Einführung in R

Clemens Brunner

14.-15.2.2019

Korrelation

Hintergrund

Oft ist es interessant zu fragen, ob zwei Variablen eine gegenseitige Abhängigkeit zeigen. Man möchte also wissen, ob sich die beiden Variablen ähnlich verhalten oder nicht - wenn die eine Variable zunimmt, nimmt dann die andere Variable auch zu (oder ab)? Die Korrelation ist ein einfaches und populäres Maß um diese Fragestellung zu beantworten.

Folgende Beispiele von der Website Spurious Correlations, welche zwei Variablen in einer Grafik darstellen, illustrieren die Gefahr der fehlerhaften Interpretation von Korrelationen. Im ersten Beispiel werden die Ausgaben für den Bereich Wissenschaft, Raumfahrt und Technik der USA im Zeitraum 1999-2009 dargestellt. Gleichzeitig werden die Anzahl der Selbstmorde durch Hängen, Strangulieren und Erstickung im selben Zeitraum gezeigt. Man sieht, dass beide Kurven einen sehr ähnlichen Verlauf haben, und der Korrelationskoeffizient (dazu später mehr) ist mit $r = 0.998$ extrem hoch.

Das zweite Beispiel zeigt den Zusammenhang zwischen der Anzahl an Leuten, die in einen Pool gefallen und ertrunken sind und der Anzahl an Filmen mit Nicolas Cage im Zeitraum 1999-2009. Auch hier kann man einen schönen Zusammenhang erkennen, welcher einen recht hohen Korrelationskoeffizienten von $r = 0.666$ aufweist.

Diese Beispiele sollen verdeutlichen, dass Korrelation nicht automatisch einen kausalen Zusammenhang darstellt ("correlation is not causation").

Produkt-Moment-Korrelation (Pearson-Korrelation)

Die Produkt-Moment-Korrelation r (auch Pearson-Korrelation genannt) ist ein Maß für den Grad des *linearen* Zusammenhangs zweier intervallskalierter Variablen. Die Korrelation nimmt Werte zwischen -1 und 1 an und wird aus den Varianzen und der Kovarianz beider Variablen berechnet.

Die Varianz einer Variable x ist wie folgt definiert:

$$\mathrm{Var}(x) = \frac{1}{N-1} \sum_{i=1}^{N} (x_i - \bar{x})^2$$

Hier ist $\bar{x}$ der Mittelwert über alle N Werte (welche als x_i bezeichnet werden), also

$$\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i.$$

Alternativ kann man die Varianz auch so schreiben:

$$\mathrm{Var}(x) = \frac{1}{N-1} \sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})$$

Die Varianz beschreibt, wie stark die Datenpunkte um den Mittelwert variieren. Dementsprechend ist die Kovarianz zwischen zwei Variablen x und y definiert als

$$\mathrm{Cov}(x, y) = \frac{1}{N-1} \sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y}).$$

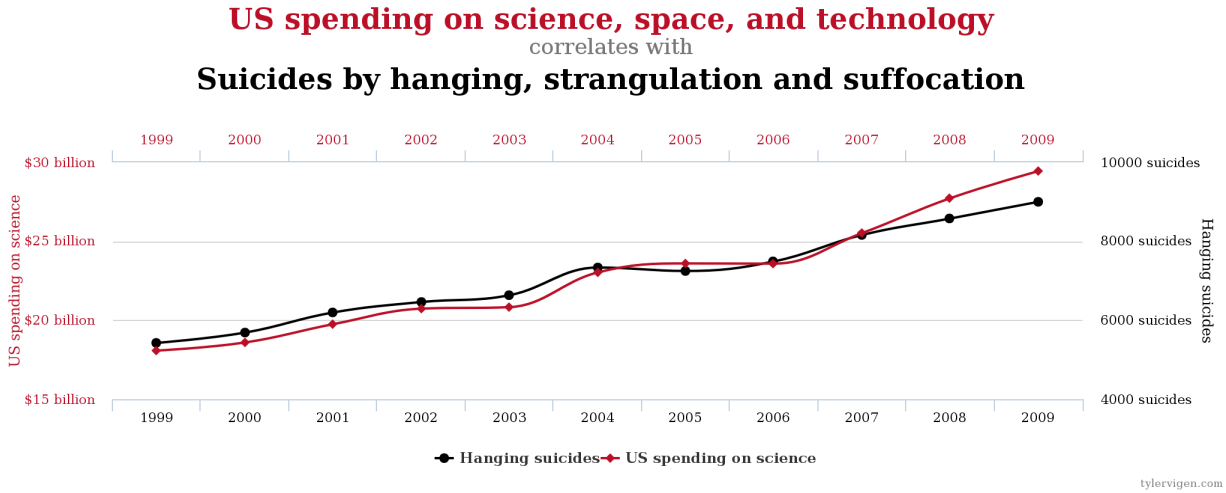


Figure 1: Beispiel 1

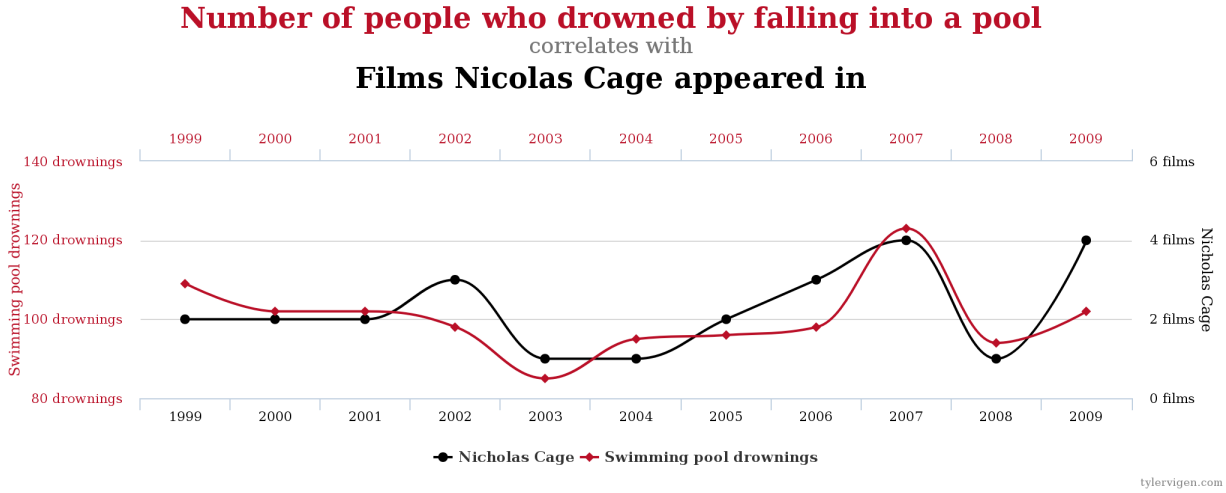


Figure 2: Beispiel 2

Die Kovarianz beschreibt, wie stark die beiden Variablen gemeinsam um den jeweiligen Mittelwert variieren. Eine positive Kovarianz bedeutet, dass beide Variablen gleichsinnig variieren (d.h. wenn eine Variable größer als der Mittelwert ist, dann ist die andere auch größer). Umgekehrt bedeutet eine negative Kovarianz, dass beide Variablen gegensinnig variieren (ist eine Variable größer als der Mittelwert, dann ist die andere Variable kleiner).

Die Kovarianz ist kein standardisiertes Maß, d.h. man kann nicht einfach zwei Kovarianzen aus unterschiedlichen Messreihen miteinander vergleichen. Die Pearson-Korrelation standardisiert nun die Kovarianz mit den Varianzen der einzelnen Variablen, so dass die Korrelation im Wertebereich zwischen -1 und 1 liegt:

$$r = \frac{\text{Cov}(x, y)}{\sqrt{\text{Var}(x)\text{Var}(y)}}$$

Signifikanz

Meist wird nach der Berechnung der Korrelation ein Test durchgeführt, welcher prüft, ob die erhaltene Korrelation signifikant von der Nullhypothese ("es existiert keine Korrelation", also $r = 0$) abweicht. Da die Stichprobenverteilung der Korrelation keine Normalverteilung aufweist, kann man den Wert von r mit Hilfe der Fisher-Transformation in eine Normalverteilung umwandeln. Der Mittelwert beträgt dann

$$z_r = \frac{1}{2} \ln \frac{1 + r}{1 - r} = \text{arctanh}(r)$$

und der Standardfehler ist

$$\text{SE}(z_r) = \frac{1}{\sqrt{N - 3}}.$$

Unter der Nullhypothese ($r = 0$) kann man den erhaltenen Wert von z_r durch den Standardfehler dividieren und das Ergebnis dann mit Werten aus einer Tabelle der Standardnormalverteilung vergleichen. So erhält man den p-Wert (siehe auch hier).

Betrachten wir dazu ein Beispiel zur Veranschaulichung der Berechnung des p-Wertes sowie des Konfidenzintervalls für eine gegebene Pearson-Korrelation. Gegeben sei eine Korrelation $r = 0.25$ berechnet aus einer Stichprobe mit dem Umfang $N = 40$:

```
r <- 0.25
N <- 40
```

Weiters geben wir ein Signifikanzniveau von $\alpha = 0.05$ vor:

```
alpha <- 0.05
```

Wir möchten nun wissen, ob die Korrelation r mit dem gegebenen Signifikanzniveau signifikant unterschiedlich von 0 ist. Dazu berechnen wir den p-Wert und das Konfidenzintervall. Um dies tun zu können, müssen wir zuerst die Fisher-Transformation von r berechnen:

```
z_r <- atanh(r)
z_r
```

```
[1] 0.2554128
```

Jetzt können wir den Standardfehler berechnen:

```
se_z <- 1 / sqrt(N - 3)
se_z
```

```
[1] 0.164399
```

Wir standardisieren nun z_r, d.h. wir dividieren durch den Standardfehler, sodass der resultierende Wert z standardnormalverteilt ist, d.h. Mittelwert 0 (Nullhypothese) und Standardabweichung 1 hat. Dies ermöglicht es uns, in einer Standardnormalverteilungstabelle die Wahrscheinlichkeit herauszusuchen, dass der erhaltene

Wert (oder ein größerer) unter der Nullhypothese auftreten würde. Diese Wahrscheinlichkeit nennt man p-Wert.

```
z <- z_r / se_z
z
```

```
[1] 1.553615
```

Da wir keine gerichtete Hypothese haben, führen wir einen zweiseitigen Test durch, d.h. wir verdoppeln den Wert aus der Tabelle. Wenn man nicht in einer Tabelle nachschlagen will, kann die Funktion `pnorm` diese Berechnungen durchführen:

```
p <- 2 * (1 - pnorm(z))
p
```

```
[1] 0.1202762
```

Dieser p-Wert ist größer als 0.05, d.h. die Nullhypothese kann nicht verworfen werden. Das Konfidenzintervall um z_r erhält man, indem man zum gegebenen Wert z_r das Produkt aus dem Signifikanzniveau entsprechenden Quantil (ca. 1.96 für $\alpha = 0.05$) mit dem Standardfehler addiert bzw. subtrahiert. Die Funktion `qnorm` gibt dieses Quantil zurück:

```
cl_z <- z_r - qnorm(1 - alpha/2) * se_z
cu_z <- z_r + qnorm(1 - alpha/2) * se_z
c(cl_z, z_r, cu_z)
```

```
[1] -0.06680328  0.25541281  0.57762891
```

Man beachte, dass es sich bei allen drei Werten um Fisher-transformierte Werte handelt. Möchte man ein Konfidenzintervall um die ursprüngliche Korrelation r angeben, so muss man diese drei Werte noch rücktransformieren:

```
cl_r <- tanh(cl_z)
cu_r <- tanh(cu_z)
print(c(cl_r, r, cu_r))
```

```
[1] -0.06670409  0.25000000  0.52093993
```

Die Tatsache, dass die Korrelation $r = 0.25$ bei einer Stichprobe von $N = 40$ nicht signifikant ist ($p = 0.12$) kann man auch daran erkennen, dass das 95%-Konfidenzintervall den Wert 0 enthält.

Spearman Rangkorrelationskoeffizient

Im Gegensatz zur Pearson-Korrelation misst der Spearman Rangkorrelationskoeffizient ρ nicht nur lineare Zusammenhänge zwischen zwei Variablen, sondern der Zusammenhang kann eine beliebige monotone Funktion sein. Die beiden Variablen müssen auch nicht intervallskaliert sein, d.h. so kann man auch ordinalskalierte Daten miteinander korrelieren.

Im Prinzip berechnet man die Spearman-Korrelation, indem man die Daten x und y vorher in Ränge konvertiert und dann die Pearson-Korrelation berechnet. Zur Berechnung kann folgende vereinfachte Formel verwendet werden:

$$\rho = 1 - \frac{6 \sum d_i^2}{N \cdot (N^2 - 1)}$$

Hier ist d_i die Differenz der Ränge einer Beobachtung.

Kendall Rangkorrelationskoeffizient

Bei kleinen Stichproben und einer relativ großen Anzahl an gleichen Rängen liefert der Kendall Rangkorrelationskoeffizient τ oft bessere Ergebnisse. Hier werden nicht die Differenzen zwischen den Rängen betrachtet

(also die Abstände der Ränge zwischen beiden Variablen), sondern nur ob es Unterschiede in den Rängen zwischen Datenpaaren gibt oder nicht.

Korrelationen mit R berechnen

Korrelationskoeffizienten kann man mit den folgenden drei Funktionen berechnen: `cor`, `cor.test` und `rcorr`. Die ersten beiden Funktionen sind Teil von R, die Funktion `rcorr` muss mit dem `Hmisc`-Paket geladen werden.

```
library(Hmisc)
```

Die drei Funktionen haben unterschiedliche Features, welche in nachfolgender Tabelle zusammengefasst sind.

	Pearson	Spearman	Kendall	p-Werte	CI	Multiple
`cor`	x	x	x			x
`cor.test`	x	x	x	x	x	
`rcorr`	x	x		x		x

Funktion `cor`

Die Funktion `cor` ruft man wie folgt auf:

```
cor(x, y, method="pearson")
```

Hier übergibt man zwei Variablen und spezifiziert, welche Korrelation berechnet werden soll (standardmäßig wird die Pearson-Korrelation berechnet). Wenn `x` ein Data Frame mit mindestens zwei Spalten ist, kann man `y` weglassen - dann wird automatisch die Korrelation zwischen allen Spalten von `x` berechnet.

Funktion `cor.test`

Der Aufruf der Funktion `cor.test` ist sehr ähnlich:

```
cor.test(x, y, alternative="t", method="pearson", conf.level=0.95)
```

Hier kann man die Form der Alternativhypothese (`two-sided`, `greater`, `less`) sowie das Konfidenzniveau angeben. Diese Funktion kann nur mit genau zwei Variablen umgehen (im Gegensatz zu den anderen beiden Funktionen, welche automatisch alle Paare von mehreren Variablen bilden können).

Funktion `rcorr`

Die Funktion `rcorr` verwendet man wie folgt:

```
rcorr(x, y, type="pearson")
```

Hier ist zu beachten, dass die Daten in einer Matrix vorliegen müssen. Weiters ist es wie bei `cor` möglich, nur das Argument `x` anzugeben (wenn dieses eine Matrix mit mindestens zwei Spalten ist).

Beispiel

Am besten können die Funktionsweisen der drei Funktionen anhand eines Beispiels veranschaulicht werden. Dazu laden wir einen Datensatz über Prüfungsangst und speichern diesen in der Variable `exam` ab:

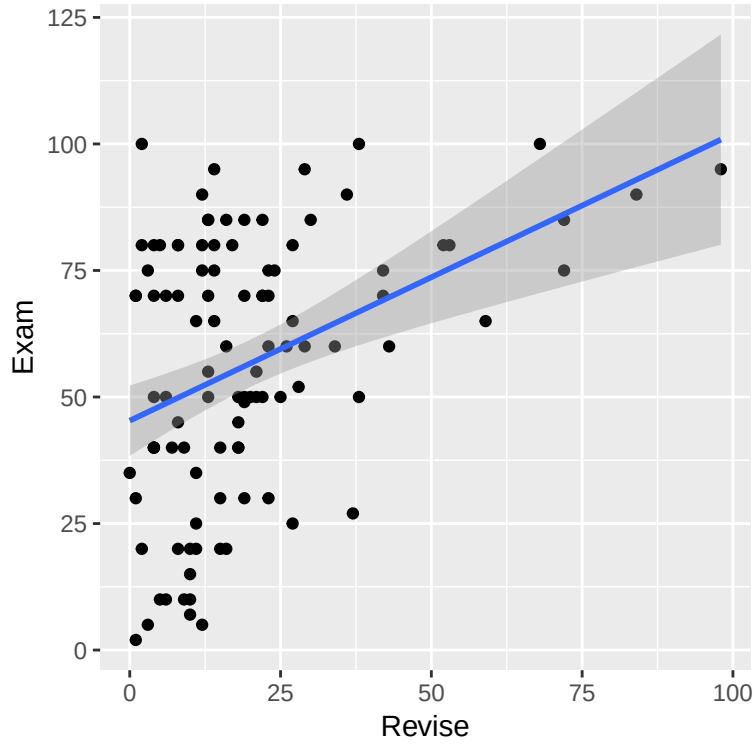
```
library(readr)
exam <- read_tsv("exam.dat")
exam
```

```
# A tibble: 103 x 4
   Revise  Exam Anxiety Gender
    <dbl> <dbl>   <dbl> <chr>
```

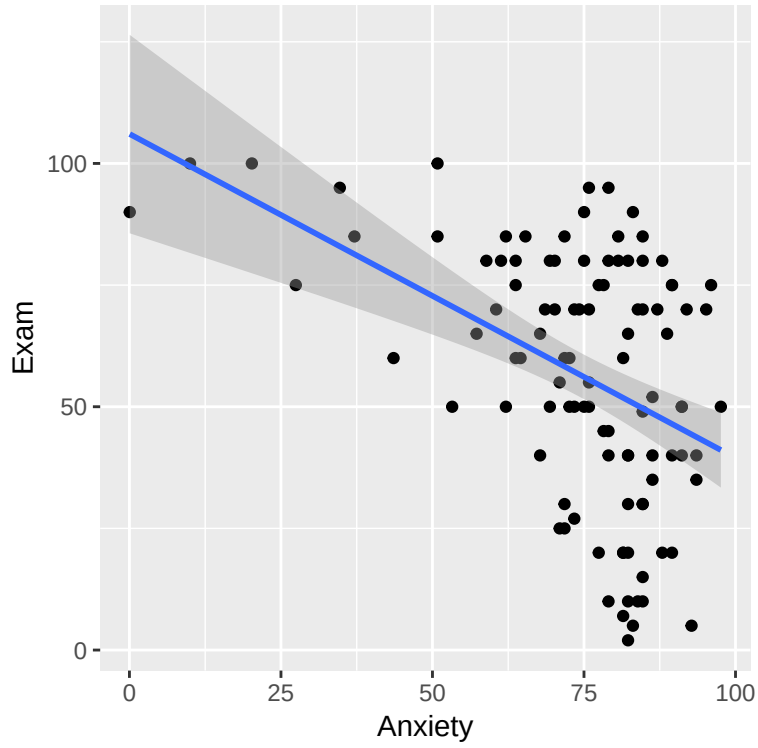
```
 1      4    40    86.3 Male
 2     11    65    88.7 Female
 3     27    80    70.2 Male
 4     53    80    61.3 Male
 5      4    40    89.5 Male
 6     22    70    60.5 Female
 7     16    20    81.5 Female
 8     21    55    75.8 Female
 9     25    50    69.4 Female
10     18    40    82.3 Female
# ... with 93 more rows
```

Es ist hilfreich, die Daten zuerst einmal grafisch darzustellen. Für den Zusammenhang zwischen zwei Variablen bietet sich ein Scatterplot (inklusive Regressionsgerade) an.

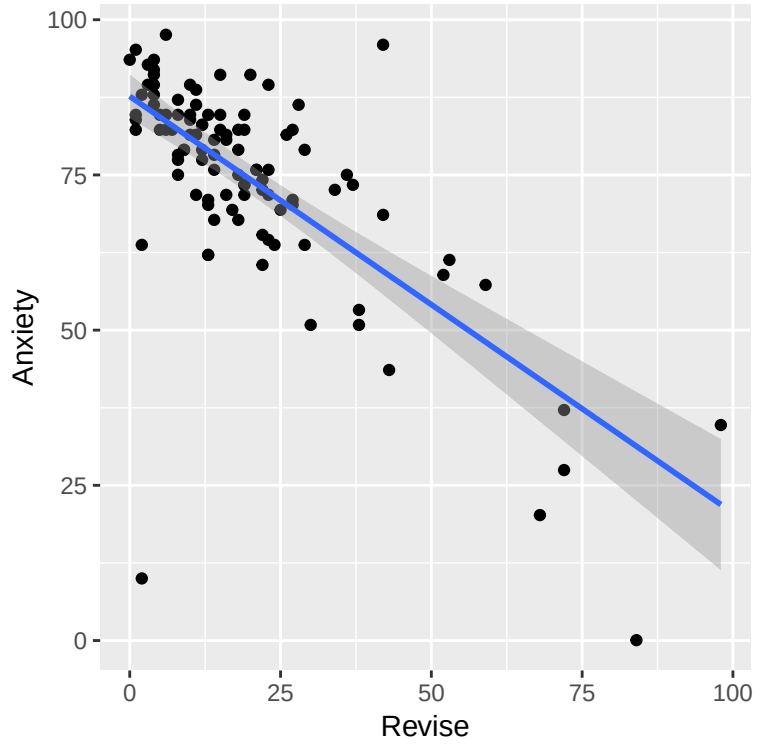
```
library(ggplot2)
ggplot(exam, aes(Revise, Exam)) + geom_point() + geom_smooth(method=lm)
```




```
ggplot(exam, aes(Anxiety, Exam)) + geom_point() + geom_smooth(method=lm)
```

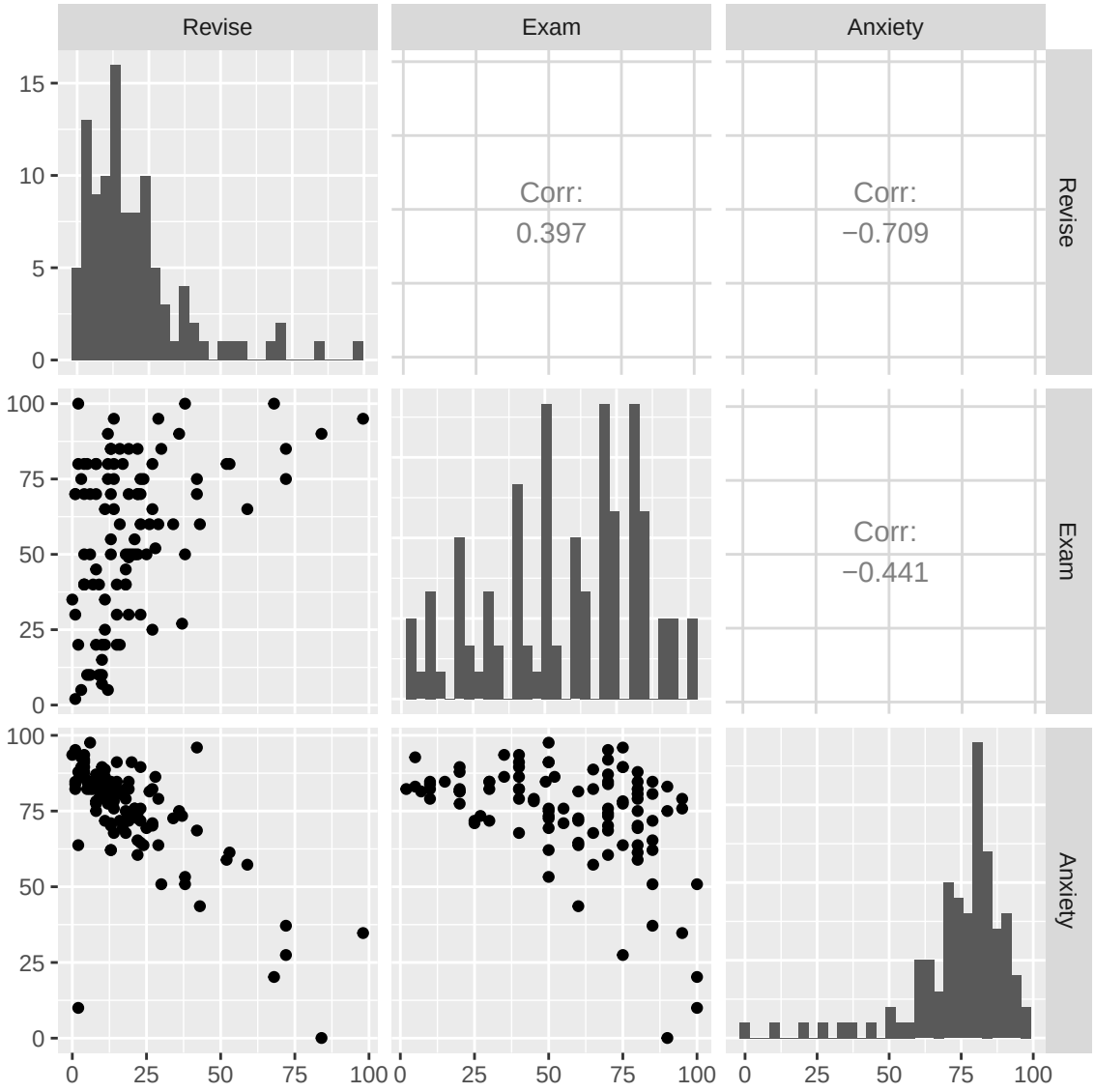



```
ggplot(exam, aes(Revise, Anxiety)) + geom_point() + geom_smooth(method=lm)
```



Bei mehr als zwei Variablen (wie in diesem Beispiel) kann man die Funktion `ggpairs` aus dem Paket `GGally` (ein Paket, welches `ggplot2` um einige praktische Funktionen erweitert) verwenden, um alle paarweisen Plots darzustellen:

```
library(GGally)
ggpairs(exam, columns=1:3, diag=list(continuous="barDiag"))
```



Hier sieht man außerdem bereits die Korrelationen zwischen allen Paaren.

Pearson-Korrelation

Nun berechnen wir die Pearson-Korrelationen zwischen den drei Variablen `Exam`, `Anxiety` und `Revise`:

```
cor(exam[, c("Exam", "Anxiety", "Revise")])
```

```
              Exam    Anxiety     Revise
Exam     1.0000000 -0.4409934  0.3967207
Anxiety -0.4409934  1.0000000 -0.7092493
Revise   0.3967207 -0.7092493  1.0000000
```

Man kann aus dieser Korrelationsmatrix direkt die einzelnen Koeffizienten für alle Variablenpaare ablesen. Die Diagonale beinhaltet die Korrelationen der Variablen mit sich selbst und besteht daher aus lauter Werten

die exakt gleich 1 sind. Außerdem ist es egal, ob man die Korrelationen in dem Dreieck unter der Diagonale oder über der Diagonale abliest, da die Korrelationsmatrix symmetrisch ist.

Möchte man jedoch auch p-Werte, muss man die Funktion `rcorr` verwenden. Diese Funktion erwartet die Daten jedoch nicht als Data Frame, sondern als Matrix. Daher müssen die Daten beim Aufruf der Funktion in eine Matrix umgewandelt werden:

```
rcorr(as.matrix(exam[, c("Exam", "Anxiety", "Revise")]))
```

```
         Exam Anxiety Revise
Exam     1.00   -0.44   0.40
Anxiety -0.44    1.00  -0.71
Revise   0.40   -0.71   1.00

n= 103


P
        Exam Anxiety Revise
Exam           0       0
Anxiety  0             0
Revise   0     0
```

Zusätzlich zur Korrelationsmatrix bekommt man auch die p-Werte geliefert. In diesem Beispiel sind alle Korrelationen signifikant, da die p-Werte sehr klein sind (gerundet Null).

Wenn man auch Konfidenzintervalle haben möchte, muss man die Funktion `cor.test` verwenden. Diese Funktion unterstützt aber nur zwei Variablen, d.h. bei mehreren Variablen muss man die Funktion mehrmals aufrufen um alle paarweisen Korrelationen zu erhalten.

```
cor.test(exam$Anxiety, exam$Exam)
```

```
    Pearson's product-moment correlation

data:  exam$Anxiety and exam$Exam
t = -4.938, df = 101, p-value = 3.128e-06
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.5846244 -0.2705591
sample estimates:
       cor
-0.4409934
```

Spearman Rangkorrelationskoeffizient

Für die Beispieldaten können wir analog auch die Spearman-Korrelation bestimmen:

```
cor(exam[, c("Exam", "Anxiety", "Revise")], method="spearman")
```

```
             Exam    Anxiety     Revise
Exam     1.0000000 -0.4046141  0.3498948
Anxiety -0.4046141  1.0000000 -0.6219694
Revise   0.3498948 -0.6219694  1.0000000
```

```
rcorr(as.matrix(exam[, c("Exam", "Anxiety", "Revise")]), type="spearman")
```

```
         Exam Anxiety Revise
Exam     1.00   -0.40   0.35
```

```
Anxiety -0.40    1.00  -0.62
Revise   0.35   -0.62   1.00

n= 103


P
        Exam  Anxiety Revise
Exam          0e+00   3e-04
Anxiety 0e+00         0e+00
Revise  3e-04 0e+00
```

```
cor.test(exam$Revise, exam$Exam, method="spearman")
```

```
Warning in cor.test.default(exam$Revise, exam$Exam, method = "spearman"): Cannot compute exact p-
value with ties


    Spearman's rank correlation rho

data:  exam$Revise and exam$Exam
S = 118390, p-value = 0.0002913
alternative hypothesis: true rho is not equal to 0
sample estimates:
      rho
0.3498948
```

Kendall Rangkorrelationskoeffizient

Die Funktion `cor.test` gibt eine Warnung aus, dass der berechnete p-Wert nicht exakt ist, da die Daten gleiche Ränge beinhalten. In solchen Fällen ist daher der Kendall-Korrelationskoeffizient die bessere Wahl:

```
cor.test(exam$Revise, exam$Exam, method="kendall")
```

```
    Kendall's rank correlation tau

data:  exam$Revise and exam$Exam
z = 3.8034, p-value = 0.0001427
alternative hypothesis: true tau is not equal to 0
sample estimates:
      tau
0.2633259
```

Übungen

Übung 1

Für diese Übung werden Sie Daten aus folgendem Artikel verwenden und ausgewählte Ergebnisse selbst nachrechnen und überprüfen: T. Chamorro-Premuzic, A. Furnham, A. N. Christopher, J. Garwood, G. N. Martin. Birds of a feather: Students' preferences for lecturers' personalities as predicted by their own personality and learning approaches. Personality and Individual Differences, 44(4), 965-976, 2008.

Tabelle 3 auf Seite 972 in diesem Artikel ist für uns interessant. Berechnen Sie selbst die Korrelationen zwischen den Persönlichkeitsmerkmalen N, E, O, A und C der Studenten und der Lehrenden mit den Daten in der Datei `birds.csv` (die Sie von http://bit.ly/r_example_data herunterladen können). Vergleichen Sie die von Ihnen berechneten 25 Werte mit den Werten aus der Tabelle. Stimmen Ihre Ergebnisse mit den

publizierten Daten überein? Berechnen Sie sowohl Korrelationen als auch Signifikanzen und geben Sie etwaige Abweichungen zum Paper an.

Übung 2

Verwenden Sie die Beispieldaten `mtcars` und analysieren Sie den Zusammenhang zwischen den Variablen `mpg`, `disp` und `hp`. Stellen Sie den Zusammenhang zwischen den Variablenpaaren grafisch dar, und berechnen Sie danach Pearson- und Spearman-Korrelationen!

Übung 3

In der Datei `pm10.csv` (die Sie von http://bit.ly/r_example_data herunterladen können) finden Sie die monatlichen Feinstaubwerte PM10 von zwei Messstationen in Graz im Zeitraum Februar 2006 bis Mai 2016. Führen Sie folgende Analysen durch:

- Erstellen Sie eine Liniengrafik, in der Sie den Verlauf der PM10-Konzentration von beiden Messstationen über die Zeit darstellen.
- Erstellen Sie einen Scatterplot, der den Zusammenhang zwischen den beiden Messstationen darstellt.

Berechnen Sie abschließend die Pearson-Korrelation zwischen den Daten beider Messstationen inklusive Konfidenzintervall und p-Wert.

Anmerkung 1: Wandeln Sie die Spalte `Datum` nach dem Einlesen mittels `as.Date` in einen Datums-Typ um.

Anmerkung 2: Für die Erstellung der ersten Grafik sollten Sie die Daten vorher ins Long-Format umwandeln:

```
library(tidyr)
pm10_long <- gather(pm10, key="Ort", value="pm10", -Datum)
```

Danach können Sie die Grafik mit der Variable `pm10_long` erstellen.

Anmerkung 3: Wenn Sie die Funktion `cor` zur Berechnung der Korrelation zwischen den beiden Variablen verwenden möchten, können Sie fehlende Werte in jeder der beiden Variablen mit dem Argument `use="complete.obs"` ausschließen.

Übung 4

In dieser Einheit haben wir die Signifikanz einer Korrelation $r = 0.25$ bei einer Stichprobengröße von $N = 40$ berechnet. Diese war mit $p = 0.12$ nicht signifikant. Welche Stichprobengröße N müssten Sie wählen, um ein auf $\alpha = 0.05$ signifikantes Ergebnis zu erhalten?

Bei welcher Stichprobengröße wird sogar eine sehr kleine Korrelation von $r = 0.05$ signifikant?

(Take-Home-Message dieser Übung: Der p-Wert allein sagt fast gar nichts über das Ergebnis aus!)

Diese Unterlagen sind lizenziert unter einer Creative Commons Namensnennung - Nicht-kommerziell - Weitergabe unter gleichen Bedingungen 4.0 International Lizenz.