

Geheimhaltung

FiRe – Ein Schritt zur Teilautomatisierung der Geheimhaltungsprüfung

von **Jakob Pohlisch, Julia Höninger, Ramona Voshage**

Das Programm FiRe kann bei der Geheimhaltungsprüfung von statistischen Ergebnissen in Forschungsdatenzentren eingesetzt werden, wenn das Statistiksoftwareprogramm Stata verwendet wird. Es übernimmt automatisch einzelne Schritte der Inputkontrolle, indem einzelne Befehle unterdrückt und nicht ausgeführt werden. Im Rahmen einer automatisierten Prozesskontrolle wird von Befehlen in Stata nur der Teil des Outputs angezeigt und im Log-File ausgegeben, der unter Geheimhaltungsgesichtspunkten unkritisch ist und veröffentlicht werden darf. Bei dem vorliegenden Beitrag handelt es sich um eine geringfügig modifizierte Fassung des FDZ-Arbeitspapiers Nr. 47, erschienen im Januar 2015, Düsseldorf.

1 Einleitung

Seit 2002 wurden bei den großen amtlichen Datenproduzenten Forschungsdatenzentren (FDZ) eingerichtet, um der Wissenschaft den Zugang zu Mikrodaten zu ermöglichen. So gibt es inzwischen unter anderem die FDZ der Statistischen Ämter des Bundes und der Länder (Zühlke et al. 2007), ein FDZ der Deutschen Rentenversicherung (Stegmann 2009) und ein FDZ der Bundesagentur für Arbeit im Institut für Arbeitsmarkt- und Berufsforschung (Allmendinger und Kohlmann 2005). Alle FDZ ermöglichen verschiedene Wege des Mikrodatenzugangs, die meisten FDZ bieten Wissenschaftlern¹ die Möglichkeit, entweder an speziell ausgestatteten Gastwissenschaftlerarbeitsplätzen in den Räumen der Datenproduzenten zu arbeiten oder die kontrollierte Datenfernverarbeitung zu nutzen. Bei letztgenannter Art des Datenzugangs senden Wissenschaftler Auswertungsprogramme an das FDZ, die FDZ-Mitarbeiter führen diese auf den Originaldaten aus und senden die Ergebnisse an die Wissenschaftler zurück. Alle Ergebnisse werden dabei vor der Freigabe an die Wissenschaftler von FDZ-Mitarbeitern auf die Einhaltung des Statistikgeheimnisses geprüft. Diese statistische Geheimhaltungsprüfung ist teilweise sehr aufwändig.

Im Projekt „infiniT – Eine informationelle Infrastruktur für das E-Science Age“ (Brandt und Zwick 2009), das vom Bundesministerium für Bildung und Forschung von Mai 2009 bis Dezember 2012 finanziert worden ist, wurden die Grundlagen für Ansätze zu einer (teil-)automatisierten Geheimhaltungsprüfung untersucht. Der Zugang zu Mikrodaten könnte durch ein echtes Fernrechnen, den sogenannten Remote Access, für die Wissenschaft komfortabler werden und in den FDZ weniger Ressourcen binden. Im Projekt wurden zwei Ansätze entwickelt: Alternative II wurde „Morpheus“ genannt (Höhne und Höninger 2012). Die Idee des ganzheitlichen Ansatzes

zur Automatisierung der Geheimhaltungsprüfung bei Morpheus ist, dass die Wissenschaftler auf anonymen Daten in Echtzeit rechnen und zu jedem Ergebnis ein Gütemaß erhalten. Bei der Alternative I wurde die Idee einer kombinierten Input-, Prozess- und manuellen Stichprobenkontrolle konzeptionell entwickelt (Hochgürtel und Brandt 2011). Inputkontrolle bedeutet hier, dass Befehle vor der Ausführung geprüft werden und solche Befehle, die in allen Fällen ein Enthüllungsrisiko darstellen, deaktiviert und nicht ausgeführt werden. Viele Befehle erzeugen jedoch Ergebnisse, die nur in manchen Konstellationen ein Enthüllungsrisiko darstellen, es müssen also jeweils die Rahmenbedingungen geprüft werden. Diese Prüfung übernimmt die Prozesskontrolle, die nur den Teil der Ergebnisse anzeigt, von denen kein Enthüllungsrisiko ausgeht. Der vorliegende Beitrag erläutert das Programm FiRe, das als ein Baustein in solch einer kombinierten Alternative I eingesetzt werden könnte. Mit FiRe kann die Input- und Prozesskontrolle bei Verwendung des Statistiksoftwareprogrammes Stata teilautomatisiert werden.

2 Grundlegende Idee von FiRe

Die Kontrollierte Datenfernverarbeitung erlaubt als einziger Zugangsweg der Forschungsdatenzentren die Analyse formal anonymisierter Einzeldaten. Für Datennutzer besteht jedoch kein direkter Zugang zu diesen Mikrodaten. Die Datennutzer erhalten Strukturdatensätze (Dummy-Dateien), die in Aufbau und Merkmalsausprägungen den Originaldaten weitestgehend gleichen. Mittels dieser Dummy-Dateien

¹ Zur sprachlichen Vereinfachung wird in diesem Text nur die männliche Form des Wortes „Wissenschaftler“ und seine Synonyme verwendet, wobei die Forschungsdatenzentren natürlich weibliche und männliche Wissenschaftler betreuen. Gleiches gilt für die im Folgenden verwendete Bezeichnung „FDZ-Mitarbeiter“.

können Auswertungsprogramme in den Analyseprogrammen SPSS, SAS, R oder Stata erstellt werden, mit denen die FDZ-Mitarbeiter in den statistischen Ämtern anschließend die Originaldaten auswerten. Die Datennutzer erhalten nach einer notwendigen Geheimhaltungsprüfung schließlich die Ergebnisse dieser Auswertung.

Die Geheimhaltungsprüfung in den FDZ wird momentan ausschließlich manuell durchgeführt und ist außerordentlich zeit- und ressourcenintensiv. Um den Aufwand zu reduzieren, wurde im Rahmen des Projektes InfiNitE nach Möglichkeiten einer automatisierten Geheimhaltung gesucht. Zwar existieren Verfahren zur automatischen Geheimhaltung von Tabellen, diese sind jedoch für Zwecke der FDZ nicht flexibel genug einsetzbar, um den Arbeitsaufwand in den FDZ zu reduzieren. Als eine gute Möglichkeit hat sich dagegen die Manipulation² von Befehlen herausgestellt. Befehle werden dabei so manipuliert, dass der zu generierende Output automatisch einer primären Geheimhaltungsprüfung unterzogen wird. Die ebenfalls notwendige sekundäre Geheimhaltung obliegt weiterhin den Mitarbeitern der FDZ.

Das Programm FiRe wurde im FDZ-Standort Berlin-Brandenburg entworfen und programmiert. Hier wurden einige erste Stata-Befehle umgeschrieben, um die grundsätzliche Machbarkeit einer Input- und Prozesskontrolle bei dem Statistiksoftwareprogramm Stata zu zeigen. Damit die umprogrammierten Befehle im Tagesgeschäft der FDZ eingesetzt werden können, wurde ein Programm namens FiRe, als Abkürzung von „Find & Replace“, entwickelt. Es ist innerhalb der Alternative I zur automatisierten Geheimhaltungsprüfung ein Baustein, um die Möglichkeit der Umsetzung der Prozesskontrolle stellvertretend am Statistiksoftwareprogramm Stata zu demonstrieren. Die von den Nutzern verfassten Do-Files werden mit Hilfe von FiRe so verändert, dass statt der Standardbefehle die neuen umprogrammierten Befehle verwendet werden. Durch diese werden die Bedingungen für eine Freigabe von statistischen Ergebnissen bis zu einem gewissen Grad automatisiert geprüft. Das Programm ist so konzipiert, dass neu umprogrammierte Befehle sehr leicht eingebunden werden können. Eine Textdatei enthält die Namen aller manipulierten Befehle. Kommen neue Befehle hinzu, kann die Textdatei entsprechend erweitert werden.

Insgesamt wurden vom FDZ-Standort Berlin-Brandenburg in einem ersten Schritt acht Stata Ado-Files manipuliert. Die Eingriffe sind im Abschnitt 4 dokumentiert. Die Prozesskontrolle durch das Programm FiRe kann bereits alleinstehend verwendet werden, ohne dass die Elemente Input- oder Stichprobenkontrolle die im InfiNitE-Projekt entwickelte Alternative I zu einem vollautomatisierten Datenfernzugang ergänzen würden.

3 FiRe

Anstelle der von den Nutzern angegebenen Originalbefehle sollen die manipulierten Befehle genutzt werden. An dieser Stelle greift FiRe ein, indem es den Aufruf der Originalbefehle durch ein „Suchen und Ersetzen“ („Find & Replace“), durch einen Aufruf der manipulierten Befehle, ersetzt. FiRe ist ein VBA-Makro, welches als add-on installiert und anschließend aus Microsoft Word gestartet werden kann. Die aktuelle Version des Programms FiRe ist Programmversion 3.5. Das Programm besteht aus den Teilen Pre- und Post-FiRe.

3.1 Pre-FiRe

Pre-FiRe durchsucht die Do-Files der Datennutzer vor deren Ausführung nach vorher definierten Befehlen und benennt diese um. So werden für die originale Kontrollierte Datenfernverarbeitung manipulierte Ado-Files verwendet. Alle Befehle, für die eine manipulierte Version vorliegt, können in der Datei „preFire.txt“ aufgelistet werden. Es wurde vereinbart, dass manipulierte Ado-Files nach folgendem Muster zu benennen sind: Dem ursprünglichen Befehlsnamen wird der Präfix „fdz“ vorangestellt. Die manipulierte Version von „summarize“ heißt dann beispielsweise „fdzsummarize“.

Das bearbeitete Do-File wird am gleichen Speicherort mit einer Erweiterung des Dateinamens um _FiRe.do gespeichert. Das originale Do-File des Wissenschaftlers wird somit nicht verändert. Wenn der FDZ-Mitarbeiter nun das geänderte Do-File startet, werden jeweils die manipulierten Befehle statt der Originalbefehle verwendet. Je mehr manipulierte Ado-Files bereitgestellt werden, umso mehr kann die manuelle Outputprüfung automatisiert werden.

Darüber hinaus fügt Pre-FiRe zu Beginn des Do-Files automatisch zusätzliche Informationen ein. Für den Wissenschaftler wird durch Kommentarzeilen deutlich und transparent gemacht, dass das Do-File mit FiRe bearbeitet wurde. Es wird angegeben, bei welchen Befehlen Namensänderungen durchgeführt wurden. Wichtig ist auch der Verweis auf den Ordner, in dem alle umgeschriebenen Befehle gesammelt werden, damit Stata diese neuen Befehle anwenden kann.

Da Befehle in Stata oft abgekürzt werden, werden auch alle zulässigen Abkürzungsvarianten geprüft. Dabei wurde darauf geachtet, dass nur Befehle den Namenszusatz „fdz“ erhalten, nicht jedoch andere Wörter oder ähnlich lautende Befehle, in denen die Buchstabenfolgen, beispielsweise „reg“ oder „sum“, vorkommen. Die neuen manipulierten Befehle müssen demnach auch in der abgekürzten Version bereitgestellt werden.

3.2 Post-FiRe

Um den Aufwand bei der Geheimhaltungsprüfung weiter zu reduzieren, können mit Post-FiRe Outputs zu einem gewissen Grad vorgeprüft werden. Es werden beispielsweise Angaben wie „1 missing value generated“ oder „1 obs not used“ automatisch gesperrt.

2 Der Begriff Manipulation wird hier in dem Sinne verwendet, dass die Originalbefehle, wie sie in dem Statistiksoftwareprogramm Stata implementiert sind oder wie sie von anderen externen Programmierern bereitgestellt werden, geändert und modifiziert werden. Sie entsprechen danach nicht mehr exakt dem Original; dieses wird verändert.

Sollten weitere Ersetzungen gewünscht sein, können diese in der Datei „postFire.txt“ ergänzt werden. In einer Zeile steht der zu ersetzende, in der darauffolgenden Zeile der Text, der stattdessen eingefügt werden soll. In jeder Zeile müssen zu Beginn zwei Rautezeichen (##) stehen, da die zu ersetzenden Textstellen auch mit einem Leerzeichen beginnen können.

Das mit Post-FiRe bearbeitete Log-File wird mit der erweiterten Dateiendung _FiRe.log im gleichen Ordner wie die Ausgangsdatei gespeichert. Somit wird auch hier sichergestellt, dass die originale Outputdatei nicht verändert wird.

4 Manipulierte Ado-Files im Detail

Die Idee besteht darin, die Befehle bezüglich ihres Outputs zu verändern. Leider liegen jedoch nicht von allen Befehlen die entsprechenden Quellcodes vor. Darüber hinaus würde eine Manipulation dieser die Nutzung der originalen Befehle unmöglich machen.

Anstatt die Befehle direkt zu manipulieren, werden die neuen Befehle auf Grundlage der existierenden programmiert. Dabei wurden bisher u. a. die Befehle regress, summarize und tabstat entsprechend verändert. Dies liegt zum einen an dem Aufwand der Manipulation der Befehle und zum anderen an der Häufigkeit, mit der diese im täglichen Betrieb der FDZ vorkommen.

Die Manipulation der Befehle unterscheidet sich grundlegend, je nach Vorhandensein eines manipulierbaren Ado-Files. Ist kein solches Programm vorhanden, muss ein solches geschrieben werden, welches den ursprünglichen Befehl unter Berücksichtigung der Geheimhaltung perfekt imitiert. Ist ein Ado-File vorhanden, kann dies meist deutlich einfacher angepasst werden.

Durch die Manipulation der Ado-files ergeben sich naturgemäß Fehlerquellen durch eine inkorrekte Programmierung. Die Vermeidung von Fehlern in den manipulierten Ado-Files muss als mindestens so wichtig eingeschätzt werden wie die Geheimhaltung selbst. Nichts wäre ein herberer Rückschlag auf dem Weg zur automatisierten Geheimhaltung als falsch generierter und anschließend durch Wissenschaftler veröffentlichter Output. Die bisher veränderten Befehle wurden daher vor der Einführung in den Alltagsbetrieb der FDZ ausgiebigen Tests unterzogen.³ Im Folgenden werden die veränderten Befehle genannt und näher erläutert.

4.1 regress

Durch den regress-Befehl kann in Stata eine lineare Regression durchgeführt werden. Die Frage, welche Enthüllungsrisiken unter anderem durch lineare Regressionen entstehen können, wurde bereits von Reznek (2003), Reznek und Riggs (2005) und Vogel (2011) untersucht. Es muss an dieser Stelle festgehalten werden, dass die hier implementierte Geheimhaltung nicht vor allen denkbaren Angriffsszenarien schützt. Eine kontinuierliche, gewissenhafte Prüfung des generierten Schätzwoutputs muss somit auch weiterhin gewährleistet sein.

In Vogel (2011) finden sich folgende Enthüllungsszenarien, denen durch die Umprogrammierung des Befehls regress begegnet wird:

- Variablen mit nur ein oder zwei von Null verschiedenen Beobachtungen,
 - Variablen, die für alle, außer für ein oder zwei Beobachtungen, nur sehr kleine Werte annehmen.
- Um diese Risiken zu eliminieren, muss jede einzelne im regress-Befehl aufgeführte unabhängige Variable auf diese beiden Risiken hin untersucht werden. Sowohl Szenario (a) als auch (b) stellen Dominanzfälle dar und können durch eine einfache Dominanzprüfung der beteiligten Variablen ausgeschlossen werden.

Der manipulierte Ado-File fdzregress enthält eine solche Prüfung. Dabei wurde sowohl eine Prüfung nach der (2,k)-Regel als auch eine Prüfung nach der p%-Regel implementiert. Die (2,k)-Regel überprüft, ob 2 Beobachtungseinheiten mehr als k % der Variablensumme auf sich vereinigen. Die p%-Regel hingegen prüft, ob sich der Wert der betragsmäßig größten Beobachtungseinheit mit Hilfe des zweitgrößten Einzelwertes schätzen lässt. Ist der Schätzfehler kleiner als p%, liegt ein Dominanzfall vor. Da sich die für k und p zu nutzenden Werte je nach Statistik unterscheiden können und darüber hinaus geheim zu halten sind, wurde der Befehl fdzregress um die Option critical() erweitert. Hier kann innerhalb der Klammern der Wert für die p%-Regel in Prozent spezifiziert werden. Soll beispielsweise p=15 sein, so lautet der Befehl:

```
fdzregress [varlist], critical(15)
```

Ist bei einer Statistik der Parameter k der (2,k)-Regel vorgegeben, so kann der Parameter p der p%-Regel, der das gleiche Schutzniveau wie die die (2,k)-Regel bietet, anhand der folgenden Formel berechnet werden (Statistisches Bundesamt 2007):

$$p = 100 \frac{100 - k}{100}$$

Soll die (2,k)-Regel mit k=85% angewendet werden, so bietet eine p%-Regel mit

$$p = 100 \frac{100 - 85}{100} = 17,6$$

den gleichen Schutz vor annäherungsweise Enthüllung einer Einzelangabe.

Sollte ein Dominanzfall vorliegen, wird die entsprechende Variable, die Art der Dominanzprüfung und der errechnete Wert für p bzw. k ausgegeben. Handelt es sich bei einer der überprüften Variablen um eine Variable mit einer Fallzahl von weniger als drei Beobachtungen für eine Ausprägung, so wird zusätzlich eine Häufigkeitstabelle der Variablen ausgegeben. Sowohl die Fallzahlprüfung als auch die Dominanzprüfung wird für jede Variable einzeln und ggf. innerhalb der Gruppierungen durchgeführt, welche beispielsweise über den Befehl „bysort“ spezifiziert werden können.

³ Die manipulierten Ado-Files können auf Anfrage zur Verfügung gestellt werden.

Der so produzierte Output enthält nun zunächst die geheim zu haltenden Informationen über p und k . Diese können jedoch leicht mit Hilfe von Post-FiRe gelöscht werden.

Die Entscheidung, ob und in welcher Weise der Output gesperrt werden muss, obliegt nach wie vor den Mitarbeitern der FDZ. Vorteil dieses Verfahrens ist jedoch, dass jede Regression auf eventuelle Risiken untersucht wird und eine manuelle Prüfung auf Dominanz entfällt.

An dieser Stelle muss erwähnt werden, dass sich die Prüfung auf Variablen beschränkt, die nur positive reelle Zahlen enthalten.⁴ Ist eine Variable auf ganz \mathbb{R} definiert, führt die Prüfung zu falschen Ergebnissen, da sich bei der Summation der Einzelwerte positive und negative Werte gegenseitig aufheben können. Somit wäre die berechnete Gesamtsumme und damit auch der berechnete Anteil der betragsmäßig größten Beobachtungseinheiten nicht korrekt. Dieses Problem muss den Prüfenden unbedingt bewusst sein, um unnötige Sperrungen zu verhindern.

4.2 tabstat

Der `tabstat`-Befehl generiert allgemeine Statistiken von zu definierenden Variablen, bezogen auf den gesamten Datensatz oder beliebig definierbare Untergruppen. Die Geheimhaltung dieser Statistiken ist in den FDZ des Bundes und der Länder einheitlich festgeschrieben. Diese Regeln wurden in die frei verfügbare Syntax des Befehls integriert. Es werden keine Statistiken auf der Grundlage von weniger als drei Beobachtungseinheiten veröffentlicht. Zusätzlich werden die Optionen `min` (Minimum) und `max` (Maximum) im Output gänzlich unterdrückt, da diese Variablenwerte einzelne Beobachtungen darstellen. Die Fallzahlregel bezieht sich hier insbesondere auch auf die Ausgabe von Perzentilen. So wird das 1%-Perzentil erst ab einer zugrundeliegenden Fallzahl von 300, das 5%-Perzentil erst ab einer Fallzahl von 60 etc. veröffentlicht.

Sollte der Befehl `tabstat` auf eine Variable angewendet werden, die für alle, außer für ein oder zwei Beobachtungen, nur sehr kleine Werte annimmt, so geben die Statistiken Werte aus, die näherungsweise Einzelwerte darstellen können. Dieses Risiko kann mit einer Dominanzprüfung eliminiert werden. Wie schon im `regress`-Befehl wurde auch im `tabstat`-Befehl eine solche Prüfung implementiert, die über die Option `critical()` steuerbar ist. Der entstandene manipulierte Befehl wurde unter dem Namen `fdztatstat.ado` gespeichert und steht zur Implementierung bereit.

4.3 summarize

Der Output des `summarize`-Befehls ist dem des `tabstat`-Befehls sehr ähnlich. Dies ist der Tatsache geschuldet, dass der `tabstat`-Befehl den `summarize`-Befehl zur Berechnung seiner Statistiken nutzt. Im Unterschied zum `tabstat`-Befehl ist die Syntax des `summarize`-Befehls jedoch nicht zugänglich und von daher nicht manipulierbar. Der Befehl `fdzsumma-`

`rize` beruht somit nicht auf einer Manipulation des ursprünglichen Befehls. Vielmehr handelt es sich um einen Klon des Befehls, dessen Rechenroutinen weiterhin auf `summarize` beruhen, jedoch mit einer integrierten Geheimhaltungsprüfung. Der Output ist dem des Originalbefehls nachempfunden. Unterliegenden Teile des Outputs der Geheimhaltung, werden sie nicht ausgegeben. Die Geheimhaltung erfolgt nach den gleichen Regeln wie bei `fdztatstat`. Auch im Befehl `fdzsummarize` wurde eine Dominanzprüfung nach dem gleichen Prinzip wie schon bei `fdzregress` und `fdztatstat` integriert.

4.4 codebook

Der Befehl „codebook“ gibt einen Überblick über die Eigenschaften und Ausprägungen einer Variable und gibt entweder die Wertelabel und eine einfache Häufigkeitsverteilung bei einem kategorialen Merkmal oder deskriptive Kennzahlen bei einem metrischen Merkmal aus. Durch die Umprogrammierung wird das Intervall, in welchem sich die Ausprägungen der aufgezählten Variablen befinden („range:“), nicht mehr angezeigt, da das Minimum und das Maximum zu schützende Einzelangaben sind. Die Anzahl der unterschiedlichen Ausprägungen („unique values:“) wird nicht mehr angezeigt, da sie in einigen Fällen einen Rückschluss auf den größten Wert zulassen. Weiterhin muss manuell überprüft werden, ob die Fallzahl zur Angabe von Perzentilen ausreicht und ob die Mindestfallzahl bei den verschiedenen Kategorien in Häufigkeitstabellen gewahrt ist.

4.5 utest

Mit dem Befehl „`utest`“ kann getestet werden, ob zwischen zwei Merkmalen eine U-förmige Beziehung besteht. Es handelt sich um einen Befehl, der als Ado-File installiert werden kann, er ist nicht standardmäßig in Stata enthalten. Im Output des Befehls wird durch die Manipulation sowohl der Extremwert des abhängigen Merkmals als auch das Intervall der Werte auf der x-Achse nicht angezeigt. Intern werden diese Werte berechnet, nur die Anzeige wird unterdrückt.

4.6 xtsum

Der Befehl gibt deskriptive Kennzahlen für eine Panelvariable aus. Dabei werden der Mittelwert und die Streuung sowie Minimum und Maximum sowohl für die ganze Variable ausgegeben als auch die Streuung in zwei Komponenten zerlegt: die Streuung zwischen den Beobachtungseinheiten im Querschnitt und die Streuung der Beobachtungseinheiten im Zeitverlauf. Die Extremwerte sollen auch bei diesem Befehl nicht herausgegeben werden. Der Befehl „`xtsum`“ wurde so umprogrammiert, dass die Extremwerte zwar berechnet werden, statt der Zahlangaben aber die Textangabe „gesperrt“ ausgegeben wird.

4.7 inspect

Dieser Befehl erzeugt ein rudimentäres Histogramm und gibt an der x-Achse den kleinsten und größten Wert an. Diese Einzelangaben müssen jedoch geheim gehalten werden. Mit FiRe kann eingestellt werden, dass der Befehl „`fdzinspect`“ anstatt des Be-

⁴ Die Null ist in diesem Fall Element der positiven reellen Zahlen.

fehls „inspect“ ausgeführt wird, wobei anstatt einer Ausführung des Befehls der Text „*FDZ: Befehl beim Fernrechnen nicht zulässig“ erscheint. FiRe bietet insofern auch eine Teilautomatisierung der Inputkontrolle.

4.8 xtdescribe

Der Befehl „xtdescribe“ gibt Kennzahlen zur Panelstruktur aus, z. B. die vorhandenen Teilnahmestrukturen der Beobachtungseinheiten und deren Häufigkeiten. Bei der Angabe der Anzahl der Teilnehmer im Querschnitt werden einige Beispiel-IDs ausgegeben, die in der manipulierten Variante des Befehls nun durch den Text „*FDZ: gesperrt“ ersetzt werden.

5 Ausblick

Die Prozesskontrolle durch FiRe und die manipulierten Ado-Files können zwar bereits eigenständig genutzt werden, stellen aber nur einen ersten Schritt in Richtung vollautomatisierter Ergebniskontrolle dar.

Durch den Einsatz von FiRe kann von einer enormen Zeitersparnis bei der manuellen Prüfung auf Geheimhaltung ausgegangen werden. Die größte Reduktion des Prüfaufwandes ist sicherlich durch die Umprogrammierung des Befehls „tabstat“ zu erzielen. Dieser Befehl wird von den Wissenschaftlern im FDZ besonders häufig verwendet. Um den Nutzen weiter zu steigern, müssen in Zukunft weitere Befehle manipuliert und eingebunden werden.

Durch die zusätzlich in die manipulierten Befehle integrierten Prüfungen auf Fallzahl- und Dominanzregelungen steigt selbstverständlich auch die für deren Ausführung benötigte Rechenzeit. Im Falle von „regress“ und „tabstat“ benötigt der manipulierte Befehl im arithmetischen Mittel ca. 3-mal länger

als das Original. Der „summarize“-Befehl hingegen benötigt im arithmetischen Mittel ca. 4-mal so lang wie der Originalbefehl. Die ermittelten Zeiten beruhen auf mehrmaligen Berechnungen mit dem gleichen Datensatz auf unterschiedlichen Computern. Dabei waren keine deutlichen Unterschiede zwischen den relativen Rechenzeiten bei unterschiedlichen Hardwarespezifikationen und Betriebssystemen feststellbar. Da es sich bei den Rechenzeiten um Sekunden handelt, ist die entstehende Verzögerung einer manuellen Prüfung jedoch in jedem Fall vorzuziehen.

Sowohl die Inputkontrolle als auch die Prozesskontrolle kommt durch FiRe teilautomatisiert schon heute in den FDZ zum Einsatz. Eine vollständige Implementierung der im Projekt infinitE entwickelten Alternative I zur (voll-) automatisierten Geheimhaltungsprüfung sind diese ersten Beispielprogrammierungen aber noch nicht. Auch können diese bisher nur eingesetzt werden, wenn die Wissenschaftler die Daten des FDZ mit der Software Stata auswerten. FiRe ist allerdings auf jedes statistische Softwareprogramm erweiterbar, dessen Syntax und Befehle in einem Texteditor bearbeitet werden können.

Jakob Pohlisch entwickelte im Rahmen seiner Tätigkeit im Referat *Mikrodaten, Analysen, Forschungsdatenzentrum* im Amt für Statistik Berlin-Brandenburg das hier vorgestellte Programm FiRe.

Julia Höninger leitet das Referat *Volkswirtschaftliche Gesamtrechnungen, Erwerbstätigkeit*. Zuvor arbeitete sie als wissenschaftliche Mitarbeiterin im Referat *Mikrodaten, Analysen, Forschungsdatenzentrum* des Amtes für Statistik Berlin-Brandenburg.

Ramona Voshage leitet das Referat *Mikrodaten, Analysen, Forschungsdatenzentrum* des Amtes für Statistik Berlin-Brandenburg.

Literatur

- Allmendinger, Jutta und Kohlmann, Annette (2005): Datenverfügbarkeit und Datenzugang am Forschungsdatenzentrum der Bundesagentur für Arbeit im Institut für Arbeitsmarkt- und Berufsforschung. *Allgemeines Statistisches Archiv* 88, S. 159–182.
- Brandt, Maurice und Zwick, Markus (2009): infinitE – Eine informationelle Infrastruktur für das E-Science Age; Verbesserung des Mikrodatenzugangs durch „Remote-Access“. *Wirtschaft und Statistik* 7/2009, Statistisches Bundesamt, Wiesbaden.
- Hochgürtel, Tim und Brandt, Maurice (2011): „InfinitE – Eine informationelle Infrastruktur für das E-Science Age: Verbesserung des Mikrodatenzugangs durch „Remote-Access“, Vortrag auf der Konferenz für Sozial- und Wirtschaftsdaten 13.–14.01.2011 in Wiesbaden.
- Höhne, Jörg und Höninger, Julia (2012): Das Verfahren Morpheus – Auf dem Weg zu Remote Access. Working Paper Series des Rates für Sozial- und Wirtschaftsdaten (RatSWD) 205/2012, Berlin. Verfügbar unter: http://www.ratswd.de/download/RatSWD_WP_2012/RatSWD_WP_205.pdf
- Reznek, Arnold P. (2003): Disclosure Risks in Cross-Section Regression Models. *American Statistical Association* 2003, Proceedings of the Section on Government Statistics and Section on Social Statistics: 3444–3451.
- Reznek, Arnold P. und Riggs, T. Lynn (2005): Disclosure Risks in Releasing Output Based on Regression Residuals. *American Statistical Association* 2005, Proceedings of the Section on Government Statistics and Section on Social Statistics: 1397–1404.
- Statistisches Bundesamt (2007): Entwurf – Leitfaden zur Festlegung eines p-Wertes für die p%-Regel zur Tabellen-Geheimhaltung. Anlage 6 zum Sachstandsbericht an den AOU vom September 2007, Internes Dokument, Statistisches Bundesamt, Wiesbaden.
- Stegmann, Michael (2009): Das aktuelle Datenangebot und Neuentwicklungen im FDZ-RV, DRV-Schriften Band 55/2009, S. 27–36.
- Vogel, Alexander (2011): Enthüllungsrisiko beim Remote Access: Die Schwerpunkteigenschaft der Regressionsgerade. FDZ-Arbeitspapier Nr. 36. Statistische Ämter des Bundes und der Länder, Düsseldorf
- Zühlke, Sylvia; Christians, Helga und Cramer, Katharina (2007): Das Forschungsdatenzentrum der Statistischen Landesämter – eine Serviceeinrichtung für die Wissenschaft. *ASTA Wirtschafts- und Sozialstatistisches Archiv* 3-4, S. 169–178.