn Angaben fehlten.
  - Nur in SOLAR II: Es wurde die Fragebogenoption angekreuzt, dass ausschließlich Jobs mit weniger als acht Wochenstunden ausgeführt wurden. Da man bei diesen Fällen keine Imputation mehr vornehmen musste, da bereits klar war, dass keine berufliche Exposition vorlag, galten diese Zeilen als vollständig.
- Aus dieser Hilfsvariable wurde eine Variable gebildet, die pro Proband die Summe der vollständig ausgefüllten Zeilen je Studie enthielt.
- Aus dieser Variable konnte dann eine Variable gebildet werden, die angab, ob alle fünf möglichen Tätigkeitsangaben je Studie vollständig waren.
  Dabei wurden die Angaben eines Proband in der jeweiligen Studie als **vollständig** definiert, wenn einer der folgenden Fälle zutraff:
  - Probanden, die bis zur jeweiligen Studie **nicht arbeiteten**
    Ein Proband galt als vollständig, wenn er nie gearbeitet hatte.
  - Probanden mit **durchgängig vollständigen Tätigkeitsangaben** bis zur jeweiligen Studie
    Die Probanden arbeiteten bis zur jeweiligen Studie, machten mindestens eine Tätigkeitsangabe und die Anzahl der ausgefüllten Zeilen stimmte mit der Anzahl der vollständig ausgefüllten Zeilen überein. Da zuvor bei Probanden, bei denen nur der Endzeitpunkt der Tätigkeit fehlte, dieser durch ein Ersatzende ersetzt wurde, galten auch diese Personen als vollständig.
    Auch die Personen, die in SOLAR II angaben, dass sie ausschließlich Jobs mit weniger als acht Wochenstunden ausübten, galten im Rahmen dieser Definition als vollständig.
  - Probanden, mit **unklarer Arbeitssituation** bis zur jeweiligen Studie
    Lag für einen Probanden keine Angabe vor, ob er in der jeweiligen Studie arbeitete, so wurde nach Absprache mit Frau Prof. Dr. Radon konservativ vorgegangen. Man nahm an, dass bei diesem Probanden keine berufliche Exposition vorlag, d.h. eine Imputation hier nicht erforderlich war.

  - Probanden, mit **komplett fehlenden Tätigkeitsangaben** bis zur jeweiligen Studie
    Bei Probanden, die angaben, dass sie in der jeweiligen Studie arbeiteten, aber keine Einträge zu den Tätigkeitsangaben vorlagen, wurde ebenfalls konservativ vorgegangen. Dieses Vorgehen wurde gewählt, da man bei diesen Personen davon ausgehen konnte, dass sie weniger als acht Wochenstunden arbeiteten und deshalb die Tätigkeitsangaben (gemäß der Fragebogenanweisung) nicht ausfüllten. Folglich ging man hier davon aus, dass keine Exposition vorlag und führte später keine Imputation bei diesen Probanden durch.

  Die Angaben eines Probanden galten in der jeweiligen Studie als **unvollständig**, wenn

  - lückenhafte Tätigkeitsangaben in der jeweiligen Studie gemacht wurden, d.h. die Anzahl der ausgefüllten Zeilen nicht mit der Anzahl der vollständig ausgefüllten Zeilen übereinstimmte.

- Für die unvollständigen Fälle war es nötig, eine Variable zu bilden, die angab, ob in der jeweiligen Zeile eine Imputation vorgenommen werden musste. In einer Zeile musste imputiert werden, wenn:

  - in dieser Zeile ein Eintrag vorlag, dieser aber als unvollständig definiert wurde.

  In einer Zeile musste in folgenden Fällen nicht imputiert werden:

  - Der Proband war vollständig
  - Die Zeile war vollständig
  - In einer Zeile lagen berechtigterweise keine Einträge vor
    z.B. da nur eine Tätigkeit ausgeübt wurde und daher die restlichen vier Zeilen leer bleiben und auch nicht imputiert werden mussten

Ablaufschema: Auswahl der Personen mit vollständigen Tätigkeitsangaben
Bildung diverser Hilfsvariablen: “Zeile mit Tätigkeitsangabe” (Ausprägungen 0 oder 1) “Summe der Zeilen mit Tätigkeitsangaben pro Proband” (Ausprägungen 0-5) “Vollständige Zeile” (Ausprägungen 0 oder 1) “Summe der vollständigen Zeilen pro Proband” (Ausprägungen 0-5)
Bildung folgender Variablen auf Basis der Hilfsvariablen: “Proband vollständig” (Ausprägungen 0 oder 1) “Imputation in Zeile” (Ausprägungen 0 oder 1)

Tabelle 5.4: Ablaufschema: Auswahl der Probanden mit vollständigen Tätigkeitsangaben

Für die beiden Studien SOLAR und SOLAR II liefert die Tabelle 5.5 eine Übersicht über die Probanden, die in der jeweiligen Studie vollständige Tätigkeitsangaben aufwiesen.

Probanden	**Anzahl Personen in SOLAR**	**Anzahl Personen in SOLAR II**
die nicht arbeiteten (keine Exposition)	477	500
mit durchgängig vollständigen Tätigkeitsangaben (Exposition möglich)	342	630
mit unklarer Arbeitssituation (Exposition als nicht vorhanden angenommen)	5	1
mit komplett fehlenden Tätigkeitsangaben (Exposition als nicht vorhanden angenommen)	320	4
mit vollständigen Tätigkeitsangaben	1.144 (98%)	1.135 (96%)

Tabelle 5.5: Übersicht über die Probanden mit vollständigen Tätigkeitsangaben

Eine Übersicht über die unvollständigen Probanden jeweils aus den beiden Studien SOLAR und SOLAR II liefert die Tabelle 5.6.

Probanden	**Anzahl Personen in SOLAR**	**Anzahl Personen in SOLAR II**
mit unvollständigen Tätigkeitsangaben	43 (mit 49 Angaben)	52 (mit 59 Angaben)

Tabelle 5.6: Übersicht über die Probanden mit unvollständigen Tätigkeitsangaben

5.2.6 Zusammenführung der Tätigkeitsangaben aus SOLAR und SOLAR II

Nach Durchführung der soeben beschriebenen Korrektur- und Kontrollschritte und der Definition der vollständigen Tätigkeitsangaben in den beiden separaten Dateien aus SOLAR und SOLAR II, wurde jeweils ein Datensatz erstellt, der mit den Tätigkeitsangaben der anderen Studie zusammengeführt werden konnte. Diese beiden separaten Dateien wurden dann zu einer Datei zusammengeführt. In dieser Datei waren pro Proband zehn Zeilen vorhanden (für zehn mögliche Tätigkeitsangaben). Damit die erste Tätigkeit jedes Probanden jeweils in der ersten der zehn Zeilen stand, wurde die Datei nach Kohortennummer(knr), Anfangsjahr und Anfangsmonat sortiert.
In dieser Datei wurden wiederum einige Hilfsvariablen gebildet.
(Details zur Bildung der Variablen vgl. Anhang A)

- Zunächst wurden folgende Variablen gebildet, die die separat vorliegenden Informationen aus SOLAR und SOLAR II zusammenführen:
  - Summe der Anzahl der Einträge
    Summiert pro Proband die Anzahl der Einträge aus SOLAR und SOLAR II auf
  - Summe der vollständigen Zeilen
    Summiert pro Proband die Anzahl der vollständigen Zeilen aus SOLAR und SOLAR II auf

- Dann war eine Variable nötig, die angab, ob die jeweilige Tätigkeit mindestens acht Wochenstunden lang ausgeführt wurde. Nur bei Tätigkeiten, die mindestens acht Wochenstunden umfassten, wurde später die Exposition betrachtet. Diese Variable war nötig, um bei der Expositionsberechnung eine Variable zu Verfügung zu haben, durch die jeder Tätigkeit mit weniger als acht Wochenstunden die Exposition 0 zugeteilt und bei allen anderen Jobs die Exposition berechnet werden konnte.
  (Details zur Berechnung der Exposition vgl. Kapitel 7)

- Auf Basis dieser Variable wurde eine Variable gebildet, die angab, wie viele Tätigkeiten mit mindestens acht Wochenstunden der jeweilige Proband hatte.

- Weiterhin wurde für die Regressionsmodelle eine Variable benötigt, die angab, ob jemals in SOLAR oder SOLAR II gearbeitet wurde. War bei einem Probanden mindestens eine Tätigkeit mit mindestens acht Wochenstunden vorhanden, so wurde diese Variable auf “Ja” gesetzt, ansonsten auf “Nein”.

- Pro Tätigkeitsangabe wurde die Dauer der jeweiligen Tätigkeit in Monaten berechnet. Diese Berechnung erfolgte mit Hilfe der Angaben zum Anfang und Ende der Tätigkeit. Falls mindestens eine Angabe zu Anfang oder Ende der Tätigkeit fehlte (Anfangsmonat, Anfangsjahr, Endmonat, Endjahr), konnte die Dauer der Tätigkeit nicht berechnet werden und musste somit vorläufig auf NA gesetzt werden. In einem späteren Imputationsschritt konnten diese fehlenden Angaben aufgefüllt werden.
  Bei Personen mit ISCO-Code 94, 95, 97 und 98 wurde die Dauer standardmäßig auf 0 gesetzt, damit bei diesen Fällen in späteren Schritten keine Imputation vorgenommen wurde. Zusätzlich wurde bei diesen Codes die Exposition durchgängig auf 0 gesetzt und somit haben diese Personen auch keine Belastung.

Ablaufschema: Übersicht über die gebildeten Variablen
Hilfsvariablen:
"Summe der Anzahl der Einträge insgesamt" (Ausprägungen 0-10)
"Summe der Anzahl der vollständigen Zeilen insgesamt" (Ausprägungen 0-10)
"Tätigkeit für mindestens 8 Stunden ausgeführt" (Ausprägungen 0 oder 1)
"Summe der für mindestens 8 Stunden ausgeführten Tätigkeiten" (Ausprägungen 0-10)
Weitere benötigte Variablen:
"Jemals gearbeitet" (Ausprägungen 0 oder 1)
"Dauer der Tätigkeit" (in Monaten)

Tabelle 5.7: Ablaufschema: Übersicht über die gebildeten Variablen

5.3 Auswahl der Probanden mit insgesamt vollständigen Tätigkeitsangaben

Die Angaben eines Probanden galten als (insgesamt) vollständig, wenn er in beiden Studien als vollständig definiert wurde, d.h. in beiden Studien keine (relevanten) Angaben zu den Tätigkeitsangaben fehlten. Von den ursprünglich 1.187 Probanden konnten gemäß obiger Definition 1.094 Personen (92%) mit vollständigen Tätigkeitsangaben ausgewählt werden. Auf Basis dieser Probanden, die sowohl in SOLAR als auch in SOLAR II vollständige Tätigkeitsangaben machten, sollten in einem späteren Schritt zwei logistische Regressionsmodell berechnet werden, um das Auftreten von Asthma und Allergischer Rhinits zu analysieren (vgl. Kapitel 8).

Probanden mit insgesamt vollständigen Tätigkeitsangaben
Insgesamt vollständige Tätigkeitsangaben lagen bei 1.094 Probanden vor, d.h. bei 92 % der ursprünglich 1.187 zur Verfügung stehenden Probanden

5.4 Auswahl der Probanden mit insgesamt unvollständigen Tätigkeitsangaben

Fehlte bei einem Probanden in SOLAR und/oder SOLAR II mindestens eine Angabe zu den Tätigkeitsangaben, so wurde er als unvollständig definiert. Bei diesen Probanden mussten noch Imputationen vorgenommen werden, um die Tätigkeitsangaben zu vervollständigen. Als Basis dieser Imputation musste zunächst analysiert werden, welche Fehlendmuster in den Tätigkeitsangaben vorlagen. Die folgende Tabelle 5.8 soll das vorliegende Fehlendmuster detailliert darstellen.

Fehlendmuster: folgende Angaben fehlten	**Anzahl Zeilen**
Wochenstunden	27
Anfangsmonat und Endmonat	17
Zeitangaben (bis auf Anfangsjahr)	15
Wochenstunden, Zeitangaben zum Ende der Tätigkeit	14
Zeitangaben zum Anfang und Ende der Tätigkeit	14
Zeitangaben und Wochenstunden	13
Wochenstunden und Zeitangaben (bis auf Anfangsjahr)	3
Anfangsjahr und Endjahr	1
Anfangsmonat	1
Anfangsmonat und Wochenstunden	1
Zeitangaben (bis auf Anfangsmonat)	1
Endjahr	1

Tabelle 5.8: Übersicht über die Fehlendmuster in den Tätigkeitsangaben

Bei diesen 108 Tätigkeitsangaben von insgesamt 93 Probanden musste an den entsprechenden Stellen eine Imputation durchgeführt werden. Das genaue Vorgehen zur Imputation der fehlenden Tätigkeitsangaben wird in Kapitel 6.2 ausführlich erläutert.

6 Praktische Umsetzung der Imputationsmethoden

Die Imputation der fehlenden Werte im Datensatz wurde in mehrere Schritte aufgeteilt. Zunächst wurden ausschließlich die Confoundervariablen betrachtet. In einem weiteren Schritt wurden dann die fehlenden Tätigkeitsangaben imputiert.
In beiden Schritten konnte jeweils von dem Fehlendmechanismus MAR (missing at random) ausgegangen werden, da es keinen Grund gab, anzunehmen, dass die Wahrscheinlichkeit für das Fehlen eines Wertes in den Confoundervariablen oder den Tätigkeitsangaben von der jeweiligen Variable selbst abhängen. Beispielsweise gab es keinen Grund dafür anzunehmen, dass die Wahrscheinlichkeit, dass die Angabe zu den Wochenstunden fehlte, höher ist, wenn 20 Stunden pro Woche gearbeitet wurden als wenn 10 Wochenstunden gearbeitet wurden.

6.1 Imputation der fehlenden Werte potenzieller Confoundervariablen

6.1.1 Übersicht über die fehlenden Werte der Confounder

Betrachtete man in einem ersten Schritt nur die potenziellen Confoundervariablen (d.h. die Tätigkeitsangaben wurden zunächst außen vor gelassen), so stellte man fest, dass 1.050 Probanden vollständige Angaben bei den Confoundern machten (88%). Das bedeutete im Umkehrschluss, dass bei 137 Probanden mindestens eine Angabe bei diesen Variablen fehlte. Folgende Tabellen sollen nun eine Übersicht über die Ausprägungen und fehlenden Werte der potenziellen Confoundervariablen geben.
Keine fehlenden Werte waren in den Variablen Studienzentrum und Geschlecht vorhanden. Bei den medizinischen Variablen für Neurodermitis, Allergische Rhinitis und Asthma wurden zuvor alle Probanden mit fehlenden Werten in diesen Variablen aus den Analysen für diese Arbeit ausgeschlossen. Folglich erfolgte bei diesen Variablen keine Imputation. Der Vollständigkeit halber werden sie allerdings in den Tabellen aufgeführt, um die Häufigkeiten der einzelnen Ausprägungen darzustellen.

Zunächst werden in Tabelle 6.1 die Variablen dargestellt, die aus ISAAC II entnommen wurden.

Variablenbeschreibung und Häufigkeit	**vorhandene Werte**	**fehlende Werte**
In Deutschland geboren Ja: n=1.128 Nein: n=57	1.185	2
Atopie der Eltern Ja: n=536 Nein: n=639	1.175	12
Anzahl Geschwister 0: n=189 1: n=654 2: n=216 3: n=66 4: n=18 5: n=8 6: n=1 7: n=2	1.154	33
als Säugling gestillt Ja: n=956 Nein: n=185	1.141	46
Passivrauch Eltern Raucher: n=332 Eltern Ex-Raucher: n=95 Eltern Nichtraucher: n=737	1.164	23
Studienzentrum Dresden: n=571 München: n=616	1.187	0
Neurodermitis zum Zeitpunkt ISAAC II Ja: n=121 Nein: n=1.066	1.187	0
Allergische Rhinitis zum Zeitpunkt ISAAC II Ja: n=86 Nein: n=1.101	1.187	0
Asthma zum Zeitpunkt ISAAC II Ja: n=47 Nein: n=1.140	1.187	0
Sozioökonomischer Status Hoch (Fachabitur/Abitur/Studium): n=693 Niedrig (Niedrigere Ausbildung): n=479	1.172	15

Tabelle 6.1: Übersicht über die potenziellen Confoundervariablen aus ISAAC II

Die Variablen, die aus SOLAR stammten, wiesen die in Tabelle 6.2 dargestellte Struktur auf.

Variablenbeschreibung und Häufigkeit	**vorhandene Werte**	**fehlende Werte**
Passivrauch Ja: n=723 Nein: n=455	1.178	9
Rauchverhalten Raucher: n=341 Nichtraucher: n=839	1.180	7
Geschlecht Männlich: n=480 Weiblich: n=707	1.187	0
Neurodermitis zum Zeitpunkt SOLAR Ja: n=137 Nein: n=1.050	1.187	0
Allergische Rhinitis zum Zeitpunkt SOLAR Ja: n=158 Nein: n=1.029	1.187	0
Asthma zum Zeitpunkt SOLAR Ja: n=43 Nein: n=1.144	1.187	0

Tabelle 6.2: Übersicht über die potenziellen Confoundervariablen aus SOLAR

Die Variablen, die aus SOLAR II stammten, wiesen die in Tabelle 6.3 dargestellte Struktur auf.

Variablenbeschreibung und Häufigkeit	**vorhandene Werte**	**fehlende Werte**
Passivrauch Ja: n=677 Nein: n=495	1.172	15
Rauchverhalten Raucher: n=425 Ex-Raucher: n=99 Nichtraucher: n=657	1.181	6
Schulbildung Höhere Schulbildung (Abi/FH): n=731 Niedrigere Schulbildung: n=452	1.183	4

Tabelle 6.3: Übersicht über die potenziellen Confoundervariablen aus SOLAR II

Die Variablen, die als Zielgrößen der Logitmodelle dienten, wiesen die in Tabelle 6.4 dargestellte Struktur auf.

Variablenbeschreibung und Häufigkeit	**vorhandene Werte**	**fehlende Werte**
Allergische Rhinitis zum Zeitpunkt SOLAR II Ja: n=181 Nein: n=1.006	1.187	0
Asthma zum Zeitpunkt SOLAR II Ja: n=60 Nein: n=1.127	1.187	0

Tabelle 6.4: Übersicht über die Zielvariablen der Logitmodelle

Die Variablen, die als Zusatzinformation für die Imputation verwendet wurden, haben die in Tabelle 6.5 dargestellte Struktur. Das Vorliegen von Zusatzinformation für die Imputation tritt sehr selten auf, und sollte deshalb an dieser Stelle als besonders vorteilhaft herausgestellt werden.

Variablenbeschreibung und Häufigkeit	**vorhandene Werte**	**fehlende Werte**
Berufssituation - SOLAR HauptschülerIn: n=8 RealschülerIn: n=137 GymnasiastIn: n=617 SchülerIn einer anderen Schule: n=103 AuszubildendeR/BerufsschülerIn: n=297 StudentIn: n=3 Angestellt: n=6 Arbeitslos&-suchend: n=5 Sonstiges: n=10	1.186	1
Berufssituation - SOLAR II AuszubildendeR/BerufsschülerIn: n=198 StudentIn (hauptberuflich): n=519 Angestellt: n=349 Selbstständig: n=12 Arbeitslos&-suchend: n=35 Aus gesundh. Gründen nicht arbeitend: n=2 Mutterschutz/Elternzeit/Beurlaubung: n=10 Sonstiges: n=60	1.185	2

Tabelle 6.5: Übersicht über die zusätzlichen Variablen für die Imputation

6.1.2 Vorgehen bei der Imputation der fehlenden Werte der Confoundervariablen

Um die potenziellen Confoundervariablen zu vervollständigen, wurden drei Datensätze mittels multipler Imputation mit Hilfe des R-Packages Amelia-II und zwei Datensätze mittels der Imputationsmethode Ziehen aus der empirischen Verteilung erstellt, so dass am Ende fünf vervollständigte Datensätze vorlagen. Beide Grundkonzepte wurden bereits in Kapitel 4 zur Theorie der fehlenden Daten beschrieben. Im Folgenden soll deren praktische Umsetzung ausführlich beschrieben werden. Zunächst soll das Schaubild 6.1 das Vorgehen bei der Imputation der fehlenden Fragebogendaten veranschaulichen.

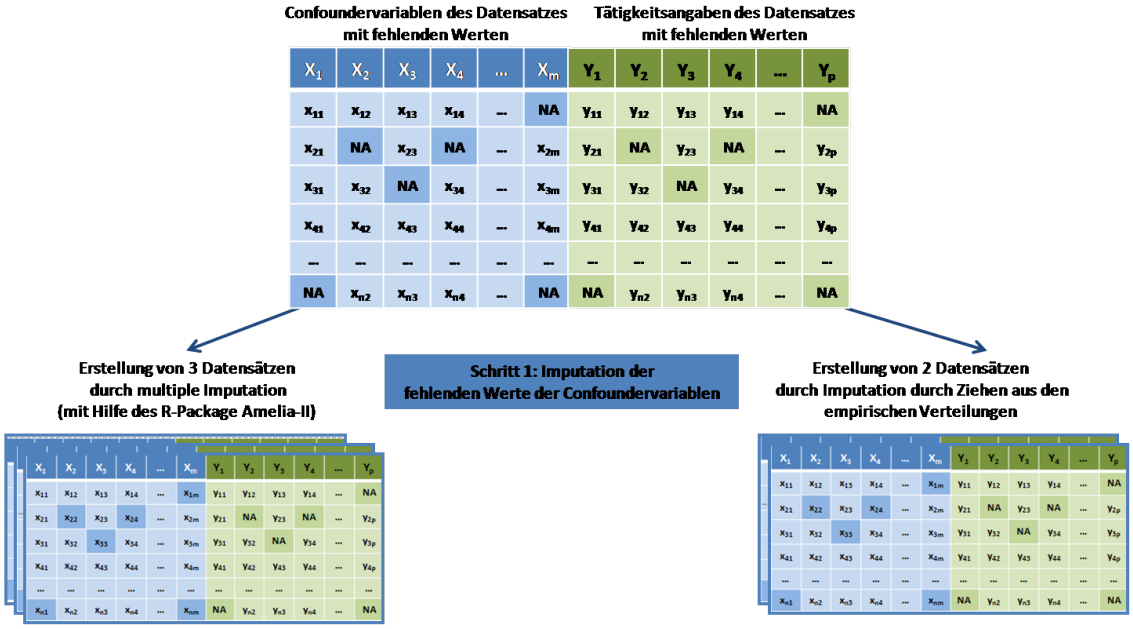

Abbildung 6.1: Vorgehensweise bei der Imputation der fehlenden Werte der Confoundervariablen

Multiple Imputation mit Hilfe des R-Packages Amelia-II

Grundsätzliches zu Amelia-II

Das R-Package Amelia-II erlaubt dem Nutzer die multiple Imputation von fehlenden Werten, um dann in die Analyse alle vorliegenden Informationen des Datensatzes einfließen zu lassen und dabei die Verzerrung und Ineffizienz zu umgehen, die beim Ausschluss von fehlenden Daten auftreten. Honaker, King und Blackwell schrieben dieses Package, um die Verwendung der multiplen Imputation durch ein einfach anzuwendendes Programm stärker zu verbreiten. [HONAKER et al. 2009] Man übergibt diesem Package einen unvollständigen Datensatz, das Package imputiert dann die fehlenden Werte und gibt dem Benutzer die gewünschte Anzahl m an vervollständigten Datensätzen zurück. Vorteil des Amelia-II Packages ist, dass mehr Variablen und Beobachtungen in kürzerer Zeit imputiert werden können als bei anderen Paketen.

Bootstrap-EM-Algorithmus
Das R-Package Amelia-II basiert auf dem Bootstrap-EM-Algorithmus (Bootstrap- Expectation-Maximization-Algorithmus). Dazu wird zunächst die Annahme getroffen, dass die vorliegenden Daten D multivariat normalverteilt mit Mittelwertvektor μ und Varianzmatrix Σ sind ($D \sim N(\mu,\Sigma)$). [HONAKER und KING 2008] Diese Annahme der multivariaten Normalverteilung hat sich in zahlreichen Situationen mit fehlenden Daten bereits als hilfreich erwiesen. [KING et al. 2001] Folgende Schritte werden dann bei der Imputation der fehlenden Werte nötig:

- Durchführen des Bootstrapping Algorithmus, d.h. es werden m Stichproben der Größe n mit Zurücklegen aus den Daten D gezogen
- In jeder Stichprobe wird der EM-Algorithmus durchgeführt und daraus resultieren dann Punktschätzer für μ und Σ
- Deterministische Berechnung des zu imputierenden Wertes im ursprünglichen Datensatz mit Hilfe der Schätzer von μ und Σ über eine lineare Regressionsbeziehung

Der EM-Algorithmus ist ein sehr allgemeiner iterativer Algorithmus für ML-Schätzprobleme bei unvollständigen Daten, der von Little und Rubin ausführlich beschrieben wird.
[LITTLE und RUBIN 2002] Wie funktioniert nun dieser Algorithmus konkret? Der EM-Algorithmus ist ein iterativer Prozess, der aus zwei Schritten besteht, dem E-Schritt (Expecation-Schritt) und dem M-Schritt (Maximization-Schritt). Im E-Schritt werden zunächst die suffizienten Statistiken [1] für μ und Σ geschätzt. Im M-Schritt werden basierend auf diesen Schätzern die Parameter wie Mittelwert, Kovarianzen und Regressionsmodellparameter neu berechnet. Dann wird mit Hilfe dieser neuen Parameterschätzer, die als richtig angenommen werden, der E-Schritt erneut durchgeführt, das bedeutet, die suffizienten Statistiken werden erneut geschätzt. In jedem Durchlauf dieser Iteration liegen somit die aktuellen Schätzer näher an den wahren Werten als im vorherigen Durchlauf der Iteration. Dieser iterative Prozess wird dann so lange fortgeführt, bis die Parameterschätzer konvergieren, d.h. sie sich von Iteration zu Iteration nur noch minimal verändern und somit in ausreichend naher Umgebung der wahren Schätzer liegen. Die Anzahl der Iterationen ist dabei bestimmt durch den Fehlendanteil in den Daten. Wäre der Fehlendanteil gleich null, so würde der Algorithmus sofort konvergieren. Je höher der Fehlendanteil hingegen ist, umso länger benötigt der Algorithums bis zur Konvergenz.
Nachdem der EM-Algorithmus konvergiert ist, erhält man feste Punktschätzer für μ und Σ. Mithilfe dieser Schätzer können dann die fehlenden Werte im ursprünglichen Datensatz ersetzt werden, da aufgrund der Annahme der multivariaten Normalverteilung aus den Punktschätzern alle benötigten Regressionsbeziehungen abgeleitet werden können.
In dem man diese Prozedur dann auf jeder der m Bootstrapping-Stichproben wiederholt, erhält man somit m vervollständigte Datensätze. Somit wird bei Amelia-II die multiple Imputation durch die Anwendung des EM-Algorithmus auf die m verschiedenen Bootstrapping-Stichproben erreicht.
In einfachen Fällen bedeutet der EM-Algorithmus also: Regression berechnen um β zu

[1] Suffiziente Statistiken sind Statistiken, die sämtliche Informationen über den geschätzten Parameter beinhalten, die in der Stichprobe enthalten sind.

schätzen, die fehlenden Werte durch vorhergesagte Werte ersetzen, β erneut schätzen und so weiter, bis zur Konvergenz der Schätzer. [KING et al. 2001]

Zuvor wurde bereits erwähnt, dass in Amelia-II die Annahme der multivariaten Normalverteilung getroffen wird. Diese Annahme kann alllerdings meist nur approximativ erfüllt werden, da sehr wenige Datensätze ausschließlich stetige und unbeschränkte Variablen enthalten. Vielmehr enthalten sehr viele Datensätze, die auf Befragungen basieren, dichotome oder kategoriale Variablen. Es konnte allerdings gezeigt werden, dass die Annahme der (approximativen) multivariaten Normalverteilung ähnlich gut funktioniert wie komplexe Methoden, die speziell für kategoriale oder gemischte Daten entwickelt wurden. [KING et al. 2001] Um die Anpassung an das Modell dennoch zu verbessern, wurden in Amelia-II Methoden zur Behandlung von nominalen Variablen implementiert.

Behandlung von nominalen Variablen
Jede nominale Variable muss in Amelia-II spezifiziert werden. Für eine Variable mit p-Kategorien bestimmt Amelia-II zunächst p, d.h. die Anzahl der Kategorien, und erstellt dann p-1 binäre Variablen. Für diese binären Variablen werden dann stetige Werte imputiert. Diese stetigen Werte werden in Wahrscheinlichkeiten für jede der p Kategorien transformiert. Eine der Kategorien wird dann auf Basis dieser Wahrscheinlichkeiten gezogen, die ursprüngliche p-kategoriale Variable wieder hergestellt und dem Nutzer ausgegeben.
Als nominale Variablen wurden alle Variablen aus dem Fragebogen definiert (Studienzentrum, Geschlecht, Sozioökonomischer Status, In Deutschland geboren, Atopie der Eltern, Geschwister, als Säugling gestillt, Passivrauch (jeweils aus ISAAC II, SOLAR und SOLAR II), Rauchverhalten (jeweils aus SOLAR und SOLAR II) und Schulbildung. Zusätzlich wurden auch die medizinischen Variablen als nominale Variablen definiert (Neurodermitis, Allergische Rhinitis und Asthma jeweils aus ISAAC II, SOLAR und SOLAR II). Weiterhin konnten als Zusatzinformation die Angaben zur Berufssituation in SOLAR und in SOLAR II für die Imputation genutzt werden, die im Rahmen der drei Studien erhoben wurden.

Behandlung von Identifikationsvariablen
Identifikationsvariablen (z.B. IDs), die nicht für die Imputation verwendet werden, aber trotzdem im Datensatz verbleiben sollen, werden in Amelia-II durch den Befehl “idvars” spezifiziert.
Die Kohortennummer (knr), die jeden Probanden der vorliegenden Studie eindeutig identifiziert, wurde durch den Befehl “idvars” als Identifikationsvariable spezifiziert, so dass sie nicht für die Imputation verwendet wurde, aber im vervollständigten Datensatz weiterhin existierte.

Auswahl der Variablen bei der Imputation
Ein entscheidender Punkt bei der Imputation ist die Auswahl der Variablen, die für die Imputation verwendet werden. Eine generelle Richtlinie ist dabei, dass für die Imputation mindestens die Variablen verwendet werden sollten, die auch später in die Analyse des Datensatzes (z.B. in ein Regressionsmodell) eingehen. [KING et al. 2001] Einen folgenschweren Fehler, den man begehen kann, ist das Weglassen einer Variable bei der

Imputation, die bei der Analyse des Datensatzes berücksichtigt wird. Dadurch werden Schätzer, die die Beziehung zwischen dieser (weggelassenen) Variable und anderen Variablen misst, gegen null verzerrt.
Gemäß dieser Richtlinie wurden in dieser Arbeit zunächst alle Variablen, die als mögliche Kovariablen in Betracht gezogen wurden, und auch die (vollständig vorliegenden) Zielvariablen (Allergische Rhinitis und Asthma in SOLAR II) für die Imputation der fehlenden Werte verwendet.

Vorgehen bei hohen Korrelationen
Enthält der Datensatz hohe Korrelationen zwischen den Variablen, so ist es sinnvoll, einen sogenannten *ridge prior* anzuwenden. Im Package Amelia-II kann dies durch das Hinzufügen der Option "empri=x" angewandt werden, das heißt, man gibt eine positive Zahl x als Prior an. Dies entspricht in etwa dem Hinzufügen von x künstlichen Beobachtungen mit den selben Mittelwerten und Varianzen wie die vorhandenen Daten, allerdings mit Kovarianzen gleich null. Dadurch wird erreicht, dass die Kovarianzen im gesamten Datensatz schrumpfen. Als ein vernünftiger Startwert für x wird 0,5 bis 1% der Beobachtungen genannt, da empfohlen wird, die künstlich hinzugefügten Fälle möglichst klein zu halten. [HONAKER et al. 2009] Im vorliegenden Datensatz mit 1.187 Beobachtungen wurden 5,9 künstliche Beobachtungen ($\approx$ 0,5%) gewählt.

Imputation durch Ziehen aus der empirischen Verteilung

Grundsätzliches zum Ziehen aus der empirischen Verteilung
Die zu imputierenden Werte werden bei dieser Methode gemäß der Randverteilung der beobachteten Daten gezogen. Dabei handelt es sich um eine Methode der Single Imputation. In Kapitel 4 zur Theorie der Imputationsmethoden wurde bereits als Problem der Single Imputation die Varianzunterschätzung angesprochen, da die Unsicherheit bei der Ersetzung fehlender Werte nicht adäquat berücksichtigt wird. Um dieses Problem zu umgehen, wurde in dieser Arbeit nicht jeder fehlende Wert durch einen einzelnen plausiblen Wert aus der empirischen Verteilung ersetzt, sondern zweimal aus der empirischen Verteilung gezogen. Somit kann jeder fehlende Wert durch zwei plausible Werte ersetzt werden und somit zwei vervollständigte Datensätze erzeugt werden.
Insgesamt wurden gemeinsam mit dem Vorgehen der multiplen Imputation mittels Amelia II fünf Datensätze erzeugt. Der Unsicherheit bei der Ersetzung fehlender Werte wurde durch dieses Vorgehen Rechnung getragen.

Behandlung von dichotomen Variablen
Handelte es sich im vorliegenden Datensatz um dichotome Variablen, so wurden die zu imputierenden Werte aus einer Bernoulli-Verteilung gezogen. Dabei wurde als Erfolgswahrscheinlichkeit die Auftretenswahrscheinlichkeit der Ausprägung, die mit 1 kodiert wurde, verwendet.

Behandlung von kategorialen Variablen
Für kategoriale Variablen wurde für jede Kategorie die Auftretenswahrscheinlichkeit angegeben. Eine der Kategorien wurde dann auf Basis dieser Wahrscheinlichkeiten gezogen und für die jeweiligen fehlenden Werte eingesetzt.

6.1.3 Zusammenfassung der Imputation der fehlenden Werte der Confounder

Durch das soeben beschriebene Vorgehen konnten fünf Datensätze erstellt werden, bei denen die potenziellen Confoundervariablen durch Imputation vervollständigt wurden. Die folgende Abbildung 6.2 gibt eine Übersicht über die Verteilung der Variablen, die zuvor fehlende Werte aufwiesen und nun durch Imputation vervollständigt wurden.

Variablen-beschreibung	Ausprägung	ursprüngl. Datensatz	Amelia 1. Datensatz	Amelia 2. Datensatz	Amelia 3. Datensatz	empir. Vert. 1. Datensatz	empir. Vert. 2. Datensatz
In Deutschland geboren	Nein(0)	57	57	57	57	57	57
	Ja (1)	1128	1130	1130	1130	1130	1130
	Fehlende Angabe(NA)	2	0	0	0	0	0
Atopie der Eltern	Nein(0)	639	646	650	646	646	642
	Ja (1)	536	541	537	541	541	545
	Fehlende Angabe(NA)	12	0	0	0	0	0
Geschwister	Nein(0)	189	193	193	193	194	196
	Ja (1)	965	994	994	994	993	991
	Fehlende Angabe(NA)	33	0	0	0	0	0
Kind gestillt	Nein(0)	185	199	196	197	193	193
	Ja (1)	956	988	991	990	994	994
	Fehlende Angabe(NA)	46	0	0	0	0	0
Passivrauch (ISAAC II)	Eltern Nichtraucher (0)	737	751	745	752	751	749
	Eltern Raucher (1)	332	339	344	337	338	341
	Eltern Ex-Raucher (2)	95	97	98	98	98	97
	NA	23	0	0	0	0	0
Passivrauch (SOLAR I)	Nein(0)	455	459	458	459	459	461
	Ja (1)	723	728	729	728	728	726
	Fehlende Angabe(NA)	9	0	0	0	0	0
Rauchverhalten (SOLAR I)	Nichtraucher (0)	839	843	845	844	843	841
	Raucher (1)	341	344	342	343	344	346
	Fehlende Angabe(NA)	7	0	0	0	0	0
Passivrauch (SOLAR II)	Nein(0)	495	497	496	499	502	504
	Ja (1)	677	690	691	688	685	683
	Fehlende Angabe(NA)	15	0	0	0	0	0
Rauchverhalten (SOLAR II)	Nichtraucher(0)	657	659	659	658	661	658
	Raucher (1)	425	427	427	429	427	429
	Ex-Raucher (2)	99	101	101	100	99	100
	Fehlende Angabe(NA)	6	0	0	0	0	0
Schulbildung (SOLAR II)	Niedrigere (0)	452	454	456	456	454	455
	Höhere (Abi/FH) (1)	731	733	731	731	733	732
	Fehlende Angabe(NA)	4	0	0	0	0	0
Sozioökonomischer Status	Niedrig (0)	479	486	490	486	485	488
	Hoch (1)	693	701	697	701	702	699
	Fehlende Angabe(NA)	15	0	0	0	0	0
Berufssituation (SOLAR I)	Hauptschüler (1)	8	8	8	8	8	8
	Realschüler (2)	137	138	137	137	138	137
	Gymnasiast (3)	617	617	617	617	617	617
	Schüler andere Schule (4)	103	103	103	104	103	104
	Azubi/Berufsschüler (5)	297	297	298	297	297	297
	Student (6)	3	3	3	3	3	3
	Angestellt (7)	6	6	6	6	6	6
	Arbeitslos&-suchend(9)	5	5	5	5	5	5
	Sonstiges (12)	10	10	10	10	10	10
	Fehlende Angabe(NA)	1	0	0	0	0	0
Berufssituation (SOLAR II)	Azubi/Berufsschüler (1)	198	198	198	198	199	199
	Student (2)	519	520	520	520	520	519
	Angestellt (3)	349	349	350	350	349	349
	Selbstständig (4)	12	12	12	12	12	12
	Arbeitslos&-suchend (5)	35	35	35	35	35	35
	aus gesundh. Gründen nicht arbeitend (6)	2	2	2	2	2	2
	Mutterschutz/Elternzeit/Beurlaubung (8)	10	11	10	10	10	10
	Sonstiges (9)	60	60	60	60	60	61
	Fehlende Angabe(NA)	2	0	0	0	0	0

Abbildung 6.2: Übersicht über die relevanten Variablen der vervollständigten Datensätze

6.2 Imputation der fehlenden Tätigkeitsangaben

Nachdem nun die potenziellen Confoundervariablen imputiert wurden, wurden in einem zweiten Schritt die Tätigkeitsangaben vervollständigt.

6.2.1 Übersicht über die fehlenden Tätigkeitsangaben

Im Rahmen der Tätigkeitsangaben wurde neben der konkreten Tätigkeit jeweils Anfangsmonat, Anfangsjahr, Endmonat, Endjahr und die Wochenstunden abgefragt. Bei 93 Probanden musste bei insgesamt 108 Tätigkeiten Imputationen vorgenommen werden. Dabei fehlten teilweise bei den Tätigkeiten mehrere Angaben gleichzeitig (vgl. Tabelle 5.8 zu den Fehlendmustern in Kapitel 5.4).
Die folgende Tabelle 6.6 gibt einen Überblick, wie viele Angaben pro Variable jeweils in den beiden Studien fehlten.

Variablen-beschreibung	**fehlende Werte SOLAR**	**fehlende Werte SOLAR II**
Anfangsmonat	32	32
Anfangsjahr	17	12
Endmonat	39	38
Endjahr	34	28
Wochenstunden	24	34

Tabelle 6.6: Übersicht über die fehlenden Tätigkeitsangaben

6.2.2 Vorgehen bei der Imputation der fehlenden Tätigkeitsangaben

Als einfachste Möglichkeit stand das Ziehen aus der empirischen Verteilung zur Auswahl, d.h. die Wochenstunden könnten direkt aus der empirischen Verteilung aller Wochenstundenangaben gezogen werden. Dabei würden aber mögliche Einflussgrößen (z.B. auf die Wochenstundenanzahl), wie beispielsweise Alter, Geschlecht, sozioökonomischer Status oder die Berufssituation (d.h. Schüler, Angestellt, Selbstständig, Arbeitslos etc.) nicht berücksichtigt. Es war zu vermuten, dass diese Parameter einen Einfluss auf die Tätigkeitsangaben haben. Daher wurde diese einfache Möglichkeit zunächst verworfen. Es wurde untersucht, ob sich die Zeitangaben mit Hilfe eines linearen Modells aus den Variablen Alter, Geschlecht, sozioökonomischer Status und Berufssituation (d.h. Schüler, Angestellt, Selbstständig, Arbeitslos etc.) vorhersagen lassen. Zunächst wurde ein Modell auf Basis aller Tätigkeitsangaben aus SOLAR und SOLAR II berechnet (Modell M0). Ein zweites Modell beschränkte sich nur auf die Tätigkeitsangaben aus SOLAR (Modell M1). In ein drittes Modell gingen nur die Tätigkeitsangaben aus SOLAR II ein (Modell M2).

Die Einflussgrößen in den jeweiligen Modellen sind der Tabelle 6.7 zu entnehmen. Mit + sind die Einflussgröße gekennzeichnet, die im jeweiligen Modell enthalten waren.

Einflussgröße	**Modell M0**	**Modell M1**	**Modell M2**
Geschlecht	+	+	+
Sozioökonomischer Status	+	+	+
Berufssituation in SOLAR	+	+	-
Berufssituation in SOLAR II	+	-	+
Alter (zum Zeitpunkt von SOLAR)	-	+	-
Alter (zum Zeitpunkt von SOLAR II)	+	-	+

Tabelle 6.7: Einflussgrößen der Modelle für die Imputation der Tätigkeitsangaben

Da das Bestimmtheitsmaß bei allen drei Modellen sehr geringe Werte aufwies (R^2 im Bereich [0,02 ; 0,10]), wurde der Ansatz verworfen, die fehlenden Tätigkeitsangaben mit Hilfe eines linearen Modells vorherzusagen.
Es wurde stattdessen entschieden, die Imputation der Tätigkeitsangaben für SOLAR und SOLAR II jeweils getrennt voneinander durchzuführen. Somit galten als Basis für die Imputation in der jeweiligen Studie ausschließlich die Angaben, die auch im Rahmen der jeweiligen Untersuchung gemacht wurden.
Weiterhin wurde entschieden, die fehlenden Zeitangaben und Wochenstunden aus der empirischen Verteilung zu ziehen, dabei allerdings zusätzlich nach bestimmten Variablen zu schichten. Auf welche Variablen bei der Ziehung jeweils bedingt wurde, wird im Folgenden ausführlich dargestellt.

Imputation der Zeitangaben

Zur Imputation der Zeitangaben (Anfangsmonat, Anfangsjahr, Endmonat, Endjahr) wurde das Ziehen aus der empirischen Verteilung gewählt. Allerdings sollte das Ziehen auf bestimmte Variablen bedingt werden. Um zu entscheiden, auf welche Variablen bedingt werden sollte, wurden erneut die drei zuvor vorgestellten Modelle betrachtet (M0 auf Basis aller Tätigkeitsangaben aus SOLAR und SOLAR II, M1 auf Basis der Tätigkeitsangaben aus SOLAR, M2 auf Basis der Tätigkeitsangaben aus SOLAR II). Aus der Liste der Kovariablen der linearen Modellen wurden die Variablen ausgewählt, die zumindest bei einem der drei Modelle ein Signifikanzniveau von mindestens 0.05 aufwiesen. Bei Faktorvariablen musste mindestens eine Faktorstufe auf diesem Niveau statistisch signifikant sein, um in die Auswahl aufgenommen zu werden. Die Tabelle 6.8 gibt eine Übersicht, welche Einflussgrößen bei welchen Zeitangaben (Anfangsmonat, Anfangsjahr, Endmonat, Endjahr) bei mindestens einem der drei Modelle dieses statistische Signifikanzniveau erreichten.

Einflussgröße	Anfangsmonat	Anfangsjahr	Endmonat	Endjahr
Geschlecht	>0,05	>0,05	>0,05	>0,05
Sozioökonomischer Status	>0,05	≤0,05	>0,05	≤0,05
Berufssituation (SOLAR)	>0,05	>0,05	>0,05	>0,05
Berufssituation (SOLAR II)	≤0,05	≤0,05	≤0,05	≤0,05
Alter (SOLAR)	>0,05	>0,05	>0,05	>0,05
Alter (SOLAR II)	>0,05	>0,05	>0,05	>0,05

Tabelle 6.8: Übersicht über das statistische Signifikanzniveau der Einflussgrößen

Auf Basis dieser Ergebnisse wurde entschieden, das Ziehen aus der empirischen Verteilung (innerhalb der jeweiligen Studie) für alle Zeitangaben einheitlich zu machen. Somit wurde das Ziehen in SOLAR auf den sozioökonomischen Status bedingt, d.h. die Imputation erfolgt durch Ziehen aus der empirisichen Verteilung geschichtet nach dem sozioökonomischen Status. Die empirischen Verteilungen wurde auf Basis aller Probanden, die bei der jeweiligen Variable Angaben machten, ermittelt. Da die Berufsituation aus SOLAR II zum Zeitpunkt von SOLAR noch nicht bekannt war, wurde diese Information für die Imputation der Zeitangaben von SOLAR nicht verwendet. Die Grafik 6.3 soll das Vorgehen am Beispiel der Imputation des Anfangsjahr veranschaulichen. Die Balkendiagramme veranschaulichen dabei die empirische Verteilung des Anfangsjahrs (in SOLAR), geschichtet nach dem sozioökonomischen Status (auf Basis aller Probanden, die in SOLAR Angaben zum Anfangsjahr machten).

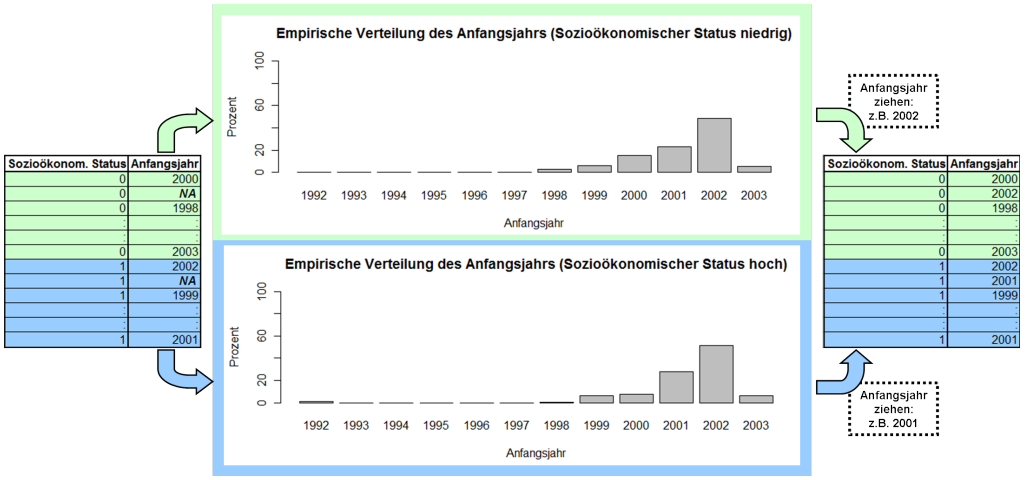

Abbildung 6.3: Imputation des Anfangsjahrs durch Ziehen aus der empirischen Verteilung geschichtet nach dem sozioökonomischen Status

Für die Imputation der Zeitangaben in SOLAR II wurde aus der empirischen Verteilung geschichtet nach sozioökonomischem Status und der Berufssituation in SOLAR II gezogen.

Bei der Imputation der Zeitangaben mussten zusätzlich einige Bedingungen erfüllt sein, damit durch die Imputation keine unplausiblen Zeitangaben erzeugt wurden.
Zum einen musste das imputierte Endjahr nach dem Anfangsjahr liegen. Um dies zu gewährleisten, verwendete man bei der Imputation des Endjahres die Methode "rejection sampling". Bei dieser Methode wurde zunächst ein Endjahr gezogen, dann wurde die Bedingung überprüft (Endjahr $\geq$ Anfangsjahr). War diese Bedingung erfüllt, so wurde das gezogene Endjahr an die Stelle des fehlenden Jahres eingesetzt. War die Bedingung nicht erfüllt, so wurde so lange aus der empirischen Verteilung gezogen, bis die Bedingung erfüllt war. Bei der Imputation des Endjahres musste weiterhin beachtet werden, dass bei einem Anfangsmonat, das nach dem Endmonat lag, das Endjahr nicht im gleichen Jahr wie das Anfangsjahr sein durfte.
Konnte beim Endjahr kein passender Wert gezogen werden, der die Bedingungen erfüllte, so wurde die Ziehung ausschließlich auf den sozioökonomischen Status bedingt.
Bei der Imputation des Anfangs- und des Endmonats musste auch die Methode "rejection sampling" angewandt werden, da bei gleichem Anfangs- und Endjahr die Bedingung Anfangsmonat $\leq$ Endmonat gelten musste.

Imputation der Wochenstunden

Die Imputation der Wochenstunden erfolgte ebenfalls durch Ziehen aus der empirischen Verteilung. Dabei wurde auf den ISCO-Code bedingt, da in den Tätigkeitsgruppen die Wochenstundenanzahl unterschiedlich sein konnte. Beispielsweise unterschieden sich die durchschnittlichen Wochenstunden im Gastronomiegewerbe deutlich von den Wochenstunden in anderen Branchen.
Weiterhin sollte auch ein möglicher Geschlechtseffekt berücksichtigt werden, indem zusätzlich bei der Ziehung auf das Geschlecht bedingt wurde. Ein solcher möglicher Geschlechtsunterschied in Bezug auf die Wochenstundenanzahl ist beim Vergleich der Mittelwerte (auf Basis der vollständigen Angaben) zu vermuten: Frauen arbeiteten durchschnittlich 26, Männer 30 Stunden pro Woche.

Bei der Imputation wurde dann wie folgt vorgegangen:
Lagen Fälle mit vorhandenen Wochenstundenangaben und gleichem ISCO-Code und Geschlecht vor (d.h. n>1), so wurden die Wochenstunden auf Basis der empirischen Verteilung bedingt auf den entsprechenden ISCO-Code und Geschlecht gezogen.
War dies nicht der Fall, so wurde bei der Ziehung der Wochenstunden ausschließlich auf das Geschlecht bedingt.

Zusammenfassung der Imputation der fehlenden Tätigkeitsangaben

Zur Imputation der fehlenden Tätigkeitsangaben wurde eine Funktion geschrieben, die obiges Vorgehen für die Imputation der fehlenden Zeitangaben und der Wochenstunden ausführte.
Dieser Funktion kann ein Datensatz mit fehlenden Tätigkeitsangaben und ein Startwert (zur Reproduzierbarkeit der Imputation) übergeben werden. Die Funktion liefert einen vervollständigten Datensatz zurück, mit dem weiterführende Analysen durchgeführt werden können.
Zu beachten ist dabei, dass der Datensatz, der der Funktion übergeben wird, keine fehlenden Angaben in den Confoundern mehr enthalten darf, d.h. diese Angaben müssen zuvor imputiert werden, da beispielsweise Angaben zum sozioökonomischen Status oder zum Geschlecht für die Imputation der Tätigkeitsangaben benötigt werden.

6.3 Zusammenfassung der Imputationsschritte

Die Imputation der fehlenden Werte im vorliegenden Datensatz wurde in zwei Schritten vorgenommen.

1. Zunächst wurden die fehlenden Angaben der Confounder imputiert. Dabei wurden
   a) drei Datensätze durch multiple Imputation mit Hilfe des R-Packages Amelia-II erstellt und
   b) zwei Datensätze mittels Ziehen aus den empirischen Verteilungen.
2. Im zweiten Schritt wurden die fehlenden Tätigkeitsangaben durch Ziehen aus den bedingten empirischen Verteilungen imputiert. Die bereits vervollständigten Confoundervariablen wurden als Basis für die Imputation der Tätigkeitsangaben verwendet, und somit erfolgte auf jedem dieser fünf Datensätze die Imputation der Tätigkeitsangaben.

Am Ende dieser Imputationsschritte lagen fünf komplett vervollständigte Datensätze vor (Abbildung 6.4).

Abbildung 6.4: Zusammenfassung der Imputationsschritte

7 Berechnung der Exposition

7.1 Grundlegendes Vorgehen

Im Rahmen des Fragebogens konnte jeder Proband in SOLAR und SOLAR II jeweils bis zu fünf Tätigkeitsangaben machen. Aufgrund der Angaben zur Tätigkeit konnte mit Hilfe der Berufssystematik ISCO-88 (vgl. Kapitel 3.1) für jede Tätigkeit ein vierstelliger Code vergeben werden. Mit Hilfe dieses Codes konnte durch die Job-Exposure-Matrix (vgl. Kapitel 3.2) für jede Tätigkeit festgelegt werden, in welcher der 22 Expositionsgruppen eine Belastung vorlag. Diese 22 Expositionsgruppen wurden zu den folgenden fünf Oberkategorien zusammengefasst: HMW, LMW, Mixed, Irrpeaks, Low Risk. Eine Oberkategorie wurde als exponiert kodiert (=1), wenn in mindestens einer der jeweiligen Unterkategorien eine Exposition vorlag (vgl. Abbildung 3.1 in Kapitel 3.2). Für die weiteren Berechnungen der Exposition wurden ausschließlich diese Oberkategorien betrachtet.

7.1.1 Erstellung der Basismatrix

Um die Erstellung der Datenmatrix zur Expositionsberechnung beispielhaft darzustellen, wurde ein Fall mit drei Tätigkeitsangaben ausgewählt. Anhand dieses Falls sollen nun die Stufen bei der Erstellung der Matrix erläutert werden. Als Basis lagen die Angaben aus dem Fragebogen vor.
Die Probandin übte während SOLAR und SOLAR II drei Tätigkeiten aus und zwar Verkäuferin im Sonnenstudio (ISCO 5141), Reinigungskraft (ISCO 9132) und Bürotätigkeiten (ISCO 4190) (vgl. Abbildung 7.1).

knr	Anf.monat	Anf.jahr	Endmonat	Endjahr	Wochenstd	ISCO	HMW	LMW	MIXED	IRRPEAKS	LOWRISK
S58154288	9	2001	1	2005	10	5141	0	1	0	0	0
S58154288	2	2002	3	2003	9	9132	0	1	0	0	1
S58154288	1	2005	1	2008	8	4190	0	0	0	0	0

Abkürzungen: knr = Kohortennummer; Anf.monat = Anfangsmonat; Anf.jahr = Anfangsjahr; Wochenstd = Wochenstunden

Abbildung 7.1: Ausschnitt aus dem Datensatz: Beispielfall mit drei Tätigkeitsangaben

Für die Analyse der Tätigkeitsangaben wurde eine Matrix erstellt, die für alle späteren Berechnungen als Basis dienen soll. Für den Beispielfall sah diese Matrix wie folgt aus (vgl. 7.2).

knr	NR_BERUF	JAHR	Anf.monat	Endmonat	Wochenstd	ISCO	gearb. Monate	HMW_jahr	LMW_jahr	MIXED_jahr	IRRPEAKS_jahr	LOWRISK_jahr
S58154288	1	2000	0	0	0	5141	0	0	0	0	0	0
S58154288	1	2001	9	12	10	5141	4	0	170	0	0	0
S58154288	1	2002	1	12	10	5141	12	0	510	0	0	0
S58154288	1	2003	1	12	10	5141	12	0	510	0	0	0
S58154288	1	2004	1	12	10	5141	12	0	510	0	0	0
S58154288	1	2005	1	1	10	5141	1	0	42.5	0	0	0
S58154288	1	2006	0	0	0	5141	0	0	0	0	0	0
S58154288	1	2007	0	0	0	5141	0	0	0	0	0	0
S58154288	1	2008	0	0	0	5141	0	0	0	0	0	0
S58154288	1	2009	0	0	0	5141	0	0	0	0	0	0
S58154288	2	2000	0	0	0	9132	0	0	0	0	0	0
S58154288	2	2001	0	0	0	9132	0	0	0	0	0	0
S58154288	2	2002	2	12	9	9132	11	0	420.75	0	0	420.75
S58154288	2	2003	1	3	9	9132	3	0	114.75	0	0	114.75
S58154288	2	2004	0	0	0	9132	0	0	0	0	0	0
S58154288	2	2005	0	0	0	9132	0	0	0	0	0	0
S58154288	2	2006	0	0	0	9132	0	0	0	0	0	0
S58154288	2	2007	0	0	0	9132	0	0	0	0	0	0
S58154288	2	2008	0	0	0	9132	0	0	0	0	0	0
S58154288	2	2009	0	0	0	9132	0	0	0	0	0	0
S58154288	3	2000	0	0	0	4190	0	0	0	0	0	0
S58154288	3	2001	0	0	0	4190	0	0	0	0	0	0
S58154288	3	2002	0	0	0	4190	0	0	0	0	0	0
S58154288	3	2003	0	0	0	4190	0	0	0	0	0	0
S58154288	3	2004	0	0	0	4190	0	0	0	0	0	0
S58154288	3	2005	1	12	8	4190	12	0	0	0	0	0
S58154288	3	2006	1	12	8	4190	12	0	0	0	0	0
S58154288	3	2007	1	12	8	4190	12	0	0	0	0	0
S58154288	3	2008	1	1	8	4190	1	0	0	0	0	0
S58154288	3	2009	0	0	0	4190	0	0	0	0	0	0

Abkürzungen: knr = Kohortennummer; NR_BERUF = Nummer der Tätigkeit; Anf.monat = Anfangsmonat; Wochenstd = Wochenstunden; gearb. Monate = gearbeitete Monate; HMW_jahr = HMW-Exposition im jeweiligen Jahr; LMW_jahr = LMW-Exposition im jeweiligen Jahr; MIXED_jahr = MIXED-Exposition im jeweiligen Jahr; IRRPEAKS_jahr = IRRPEAKS-Exposition im jeweiligen Jahr; LOWRISK_jahr = LOWRISK-Exposition im jeweiligen Jahr

Abbildung 7.2: Ausschnitt aus der Basismatrix zur Expositionsberechnung: Beispielfall mit drei Tätigkeitsangaben

In dieser Basismatrix wurden zunächst die Tätigkeiten aufgesplittet, in einem zweiten Schritt wurden die Jahre aufgesplittet. Somit gab es für jeden Probanden jeweils für jede der möglichen zehn Tätigkeiten einen Block von zehn Zeilen, der die Jahre 2000 bis 2009 umfasste (d.h. insgesamt 100 Zeilen pro Proband). Die Jahre 2000 bis 2009 wurden gewählt, da die meisten Probanden nicht vor 2000 arbeiteten. Nur einige wenige Probanden (25 Fälle) arbeiteten bereits vor dem Jahr 2000. Für diese Fälle enthielt jeder Block pro Tätigkeit 18 Zeilen, der die Jahre 1992 bis 2009 umfasste (d.h. insgesamt 180 Zeilen pro Proband). Dann wurden aus den ursprünglichen Tätigkeitsangaben die Zeitangaben, die Wochenstunden und der ISCO-Code in die passende Jahreszeile dieser Matrix übertragen. In jeder Jahreszeile wurden dann die gearbeiteten Monate berechnet.
Aus den gearbeiteten Monaten und den Wochenstunden konnten die gearbeiteten Stunden im jeweiligen Jahr folgendermaßen berechnet werden:

Berechnung der gearbeiteten Stunden im jeweiligen Jahr
gearbeitete Stunden pro Jahr = Wochenstunden × 4,25 × gearbeitete Monate
(Der Faktor 4,25 steht für 4,25 Wochen pro Monat)

Wie bereits erwähnt, gab es im Datensatz für jede der fünf Expositionsobergruppen (HMW, LMW, Mixed, Irrpeaks, Low Risk) eine Variable mit den Einträgen 0 (nicht exponiert) oder 1 (exponiert). Multipliziert man nun jede einzelne Variable der Expositionsgruppen mit den gearbeiteten Stunden in dem entsprechenden Jahr, so erhält man die Exposition, die bei einem Probanden bei einer bestimmten Tätigkeit und dem entsprechendem Jahr vorlag (HMW_jahr, LMW_jahr, MIXED_jahr, IRRPEAKS_jahr, LOWRISK_jahr).

Berechnung der HMW-Belastung in Stunden im jeweiligen Jahr
HMW_jahr = gearbeitete Stunden pro Jahr × HMW

7.1.2 Zusammenfassung der Erstellung der Basismatrix

Die Grafik 7.3 fasst die Erstellung der Basismatrix zusammen, die zur Berechnung der Expositionen benötigt wurde.

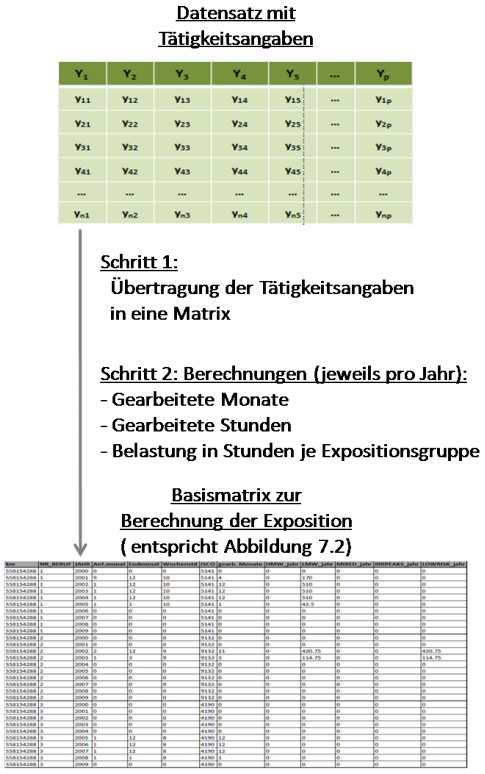

Abbildung 7.3: Übersicht über die Erstellung der Basismatrix zur Berechnung der Expositionen

7.1.3 Extraktion diverser Expositionen aus der Basismatrix

Aus der so entstandenen Basismatrix sollten für spätere Analysen pro Proband folgende Expositionen extrahiert werden:

- **binäre Exposition über alle Tätigkeiten und Jahre hinweg**
  Die binäre Exposition gibt wieder, ob der Proband irgendwann im Laufe seines bisherigen Arbeitslebens bei irgendeiner Tätigkeit einer beruflichen Exposition der jeweiligen Expositionsgruppe ausgesetzt war (=1) oder nicht (=0).
  Probanden, die nie gearbeitet haben, erhielten bei allen Expositionsgruppen den Status nicht exponiert (=0).

- **kumulierte Exposition über alle Tätigkeiten und Jahre hinweg**
  Für die kumulierte Exposition wurden pro Proband über alle Tätigkeiten, sowohl aus SOLAR als auch aus SOLAR II, und über alle Jahre hinweg die jeweiligen Einträge pro Variablen HMW_jahr, LMW_jahr, MIXED_jahr, IRRPEAKS_jahr und LOWRISK_jahr aufsummiert. Jeder Proband erhielt somit fünf Variablen, die pro Expositionsgruppe die Belastung in Stunden wiedergab, die der Proband seit Beginn seines Arbeitslebens über alle Tätigkeiten hinweg ausgesetzt war.

- **binäre Exposition während des ersten Tätigkeitsjahrs**
  Die binäre Exposition gibt an, ob der jeweilige Proband während seines ersten Tätigkeitsjahres der jeweiligen Expositionsgruppe ausgesetzt war.

- **kumulierte Exposition während des ersten Tätigkeitsjahrs**
  In die kumulierte Exposition des ersten Tätigkeitsjahrs geht der Zeitraum innerhalb der 12 Monate ab Beginn der ersten Tätigkeit ein. Somit wurde zunächst der Anfangszeitpunkt der ersten Tätigkeit und der hier relevante 12-Monatszeitraum ermittelt. Für alle Tätigkeiten musste weiterhin ermittelt werden, ob diese (teilweise) innerhalb dieses 12-Monatszeitraums lagen. War dies der Fall, so wurde die Anzahl der Monate ermittelt, die sich innerhalb dieses Zeitraums befanden. Für die Berechnung der Exposition wurden nun die entsprechenden Expositionskategorien mit der Anzahl der Monate innerhalb des Zeitraums multipliziert und dann über alle Tätigkeiten und Jahre hinweg aufsummiert. Somit wurden nur Expositionen berücksichtigt, die innerhalb des ersten Tätigkeitsjahres lagen. Auch wenn mehrere Tätigkeiten innerhalb des ersten Jahres ausgeübt wurden, wurde dies berücksichtigt, da über alle Tätigkeiten hinweg summiert wurde.
  Hatte ein Proband nur einen Teil des ersten Jahres gearbeitet (*z.B. 6 Monate*), so wurden ausschließlich die Expositionen in diesem Zeitraum aufsummiert.
  Dieser Zeitraum wurde ausgewählt, da gezeigt wurde, dass eine Exposition am Anfang des Berufslebens das Entstehen von berufsbedingtem Asthma tendenziell stärker beeinflusst, als eine spätere Exposition. [BENKE et al. 2008]

- **binäre Exposition während der ersten ausgeübten Tätigkeit**
  Die binäre Exposition gibt an, ob der Proband während seiner ersten ausgeübten Tätigkeit einer Exposition der jeweiligen Kategorie ausgesetzt war.

- **kumulierte Exposition während der ersten ausgeübten Tätigkeit**
  Zur Berechnung der kumulierten Exposition der ersten ausgeübten Tätigkeit wur-

den ausschließlich die Expositionsstunden über alle Jahre hinweg aufsummiert, die während der ersten Tätigkeit auftraten.
Dieser Zeitraum wurde betrachtet, da in Bezug auf Atemwegserkrankungen gezeigt werden konnte, dass sich die Exposition im ersten ausgeübte Job ersatzweise an Stelle der Exposition aller ausgeübten Tätigkeiten eignen kann. [BENKE et al. 2008]

Für den vorliegenden Beispielfall erhielt man folgende berechnete Expositionen (vgl. Abbildung 7.4).

ÜBER ALLE JAHRE UND TÄTIGKEITEN HINWEG					
Kohortennummer	HMW-Exposition binär	LMW-Exposition binär	MIXED-Exposition binär	IRRPEAKS-Exposition binär	LOWRISK-Exposition binär
S58154288	0	1	0	0	1
	HMW-Exposition kumuliert	LMW-Exposition kumuliert	MIXED-Exposition kumuliert	IRRPEAKS-Exposition kumuliert	LOWRISK-Exposition kumuliert
	0	2278	0	0	535.5

1. TÄTIGKEIT					
Kohortennummer	HMW-Exposition binär	LMW-Exposition binär	MIXED-Exposition binär	IRRPEAKS-Exposition binär	LOWRISK-Exposition binär
S58154288	0	1	0	0	0
	HMW-Exposition kumuliert	LMW-Exposition kumuliert	MIXED-Exposition kumuliert	IRRPEAKS-Exposition kumuliert	LOWRISK-Exposition kumuliert
	0	1742.5	0	0	0

1.TÄTIGKEITSJAHR					
Kohortennummer	HMW-Exposition binär	LMW-Exposition binär	MIXED-Exposition binär	IRRPEAKS-Exposition binär	LOWRISK-Exposition binär
S58154288	0	1	0	0	1
	HMW-Exposition kumuliert	LMW-Exposition kumuliert	MIXED-Exposition kumuliert	IRRPEAKS-Exposition kumuliert	LOWRISK-Exposition kumuliert
	0	777.75	0	0	267.75

Abbildung 7.4: Ausschnitt aus den berechneten Expositionen: Beispielfall mit drei Tätigkeitsangaben

Die Berechnung der kumulierten Exposition während des ersten Tätigkeitsjahres soll nun anhand dieses Beispiels ausführlich dargestellt werden.

Die Probandin übte als erste Tätigkeit Verkäuferin in einem Sonnenstudio aus, die sie im September 2001 begann. Somit läuft der 12-Monatszeitraum, der hier eingehen sollte, bis August 2002.
Zunächst wurde diese erste Tätigkeit betrachtet. Im Jahr 2001 wurden 4 Monate gearbeitet (von September bis Dezember), diese lagen komplett im 12-Monatszeitraum. Somit betrug die relevante Exposition z.B. für LMW in dieser Zeile 170 Stunden
*(170 Stunden = 10 Wochenstunden * 4,25 Wochen * 4 Monate * 1 (exponiert)).*
Im Jahr 2002 war die Probandin von Januar bis Dezember Verkäuferin. In den Zeitraum für das erste Tätigkeitsjahr fielen allerdings nur acht Monate (von Januar bis August). Hier betrug die relevante Exposition z.B. für LMW in dieser Zeile 340 Stunden
*(340 Stunden = 10 Wochenstunden * 4,25 Wochen * 8 Monate * 1 (exponiert)).*
Dann wurde die Tätigkeit als Reinigungskraft betrachtet. Von der Tätigkeit als Reinigungskraft fielen noch 7 Monate in den 12-Monatszeitraum (Februar bis August). Somit erhielt man als relevante Exposition z.B. für LMW 267,75 Stunden
*(267,75 Stunden = 9 Wochenstunden * 4,25 Wochen * 7 Monate * 1 (exponiert)).*
Die Bürotätigkeit lag komplett außerhalb des Zeitraums und musste für diese Exposition nicht mehr beachtet werden.
Summiert man also abschließend alle relevanten Expositionen auf, z.B. für LMW, so er-

hält man für diese Probandin eine Belastung von 777,75 Stunden innerhalb ihres ersten Tätigkeitsjahres
(777,75 Stunden = 170 Stunden + 340 Stunden + 267,75 Stunden).

Alle eben ausführlich beschriebenen Berechnungen wurden nur für Tätigkeiten durchgeführt, die mindestens für acht Wochenstunden ausgeführt wurden. Alle Tätigkeiten mit weniger als acht Wochenstunden wurden in den Expositionsobergruppen jeweils auf 0 (nicht exponiert) gesetzt. Als Sonderfall wurden auch die Tätigkeiten behandelt, bei denen man konservativ vorging und die Exposition in allen Spalten auf 0 setzte. Dies waren die Fälle, die zuvor die ISCO-Codes 94, 95, 97 und 98 erhielten. Bei all diesen Fällen lag somit keine Exposition vor.

7.1.4 Vereinfachte Berechnung der Exposition

Zunächst wurde im Rahmen der Themenstellung dieser Bachelorarbeit gefordert, die Exposition auf Basis der soeben beschriebenen Basismatrix, die für die Jahre 2000-2009 (bzw. 1992 bis 2009) jeweils eine Zeile enthielt, zu berechnen. Diese Matrix und die darauf basierenden Berechnungen wurden dann in R programmiert. Wie zuvor erläutert, lagen in dieser Matrix pro Proband 100 (bzw. 180 Zeilen) vor. Daher nahm die Berechnung dieser Matrix sehr viel Zeit in Anspruch (mehr als 24 Stunden pro Durchlauf). Aus diesem Grund wurde eine vereinfachte Berechnung der Exposition entworfen. Diese Berechnung konnte direkt auf Basis der ursprünglichen Tätigkeitsangaben vorgenommen werden. Eine Aufsplittung auf die einzelnen Jahre war nicht mehr nötig. Das Vorgehen zur Berechnung der interessierenden Expositionen (kumulierte Exposition, Exposition im ersten Tätigkeitsjahr und Exposition der ersten Tätigkeit) erfolgte analog zum oben beschriebenen Vorgehen. Die daraus resultierenden berechneten Expositionen entsprachen den Expositionen, die auf Basis der Matrix berechnet wurden. Das Ergebnis war somit identisch. Ein großer Vorteil war allerdings die Dauer der Berechnungen, die bei dem vereinfachten Vorgehen sehr kurz war (ca. 10 Minuten) und somit problemlos mehrmals (auf verschiedenen Datensätzen) angewandt werden konnte.
Ist man allerdings daran interessiert, die Expositionen ausschließlich für bestimmte Jahre zu betrachten, beispielsweise getrennt für SOLAR (bis 2003) und für SOLAR II (ab 2003), so ist die Anwendung der komplexen Matrix vorteilhafter, da hier die Information bereits jahreweise aufgesplittet wurde.

7.2 Berechnung der Exposition auf Basis der Probanden mit vollständigen Tätigkeitsangaben

Die Expositionen wurden zunächst auf den vollständigen Tätigkeitsangaben berechnet. Von den 1.094 Probanden, die vollständige Tätigkeitsangaben machten, konnten folgende interessierende Expositionen berechnet werden, die später in die logistischen Regressionsmodelle als mögliche Einflussgrößen eingingen:

- kumulierte Exposition über alle Tätigkeiten und Jahre
- Exposition der ersten Tätigkeit
- Exposition im ersten Tätigkeitsjahr

Zusätzlich zu den kumulierten Expositionen, die die Belastungen im jeweiligen Zeitraum in Stunden messen, wurden auch binäre Variablen angelegt, die angaben, ob überhaupt eine Exposition in der jeweiligen Kategorie vorlag. Auch diese binären Variablen konnten als mögliche Einflussgrößen in den Logitmodellen dienen.
Die folgende Grafik 7.5 gibt eine Übersicht über die Anzahl der Personen, die exponiert waren.

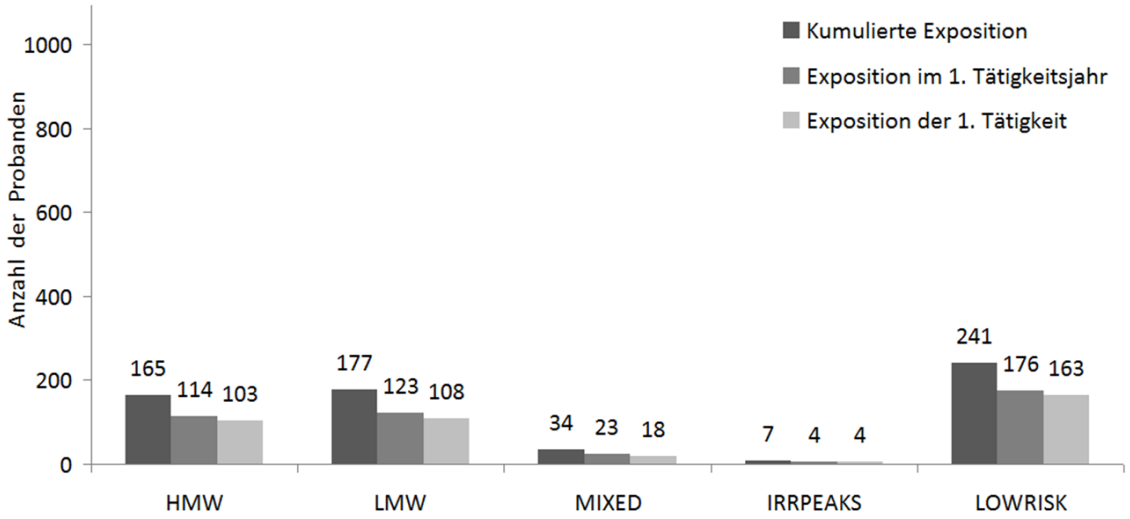

Abbildung 7.5: Übersicht über die Anzahl der exponierten Personen

Die kumulierten Expositionen werden im folgenden Abschnitt anhand von Tabellen und Boxplots analysiert.

7.2.1 Exposition über alle Tätigkeiten und Jahre

Die Tabelle 7.1 zeigt das Auftreten der Expositionen über alle Tätigkeiten und Jahre hinweg, sowie Median, Minimum und Maximum der berechneten Expositionen. Am häufigsten tratt die Exposition LOWRISK auf, gefolgt von LMW und HMW. Die Expositionen MIXED und vor allem IRRPEAKS traten sehr selten auf. Die längste Exposition beim Vergleich der Mediane lag bei IRRPEAKS und LMW vor, gefolgt von HMW. Die Mediane der MIXED- und LOWRISK-Exposition lagen deutlich darunter.
426 Personen, die keiner Exposition ausgesetzt waren, hatten nie gearbeitet. Die anderen Probanden, die keiner Exposition ausgesetzt waren, hatten zwar gearbeitet, waren allerdings in dieser Tätigkeit nicht exponiert.

kumulierte Exposition	Exposition vorhanden	Keine Exposition vorhanden
HMW	Anzahl Fälle: 165 Median: 1.360 Stunden Range in Stunden: [68,12.240]	Anzahl Fälle ohne Exposition: 929 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 503
LMW	Anzahl Fälle: 177 Median: 1.530 Stunden Range in Stunden: [34,13.090]	Anzahl Fälle ohne Exposition: 917 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 491
MIXED	Anzahl Fälle: 34 Median: 669 Stunden Range in Stunden: [68,12.240]	Anzahl Fälle ohne Exposition: 1.060 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 634
IRRPEAKS	Anzahl Fälle: 7 Median: 1.530 Stunden Range in Stunden: [170,7.990]	Anzahl Fälle ohne Exposition: 1.087 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 661
LOWRISK	Anzahl Fälle: 241 Median: 850 Stunden Range in Stunden: [34,15.980]	Anzahl Fälle ohne Exposition: 853 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 427

Tabelle 7.1: Übersicht über die Expositionen über alle Tätigkeiten und Jahre hinweg

Boxplots der Exposition über alle Tätigkeiten und Jahre

In Abbildung 7.6 werden die Boxplots für die kumulierten Expositionen dargestellt.
Zunächst erkennt man, dass nur sehr wenige Probanden einer IRRPEAKS-Belastung ausgesetzt waren. Gleichzeitig waren diese Personen mit am längsten exponiert (gemeinsam mit den Personen, die einer LMW-Exposition ausgesetzt waren).
Am kürzesten exponiert waren die Probanden, die einer MIXED-Exposition ausgesetzt waren. Generell ist zu sagen, dass der Interquartilsabstand der Expositionen über alle fünf Gruppen relativ breit ist, am stärksten sticht hier die Kategorie IRRPEAKS heraus.
Ausreißer, mit vergleichsweise hohen Expositionsdauern, traten bei allen Expositionsgruppen, bis auf IRRPEAKS, auf.

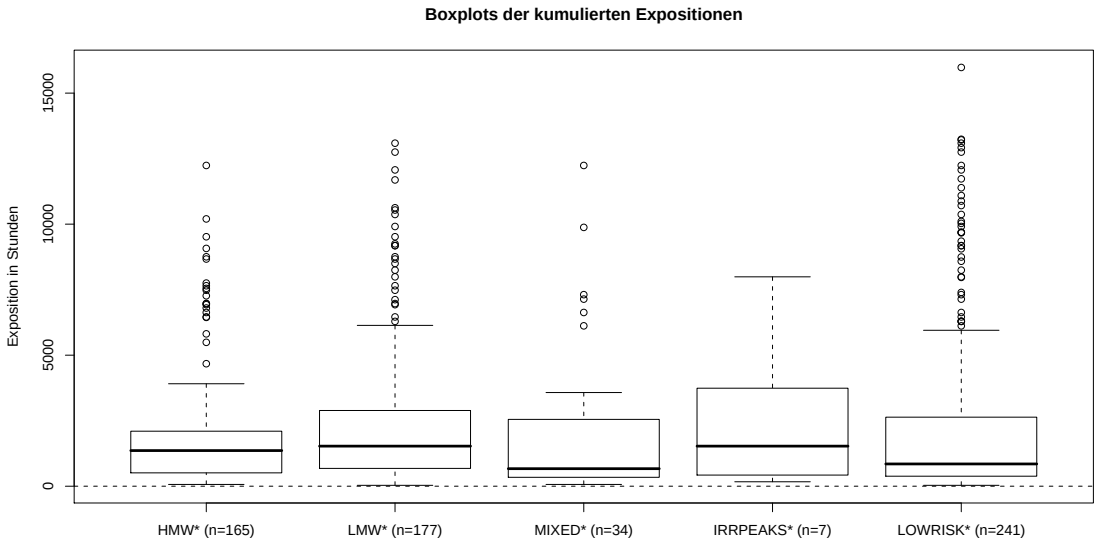

Abbildung 7.6: Boxplots der kumulierten Expositionen auf Basis der vollständigen Tätigkeitsangaben

7.2.2 Exposition im 1. Tätigkeitsjahr

Die Expositionen im ersten Tätigkeitsjahr beschreibt in analoger Weise Tabelle 7.2.
Das Auftreten der Expositionen im ersten Tätigkeitsjahr war sehr ähnlich zur gesamten Exposition. Allerdings waren nur rund 2/3 der Personen, die während ihres gesamten bisherigen Arbeitslebens exponiert waren, auch bereits im 1. Tätigkeitsjahr einer Exposition ausgesetzt. Am häufigsten traten während des 1. Tätigkeitsjahrs ebenfalls LOWRISK, LMW und HMW-Exposition auf. Relativ selten waren die Probanden gegenüber MIXED und IRRPEAKS exponiert. Im Hinblick auf die LOWRISK-Exposition gab es im 1. Tätigkeitsjahr insgesamt deutlich mehr Nichtexponierte als dies bei den anderen Expositionsgruppen (im Vergleich zur Exposition über alle Tätigkeiten und Jahre hinweg) der Fall war.

Exposition im 1. Tätigkeitsjahr	Exposition vorhanden	Keine Exposition vorhanden
HMW	Anzahl Fälle: 114 Median: 697 Stunden Range in Stunden: [68,2.295]	Anzahl Fälle ohne Exposition: 980 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 554
LMW	Anzahl Fälle: 123 Median: 850 Stunden Range in Stunden: [34,3.672]	Anzahl Fälle ohne Exposition: 971 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 545
MIXED	Anzahl Fälle: 23 Median: 574 Stunden Range in Stunden: [68,2.550]	Anzahl Fälle ohne Exposition: 1.071 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 645
IRRPEAKS	Anzahl Fälle: 4 Median: 1.071 Stunden Range in Stunden: [170,2.040]	Anzahl Fälle ohne Exposition: 1.090 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 664
LOWRISK	Anzahl Fälle: 176 Median: 510 Stunden Range in Stunden: [34,3.672]	Anzahl Fälle ohne Exposition: 918 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 492

Tabelle 7.2: Übersicht über die Expositionen im ersten Tätigkeitsjahr

Boxplots der Exposition im 1. Tätigkeitsjahr

Abbildung 7.7 stellt die Boxplots für die Expositionen im ersten Tätigkeitsjahr dar. Zunächst fällt auf, dass die Interquartilsabstände der Expositionen deutlich schmaler sind im Vergleich zur Exposition über alle Jahre und Tätigkeiten hinweg. Die Personen, die einer IRRPEAKS- oder einer LMW-Exposition ausgesetzt waren, waren am längsten exponiert. Erkennbar weniger lang war die Dauer, die die Probanden einer HMW-, MIXED- oder LOWRISK-Exposition ausgesetzt waren (bei Betrachtung des Medians).
Einige wenige Ausreißer, mit vergleichsweise hohen Expositionsdauern, lagen ausschließlich bei der LOWRISK-Exposition vor.

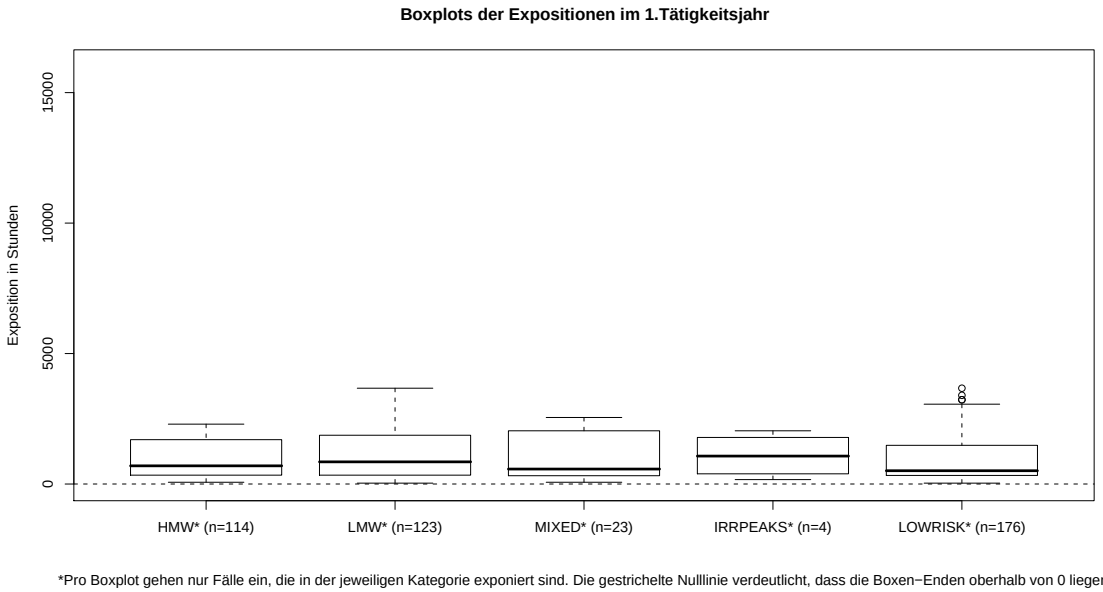

Abbildung 7.7: Boxplots der Expositionen im ersten Tätigkeitsjahr auf Basis der vollständigen Tätigkeitsangaben

7.2.3 Exposition der ersten Tätigkeit

Die Tabelle 7.3 zeigt den Median, das Minimum und das Maximum der berechneten Expositionen, die während der ersten Tätigkeit auftraten.
Von den Personen, die während ihres gesamten bisherigen Arbeitslebens einer Exposition ausgesetzt waren, waren etwa rund 60% während ihrer ersten Tätigkeit bereits exponiert. Am häufigsten trat erneut eine LOWRISK-Exposition auf, gefolgt von LMW und HMW. IRRPEAKS- und LMW-Exposition wiesen die längste Belastung auf (beim Vergleich der Mediane), etwas darunter lag HMW.

Exposition der 1. Tätigkeit	**Exposition vorhanden**	**Keine Exposition vorhanden**
HMW	Anzahl Fälle: 103 Median: 893 Stunden Range in Stunden: [68,12.240]	Anzahl Fälle ohne Exposition: 991 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 565
LMW	Anzahl Fälle: 108 Median: 1.077 Stunden Range in Stunden: [34,13.090]	Anzahl Fälle ohne Exposition: 986 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 560
MIXED	Anzahl Fälle: 18 Median: 510 Stunden Range in Stunden: [68,10.710]	Anzahl Fälle ohne Exposition: 1.076 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 650
IRRPEAKS	Anzahl Fälle: 4 Median: 1.071 Stunden Range in Stunden: [170,3.570]	Anzahl Fälle ohne Exposition: 1.090 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 664
LOWRISK	Anzahl Fälle: 163 Median: 612 Stunden Range in Stunden: [34,15.980]	Anzahl Fälle ohne Exposition: 931 nie gearbeitet: 426 gearbeitet, aber keine Exposition: 505

Tabelle 7.3: Übersicht über die Expositionen in der ersten Tätigkeit

Boxplots der Exposition der ersten Tätigkeit

Die Boxplots der Exposition in der ersten Tätigkeit ist in Abbildung 7.8 dargestellt. Der Interquartilsabstand der Expositionen in der ersten Tätigkeit war deutlich schmaler als der der Expositionen über alle Tätigkeiten und Jahre hinweg, allerdings etwas breiter als der des 1. Tätigkeitsjahres. Die längsten Expositionsdauern herrschten erneut in den Kategorien LMW und IRRPEAKS vor. Die kürzesten Dauern wiesen MIXED und LOWRISK auf. Ausreißer mit besonders langen Expositionsdauern traten hauptsächlich in der Kategorie LOWRISK, HMW und LMW auf.

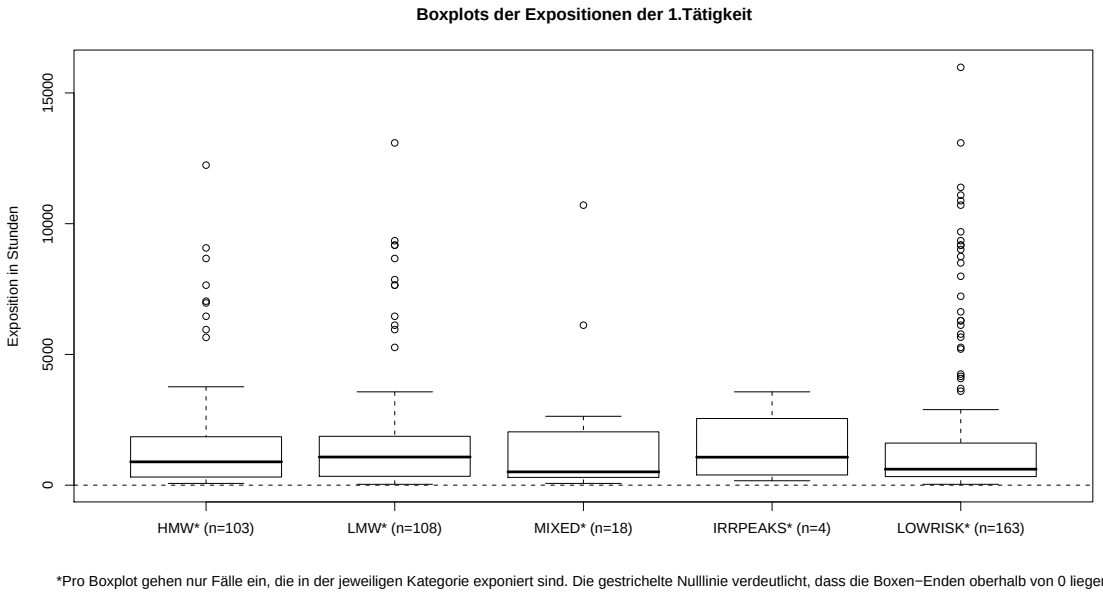

Abbildung 7.8: Boxplots der Expositionen in der ersten Tätigkeit auf Basis der vollständigen Tätigkeitsangaben

7.3 Berechnung der Exposition auf Basis aller Probanden

Nach der Imputation der fehlenden Tätigkeitsangaben wurden die Expositionen erneut auf Basis aller 1.187 Probanden jeweils für jeden der fünf vervollständigten Datensätze berechnet. Da die Berechnung der Expositionen in analoger Weise durchgeführt wurde, wird darauf an dieser Stelle nicht mehr detailliert eingegangen.

8 Logistische Regressionsmodelle

8.1 Modellannahmen

Ein weiteres Ziel dieser Arbeit war die Erstellung eines Regressionsmodells für Asthma und allergische Rhinitis. Da beide Zielvariablen binäre Variablen sind ($y_i \in \{0, 1\}$), ist ein logistisches Regressionsmodell geeignet. Ein solches Logit-Modell hat die Modellierung und Analyse der (bedingten) Wahrscheinlichkeit

$$\pi_i = \mathrm{P}(y_i = 1 \mid x_{i1}, ..., x_{ip}) = \mathrm{E}(y_i = 1 \mid x_{i1}, ..., x_{ip})$$

in Abhängigkeit der Kovariablen zum Ziel. Die Zielvariablen werden dabei bei gegebenen Kovariablen x_{i1}, ..., x_{ip} als (bedingt) unabhängig angenommen.
Das Modell ist durch folgende Formen darstellbar:

$$\pi_i = \frac{\exp(\beta_0 + x_{i1}\beta_1 + ... + x_{ip}\beta_p)}{1 + \exp(\beta_0 + x_{i1}\beta_1 + ... + x_{ip}\beta_p)} \tag{8.1}$$

$$\log(\frac{\pi_i}{1 - \pi_i}) = \beta_0 + x_{i1}\beta_1 + ... + x_{ip}\beta_p \tag{8.2}$$

$$\frac{\pi_i}{1 - \pi_i} = \exp(\beta_0 + x_{i1}\beta_1 + ... + x_{ip}\beta_p) \tag{8.3}$$

In Gleichung 8.1 wird durch die Wahrscheinlichkeit π_i der bedingte Erwartungswert von Y_i modelliert. Stellt man einige Umformungen an, so erhält man Gleichung 8.2, die die logarithmischen Chancen ("Logits") darstellt. Zur Darstellung der Chancen ("odds") eignet sich Gleichung 8.3.

8.2 Parameterschätzung

Die Parameterschätzung erfolgt im Logit-Modell nach der Maximum-Likelihood-Methode. Das Maximum-Likelihood-Prinzip besagt: Wähle zu den Realisationen denjenigen Parameter, für den die Wahrscheinlichkeit bzw. die Dichte, dass gerade diese Werte auftreten maximal wird, d.h. der die plausibelste Erklärung für das Zustandekommen dieser Werte liefert. [FAHRMEIR et al. 2004] Bei der Anwendung dieses Prinzips werden die Parameter bei gegebenen Daten durch Maximierung der (vollständigen und korrekt spezifizierten) Likelihood geschätzt. Dabei wird zunächst die Likelihood und daraus die log-Likelihood bestimmt. Leitet man diese log-Likelihood nach dem Parametervektor β ab, so erhält man die Score-Funktion. Nullsetzen der Score-Funktion liefert dann die ML-Gleichung. Diese

Gleichung wird üblicherweise iterativ durch den Fisher-Scoring-Algorithmus gelöst und liefert dann den ML-Schätzer für β. Zur Schätzung der Koeffizienten und der Kovarianzmatrix des ML-Schätzers muss zusätzlich noch die Informationsmatrix bestimmt werden. (Ausführliche Darstellungen zur Paramterschätzung siehe [FAHRMEIR et al. 2007].)

8.3 Parameterinterpretation

Interpretation der Logits
Möchte man auf Basis der Gleichung 8.2 die Parameter interpretieren, so gilt die übliche Interpretation des linearen Modells: nimmt die Variable x_{ij} um eine Einheit zu (von x_{ij} auf $x_{ij} + 1$) und hält man alle anderen Kovariablenwerte konstant, so verändern sich die Logits, also die logarithmierten Chancen, additiv um den Wert β_j.

Interpretation der Chance
Eine anschaulichere Interpretation basiert auf Gleichung 8.3, in der die Kovariablen in exponentiell-multiplikativer Form auf die Chancen wirken. Wird die Variable x_{ij} um eine Einheit auf $x_{ij} + 1$ erhöht und hält man alle anderen Kovariablenwerte fest, so verändert sich die Chance multiplikativ um den Faktor $\exp(\beta_j)$. Ist β_j positiv, so vergrößert sich die Chance (da $\exp(\beta_j) > 1$), für negative β_j verkleinert sie sich (da $\exp(\beta_j) < 1$) und für $\beta_j = 0$ bleibt die Chance unverändert (da $\exp(0) = 1$).

Interpretation der Odds Ratios (Chancenverhältnisse)
Weiterhin können auch die Odds Ratios, die Chancenverhältnisse, interpretiert werden. Es gilt dabei, dass das Chancenverhältnis zwischen Y bei x_{ij} und Y bei $x_{ij} + 1$ gleich $\exp(\beta_j)$ ist. Dies soll anhand eines Beispiels erläutert werden:
Wählt man beispielsweise die Einflussgröße Rauchen, die einen möglichen Einfluss auf eine bestimmte Erkrankung hat, so kann man die Chance eines Rauchers (x = 1) zu erkranken (im Vergleich nicht zu erkranken) folgendermaßen berechnen:

$$\gamma(x = 1) = \frac{\pi(x=1)}{1-\pi(x=1)}.$$

Die Chance eines Nichtrauchers (x = 0) berechnet sich analog:

$$\gamma(x = 0) = \frac{\pi(x=0)}{1-\pi(x=0)}.$$

Das Odds-Ratio, also das Chancenverhältniss, ist somit der Quotient der beiden Chancen:

$$OR = \frac{\gamma(x=1)}{\gamma(x=0)} = \frac{\frac{\pi(x=1)}{1-\pi(x=1)}}{\frac{\pi(x=0)}{1-\pi(x=0)}} = \frac{\exp(\beta_0)\cdot\exp(\beta_1)}{\exp(\beta_0)} = \exp(\beta_1).$$

Möchte man nun das Odds Ratio von Rauchern gegenüber Nichtrauchern im Zusammenhang mit dieser Erkrankung (bei festgehaltenen restlichen Kovariablenwerten) interpretieren, so gilt:

- Ist das Odds Ratio gleich 1, so sind die Chancen zu erkranken von Rauchern und Nichtrauchern gleich.
- Ist das Odds Ratio größer (kleiner) als 1, so sind die Chancen zu erkranken für Raucher um den Faktor $\exp(\beta_1)$ höher (niedriger) als die Chancen für Nichtraucher.

8.4 Zielvariablen und potenzielle Einflussgrößen der Modelle

In die logistischen Modelle, die im Rahmen dieser Arbeit berechnet und analysiert wurden, gingen Variablen aus allen drei Studien (ISAAC II, SOLAR, SOLAR II) ein. Diverse Rekodierungen waren nötig, um die Variablen in der benötigten Form vorliegen zu haben. Details zu den einzelnen Rekodierungen können dem Anhang entnommen werden (vgl. Anhang A).
Folgende Variablen aus SOLAR II dienten als Zielvariablen in den Logit-Modellen und wurden getrennt voneinander modelliert:

- Allergische Rhinitis

$$s2CURHAYV = \begin{cases} 1, & Allergische\ Rhinitis\ in\ SOLAR\ II \\ 0, & Keine\ allergische\ Rhinitis\ in\ SOLAR\ II \end{cases}$$

  In diesem Modell wurden die Personen mit Asthma während ISAAC II und/oder SOLAR ausgeschlossen.

- Asthma

$$s2CURASTHV = \begin{cases} 1, & Asthma\ in\ SOLAR\ II \\ 0, & Kein\ Asthma\ in\ SOLAR\ II \end{cases}$$

  In diesem Modell wurden die Variablen Allergische Rhinitis (in ISAAC II und SOLAR) als Kovariablen verwendet.

Als Kovariablen konnten in die Modelle folgende Einflussgrößen eingehen.
Aus **ISAAC II** galten folgende Angaben aus dem Fragebogen als mögliche Confoundervariablen für die Modelle:

- Studienzentrum (Dresden/München)
- Sozioökonomischer Status (Hoch/Niedrig)
- In Deutschland geboren (Ja/Nein)
- Atopie der Eltern (Ja/Nein)
- Geschwister (Ja/Nein)
- Als Säugling gestillt (Ja/Nein)
- Neurodermitis (Ja/Nein)
- Allergische Rhinitis (Ja/Nein)
- Asthma (Ja/Nein)
- Passivrauch (Eltern Raucher/Eltern Ex-Raucher/Eltern Nichtraucher)

Aus **SOLAR** konnten folgende Angaben aus dem Fragebogen als potenzielle Confoundervariablen in die Modelle eingehen:

- Geschlecht (Männlich/Weiblich)
- Neurodermitis (Ja/Nein)
- Allergische Rhinitis (Ja/Nein)
- Asthma (Ja/Nein)
- Passivrauch (Ja/Nein)
- Rauchverhalten (Raucher/Nichtraucher)

Aus **SOLAR II** standen folgende Angaben aus dem Fragebogen als mögliche Confoundervariablen zur Verfügung:

- Passivrauch (Ja/Nein)
- Rauchverhalten (Raucher/Ex-Raucher/Nichtraucher)
- Schulbildung (Höhere/Niedrigere)

Weiterhin wurden die Tätigkeitsangaben in den Fragebögen aus **SOLAR** und **SOLAR II** zusammengefasst, um folgende Einflussgrößen verwenden zu können:

- Jemals gearbeitet (Ja/Nein)
- Job-Exposition (Exposition kumuliert über alle Tätigkeiten und Jahre, Exposition im 1. Tätigkeitsjahr, Exposition während der 1. Tätigkeit)

8.5 Vorgehen bei der Modellwahl

Durch die zuvor aufgelisteten Variablen standen nun eine Vielzahl an potenziellen Einflussgrößen zur Verfügung. Nun musste entschieden werden, welche der zur Verfügung stehenden Variablen in die Logitmodelle aufgenommen werden sollten.
Um ein geeignetes Modell auszuwählen, wurde im vorliegenden Datensatz zunächst auf Basis der Confoundervariablen (alle zuvor aufgelisteten Variablen mit Ausnahme der Expositionsvariablen) ein Confounder-Modell erstellt. Um solch ein Confounder-Modell erstellen zu können, mussten aus den zwanzig potenziellen Confoundervariablen die relevanten Variablen ausgewählt werden. Da aus zwanzig Variablen allerdings 2^{20} Modelle (mehr als 1 Million Modelle) resultieren, war es offensichtlich, dass nicht alle möglichen Modelle miteinander verglichen werden konnten. Aus diesem Grund wurde Variablenselektion durch Schrittweise Selektion über das AIC (Akaikes Informationskriterium) durchgeführt. Nachdem durch dieses Vorgehen ein geeignetes Confounder-Modell festgelegt wurde, konnten die Expositionsvariablen einbezogen werden. Dabei wurde jeweils separat überprüft, ob bzw. welche der folgenden Expositionen zusätzlich zu den Confoundervariablen in das Logitmodell aufgenommen werden sollten:

- kumulierte Exposition über alle Tätigkeiten und Jahre
- binäre Exposition über alle Tätigkeiten und Jahre
- kumulierte Exposition im 1. Tätigkeitsjahr
- binäre Exposition im 1. Tätigkeitsjahr
- kumulierte Exposition während der 1. Tätigkeit
- binäre Exposition während der 1. Tätigkeit

Zunächst wurde mit Hilfe eines GAMs (Generalized Additive Models) überprüft, ob der Einfluss der (kumulierten) Expositionsvariablen auf die Zielvariablen linear ist. Dieser Kontrollschritt wurde durchgeführt, um abzusichern, ob die Expositionsvariablen als lineare Terme in das Logitmodell aufgenommen werden konnten.
Für jede der oben aufgeführten Expositionen wurde im nächsten Schritt ein Likelihood-Ratio-Test durchgeführt. Darin wurde das Confoundermodell dem Modell inklusive der jeweiligen fünf Expositionsvariablen (HMW, LMW, MIXED, IRRPEAKS, LOWRISK) gegenüber gestellt. Somit konnte die Nullhypothese "Der Einfluss der fünf Expositionsvariable ist gleich 0" überprüft werden. Wurde die Nullhypothese auf dem Signifikanzniveau von 5% verworfen, so wurden alle fünf Expositionsvariablen zusätzlich zu den Confoundervariablen in das Modell aufgenommen.
Bevor die einzelnen Schritte der Modellwahl ausführlich erläutert werden, soll durch die Grafik 8.1 das Vorgehen schematisch zusammengefasst werden.

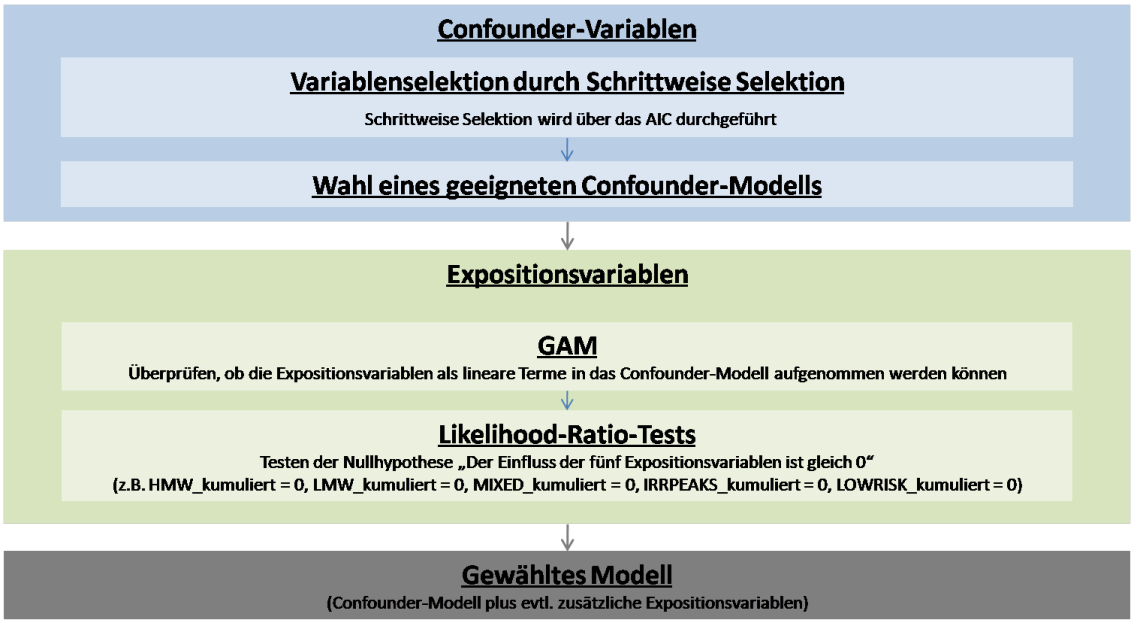

Abbildung 8.1: Schema der Vorgehensweise bei der Modellwahl

8.5.1 Variablenselektion durch Schrittweise-Selektion

Die Variablenselektion wurde durch Schrittweise-Selektion auf Basis des AIC-Kriteriums (Akaikes Informationskriterium) durchgeführt. Das Vorgehen soll im folgenden Abschnitt erläutert werden.

AIC - Akaikes Informationskriterium

Bei der Wahl mehrerer konkurrierender Modelle, die verschiedene Prädiktoren und unterschiedliche Anzahlen an Parametern enthalten, ist man stets mit einem Tradeoff zwischen Modellgenauigkeit, d.h. guter Datenanpassung, und Modelleinfachheit, d.h. geringer Anzahl an Parametern, konfrontiert. Aus diesem Grund wurden Modellwahlkriterien entwickelt, die eine Überanpassung an den Datensatz vorbeugen, indem sie eine zu hohe Modellkomplexität, d.h. eine zu hohe Anzahl an Parametern, bestrafen. Das Modellwahlkriterium, das im Rahmen der Maximum-Likelihood-Inferenz wohl am häufigsten verwendet wird, ist das AIC: [FAHRMEIR et al. 2007]

$$AIC = -2l(\widehat{\beta}) + 2p.$$

Der erste Term dieser Gleichung ist der Wert der Log-Likelihood, wenn der ML-Schätzer $\widehat{\beta}$ eingesetzt wurde, wodurch man den maximalen Wert der Log-Likelihood erhält. Dieser Term steht für die Modellanpassung, d.h. wie gut das Modell zu den vorliegenden Daten passt. Im zweiten Term ist die Anzahl der Parameter p enthalten. Dieser Term gilt als Strafterm für die Modellkomplexität aufgrund der entsprechenden Anzahl an Parameter. Bei der Wahl zwischen den konkurrierenden Modellen wird das Modell mit dem kleinsten AIC bevorzugt.

Variablenselektionsverfahren auf Basis des AIC-Kriteriums

Wie bereits zuvor angesprochen wurde, war die Berechnung und der Vergleich aller möglichen Modelle in diesem Fall nicht durchführbar, da es bei 20 potenziellen Confoundervariablen 2^{20} (mehr als 1 Million) mögliche Modelle gäbe. Um auch ohne die Berechnung aller möglichen Modelle ein geeignetes Modell zu erhalten, eignen sich Selektionsverfahren, die beispielsweise auf dem AIC-Kriterium basieren.
Bevor man ein Selektionsverfahren anwendet, sollte man durch substanzwissenschaftliche Überlegungen und Vorwissen eine Vorauswahl an potenziellen Modellen treffen. [FAHRMEIR et al. 2007] In dieser Arbeit wurde vorab festgelegt, dass ausschließlich Haupteffekte in die Modelle eingehen und Interaktionseffekte nicht beachtet werden. Weiterhin sollten die Variablen Geschlecht und Sozioökonomischer Status fest im Modell bleiben.
Das Geschlecht sollte im Modell bleiben, da man davon ausgehen kann, dass sich Abläufe zwischen den Geschlechtern unterscheiden, die sich in Bezug auf Krankheiten, im Zusammenhang mit der Auswirkung von möglichen Risikofaktoren oder ganz allgemein in Bezug auf Körperfunktionen abspielen. Da diese Unterschiede gerade auch in der Pubertät zum Tragen kommen, war es in der vorliegenden Studie, in der dieser Zeitraum mitbetrachtet wurde, sehr wichtig, diese Variable als Einflussgröße im Modell festzulegen. Vor allem in Bezug auf Asthmaerkrankungen zeigten diverse Studien, dass sich das Auftreten von Asthma bei den Geschlechtern unterscheidet. Bei Kindern bis 16 Jahren litten Jungen

häufiger an Asthma als Mädchen. Später im Jugendlichenalter (17-23 Jahre) drehte sich dieses Verhältnis um. [ANDERSON et al. 1992]
Der Sozioökonomische Status wurde als wichtige Einflussgröße erachtet, da man davon ausgehen kann, dass es für die generelle Gesundheit und die Entwicklung eines Kindes einen Unterschied macht, ob es in einem Umfeld mit "höherem" oder "niedrigerem" Sozialstatus aufwächst. Diese Aussage ist aus der Hygiene-Hypothese von David Strachan abgeleitet. [STRACHAN 1989] Dabei bedeutet ein höherer Status oft, dass ein Kind "steriler" aufwächst, somit beispielsweise weniger Keimen ausgesetzt ist und folglich weniger abgehärtet wird.
Die dadurch gewählten potenziellen Modelle konnten anhand des AIC-Kriteriums verglichen werden. Dabei gibt es drei generelle Vorgehensweisen:

- **Vorwärts-Selektion** (Forward-Selection)
  Bei der Vorwärts-Selektion wird zunächst ein minimales Modell (kleinstmögliche Anzahl an Einflussgrößen) festgelegt. In jedem Selektionsschritt wird die Kovariable ins Modell aufgenommen, die die größte Reduktion des AICs liefert. Dieses Verfahren wird so lange durchgeführt, bis keine Reduktion des AIC-Kriteriums mehr möglich ist.

- **Rückwärts-Selektion** (Backward-Selection)
  Bei der Rückwärts-Selektion wird mit einem maximalen Modell (größtmögliche Anzahl an Einflussgrößen) gestartet. In jedem Selektionsschritt wird die Kovariable aus dem Modell entfernt, die die größte Reduktion des AIC-Kriteriums liefert. Das Selektionsverfahren ist beendet, so bald keine Reduktion des AICs mehr möglich ist.

- **Schrittweise-Selektion** (Stepwise-Selection)
  Die Schrittweise-Selektion ist eine Kombination aus Vorwärts- und Rückwärts- Selektion. In jedem Schritt wird geprüft, ob die Aufnahme oder Entfernung einer Variable die größte Reduktion des AICs liefert. Somit kann in jedem Schritt sowohl eine Einflussgröße aufgenommen als auch entfernt werden. Es ist möglich, dass in einem früheren Selektionsschritt bereits entfernte Variablen in einem späteren Schritt erneut in das Modell aufgenommen werden. Auch hier ist das Selektionsverfahren beendet, wenn keine Reduktion des AICs mehr möglich ist.

In der vorliegenden Arbeit wurde als Selektionsverfahren die Schrittweise-Selektion verwendet, da die Kombination aus Vorwärts- und Rückwärts-Selektion am sinnvollsten erschien.
Wichtig ist die Tatsache, dass diese Selektionsverfahren im Allgemeinen nicht zu dem besten Modell - im Sinne des AIC-Kriteriums - führen, da nicht alle potenziellen Modelle miteinander verglichen werden. Allerdings liefern sie in der Regel ein sehr gutes Modell. [FAHRMEIR et al. 2007]

8.5.2 Prüfung des linearen Einflusses der Expositionsvariablen mit Hilfe eines GAMs (Generalized Additive Models)

Die generalisierten additiven Modelle (GAM) stellen eine Art Erweiterung zu den generalisierten linearen Modellen (GLM) dar, zu dem das Logitmodell gehört. Die Anwendung eines GAMs eignet sich in folgender Situation:
Für einige Kovariablen $(x_{i1}, ..., x_{ik})$ kann man davon ausgehen, dass der Einfluss auf die Zielvariable (y_i) durch einen linearen Prädiktor beschreiben werden kann. Zusätzlich gibt es allerdings weitere metrische Einflussgrößen $(z_{i1}, ..., z_{iq})$, bei denen man - zumindest a priori - nicht davon ausgehen kann, dass deren Einfluss auf die Zielvariable linear ist. Ziel ist deshalb, den Einfluss dieser Kovariablen flexibel durch eine Funktion zu modellieren. Um die Schätzung dieser Funktion zu vereinfachen, unterstellt man für den Einfluss der Kovariablen zusätzlich eine additive Struktur. [FAHRMEIR et al. 2007] Generalisierte additive Modelle stellen somit eine Erweiterung der generalisierten linearen Modelle, z.B. des Logitmodells, dar:

$$\log \frac{\pi_i}{1-\pi_i} = \beta_0 + x_{i1}\beta_1 + ... + x_{ik}\beta_k + f_1(z_{i1}) + ... + f_q(z_{iq})$$

Die Funktionen $f_1(z_{i1}), ..., f_q(z_{iq})$ werden im Rahmen des GAMs geschätzt.
Im Rahmen dieser Arbeit wurde das GAM genutzt, um den Einfluss der metrischen Expositionsvariablen zu modellieren. Dazu wurden die Variablen aus dem Confoundermodell als lineare Terme aufgenommen. Zusätzlich wurden die Einflüsse der (kumulierten) Expositionsvariablen durch Funktionen modelliert. Die geschätzten Funktionen wurden daraufhin betrachtet, um zu entscheiden, ob die Exposition als lineare (oder quadratische) Terme in das Logit-Modell aufgenommen wurden.

8.5.3 Likelihood-Ratio-Test

Unter Anwendung eines Likelihood-Ratio-Tests wurde geprüft, ob durch die Aufnahme der Expositionsvariablen als zusätzliche Variablen in das Confoundermodell eine Verbesserung des Modells erreicht werden kann.
Bei diesem Test wird die Likelihood-Ratio-Statistik verwendet:

$$lq = -2(l(\tilde{\beta}) - l(\widehat{\beta}))$$

Diese Statistik misst die Abweichung zwischen der Log-Likelihood des unrestringierten Modells $l(\widehat{\beta})$ und der Log-Likelhood des unter H_0 restringierten Modells $l(\tilde{\beta})$.
Im Likelihood-Ratio-Test können lineare Hypothesen der Form

$$H_0\text{: } \mathrm{C}\beta = \mathrm{d} \text{ gegen } H_1\text{: } \mathrm{C}\beta \neq \mathrm{d} \text{ mit } \mathrm{rg}(\mathrm{C}) = \mathrm{r}$$

formuliert werden. Der Likelihood-Ratio-Test eignet sich zum Vergleich von hierarchischen Modellen. Dabei kann geprüft werden, ob die Aufnahme einer oder mehrerer Variablen zu einer Modellverbesserung führt.
Die Teststatistik ist unter H_0 asymptotisch χ^2-verteilt mit r Freiheitsgraden. Ist das unrestringierte Maximum $l(\widehat{\beta})$ deutlich größer als $l(\tilde{\beta})$, so wird die Teststatistik lq groß und folglich wird H_0 zugunsten von H_1 abgelehnt. [FAHRMEIR et al. 2007]
Konkret galt auf dem vorliegenden Datensatz die Nullhypothese
H_0: “Der Einfluss der fünf Expositionsvariablen ist gleich 0”.

Das unrestringierte Modell war folglich das Confoundermodell inklusive der jeweiligen fünf Expositionsvariablen. Das restringierte Modell entsprach dem Confoundermodell (ohne den fünf Expositionsvariablen). In diesem Fall testete man auf Signifikanz der jeweiligen Expositionsvariablen. War der Wert der Teststatistik größer als der kritische Wert (in diesem Fall $\chi^2_{0.95}(5) = 11.07$) so wurde die Nullhypothese auf dem 5%-Signifikanzniveau abgelehnt. Dies bedeutet, dass mindestens eine der Expositionsvariablen einen Einfluss auf die Zielvariable hatte. In diesem Fall wurden alle fünf Expositionsvariablen gemeinsam in das Modell aufgenommen.

8.5.4 ROC-Analyse des gewählten Modells

Aufgrund von diagnostischen Tests wird in der Epidemiologie angenommen, dass eine bestimmte Krankheit vorliegt (K) oder dass die Krankheit nicht vorliegt ($\overline{K}$). Diese Tests können positiv (T^+) oder negativ (T^-) ausfallen. Um die Brauchbarkeit eines Tests zur Erkennung einer Krankheit anzugeben, gibt es in diesem Zusammenhang zwei wichtige bedingte Wahrscheinlichkeiten:

- **Sensitivität:** $P(T^+|K)$
  Wahrscheinlichkeit eines positiven Testergebnisses bei einer kranken Person
- **Spezifität:** $P(T^-|\overline{K})$
  Wahrscheinlichkeit eines negativen Testergebnisses bei einer gesunden Person

Ein guter Test liegt vor, wenn diese beiden Wahrscheinlichkeiten möglichst groß sind, d.h. nahe bei 1 liegen.

Das Ergebnis eines Tests ist oft eine kontinuierliche Messgröße. In der Regel gibt es also keine "natürliche" Grenze zwischen erkrankt und nicht erkrankt. Folglich sind Sensitivität und Spezifität abhängig von der Festlegung eines Trennwertes, der "cut off value" genannt wird. Ziel dieses Cut-off-Wertes ist es, den Anteil falsch positiver und/oder falsch negativer Entscheidungen möglichst minimal zu halten. Bei der Wahl eines "optimalen" Trennwertes müssen sowohl Risiken für falsche Entscheidungen als auch substanzwissenschaftliche Überlegungen beachtet werden.
Eine Methode, die häufig zur Festlegung eines objektiven Cut-off-Wertes verwendet wird, ist das ROC-Verfahren ("receiver operating characteristic"). Bei diesem Verfahren werden die Sensitivitäten und Spezifitäten an möglichst vielen Stellen des Definitionsbereiches berechnet, der in diskreten Schritten durchlaufen wird. Als Ergebnis dieser Analyse erhält man die sogenannte ROC-Kurve.
Der Trennwert wird dann als optimal eingeschätzt, wenn die Werte für Sensitivität und Spezifität möglichst hoch liegen. Weiterhin kann die Fläche unter der ROC-Kurve ("Area Under Curve", AUC) betrachtet werden. Je größer diese Fläche ist, umso besser trennt der vorliegende Test. Eine maximale Fläche wird erreicht, wenn sowohl Spezifität als auch Sensitivität 100% betragen, d.h. alle kranken Personen erhalten ein positives Testergebnis und alle gesunden Personen ein negatives Testergebnis. Verläuft die ROC-Kurve entlang der Winkelhalbierenden, so beträgt die Fläche 0.5 und somit ist keine Trennung möglich. Die Theorie zur ROC-Analyse in diesem Kapitel basiert auf [SACHS und HEDDERICH 2006].

Die in dieser Arbeit berechneten logistischen Regressionsmodelle für die Zielgrößen Asthma und Allergische Rhinitis (jeweils in SOLAR II), konnten in diesem Zusammenhang als (diagnostische) Tests aufgefasst werden. Das Ergebnis dieser Logitmodelle waren Wahrscheinlichkeiten, mit denen ein Proband unter Vorliegen bestimmter Kovariablenwerte an der jeweiligen Krankheit litt. Aus diesem Grund erschien eine ROC-Analyse des gewählten Modells als sehr sinnvoll.

8.6 Logistische Regressionsmodelle für die Probanden mit vollständigen Tätigkeitsangaben

Zunächst wurden ausschließlich die Probanden betrachtet, die vollständige Tätigkeitsangaben gemacht hatten (Definition der vollständigen Tätigkeitsangaben siehe Kapitel 5.3). Probanden, die unvollständige Tätigkeitsangaben hatten, wurden erst in einem späteren Analyseschritt betrachtet (vgl. Kapitel 8.7). Dieses Vorgehen wurde gewählt, um zunächst auf Basis der "sicheren" Expositionen, die von den Probanden tatsächlich so angegeben wurden, ein Modell zu erstellen. Dieses Modell sollte dann mit dem resultierenden Modell auf Basis aller Probanden (inklusive imputierter Tätigkeitsangaben) verglichen werden.
Auf Basis der Probanden mit vollständigen Tätigkeitsangaben wurden zwei logistische Regressionsmodelle angepasst:

- **Logit-Modell 1** - Zielgröße: Allergische Rhinitis in SOLAR II
  Datenbasis: Probanden mit vollständigen Tätigkeitsangaben, die während ISAAC II und/oder SOLAR kein Asthma hatten (n = 1.032)
- **Logit-Modell 2** - Zielgröße: Asthma in SOLAR II
  Datenbasis: Probanden mit vollständigen Tätigkeitsangaben (n = 1.094)

Es lagen insgesamt fünf Datensätze vor, bei denen vollständige Tätigkeitsangaben und imputierte Werte bei den Confoundervariablen enthalten waren. Diese Datensätze waren folgendermaßen entstanden: Zunächst wurden die Confoundervariablen, die fehlende Werte enthielten, fünf mal imputiert, so dass fünf vervollständigte Datensätze (mit Confoundervariablen) vorlagen. In einem zweiten Schritt wurde jeder dieser fünf Datensätze jeweils mit dem Datensatz, der ausschließlich die Probanden mit vollständigen Tätigkeitsangaben enthielt, kombiniert (vgl. Kapitel 6.1).
Auf Basis dieser fünf Datensätze konnten die beiden logistischen Regressionsmodelle gemäß des zuvor beschriebenen Vorgehens der Modellwahl ausgewählt werden.

8.6.1 Logit-Modell 1: Allergische Rhinitis auf Basis der Personen mit vollständigen Tätigkeitsangaben

Variablenselektion und Wahl eines Confoundermodells

Da in diesem Modell ausschließlich Personen betrachtet wurden, die während ISAAC II bzw. SOLAR kein Asthma hatten, wurden die Variablen "Asthma in ISAAC II" und "Asthma in SOLAR" nicht in die Liste der möglichen Einflussgrößen aufgenommen. Für die Variablenselektion wurde als minimales Modell das Modell mit Intercept und den Einflussgrößen Geschlecht und Sozioökonomischer Status definiert. Als maximales Modell standen zusätzlich zum Intercept insgesamt 18 Variablen als potenzielle Confoundervariablen zur Verfügung.
Die Variablenselektion durch Schrittweise-Selektion auf Basis des AIC-Kriteriums wurde auf jedem der fünf Datensätze durchgeführt. Tabelle 8.1 stellt dar, welche Variablen zusätzlich zu den Variablen Geschlecht und sozioökonomischer Status für das jeweilige Modell selektiert wurden.

Daten	Atopie der Eltern	Allergische Rhinitis (ISAAC II)	Allergische Rhinitis (SOLAR)	Als Säugling gestillt	Passivrauch (SOLAR)
1	+	+	+	-	-
2	+	+	+	+	-
3	+	+	+	-	-
4	+	+	+	+	+
5	+	+	+	+	+

Tabelle 8.1: Allergische Rhinitis-Modelle auf Basis der Probanden mit vollständigen Tätigkeitsangaben: Selektierte Confoundervariablen (zusätzlich zu Geschlecht und Sozioökonomischer Status)

Folglich standen drei potenzielle Confoundermodelle zur Auswahl:

- **Confoundermodell A**:
  Einflussgrößen: Geschlecht, Sozioökonomischer Status, Atopie der Eltern, Allergische Rhinits (ISAAC II), Allergische Rhinitis (SOLAR)
- **Confoundermodell B**:
  Einflussgrößen: Geschlecht, Sozioökonomischer Status, Atopie der Eltern, Allergische Rhinits (ISAAC II), Allergische Rhinitis (SOLAR), als Säugling gestillt
- **Confoundermodell C**:
  Einflussgrößen: Geschlecht, Sozioökonomischer Status, Atopie der Eltern, Allergische Rhinits (ISAAC II), Allergische Rhinitis (SOLAR), als Säugling gestillt, Passivrauch (SOLAR)

Wünschenswert war an dieser Stelle allerdings ein einziges Confoundermodell auszuwählen, das dann für die weiteren Modellwahl-Schritte verwendet wurde. Um dieses eine Confoundermodell auszuwählen, wurden die drei potenziellen Modelle auf jeden der fünf Datensätze unter Anwendung von Likelihood-Ratio-Tests gegeneinander getestet. Lieferte ein größeres Modell (*z.B. Confoundermodell B*) auf Basis mindestens eines Datensatzes eine Verbesserung gegenüber eines kleineren Modells (*z.B. Confoundermodell A*), so wurde die Entscheidung getroffen, das größere Modell zu bevorzugen. Durch dieses Vorgehen

nahm man eher die Tatsache in Kauf, eine zusätzliche (evtl. irrelevante) Variable in das Confoundermodell aufzunehmen, als eine möglicherweise relevante Variable im Modell nicht zu berücksichtigen, da man an unverzerrten Schätzern interessiert war. Das Nichtberücksichtigen einer relevanten Variable führt nämlich zu einer Verzerrung der Schätzer. Die Aufnahme einer irrelevanten Variable hingegen liefert unverzerrte Schätzer, führt allerdings zu einer Erhöhung der Varianz, d.h. lediglich zu einem Genauigkeitsverlust bei der Schätzung. [FAHRMEIR et al. 2007]
Bei den vorliegenden Datensätzen lieferte das Confoundermodell B auf einem Datensatz (Datensatz 2) eine Verbesserung gegenüber des kleineren Modells A. Das Confoundermodell C lieferte auf keinem Datensatz eine Verbesserung. Folglich fiel die Entscheidung auf das Confoundermodell B.

Einflussgrößen des gewählten Confoundermodells für Allergische Rhinitis auf Basis der Probanden mit vollständigen Tätigkeitsangaben
Geschlecht, Sozioökonomischer Status, Atopie der Eltern
Allergische Rhinits (ISAAC II), Allergische Rhinitis (SOLAR), als Säugling gestillt

Aufnahme von Expositionsvariablen in das Confoundermodell

Wie zuvor bereits erläutert, konnten nun folgende Gruppen von Expositionsvariablen zusätzlich zu den Confoundervariablen in das Logitmodell aufgenommen werden:

- kumulierte Exposition über alle Tätigkeiten und Jahre
- binäre Exposition über alle Tätigkeiten und Jahre
- kumulierte Exposition im 1. Tätigkeitsjahr
- binäre Exposition im 1. Tätigkeitsjahr
- kumulierte Exposition während der 1. Tätigkeit
- binäre Exposition während der 1. Tätigkeit

Zunächst wurde für die kumulierten Expositionen anhand eines GAMs überprüft, ob der Einfluss dieser stetigen Expositionsvariablen linear modelliert werden könnte. Dafür wurden die geschätzten Funktionen betrachtet. Beispielhaft sind die geschätzten Funktionen für die kumulierten Expositionen über alle Tätigkeiten und Jahre hinweg (aus dem Datensatz 1) in Abbildung 8.2 dargestellt: HMW-Exposition kumuliert, LMW-Exposition kumuliert, MIXED-Exposition kumuliert und LOWRISK-Exposition kumuliert. Für die fünfte Einflussgröße (IRRPEAKS-Exposition kumuliert) konnte aufgrund zu geringem Auftretens dieser Belastung (nur in sieben Fällen trat diese Belastung auf) keine Funktion geschätzt werden. Die geschätzten Funktionen der anderen Expositionsvariablen (kumulierte Exposition im 1. Tätigkeitsjahr bzw. während der 1. Tätigkeit) sowohl auf Basis dieses Datensatzes, als auch auf Basis der anderen vier Datensätze wiesen nahezu identische Verläufe auf.

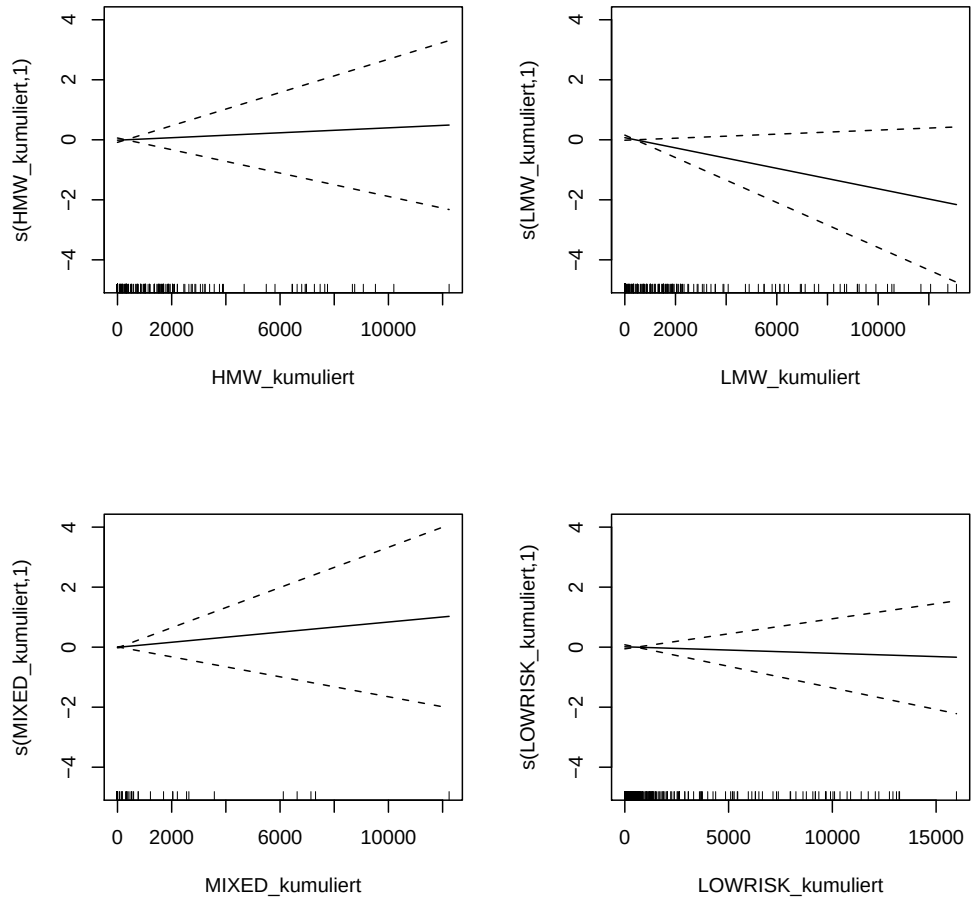

Abbildung 8.2: Geschätzter Funktionsverlauf des Einflusses der kumulierten Expositionsvariablen auf die Zielgröße Allergische Rhinitis unter Anwendung eines GAMs

Wie auf der Abbildung deutlich zu erkennen, konnten die Expositionsvariablen als lineare Terme in die Logitmodelle aufgenommen werden.
Um beurteilen zu können, ob die Aufnahme der Expositionsvariablen in das Confoundermodell eine Verbesserung liefert, wurden auf jedem der fünf Datensätze jeweils Likelihood-Ratio-Tests (LR-Tests) durchgeführt. Dafür wurden die Expositionsvariablen für alle Kategorien (HMW, LMW, MIXED, IRRPEAKS und LOWRISK) zusätzlich in das ausgewählte Confoundermodell aufgenommen und dem "reinen" Confoundermodell gegenübergestellt. Diese LR-Tests wurden jeweils separat für die kumulierte und binäre Exposition über alle Tätigkeiten und Jahre hinweg, für die kumulierte und binäre Exposition im 1. Tätigkeitsjahr und für die kumulierte und binäre Exposition während der 1. Tätigkeit durchgeführt. Insgesamt wurden also auf jedem der fünf Datensätze sechs LR-Tests durchgeführt. Die Tabelle 8.2 liefert eine Übersicht, über die p-Werte der LR-Tests.

Likelihood-Ratio-Test Confoundermodell vs. Modell inklusive ...	**p-Wert** Datensatz 1	**p-Wert** Datensatz 2	**p-Wert** Datensatz 3	**p-Wert** Datensatz 4	**p-Wert** Datensatz 5
kumulierte Expositionen	0,45	0,43	0,45	0,44	0,45
binäre Expositionen	0,17	0,16	0,17	0,16	0,17
Expositionen des 1. Tätigkeitsjahres	0,59	0,57	0,59	0,59	0,59
binäre Expositionen des 1. Jahres	0,26	0,26	0,26	0,25	0,26
Expositionen der 1. Tätigkeit	0,27	0,25	0,28	0,27	0,27
binäre Expositionen der 1. Tätigkeit	0,67	0,66	0,67	0,66	0,67

Tabelle 8.2: Übersicht über die p-Werte der durchgeführten Likelihood-Ratio-Tests

In keinem der Fälle lieferte allerdings die zusätzliche Aufnahme der Expositionsvariablen eine (statistische) Verbesserung des Modells für Allergische Rhinitis. Aus inhaltlichen Gründen war es allerdings nötig, Expositionsvariablen in das finale Modell aufzunehmen. Da das Modell mit den binären Expositionen im Vergleich zu den anderen Expositionen auf Basis des AICs am besten geeignet war, wurde die binäre Exposition über alle Tätigkeiten und Jahre in das Modell aufgenommen. Die Tabelle 8.3 liefert eine Übersicht über die AICs der verschiedenen Modelle, wobei pro Datensatz für die Modelle mit Exposition das Modell mit dem kleinsten AIC hervorgehoben ist.

Modell	**AIC** Datensatz 1	**AIC** Datensatz 2	**AIC** Datensatz 3	**AIC** Datensatz 4	**AIC** Datensatz 5
Confoundermodell (ohne Expositionsvariablen)	592,67	588,85	591,58	590,83	591,65
Confoundermodell inkl. kumulierte Expositionen	597,97	593,95	596,87	596,06	596,93
Confoundermodell inkl. binäre Expositionen	**594,96**	**590,99**	**593,75**	**592,85**	**593,92**
Confoundermodell inkl. Expositionen des 1. Tätigkeitsjahres	598,96	595,03	597,88	597,08	597,95
Confoundermodell inkl. binäre Expositionen des 1. Jahres	596,21	592,30	595,07	594,19	595,18
Confoundermodell inkl. Expositionen der 1. Tätigkeit	596,31	592,22	595,26	594,46	595,31
Confoundermodell inkl. binäre Expositionen der 1. Tätigkeit	599,47	595,62	598,38	597,54	598,45

Tabelle 8.3: Übersicht über die AICs der unterschiedlichen Modelle

Somit wurde als finales Modell das Confoundermodell inklusive der binären Expositionen (HMW-Exposition binär, LMW-Exposition binär, MIXED-Exposition binär, IRRPEAKS-Exposition binär und LOWRISK-Exposition binär) ausgewählt.

Analyse des finalen Modells für Allergische Rhinits

Als finales Modell für Allergische Rhinits wurde folgendes Modell ausgewählt:

Einflussgrößen des finalen Modells für Allergische Rhinitis auf Basis der Probanden mit vollständigen Tätigkeitsangaben
Geschlecht, Sozioökonomischer Status, Atopie der Eltern
Allergische Rhinits (ISAAC II), Allergische Rhinitis (SOLAR), als Säugling gestillt,
HMW-Exposition binär, LMW-Exposition binär, MIXED-Exposition binär,
IRRPEAKS-Exposition binär, LOWRISK-Exposition binär

Dieses ausgewählte Modell konnte nun auf allen fünf Datensätzen durch eine ROC-Kurve analysiert werden. Abbildung 8.3 zeigt nun beispielhaft die ROC-Kurve für den ersten Datensatz. Die ROC-Kurven der anderen Datensätze wiesen einen sehr ähnlichen Verlauf auf, die Flächen unter den ROC-Kurven lag bei den fünf Datensätzen im Bereich [0.837,0.840].

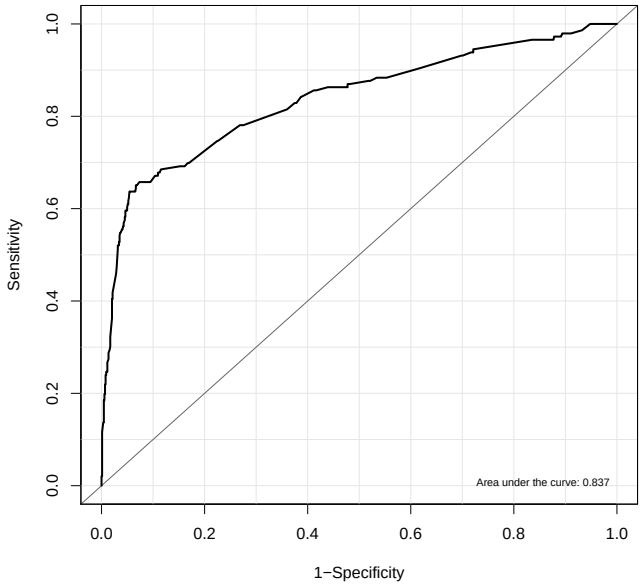

Abbildung 8.3: Beispielhafte ROC-Kurve für das Modell mit der Zielgröße Allergische Rhinitis in SOLAR II

8.6.2 Kombination der Schätzer des finalen Modells für Allergische Rhinits

Nachdem nun das finale Modell auf allen fünf Datensätzen gerechnet wurde, fanden die Kombinationsregeln Anwendung, die von Donald Rubin im Zusammenhang mit der multiplen Imputation entwickelt wurden (vgl. Kapitel 4). Die kombinierten Schätzer für die einzelnen β-Koeffizienten wurden dabei als arithmetisches Mittel aus den entsprechenden β-Koeffizienten der fünf Datensätze berechnet. Die Varianz der jeweiligen Schätzer setzte sich dabei aus der Varianz innerhalb jedes imputierten Datensatzes und der Varianz zwischen den fünf imputierten Datensätzen zusammen.
Exponiert man die kombinierten Parameterschätzer, so erhält man die (kombinierten) Odds-Ratios. Darauf aufbauend konnten 95%-Konfidenzintervalle berechnet werden:

$$95\%\text{-KI} = [\exp(\widehat{\beta} - 1.96\ \widehat{\sigma}),\ \exp(\widehat{\beta} + 1.96\ \widehat{\sigma})]$$

Folgende Abbildung 8.4 gibt einen Überblick über die Kombination der Schätzer und das weitere Vorgehen bei der Analyse, Interpretation und Diskussion der Ergebnisse, bei der ausschließlich das finale (kombinierte) Modell betrachtet wird.

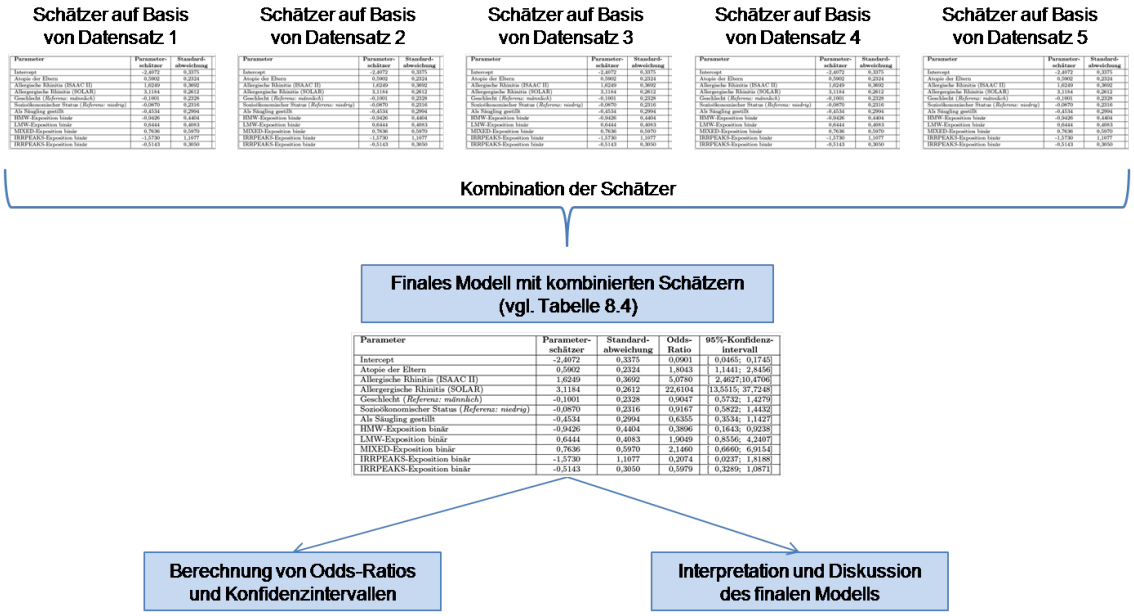

Abbildung 8.4: Übersicht über die Kombination der Schätzer und das weitere Vorgehen bei der Analyse

Einen ersten Überblick über die (kombinierten) Odds-Ratios sollen die 95%-Konfidenzintervalle, die in Abbildung 8.5 dargestellt sind, liefern.

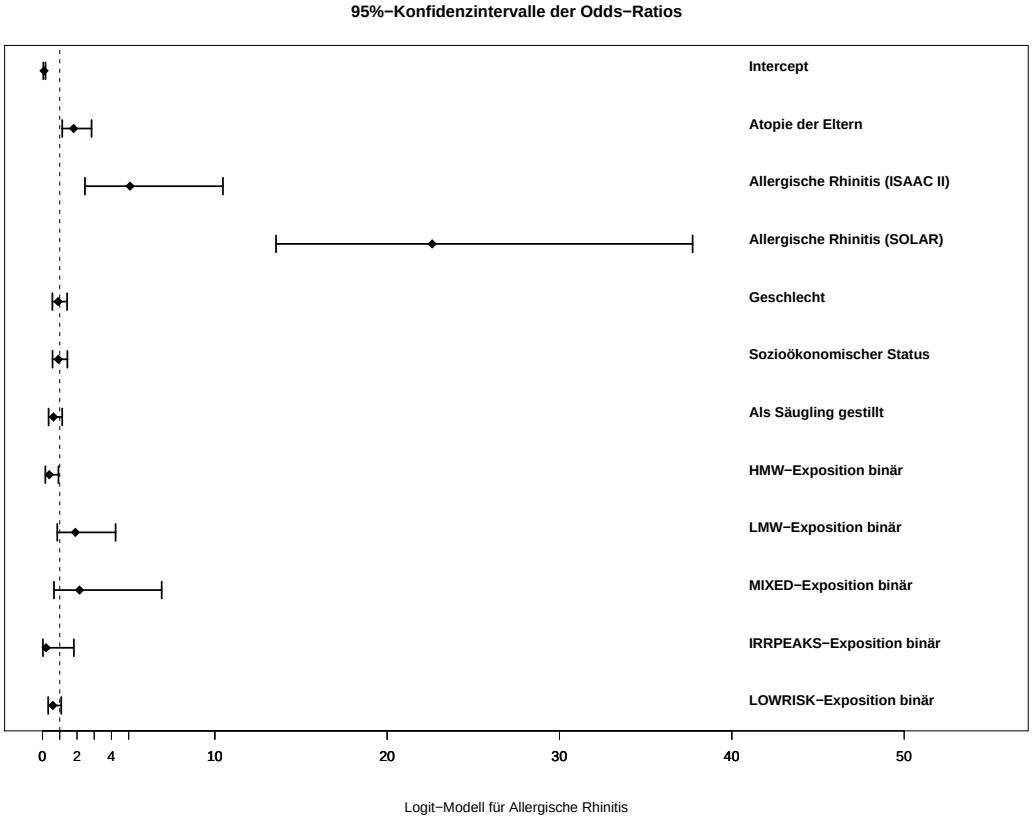

Abbildung 8.5: 95%-Konfidenzintervalle im Logit-Modell für Allergische Rhinitis (auf Basis der Probanden mit vollständigen Tätigkeitsangaben)

Tabelle 8.4 liefert für das finale Modell für Allergische Rhinitis eine ausführliche Übersicht über die kombinierten Parameterschätzer, die zugehörigen kombinierten geschätzten Standardabweichungen, die Odds-Ratios und die 95%-Konfidenzintervalle. [1]

Parameter	**Parameter-schätzer**	**Standard-abweichung**	**Odds-Ratio**	**95%-Konfidenz-intervall**
Intercept	-2,4072	0,3375	0,0901	[0,0465; 0,1745]
Atopie der Eltern	0,5902	0,2324	1,8043	[1,1441; 2,8456]
Allergische Rhinitis (ISAAC II)	1,6249	0,3692	5,0780	[2,4627;10,4706]
Allergische Rhinitis (SOLAR)	3,1184	0,2612	22,6104	[13,5515; 37,7248]
Geschlecht (*Referenz: männlich*)	-0,1001	0,2328	0,9047	[0,5732; 1,4279]
Sozioökonomischer Status (*Referenz: niedrig*)	-0,0870	0,2316	0,9167	[0,5822; 1,4432]
Als Säugling gestillt	-0,4534	0,2994	0,6355	[0,3534; 1,1427]
HMW-Exposition binär	-0,9426	0,4404	0,3896	[0,1643; 0,9238]
LMW-Exposition binär	0,6444	0,4083	1,9049	[0,8556; 4,2407]
MIXED-Exposition binär	0,7636	0,5970	2,1460	[0,6660; 6,9154]
IRRPEAKS-Exposition binär	-1,5730	1,1077	0,2074	[0,0237; 1,8188]
LOWRISK-Exposition binär	-0,5143	0,3050	0,5979	[0,3289; 1,0871]

Tabelle 8.4: Finales Modell für Allergische Rhinitis (auf Basis der Probanden mit vollständigen Tätigkeitsangaben): Kombinierte Parameterschätzer, Standardabweichungen, Odds-Ratios und 95%-Konfidenzintervalle

[1] In den Tabellen werden einheitlich 4 Nachkommastellen dargestellt, um bei den Expositionsvariablen eine Tendenz erkennen zu können. (Dadurch sollen keine Aussagen über die Schätzgenauigkeit getroffen werden.)

8.6.3 Interpretation des finalen Modells für Allergische Rhinitis auf Basis der Probanden mit vollständigen Tätigkeitsangaben

Verwendet man zur Interpretation des finalen Modells auf Basis der Probanden mit vollständigen Tätigkeitsangaben die (kombinierten) Odds-Ratios, so lassen sich aus der zuvor dargestellten Tabelle folgende Aussagen generieren.
(Zur Interpretation des Odds-Ratios vgl. das entsprechende Beispiel in Abschnitt 8.3).
Die Chance zum Zeitpunkt der SOLAR II-Studie (2007/2009) an Allergischer Rhinitis zu leiden...

- ... ist für Personen, von denen mindestens ein Elternteil bereits eine Atopie (d.h. Neurodermitis, Allergische Rhinitis oder Asthmaerkrankung) aufwies, knapp doppelt so hoch (Faktor 1,8) als die Chance für Personen, deren Eltern keine Atopie aufwiesen.

- ... ist für Personen, die bereits bis zum Zeitpunkt der ISAAC II-Studie (1995/96) an Allergischer Rhinitis erkrankt waren, fünf mal so hoch (Faktor 5,1) als die Chance für Personen, die zum damaligen Zeitpunkt keine Allergische Rhinitis aufwiesen.

- ... ist für Personen, die bereits zum Zeitpunkt der SOLAR-Studie (2002/03) Allergische Rhinitis hatten, etwas mehr als 22 mal so hoch (Faktor 22,6) als die Chance für Personen, die damals keine Allergische Rhinitis aufwiesen.
  Bei diesem Parameter ist allerdings zu beachten, dass die Schätzung mit einer relativ großen Unsicherheit behaftet war, da die Spannweite des Konfidenzintervalls einen relative großen Wert aufwies. Der absolute Schätzwert ist somit mit Vorsicht zu behandeln. Allerdings kann mit 95%-igen Sicherheit festgelegt werden, dass die Chance auf Allergische Rhinits in SOLAR II für Personen, die bereits in der Vorstudie an dieser Atopie litten, zwischen 14 und 38 mal so hoch ist (im Vergleich zur Chance von Personen ohne dieser Atopie).

Die Variablen “Geschlecht” und “Sozioökonomischer Status” wurden aufgrund von Vorüberlegungen und Hypothesen in dem Modell belassen. Ein signifikanter Unterschied zwischen Mädchen und Jungen bzw. zwischen niedrigem und hohem Sozialstatus konnte in Bezug auf Allergische Rhinitis nicht nachgewiesen werden.
Ob tatsächlich ein Unterschied zwischen als Säugling gestillten und nicht gestillten Probanden in Bezug auf das Auftreten von allergischer Rhinitis besteht, konnte nicht abschließend geklärt werden. Da es sich bei dem finalen Modell um ein aus fünf Datensätzen kombiniertes Modell handelte, konnte es vorkommen, dass eine Variablen auf einem der Datensätze einen statistisch signifikanten Einfluss hatte, der Effekt allerdings auf den anderen Datensätzen und letztendlich auch im kombinierten Modell verschwand. Dies war bei der Variable “Als Säugling gestillt” der Fall. Da man allerdings eher eine zusätzliche (evtl. irrelevante) Variable in das finale Modell aufnehmen möchte, als eine möglicherweise relevante Variable im Modell nicht zu berücksichtigen, entschied man sich bis auf Weiteres für die Aufnahme dieser Variable.

8.6.4 Logit-Modell 2: Asthma auf Basis der Probanden mit vollständigen Tätigkeitsangaben

Variablenselektion und Wahl eines Confoundermodells

Als Basis für die Variablenselektion wurde als minimales Modell erneut das Modell mit Intercept und den Einflussgrößen Geschlecht und Sozioökonomischer Status definiert. Als maximales Modell standen zusätzlich zum Intercept insgesamt 19 Variablen als potenzielle Confoundervariablen zur Verfügung. Die Variable "In Deutschland geboren" musste aus der ursprünglichen Liste der möglichen 20 Confoundervariablen entfernt werden, da alle Personen, die nicht in Deutschland geboren waren (n = 53), keine Asthmaerkrankung in SOLAR II aufwiesen. Würde man diese Variable aufnehmen, so wären die Daten trennbar, was eine Divergenz des ML-Schätzers zur Folge hätte. Um den Einfluss dieser Variable untersuchen zu können, wären weitere Daten von nicht in Deutschland geborenen Personen nötig.
Auf jedem der fünf Datensätze wurde die Variablenselektion durch Schrittweise-Selektion auf Basis des AIC-Kriteriums durchgeführt. Die folgende Tabelle 8.5 liefert eine Übersicht über die Variablen, die zusätzlich zu dem minimalen Modell mit Intercept, Geschlecht und sozioökonomischer Status selektiert wurden.

Daten	Asthma (I)	Asthma (S)	Neuroder-mitis (S)	Allergische Rhinitis (S)	Rauchen (S)	Passivrauch (S II)	Neuroder-mitis (I)
1	+	+	+	+	+	-	-
2	+	+	+	+	+	-	-
3	+	+	+	+	+	+	+
4	+	+	+	+	+	+	+
5	+	+	+	+	+	+	+

Tabelle 8.5: Asthma-Modelle auf Basis der Probanden mit vollständigen Tätigkeitsangaben: Selektierte Confoundervariablen (zusätzlich zu Geschlecht und Sozioökonomischer Status)
(Abkürzungen: I: ISAAC II, S: SOLAR, S II: SOLAR II)

Folglich standen zwei potenzielle Confoundermodelle zur Auswahl:

- **Confoundermodell A**:
  Einflussgrößen: Geschlecht, Sozioökonomischer Status, Asthma (ISSAC II), Asthma (SOLAR), Neurodermitis (SOLAR), Allergische Rhinits (SOLAR), Rauchen (SOLAR)

- **Confoundermodell B**:
  Einflussgrößen: Geschlecht, Sozioökonomischer Status, Asthma (ISSAC II), Asthma (SOLAR), Neurodermitis (SOLAR), Allergische Rhinits (SOLAR), Rauchen (SOLAR), Passivrauch (SOLAR II), Neurodermitis (ISAAC II)

Um auch für Asthma ein einziges Confoundermodell auswählen zu können, wurden auf jedem der fünf Datensätze Likelihood-Ratio-Tests der beiden Modellen durchgeführt. Da das größere Modell B auf keinem der fünf Datensätze eine Verbesserung gegenüber des kleineren Modells A lieferte, fiel die Entscheidung auf das kleinere Modell.

Einflussgrößen des gewählten Confoundermodells für Asthma auf Basis der Probanden mit vollständigen Tätigkeitsangaben
Geschlecht, Sozioökonomischer Status, Asthma (ISSAC II), Asthma (SOLAR), Neurodermitis (SOLAR), Allergische Rhinits (SOLAR), Rauchen (SOLAR)

Aufnahme von Expositionsvariablen in das Confoundermodell

Im Rahmen des Asthma-Modells wurde zunächst erneut anhand eines GAMs geprüft, ob die kumulierten Expositionsvariablen als lineare Terme in die Logitmodelle eingehen könnten. Beispielhaft wurde hier der Datensatz 1 ausgewählt und die geschätzten Funktionen für die kumulierten Expositionen während der ersten Tätigkeit in Abbildung 8.6 dargestellt. Für die fünfte Einflussgröße (IRRPEAKS) kann hier keine Funktion geschätzt werden, da nur vier Personen dieser Belastung im ersten Jahr ausgesetzt waren. Ähnliche Verläufe wiesen die geschätzten Funktion der anderen vier Datensätze auf.

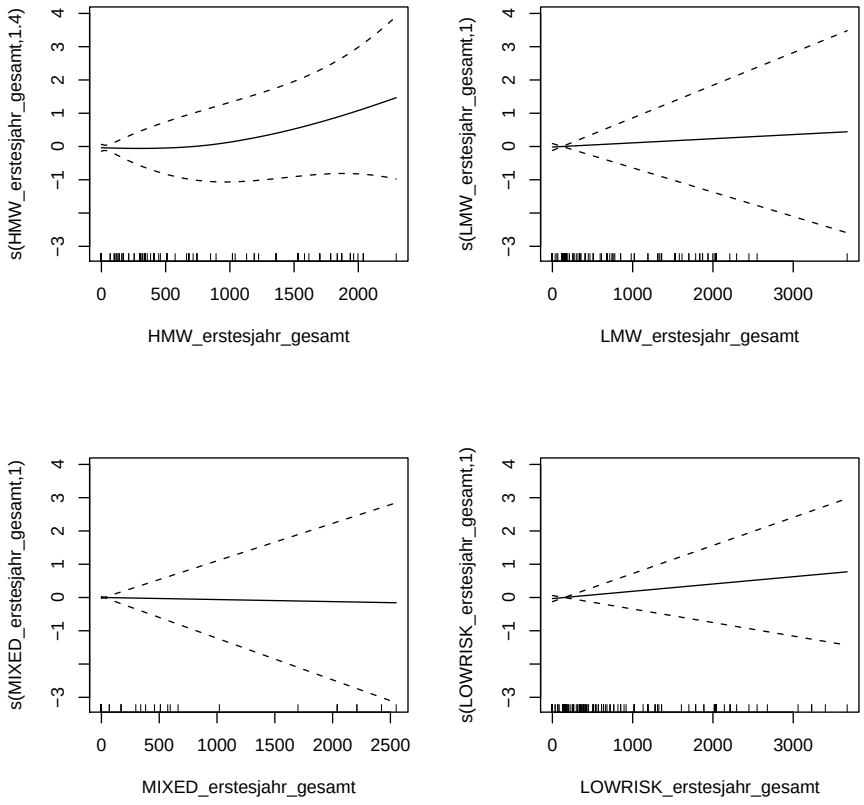

Abbildung 8.6: Geschätzter Funktionsverlauf des Einflusses der kumulierten Expositionsvariablen während des ersten Tätigkeitsjahres auf die Zielgröße Asthma unter Anwendung eines GAMs

Wie auf der Abbildung deutlich zu erkennen, konnten die Expositionsvariablen LMW, MIXED und LOWRISK des ersten Tätigkeitsjahres als lineare Terme in die Logitmodelle aufgenommen werden. Der Funktionsverlauf von HMW im Zusammenhang mit dem ersten Jahr wies allerdings eher auf einen quadratischen Einfluss hin. Aus diesem Grund wurde diese Exposition sowohl linear als auch quadratisch aufgenommen.
Die kumulierten Expositionsvariablen (über alle Tätigkeiten und Jahre hinweg bzw. während der 1. Tätigkeit) konnten linear aufgenommen werden.

Die Durchführung der Likelhood-Ratio-Tests lieferte auf jedem Datensatz das Ergebnis, dass die linearen Terme der kumulierten Expositionsvariablen über alle Tätigkeiten und Jahre hinweg zu einer Verbesserung des Confoundermodells führten. Die Tabelle 8.6 liefert einen Überblick über die p-Werte der durchgeführten Likelihood-Ratio-Tests. Aus diesem Grund wurden zusätzlich zu den gewählten Confoundervariablen die kumulierten Expositionsvariablen für alle fünf Kategorien aufgenommen.

Likelihood-Ratio-Test Confoundermodell vs. Modell inklusive ...	**p-Wert** Datensatz 1	**p-Wert** Datensatz 2	**p-Wert** Datensatz 3	**p-Wert** Datensatz 4	**p-Wert** Datensatz 5
kumulierte Expositionen	**0,04**	**0,04**	**0,04**	**0,04**	**0,04**
binäre Expositionen	0,38	0,38	0,38	0,39	0,38
Expositionen des 1. Tätigkeitsjahres	0,32	0,31	0,31	0,31	0,31
Expositionen des 1. Jahres (inkl. HMW als quadratischer Term)	0,22	0,21	0,21	0,21	0,21
binäre Expositionen des 1. Jahres	0,90	0,90	0,90	0,90	0,90
Expositionen der 1. Tätigkeit	0,15	0,15	0,15	0,15	0,15
binäre Expositionen der 1. Tätigkeit	0,79	0,79	0,79	0,79	0,80

Tabelle 8.6: Übersicht über die p-Werte der durchgeführten Likelihood-Ratio-Tests

Analyse des finalen Modells für Asthma

Als finales Modell für Asthma wurde folgendes Modell ausgewählt:

Einflussgrößen des finalen Modells für Asthma auf Basis der Probanden mit vollständigen Tätigkeitsangaben
Geschlecht, Sozioökonomischer Status, Asthma (ISSAC II), Asthma (SOLAR), Neurodermitis (SOLAR), Allergische Rhinits (SOLAR), Rauchen (SOLAR), HMW-Exposition kumuliert, LMW-Exposition kumuliert, MIXED-Exposition kumuliert, IRRPEAKS-Exposition kumuliert, LOWRISK-Exposition kumuliert

Das finale Modell wurde durch ROC-Kurven auf allen fünf Datensätzen analysiert. Die ROC-Kurve für den Datensatz 1 ist Abbildung 8.7 zu entnehmen. Die anderen vier Datensätze wiesen bei den ROC-Kurven einen sehr ähnlichen Verlauf auf, die Flächen unter den ROC-Kurven lagen im Bereich [0.882,0.887].

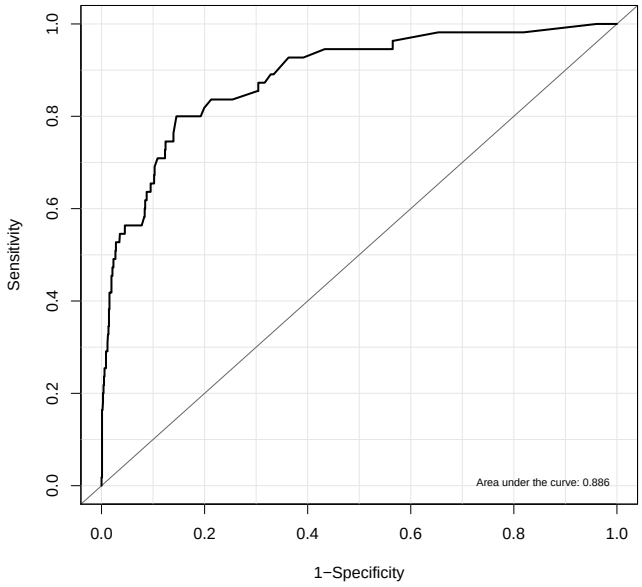

Abbildung 8.7: Beispielhafte ROC-Kurve für das Modell mit der Zielgröße Asthma in SOLAR II

8.6.5 Kombination der Schätzer des finalen Modells für Asthma

Die Parameterschätzer und Varianzen des finalen Modells aller fünf Datensätze wurden gemäß der Kombinationsregeln (vgl. Kapitel 4) zusammengefasst. Durch Exponieren der kombinierten Parameterschätzer erhielt man die (kombinierten) Odds-Ratios. Darauf aufbauend konnten die 95%-Konfidenzintervalle berechnet werden:

$$95\%\text{-KI} = [\exp(\widehat{\beta} - 1.96\ \widehat{\sigma}),\ \exp(\widehat{\beta} + 1.96\ \widehat{\sigma})]$$

In Abbildung 8.8 sind diese Intervalle dargestellt.

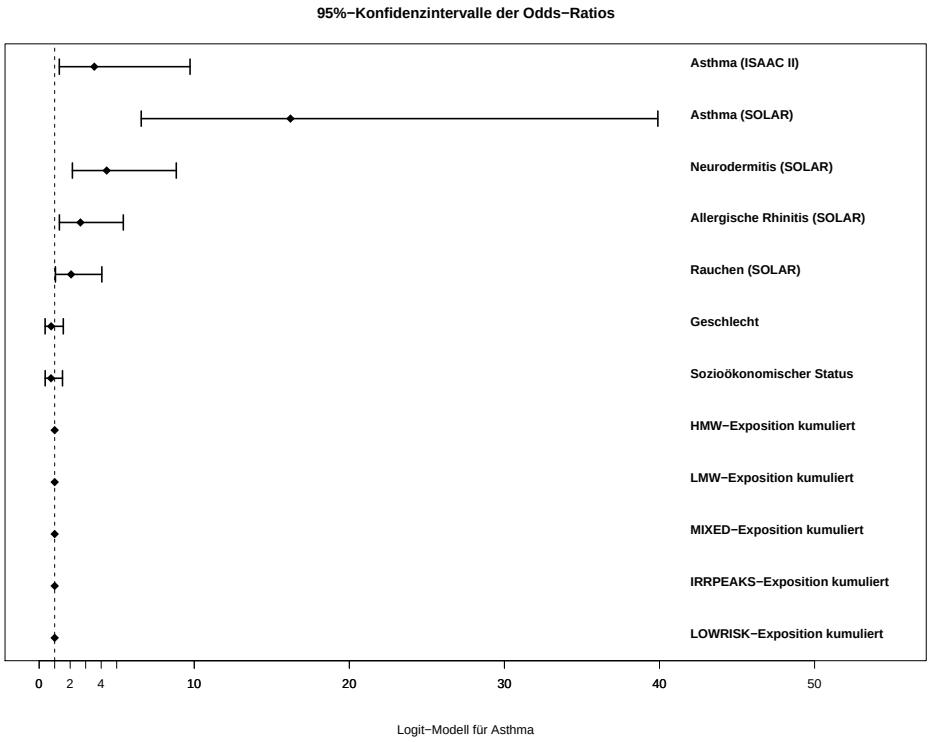

Abbildung 8.8: 95%-Konfidenzintervalle im Logit-Modell für Asthma (auf Basis der Probanden mit vollständigen Tätigkeitsangaben)

Für das finale Modell für Asthma liefert Tabelle 8.7 eine ausführliche Übersicht über die kombinierten Parameterschätzer, die zugehörigen kombinierten geschätzten Standardabweichungen, die Odds-Ratios und die 95%-Konfidenzintervalle. [2]

Parameter	**Parameter-schätzer**	**Standard-abweichung**	**Odds-Ratio**	**95%-Konfidenz-intervall**
Intercept	-4,0788	0,4155	0,0169	[0,0075; 0,0382]
Asthma (ISAAC II)	1,2685	0,5136	3,5557	[1,2994; 9,7299]
Asthma (SOLAR)	2,7853	0,4597	16,2040	[6,5816; 39,8943]
Neurodermitis (SOLAR)	1,4710	0,3615	4,3535	[2,1435; 8,8421]
Allergische Rhinitis (SOLAR)	0,9787	0,3635	2,6611	[1,3050; 5,4266]
Rauchen (SOLAR)	0,7229	0,3439	2,0605	[1,0502; 4,0427]
Geschlecht (*Referenz: männlich*)	-0,2562	0,3568	0,7740	[0,3846; 1,5574]
Sozioökonomischer Status (*Referenz: niedrig*)	-0,2687	0,3456	0,7644	[0,3883; 1,5078]
HMW-Exposition kumuliert (pro Stunde)	0.0003	0.0001	1.0003	[1.0000; 1.0005]
LMW-Exposition kumuliert (pro Stunde)	-0,0001	0,0001	0,9999	[0,9997; 1,0002]
MIXED-Exposition kumuliert (pro Stunde)	-0,0000	0,0002	1,0000	[0,9997; 1,0003]
IRRPEAKS-Exposition kumuliert (pro Stunde)	-0,0008	0,0004	1,0008	[0,9999; 1,0016]
LOWRISK-Exposition kumuliert (pro Stunde)	-0,0000	0,0001	1,0000	[0,9998; 1,0002]

Tabelle 8.7: Finales Modell für Asthma (auf Basis der Probanden mit vollständigen Tätigkeitsangaben): Kombinierte Parameterschätzer, Standardabweichungen, Odds-Ratios und 95%-Konfidenzintervalle

8.6.6 Interpretation des finalen Modells für Asthma auf Basis der Probanden mit vollständigen Tätigkeitsangaben

Zur Interpretation des Asthma-Modells auf Basis der Probanden mit vollständigen Tätigkeitsangaben wurden ebenfalls die (kombinierten) Odds-Ratios verwendet. Aus der zuvor dargestellten Tabelle lassen sich folgende Aussagen ableiten.
Die Chance zum Zeitpunkt der SOLAR II-Studie an Asthma zu leiden...

- ... ist für Personen, die bereits zum Zeitpunkt von ISAAC II (1995/96) Asthma hatten, knapp 4 mal so hoch (Faktor 3,6) als die Chance für Personen, die zum damaligen Zeitpunkt kein Asthma hatten.

- ... ist für Personen, die zum Zeitpunkt der SOLAR-Studie (2002/03) an Asthma erkrankt waren, 16 mal so hoch (Faktor 16,2) als die Chance für Personen, die damals kein Asthma hatten.
  Da die Spannweite des Konfidenzintervalls einen relativ großen Wert aufwies, ist die Schätzung dieses Parameters mit einer relativ großen Unsicherheit behaftet. Folglich sollte der absolute Schätzwert mit Vorsicht behandelt werden. Allerdings konnte mit 95%-igen Sicherheit folgende Aussage getroffen werden: die Chance einer Asthmaerkrankung in SOLAR II ist bei einer Person, die bereits in der vorhergehenden Studie an Asthma litt, zwischen 7 und 40 mal so hoch, als die Chance bei einer zuvor gesunden Person.

[2] In den Tabellen werden einheitlich 4 Nachkommastellen dargestellt, um bei den Expositionsvariablen eine Tendenz erkennen zu können. (Dadurch sollen keine Aussagen über die Schätzgenauigkeit getroffen werden.)

- ... ist für Personen, die bereits zum Zeitpunkt der SOLAR-Studie (2002/03) an Neurodermitis erkrankt waren, 4 mal so hoch (Faktor 4,4) als die Chance bei Personen, die damals kein Neurodermitis hatten.
- ... ist für Personen, die während der SOLAR-Studie Allergische Rhinitis hatten, knapp 3 mal so hoch (Faktor 2,7) als die Chance für Personen, die damals keine Allergische Rhinitis aufwiesen.
- ... ist für Personen, die zum Zeitpunkt der SOLAR-Studie rauchten, doppelt so hoch (Faktor 2,1) als die Chance für Personen, die damals Nichtraucher waren.

Aufgrund von substanzwissenschaftlichen Überlegungen und Hypothesen wurden die Variablen "Geschlecht" und "Sozioökonomischer Status" im Modell belassen. Ein statistisch signifikanter Unterschied in Bezug auf Asthma konnte zwischen Mädchen und Jungen bzw. zwischen niedrigem und hohem Sozialstatus nicht nachgewiesen werden.
Bei genauerer Analyse der kumulierten Expositionen stellte sich heraus, dass die fünf Variablen in die Modelle aufgenommen wurden, da vor allem bei den Expositionsgruppen HMW und IRRPEAKS auf Basis der einzelnen Datensätze ein relevanter Einfluss nachgewiesen werden konnte. Dies erschien (v.a. im Hinblick auf IRRPEAKS) plausibel: Je mehr Stunden man in einer Tätigkeit arbeitete, bei der es zu einer Spitzenexposition kommen konnte, umso größer war die Wahrscheinlichkeit, dass solch eine Exposition tatsächlich auftrat und dadurch das Asthmarisiko der jeweiligen Person erhöht wurde.
Weiterhin ist zu beachten, dass ein Odds-Ratio in der Nähe von 1, wie dies bei den Expositionsvariablen der Fall war, zunächst sehr gering und irrelevant erscheinen mag. Allerdings ist dies keineswegs der Fall, da es sich bei den Expositionsvariablen um metrische Einflussgrößen handelt. Der Wert des Odds-Ratios musste in diesem Fall mit der konkreten Stundenanzahl der Variable exponiert werden.
Beispiel: Eine Person A, die zwei Jahre lang einen Beruf mit HMW-Exposition für 40 Stunden pro Woche ausübte, war folglich der Belastung für insgesamt 4.080 Stunden ausgesetzt (4.080 Stunden = 2 × 12 Monate × 4,25 Wochen pro Monat × 40 Stunden pro Woche). Vergleicht man die Person A, mit einer Person B, die keiner HMW-Exposition ausgesetzt war (unter sonst gleichen Voraussetzungen), so ist das Asthmarisiko von Person A um den Faktor 3,4 ($1,0003^{4.080} = 3,4$) höher als bei Person B.

8.7 Logistische Regressionsmodelle für alle Probanden

Nachdem nun die Modelle auf Basis der Probanden mit vollständigen Tätigkeitsangaben ausführlich analysiert wurden, sollen im nächsten Schritt Modelle auf Basis aller Probanden berechnet werden. Um diesen Schritt zu ermöglichen, war die Imputation der fehlenden Tätigkeitsangaben bei 93 Probanden nötig. Dabei wurde für jeden der fünf Datensätze, in denen die fehlenden Werte der Confoundervariablen bereits in einem früheren Schritt imputiert wurden, in einem zweiten Schritt jeweils für jeden Datensatz die Tätigkeitsangaben imputiert (Details zur Imputation der fehlenden Tätigkeitsangaben vergleiche 6.2). Somit standen auch im Folgenden fünf Datensätze zur Verfügung, auf denen jeweils zwei logistische Regressionsmodelle angepasst werden konnten:

- **Logit-Modell 1** - Zielgröße: Allergische Rhinitis in SOLAR II
  Datenbasis: alle Probanden, die während ISAAC II und/oder SOLAR kein Asthma hatten (n = 1.118)
- **Logit-Modell 2** - Zielgröße: Asthma in SOLAR II
  Datenbasis: alle Probanden (n = 1.187)

Das Vorgehen bei der Modellwahl entspricht dem zuvor ausführlich beschriebenen Vorgehen. Die Ergebnisse werden aus diesem Grund in komprimierterer Form dargestellt.

8.7.1 Logit-Modell 1: Allergische Rhinitis auf Basis aller Probanden

Variablenselektion und Wahl eines Confoundermodells

Für die Wahl eines Confoundermodells wurde die Variablenselektion (erneut aus 18 Variablen) durch Schrittweise-Selektion auf Basis des AIC-Kriteriums durchgeführt. Welche Variablen zusätzlich zu den Variablen Geschlecht und Sozioökonomischer Status in die jeweiligen Modelle aufgenommen wurden, stellt Tabelle 8.8 dar.

Daten	Atopie der Eltern	Allergische Rhinitis (I)	Allergische Rhinitis (S)	Als Säugling gestillt	Passivrauch (S)	Geschwister
1	+	+	+	-	+	+
2	+	+	+	+	+	+
3	+	+	+	-	+	+
4	+	+	+	+	+	-
5	+	+	+	+	+	+

Tabelle 8.8: Allergische Rhinitis-Modelle auf Basis aller Probanden: Selektierte Confoundervariablen (zusätzlich zu Geschlecht und Sozioökonomischer Status) (Abkürzungen: I: ISAAC II, S: SOLAR, S II: SOLAR II)

Folgende drei mögliche Confoundermodelle stehen zur Auswahl:

- **Confoundermodell A**:
  Einflussgrößen: Geschlecht, Sozioökonomischer Status, Atopie der Eltern, Allergische Rhinits (ISAAC II), Allergische Rhinitis (SOLAR), als Säugling gestillt, Passivrauch (SOLAR)
- **Confoundermodell B**:
  Einflussgrößen: Geschlecht, Sozioökonomischer Status, Atopie der Eltern, Allergische Rhinits (ISAAC II), Allergische Rhinitis (SOLAR), als Säugling gestillt, Passivrauch (SOLAR), Geschwister
- **Confoundermodell C**:
  Einflussgrößen: Geschlecht, Sozioökonomischer Status, Atopie der Eltern, Allergische Rhinits (ISAAC II), Allergische Rhinitis (SOLAR), Passivrauch (SOLAR), Geschwister

Unter Anwendung von Likelihood-Ratio-Tests wurde schlußendlich das Confoundermodell A ausgewählt.

Einflussgrößen des gewählten Confoundermodells für Allergische Rhinitis auf Basis aller Probanden
Geschlecht, Sozioökonomischer Status, Atopie der Eltern
Allergische Rhinits (ISAAC II), Allergische Rhinitis (SOLAR), als Säugling gestillt
Passivrauch (SOLAR)

Aufnahme von Expositionsvariablen in das Confoundermodell

Die Likelihood-Ratio-Tests zur Aufnahme der Expositionsvariablen sprachen (aus statistischer Sicht) alle für eine Beibehaltung des Confoundermodells. Tabelle 8.9 fasst die p-Werte der durchgeführten Likelihood-Ratio-Tests zusammen.

Likelihood-Ratio-Test Confoundermodell vs. Modell inklusive ...	**p-Wert** Datensatz 1	**p-Wert** Datensatz 2	**p-Wert** Datensatz 3	**p-Wert** Datensatz 4	**p-Wert** Datensatz 5
kumulierte Expositionen	0,34	0,30	0,36	0,34	0,35
binäre Expositionen	0,36	0,35	0,35	0,35	0,36
Expositionen des 1. Tätigkeitsjahres	0,68	0,54	0,71	0,63	0,75
binäre Expositionen des 1. Jahres	0,45	0,46	0,45	0,43	0,45
Expositionen der 1. Tätigkeit	0,36	0,21	0,32	0,32	0,34
binäre Expositionen der 1. Tätigkeit	0,79	0,80	0,79	0,79	0,80

Tabelle 8.9: Übersicht über die p-Werte der durchgeführten Likelihood-Ratio-Tests

Aus inhaltlichen Gründen war es allerdings nötig, Expositionsvariablen in das finale Modell aufzunehmen. Welche Expositionsvariablen aufgenommen wurden, wurde auf Basis des AIC-Kriteriums entschieden. Die Tabelle 8.10 liefert eine Übersicht über die AICs der verschiedenen Modelle, wobei pro Datensatz für die Modelle mit Exposition das Modell mit dem kleinsten AIC hervorgehoben ist.

Modell	**AIC** Datensatz 1	**AIC** Datensatz 2	**AIC** Datensatz 3	**AIC** Datensatz 4	**AIC** Datensatz 5
Confoundermodell (ohne Expositionsvariablen)	638,25	634,48	637,40	635,71	636,40
Confoundermodell inkl. kumulierte Expositionen	**642,57**	638,37	641,88	640,02	640,80
Confoundermodell inkl. binäre Expositionen	642,73	638,93	641,82	640,11	640,93
Confoundermodell inkl. Expositionen des 1. Tätigkeitsjahres	645,12	640,39	644,44	642,28	643,70
Confoundermodell inkl. binäre Expositionen des 1. Jahres	643,54	639,80	642,67	640,86	641,68
Confoundermodell inkl. Expositionen der 1. Tätigkeit	642,75	**637,35**	**641,52**	**639,83**	**640,70**
Confoundermodell inkl. binäre Expositionen der 1. Tätigkeit	645,84	642,15	645,02	643,31	644,03

Tabelle 8.10: Übersicht über die AICs der unterschiedlichen Modelle

Zu beachten ist, dass die AIC-Werte der Modelle inklusive kumulierter Exposition, inklusive binärer Exposition und inklusive Expositionen der 1. Tätigkeit sehr eng zusammenliegen. In 4 von 5 Datensätzen liefert das Modell inklusive der Expositionen der 1. Tätigkeit das geringste AIC. Zusätzlich ist die Erhebung der Exposition in der 1. Tätigkeit mit weniger Aufwand verbunden, als die Erhebung der (kumulierten) Expositionen über alle Jahre und Tätigkeiten hinweg. Somit wurde als finales Modell das Confoundermodell inklusive der Expositionen der 1. Tätigkeit (HMW-Exposition 1. Tätigkeit, LMW-Exposition 1. Tätigkeit, MIXED-Exposition 1. Tätigkeit, IRRPEAKS-Exposition 1. Tätigkeit und LOWRISK-Exposition 1. Tätigkeit) ausgewählt.

Analyse des finalen Modells

Als finales Modell für Allergische Rhinits wurde folgendes Modell ausgewählt:

Einflussgrößen des finalen Modells für Allergische Rhinitis auf Basis aller Probanden
Geschlecht, Sozioökonomischer Status, Atopie der Eltern Allergische Rhinits (ISAAC II), Allergische Rhinitis (SOLAR), als Säugling gestillt, Passivrauch (SOLAR), HMW-Exposition 1. Tätigkeit, LMW-Exposition 1. Tätigkeit, MIXED-Exposition 1. Tätigkeit, IRRPEAKS-Exposition 1. Tätigkeit, LOWRISK-Exposition 1. Tätigkeit

Das finale Modell wurde anhand von ROC-Kurven analysiert. Beispielhaft wird die ROC-Kurve des ersten Datensatzes in Abbildung 8.9 dargestellt. Die ROC-Kurven, die alle einen sehr ähnlichen Verlauf aufweisen, begrenzen eine Fläche, die im Bereich [0.842,0.844] liegt.

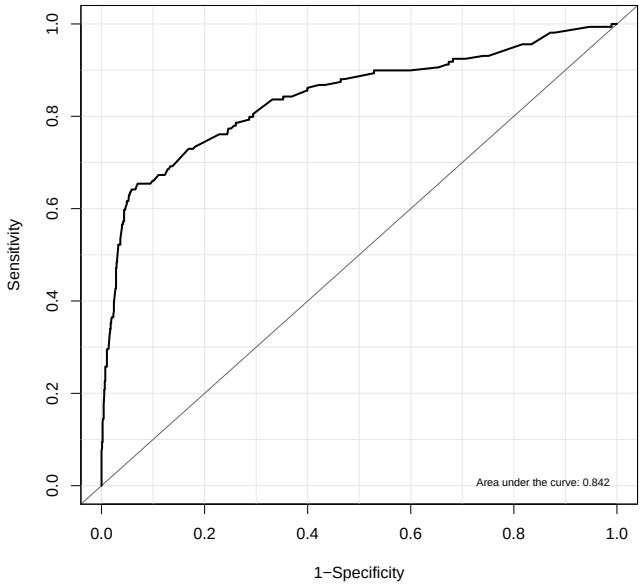

Abbildung 8.9: Beispielhafte ROC-Kurve für das Modell mit der Zielgröße Allergische Rhinitis in SOLAR II

8.7.2 Kombination der Schätzer des finalen Modells

Die Ergebnisse des finalen Modells auf allen fünf Datensätzen wurden kombiniert. Die dadurch erhaltenen kombinierten Parameterschätzer, die zugehörigen kombinierten geschätzten Standardabweichungen, die Odds-Ratios und die 95%-Konfidenzintervalle wurden in Tabelle 8.11 zusammengestellt. [3]

Parameter	**Parameter-schätzer**	**Standard-abweichung**	**Odds-Ratio**	**95%-Konfidenz-intervall**
Intercept	-2,1744	0,3543	0,1137	[0,0568; 0,2277]
Atopie der Eltern	0,5446	0,2233	1,7239	[1,1129; 2,6704]
Allergische Rhinitis (ISAAC II)	1,6206	0,3626	5,0561	[2,4842; 10,2908]
Allergische Rhinitis (SOLAR)	3,0999	0,2479	22,1966	[13,6538; 36,0846]
Geschlecht (*Referenz: männlich*)	-0,0277	0,2250	0,9727	[0,6258; 1,5117]
Sozioökonomischer Status (*Referenz: niedrig*)	-0,1910	0,2250	0,8261	[0,5315; 1,2841]
Als Säugling gestillt	-0,4603	0,2902	0,6311	[0,3573; 1,1146]
Passivrauch (SOLAR)	-0,4320	0,2241	0,6492	[0,4184; 1,0073]
HMW-Exposition 1.Tätigkeit	-0,0001	0,0002	0,9999	[0,9995; 1,0004]
LMW-Exposition 1.Tätigkeit	-0,0003	0,0002	0,9997	[0,9994; 1,0001]
MIXED-Exposition 1.Tätigkeit	0,0003	0,0004	1,0003	[0,9996; 1,0010]
IRRPEAKS-Exposition 1.Tätigkeit	-0,0003	0,0006	0,9997	[0,9985; 1,0009]
LOWRISK-Exposition 1.Tätigkeit	0,0000	0,0001	1,0000	[0,9999; 1,0002]

Tabelle 8.11: Finales Modell für Allergische Rhinitis auf Basis aller Probanden: Kombinierte Parameterschätzer, Standardabweichungen, Odds-Ratios und 95%-Konfidenzintervalle

[3] In den Tabellen werden einheitlich 4 Nachkommastellen dargestellt, um bei den Expositionsvariablen eine Tendenz erkennen zu können. (Dadurch sollen keine Aussagen über die Schätzgenauigkeit getroffen werden.)

8.7.3 Interpretation des finalen Modells für Allergische Rhinitis auf Basis aller Probanden

Aus der zuvor dargestellten Tabelle lassen sich auf Basis der (kombinierten) Odds-Ratios folgende Aussagen auf Basis aller Probanden generieren.
Die Chance in SOLAR II an Allergischer Rhinitis zu leiden...

- ... ist für Personen, von denen mindestens ein Elternteil bereits eine Atopie (d.h. Neurodermitis-, Allergische Rhinitis- oder Asthmaerkrankung) aufwies, knapp doppelt so hoch (Faktor 1,7) als die Chance für Personen, deren Eltern keine Atopie aufwiesen.

- ... ist für Personen, die bereits während der Zeit, in der die Studie ISAAC II durchgeführt wurde (1995/96), an Allergischer Rhinitis erkrankt waren, etwa fünf mal so hoch (Faktor 5,1) als die Chance für Personen, die zum damaligen Zeitpunkt keine Allergische Rhinitis aufwiesen.

- ... ist für Personen, die bereits während der Zeit, in der die SOLAR-Studie durchgeführt wurde (2002/03), Allergische Rhinitis hatten, etwa 22 mal so hoch (Faktor 22,2) als die Chance für Personen, die damals keine Allergische Rhinitis aufwiesen. Aufgrund der großen Spannweite des Konfidenzintervalls, sollte die Interpretation des absoluten Parameters mit Vorsicht behandelt werden. Vielmehr kann man mit 95%-iger Sicherheit aussagen, dass die Chance bei Personen mit zuvoriger Allergischer Rhinitis um einen Faktor zwischen 14 und 36 erhöht ist (im Vergleich zur Chance von zuvor nicht Erkrankten).

Wie bereits in den vorhergehenden Modellen, wurden auch hier aufgrund von substanzwissenschaftlichen Überlegungen und Hypothesen die Variablen “Geschlecht” und “Sozioökonomischer Status” im Modell belassen. Ein signifikanter Unterschied in Bezug auf Allergische Rhinitis konnte zwischen Mädchen und Jungen bzw. zwischen niedrigem und hohem Sozialstatus nicht nachgewiesen werden.
Weiterhin konnte bei den Variablen “Passivrauch” und “als Säugling gestillt” nicht abschließend geklärt werden, ob ein Einfluss auf das Auftreten von allergischer Rhinitis besteht. Da es sich bei dem finalen Modell um ein aus fünf Datensätzen kombiniertes Modell handelte, konnte es vorkommen, dass eine Variablen auf einem der Datensätze einen statistisch signifikanten Einfluss hatte, der Effekt allerdings auf den anderen Datensätzen und letztendlich auch im kombinierten Modell verschwand. Dies war bei diesen Variablen der Fall. Da man allerdings eher diese beiden zusätzlichen (evtl. irrelevanten) Variablen in das finale Modell aufnehmen möchte, als eine möglicherweise relevante Variable im Modell nicht zu berücksichtigen, entschied man sich bis auf Weiteres für die Aufnahme dieser Variablen.

Vergleich der Modelle für Allergische Rhinitis

Vergleicht man nun diese Modell auf Basis aller Probanden, d.h. auf Basis der durch Imputation vervollständigten Datensätze, mit dem vorherigen Modell, auf Basis der Probanden mit vollständigen Tätigkeitsangaben, so erhält man im Hinblick auf die Confoundervariablen ein sehr ähnliches Bild (vgl. Tabelle 8.12). [4]

Parameter	**Odds-Ratio** Modell auf Basis der Probanden mit vollständigen Tätigkeitsangaben	**Odds-Ratio** Modell auf Basis aller Probanden (inkl. imputierter Tätigkeitsangaben)
Intercept	0,0901	0,1137
Atopie der Eltern	1,8043	1,7239
Allergische Rhinitis (ISAAC II)	5,0780	5,0561
Allergische Rhinitis (SOLAR)	22,6104	22,1966
Geschlecht (*Referenz: männlich*)	0,9047	0,9727
Sozioökonomischer Status (*Referenz: niedrig*)	0,9167	0,8261
Als Säugling gestillt	0,6355	0,6311
Passivrauch	-	0,6492
HMW-Exposition binär	0,3896	-
LMW-Exposition binär	1,9049	-
MIXED-Exposition binär	2,1460	-
IRRPEAKS-Exposition binär	0,2074	-
LOWRISK-Exposition binär	0,5979	-
HMW-Exposition 1. Tätigkeit	-	0,9999
LMW-Exposition 1. Tätigkeit	-	0,9997
MIXED-Exposition 1. Tätigkeit	-	1,0003
IRRPEAKS-Exposition 1. Tätigkeit	-	0,9997
LOWRISK-Exposition 1. Tätigkeit	-	1,0000

Tabelle 8.12: Vergleich der Modelle für Allergische Rhinitis

Die Modelle unterscheiden sich nur gering in den einzelnen Odds-Ratios der Confoundervariablen. Allerdings enthält das nun erstellte Modell die zusätzliche Variable Passivrauch. Nun stellte sich die Frage, welches Modell sich für die weitere Anwendung am Besten eignet. Um diese Fragestellung beantworten zu können, wurde ein Likelihood-Ratio-Test durchgeführt. Das Ergebnis dieses Tests war, dass sich das minimale Confoundermodell (ohne die Variable Passivrauch), dass auf Basis der Probanden mit vollständigen Tätigkeitsangaben als Modell gewählt wurde, sehr gut als finales Confoundermodell eignet.
Weiterhin unterscheiden sich die Modelle in Bezug auf die Expositionsvariablen. Das Modell auf Basis der Probanden mit vollständigen Tätigkeitsangaben enthielt die binäre Exposition über alle Tätigkeiten und Jahre hinweg. Das Modell auf Basis aller Probanden beinhaltete die (metrische) Exposition der 1. Tätigkeit. Vorteil der Exposition der 1. Tätigkeit ist sicherlich, dass diese Größe relativ schnell vorliegt - nämlich sobald jemand seine 1. Tätigkeit beendet hat. Allerdings sind für diese metrische Größe die Angaben zur Wochenstundenanzahl und zur Dauer der Tätigkeit nötig. Wählt man hingegen die binäre Exposition, so benötigt man ausschließlich die Tätigkeit und die Angabe, ob diese Tätigkeit für mindestens 8 Wochenstunden ausgeübt wurde, um die Person in die Kategorien

[4] In den Tabellen werden einheitlich 4 Nachkommastellen dargestellt, um bei den Expositionsvariablen eine Tendenz erkennen zu können. (Dadurch sollen keine Aussagen über die Schätzgenauigkeit getroffen werden.)

exponiert vs. nicht exponiert einteilen zu können. Allerdings muss für diese Größe das gesamte bisherige Arbeitsleben einer Person betrachtet werden, und nicht nur die erste Tätigkeit.

8.7.4 Logit-Modell 2: Asthma auf Basis aller Probanden

Variablenselektion und Wahl eines Confoundermodells

Die Variablenselektion durch Schrittweise-Selektion auf Basis des AIC-Kriteriums wurde durchgeführt, um ein Confoundermodell zu erhalten. Die Tabelle 8.13 fasst zusammen, welche Variablen zusätzlich zu den Variablen Geschlecht und Sozioökonomischer Status in die jeweiligen Modelle aufgenommen wurden.

Daten	Asthma (I)	Asthma (S)	Neuroder-mitis (I)	Neuroder-mitis (S)	Allergische Rhinitis (S)	Rauchen (S)	Passivrauch (S II)
1	+	+	+	+	+	+	-
2	+	+	+	+	+	+	-
3	+	+	+	+	+	+	+
4	+	+	+	+	+	+	+
5	+	+	+	+	+	+	+

Tabelle 8.13: Asthma-Modelle auf Basis aller Probanden: Selektierte Confoundervariablen (zusätzlich zu Geschlecht und Sozioökonomischer Status) (Abkürzungen: I: ISAAC II, S: SOLAR, S II: SOLAR II)

Zur Auswahl stehen folgende zwei Confoundermodelle:

- **Confoundermodell A**:
  Einflussgrößen: Geschlecht, Sozioökonomischer Status, Asthma (ISAAC II), Asthma (SOLAR), Neurodermitis (ISSAC II), Neurodermitis (SOLAR), Allergische Rhinits (SOLAR), Rauchen (SOLAR)
- **Confoundermodell B**:
  Einflussgrößen: Geschlecht, Sozioökonomischer Status, Asthma (ISAAC II), Asthma (SOLAR), Neurodermitis (ISAAC II), Neurodermitis (SOLAR), Allergische Rhinits (SOLAR), Rauchen (SOLAR), Passivrauch (SOLAR II)

Unter Anwendung von Likelihood-Ratio-Tests wurde schlußendlich das Confoundermodell A ausgewählt.

Einflussgrößen des gewählten Confoundermodells für Asthma auf Basis aller Probanden

Geschlecht, Sozioökonomischer Status, Asthma (ISAAC II), Asthma (SOLAR), Neurodermitis (ISAAC II), Neurodermitis (SOLAR), Allergische Rhinits (SOLAR), Rauchen (SOLAR)

Aufnahme von Expositionsvariablen in das Confoundermodell

Die Likelihood-Ratio-Tests zur Aufnahme der Expositionsvariablen sprachen (aus statistischer Sicht) alle für eine Beibehaltung des Confoundermodells. Tabelle 8.14 fasst die p-Werte der durchgeführten Likelihood-Ratio-Tests zusammen.

Likelihood-Ratio-Test Confoundermodell vs. Modell inklusive ...	**p-Wert** Datensatz 1	**p-Wert** Datensatz 2	**p-Wert** Datensatz 3	**p-Wert** Datensatz 4	**p-Wert** Datensatz 5
kumulierte Expositionen	0,09	0,12	0,12	0,14	0,11
binäre Expositionen	0,37	0,37	0,37	0,38	0,37
Expositionen des 1. Tätigkeitsjahres	0,67	0,72	0,79	0,74	0,77
Expositionen des 1. Jahres (inkl. HMW als quadratischer Term)	0,55	0,43	0,46	0,50	0,47
binäre Expositionen des 1. Jahres	0,92	0,92	0,92	0,92	0,92
Expositionen der 1. Tätigkeit	0,49	0,52	0,52	0,64	0,51
binäre Expositionen der 1. Tätigkeit	0,95	0,95	0,96	0,96	0,96

Tabelle 8.14: Übersicht über die p-Werte der durchgeführten Likelihood-Ratio-Tests

Aus inhaltlichen Gründen wurden allerdings Expositionsvariablen in das finale Modell aufgenommen. Welche Expositionsvariablen aufgenommen wurden, wurde auf Basis des AIC-Kriteriums entschieden. Die Tabelle 8.15 liefert eine Übersicht über die AICs der verschiedenen Modelle, wobei pro Datensatz für die Modelle mit Exposition das Modell mit dem kleinsten AIC hervorgehoben ist.

Modell	**AIC** Datensatz 1	**AIC** Datensatz 2	**AIC** Datensatz 3	**AIC** Datensatz 4	**AIC** Datensatz 5
Confoundermodell (ohne Expositionsvariablen)	359,36	359,95	360,22	360,24	360,65
Confoundermodell inkl. kumulierte Expositionen	**359,91**	**361,18**	**361,40**	**361,93**	**361,56**
Confoundermodell inkl. binäre Expositionen	364,01	364,56	364,87	364,95	365,29
Confoundermodell inkl. Expositionen des 1. Tätigkeitsjahres	366,18	367,10	367,79	367,52	368,11
Confoundermodell inkl. Expositionen des 1. Tätigkeitsjahres und HMW-Exposition als quadratischer Term	366,45	366,04	366,57	366,89	367,07
Confoundermodell inkl. binäre Expositionen des 1. Jahres	367,89	368,48	368,76	368,78	369,21
Confoundermodell inkl. Expositionen der 1. Tätigkeit	364,91	365,76	366,01	366,83	366,38
Confoundermodell inkl. binäre Expositionen der 1. Tätigkeit	368,26	368,85	369,14	369,16	369,58

Tabelle 8.15: Übersicht über die AICs der unterschiedlichen Modelle

Da das Modell mit den kumulierten Expositionen im Vergleich zu den anderen Expositionen auf Basis des AICs am besten geeignet war, wurde die kumulierte Exposition über alle Tätigkeiten und Jahre in das Modell aufgenommen.

Analyse des finalen Modells für Asthma

Als finales Modell für Astma wurde folgendes Modell ausgewählt:

Einflussgrößen des finalen Modells für Asthma auf Basis aller Probanden
Geschlecht, Sozioökonomischer Status, Asthma (ISAAC II), Asthma (SOLAR), Neurodermitis (ISAAC II), Neurodermitis (SOLAR), Allergische Rhinits (SOLAR), Rauchen (SOLAR), HMW-Exposition kumuliert, LMW-Exposition kumuliert, MIXED-Exposition kumuliert, IRRPEAKS-Exposition kumuliert, LOWRISK-Exposition kumuliert

Anhand von ROC-Kurven wurde das finale Modell analysiert. Die Abbildung 8.10 stellt beispielhaft die ROC-Kurve des ersten Datensatzes dar. Die ROC-Kurven weisen alle einen sehr ähnlichen Verlauf auf, die Flächen liegen im Bereich [0.875,0.880].

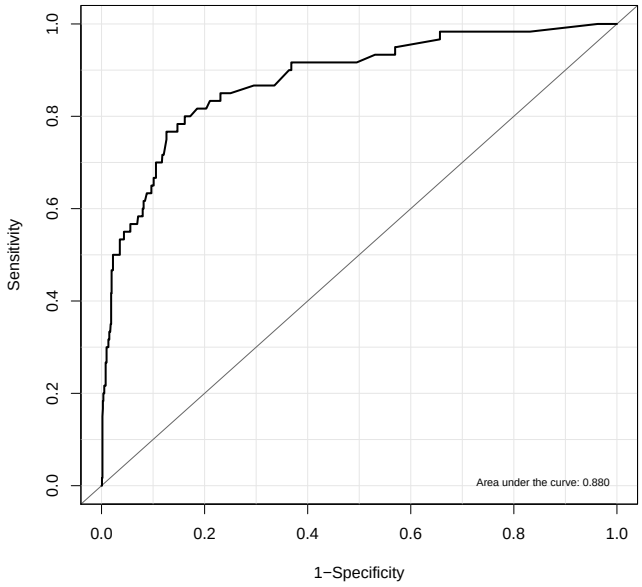

Abbildung 8.10: Beispielhafte ROC-Kurve für das Modell mit der Zielgröße Asthma in SOLAR II

8.7.5 Kombination der Schätzer des finalen Modells

Unter Anwendung der Kombinationsregeln wurden die Ergebnisse der fünf Datensätze kombiniert. Eine Übersicht über die dadurch erhaltenen kombinierten Parameterschätzer, die zugehörigen kombinierten geschätzten Standardabweichungen, die Odds-Ratios und die 95%-Konfidenzintervalle sind in Tabelle 8.16 zusammengestellt. [5]

Parameter	**Parameter-schätzer**	**Standard-abweichung**	**Odds-Ratio**	**95%-Konfidenz-intervall**
Intercept	-3,9280	0,3778	0,0197	[0,0094; 0,0413]
Asthma (ISAAC II)	0,9849	0,4787	2,6775	[1,0477; 6,8425]
Asthma (SOLAR)	2,8572	0,4394	17,4131	[7,3601; 41,1972]
Neurodermitis (ISAAC II)	0,8329	0,4720	2,2999	[0,9118; 5,8009]
Neurodermitis (SOLAR)	0,8151	0,4603	2,2595	[0,9166; 5,5698]
Allergische Rhinitis (SOLAR)	0,8920	0,3508	2,4398	[1,2266; 4,8527]
Rauchen (SOLAR)	0,6958	0,3225	2,0053	[1,0657; 3,7732]
Geschlecht (*Referenz: männlich*)	-0,2991	0,3336	0,7415	[0,3856; 1,4258]
Sozioökonomischer Status (*Referenz: niedrig*)	-0,4086	0,3241	0,6646	[0,3521, 1,2544]
HMW-Exposition kumuliert (pro Stunde)	0,0002	0,0001	1,0002	[1,0000; 1,0004]
LMW-Exposition kumuliert (pro Stunde)	0,0000	0,0001	1,0000	[0,9998; 1,0002]
MIXED-Exposition kumuliert (pro Stunde)	0,0000	0,0002	1,0000	[0,9997; 1,0003]
IRRPEAKS-Exposition kumuliert (pro Stunde)	0,0003	0,0002	1,0003	[0,9999; 1,0007]
LOWRISK-Exposition kumuliert (pro Stunde)	-0,0000	0,0001	1,0000	[0,9998; 1,0002]

Tabelle 8.16: Finales Modell für Asthma auf Basis aller Probanden: Kombinierte Parameterschätzer, Standardabweichungen, Odds-Ratios und 95%-Konfidenzintervalle

[5] In den Tabellen werden einheitlich 4 Nachkommastellen dargestellt, um bei den Expositionsvariablen eine Tendenz erkennen zu können. (Dadurch sollen keine Aussagen über die Schätzgenauigkeit getroffen werden.)

8.7.6 Interpretation des finalen Modells für Asthma auf Basis aller Probanden

Folgende Aussagen lassen sich auf Basis der (kombinierten) Odds-Ratios aus der zuvor dargestellten Tabelle auf Basis aller Probanden ableiten.
Die Chance in SOLAR II an Asthma zu leiden...

- ... ist für Personen, die bereits während der Zeit, in der die Studie ISAAC II durchgeführt wurde (1995/96), Asthma hatten, knapp 3 mal so hoch (Faktor 2,7) als die Chance für Personen, die zum damaligen Zeitpunkt kein Asthma hatten.

- ... ist für Personen, die zum Zeitpunkt der SOLAR-Studie an Asthma erkrankt waren, 17 mal so hoch (Faktor 17,4) als die Chance für Personen, die damals kein Asthma hatten.
  Der absolute Faktor sollte allerdings mit Vorsicht interpretiert werden, da die große Spannweite des Konfidenzintervalls auf eine starke Schätzungenauigkeit hinweist. Allerdings kann mit 95%-iger Sicherheit die Aussage getroffen werden, dass eine Asthmaerkrankung zum Zeitpunkt der SOLAR-Studie die Chance auf ein Auftreten dieser Erkrankung zum Zeitpunkt der SOLAR II-Studie um das 7- bis 41-fache erhöht (im Vergleich zur Chance für damals Nichterkrankte).

- ... ist für Personen, die während der SOLAR-Studie Allergischer Rhinitis hatten, doppelt so hoch (Faktor 2,4) als die Chance für Personen, die damals keine Allergische Rhinitis aufwiesen.

- ... ist für Personen, die zum Zeitpunkt der SOLAR-Studie rauchten, doppelt so hoch (Faktor 2,0) als die Chance bei damaligen Nichtrauchern.

Die Variablen “Geschlecht” und “Sozioökonomischer Status” verblieben wie bereits in den vorhergehenden Modellen aufgrund von substanzwissenschaftlichen Überlegungen und Hypothesen im Modell. Ein signifikanter Unterschied in Bezug auf Asthma konnte zwischen Mädchen und Jungen bzw. zwischen niedrigem und hohem Sozialstatus nicht nachgewiesen werden.
Ob die Variablen Neurodermitis in ISAAC II und SOLAR tatsächlich einen Einfluss auf das Auftreten von Asthma haben, konnte auf Basis dieses finalen Modells nicht abschließend geklärt werden. Da es sich bei dem finalen Modell um ein aus fünf Datensätzen kombiniertes Modell handelte, konnte es durchaus vorkommen, dass eine Variable auf einem der Datensätze einen signifikanten Einfluss hatte, der Effekt allerdings auf den anderen Datensätzen und letztendlich auch im kombinierten Modell verschwand. Da man allerdings eher eine zusätzliche (evtl. irrelevante) Variable in das finale Modell aufnehmen möchte, als eine möglicherweise relevante Variable im Modell nicht zu berücksichtigen, entschied man sich bis auf Weiteres für die Aufnahme dieser Variablen.

Vergleich der Modelle für Asthma

Vergleicht man nun diese Modell auf Basis aller Probanden, d.h. auf Basis der durch Imputation vervollständigten Datensätze, mit dem vorherigen Modell, auf Basis der Probanden mit vollständigen Tätigkeitsangaben, so erhält man folgendes Bild (vgl. Tabelle 8.17. [6]

Parameter	**Odds-Ratio** Modell auf Basis der Probanden mit vollständigen Tätigkeitsangaben	**Odds-Ratio** Modell auf Basis aller Probanden (inkl. imputierter Tätigkeitsangaben)
Intercept	0,0169	0,0197
Asthma (ISAAC II)	3,5557	2,6775
Asthma (SOLAR)	16,2040	17,4131
Neurodermitis (ISAAC II)	-	2,2999
Neurodermitis (SOLAR)	4,3535	2,2595
Allergische Rhinitis (SOLAR)	2,6611	2,4398
Rauchen (SOLAR)	2,0605	2,0053
Geschlecht (*Referenz: männlich*)	0,7740	0,7415
Sozioökonomischer Status (*Referenz: niedrig*)	0,7644	0,6646
HMW-Exposition kumuliert (pro Stunde)	1,0003	1,0002
LMW-Exposition kumuliert (pro Stunde)	0,9999	1,0000
MIXED-Exposition kumuliert (pro Stunde)	1,0000	1,0000
IRRPEAKS-Exposition kumuliert (pro Stunde)	1,0008	1,0003
LOWRISK-Exposition kumuliert (pro Stunde)	1,0000	1,0000

Tabelle 8.17: Vergleich der Modelle für Asthma

Betrachtet man zunächst die Odds-Ratios, so unterscheiden sich die Modelle nur geringfügig.
In Bezug auf die Confoundervariablen unterscheiden sich die Modelle in der Variable Neurodermitis in ISAAC II. Um Feststellen zu können, ob diese zusätzliche Variable eine Modellverbesserung liefert, wurde ein Likelihood-Ratio-Test durchgeführt. Das Ergebnis dieses Tests war, dass sich das minimale Confoundermodell (ohne die Variable Neurodermitis in ISAAC II), dass auf Basis der Probanden mit vollständigen Tätigkeitsangaben als Modell gewählt wurde, sehr gut als finales Confoundermodell eignet.
Weiterhin ist anzumerken, dass in dem Modell auf Basis der Probanden mit vollständigen Tätigkeitsangaben die Expositionen (statistisch) signifikant waren. Auf Basis aller Probanden waren diese Expositionsvariablen allerdings (statistisch) nicht mehr signifikant, wurden aber aus inhaltlichen Gründen aufgenommen. Es ist zu vermuten, dass das konservative Vorgehen bei der Exposition der Tätigkeitsangaben und die damit verbundene Unterschätzung der Exposition, dazu führte, dass der Effekt verschwand.

[6] In den Tabellen werden einheitlich 4 Nachkommastellen dargestellt, um bei den Expositionsvariablen eine Tendenz erkennen zu können. (Dadurch sollen keine Aussagen über die Schätzgenauigkeit getroffen werden.)

9 Zusammenfassung der Ergebnisse

9.1 Grundsätzliches Vorgehen

Hauptziel dieser Bachelorarbeit war die Erstellung von logistischen Regressionsmodellen, um das Auftreten von Allergischer Rhinitis und Asthma zu untersuchen.

Folgende Schritte waren nötig, um dieses Ziel zu erreichen:

Zunächst wurden diverse Bereinigungs- und Rekodierungsschritte am ursprünglichen Datensatz durchgeführt. Auf Basis dieses korrigierten Datensatzes konnten die Expositionen in den entsprechenden Tätigkeiten unter zu Hilfenahme der ISCO-88-Berufssystematik und der Job-Exposure-Matrix berechnet und zusammengefasst werden.

Die Imputation der fehlenden Angaben erfolgte in zwei Stufen.
Zunächst wurden die Confoundervariablen vervollständigt, indem unter Verwendung der Methoden "Multiple Imputation" und "Ziehen aus der empirischen Verteilung" fünf Datensätze erstellt wurden.
In einem zweiten Schritt wurden die Tätigkeitsangaben imputiert, indem man die Methode "Ziehen aus der empirischen Verteilung" anwandte, die zusätzlich auf bestimmte Variablen bedingt wurde. Durch die Erstellung mehrerer Datensätze wurde der Unsicherheit bei der Ersetzung fehlender Werte Rechnung getragen.

Schlussendlich wurden logistische Regressionsmodelle an die Daten angepasst. Bei der Wahl der Modelle wählte man erneut ein zweistufiges Vorgehen:
Im ersten Schritt wurde aus allen potenziellen Confoundervariablen ein geeignetes Confoundermodell ausgewählt.
Im zweiten Schritt wurde überprüft, ob die zusätzliche Aufnahme von Expositionsvariablen in das gewählte Confoundermodell zu einer Modellverbesserung führt, bzw. welche Expositionsvariablen in das finale Modell aufgenommen werden.

9.2 Finales Modell für Allergische Rhinitis

Im finalen Modell für das Auftreten von Allergischer Rhinitis wurden folgende Confounderariablen ausgewählt. Eine zusätzliche Aufnahme der Expositionsvariablen erfolgte aus inhaltlichen Gründen. Dabei wurden unterschiedliche Expositionsvariablen auf den beiden Datensätzen zum einen auf Basis der Probanden mit vollständigen Tätigkeitsangaben, zum anderen auf Basis aller Probanden (inklusive imputierter Tätigkeitsangaben) ausgewählt.

Einflussgrößen des finalen Modells für Allergische Rhinitis
Confoundervariablen:
Geschlecht, Sozioökonomischer Status, Atopie der Eltern,
Allergische Rhinits (ISAAC II), Allergische Rhinitis (SOLAR), Als Säugling gestillt,
Expositionsvariablen auf Basis der Probanden mit vollständigen Tätigkeitsangaben:
HMW-Exposition binär, LMW-Exposition binär, MIXED-Exposition binär,
IRRPEAKS-Exposition binär, LOWRISK-Exposition binär
Expositionsvariablen auf Basis aller Probanden
(inklusive imputierter Tätigkeitsangaben):
HMW-Exposition 1. Tätigkeit, LMW-Exposition 1. Tätigkeit,
MIXED-Exposition 1. Tätigkeit, IRRPEAKS-Exposition 1. Tätigkeit,
LOWRISK-Exposition 1. Tätigkeit

Dabei stellte sich in beiden Modellen eine frühere Allergische Rhinitis (entweder in SOLAR oder in ISAAC II) als Haupteinflussgröße heraus. Das bedeutet, dass diese Atopie am besten durch das bisherige Auftreten einer solchen Atopie vorhergesagt werden kann. Litten zusätzlich die Eltern bereits an einer atopischen Erkrankung (Neurodermitis, Allergische Rhinitis oder Asthma), so war die Wahrscheinlichkeit einer Asthmaerkrankung für die Kinder deutlich erhöht. Die Variablen “Geschlecht” und “Sozioökonomischer Status” wurden aufgrund von Vorüberlegungen und Hypothesen in dem Modell belassen. Ein signifikanter Unterschied zwischen Mädchen und Jungen bzw. zwischen niedrigem und hohem Sozialstatus konnte in Bezug auf Allergische Rhinitis nicht nachgewiesen werden. Ob tatsächlich ein Unterschied zwischen als Säugling gestillten und nicht gestillten Probanden in Bezug auf das Auftreten von allergischer Rhinitis besteht, konnte nicht abschließend geklärt werden. Da es sich bei dem finalen Modell um ein aus fünf Datensätzen kombiniertes Modell handelte, konnte es durchaus vorkommen, dass eine Variable auf einem der Datensätze einen signifikanten Einfluss hatte, der Effekt allerdings auf den anderen Datensätzen und letztendlich auch im kombinierten Modell verschwand. Dies war bei der Variable “Als Säugling gestillt” der Fall. Da man allerdings eher eine zusätzliche (evtl. irrelevante) Variable in das finale Modell aufnehmen möchte, als eine möglicherweise relevante Variable im Modell nicht zu berücksichtigen, entschied man sich bis auf Weiteres für die Aufnahme dieser Variable.
Die Modelle unterschieden sich in Bezug auf die Expositionsvariablen. Das Modell auf

Basis der Probanden mit vollständigen Tätigkeitsangaben enthielt die binäre Exposition über alle Tätigkeiten und Jahre hinweg. Das Modell auf Basis aller Probanden beinhaltete die (metrische) Exposition der 1. Tätigkeit. Vorteil der Exposition der 1. Tätigkeit ist sicherlich, dass diese Größe relativ schnell vorliegt - nämlich sobald jemand seine 1. Tätigkeit beendet hat. Allerdings sind für diese metrische Größe die Angaben zur Wochenstundenanzahl und zur Dauer der Tätigkeit nötig. Wählt man hingegen die binäre Exposition, so benötigt man ausschließlich die Tätigkeit und die Angabe, ob diese Tätigkeit für mindestens 8 Wochenstunden ausgeübt wurde, um die Person in die Kategorien exponiert vs. nicht exponiert einteilen zu können. Allerdings muss für diese Größe das gesamte bisherige Arbeitsleben einer Person betrachtet werden, und nicht nur die erste Tätigkeit.

9.3 Finales Modell für Asthma

Das finale Modell für das Auftreten von Asthma enthielt folgende Confoundervariablen. Weiterhin führte die Aufnahme der kumulierten Expositionsvariablen (auf Basis der Probanden mit vollständigen Tätigkeitsangaben) zu einer signifikanten Modellverbesserung.

Einflussgrößen des finalen Modells für Asthma
Confoundervariablen:
Geschlecht, Sozioökonomischer Status, Asthma (ISSAC II), Asthma (SOlAR), Neurodermitis (SOLAR), Allergische Rhinits (SOLAR), Rauchen (SOLAR)
Expositionsvariablen:
HMW-Exposition kumuliert, LMW-Exposition kumuliert, MIXED-Exposition kumuliert, IRRPEAKS-Exposition kumuliert, LOWRISK-Exposition kumuliert

In diesem Modell stellten atopische Erkrankungen (Asthma, Neurodermitis und Allergische Rhinitis) in der Vergangenheit - vor allem in der vorangehenden Studie - die Haupteinflussgrößen dar. Weiterhin ist die Wahrscheinlichkeit einer Asthmaerkrankung bei Rauchern deutlich erhöht.
Die kumulierten Expositionen stehen ebenfalls in Zusammenhang mit dem Auftreten von Asthma. Dabei erscheint es im Zusammenhang mit dem Auftreten einer Asthmaerkrankung nicht als sinnvoll, die kumulierten Expositionen durch die Exposition während der ersten Tätigkeit bzw. während des ersten Tätigkeitsjahres zu ersetzen.
Auch bei diesem Modell verblieben die Variablen “Geschlecht” und “Sozioökonomischer Status” aufgrund von substanzwissenschaftlichen Vorüberlegungen und Hypothesen im Modell. In Bezug auf Asthma konnte allerdings kein signifikanter Unterschied zwischen Mädchen und Jungen bzw. niedrigem und hohen Sozialstatus nachgewiesen werden.

10 Abschließende Diskussion der Ergebnisse

Die Wahl eines geeigneten logistischen Regressionsmodells jeweils für Allergische Rhinitis und Asthma wurde in der vorliegenden Arbeit in zwei Schritte untergliedert:

1. Zunächst wurde auf Basis von etwa 20 potenziellen Confoundervariablen ein Confoundermodell erstellt.

2. Im zweiten Schritt wurde mit Hilfe von Likelihood-Ratio-Test überprüft, ob eine zusätzliche Aufnahme von Expositionsvariablen zu einer Modellverbesserung führte bzw. welche Expositionsvariablen den besten Beitrag zur Erklärung der jeweiligen Krankheit lieferten.

Beide Schritte sollen im folgenden Abschnitt diskutiert werden.

Zu 1) ist Folgendes anzumerken:
Als Confoundervariablen gingen in die Modelle unter anderem der jeweilige Erkrankungsstatus aus ISAAC II und SOLAR ein. Somit gingen in das Modell für die Zielgröße "Allergische Rhinitis in SOLAR II" die Kovariablen "Allergische Rhinitis in ISAAC II" und "Allergische Rhinitis in SOLAR" ein. Analog beinhaltete das Modell für die Zielgröße "Asthma in SOLAR II" die Confoundervariablen "Asthma in ISAAC II" und "Asthma in SOLAR". Durch diese Einflussgrößen konnte der jeweilige Erkrankungsstatus in SOLAR II relativ gut erklärt werden, da deren Einfluss sehr hoch war. Hauptsächlich auf diese Einflussgrößen sind die guten Ergebnisse der ROC-Analysen der verschiedenen Modelle zurückzuführen.
Als interessant würde sich weiterhin die Fragestellung erweisen, welche Modelle im Rahmen der Variablenselektion resultieren würden, wenn diese Einflussgrößen, also die vergangenen Responsewerte, aus der Liste der potenziellen Confoundervariablen entfernt würden.
Als problematisch muss die Tatsache erachtet werden, dass in den derzeitigen Modellen die zeitliche Abfolge zwischen Exposition und Erkrankung nicht ausreichend berücksichtigt wurde. Beispielsweise kann es sein, dass ein Proband bis zum Zeitpunkt der SOLAR-Studie bereits einer Exposition ausgesetzt war, die die Entwicklung der jeweiligen Krankheit (Allergische Rhinitis oder Asthma) in SOLAR beeinflusste. Dadurch, dass nun sowohl die gesamte Exposition (also die Exposition in SOLAR und SOLAR II) und die Variable "Allergische Rhinitis in SOLAR" bzw. "Asthma in SOLAR" als Einflussgrößen in die Modelle eingehen, erfolgt keine Berücksichtigung des zeitlichen Verlaufs.
Eine mögliche Lösung wäre, die Probanden mit einer entsprechenden Erkrankung in SOLAR (und eventuell diejenigen mit Erkrankung in ISAAC II) auszuschließen und eine reine Inzidenzanalyse durchzuführen.
Weiterhin könnte man für die Probanden mit einer Erkrankung in SOLAR ausschließlich die Exposition bis zum Zeitpunkt der SOLAR-Studie betrachten. Für die Probanden, die erst in SOLAR II erkrankt sind, würde man hingegen die gesamte Exposition berücksichtigen. In beiden Fällen würde der jeweilige Proband als erkrankt gelten. Die Zielgrößen

aus ISAAC II ("Allergische Rhinitis in ISAAC II" bzw. "Asthma in ISAAC II") könnten in diesem Fall als Confoundervariablen berücksichtigt werden, da zum Zeitpunkt von ISAAC noch keine (erfasste) Exposition vorlag.
Welche Ergebnisse diese Berücksichtigung des zeitlichen Verlaufs liefert, müssen künftige Arbeiten zeigen.

Zu 2) ist Folgendes anzumerken:
Die Expositionsvariablen wurden in die logistischen Regressionsmodelle als lineare Einflussgrößen aufgenommen. Dieser lineare Einfluss wurde mit Hilfe von GAMs überprüft. Da allerdings der Großteil der Probanden in den jeweiligen Expositionskategorien keiner Exposition ausgesetzt war, lag nur bei verhältnismäßig wenigen Probanden eine Exposition vor. Je nach Expositionskategorie waren zwischen 22% und weniger als 1% der Probanden exponiert.
Als mögliche Vorgehensweisen bei der Modellierung des Einflusses der Expositionsvariablen sollen zwei Möglichkeiten vorgestellt werden, die in einem nächsten Schritt betrachtet werden könnten.
Eine Möglichkeit bestünde darin, die Probanden ohne Exposition in der jeweiligen Kategorie anders zu behandeln als die Probanden mit vorhandener Exposition. Beispielsweise könnte die HMW-Exposition als Einflussgröße folgendermaßen modelliert werden:
Zunächst wird eine Indikatorvariable benötigt, die angibt, ob für den jeweiligen Probanden in der Kategorie HMW eine Exposition vorliegt:

$$I_{HMW} = \begin{cases} 1, & in\ der\ Kategorie\ HMW\ exponiert \\ 0, & in\ der\ Kategorie\ HMW\ nicht\ exponiert \end{cases}$$

Diese Indikatorvariable würde man dann wie folgt verwenden:

$$\beta \cdot (1 - I) + \gamma \cdot I \cdot HMW\_kumuliert$$

Dabei gibt "HMW_kumuliert" die kumulierte HMW-Exposition über alle Tätigkeiten und Jahre hinweg an.
Für die Probanden, die keiner HMW-Exposition ausgesetzt waren (I = 0), würde in die Modellgleichung ausschließlich der konstante Term β eingehen.
Für die exponierten Probanden (I = 1) enthält die Modellgleichung den Term $\gamma \cdot$ HMW_kumuliert.
In analoger Weise würde man für die restlichen Expositionskategorien vorgehen.
In diesem Fall würde man zunächst zwischen exponierten und nicht exponierten Probanden unterscheiden, die tatsächliche Exposition ginge aber nach wie vor als linearer Term in die Modelle ein.

Möchte man die Expositionen hingegen nicht linear modellieren, so könnte man die Exposition in folgende Kategorien unterteilen:

- Tätigkeit mit hohem Asthmarisiko
  (Expositionskategorien HMW, LMW, MIXED, IRRPEAKS)
  (keine Exposition vs. Expositions-Quartile)
- Tätigkeit mit niedrigem Asthmarisiko
  (Expositionskategorie LOWRISK)
  (keine Exposition vs. Expositions-Quartile)
- Tätigkeit mit keiner asthmaspezifischen Exposition (im Sinne der JEM)
  (binär)
- Nie gearbeitet
  (binär)

Diese 14 Kategorien könnten dann in die Modelle aufgenommen werden, wobei die Personen, die nie gearbeitet haben, als Referenzkategorie gewählt werden.
Inwiefern diese Lösung aufgrund der Fallzahlen in dem vorliegenden Datensatz praktikabel ist, muss separat geprüft werden. Alternativ könnten neben der Aufteilung in die Expositions-Quartile auch die Einteilung "Exposition unterhalb des Medians" vs. "Exposition überhalb des Medians" gewählt werden.

Zusammenfassend ist zu sagen, dass die in dieser Arbeit entwickelten Modelle als erklärende Modelle verstanden werden sollen. Um die Modellgüte auf den vorliegenden Daten zu beurteilen, wurden als Instrumente das AIC-Kriterium und die ROC-Analyse verwendet. Diese ROC-Kurven zeichnen allerdings eher ein zu optimistisches Bild und überschätzen die AUC ("area under the curve") tendenziell, da keine Kreuzvalidierung vorgenommen wurde. Ist man statt an einem erklärenden Modell an einem Prognosemodell interessiert, so ist die Durchführung einer Kreuzvalidierung zu empfehlen, um die Prognosegüte dieser Modelle abzuschätzen.
Weiterhin wurden für die vorliegende Bachelorarbeit ausschließlich die Probanden in der Analyse betrachtet, die zum Zeitpunkt des Beginns dieser Arbeit (31. März 2009) bereits an der Studie teilnahmen und deren Tätigkeitsangaben abschließend kodiert waren. Nach Ende der Feldphase können die Ergebnisse dieser Arbeit erneut auf Basis aller Teilnehmer der Studie überprüft werden.

A Variablenkodierung

A.1 Variablen aus ISAAC II

A.1.1 In Deutschland geboren

Die Angaben zum Geburtsort aus ISAAC II wurden so kodiert, dass man die Unterscheidung treffen konnte, ob jemand in Deutschland geboren wurde oder nicht. "In Deutschland geboren" wurde dadurch definiert, ob das Kind die deutsche Staatsangehörigkeit hat.

Verwendete Variablen:

Variablenname	**Frage & Kodierung**
STAATS1	Welche Staatsangehörigkeit hat Ihr Kind? 01=deutsch, 02=russisch, ..., 15=sonstiges, NA=Missing

Bildung der Variable d_geb

Kodierungsbeschreibung:
d_geb wurde auf 1 gesetzt, wenn STAATS1 den Wert 1 hatte.
d_geb wurde auf missing (NA) gesetzt, wenn STAATS1 missing (NA) war.
In allen anderen Fällen stand in der Variablen d_geb der Wert 0.

Kodierungsübersicht:

Verwendete Variablen	**Abgeleitete Variablen**
STAATS1	**geb_d**
1	1
2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15	0
NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
geb_d	In Deutschland geboren 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> d_geb <- 0
> d_geb[is.na(STAATS1)] <- NA
> d_geb[!is.na(STAATS1)&(STAATS1==1)] <- 1
```

A.1.2 Atopie der Eltern

Die Variable PAR_ALL aus ISAAC II wurde rekodiert, um eine 0-1-Kodierung zu erhalten, um die Interpretation und Vergleichbarkeit mit den (0-1-kodierten) Variablen aus SOLAR und SOLAR II zu vereinfachen.

Verwendete Variablen:

Variablenname	Frage & Kodierung
PAR_ALL	Atopie der Eltern Hatte mindestens ein Elternteil des Kindes irgendwann einmal Asthma, Heutschnupfen oder Neurodermitis? 1=Ja, 2=Nein, NA=Keine Angabe

Bildung der Variable PAR_ALL_r

Kodierungsbeschreibung:
PAR_ALL_r wurde auf 0 (Nein) gesetzt, wenn PAR_ALL den Wert 2 hatte.
PAR_ALL_r blieb auf 1 (Ja) gesetzt, wenn PAR_ALL den Wert 1 hatte.
PAR_ALL_r war missing (NA), wenn PAR_ALL missing (NA) war.

Kodierungsübersicht:

Verwendete Variablen	Abgeleitete Variablen
PAR_ALL	**PAR_ALL_r**
1	1
2	0
NA	NA

Code-Übersicht:

Variablenname	Frage & Kodierung
PAR_ALL_r	Atopie der Eltern 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> PAR_ALL_r[PAR_ALL==2] <- 0
> PAR_ALL_r[PAR_ALL==1] <- 1
```

A.1.3 Als Säugling gestillt

Die Variable STILL aus ISAAC II wurde rekodiert, um eine 0-1-Kodierung zu erhalten, um die Interpretation und Vergleichbarkeit mit den (0-1-kodierten) Variablen aus SOLAR und SOLAR II zu vereinfachen.

Verwendete Variablen:

Variablenname	Frage & Kodierung
STILL	Als Säugling gestillt (Wurde Ihr Kind gestillt?) 1=Ja, 2=Nein, NA=Keine Angabe

Bildung der Variable STILL_r

Kodierungsbeschreibung:
STILL_r wurde auf 0 (Nein) gesetzt, wenn STILL den Wert 2 hatte.
STILL_r blieb auf 1 (Ja) gesetzt, wenn STILL den Wert 1 hatte.
STILL_r war missing (NA), wenn STILL missing (NA) war.

Kodierungsübersicht:

Verwendete Variablen	Abgeleitete Variablen
STILL	**STILL_r**
1	1
2	0
NA	NA

Code-Übersicht:

Variablenname	Frage & Kodierung
STILL_r	Als Säugling gestillt 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> STILL_r[STILL==2] <- 0
> STILL_r[STILL==1] <- 1
```

A.1.4 Neurodermitis

Die Variable CUR_DERM aus ISAAC II wurde rekodiert, um eine 0-1-Kodierung zu erhalten, um die Interpretation und Vergleichbarkeit mit den (0-1-kodierten) Variablen aus SOLAR und SOLAR II zu vereinfachen.

Verwendete Variablen:

Variablenname	Frage & Kodierung
CUR_DERM	Neurodermitis 1=Ja, 2=Nein, NA=Keine Angabe

Bildung der Variable CUR_DERM_r

Kodierungsbeschreibung:
CUR_DERM_r wurde auf 0 (Nein) gesetzt, wenn CUR_DERM den Wert 2 hatte.
CUR_DERM_r blieb auf 1 (Ja) gesetzt, wenn CUR_DERM den Wert 1 hatte.
CUR_DERM_r war missing (NA), wenn CUR_DERM missing (NA) war.

Kodierungsübersicht:

Verwendete Variablen	Abgeleitete Variablen
CUR_DERM	**CUR_DERM_r**
1	1
2	0
NA	NA

Code-Übersicht:

Variablenname	Frage & Kodierung
CUR_DERM_r	Neurodermitis 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> CUR_DERM_r[CUR_DERM==2] <- 0
> CUR_DERM_r[CUR_DERM==1] <- 1
```

A.1.5 Allergische Rhinitis

Die Variable CUR_HAY aus ISAAC II wurde rekodiert, um eine 0-1-Kodierung zu erhalten, um die Interpretation und Vergleichbarkeit mit den (0-1-kodierten) Variablen aus SOLAR und SOLAR II zu vereinfachen.

Verwendete Variablen:

Variablenname	Frage & Kodierung
CUR_HAY	Allergische Rhinitis 1=Ja, 2=Nein, NA=Keine Angabe

Bildung der Variable CUR_HAY_r

Kodierungsbeschreibung:
CUR_HAY_r wurde auf 0 (Nein) gesetzt, wenn CUR_HAY den Wert 2 hatte.
CUR_HAY_r blieb auf 1 (Ja) gesetzt, wenn CUR_HAY den Wert 1 hatte.
CUR_HAY_r war missing (NA), wenn CUR_HAY missing (NA) war.

Kodierungsübersicht:

Verwendete Variablen	Abgeleitete Variablen
CUR_HAY	**CUR_HAY_r**
1	1
2	0
NA	NA

Code-Übersicht:

Variablenname	Frage & Kodierung
CUR_HAY_r	Allergische Rhinitis 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> CUR_HAY_r[CUR_HAY==2] <- 0
> CUR_HAY_r[CUR_HAY==1] <- 1
```

A.1.6 Asthma

Die Variable CUR_ASTH aus ISAAC II wurde rekodiert, um eine 0-1-Kodierung zu erhalten, um die Interpretation und Vergleichbarkeit mit den (0-1-kodierten) Variablen aus SOLAR und SOLAR II zu vereinfachen.

Verwendete Variablen:

Variablenname	**Frage & Kodierung**
CUR_ASTH	Asthma 1=Ja, 2=Nein, NA=Keine Angabe

Bildung der Variable CUR_ASTH_r

Kodierungsbeschreibung:
CUR_ASTH_r wurd auf 0 (Nein) gesetzt, wenn CUR_ASTH den Wert 2 hatte.
CUR_ASTH_r blieb auf 1 (Ja) gesetzt, wenn CUR_ASTH den Wert 1 hatte.
CUR_ASTH_r war missing (NA), wenn CUR_ASTH missing (NA) war.

Kodierungsübersicht:

Verwendete Variablen	**Abgeleitete Variablen**
CUR_ASTH	**CUR_ASTH_r**
1	1
2	0
NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
CUR_ASTH_r	Asthma 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> CUR_ASTH_r[CUR_ASTH==2] <- 0
> CUR_ASTH_r[CUR_ASTH==1] <- 1
```

A.1.7 Passivrauch

Die Variable ETSNOW aus ISAAC II wurde rekodiert, so dass sie mit den Variablen aus SOLAR und SOLAR II einfacher vergleichbar war.

Verwendete Variablen:

Variablenname	Frage & Kodierung
ETSNOW	Passivrauch Ist das Kind in der Wohnung Tabakrauch ausgesetzt? 1=Eltern zur Zeit Raucher, 2=Eltern ehemalige Raucher, 3=Eltern nie geraucht, NA=Keine Angabe

Bildung der Variable ETSNOW_r

Kodierungsbeschreibung:
ETSNOW_r blieb auf 1 (Eltern zur Zeit Raucher) gesetzt, wenn ETSNOW den Wert 1 hatte.
ETSNOW_r blieb auf 2 (Eltern ehemalige Raucher) gesetzt, wenn ETSNOW den Wert 2 hatte.
ETSNOW_r wurde auf 0 (Eltern nie geraucht) gesetzt, wenn ETSNOW den Wert 3 hatte.
ETSNOW_r war missing (NA), wenn ETSNOW missing (NA) war.

Kodierungsübersicht:

Verwendete Variablen	Abgeleitete Variablen
ETSNOW	**ETSNOW_r**
1	1
2	2
3	0
NA	NA

Code-Übersicht:

Variablenname	Frage & Kodierung
ETSNOW_r	Asthma 0=Eltern nie geraucht, 1=Eltern zur Zeit Raucher, 2=Eltern ehemalige Raucher, NA=Keine Angabe

R-Code:

```
> ETSNOW_r[ETSNOW==2] <- 2
> ETSNOW_r[ETSNOW==1] <- 1
> ETSNOW_r[ETSNOW==3] <- 0
```

A.1.8 Sozioökonomischer Status

Die Variable SES aus ISAAC II wurde rekodiert, um eine 0-1-Kodierung zu erhalten, um die Interpretation und Vergleichbarkeit mit den (0-1-kodierten) Variablen aus SOLAR und SOLAR II zu vereinfachen.

Verwendete Variablen:

Variablenname	Frage & Kodierung
SES	Sozioökonomischer Status Schulabschluss (bzw. Dauer des Schulbesuchs) der Eltern 1=Hoch (Fachabitur/Abitur/Studium), 2=Niedrig (Niedrigere Ausbildung), NA=Keine Angabe

Bildung der Variable SES_r

Kodierungsbeschreibung:
SES_r wurde auf 0 (Niedrig) gesetzt, wenn SES den Wert 2 hatte.
SES_r blieb auf 1 (Hoch) gesetzt, wenn SES den Wert 1 hatte.
SES_r war missing (NA), wenn SES missing (NA) war.

Kodierungsübersicht:

Verwendete Variablen	Abgeleitete Variablen
SES	**SES_r**
1	1
2	0
NA	NA

Code-Übersicht:

Variablenname	Frage & Kodierung
SES_r	Sozioökonomischer Status 0=Niedirg (Niedrigere Ausbildung), 1=Hoch (Fachabitur/Abitur/Studium), NA=Missing

R-Code:

```
> SES_r[SES==2] <- 0
> SES_r[SES==1] <- 1
```

A.1.9 Studienzentrum

Die Variable zentrum aus ISAAC II wurde rekodiert, um eine 0-1-Kodierung zu erhalten, um die Interpretation und Vergleichbarkeit mit den (0-1-kodierten) Variablen aus SOLAR und SOLAR II zu vereinfachen.

Verwendete Variablen:

Variablenname	Frage & Kodierung
zentrum	Studienzentrum 23=Dresden, 24=München

Bildung der Variable zentrum_r

Kodierungsbeschreibung:
zentrum_r wurde auf 0 (Dresden) gesetzt, wenn zentrum den Wert 23 hatte.
zentrum_r wurde auf 1 (München) gesetzt, wenn zentrum den Wert 24 hatte.

Kodierungsübersicht:

Verwendete Variablen	Abgeleitete Variablen
zentrum	**zentrum_r**
23	0
24	1

Code-Übersicht:

Variablenname	Frage & Kodierung
zentrum_r	Studienzentrum 0=Dresden, 1=München, NA=Missing

R-Code:

```
> zentrum_r <- NA
> zentrum_r[zentrum==23] <- 0
> zentrum_r[zentrum==24] <- 1
```

A.1.10 Geschwister

Die Variable siblings (Anzahl der Geschwister) aus ISAAC II wurde zu einer dichotomen Variable GESCHW zusammengefasst, die angab, ob die Person Geschwister hat (=1) oder nicht (=0).

Verwendete Variablen:

Variablenname	Frage & Kodierung
siblings	Anzahl Geschwister 0-7=Anzahl der Geschwister, NA=Keine Angabe

Bildung der Variable GESCHW

Kodierungsbeschreibung:
GESCHW wurde auf 0 gesetzt, wenn die Anzahl der Geschwister gleich 0 war.
GESCHW wurde auf 1 gesetzt, wenn 1-7 Geschwister angegeben wurden.

Kodierungsübersicht:

Verwendete Variablen	Abgeleitete Variablen
siblings	**GESCHW**
0	0
1-7	1

Code-Übersicht:

Variablenname	Frage & Kodierung
GESCHW	Geschwister vorhanden 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> GESCHW <- NA
> GESCHW[siblings==0] <- 0
> GESCHW[siblings!=0] <- 1
```

A.2 Variablen aus SOLAR

A.2.1 Rauchverhalten

Die Angaben zum Rauchverhalten aus SOLAR wurden so kodiert, dass man die Unterscheidung zwischen Raucher und Nichtraucher treffen konnte.

Verwendete Variablen:

Variablenname	**Frage & Kodierung**
f54	Haben Sie selbst schon einmal Zigaretten geraucht? 0=Nein, 1=Ja probiert, 2=Ja öfter NA=Keine Angabe
f55	Haben Sie schon einmal ein Jahr lang geraucht? 0=Nein, 1=Ja, NA=Keine Angabe

In zwei Schritten, die im Folgenden ausführlich dargestellt werden, wurde aus den soeben aufgeführten Variablen die Variable RAUCHEN gebildet. Raucher war hier jemand, wenn er schon einmal ein Jahr lang geraucht hatte. Nichtraucher war somit jemand, wenn nicht bereits ein Jahr lang geraucht wurde.

Gebildete Variable:

Variablenname	**Frage & Kodierung**
RAUCHEN	Rauchverhalten in SOLAR 0=Nichtraucher, 1=Raucher, NA=Missing

Schritt 1: Bildung der Variable f55xx

Kodierungsbeschreibung:
Der Variablen f55xx wurden zunächst die Werte aus f55 zugewiesen.
War die Variable f55 missing (NA) und hatte f54 den Wert 0 oder 1, so wurde der Variable f55xx der Wert 0 zugewiesen.
War die Variable f55 missing (NA) und hatte f54 den Wert 2 oder war missing (NA), so wurde die Variable f55xx auf missing (NA) gesetzt.

Kodierungsübersicht:

Verwendete Variablen		**Abgeleitete Variablen**
f54	**f55**	**f55xx**
0, 1, 2, NA	0, 1	0, 1
0, 1	NA	0
2, NA	NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
f55xx	Haben Sie schon einmal ein Jahr lang geraucht? 0=Nein, 1=Ja, NA=Missing

R-Code:
Diese Variable wurde bereits vom Datenzentrum kodiert und war so im Datensatz enthalten, der als Grundlage für diese Arbeit galt. Eine selbstständige Kodierung mit Hilfe von R war daher nicht nötig.

Schritt 2: Bildung der Variable RAUCHEN

Kodierungsbeschreibung:
RAUCHEN wurde auf 1 gesetzt, wenn f55xx den Wert 1 hatte.
RAUCHEN wurde auf missing (NA) gesetzt, wenn f55xx missing (NA) war.
In allen anderen Fällen stand in der Variablen RAUCHEN der Wert 0.

Kodierungsübersicht:

Verwendete Variablen	**Abgeleitete Variablen**
f55xx	**RAUCHEN**
1	1
0	0
NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
RAUCHEN	Rauchverhalten in SOLAR 0=Nichtraucher, 1=Raucher, NA=Missing

R-Code:

```
> RAUCHEN <- 0
> RAUCHEN[!is.na(f55xx) & (f55xx==1)] <- 1
> RAUCHEN[is.na(f55xx)] <- NA
```

A.2.2 Berufssituation

Die Angaben zur Berufssituation aus SOLAR lagen ursprünglich als mehrere Dummy-Variablen vor. Diese Variablen wurden so kodiert, dass sie als eine kategoriale Variable vorlagen. Im Rahmen dieser Kodierung wurden zusätzlich Doppelnennungen korrigiert, die in dieser Frage nicht erlaubt waren. Wie mit diesen Doppelnennungen umzugehen war, wurde vorab gemeinsam mit Frau Kellberger und Herrn Heumann besprochen.

Verwendete Variablen:

Variablenname	**Frage & Kodierung**
f61_01xx	Sind Sie zur Zeit - HauptschülerIn 0=Nein, 1=Ja, NA=Missing
f61_02xx	Sind Sie zur Zeit - RealschülerIn 0=Nein, 1=Ja, NA=Missing
f61_03xx	Sind Sie zur Zeit - GymnasiastIn 0=Nein, 1=Ja, NA=Missing
f61_04xx	Sind Sie zur Zeit - SchülerIn einer anderen Schule 0=Nein, 1=Ja, NA=Missing
f61_05xx	Sind Sie zur Zeit - AuszubildendeR/BerufsschülerIn 0=Nein, 1=Ja, NA=Missing
f61_06xx	Sind Sie zur Zeit - StudentIn 0=Nein, 1=Ja, NA=Missing
f61_07xx	Sind Sie zur Zeit - angestellt 0=Nein, 1=Ja, NA=Missing
f61_08xx	Sind Sie zur Zeit - selbstständig 0=Nein, 1=Ja, NA=Missing
f61_09xx	Sind Sie zur Zeit - arbeitslos und arbeitssuchend 0=Nein, 1=Ja, NA=Missing
f61_10xx	Sind Sie zur Zeit - aus gesundheitl. Gründen nicht arbeitend 0=Nein, 1=Ja, NA=Missing
f61_11xx	Sind Sie zur Zeit - Hausfrau/Hausmann 0=Nein, 1=Ja, NA=Missing
f61_12xx	Sind Sie zur Zeit - sonstiges 0=Nein, 1=Ja, NA=Missing

Kommentar:
Die Variablen f61_01xx bis f61_12xx wurde bereits vom Datenzentrum aus den ursprünglichen Variablen f61_01 bis f61_12 kodiert und waren so im Datensatz enthalten, der als Grundlage für diese Arbeit galt. Wie bei der Kodierung vorgegangen wurde, zeigt folgende Tabelle.

Verwendete Variablen	**Abgeleitete Variablen**
f61_01 - f61_12	**f61_01xx - f61_12xx**
1	1
NA, aber nicht alle 12 Variablen NA	0
alle 12 Variablen NA	NA

Bildung der Variable BERUF

Kodierungsbeschreibung:
Hatte die dichotome Variable f61_01xx den Wert 1, so wurde die Variable BERUF auf 1 gesetzt. Hatte die dichotome Variable f61_02xx den Wert 1, so wurde die Variable BERUF auf 2 gesetzt. Nach diesem Schema wurde für alle Variablen vorgegangen.
In einigen Fällen nahmen mehr als eine dichotome Variable den Wert 1 an. Wie diese Sonderfälle kodiert wurden, kann folgender Tabelle entnommen werden.

Kodierungsübersicht - Sonderfälle:

Variablen	**Zugewiesener Wert für BERUF**
f61_01xx = 1 und f61_05xx = 1	5
f61_02xx = 1 und f61_12xx = 1	2
f61_03xx = 1 und f61_12xx = 1	3
f61_03xx = 1 und f61_07xx = 1	3
f61_03xx = 1 und f61_04xx = 1	3
f61_04xx = 1 und f61_09xx = 1	4
f61_05xx = 1 und f61_12xx = 1	5
f61_05xx = 1 und f61_10xx = 1	5
f61_05xx = 1 und f61_07xx = 1	5
f61_05xx = 1 und f61_12xx = 1	5

Code-Übersicht:

Variablenname	**Frage & Kodierung**
BERUF	Berufssituation - SOLAR
	1=HauptschülerIn
	2=RealschülerIn
	3=GymnasiastIn
	4=SchülerIn einer anderen Schule
	5=AuszubildendeR/BerufsschülerIn
	6=StudentIn
	7=Angestellt
	8=Selbstständig
	9=Arbeitslos und arbeitssuchend
	10=Aus gesundheitl. Gründen nicht arbeitend
	11=Hausfrau/Hausmann
	12=Sonstiges
	NA=Missing

R-Code:

```
> BERUF<-NA
> BERUF[!is.na(f61_01xx) & (f61_01xx==1)] <- 1
> BERUF[!is.na(f61_02xx) & (f61_02xx==1)] <- 2
> BERUF[!is.na(f61_03xx) & (f61_03xx==1)] <- 3
> BERUF[!is.na(f61_04xx) & (f61_04xx==1)] <- 4
> BERUF[!is.na(f61_05xx) & (f61_05xx==1)] <- 5
> BERUF[!is.na(f61_06xx) & (f61_06xx==1)] <- 6
> BERUF[!is.na(f61_07xx) & (f61_07xx==1)] <- 7
> BERUF[!is.na(f61_08xx) & (f61_08xx==1)] <- 8
> BERUF[!is.na(f61_09xx) & (f61_09xx==1)] <- 9
> BERUF[!is.na(f61_10xx) & (f61_10xx==1)] <- 10
> BERUF[!is.na(f61_11xx) & (f61_11xx==1)] <- 11
> BERUF[!is.na(f61_12xx) & (f61_12xx==1)] <- 12
> BERUF[!is.na(f61_01xx) & (f61_01xx==1)
+ & is.na(f61_05xx) & (f61_05xx==1)] <- 5
> BERUF[!is.na(f61_02xx) & (f61_02xx==1)
+ & !is.na(f61_12xx) & (f61_12xx==1)] <- 2
> BERUF[!is.na(f61_03xx) & (f61_03xx==1)
+ & !is.na(f61_12xx) & (f61_12xx==1)] <- 3
> BERUF[!is.na(f61_03xx) & (f61_03xx==1)
+ & !is.na(f61_07xx) & (f61_07xx==1)] <- 3
> BERUF[!is.na(f61_03xx) & (f61_03xx==1)
+ & !is.na(f61_04xx) & (f61_04xx==1)] <- 3
> BERUF[!is.na(f61_04xx) & (f61_04xx==1)
+ & !is.na(f61_09xx) & (f61_09xx==1)] <- 4
> BERUF[!is.na(f61_05xx) & (f61_05xx==1)
+ & !is.na(f61_12xx) & (f61_12xx==1)] <- 5
> BERUF[!is.na(f61_05xx) & (f61_05xx==1)
+ & !is.na(f61_10xx) & (f61_10xx==1)] <- 5
> BERUF[!is.na(f61_05xx) & (f61_05xx==1)
+ & !is.na(f61_07xx) & (f61_07xx==1)] <- 5
> BERUF[!is.na(f61_05xx) & (f61_05xx==1)
+ & !is.na(f61_12xx) & (f61_12xx==1)] <- 5
```

A.3 Variablen aus SOLAR II

A.3.1 Asthma

Diese Kodierung für SOLAR II entsprach der Kodierung der Variable CURASTHV in SOLAR.

Verwendete Variablen:

Variablenname	Frage & Kodierung
s2f07	Haben Sie jemals in den letzten 12 Monaten ein pfeifendes oder brummendes Geräusch in Ihrem Brustkorb gehört? 0=Nein, 1=Ja, NA=Keine Angabe
s2f19	Haben Sie jemals Asthma gehabt? 0=Nein, 1=Ja, NA=Keine Angabe
s2f20_01	Wurde bei Ihnen von einem Arzt schon einmal eine der folgenden Erkrankungen festgestellt? Asthma 0=Noch nie, 1=Einmal, 2=Mehrmals, NA=Keine Angabe
s2f20_02	Wurde bei Ihnen von einem Arzt schon einmal eine der folgenden Erkrankungen festgestellt? Spastische/asthmatische Bronchitis 0=Noch nie, 1=Einmal, 2=Mehrmals, NA=Keine Angabe

In fünf Schritten, die im Folgenden ausführlich dargestellt werden, wurde aus den soeben aufgeführten Variablen die Variable s2CURASTHV gebildet. Diese Variable zeigte an, ob bei einem Probanden zum damaligen Zeitpunkt Asthma vorlag oder nicht. Asthma lag bei einem Probanden vor, wenn diese Person bei sich selbst Asthmasymptome (pfeifendes oder brummendes Geräusch im Brustkorb) innerhalb der letzten 12 Monate beobachtet hatte und gleichzeitig eine Arztdiagnose Asthma oder spastische/asthmatische Bronchitis vorlag (d.h. Asthma bereits mindestens einmal oder spastische/asthmatische Bronchitis mehrmals von einem Arzt diagnostiziert wurde).

Gebildete Variable:

Variablenname	Frage & Kodierung
s2CURASTHV	Current Asthma (derzeit Asthma) 0=Nein, 1=Ja, NA=Missing

Schritt 1: Bildung der Variablen s2f20_01x und s2f20_02x

Kodierungsbeschreibung:
Den Variablen s2f20_01x und s2f20_02x wurden zunächst die Werte aus s2f20_01 bzw. s2f20_02 zugewiesen. Wurde Frage 19 mit 0 oder 1 beantwortet und wurde in Frage 20 auf mindestens eine der Variablen s2f20_01 oder s2f20_02 keine Angabe gegeben, so wurde die jeweilige Variable auf missing (NA) gesetzt. Fehlten die Angaben zu Frage 19 und 20 komplett, so wurden die Variablen s2f20_01x und s2f20_02x beide auf missing (NA) gesetzt. In allen anderen Fällen stimmten die Werte von s2f20_01x und s2f20_02x mit den Werten aus s2f20_01 und s2f20_02 überein.

Kodierungsübersicht:

Verwendete Variablen			**Abgeleitete Variablen**	
s2f19	**s2f20_01**	**s2f20_02**	**s2f20_01x**	**s2f20_02x**
0, 1, NA	0, 1, 2	0, 1, 2	0, 1, 2	0, 1, 2
0, 1	0, 1, 2	NA	0, 1, 2	NA
0, 1	NA	0, 1, 2	NA	0, 1, 2
0, 1	NA	NA	NA	NA
NA	NA	NA	NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
s2f20_01x	Asthma 0=Noch nie, 1=Einmal, 2=Mehrmals, NA=Missing
s2f20_02x	Spastische/asthmatische Bronchitis 0=Noch nie, 1=Einmal, 2=Mehrmals, NA=Missing

R-Code:

Diese Variable wurde bereits vom Datenzentrum kodiert und war so im Datensatz enthalten, der als Grundlage für diese Arbeit galt. Eine selbstständige Kodierung mit Hilfe von R war daher nicht nötig.

Schritt 2: Bildung der Hilfsvariablen s2f20_01n und s2f20_02n

Kodierungsbeschreibung:
Den Hilfsvariablen wurden zunächst die Werte aus s2f20_01x bzw. s2f20_02x zugewiesen. War in Frage 20 eine der Variablen s2f20_01x und s2f20_02x mit 0, 1 oder 2 beantwortet, so galt die komplette Frage als beantwortet. Für diesen Fall wurden die Hilfsvariablen s2f20_01n und s2f20_02n von missing (NA) auf 0 gesetzt. Man nahm in diesen Fällen an, dass die jeweilige Erkrankung noch nie von einem Arzt festgestellt wurde. In allen anderen Fällen stimmten die Werte von s2f20_01n und s2f20_02n mit den Werten aus s2f20_01x und s2f20_02x überein.

Kodierungsübersicht:

Verwendete Variablen		**Abgeleitete Variablen**	
s2f20_01x	**s2f20_02x**	**s2f20_01n**	**s2f20_02n**
0, 1, 2	0, 1, 2	0, 1, 2	0, 1, 2
0, 1, 2	NA	0, 1, 2	0
NA	0, 1, 2	0	0, 1, 2
NA	NA	NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
s2f20_01n	Asthma 0=Noch nie, 1=Einmal, 2=Mehrmals, NA=Missing
s2f20_02n	Spastische/asthmatische Bronchitis 0=Noch nie, 1=Einmal, 2=Mehrmals, NA=Missing

R-Code:

```
> s2f20_01n <- s2f20_01x
> s2f20_02n <- s2f20_02x
> s2f20_01n[is.na(s2f20_01x) & !is.na(s2f20_02x)] <- 0
> s2f20_02n[is.na(s2f20_02x) & !is.na(s2f20_01x)] <- 0
```

Schritt 3: Bildung der Variable s2ARASOBS

Kodierungsbeschreibung:
s2ARASOBS wurde auf 1 gesetzt, wenn s2f20_01n den Wert 1 oder 2 oder s2f20_02n den Wert 2 hatte (d.h. Asthma bereits mindestens einmal oder spastische/asthmatische Bronchitis mehrmals von einem Arzt festgestellt wurde).
s2ARASOBS wurde auf 0 gesetzt, wenn s2f20_01n den Wert 0 und s2f20_02n den Wert 1 oder 0 hatte.
In allen anderen Fällen war s2ARASOBS missing (NA).

Kodierungsübersicht:

Verwendete Variablen		**Abgeleitete Variablen**
s2f20_01n	**s2f20_02n**	**s2ARASOBS**
1, 2	0, 1, 2	1
0, 1, 2	2	1
0	0,1	0
NA	NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
s2ARASOBS	Arztdiagnose Asthma oder spastische/asthmatische Bronchitis 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> s2ARASOBS <- NA
> s2ARASOBS[!is.na(s2f20_01n) & (s2f20_01n==1)] <- 1
> s2ARASOBS[!is.na(s2f20_01n) & (s2f20_01n==2)] <- 1
> s2ARASOBS[!is.na(s2f20_02n) & (s2f20_02n==2)] <- 1
> s2ARASOBS[!is.na(s2f20_01n) & (s2f20_01n==0)
+ & !is.na(s2f20_02n) & (s2f20_02n==1)] <- 0
> s2ARASOBS[!is.na(s2f20_01n) & (s2f20_01n==0)
+ & !is.na(s2f20_02n) & (s2f20_02n==0)] <- 0
```

Schritt 4: Bildung der Variable s2KEUASOBV

Kodierungsbeschreibung:
In dieser Variable wurden die Informationen über das Vorliegen eines Asthmasymptoms aus den Fragen 7 und 20 kombiniert.
s2KEUASOBV wurde auf 1 gesetzt, wenn s2f07 und s2ARASOBS beide den Wert 1 hatten (d.h. Arztdiagnose Asthma oder spastische/asthmatische Bronchitis und gleichzeitig Asthmasymptome bei sich selbst beobachtet wurden).
s2KEUASOBV wurde auf 3 gesetzt, wenn s2f07 und s2ARASOBS beide den Wert 0 hatten (d.h. weder eine Arztdiagnose vorlag, noch Symptome bei sich selbst beobachtet wurden).
s2KEUASOBV wurde auf 2 gesetzt, wenn nur zu einer der beiden Variablen s2f07 und s2ARASOBS eine Angabe vorlag oder wenn sich die Angaben unterschieden.
s2KEUASOBV war nur dann missing (NA), wenn weder zu s2f07 noch zu s2ARASOBS Angaben vorlagen.

Kodierungsübersicht:

Verwendete Variablen		**Abgeleitete Variablen**
s2f07	**s2ARASOBS**	**s2KEUASOBV**
1	1	1
0	0	3
1	0	2
0	1	2
1, 0	NA	2
NA	0, 1	2
NA	NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
s2KEUASOBV	Wheezing und ARASOBS Asthmasymptome bei sich selbst beobachtet und gleichzeitig Arztdiagnose Asthma oder spastische/asthmatische Bronchitis 1=Positiv, 2=Intermediate, 3=Negativ, NA=Missing

R-Code:

```
> s2KEUASOBV <- 2
> s2KEUASOBV[!is.na(s2f07) & (s2f07==1)
+ & !is.na(s2ARASOBS) & (s2ARASOBS==1)] <- 1
> s2KEUASOBV[!is.na(s2f07) & (s2f07==0)
+ & !is.na(s2ARASOBS) & (s2ARASOBS==0)] <- 3
> s2KEUASOBV[is.na(s2f07) & is.na(s2ARASOBS)] <- NA
```

Schritt 5: Bildung der Variablen s2CURASTHV

Kodierungsbeschreibung:
s2CURASTHV wurde auf 1 gesetzt, wenn s2KEUASOBV mit 1 (positiv) kodiert war.
s2CURASTHV war missing (NA), wenn s2KEUASOBV missing (NA) war.
In allen anderen Fällen (s2KEUASOBV = 2 oder s2KEUASOBV = 3) stand in s2CURASTHV der Wert 0.

Kodierungsübersicht:

Verwendete Variablen	**Abgeleitete Variablen**
s2KEUASOBV	**s2CURASTHV**
1	1
2, 3	0
NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
s2CURASTHV	Current Asthma (derzeit Asthma) 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> s2CURASTHV <- 0
> s2CURASTHV[!is.na(s2KEUASOBV) & (s2KEUASOBV==1)] <- 1
> s2CURASTHV[is.na(s2KEUASOBV)] <- NA
```

A.3.2 Allergische Rhinitis

Diese Kodierung für SOLAR II entsprach der Kodierung der Variable CURHAYV in SOLAR.

Verwendete Variablen:

Variablenname	Frage & Kodierung
s2f26	Hatten Sie in den letzten 12 Monaten Probleme mit Niesanfällen oder einer laufenden, verstopften Nase, ohne erkältet zu sein? 0=Nein, 1=Ja, NA=Keine Angabe
s2f27	Traten diese Nasenprobleme zusammen mit juckenden, tränenden Augen auf? 0=Nein, 1=Ja, NA=Keine Angabe
s2f30	Hatten Sie in den letzten 12 Monaten allergischen Schupfen, z.B. “Heuschnupfen”? 0=Nein, 1=Ja, NA=Keine Angabe
s2f33	Hat ein Arzt bei Ihnen schon einmal allergischen Schnupfen, zum Beispiel “Heuschnupfen” festgestellt? 0=Nein, 1=Ja, NA=Keine Angabe

In drei Schritten, die im Folgenden ausführlich dargestellt werden, wurde aus den soeben aufgeführten Variablen die Variable s2CURHAYV gebildet. Diese Variable zeigte an, ob bei einem Probanden zu diesem Zeitpunkt Allergische Rhinitis vorlag oder nicht. Allergische Rhinitis lag bei einem Probanden vor, wenn bei dieser Person innerhalb der letzten 12 Monate Nasenprobleme (Niesanfälle oder laufende, verstopfte Nase ohne Erkältung) zusammen mit juckenden, tränenden Augen auftraten und gleichzeitig schon einmal von einem Arzt allergischer Schnupfen diagnostiziert wurde.

Gebildete Variable

Variablenname	Frage & Kodierung
s2CURHAYV	Current Hayfever (derzeit Allergische Rhinitis) 0=Nein, 1=Ja, NA=Missing

Schritt 1: Bildung der Variable s2f27xx

Kodierungsbeschreibung:
Der Variablen s2f27xx wurden zunächst die Werte aus s2f27 zugewiesen. War die Variable s2f27 missing (NA) und hatte s2f26 den Wert 0, so hatte s2f27xx ebenfalls den Wert 0. War die Variable s2f27 missing (NA) und hatte s2f26 den Wert 1 oder war missing (NA), so wurde die Variable s2f27xx auf missing (NA) gesetzt.

Kodierungsübersicht:

Verwendete Variablen		**Abgeleitete Variablen**
s2f26	**s2f27**	**s2f27xx**
0, 1, NA	0, 1	0, 1
0	NA	0
1, NA	NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
s2f27xx	Traten diese Nasenprobleme zusammen mit juckenden, tränenden Augen auf? 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> s2f27xx <- s2f27
> s2f27xx[!is.na(s2f26) & (s2f26==0) & is.na(s2f27)] <- 0
> s2f27xx[!is.na(s2f26) & (s2f26==1) & is.na(s2f27)] <- NA
> s2f27xx[is.na(s2f26) & is.na(s2f27)] <- NA
```

Schritt 2: Bildung der Variable s2f33xx

Kodierungsbeschreibung:
Der Variablen s2f33xx wurden zunächst die Werte aus s2f33 zugewiesen. War die Variable s2f33 missing (NA) und hatte s2f30 den Wert 0, so hatte s2f33xx ebenfalls den Wert 0. War die Variable s2f33 missing (NA) und hatte s2f30 den Wert 1 oder war missing (NA), so wurde die Variable s2f33xx auf missing (NA) gesetzt.

Kodierungsübersicht:

Verwendete Variablen		**Abgeleitete Variablen**
s2f30	**s2f33**	**s2f33xx**
0, 1, NA	0, 1	0, 1
0	NA	0
1, NA	NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
s2f33xx	Hat ein Arzt bei Ihnen schon einmal allergischen Schnupfen, zum Beispiel "Heuschnupfen" festgestellt? 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> s2f33xx <- s2f33
> s2f33xx[!is.na(s2f30) & (s2f30==0) & is.na(s2f33)] <- 0
> s2f33xx[!is.na(s2f30) & (s2f30==1) & is.na(s2f33)] <- NA
> s2f33xx[is.na(s2f30) & is.na(s2f33)] <- NA
```

Schritt 3: Bildung der Variable s2CURHAYV

Kodierungsbeschreibung:
s2CURHAYV wurde auf 1 gesetzt, wenn s2f27xx und s2f33xx beide den Wert 1 annahmen. s2CURHAYV wurde auf missing (NA) gesetzt, wenn s2f27xx und s2f33xx beide missing (NA) waren. In allen anderen Fällen stand in der Variablen s2CURHAYV der Wert 0.

Kodierungsübersicht:

Verwendete Variablen		**Abgeleitete Variablen**
s2f27xx	**s2f33xx**	**s2CURHAYV**
1	1	1
1,0	0,NA	0
0, NA	1,0	0
NA	NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
s2CURHAYV	Current Hayfever (derzeit Allergische Rhinitis) 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> s2CURHAYV <- 0
> s2CURHAYV[!is.na(s2f27xx) & (s2f27xx==1)
+ & !is.na(s2f33xx) & (s2f33xx==1)] <- 1
> s2CURHAYV[is.na(s2f27xx) & is.na(s2f33xx)] <- NA
```

A.3.3 Rauchverhalten

Die Angaben zum Rauchverhalten aus SOLAR II wurden so kodiert, dass man die Unterscheidung zwischen Raucher, Ex-Raucher und Nichtraucher treffen konnte. Nichtraucher wurden analog zu SOLAR definiert. In SOLAR II traff man (im Vergleich zu SOLAR) noch zusätzlich die Unterscheidung zwischen Raucher und Ex-Raucher, da man davon ausgehen konnte, dass sich bei den Probanden das Rauchverhalten im Gegensatz zum Zeitpunkt der SOLAR-Studie nun weitestgehend stabilisiert hatte. Raucher waren Personen, die schon einmal ein Jahr lang geraucht hatten und dies auch innerhalb des letzten Monats getan hatten. Ex-Raucher waren Personen, die zwar schon einmal ein Jahr lang geraucht hatten, aber dies nicht innerhalb des letzten Monats taten.

Verwendete Variablen:

Variablenname	**Frage & Kodierung**
s2f73	Haben Sie schon einmal ein Jahr lang geraucht? 0=Nein, 1=Ja, NA=Keine Angabe
s2f75	Haben Sie innerhalb des letzten Monats geraucht? 0=Nein, 1=Ja, NA=Keine Angabe

Bildung der Variable s2RAUCHEN

Kodierungsbeschreibung:
s2RAUCHEN wurde auf 0 (Nichtraucher) gesetzt, wenn s2f73 den Wert 0 hatte.
s2RAUCHEN wurde auf 2 (Ex-Raucher) gesetzt, wenn s2f73 den Wert 1 und s2f75 den Wert 0 hatte.
s2RAUCHEN wurde auf missing (NA) gesetzt, wenn sowohl s2f73 als auch s2f75 missing (NA) waren.
In allen anderen Fällen stand in der Variablen s2RAUCHEN der Wert 1 (Raucher).

Kodierungsübersicht:

Verwendete Variablen		**Abgeleitete Variablen**
s2f73	**s2f75**	**s2RAUCHEN**
0	1,0	0
1	1	1
NA	1	1
1	NA	1
1	0	2
NA	NA	NA

Code-Übersicht:

Variablenname	Frage & Kodierung
s2RAUCHEN	Rauchverhalten in SOLAR II 0=Nichtraucher, 1=Raucher, 2=Ex-Raucher, NA=Missing

R-Code:

```
> s2RAUCHEN <- 1
> s2RAUCHEN[is.na(s2f73) & is.na(s2f75)] <- NA
> s2RAUCHEN[!is.na(s2f73) & (s2f73==1)
+ & !is.na(s2f75) & (s2f75==0)] <- 2
> s2RAUCHEN[!is.na(s2f73) & (s2f73==0)] <- 0
```

A.3.4 Berufssituation

Die Angaben zur Berufssituation aus SOLAR II lagen ursprünglich als mehrere Dummy-Variablen vor. Diese Variablen wurden so kodiert, dass sie als eine kategoriale Variable vorlagen. Im Rahmen dieser Kodierung wurden zusätzlich Doppelnennungen korrigiert, die in dieser Frage nicht erlaubt waren. Wie mit diesen Doppelnennungen umzugehen war, wurde vorab gemeinsam mit Frau Kellberger und Herrn Heumann besprochen.

Verwendete Variablen:

Variablenname	**Frage & Kodierung**
s2f81_01	Sind Sie zur Zeit - AuszubildendeR/BerufsschülerIn 0=Nein, 1=Ja, NA=Missing
s2f81_02	Sind Sie zur Zeit - StudentIn (hauptberuflich) 0=Nein, 1=Ja, NA=Missing
s2f81_03	Sind Sie zur Zeit - Angestellt 0=Nein, 1=Ja, NA=Missing
s2f81_04	Sind Sie zur Zeit - Selbstständig 0=Nein, 1=Ja, NA=Missing
s2f81_05	Sind Sie zur Zeit - Arbeitslos und arbeitssuchend 0=Nein, 1=Ja, NA=Missing
s2f81_06	Sind Sie zur Zeit - Aus gesundheitl. Gründen nicht arbeitend 0=Nein, 1=Ja, NA=Missing
s2f81_07	Sind Sie zur Zeit - Hausmann/Hausfrau (hauptberuflich) 0=Nein, 1=Ja, NA=Missing
s2f81_08	Sind Sie zur Zeit - In Mutterschutz / Elternzeit oder sonstige Beurlaubung 0=Nein, 1=Ja, NA=Missing
s2f81_09	Sind Sie zur Zeit - Sonstiges 0=Nein, 1=Ja, NA=Missing

Bildung der Variable s2BERUF

Kodierungsbeschreibung:
Hatte die dichotome Variable s2f81_01 den Wert 1, so wurde die Variable s2BERUF auf 1 gesetzt. Hatte die dichotome Variable s2f81_02 den Wert 1, so wurde die Variable s2BERUF auf 2 gesetzt. Nach diesem Schema wurde für alle Variablen vorgegangen.
In einigen Fällen nahmen mehr als eine dichotome Variable den Wert 1 an. Wie diese Sonderfälle kodiert wurden, kann folgender Tabelle entnommen werden.

Kodierungsübersicht - Sonderfälle:

Variablen	**Zugewiesener Wert für s2BERUF**
s2f81_05 = 1 und s2f81_09 = 1 (Nebenjob)	3
s2f81_03 = 1 und s2f81_09 = 1	3
s2f81_02 = 1 und s2f81_09 = 1	2
s2f81_02 = 1 und s2f81_03 = 1	2
s2f81_02 = 1 und s2f81_04 = 1	2
s2f81_01 = 1 und s2f81_09 = 1	1
s2f81_01 = 1 und s2f81_08 = 1	1
s2f81_01 = 1 und s2f81_05 = 1	1
s2f81_01 = 1 und s2f81_04 = 1	1
s2f81_01 = 1 und s2f81_03 = 1	1

Code-Übersicht:

Variablenname	**Frage & Kodierung**
s2BERUF	Berufssituation - SOLAR II 1=AuszubildendeR/BerufsschülerIn 2=StudentIn (hauptberuflich) 3=Angestellt 4=Selbstständig 5=Arbeitslos und arbeitssuchend 6=Aus gesundheitl. Gründen nicht arbeitend 7=Hausfrau/Hausmann (hauptberuflich) 8=In Mutterschutz / Elternzeit oder sonstige Beurlaubung 9=Sonstiges

R-Code:

```
> s2BERUF<-NA
> s2BERUF[!is.na(s2f81_01) & (s2f81_01==1)] <- 1
> s2BERUF[!is.na(s2f81_02) & (s2f81_02==1)] <- 2
> s2BERUF[!is.na(s2f81_03) & (s2f81_03==1)] <- 3
> s2BERUF[!is.na(s2f81_04) & (s2f81_04==1)] <- 4
> s2BERUF[!is.na(s2f81_05) & (s2f81_05==1)] <- 5
> s2BERUF[!is.na(s2f81_06) & (s2f81_06==1)] <- 6
> s2BERUF[!is.na(s2f81_07) & (s2f81_07==1)] <- 7
> s2BERUF[!is.na(s2f81_08) & (s2f81_08==1)] <- 8
> s2BERUF[!is.na(s2f81_09) & (s2f81_09==1)] <- 9
> s2BERUF[!is.na(s2f81_05) & (s2f81_05==1)
+ & !is.na(s2f81_09) & (s2f81_09==1)] <- 3
> s2BERUF[!is.na(s2f81_03) & (s2f81_03==1)
+ & !is.na(s2f81_09) & (s2f81_09==1)] <- 3
> s2BERUF[!is.na(s2f81_02) & (s2f81_02==1)
+ & !is.na(s2f81_09) & (s2f81_09==1)] <- 2
> s2BERUF[!is.na(s2f81_02) & (s2f81_02==1)
+ & !is.na(s2f81_03) & (s2f81_03==1)] <- 2
> s2BERUF[!is.na(s2f81_02) & (s2f81_02==1)
+ & !is.na(s2f81_04) & (s2f81_04==1)] <- 2
> s2BERUF[!is.na(s2f81_01) & (s2f81_01==1)
+ & !is.na(s2f81_09) & (s2f81_09==1)] <- 1
> s2BERUF[!is.na(s2f81_01) & (s2f81_01==1)
+ & !is.na(s2f81_08) & (s2f81_08==1)] <- 1
> s2BERUF[!is.na(s2f81_01) & (s2f81_01==1)
+ & !is.na(s2f81_05) & (s2f81_05==1)] <- 1
> s2BERUF[!is.na(s2f81_01) & (s2f81_01==1)
+ & !is.na(s2f81_04) & (s2f81_04==1)] <- 1
> s2BERUF[!is.na(s2f81_01) & (s2f81_01==1)
+ & !is.na(s2f81_03) & (s2f81_03==1)] <- 1
```

A.3.5 Schulbildung

Die Angaben zur Schulbildung aus SOLAR II wurden so kodiert, dass man die Unterscheidung zwischen höherere und niedrigerer Schulbildung treffen konnte. "Höhere Schulbildung" hatte ein Proband, wenn er als höchsten Schulabschluss Fachhochschule, fachgebundene Hochschulreife, Abitur oder allgemeine Hochschulreife angab. Bei allen anderen Angaben wurde ihm "niedrigere Schulbildung" zugeordnet.

Verwendete Variablen:

Variablenname	Frage & Kodierung
s2f80	Welchen Schulabschluss haben Sie? Wenn Sie mehrere Abschlüsse haben, nennen Sie nur den höchsten! 0=Hauptschulabschluss/Volksschulabschluss(Mittelschule) 1=Realschulabschluss(mittlere Reife, Mittelschule) 2=Fachhochschulreife/fachgebundene Hochschulreife 3=Abitur/allgemeine Hochschulreife 4=Anderen Schulabschluss 5=Schule beendet ohne Abschluss 6=Noch keinen Schulabschluss

Bildung der Variable s2SCHULE

Kodierungsbeschreibung:
s2SCHULE wurde auf 1 (höhere Schulbildung) gesetzt, wenn s2f80 den Wert 2 (FH) oder 3 (Abitur) hatte.
s2SCHULE wurde auf missing (NA) gesetzt, wenn s2f80 missing (NA) war.
In allen anderen Fällen stand in der Variablen s2SCHULE der Wert 0 (niedrigere Schulbildung).

Kodierungsübersicht:

Variablen	**Zugewiesener Wert für s2SCHULE**
s2f80	**s2SCHULE**
2,3	1
0,1,4,5,6	0
NA	NA

Code-Übersicht:

Variablenname	Frage & Kodierung
s2SCHULE	Schulbildung in SOLAR II 0=Niedrigere Schulbildung, 1=Höhere Schulbildung, NA=Missing

R-Code:

```
> s2SCHULE <- 0
> s2SCHULE[is.na(s2f80)] <- NA
> s2SCHULE[!is.na(s2f80) & (s2f80==2)] <- 1
> s2SCHULE[!is.na(s2f80) & (s2f80==3)] <- 1
```

A.4 Benötigte Variablen für die Tätigkeitsangaben

Zu den beruflichen Tätigkeiten wurden in SOLAR und SOLAR II jeweils zwei Fragen gestellt. Alle im weiteren durchgeführten Rekodierungen basierten auf diesen Fragen.

Verwendete Fragen aus SOLAR:

Fragennummer	**Frage & Kodierung**
Frage 65	Haben Sie schon einmal irgendeine Arbeit / irgendeinen Ferienjob gehabt? 0=Nein, 1=Ja, NA=Missing
Frage 66	Welche Art von Arbeitsstellen und/oder Ferienjobs etc. hatten Sie bis jetzt (mind. 1 Monat lang, mind. 8 Stunden pro Woche)? offene Angaben zu Tätigkeit, Branche, Beginn und Ende der Tätigkeit sowie zu den Stunden pro Woche

Verwendete Fragen aus SOLAR II:

Fragennummer	**Frage & Kodierung**
Frage 92	Haben Sie seit der letzten SOLAR-Studie (2003/2004) irgendeine Arbeit/irgendeinen Ferienjob für mindestens 1 Monat gehabt? 0=Nein, 1=Ja, NA=Missing
Frage 93	Welche Art von Arbeitsstellen und/oder Ferienjobs etc. hatten Sie seit der letzten SOLAR-Studie (2003/2004) (mind. 1 Monat lang, mind. 8 Stunden pro Woche)? offene Angaben zu Tätigkeit, Branche, Beginn und Ende der Tätigkeit sowie zu den Stunden pro Woche

A.4.1 Gearbeitet in SOLAR

Eine Person, die mindestens einen Eintrag bei den Tätigkeitsangaben machte (Frage 66) musste auch die vorherigen Frage (Frage 65) bejahen, ob in diesem Zeitraum gearbeitet wurde. Bei den Probanden, bei denen das nicht der Fall war, wurde die Variable f65x in einer neuen Variable f65xx entsprechend korrigiert. Diese Variable gab an, ob überhaupt gearbeitet wurde, unabhängig davon, wie viele Wochenstunden gearbeitet wurden.

Bildung der Variablen f65xx und GEARB_s1 (pro Proband)

Kodierungsbeschreibung:
Der Variablen f65xx wurden zunächst die Werte aus f65x zugewiesen. Wurde mindestens eine Tätigkeitsangaben in Frage 66 getätigt (d.h. Variable n_jobs ≥ 1), so wurde die Variable f65xx auf 1 gesetzt (wenn sie zuvor nicht bereits den Wert 1 hatte). Alle Werte aus f65xx wurden dann in die Variable GEARB_s1 kopiert.

Kodierungsübersicht:

Verwendete Variablen		**Abgeleitete Variablen**	
f65x	**Frage66**	**f65xx**	**GEARB_s1**
0	NA (d.h. n_jobs = 0)	0	0
0	Angaben (d.h. n_jobs ≥ 1)	1	1
1	Angaben (d.h. n_jobs ≥ 1)	1	1
NA	NA	NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
f65xx	Haben Sie schon einmal irgendeine Arbeit / irgendeinen Ferienjob gehabt? 0=Nein, 1=Ja, NA=Missing
GEARB_s1	Haben Sie schon einmal irgendeine Arbeit / irgendeinen Ferienjob gehabt? 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> f65xx <- f65x
> f65xx[(n_jobs==1) & (f65x==0)] <- 1
Alle anderen Kombinationen waren bereits richtig kodiert. Die Übertragung der Variable f
im Rahmen der Datensatzerstellung.
```

A.4.2 Gearbeitet in SOLAR II

Eine Person, die mindestens einen Eintrag bei den Tätigkeitsangaben machte (Frage 93), musste auch die vorherigen Frage (Frage 92) bejahen, ob in diesem Zeitraum gearbeitet wurde. Bei den Probanden, bei denen das nicht der Fall war, wurde die Variable s2f92 in einer neuen Variable GEARB_s2 entsprechend korrigiert. Diese Variable gab an, ob überhaupt gearbeitet wurde, unabhängig davon, wie viele Wochenstunden gearbeitet wurden.

Bildung der Variable GEARB_s2 (pro Proband)

Kodierungsbeschreibung:
Der Variable GEARB_s2 wurden zunächst die Werte aus s2f92 zugewiesen. Wurde mindestens eine Tätigkeitsangaben in Frage 93 getätigt, so wurde die Variable GEARB_s2 auf 1 gesetzt (wenn sie zuvor nicht bereits den Wert 1 hatte). Wurde in Frage 93 die Antwortoption "Keine Tätigkeit für mindestens 8 Stunden pro Woche ausgeführt" genutzt (s2f93_01 = 1), so wurde GEARB_s2 ebenfalls mit 1 kodiert.

Kodierungsübersicht:

Verwendete Variablen		**Abgeleitete Variablen**
s2f92	**Frage93**	**GEARB_s2**
0	NA	0
0	Angaben	1
1	Angaben	1
NA	NA	NA
1,0,NA	s2f93_01 = 1	1

Code-Übersicht:

Variablenname	**Frage & Kodierung**
GEARB_s2	Haben Sie seit der letzten SOLAR-Studie (2003/2004) irgendeine Arbeit/irgendeinen Ferienjob für mindestens 1 Monat gehabt? 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> GEARB_s2 <- s2f92
> GEARB_s2[(knr=="A59355296")|(knr=="B56628279")|(knr=="B58427289")
+ |(knr=="B59327287")|(knr=="D56158285")|(knr=="K54357297")
+ |(knr=="N53657299")|(knr=="N55156285")|(knr=="N58928280")
+ |(knr=="N59156296")|(knr=="P53858298")|(knr=="P56028289")
+ |(knr=="P59827272")|(knr=="R50927282")|(knr=="R59827287")
+ |(knr=="S52627272")|(knr=="S53151284")|(knr=="T54857294")
+ |(knr=="T56158280")|(knr=="T57626276")|(knr=="U56726296")
+ |(knr=="U57955278")|(knr=="U59828294")|(knr=="D54726284")
+ |(knr=="P56957272")|(knr=="P58027298")|(knr=="U53256284")] <- 1
> GEARB_s2[s2f93_01==1] <- 1
```

A.4.3 Ende der Tätigkeit in SOLAR

Bei den Datensätzen, bei denen nur die Angaben zum Ende der Tätigkeit fehlte und die restlichen Tätigkeitsangaben vollständig waren (d.h. ISCO-Code, Anfang der Tätigkeit und Wochenstunden), wurd davon ausgegangen, dass diese Person die Tätigkeit zum Zeitpunkt der Befragung noch ausführte. Aus diesem Grund wurde hierfür ein Ersatzende eingesetzt. Als Ersatzende wurde falls Vorhanden das Ausfülldatum, ansonsten das Einscanndatum des Fragebogens verwendet.

Bildung der Variablen END_MONATx und END_JAHRx (pro Tätigkeit)

Kodierungsbeschreibung:
In die neue Variable END_MONATx bzw. END_JAHRx wurde zunächst das tatsächlich angegebenen Ende der Tätigkeit eingefügt (END_MONAT und END_JAHR). Fehlte diese Angabe, so wurde das Ersatzende verwendet (s1Ersatzende_Monat und s1Ersatzende_Jahr).

Kodierungsübersicht:

Verwendete Variablen		**Abgeleitete Variablen**
END_MONAT **END_JAHR**	**s1Ersatzende_Monat** **s1Ersatzende_Jahr**	**END_MONATx** **END_JAHRx**
Angaben	Ersatzende	Angaben
NA	Ersatzende	Ersatzende
NA	NA	NA

R-Code:

```
> END_MONATx <- END_MONAT
> END_JAHRx <- END_JAHR
> for (i in 1:nrow()){
+ if((ISCO[i]!="9999") & !is.na(ANF_MONAT[i]) & !is.na(ANF_JAHR[i])
+ & is.na(END_MONAT[i]) & is.na(END_JAHR[i])  & !is.na(WST[i])){
+ END_MONATx[i] <- s1Ersatzende_Monat[i]
+ END_JAHRx[i] <- s1Ersatzende_Jahr[i]
+ }
+ }
```

A.4.4 Ende der Tätigkeit in SOLAR II

Bei den Datensätzen, bei denen nur die Angaben zum Ende der Tätigkeit fehlte und die restlichen Tätigkeitsangaben vollständig waren (d.h. ISCO-Code, Anfang der Tätigkeit und Wochenstunden), wurde davon ausgegangen, dass diese Person die Tätigkeit zum Zeitpunkt der Befragung noch ausführte. Aus diesem Grund wurde hierfür ein Ersatzende eingesetzt. Als Ersatzende wurde falls Vorhanden das Ausfülldatum, ansonsten das Einscanndatum des Fragebogens verwendet.

Bildung der Variablen END_MONATx und END_JAHRx (pro Tätigkeit)

Kodierungsbeschreibung:
In die neue Variable END_MONATx bzw. END_JAHRx wurde zunächst das tatsächlich angegebenen Ende der Tätigkeit (Ende_Monat und Ende_Jahr) eingefügt. Fehlte diese Angabe, so wurde das Ersatzende verwendet (s2Ersatzende_Monat und s2Ersatzende_Jahr).

Kodierungsübersicht:

Verwendete Variablen		**Abgeleitete Variablen**
Ende_Monat **Ende_Jahr**	**s2Ersatzende_Monat** **s2Ersatzende_Jahr**	**END_MONATx** **END_JAHRx**
Angaben	Ersatzende	Angaben
NA	Ersatzende	Ersatzende
NA	NA	NA

R-Code:

```
> END_MONATx <- Ende_Monat
> END_JAHRx <- Ende_Jahr
> for (i in 1:nrow()){
+ if((ISCO[i]!="9999") & (ISCO[i]!="8888") & !is.na(Beginn_Monat[i])
+ & !is.na(Beginn_Jahr[i]) & is.na(Ende_Monat[i]) & is.na(Ende_Jahr[i])
+ & !is.na(Wochenstunden[i])){
+ END_MONATx[i] <- s2Ersatzende_Monat[i]
+ END_JAHRx[i] <- s2Ersatzende_Jahr[i]
+ }
+ }
```

A.4.5 Jemals (mindestens acht Wochenstunden) gearbeitet in SOLAR und SOLAR II

In drei Schritten, die im Folgenden ausführlich dargestellt werden, wurde aus den Tätigkeitsangaben aus SOLAR und SOLAR II die Variable JEMALS_GEARB gebildet. Jemand, der während SOLAR und SOLAR II mindestens eine Tätigkeit mit mindestens acht Wochenstunden durchgeführte, wurde hier mit "Ja" kodiert.

Gebildete Variable:

Variablenname	**Frage & Kodierung**
JEMALS_GEARB	Jemals (mind. acht Wochenstunden) gearbeitet in SOLAR und SOLAR II 0=Nein, 1=Ja

Schritt 1: Bildung der Variable MIND_8WST (pro Tätigkeit)

Kodierungsbeschreibung:
Betrugen die angegebenen Wochenstunden der jeweiligen Tätigkeit mindestens acht Stunden, so wurde die Variable MIND_8WST für diese Tätigkeit auf 1 gesetzt. Ansonsten wurde die Variable auf 0 gesetzt. Fehlte die Angabe zu den Wochenstunden, so wurde auch die Variable MIND_8WST auf NA gesetzt.

Kodierungsübersicht:

Verwendete Variablen	**Abgeleitete Variablen**
WST	**MIND_8WST**
< 8	0
≥ 8	1
NA	NA

Code-Übersicht:

Variablenname	**Frage & Kodierung**
MIND_8WST	jeweilige Tätigkeit mind. 8 Wochenstunden ausgeführt 0=Nein, 1=Ja, NA=Missing

R-Code:

```
> MIND_8WST <- 0
> MIND_8WST[!is.na(WST)&WST>=8]<-1
> MIND_8WST[is.na(WST)]<-NA
```

Schritt 2: Bildung der Variable SUM_BERUF_MIND_8WST (pro Proband)

Kodierungsbeschreibung:
Alle Tätigkeiten eines Probanden, die mindestens acht Wochenstunden durchgeführt wurden, wurden aufsummiert und in der Variable SUM_BERUF_MIND_8WST abgespeichert.

Code-Übersicht:

Variablenname	Frage & Kodierung
SUM_BERUF_MIND_8WST	Summe der Tätigkeiten, die mind. acht Wochenstunden ausgeführt wurden 0-10=Summe

R-Code:

```
> j <- 1
> i <- 1
> while (j <= nrow()) {
+ SUMME <- sum(MIND_8WST[j],MIND_8WST[j+1],MIND_8WST[j+2],
+ MIND_8WST[j+3],MIND_8WST[j+4],MIND_8WST[j+5],MIND_8WST[j+6],
+ MIND_8WST[j+7],MIND_8WST[j+8],MIND_8WST[j+9],na.rm=TRUE)
+ SUM_BERUF_MIND_8WST[i] <- SUMME
+ SUM_BERUF_MIND_8WST[i+1] <- SUMME
+ SUM_BERUF_MIND_8WST[i+2] <- SUMME
+ SUM_BERUF_MIND_8WST[i+3] <- SUMME
+ SUM_BERUF_MIND_8WST[i+4] <- SUMME
+ SUM_BERUF_MIND_8WST[i+5] <- SUMME
+ SUM_BERUF_MIND_8WST[i+6] <- SUMME
+ SUM_BERUF_MIND_8WST[i+7] <- SUMME
+ SUM_BERUF_MIND_8WST[i+8] <- SUMME
+ SUM_BERUF_MIND_8WST[i+9] <- SUMME
+ j <- j+10
+ i <- i+10
+ }
```

Schritt 3: Bildung der Variable JEMALS_GEARB (pro Proband)

Kodierungsbeschreibung:
Hatte ein Proband mindestens eine Tätigkeit mit mindestens acht Wochenstunden während der Studien SOLAR oder SOLAR II durchgeführt, so wurde die Variable JEMALS_GEARB für denjenigen Probanden auf 1 gesetzt. Ansonsten wurde sie auf 0 gesetzt.

Kodierungsübersicht:

Verwendete Variablen	**Abgeleitete Variablen**
SUM_BERUF_MIN_8WST	**JEMALS_GEARB**
0	0
≥ 0	1

Code-Übersicht:

Variablenname	**Frage & Kodierung**
JEMALS_GEARB	Jemals (mind. acht Wochenstunden) gearbeitet in SOLAR und SOLAR II 0=Nein, 1=Ja

R-Code:

```
> JEMALS_GEARB <- 0
> JEMALS_GEARB[SUM_BERUF_MIND_8WST>0] <- 1
```

A.4.6 Anzahl Tätigkeitsangaben in SOLAR und SOLAR II

In mehreren Schritten, die im Folgenden ausführlich dargestellt werden, wurde aus den Tätigkeitsangaben aus SOLAR und SOLAR II die Variable SUM_ANZAHL_EINTRAEGE gebildet, die angab, wie viele Tätigkeiten der Proband nannte (unabhängig davon, wie viele Wochenstunden gearbeitet wurden).

Gebildete Variable:

Variablenname	**Frage & Kodierung**
SUM_ANZAHL_EINTRAEGE	Anzahl der Tätigkeiten (unabhängig von der Wochenstundenanzahl) 0-10=Summe

Schritt 1: Bildung der Variable ANZAHL_EINTRAEGE (pro Tätigkeit) in SOLAR

Kodierungsbeschreibung:
Die Variable ANZAHL_EINTRAEGE gab wieder, ob ein Proband in einer Zeile Tätigkeitsangaben gemacht hatte (=1) oder nicht (=0).

Kodierungsübersicht:

Verwendete Variablen	**Abgeleitete Variablen**
ISCO	**ANZAHL_EINTRAEGE**
9999	0
alle anderen Codes	1

Code-Übersicht:

Variablenname	**Frage & Kodierung**
ANZAHL_EINTRAEGE	Tätigkeit angegeben 0=Nein, 1=Ja

R-Code:

```
> ANZAHL_EINTRAEGE <- 0
> ANZAHL_EINTRAEGE[ISCO != 9999] <- 1
```

Schritt 2: Bildung der Variable ANZAHL_EINTRAEGE (pro Tätigkeit) in SOLAR II

Kodierungsbeschreibung:
Die Variable ANZAHL_EINTRAEGE gab wieder, ob ein Proband in einer Zeile Tätigkeitsangaben gemacht hatte (=1) oder nicht (=0).

Kodierungsübersicht:

Verwendete Variablen	**Abgeleitete Variablen**
ISCO	**ANZAHL_EINTRAEGE**
8888	0
9999	0
alle anderen Codes	1

Zusätzlich wurde die Variable ANZAHL_EINTRAEGE auf 1 gesetzt, wenn die Antwortoption "Keine Tätigkeit für mindestens 8 Stunden pro Woche ausgeführt" angegeben wurde (s2f93_01 = 1).

Code-Übersicht:

Variablenname	**Frage & Kodierung**
ANZAHL_EINTRAEGE	Tätigkeit angegeben 0=Nein, 1=Ja

R-Code:

```
> ANZAHL_EINTRAEGE <- 0
> ANZAHL_EINTRAEGE[ISCO != 9999 & ISCO != 8888] <- 1
> ANZAHL_EINTRAEGE[(s2f93_01==1)] <- 1
```

Schritt 3: Bildung der Variable SUM_ANZAHL_EINTRAEGE_s1 (pro Proband) für SOLAR

Kodierungsbeschreibung:
Alle Tätigkeiten eines Probanden innerhalb von SOLAR (unabhängig von der Anzahl der Wochenstunden) wurden aufsummiert und in der Variable SUM_ANZAHL_EINTRAEGE_s1 abgespeichert.

Code-Übersicht:

Variablenname	Frage & Kodierung
SUM_ANZAHL_EINTRAEGE_s1	Anzahl der Tätigkeiten in SOLAR (unabhängig von der Wochenstundenanzahl) 0-5=Summe

R-Code:

```
> j <- 1
> i <- 1
> while (j <= nrow()) {
+ SUMME <- sum(ANZAHL_EINTRAEGE[j],ANZAHL_EINTRAEGE[j+1],
+ ANZAHL_EINTRAEGE[j+2],ANZAHL_EINTRAEGE[j+3],ANZAHL_EINTRAEGE[j+4],
+ na.rm=TRUE)
+ SUM_ANZAHL_EINTRAEGE_s1[i] <- SUMME
+ SUM_ANZAHL_EINTRAEGE_s1[i+1] <- SUMME
+ SUM_ANZAHL_EINTRAEGE_s1[i+2] <- SUMME
+ SUM_ANZAHL_EINTRAEGE_s1[i+3] <- SUMME
+ SUM_ANZAHL_EINTRAEGE_s1[i+4] <- SUMME
+ j <- j+5
+ i <- i+5
+ }
```

Schritt 4: Bildung der Variable SUM_ANZAHL_EINTRAEGE_s2 (pro Proband) für SOLAR II

Kodierungsbeschreibung:
Alle Tätigkeiten eines Probanden innerhalb von SOLAR II (unabhängig von der Anzahl der Wochenstunden) wurden aufsummiert und in der Variable SUM_ANZAHL_EINTRAEGE_s2 abgespeichert.

Code-Übersicht:

Variablenname	Frage & Kodierung
SUM_ANZAHL_EINTRAEGE_s2	Anzahl der Tätigkeiten in SOLAR II (unabhängig von der Wochenstundenanzahl) 0-5=Summe

R-Code:

```
> j <- 1
> i <- 1
> while (j <= nrow()) {
+ SUMME <- sum(ANZAHL_EINTRAEGE[j],ANZAHL_EINTRAEGE[j+1],
+ ANZAHL_EINTRAEGE[j+2],ANZAHL_EINTRAEGE[j+3],ANZAHL_EINTRAEGE[j+4],
+ na.rm=TRUE)
+ SUM_ANZAHL_EINTRAEGE_s2[i] <- SUMME
+ SUM_ANZAHL_EINTRAEGE_s2[i+1] <- SUMME
+ SUM_ANZAHL_EINTRAEGE_s2[i+2] <- SUMME
+ SUM_ANZAHL_EINTRAEGE_s2[i+3] <- SUMME
+ SUM_ANZAHL_EINTRAEGE_s2[i+4] <- SUMME
+ j <- j+5
+ i <- i+5
+ }
```

Schritt 5: Bildung der Variable SUM_ANZAHL_EINTRAEGE (pro Proband) für SOLAR und SOLAR II

Kodierungsbeschreibung:
Die Anzahl der Tätigkeiten aus SOLAR (SUM_ANZAHL_EINTRAEGE_s1) und SOLAR II (SUM_ANZAHL_EINTRAEGE_s2) wurden aufsummiert und in der Variable SUM_ANZAHL_EINTRAEGE abgespeichert.

Code-Übersicht:

Variablenname	**Frage & Kodierung**
SUM_ANZAHL_EINTRAEGE	Anzahl der Tätigkeiten (unabhängig von der Wochenstundenanzahl) 0-10=Summe

R-Code:

```
> SUM_ANZAHL_EINTRAEGE <- SUM_ANZAHL_EINTRAEGE_s1 + SUM_ANZAHL_EINTRAEGE_s2
```

A.4.7 Dauer der Tätigkeit

Die Dauer der jeweiligen Tätigkeit wurde berechnet, indem jeweils das Ende der Tätigkeit vom Anfang der Tätigkeit abgezogen wurde. Die genaue Berechnung musste in einer längeren Schleife programmiert werden und ist deshalb nicht hier dargestellt, sondern kann dem R-Code (auf der beigelegten CD) entnommen werden.

A.4.8 Zeilen mit vollständig ausgefüllten Tätigkeitsangaben

In mehreren Schritten, die im Folgenden ausführlich dargestellt werden, wurde aus den Tätigkeitsangaben aus SOLAR und SOLAR II die Variable SUM_ZEILE_VOLLST gebildet, die angab, wie viele Zeilen mit Tätigkeitsangaben der Proband vollständig ausgefüllt hatte. Eine Zeile galt als vollständig ausgefüllt, wenn der ISCO-Code 94, 95, 97 oder 98 auftratt, nie gearbeitet wurde oder wenn alle Jobs, die angegeben wurden, vollständig ausgefüllt waren (d.h. Angaben zu Beginn und Ende der Tätigkeit, Wochenstunden und der ISCO-Code vorlagen). Weiterhin galt die Zeile als vollständig, wenn weniger als acht Wochenstunden gearbeitet wurde oder die Antwortoption "Keine Tätigkeit für mindestens 8 Stunden pro Woche ausgeführt" ausgewählt wurde.

Gebildete Variable:

Variablenname	**Frage & Kodierung**
SUM_ZEILE_VOLLST	Summe der vollständig ausgefüllten Zeilen 0-10=Summe

Schritt 1: Bildung der Variable ZEILE_VOLLST (pro Tätigkeit) in SOLAR

Kodierungsbeschreibung:
Zunächst wurde eine Hilfsvariable ZEILE_VOLLST für SOLAR gebildet, die angab, ob die Zeile vollständig ausgefüllt wurde (=1) oder später imputiert werden musste (=0).

Code-Übersicht:

Variablenname	Frage & Kodierung
ZEILE_VOLLST	Zeile vollständig ausgefüllt 0=Nein, 1=Ja

R-Code:

```
> ZEILE_VOLLST <- 0
> ZEILE_VOLLST[!is.na(ANF_JAHR) & !is.na(ANF_MONAT) & !is.na(END_JAHRx)
+ & !is.na(END_MONATx) & !is.na(WST) & !is.na(ISCO)] <- 1
> ZEILE_VOLLST[ISCO==94 | ISCO==95 | ISCO==98 | ISCO==97] <- 1
> ZEILE_VOLLST[WST<8] <- 1
```

Schritt 2: Bildung der Variable ZEILE_VOLLST (pro Tätigkeit) in SOLAR II

Kodierungsbeschreibung:
Zunächst wurde eine Hilfsvariable ZEILE_VOLLST für SOLAR II gebildet, die angab, ob die Zeile vollständig ausgefüllt wurde (=1) oder später imputiert werden musste (=0).

Code-Übersicht:

Variablenname	Frage & Kodierung
ZEILE_VOLLST	Zeile vollständig ausgefüllt 0=Nein, 1=Ja

R-Code:

```
> ZEILE_VOLLST <- 0
> ZEILE_VOLLST[!is.na(ANF_JAHR) & !is.na(ANF_MONAT) & !is.na($END_JAHRx)
+ & !is.na(END_MONATx) & !is.na(WST) & !is.na(ISCO)] <- 1
> ZEILE_VOLLST[$ISCO==94 | ISCO==95 | ISCO==98 | ISCO==97] <- 1
> ZEILE_VOLLST[WST<8] <- 1
> ZEILE_VOLLST[(s2f93_01==1)] <- 1
```

Schritt 3: Bildung der Variable SUM_ZEILE_VOLLST_s1 (pro Proband) für SOLAR

Kodierungsbeschreibung:
Aus der Hilfsvariable ZEILE_VOLLST wurde dann für SOLAR eine Variable SUM_ZEILE_VOLLST_s1 gebildet, die pro Probanden alle fünf möglichen Zeilen aufsummierte und die Summe der vollständig ausgefüllten Zeilen enthielt.

Code-Übersicht:

Variablenname	Frage & Kodierung
SUM_ZEILE_VOLLST_s1	Summe der vollständig ausgefüllten Zeilen in SOLAR 0-5=Summe

R-Code:

```
> j <- 1
> i <- 1
> while (j <= nrow()) {
+ SUMME <- sum(ZEILE_VOLLST[j],ZEILE_VOLLST[j+1],ZEILE_VOLLST[j+2],
+ ZEILE_VOLLST[j+3],ZEILE_VOLLST[j+4],na.rm=TRUE)
+ SUM_ZEILE_VOLLST_s1[i] <- SUMME
+ SUM_ZEILE_VOLLST_s1[i+1] <- SUMME
+ SUM_ZEILE_VOLLST_s1[i+2] <- SUMME
+ SUM_ZEILE_VOLLST_s1[i+3] <- SUMME
+ SUM_ZEILE_VOLLST_s1[i+4] <- SUMME
+ j <- j+5
+ i <- i+5
+ }
```

Schritt 4: Bildung der Variable SUM_ZEILE_VOLLST_s2 (pro Proband) für SOLAR II

Kodierungsbeschreibung:
Aus der Hilfsvariable ZEILE_VOLLST wurd dann für SOLAR II eine Variable SUM_ZEILE_VOLLST_s2 gebildet, die pro Probanden alle fünf möglichen Zeilen aufsummierte und die Summe der vollständig ausgefüllten Zeilen enthielt.

Code-Übersicht:

Variablenname	Frage & Kodierung
SUM_ZEILE_VOLLST_s2	Summe der vollständig ausgefüllten Zeilen in SOLAR II 0-5=Summe

R-Code:

```
> j <- 1
> i <- 1
> while (j <= nrow()) {
+ SUMME <- sum(ZEILE_VOLLST[j],ZEILE_VOLLST[j+1],ZEILE_VOLLST[j+2],
+ ZEILE_VOLLST[j+3],ZEILE_VOLLST[j+4],na.rm=TRUE)
+ SUM_ZEILE_VOLLST_s2[i] <- SUMME
+ SUM_ZEILE_VOLLST_s2[i+1] <- SUMME
+ SUM_ZEILE_VOLLST_s2[i+2] <- SUMME
+ SUM_ZEILE_VOLLST_s2[i+3] <- SUMME
+ SUM_ZEILE_VOLLST_s2[i+4] <- SUMME
+ j <- j+5
+ i <- i+5
+ }
```

Schritt 5: Bildung der Variable SUM_ZEILE_VOLLST (pro Proband) für SOLAR und SOLAR II

Kodierungsbeschreibung:
Aus den Variablen SUM_ZEILE_VOLLST_s1 und SUM_ZEILE_VOLLST_s2 wurde dann eine gemeinsame Variable SUM_ZEILE_VOLLST gebildet, die pro Probanden alle zehn möglichen Zeilen aus SOLAR und SOLAR II aufsummierte und die Summe der vollständig ausgefüllten Zeilen enthielt.

Code-Übersicht:

Variablenname	**Frage & Kodierung**
SUM_ZEILE_VOLLST	Summe der vollständig ausgefüllten Zeilen 0-10=Summe

R-Code:

```
> SUM_ZEILE_VOLLST <- SUM_ZEILE_VOLLST_s1 + SUM_ZEILE_VOLLST_s2
```

A.4.9 Probanden mit vollständig ausgefüllten Tätigkeitsangaben

In mehreren Schritten, die im Folgenden ausführlich dargestellt werden, wurde aus den Tätigkeitsangaben aus SOLAR und SOLAR II die Variable PROB_VOLLST gebildet, die angab, ob ein Proband alle gemachten Tätigkeitsangaben vollständig ausgefüllt hatte. Die Probanden die alle gemachten Tätigkeitsangaben vollständig ausgefüllt hatten, erhielten in der Variable PROB_VOLLST den Eintrag 1. Fehlte mindestens eine Tätigkeitsangabe, so erhielt der Proband den Eintrag 0. Bei all diesen Probanden mit Eintrag 0 konnten in einem späteren Schritt die fehlenden Angaben imputiert werden.

Gebildete Variable:

Variablenname	**Frage & Kodierung**
PROB_VOLLST	Proband mit vollständig ausgefüllten Tätigkeitsangaben 0=Nein, 1=Ja

Schritt 1: Bildung der Variable PROB_VOLLST_s1 (pro Proband) in SOLAR

Kodierungsbeschreibung:
Zunächst wurde eine Hilfsvariable PROB_VOLLST_s1 für SOLAR gebildet, die angab, ob der Proband in SOLAR alle gemachten Tätigkeitsangaben vollständig ausgefüllt hatte. Ein Proband galt als vollständig, wenn er in SOLAR nicht gearbeitet hatte und auch keine Tätigkeitsangaben gemacht hatte. Wurde in SOLAR gearbeitet, so musste die Anzahl der ausgefüllten Zeilen und die Anzahl der vollständig ausgefüllten Zeilen bei diesen Probanden übereinstimmen (SUM_ANZAHL_EINTRAEGE = SUM_ZEILE_VOLLST). War das der Fall, galten sie als vollständig. Weiterhin galten die Probanden als vollständig, wenn zwar gearbeitet wurde, aber keine Einträge vorlagen oder wenn die Angabe, ob gearbeitet wurde komplett fehlte. In diesen Fällen wurde konservativ vorgegangen und somit keine Tätigkeiten und mögliche Expositionen imputiert. Als unvollständig galt ein Proband in SOLAR, wenn die gemachten Tätigkeitsangaben fehlende Werte aufwiesen.

Code-Übersicht:

Variablenname	Frage & Kodierung
PROB_VOLLST_s1	Proband vollständig in SOLAR 0=Nein, 1=Ja

R-Code:

```
> PROB_VOLLST_s1 <- NA
> PROB_VOLLST_s1[GEARB_s1 == 0] <- 1
> PROB_VOLLST_s1[(GEARB_s1==1) & (SUM_ANZAHL_EINTRAEGE_s1!=0)
+ & (SUM_ZEILE_VOLLST_s1!=0)
+ & (SUM_ANZAHL_EINTRAEGE_s1==SUM_ZEILE_VOLLST_s1)] <- 1
> PROB_VOLLST_s1[(GEARB_s1==1)
+ & (SUM_ANZAHL_EINTRAEGE_s1!=SUM_ZEILE_VOLLST_s1)] <- 0
> PROB_VOLLST_s1[(GEARB_s1==1) & (SUM_ANZAHL_EINTRAEGE_s1==0)
+ & (SUM_ZEILE_VOLLST_s1==0)] <- 1
> PROB_VOLLST_s1[is.na(GEARB_s1)] <- 1
```

Schritt 2: Bildung der Variable PROB_VOLLST_s2 (pro Proband) in SOLAR II

Kodierungsbeschreibung:
Zunächst wurde eine Hilfsvariable PROB_VOLLST_s2 für SOLAR II gebildet, die angab, ob der Proband in SOLAR II alle gemachten Tätigkeitsangaben vollständig ausgefüllt hatte. Ein Proband galt als vollständig, wenn er in SOLAR II nicht gearbeitet und auch keine Tätigkeitsangaben gemacht hatte. Wurde in SOLAR II gearbeitet, so musste die Anzahl der ausgefüllten Zeilen und die Anzahl der vollständig ausgefüllten Zeilen bei diesen Probanden übereinstimmen (SUM_ANZAHL_EINTRAEGE = SUM_ZEILE_VOLLST). War das der Fall, galten sie als vollständig. Weiterhin galten die Probanden als vollständig, wenn zwar gearbeitet wurde, aber keine Einträge vorlagen oder wenn die Angabe, ob gearbeitet wurde komplett fehlte. In diesen Fällen wurde konservativ vorgegangen und somit keine Tätigkeiten und mögliche Expositionen imputiert. Als unvollständig galt ein Proband in SOLAR II, wenn die gemachten Tätigkeitsangaben fehlende Werte aufwiesen.

Code-Übersicht:

Variablenname	Frage & Kodierung
PROB_VOLLST_s2	Proband vollständig in SOLAR II
	0=Nein, 1=Ja

R-Code:

```
> PROB_VOLLST_s2 <- NA
> PROB_VOLLST_s2[GEARB_s2 == 0] <- 1
> PROB_VOLLST_s2[(GEARB_s2==1) & (SUM_ANZAHL_EINTRAEGE_s2!=0)
+ & (SUM_ZEILE_VOLLST_s2!=0)
+ & (SUM_ANZAHL_EINTRAEGE_s2==SUM_ZEILE_VOLLST_s2)] <- 1
> PROB_VOLLST_s2[(GEARB_s2==1)
+ & (SUM_ANZAHL_EINTRAEGE_s2!=SUM_ZEILE_VOLLST_s2)] <- 0
> PROB_VOLLST_s2[(GEARB_s2==1) & (SUM_ANZAHL_EINTRAEGE_s2==0)
+ & (SUM_ZEILE_VOLLST_s2==0)] <- 1
> PROB_VOLLST_s2[is.na(GEARB_s2)] <- 1
```

Schritt 3: Bildung der Variable PROB_VOLLST (pro Proband) für SOLAR und SOLAR II

Kodierungsbeschreibung:
Aus den Variablen PROB_VOLLST_s1 und PROB_VOLLST_s2 wurde eine Variable PROB_VOLLST gebildet, die angab, ob der Proband sowohl in SOLAR als auch in SOLAR II vollständig war.

Code-Übersicht:

Variablenname	Frage & Kodierung
PROB_VOLLST	Proband in SOLAR und SOLAR II vollständig 0=Nein, 1=Ja

R-Code:

```
> PROB_VOLLST <- NA
> PROB_VOLLST[PROB_VOLLST_s1 == 1 & PROB_VOLLST_s2 == 1] <- 1
> PROB_VOLLST[PROB_VOLLST_s1 != 1 | PROB_VOLLST_s2 != 1] <- 0
```

A.4.10 Zeilen in denen imputiert werden musste

In mehreren Schritten, die im Folgenden ausführlich dargestellt werden, wurde aus den Tätigkeitsangaben aus SOLAR und SOLAR II die Variable SUM_ZEILE_IMPUTE gebildet, die angab, wie viele Zeilen pro Proband in einem späteren Schritt noch imputiert werden mussten.

Gebildete Variable:

Variablenname	**Frage & Kodierung**
SUM_ZEILE_IMPUTE	Summe der zu imputierenden Zeilen 0-10=Summe

Schritt 1: Bildung der Variable ZEILE_IMPUTE (pro Tätigkeit) in SOLAR

Kodierungsbeschreibung:
Zunächst wurd eine Hilfsvariable ZEILE_IMPUTE für SOLAR gebildet, die angab, ob die Zeile später imputiert werden musste (=1) oder nicht (=0).

Code-Übersicht:

Variablenname	Frage & Kodierung
ZEILE_IMPUTE	Zeile muss imputiert werden 0=Nein, 1=Ja

R-Code:

```
> ZEILE_IMPUTE <- NA
> ZEILE_IMPUTE[PROB_VOLLST_s1 == 1] <- 0
> ZEILE_IMPUTE[(ZEILE_VOLLST==1)] <- 0
> ZEILE_IMPUTE[(ANZAHL_EINTRAEGE==1) & (ZEILE_VOLLST==0)] <- 1
> ZEILE_IMPUTE[is.na(ZEILE_IMPUTE) & (ANZAHL_EINTRAEGE==0)] <- 0
```

Schritt 2: Bildung der Variable SUM_ZEILE_IMPUTE_s1 (pro Proband) für SOLAR

Kodierungsbeschreibung:
Aus der Hilfsvariable ZEILE_IMPUTE wurde dann für SOLAR eine Variable SUM_ZEILE_IMPUTE_s1 gebildet, die pro Probanden alle fünf möglichen Zeilen aufsummierte und die Summe der zu imputierenden Zeilen enthielt.

Code-Übersicht:

Variablenname	Frage & Kodierung
SUM_ZEILE_IMPUTE_s1	Summe der zu imputierenden Zeilen in SOLAR 0-5=Summe

R-Code:

```
> j <- 1
> i <- 1
> while (j <= nrow()) {
+ SUMME <- sum(ZEILE_IMPUTE[j],ZEILE_IMPUTE[j+1],ZEILE_IMPUTE[j+2],
+ ZEILE_IMPUTE[j+3],ZEILE_IMPUTE[j+4])
+ SUM_ZEILE_IMPUTE_s1[i] <- SUMME
+ SUM_ZEILE_IMPUTE_s1[i+1] <- SUMME
+ SUM_ZEILE_IMPUTE_s1[i+2] <- SUMME
+ SUM_ZEILE_IMPUTE_s1[i+3] <- SUMME
+ SUM_ZEILE_IMPUTE_s1[i+4] <- SUMME
+ j <- j+5
+ i <- i+5
+ }
```

Schritt 3: Bildung der Variable ZEILE_IMPUTE (pro Tätigkeit) in SOLAR II

Kodierungsbeschreibung:
Zunächst wurde eine Hilfsvariable ZEILE_IMPUTE für SOLAR II gebildet, die angab, ob die Zeile später imputiert werden musste (=1) oder nicht (=0).

Code-Übersicht:

Variablenname	Frage & Kodierung
ZEILE_IMPUTE	Zeile muss imputiert werden 0=Nein, 1=Ja

R-Code:

```
> ZEILE_IMPUTE <- NA
> ZEILE_IMPUTE[PROB_VOLLST_s2 == 1] <- 0
> ZEILE_IMPUTE[is.na(GEARB_s2)] <- 1
> ZEILE_IMPUTE[(GEARB_s2==1) & (SUM_ANZAHL_EINTRAEGE_s2==0)
+ & (SUM_ZEILE_VOLLST_s2==0)] <- 1
> ZEILE_IMPUTE[(ZEILE_VOLLST==1)] <- 0
> ZEILE_IMPUTE[(ANZAHL_EINTRAEGE==1) & (ZEILE_VOLLST==0)] <- 1
> ZEILE_IMPUTE[is.na(ZEILE_IMPUTE) & (ANZAHL_EINTRAEGE==0)] <- 0
```

Schritt 4: Bildung der Variable SUM_ZEILE_IMPUTE_s2 (pro Proband) für SOLAR II

Kodierungsbeschreibung:
Aus der Hilfsvariable ZEILE_IMPUTE wurde dann für SOLAR II eine Variable SUM_ZEILE_IMPUTE_s2 gebildet, die pro Probanden alle fünf möglichen Zeilen aufsummierte und die Summe der zu imputierenden Zeilen enthielt.

Code-Übersicht:

Variablenname	Frage & Kodierung
SUM_ZEILE_IMPUTE_s2	Summe der zu imputierenden Zeilen in SOLAR II 0-5=Summe

R-Code:

```
> j <- 1
> i <- 1
> while (j <= nrow()) {
+ SUMME <- sum(ZEILE_IMPUTE[j],ZEILE_IMPUTE[j+1],ZEILE_IMPUTE[j+2],
+ ZEILE_IMPUTE[j+3],ZEILE_IMPUTE[j+4])
+ SUM_ZEILE_IMPUTE_s2[i] <- SUMME
+ SUM_ZEILE_IMPUTE_s2[i+1] <- SUMME
+ SUM_ZEILE_IMPUTE_s2[i+2] <- SUMME
+ SUM_ZEILE_IMPUTE_s2[i+3] <- SUMME
+ SUM_ZEILE_IMPUTE_s2[i+4] <- SUMME
+ j <- j+5
+ i <- i+5
+ }
```

Schritt 5: Bildung der Variable SUM_ZEILE_IMPUTE (pro Proband) für SOLAR und SOLAR II

Kodierungsbeschreibung:
Aus den Variablen SUM_ZEILE_IMPUTE_s1 und SUM_ZEILE_IMPUTE_s2 wurde dann eine gemeinsame Variable SUM_ZEILE_IMPUTE gebildet, die pro Probanden alle zehn möglichen Zeilen aus SOLAR und SOLAR II aufsummierte und die Summe der zu imputierenden Zeilen enthielt.

Code-Übersicht:

Variablenname	**Frage & Kodierung**
SUM_ZEILE_IMPUTE	Summe der zu imputierenden Zeilen 0-10=Summe

R-Code:

```
> SUM_ZEILE_IMPUTE <- SUM_ZEILE_IMPUTE_s1 + SUM_ZEILE_IMPUTE_s2
```

A.5 Benötigte Variablen für die Berechnung der Exposition

A.5.1 Kurzbeschreibung der in der Basis-Matrix enthaltenen Variablen

Folgende Variablen waren in der Basis-Matrix enthalten:

Variablenname	**Variablenbeschreibung**
knr	Kohortennummer des Probanden
JAHR	Pro Proband und Tätigkeit die Jahre 2000-2009 (bzw. bei den Probanden, die schon vor 2000 gearbeitet hatten, die Jahre 1992-2009)
NR_BERUF	Gab an, in welcher Reihenfolge die Tätigkeiten ausgeübt wurden; Tätigkeit mit NR_BERUF=1 war 1. Tätigkeit des jeweiligen Probanden
ANF_MONAT	Anfangsmonat der Tätigkeit im entsprechenden Jahr
END_MONATx	Endmonat der Tätigkeit im entsprechenden Jahr
WST	Anzahl der Wochenstunden, die in der entsprechenden Tätigkeit gearbeitet wurden
ISCO	ISCO-Code des Tätigkeit
HMW*	Gab an, ob Exposition in der Kategorie HMW bestand (0=nein, 1=ja)
GEARB_MON	Gab an, wie viele Monate im entsprechenden Jahr gearbeitet wurden
MIND_8WST	Gab an, ob mindestens acht Stunden pro Woche gearbeitet wurde (0=nein: WST < 8, 1=ja: WST ≥ 8)
HMW_jahr*	Exposition in der Kategorie HMW in Stunden pro Jahr

*(Analog für LMW/MIXED/IRRPEAKS/LOWRISK)

A.5.2 Kurzbeschreibung der aus der Basis-Matrix gebildeten Variablen

Auf Grundlage der Basis-Matrix konnten folgende Variablen berechnet werden, die als mögliche Kovariablen in die Regressionsmodelle eingehen konnten:

Variablenname	**Variablenbeschreibung**
HMW_kumuliert*	kumulierte Exposition pro Proband in der Kategorie HMW über alle Tätigkeiten und Jahre hinweg (in Stunden)
HMW_binaer*	binäre Exposition pro Proband in der Kategorie HMW über alle Tätigkeiten und Jahre hinweg (0=nein, 1=ja)
HMW_erstesjahr_gesamt*	kumulierte Exposition pro Proband in der Kategorie HMW innerhalb des 1. Tätigkeitsjahres (in Stunden)
HMW_erstesjahr_binaer*	binäre Exposition pro Proband in der Kategorie HMW innerhalb des 1. Tätigkeitsjahres (0=nein, 1=ja)
HMW_ersterberuf_gesamt*	kumulierte Exposition pro Proband in der Kategorie HMW während der 1. Tätigkeit (in Stunden)
HMW_ersterberuf_binaer*	binäre Exposition pro Proband in der Kategorie HMW während der 1. Tätigkeit (0=nein, 1=ja)

*(Analog für LMW/MIXED/IRRPEAKS/LOWRISK)

B Imputation der Confoundervariablen

B.1 Imputation von drei Datensätzen durch multiple Imputation unter Anwendung des R-Packages Amelia-II

```
##############################################################################
##############################################################################
#   Imputation mit Amelia II - Erstellung von 3 vervollständigten Datensätzen  #
##############################################################################
##############################################################################

library(Amelia)

load("daten_fragebogen.RData")

# noms = ... Nominale Variablen angeben;
# idvars=knr (wird nicht bei Imputation verwendet, bleibt aber im Datensatz)

set.seed(123)# für Reproduzierbarkeit
amelia_all <- amelia(daten_fragebogen, m = 5, idvars=c("knr"), noms=c("zentrum","f02x",
"d_geb","PAR_ALL_r","siblings","STILL_r","ETSNOW_r","SES_r","f58x","RAUCHEN","BERUF",
"s2f78","s2SCHULE","s2RAUCHEN","s2BERUF","CUR_DERM_r","CUR_HAY_r","CUR_ASTH_r",
"CURDERMV","CURHAYV","CURASTHV","s2CURHAYV","s2CURASTHV"),empri=5,9)

#################################
#einzelne Datensätze abspeichern
#################################

amelia_imputed1 <- amelia_all$m1
save(amelia_imputed1, file="amelia_imputed1.RData")

amelia_imputed2 <- amelia_all$m2
save(amelia_imputed2, file="amelia_imputed2.RData")

amelia_imputed3 <- amelia_all$m3
save(amelia_imputed3, file="amelia_imputed3.RData")
```

B.2 Imputation von zwei Datensätzen durch Ziehen aus der empirischen Verteilung

Beispielhaft ist hier die Imputation eines Datensatzes dargestellt. Die Imputation des zweiten Datensatzes erfolgte analog. Details sind der CD-Rom zu entnehmen.

```
##############################################################################
##############################################################################
# Imputation:Ziehen aus emp.Verteilung - Erstellung des 1.vervollst.Datensatzes#
##############################################################################
##############################################################################

###### 2 zufällige Startwerte auswählen #############
set.seed(1)
startwerte <- runif(2, min=1, max=1000)
startwert1 <- round(startwerte[1])
startwert2 <- round(startwerte[2])

##############################################################################
################## Imputation des  1. Datensatzes ############################
##############################################################################

load("daten_fragebogen.RData")

###################
# d_geb imputieren
###################

n0 <- table(daten_fragebogen$d_geb, useNA="always")[[1]]
n1 <- table(daten_fragebogen$d_geb, useNA="always")[[2]]
n_miss <- table(daten_fragebogen$d_geb, useNA="always")[[3]]
prob1 <- n1/(n0+n1)

set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- rbinom(n=n_miss, size=1, prob=prob1)
```

```
j <- 1
for (i in 1:nrow(daten_fragebogen)){
  if(is.na(daten_fragebogen$d_geb)[i]){
    daten_fragebogen$d_geb[i] <- x[j]
    j <- j+1
  }
}

########################
# PAR_ALL_r imputieren
########################

n0 <- table(daten_fragebogen$PAR_ALL_r, useNA="always")[[1]]
n1 <- table(daten_fragebogen$PAR_ALL_r, useNA="always")[[2]]
n_miss <- table(daten_fragebogen$PAR_ALL_r, useNA="always")[[3]]
prob1 <- n1/(n0+n1)

set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- rbinom(n=n_miss, size=1, prob=prob1)

j <- 1
for (i in 1:nrow(daten_fragebogen)){
  if(is.na(daten_fragebogen$PAR_ALL_r)[i]){
    daten_fragebogen$PAR_ALL_r[i] <- x[j]
    j <- j+1
  }
}

########################
# siblings imputieren
########################

n0 <- table(daten_fragebogen$siblings, useNA="always")[[1]]
n1 <- table(daten_fragebogen$siblings, useNA="always")[[2]]
n2 <- table(daten_fragebogen$siblings, useNA="always")[[3]]
n3 <- table(daten_fragebogen$siblings, useNA="always")[[4]]
n4 <- table(daten_fragebogen$siblings, useNA="always")[[5]]
n5 <- table(daten_fragebogen$siblings, useNA="always")[[6]]
n6 <- table(daten_fragebogen$siblings, useNA="always")[[7]]
n7 <- table(daten_fragebogen$siblings, useNA="always")[[8]]
n_miss <- table(daten_fragebogen$siblings, useNA="always")[[9]]
prob0 <- n0/(n0+n1+n2+n3+n4+n5+n6+n7)
prob1 <- n1/(n0+n1+n2+n3+n4+n5+n6+n7)
prob2 <- n2/(n0+n1+n2+n3+n4+n5+n6+n7)
prob3 <- n3/(n0+n1+n2+n3+n4+n5+n6+n7)
prob4 <- n4/(n0+n1+n2+n3+n4+n5+n6+n7)
prob5 <- n5/(n0+n1+n2+n3+n4+n5+n6+n7)
prob6 <- n6/(n0+n1+n2+n3+n4+n5+n6+n7)
prob7 <- n7/(n0+n1+n2+n3+n4+n5+n6+n7)

set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- sample(0:7, size=n_miss, replace=TRUE, prob=c(prob0,prob1,prob2,prob3,prob4,prob5,prob6,prob7))

j <- 1
for (i in 1:nrow(daten_fragebogen)){
  if(is.na(daten_fragebogen$siblings)[i]){
    daten_fragebogen$siblings[i] <- x[j]
    j <- j+1
  }
}

####################
# STILL imputieren
####################

n0 <- table(daten_fragebogen$STILL_r, useNA="always")[[1]]
n1 <- table(daten_fragebogen$STILL_r, useNA="always")[[2]]
n_miss <- table(daten_fragebogen$STILL_r, useNA="always")[[3]]
prob1 <- n1/(n0+n1)

set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- rbinom(n=n_miss, size=1, prob=prob1)

j <- 1
for (i in 1:nrow(daten_fragebogen)){
  if(is.na(daten_fragebogen$STILL_r)[i]){
    daten_fragebogen$STILL_r[i] <- x[j]
    j <- j+1
  }
}

####################
# ETSNOW imputieren
####################

n0 <- table(daten_fragebogen$ETSNOW_r, useNA="always")[[1]]
n1 <- table(daten_fragebogen$ETSNOW_r, useNA="always")[[2]]
n2 <- table(daten_fragebogen$ETSNOW_r, useNA="always")[[3]]
```

```
n_miss <- table(daten_fragebogen$ETSNOW, useNA="always")[[4]] # Achtung: hier muss [[4]] stehen!
prob0 <- n0/(n0+n1+n2)
prob1 <- n1/(n0+n1+n2)
prob2 <- n2/(n0+n1+n2)

set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- sample(0:2, size=n_miss, replace=TRUE, prob=c(prob0,prob1,prob2))

j <- 1
for (i in 1:nrow(daten_fragebogen)){
  if(is.na(daten_fragebogen$ETSNOW_r)[i]){
    daten_fragebogen$ETSNOW_r[i] <- x[j]
    j <- j+1
  }
}

####################
# f58x imputieren
####################

n0 <- table(daten_fragebogen$f58x, useNA="always")[[1]]
n1 <- table(daten_fragebogen$f58x, useNA="always")[[2]]
n_miss <- table(daten_fragebogen$f58x, useNA="always")[[3]]
prob1 <- n1/(n0+n1)

set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- rbinom(n=n_miss, size=1, prob=prob1)

j <- 1
for (i in 1:nrow(daten_fragebogen)){
  if(is.na(daten_fragebogen$f58x)[i]){
    daten_fragebogen$f58x[i] <- x[j]
    j <- j+1
  }
}

####################
# RAUCHEN imputieren
####################

n0 <- table(daten_fragebogen$RAUCHEN, useNA="always")[[1]]
n1 <- table(daten_fragebogen$RAUCHEN, useNA="always")[[2]]
n_miss <- table(daten_fragebogen$RAUCHEN, useNA="always")[[3]]
prob1 <- n1/(n0+n1)

set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- rbinom(n=n_miss, size=1, prob=prob1)

j <- 1
for (i in 1:nrow(daten_fragebogen)){
  if(is.na(daten_fragebogen$RAUCHEN)[i]){
    daten_fragebogen$RAUCHEN[i] <- x[j]
    j <- j+1
  }
}

####################
# s2f78 imputieren
####################

n0 <- table(daten_fragebogen$s2f78, useNA="always")[[1]]
n1 <- table(daten_fragebogen$s2f78, useNA="always")[[2]]
n_miss <- table(daten_fragebogen$s2f78, useNA="always")[[3]]
prob1 <- n1/(n0+n1)

set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- rbinom(n=n_miss, size=1, prob=prob1)

j <- 1
for (i in 1:nrow(daten_fragebogen)){
  if(is.na(daten_fragebogen$s2f78)[i]){
    daten_fragebogen$s2f78[i] <- x[j]
    j <- j+1
  }
}

#####################
# s2RAUCHEN imputieren
#####################

n0 <- table(daten_fragebogen$s2RAUCHEN, useNA="always")[[1]]
n1 <- table(daten_fragebogen$s2RAUCHEN, useNA="always")[[2]]
n2 <- table(daten_fragebogen$s2RAUCHEN, useNA="always")[[3]]
n_miss <- table(daten_fragebogen$s2RAUCHEN, useNA="always")[[4]]  # Achtung: hier muss [[4]] stehen!
prob0 <- n0/(n0+n1+n2)
prob1 <- n1/(n0+n1+n2)
prob2 <- n2/(n0+n1+n2)
```

```
set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- sample(0:2, size=n_miss, replace=TRUE, prob=c(prob0,prob1,prob2))

j <- 1
for (i in 1:nrow(daten_fragebogen)){
  if(is.na(daten_fragebogen$s2RAUCHEN)[i]){
    daten_fragebogen$s2RAUCHEN[i] <- x[j]
    j <- j+1
  }
}

######################
# s2SCHULE imputieren
######################

n0 <- table(daten_fragebogen$s2SCHULE, useNA="always")[[1]]
n1 <- table(daten_fragebogen$s2SCHULE, useNA="always")[[2]]
n_miss <- table(daten_fragebogen$s2SCHULE, useNA="always")[[3]]
prob1 <- n1/(n0+n1)

set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- rbinom(n=n_miss, size=1, prob=prob1)

j <- 1
for (i in 1:nrow(daten_fragebogen)){
  if(is.na(daten_fragebogen$s2SCHULE)[i]){
    daten_fragebogen$s2SCHULE[i] <- x[j]
    j <- j+1
  }
}

############################
### Datensatz zwischenspeichern
############################

empVert_imputed1 <- daten_fragebogen
save(empVert_imputed1, file="empVert_imputed1.RData")
load("empVert_imputed1.RData")

##################
# SES_r imputieren
##################

n0 <- table(empVert_imputed1$SES_r, useNA="always")[[1]]
n1 <- table(empVert_imputed1$SES_r, useNA="always")[[2]]
n_miss <- table(empVert_imputed1$SES_r, useNA="always")[[3]]
prob1 <- n1/(n0+n1)

set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- rbinom(n=n_miss, size=1, prob=prob1)

j <- 1
for (i in 1:nrow(empVert_imputed1)){
  if(is.na(empVert_imputed1$SES_r)[i]){
    empVert_imputed1$SES_r[i] <- x[j]
    j <- j+1
  }
}

########################
# BERUF imputieren
########################

n1 <- table(empVert_imputed1$BERUF, useNA="always")[[1]]
n2 <- table(empVert_imputed1$BERUF, useNA="always")[[2]]
n3 <- table(empVert_imputed1$BERUF, useNA="always")[[3]]
n4 <- table(empVert_imputed1$BERUF, useNA="always")[[4]]
n5 <- table(empVert_imputed1$BERUF, useNA="always")[[5]]
n6 <- table(empVert_imputed1$BERUF, useNA="always")[[6]]
n7 <- table(empVert_imputed1$BERUF, useNA="always")[[7]]
n9 <- table(empVert_imputed1$BERUF, useNA="always")[[8]]
n12 <- table(empVert_imputed1$BERUF, useNA="always")[[9]]
n_miss <- table(empVert_imputed1$BERUF, useNA="always")[[10]]
prob1 <- n1/(n1+n2+n3+n4+n5+n6+n7+n9+n12)
prob2 <- n2/(n1+n2+n3+n4+n5+n6+n7+n9+n12)
prob3 <- n3/(n1+n2+n3+n4+n5+n6+n7+n9+n12)
prob4 <- n4/(n1+n2+n3+n4+n5+n6+n7+n9+n12)
prob5 <- n5/(n1+n2+n3+n4+n5+n6+n7+n9+n12)
prob6 <- n6/(n1+n2+n3+n4+n5+n6+n7+n9+n12)
prob7 <- n7/(n1+n2+n3+n4+n5+n6+n7+n9+n12)
prob9 <- n9/(n1+n2+n3+n4+n5+n6+n7+n9+n12)
prob12 <- n12/(n1+n2+n3+n4+n5+n6+n7+n9+n12)

set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- sample(c(1,2,3,4,5,6,7,9,12), size=n_miss, replace=TRUE, prob=c(prob1,prob2,prob3,prob4,prob5,prob6,prob7,prob9,prob12))

j <- 1
for (i in 1:nrow(empVert_imputed1)){
```

```
  if(is.na(empVert_imputed1$BERUF)[i]){
    empVert_imputed1$BERUF[i] <- x[j]
    j <- j+1
  }
}

########################
# s2BERUF imputieren
########################

n1 <- table(empVert_imputed1$s2BERUF, useNA="always")[[1]]
n2 <- table(empVert_imputed1$s2BERUF, useNA="always")[[2]]
n3 <- table(empVert_imputed1$s2BERUF, useNA="always")[[3]]
n4 <- table(empVert_imputed1$s2BERUF, useNA="always")[[4]]
n5 <- table(empVert_imputed1$s2BERUF, useNA="always")[[5]]
n6 <- table(empVert_imputed1$s2BERUF, useNA="always")[[6]]
n8 <- table(empVert_imputed1$s2BERUF, useNA="always")[[7]]
n9 <- table(empVert_imputed1$s2BERUF, useNA="always")[[8]]
n_miss <- table(empVert_imputed1$s2BERUF, useNA="always")[[9]]
prob1 <- n1/(n1+n2+n3+n4+n5+n6+n8+n9)
prob2 <- n2/(n1+n2+n3+n4+n5+n6+n8+n9)
prob3 <- n3/(n1+n2+n3+n4+n5+n6+n8+n9)
prob4 <- n4/(n1+n2+n3+n4+n5+n6+n8+n9)
prob5 <- n5/(n1+n2+n3+n4+n5+n6+n8+n9)
prob6 <- n6/(n1+n2+n3+n4+n5+n6+n8+n9)
prob8 <- n8/(n1+n2+n3+n4+n5+n6+n8+n9)
prob9 <- n9/(n1+n2+n3+n4+n5+n6+n8+n9)

set.seed(startwert1)  # muss vor jedem Befehl laufen, der Zufallszahlen zieht
x <- sample(c(1,2,3,4,5,6,8,9), size=n_miss, replace=TRUE, prob=c(prob1,prob2,prob3,prob4,prob5,prob6,prob8,prob9))

j <- 1
for (i in 1:nrow(empVert_imputed1)){
  if(is.na(empVert_imputed1$s2BERUF)[i]){
    empVert_imputed1$s2BERUF[i] <- x[j]
    j <- j+1
  }
}

############################
### Datensatz abspeichern
############################

save(empVert_imputed1, file="empVert_imputed1.RData")
load("empVert_imputed1.RData")
```

C Expositionsberechnung

Die Expositionsberechnung wurde auf diversen Datensätzen nach dem gleichen Prinzip durchgeführt. Beispielhaft soll hier die Berechnung der Exposition auf Basis der Probanden mit vollständigen Tätigkeitsangaben abgedruckt werden. Die Expositionsberechnungen auf Basis der anderen Datensätze kann der beiliegenden CD-Rom entnommen werden.

```
##############################################################################
################### Datensatz laden ##########################################
##############################################################################

# Für die Job-Matrix werden nur die Probanden mit vollständigen Tätigkeitsangaben verwendet
load("berufsdaten_vollstaendig.RData")
nrow(berufsdaten_vollstaendig)
# enthält 10940 Zeilen also 1094 Probanden (weil pro Proband 10 Zeilen für maximal 10 Jobs)

basis <- berufsdaten_vollstaendig

##############################################################################
################### Basisdatensatz erstellen #################################
##############################################################################

# JEM auf 5 Kategorien (HMW, LMW, MIXED, IRRPEAKS, LOWRISK) reduzieren
# Dabei steht jeweils in der Obergruppe (z.B. HMW) eine 1 (für exponiert), wenn
# in mindestens einer Unterkategorie (z.B. anim, fish, flour ... für HMW) eine 1
# (für exponiert) steht
basis$HMW <- 0
basis$LMW <- 0
basis$MIXED <- 0
basis$IRRPEAKS <- 0
basis$LOWRISK <- 0

# anim, fish, flour, plants, mites, enzymes, latex, bioaero, drugs zur
# Obergruppe HMW zusammenfassen
basis$HMW[(basis$anim==1) |
(basis$fish==1) | (basis$flour==1) |
(basis$plants==1) | (basis$mites==1) |
(basis$enzymes==1) | (basis$latex==1) |
(basis$bioaero==1) | (basis$drugs==1)] <-1

# react, isocy, clean, wood, metals zur Obergruppe LMW zusammenfassen
basis$LMW[(basis$react==1) |
(basis$isocy==1) | (basis$clean==1) |
(basis$wood==1) | (basis$metals==1)] <-1

# mwf, textile, agric zur Obergruppe MIXED zusammenfassen
basis$MIXED[(basis$mwf==1) |
(basis$textile==1) | (basis$agric==1)] <-1

# IRRPEAKS (Obergruppe) kann direkt von irrpeaks(Untergruppe) übernommen werden,
# es gibt hier keine anderen Untergruppen
basis$IRRPEAKS[(basis$irrpeaks==1)] <-1

# exhaust, ets, pos_irr zur Obergruppe LOWRISK zusammenfassen
basis$LOWRISK[(basis$exhaust==1) |
(basis$ets==1) | (basis$pos_irr==1) |
(basis$low_anti==1)] <-1

# basis umbennen in basis_5kat und
basis_5kat <- basis

# Sortierung entsprechend vornehmen, so dass 1. Tätigkeit in der ersten Zeile steht etc.
# Sortieren nach knr, dann nach ANF_JAHR und wenn Jahr gleich ist nach ANF_MONAT
# => die erste Tätigkeit steht immer in der ersten Zeile
# wurde bereits in daten_beruf_alle_sort vorgenommen!!!
basis_5kat_sort <- basis_5kat

# Variable NR_BERUF anlegen, gibt an in welcher Reihenfolge die Tätigkeiten ausgeübt
# wurden
basis_5kat_sort$NR_BERUF <- rep(1:10, times=nrow(basis)/10)

# Nur die Spalten die auch wirklich benötigt werden als Subset rausziehen
expoberechnung_basis <- subset(basis_5kat_sort, select=c("knr",
"NR_BERUF","ANF_MONAT","ANF_JAHR","END_MONATx","END_JAHRx","WST","ISCO","HMW",
"LMW","MIXED","IRRPEAKS","LOWRISK","DAUER","JEMALS_GEARB"))

# Für spätere Berechnungen: DAUER, WST wenn == NA auf 0 setzten
# WST == NA: das sind die Fälle mit ISCO 94, 95, 97, 98 (da dürfen die WST fehlen)
# DAUER == NA: das sind die Fälle mit WST < 8 (da dürfen die WST fehlen)
# oder die Zeilen ohne Tätigkeitsangaben (also mit ISCO 8888 bzw. 9999)
expoberechnung_basis$DAUER[is.na(expoberechnung_basis$DAUER)] <- 0
expoberechnung_basis$WST[is.na(expoberechnung_basis$WST)] <- 0
```

```
# Variable MIND_8_WST nochmal neu anlegen damit überall 0 oder 1 drin steht
expoberechnung_basis$MIND_8_WST <- 0
expoberechnung_basis$MIND_8_WST[expoberechnung_basis$WST>=8] <- 1

# Jetzt die Exposition pro Tätigkeit (d.h. für jede Zeile) berechnen

# Erst mal alle Variablen mit 0 initialisieren
expoberechnung_basis$HMW_beruf <- 0
expoberechnung_basis$LMW_beruf <- 0
expoberechnung_basis$MIXED_beruf <- 0
expoberechnung_basis$IRRPEAKS_beruf <- 0
expoberechnung_basis$LOWRISK_beruf <- 0

# Expositionen nur berechnen wenn MIND_8_WST == 1 ist
# für ISCO 94, 95, 97, 98 ist sowieso überall Expo == 0 (passt also)
for (i in 1:nrow(expoberechnung_basis)){
if (expoberechnung_basis$MIND_8_WST[i] == 1){
expoberechnung_basis$HMW_beruf[i] <- (4.25 * expoberechnung_basis$WST[i] *
expoberechnung_basis$HMW[i] * expoberechnung_basis$DAUER[i])
expoberechnung_basis$LMW_beruf[i] <- (4.25 * expoberechnung_basis$WST[i] *
expoberechnung_basis$LMW[i] * expoberechnung_basis$DAUER[i])
expoberechnung_basis$MIXED_beruf[i] <- (4.25 * expoberechnung_basis$WST[i] *
expoberechnung_basis$MIXED[i] * expoberechnung_basis$DAUER[i])
expoberechnung_basis$IRRPEAKS_beruf[i] <- (4.25 * expoberechnung_basis$WST[i] *
expoberechnung_basis$IRRPEAKS[i] * expoberechnung_basis$DAUER[i])
expoberechnung_basis$LOWRISK_beruf[i] <- (4.25 * expoberechnung_basis$WST[i] *
expoberechnung_basis$LOWRISK[i] * expoberechnung_basis$DAUER[i])
}
}

# Datensatz abspeichern
save(expoberechnung_basis, file="expoberechnung_basis.RData")
load("expoberechnung_basis.RData")

################################################################################
##################### Exposition kumuliert #####################################
################################################################################

load("expoberechnung_basis.RData")

# Jetzt über die Jobs aufsummieren => pro Proband eine Zeile
expo_kumuliert <- data.frame(HMW_kumuliert=numeric(nrow(basis)/10),
LMW_kumuliert=numeric(nrow(basis)/10), MIXED_kumuliert=numeric(nrow(basis)/10),
IRRPEAKS_kumuliert=numeric(nrow(basis)/10), LOWRISK_kumuliert=numeric(nrow(basis)/10),
HMW_binaer=numeric(nrow(basis)/10), LMW_binaer=numeric(nrow(basis)/10), MIXED_binaer=numeric(nrow(basis)/10),
IRRPEAKS_binaer=numeric(nrow(basis)/10), LOWRISK_binaer=numeric(nrow(basis)/10))

# alle initialisieren mit NA
expo_kumuliert$HMW_kumuliert <- NA
expo_kumuliert$LMW_kumuliert <- NA
expo_kumuliert$MIXED_kumuliert <- NA
expo_kumuliert$IRRPEAKS_kumuliert <- NA
expo_kumuliert$LOWRISK_kumuliert <- NA
expo_kumuliert$HMW_binaer <- NA
expo_kumuliert$LMW_binaer <- NA
expo_kumuliert$MIXED_binaer <- NA
expo_kumuliert$IRRPEAKS_binaer <- NA
expo_kumuliert$LOWRISK_binaer <- NA

# KNR übertragen
i <- 1
j <- 1
while (i <= nrow(expoberechnung_basis)){
expo_kumuliert$knr[j] <- as.character(expoberechnung_basis$knr[i])
i <- i+10 # nächster Proband
j <- j+1  # nächster Proband in der neuen Matrix
}

# HMW_beruf pro Proband über die 10 Jobs aufsummieren
i <- 1
j <- 1
while (i <= nrow(expoberechnung_basis)){
expo_kumuliert$HMW_kumuliert[j] <- (
expoberechnung_basis$HMW_beruf[i] +
expoberechnung_basis$HMW_beruf[i+1] +
expoberechnung_basis$HMW_beruf[i+2] +
expoberechnung_basis$HMW_beruf[i+3] +
expoberechnung_basis$HMW_beruf[i+4] +
expoberechnung_basis$HMW_beruf[i+5] +
expoberechnung_basis$HMW_beruf[i+6] +
expoberechnung_basis$HMW_beruf[i+7] +
expoberechnung_basis$HMW_beruf[i+8] +
expoberechnung_basis$HMW_beruf[i+9])
i <- i+10  # nächster Proband in expoberechnung_basis
j <- j+1   # nächste Stelle in der neuen Matrix expo_kumuliert
}

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
```

```
expo_kumuliert$HMW_binaer[expo_kumuliert$HMW_kumuliert > 0] <- 1
expo_kumuliert$HMW_binaer[expo_kumuliert$HMW_kumuliert == 0] <- 0

# LMW_beruf pro Proband über die 10 Jobs aufsummieren
i <- 1
j <- 1
while (i <= nrow(expoberechnung_basis)){
expo_kumuliert$LMW_kumuliert[j] <- (
expoberechnung_basis$LMW_beruf[i] +
expoberechnung_basis$LMW_beruf[i+1] +
expoberechnung_basis$LMW_beruf[i+2] +
expoberechnung_basis$LMW_beruf[i+3] +
expoberechnung_basis$LMW_beruf[i+4] +
expoberechnung_basis$LMW_beruf[i+5] +
expoberechnung_basis$LMW_beruf[i+6] +
expoberechnung_basis$LMW_beruf[i+7] +
expoberechnung_basis$LMW_beruf[i+8] +
expoberechnung_basis$LMW_beruf[i+9])
i <- i+10  # nächster Proband in expoberechnung_basis
j <- j+1   # nächste Stelle in der neuen Matrix expo_kumuliert
}

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_kumuliert$LMW_binaer[expo_kumuliert$LMW_kumuliert > 0] <- 1
expo_kumuliert$LMW_binaer[expo_kumuliert$LMW_kumuliert == 0] <- 0

# MIXED_beruf pro Proband über die 10 Jobs aufsummieren
i <- 1
j <- 1
while (i <= nrow(expoberechnung_basis)){
expo_kumuliert$MIXED_kumuliert[j] <- (
expoberechnung_basis$MIXED_beruf[i] +
expoberechnung_basis$MIXED_beruf[i+1] +
expoberechnung_basis$MIXED_beruf[i+2] +
expoberechnung_basis$MIXED_beruf[i+3] +
expoberechnung_basis$MIXED_beruf[i+4] +
expoberechnung_basis$MIXED_beruf[i+5] +
expoberechnung_basis$MIXED_beruf[i+6] +
expoberechnung_basis$MIXED_beruf[i+7] +
expoberechnung_basis$MIXED_beruf[i+8] +
expoberechnung_basis$MIXED_beruf[i+9])
i <- i+10  # nächster Proband in expoberechnung_basis
j <- j+1   # nächste Stelle in der neuen Matrix expo_kumuliert
}

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_kumuliert$MIXED_binaer[expo_kumuliert$MIXED_kumuliert > 0] <- 1
expo_kumuliert$MIXED_binaer[expo_kumuliert$MIXED_kumuliert == 0] <- 0

# IRRPEAKS_beruf pro Proband über die 10 Jobs aufsummieren
i <- 1
j <- 1
while (i <= nrow(expoberechnung_basis)){
expo_kumuliert$IRRPEAKS_kumuliert[j] <- (
expoberechnung_basis$IRRPEAKS_beruf[i] +
expoberechnung_basis$IRRPEAKS_beruf[i+1] +
expoberechnung_basis$IRRPEAKS_beruf[i+2] +
expoberechnung_basis$IRRPEAKS_beruf[i+3] +
expoberechnung_basis$IRRPEAKS_beruf[i+4] +
expoberechnung_basis$IRRPEAKS_beruf[i+5] +
expoberechnung_basis$IRRPEAKS_beruf[i+6] +
expoberechnung_basis$IRRPEAKS_beruf[i+7] +
expoberechnung_basis$IRRPEAKS_beruf[i+8] +
expoberechnung_basis$IRRPEAKS_beruf[i+9])
i <- i+10  # nächster Proband in expoberechnung_basis
j <- j+1   # nächste Stelle in der neuen Matrix expo_kumuliert
}

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_kumuliert$IRRPEAKS_binaer[expo_kumuliert$IRRPEAKS_kumuliert > 0] <- 1
expo_kumuliert$IRRPEAKS_binaer[expo_kumuliert$IRRPEAKS_kumuliert == 0] <- 0

# LOWRISK_beruf pro Proband über die 10 Jobs aufsummieren
i <- 1
j <- 1
while (i <= nrow(expoberechnung_basis)){
expo_kumuliert$LOWRISK_kumuliert[j] <- (
expoberechnung_basis$LOWRISK_beruf[i] +
expoberechnung_basis$LOWRISK_beruf[i+1] +
expoberechnung_basis$LOWRISK_beruf[i+2] +
expoberechnung_basis$LOWRISK_beruf[i+3] +
expoberechnung_basis$LOWRISK_beruf[i+4] +
expoberechnung_basis$LOWRISK_beruf[i+5] +
expoberechnung_basis$LOWRISK_beruf[i+6] +
expoberechnung_basis$LOWRISK_beruf[i+7] +
expoberechnung_basis$LOWRISK_beruf[i+8] +
expoberechnung_basis$LOWRISK_beruf[i+9])
i <- i+10  # nächster Proband in expoberechnung_basis
```

```
j <- j+1   # nächste Stelle in der neuen Matrix expo_kumuliert
}

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_kumuliert$LOWRISK_binaer[expo_kumuliert$LOWRISK_kumuliert > 0] <- 1
expo_kumuliert$LOWRISK_binaer[expo_kumuliert$LOWRISK_kumuliert == 0] <- 0

# Datensatz abspeichern
save(expo_kumuliert, file="expo_kumuliert.RData")

################################################################################
###         Die Exposition in der ersten Tätigkeit berechnen                 ###
################################################################################

load("expoberechnung_basis.RData")
expo_ersterberuf_basis <- subset(expoberechnung_basis, NR_BERUF == 1)

# Datensatz abspeichern
#save(expo_ersterberuf_basis, file="expo_ersterberuf_basis.RData")

expo_ersterberuf_basis$HMW_ersterberuf_gesamt <- expo_ersterberuf_basis$HMW_beruf
expo_ersterberuf_basis$LMW_ersterberuf_gesamt <- expo_ersterberuf_basis$LMW_beruf
expo_ersterberuf_basis$MIXED_ersterberuf_gesamt <- expo_ersterberuf_basis$MIXED_beruf
expo_ersterberuf_basis$IRRPEAKS_ersterberuf_gesamt <- expo_ersterberuf_basis$IRRPEAKS_beruf
expo_ersterberuf_basis$LOWRISK_ersterberuf_gesamt <- expo_ersterberuf_basis$LOWRISK_beruf

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_ersterberuf_basis$HMW_ersterberuf_binaer[
expo_ersterberuf_basis$HMW_ersterberuf_gesamt > 0] <- 1
expo_ersterberuf_basis$HMW_ersterberuf_binaer[
expo_ersterberuf_basis$HMW_ersterberuf_gesamt == 0] <- 0

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_ersterberuf_basis$LMW_ersterberuf_binaer[
expo_ersterberuf_basis$LMW_ersterberuf_gesamt > 0] <- 1
expo_ersterberuf_basis$LMW_ersterberuf_binaer[
expo_ersterberuf_basis$LMW_ersterberuf_gesamt == 0] <- 0

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_ersterberuf_basis$MIXED_ersterberuf_binaer[
expo_ersterberuf_basis$MIXED_ersterberuf_gesamt > 0] <- 1
expo_ersterberuf_basis$MIXED_ersterberuf_binaer[
expo_ersterberuf_basis$MIXED_ersterberuf_gesamt == 0] <- 0

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_ersterberuf_basis$IRRPEAKS_ersterberuf_binaer[
expo_ersterberuf_basis$IRRPEAKS_ersterberuf_gesamt > 0] <- 1
expo_ersterberuf_basis$IRRPEAKS_ersterberuf_binaer[
expo_ersterberuf_basis$IRRPEAKS_ersterberuf_gesamt == 0] <- 0

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_ersterberuf_basis$LOWRISK_ersterberuf_binaer[
expo_ersterberuf_basis$LOWRISK_ersterberuf_gesamt > 0] <- 1
expo_ersterberuf_basis$LOWRISK_ersterberuf_binaer[
expo_ersterberuf_basis$LOWRISK_ersterberuf_gesamt == 0] <- 0

# nur die relevanten Variablen bleiben im Datensatz
expo_ersterberuf <- subset(expo_ersterberuf_basis, select=c("knr",
"HMW_ersterberuf_gesamt","LMW_ersterberuf_gesamt","MIXED_ersterberuf_gesamt",
"IRRPEAKS_ersterberuf_gesamt","LOWRISK_ersterberuf_gesamt",
"HMW_ersterberuf_binaer","LMW_ersterberuf_binaer","MIXED_ersterberuf_binaer",
"IRRPEAKS_ersterberuf_binaer","LOWRISK_ersterberuf_binaer"))

# Datensatz abspeichern
save(expo_ersterberuf, file="expo_ersterberuf.RData")

################################################################################
###             Die Exposition im ersten Tätigkeitssjahr berechnen           ###
################################################################################

load("expoberechnung_basis.RData")
expo_erstesjahr_basis <- expoberechnung_basis

# Variable ANF_BERUF_CHAR erstellen: in ihr sollen ANF_JAHR und ANF_MONAT in
# folgender Form zusammengefasst werden:"JJJJMM" also zb. "200103" für März 2001

# Variable ANF_BERUF_CHAR mit 0 initialisieren
expo_erstesjahr_basis$ANF_BERUF_CHAR <- 0

# ANF_BERUF_CHAR wird nur da eingetragen, wo ANF_MONAT und ANF_JAHR vorhanden sind
for (i in 1:nrow(expo_erstesjahr_basis)){
if (!is.na(expo_erstesjahr_basis$ANF_MONAT[i])
& (!is.na(expo_erstesjahr_basis$ANF_JAHR[i]))){
# Wenn der ANF_MONAT kleiner als 10 ist, dann muss das Format so aussehen:
# "JJJJOM" weil z.B. der Monat März im numerischen "3" ist und nicht "03"
if (expo_erstesjahr_basis$ANF_MONAT[i] < 10){
expo_erstesjahr_basis$ANF_BERUF_CHAR[i]<-paste(expo_erstesjahr_basis$ANF_JAHR[i]
,"0",expo_erstesjahr_basis$ANF_MONAT[i],sep="")
```

```
}
# Wenn der ANF_MONAT größer oder gleich 10 ist,dann muss das Format so aussehen:
# "JJJJMM", d.h. man kann Jahr und Monat einfach hintereinander zusammenfügen
if (expo_erstesjahr_basis$ANF_MONAT[i] >= 10){
expo_erstesjahr_basis$ANF_BERUF_CHAR[i]<-paste(expo_erstesjahr_basis$ANF_JAHR[i]
,expo_erstesjahr_basis$ANF_MONAT[i],sep="")
}
}
}

# Variable ENDE_12_MONATE erstellen: in ihr steht das Ende der ersten 12 Monate
# der Tätigkeit (also des ersten Tätigkeitsjahres), d.h. zum Beginn der ersten Tätigkeit
# eines Probanden werden 12 Monate addiert, dann hat man den Endzeitpunkt des
# ersten Tätigkeitsjahres (wieder im Format "JJJJMM" also zb. "200103" für März 2001
# (so wie bei ANF_BERUF_CHAR))

# Variable ENDE_12_MONATE mit 0 initialisieren
expo_erstesjahr_basis$ENDE_12_MONATE <- 0

# Wenn ANF_BERUF_CHAR ungleich 0 ist und NR_BERUF = 1, d.h. bei der ersten Tätigkeit
# jedes Probanden, wenn dort ein Beginn steht (1. Tätigkeit ist immer in der ersten
# Zeile da zuvor sortiert wurde), soll ein ENDE_12_MONATE berechnet werden
for (i in 1:nrow(expo_erstesjahr_basis)){
if (expo_erstesjahr_basis$ANF_BERUF_CHAR[i]!=0
& expo_erstesjahr_basis$ANF_BERUF_CHAR[i]!="000000"
& expo_erstesjahr_basis$NR_BERUF[i]==1){
# Wenn der ANF_MONAT 1 ist, dann ist das ENDE_12_MONATE im gleichen Jahr wie
# ANF_BERUF_CHAR, d.h. das Jahr kann so übernommen werden; beim Monat müssen 11
# Monate zum ANF_MONAT dazu addiert werden. Bsp: ANF_BERUF_CHAR ist 200001, dann
# ist ENDE_12_MONATE 200012 (genau ein Jahr = 12 Monate)
if (expo_erstesjahr_basis$ANF_MONAT[i]==1){
ende12monate <- paste(expo_erstesjahr_basis$ANF_JAHR[i],
expo_erstesjahr_basis$ANF_MONAT[i]+11,sep="")
expo_erstesjahr_basis$ENDE_12_MONATE[i] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+1] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+2] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+3] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+4] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+5] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+6] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+7] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+8] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+9] <- ende12monate
}
# Wenn der Anfang Monat ungleich 1 ist
if (expo_erstesjahr_basis$ANF_MONAT[i]!=1){
# Wenn der ANF_MONAT 11 oder 12 ist, dann wird zum Jahr in ANF_BERUF_CHAR noch
# ein Jahr addiert (Ende der 12 Monate liegt im nächsten Jahr); Vom ANF_MONAT
# muss ein Monat abgezogen werden. Bsp: ANF_BERUF_CHAR ist 200211, dann ist
# ENDE_12_MONATE 200310.
if (expo_erstesjahr_basis$ANF_MONAT[i] > 10){
ende12monate <- paste(expo_erstesjahr_basis$ANF_JAHR[i]+1,
expo_erstesjahr_basis$ANF_MONAT[i]-1,sep="")
expo_erstesjahr_basis$ENDE_12_MONATE[i] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+1] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+2] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+3] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+4] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+5] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+6] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+7] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+8] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+9] <- ende12monate

}
# Wenn der ANF_MONAT kleiner gleich 10 ist (2,3,...,10), dann wird zum Jahr in
# ANF_BERUF_CHAR noch ein Jahr addiert (Ende der 12 Monate liegt im nächsten
# Jahr); Vom ANF_MONAT muss ein Monat abgezogen werden, da dann ANF_MONAT-1 in
# Menge (1,2,...,9) liegt muss zwischen JAHR und ANF_MONAT-1 noch eine "0"
# eingefügt werden. Bsp: ANF_BERUF_CHAR ist 200210, dann ist ENDE_12_MONATE
# 200309.
if (expo_erstesjahr_basis$ANF_MONAT[i] <= 10){
ende12monate <- paste(expo_erstesjahr_basis$ANF_JAHR[i]+1,"0",
expo_erstesjahr_basis$ANF_MONAT[i]-1,sep="")
expo_erstesjahr_basis$ENDE_12_MONATE[i] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+1] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+2] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+3] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+4] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+5] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+6] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+7] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+8] <- ende12monate
expo_erstesjahr_basis$ENDE_12_MONATE[i+9] <- ende12monate
}
}
}
}
```

```
# Indikatorvariable BERUF_BEACHTEN erstellen, die angibt, ob die Tätigkeit in der
# enstprechenden Zeile für die Exposition im ersten Tätigkeitsjahr berücksichtigt
# werden muss oder nicht (0=nein /1=ja)
expo_erstesjahr_basis$BERUF_BEACHTEN <- 0

for(i in 1:nrow(expo_erstesjahr_basis)){
if(expo_erstesjahr_basis$ANF_BERUF_CHAR[i]!=0 &
(expo_erstesjahr_basis$ANF_BERUF_CHAR[i]
< expo_erstesjahr_basis$ENDE_12_MONATE[i])){
expo_erstesjahr_basis$BERUF_BEACHTEN[i] <- 1
}
}

# Variable ENDE_12_MONATE auftrennen in ENDE_12_MONATE_MONAT und
# ENDE_12_MONATE_JAHR. Bsp: Wenn ENDE_12_MONATE = "200103" ist dann steht jetzt
# in ENDE_12_MONATE_MONAT "3" und in ENDE_12_MONATE_JAHR "2001".

# Beide Variablen ENDE_12_MONATE_MONAT und ENDE_12_MONATE_JAHR mit NA initialis.
expo_erstesjahr_basis$ENDE_12_MONATE_MONAT <- NA
expo_erstesjahr_basis$ENDE_12_MONATE_JAHR <- NA

# ENDE_12_MONATE_MONAT steht in ENDE_12_MONATE an den Stellen 5-6 (substring)
# ENDE_12_MONATE_JAHR steht in ENDE_12_MONATE an den Stellen 1-4 (substring)
for(i in 1:nrow(expo_erstesjahr_basis)){
if(expo_erstesjahr_basis$ENDE_12_MONATE[i]!=0){
expo_erstesjahr_basis$ENDE_12_MONATE_MONAT[i] <- as.numeric(
substring(expo_erstesjahr_basis$ENDE_12_MONATE[i], 5, 6))
expo_erstesjahr_basis$ENDE_12_MONATE_JAHR[i] <- as.numeric(
substring(expo_erstesjahr_basis$ENDE_12_MONATE[i], 1, 4))
}
}

# Jetzt soll bei den Tätigkeiten die für das erste Tätigkeitsjahr berücksichtigt werden
# müssen (BERUF_BEACHTEN = 1) berechnet werden, wieviele Monate noch in den
# 12-Monats-Zeitraum fallen. Bsp: wenn die Tätigkeit "200103" beginnt und das
# ENDE_12_MONATE ist "200105", dann fallen 3 Monate noch in den 12-Monats-
# Zeitraum (es wird jeweils einschließlich Anfangs- und Endmonat gerechnet) !

# Variable DIFF_MONAT mit NA initialisieren
expo_erstesjahr_basis$DIFF_MONAT <- NA

for(i in 1:nrow(expo_erstesjahr_basis)){
if(expo_erstesjahr_basis$BERUF_BEACHTEN[i]==1){
# Wenn das ANF_JAHR gleich dem ENDE_12_MONATE_JAHR ist, dann ist die DIFF_MONAT:
# (ENDE_12_MONATE_MONAT - ANF_MONAT +1); ist also die Anzahl der Monate die für
# die jeweilige Tätigkeit für das erste Tätigkeitsjahr noch berücksichtigt werden
# müssen
if(expo_erstesjahr_basis$ANF_JAHR[i] ==
expo_erstesjahr_basis$ENDE_12_MONATE_JAHR[i]){
expo_erstesjahr_basis$DIFF_MONAT[i] <- (
expo_erstesjahr_basis$ENDE_12_MONATE_MONAT[i] -
expo_erstesjahr_basis$ANF_MONAT[i] + 1)
}
# Wenn das ANF_JAHR kleiner ist als das ENDE_12_MONATE_JAHR, dann ist die
# DIFF_MONAT: ((12 - ANF_MONAT) + ENDE_12_MONATE_MONAT +1); ist also die Anzahl
# der Monate die für die jeweilige Tätigkeit für das erste Tätigkeitsjahr noch
# berücksichtigt werden müssen
if(expo_erstesjahr_basis$ANF_JAHR[i] <
expo_erstesjahr_basis$ENDE_12_MONATE_JAHR[i]){
expo_erstesjahr_basis$DIFF_MONAT[i] <-((12 - expo_erstesjahr_basis$ANF_MONAT[i])
+ expo_erstesjahr_basis$ENDE_12_MONATE_MONAT[i] + 1)
}
}
}

# Jetzt wird noch abgeglichen, wieviele Monate in der Tätigkeit insgesamt gearbeitet
# wurden (DAUER) und wieviele Monate noch in den 12-Monats-Zeitraum fallen
# (DIFF_MONAT)

# Variable MONATE_BEACHTEN mit 0 initialisieren
expo_erstesjahr_basis$MONATE_BEACHTEN <- 0

# Nur für die Fälle abgleichen, bei denen DIFF_MONAT nicht NA ist
for(i in 1:nrow(expo_erstesjahr_basis)){
if(!is.na(expo_erstesjahr_basis$DIFF_MONAT[i])){
# Wenn die Anzahl der Monate die in der Tätigkeit gearbeitet wurden (DAUER)
# größer ist als die Anzahl der Monate, die noch in den 12-Monats-Zeitraum
# fallen (DIFF_MONAT), dann ist die Anzahl der Monate, die für das erste
# Tätigkeitsjahr noch beachtet werden muss (MONATE_BEACHTEN) gleich DIFF_MONAT
# Wenn die Anzahl der Monate die in der Tätigkeit gearbeitet wurden (DAUER)
# kleiner gleich der Anzahl der Monate, die noch in den 12-Monats-Zeitraum
# fallen (DIFF_MONAT) ist, dann ist die Anzahl der Monate, die für das
# erste Tätigkeitsjahr noch beachtet werden muss (MONATE_BEACHTEN) gleich
# DAUER;
# => MONATE_BEACHTEN entspricht also dem Minimum von DAUER und DIFF_MONAT
expo_erstesjahr_basis$MONATE_BEACHTEN[i] <- min(
expo_erstesjahr_basis$DIFF_MONAT[i],expo_erstesjahr_basis$DAUER[i])}
```

```
}

# Wo BERUF_BEACHTEN==1 und MIND_8_WST == 1 wird Expo pro Zeile berechnet, sonst 0

# zunächst Variablen mit 0 initialisieren
expo_erstesjahr_basis$HMW_beruf_erstesjahr <- 0
expo_erstesjahr_basis$LMW_beruf_erstesjahr <- 0
expo_erstesjahr_basis$MIXED_beruf_erstesjahr <- 0
expo_erstesjahr_basis$IRRPEAKS_beruf_erstesjahr <- 0
expo_erstesjahr_basis$LOWRISK_beruf_erstesjahr <- 0

# Wo BERUF_BEACHTEN==1 und MIND_8_WST == 1 wird Expo pro Zeile berechnet
for(i in 1:nrow(expo_erstesjahr_basis)){
if((expo_erstesjahr_basis$BERUF_BEACHTEN[i]==1)
&(expo_erstesjahr_basis$MIND_8_WST[i]==1)){
expo_erstesjahr_basis$HMW_beruf_erstesjahr[i]  <- (4.25 *
expo_erstesjahr_basis$WST[i] * expo_erstesjahr_basis$HMW[i] *
expo_erstesjahr_basis$MONATE_BEACHTEN[i])
expo_erstesjahr_basis$LMW_beruf_erstesjahr[i]  <- (4.25 *
expo_erstesjahr_basis$WST[i] * expo_erstesjahr_basis$LMW[i] *
expo_erstesjahr_basis$MONATE_BEACHTEN[i])
expo_erstesjahr_basis$MIXED_beruf_erstesjahr[i]  <- (4.25 *
expo_erstesjahr_basis$WST[i] * expo_erstesjahr_basis$MIXED[i] *
expo_erstesjahr_basis$MONATE_BEACHTEN[i])
expo_erstesjahr_basis$IRRPEAKS_beruf_erstesjahr[i]  <- (4.25 *
expo_erstesjahr_basis$WST[i] * expo_erstesjahr_basis$IRRPEAKS[i] *
expo_erstesjahr_basis$MONATE_BEACHTEN[i])
expo_erstesjahr_basis$LOWRISK_beruf_erstesjahr[i]  <- (4.25 *
expo_erstesjahr_basis$WST[i] * expo_erstesjahr_basis$LOWRISK[i] *
expo_erstesjahr_basis$MONATE_BEACHTEN[i])
}
}

# Jetzt über die Jobs aufsummieren
# pro Proband eine Zeile
expo_erstesjahr <- data.frame(HMW_erstesjahr_gesamt=numeric(nrow(basis)/10),
LMW_erstesjahr_gesamt=numeric(nrow(basis)/10), MIXED_erstesjahr_gesamt=numeric(nrow(basis)/10),
IRRPEAKS_erstesjahr_gesamt=numeric(nrow(basis)/10),
LOWRISK_erstesjahr_gesamt=numeric(nrow(basis)/10), HMW_erstesjahr_binaer=numeric(nrow(basis)/10),
LMW_erstesjahr_binaer=numeric(nrow(basis)/10), MIXED_erstesjahr_binaer=numeric(nrow(basis)/10),
IRRPEAKS_erstesjahr_binaer=numeric(nrow(basis)/10),
LOWRISK_erstesjahr_binaer=numeric(nrow(basis)/10))

# alle Variablen initialisieren mit NA
expo_erstesjahr$HMW_erstesjahr_gesamt <- NA
expo_erstesjahr$LMW_erstesjahr_gesamt <- NA
expo_erstesjahr$MIXED_erstesjahr_gesamt <- NA
expo_erstesjahr$IRRPEAKS_erstesjahr_gesamt <- NA
expo_erstesjahr$LOWRISK_erstesjahr_gesamt <- NA
expo_erstesjahr$HMW_erstesjahr_binaer <- NA
expo_erstesjahr$LMW_erstesjahr_binaer <- NA
expo_erstesjahr$MIXED_erstesjahr_binaer <- NA
expo_erstesjahr$IRRPEAKS_erstesjahr_binaer <- NA
expo_erstesjahr$LOWRISK_erstesjahr_binaer <- NA

# KNR übertragen
i <- 1
j <- 1
while (i <= nrow(expo_erstesjahr_basis)){
expo_erstesjahr$knr[j] <- as.character(expo_erstesjahr_basis$knr[i])
i <- i+10 # nächster Proband
j <- j+1  # nächster Proband in der neuen Matrix
}

# HMW_beruf_erstesjahr pro Proband über die 10 Jobs aufsummieren
i <- 1
j <- 1
while (i <= nrow(expo_erstesjahr_basis)){
expo_erstesjahr$HMW_erstesjahr_gesamt[j] <- (
expo_erstesjahr_basis$HMW_beruf_erstesjahr[i] +
expo_erstesjahr_basis$HMW_beruf_erstesjahr[i+1] +
expo_erstesjahr_basis$HMW_beruf_erstesjahr[i+2] +
expo_erstesjahr_basis$HMW_beruf_erstesjahr[i+3] +
expo_erstesjahr_basis$HMW_beruf_erstesjahr[i+4] +
expo_erstesjahr_basis$HMW_beruf_erstesjahr[i+5] +
expo_erstesjahr_basis$HMW_beruf_erstesjahr[i+6] +
expo_erstesjahr_basis$HMW_beruf_erstesjahr[i+7] +
expo_erstesjahr_basis$HMW_beruf_erstesjahr[i+8] +
expo_erstesjahr_basis$HMW_beruf_erstesjahr[i+9])
i <- i+10  # nächster Proband in expo_erstesjahr_basis
j <- j+1   # nächste Stelle in der neuen Matrix expo_erstesjahr
}

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_erstesjahr$HMW_erstesjahr_binaer[
expo_erstesjahr$HMW_erstesjahr_gesamt > 0] <- 1
expo_erstesjahr$HMW_erstesjahr_binaer[
expo_erstesjahr$HMW_erstesjahr_gesamt == 0] <- 0
```

```
# LMW_beruf_erstesjahr pro Proband über die 10 Jobs aufsummieren
i <- 1
j <- 1
while (i <= nrow(expo_erstesjahr_basis)){
expo_erstesjahr$LMW_erstesjahr_gesamt[j] <- (
expo_erstesjahr_basis$LMW_beruf_erstesjahr[i] +
expo_erstesjahr_basis$LMW_beruf_erstesjahr[i+1] +
expo_erstesjahr_basis$LMW_beruf_erstesjahr[i+2] +
expo_erstesjahr_basis$LMW_beruf_erstesjahr[i+3] +
expo_erstesjahr_basis$LMW_beruf_erstesjahr[i+4] +
expo_erstesjahr_basis$LMW_beruf_erstesjahr[i+5] +
expo_erstesjahr_basis$LMW_beruf_erstesjahr[i+6] +
expo_erstesjahr_basis$LMW_beruf_erstesjahr[i+7] +
expo_erstesjahr_basis$LMW_beruf_erstesjahr[i+8] +
expo_erstesjahr_basis$LMW_beruf_erstesjahr[i+9])
i <- i+10  # nächster Proband in expo_erstesjahr_basis
j <- j+1   # nächste Stelle in der neuen Matrix expo_erstesjahr
}

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_erstesjahr$LMW_erstesjahr_binaer[
expo_erstesjahr$LMW_erstesjahr_gesamt > 0] <- 1
expo_erstesjahr$LMW_erstesjahr_binaer[
expo_erstesjahr$LMW_erstesjahr_gesamt == 0] <- 0

# MIXED_beruf_erstesjahr pro Proband über die 10 Jobs aufsummieren
i <- 1
j <- 1
while (i <= nrow(expo_erstesjahr_basis)){
expo_erstesjahr$MIXED_erstesjahr_gesamt[j] <- (
expo_erstesjahr_basis$MIXED_beruf_erstesjahr[i] +
expo_erstesjahr_basis$MIXED_beruf_erstesjahr[i+1] +
expo_erstesjahr_basis$MIXED_beruf_erstesjahr[i+2] +
expo_erstesjahr_basis$MIXED_beruf_erstesjahr[i+3] +
expo_erstesjahr_basis$MIXED_beruf_erstesjahr[i+4] +
expo_erstesjahr_basis$MIXED_beruf_erstesjahr[i+5] +
expo_erstesjahr_basis$MIXED_beruf_erstesjahr[i+6] +
expo_erstesjahr_basis$MIXED_beruf_erstesjahr[i+7] +
expo_erstesjahr_basis$MIXED_beruf_erstesjahr[i+8] +
expo_erstesjahr_basis$MIXED_beruf_erstesjahr[i+9])
i <- i+10  # nächster Proband in expo_erstesjahr_basis
j <- j+1   # nächste Stelle in der neuen Matrix expo_erstesjahr
}

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_erstesjahr$MIXED_erstesjahr_binaer[
expo_erstesjahr$MIXED_erstesjahr_gesamt > 0] <- 1
expo_erstesjahr$MIXED_erstesjahr_binaer[
expo_erstesjahr$MIXED_erstesjahr_gesamt == 0] <- 0

# IRRPEAKS_beruf_erstesjahr pro Proband über die 10 Jobs aufsummieren
i <- 1
j <- 1
while (i <= nrow(expo_erstesjahr_basis)){
expo_erstesjahr$IRRPEAKS_erstesjahr_gesamt[j] <- (
expo_erstesjahr_basis$IRRPEAKS_beruf_erstesjahr[i] +
expo_erstesjahr_basis$IRRPEAKS_beruf_erstesjahr[i+1] +
expo_erstesjahr_basis$IRRPEAKS_beruf_erstesjahr[i+2] +
expo_erstesjahr_basis$IRRPEAKS_beruf_erstesjahr[i+3] +
expo_erstesjahr_basis$IRRPEAKS_beruf_erstesjahr[i+4] +
expo_erstesjahr_basis$IRRPEAKS_beruf_erstesjahr[i+5] +
expo_erstesjahr_basis$IRRPEAKS_beruf_erstesjahr[i+6] +
expo_erstesjahr_basis$IRRPEAKS_beruf_erstesjahr[i+7] +
expo_erstesjahr_basis$IRRPEAKS_beruf_erstesjahr[i+8] +
expo_erstesjahr_basis$IRRPEAKS_beruf_erstesjahr[i+9])
i <- i+10  # nächster Proband in expo_erstesjahr_basis
j <- j+1   # nächste Stelle in der neuen Matrix expo_erstesjahr
}

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_erstesjahr$IRRPEAKS_erstesjahr_binaer[
expo_erstesjahr$IRRPEAKS_erstesjahr_gesamt > 0] <- 1
expo_erstesjahr$IRRPEAKS_erstesjahr_binaer[
expo_erstesjahr$IRRPEAKS_erstesjahr_gesamt == 0] <- 0

# LOWRISK_beruf_erstesjahr pro Proband über die 10 Jobs aufsummieren
i <- 1
j <- 1
while (i <= nrow(expo_erstesjahr_basis)){
expo_erstesjahr$LOWRISK_erstesjahr_gesamt[j] <- (
expo_erstesjahr_basis$LOWRISK_beruf_erstesjahr[i] +
expo_erstesjahr_basis$LOWRISK_beruf_erstesjahr[i+1] +
expo_erstesjahr_basis$LOWRISK_beruf_erstesjahr[i+2] +
expo_erstesjahr_basis$LOWRISK_beruf_erstesjahr[i+3] +
expo_erstesjahr_basis$LOWRISK_beruf_erstesjahr[i+4] +
expo_erstesjahr_basis$LOWRISK_beruf_erstesjahr[i+5] +
expo_erstesjahr_basis$LOWRISK_beruf_erstesjahr[i+6] +
```

```
expo_erstesjahr_basis$LOWRISK_beruf_erstesjahr[i+7] +
expo_erstesjahr_basis$LOWRISK_beruf_erstesjahr[i+8] +
expo_erstesjahr_basis$LOWRISK_beruf_erstesjahr[i+9])
i <- i+10  # nächster Proband in expo_erstesjahr_basis
j <- j+1   # nächste Stelle in der neuen Matrix expo_erstesjahr
}

# binäre Variable erstellen - gibt an ob Exposition bestand oder nicht
expo_erstesjahr$LOWRISK_erstesjahr_binaer[
expo_erstesjahr$LOWRISK_erstesjahr_gesamt > 0] <- 1
expo_erstesjahr$LOWRISK_erstesjahr_binaer[
expo_erstesjahr$LOWRISK_erstesjahr_gesamt == 0] <- 0

# Datensatz abspeichern
save(expo_erstesjahr, file="expo_erstesjahr.RData")
```

D Imputation der Tätigkeitsangaben

Für die Imputation der Tätigkeitsangaben wurde eine Funktion geschrieben, in der automatisch die Zeitangaben und die Wochenstunden imputiert werden. Dieser Funktion übergibt man einen Datensatz und einen Startwert. Als Output erhält man den vervollständigten Datensatz.
Als Zusatzfunktion wurde auch eine Option für das Imputieren der Expositionen eingefügt (diese ist derzeit allerdings auskommentiert), da im Laufe der Arbeit das Imputieren der Expositionen in Betracht gezogen, später aber verworfen wurde. Folgender Abschnitt erläutert kurz, wie man dabei vorgehen könnte.

Imputation der fehlenden Exposition
Werden von einem Probanden keine Tätigkeiten angegeben, so kann nicht mit Hilfe eines ISCO-Codes kodiert werden. Als Folge des fehlenden ISCO-Codes kann demnach über die Job-Exposure-Matrix keine Exposition zugewiesen werden. Diese Exposition muss daher imputiert werden. Die Exposition besteht aus 22 Variablen (Tier-, Latex-, Mehlexposition usw.), die jeweils mit 0 (=keine Exposition) oder 1 (=Exposition) kodiert sind. Die Expositionen in den einzelnen Variablen sind allerdings nicht frei kombinierbar, sondern treten in bestimmten Mustern auf. Beispielsweise gibt es die Kombination Tier- gemeinsam mit Latexexposition (z.B. bei Tierärzten). Die Kombination Tier- gemeinsam mit Mehlexposition tritt allerdings nicht auf. Daher werden für die Imputation der Exposition zunächst die verschiedenen Expositionsmuster, die in den beobachteten Daten vorkommen, festgestellt und deren Häufigkeiten ermittelt. Dann kann das jeweilige Expositionsmuster aus der vorliegenden empirischen Verteilung gezogen werden. In einem letzten Schritt wird das Expomuster auf die 22 einzelnen Variablen übertragen, so dass für diese Probanden in jeder dieser Variablen entweder eine 0 oder eine 1 eingetragen wird. Die Zusammenfassung der Exposition zu den fünf Expositionshauptgruppen erfolgt erst nach der Imputation auf Basis der Detailebene. Durch dieses Vorgehen wäre auch eine spätere Analyse der einzelnen Expositonsuntergruppen möglich.

```
##############################################################################
####           FUNKTION FÜR DIE IMPUTATION DER TÄTIGKEITSANGABEN           ####
##############################################################################

# Funktion für die Imputation der Tätigkeitsangaben: Als Argumente werden eingegeben:
# datensatz (enthält fehlende Werte in den Tätigkeitsangaben, Confoundervariablen vollst.)
# startwert (um immer eine andere zufällige Ziehung der Werte für die Imputation
# zu erhalten und es nachvollziehbar zu machen)

Imputation_Berufsdaten <- function(datensatz, startwert){
# Subset für SOLAR I:
datensatz_s1 <- subset(datensatz, STUDIE == 1)
# Subset für SOLAR II:
datensatz_s2 <- subset(datensatz, STUDIE == 2)

##############################################################################
####                  Wochenstunden (WST) imputieren                      ####
##############################################################################

# Indikator, der angibt ob die WST imputiert wurden anlegen
datensatz$IMP_WST <- 0

############################### SOLAR I ########################################
print("***************Imputation der Wochenstunden in SOLAR I***************")
# Für solar 1:
a1 <- subset(datensatz_s1, is.na(WST) & ISCO != 8888 & ISCO != 9999
& ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
# hier nur die Zeilen drin in die imputiert werden soll

# Datensatz anlegen, in den jeweils index der Zeile und imputierte WST
# geschrieben werden (1.Spalte: index, 2.Spalte: wochenstunden)

imput_werte1 <- data.frame(index=numeric(nrow(a1)),
wochenstunden = numeric(nrow(a1)))

# j auf 1 setzen
j <- 1

# Startwert setzen (der der Funktion als Argument übergeben wurde)
set.seed(startwert)

for (i in 1: nrow(a1)){
print("Nächste Stelle i")
# Geschlecht,ISCO und Index an der i-ten Stelle aus subset betrachten
isco <- a1$ISCO[i]
print("ISCO an der Stelle i")
print(isco)
geschlecht <- a1$f02x[i]
print("Geschlecht an der Stelle i")
print(geschlecht)
index <- a1$index[i]
print("Index an der Stelle i")
print(index)
# aus großen Datensatz alle mit gleichem ISCO und Geschlecht ziehen, durch
# IMP_WST == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen die WST bereits imputiert worden sind
b <- subset(datensatz, ISCO==isco & f02x==geschlecht & !is.na(WST) & STUDIE == 1
 & IMP_WST==0 )
print("Anzahl der Fälle mit gleichem ISCO und gleichem Geschlecht")
```

```
print(nrow(b))

if(nrow(b)>0){ # Wenn es noch andere Fälle mit gleichem ISCO und WST gibt
print("b groesser als 0 also auf Geschlecht und ISCO bedingen")
table1 <- table(b$WST)
print("Table der Wochenstunden bedingt auf Geschlecht und ISCO")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für WST berechnen
print("Wahrscheinlichkeiten der Wochenstunden bedingt auf Geschlecht und ISCO")
print(prob1)
# dann aus WST mit diesen Wahrscheinlichkeiten ziehen
}

if(nrow(b)==0){ # also wenn es keinen entsprechenden Fall mit gleichem ISCO und
# gleichem Geschlecht gibt bei dem die WST fehlen (d.h. WST fehlen nur in diesem
# einen Fall mit diesem ISCO und diesem Geschlecht)
print("b gleich 0 also nur auf Geschlecht bedingen")
# aus großem Datensatz alle mit gleichem Geschlecht ziehen, durch
# IMP_WST == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen die WST bereits imputiert worden sind
c <- subset(datensatz, f02x==geschlecht & !is.na(WST) & STUDIE == 1
& IMP_WST==0 )
table1 <- table(c$WST)
print("Table der Wochenstunden bedingt auf Geschlecht")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für WST berechnen
print("Wahrscheinlichkeiten der Wochenstunden bedingt auf Geschlecht")
print(prob1)
# dann aus WST mit diesen Wahrscheinlichkeiten ziehen
}
# Wochenstundenwert für die Imputation ziehen mit der Funktion sample
imput <- sample(names(prob1), size = 1, replace=TRUE, prob = prob1)
# Im Datensatz imputieren, der die Daten aus SOLAR I und SOLAR II enthält
print("Wochenstunden-Wert der an dieser Stelle imputiert werden soll")
print(imput)
# Index der Zeile (index) in 1.Spalte des Datensatzes imput_werte1 speichern
imput_werte1$index[j] <- index
# Wochenstundenwert (imput) in 2.Spalte des Datensatzes imput_werte1 speichern
imput_werte1$wochenstunden[j] <- imput
print("Matrix,1.Spalte:Index der Zeile,2.Spalte:Zu imputierende Wochenstunden")
print(imput_werte1)
# Jetzt den Wert im "großen" Datensatz imputieren
for(k in 1:nrow(datensatz)){
# Wenn der Index übereinstimmt
if(datensatz$index[k] == imput_werte1$index[j]){
# Zu imputierende Wochenstunden in der Zeile mit diesem Index imputieren
datensatz$WST[k] <- as.numeric(imput_werte1$wochenstunden[j])
print("Wochenstunden-Wert der imputiert wird")
print(datensatz$WST[k])
# Indikator IMP_WST auf 1 setzen, d.h. die Wochenstunden werden imputiert
datensatz$IMP_WST[k] <- 1
}
}
j <- j+1
}

############################### SOLAR II #####################################
print("***************Imputation der Wochenstunden in SOLAR II***************")
# Für solar 2:
a2 <- subset(datensatz_s2, is.na(WST) & ISCO != 8888 & ISCO != 9999
& ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
# hier nur die Zeilen drin in die imputiert werden soll

# Datensatz anlegen, in den jeweils index der Zeile und imputierte WST
# geschrieben werden (1.Spalte: index, 2.Spalte: wochenstunden)

imput_werte2 <- data.frame(index=numeric(nrow(a2)),
wochenstunden = numeric(nrow(a2)))

# j auf 1 setzen
j <- 1

# Startwert setzen (der der Funktion als Argument übergeben wurde)
set.seed(startwert)

for (i in 1:nrow(a2)){
print("Nächste Stelle i")
# Geschlecht,ISCO und Index an der i-ten Stelle aus subset betrachten
isco <- a2$ISCO[i]
print("ISCO an der Stelle i")
print(isco)
geschlecht <- a2$f02x[i]
print("Geschlecht an der Stelle i")
print(geschlecht)
index <- a2$index[i]
print("Index an der Stelle i")
print(index)
# aus großen Datensatz alle mit gleichem ISCO und Geschlecht ziehen, durch
```

```
# IMP_WST == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen die WST bereits imputiert worden sind
b <- subset(datensatz, ISCO==isco & f02x==geschlecht & !is.na(WST) & STUDIE == 2
 & IMP_WST==0 )
print("Anzahl der Fälle mit gleichem ISCO und gleichem Geschlecht")
print(nrow(b))

if(nrow(b)>0){ # Wenn es noch andere Fälle mit gleichem ISCO und WST gibt
print("b groesser als 0 also auf Geschlecht und ISCO bedingen")
table1 <- table(b$WST)
print("Table der Wochenstunden bedingt auf Geschlecht und ISCO")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für WST berechnen
print("Wahrscheinlichkeiten der Wochenstunden bedingt auf Geschlecht und ISCO")
print(prob1)
# dann aus WST mit diesen Wahrscheinlichkeiten ziehen
}

if(nrow(b)==0){ # also wenn es keinen entsprechenden Fall mit gleichem ISCO und
# gleichem Geschlecht gibt bei dem die WST fehlen (d.h. WST fehlen nur in diesem
# einen Fall mit diesem ISCO und diesem Geschlecht)
print("b gleich 0 also nur auf Geschlecht bedingen")
# aus großem Datensatz alle mit gleichem Geschlecht ziehen, durch
# IMP_WST == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen die WST bereits imputiert worden sind
c <- subset(datensatz, f02x==geschlecht & !is.na(WST) & STUDIE == 2
& IMP_WST==0 )
table1 <- table(c$WST)
print("Table der Wochenstunden bedingt auf Geschlecht")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für WST berechnen
print("Wahrscheinlichkeiten der Wochenstunden bedingt auf Geschlecht")
print(prob1)
# dann aus WST mit diesen Wahrscheinlichkeiten ziehen
}
# Wochenstundenwert für die Imputation ziehen mit der Funktion sample
imput <- sample(names(prob1), size = 1, replace=TRUE, prob = prob1)
# Im Datensatz imputieren, der die Daten aus SOLAR I und SOLAR II enthält
print("Wochenstunden-Wert der an dieser Stelle imputiert werden soll")
print(imput)
# Index der Zeile (index) in 1.Spalte des Datensatzes imput_werte2 speichern
imput_werte2$index[j] <- index
# Wochenstundenwert (imput) in 2.Spalte des Datensatzes imput_werte2 speichern
imput_werte2$wochenstunden[j] <- imput
print("Matrix,1.Spalte:Index der Zeile,2.Spalte:Zu imputierende Wochenstunden")
print(imput_werte2)
# Jetzt den Wert im "großen" Datensatz imputieren
for(k in 1:nrow(datensatz)){
# Wenn der Index übereinstimmt
if(datensatz$index[k] == imput_werte2$index[j]){
# Zu imputierende Wochenstunden in der Zeile mit diesem Index imputieren
datensatz$WST[k] <- as.numeric(imput_werte2$wochenstunden[j])
print("Wochenstunden-Wert der imputiert wird")
print(datensatz$WST[k])
# Indikator IMP_WST auf 1 setzen, d.h. die Wochenstunden werden imputiert
datensatz$IMP_WST[k] <- 1
}
}
j <- j+1
}

################################################################################
####                  Anfangsjahr (ANF_JAHR) imputieren                     ####
################################################################################

# Indikator, der angibt ob das ANF_JAHR imputiert wurde anlegen
datensatz$IMP_AJ <- 0

############################### SOLAR I ########################################
# in SOLAR I nur bedingen auf SES_r
print("***************Imputation des Anfangsjahrs in SOLAR I***************")
# Für solar 1:
a3 <- subset(datensatz_s1, is.na(ANF_JAHR) & ISCO != 8888 & ISCO != 9999
& ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
# hier nur die Zeilen drin in die imputiert werden soll

# Datensatz anlegen, in den jeweils index der Zeile und imputiertes ANF_JAHR
# geschrieben werden (1.Spalte: index, 2.Spalte: anfangsjahr)

imput_werte3 <- data.frame(index=numeric(nrow(a3)),
anfangsjahr = numeric(nrow(a3)))

# j auf 1 setzen
j <- 1

# Startwert setzen (der der Funktion als Argument übergeben wurde)
set.seed(startwert)
```

```
for (i in 1: nrow(a3)){
print("Nächste Stelle i")
# SES und Index an der i-ten Stelle aus subset betrachten
ses <- a3$SES_r[i]
print("SES an der Stelle i")
print(ses)
index <- a3$index[i]
print("Index an der Stelle i")
print(index)
# aus großen Datensatz alle mit gleichem SES ziehen, durch
# IMP_AJ == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das ANF_JAHR bereits imputiert wurde
b <- subset(datensatz, SES_r==ses & !is.na(ANF_JAHR) & STUDIE == 1
 & IMP_AJ==0 )
print("Anzahl der Fälle mit gleichem SES")
print(nrow(b))
# Hier gibt es auf jeden Fall noch andere Fälle mit gleichem SES ! D.h.
# nrow(b) > 0 immer !
table1 <- table(b$ANF_JAHR)
print("Table der Anfangsjahre bedingt auf SES_r")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für ANF_JAHR
# berechnen
print("Wahrscheinlichkeiten der Anfangsjahre bedingt auf SES_r")
print(prob1)
# dann aus den Anfangsjahren mit diesen Wahrscheinlichkeiten ziehen
# Anfangsjahr für die Imputation ziehen mit der Funktion sample
imput <- sample(names(prob1), size = 1, replace=TRUE, prob = prob1)
# Im Datensatz imputieren, der die Daten aus SOLAR I und SOLAR II enthält
print("Anfangsjahr das an dieser Stelle imputiert werden soll")
print(imput)
# Index der Zeile (index) in 1.Spalte des Datensatzes imput_werte3 speichern
imput_werte3$index[j] <- index
# Anfangsjahr (imput) in 2.Spalte des Datensatzes imput_werte3 speichern
imput_werte3$anfangsjahr[j] <- imput
print("Matrix,1.Spalte:Index der Zeile,2.Spalte:Zu imputierendes Anfangsjahr")
print(imput_werte3)
# Jetzt den Wert im "großen" Datensatz imputieren
for(k in 1:nrow(datensatz)){
# Wenn der Index übereinstimmt
if(datensatz$index[k] == imput_werte3$index[j]){
# Zu imputierendes Anfangsjahr in der Zeile mit diesem Index imputieren
datensatz$ANF_JAHR[k] <- as.numeric(imput_werte3$anfangsjahr[j])
print("Anfangsjahr das imputiert wird")
print(datensatz$ANF_JAHR[k])
# Indikator IMP_AJ auf 1 setzen, d.h. das Anfangsjahr werden imputiert
datensatz$IMP_AJ[k] <- 1
}
}
j <- j+1
}

############################### SOLAR II #####################################
# In SOLAR II bedingen auf s2BERUF und SES_r
print("**************Imputation des Anfangsjahrs in SOLAR II**************")
# Für solar 2:
a4 <- subset(datensatz_s2, is.na(ANF_JAHR) & ISCO != 8888 & ISCO != 9999
& ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
# hier nur die Zeilen drin in die imputiert werden soll

# Datensatz anlegen, in den jeweils index der Zeile und imputiertes ANF_JAHR
# geschrieben werden (1.Spalte: index, 2.Spalte: anfangsjahr)

imput_werte4 <- data.frame(index=numeric(nrow(a4)),
anfangsjahr = numeric(nrow(a4)))

# j auf 1 setzen
j <- 1

# Startwert setzen (der der Funktion als Argument übergeben wurde)
set.seed(startwert)

for (i in 1:nrow(a4)){
print("Nächste Stelle i")
# SES_r, s2BERUF und Index an der i-ten Stelle aus subset betrachten
beruf <- a4$s2BERUF[i]
print("s2BERUF an der Stelle i")
print(beruf)
ses <- a4$SES_r[i]
print("SES an der Stelle i")
print(ses)
index <- a4$index[i]
print("Index an der Stelle i")
print(index)
# aus großen Datensatz alle mit gleichem s2BERUF und SES_r ziehen, durch
# IMP_AJ == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das ANF_JAHR bereits imputiert worden sind
b <- subset(datensatz, SES_r==ses & s2BERUF==beruf & !is.na(ANF_JAHR)
```

```
& STUDIE == 2 & IMP_AJ==0 )
print("Anzahl der Fälle mit gleichem s2BERUF und gleichem SES_r")
print(nrow(b))

if(nrow(b)>0){ # Wenn es noch andere Fälle mit gleichem s2BERUF und SES_r gibt
print("b groesser als 0 also auf s2BERUF und SES_r bedingen")
table1 <- table(b$ANF_JAHR)
print("Table der Anfangsjahre bedingt auf s2BERUF und SES_r")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für ANF_JAHR
# berechnen
print("Wahrscheinlichkeiten der Anfangsjahre bedingt auf s2BERUF und SES_r")
print(prob1)
# dann aus ANF_JAHR mit diesen Wahrscheinlichkeiten ziehen
}

if(nrow(b)==0){ # also wenn es keinen entsprechenden Fall mit gleichem s2BERUF,
# gleichem SES_r gibt bei dem das ANF_JAHR fehlt (d.h. ANF_JAHR fehlt nur in
# diesem einen Fall mit diesem s2BERUF und diesem SES_r)
print("b gleich 0 also nur auf SES_r bedingen")
# aus großem Datensatz alle mit gleichem SES_r ziehen, durch
# IMP_AJ == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das ANF_JAHR bereits imputiert wurde
c <- subset(datensatz, SES_r==ses & !is.na(ANF_JAHR) & STUDIE == 2
& IMP_AJ==0 )
table1 <- table(c$ANF_JAHR)
print("Table der Anfangsjahre bedingt auf SES_r")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für ANF_JAHR
# berechnen
print("Wahrscheinlichkeiten der Anfangsjahre bedingt auf SES_r")
print(prob1)
# dann aus ANF_JAHR mit diesen Wahrscheinlichkeiten ziehen
}
# Anfangsjahr für die Imputation ziehen mit der Funktion sample
imput <- sample(names(prob1), size = 1, replace=TRUE, prob = prob1)
# Im Datensatz imputieren, der die Daten aus SOLAR I und SOLAR II enthält
print("Anfangsjahr das an dieser Stelle imputiert werden soll")
print(imput)
# Index der Zeile (index) in 1.Spalte des Datensatzes imput_werte4 speichern
imput_werte4$index[j] <- index
# Anfangsjahr (imput) in 2.Spalte des Datensatzes imput_werte4 speichern
imput_werte4$anfangsjahr[j] <- imput
print("Matrix,1.Spalte:Index der Zeile,2.Spalte:Zu imputierendes Anfangsjahr")
print(imput_werte4)
# Jetzt den Wert im "großen" Datensatz imputieren
for(k in 1:nrow(datensatz)){
# Wenn der Index übereinstimmt
if(datensatz$index[k] == imput_werte4$index[j]){
  # Zu imputierendes Anfangsjahr in der Zeile mit diesem Index imputieren
  datensatz$ANF_JAHR[k] <- as.numeric(imput_werte4$anfangsjahr[j])
  print("Anfangsjahr das imputiert wird")
  print(datensatz$ANF_JAHR[k])
  # Indikator IMP_AJ auf 1 setzen, d.h. das Anfangsjahr wurde imputiert
  datensatz$IMP_AJ[k] <- 1
  }
  }
  j <- j+1
  # immer: j um 1 erhöhen, d.h. der nächste Wert im Vektor wird betrachet
  }

################################################################################
####                  Endjahr (END_JAHRx) imputieren                        ####
################################################################################

# Indikator, der angibt ob das END_JAHRx imputiert wurde anlegen
datensatz$IMP_EJ <- 0

############################### SOLAR I ###########################################
# in SOLAR I nur bedingen auf SES_r
print("***************Imputation des Endjahrs in SOLAR I***************")
# Für solar 1:
a7 <- subset(datensatz_s1, is.na(END_JAHRx) & ISCO != 8888 & ISCO != 9999
& ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
# hier nur die Zeilen drin in die imputiert werden soll

# Datensatz anlegen, in den jeweils index der Zeile und imputiertes END_JAHRx
# geschrieben werden (1.Spalte: index, 2.Spalte: endjahr)

imput_werte7 <- data.frame(index=numeric(nrow(a7)),
endjahr1 = numeric(nrow(a7)), endjahr2 = numeric(nrow(a7)),
endjahr3 = numeric(nrow(a7)), endjahr4 = numeric(nrow(a7)),
endjahr5 = numeric(nrow(a7)), endjahr6 = numeric(nrow(a7)),
endjahr7 = numeric(nrow(a7)), endjahr8 = numeric(nrow(a7)),
endjahr9 = numeric(nrow(a7)), endjahr10 = numeric(nrow(a7)),
endjahr11 = numeric(nrow(a7)), endjahr12 = numeric(nrow(a7)),
endjahr13 = numeric(nrow(a7)), endjahr14 = numeric(nrow(a7)),
endjahr15 = numeric(nrow(a7)), endjahr16 = numeric(nrow(a7)),
```

```
endjahr17 = numeric(nrow(a7)), endjahr18 = numeric(nrow(a7)),
endjahr19 = numeric(nrow(a7)), endjahr20 = numeric(nrow(a7))
)

# j auf 1 setzen
j <- 1

# Startwert setzen (der der Funktion als Argument übergeben wurde)
set.seed(startwert)

for (i in 1: nrow(a7)){
print("Nächste Stelle i")
# SES und Index an der i-ten Stelle aus subset betrachten
ses <- a7$SES_r[i]
print("SES an der Stelle i")
print(ses)
index <- a7$index[i]
print("Index an der Stelle i")
print(index)
# aus großen Datensatz alle mit gleichem SES ziehen, durch
# IMP_EJ == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das END_JAHRx bereits imputiert wurde
b <- subset(datensatz, SES_r==ses & !is.na(END_JAHRx) & STUDIE == 1
 & IMP_EJ==0 )
print("Anzahl der Fälle mit gleichem SES")
print(nrow(b))
# Hier gibt es auf jeden Fall noch andere Fälle mit gleichem SES ! D.h.
# nrow(b) > 0 immer !
table1 <- table(b$END_JAHRx)
print("Table der Endjahre bedingt auf SES_r")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für END_JAHRx
# berechnen
print("Wahrscheinlichkeiten der Endjahre bedingt auf SES_r")
print(prob1)
# dann aus den Endjahren mit diesen Wahrscheinlichkeiten ziehen
# Endjahr für die Imputation ziehen mit der Funktion sample
imput <- as.list(sample(names(prob1), size = 20, replace=TRUE, prob = prob1))
# Im Datensatz imputieren, der die Daten aus SOLAR I und SOLAR II enthält
print("Endjahr das an dieser Stelle imputiert werden soll")
print(imput)
# Index der Zeile (index) in 1.Spalte des Datensatzes imput_werte7 speichern
imput_werte7$index[j] <- index
# Endjahr (imput) in 2.Spalte des Datensatzes imput_werte7 speichern
imput_werte7$endjahr1[j] <- as.numeric(imput[[1]])
imput_werte7$endjahr2[j] <- as.numeric(imput[[2]])
imput_werte7$endjahr3[j] <- as.numeric(imput[[3]])
imput_werte7$endjahr4[j] <- as.numeric(imput[[4]])
imput_werte7$endjahr5[j] <- as.numeric(imput[[5]])
imput_werte7$endjahr6[j] <- as.numeric(imput[[6]])
imput_werte7$endjahr7[j] <- as.numeric(imput[[7]])
imput_werte7$endjahr8[j] <- as.numeric(imput[[8]])
imput_werte7$endjahr9[j] <- as.numeric(imput[[9]])
imput_werte7$endjahr10[j] <- as.numeric(imput[[10]])
imput_werte7$endjahr11[j] <- as.numeric(imput[[11]])
imput_werte7$endjahr12[j] <- as.numeric(imput[[12]])
imput_werte7$endjahr13[j] <- as.numeric(imput[[13]])
imput_werte7$endjahr14[j] <- as.numeric(imput[[14]])
imput_werte7$endjahr15[j] <- as.numeric(imput[[15]])
imput_werte7$endjahr16[j] <- as.numeric(imput[[16]])
imput_werte7$endjahr17[j] <- as.numeric(imput[[17]])
imput_werte7$endjahr18[j] <- as.numeric(imput[[18]])
imput_werte7$endjahr19[j] <- as.numeric(imput[[19]])
imput_werte7$endjahr20[j] <- as.numeric(imput[[20]])
print("Matrix,1.Spalte:Index der Zeile,2.Spalte:Zu imputierendes Endjahr")
print(imput_werte7)
# Jetzt den Wert im "großen" Datensatz imputieren
for(k in 1:nrow(datensatz)){
# Wenn der Index übereinstimmt
if(datensatz$index[k] == imput_werte7$index[j]){
# Zu imputierendes Endjahr in der Zeile mit diesem Index imputieren
if (!is.na(datensatz$ANF_MONAT[k])&!is.na(datensatz$END_MONATx[k])&
(datensatz$ANF_MONAT[k] > datensatz$END_MONATx[k])){
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr1[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr1[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr2[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr2[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
```

```
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr3[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr3[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr4[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr4[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr5[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr5[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr6[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr6[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr7[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr7[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr8[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr8[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr9[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr9[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr10[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr10[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr11[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr11[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr12[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr12[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr13[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr13[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
```

```
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr14[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr14[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr15[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr15[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr16[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr16[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr17[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr17[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr18[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr18[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr19[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr19[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte7$endjahr20[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr20[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}}}}}}}}}}}}}}}}}}}}
}
if (is.na(datensatz$ANF_MONAT[k]) | is.na(datensatz$END_MONATx[k]) |
!is.na(datensatz$ANF_MONAT[k])&!is.na(datensatz$END_MONATx[k])
& (datensatz$ANF_MONAT[k] <= datensatz$END_MONATx[k])){
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr1[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr1[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr2[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr2[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr3[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr3[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr4[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr4[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
```

```
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr5[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr5[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr6[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr6[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr7[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr7[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr8[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr8[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr9[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr9[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr10[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr10[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr11[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr11[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr12[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr12[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr13[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr13[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr14[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr14[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr15[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr15[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
```

```
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr16[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr16[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr17[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr17[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr18[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr18[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr19[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr19[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte7$endjahr20[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte7$endjahr20[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}}}}}}}}}}}}}}}}}}}}
j <- j+1
}

############################## SOLAR II #######################################
# in SOLAR II bedingen auf s2BERUF und SES_r
print("***************Imputation des Endjahrs in SOLAR II***************")
# Für solar 2:
a8 <- subset(datensatz, STUDIE == 2 & is.na(END_JAHRx) & ISCO != 8888 & ISCO != 9999
& ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
# hier nur die Zeilen drin in die imputiert werden soll


# Datensatz anlegen, in den jeweils index der Zeile und imputiertes END_JAHRx
# geschrieben werden (1.Spalte: index, 2.Spalte: endjahr)

imput_werte8 <- data.frame(index=numeric(nrow(a8)),
endjahr1 = numeric(nrow(a8)), endjahr2 = numeric(nrow(a8)),
endjahr3 = numeric(nrow(a8)), endjahr4 = numeric(nrow(a8)),
endjahr5 = numeric(nrow(a8)), endjahr6 = numeric(nrow(a8)),
endjahr7 = numeric(nrow(a8)), endjahr8 = numeric(nrow(a8)),
endjahr9 = numeric(nrow(a8)), endjahr10 = numeric(nrow(a8)),
endjahr11 = numeric(nrow(a8)), endjahr12 = numeric(nrow(a8)),
endjahr13 = numeric(nrow(a8)), endjahr14 = numeric(nrow(a8)),
endjahr15 = numeric(nrow(a8)), endjahr16 = numeric(nrow(a8)),
endjahr17 = numeric(nrow(a8)), endjahr18 = numeric(nrow(a8)),
endjahr19 = numeric(nrow(a8)), endjahr20 = numeric(nrow(a8))
)

# j auf 1 setzen
j <- 1

# Startwert setzen (der der Funktion als Argument übergeben wurde)
set.seed(startwert)

for (i in 1: nrow(a8)){
print("Nächste Stelle i")
# SES_r, s2Beruf und Index an der i-ten Stelle aus subset betrachten
beruf <- a8$s2BERUF[i]
print("s2BERUF an der Stelle i")
print(beruf)
ses <- a8$SES_r[i]
print("SES an der Stelle i")
print(ses)
index <- a8$index[i]
print("Index an der Stelle i")
```

```
print(index)
# aus großen Datensatz alle mit gleichem s2BERUF SES_r ziehen, durch
# IMP_EJ == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das END_JAHR bereits imputiert wurde
b <- subset(datensatz, s2BERUF == beruf & SES_r==ses & !is.na(END_JAHRx)
& STUDIE == 2 & IMP_EJ==0 )
print("Anzahl der Fälle mit gleichem SES und s2BERUF")
print(nrow(b))

if(nrow(b)>0){ # Wenn es noch andere Fälle mit gleichem s2BERUF und SES_r gibt
print("b groesser als 0 also auf s2BERUF und SES_r bedingen")
table1 <- table(b$END_JAHRx)
print("Table der Endjahre bedingt auf SES_r und s2BERUF")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für END_JAHRx
# berechnen
print("Wahrscheinlichkeiten der Endjahre bedingt auf SES_r und s2BERUF")
print(prob1)
}
if(nrow(b)==0 | (nrow(b)==1 & b$END_JAHRx[1] < a8$ANF_JAHR[i]) |
(nrow(b)==2 & b$END_JAHRx[1] < a8$ANF_JAHR[i] & b$END_JAHRx[2] < a8$ANF_JAHR[i])
|(nrow(b)==3 & b$END_JAHRx[1] < a8$ANF_JAHR[i] & b$END_JAHRx[2] < a8$ANF_JAHR[i]
& b$END_JAHRx[3] < a8$ANF_JAHR[i])){
# Wenn es sonst keinen Fall mit gleichem s2BERUF und
# gleichem SES_r gibt bei dem das END_JAHRx fehlt (d.h. END_JAHRx fehlt nur in
# diesem einen Fall mit diesem s2BERUF und diesem SES_r)
# oder es gibt nur 1(2/3) fälle und bei denen wäre dann das Anfangsjahr >
# Endjahr
print("b gleich 0 also nur auf SES_r bedingen")
# aus großem Datensatz alle mit gleichem SES_r ziehen, durch
# IMP_EJ == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das END_JAHRx bereits imputiert wurde
c <- subset(datensatz, SES_r==ses & !is.na(END_JAHRx) & STUDIE == 2
& IMP_EJ==0 )
table1 <- table(c$END_JAHRx)
print("Table der Endjahre bedingt auf SES_r")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für END_JAHRx
# berechnen
print("Wahrscheinlichkeiten der Endjahre bedingt auf Geschlecht")
print(prob1)
}
# dann aus den Endjahren mit diesen Wahrscheinlichkeiten ziehen
# Endjahr für die Imputation ziehen mit der Funktion sample
imput <- as.list(sample(names(prob1), size = 20, replace=TRUE, prob = prob1))
# Im Datensatz imputieren, der die Daten aus SOLAR I und SOLAR II enthält
print("Endjahr das an dieser Stelle imputiert werden soll")
print(imput)
# Index der Zeile (index) in 1.Spalte des Datensatzes imput_werte8 speichern
imput_werte8$index[j] <- index
# Anfangsjahr (imput) in 2.Spalte des Datensatzes imput_werte8 speichern
imput_werte8$endjahr1[j] <- as.numeric(imput[[1]])
imput_werte8$endjahr2[j] <- as.numeric(imput[[2]])
imput_werte8$endjahr3[j] <- as.numeric(imput[[3]])
imput_werte8$endjahr4[j] <- as.numeric(imput[[4]])
imput_werte8$endjahr5[j] <- as.numeric(imput[[5]])
imput_werte8$endjahr6[j] <- as.numeric(imput[[6]])
imput_werte8$endjahr7[j] <- as.numeric(imput[[7]])
imput_werte8$endjahr8[j] <- as.numeric(imput[[8]])
imput_werte8$endjahr9[j] <- as.numeric(imput[[9]])
imput_werte8$endjahr10[j] <- as.numeric(imput[[10]])
imput_werte8$endjahr11[j] <- as.numeric(imput[[11]])
imput_werte8$endjahr12[j] <- as.numeric(imput[[12]])
imput_werte8$endjahr13[j] <- as.numeric(imput[[13]])
imput_werte8$endjahr14[j] <- as.numeric(imput[[14]])
imput_werte8$endjahr15[j] <- as.numeric(imput[[15]])
imput_werte8$endjahr16[j] <- as.numeric(imput[[16]])
imput_werte8$endjahr17[j] <- as.numeric(imput[[17]])
imput_werte8$endjahr18[j] <- as.numeric(imput[[18]])
imput_werte8$endjahr19[j] <- as.numeric(imput[[19]])
imput_werte8$endjahr20[j] <- as.numeric(imput[[20]])
print("Matrix,1.Spalte:Index der Zeile,2.Spalte:Zu imputierendes Endjahr")
print(imput_werte8)
# Jetzt den Wert im "großen" Datensatz imputieren
for(k in 1:nrow(datensatz)){
# Wenn der Index übereinstimmt
if(datensatz$index[k] == imput_werte8$index[j]){
# Zu imputierendes Endjahr in der Zeile mit diesem Index imputieren
if (!is.na(datensatz$ANF_MONAT[k])&!is.na(datensatz$END_MONATx[k])&
(datensatz$ANF_MONAT[k] > datensatz$END_MONATx[k])){
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr1[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr1[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
```

```
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr2[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr2[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr3[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr3[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr4[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr4[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr5[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr5[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr6[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr6[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr7[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr7[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr8[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr8[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr9[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr9[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr10[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr10[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr11[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr11[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr12[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr12[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
```

```
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr13[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr13[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr14[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr14[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr15[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr15[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr16[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr16[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr17[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr17[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr18[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr18[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr19[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr19[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] < imput_werte8$endjahr20[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr20[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}}}}}}}}}}}}}}}}}}}}
}
if (is.na(datensatz$ANF_MONAT[k]) | is.na(datensatz$END_MONATx[k]) |
!is.na(datensatz$ANF_MONAT[k])&!is.na(datensatz$END_MONATx[k])
& (datensatz$ANF_MONAT[k] <= datensatz$END_MONATx[k])){
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr1[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr1[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr2[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr2[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr3[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr3[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
```

```
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr4[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr4[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr5[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr5[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr6[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr6[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr7[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr7[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr8[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr8[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr9[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr9[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr10[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr10[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr11[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr11[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr12[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr12[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr13[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr13[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr14[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr14[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
```

```
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr15[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr15[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr16[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr16[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr17[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr17[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr18[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr18[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr19[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr19[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}
else{
if (datensatz$ANF_JAHR[k] <= imput_werte8$endjahr20[j]){
datensatz$END_JAHRx[k] <- as.numeric(imput_werte8$endjahr20[j])
print("Endjahr das imputiert wird")
print(datensatz$END_JAHRx[k])
# Indikator IMP_EJ auf 1 setzen, d.h. das Endjahr wurde imputiert
datensatz$IMP_EJ[k] <- 1
}}}}}}}}}}}}}}}}}}}}
}
j <- j+1
}

#############################################################################
####                  Anfangsmonat (ANF_MONAT) imputieren                  ####
#############################################################################

# Indikator, der angibt ob das ANF_MONAT imputiert wurde anlegen
datensatz$IMP_AM <- 0

############################### SOLAR I #######################################
# in SOLAR I nur bedingen auf SES_r
print("***************Imputation des Anfangsmonats in SOLAR I***************")
# Für solar 1:
a5 <- subset(datensatz_s1, is.na(ANF_MONAT) & ISCO != 8888 & ISCO != 9999
& ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
# hier nur die Zeilen drin in die imputiert werden soll

# Datensatz anlegen, in den jeweils index der Zeile und imputiertes ANF_MONAT
# geschrieben werden (1.Spalte: index, 2.Spalte: anfangsmonat)

imput_werte5 <- data.frame(index=numeric(nrow(a5)),
anfmonat1 = numeric(nrow(a5)), anfmonat2 = numeric(nrow(a5)),
anfmonat3 = numeric(nrow(a5)), anfmonat4 = numeric(nrow(a5)),
anfmonat5 = numeric(nrow(a5)), anfmonat6 = numeric(nrow(a5)),
anfmonat7 = numeric(nrow(a5)), anfmonat8 = numeric(nrow(a5)),
anfmonat9 = numeric(nrow(a5)), anfmonat10 = numeric(nrow(a5)),
anfmonat11 = numeric(nrow(a5)), anfmonat12 = numeric(nrow(a5)),
anfmonat13 = numeric(nrow(a5)), anfmonat14 = numeric(nrow(a5)),
anfmonat15 = numeric(nrow(a5)), anfmonat16 = numeric(nrow(a5)),
anfmonat17 = numeric(nrow(a5)), anfmonat18 = numeric(nrow(a5)),
anfmonat19 = numeric(nrow(a5)), anfmonat20 = numeric(nrow(a5))
)

# j auf 1 setzen
j <- 1
```

```
# Startwert setzen (der der Funktion als Argument übergeben wurde)
set.seed(startwert)

for (i in 1: nrow(a5)){
print("Nächste Stelle i")
# SES und Index an der i-ten Stelle aus subset betrachten
ses <- a5$SES_r[i]
print("SES an der Stelle i")
print(ses)
index <- a5$index[i]
print("Index an der Stelle i")
print(index)
# aus großen Datensatz alle mit gleichem SES ziehen, durch
# IMP_AM == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das ANF_MONAT bereits imputiert wurde
b <- subset(datensatz, SES_r==ses & !is.na(ANF_MONAT) & STUDIE == 1
 & IMP_AM==0 )
print("Anzahl der Fälle mit gleichem SES")
print(nrow(b))
# Hier gibt es auf jeden Fall noch andere Fälle mit gleichem SES ! D.h.
# nrow(b) > 0 immer !
table1 <- table(b$ANF_MONAT)
print("Table der Anfangsmonate bedingt auf SES_r")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für ANF_MONAT
# berechnen
print("Wahrscheinlichkeiten der Anfangsmonate bedingt auf SES_r")
print(prob1)
# dann aus den Anfangsmonaten mit diesen Wahrscheinlichkeiten ziehen
# Anfangsmonat für die Imputation ziehen mit der Funktion sample
imput <- as.list(sample(names(prob1), size = 20, replace=TRUE, prob = prob1))
# Im Datensatz imputieren, der die Daten aus SOLAR I und SOLAR II enthält
print("Anfangsmonat das an dieser Stelle imputiert werden soll")
print(imput)
# Index der Zeile (index) in 1.Spalte des Datensatzes imput_werte5 speichern
imput_werte5$index[j] <- index
# Anfangsmonat (imput) in 2.Spalte des Datensatzes imput_werte5 speichern
imput_werte5$anfmonat1[j] <- as.numeric(imput[[1]])
imput_werte5$anfmonat2[j] <- as.numeric(imput[[2]])
imput_werte5$anfmonat3[j] <- as.numeric(imput[[3]])
imput_werte5$anfmonat4[j] <- as.numeric(imput[[4]])
imput_werte5$anfmonat5[j] <- as.numeric(imput[[5]])
imput_werte5$anfmonat6[j] <- as.numeric(imput[[6]])
imput_werte5$anfmonat7[j] <- as.numeric(imput[[7]])
imput_werte5$anfmonat8[j] <- as.numeric(imput[[8]])
imput_werte5$anfmonat9[j] <- as.numeric(imput[[9]])
imput_werte5$anfmonat10[j] <- as.numeric(imput[[10]])
imput_werte5$anfmonat11[j] <- as.numeric(imput[[11]])
imput_werte5$anfmonat12[j] <- as.numeric(imput[[12]])
imput_werte5$anfmonat13[j] <- as.numeric(imput[[13]])
imput_werte5$anfmonat14[j] <- as.numeric(imput[[14]])
imput_werte5$anfmonat15[j] <- as.numeric(imput[[15]])
imput_werte5$anfmonat16[j] <- as.numeric(imput[[16]])
imput_werte5$anfmonat17[j] <- as.numeric(imput[[17]])
imput_werte5$anfmonat18[j] <- as.numeric(imput[[18]])
imput_werte5$anfmonat19[j] <- as.numeric(imput[[19]])
imput_werte5$anfmonat20[j] <- as.numeric(imput[[20]])
print("Matrix,1.Spalte:Index der Zeile,2.Spalte:Zu imputierendes Anfangsmonat")
print(imput_werte5)
# Jetzt den Wert im "großen" Datensatz imputieren
for(k in 1:nrow(datensatz)){
# Wenn der Index übereinstimmt
if(datensatz$index[k] == imput_werte5$index[j]){
# Zu imputierendes Anfangsmonat in der Zeile mit diesem Index imputieren
if (datensatz$ANF_JAHR[k] != datensatz$END_JAHRx[k] |
(datensatz$ANF_JAHR[k] == datensatz$END_JAHRx[k]
& is.na(datensatz$END_MONATx[k]))){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat1[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
if (datensatz$ANF_JAHR[k] == datensatz$END_JAHRx[k]){
if (imput_werte5$anfmonat1[j] <= datensatz$END_MONATx[k]
| is.na(datensatz$END_MONATx[k])){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat1[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat2[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat2[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
```

```
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat3[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat3[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat4[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat4[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat5[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat5[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat6[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat6[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat7[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat7[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat8[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat8[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat9[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat9[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat10[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat10[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat11[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat11[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat12[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat12[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat13[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat13[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
```

```
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat14[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat14[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat15[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat15[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat16[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat16[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat17[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat17[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat18[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat18[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat19[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat19[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte5$anfmonat20[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte5$anfmonat20[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}}}}}}}}}}}}}}}}}}}}
}
j <- j+1
}

################################ SOLAR II #####################################
# in SOLAR II bedingen auf SES_r und s2BERUF
print("***************Imputation des Anfangsmonats in SOLAR II***************")
# Für solar 2:
a6 <- subset(datensatz_s2, is.na(ANF_MONAT) & ISCO != 8888 & ISCO != 9999
& ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
# hier nur die Zeilen drin in die imputiert werden soll

# Datensatz anlegen, in den jeweils index der Zeile und imputiertes ANF_MONAT
# geschrieben werden (1.Spalte: index, 2.Spalte: anfangsmonat)

imput_werte6 <- data.frame(index=numeric(nrow(a6)),
anfmonat1 = numeric(nrow(a6)), anfmonat2 = numeric(nrow(a6)),
anfmonat3 = numeric(nrow(a6)), anfmonat4 = numeric(nrow(a6)),
anfmonat5 = numeric(nrow(a6)), anfmonat6 = numeric(nrow(a6)),
anfmonat7 = numeric(nrow(a6)), anfmonat8 = numeric(nrow(a6)),
anfmonat9 = numeric(nrow(a6)), anfmonat10 = numeric(nrow(a6)),
anfmonat11 = numeric(nrow(a6)), anfmonat12 = numeric(nrow(a6)),
anfmonat13 = numeric(nrow(a6)), anfmonat14 = numeric(nrow(a6)),
anfmonat15 = numeric(nrow(a6)), anfmonat16 = numeric(nrow(a6)),
anfmonat17 = numeric(nrow(a6)), anfmonat18 = numeric(nrow(a6)),
anfmonat19 = numeric(nrow(a6)), anfmonat20 = numeric(nrow(a6))
)

# j auf 1 setzen
j <- 1
```

```
# Startwert setzen (der der Funktion als Argument übergeben wurde)
set.seed(startwert)

for (i in 1: nrow(a6)){
print("Nächste Stelle i")
# SES und Index und Beruf an der i-ten Stelle aus subset betrachten
beruf <- a6$s2BERUF[i]
print("s2BERUF an der Stelle i")
print(beruf)
ses <- a6$SES_r[i]
print("SES an der Stelle i")
print(ses)
index <- a6$index[i]
print("Index an der Stelle i")
print(index)
# aus großen Datensatz alle mit gleichem SES und s2Beruf ziehen, durch
# IMP_AM == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das ANF_MONAT bereits imputiert wurde
b <- subset(datensatz, SES_r==ses & s2BERUF == beruf & !is.na(ANF_MONAT) & STUDIE == 2
 & IMP_AM==0 )
print("Anzahl der Fälle mit gleichem SES und s2BERUF")
print(nrow(b))

if(nrow(b)>0){ # Wenn es noch andere Fälle mit gleichem s2BERUF und SES_r gibt
print("b groesser als 0 also auf s2BERUF und SES_r bedingen")
table1 <- table(b$ANF_MONAT)
print("Table der Anfangsmonate bedingt auf SES_r und s2BERUF")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für ANF_MONAT
# berechnen
print("Wahrscheinlichkeiten der Anfangsmonate bedingt auf SES_r und s2BERUF")
print(prob1)
}
if(nrow(b)==0 | (nrow(b)==1 & b$END_MONATx[1] < a6$ANF_MONAT[i]) |
(nrow(b)==2 & b$END_MONATx[1] < a6$ANF_MONAT[i]
& b$END_MONATx[2] < a6$ANF_MONAT[i])| (nrow(b)==3
& b$END_MONATx[1] < a6$ANF_MONAT[i] & b$END_MONATx[2] < a6$ANF_MONAT[i]
& b$END_MONATx[3] < a6$ANF_MONAT[i])){
# Wenn es sonst keinen Fall mit gleichem s2BERUF und
# gleichem SES_r gibt bei dem das ANF_MONAT fehlt (d.h. ANF_MONAT fehlt nur in
# diesem einen Fall mit diesem s2BERUF und diesem SES_r)
# oder es gibt nur 1(2/3) fälle und bei denen wäre dann das Anfangsmonat >
# Endmonat
print("b gleich 0 also nur auf SES_r bedingen")
# aus großem Datensatz alle mit gleichem SES_r ziehen, durch
# IMP_AM == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das ANF_MONAT bereits imputiert wurde
c <- subset(datensatz, SES_r==ses & !is.na(ANF_MONAT) & STUDIE == 2
& IMP_AM==0 )
table1 <- table(c$ANF_MONAT)
print("Table der Anfangsmonate bedingt auf SES_r")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für ANF_MONAT
# berechnen
print("Wahrscheinlichkeiten der Anfangsmonate bedingt auf SES_r")
print(prob1)
}
# dann aus den Anfangsmonaten mit diesen Wahrscheinlichkeiten ziehen
# Anfangsmonat für die Imputation ziehen mit der Funktion sample
imput <- as.list(sample(names(prob1), size = 20, replace=TRUE, prob = prob1))
# Im Datensatz imputieren, der die Daten aus SOLAR I und SOLAR II enthält
print("Anfangsmonat das an dieser Stelle imputiert werden soll")
print(imput)
# Index der Zeile (index) in 1.Spalte des Datensatzes imput_werte6 speichern
imput_werte6$index[j] <- index
# Anfangsmonat (imput) in 2.Spalte des Datensatzes imput_werte6 speichern
imput_werte6$anfmonat1[j] <- as.numeric(imput[[1]])
imput_werte6$anfmonat2[j] <- as.numeric(imput[[2]])
imput_werte6$anfmonat3[j] <- as.numeric(imput[[3]])
imput_werte6$anfmonat4[j] <- as.numeric(imput[[4]])
imput_werte6$anfmonat5[j] <- as.numeric(imput[[5]])
imput_werte6$anfmonat6[j] <- as.numeric(imput[[6]])
imput_werte6$anfmonat7[j] <- as.numeric(imput[[7]])
imput_werte6$anfmonat8[j] <- as.numeric(imput[[8]])
imput_werte6$anfmonat9[j] <- as.numeric(imput[[9]])
imput_werte6$anfmonat10[j] <- as.numeric(imput[[10]])
imput_werte6$anfmonat11[j] <- as.numeric(imput[[11]])
imput_werte6$anfmonat12[j] <- as.numeric(imput[[12]])
imput_werte6$anfmonat13[j] <- as.numeric(imput[[13]])
imput_werte6$anfmonat14[j] <- as.numeric(imput[[14]])
imput_werte6$anfmonat15[j] <- as.numeric(imput[[15]])
imput_werte6$anfmonat16[j] <- as.numeric(imput[[16]])
imput_werte6$anfmonat17[j] <- as.numeric(imput[[17]])
imput_werte6$anfmonat18[j] <- as.numeric(imput[[18]])
imput_werte6$anfmonat19[j] <- as.numeric(imput[[19]])
imput_werte6$anfmonat20[j] <- as.numeric(imput[[20]])
print("Matrix,1.Spalte:Index der Zeile,2.Spalte:Zu imputierendes Anfangsmonat")
```

```
print(imput_werte6)
# Jetzt den Wert im "großen" Datensatz imputieren
for(k in 1:nrow(datensatz)){
# Wenn der Index übereinstimmt
if(datensatz$index[k] == imput_werte6$index[j]){
# Zu imputierendes Anfangsmonat in der Zeile mit diesem Index imputieren
if (datensatz$ANF_JAHR[k] != datensatz$END_JAHRx[k] |
(datensatz$ANF_JAHR[k] == datensatz$END_JAHRx[k]
& is.na(datensatz$END_MONATx[k]))){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat1[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
if (datensatz$ANF_JAHR[k] == datensatz$END_JAHRx[k]){
if (imput_werte6$anfmonat1[j] <= datensatz$END_MONATx[k]
| is.na(datensatz$END_MONATx[k])){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat1[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat2[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat2[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat3[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat3[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat4[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat4[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat5[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat5[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat6[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat6[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat7[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat7[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat8[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat8[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat9[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat9[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
```

```
else{
if (imput_werte6$anfmonat10[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat10[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat11[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat11[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat12[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat12[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat13[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat13[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat14[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat14[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat15[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat15[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat16[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat16[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat17[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat17[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat18[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat18[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat19[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat19[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}
else{
if (imput_werte6$anfmonat20[j] <= datensatz$END_MONATx[k]){
datensatz$ANF_MONAT[k] <- as.numeric(imput_werte6$anfmonat20[j])
print("Anfangsmonat das imputiert wird")
print(datensatz$ANF_MONAT[k])
# Indikator IMP_AM auf 1 setzen, d.h. das Anfangsmonat wurde imputiert
datensatz$IMP_AM[k] <- 1
}}}}}}}}}}}}}}}}}}}}
```

```
}
j <- j+1
}


################################################################################
####                  Endmonat (END_MONATx) imputieren                      ####
################################################################################

# Indikator, der angibt ob das END_MONATx imputiert wurde anlegen
datensatz$IMP_EM <- 0

############################### SOLAR I ########################################
# in SOLAR I nur bedingen auf SES_r
print("***************Imputation des Endmonats in SOLAR I***************")
# Für solar 1:
a9 <- subset(datensatz_s1, is.na(END_MONATx) & ISCO != 8888 & ISCO != 9999
& ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
# hier nur die Zeilen drin in die imputiert werden soll

# Datensatz anlegen, in den jeweils index der Zeile und imputiertes END_JAHRx
# geschrieben werden (1.Spalte: index, 2.Spalte: endmonat)

imput_werte9 <- data.frame(index=numeric(nrow(a9)),
endmonat1 = numeric(nrow(a9)), endmonat2 = numeric(nrow(a9)),
endmonat3 = numeric(nrow(a9)), endmonat4 = numeric(nrow(a9)),
endmonat5 = numeric(nrow(a9)), endmonat6 = numeric(nrow(a9)),
endmonat7 = numeric(nrow(a9)), endmonat8 = numeric(nrow(a9)),
endmonat9 = numeric(nrow(a9)), endmonat10 = numeric(nrow(a9)),
endmonat11 = numeric(nrow(a9)), endmonat12 = numeric(nrow(a9)),
endmonat13 = numeric(nrow(a9)), endmonat14 = numeric(nrow(a9)),
endmonat15 = numeric(nrow(a9)), endmonat16 = numeric(nrow(a9)),
endmonat17 = numeric(nrow(a9)), endmonat18 = numeric(nrow(a9)),
endmonat19 = numeric(nrow(a9)), endmonat20 = numeric(nrow(a9))
)

# j auf 1 setzen
j <- 1

# Startwert setzen (der der Funktion als Argument übergeben wurde)
set.seed(startwert)

for (i in 1: nrow(a9)){
print("Nächste Stelle i")
# SES und Index an der i-ten Stelle aus subset betrachten
ses <- a9$SES_r[i]
print("SES an der Stelle i")
print(ses)
index <- a9$index[i]
print("Index an der Stelle i")
print(index)
# aus großen Datensatz alle mit gleichem SES ziehen, durch
# IMP_EM == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das END_MONATx bereits imputiert wurde
b <- subset(datensatz, SES_r==ses & !is.na(END_MONATx) & STUDIE == 1
 & IMP_EM==0 )
print("Anzahl der Fälle mit gleichem SES")
print(nrow(b))
# Hier gibt es auf jeden Fall noch andere Fälle mit gleichem SES ! D.h.
# nrow(b) > 0 immer !
table1 <- table(b$END_MONATx)
print("Table der Endmonate bedingt auf SES_r")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für END_MONATx
# berechnen
print("Wahrscheinlichkeiten der Endmonate bedingt auf SES_r")
print(prob1)
# dann aus den Endmonaten mit diesen Wahrscheinlichkeiten ziehen
# Endmonat für die Imputation ziehen mit der Funktion sample
imput <- as.list(sample(names(prob1), size = 20, replace=TRUE, prob = prob1))
# Im Datensatz imputieren, der die Daten aus SOLAR I und SOLAR II enthält
print("Endmonat das an dieser Stelle imputiert werden soll")
print(imput)
# Index der Zeile (index) in 1.Spalte des Datensatzes imput_werte7 speichern
imput_werte9$index[j] <- index
# Anfangsjahr (imput) in 2.Spalte des Datensatzes imput_werte7 speichern
imput_werte9$endmonat1[j] <- as.numeric(imput[[1]])
imput_werte9$endmonat2[j] <- as.numeric(imput[[2]])
imput_werte9$endmonat3[j] <- as.numeric(imput[[3]])
imput_werte9$endmonat4[j] <- as.numeric(imput[[4]])
imput_werte9$endmonat5[j] <- as.numeric(imput[[5]])
imput_werte9$endmonat6[j] <- as.numeric(imput[[6]])
imput_werte9$endmonat7[j] <- as.numeric(imput[[7]])
imput_werte9$endmonat8[j] <- as.numeric(imput[[8]])
imput_werte9$endmonat9[j] <- as.numeric(imput[[9]])
imput_werte9$endmonat10[j] <- as.numeric(imput[[10]])
imput_werte9$endmonat11[j] <- as.numeric(imput[[11]])
imput_werte9$endmonat12[j] <- as.numeric(imput[[12]])
```

```
imput_werte9$endmonat13[j] <- as.numeric(imput[[13]])
imput_werte9$endmonat14[j] <- as.numeric(imput[[14]])
imput_werte9$endmonat15[j] <- as.numeric(imput[[15]])
imput_werte9$endmonat16[j] <- as.numeric(imput[[16]])
imput_werte9$endmonat17[j] <- as.numeric(imput[[17]])
imput_werte9$endmonat18[j] <- as.numeric(imput[[18]])
imput_werte9$endmonat19[j] <- as.numeric(imput[[19]])
imput_werte9$endmonat20[j] <- as.numeric(imput[[20]])
print("Matrix,1.Spalte:Index der Zeile,2.Spalte:Zu imputierendes Endmonat")
print(imput_werte9)
# Jetzt den Wert im "großen" Datensatz imputieren
for(k in 1:nrow(datensatz)){
# Wenn der Index übereinstimmt
if(datensatz$index[k] == imput_werte9$index[j]){
# Zu imputierendes Endmonat in der Zeile mit diesem Index imputieren
if (datensatz$ANF_JAHR[k] != datensatz$END_JAHRx[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat1[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
if (datensatz$ANF_JAHR[k] == datensatz$END_JAHRx[k]){
if (imput_werte9$endmonat1[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat1[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat2[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat2[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat3[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat3[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat4[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat4[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat5[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat5[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat6[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat6[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat7[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat7[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat8[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat8[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat9[j] >= datensatz$ANF_MONAT[k]){
```

```
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat9[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat10[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat10[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat11[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat11[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat12[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat12[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat13[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat13[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat14[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat14[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat15[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat15[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat16[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat16[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat17[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat17[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat18[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat18[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat19[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat19[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte9$endmonat20[j] >= datensatz$ANF_MONAT[k]){
```

```
datensatz$END_MONATx[k] <- as.numeric(imput_werte9$endmonat20[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}}}}}}}}}}}}}}}}}}}}}
}
j <- j+1
}

############################## SOLAR II ######################################
# in SOLAR II bedingen auf SES_r und s2BERUF
print("***************Imputation des Endmonats in SOLAR II***************")
# Für solar 2:
a10 <- subset(datensatz_s2, is.na(END_MONATx) & ISCO != 8888 & ISCO != 9999
& ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
# hier nur die Zeilen drin in die imputiert werden soll

# Datensatz anlegen, in den jeweils index der Zeile und imputiertes END_MONATx
# geschrieben werden (1.Spalte: index, 2.Spalte: wochenstunden)

imput_werte10 <- data.frame(index=numeric(nrow(a10)),
endmonat1 = numeric(nrow(a10)), endmonat2 = numeric(nrow(a10)),
endmonat3 = numeric(nrow(a10)), endmonat4 = numeric(nrow(a10)),
endmonat5 = numeric(nrow(a10)), endmonat6 = numeric(nrow(a10)),
endmonat7 = numeric(nrow(a10)), endmonat8 = numeric(nrow(a10)),
endmonat9 = numeric(nrow(a10)), endmonat10 = numeric(nrow(a10)),
endmonat11 = numeric(nrow(a10)), endmonat12 = numeric(nrow(a10)),
endmonat13 = numeric(nrow(a10)), endmonat14 = numeric(nrow(a10)),
endmonat15 = numeric(nrow(a10)), endmonat16 = numeric(nrow(a10)),
endmonat17 = numeric(nrow(a10)), endmonat18 = numeric(nrow(a10)),
endmonat19 = numeric(nrow(a10)), endmonat20 = numeric(nrow(a10))
)

# j auf 1 setzen
j <- 1

# Startwert setzen (der der Funktion als Argument übergeben wurde)
set.seed(startwert)

for (i in 1: nrow(a10)){
print("Nächste Stelle i")
# SES, s2BERUF und Index an der i-ten Stelle aus subset betrachten
beruf <- a10$s2BERUF[i]
print("s2BERUF an der Stelle i")
print(beruf)
ses <- a10$SES_r[i]
print("SES an der Stelle i")
print(ses)
index <- a10$index[i]
print("Index an der Stelle i")
print(index)
# aus großen Datensatz alle mit gleichem SES und s2BERUF ziehen, durch
# IMP_EM == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das END_MONATx bereits imputiert wurde
b <- subset(datensatz, SES_r==ses & s2BERUF == beruf & !is.na(END_MONATx) & STUDIE == 2
 & IMP_EM==0 )
print("Anzahl der Fälle mit gleichem SES")
print(nrow(b))


if(nrow(b)>0){ # Wenn es noch andere Fälle mit gleichem s2BERUF und SES_r gibt
print("b groesser als 0 also auf s2BERUF und SES_r bedingen")
table1 <- table(b$END_MONATx)
print("Table der Endmonate bedingt auf SES_r und s2BERUF")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für END_MONATx
# berechnen
print("Wahrscheinlichkeiten der Endmonate bedingt auf SES_r und s2BERUF")
print(prob1)
}
if(nrow(b)==0 | (nrow(b)==1 & b$END_MONATx[1] < a10$ANF_MONAT[i]) |
(nrow(b)==2 & b$END_MONATx[1] < a10$ANF_MONAT[i]
& b$END_MONATx[2] < a10$ANF_MONAT[i])| (nrow(b)==3
& b$END_MONATx[1] < a10$ANF_MONAT[i] & b$END_MONATx[2] < a10$ANF_MONAT[i]
& b$END_MONATx[3] < a10$ANF_MONAT[i])){
# Wenn es sonst keinen Fall mit gleichem s2BERUF und
# gleichem SES_r gibt bei dem das END_MONATx fehlt (d.h. END_MONATx fehlt nur in
# diesem einen Fall mit diesem s2BERUF und diesem SES_r)
# oder es gibt nur 1(2/3) fälle und bei denen wäre dann das Anfangsmonat >
# Endmonat
print("b gleich 0 also nur auf SES_r bedingen")
# aus großem Datensatz alle mit gleichem SES_r ziehen, durch
# IMP_AM == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das ANF_MONAT bereits imputiert wurde
c <- subset(datensatz, SES_r==ses & !is.na(END_MONATx) & STUDIE == 2
& IMP_EM==0 )
table1 <- table(c$END_MONATx)
```

```
print("Table der Endmonate bedingt auf SES_r")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für END_MONATx
# berechnen
print("Wahrscheinlichkeiten der Endmonate bedingt auf Geschlecht")
print(prob1)
}

# dann aus den Endmonaten mit diesen Wahrscheinlichkeiten ziehen
# Endmonate für die Imputation ziehen mit der Funktion sample
imput <- as.list(sample(names(prob1), size = 20, replace=TRUE, prob = prob1))
# Im Datensatz imputieren, der die Daten aus SOLAR I und SOLAR II enthält
print("Endmonat das an dieser Stelle imputiert werden soll")
print(imput)
# Index der Zeile (index) in 1.Spalte des Datensatzes imput_werte10 speichern
imput_werte10$index[j] <- index
# Anfangsjahr (imput) in 2.Spalte des Datensatzes imput_werte10 speichern
imput_werte10$endmonat1[j] <- as.numeric(imput[[1]])
imput_werte10$endmonat2[j] <- as.numeric(imput[[2]])
imput_werte10$endmonat3[j] <- as.numeric(imput[[3]])
imput_werte10$endmonat4[j] <- as.numeric(imput[[4]])
imput_werte10$endmonat5[j] <- as.numeric(imput[[5]])
imput_werte10$endmonat6[j] <- as.numeric(imput[[6]])
imput_werte10$endmonat7[j] <- as.numeric(imput[[7]])
imput_werte10$endmonat8[j] <- as.numeric(imput[[8]])
imput_werte10$endmonat9[j] <- as.numeric(imput[[9]])
imput_werte10$endmonat10[j] <- as.numeric(imput[[10]])
imput_werte10$endmonat11[j] <- as.numeric(imput[[11]])
imput_werte10$endmonat12[j] <- as.numeric(imput[[12]])
imput_werte10$endmonat13[j] <- as.numeric(imput[[13]])
imput_werte10$endmonat14[j] <- as.numeric(imput[[14]])
imput_werte10$endmonat15[j] <- as.numeric(imput[[15]])
imput_werte10$endmonat16[j] <- as.numeric(imput[[16]])
imput_werte10$endmonat17[j] <- as.numeric(imput[[17]])
imput_werte10$endmonat18[j] <- as.numeric(imput[[18]])
imput_werte10$endmonat19[j] <- as.numeric(imput[[19]])
imput_werte10$endmonat20[j] <- as.numeric(imput[[20]])
print("Matrix,1.Spalte:Index der Zeile,2.Spalte:Zu imputierendes Endmonat")
print(imput_werte10)
# Jetzt den Wert im "großen" Datensatz imputieren
for(k in 1:nrow(datensatz)){
# Wenn der Index übereinstimmt
if(datensatz$index[k] == imput_werte10$index[j]){
# Zu imputierendes Endmonat in der Zeile mit diesem Index imputieren
if (datensatz$ANF_JAHR[k] != datensatz$END_JAHRx[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat1[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
if (datensatz$ANF_JAHR[k] == datensatz$END_JAHRx[k]){
if (imput_werte10$endmonat1[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat1[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat2[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat2[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat3[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat3[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat4[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat4[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat5[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat5[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
```

```
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat6[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat6[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat7[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat7[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat8[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat8[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat9[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat9[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat10[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat10[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat11[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat11[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat12[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat12[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat13[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat13[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat14[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat14[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat15[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat15[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat16[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat16[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
```

```
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat17[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat17[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat18[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat18[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat19[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat19[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte10$endmonat20[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte10$endmonat20[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}}}}}}}}}}}}}}}}}}}}
}
j <- j+1
}

# Schauen wo der END_MONATx noch fehlt:
endmonatfehlt <- subset(datensatz, is.na(END_MONATx) & ISCO != 8888 & ISCO != 9999
& ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
# Für diesen Fall nochmal aus der Verteilung nur bedingt auf SES_r ziehen
# da dann ein Anfangsmonat gezogen wurde für den es keine Endmonat >= Anf.monat
# gibt

############################### SOLAR II #####################################
# hier nur bedingen auf SES_r
print("***************Imputation des Endmonats in SOLAR II (2)***************")
# Für solar 2:
a11 <- subset(datensatz, STUDIE==2 & is.na(END_MONATx) & ISCO != 8888 & ISCO != 9999
& ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
# hier nur die Zeilen drin in die imputiert werden soll

# Datensatz anlegen, in den jeweils index der Zeile und imputiertes END_JAHRx
# geschrieben werden (1.Spalte: index, 2.Spalte: wochenstunden)
if(nrow(a11)>0){
imput_werte11 <- data.frame(index=numeric(nrow(a11)),
endmonat1 = numeric(nrow(a11)), endmonat2 = numeric(nrow(a11)),
endmonat3 = numeric(nrow(a11)), endmonat4 = numeric(nrow(a11)),
endmonat5 = numeric(nrow(a11)), endmonat6 = numeric(nrow(a11)),
endmonat7 = numeric(nrow(a11)), endmonat8 = numeric(nrow(a11)),
endmonat9 = numeric(nrow(a11)), endmonat10 = numeric(nrow(a11)),
endmonat11 = numeric(nrow(a11)), endmonat12 = numeric(nrow(a11)),
endmonat13 = numeric(nrow(a11)), endmonat14 = numeric(nrow(a11)),
endmonat15 = numeric(nrow(a11)), endmonat16 = numeric(nrow(a11)),
endmonat17 = numeric(nrow(a11)), endmonat18 = numeric(nrow(a11)),
endmonat19 = numeric(nrow(a11)), endmonat20 = numeric(nrow(a11))
)

# j auf 1 setzen
j <- 1

# Startwert setzen (der der Funktion als Argument übergeben wurde)
set.seed(startwert)

for (i in 1: nrow(a11)){
print("Nächste Stelle i")
# SES und Index an der i-ten Stelle aus subset betrachten
ses <- a11$SES_r[i]
print("SES an der Stelle i")
print(ses)
index <- a11$index[i]
print("Index an der Stelle i")
print(index)
# aus großen Datensatz alle mit gleichem SES ziehen, durch
# IMP_EM == 0 werden diejenigen Fälle NICHT zur Berechnung der W.keiten
# verwendet, bei denen das END_MONATx bereits imputiert wurde
```

```
b <- subset(datensatz, SES_r==ses & !is.na(END_MONATx) & STUDIE == 2
 & IMP_EM==0 )
print("Anzahl der Fälle mit gleichem SES")
print(nrow(b))

table1 <- table(b$END_MONATx)
print("Table der Endmonate bedingt auf SES_r")
print(table1)
prob1 <- prop.table(as.array(table1))  # Wahrscheinlichkeiten für END_MONATx
# berechnen
print("Wahrscheinlichkeiten der Endmonate bedingt auf SES_r")
print(prob1)

# dann aus den Endmonaten mit diesen Wahrscheinlichkeiten ziehen
# Endmonat für die Imputation ziehen mit der Funktion sample
imput <- as.list(sample(names(prob1), size = 20, replace=TRUE, prob = prob1))
# Im Datensatz imputieren, der die Daten aus SOLAR I und SOLAR II enthält
print("Endmonat das an dieser Stelle imputiert werden soll")
print(imput)
# Index der Zeile (index) in 1.Spalte des Datensatzes imput_werte11 speichern
imput_werte11$index[j] <- index
# Anfangsjahr (imput) in 2.Spalte des Datensatzes imput_werte11 speichern
imput_werte11$endmonat1[j] <- as.numeric(imput[[1]])
imput_werte11$endmonat2[j] <- as.numeric(imput[[2]])
imput_werte11$endmonat3[j] <- as.numeric(imput[[3]])
imput_werte11$endmonat4[j] <- as.numeric(imput[[4]])
imput_werte11$endmonat5[j] <- as.numeric(imput[[5]])
imput_werte11$endmonat6[j] <- as.numeric(imput[[6]])
imput_werte11$endmonat7[j] <- as.numeric(imput[[7]])
imput_werte11$endmonat8[j] <- as.numeric(imput[[8]])
imput_werte11$endmonat9[j] <- as.numeric(imput[[9]])
imput_werte11$endmonat10[j] <- as.numeric(imput[[10]])
imput_werte11$endmonat11[j] <- as.numeric(imput[[11]])
imput_werte11$endmonat12[j] <- as.numeric(imput[[12]])
imput_werte11$endmonat13[j] <- as.numeric(imput[[13]])
imput_werte11$endmonat14[j] <- as.numeric(imput[[14]])
imput_werte11$endmonat15[j] <- as.numeric(imput[[15]])
imput_werte11$endmonat16[j] <- as.numeric(imput[[16]])
imput_werte11$endmonat17[j] <- as.numeric(imput[[17]])
imput_werte11$endmonat18[j] <- as.numeric(imput[[18]])
imput_werte11$endmonat19[j] <- as.numeric(imput[[19]])
imput_werte11$endmonat20[j] <- as.numeric(imput[[20]])
print("Matrix,1.Spalte:Index der Zeile,2.Spalte:Zu imputierendes Endmonat")
print(imput_werte11)
# Jetzt den Wert im "großen" Datensatz imputieren
for(k in 1:nrow(datensatz)){
# Wenn der Index übereinstimmt
if(datensatz$index[k] == imput_werte11$index[j]){
# Zu imputierendes Endmonat in der Zeile mit diesem Index imputieren
if (datensatz$ANF_JAHR[k] != datensatz$END_JAHRx[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat1[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
if (datensatz$ANF_JAHR[k] == datensatz$END_JAHRx[k]){
if (imput_werte11$endmonat1[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat1[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat2[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat2[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat3[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat3[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat4[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat4[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
```

```
else{
if (imput_werte11$endmonat5[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat5[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat6[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat6[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat7[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat7[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat8[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat8[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat9[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat9[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat10[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat10[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat11[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat11[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat12[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat12[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat13[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat13[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat14[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat14[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat15[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat15[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
```

```
else{
if (imput_werte11$endmonat16[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat16[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat17[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat17[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat18[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat18[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat19[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat19[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}
else{
if (imput_werte11$endmonat20[j] >= datensatz$ANF_MONAT[k]){
datensatz$END_MONATx[k] <- as.numeric(imput_werte11$endmonat20[j])
print("Endmonat das imputiert wird")
print(datensatz$END_MONATx[k])
# Indikator IMP_EM auf 1 setzen, d.h. das Endmonat wurde imputiert
datensatz$IMP_EM[k] <- 1
}}}}}}}}}}}}}}}}}}}}
}
j <- j+1
}
}

#############################################################################
#################  EXPOSITION imputieren ####################################
#############################################################################

#datensatz$IMP_EXPO <- 0

################################ SOLAR-I ###################################
#print("***************Imputation der Exposition in SOLAR I***************")

# Daten rausziehen, bei denen imputiert werden muss,nur
# SOLAR I (STUDIE=1) und ISCO fehlt (da Expo nur fehlt wenn ISCO fehlt)
# EXPO_s1 <-subset(datensatz_s1, is.na(ISCO))
# hier nur die Zeilen drin in die imputiert werden soll

#if(nrow(EXPO_s1)>0){
#j <- 1
#set.seed(startwert)
#for(i in 1:nrow(EXPO_s1)){
#index <- EXPO_s1$index[i]
#print("Index an der Stelle i")
#print(index)
#expo1 <- subset(datensatz_s1, select=c("anim", "fish", "flour", "plants", "mites", "enzymes", "latex", "bioaero",
#"drugs", "react", "isocy", "clean", "wood", "metals", "mwf", "textile", "agric",
#"irrpeaks", "exhaust", "ets", "pos_irr", "low_anti"), ISCO != 8888 & ISCO != 9999 &
#ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
#expo1$muster <- ""

#for (i in 1:nrow(expo1)){
#expo1$muster[i] <- paste(expo1[i,],sep="",collapse="")
#}

#table_EXPO_s1 <- table(expo1$muster)
#table_EXPO_s1
#print("Table der Expositionsmuster")
#print(table_EXPO_s1)
#prop_EXPO_s1 <- prop.table(as.array(table_EXPO_s1))
#prop_EXPO_s1
#print("Table der Wahrscheinlichkeiten der Expositionsmuster")
#print(prop_EXPO_s1)

#EXPO_s1_imput <- sample(names(prop_EXPO_s1), size = nrow(EXPO_s1),
#replace=TRUE, prob = prop_EXPO_s1)
#table(EXPO_s1_imput)
```

```
#print("Expositionsmuster das imputiert wird")
#print(EXPO_s1_imput)

# Exposition im Datensatz ersetzen
#j <- 1
#for (i in 1:nrow(datensatz)){
#  if(is.na(datensatz$ISCO)[i] & datensatz$STUDIE[i] == 1){
#    datensatz$anim[i] <- as.numeric(substring(EXPO_s1_imput,1,1))
#    datensatz$fish[i] <- as.numeric(substring(EXPO_s1_imput,2,2))
#    datensatz$flour[i] <- as.numeric(substring(EXPO_s1_imput,3,3))
#    datensatz$plants[i] <- as.numeric(substring(EXPO_s1_imput,4,4))
#    datensatz$mites[i] <- as.numeric(substring(EXPO_s1_imput,5,5))
#    datensatz$enzymes[i] <- as.numeric(substring(EXPO_s1_imput,6,6))
#    datensatz$latex[i] <- as.numeric(substring(EXPO_s1_imput,7,7))
#    datensatz$bioaero[i] <- as.numeric(substring(EXPO_s1_imput,8,8))
#    datensatz$drugs[i] <- as.numeric(substring(EXPO_s1_imput,9,9))
#    datensatz$react[i] <- as.numeric(substring(EXPO_s1_imput,10,10))
#    datensatz$isocy[i] <- as.numeric(substring(EXPO_s1_imput,11,11))
#    datensatz$clean[i] <- as.numeric(substring(EXPO_s1_imput,12,12))
#    datensatz$wood[i] <- as.numeric(substring(EXPO_s1_imput,13,13))
#    datensatz$metals[i] <- as.numeric(substring(EXPO_s1_imput,14,14))
#    datensatz$mwf[i] <- as.numeric(substring(EXPO_s1_imput,15,15))
#    datensatz$textile[i] <- as.numeric(substring(EXPO_s1_imput,16,16))
#    datensatz$agric[i] <- as.numeric(substring(EXPO_s1_imput,17,17))
#    datensatz$irrpeaks[i] <- as.numeric(substring(EXPO_s1_imput,18,18))
#    datensatz$exhaust[i] <- as.numeric(substring(EXPO_s1_imput,19,19))
#    datensatz$ets[i] <- as.numeric(substring(EXPO_s1_imput,20,20))
#    datensatz$pos_irr[i] <- as.numeric(substring(EXPO_s1_imput,21,21))
#    datensatz$low_anti[i] <- as.numeric(substring(EXPO_s1_imput,22,22))
#    datensatz$IMP_EXPO[i] <- 1
#    j <- j+1
#  }
#}
#}
#}


################################ SOLAR-II ###################################
#print("**************Imputation der Exposition in SOLAR II**************")
# Daten rausziehen, bei denen imputiert werden muss, nur
# SOLAR II (STUDIE=2) und ISCO fehlt (da Expo nur fehlt wenn ISCO fehlt)
#EXPO_s2 <-subset(datensatz_s2, is.na(ISCO))
# hier nur die Zeilen drin in die imputiert werden soll
#if(nrow(EXPO_s2)>0){
#j <- 1
#set.seed(startwert)
#for(i in 1:nrow(EXPO_s2)){
#index <- EXPO_s2$index[i]
#print("Index an der Stelle i")
#print(index)
#expo2 <- subset(datensatz_s2, select=c("anim", "fish", "flour", "plants", "mites", "enzymes", "latex", "bioaero",
#"drugs", "react", "isocy", "clean", "wood", "metals", "mwf", "textile", "agric",
#"irrpeaks", "exhaust", "ets", "pos_irr", "low_anti"), ISCO != 8888 & ISCO != 9999 &
#ISCO != 94 & ISCO != 95 & ISCO != 97 & ISCO != 98)
#expo2$muster <- ""

#for (i in 1:nrow(expo2)){
#expo2$muster[i] <- paste(expo2[i,],sep="",collapse="")
#}

#table_EXPO_s2 <- table(expo2$muster)
#table_EXPO_s2
#print("Table der Expositionsmuster")
#print(table_EXPO_s2)
#prop_EXPO_s2 <- prop.table(as.array(table_EXPO_s2))
#prop_EXPO_s2
#print("Table der Wahrscheinlichkeiten der Expositionsmuster")
#print(prop_EXPO_s2)


#EXPO_s2_imput <- sample(names(prop_EXPO_s2), size = nrow(EXPO_s2),
#replace=TRUE, prob = prop_EXPO_s2)
#table(EXPO_s2_imput)
#print("Expositionsmuster das imputiert wird")
#print(EXPO_s2_imput)

# Exposition im Datensatz ersetzen
#j <- 1
#for (i in 1:nrow(datensatz)){
#  if(is.na(datensatz$ISCO)[i] & datensatz$STUDIE[i] == 2){
#    datensatz$anim[i] <- as.numeric(substring(EXPO_s2_imput,1,1))
#    datensatz$fish[i] <- as.numeric(substring(EXPO_s2_imput,2,2))
#    datensatz$flour[i] <- as.numeric(substring(EXPO_s2_imput,3,3))
#    datensatz$plants[i] <- as.numeric(substring(EXPO_s2_imput,4,4))
#    datensatz$mites[i] <- as.numeric(substring(EXPO_s2_imput,5,5))
#    datensatz$enzymes[i] <- as.numeric(substring(EXPO_s2_imput,6,6))
#    datensatz$latex[i] <- as.numeric(substring(EXPO_s2_imput,7,7))
#    datensatz$bioaero[i] <- as.numeric(substring(EXPO_s2_imput,8,8))
```

```
#    datensatz$drugs[i] <- as.numeric(substring(EXPO_s2_imput,9,9))
#    datensatz$react[i] <- as.numeric(substring(EXPO_s2_imput,10,10))
#    datensatz$isocy[i] <- as.numeric(substring(EXPO_s2_imput,11,11))
#    datensatz$clean[i] <- as.numeric(substring(EXPO_s2_imput,12,12))
#    datensatz$wood[i] <- as.numeric(substring(EXPO_s2_imput,13,13))
#    datensatz$metals[i] <- as.numeric(substring(EXPO_s2_imput,14,14))
#    datensatz$mwf[i] <- as.numeric(substring(EXPO_s2_imput,15,15))
#    datensatz$textile[i] <- as.numeric(substring(EXPO_s2_imput,16,16))
#    datensatz$agric[i] <- as.numeric(substring(EXPO_s2_imput,17,17))
#    datensatz$irrpeaks[i] <- as.numeric(substring(EXPO_s2_imput,18,18))
#    datensatz$exhaust[i] <- as.numeric(substring(EXPO_s2_imput,19,19))
#    datensatz$ets[i] <- as.numeric(substring(EXPO_s2_imput,20,20))
#   datensatz$pos_irr[i] <- as.numeric(substring(EXPO_s2_imput,21,21))
#    datensatz$low_anti[i] <- as.numeric(substring(EXPO_s2_imput,22,22))
#    datensatz$IMP_EXPO[i] <- 1
#    j <- j+1
#  }
#}
#}
#}

################################################################################
####                 Erst wenn alles imputiert wurde                        ####
################################################################################

# Datensatz der jetzt die imputierten Werte enthält soll zurückgegeben werden
return(datensatz)
} # Endklammer von function()
```

E Modellwahl

Auf Basis eines Datensatzes und des Modells für Allergische Rhinitis soll nun beispielhaft der R-Code abgedruckt werden, um die Modellwahl (auf Basis der vollständigen Tätigkeitsangaben) zu verdeutlichen. Alle anderen R-Codes sind der CD-Rom zu entnehmen.

E.1 Schritt 1: Confounder-Modell festlegen

```
################################################################################
########## Logit-Modell - Datensatz Amelia 1 ###################################
################################################################################

load("basis_modell_HEUSCHNUPFEN_amelia1.RData")

################################################################################
# Confounder-Modell auswählen #
################################################################################

HAY_confounder <- glm(s2CURHAYV ~ zentrum_r + d_geb + PAR_ALL_r + GESCHW +
STILL_r + CUR_DERM_r + CUR_HAY_r + ETSNOW_r + f02x + CURDERMV + CURHAYV + f58x
+ RAUCHEN + s2f78 + s2RAUCHEN + s2SCHULE + SES_r + JEMALS_GEARB,
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)
summary(HAY_confounder)

##############################
# Both-Selektion #
##############################

library(MASS)
HAY_confounder_bothAIC <- stepAIC(HAY_confounder,direction="both",
scope = list(upper = HAY_confounder, lower = ~SES_r + f02x))

summary(HAY_confounder_bothAIC)
# selektiertes Modell: s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + f02x + CURHAYV + SES_r
```

E.2 Schritt 2: Likelihood-Ratio-Tests der potenziellen Confounder-Modelle

```
################################################################################
########## Logit-Modell - Datensatz Amelia1   ##################################
################################################################################

load("basis_modell_HEUSCHNUPFEN_amelia1.RData")

##############################
# Modell 1 #
##############################

HAY_modell1 <- glm(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r,
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)
summary(HAY_modell1)

##############################
# Modell 2 #
##############################

HAY_modell2 <- glm(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+ STILL_r,
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)
summary(HAY_modell2)

# Likelihood-Ratio-Test: Modell1 vs. Modell2
library(lmtest)
lrtest(HAY_modell1, HAY_modell2) # keine Verbesserung

##############################
# Modell 3 #
##############################

HAY_modell3 <- glm(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+ STILL_r + f58x,
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)
summary(HAY_modell3)

# Likelihood-Ratio-Test: Modell2 vs. Modell3
library(lmtest)
lrtest(HAY_modell2, HAY_modell3) # keine Verbesserung
```

E.3 Schritt 3: Berechnung von GAMs

```
##############################################################################
########## GAM - Datensatz Amelia1 ###########################################
##############################################################################

#### Überprüfen, ob die Expositionen als lineare Terme oder als andere Funktionen
#### in das Modell eingehen können

load("basis_modell_HEUSCHNUPFEN_amelia1.RData")

########## Exposition kumuliert ##############################################
library(mgcv)
gam_kumuliert <- gam(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+  STILL_r + s(HMW_kumuliert) + s(LMW_kumuliert) + s(MIXED_kumuliert) + s(LOWRISK_kumuliert),
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)
plot(gam_kumuliert) # nichts quadratisch aufnehmen

########## Exposition 1.Beruf ################################################
library(mgcv)
gam_ersterberuf <- gam(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+  STILL_r + s(HMW_ersterberuf_gesamt) + s(LMW_ersterberuf_gesamt) +
s(MIXED_ersterberuf_gesamt) + s(LOWRISK_ersterberuf_gesamt),
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)
plot(gam_ersterberuf) # nichts quadratisch aufnehmen

########## Exposition 1.Jahr #################################################
library(mgcv)
gam_erstesjahr <- gam(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+  STILL_r + s(HMW_erstesjahr_gesamt) + s(LMW_erstesjahr_gesamt) +
s(MIXED_erstesjahr_gesamt) + s(LOWRISK_erstesjahr_gesamt),
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)
plot(gam_erstesjahr) # nichts quadratisch aufnehmen
```

E.4 Schritt 3: Durchführung von Likelihood-Ratio-Tests für die Expositionen

```
##############################################################################
########## Test, ob Expo-Variablen Einfluss haben - Datensatz Amelia 1 #########
##############################################################################

load("basis_modell_HEUSCHNUPFEN_amelia1.RData")

###########################################
##### Confounder-Modell definieren    #####
###########################################

HAY_confounder_amelia1 <- glm(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+ STILL_r ,family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)
summary(HAY_confounder_amelia1)

###########################################
##### kumulierte Exposition aufnehmen #####
###########################################
HAY_kum_amelia1 <- glm(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+ STILL_r + HMW_kumuliert + LMW_kumuliert + MIXED_kumuliert + IRRPEAKS_kumuliert + LOWRISK_kumuliert,
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)

# Likelihood-Ratio-Test: Confounder-Modell vs. Confounder-Modell mit allen Expo-Variablen
library(lmtest)
lrtest(HAY_confounder_amelia1, HAY_kum_amelia1) # keine Verbesserung

###########################################
##### binäre Exposition aufnehmen #####
###########################################
HAY_binaer_amelia1 <- glm(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+ STILL_r + HMW_binaer + LMW_binaer + MIXED_binaer + IRRPEAKS_binaer + LOWRISK_binaer,
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)

# Likelihood-Ratio-Test: Confounder-Modell vs. Confounder-Modell mit allen Expo-Variablen
library(lmtest)
lrtest(HAY_confounder_amelia1, HAY_binaer_amelia1) # keine Verbesserung

###########################################
##### Exposition 1.Jahr aufnehmen #####
###########################################
HAY_erstesjahr_amelia1 <- glm(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+ STILL_r + HMW_erstesjahr_gesamt + LMW_erstesjahr_gesamt + MIXED_erstesjahr_gesamt
+ IRRPEAKS_erstesjahr_gesamt + LOWRISK_erstesjahr_gesamt,
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)

# Likelihood-Ratio-Test: Confounder-Modell vs. Confounder-Modell mit allen Expo-Variablen
library(lmtest)
lrtest(HAY_confounder_amelia1, HAY_erstesjahr_amelia1) # keine Verbesserung

###########################################
##### Exposition 1.Jahr binär aufnehmen #####
###########################################
```

```
HAY_erstesjahrbin_amelia1 <- glm(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+ STILL_r + HMW_erstesjahr_binaer + LMW_erstesjahr_binaer + MIXED_erstesjahr_binaer
+ IRRPEAKS_erstesjahr_binaer + LOWRISK_erstesjahr_binaer,
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)

# Likelihood-Ratio-Test: Confounder-Modell vs. Confounder-Modell mit allen Expo-Variablen
library(lmtest)
lrtest(HAY_confounder_amelia1, HAY_erstesjahrbin_amelia1) # keine Verbesserung

##########################################
##### Exposition 1.Beruf aufnehmen #####
##########################################
HAY_ersterberuf_amelia1 <- glm(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+ STILL_r + HMW_ersterberuf_gesamt + LMW_ersterberuf_gesamt + MIXED_ersterberuf_gesamt
+ IRRPEAKS_ersterberuf_gesamt + LOWRISK_ersterberuf_gesamt,
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)

# Likelihood-Ratio-Test: Confounder-Modell vs. Confounder-Modell mit allen Expo-Variablen
library(lmtest)
lrtest(HAY_confounder_amelia1, HAY_ersterberuf_amelia1) # keine Verbesserung

##########################################
##### Exposition 1.Beruf binär aufnehmen #####
##########################################
HAY_ersterberufbin_amelia1 <- glm(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+ STILL_r + HMW_ersterberuf_binaer + LMW_ersterberuf_binaer + MIXED_ersterberuf_binaer +
IRRPEAKS_ersterberuf_binaer + LOWRISK_ersterberuf_binaer,
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)

# Likelihood-Ratio-Test: Confounder-Modell vs. Confounder-Modell mit allen Expo-Variablen
library(lmtest)
lrtest(HAY_confounder_amelia1, HAY_ersterberufbin_amelia1) # keine Verbesserung
```

E.5 Schritt 5: Gewähltes Modell analysieren

```
#############################################################################
########## Gewähltes Modell - Datensatz Amelia 1 ############################
#############################################################################

load("basis_modell_HEUSCHNUPFEN_amelia1.RData")

######################
### gewähltes Modell (inklusive binäre Expositionsvariablen)
######################
HAY_bestes_modell_amelia1 <- glm(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+ STILL_r + HMW_binaer + LMW_binaer + MIXED_binaer + IRRPEAKS_binaer + LOWRISK_binaer,
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)
summary(HAY_bestes_modell_amelia1)

######################
# Schätzer exponieren
######################
exp(coefficients(HAY_bestes_modell_amelia1))

######################
### Konfidenzintervall
######################
KI <- exp(confint(HAY_bestes_modell_amelia1))

######################
### ROC-Kurve
######################
library(Epi)
attach(basis_modell_HEUSCHNUPFEN_amelia1)
roc_am1 <- ROC(form = s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r
+  STILL_r + HMW_binaer + LMW_binaer + MIXED_binaer + IRRPEAKS_binaer + LOWRISK_binaer,
plot = "ROC", # nur ROC Kurve soll gezeichnet werden
PV = FALSE, # Angabe Sensitivität, Spezifität am optimalen Cutpoint
MX = FALSE, # Angabe Optimaler Cutpoint (wo Sensitivität und Spezifität optimal)
AUC = TRUE, # Area under Curve zeichnen lassen
lwd = 2, MI=FALSE)  # Model summary des logist. Modells mit reinschreiben lassen
detach(basis_modell_HEUSCHNUPFEN_amelia1)
```

E.6 Schritt 6: Schätzer des finalen Modells kombinieren

```
#############################################################################
########## Finales Modell - Datensatz Amelia 1 ##############################
#############################################################################

load("basis_modell_HEUSCHNUPFEN_amelia1.RData")

####################
### finales Modell
####################
HAY_bestes_modell_amelia1 <- glm(s2CURHAYV ~ PAR_ALL_r + CUR_HAY_r + CURHAYV + f02x + SES_r + STILL_r
```

```
+ HMW_binaer + LMW_binaer + MIXED_binaer + IRRPEAKS_binaer + LOWRISK_binaer,
family = binomial(link="logit"), data=basis_modell_HEUSCHNUPFEN_amelia1)
summary(HAY_bestes_modell_amelia1)

# analoges Vorgehen für die anderen Datensätze

################################################################################
#########               Kombination der Ergebnisse                     #########
################################################################################

####################
### Schätzer
####################

###### Schätzer aus den einzelnen Datensätzen
ca1 <- coefficients(HAY_bestes_modell_amelia1)
ca2 <- coefficients(HAY_bestes_modell_amelia2)
ca3 <- coefficients(HAY_bestes_modell_amelia3)
ce1 <- coefficients(HAY_bestes_modell_empVert1)
ce2 <- coefficients(HAY_bestes_modell_empVert2)

### kombinierter Parameterschätzer: Intercept
intercept_quer <- 1/5 * (ca1[1]+ca2[1]+ca3[1]+ce1[1]+ce2[1])
### kombinierter Parameterschätzer: PAR_ALL_r
PAR_ALL_r_quer <- 1/5 * (ca1[2]+ca2[2]+ca3[2]+ce1[2]+ce2[2])
### kombinierter Parameterschätzer: CUR_HAY_r
CUR_HAY_r_quer <- 1/5 * (ca1[3]+ca2[3]+ca3[3]+ce1[3]+ce2[3])
### kombinierter Parameterschätzer: CURHAYV
CURHAYV_quer <- 1/5 * (ca1[4]+ca2[4]+ca3[4]+ce1[4]+ce2[4])
### kombinierter Parameterschätzer: Geschlecht (f02x)
f02x_quer <- 1/5 * (ca1[5]+ca2[5]+ca3[5]+ce1[5]+ce2[5])
### kombinierter kombinierter Parameterschätzer: SES_r
SES_r_quer <- 1/5 * (ca1[6]+ca2[6]+ca3[6]+ce1[6]+ce2[6])
### kombinierter Parameterschätzer: STILL_r
STILL_r_quer <- 1/5 * (ca1[7]+ca2[7]+ca3[7]+ce1[7]+ce2[7])
### Parameterschätzer: HMW_binaer
HMW_binaer_quer <- 1/5 * (ca1[8]+ca2[8]+ca3[8]+ce1[8]+ce2[8])
### Parameterschätzer: LMW_binaer
LMW_binaer_quer <- 1/5 * (ca1[9]+ca2[9]+ca3[9]+ce1[9]+ce2[9])
### Parameterschätzer: MIXED_binaer
MIXED_binaer_quer <- 1/5 * (ca1[10]+ca2[10]+ca3[10]+ce1[10]+ce2[10])
### Parameterschätzer: IRRPEAKS_binaer
IRRPEAKS_binaer_quer <- 1/5 * (ca1[11]+ca2[11]+ca3[11]+ce1[11]+ce2[11])
### Parameterschätzer: LOWRISK_binaer
LOWRISK_binaer_quer <- 1/5 * (ca1[12]+ca2[12]+ca3[12]+ce1[12]+ce2[12])

####################
### Varianzen
####################

###### Varianz-Kovarianz-Matrizen der Schätzer
vcov_a1 <- vcov(HAY_bestes_modell_amelia1)
vcov_a2 <- vcov(HAY_bestes_modell_amelia2)
vcov_a3 <- vcov(HAY_bestes_modell_amelia3)
vcov_e1 <- vcov(HAY_bestes_modell_empVert1)
vcov_e2 <- vcov(HAY_bestes_modell_empVert2)

###### Varianz innerhalb jedes Datensatzes ist das arithmetische Mittel
###### der geschaetzten Varianzen:

### Varianz innerhalb des Datensatzes: Intercept
intercept_var_innerhalb <- 1/5 * (vcov_a1[1,1]+vcov_a2[1,1]+vcov_a3[1,1]
+vcov_e1[1,1]+vcov_e2[1,1])
### Varianz innerhalb des Datensatzes: PAR_ALL_r
PAR_ALL_r_var_innerhalb <- 1/5 * (vcov_a1[2,2]+vcov_a2[2,2]+vcov_a3[2,2]
+vcov_e1[2,2]+vcov_e2[2,2])
### Varianz innerhalb des Datensatzes: CUR_HAY_r
CUR_HAY_r_var_innerhalb <- 1/5 * (vcov_a1[3,3]+vcov_a2[3,3]+vcov_a3[3,3]
+vcov_e1[3,3]+vcov_e2[3,3])
### Varianz innerhalb des Datensatzes: CURHAYV
CURHAYV_var_innerhalb <- 1/5 * (vcov_a1[4,4]+vcov_a2[4,4]+vcov_a3[4,4]
+vcov_e1[4,4]+vcov_e2[4,4])
### Varianz innerhalb des Datensatzes: Geschlecht (f02x)
f02x_var_innerhalb <- 1/5 * (vcov_a1[5,5]+vcov_a2[5,5]+vcov_a3[5,5]
+vcov_e1[5,5]+vcov_e2[5,5])
### Varianz innerhalb des Datensatzes: SES_r
SES_r_var_innerhalb <- 1/5 * (vcov_a1[6,6]+vcov_a2[6,6]+vcov_a3[6,6]
+vcov_e1[6,6]+vcov_e2[6,6])
### Varianz innerhalb des Datensatzes: STILL_r
STILL_r_var_innerhalb <- 1/5 * (vcov_a1[7,7]+vcov_a2[7,7]+vcov_a3[7,7]
+vcov_e1[7,7]+vcov_e2[7,7])
### Varianz innerhalb des Datensatzes: HMW_binaer
HMW_binaer_var_innerhalb <- 1/5 * (vcov_a1[8,8]+vcov_a2[8,8]+vcov_a3[8,8]
+vcov_e1[8,8]+vcov_e2[8,8])
### Varianz innerhalb des Datensatzes: LMW_binaer
LMW_binaer_var_innerhalb <- 1/5 * (vcov_a1[9,9]+vcov_a2[9,9]+vcov_a3[9,9]
+vcov_e1[9,9]+vcov_e2[9,9])
### Varianz innerhalb des Datensatzes: MIXED_binaer
```

```
MIXED_binaer_var_innerhalb <- 1/5 * (vcov_a1[10,10]+vcov_a2[10,10]+vcov_a3[10,10]
+vcov_e1[10,10]+vcov_e2[10,10])
### Varianz innerhalb des Datensatzes: IRRPEAKS_binaer
IRRPEAKS_binaer_var_innerhalb <- 1/5 * (vcov_a1[11,11]+vcov_a2[11,11]+vcov_a3[11,11]
+vcov_e1[11,11]+vcov_e2[11,11])
### Varianz innerhalb des Datensatzes: LOWRISK_binaer
LOWRISK_binaer_var_innerhalb <- 1/5 * (vcov_a1[12,12]+vcov_a2[12,12]+vcov_a3[12,12]
+vcov_e1[12,12]+vcov_e2[12,12])

####Die Varianz zwischen den Datensaetzen ist die Stichprobenvarianz der Schaetzer selbst:

### Varianz zwischen den Datensaetzen: Intercept
intercept_var_zwischen <- 1/4 * ((ca1[1]-intercept_quer)^2
+(ca2[1]-intercept_quer)^2+(ca3[1]-intercept_quer)^2
+(ce1[1]-intercept_quer)^2+(ce2[1]-intercept_quer)^2)
### Varianz zwischen den Datensaetzen: PAR_ALL_r
PAR_ALL_r_var_zwischen <- 1/4 * ((ca1[2]-PAR_ALL_r_quer)^2
+(ca2[2]-PAR_ALL_r_quer)^2+(ca3[2]-PAR_ALL_r_quer)^2
+(ce1[2]-PAR_ALL_r_quer)^2+(ce2[2]-PAR_ALL_r_quer)^2)
### Varianz zwischen den Datensaetzen: CUR_HAY_r
CUR_HAY_r_var_zwischen <- 1/4 * ((ca1[3]-CUR_HAY_r_quer)^2
+(ca2[3]-CUR_HAY_r_quer)^2+(ca3[3]-CUR_HAY_r_quer)^2
+(ce1[3]-CUR_HAY_r_quer)^2+(ce2[3]-CUR_HAY_r_quer)^2)
### Varianz zwischen den Datensaetzen: CURHAYV
CURHAYV_var_zwischen <- 1/4 * ((ca1[4]-CURHAYV_quer)^2
+(ca2[4]-CURHAYV_quer)^2+(ca3[4]-CURHAYV_quer)^2
+(ce1[4]-CURHAYV_quer)^2+(ce2[4]-CURHAYV_quer)^2)
### Varianz zwischen den Datensaetzen: Geschlecht (f02x)
f02x_var_zwischen <- 1/4 * ((ca1[5]-f02x_quer)^2
+(ca2[5]-f02x_quer)^2+(ca3[5]-f02x_quer)^2
+(ce1[5]-f02x_quer)^2+(ce2[5]-f02x_quer)^2)
### Varianz zwischen den Datensaetzen: SES_r
SES_r_var_zwischen <- 1/4 * ((ca1[6]-SES_r_quer)^2
+(ca2[6]-SES_r_quer)^2+(ca3[6]-SES_r_quer)^2
+(ce1[6]-SES_r_quer)^2+(ce2[6]-SES_r_quer)^2)
### Varianz zwischen den Datensaetzen: STILL_r
STILL_r_var_zwischen <- 1/4 * ((ca1[7]-STILL_r_quer)^2
+(ca2[7]-STILL_r_quer)^2+(ca3[7]-STILL_r_quer)^2
+(ce1[7]-STILL_r_quer)^2+(ce2[7]-STILL_r_quer)^2)
### Varianz zwischen den Datensaetzen: HMW_binaer
HMW_binaer_var_zwischen <- 1/4 * ((ca1[8]-HMW_binaer_quer)^2
+(ca2[8]-HMW_binaer_quer)^2+(ca3[8]-HMW_binaer_quer)^2
+(ce1[8]-HMW_binaer_quer)^2+(ce2[8]-HMW_binaer_quer)^2)
### Varianz zwischen den Datensaetzen: LMW_binaer
LMW_binaer_var_zwischen <- 1/4 * ((ca1[9]-LMW_binaer_quer)^2
+(ca2[9]-LMW_binaer_quer)^2+(ca3[9]-LMW_binaer_quer)^2
+(ce1[9]-LMW_binaer_quer)^2+(ce2[9]-LMW_binaer_quer)^2)
### Varianz zwischen den Datensaetzen: MIXED_binaer
MIXED_binaer_var_zwischen <- 1/4 * ((ca1[10]-MIXED_binaer_quer)^2
+(ca2[10]-MIXED_binaer_quer)^2+(ca3[10]-MIXED_binaer_quer)^2
+(ce1[10]-MIXED_binaer_quer)^2+(ce2[10]-MIXED_binaer_quer)^2)
### Varianz zwischen den Datensaetzen: IRRPEAKS_binaer
IRRPEAKS_binaer_var_zwischen <- 1/4 * ((ca1[11]-IRRPEAKS_binaer_quer)^2
+(ca2[11]-IRRPEAKS_binaer_quer)^2+(ca3[11]-IRRPEAKS_binaer_quer)^2
+(ce1[11]-IRRPEAKS_binaer_quer)^2+(ce2[11]-IRRPEAKS_binaer_quer)^2)
### Varianz zwischen den Datensaetzen: LOWRISK_binaer
LOWRISK_binaer_var_zwischen <- 1/4 * ((ca1[12]-LOWRISK_binaer_quer)^2
+(ca2[12]-LOWRISK_binaer_quer)^2+(ca3[12]-LOWRISK_binaer_quer)^2
+(ce1[12]-LOWRISK_binaer_quer)^2+(ce2[12]-LOWRISK_binaer_quer)^2)

###### Die Gesamtvarianz T entspricht der Summe der beiden Komponenten mit einem
###### zusaetzlichen Korrekturfaktor fuer den Simulationsfehler in Q_quer

### Gesamtvarianz: Intercept
intercept_var_gesamt <- (intercept_var_innerhalb
+(1+(1/5))*intercept_var_zwischen)
### Gesamtvarianz: PAR_ALL_r
PAR_ALL_r_var_gesamt <- (PAR_ALL_r_var_innerhalb
+(1+(1/5))*PAR_ALL_r_var_zwischen)
### Gesamtvarianz: CUR_HAY_r
CUR_HAY_r_var_gesamt <- (CUR_HAY_r_var_innerhalb+(1+(1/5))*CUR_HAY_r_var_zwischen)
### Gesamtvarianz: CURHAYV
CURHAYV_var_gesamt <- (CURHAYV_var_innerhalb+(1+(1/5))*CURHAYV_var_zwischen)
### Gesamtvarianz: Geschlecht (f02x)
f02x_var_gesamt <- (f02x_var_innerhalb+(1+(1/5))*f02x_var_zwischen)
### Gesamtvarianz: SES_r
SES_r_var_gesamt <- (SES_r_var_innerhalb+(1+(1/5))*SES_r_var_zwischen)
### Gesamtvarianz: STILL_r
STILL_r_var_gesamt <- (STILL_r_var_innerhalb+(1+(1/5))*STILL_r_var_zwischen)
### Gesamtvarianz: HMW_binaer
HMW_binaer_var_gesamt <- (HMW_binaer_var_innerhalb+(1+(1/5))*HMW_binaer_var_zwischen)
### Gesamtvarianz: LMW_binaer
LMW_binaer_var_gesamt <- (LMW_binaer_var_innerhalb+(1+(1/5))*LMW_binaer_var_zwischen)
### Gesamtvarianz: MIXED_binaer
MIXED_binaer_var_gesamt <- (MIXED_binaer_var_innerhalb+(1+(1/5))*MIXED_binaer_var_zwischen)
### Gesamtvarianz: IRRPEAKS_binaer
IRRPEAKS_binaer_var_gesamt <- (IRRPEAKS_binaer_var_innerhalb+(1+(1/5))*IRRPEAKS_binaer_var_zwischen)
### Gesamtvarianz: LOWRISK_binaer
```

```
LOWRISK_binaer_var_gesamt <- (LOWRISK_binaer_var_innerhalb+(1+(1/5))*LOWRISK_binaer_var_zwischen)

####################
# Standardabweichungen der Schätzer: sqrt(Varianz)
####################

### Standardabweichung: Intercept
intercept_stdabw <- sqrt(intercept_var_gesamt)
### Standardabweichung: PAR_ALL_r
PAR_ALL_r_stdabw <- sqrt(PAR_ALL_r_var_gesamt)
### Standardabweichung: CUR_HAY_r
CUR_HAY_r_stdabw <- sqrt(CUR_HAY_r_var_gesamt)
### Standardabweichung: CURHAYV
CURHAYV_stdabw <- sqrt(CURHAYV_var_gesamt)
### Standardabweichung: Geschlecht (f02x)
f02x_stdabw <- sqrt(f02x_var_gesamt)
### Standardabweichung: SES_r
SES_r_stdabw <- sqrt(SES_r_var_gesamt)
### Standardabweichung: STILL_r
STILL_r_stdabw <- sqrt(STILL_r_var_gesamt)
### Standardabweichung: HMW_binaer
HMW_binaer_stdabw <- sqrt(HMW_binaer_var_gesamt)
### Standardabweichung: LMW_binaer
LMW_binaer_stdabw <- sqrt(LMW_binaer_var_gesamt)
### Standardabweichung: MIXED_binaer
MIXED_binaer_stdabw <- sqrt(MIXED_binaer_var_gesamt)
### Standardabweichung: IRRPEAKS_binaer
IRRPEAKS_binaer_stdabw <- sqrt(IRRPEAKS_binaer_var_gesamt)
### Standardabweichung: LOWRISK_binaer
LOWRISK_binaer_stdabw <- sqrt(LOWRISK_binaer_var_gesamt)

##############################################################################
#########    (kombinierte) Konfidenzintervalle berechnen und zeichen   #########
##############################################################################

KI_u_intercept <- exp(intercept_quer - 1.96 * intercept_stdabw)
KI_o_intercept <- exp(intercept_quer + 1.96 * intercept_stdabw)

KI_u_PAR_ALL_r <- exp(PAR_ALL_r_quer - 1.96 * PAR_ALL_r_stdabw)
KI_o_PAR_ALL_r <- exp(PAR_ALL_r_quer + 1.96 * PAR_ALL_r_stdabw)

KI_u_CUR_HAY_r <- exp(CUR_HAY_r_quer - 1.96 * CUR_HAY_r_stdabw)
KI_o_CUR_HAY_r <- exp(CUR_HAY_r_quer + 1.96 * CUR_HAY_r_stdabw)

KI_u_CURHAYV <- exp(CURHAYV_quer - 1.96 * CURHAYV_stdabw)
KI_o_CURHAYV <- exp(CURHAYV_quer + 1.96 * CURHAYV_stdabw)

KI_u_f02x <- exp(f02x_quer - 1.96 * f02x_stdabw)
KI_o_f02x <- exp(f02x_quer + 1.96 * f02x_stdabw)

KI_u_SES_r <- exp(SES_r_quer - 1.96 * SES_r_stdabw)
KI_o_SES_r <- exp(SES_r_quer + 1.96 * SES_r_stdabw)

KI_u_STILL_r <- exp(STILL_r_quer - 1.96 * STILL_r_stdabw)
KI_o_STILL_r <- exp(STILL_r_quer + 1.96 * STILL_r_stdabw)

KI_u_HMW_binaer <- exp(HMW_binaer_quer - 1.96 * HMW_binaer_stdabw)
KI_o_HMW_binaer <- exp(HMW_binaer_quer + 1.96 * HMW_binaer_stdabw)

KI_u_LMW_binaer <- exp(LMW_binaer_quer - 1.96 * LMW_binaer_stdabw)
KI_o_LMW_binaer <- exp(LMW_binaer_quer + 1.96 * LMW_binaer_stdabw)

KI_u_MIXED_binaer <- exp(MIXED_binaer_quer - 1.96 * MIXED_binaer_stdabw)
KI_o_MIXED_binaer <- exp(MIXED_binaer_quer + 1.96 * MIXED_binaer_stdabw)

KI_u_IRRPEAKS_binaer <- exp(IRRPEAKS_binaer_quer - 1.96 * IRRPEAKS_binaer_stdabw)
KI_o_IRRPEAKS_binaer <- exp(IRRPEAKS_binaer_quer + 1.96 * IRRPEAKS_binaer_stdabw)

KI_u_LOWRISK_binaer <- exp(LOWRISK_binaer_quer - 1.96 * LOWRISK_binaer_stdabw)
KI_o_LOWRISK_binaer <- exp(LOWRISK_binaer_quer + 1.96 * LOWRISK_binaer_stdabw)

plot( c(0, 55) , y=c(0, 11), type="n", xlab="",ylab="", yaxt="n", yaxs="r",
main="95%-Konfidenzintervalle der Odds-Ratios",
sub="Logit-Modell für Allergische Rhinitis")
axis(1, at=c(0,1,2,3,4,5,10,20,30,40,50))
axis(2, labels=F, tick=F)
# "Pfeil" in beide Richtungen zeichnen
axis(1,at=c(0,1,2,3,4,5,10,20,30,40))
arrows(KI_u_intercept, 11, KI_o_intercept, 11, angle=90, length=0.1, code=3, lwd=2)
points(exp(intercept_quer), 11, pch=18, cex=1.5)
text( 41, 11, "Intercept",adj=c(0,0),font=2)
arrows(KI_u_PAR_ALL_r, 10, KI_o_PAR_ALL_r, 10, angle=90, length=0.1, code=3, lwd=2)
points(exp(PAR_ALL_r_quer), 10, pch=18, cex=1.5)
text( 41, 10, "Atopie der Eltern",adj=c(0,0),font=2)
arrows(KI_u_CUR_HAY_r, 9, KI_o_CUR_HAY_r, 9, angle=90, length=0.1, code=3, lwd=2)
points(exp(CUR_HAY_r_quer), 9, pch=18, cex=1.5)
text( 41, 9, "Allergische Rhinitis (ISAAC II)",adj=c(0,0),font=2)
arrows(KI_u_CURHAYV, 8, KI_o_CURHAYV, 8, angle=90, length=0.1, code=3, lwd=2)
```

```
points(exp(CURHAYV_quer), 8, pch=18, cex=1.5)
text( 41, 8, "Allergische Rhinitis (SOLAR)",adj=c(0,0),font=2)
arrows(KI_u_f02x, 7, KI_o_f02x, 7, angle=90, length=0.1, code=3, lwd=2)
points(exp(f02x_quer), 7, pch=18, cex=1.5)
text( 41, 7, "Geschlecht",adj=c(0,0),font=2)
arrows(KI_u_SES_r, 6, KI_o_SES_r, 6, angle=90, length=0.1, code=3, lwd=2)
points(exp(SES_r_quer), 6, pch=18, cex=1.5)
text( 41, 6, "Sozioökonomischer Status",adj=c(0,0),font=2)
arrows(KI_u_STILL_r, 5, KI_o_STILL_r, 5, angle=90, length=0.1, code=3, lwd=2)
points(exp(STILL_r_quer), 5, pch=18, cex=1.5)
text( 41, 5, "Als Säugling gestillt",adj=c(0,0),font=2)
arrows(KI_u_HMW_binaer, 4, KI_o_HMW_binaer, 4, angle=90, length=0.1, code=3, lwd=2)
points(exp(HMW_binaer_quer), 4, pch=18, cex=1.5)
text( 41, 4, "HMW-Exposition binär",adj=c(0,0),font=2)
arrows(KI_u_LMW_binaer, 3, KI_o_LMW_binaer, 3, angle=90, length=0.1, code=3, lwd=2)
points(exp(LMW_binaer_quer), 3, pch=18, cex=1.5)
text( 41, 3, "LMW-Exposition binär",adj=c(0,0),font=2)
arrows(KI_u_MIXED_binaer, 2, KI_o_MIXED_binaer, 2, angle=90, length=0.1, code=3, lwd=2)
points(exp(MIXED_binaer_quer), 2, pch=18, cex=1.5)
text( 41, 2, "MIXED-Exposition binär",adj=c(0,0),font=2)
arrows(KI_u_IRRPEAKS_binaer, 1, KI_o_IRRPEAKS_binaer, 1,angle=90,length=0.1,code=3,lwd=2)
points(exp(IRRPEAKS_binaer_quer), 1, pch=18, cex=1.5)
text( 41, 1, "IRRPEAKS-Exposition binär",adj=c(0,0),font=2)
arrows(KI_u_LOWRISK_binaer, 0, KI_o_LOWRISK_binaer, 0, angle=90, length=0.1, code=3,lwd=2)
points(exp(LOWRISK_binaer_quer), 0, pch=18, cex=1.5)
text( 41, 0, "LOWRISK-Exposition binär",adj=c(0,0),font=2)
abline(v=1,lty=2)
```

F CD Inhalt

Die beiliegende CD enthält die digitale Ausgabe der vorliegenden Arbeit im PDF-Format, sämtliche Datensätze und R-Files, die zugrundeliegenden Fragebögen und die Quellen. Nachfolgend ist in Abbildung F.1 die Ordnerstruktur der CD abgedruckt.

Abbildung F.1: Ordnerstruktur der beiligenden CD

G Eidesstattliche Erklärung

Hiermit versichere ich, dass ich die vorliegende Bachelorarbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe.

Engelbrechtsmünster, den 29.Juni 2009

Ort, Datum *(Unterschrift des Autors)*

Literaturverzeichnis

[ANDERSON et al. 1992] Anderson, H., Pottier, A., Strachan, D., 1992, *Asthma from birth to age 23: incidence and relation to prior and concurrent atopic disease*: Thorax, 47: Pp. 537-542

[BENKE et al. 2008] Benke, G. et al., 2008, *Comparison of First, Last and Longest-Held Jobs as Surrogates for All Jobs in Estimating Cumulative Exposure in Cross-Sectional Studies of Work-Related Asthma*: Annals of Epidemiology, 18: Pp. 23-27

[FAHRMEIR et al. 2004] Fahrmeir, L., Künstler, R., Pigeot, I., Tutz, G., 2004, *Statistik - Der Weg zur Datenanalyse, 5. Auflage*: Springer Verlag

[FAHRMEIR et al. 2007] Fahrmeir, L., Kneib, T., Lang, S., 2007, *Regression - Modelle, Methoden und Anwendungen*: Springer Verlag

[GEIS 2007] Geis, A., 2007, *Handbuch für die Berufsvercodung*: ZUMA

[HONAKER und KING 2008] Honaker, J., King, G., 2008, *What to do about Missing Values in Time Series Cross-Section Data*: http://gking.harvard.edu/files/pr.pdf

[HONAKER et al. 2009] Honaker, J., King, G., Blackwell, M., 2009, *Amelia II: A Program for Missing Data*: http://gking.harvard.edu/amelia/docs/amelia.pdf

[KENNEDY et al. 2000] Kennedy, S., Le Moual, N., Choudat, D., Kauffmann, F., 2000, *Development of an asthma specific job exposure matrix and its application in the epidemiological study of genetics and environment in asthma (EGEA)*: Occupational and Environmental Medicine, 57: Pp. 635-641

[KING et al. 2001] King, G., Honaker, J., Joseph, A., Scheve, K., 2001, *Analyzing Incomplete Political Science Data: An Alternative Algorithm for Multiple Imputation*: American Political Science Review, 95 (1): Pp. 49-69 (http://gking.harvard.edu/files/evil.pdf)

[LITTLE und RUBIN 2002] Little, R., Rubin, D., 2002, *Statistical Analysis with Missing Data, Second Edition*: John Wiley & Sons

[RADON et al. 2005] Radon, K. et al., 2005, *Berufliche Allergierisiken - Die SOLAR-Kohortenstudie*: Schriftenreihe der Bundesanstalt für Arbeitsschutz und Arbeitsmedizin, Forschungsbericht 1045

[RADON et al. 2008] Radon, K. et al., 2008, *Manifestation allergischer Krankheiten bei jungen Erwachsenen in Zusammenhang mit dem Eintritt in das Berufsleben*: Sachstandsbericht November 2008

[RADON 2008] Radon, K., 2008, *Manifestation allergischer Krankheiten bei Jugendlichen in Zusammenhang mit dem Eintritt in das Berufsleben*: Stellungnahme zum Forschungsvorhaben Nr. 50/05

[SACHS und HEDDERICH 2006] Sachs, L., Hedderich, J., 2006, *Angewandte Statistik - Methodensammlung mit R, 12. Auflage*: Springer Verlag

[SCHAFER und OLSEN 2007] Schafer, J., Olsen, M., 1998, *Multiple Imputation for Multivariate Missing-Data Problems: A Data Analyst's Perspective*: Multivariate Behavioral Research, 33: Pp. 545-571

[STRACHAN 1989] Strachan, D., 1989, *Hay fever, hygiene, and household size*: BMJ, 299: Pp. 1259-1260

[TOUTENBURG 2003] Toutenburg, H., 2003, *Lineare Modelle, 2. Auflage*: Physica-Verlag

[TOUTENBURG und HEUMANN 2006] Toutenburg, H., Heumann, C., 2006, *Deskriptive Statistik - Eine Einführung in Methoden und Anwendungen mit SPSS, 5. Auflage*: Springer-Verlag