Ludwig-Maximilians-Universität
München

Institut für Statistik

Bachelorarbeit

Vergleich verschiedener Verfahren zur Datenimputation

Autor:
Susanne Rubenbauer

Betreuer:
Prof. Dr. Christian Heumann

Datum:
10. Juli 2015

In dieser Bachelorarbeit werden verschiedene Methoden zur Datenimputation vorgestellt, durchgeführt und miteinander verglichen. Das Hauptaugenmerk liegt dabei auf einer einfachen Imputationmethode, bei der fehlende Werte mithilfe von Regression imputiert werden. Die Ergebnisse werden anschließend mit bekannten Methoden zur multiplen Datenimputation verglichen.
Um diesen Vergleich durchführen zu können, werden die Daten zu Beginn nach einem vorgegebenen Algorithmus simuliert, danach fehlende Werte erzeugt und die Daten anschließend mit den verschiedenen Methoden wieder imputiert.
Die interessierenden Größen, nämlich die Koeffizienten einer Regression auf Grundlage des imputierten Datensatzes, werden anschließend untereinander und mit den wahren Koeffizienten verglichen.
Es stellt sich heraus, dass die Imputation kategorialer Variablen bei der Regressionsimputation Schwierigkeiten bereitet. Ebenso wird der wahre Zusammenhang in den Daten für die multiplen Imputationsmethoden tendenziell besser abgebildet als für die einfache Imputationsmethode.

Inhaltsverzeichnis

1. Einleitung **1**

2. Simulation der Daten **3**

2.1. Algorithmus zur Erzeugung der Daten 3

2.2. Umsetzung der Simulation in R 7

2.2.1. Gamma-Verteilung in R 7

2.2.2. Funktion zur Durchführung der Simulation 7

2.3. Datensätze 8

3. Fehlende Werte **10**

3.1. Klassifikation fehlender Werte 10

3.2. Erzeugen der fehlenden Werte 11

3.3. Mittlere Fehlerraten in den Datensätzen 12

4. Imputation fehlender Daten **16**

4.1. Einfache Imputationsverfahren 16

4.2. Multiple Imputationsverfahren 17

4.3. Umsetzung in R 18

5. Imputation mit Amelia II **21**

5.1. Theorie 21

5.1.1. Annahmen 21

5.1.2. Algorithmus 22

5.2. Umsetzung in R 24

5.3. Ergebnisse 25

5.3.1. Kleinerer Datensatz 25

5.3.2. Größerer Datensatz 28

6. Imputation mit mice **30**
6.1. Theorie . . . 30
6.1.1. Annahmen . . . 30
6.1.2. Algorithmus . . . 31
6.2. Umsetzung in R . . . 32
6.3. Ergebnisse . . . 34
6.3.1. Kleinerer Datensatz . . . 34
6.3.2. Größerer Datensatz . . . 36

7. Regressionsimputation **38**
7.1. Theorie . . . 38
7.1.1. Annahmen . . . 38
7.1.2. Algorithmus . . . 38
7.2. Umsetzung in R . . . 45
7.3. Ergebnisse . . . 46
7.3.1. Kleinerer Datensatz . . . 46
7.3.2. Größerer Datensatz . . . 55

8. Vergleich der Ergebnisse **59**
8.1. Kleinerer Datensatz . . . 59
8.2. Größerer Datensatz . . . 64
8.3. Vorteile und Nachteile bei der Umsetzung in R . . . 67

9. Zusammenfassung **68**

Literaturverzeichnis **69**

A. Elektronischer Anhang **71**

Abbildungsverzeichnis

3.1. Übersicht über den Anteil fehlender Werte pro Variable in jeder Runde. Es wird der Datensatz mit zehn Variablen betrachtet, wobei die mittlere Fehlerrate knapp unter 20 % liegt. 13
3.2. Übersicht über den Anteil fehlender Werte pro Variable in jeder Runde. Es wird der Datensatz mit zehn Variablen betrachtet, wobei die mittlere Fehlerrate knapp unter 10 % liegt. 14

5.1. Schematische Darstellung der Imputation mit *Amelia* mithilfe des EMB-Algorithmus aus (Honaker et al.; 2011). 22
5.2. Übersicht über die geschätzten Regressionskoeffizienten nach der Imputation mit *Amelia* aus 500 Durchgängen. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 10 % betrachtet. . 26
5.3. Übersicht über die geschätzten Regressionskoeffizienten nach der Imputation mit *Amelia* aus 500 Durchgängen. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet. . 27
5.4. Übersicht über die geschätzten Regressionskoeffizienten nach der Imputation mit *Amelia* aus 500 Durchgängen. Es wird der Datensatz mit 20 Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet. . 28

6.1. Schematische Darstellung der Imputation mit *mice* in R aus (van Buuren und Groothuis-Oudshoorn; 2011). 32
6.2. Übersicht über die geschätzten Regressionskoeffizienten nach der Imputation mit *mice* aus 500 Durchgängen. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 10 % betrachtet. . 35
6.3. Übersicht über die geschätzten Regressionskoeffizienten nach der Imputation mit *mice* aus 500 Durchgängen. Es wird der Datensatz mit 20 Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet. . 36

7.1. Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 10 % betrachtet. . 47
7.2. Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet. . 48
7.3. Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen, beim zugrundeliegenden Algorithmus wird zu den geschätzten Koeffizienten kein zusätzlicher Standardfehler addiert. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet. 50
7.4. Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen, wobei der Algorithmus mehrmals durchlaufen wird. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet. 51
7.5. Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen. Die Variablen werden dabei in entgegengesetzter Reihenfolge wie bei der Simulation imputiert. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet. 53
7.6. Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen, beim zugrundeliegenden Algorithmus wird zu den geschätzten Koeffizienten kein zusätzlicher Standardfehler addiert. Die Variablen werden dabei in entgegengesetzter Reihenfolge wie bei der Simulation imputiert. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet. 54
7.7. Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen, beim zugrundeliegenden Algorithmus wird zu den geschätzten Koeffizienten kein zusätzlicher Standardfehler addiert. Die Variablen werden dabei in analoger Reihenfolge wie bei der Simulation imputiert. Es wird der Datensatz mit 20 Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet. 56

7.8. Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen, beim zugrundeliegenden Algorithmus wird zu den geschätzten Koeffizienten kein zusätzlicher Standardfehler addiert. Die Variablen werden dabei in entgegengesetzter Reihenfolge wie bei der Simulation imputiert. Es wird der Datensatz mit 20 Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet. 57

8.1. Übersicht über die Schätzungen des Koeffizienten β_2 mit den verschiedenen Imputationsmethoden aus 500 Durchgängen. Den Schätzungen liegt der kleinere Datensatz und eine mittlere Fehlerrate um die 10 % zugrunde. 60
8.2. Übersicht über die Schätzungen des Koeffizienten β_2 mit den verschiedenen Imputationsmethoden aus 500 Durchgängen. Den Schätzungen liegt der kleinere Datensatz und eine mittlere Fehlerrate um die 20 % zugrunde. 62
8.3. Übersicht über die Schätzungen des Koeffizienten β_5 mit den verschiedenen Imputationsmethoden aus 500 Durchgängen. Den Schätzungen liegt der kleinere Datensatz und eine mittlere Fehlerrate um die 20 % zugrunde. 63
8.4. Übersicht über die Schätzungen des Koeffizienten β_2 mit den verschiedenen Imputationsmethoden aus 500 Durchgängen. Den Schätzungen liegt der größere Datensatz und eine mittlere Fehlerrate um die 20 % zugrunde. 65
8.5. Übersicht über die Schätzungen des Koeffizienten β_3 mit den verschiedenen Imputationsmethoden aus 500 Durchgängen. Den Schätzungen liegt der größere Datensatz und eine mittlere Fehlerrate um die 20 % zugrunde. 66

Tabellenverzeichnis

2.1. Verteilungstypen und benötigte Übergabeparameter zur Simulation der ersten Variable. 3
2.2. Verteilungstypen, zugrundeliegende Linkfunktion und Zufallsziehung in R für die Simulation aus Regressionsmodellen. 6

5.1. Liste möglicher Angaben für das Regressionsmodell bei der Funktion *zelig*. 25

6.1. Liste einiger univariaten Imputationsmethoden der Funktion *mice* in R. . 33

7.1. Datenbeispiel mit perfekter Trennung 43

8.1. Darstellung der verwendeten Imputationsmethoden für die beiden Datensätze. 59

1. Einleitung

Ein häufiges Problem bei Umfragen und Datenerhebungen ist die Unvollständigkeit der Antworten. Oft geben Befragte bewusst keine Auskunft zu bestimmten Themen, vor allem bei delikaten Fragen wie etwa zum Gehalt.
Durch die fehlenden Antworten kann es zu Verzerrungen kommen, bei einer Analyse nur auf Grundlage der vorhandenen Daten wird die Situation oft falsch dargestellt. Ein Verfahren, das diese Verzerrung verringern soll, ist die Datenimputation. Dabei werden die fehlenden Werte im Datensatz durch möglichst plausible Werte vervollständigt. Dafür existieren mehrere Ansätze, wovon einige in dieser Arbeit genauer vorgestellt werden.

In dieser Bachelorarbeit sollen verschiedene Methoden zur Datenimputation angewendet und die Ergebnisse miteinander verglichen werden. Die Auswertungen basieren auf selbst simulierten Datensätzen, da so der wahre Zusammenhang in den Daten bekannt ist und mit den geschätzten Zusammenhängen verglichen werden kann. Angewendet werden dabei ein selbst programmierter Algorithmus der einfachen Regressionsimputation sowie einige Modifikationen dieses Algorithmus, die mit zwei multiplen Imputationsmethoden aus den bestehenden R-Paketen *Amelia* und *mice* verglichen werden. Alle Auswertungen werden dabei mit dem Programmpaket R (R Development Core Team; 2008) durchgeführt.

In Kapitel 2 wird zunächt die genaue Simulation der Daten erklärt und die zwei erzeugten Datensätze beschrieben. Zusätzlich wird auf die Durchführung in R eingegangen.

Kapitel 3 befasst sich mit der Klassifikation fehlender Werte, der künstlichen Erzeugung der Missings, der Durchführung in R und den letztendlichen mittleren Feherraten in den Datensätzen.

Die allgemein existierenden Arten von Imputationsmethoden werden in Kapitel 4 dargestellt. Ebenso wird auf den genauen Vorgang der Imputationen in dieser Arbeit ein-

gegangen sowie auf deren Umsetzung in R.

In Kapitel 5 wird zuerst das R-Paket *Amelia* vorgestellt und der zugrundeliegende Imputationsmechanismus erklärt. Zusätzlich werden die Durchführung in R sowie die Ergebnisse für die beiden Datensätze dargestellt.

Kapitel 6 ist analog aufgebaut wie Kapitel 5, nur dass die Imputation mit dem R-Paket *mice* durchgeführt wird.

Kapitel 7 befasst sich mit der zu testenden Regressionsimputation, die mit den multiplen Imputationsmethoden verglichen werden soll. Der Aufbau des Kapitels ist analog wie in Kapitel 5 und 6.

In Kapitel 8 werden die Ergebnisse der verschiedenen Imputationsmethoden miteinander verglichen. Ebenso wird kurz auf die Vor- und Nachteile der Imputationsmethoden bei der Umsetzung in R eingegangen.

Kapitel 9 fasst schlussendlich die wichtigsten Punkte dieser Arbeit noch einmal zusammen.

2. Simulation der Daten

Der Vergleich der verschiedenen Imputationsmethoden wird mithilfe selbst simulierter Datensätze durchgeführt. Erzeugt werden dabei zwei verschieden große Datensätze. Der erste Datensatz hat einen Umfang von zehn Variablen mit jeweils 1000 Beobachtungen, der zweite Datensatz ist etwas größer und umfasst 20 Variablen und 1000 Beobachtungen. Die Erzeugung der Daten folgt dabei einem vorgegebenem Schema, welches in diesem Kapitel genauer beschrieben wird.

2.1. Algorithmus zur Erzeugung der Daten

Zur Erzeugung der ersten Variablen des Datensatzes werden einfache Zufallszahlen gezogen. Zur Auswahl stehen normal-, poisson-, gamma- und binomial-verteilte Variablen sowie für multinomial-verteilte Variablen nominal- und ordinal-skalierte Daten. Es werden dabei, je nach Verteilungstyp, die benötigten Parameter beliebig festgelegt:

Verteilungstyp	**Übergabeparameter**
Normal	Erwartungswert μ
	Standardabweichung σ
Poisson	Erwartungswert λ
Gamma	Shape-Parameter ν
	Scale-Parameter $\frac{\mu}{\nu}$
Binomial	$P(X_1 = 0), P(X_1 = 1)$
Multinomial	$P(X_1 = 1), ..., P(X_1 = k)$
(nominal oder ordinal)	

Tabelle 2.1.: Verteilungstypen und benötigte Übergabeparameter zur Simulation der ersten Variable.

Soll also beispielsweise eine standardnormal-verteilte Variable erzeugt werden, müssen der Erwartungswert $\mu = 0$ sowie die Standardabweichung $\sigma = 1$ festgelegt werden. Für kategoriale Variablen müssen die Wahrscheinlichkeiten für jede Kategorie $1, ..., k$ angegeben werden, die Wahrscheinlichkeiten müssen sich dabei insgesamt zu eins aufsummieren.

In jedem weiteren Schritt wird die neue Variable aus einem Regressionsmodell simuliert. Für normal-, poisson-, gamma- und binomial-verteilte Variablen wird dabei aus einem generalisierten linearen Modell simuliert, bei nominalen und ordinalen Variablen aus einem multikategorialen Modell. Die Theorie zu diesem Kapitel stützt sich auf (Fahrmeir et al.; 2009).

Zur Erzeugung der Variable werden zuerst die nötigen Regressionskoeffizienten $\beta_0, ..., \beta_p$ beliebig, aber sinnvoll festgelegt. Eine sinnvolle Festlegung bedeutet dabei, dass beispielsweise für kategoriale Variablen auch schlussendlich jede Kategorie im Datensatz vorkommt, beziehungsweise die Wahrscheinlichkeiten nicht zu extreme Werte nahe 0 oder 1 annehmen.
Für jede Beobachtung wird dann der Prädiktor η mithilfe des festgelegten Koeffizientenvektors β und der schon erzeugten Variablen errechnet:

$$\eta = x'\beta \tag{2.1}$$

Mithilfe der Linkfunktion g wird der Erwartungswert μ anschließend transformiert:

$$g(\mu) = \eta = x'\beta \tag{2.2}$$

Für normal-, poisson- und binomial-verteilte Variablen wird dabei die natürliche Linkfunktion verwendet. Um zu gewährleisten, dass bei gamma-verteilten Variablen nur positive Werte simuliert werden, wird hier der Log-Link angewendet.

Für nominale, ungeordnete Variablen wird ein multinomiales Logit-Modell mit der letzten Kategorie k als Referenz aufgestellt. Die Wahrscheinlichkeit für jede Kategorie (außer der Referenzkategorie) errechnet sich dabei wie folgt:

$$P(y = r|x) = \pi_r = \frac{exp(x'\beta_r)}{1 + \sum_{s=1}^{k-1} exp(x'\beta_s)}, \qquad r = 1, ..., k-1 \tag{2.3}$$

Die Wahrscheinlichkeit für die Referenzkategorie k errechnet sich durch:

$$P(y = k|x) = \pi_k = \frac{1}{1 + \sum_{s=1}^{k-1} exp(x'\beta_s)} \tag{2.4}$$

Für ordinale Variablen wird ein kumulatives Logit-Modell verwendet. Die Wahrscheinlichkeit für Kategorie r oder einer niedrigeren Kategorie errechnet sich dabei durch:

$$P(y \leq r|x) = \frac{exp(\gamma_{0r} + x'\gamma)}{1 + exp(\gamma_{0r} + x'\gamma)}, \qquad r = 1, ..., k-1 \tag{2.5}$$

Daraus lassen sich dann einfach die nicht kumulierten Wahrscheinlichkeiten errechnen:

$$P(y = r|x) = \pi_r = \begin{cases} P(y \leq r|x) & \text{für } r = 1 \\ P(y \leq r|x) - P(y \leq r-1|x) & \text{für } r = 2, ..., k-1 \\ 1 - P(y \leq k-1|x) & \text{für } r = k \end{cases} \tag{2.6}$$

Mithilfe des errechneten Erwartungswertes μ, beziehungsweise der jeweiligen Wahrscheinlichkeiten für die Kategorien, werden nun Zufallszahlen aus der zugrundeliegenden Verteilung der Variablen gezogen. Für jede Beobachtung der Variablen ist dabei der Erwartungswert oder der Wahrscheinlichkeitsvektor unterschiedlich, abhängig von dem errechneten Prädiktor.

Eine Zusammenfassung über die Verteilungstypen, gewählten Linkfunktionen und den schematischen Vorgang der Zufallsziehung in R wie in Kapitel 2.1 beschrieben, wird in folgender Tabelle gegeben:

Verteilung	**Link**		**Zufallsziehung in R**
Normal	Identität:	$\mu = x'\beta$	$y \sim rnorm(x'\beta, \sigma)$
Poisson	Log:	$log(\mu) = x'\beta$	$y \sim rpois(exp(x'\beta))$
Gamma	Log:	$log(\mu) = x'\beta$	$y \sim rgamma(\nu, \frac{exp(x'\beta)}{\nu})$
Binomial	Logit:	$log(\frac{\mu}{1-\mu}) = x'\beta$	$y \sim sample(\pi_1, \pi_0)$
Mulitnomial (nominal)	Logit:	$log(\frac{P(y=r\|x)}{P(y=k\|x)}) = x'\beta_r$	$y \sim sample(\pi_1, ..., \pi_k)$
Multinomial (ordinal)	Logit:	$log(\frac{P(y\leq r\|x)}{P(y>r\|x)}) = \gamma_{0r} + x'\gamma$	$y \sim sample(\pi_1, ..., \pi_k)$

Tabelle 2.2.: Verteilungstypen, zugrundeliegende Linkfunktion und Zufallsziehung in R für die Simulation aus Regressionsmodellen.

Ein Beispiel zur Erzeugung der binomial-verteilten Variable X_2 sei folgendes:
Die normal-verteilte Variable X_1 hat für die erste Beobachtung den Wert $x_{11} = 4.57$.
Die Koeffizienten werden beliebig festgelegt als $\beta_0 = 1.2, \beta_1 = 0.2$.
Der Prädiktor errechnet sich somit zu

$$\eta_1 = \beta_0 + \beta_1 \cdot x_{11} = 1.2 + 0.2 \cdot 4.57 = 2.11 \tag{2.7}$$

Die Wahrscheinlichkeit $P(x_{21} = 1)$ errechnet sich durch Auflösen der Link-Funktion nach μ zu

$$P(x_{21} = 1) = \mu_1 = \frac{exp(\eta_1)}{1 + exp(\eta_1)} = \frac{exp(2.11)}{1 + exp(2.11)} = 0.89 \tag{2.8}$$

Hieraus werden in R nun Zufallszahlen gezogen, dabei gilt

$$P(x_{21} = i) = \begin{cases} 0.89 & \text{für } i = 1 \\ 0.11 & \text{für } i = 0 \end{cases} \tag{2.9}$$

Dieser Vorgang wird anschließend für jede Beobachtung wiederholt, um die Variable X_2 komplett zu erzeugen.

2.2. Umsetzung der Simulation in R

In diesem Kapitel wird kurz auf die Implementierung der Gamma-Verteilung in R eingegangen, da hier eine spezielle Parametrisierung vorliegt. Zusätzlich wurde eine Funktion geschrieben, mit der mithilfe weniger Übergabeparameter die Daten nach dem Algorithmus aus Kapitel 2.1 erzeugt werden können. Die nötigen Übergabeparameter der Funktion werden kurz vorgestellt.

2.2.1. Gamma-Verteilung in R

Da die Gamma-Verteilung in R nicht in der Darstellung der Exponentialfamilie parametrisiert ist, müssen die Übergabeparameter entsprechend angepasst werden. Die Exponentialfamilien-Darstellung ist wie folgt:

$$f(y|\mu,\nu) = \frac{1}{\Gamma(\nu)} \cdot (\frac{\nu}{\mu})^{\nu} \cdot y^{\nu-1} \cdot exp(-\frac{\nu}{\mu} \cdot y) \tag{2.10}$$

Dabei entspricht μ dem Erwartungswert und ν dem Shape-Parameter.
Bei der Parametrisierung in R wird y dagegen bedingt auf den Shape-Parameter a und den Scale-Parameter s dargestellt:

$$f(y|a,s) = \frac{1}{s^a \cdot \Gamma(a)} \cdot y^{a-1} \cdot exp(-\frac{y}{s}) \tag{2.11}$$

Durch Umformung lassen sich die beiden Gleichungen jedoch leicht ineinander überführen, es gilt $a = \nu$ und $s = \frac{\mu}{\nu}$.
So kann in R also wie gewünscht eine gamma-verteilte Variable aus einem generalisierten linearen Modell mit festgelegtem ν und abhängig von dem errechneten Prädiktor $\eta = g(\mu)$ simuliert werden.

2.2.2. Funktion zur Durchführung der Simulation

Zur vereinfachten Umsetzung in R wurde eine Funktion geschrieben, die Daten nach dem Algorithmus aus Kapitel 2.1 erzeugt.
Der Funktion muss zum einen ein Vektor *variable.type* mit dem Typ der jeweiligen Variable übergeben werden, mögliche Angaben sind “normal“, “poisson“, “gamma“, “binomial“, “nominal“ und “ordinal“. Dieser Vektor hat logischerweise dieselbe Länge wie Variablen erzeugt werden sollen.

Desweiteren benötigt die Funktion einen Vektor *variable.cat*, der bei kategorialen Variablen die Anzahl an Kategorien angibt und für metrische Variablen den Eintrag NA enthält.
Der dritte Übergabeparameter *first.param* enthält die nötigen Informationen zur Erzeugung der ersten Variablen, vergleiche Tabelle 2.1.
Zur Erzeugung von normal- und gamma-verteilten Variablen wird die Standardabweichung σ beziehungsweise der Shape-Parameter ν benötigt. Dazu wird der Funktion ein Vektor *sigma* übergeben, der an der Position solcher Variablen eine Zahl, sonst den Eintrag NA enthält.
Desweiteren benötigt die Funktion eine Liste *coeff.list*, in der jeder Listeneintrag den Koeffizienten zur Erzeugung einer Variablen entspricht. Diese Koeffizienten sind wie schon angesprochen frei wählbar, sollten aber sinnvoll sein, um realitätsnahe Werte erzeugen zu können. Für normal-, poisson-, gamma- und binomial-verteile Variablen entspricht dieser Eintrag einem Vektor, für nominal- und ordinal-skalierte Daten einer Matrix mit $k-1$ Zeilen.
Der Übergabeparameter n gibt die Anzahl an Beobachtung im Datensatz an.
Falls gewünscht kann der Funktion zur Reproduzierbarkeit noch ein *seed* übergeben werden, falls nicht wird dieser auf NA gesetzt.

2.3. Datensätze

Zum Testen und Vergleichen der Imputationsmethoden wurden zwei Datensätze nach dem Algorithmus aus Kapitel 2.1 und mithilfe der Funktion aus Kapitel 2.2.2 erzeugt.
Der erste, kleinere Datensatz hat zehn Variablen mit je 1000 Beobachtungen, der zweite Datensatz hat 20 Variablen mit je 1000 Beobachtungen.

Im Datensatz aufgenommen wurden dabei schlussendlich nur normal-, poisson-, gamma- und binomial-verteilte Variablen, da kategoriale Variablen bei der Imputation zu Problemen und letztendlich zum Funktionsabbruch führten. Die Instabilität multinomialer Modelle ist ein bekanntes Problem, um Ergebnisse zu erhalten wurden diese also rausgelassen.

Im kleineren Datensatz sind die 10 Variablen wie folgt verteilt:

X_1	X_2	X_3	X_4	X_5
normal	binomial	gamma	normal	poisson

X_6	X_7	X_8	X_9	X_{10}
binomial	poisson	gamma	binomial	normal

Im größeren Datensatz sind die 20 Variablen folgendermaßen verteilt:

X_1	X_2	X_3	X_4	X_5
binomial	normal	gamma	poisson	binomial

X_6	X_7	X_8	X_9	X_{10}
normal	gamma	normal	poisson	gamma

X_{11}	X_{12}	X_{13}	X_{14}	X_{15}
binomial	poisson	normal	normal	gamma

X_{16}	X_{17}	X_{18}	X_{19}	X_{20}
poisson	binomial	gamma	poisson	binomial

3. Fehlende Werte

Um später die Imputationsmethoden testen zu können, müssen in den simulierten Datensätzen zuerst Werte entfernt werden. Dafür gibt es verschiedene Ansätze und Methoden, die im Folgenden dargestellt werden.

3.1. Klassifikation fehlender Werte

Eine anerkannte und häufig verwendete Klassifikation von fehlenden Werten geht auf Donald B. Rubin zurück. Danach kann der Mechanismus, durch den fehlende Werte entstehen (sogenannter Missingmechanismus), in drei Gruppen eingeteilt werden. Dies wird in (Spiess; 2008) genauer beschrieben.

Missing completely at random (MCAR)

Unter der MCAR-Annahme ist ein beobachtetes Fehlermuster sowohl unabhängig von den beobachteten Daten D^{obs} als auch von den unbeobachteten Daten D^{mis}. Anders ausgedrückt unterliegen fehlende Daten also der MCAR-Annahme, falls

$$p(M|D) = P(M) \tag{3.1}$$

M ist dabei die Fehlermatrix, mit Einträgen $m_{ij} = 1$ falls $d_{ij} \in D^{mis}$ und $m_{ij} = 0$ sonst. Das Fehlen einer Beobachtung unterliegt also komplett dem Zufall. Würde man in einem Datensatz also komplett beliebig und unabhängig von anderen Variablen Beobachtungen löschen, wäre die MCAR-Annahme erfüllt.
Insgesamt ist MCAR der unproblematischste Fehler-Mechanismus, unter dem keine Verzerrung der wahren Daten entsteht.

Missing at random (MAR)

Unter der MAR-Annahme ist ein beobachtetes Fehlermuster zwar wie bei MCAR unabhängig von den unbeobachteten Werten D^{mis}, jedoch abhängig von den beobachteten

Werten D^{obs}:

$$p(M|D) = p(M|D^{obs}) \tag{3.2}$$

Das Fehlermuster ist unter MAR also abhängig von anderen Variablen, beispielsweise wenn die Angabe des Einkommens vom Alter einer bestimmten Person abhängt. Vernachlässigt man die fehlenden Werte, wird das Gesamtbild bei der Betrachtung zwar verzerrt, die wahre Regressionsbeziehung bleibt unter MAR jedoch erhalten.
MCAR und MAR sind zufällige (at random) Fehlermuster, und werden oft als Voraussetzung für Methoden zur multiplen Imputation wie beispielsweise bei *Amelia* benötigt.

Not missing at random (NMAR)

Unter der NMAR-Annahme ist ein beobachtetes Fehlermuster sowohl von D^{obs} als auch von D^{mis} abhängig, das Fehlermuster ist also nicht zufällig. Dies trifft beispielsweise zu, falls häufiger die Angaben von Personen mit hohem Einkommen fehlen. Die Daten sowie die Regressionsbeziehung werden bei NMAR verzerrt dargestellt.

3.2. Erzeugen der fehlenden Werte

In dieser Arbeit werden die fehlenden Werte so erzeugt, dass die MAR-Annahme erfüllt ist. Dafür bleibt die zuletzt erzeugte Variable, welche beim Durchführen der Regression nach der Imputation die abhängige Y-Variable darstellt, vollständig. Die Wahrscheinlichkeit, dass eine Beobachtung einer unabhängigen Variable fehlt, ist immer abhängig von der Y-Variablen.
Dabei wird folgende Formel verwendet:

$$P(x_{ij} = NA) = 1 - \frac{1}{(\alpha_j \cdot y_i)^2 + \beta_j}, \qquad i = 1, ..., n,\ j = 1, ..., p \tag{3.3}$$

n entspricht dabei der Anzahl an Beobachtungen im Datensatz und p der Anzahl an Variablen.
Dabei muss darauf geachtet werden, dass keine negativen Wahrscheinlichkeiten errechnet werden. Unter der Bedingung $\beta_j \geq 1\ \forall j$ ist dieses Problem sicher behoben.

Insgesamt sind die Missing-Wahrscheinlichkeiten für alle Beobachtungen einer Variablen immer gleich, von Variable zu Variable jedoch unterschiedlich. Für den Datensatz mit zehn Variablen existieren also $\alpha_1, ..., \alpha_{10}$ und $\beta_1, ..., \beta_{10}$.

3.3. Mittlere Fehlerraten in den Datensätzen

Die Imputationsmethoden werden für beide Datensätze jeweils einmal für geringere Fehlerraten und einmal mit etwas höheren Fehlerraten durchgeführt. Bei der Erzeugung fehlender Werte liegt die mittlere Fehlerrate für beide Datensätze also jeweils einmal knapp unter 10 % und einmal knapp unter 20 %. Viel höhere Fehlerraten sind im Allgemeinen kritisch zu betrachten und werden deswegen nicht getestet.

Ein Problem bei zu hoher Fehlerrate, das beispielsweise bei der Anwendung von *Amelia* auftreten kann, betrifft kategoriale Variablen. Mit steigender Anzahl an fehlenden Werten sinken logischerweise die Ausprägungen pro Kategorie. Wie später in Kapitel 5.1 genauer beschrieben wird, verwendet *Amelia* Bootstrapping, das heißt es werden mit Zurücklegen Stichproben mit gleichem Umfang aus dem ursprünglichen Datensatz gezogen. Dadurch kann es also vorkommen, dass eine bestimmte Ausprägung der kategorialen Variable gar nicht in der Bootstrap-Stichprobe vorkommt. Dies führt beispielsweise dazu, dass von einer kategoriale Variable mit drei Ausprägungen in der Bootstrap-Stichprobe nur zwei Ausprägungen existieren. Dadurch kommt es in *Amelia* zu einem Problem bei der Imputation und zum Funktionsabbruch.
Für geringere Fehlerraten kann dieses Problem rein theoretisch natürlich ebenso auftreten, die Wahrscheinlichkeit ist jedoch viel geringer.

Die verschiedenen Imputationsmethoden werden jeweils 500 mal durchlaufen, wobei jede Runde die fehlenden Werte mit selber Wahrscheinlichkeit neu erzeugt werden. Der genaue Ablauf wird in Kapitel 4 noch näher erklärt. Dabei werden in jeder Runde die Anzahl an fehlenden Werten im Datensatz abgespeichert, um schlussendlich einen Überblick über die mittlere Fehlerrate zu bekommen.

Für den kleineren Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % für jede Variable ergibt sich folgendes Bild:

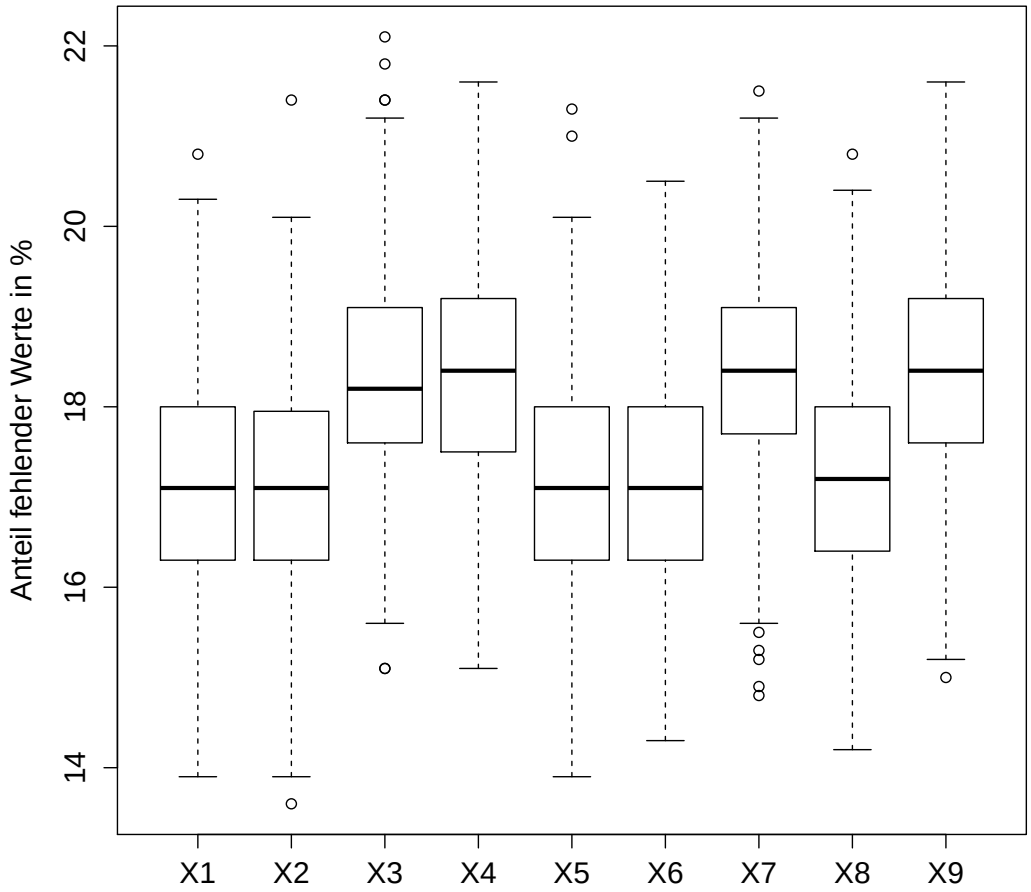

Abbildung 3.1.: Übersicht über den Anteil fehlender Werte pro Variable in jeder Runde. Es wird der Datensatz mit zehn Variablen betrachtet, wobei die mittlere Fehlerrate knapp unter 20 % liegt.

Die minimale Fehlerrate aus allen Durchgängen liegt bei 13.6 % (Variable X_2), die maximale bei 22.1% (Variable X_3). Die Mittelwerte der Fehlerraten über alle Durchgänge liegen zwischen 17.1 % und 18.5 %, der Wertebereich der Mediane ist sehr ähnlich, wie in Abbildung 3.1 zu erkennen ist. Die Werte weisen für alle Variablen eine ähnliche Spannweite auf, es existieren keine extremen Ausreißer.

Für eine geringere mittlere Fehlerrate knapp unter 10 % ergibt sich folgende Abbildung:

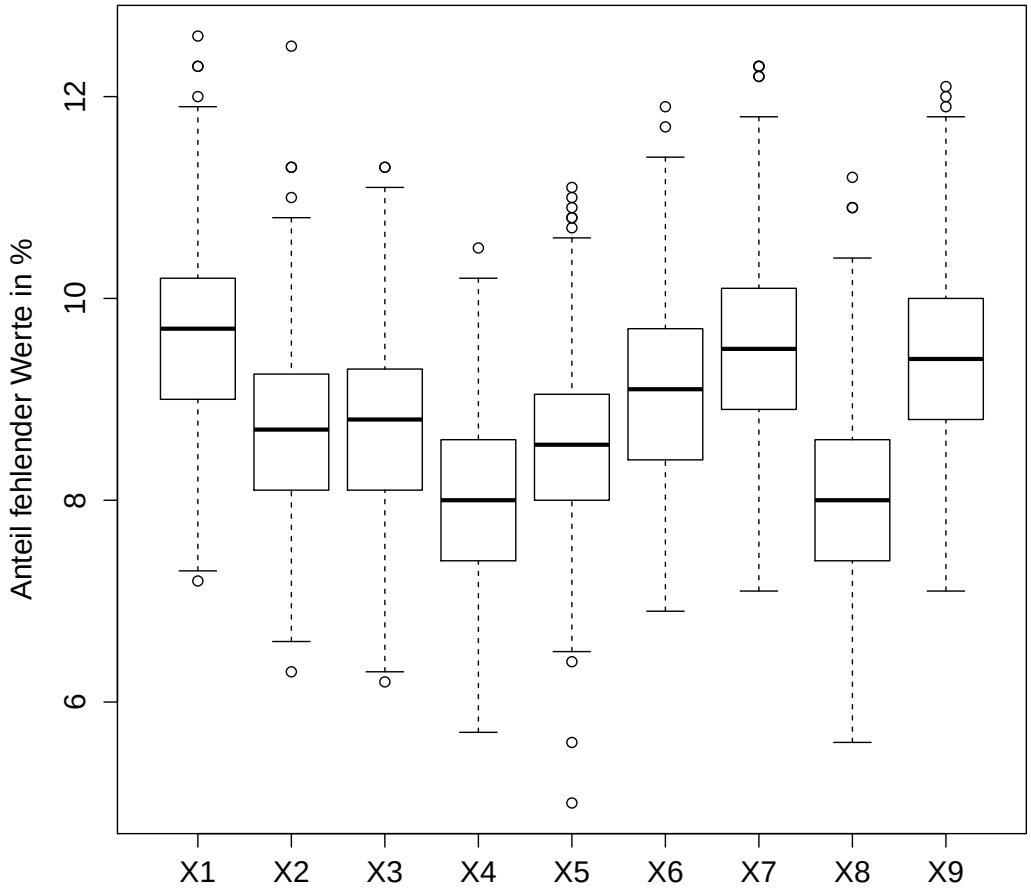

Abbildung 3.2.: Übersicht über den Anteil fehlender Werte pro Variable in jeder Runde. Es wird der Datensatz mit zehn Variablen betrachtet, wobei die mittlere Fehlerrate knapp unter 10 % liegt.

Hier liegt der Wertebereich zwischen einem minimalen Anteil fehlender Werte von 5 % (Variable X_5) und einem maximalen Anteil von 12.6 % (Variable X_1). Die Mittelwerte der Fehlerraten liegen zwischen 7.9 % und 9.7 %, die Mediane liegen erneut in einem ähnlichen Wertebereich. Auch in Abbildung 3.2 ist zu erkennen, dass die Anteile fehlender Daten über alle Variablen hinweg eine ähnliche Spannweite aufweisen und auch hier keine extremen Ausreißer vorhanden sind.

Die mittleren Fehlerraten für den größeren Datensatz liegen erneut knapp unter 10 % und knapp unter 20 %, die Abbildungen befinden sich im Anhang.

4. Imputation fehlender Daten

Um mit unvollständigen Datensätzen Analysen durchzuführen gibt es verschiedene Ansätze. Beispielsweise gibt es die sogenannte **Complete Case Analyse**, ein **Ad-hoc-Verfahren**, bei dem alle Beobachtungen mit einem oder mehreren fehlenden Werten in einer beliebigen Variable nicht in die Analyse mit einbezogen werden („listwise deletion“). Nur in allen Variablen vollständige Beobachtungen werden also berücksichtigt. Für Datensätze mit vielen fehlenden Werten ist dieses Vorgehen problematisch, da es einen hohen Informationsverlust zur Folge hat. Würde theoretisch für jede Beobachtung nur die Angabe einer Variable von vielen fehlen, würde für die Auswertung keine Beobachtung mehr übrig bleiben. Ebenso ist ein MCAR-Fehlermuster zwar unproblematisch, falls dies jedoch nicht gilt kommt es meist zu einer Verzerrung der Schätzer.
Sinnvoller ist teilweise die sogenannte **Available Case Analyse**, bei der alle Beobachtungen verwendet werden, die für die interessierende Variable einer Auswertung vollständig sind. Auch hier existiert jedoch die Problematik der verzerrten Schätzung, falls die MCAR-Annahme nicht zutrifft.
Um diese Verzerrungen zu vermeiden, ist es manchmal sinnvoll, die fehlenden Werte durch möglichst plausible Werte zu ersetzen. Dieses Vorgehen wird auch als Imputation bezeichnet, die möglichen Methoden werden in diesem Kapitel dargestellt. Die Grundlagen zu diesem Kapitel sind aus (Spiess; 2008) und können dort nachgelesen werden.

4.1. Einfache Imputationsverfahren

Bei einfachen Imputationsverfahren wird für jeden fehlenden Wert genau eine Imputation erzeugt. Ein paar mögliche Methoden lauten wie folgt:

- **Mittelwertsimputation**
  Dabei wird für jeden fehlenden Wert das arithmetische Mittel der beobachteten Werte der Variablen eingesetzt. Bei nicht-metrischen Daten kann alternativ auch der Median oder Modus imputiert werden. Der Variablen-Mittelwert (beziehungsweise -Modus oder -Median) bleibt dabei gleich, die Varianz wird jedoch

unterschätzt ebenso wie die Kovarianz mit einer anderen Variablen. Außerdem bleibt die Problematik der verzerrten Schätzung bestehen, oft sogar selbst unter der MCAR-Annahme.

- **Regressionsimputation**
  Dabei wird der fehlende Wert durch den Vorhersagewert eines Regressionsmodells auf Basis der beobachteten Werte anderer Variablen ersetzt. Eine konsistente Schätzung von Erwartungswerten ist mit dieser Methode unter schwachen Annahmen möglich, unter anderem muss die MCAR- oder MAR-Annahme erfüllt sein. Zur Schätzung von Varianzen und Kovarianzen hingegen müssen Korrekturen vorgenommen werden, da diese sonst unterschätzt werden.

- **Hot Deck Imputation**
  Bei der Mittelwerts- und Regressionsimputation können Werte geschätzt werden, die außerhalb des Wertebereichs der wahren Daten liegen. Die Hot Deck Imputation ist eine alternative Imputationsmethode, bei der dieses Problem nicht auftreten kann. Dabei werden fehlende Werte durch in den Daten tatsächlich beobachtete Werte ersetzt. Eine Möglichkeit hierfür ist die „Random Overall"-Imputation, bei der absolut zufällig mit oder ohne Zurücklegen beziehungsweise mithilfe spezieller Ziehungsdesigns ein Wert aus den beobachteten Daten ausgewählt wird. Dieses Verfahren führt nur zu einer konsistenten Schätzung, falls die MCAR-Annahme zutrifft.

- **Cold Deck Imputation**
  Diese ist sehr ähnlich zur Hot Deck Imputation, nur werden die Werte aus denen gezogen wird nicht aus den wahren Daten, sondern aus anderen Datensätzen oder Quellen gewonnen.

Bei den meisten einfachen Imputationsmethoden, außer der stochastischen Regressionsimputation, wird die Unsicherheit in den Daten nicht angemessen berücksichtigt. Dies führt unter anderem dazu, dass die wahre Varianz in den Daten unterschätzt wird.

4.2. Multiple Imputationsverfahren

Im Gegensatz zur einfachen Imputation, bei der für jeden fehlenden Wert nur eine Imputation erzeugt wird, werden bei der multiplen Imputation für jeden Wert mehrere

Imputationen erzeugt. Dadurch entstehen mehrere vollständige Exemplare des ursprünglich unvollständigen Datensatzes, wobei die beobachteten Werte jeweils gleich sind. Das Verfahren lässt sich in drei Schritten darstellen:

- **1. Imputation:**
  Im ersten Schritt werden mithilfe eines ausgewählten Verfahrens m imputierte Datensätze erstellt. Dabei entspricht m der Anzahl an Werten, die für ein fehlendes Feld erzeugt werden sollen.

- **2. Analyse:**
  Im zweiten Schritt werden die Datensätze einzeln analysiert, wodurch m Auswertungen entstehen.

- **3. Kombination:**
  Im letzten Schritt werden die Einzelergebnisse zu einem Gesamtergebnis zusammengefasst. Für Q, eine beliebige statistische Größe von Interesse, kann man dabei die separaten Schätzer q_j $(j = 1, ..., m)$ beispielsweise durch den Mittelwert zu einem Gesamtergebnis zusammenfassen:

$$\bar{q} = \frac{1}{m} \sum_{j=1}^{m} q_j \tag{4.1}$$

Multiple Imputationsmethoden haben ein paar Vorteile gegenüber den einfachen. Zum einen wird durch das mehrmalige Schätzen die Unsicherheit in den Daten berücksichtigt und als Konsequenz daraus die wahre Varianz der Daten besser abgebildet. Zusätzlich sind die Ergebnisse aus multipler Imputation der Erfahrung nach meistens besser als die Ergebnisse aus einfacher Imputation.

4.3. Umsetzung in R

Folgend wird auf den genauen Ablauf der Imputationen und die Gewinnung der Auswertungen eingegangen. Zu Beginn steht der vollständige Datensatz, simuliert wie in Kapitel 2.1 beschrieben. In diesem Datensatz werden anschließend fehlende Werte erzeugt. Dies funktioniert wie in Kapitel 3.2 erklärt, also mit einer Wahrscheinlichkeit abhängig von der zuletzt erzeugten Variablen, die als einzige vollständig bleibt. Dieser Datensatz mit den fehlenden Werten wird anschließend auf mehreren Wegen imputiert, die zugrundeliegenden R-Pakete und Methoden werden in Kapitel 5, 6 und 7 genauer vorgestellt.

Durch diesen Schritt entsteht ein vollständiger Datensatz für jede einfache Imputationsmethode, für jede multiple Imputationsmethode entstehen je nach Angabe m Stück. Die Qualität der Imputation soll letztendlich dahingehend getestet werden, wie gut der Zusammenhang in den imputierten Daten dem wahren Zusammenhang angenähert wird. Deswegen wird mithilfe jedes Datensatzes ein Regressionsmodell berechnet, in dem die letzte Variable (X_{10} im Falle des kleineren Datensatzes, X_{20} im Falle des größeren) die abhängige Größe ist und alle anderen Variablen als Einflussgrößen aufgenommen werden. Für die multiplen Imputationsverfahren werden die m Schätzer anschließend zu einem Gesamtergebnis zusammengefasst. Für jede Imputationsmethode wird letztendlich ein Parametervektor $\beta = (\beta_0, ..., \beta_p)$ in einer Ergebnismatrix abgespeichert.
Dieser Vorgang wird anschließend 500 mal wiederholt, mit der einzigen Änderung, dass die abhängige Variable des Modells vor der Erzeugung fehlender Werte neu simuliert wird. Durch diese erneute Simulation werden mithilfe der aus dem Prädiktor η errechneten Erwartungswerte in jeder Runde neue Zufallszahlen gezogen. Dadurch kann insgesamt ein besseres Abbild der wahren Situation dargestellt werden. Die Erzeugung erfolgt dabei genau wie bei der Simulation des Datensatzes, also auf Grundlage eines Regressionsmodells mit allen anderen Variablen als Einflussgrößen, wobei genau die selben Koeffizienten verwendet werden.
Aus jedem Schleifendurchgang resultiert ein Parametervektor für jede Imputationsmethode, dieser wird in der jeweiligen Ergebnismatrix abgespeichert. Für jedes $\beta_0, ..., \beta_p$ für jede Imputationsmethode resultieren also 500 Schätzer, die zusammengefasst in einem Boxplot dargestellt werden. Der Intercept wird dabei herausgelassen, da dieser für die Interpretation eher unwichtig ist. Die wahren Koeffizienten aus der Simulation werden jeweils mit eingezeichnet und dienen zum Vergleich.

Der schematische Ablauf eines Schleifendurchganges ist im Folgenden noch einmal vereinfacht dargestellt:

vollständiger Datensatz

$\downarrow$

Erzeugen von fehlenden Werten

$\downarrow$

unvollständiger Datensatz

$\downarrow$

Imputation

Methode 1 Methode 2 ... Methode l

$\downarrow$

ein vollständiger Datensatz bei einfacher Imputation

m vollständige Datensätze bei multipler Imputation

$\downarrow$

jeweils Aufstellen einer Regressionsgleichung,

letzte Variable im Datensatz ist die abhängige

$\downarrow$

Zusammenfassen der Ergebnisse bei multipler Imputation

$\downarrow$

Modell 1 Modell 2 ... Modell l

$\downarrow$ $\downarrow$ $\downarrow$

$\hat{\beta}_{Methode1}$ $\hat{\beta}_{Methode2}$ $\cdots$ $\hat{\beta}_{Methode_l}$

Desweiteren werden verschieden hohe Fehlerraten an unterschiedlichen Datensätzen getestet. Deshalb werden schlussendlich vier dieser Auswertungen mit jeweils 500 Durchgängen ausgeführt. Diese sind die möglichen Verknüpfungen aus

- Anteil fehlender Werte knapp unter 10 % beziehungsweise knapp unter 20 %
- Datensatz mit zehn Variablen beziehungsweise Datensatz mit 20 Variablen

5. Imputation mit Amelia II

5.1. Theorie

Amelia ist ein R-Paket zur multiplen Imputation, es werden für jeden fehlenden Wert also mehrere imputierte Werte erzeugt.
Es wird der sogenannte EMB-Algorithmus (expectation-maximization with bootstrapping) verwendet. Hierbei wird der EM-Algorithmus auf mehrere durch Bootstrapping ermittelte Datensätze, gezogen aus dem ursprünglichen, unvollständigen Datensatz, angewendet. Die fehlenden Werte in den Datensätzen werden daraufhin durch die gezogenen Imputationen ersetzt.
Die zugrundeliegenden Annahmen, der Algorithmus und die Durchführung in R sind entnommen aus (Honaker et al.; 2011) und werden dort genauer beschrieben.

5.1.1. Annahmen

Das Annahme-Modell unter *Amelia* besagt, dass die kompletten Daten multivariat normalverteilt sind mit Mittelwertsvektor μ und Kovarianzmatrix Σ:

$$D \sim N_k(\mu, \Sigma) \tag{5.1}$$

Auch wenn diese Annahme für viele Daten nicht immer sinnvoll erscheint, ermöglichen verschiedene Variablentransformationen eine Annäherung an diese Voraussetzung.
Desweiteren wird die sogenannte MAR-Annahme (missing at random) getroffen, diese wurde in Kapitel 3.1 genauer beschrieben.
Auch der speziellere Fall, die sogenannte MCAR-Annahme (missing completely at random), ist natürlich ausreichend.

5.1.2. Algorithmus

Eine schematischer Ablauf der Imputation ist in folgender Grafik dargestellt:

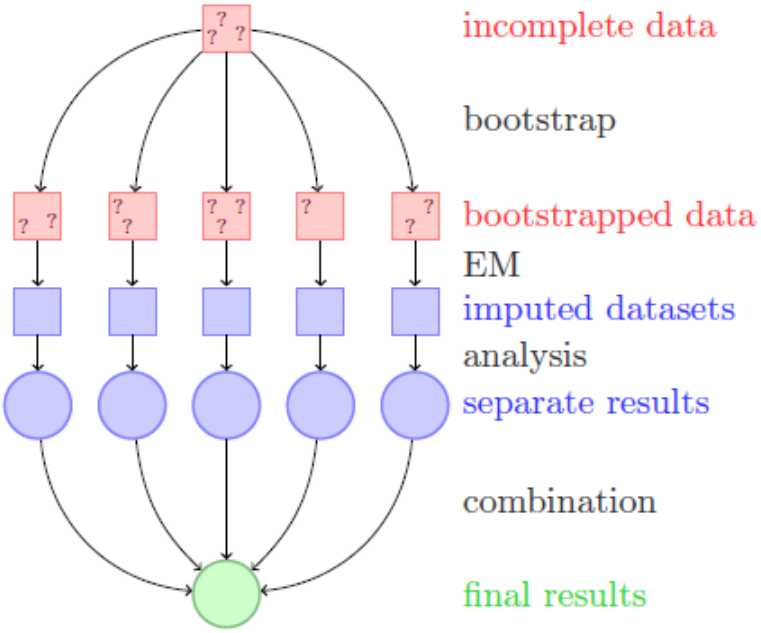

Abbildung 5.1.: Schematische Darstellung der Imputation mit *Amelia* mithilfe des EMB-Algorithmus aus (Honaker et al.; 2011).

Am Anfang steht der unvollständige Datensatz D, der sich zusammensetzt aus den beobachteten Daten D^{obs} und den fehlenden Daten D^{mis}.

Anschließend wird Bootstrapping angewandt, um die Unsicherheit der Schätzung nachzubilden.
Dabei wird n-mal (wobei n der Anzahl an Beobachtungen im Datensatz entspricht) mit Zurücklegen aus dem ursprünglichen Datensatz gezogen, wodurch eine Stichprobe des Datensatzes entsteht. Eine Beobachtung kann also einmal, mehrmals oder gar nicht in der Bootstrap-Stichprobe vorkommen. (Heumann und Schmid; 2013)
Durch mehrmaliges Durchführen von Bootstrapping erhält man mehrere Stichproben des Datensatzes, wie in Abbildung 5.1 zu sehen ist.

Im nächsten Schritt sollen aus der Posteriori Werte gezogen werden, um damit schlussendlich abhängig von den gezogenen Parameterschätzern und von D^{obs} die unvollständigen Bootstrap-Stichproben zu imputieren. Die Posteriori lässt sich aus folgenden Schrit-

ten errechnen:
Die Likelihood der beobachteten Daten D^{obs} ist $p(D^{obs}, M|\theta)$, wobei $\theta = (\mu, \Sigma)$ und M der Fehlermatrix entspricht wie in Kapitel 3.1 definiert, also mit den Einträgen $m_{ij} = 1$ falls $d_{ij} \in D^{mis}$ und $m_{ij} = 0$ sonst. Unter der MAR-Annahme (und der weiteren Annahme, dass M nicht von θ abhängt) gilt:

$$p(D^{obs}, M|\theta) = p(M|D^{obs}) \cdot p(D^{obs}|\theta) \tag{5.2}$$

Da nur die Inferenz der Parameter der kompletten Daten von Interesse ist, lässt sich die Likelihood auch darstellen als

$$L(\theta|D^{obs}) \propto p(D^{obs}|\theta) \tag{5.3}$$

Mit dem „Satz vom iterierten Erwartungswert“ kann das wiederum umgeschrieben werden zu

$$p(D^{obs}|\theta) = \int p(D|\theta)\mathrm{d}D^{mis} \tag{5.4}$$

Durch diese Likelihood, verbunden mit der flachen Priori von θ (dabei handelt es sich um die nicht-informative Gleichverteilungspriori), ergibt sich die Posteriori zu

$$p(\theta|D^{obs}) \propto p(D^{obs}|\theta) = \int p(D|\theta)\mathrm{d}D^{mis} \tag{5.5}$$

Um aus dieser Posteriori Werte zu ziehen und mithilfe des damit erhaltenen Parameterschätzers $\hat{\theta}$ und den beobachteten Daten D^{obs} die fehlenden Werte D^{mis} zu vervollständigen, wird nun der EM-Algorithmus angewendet. Dieser wird genauer beschrieben in (Honaker und King; 2010).
Der EM-Algorithmus setzt sich zusammen aus dem Estimation-Schritt (E-Schritt) und dem Maximization-Schritt (M-Schritt). Im E-Schritt werden die fehlenden Daten D^{mis} mithilfe der beobachteten Werte D^{obs} und des Parameters $\hat{\theta}$, bei dem es sich um einen Schätzer auf Grundlage der letzten Imputation handelt, aufgefüllt. Für den ersten Durchgang wird dabei für den Parameter θ ein zufälliger Startwert generiert, da noch keine aktuelle Imputation vorhanden ist. Im M-Schritt wird anschließend der Parameter des Modells auf Grundlage der neuen Imputation mit der Maximum-Likelihood-Methode geschätzt. Der Algorithmus iteriert so lange zwischen dem E-Schritt und dem M-Schritt bis Konvergenz eintritt, also bis sich der Parameterschätzer $\hat{\theta}$ im Vergleich zum vorherigen Durchgang nur noch minimal verändert.

Mithilfe der Funktion *zelig* aus dem R-Paket *Zelig* (Imai et al.; 2015) können nun die fehlenden Schritte aus Abbildung 5.1 einfach durchgeführt werden. Mithilfe der vollständigen, imputierten Datensätze werden Analysen, in diesem Falle eine Regression, durchgeführt und die Einzelergebnisse zu einem Gesamtergebnis kombiniert.

5.2. Umsetzung in R

Für die Durchführung der multiplen Imputation mit dem R-Paket *Amelia* muss der Funktion der unvollständige Datensatz, die gewünschte Anzahl an Imputationen m und die Information, bei welchen Variablen es sich um nominale beziehungsweise ordinale Variablen handelt, übergeben werden.
Desweiteren gibt es die Möglichkeit, die Imputationen einer Variablen auf einen bestimmten Wertebereich zu beschränken. Dies würde beispielsweise verhindern, dass für eine gamma-verteilte Variable negative Werte imputiert werden können. In (Honaker et al.; 2011) wird jedoch empfohlen, auf das Verwenden dieser Restriktionen zu verzichten, da durch das Überschreiten der logischen Beschränkung einer Variablen ein Teil der Unsicherheit beim Imputieren wiedergespiegelt wird. Da die Werte der Variablen selbst hier letztendlich nicht weiter interessant sind, sondern nur die Regressionskoeffizienten betrachtet werden, wird auf diese Einschränkung verzichtet.
Für den größeren Datensatz mit einem Anteil fehlender Werte knapp unter 20 % war es nötig, den Befehl *incheck=FALSE* einzufügen. Dadurch werden die Übergabeparameter der Funktion vor der Imputation nicht überprüft. Ohne diesen Befehl resultierte eine Fehlermeldung, nach der mehrere Variablen im Datensatz perfekt kollinear zu anderen Variablen seien. Auch mithilfe des Übergabeparameters *empri*, durch den die Kovarianz zwischen den Variablen gesenkt wird, konnte dieses Problem nicht behoben werden. Wie in Kapitel 5.3.2 noch gezeigt wird, scheint die Güte der Imputation darunter jedoch nicht zu leiden.
Es sind noch einige weitere Übergabeparameter vorhanden, welche eine bessere Anpassung an verschiedene Datengrundlagen ermöglichen. Diese sind für die betrachteten Datensätze jedoch nicht von Relevanz und werden deswegen weggelassen, können aber in (Honaker et al.; 2011) nachgelesen werden.

Die Funktion *zelig* aus dem R-Paket *Zelig* benötigt als Übergabeparameter wiederum die m imputierten Datensätze, den Prädiktor der Regressionsgleichung und die Art der Regression.

Mögliche Angaben für das Regressionsmodell sind nach (Owen et al.; 2013) folgende:

„model“ in R	**Regression**	**Skalierung der abhängigen Variable**
gamma	Gamma	stetig, positiv
logit	Binomial (Logit)	dichotom
ls	linear (KQ-Methode)	stetig
negbinom	Negativ Binomial	Zähldaten
normal	linear (ML-Methode)	stetig
poisson	Poisson	Zähldaten
probit	Binomial (Probit)	dichotom

Tabelle 5.1.: Liste möglicher Angaben für das Regressionsmodell bei der Funktion *zelig*.

Die Methoden *ls* und *normal* unterscheiden sich dabei lediglich in der Schätzung für den Parameter σ.
Die Theorie zu den generalisierten linearen Modellen kann nachgelesen werden in (Fahrmeir et al.; 2009).

5.3. Ergebnisse

Die Ergebnisse nach der Imputation mit *Amelia* werden getrennt nach den beiden Datensätzen und für verschiedene Fehlerraten dargestellt.

5.3.1. Kleinerer Datensatz

Für einen Anteil fehlender Daten knapp unter 10 % werden die 500 Schätzungen der Regressionskoeffizienten in einem Boxplot dargestellt:

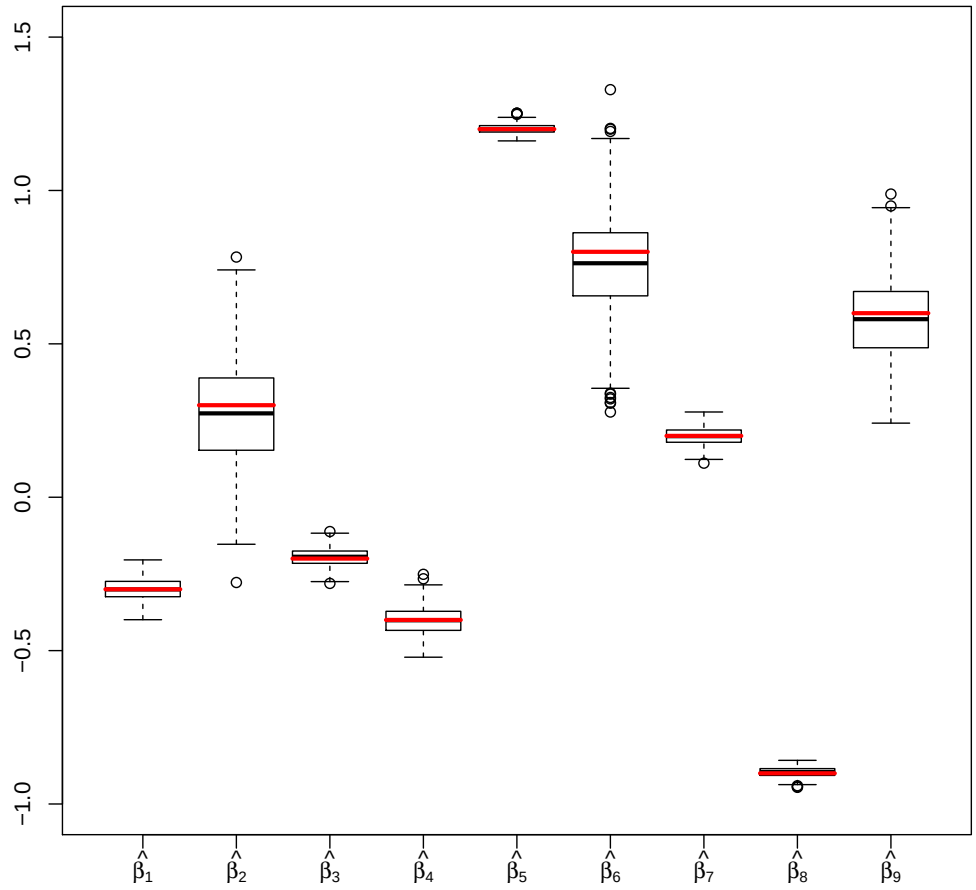

Abbildung 5.2.: Übersicht über die geschätzten Regressionskoeffizienten nach der Imputation mit *Amelia* aus 500 Durchgängen. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 10 % betrachtet.

Sehr auffällig ist hierbei, dass die Koeffizienten für die binomial-verteilten Variablen X_2, X_6 und X_9 viel mehr streuen als die Koeffizienten für numerische Variablen. Insgesamt lässt sich jedoch erkennen, dass der Median der Schätzwerte in allen Fällen ziemlich nah am wahren Koeffizienten liegt. Auch existieren für keinen Koeffizienten extreme Ausreißer.

Für einen höheren Anteil fehlender Daten knapp unter 20 % ergibt sich ein ähnliches Bild:

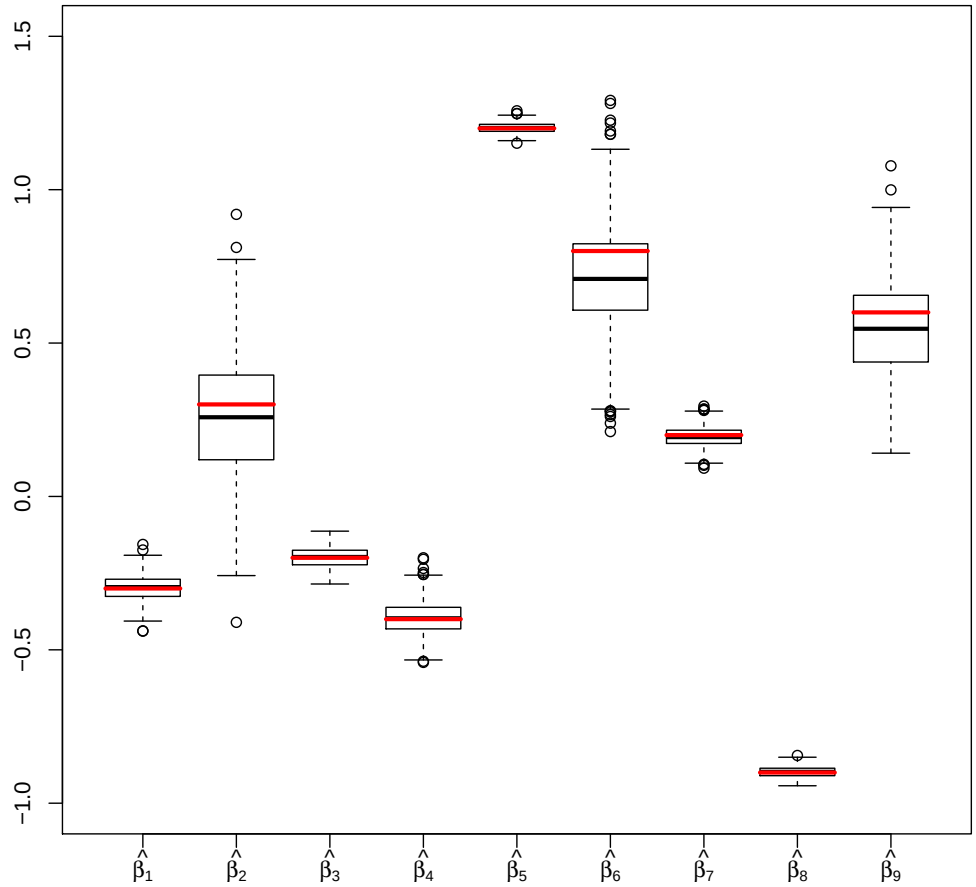

Abbildung 5.3.: Übersicht über die geschätzten Regressionskoeffizienten nach der Imputation mit *Amelia* aus 500 Durchgängen. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet.

Auch in Abbildung 5.3 streuen die Schätzer der binomial-verteilten Variablen weit mehr als die Schätzer der numerischen Variablen. Während bei geringerer Fehlerrate in Abbildung 5.2 die Koeffizienten der dichotomen Variablen jedoch tendenziell nur leicht unterschätzt werden, ist diese Tendenz für eine höhere Fehlerrate schon deutlicher zu erkennen. Auch ist insgesamt zu sehen, dass die Spannweite der Schätzer für alle Koeffizienten größer ist als bei der geringeren Fehlerrate.

5.3.2. Größerer Datensatz

Für den Datensatz mit 20 Variablen sowie einem Anteil fehlender Daten knapp unter 20 % werden die 500 Schätzungen der Regressionskoeffizienten erneut in einem Boxplot dargestellt:

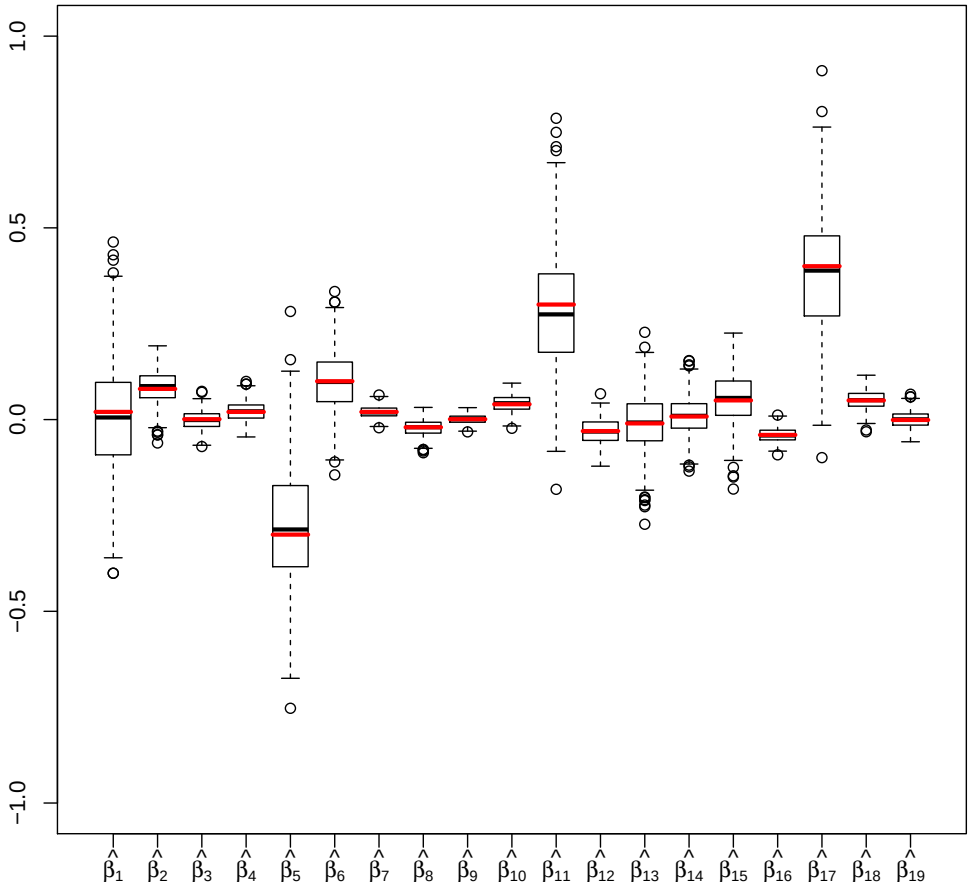

Abbildung 5.4.: Übersicht über die geschätzten Regressionskoeffizienten nach der Imputation mit *Amelia* aus 500 Durchgängen. Es wird der Datensatz mit 20 Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet.

Auch in Abbildung 5.4 ist deutlich zu erkennen, dass die Koeffizientenschätzer der binomial-verteilten Variablen X_1, X_5, X_{11} und X_{17} eine deutlich höhere Spannweite und auch Varianz aufweisen als die Koeffizientenschätzer aller numerischen Variablen. Der

wahre Koeffizient wird jedoch erneut für alle Schätzer im Mittel relativ gut angenähert, der Median entspricht meistens etwa dem wahren Koeffizienten. Ebenso existieren erneut keine extremen Ausreißer.

Für die geringere Fehlerrate ergibt sich ein ähnliches Bild, wobei die Koeffizienten analog zu Kapitel 5.3.1 etwas besser angenähert werden sowie die Spannweite der Schätzer tendenziell geringer ist. Die zugehörige Grafik befindet sich im Anhang.

6. Imputation mit mice

6.1. Theorie

mice ist ein R-Paket zur multiplen Datenimputation. Hier werden, ebenso wie bei *Amelia*, für jeden fehlenden Wert mehrere imputierte Werte erzeugt. *mice* steht dabei für „multivariate imputation by chained equations“, auch bekannt als FCS (fully conditional specification).
Eine bekannte Ansatzweise bei multipler Imputation ist das sogenannte „joint modeling“, bei dem die multivariate Verteilung der fehlenden Daten spezifiziert wird und anschließend mithilfe von MCMC-Verfahren (Markov Chain Monte Carlo) aus den bedingten Verteilungen Imputationen gezogen werden. Dieses Verfahren ist sinnvoll, falls die spezifizierte multivariate Verteilung die Daten gut beschreibt. Kann jedoch keine passende multivariate Verteilung gefunden werden, ist die Verwendung von *mice* eine mögliche Alternative. Dabei wird für jede unvollständige Variable die bedingte Verteilung definiert, welche auf einem univariaten Regressionsmodell basiert. Dann werden mithilfe des FCS-Algorithmus Imputationen erzeugt, wobei wiederholt mithilfe der bedingten Verteilungen Werte gezogen werden.
Die zugrundeliegende Theorie sowie die Durchführung in R werden genauer beschrieben in (van Buuren und Groothuis-Oudshoorn; 2011).

6.1.1. Annahmen

Es wird angenommen, dass der Datensatz D einer p-variaten Verteilung $P(D|\theta)$ folgt, welche durch den unbekannten Parametervektor θ komplett spezifiziert ist. Das eigentliche Problem, nämlich die multivariate Verteilung von θ zu erhalten, wird dabei mit bayesianischen Verfahren gelöst.
mice kann, im Gegensatz zu *Amelia*, mit MAR- und NMAR-Daten umgehen. Jedoch muss vor der Auswertung entschieden werden welche Annahme sinnvoll ist, da unter Gültigkeit des NMAR-Falls eventuell zusätzliche Modifikationen vorgenommen werden müssen.

6.1.2. Algorithmus

Der zugrundeliegende Algorithmus der Funktion *mice* kann in vier generelle Schritte eingeteilt werden und wird in (Azur et al.; 2011) genauer beschrieben:

- Im ersten Schritt wird für jeden fehlenden Wert einer Variablen eine einfache Stichprobe aus den beobachteten Werten gezogen. Dieser Wert wird statt des NA-Eintrags eingesetzt, sodass schlussendlich jede Beobachtung vollständig ist. Die eingesetzten Werte können dabei als „Platzhalter“ gesehen werden.

- Die „Platzhalter“ einer einzigen Variablen werden wieder gelöscht, sodass sie sich wieder im ursprünglichen Zustand befindet. Alle anderen Variablen bleiben vervollständigt. Die Variable im ursprünglichen Zustand wird im Folgenden als Y bezeichnet.

- Es wird eine Regressionsgleichung auf Grundlage der beobachteten Werte von Y durchgeführt, die bedingt wird auf alle anderen Variablen im Datensatz. Bei Y handelt es sich also um die abhängige Variable, die restlichen sind unabhängige Einflussgrößen. Bei der Aufstellung des Regressionsmodells wird dabei die Verteilung der abhängigen Variablen berücksichtigt. Gilt Y also beispielsweise als normalverteilt, wird ein einfaches lineares Modell aufgestellt, für ein ordinal-skaliertes Y wird hingegen ein kumulatives Logit-Modell berechnet. Mögliche Angaben in R werden in Kapitel 6.2 genauer beschrieben.

- Die fehlenden Werte von Y werden mithilfe von Vorhersagen auf Grundlage des aufgestellten Regressionsmodells ersetzt. Bei den unabhängigen Variablen wird als Datengrundlage zur Vorhersage für jede Beobachtung der wahre Wert verwendet, falls dieser vorhanden ist, sonst der imputierte Wert.

Die Schritte zwei bis vier werden nun für jede Variable im Datensatz, die imputiert werden soll, durchgeführt. Danach ist ein Durchgang der Imputation abgeschlossen. Die Schritte zwei bis vier werden nun mehrere Durchgänge lang wiederholt, wobei die Imputationen in jedem Durchgang aktualisiert werden. Eine sinnvolle Anzahl an Imputationsdurchgängen ist dabei von Situation zu Situation unterschiedlich. Ziel ist aber immer, Konvergenz in dem Sinne zu erhalten, dass Parameter und Regressionskoeffizienten am Ende der Durchgänge keine großen Veränderungen zum vorherigen Durchgang mehr aufweisen.

6.2. Umsetzung in R

Eine schematische Darstellung der Datenimputation mit dem R-Paket *mice* ist in folgender Grafik zu sehen:

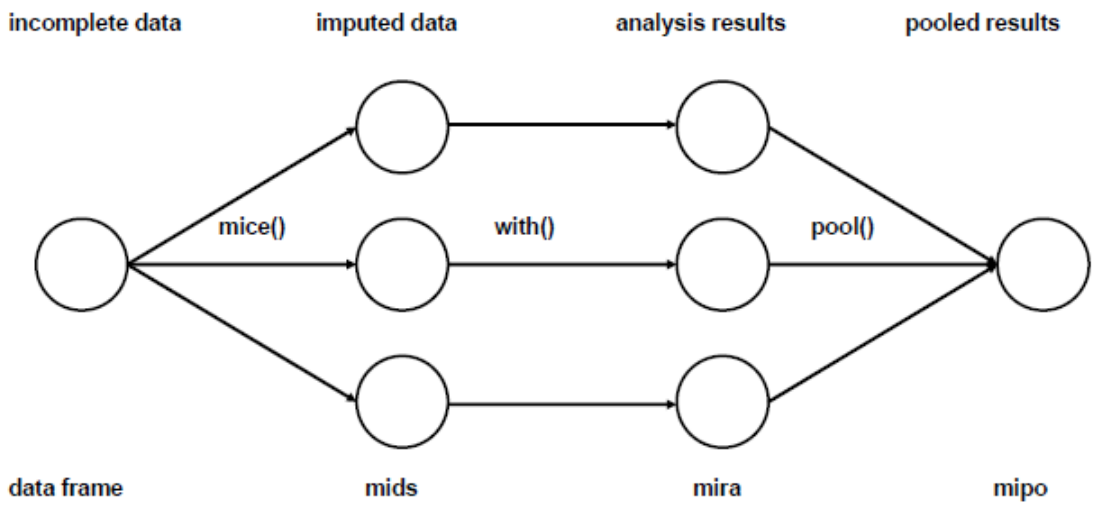

Abbildung 6.1.: Schematische Darstellung der Imputation mit *mice* in R aus (van Buuren und Groothuis-Oudshoorn; 2011).

Zu Beginn steht also der unvollständige Datensatz D, welcher der Funktion als ein Dataframe übergeben wird. Mithilfe der Funktion *mice* werden nun m imputierte Datensätze erzeugt. Standardmäßig werden pro Imputation fünf Durchgänge des in Kapitel 6.1.2 beschriebenen Algorithmus durchgeführt. Eine Erhöhung dieser Zahl zum Erreichen besserer Ergebnisse ist oftmals sinnvoll und muss von Fall zu Fall betrachtet werden.
Der Funktion kann auch die jeweilige Imputationsmethode übergeben werden, bei fehlendem Übergabeparameter werden je nach Datentyp default-Werte verwendet. Folgende Angaben sind möglich:

Methode	Regressionsmodell	Skalierung der Variable
pmm	predictive mean matching	numerisch
norm	bayesianische lineare Regression	numerisch
logreg	logistische Regression	nominal (2 Level)
polyreg	multinomiales Logit-Modell	nominal ($\geq$ 2 Level)
polr	kumulatives Logit-Modell	ordinal ($\geq$ 2 Level)

Tabelle 6.1.: Liste einiger univariaten Imputationsmethoden der Funktion *mice* in R.

Beim „predictive mean matching“ handelt es sich um eine semi-parametrische Imputationsmethode mit dem Vorteil, dass die Imputationen auf den Wertebereich der beobachteten Werte eingegrenzt werden. Ebenso können nicht-lineare Beziehungen erhalten werden, auch wenn der strukturelle Teil des Imputationsmodells nicht korrekt ist. Es handelt sich damit um eine gute Methode über alle numerischen Datentypen, kann aber auch bei kategorialen Daten angewendet werden. Die bayesianische lineare Regression ist eine effiziente Imputationsmethode falls die Modell-Residuen annähernd normalverteilt sind. Das multinomiale Logit-Modell wird mit der Funktion *multinom* aus dem *nnet*-Paket (Venables und Ripley; 2002) aufgestellt und ist gedacht für ungeordnete, kategoriale Variablen mit zwei oder mehr Kategorien. Dabei wird immer die erste Kategorie als Referenz verwendet. Für geordnete, kategoriale Variablen mit zwei oder mehr Kategorien wird mithilfe der *polr*-Funktion aus dem *MASS*-Paket (Venables und Ripley; 2002) ein kumulatives Logit-Modell aufgestellt, wobei auch hier die erste Kategorie als Referenz verwendet wird. Es existieren noch einige weitere Möglichkeiten, die in (van Buuren und Groothuis-Oudshoorn; 2011) nachgelesen werden können. Diese sind für die hier betrachteten Datensätze jedoch nicht von Relevanz und werden deswegen der Einfachheit halber weggelassen.
Ein weiterer Übergabeparameter für die Funktion *mice* ist die Angabe, in welcher Reihenfolge die Imputationen in jedem Durchgang durchgeführt werden sollen. Standardmäßig werden die Variablen im Datensatz von links nach rechts imputiert. Um eine schnellere Konvergenz des Algorithmus zu erreichen ist es manchmal sinnvoll, die Reihenfolge der Imputationen anzupassen. Eine Möglichkeit ist es, die Variablen mit aufsteigender Anzahl an fehlenden Werten zu imputieren, beginnend mit der geringsten Anzahl.

Es existieren einige weitere Übergabeparameter, die eine bessere Anpassung an verschiedene Datengrundlagen ermöglichen. Diese sind für die zugrundeliegenden Daten jedoch nicht von Relevanz und werden deswegen weggelassen, können aber in (van Buuren und Groothuis-Oudshoorn; 2011) nachgelesen werden.

Die imputierten Datensätze, in Abbildung 6.1 sind es drei Stück, werden dabei abgespeichert als ein Objekt der Klasse *mids*. Die drei Imputationen sind dabei identisch für die existierenden Werte und unterscheiden sich in den imputierten Werten. Das *mira*-Objekt wird anschließend der Funktion *with* übergeben, zusätzlich mit der gewünschten Auswertung wie beispielsweise einer Regressionsgleichung. Das entstehende Objekt der Klasse *mira* enthält mehrere unterschiedliche Analyseresultate, die letztendlich mit der Funktion *pool* zu einem Gesamtergebnis zusammengesetzt werden können. Das Gesamtergebnis stellt dabei den Mittelwert aus allen Einzelergebnissen dar, die Varianz des Schätzers wird dabei nach einem Vorschlag von Donald B. Rubin errechnet.

6.3. Ergebnisse

Die Ergebnisse werden erneut getrennt nach den Datensätzen und für unterschiedliche Fehlerraten dargestellt.

6.3.1. Kleinerer Datensatz

Für einen Anteil fehlender Daten knapp unter 10 % werden die 500 geschätzten Regressionskoeffizienten in einem Boxplot zusammengefasst.

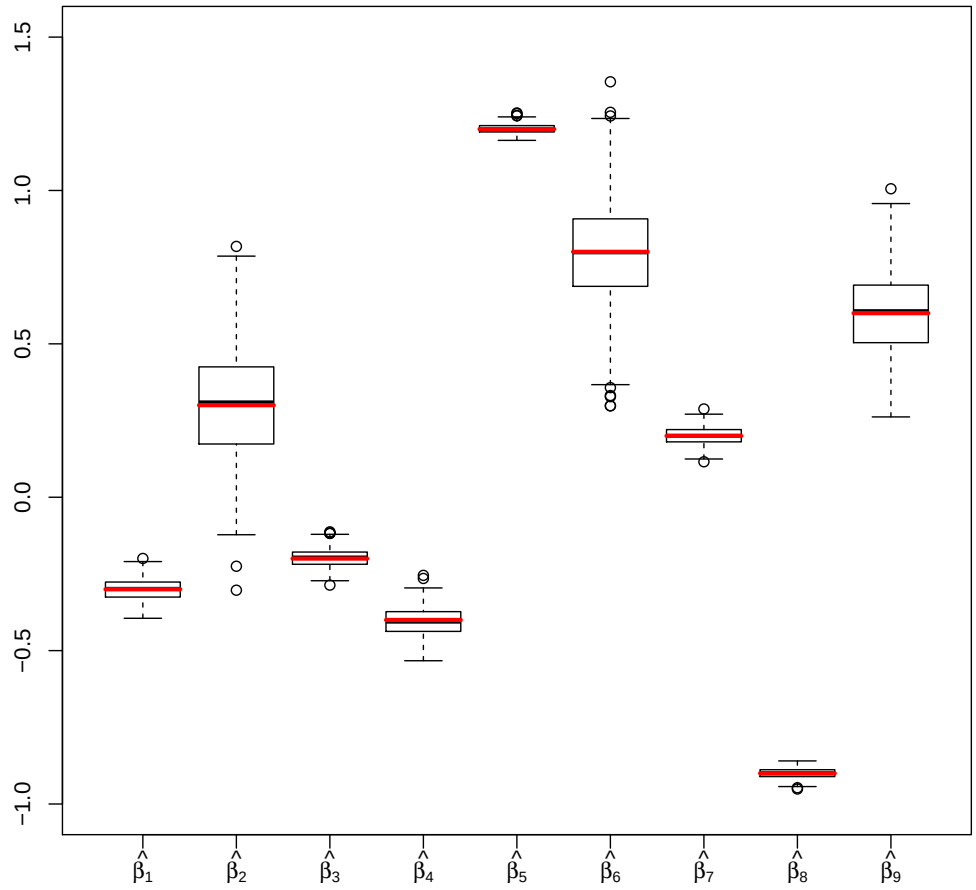

Abbildung 6.2.: Übersicht über die geschätzten Regressionskoeffizienten nach der Imputation mit *mice* aus 500 Durchgängen. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 10 % betrachtet.

Analog zu den Ergebnissen aus Kapitel 5.3 ist eine erhöhte Varianz der Koeffizientenschätzer für die binomial-verteilten Variablen X_2, X_6 und X_9 im Vergleich zu den Koeffizientenschätzern numerischer Variablen zu erkennen. Es werden jedoch alle wahren Koeffizienten tendenziell weder unter- noch überschätzt, ebenso wie keine extremen Ausreißer existieren.

Für eine höhere Fehlerrate sind die Ergebnisse sehr ähnlich. Die Koeffizientenschätzer streuen zwar etwas mehr, der wahre Wert der Koeffizienten wird jedoch im Mittel nahezu genauso gut angenähert. Die zugehörige Grafik befindet sich im Anhang.

6.3.2. Größerer Datensatz

Betrachtet man die Ergebnisse für eine Fehlerrate knapp unter 20 % bei der Imputation des Datensatzes mit 20 Variaben ergibt sich folgendes Bild:

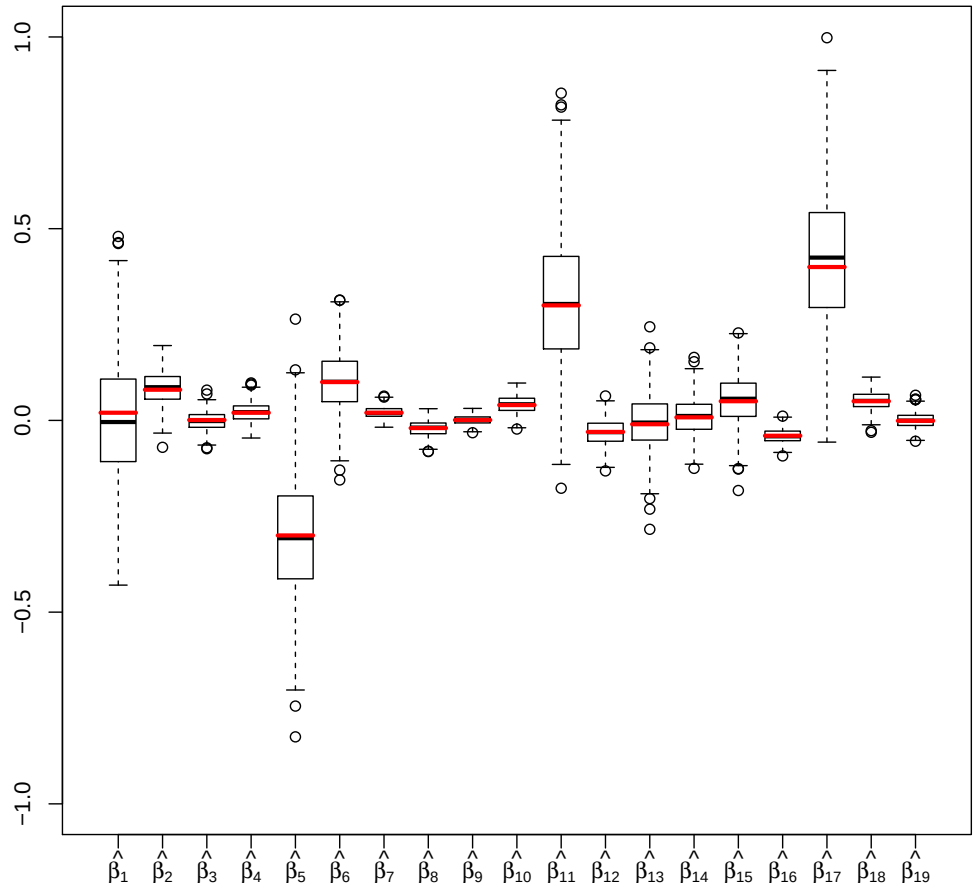

Abbildung 6.3.: Übersicht über die geschätzten Regressionskoeffizienten nach der Imputation mit *mice* aus 500 Durchgängen. Es wird der Datensatz mit 20 Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet.

Erneut ist die Spannweite der Koeffizientenschätzer binomial-verteilter Variablen weitaus größer als die der Koeffizientenschätzer numerischer Variablen. Für die dichotomen Variablen wird der wahre Wert von $\hat{\beta}_1$ und $\hat{\beta}_5$ tendenziell leicht unterschätzt, für $\hat{\beta}_{17}$ leicht überschätzt. Jedoch handelt es sich in absoluten Zahlen um sehr geringe Differenzen nahe 0. Für die numerischen Variablen ist im Mittel eine sichere Schätzung des wahren Regressionskoeffizienten zu erkennen, die maximalen absoluten Abweichungen der Schätzer vom wahren Wert sind dabei nahe 0.

Für die geringere Fehlerrate sind die Ergebnisse erneut sehr ähnlich, die zugehörige Grafik kann im Anhang betrachtet werden.

7. Regressionsimputation

In diesem Kapitel wird eine Methode zur einfachen Imputation mithilfe von Regressionsgleichungen vorgestellt. Für jeden fehlenden Wert im Datensatz wird eine Imputation erzeugt, woraus ein einziger, vollständiger Datensatz resultiert. Das Ergebnis muss also nicht wie in Kapitel 5 und 6 aus Einzelergebnissen zusammengesetzt werden. Die beobachteten Daten D^{obs} bleiben natürlich auch hier unberührt.
Diese Imputationsmethode ist keine sehr verbreitete oder vielgetestete Möglichkeit zur Vervollständigung von Datensätzen und soll mit bekannten Methoden verglichen werden.

7.1. Theorie

7.1.1. Annahmen

Wie schon in Kapitel 4.1 erwähnt, muss bei den meisten einfachen Imputationsmethoden die MCAR- oder MAR-Annahme erfüllt sein, um möglichst unverzerrte Schätzer zu erhalten. Zusätzlich wurde in Kapitel 4.1 das Problem angesprochen, dass die wahre Variabilität der Daten durch einfache Imputationsmethoden oft unterschätzt wird. Auf dieses Problem wird reagiert durch das künstliche Erzeugen von Zufallsfehlern im Laufe der Imputation, näher beschrieben in Kapitel 7.1.2.

7.1.2. Algorithmus

Im Grunde basiert diese Imputationsmethode auf der Definition bedingter Dichten, die wie folgt aussieht:

$$f(x|y) = \frac{f(x,y)}{f(y)} \tag{7.1}$$

Dies lässt sich umformen zu:

$$f(x,y) = f(x|y) \cdot f(y) \tag{7.2}$$

Die gemeinsame Dichte von X und Y lässt sich auch darstellen als ein Produkt aus der Dichte von Y und der bedingten Dichte von X auf Y. Dies lässt sich für mehrere Variablen weiterführen, für vier Variablen $X_1, ..., X_4$ gilt also beispielsweise

$$f(x_1, x_2, x_3, x_4) = f(x_4|x_1, x_2, x_3) \cdot f(x_3|x_1, x_2) \cdot f(x_2|x_1) \cdot f(x_1) \tag{7.3}$$

Ebenso könnte theoretisch die Reihenfolge beliebig vertauscht werden und die gemeinsame Dichte dargestellt werden als

$$f(x_1, x_2, x_3, x_4) = f(x_1|x_2, x_3, x_4) \cdot f(x_2|x_3, x_4) \cdot f(x_3|x_4) \cdot f(x_4) \tag{7.4}$$

Der Einfluss der Reihenfolge auf das Endergebnis wird ebenfalls untersucht.

Grundsätzlicher Algorithmus

Aufbauend auf dieser Definition werden die fehlenden Werte jeder Variablen mithilfe einer Regression imputiert. Nach der Reihenfolge in Formel 7.3 würde beispielsweise zuerst X_1 mithilfe von einfachen Zufallszahlen erzeugt werden. X_2 wird anschließend mit einer Regression mit X_1 als unabhängiger Variable imputiert. Dabei wird das Regressionsmodell passend zum Verteilungstyp der abhängigen Variablen gewählt, beispielsweise ein kumulatives Logit-Modell für ein ordinal-skaliertes X_2. X_3 wird anschließend durch ein Regressionsmodell mit den unabhängigen Variablen X_1 und X_2 erzeugt, die Imputation von X_4 funktioniert analog.
Dieser Imputationsvorgang ähnelt sehr der ursprünglichen Erzeugung der Daten wie in Kapitel 2.1 beschrieben. Die Koeffizienten des Regressionsmodells werden jedoch nicht fest vorgegeben, da der wahre Zusammenhang in realen Situationen nicht bekannt ist. Stattdessen werden die Regressionskoeffizienten mithilfe der nicht fehlenden Daten geschätzt.
Genau an dieser Stelle wird auch auf das Problem der tendenziellen Varianzunterschätzung bei einfachen Imputationsmethoden eingegangen. Zu den errechneten Regressionskoeffizienten auf Grundlage der vorhandenen Daten wird ein Zufallsfehler addiert. Dazu wird aus einer multivariaten Normalverteilung gezogen mit Mittelwertsvektor $\mu = \hat{\beta} = (\hat{\beta}_0, \hat{\beta}_1, ..., \hat{\beta}_p)$ und der geschätzten Varianz-Kovarianz-Matrix $\Sigma = \hat{cov}(\hat{\beta})$, die den Zusammenhang zwischen den Koeffizienten auf Grundlage des Regressionsmodells wiedergibt:

$$\tilde{\beta} \sim N_p(\hat{\beta}, \hat{cov}(\hat{\beta})) \tag{7.5}$$

Ab hier verläuft der Imputationsvorgang komplett analog zur Simulation in Kapitel 2.1. Mithilfe der Koeffizienten wird ein Regressionsmodell aufgestellt und mit dem dadurch errechneten Erwartungswert werden schlussendlich Zufallszahlen gezogen. Hierbei wird verwiesen auf Tabelle 2.2, in der genau die möglichen Variablentypen, die verwendeten Linkfunktionen sowie der Vorgang der Zufallsziehung in R beschrieben werden.
Auch bei der Zufallsziehung wird erneut auf das Problem der Varianzunterschätzung eingegangen, da durch das Ziehen von Zufallszahlen zusätzliche Variabilität entsteht.
Bei der Ausführung bleibt lediglich zu beachten, dass die Standardabweichung σ für normal-verteilte Variablen sowie der Shape-Parameter ν für gamma-verteilte Variablen in realen Situationen natürlich ebenso wie die wahren Regressionskoeffizienten nicht bekannt sind. Deshalb werden diese Parameter aus den Daten geschätzt. Dazu wird der geschätzte Dispersionsparameter $\hat{\phi}$ des Regressionsmodells betrachtet und folgendermaßen transformiert, siehe (Fahrmeir et al.; 2009):

- normal-verteilte Variable:
  der Dispersionsparameter ϕ entspricht der Varianz σ^2. Um die Standardabweichung σ zu erhalten, wird die Wurzel aus dem Dispersionsparameter gezogen:

$$\sigma = \sqrt{\phi} \tag{7.6}$$

- gamma-verteilte Variable:
  der Dispersionsparameter ϕ entspricht dem Kehrwert des Shape-Parameters ν, umgekehrt gilt

$$\nu = \frac{1}{\phi} \tag{7.7}$$

Im Folgenden sind die einzelnen Schritte für dieses Imputationsverfahren an einer poissonverteilten Variablen zu sehen:

$$
\begin{gathered}
\text{GLM} \\
\downarrow \\
\hat{\beta}, \hat{cov}(\hat{\beta}) \\
\downarrow \\
\tilde{\beta} \sim N_p(\hat{\beta}, \hat{cov}(\hat{\beta})) \\
\downarrow \\
\eta = x'\tilde{\beta} \\
\downarrow \\
\mu = exp(\eta) \\
\downarrow \\
y \sim rpois(lambda = \mu)
\end{gathered}
$$

Die Imputation wird analog für jede Variable der Reihe nach durchgeführt bis der Datensatz vollständig ist, wobei die Anzahl an unabhängigen Variablen im Modell mit jedem mal um eins steigt. Dabei wird das zugrundeliegende Regressionsmodell natürlich an die unabhängige Variable angepasst, ebenso wie die Errechnung des Erwartungswertes und der Zufallsziehungsprozess.

Die Ergebnisse dieses Imputationsverfahrens sind, wie später in Kapitel 7.3 gezeigt wird, im Vergleich zur multiplen Imputation mit *Amelia* oder *mice*, eher keine Verbesserung. Auch tauchten im Laufe der Durchführung einige Probleme auf, die eine Auswertung unmöglich machten. Deswegen wurden Modifikationen an den Daten und am Algorithmus vorgenommen und untersucht, ob unter diesen die wahre Situation eventuell besser dargestellt wird.

Weglassen von kategorialen Variablen im Datensatz

Wie schon in Kapitel 2.3 erwähnt, wurden schlussendlich nur normal-, poisson-, gamma- und binomial-verteilte Variablen in den Datensatz aufgenommen. Problematischer Schritt bei kategorialen Variablen ist das Aufstellen des Regressionsmodells, um daraus den Koeffizientenschätzer $\hat{\beta} = (\hat{\beta}_1, ..., \hat{\beta}_p)$ zu erhalten.
Schwierigkeiten treten beispielsweise auf, falls bei der Maximum-Likelihood-Schätzung

der Koeffizienten in einem beliebigen Durchgang k kein Maximum für ein endliches β existiert, also mindestens eine Komponente von $\hat{\beta}^{(k)}$ gegen unendlich geht (Fahrmeir et al.; 2009). Der ML-Algorithmus konvergiert in diesem Falle nicht und es kommt zum Funktionsabbruch.
Ebenso ist es problematisch, falls die geschätzten Wahrscheinlichkeiten für eine Kategorie sehr nahe bei 1 oder 0 liegen, was sehr hohe Schätzer $\hat{\beta}$ mit unverhältnismäßig hoher Standardabweichung zur Folge hat. Daraus resultieren weitere Probleme, die schlussendlich ebenso zum Funktionsabbruch führen.
Schwierigkeiten entstehen ebenfalls, wenn für die Kombination aus zwei kategorialen Variablen eine Ausprägung in den Daten nicht auftritt. Ein Beispiel hierfür ist die folgende Tabelle, die die Anzahl an Beobachtungen für jede Verknüpfung aus $X = \{1, 2, 3\}$ und $Y = \{a, b\}$ angibt:

	a	**b**
1	5	0
2	4	10
3	3	14

Die Chance, dass bei Kategorie 1 das Ereignis a eintritt, errechnet sich hierbei durch

$$R(Y = a|X = 1) = \frac{P(Y = a|X = 1)}{1 - P(Y = a|X = 1)} = \frac{P(Y = a|X = 1)}{P(Y = b|X = 1)} = \frac{5}{0} \tag{7.8}$$

Das Odds ist in diesem Falle also nicht definiert. Bei einem Logit-Modell wird das Odds jedoch benötigt, um den Erwartungswert μ zu errechnen:

$$R(Y = a) = \frac{P(Y = a)}{P(Y = b)} \underset{\text{Modell}}{\overset{\text{Logit-}}{=}} \frac{exp(\eta)}{1 + exp(\eta)} / \frac{1}{1 + exp(\eta)} = exp(\eta) \tag{7.9}$$

Auch für solche Datensituationen entstehen also Probleme bei der Aufstellung des Regressionsmodells.
Es exisiteren natürlich noch einige weitere Beispiele, bei denen das Aufstellen eines Logit-Modells nicht problemfrei funktioniert.

Verwendung einer penalisierten logistischen Regression anstatt des normalen Logit-Modells für binomial-verteilte Variablen

Die gerade geschilderten Probleme gelten analog für das Logit-Modell bei binomial-verteilten Variablen. Ein ebenso bekanntes Problem ist, falls eine perfekte Trennung in den Daten auftaucht. Um das Problem genauer zu verstehen, sei folgend ein Beispiel gegeben.
Die numerische Variable Y soll durch die binomial-verteilte Variable X erklärt werden. Die Datensituation ist dabei wie folgt:

X	**Y**
0	-5
0	-4
0	-3
0	-2
0	-1
1	1
1	2
1	3
1	4
1	5

Tabelle 7.1.: Datenbeispiel mit perfekter Trennung

Obwohl der Wert von Y durch die Variable X perfekt vorhergesagt werden kann, existiert bei der Maximum-Likelihood-Schätzung kein Maximum und der Koeffizientenschätzer $\hat{\beta}_{ML}$ deshalb auch nicht.

Um dieses Problem zu vermeiden wird zu Beginn der Imputation eine penalisierte logistische Regression mithilfe der Funktion *logistf* aus dem R-Paket *logistf* statt des normalen Logit-Modells durchgeführt. Diese von Firth 1993 entwickelte Methode, die eigentlich zum reduzieren des Bias der Maximum-Likelihood-Schätzer gedacht ist, eignet sich sehr gut im Umgang mit perfekter Trennung in den Daten. Die Methodik wird in (Heinze und Schemper; 2002) wie folgt erklärt:

Die Maximum-Likelihood Schätzer der Regressionsparameter β_r, $r = 1, ..., k$, erhält man durch Nullsetzen der Score-Funktion:

$$\frac{\partial log(L)}{\partial \beta_r} = U(\beta_r) = 0 \tag{7.10}$$

wobei L die Likelihood-Funktion ist. Um den Bias zu reduzieren, schlug Firth eine Modifikation dieser Formel vor:

$$U(\beta_r)^* = U(\beta_r) + \frac{1}{2} \cdot spur[I(\beta)^{-1}(\frac{\partial I(\beta)}{\partial \beta_r})] = 0, \qquad r = 1, ..., k \tag{7.11}$$

wobei $I(\beta)^{-1}$ die Inverse der Informationsmatrix ist. Dieser Schätzer existiert im Gegensatz zum ML-Schätzer auch, wenn perfekte Trennung in den Daten vorliegt.

Algorithmus unter Verwendung der geschätzten Werte $\hat{\beta}$

Um eine mögliche Überschätzung der Variabilität in den Daten zu vermeiden, wird zum einen anstatt des Koeffizientenvektors $\tilde{\beta}$ mit addiertem Zufallsfehler der wahre Koeffizientenschätzer $\hat{\beta}$ verwendet. Der schematische Ablauf aus Kapitel 7.1.2 lässt sich wie folgt anpassen:

$$\begin{gathered} \text{GLM} \\ \downarrow \\ \hat{\beta} \\ \downarrow \\ \eta = x'\hat{\beta} \\ \downarrow \\ \mu = exp(\eta) \\ \downarrow \\ y \sim rpois(lambda = \mu) \end{gathered}$$

Algorithmus mit mehrmaligem Durchlaufen des Imputationsvorganges

Um das Risiko zu verringern, dass die imputierten Werte in der ersten Runde zufällig sehr ungenau sind, wird der ursprüngliche Algorithmus mehrmals durchlaufen. Dabei werden natürlich weiterhin die selben, fehlenden Werte imputiert. Der Unterschied be-

steht darin, dass die Regressionskoeffizienten $\hat{\beta}$ und die Varianz-Kovarianz-Matrix Σ ab dem zweiten Komplettdurchgang auf Grundlage der vorherigen Imputation geschätzt werden, nicht auf Grundlage des unvollständigen Datensatzes. Dies kann beliebig viele Runden wiederholt werden.

7.2. Umsetzung in R

Zur vereinfachten Durchführung der Imputation wurden Funktionen geschrieben, die für wenige Übergabeparameter nach dem Algorithmus aus Kapitel 7.1.2 fehlende Daten imputieren. Eine Funktion verwendet dabei zum Errechnen des Prädiktors die wahren Regressionskoeffizienten $\hat{\beta}$, die andere Funktion benutzt den Regressionskoeffizienten $\tilde{\beta}$ mit zusätzlich addiertem Zufallsfehler.
Dabei werden einige bestehende Funktionen aus R verwendet. Zum einen wird die Funktion *glm* aus dem *stats*-Paket zum fitten von generalisierten linearen Modellen verwendet und die Funktion *vglm* aus dem Paket *VGAM* (Yee; 2010), um ein multinomiales oder kumulatives Logit-Modell zu fitten. Auch wird auf die Funktion *rmvnorm* aus dem Paket *mvtnorm* (Genz et al.; 2014) zugegriffen, um aus einer multivariaten Normalverteilung zu ziehen und damit $\tilde{\beta}$ zu erhalten.

Als Übergabeparameter benötigen die Funktionen zum einen den zu imputierenden Datensatz *data.missing* mit den fehlenden Werten. Auch muss der Datensatz *data.mod* übergeben werden, auf Grundlage dessen die Regressionsmodelle berechnet werden sollen. Für die mehrmalige Durchführung der Imputation ist das in der ersten Runde der unvollständige Datensatz selbst, ab der zweiten Runde jeweils der erzeugte, imputierte Datensatz. Für die beiden anderen Methoden ist *data.missing* und *data.mod* jeweils der unvollständige Datensatz.
Der Übergabeparameter *variable* ist ein Vektor, der die Spaltennummer der abhängigen, zu imputierenden Variablen für jeden Durchgang angibt. Die Länge dieses Vektors entspricht dadurch der Anzahl unvollständiger Variablen, die imputiert werden sollen. Im Vektor *variable.type* der selben Länge wird jeweils angegeben, welcher Verteilung beziehungsweise Skalierung die Variable folgt, mögliche Angaben sind hierbei “normal“, “poisson“, “gamma“, “binomial“, “nominal“ und “ordinal“.
In der Liste *independent.variables* werden jeweils die Spaltennummern der Variablen angegeben, die als unabhängige Einflussgrößen in das Modell mit aufgenommen werden sollen. Im ersten Listeneintrag ist das nur eine Variable, für jeden weiteren Listeneintrag

kommt die zuvor abhängige Variable dazu.
Bei Bedarf kann den Funktionen noch ein *seed* übergeben werden, sonst wird dieser auf NA gesetzt.

7.3. Ergebnisse

Die Ergebnisse werden für jeden Datensatz zuerst getrennt nach der Imputationsreihenfolge betrachtet. Durchgeführt wird zum einen die Imputation in der selben Reihenfolge wie bei der Datensimulation, das heißt zuerst wird X_1 mithilfe von Zufallszahlen vervollständigt und dann die restlichen Variablen $X_2, ...$ mithilfe von Regressionsmodellen. Die zweite Reihenfolge ist entgegengesetzt zur Simulation, das heißt von der vorletzten Variablen absteigend bis zur ersten. Hierbei wird die vorletzte Variable (X_9 im kleineren Datensatz, X_{19} im größeren) ebenso mithilfe von Zufallszahlen, alle restlichen Variablen von X_8 (beziehungsweise X_{18}) absteigend mithilfe von Regressionsmodellen aufgefüllt. Natürlich ist eine komplett beliebige Reihenfolge ebenso denkbar.
Zusätzlich werden für jeden Datensatz und jede Imputationsmethode verschiedene Anteile fehlender Daten betrachtet.

7.3.1. Kleinerer Datensatz

Für den kleineren Datensatz werden der ursprüngliche Algorithmus sowie beide Anpassungen durchgeführt, also einmal die Benutzung der geschätzten Koeffizienten ohne Zufallsfehler und einmal die Imputation mit mehrmaligem Durchlaufen des Algorithmus.

Ergebnisse bei der Imputationsreihenfolge analog zur Simulation

Für die Imputation in der selben Reihenfolge wie bei der Simulation werden alle drei in Kapitel 7.1.2 erwähnten Verfahren angewendet. Dabei wird zuerst die normal-verteilte Variable X_1 mithilfe der aus dem unvollständigen Datensatz geschätzten Parameter μ und σ zufällig erzeugt, wobei der Erwartungswert μ durch den Mittelwert geschätzt wird. Danach werden $X_2, ..., X_9$ mithilfe des Verfahrens imputiert.

Ergebnisse unter dem ursprünglichen Algorithmus

Für eine niedrige Fehlerrate um die 10 % ergibt sich folgendes Bild für die Koeffizientenschätzer:

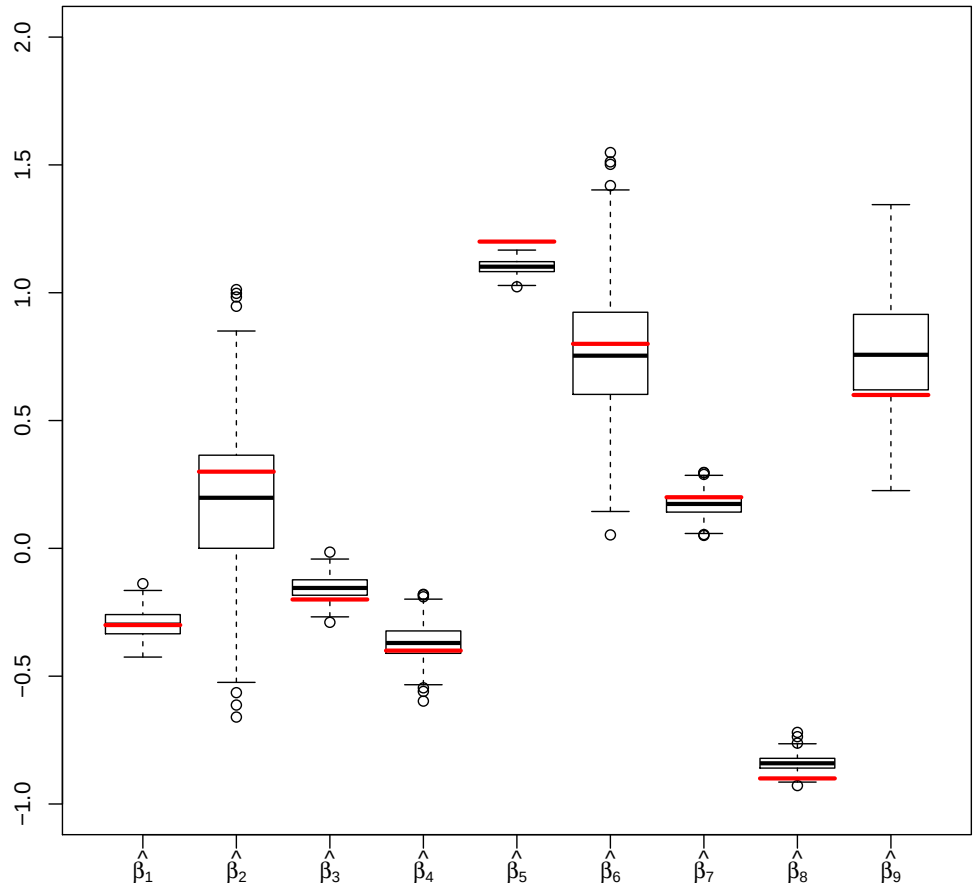

Abbildung 7.1.: Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 10 % betrachtet.

Auch bei dieser Methode ist, genauso wie bei *mice* und *Amelia*, eine deutlich erhöhte Varianz der Schätzer dichotomer Variablen im Vergleich zu den Schätzern numerischer Variablen zu erkennen. Jedoch werden hier bei den Schätzern durchaus Abweichungen vom wahren Koeffizienten deutlich. Beispielsweise wurde der Wert von β_5 in allen 500 Durchgängen unterschätzt. Auch Median und Mittelwert weichen für die meisten Schätzer vom wahren Koeffizienten ab.

Für eine erhöhte Fehlerrate ergibt sich ein leicht abgeändertes Bild:

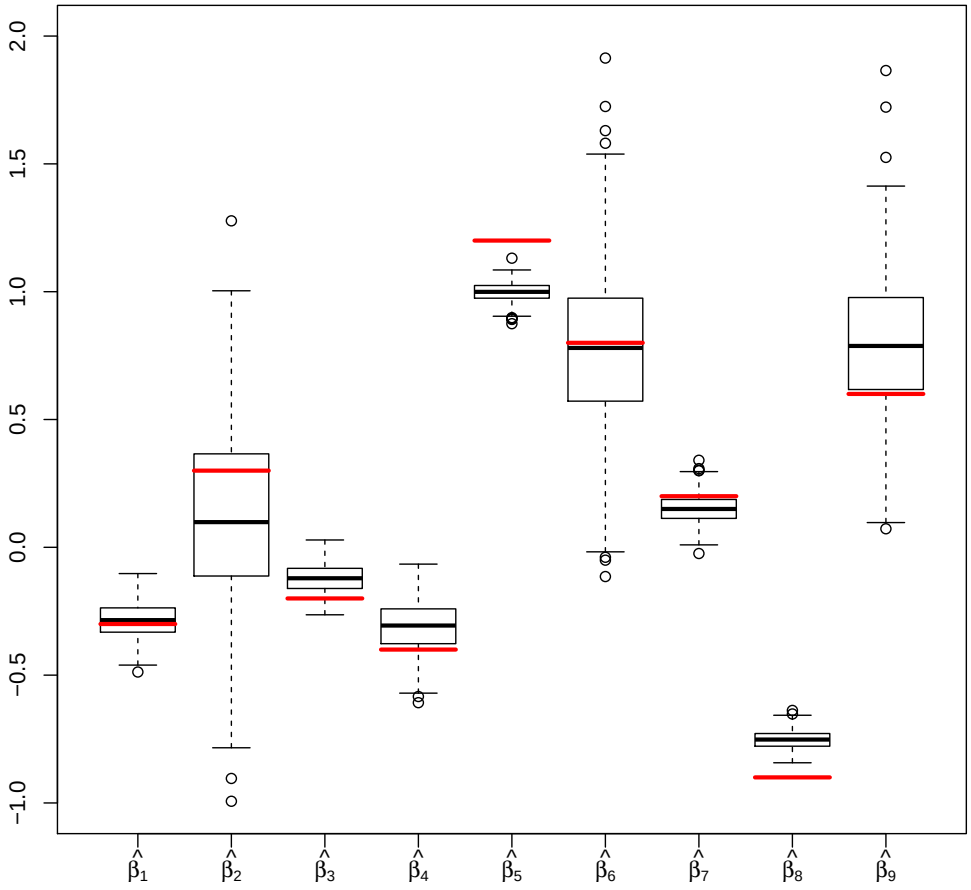

Abbildung 7.2.: Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet.

Die Tendenzen aus Abbildung 7.1 sind hier ganz analog zu erkennen. Die Schätzungen sind allerdings etwas schlechter, die Spannweite ist erhöht und der wahre Koeffizient wird schlechter abgebildet. Bei sechs von neun Koeffizienten schließt das 25 % bis 75 %-Quantil der 500 Schätzwerte den wahren Wert nicht ein. Ebenso wird der Wert von β_5 weiterhin in allen Durchgängen unterschätzt, der Wert von β_8 zusätzlich in allen 500 Durchgängen überschätzt.

Da die Unterschiede zwischen niedrigerer und höherer Fehlerrate für alle weiteren Imputationsvorgänge die gleiche Tendenz aufweisen, wird der Vergleichbarkeit halber immer eine mittlere Fehlerrate knapp unter 20 % betrachtet. Die analogen Grafiken mit niedrigerer Fehlerrate befinden sich im Anhang.

Ergebnisse unter Verwendung der geschätzten Werte $\hat{\beta}$

Für den Algorithmus unter Verwendung der geschätzten Werte $\hat{\beta}$ ohne zusätzlichen Standardfehler zur Errechnung des Erwartungswertes sehen die Schätzwerte wie folgt aus:

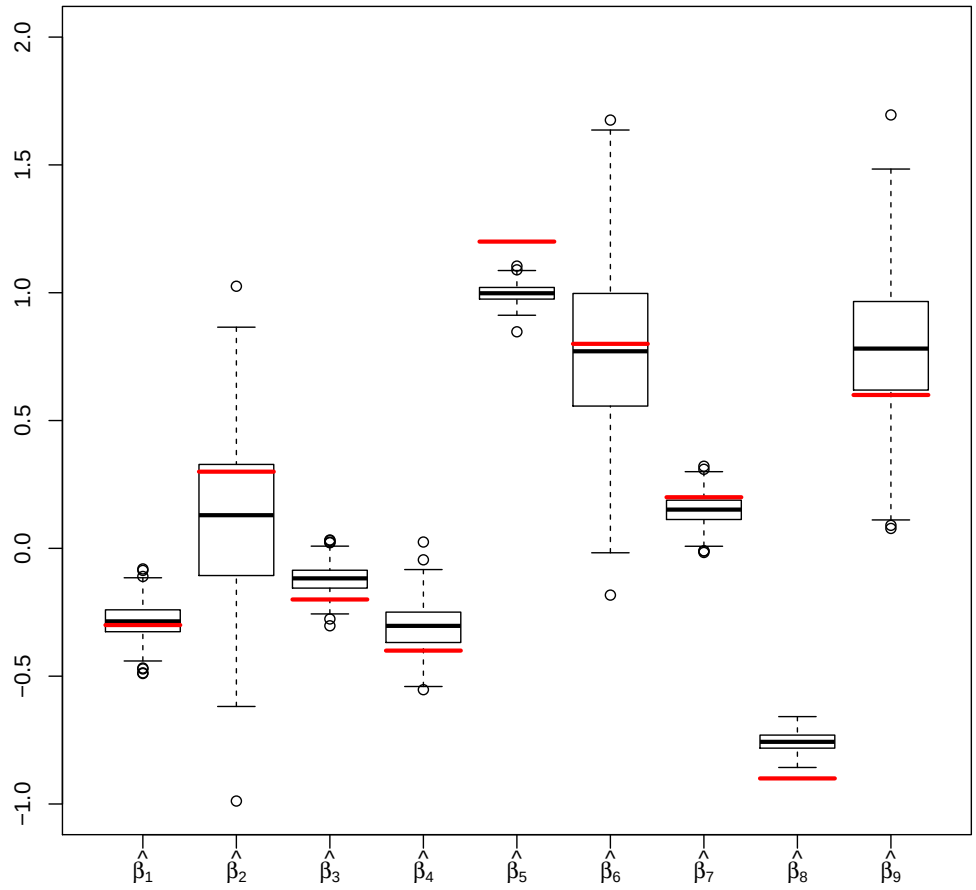

Abbildung 7.3.: Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen, beim zugrundeliegenden Algorithmus wird zu den geschätzten Koeffizienten kein zusätzlicher Standardfehler addiert. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet.

Das Abbild der Schätzwerte ähnelt dabei sehr den Werten unter dem Standardalgorithmus, dargestellt in Abbildung 7.2. Die Güte der Koeffizientenschätzer ist im Mittel ähnlich zu denen unter dem Standardalgorithmus. Für die meisten Schätzer ist jedoch eine etwas geringere Spannweite zu erkennen, vor allem für die der binomial-verteilten Variablen. Ebenso ist die Anzahl an Ausreißern tendenziell geringer.

Ergebnisse unter mehrmaligem Durchlaufen des Imputationsvorganges

Für den Algorithmus, bei dem der Imputationsvorgang mehrmals durchlaufen wird, sieht der Boxplot folgendermaßen aus:

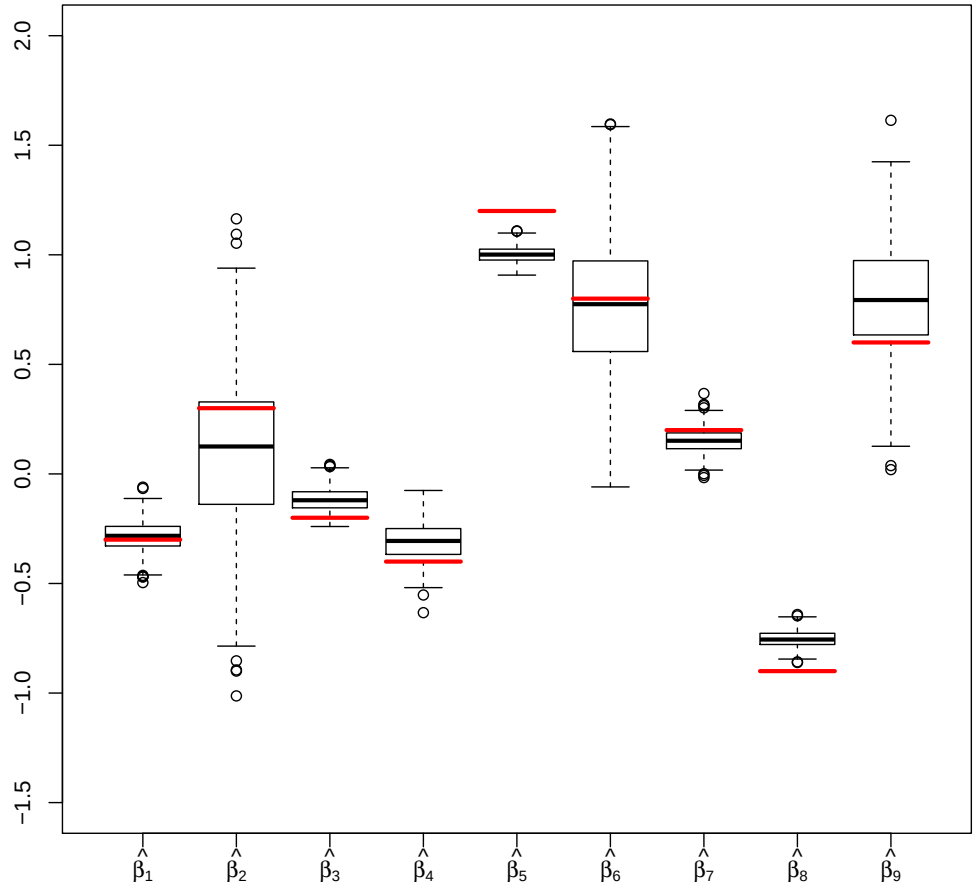

Abbildung 7.4.: Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen, wobei der Algorithmus mehrmals durchlaufen wird. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet.

Die Ergebnisse sind erneut sehr ähnlich zu denen aus Abbildung 7.2 und 7.3. Der Wertebereich der Schätzer ist dabei im Gegensatz zu dem Standardverfahren erneut tendenziell

etwas geringer, die Spannweite der dichotomen Variablen niedriger. Die Differenz zwischen den wahren Koeffizienten und dem Median und Mittelwert der Schätzer ist ähnlich wie bei den beiden anderen Verfahren.

Ergebnisse bei der Imputationsreihenfolge entgegengesetzt zur Simulation

Für die Imputation in entgegengesetzter Reihenfolge wie bei der Simulation werden der Standardalgorithmus und der Algorithmus unter Verwendung der $\hat{\beta}$-Werte verwendet. Beim mehrmals nacheinander ausgeführten Algorithmus gab es Probleme beim Berechnen des Logit-Modells, woraus ein Funktionsabbruch resultierte. Diese Problematik wurde bereits diskutiert.
Um eine Imputation in entgegengesetzter Reihenfolge durchzuführen wird zuerst die binomial-verteilte Variable X_9 zufällig erzeugt mithilfe der aus den vorhandenen Daten errechneten Wahrscheinlichkeiten für beide Kategorien. Danach werden X_8,..., X_1 mithilfe der Verfahren imputiert.

Ergebnisse unter dem ursprünglichen Algorithmus

Für den ursprünglichen Algorithmus ohne Modifikationen werden die Schätzwerte aus den 500 Durchgängen in einem Boxplot zusammengefasst:

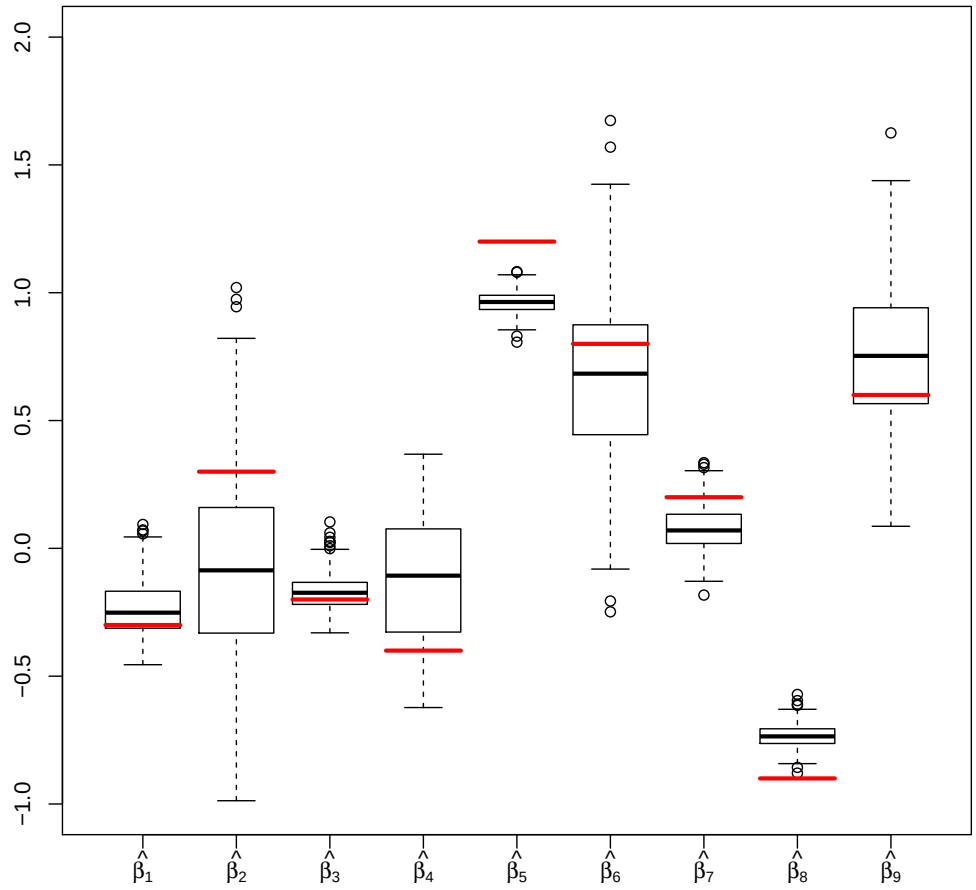

Abbildung 7.5.: Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen. Die Variablen werden dabei in entgegengesetzter Reihenfolge wie bei der Simulation imputiert. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet.

Analog zu allen bisherigen Imputationsverfahren weisen die Koeffizientenschätzer der dichotomen Variablen eine verhältnismäßig hohe Varianz auf. Auffällig ist hier der Schätzer für die normal-verteilte Variable X_4, der im Vergleich zu allen anderen Schätzern von numerischen Variablen weitaus mehr streut. Auch für diesen Imputationsvorgang wird der Koeffizient β_5 in allen Durchgängen unterschätzt, Mittelwert und Median weichen für die meisten Koeffizienten erkennbar von den wahren Werten ab.

Ergebnisse unter Verwendung der geschätzten Werte $\hat{\beta}$

Das Ergebnis unter Verwendung der geschätzten Koeffizienten ohne Standardfehler bei der Imputation ist wie folgt:

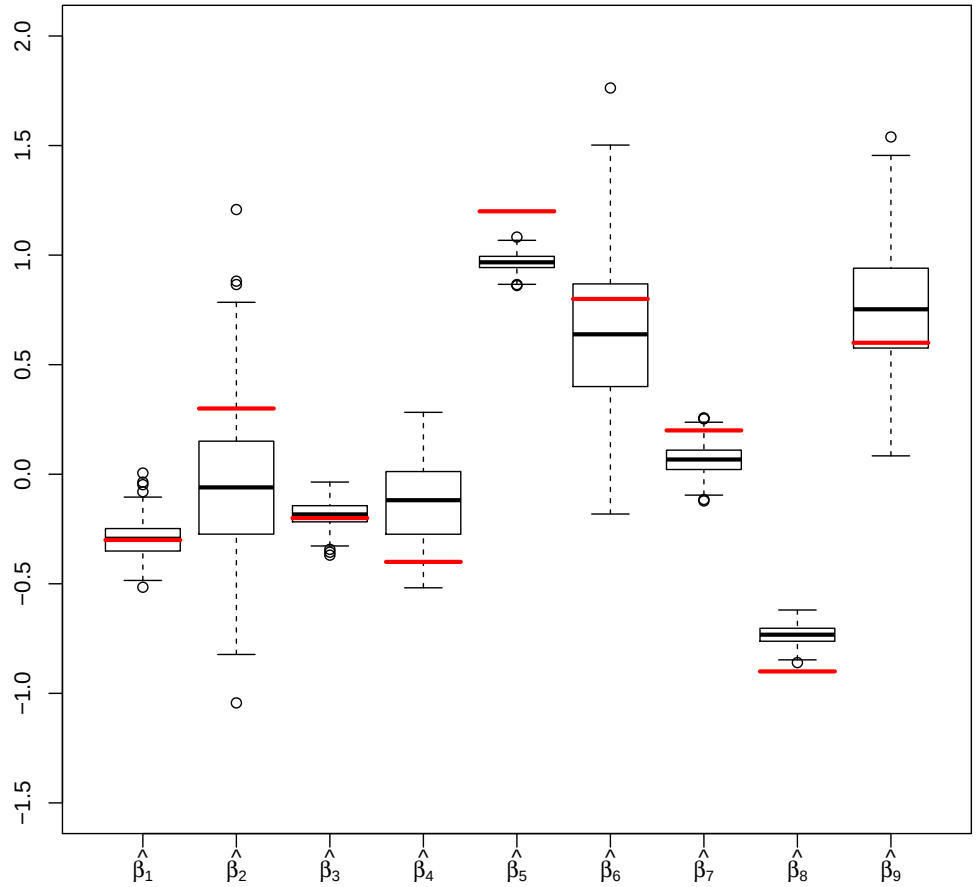

Abbildung 7.6.: Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen, beim zugrundeliegenden Algorithmus wird zu den geschätzten Koeffizienten kein zusätzlicher Standardfehler addiert. Die Variablen werden dabei in entgegengesetzter Reihenfolge wie bei der Simulation imputiert. Es wird der Datensatz mit zehn Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet.

Die Ergebnisse unterscheiden sich hier nur gering von denen aus Abbildung 7.5, die Interpretation erfolgt also ganz analog. Hier ist, im Vergleich zu den Ergebnissen bei der Imputation analog zur Simulationsreihenfolge, keine tendenzielle Verringerung bei der Spannweite der Schätzwerte im Vergleich zum Standardalgorithmus zu erkennen.

7.3.2. Größerer Datensatz

Für den größeren Datensatz mit 1000 Beobachtungen und 20 Variablen wird nur die Imputation mit den geschätzten $\hat{\beta}$-Werten ohne zusätzlichen Standardfehler betrachtet. Beim Aufstellen des Logit-Modells bei den beiden anderen Algorithmen kam es im Laufe der 500 Wiederholungen zu Problemen und letztendlich zum Funktionsabbruch. Die möglichen Ursachen dafür wurden bereits diskutiert.
Die Daten werden dabei wie in Kapitel 7.3.1 einmal in analoger Reihenfolge wie bei der Erzeugung und einmal in entgegengesetzter Reihenfolge imputiert.

Ergebnisse bei der Imputationsreihenfolge analog zur Simulation

Bei der Imputation der Variablen in der selben Reihenfolge wie bei der Simulation ergibt sich für die 19 Koeffizientenschätzer folgender Boxplot:

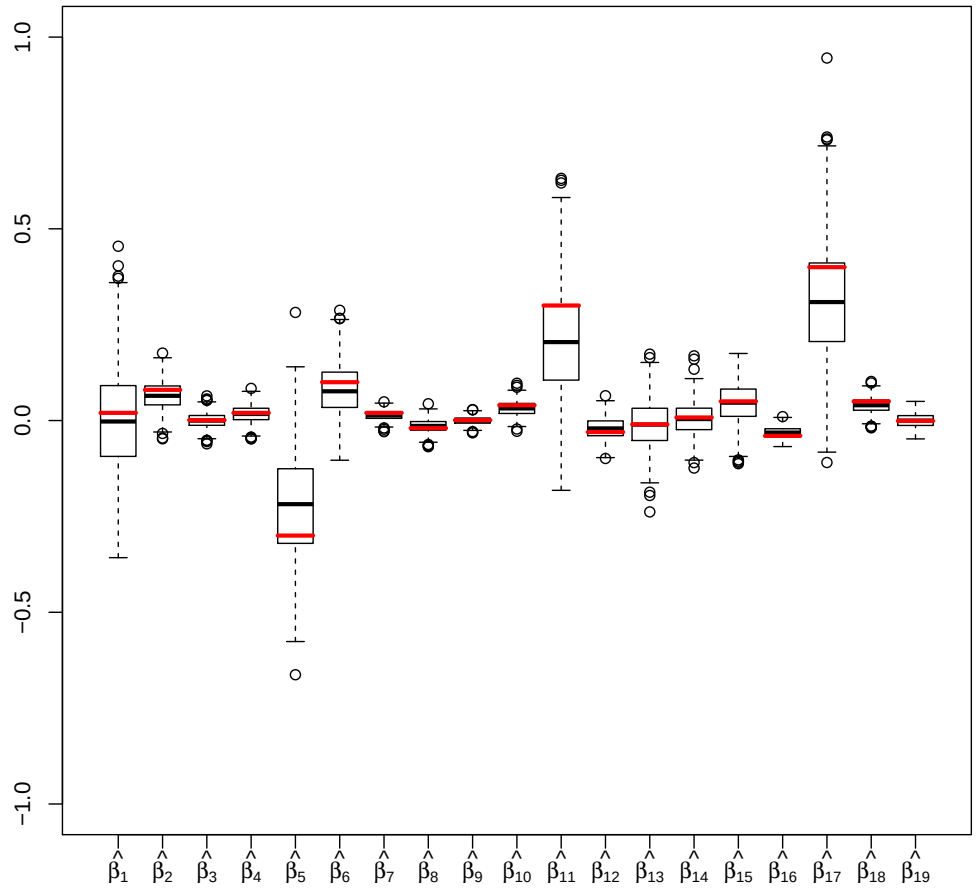

Abbildung 7.7.: Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen, beim zugrundeliegenden Algorithmus wird zu den geschätzten Koeffizienten kein zusätzlicher Standardfehler addiert. Die Variablen werden dabei in analoger Reihenfolge wie bei der Simulation imputiert. Es wird der Datensatz mit 20 Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet.

Auch hier gibt es keine Verbesserung bei der erhöhten Varianz der Koeffizientenschätzer binomial-verteilter Variablen. Im Mittel werden die Koeffizienten numerischer Variablen relativ genau angenähert, eine dauerhafte Über- oder Unterschätzung existiert für keinen Koeffizienten.

Ergebnisse bei der Imputationsreihenfolge entgegengesetzt zur Simulation

Für die entgegengesetzte Imputationsreihenfolge wird erneut ein Boxplot betrachtet, in dem die Schätzer aus 500 Durchgängen zusammengefasst dargestellt sind:

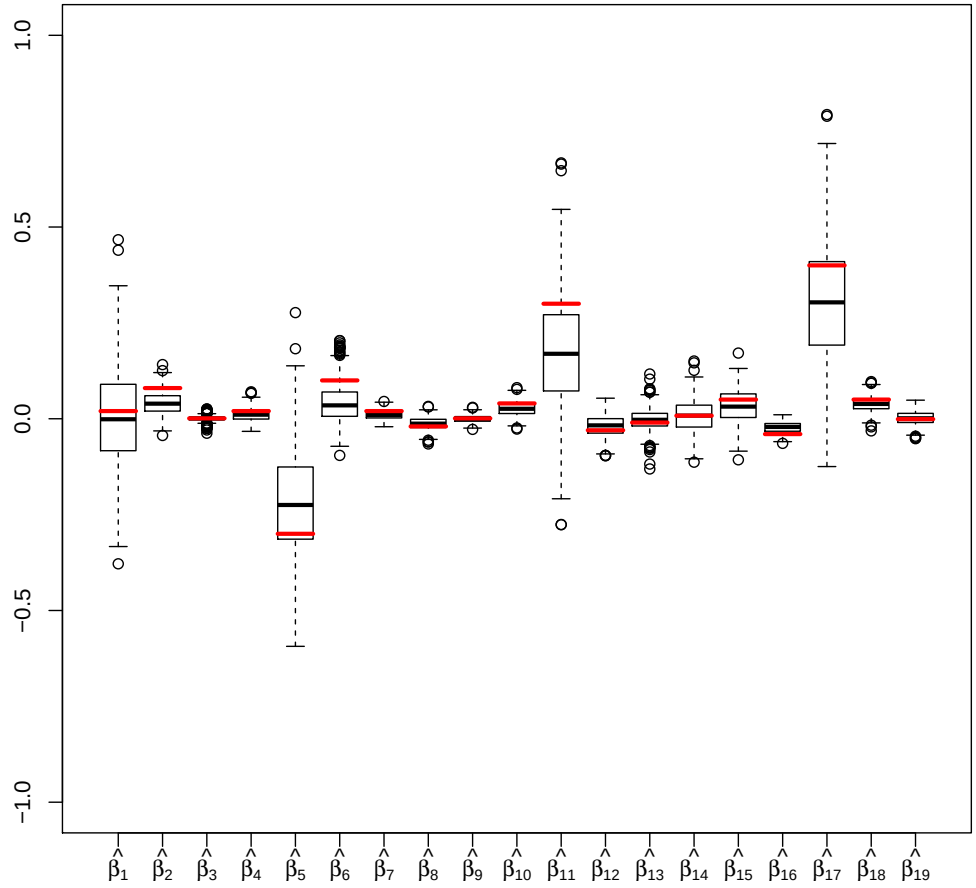

Abbildung 7.8.: Übersicht über die geschätzten Regressionskoeffizienten nach der Regressionsimputation aus 500 Durchgängen, beim zugrundeliegenden Algorithmus wird zu den geschätzten Koeffizienten kein zusätzlicher Standardfehler addiert. Die Variablen werden dabei in entgegengesetzter Reihenfolge wie bei der Simulation imputiert. Es wird der Datensatz mit 20 Variablen und einer mittleren Fehlerrate knapp unter 20 % betrachtet.

Auch hier ergeben sich keine Neuerungen zu der Interpretation von Abbildung 7.7. Die Differenz zwischen dem Median der Schätzwerte und dem wahren Wert scheint jedoch tendenziell etwas größer zu sein, jedoch nicht bei allen Koeffizienten.

8. Vergleich der Ergebnisse

Die Ergebnisse der getesteten Imputationsmethoden werden in diesem Kapitel noch einmal miteinander verglichen. Dafür werden für jeden Koeffizienten die Schätzer der verschiedenen Imputationsmethoden in einem Boxplot dargestellt. Bei dem kleineren Datensatz werden sieben Methoden verglichen, bei dem größeren Datensatz nur vier:

Kleinerer Datensatz	**Größerer Datensatz**
Amelia	*Amelia*
mice	*mice*
$X_1 \to X_9$	$X_1 \to X_{19}$ mit den $\hat{\beta}$-Werten
$X_1 \to X_9$ mit den $\hat{\beta}$-Werten	$X_{19} \to X_1$ mit den $\hat{\beta}$-Werten
$X_1 \to X_9$ mehrmals	
$X_9 \to X_1$	
$X_9 \to X_1$ mit den $\hat{\beta}$-Werten	

Tabelle 8.1.: Darstellung der verwendeten Imputationsmethoden für die beiden Datensätze.

Die Auswertungen wurden für beide Datensätze jeweils mit niedrigerer und höherer Fehlerrate und für jeden Koeffizienten durchgeführt. Einige Ergebnisse werden nachfolgend vorgestellt, alle übrigen Grafiken befinden sich im Anhang.

8.1. Kleinerer Datensatz

Für die Schätzungen des Koeffizienten β_2 ergibt sich ein häufiger vorkommendes Schema, weswegen die zugehörige Grafik zuerst betrachtet wird. Dargestellt wird eine mittlere Fehlerrate knapp unter 10 %.

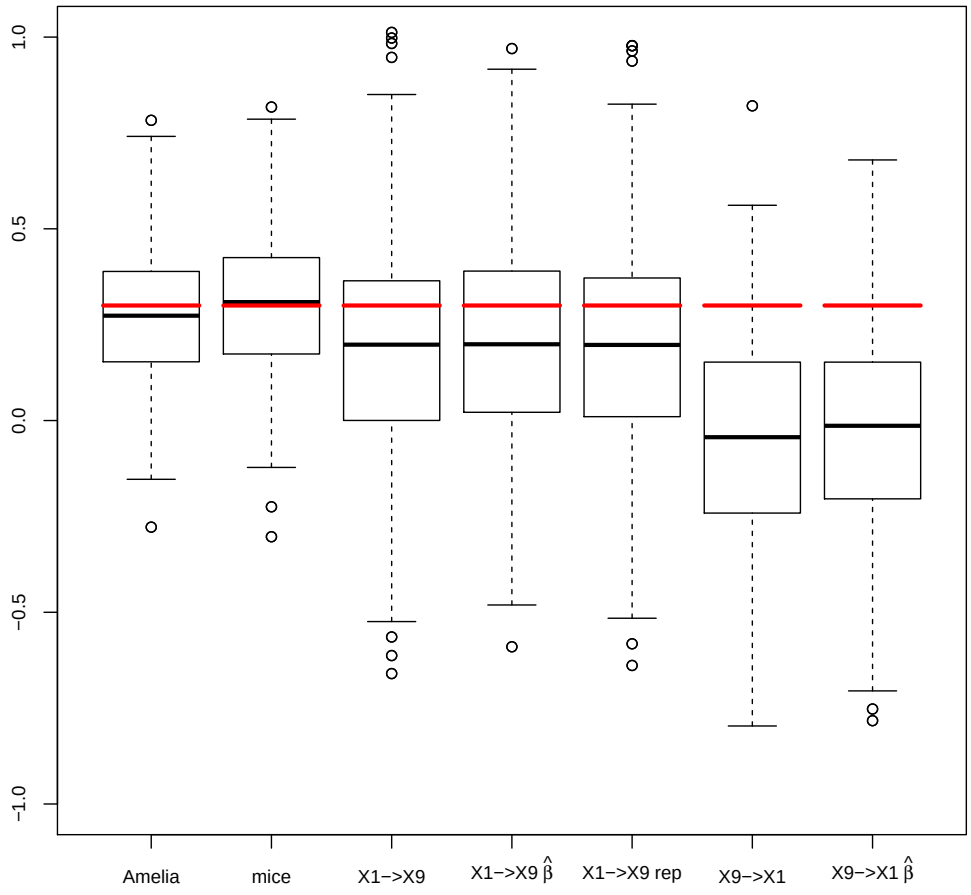

Abbildung 8.1.: Übersicht über die Schätzungen des Koeffizienten β_2 mit den verschiedenen Imputationsmethoden aus 500 Durchgängen. Den Schätzungen liegt der kleinere Datensatz und eine mittlere Fehlerrate um die 10 % zugrunde.

Es ist klar erkennbar, dass für die Imputation mit *Amelia* und *mice* der wahre Koeffizient und der Median der Schätzwerte sehr nah beieinander liegen, wobei *mice* etwas genauer ist. Der Wertebereich der Schätzer nach der Imputation mit *Amelia* und *mice* ist im Vergleich zu den einfachen Imputationsmethoden sichtbar kleiner.
Bei der Regressionsimputation in analoger Reihenfolge zur Simulation befindet sich für alle drei Algorithmen der wahre Wert im Bereich des 50 %- bis 75 %-Quantils der Schätzwerte. Der wahre Koeffizient wird also tendenziell unterschätzt. Der Median der Schätzungen ist für den Standardalgorithmus sowie für die zwei Modifikationen in etwa

gleich, der Wertebereich der Schätzer für die Anpassungen ist jedoch etwas geringer.
Für die Regressionsimputation in entgegengesetzter Reihenfolge zur Imputation befindet sich für die betrachteten Algorithmen der wahre Wert von β_2 außerhalb des 75 %-Quantils. Der Koeffizient wird also tendenziell noch mehr unterschätzt als nach der Imputation in analoger Reihenfolge zur Simulation. Die Spannweite der Werte ist dabei für beide Reihenfolgen bei der Imputation vergleichbar.
Insgesamt existieren für alle Imputationsmethoden wenig Ausreißer und vor allem keine extremen Ausreißer. Auch eine Unter- oder Überschätzung in allen 500 Imputationsdurchgängen exisitiert für den Koeffizienten β_2 nicht.

Vergleicht man die Schätzwerte mit denen bei höherer Fehlerrate knapp unter 20 % ergibt sich folgendes:

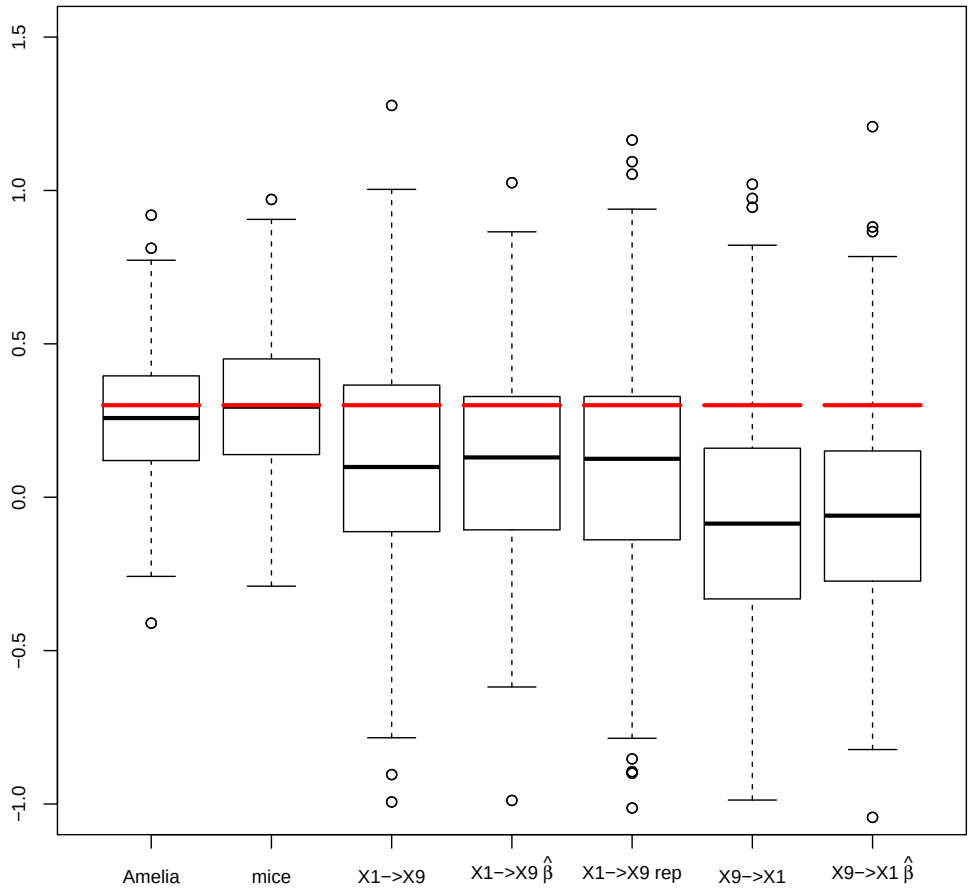

Abbildung 8.2.: Übersicht über die Schätzungen des Koeffizienten β_2 mit den verschiedenen Imputationsmethoden aus 500 Durchgängen. Den Schätzungen liegt der kleinere Datensatz und eine mittlere Fehlerrate um die 20 % zugrunde.

Im Vergleich zu Abbildung 8.1 ist klar ersichtlich, dass die Spannweite der Schätzwerte für alle Imputationsmethoden erhöht ist. Während der Median für *Amelia* und *mice* ähnlich nah am wahren Koeffizienten liegt, erkennt man für die einfache Regressionsimputation eine tendenziell verstärkte Unterschätzung des wahren Wertes im Vergleich zu einer geringeren Fehlerrate.

Die tendenziell schlechtere Schätzung bei Erhöhen der Fehlerrate ist für alle Koeffizienten und für den kleineren sowie den größeren Datensatz zu erkennen, weswegen in allen weiteren Grafiken der Vergleichbarkeit halber nur noch die Auswertungen mit einer mittleren Fehlerrate knapp unter 20 % betrachtet werden.

Die Tendenzen aus Abbildung 8.3 sind verstärkt für den Koeffizienten β_5 zu erkennen:

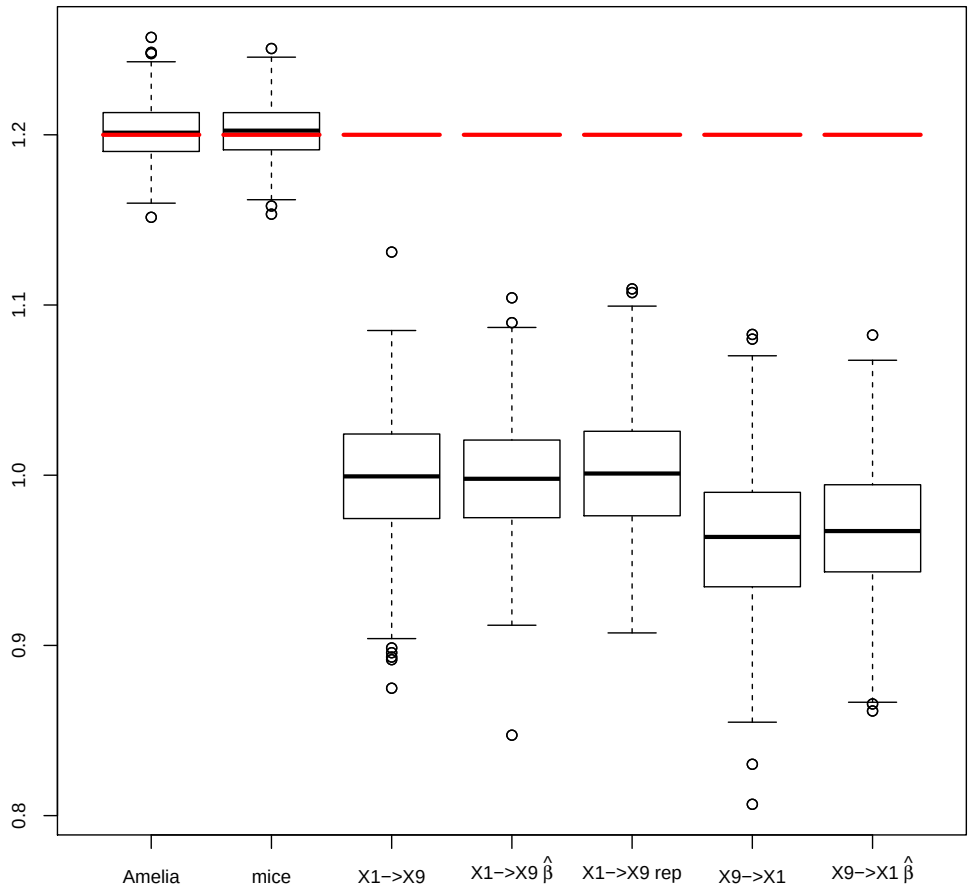

Abbildung 8.3.: Übersicht über die Schätzungen des Koeffizienten β_5 mit den verschiedenen Imputationsmethoden aus 500 Durchgängen. Den Schätzungen liegt der kleinere Datensatz und eine mittlere Fehlerrate um die 20 % zugrunde.

Erneut sind die Koeffizientenschätzer für β_5 nach der Imputation mit *Amelia* und *mice*

im Mittel recht präzise und weisen eine vergleichsweise geringe Spannweite auf. Für alle Algorithmen der einfachen Imputationsmethode wird der wahre Wert von $\beta_5 = 1.2$ in allen 500 Durchgängen unterschätzt. Dabei ist die Schätzung bei der Imputation in analoger Reihenfolge zur Simulation tendenziell noch etwas näher am wahren Wert als für die entgegengesetzte Reihenfolge.

Insgesamt wird für den kleineren Datensatz keiner der Koeffizienten nach der Regressionsimputation besser abgebildet als nach der Imputation mit *Amelia* oder *mice*. Es ist für nahezu jeden Koeffizienten eine größere Differenz zwischen dem wahren Wert und dem Median sowie auch dem Mittelwert zu erkennen. Ebenso ist die Varianz und Spannweite der Schätzer für *mice* und *Amelia* immer geringer.

8.2. Größerer Datensatz

Für den größeren Datensatz mit 20 Variablen ergibt sich ein leicht abgeändertes Bild. Ein häufig vorkommendes Schema ist für den Koeffizienten β_2 zu erkennen, der zugehörige Boxplot sieht wie folgt aus:

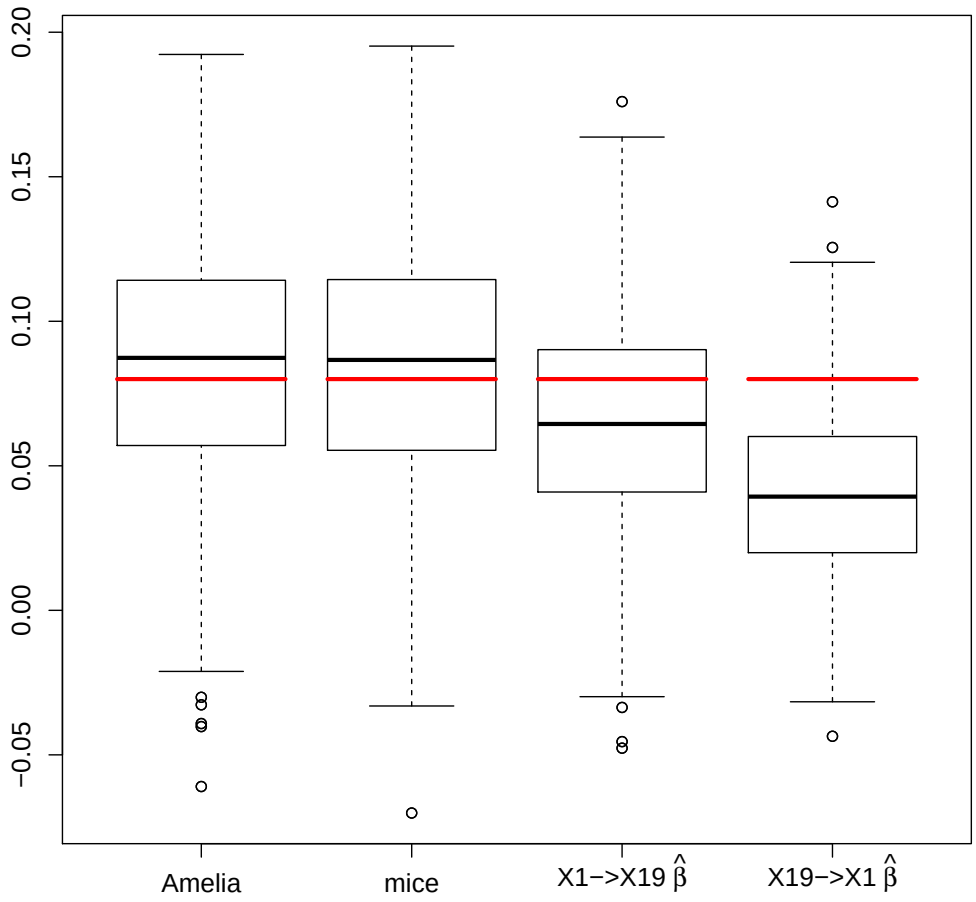

Abbildung 8.4.: Übersicht über die Schätzungen des Koeffizienten β_2 mit den verschiedenen Imputationsmethoden aus 500 Durchgängen. Den Schätzungen liegt der größere Datensatz und eine mittlere Fehlerrate um die 20 % zugrunde.

Analog zum kleineren Datensatz ist der Median der Schätzer aus 500 Durchgängen nach der Imputation mit *mice* und *Amelia* sehr ähnlich und liegt näher am wahren Wert als bei der Regressionsimputation. Die Imputation mit analoger Reihenfolge zur Simulation ist dabei im Mittel noch etwas genauer. Was hier jedoch von Kapitel 8.1 tendenziell abweicht, ist eine verringerte Spannweite bei den Schätzern nach der Regressionsimputation im Vergleich zu *Amelia* und *mice*.

Natürlich gibt es auch einige wenige Fälle, bei denen sich ein komplett anderes Bild ergibt. Das betrifft den Koeffizienten β_3, für den die Schätzer wie folgt aussehen:

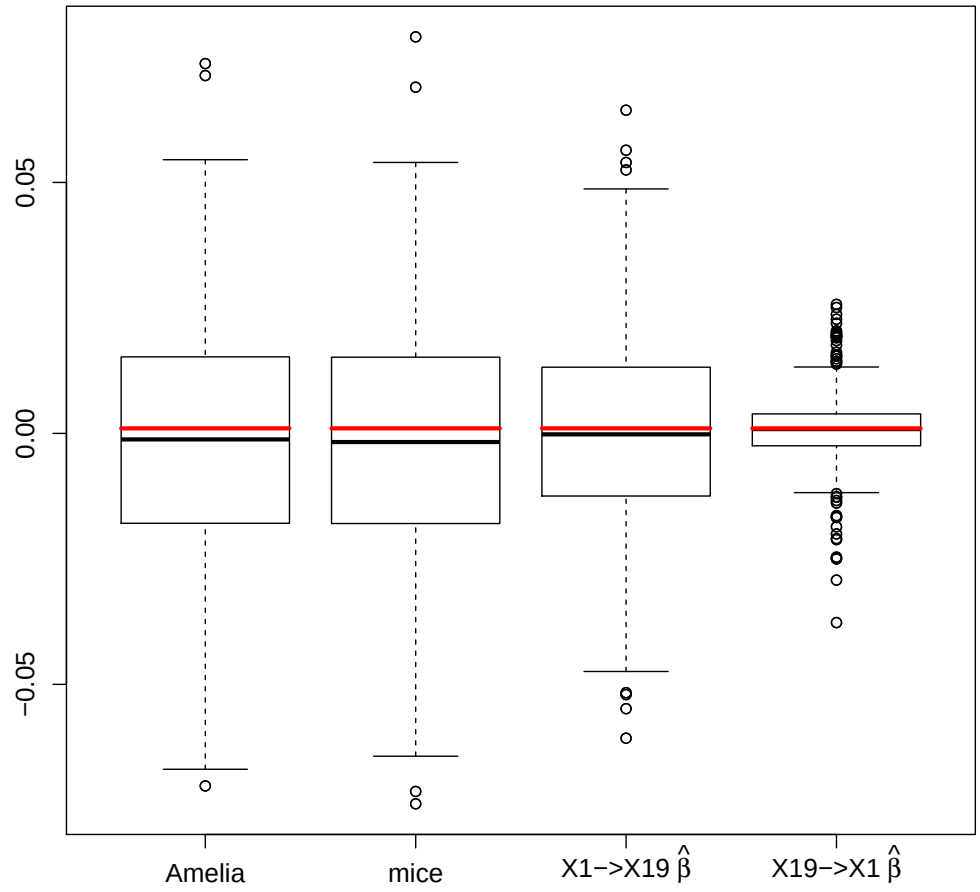

Abbildung 8.5.: Übersicht über die Schätzungen des Koeffizienten β_3 mit den verschiedenen Imputationsmethoden aus 500 Durchgängen. Den Schätzungen liegt der größere Datensatz und eine mittlere Fehlerrate um die 20 % zugrunde.

Die Differenz zwischen dem Median der 500 Schätzer und dem wahren Koeffizienten ist hier am geringsten für die Regressionsimputation in entgegengesetzter Reihenfolge wie bei der Simulation, am zweitgeringsten für die analoge Reihenfolge. Die Unterschiede sind in absoluten Zahlen jedoch minimal, wie an der Skala zu erkennen ist. Auffällig ist hier vor allem die Varianz sowie die Spannweite der Schätzer für die Regressionsimputation

in entgegengesetzter Reihenfolge wie bei der Simulation. Diese ist verhältnismäßig viel geringer.

8.3. Vorteile und Nachteile bei der Umsetzung in R

Was ebenso ein wichtiger Punkt bei der Durchführung der Imputation ist und worauf deswegen noch kurz eingegangen wird, sind die Vor- und Nachteile der verschiedenen Methoden bei der Umsetzung in R.
Zuerst sei angemerkt, dass die Anwendung der Funktionen *amelia* und *mice* in R sehr unkompliziert ist. Für die Imputation reichen für nicht zu spezielle Datengrundlagen einige wenige Übergabeparameter, jedoch gibt es einige Anpassungsmöglichkeiten an verschiedene Datensituationen.
Ein weiterer Punkt betrifft die Laufzeit der Imputationen. Während die Funktion *amelia* sowie die Regressionsimputation relativ schnell durchlaufen wird, benötigt die Funktion *mice* ein Vielfaches der Zeit für die Imputation. Vor allem, wenn die Anzahl an Imputationen m sowie die Anzahl an Durchläufen pro Imputation erhöht wird. Dadurch können jedoch tendenziell bessere Ergebnisse erreicht werden.
Wie schon mehrmals angemerkt wurde, war die Anwendung der Regressionsimputation für kategoriale Größen mit den gewählten Algorithmen nicht möglich und die Imputation von binomial-verteilten Variablen problematisch. Auch einige Modifikationen am Algorithmus brachten keine Lösung für das Problem.
Bei der Imputation mit *Amelia* tauchten einige Probleme auf, auch diese wurden schon angesprochen. Problematisch war hierbei vor allem die Imputation des größeren Datensatzes mit einer höheren Fehlerrate um die 20 %.

9. Zusammenfassung

Insgesamt sind einige Trends bei dem Vergleich der Imputationsmethoden erkennbar.
Erstens ist deutlich zu sehen, dass die Koeffizienten für binomial-verteilte Variablen nach der Imputation bei allen Methoden deutlich ungenauer und mit höherer Streuung geschätzt werden als die Koeffizienten numerischer Variablen.
Ebenso ist klar erkennbar, dass für eine geringere Fehlerrate die Spannweite und Streuung der Schätzer verkleinert wird, ebenso wie die Koeffizienten tendenziell besser angenähert werden.
Das angewendete Verfahren der Regressionsimputation in Verbindung mit kategorialen Variablen ist tendenziell problematisch, auch die getesteten Modifikationen am Algorithmus lösen dieses Problem nicht.
Was die Durchführung in R betrifft, ist *Amelia* sehr benutzerfreundlich und hat eine geringe Laufzeit. Bei *mice* dauert die Imputation dagegen um ein Vielfaches länger.
Beim Vergleich der Methoden sind die Koeffizientenschätzer nach der Imputation mit *Amelia* oder *mice* im Mittel meistens näher am wahren Wert als die Schätzer nach der Regressionsimputation. Auch liefert die Regressionsimputation in analoger Reihenfolge zur Simulation tendenziell bessere Ergebnisse als bei Verwendung der entgegengesetzten Reihenfolge. Hier existieren natürlich Ausnahmen.
Insgesamt liefert die Regressionsimputation also eher selten genauere Ergebnisse als die multiplen Imputationsmethoden. Eventuell kann durch weitere Modifikationen am Algorithmus eine Verbesserung erzielt werden, die Änderungen bei den getesteten Modifikationen sind jedoch minimal.

Literaturverzeichnis

Azur, M. J., Stuart, E. A., Frangakis, C. und Leaf, P. J. (2011). Multiple Imputation by Chained Equations: What is it and how does it work?, *International Journal of Methods in Psychiatric Research* **20**(1): 40–49. Letzter Abruf am 08.07.2015.
URL: *http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3074241/*

Fahrmeir, L., Kneib, T. und Lang, S. (2009). *Regression - Modelle, Methoden und Anwendungen*, Springer-Verlag Berlin Heidelberg.

Genz, A., Bretz, F., Miwa, T., Mi, X., Leisch, F., Scheipl, F. und Hothorn, T. (2014). mvtnorm: Multivariate normal and t distributions. R package version 1.0-2. Letzter Abruf am 08.07.2015.
URL: *http://CRAN.R-project.org/package=mvtnorm*

Heinze, G. und Schemper, M. (2002). A solution to the problem of separation in logistic regression, *Statistics in Medicine* **21**(16): 2409–2419.

Heumann, C. und Schmid, V. (2013). Schätzen und Testen II. Letzter Abruf am 08.07.2015.
URL: *http://www.statistik.lmu.de/∼bothmann/st2_2013/Vorlesung/Skript/Kapitel_5.pdf*

Honaker, J. und King, G. (2010). What to Do about Missing Values in Time-Series Cross-Section Data, *American Journal of Political Science* **54**(2): 561–581. Letzter Abruf am 08.07.2015.
URL: *http://gking.harvard.edu/files/pr.pdf*

Honaker, J., King, G. und Blackwell, M. (2011). Amelia II: A program for missing data, *Journal of Statistical Software* **45**(7): 1–47. Letzter Abruf am 08.07.2015.
URL: *http://www.jstatsoft.org/v45/i07/*

Imai, K., King, G. und Lau, O. (2015). Zelig: Everyone's Statistical Software. Letzter Abruf am 08.07.2015.
URL: *http://gking.harvard.edu/zelig*

Owen, M., Lau, O., Imai, K. und King, G. (2013). Zelig v4.0-10 Core Model Reference Manual. Letzter Abruf am 08.07.2015.
URL: *http://cran.r-project.org/web/packages/Zelig/vignettes/manual.pdf*

R Development Core Team (2008). *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0. Letzter Abruf am 08.07.2015.
URL: *http://www.R-project.org*

Spiess, M. (2008). *Missing-Data Techniken: Analyse von Daten mit fehlenden Werten*, Lit Verlag.

van Buuren, S. und Groothuis-Oudshoorn, K. (2011). mice: Multivariate Imputation by Chained Equations in R, *Journal of Statistical Software* **45**(3): 1–67. Letzter Abruf am 08.07.2015.
URL: *http://www.jstatsoft.org/v45/i03/*

Venables, W. und Ripley, B. (2002). *Modern Applied Statistics with S*, 4 edn, Springer-Verlag New York.

Yee, T. W. (2010). The VGAM package for categorical data analysis, *Journal of Statistical Software* **32**(10): 1–34. Letzter Abruf am 08.07.2015.
URL: *http://www.jstatsoft.org/v32/i10/*

A. Elektronischer Anhang

Der elektronische Anhang enthält die Ordner „Daten“, „Ergebnisse“ und „Programme“.

Im Ordner „Daten“ befinden sich die beiden simulierten Datensätze, mit denen in den Analysen gearbeitet wird.

Im Ordner „Ergebnisse“ befinden sich zum einen PDF-Dateien mit allen erstellten Grafiken, zusammengefasst nach dem jeweiligen Themenbereich. Im Unterordner „Koeffizientenmatrizen_nach_Imputation“ befinden sich für jede Imputationsmethode und jede Verknüpfung aus Datensatzgröße und Fehlerrate die Ergebnismatrizen, in denen für jeden der 500 Durchgänge die Koeffizientenschätzer abgespeichert sind. Ebenso sind dort die Matrizen abgespeichert, die für die vier Verknüpfungen die jeweilige Anzahl fehlender Daten pro Durchgang angeben. „Big“ und „small“ stehen dabei für die Größe des Datensatzes, „much“ und „less“ für die Fehlerrate.

Im Ordner „Programme“ befinden sich alle erstellten R-Codes. Die Codes sind jeweils nach Themenbereich getrennt.

Erklärung zur Urheberschaft

Hiermit versichere ich, dass ich die vorliegende Bachelor-Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

München, den 10. Juli 2015

(Susanne Rubenbauer)