Übungszettel Indexerstellung

M.Psy.205, Dozent: Dr. Peter Zezula

Joshua Driesen (joshua.driesen@stud.uni-goettingen.de)

#German

Links

Übungszettel als PDF-Datei zum Drucken

Hinweise zur Bearbeitung

1. Bitte beantworten Sie die Fragen in einer .Rmd Datei. Sie können Sie über `Datei > Neue Datei > R Markdown...` eine neue R Markdown Datei erstellen. Den Text unter dem *Setup Chunk* (ab Zeile 11) können Sie löschen. Unter diesem Link können Sie auch unsere Vorlage-Datei herunterladen (Rechtsklick > Speichern unter…).
2. Informationen, die Sie für die Bearbeitung benötigen, finden Sie auf der Website der Veranstaltung
3. Zögern Sie nicht, im Internet nach Lösungen zu suchen. Das effektive Suchen nach Lösungen für R-Probleme im Internet ist tatsächlich eine sehr nützliche Fähigkeit, auch Profis arbeiten auf diese Weise. Die beste Anlaufstelle dafür ist der R-Bereich der Programmiererplattform Stackoverflow
4. Auf der Website von R Studio finden Sie sehr hilfreiche Übersichtszettel zu vielen verschiedenen R-bezogenen Themen. Ein guter Anfang ist der Base R Cheat Sheet

Aufgabe 1: SerienmörderInnen

Erste Schritte

Setzen Sie das gewünschte Arbeitsverzeichnis, und laden Sie das `tidyverse`-Paket. Laden Sie im Anschluss den Datensatz `https://pzezula.pages.gwdg.de/data/serial_killers.txt` (Quelle) herunter, oder lesen Sie ihn direkt aus der URL ein. Machen Sie sich die Struktur des Datensatzes klar.

Index Erstellen

Erstellen Sie einen Index, in dem Sie die Häufigkeit von SerienmörderInnen innerhalb eines Landes mit der Bevölkerung des Landes in Beziehung setzen. Skalieren Sie diesen so, dass ein Land, das genauso viele SerienmörderInnen hat, wie bei der Größe seiner Bevölkerung zu erwarten wäre, den Wert 10 erhält, und höhere Werte einer größeren Häufigkeit entsprechen. (Hinweis: Hierzu dürfen Sie voraussetzen, dass aus Ländern, die nicht im Datensatz vorkommen, keine Serienmorde bekannt sind. In Wahrheit verteilen sich "nur" 93,2% aller SerienmörderInnen auf die aufgeführten Länder.)

Kritik

Reflektieren Sie kritisch über die verwendeten Daten und mögliche Folgen für den von Ihnen erstellten Index. Gehen Sie hierbei insbesondere auf mögliche Konfundierungen ein.

Aufgabe 2: Geschlechterverhältnisse deutscher Städte

Datensatz einlesen

Lesen Sie den Datensatz `https://pzezula.pages.gwdg.de/data/Staedte.txt` ein. (Quelle: Statistisches Bundesamt, 2016) Hierin befindet sich eine Liste aller deutscher Städte, sortiert nach der Bevölkerungszahl. Die Bevölkerung finden Sie auch im Datensatz, einmal insgesamt, und einmal aufgeschlüsselt nach Geschlecht.

Index erstellen

Erstellen Sie einen Index, der das Geschlechterverhältnis der deutschen Städte abbildet. Dabei soll der Wertebereich *symmetrisch* um null herum verteilt sein, sprich: Der Betrag des Indexes soll für zwei gleich große, entgegengerichtete Ungleichverteilungen gleich groß sein.

Tipp: schauen Sie sich die mögliche Verteilung Ihres Index über den Wertebereich an.

Extremwerte verstehen

Suchen Sie anhand des von Ihnen erstellten Index die drei Städte mit dem größten Ungleichgewicht der beiden Geschlechter. Versuchen Sie mit einer Internetrecherche erste Ansatzpunkte für mögliche Gründe für diese Extremwerte zu ermitteln.

Mit der Zeit gehen

Die Daten wurden vor der Einführung des sog. dritten Geschlechts erhoben, und spiegeln daher die binäre Kategorisierung damaliger Rechtssprechung wieder:

```
all(Staedte$weiblich + Staedte$maennlich == Staedte$Gesamtbevoelkerung)
```

```
## [1] TRUE
```

Überprüfen Sie den von Ihnen erstellten Index darauf, ob er gegenüber der Einführung einer dritten Geschlechtskategorie robust ist, also bei gleich vielen Frauen und Männern trotzdem den Wert 0 annimmt. Falls nicht, überlegen Sie sich auch eine mögliche Korrektur!

Bonusaufgabe: Zipf's Law

Das Zipfsche Gesetz (Wikipedia) stammt ursprünglich aus der quantitativen Linguistik, und besagt: "Wenn die Elemente einer Menge – beispielsweise die Wörter eines Textes – nach ihrer Häufigkeit geordnet werden, ist die Wahrscheinlichkeit *p* ihres Auftretens umgekehrt proportional zur Position *n* innerhalb der Rangfolge: $p(n) \sim \frac{1}{n}$."

Es gibt einige Hinweise darauf, dass auch Städte innerhalb eines Landes diesem Gesetz folgen. Konkret heißt das: Die zweitgrößte Stadt hat halb so viele Einwohner wie die größte, die drittgrößte ein Drittel so viele, die viertgrößte ein Viertel, und so weiter.

Generieren Sie in Ihrem Datensatz eine neue Variable, in der sich die gemäß Zipf erwarteten Einwohnerzahlen der Städte befinden. Nutzen Sie im Anschluss einen Chi-Quadrat-Test (`chisq.test()`), um zu überprüfen, ob sich die tatsächliche und die erwartete Einwohnerverteilung signifikant voneinander unterscheiden.

Aufgabe 3: Bachelorbewerbungen

Datensatz einlesen

Lesen Sie den Datensatz `https://pzezula.pages.gwdg.de/data/bsc_bewerbung.txt` ein. In diesem finden Sie die Abiturnoten von 200 (fiktiven) BewerberInnen auf einen Bachelor-Studienplatz am GEMI. Machen Sie sich die Struktur des Datensatzes klar.

Notendurchschnitt erstellen

Erzeugen Sie zwei neue Variablen in dem Datensatz, in denen der Notendurchschnitt vermerkt wird. Behalten Sie für die erste Variable die gegebene Skalierung mit 0 bis 15 Punkten bei. Für die zweite Variable skalieren Sie bitte auf klassische Schulnoten von 1 bis 6 um. Wichtig: Achten Sie darauf, die Bewerbendennummer in der ersten Spalte *nicht* in den Notendurchschnitt miteinzubeziehen!

GEMI-Score

Bachelorstudienplätze am GEMI werden nicht bloß auf Basis der Abinote vergeben, sondern anhand einer eigenen Notengewichtungsformel. In diese geht die Gesamtnote zu 80%, die Englisch-Note zu 10%, und Deutsch und Mathe zu je 5% ein. Bilden Sie diesen Score in einer neuen Variable nach. Transformieren Sie diesen Score dabei auf den Wertebereich 0 bis 100!

Eigener Score

Reflektieren Sie über Ihr eigenes Bachelorstudium: Welche Fähigkeiten und welches Vorwissen erwies sich hierfür als nützlich? Erstellen Sie anhand dieser Überlegungen Ihren eigenen gewichteten Bewerbungsscore, ebenfalls mit dem Wertebereich 0 bis 100. Schreiben Sie ggf. ein paar Worte zur Begründung Ihrer Entscheidungen!

Zulassung

Leider können Sie von den 200 BewerberInnen nur 100 für ein Bachelorstudium zulassen. Erzeugen Sie sowohl für den GEMI-Score, als auch für Ihren selbst entwickelten Score eine neue Variable, in der die Zulassungsentscheidung anhand des jeweiligen Scores kodiert wird. Hierfür bietet sich der Variablentyp `logical` an.

Score-Vergleich

Analysieren Sie, in wieviel Prozent der Fälle der von Ihnen entwickelte und der GEMI-Score zu verschiedenen Ergebnissen kommen!

English

Links

Exercise sheet in PDF

Some hints

1. Please give your answers in a .Rmd file. You may generate one from scratch using the file menu: 'File > new file > R Markdown …' Delete the text below *Setup Chunk* (starting from line 11). Alternatively you may use this sample Rmd by donloading it.
2. You may find the informations useful that you can find on the start page of this course.
3. Don't hesitate to google for solutions. Effective web searches to find solutions for R-problems is a very useful ability, professionals to that too … A really good starting point might be the R area of the programmers platform Stackoverflow
4. You can find very useful cheat sheets for various R-related topics. A good starting point is the Base R Cheat Sheet.

Task 1: Serial killers

First steps

Load the tidyverse and download the data frame at `https://pzezula.pages.gwdg.de/data/serial_killers.txt` (Source). You can also read in the data frame directly from the URL. Take a look at the the data structure.

Index creation

Create an index that puts the frequency of serial killers in a country in relation to the population of that country. It should be scaled in a way that gives a value of 10 to countries having exactly as many serial killers as would be expected from their population size. Higher values should indicate an elevated frequency. (Note: For this task, you can assume that countries not listed here don't have any known serial killers. In reality, 'only' 93.2% of all known serial killers come from the countries listed here.)

Critique

Reflect critically on the data used here and possible consequences for the index you created. Take special note of possible confounders.

Task 2: Gender ratios of German cities

Read data

Read in the data frame `https://pzezula.pages.gwdg.de/data/Staedte.txt`. (Source: Statistisches Bundesamt, 2016) In it, you'll find a list of all German cities, sorted by size of population. The population's in the data frame, too, once as total and once broken down by sex.

German	English
Staedte	cities
Rang	rank
Bundesland	federal state
Stadt	city
Flaeche	area
Gesamtbevoelkerung	total population
maennlich	male
weiblich	female

Index creation

Create an index representing the ratio of the two sexes. The scale has to be *symmetrical* around zero, i.e. the absolute value of the score of two inequalities in opposing directions, but of the same size, should be the same.

Understanding extreme values

Use your index to find the three cities with the biggest inequalities between male and female inhabitants. Try researching the web to find clues how these inequalities come to be. (Note: this might prove impossible without German Wikipedia for the smaller cities. If so, skip to the next task.)

Keeping up with the times

The present data were aggregated before the legal introduction of the so-called 'third gender', which is why they still contain the binary gender categorization:

```
all(Staedte$weiblich + Staedte$maennlich == Staedte$Gesamtbevoelkerung)
```

```
## [1] TRUE
```

Check whether your index could handle the introduction of a third gender categorie, i.e. if it would still be zero when there are the same number of men and women. If not, try to fix that!

Bonus task: Zipf's Law

Zipf's Law (Wikipedia) originally stems from quantitative linguistics and states: If you sort the elements of a set - e.g. words from a text - by their frequency, then the probability of one element's occurence *p* is inversely proportianal to its position *n* in the ranked order: $p(n) \sim \frac{1}{n}$."

There's some evidence that cities within one country follow this law as well. In practice, this means: There's half as many inhabitants in the second biggest city as in the biggest, a third as many in the third biggest as in the biggest, a fourth as many in the fourth biggest etc.

Add a new variable to your data frame containing the population size predictions based on Zipf's law. Next, use a chi-square-test (`chisq.test()`) to check whether the real and the expected distributions of population differ significantly.

Task 3: Bachelor applications

Read data

Read in the data at `https://pzezula.pages.gwdg.de/data/bsc_bewerbung.txt`. In it, you'll find the "Abitur" (high school diploma) grades from 200 (fictional) applicants for the psychology bachelor at the GEMI. Check out the structure of the data.

German	English
bsc_bewerbung	bachelor application
Nr	no.
Deutsch	German
Englisch	English
ZweiteFremdsprache	SecondForeignLanguage
Mathe	Maths
Bio	biology
Chemie	chemistry
Physik	physics
Geschichte	history
Erdkunde	geography
Kunst	art
Sport	P.E.

Create grade average

Create two new variables for the grade average. In the first, keep the original scaling from 0 to 15 points. For the second, rescale it to classic school grades from 1 to 6. Note: Take care *not* to include the applicant number in the first column into the grade average!

GEMI-Score

At the GEMI, bachelor spots are not distributed by the grade average alone, but by a special weighted average. This contains 80% overall average, an extra 10% for the English grade, and 5% for German and for maths, respectively. Add a new variable for this score. Rescale the score so that its theoretical range is 0 to 100!

Your own Score

Reflect on your own bachelor's study: Which skills were useful? Which pre-existing knowledge helped you? Use these considerations to create your own weighted sum score, again scaling it to 0 to 100. You might want to write down a few points explaining your reasoning.

Admission

Unfortunately, you can only accept 100 of the 200 applicants. Create a new variable for the GEMI score and another for your own score in which the admission or non-admission of the applicants is recorded. The variable-type `logical` is probably best for this.

Comparing the scores

Analyse in how many percent your own score and the GEMI score would lead to different decisions!