Die t-Verteilung und der t-Test

Jonathan Harrington

21 Mai 2019

```
library(ggplot2)
source(file.path(pfadu, "proben.R"))
form = read.table(file.path(pfadu, "bet.txt"))
e.df = read.table(file.path(pfadu, "e.txt"))
```

1. SE (Standard Error) und Konfidenzintervall

Zur Erinnerung: Der Standard Error (SE) ist die Populationsstandardabweichung von Mittelwerten. Wir werfen beispielsweise 12 Würfel gleichzeitig, und berechnen den Mittelwert der Zahlen. Der SE lässt sich berechnen mit $\sigma/\sqrt{k}$:

- σ ist die Populationsstandardabweichung = `sd(1:6) * sqrt(5/6)`
- k ist die Anzahl der Würfel = 12

```
SE = sd(1:6) * sqrt(5/6) / sqrt(12)
SE
```

```
## [1] 0.4930066
```

Die Bedeutung hiervon ist: wenn wir 12 Würfel gleichzeitg werfen, davon den Mittelwert berechnen, diesen Vorgang unendlich viel Mal wiederholen, sodass wir unendlich viele Mittelwerte hätten, dann davon die Stichprobenstandardabweichung berechnen, dann wäre diese Standardabweichung genau `SE = sd(1:6) * sqrt(5/6) / sqrt(12)`. Wir müssten ziemlich nah an diesen SE mit z.B. 50000 Mittelwerten kommen:

```
o = proben(k=12, N = 50000)
sd(o)
```

```
## [1] 0.4913101
```

Konfidenzintervall

Wir wollen zwei Werte a und b auf eine solche Weise berechnen, sodass der Mittelwert zwischen a und b mit einer Wahrscheinlichkeit von 0.95 liegt. Hier benötigen wir den Populations-Mittelwert:

```
mu = mean(1:6)
# und SE
SE = sd(1:6) * sqrt(5/6) / sqrt(12)
# a und b
a = qnorm(0.025, mu, SE)
b = qnorm(0.975, mu, SE)
a
```

```
## [1] 2.533725
```

```
b
```

```
## [1] 4.466275
```

Das heißt also, dass, wenn wir 12 Würfel zusammen werfen und davon den Mittelwert berechnen, der Mittelwert dann mit einer Wahrscheinlichkeit von 0.95 zwischen (a) 2.533725 und (b) 4.466275 (= ein Mittelwert von weniger als 2.533725 oder mehr als 4.466275 wird meistens nur in 5/100 Fällen vorkommen) fällt.

Prüfen wir das: 12 Würfel werfen, davon den Mittelwert berechnen, diesen Vorgang 100 Mal wiederholen (daher 100 Mittelwerte):

```
m = proben(k = 12, N = 100)
```

Wieviele dieser Mittelwerte sind kleiner als 2.533725 oder größer als 4.466275?

```
sum(m < 2.533725 | m > 4.466275)
```

```
## [1] 5
```

2. Konfidenzintervall einer Stichprobe

Hier sind 12 Dauerwerte von einem /a:/ Vokal:

```
d = c(119, 111, 105, 130, 133, 122, 124, 129,  95, 100, 109, 111)
```

Wir wollen aufgrund der Stichprobe ein Konfidenzintervall für den Dauer-Mittelwert von /a:/ erstellen (= dieser Mittelwert fällt zwischen a und b mit einer Wahrscheinlichkeit von 95%). Zu diesem Zweck müssen ...

i. (mu, SE) aufgrund der Stichprobe wie folgt eingeschätzt werden, angenommen dass es sich um eine randomisiert ausgewählte Stichprobe handelt: beste Einschätzung von mu:

```
mu = mean(d)
```

Die beste Einschätzung von SE:

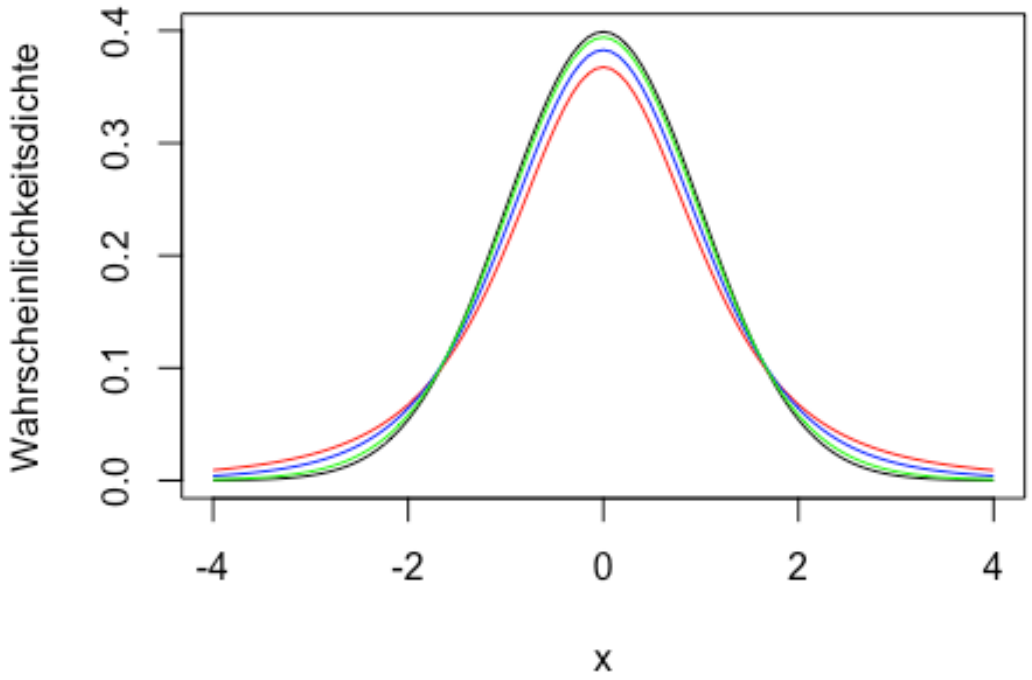
```
SE = sd(d)/sqrt(12)
```

ii. Wenn (mu, SE) nicht aus theoretischen Überlegungen (wie beim Würfel-Spiel) heraus berechnet werden können, dann wird das Konfidenzintervall *nicht durch die Normalverteilung*, sondern durch die **t-Verteilung** mit einer gewissen Anzahl von sogenannten *Freiheitsgraden* erstellt.

Die t-Verteilung ist der Normalverteilung sehr ähnlich, und nähert sich der Normalverteilung zunehmend an, je höher die Anzahl der Freiheitsgrade ist, z.B.

```
# t-Verteilung
# mit 3, 6, 20 Freiheitsgrade auf die Normalverteilung überlagern:
ylab = "Wahrscheinlichkeitsdichte"
curve(dnorm(x, 0, 1), xlim = c(-4, 4), ylab = ylab)
```

```r
# t-Verteilung mit 3 df
curve(dt(x, 3), add = T, col = "red")
# t-Verteilung mit 6 df
curve(dt(x, 6), add = T, col = "blue")
# t-Verteilung mit 20 df
curve(dt(x, 20), add = T, col = "green")
```



Die Anzahl der **Freiheitsgrade** entspricht der **Anzahl der Stichproben minus 1**:

```r
df = 11
```

Das Konfidenzintervall für eine Stichprobe wie die obige wird mit `qt()` statt mit `qnorm()` berechnet:

```r
a = mu + SE * qt(0.025, df)
b1 = mu - SE * qt(0.025, df)
# oder
b2 = mu + SE * qt(0.975, df)
a
```

```
## [1] 107.8177
```

```r
b1
```

```
## [1] 123.5156
```

```
b2

## [1] 123.5156
```

Bedeutung: aufgrund dieser Stichprobe fällt der Dauermittelwert für /a/ zwischen 107.8 ms und 123.5 ms mit einer Wahrscheinlichkeit von 0.95 (also 95%).

3. Konfidenzintervall für den Unterschied zwischen 2 Stichproben

Die Dauerwerte, wenn 12 Sprecher (Vpn1, Vpn2… Vpn12) ein betontes /a/ produzierten, waren wie folgt:

```
bet = c(119, 111, 105, 130, 133, 122, 124, 129,  95, 100, 109, 111)
```

Die Dauerwerte derselben 12 Sprecher (Vpn1, Vpn2… Vpn12), die ein unbetontes /a/ produzierten, waren:

```
un = c(110, 95, 108, 80, 120, 110, 120, 95, 72, 83, 90, 95)
```

Wir wollen ein 95%-Konfidenzintervall erstellen für den **Unterschied** zwischen den Dauern für *betont* und *unbetont*:

```
mu = mean(bet - un)
SE = sd(bet - un)/sqrt(12)
# 95% Konfidenzintervall
mu + SE * qt(0.025, 11)

## [1] 8.739938

mu + SE * qt(0.975, 11)

## [1] 26.26006
```

Das 95% Konfidenzintervall für den Unterschied zwischen den Dauerwerten ist 8.7 ms ≤ mu ≤ 26.3 ms. Bedeutung: der Unterschied zwischen dem Dauer-Mittelwert von einem betontem /a/ und dem Dauer-Mittelwert von einem unbetonten /a/ liegt zwischen 8.7 ms und 26.3 ms mit einer Wahrscheinlichkeit von 0.95.

4. Prüfen einer Hypothese, oder: unterscheiden sich Mittelwerte *signifikant*?

*H*0 (die Null-Hypothese):

- "Betonung hat keinen Einfluss auf die Dauer."
  - Bedeutung: der Unterschied zwischen den Mittelwerten ist Null (0).

*H*1 (die Alternativ-Hypothese):

- "Betonung beeinflusst die Dauer.""
  - Bedeutung: der Unterschied zwischen den Mittelwerten weicht von Null ab.

α-Wert (alpha-Wert):

- 0.05 ist der hier gewählte, sogenannte α-Wert (alpha-Wert), bei dem wir $H0$ *verwerfen*
  - Ein Konfidenzintervall von 0.95 bedeutet einen α-Wert von 1 - 0.95 = 0.05.

Prüfen: wenn 0 außerhalb des Konfidenzintervalls 8.7 ms ≤ mu ≤ 26.3 ms fällt, verwerfen wir $H0$ und akzepierten $H1$.

Schlussfolgerung für das obige Beispiel: Wir verwerfen $H0$ und akzeptieren $H1$ ("Betonung beeinflusst die Dauer.").

Berichten:

"Die Dauer wird signifikant von der Betonung beeinflusst (p < 0.05)."

Bedeutung (i):

- die Wahrscheinlichkeit, dass die Dauer *nicht* von der Betonung beeinflusst wird, ist weniger als 0.05 (weniger als 5%).

Bedeutung (ii):

- die Wahrscheinlichkeit, dass der Unterschied zwischen dem Dauer-Mittelwert vom betonten /a/ und dem Dauer-Mittelwert vom unbetonten /a/ 0 (Null) sein könnte, ist kleiner als 0.05.

5. Der t-Test

Wir bekommen dieselbe Auswertung einfacher durch den sogenannten **t-Test**.

Manuell noch einmal:

```
mu = mean(bet - un)
SE = sd(bet - un)/sqrt(12)
# 95% Konfidenzintervall
mu + SE * qt(0.025, 11)
```

```
## [1] 8.739938
```

```
mu + SE * qt(0.975, 11)
```

```
## [1] 26.26006
```

Mit dem t-Test:

```
t.test(bet - un)
```

```
##
##  One Sample t-test
##
## data:  bet - un
## t = 4.3969, df = 11, p-value = 0.001069
## alternative hypothesis: true mean is not equal to 0
```

```
## 95 percent confidence interval:
##   8.739938 26.260062
## sample estimates:
## mean of x
##      17.5
```

Identifizieren:

```
mu
```

```
## [1] 17.5
```

- μ (`mean of x: 17.5`)
- Freiheitgrade (`df = 11`)
- 95%-Konfidenzintervall (`8.739938` — `26.260062`)
- `t = 4.3969` (die t-Statistik).
  - Der Abstand zwischen μ und 0 (Null) in `SE`-Einheiten.

```
(mu - 0)/SE
```

```
## [1] 4.396914
```

- `p-value = 0.001069`
  - Bedeutung (i): Die Wahrscheinlichkeit, dass der Unterschied zwischen den Mittelwerten Null sein könnte.
  - Bedeutung (ii): Die Wahrscheinlichkeit, dass $H0$ zutrifft.
  - Bedeutung (iii): die Wahrscheinlichkeit, dass der Unterschied zwischen den Mittelwerten außerhalb des Konfidenzintervalls fällt, auch:

```
(1- pt(4.396914, 11)) * 2
```

```
## [1] 0.001068659
```

Ergebnis berichten:

Wir wählen immer drei α-Werte aus: $p < 0.05$, $p < 0.01$, $p < 0.001$ und wählen den α-Wert, der am nächsten über dem p-Wert liegt

Hier wählen wir $p < 0.01$, da der p-Wert (0.001069) über 0.001 aber unter 0.01 liegt. Wir berichten entweder:

- "Betonung hatte einen signifikanten Einfluss auf die Dauer ($t[11] = 4.4, p < 0.01$)"

Oder:

- "Die Betonung wurde signifikant von der Dauer beeinflusst ($t[11] = 4.4, p < 0.01$)"

Sollte $p > 0.05$ sein, dann ist das Ergebnis nicht signifikant ("n.s.") (Wir verwerfen nicht H0), und schreiben:

- "Die Betonung hatte keinen signifikanten Einfluss auf die Dauer"

oder

- "Die Dauer wurde nicht signifikant von der Dauer beinflusst."

6. Gepaarter t-Test

Das obige Beispiel ist ein sogenannter *gepaarter t-Test*, weil eine Differenz — ob betont oder unbetont — pro Paar berechnet wird (daher 12 Paare in dem obigen Beispiel).

Ein gepaarter t-test kommt in der Phonetik meistens dann vor, wenn Stichproben-Paare pro Versuchsperson verglichen werden.

Ein zweites Beispiel, bei dem (wie üblich) die Werte in einem Data-Frame stecken:

- *12 Versuchspersonen produzierten jeweils ein betontes und ein unbetontes /i/. Unterscheiden sich das betonte und das unbetonte /i/ voneinander in F2?*

```
dim(form)
```

```
## [1] 24  3
```

```
head(form)
```

```
##     F2 Bet Vpn
## 1 2577   b  S1
## 2 2122   b  S2
## 3 2192   b  S3
## 4 2581   b  S4
## 5 2227   b  S5
## 6 2481   b  S6
```

```
summary(form)
```

```
##        F2       Bet         Vpn
##  Min.   :1728   b:12   S1     : 2
##  1st Qu.:2047   u:12   S10    : 2
##  Median :2150          S11    : 2
##  Mean   :2176          S12    : 2
##  3rd Qu.:2333          S2     : 2
##  Max.   :2581          S3     : 2
##                        (Other):12
```

Die Frage bitte immer umstellen ("Wird y von x beeinflusst?"):

- *"Wurde F2 (abhängige Variable) von der Betonung (unabhängiger Faktor mit 2 Stufen: betont/unbetont) beeinflusst?"*

Der Test ist gepaart: es gibt jeweils ein Wertepaar pro Versuchsperson.

6.1. Boxplot

6.1.1. Unterschiede pro Paar (hier Versuchsperson) berechnen

Entweder mit `aggregate()`; dort in der Formel den unterscheidenden Faktor weglassen, und als Funktion `diff()` angeben:

```
unterschied  = aggregate(F2 ~ Vpn, FUN = diff, data = form)
unterschied
```

```
##    Vpn   F2
## 1   S1 -495
## 2  S10 -497
## 3  S11   32
## 4  S12 -308
## 5   S2  -99
## 6   S3   15
## 7   S4 -480
## 8   S5  -96
## 9   S6 -508
## 10  S7 -146
## 11  S8 -495
## 12  S9 -175
```

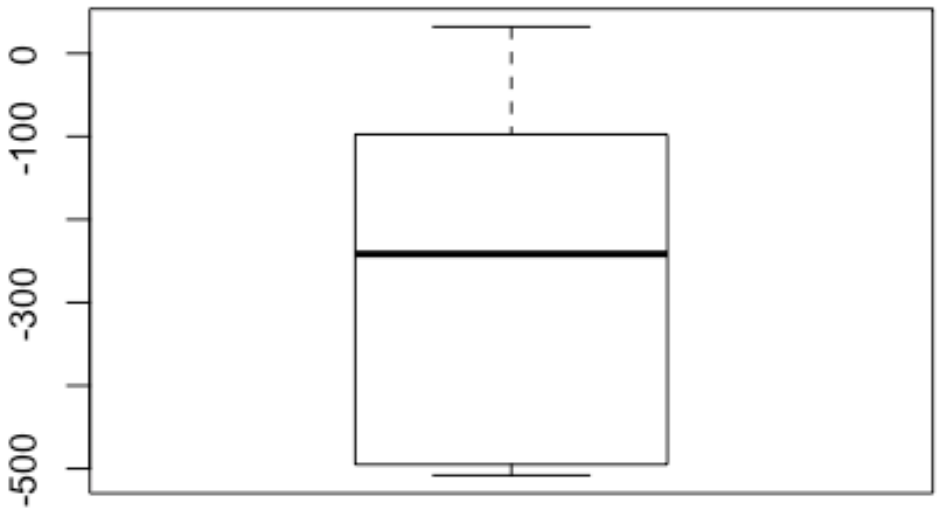
Oder mit dem Package `dplyr`:

```
library(dplyr)

unterschied = form%>%
  group_by(Vpn)%>%
  summarise(F2=diff(F2))

unterschied
```

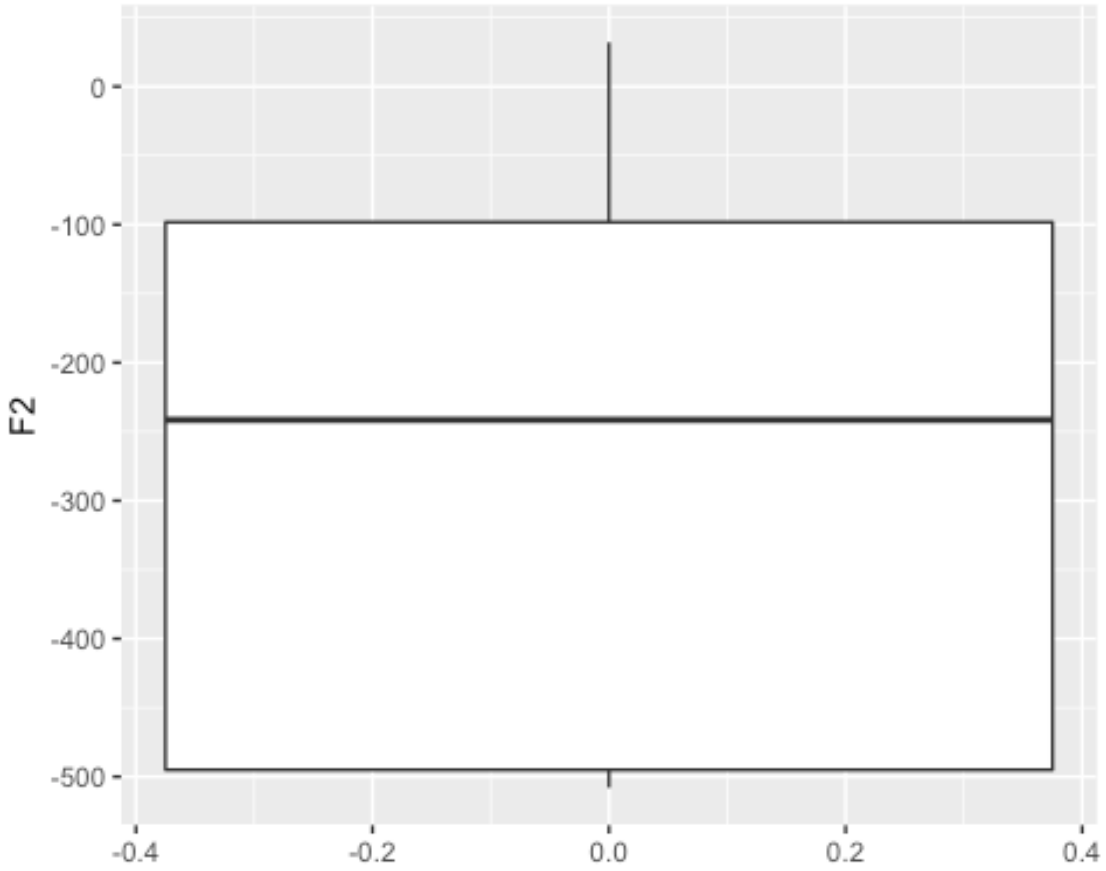
```
## # A tibble: 12 x 2
##    Vpn      F2
##    <fct> <int>
##  1 S1     -495
##  2 S10    -497
##  3 S11      32
##  4 S12    -308
##  5 S2      -99
##  6 S3       15
##  7 S4     -480
##  8 S5      -96
##  9 S6     -508
## 10 S7     -146
## 11 S8     -495
## 12 S9     -175
```

6.1.2. Boxplot der Unterschiede

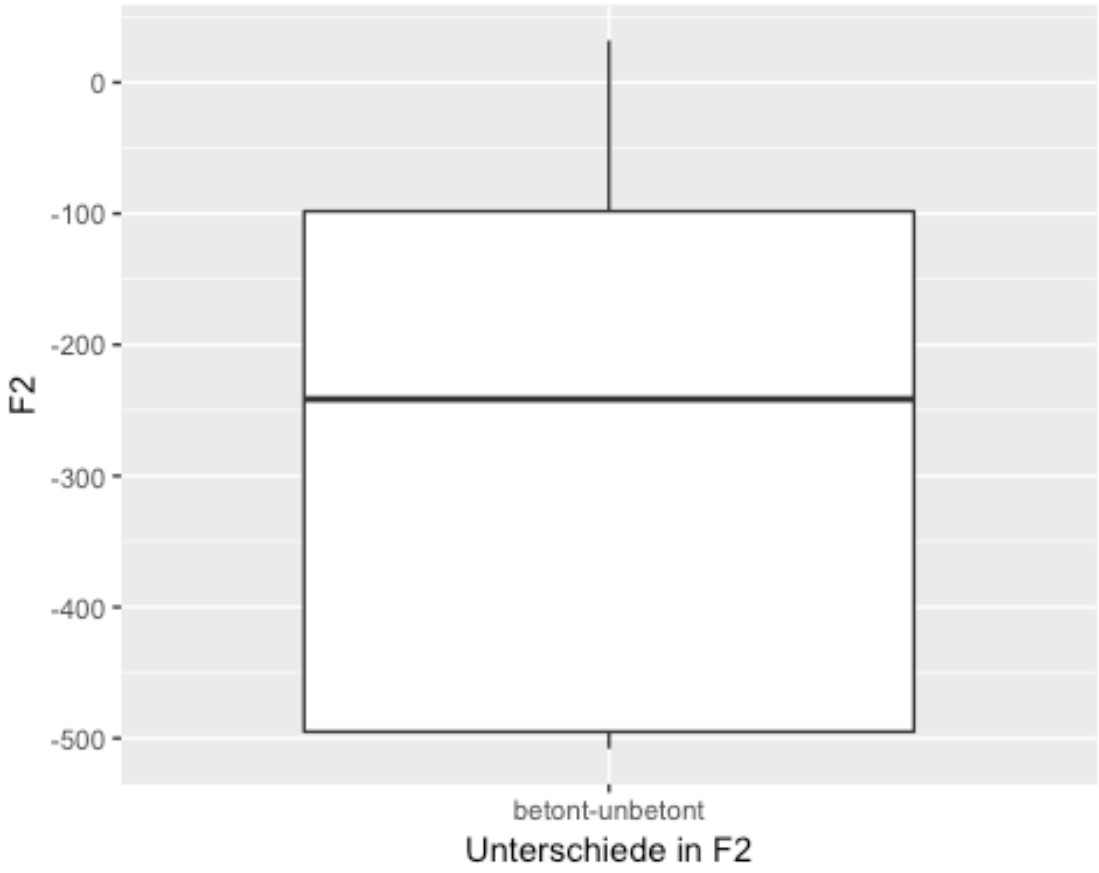
```
# am einfachsten ohne ggplot2:
boxplot(unterschied$F2)
```




```
# mit ggplot2 (x einfach weglassen):
ggplot(unterschied) +
  aes(y = F2)  +
  geom_boxplot()
```




```
#oder:
ggplot(unterschied) +
  aes(x = "betont-unbetont", y = F2)  +
  geom_boxplot() +
  xlab("Unterschiede in F2")
```



6.2. `t.test()`

Der t-Test prüft, ob der Mittelwert der Unterschiede signifikant von 0 abweicht oder nicht:

```
t.test(unterschied$F2)

##
##  One Sample t-test
##
## data:  unterschied$F2
## t = -4.3543, df = 11, p-value = 0.001147
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -407.9837 -134.0163
## sample estimates:
## mean of x
##      -271
```

Antwort:

- "F2 wurde signifikant von der Betonung beeinflusst ($t[11] = 4.4, p < 0.01$)."

Alternative Berechnung

- **y ~ x** unter der Berüchsichtigung, dass x (hier `Bet`) im ursprünglichen data.frame `form` gepaarte Werte für y (hier `F2`) kodiert hat:

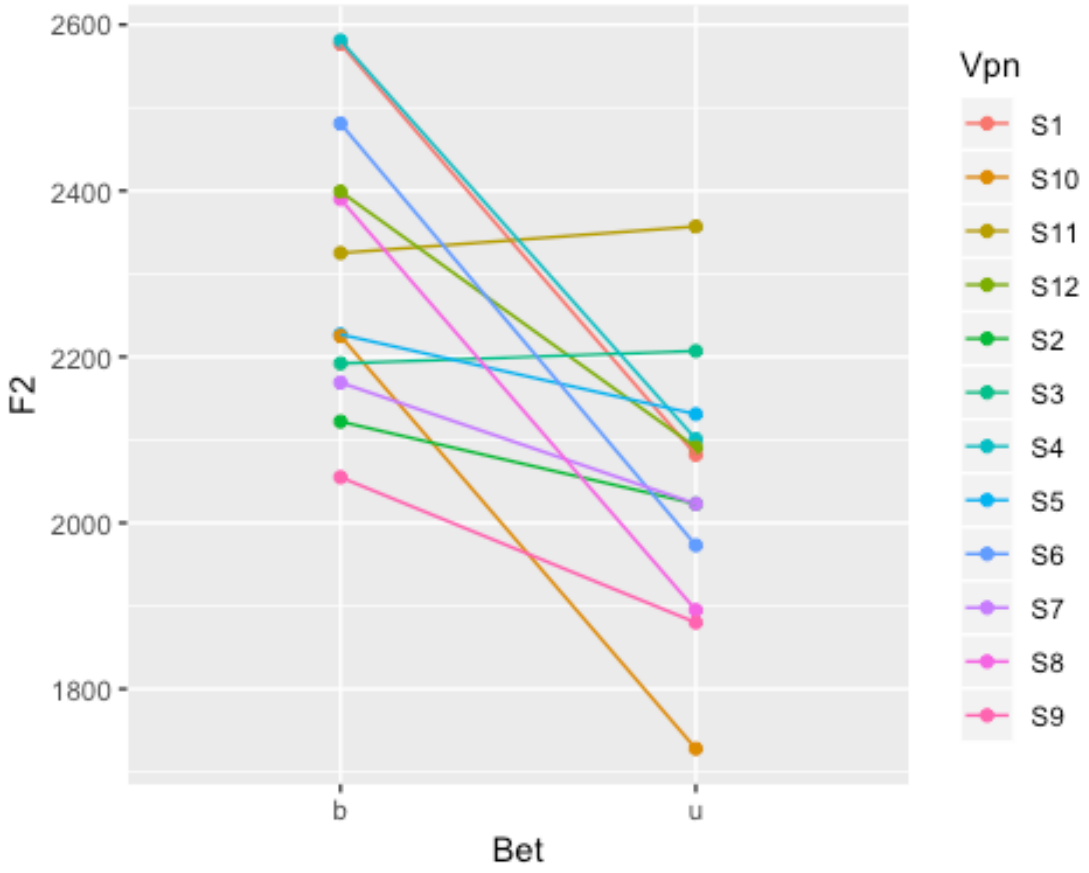
```r
t.test(F2 ~ Bet, paired = T, data = form)
```

```
##
##  Paired t-test
##
## data:  F2 by Bet
## t = 4.3543, df = 11, p-value = 0.001147
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  134.0163 407.9837
## sample estimates:
## mean of the differences
##                     271
```

Alternative Abbildung

Manche Leute bevorzugen auch, beide Datensätze als Punkte abzubilden, und die Paarungen als Linienverbindungen darzustellen:

```r
ggplot(form) +
  aes(x = Bet, y = F2,col = Vpn, group = Vpn) +
  geom_point() +
  geom_line()
```



In unserem Kurs wollen wir aber lieber bei der Abbildung mit einer Box, die die Paar-Differenz-Werte statistisch-deskriptiv beschreibt, bleiben!

7. Ungepaarter t-test

Ein *ungepaarter t-Test* liegt vor, wenn nicht Paare von Stichproben, sondern zwei Gruppen miteinander verglichen werden.

In der Phonetik werden z.B. oft zwei verschiedene Sprecher-Gruppen (männlich/weiblich; Bayern/Hessen; englisch/deutsch) verglichen.

- *Unterscheiden sich deutsch und englisch in F2 von /e/?*
  - = Wird F2 (abhängige Variable) von der Sprache (unabhängige Variable mit 2 Stufen: englisch/deutsch) beeinflusst?

```
head(e.df)
```

```
##         F2 Sprache Vpn
## 1 1904.296       E  S1
## 2 1938.509       E  S2
## 3 1519.615       E  S3
## 4 1904.047       E  S4
## 5 1891.154       E  S5
## 6 1938.889       E  S6
```

```
dim(e.df)
```

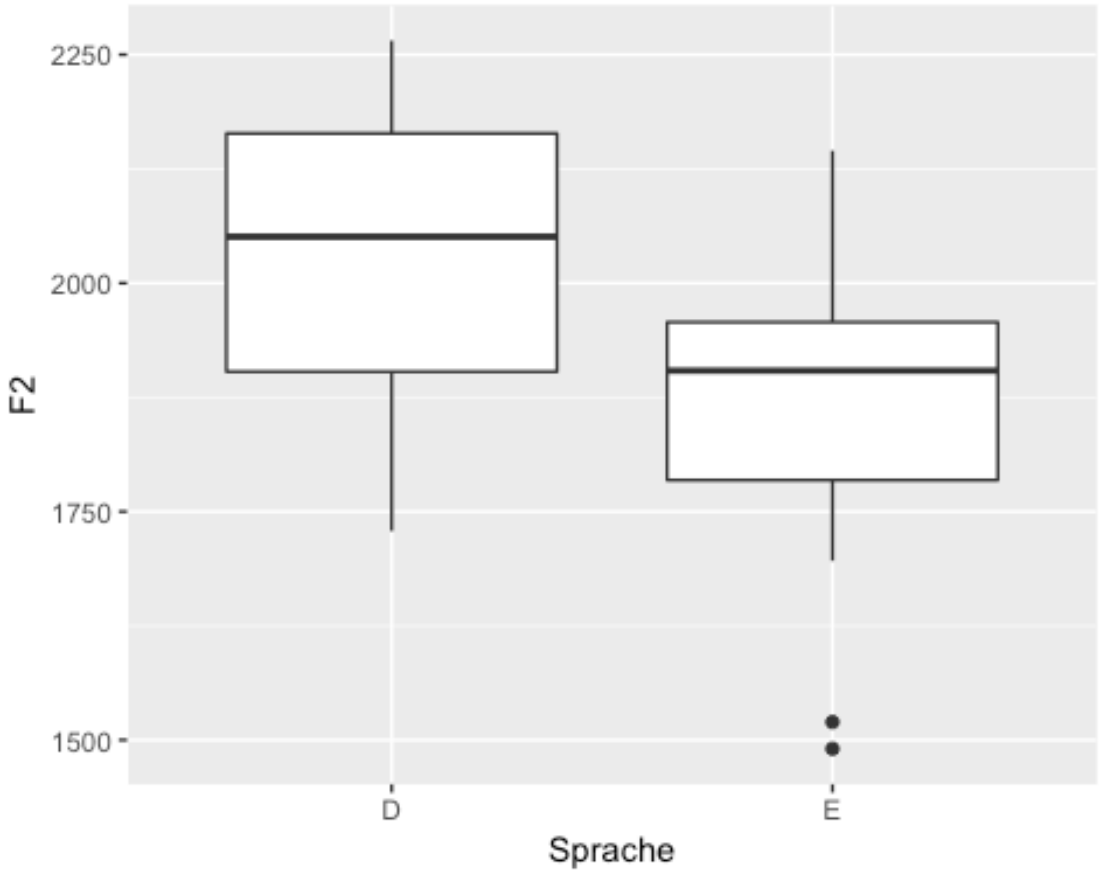
```
## [1] 27  3
```

Vorgang: wie oben, aber die Stichproben werden nicht paarweise voneinander subtrahiert (da sie nicht gepaart sind). Daher ist eine genaue Übereinstimmung der Gruppengrößen auch nicht erforderlich (dennoch sollten die Gruppengrößen auch nicht zu sehr voneinander abweichen), hier z.B.

```
table(e.df$Sprache)
```

```
##
##  D  E
## 15 12
```

```
# 1. Boxplot oder Densitplot der Unterschiede
ggplot(e.df) +
  aes(y = F2, x = Sprache) +
  geom_boxplot()
```




```r
# density plot
ggplot(e.df) +
  aes(x = F2, col = Sprache) +
  geom_density()
```

Hier prüfen wir, ob signifikante Unterschiede zwischen den Mittelwerten der beiden Gruppen vorliegen (NB: nicht gepaart!):

```
t.test(F2 ~ Sprache, data = e.df)

##
##  Welch Two Sample t-test
##
## data:  F2 by Sprache
## t = 2.2613, df = 21.101, p-value = 0.03443
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   13.46719 320.73097
## sample estimates:
## mean in group D mean in group E
##        2031.672        1864.573
```

Antwort:

- "F2 wurde signifikant ($t[21.1] = 2.3, p < 0.05$) von der Sprache beeinflusst."

(In einem ungepaarten Welch Two Sample t-Test bekommt man immer Gleitkommazahlen als Freiheitsgrade, da die Freiheitsgrade mittels einer Formel (der sogenannten Welch-Satterthwaite-Formel) verändert ("approximiert") werden. Durch diese Änderung müssen wir uns weniger Gedanken darüber machen, wann ein solcher Test erlaubt ist oder nicht.)

8. Voraussetzung für den t-Test prüfen: `shapiro.test()` zur Überprüfung von Verteilungen

Der `shapiro.test()` überprüft, ob die Verteilungen der Daten signifikant von einer Normalverteilung abweichen. Sollte dies der Fall sein, dürfen wir den t-Test nicht anwenden, sondern müssen statt dessen den Wilcoxon-Vorzeichen-Rang-Test anwenden (`wilcox.test()`). Ergibt der `shapiro.test()` keine signifikante Abweichung der Verteilungen unserer Stichproben von einer Normalverteilung, bleiben wir beim t-Test.

Beispiel für den gepaarten Fall:

```
shapiro.test(unterschied$F2)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  unterschied$F2
## W = 0.84496, p-value = 0.03183
```

```
# da hier p < 0.05, ist der oben angewandte t-Test eigentlich nicht zulässig
...
t.test(unterschied$F2)
```

```
##
##  One Sample t-test
##
## data:  unterschied$F2
## t = -4.3543, df = 11, p-value = 0.001147
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -407.9837 -134.0163
## sample estimates:
## mean of x
##      -271
```

```
# ... und muss daher mit dem Wilcoxon signed rank test ersetzt werden:
wilcox.test(unterschied$F2)
```

```
## Warning in wilcox.test.default(unterschied$F2): cannot compute exact p-
## value with ties
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  unterschied$F2
## V = 3, p-value = 0.005338
## alternative hypothesis: true location is not equal to 0
```

Wir müssen also unsere Antwort ändern zu: “F2 wurde signifikant von der Betonung beeinflusst, wie ein Wilcoxon-Test zeigte ($V = 3, p < 0.01$).”

Im Beispiel für den ungepaarten Fall war der t-Test allerdings durchaus zulässig:

```r
with(e.df, tapply(F2, Sprache, shapiro.test))
```

```
## $D
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.94892, p-value = 0.5076
##
##
## $E
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.91579, p-value = 0.2529
```

```r
#da beide nicht signifikant -->
t.test(F2 ~ Sprache, data = e.df)
```

```
##
##  Welch Two Sample t-test
##
## data:  F2 by Sprache
## t = 2.2613, df = 21.101, p-value = 0.03443
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   13.46719 320.73097
## sample estimates:
## mean in group D mean in group E
##        2031.672        1864.573
```

```r
#wäre zumindest ein Datensatz signifikant von einer Normalverteilung
abweichend, wäre die Syntax für die Alternative gewesen:
wilcox.test(F2 ~ Sprache, data = e.df)
```

```
##
##  Wilcoxon rank sum test
##
## data:  F2 by Sprache
## W = 131, p-value = 0.04687
## alternative hypothesis: true location shift is not equal to 0
```

Da der Shapiro-Test hier für keine der Datensätze eine Abweichung von der Normalverteil gezeigt hatte, können wir beim t-Test bleiben und schreiben auch weiterhin:

"F2 wurde signifikant ($t[21.1] = 2.3, p < 0.05$) von der Sprache beeinflusst."