Merkblatt für gute Programmierpraxis

Regeln für den Programmcode für das Arbeiten mit den KombiFiD-Mikrodaten

Sehr geehrte Datennutzerin, sehr geehrter Datennutzer,

im Rahmen des Projekts „KombiFiD – Kombinierte Firmendaten für Deutschland“ wurde ein neuer Unternehmensdatensatz aufbereitet, in welchem ausgewählte Daten der statistischen Ämter des Bundes und der Länder, der Bundesagentur für Arbeit und der Deutschen Bundesbank zusammengeführt wurden. Dieser Datensatz steht Ihnen nun über folgende Datenzugangswege zur Verfügung:

- Gastaufenthalt bei den Forschungsdatenzentren (FDZ) der Statistischen Ämter des Bundes und der Länder
- Gastaufenthalt am FDZ der Bundesagentur für Arbeit (BA) im Institut für Arbeitsmarkt- und Berufsforschung (IAB)
- Gastaufenthalt im Forschungszentrum der Deutschen Bundesbank. Die Nutzung der Bundesbankdaten ist ausschließlich im Forschungszentrum der Bundesbank möglich.
- Bei Nutzung von Daten der FDZ der Statistischen Ämter des Bundes und der Länder und des FDZ der BA im IAB ist *nach vorherigem Gastaufenthalt* der Zugang über die kontrollierte Datenfernverarbeitung möglich. Bei Nutzung von Daten der Deutschen Bundesbank ist die kontrollierte Fernverarbeitung ausgeschlossen.

Der KombiFiD-Datensatz bietet gegenüber der Verwendung der einzelnen Datensätze ein höheres Analysepotenzial. Seine Nutzung unterliegt allerdings besonders strengen Vertraulichkeitsanforderungen. Bitte beachten Sie, dass aus Gründen des Datenschutzes nach Gastaufenthalt bzw. bei kontrollierter Fernverarbeitung die Ergebnisse an Sie nur dann übermittelt werden, wenn einzelne Unternehmen nicht reidentifiziert werden können. Für eine effiziente und schnelle Prüfung Ihrer Ergebnisse auf Geheimhaltung möchten wir Sie bitten, bei der Aufbereitung Ihrer Programme folgende Regeln anzuwenden:

1. Betreffzeile der E-Mail:

Geben Sie in die Betreffzeile der E-Mail die ID-Nummer Ihres Nutzungsantrags ein.

2. Programmaufbau und Dateiennamen:

a) Erstellen Sie eine Master-Datei, die Informationen zu Ihrem Projekt enthält und von welcher weitere untergeordnete Programme aufgerufen werden (vgl. Punkt 3 unten). Die Master-Datei sollte exakt „master“[1] heißen.

Beachten Sie, dass nach einem Gastaufenthalt im FDZ der BA im IAB lediglich die Master-Datei von Mitarbeiterinnen bzw. Mitarbeitern des FDZ gestartet wird. Nach Ende des Programmdurchlaufs erhalten Sie ausschließlich die in diesem Programmdurchlauf erzeugten und auf Einhaltung des Datenschutzes geprüften Ergebnisdateien.

[1] z.B. „master.do“ in Stata. Bei Nutzung von SPSS legen Sie eine Textdatei als Master-Datei an.

b) Zur besseren Nachvollziehbarkeit teilen Sie lange Programme bzw. Analyseschritte auf kleinere Programmdateien/Module auf, z.B. Datenaufbereitung, Deskriptionen, Analysen. Jedes Modul kann auch aus mehreren Dateien bestehen. Die Unterprogramme sind dabei so zu benennen, dass ihre Abfolge klar erkennbar ist.

c) Vergeben Sie den erzeugten Daten- und Ergebnisdateien den gleichen Namen wie dem Programm, von dem sie generiert wurden.[2] Werden mehrere Ergebnisdateien von einem Programm generiert, machen Sie dies durch Endungen kenntlich.

d) Aktivieren Sie in Ihrem Programmcode die Protokollfunktion (Anlegen von Logfiles). Die Protokollfunktion darf nicht ausgeschaltet werden (abhängig von der Statistik-Software).

e) Geben Sie die Ergebnisse in ASCII-Text-Dateien aus. Bei Formaten wie `xls, doc` oder `rtf` ist der Aufwand für die Durchsicht für uns hoch.

f) Bei Verwendung von Stata:

   - Vergeben Sie der Stata-Ergebnisdatei die Endung „.log“. Die Endung „.scml“ ist zur Vereinfachung der Datenschutzprüfung nicht zulässig.
   - Da aus STATA-Grafiken sich teilweise die Ursprungsdaten erzeugen lassen, ist nur das Grafik-Format `gph` mit Option `asis` zulässig, z.B. `graph save filename, replace asis`. Die Formate `wmf`, `eps`, `png` sind ebenfalls zulässig.
   - Mit Stata verwendete ado-Dateien, die nicht in der Standard-Stata-Version enthalten sind, müssen vor dem Gastaufenthalt an die datenbereitstellende Stelle geschickt werden.

g) Struktur der Projektverzeichnisse: Für jedes Projekt wird ein eigener Ordner eingerichtet. Dieser Ordner enthält folgende Unterverzeichnisse:

   - `orig:` Hier werden die Originaldaten bereitgestellt. Für diesen Ordner erhalten Sie nur Leserechte.
   - `data:` Legen Sie alle von Ihnen generierten Datensätze hier ab.
   - `prog:` Hier sollten sich alle Auswertungsprogramme (z.B. do-, ado-, sps-Dateien) befinden.
   - `log:` Speichern Sie in diesem Ordner alle Ergebnisdateien, die in Ihren Analysen erzeugt werden einschl. Grafiken, ab.
   - `doc:` Dieser Ordner ist für alle weiteren Dateien, z.B. Word und Excel.

   Der Speicherplatz ist auf 20 GB pro Projekt begrenzt.

h) Die für den Gastaufenthalt vorbereiteten Auswertungsprogramme (und ggf. weitere benötigte Dateien) sollten spätestens drei Arbeitstage vor dem ersten Aufenthaltstag an den betreuenden Standort geschickt werden.

3. Dokumentation in der Master-Datei

Die Master-Datei soll einen Überblick über Ihr Projekt und Ihre Analysen verschaffen. Sie soll die folgenden Informationen enthalten:

a) Projekttitel, verwendetes Datenmaterial und Ihre Kontaktinformationen.

b) Ziel und Zweck des Programms und ggf. Einordnung der aktuellen Auswertungen in den Stand und Fortschritt des Projekts, Bezüge zu vorhergehenden und zukünftigen Auswertungen bzw. eventuelle Änderungen der Auswertungsstrategie.

[2] In Stata z.B. modul_1.do, modul_1.dta. modul_1.log.

c) Pfadreferenzen auf verwendete Dateien

d) Liste der verwendeten Originalvariablen

e) Liste der erzeugten Variablen mit ihren Labels

   Geben Sie bitte den neu generierten Variablen sprechende Namen und versehen Sie diese Variablen mit verständlichen Variablenlabeln und ggf. Wertelabeln.

f) Aufruf aller weiteren Unterprogramme bzw. Module

   Dazu ist es wichtig, dass hinter jedem Programmaufruf eine kurze Beschreibung der Inhalte des jeweiligen Unterprogramms erfolgt.

g) Werden die Ausgangsdaten in weiteren Analysen aggregiert (z.B. auf Ebene von Wirtschaftszweigen), muss dies im Master-Programm entsprechend deutlich gemacht werden. Geben Sie auch an, in welchem Programmschritt und auf welche Ebene aggregiert wurde (vgl. Punkt 5 l).

4. Dokumentation innerhalb der Auswertungsprogramme

Jedes Auswertungsprogramm muss *optisch klar* strukturiert und *gut kommentiert* sein. Dabei ist Folgendes zu beachten:

a) Nummerieren Sie die einzelnen Programmabschnitte und fügen Sie Abstände zwischen den Abschnitten ein.

b) Versehen Sie alle Programmblöcke und einzelne erklärungsbedürftige Befehle mit verständlichen und ausreichend detaillierten Kommentaren, damit wir die Programmschritte leichter nachvollziehen können. Insbesondere sind Makros ausführlich zu dokumentieren.

c) Geben Sie Variablen in Kommentaren mit Ihren Labeln an.

d) Rücken Sie Schleifen ein und beschreiben Sie ihren Inhalt bzw. Funktion.

e) Bleiben Sie einheitlich, z. B. bei der Groß- und Kleinschreibung, bei der Bezeichnung von fehlenden Werten („.“, „0“ oder „-x“) etc.

5. Dokumentation der Ergebnisse

Die einzelnen Ergebnisse müssen im Programmcode deutlich gekennzeichnet und gut dokumentiert sein. Bitte beachten Sie dabei die folgenden Punkte:

<u>Allgemeine Vorgaben</u>

a) Jedes Output-Element muss mit einem Titel versehen und fortlaufend durchnummeriert werden.

b) Bei allen statistischen Kennzahlen oder multivariaten Analysen sind die zugrunde liegende Anzahl der Beobachtungen (Fallzahlen) auszuweisen. Bitte achten Sie darauf, dass jede Auswertung auf Basis von mindestens 20 Beobachtungen erzeugt wird. Analyseergebnisse, die auf weniger als 20 Beobachtungen basieren, werden nicht übermittelt.

c) Bei der Geheimhaltung wird zusätzlich zu der Fallzahlregelung die (2,k)-Dominanzregel angewendet, d.h. ein Wert (z.B. Umsatz) ist dann geheim zu halten,

wenn die Summe der beiden größten Einheiten mehr als k % der totalen Summe betragen.

Zur Überprüfung der (2,k)-Dominanzregel wird Ihnen ein Programm zur Verfügung gestellt (in STATA dominance-ado-File).

d) Bei Verwendung von Hochrechnungsfaktoren bitte parallel ohne Hochrechnung rechnen.

e) Falls Sie Gewichtungsfaktoren verwenden, weisen Sie auch die ungewichteten Fallzahlen aus.

f) Da bei Variablen mit nur zwei Ausprägungen (z.B. Dummys) anhand des Mittelwertes die Anzahl der Fälle in den beiden Ausprägungen berechnet werden kann, weisen Sie für Dummyvariablen bei Mittelwerten neben der Fallzahl immer auch Minimum, Maximum und Standardabweichung aus. Anhand dieser Angaben können wir Variablen mit zwei Ausprägungen von anderen Variablen unterscheiden.

g) Minima und Maxima werden im Allgemeinen nicht herausgegeben.

h) Die oben genannte Fallzahlkontrolle trifft auch auf Graphiken zu. So werden z.B. grundsätzlich keine scatterplots übermittelt, bei denen jeder Punkt eine einzige Beobachtung darstellt.

i) Residualsplots sind grundsätzlich schutzbedürftig.

<u>Bei Aggregation der Ausgangsdaten</u>

Falls Sie die ursprünglichen Mikrodaten aggregieren (z.B. auf Ebene von Wirtschaftszweigen, Berufen) und in weiteren Analysen verenden, sollte Ihr Analyseprogramm folgende Angaben enthalten:

j) Machen Sie bitte die Aggregation gut kenntlich.

k) Bilden Sie eine Variable, welche die Anzahl der Beobachtungen pro Aggregat enthält.[3]

l) Zu allen im Anschluss an eine Aggregation erstellten Ergebnissen sind *in der Master-Datei für die entsprechenden Unterprogramme* folgende Informationen anzugeben:

- Aggregationsebene,
- Anzahl der Einheiten, die mindestens pro Datenzeile eingegangen sind,
- Name der Variable, welche die Unternehmensanzahl pro Datenzeile enthält.

<u>Tabellenoutput</u>

m) Bitte erzeugen Sie Tabellen, die handhabbar sind und eine sinnvoll darstellbare Größe nicht überschreiten.

n) Tabellen müssen dabei selbsterklärend sein, d.h. aus den Tabellenvorspalten und -zeilen muss der Tabelleninhalt eindeutig hervorgehen.

o) Beachten Sie die tabellenübergreifende Geheimhaltung, z.B. bei der Ausgabe der Tabellen „insgesamt“ und „weiblich“ würde sich die Tabelle „männlich“ durch die Differenzbildung der beiden Tabellen konstruieren lassen. Deswegen geben Sie gleich die Tabelle „männlich“ aus.

[3] Z.B. in Stata mit Hilfe des Befehls `bysort branche: gen N=_N.`