

# Übung Wahrscheinlichkeitsrechnung und Statistik in den Ingenieurwissenschaften

## Übung 12

Datensatz BLECH.DAT (Dicken von 15 zufällig ausgewählten Blechen in  $10\mu\text{m}$ :)

346, 363, 360, 318, 346, 268, 299, 287, 310, 349, 333, 365, 281, 265, 344.

Betrachten das Testproblem  $H_0 : \mu = 310$  (Hypothese) gegen  $H_1 : \mu \neq 310$  (Alternative) für diesen Datensatz.  
Benutze den sogenannten **t-Test**.

# Testproblem $H_0 : \mu = 310$ gegen $H_1 : \mu \neq 310$ .

Wähle als Teststatistik  $T(x) = \sqrt{N} \frac{|\bar{x} - \mu_0|}{s(x)}$ .

```
> BLECH<-c(346,363,360,318,346,268,299,287,310,349,333,365,  
+ 281,265,344)  
> mean(BLECH)  
[1] 322.2667  
> sd(BLECH)  
[1] 35.07027  
> sqrt(15)*abs(mean(BLECH)-310)/sd(BLECH)  
[1] 1.354669
```

Testproblem  $H_0 : \mu = 310$  gegen  $H_1 : \mu \neq 310$ .

- P-Wert:  $P_{H_0}(T(X) \geq T(x))$ , lehne  $H_0$  ab, wenn P-Wert kleiner als  $\alpha (= 0.05)$

Testproblem  $H_0 : \mu = 310$  gegen  $H_1 : \mu \neq 310$ .

- P-Wert:  $P_{H_0}(T(X) \geq T(x))$ , lehne  $H_0$  ab, wenn P-Wert kleiner als  $\alpha (= 0.05)$
- unter  $H_0 : \mu = 310$  gilt  $\sqrt{N} \frac{|\bar{X} - 310|}{s(x)} \sim t_{N-1}$

Testproblem  $H_0 : \mu = 310$  gegen  $H_1 : \mu \neq 310$ .

- P-Wert:  $P_{H_0}(T(X) \geq T(x))$ , lehne  $H_0$  ab, wenn P-Wert kleiner als  $\alpha (= 0.05)$
- unter  $H_0 : \mu = 310$  gilt  $\sqrt{N} \frac{|\bar{X} - 310|}{s(x)} \sim t_{N-1}$
- P-Wert hier  $2F_{t_{N-1}}(-T(x)) = 2F_{t_{14}}(-1.354669)$

# Testproblem $H_0 : \mu = 310$ gegen $H_1 : \mu \neq 310$ .

- P-Wert:  $P_{H_0}(T(X) \geq T(x))$ , lehne  $H_0$  ab, wenn P-Wert kleiner als  $\alpha (= 0.05)$
- unter  $H_0 : \mu = 310$  gilt  $\sqrt{N} \frac{|\bar{X} - 310|}{s(x)} \sim t_{N-1}$
- P-Wert hier  $2F_{t_{N-1}}(-T(x)) = 2F_{t_{14}}(-1.354669)$
- `2*pt(-T,df=14)` in R liefert 0.3342819

## Testproblem $H_0 : \mu = 310$ gegen $H_1 : \mu \neq 310$ .

- P-Wert:  $P_{H_0}(T(X) \geq T(x))$ , lehne  $H_0$  ab, wenn P-Wert kleiner als  $\alpha (= 0.05)$
- unter  $H_0 : \mu = 310$  gilt  $\sqrt{N} \frac{|\bar{X} - 310|}{s(x)} \sim t_{N-1}$
- P-Wert hier  $2F_{t_{N-1}}(-T(x)) = 2F_{t_{14}}(-1.354669)$
- `2*pt(-T,df=14)` in R liefert 0.3342819
- dasselbe Ergebnis liefert auch  
`t.test(BLECH,alternative="two.sided",mu=310)`



# Testproblem $H_0 : \mu = 310$ gegen $H_1 : \mu \neq 310$ .

Kennt man die Verteilung der Teststatistik nicht, kann der P-Wert auch durch Simulationen (näherungsweise) bestimmt werden:

Simuliere P-Wert:

- erzeuge  $M = 1000$  Stichproben  $x^1, \dots, x^M$  mit je  $N = 15$  Daten  $x^m = (x_1^m, \dots, x_{15}^m)$ ,  $m = 1, \dots, M$  mit Normalverteilung,  $\mu = 310$  und  $\sigma^2$  (sodass  $H_0$  gilt)

# Testproblem $H_0 : \mu = 310$ gegen $H_1 : \mu \neq 310$ .

Kennt man die Verteilung der Teststatistik nicht, kann der P-Wert auch durch Simulationen (näherungsweise) bestimmt werden:

Simuliere P-Wert:

- erzeuge  $M = 1000$  Stichproben  $x^1, \dots, x^M$  mit je  $N = 15$  Daten  $x^m = (x_1^m, \dots, x_{15}^m)$ ,  $m = 1, \dots, M$  mit Normalverteilung,  $\mu = 310$  und  $\sigma^2$  (sodass  $H_0$  gilt)
- berechne für alle  $x^m$  die Teststatistik

# Testproblem $H_0 : \mu = 310$ gegen $H_1 : \mu \neq 310$ .

Kennt man die Verteilung der Teststatistik nicht, kann der P-Wert auch durch Simulationen (näherungsweise) bestimmt werden:

Simuliere P-Wert:

- erzeuge  $M = 1000$  Stichproben  $x^1, \dots, x^M$  mit je  $N = 15$  Daten  $x^m = (x_1^m, \dots, x_{15}^m)$ ,  $m = 1, \dots, M$  mit Normalverteilung,  $\mu = 310$  und  $\sigma^2$  (sodass  $H_0$  gilt)
- berechne für alle  $x^m$  die Teststatistik
- bestimme den Anteil der Teststatistiken, die größer als die beobachtete: Dieser Anteil entspricht dem P-Wert

Testproblem  $H_0 : \mu = 310$  gegen  $H_1 : \mu \neq 310$ .

$$\sigma = 1$$

```
> ts<-numeric(10000)
> for (i in (1:10000))
+ {
+   testdaten<-rnorm(15,310,1)
+   ts[i]<-sqrt(15)*abs(mean(testdaten-310))/sd(testdaten)
+ }
> 1/10000*sum(ts>=T)
[1] 0.2001
```

Testproblem  $H_0 : \mu = 310$  gegen  $H_1 : \mu \neq 310$ .

$$\sigma = 30$$

```
> ts<-numeric(10000)
> for (i in (1:10000))
+ {
+   testdaten<-rnorm(15,310,30)
+   ts[i]<-sqrt(15)*abs(mean(testdaten-310))/sd(testdaten)
+ }
> 1/10000*sum(ts>=T)
[1] 0.1947
```

## Aufgabe 1 (c):

Testproblem  $H_0 : \mu = 310$  gegen  $H_1 : \mu \neq 310$ .

Entscheidungsregel, die nicht den P-Wert benutzt: Lehne  $H_0$  ab, falls  $T(x) > t_{N-1; 1-\frac{\alpha}{2}}$  gilt.

```
> qt(0.975,df=14)
```

```
[1] 2.144787
```

## Aufgabe zur Entspannung:

Datensatz:  $x_1 = 1, x_2 = 4, x_3 = 2, x_4 = 3, x_5 = 1, x_6 = 5$

- Ordne den Datensatz
- Bestimme Ränge der Daten
- Median=?
- Quartilsabstand=?
- arithmetisches Mittel=?

## Aufgabe 1 (d)

Fehler 2. Art: Entscheidung für  $H_0$ , obwohl  $H_1$  gilt.



## Aufgabe 1 (d)

Fehler 2. Art: Entscheidung für  $H_0$ , obwohl  $H_1$  gilt.

Für den t-Test: Wahrscheinlichkeit für Fehler 2. Art, wenn der wahre Mittelwert  $\mu$  ist:

$$F_{t_{N-1}\left(\frac{\sqrt{N}(\mu-\mu_0)}{\sigma}\right)}\left(t_{N-1;1-\frac{\alpha}{2}}\right) - F_{t_{N-1}\left(\frac{\sqrt{N}(\mu-\mu_0)}{\sigma}\right)}\left(-t_{N-1;1-\frac{\alpha}{2}}\right)$$

also hier:

$$P_{H1}(T(X) \leq t_{14;1-\frac{\alpha}{2}}) = F_{t_{14}\left(\frac{\sqrt{15}(300-310)}{\sigma}\right)}\left(t_{14;1-\frac{\alpha}{2}}\right) - F_{t_{14}\left(\frac{\sqrt{15}(300-310)}{\sigma}\right)}\left(-t_{14;1-\frac{\alpha}{2}}\right).$$

## Aufgabe 1 (d) und (e)

$$\sigma = 1$$

```
> ncp<-sqrt(15)*(300-310)/1  
> pt(qt(0.975,df=14),df=14,ncp=ncp)  
-pt(-qt(0.975,df=14),df=14,ncp=ncp)  
[1] 0
```

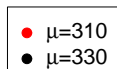
## Aufgabe 1 (d) und (e)

$$\sigma = 1$$

```
> ncp<-sqrt(15)*(300-310)/1  
> pt(qt(0.975,df=14),df=14,ncp=ncp)  
-pt(-qt(0.975,df=14),df=14,ncp=ncp)  
[1] 0
```

$$\sigma = 30$$

```
> ncp<-sqrt(15)*(300-310)/30  
> pt(qt(0.975,df=14),df=14,ncp=ncp)  
-pt(-qt(0.975,df=14),df=14,ncp=ncp)  
[1] 0.7745203
```



280 300 320 340 360



280 300 320 340 360

Links: Varianz  $\sigma = 30$ , Rechts: Varianz  $\sigma = 1$

Testproblem  $H_0 : \mu = 310$  gegen  $H_1 : \mu \neq 310$ .

Aufgabe 1 (f)

Simulation der Fehlerwahrscheinlichkeit 2. Art, wenn  $\mu = 300$  gilt.

- erzeuge  $M = 1000$  Stichproben  $x^1, \dots, x^M$  mit je  $N = 15$  Daten  $x^m = (x_1^m, \dots, x_{15}^m)$ ,  $m = 1, \dots, M$  mit Normalverteilung,  $\mu = 300$  und  $\sigma^2$

# Testproblem $H_0 : \mu = 310$ gegen $H_1 : \mu \neq 310$ .

## Aufgabe 1 (f)

Simulation der Fehlerwahrscheinlichkeit 2. Art, wenn  $\mu = 300$  gilt.

- erzeuge  $M = 1000$  Stichproben  $x^1, \dots, x^M$  mit je  $N = 15$  Daten  $x^m = (x_1^m, \dots, x_{15}^m)$ ,  $m = 1, \dots, M$  mit Normalverteilung,  $\mu = 300$  und  $\sigma^2$
- Berechne für jede Stichprobe die zugehörige Teststatistik

Testproblem  $H_0 : \mu = 310$  gegen  $H_1 : \mu \neq 310$ .

## Aufgabe 1 (f)

Simulation der Fehlerwahrscheinlichkeit 2. Art, wenn  $\mu = 300$  gilt.

- erzeuge  $M = 1000$  Stichproben  $x^1, \dots, x^M$  mit je  $N = 15$  Daten  $x^m = (x_1^m, \dots, x_{15}^m)$ ,  $m = 1, \dots, M$  mit Normalverteilung,  $\mu = 300$  und  $\sigma^2$
- Berechne für jede Stichprobe die zugehörige Teststatistik
- zähle Anteil, wie oft Teststatistik unter kritischem Wert, also wie oft  $H_0$  nicht abgelehnt wird = Fehlerwahrscheinlichkeit 2. Art für  $\mu = 300$

Testproblem  $H_0 : \mu = 310$  gegen  $H_1 : \mu \neq 310$ .

```
> ts<-numeric(10000)
> for (i in (1:10000))
+ {
+   testdaten<-rnorm(15,300,1)
+   ts[i]<-sqrt(15)*abs(mean(testdaten-310))/sd(testdaten)
+ }
> sum(ts<=qt(0.975,df=14))
[1] 0
```



Testproblem  $H_0 : \mu = 310$  gegen  $H_1 : \mu \neq 310$ .

```
> ts<-numeric(1000)
> for (i in (1:1000))
+ {
+   testdaten<-rnorm(15,300,30)
+   ts[i]<-sqrt(15)*abs(mean(testdaten-310))/sd(testdaten)
+ }
> sum(ts<=qt(0.975,df=14))/1000
[1] 0.78
```

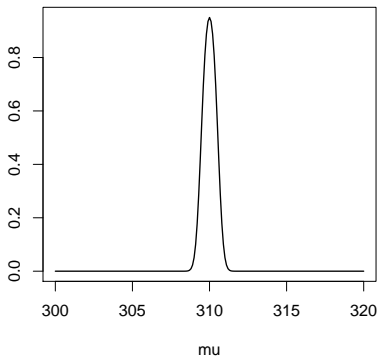
## Aufgabe 1 (h)

Die Gütefunktion ist 1 - "Fehlerwahrscheinlichkeit 2.Art", sie gibt die Wahrscheinlichkeit für die Entscheidung für  $H_1$  (also Ablehnung von  $H_0$ ) an.

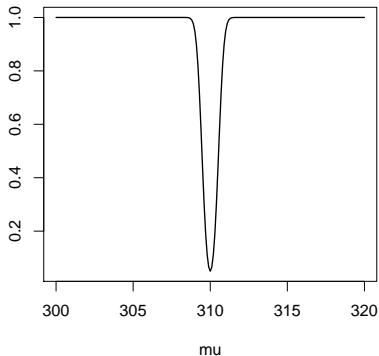
```
mu<-seq(300,320,0.1)
ncp<-sqrt(15)*(mu-310)
plot(mu,pt(qt(0.975,df=14),df=14,ncp=ncp)
      -pt(-qt(0.975,df=14),df=14,ncp=ncp),
      type="l",ylab="",
      main="Wahrscheinlichkeit Fehler 2. Art")
plot(mu,1-pt(qt(0.975,df=14),df=14,ncp=ncp)
      +pt(-qt(0.975,df=14),df=14,ncp=ncp)
      ,type="l",ylab="",main="Gütefunktion")
```

# Aufgabe 1 (h)

**Wahrscheinlichkeit Fehler 2. Art**



**Gütefunktion**



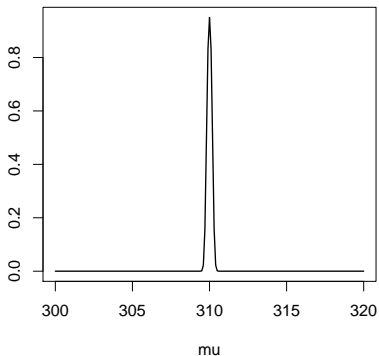
# Aufgabe 1 (i)

$$N = 100$$

```
mu<-seq(300,320,0.1)
ncp<-sqrt(100)*(mu-310)
plot(mu,pt(qt(0.975,df=99),df=99,ncp=ncp)
      -pt(-qt(0.975,df=99),df=99,ncp=ncp),
      type="l",ylab="",
      main="Wahrscheinlichkeit Fehler 2. Art")
plot(mu,1-pt(qt(0.975,df=99),df=99,ncp=ncp)
      +pt(-qt(0.975,df=99),df=99,ncp=ncp),
      type="l",ylab="",main="Gütefunktion")
```

# Aufgabe 1 (i)

Wahrscheinlichkeit Fehler 2. Art



Gütefunktion

