# Lösungen Testat STOC SW12

Daniel Winz

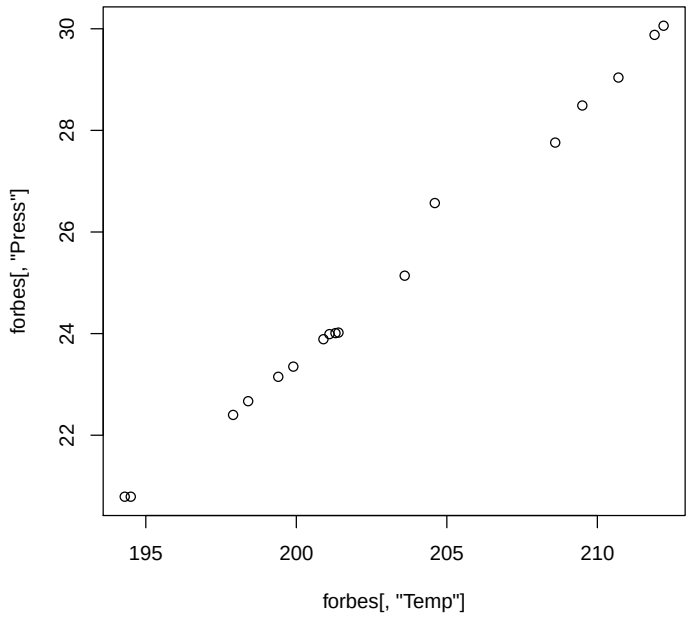
30. Mai 2013 11:46

## Inhaltsverzeichnis

# 1 Aufgabe 1

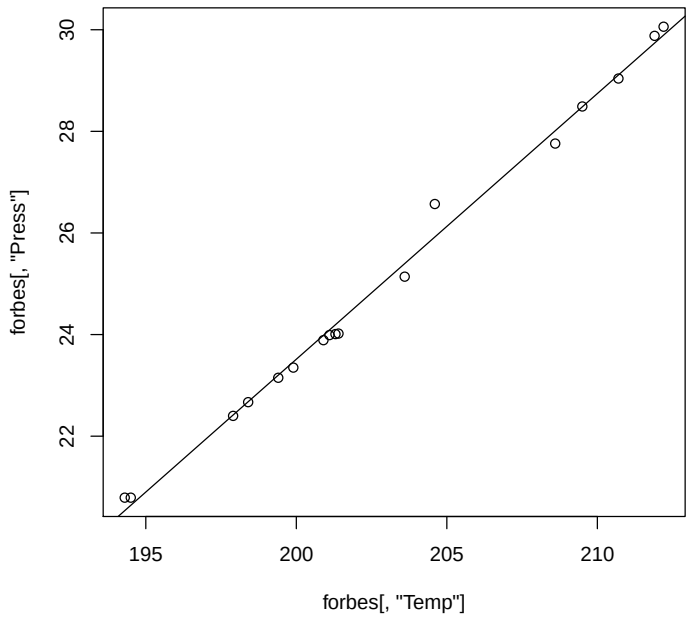
## 1.1 a

```
> # forbes <- read.table("http://stat.ethz.ch/Teaching/Datasets/forbes.dat",header=TRUE)
> forbes <- read.table("forbes.dat",header=TRUE)
> plot(forbes[,"Temp"], forbes[,"Press"])
```



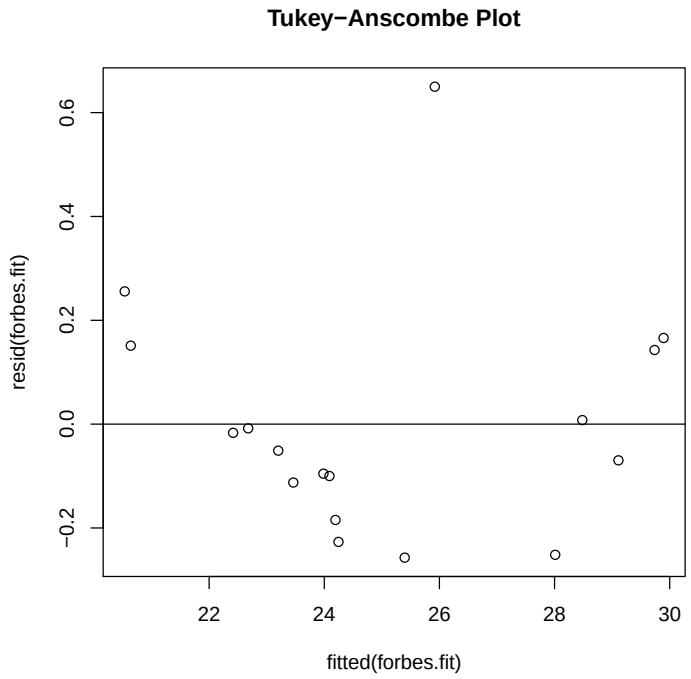

## 1.2 b

```
> forbes.fit <- lm(Press ~ Temp, data = forbes)
> # summary(forbes.fit)
> plot(forbes[,"Temp"], forbes[,"Press"])
> abline(forbes.fit)
```

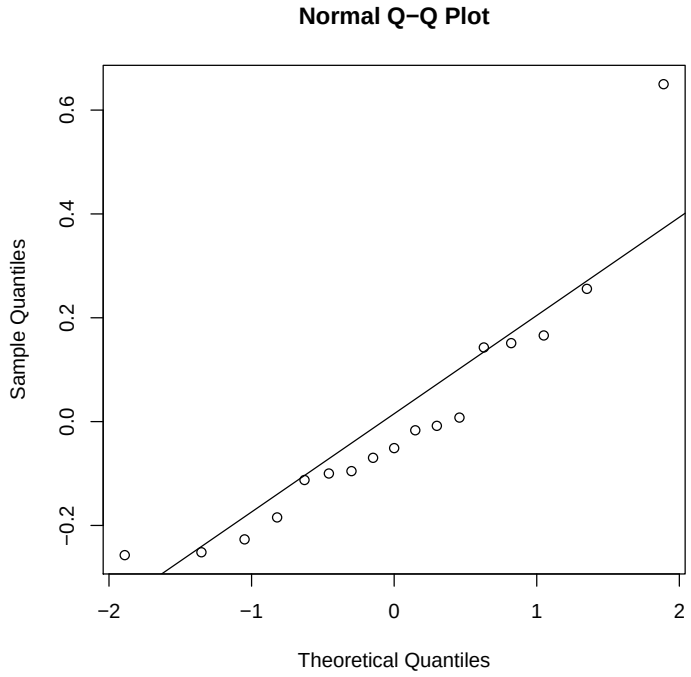



### 1.3 c

```
> plot(fitted(forbes.fit), resid(forbes.fit), main="Tukey-Anscombe Plot")
> abline(h=0)
```

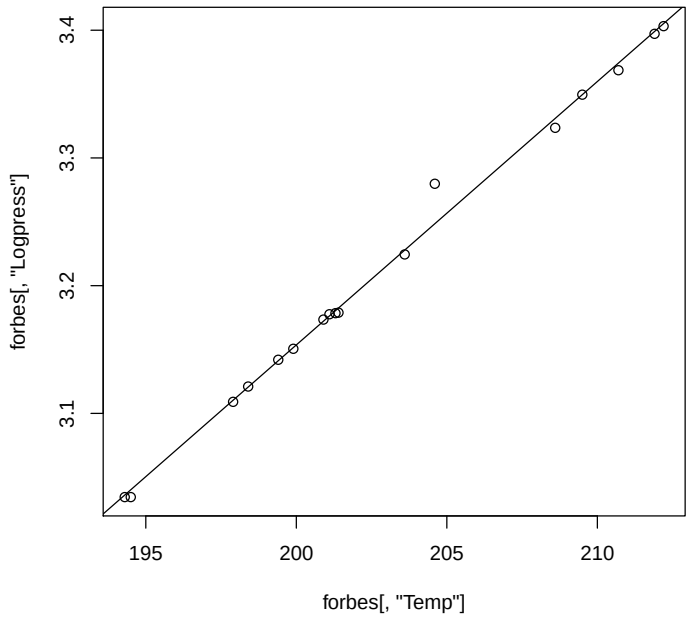



```
> qqnorm(resid(forbes.fit))
> qqline(resid(forbes.fit))
```



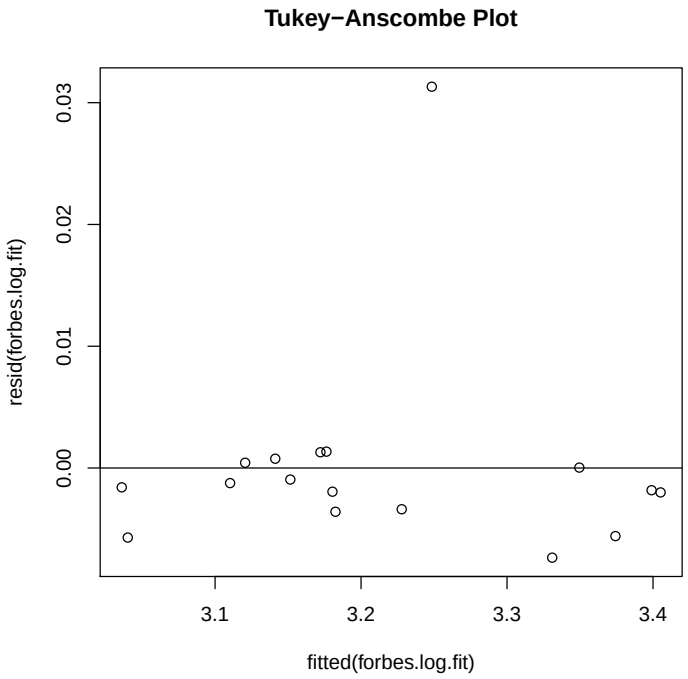
## 1.4 d

```
> forbes[,"Logpress"] <- log(forbes[,"Press"])
> plot(forbes[,"Temp"], forbes[,"Logpress"])
> forbes.log.fit <- lm(Logpress ~ Temp, data = forbes)
> abline(forbes.log.fit)
```

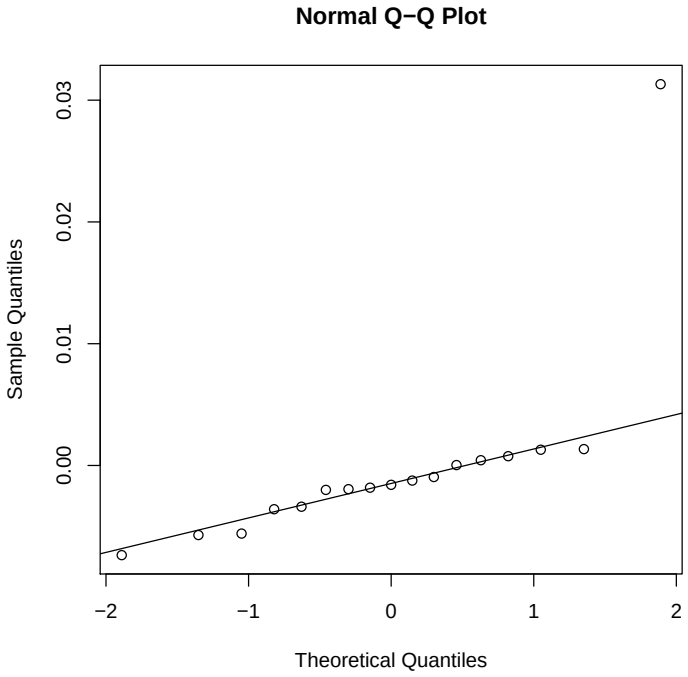



## 1.5 e

```
> plot(fitted(forbes.log.fit), resid(forbes.log.fit), main="Tukey-Anscombe Plot")
> abline(h=0)
```

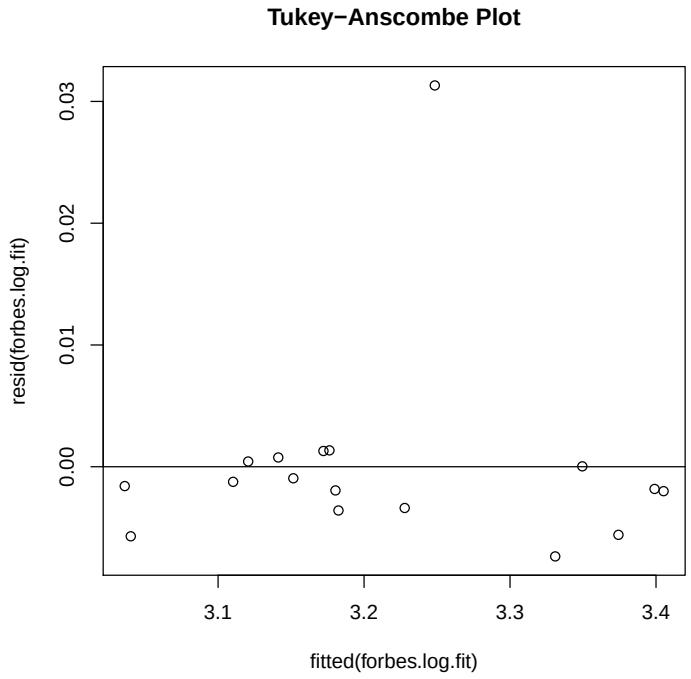



```
> qqnorm(resid(forbes.log.fit))
> qqline(resid(forbes.log.fit))
```

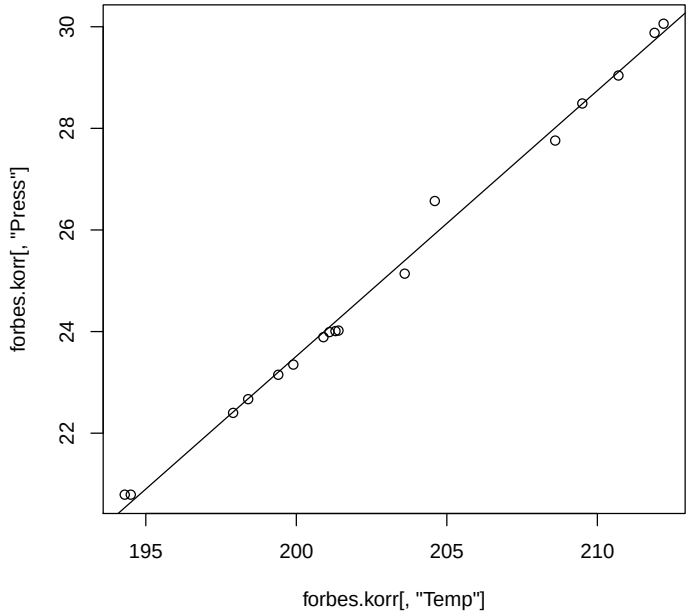


## 1.6 f

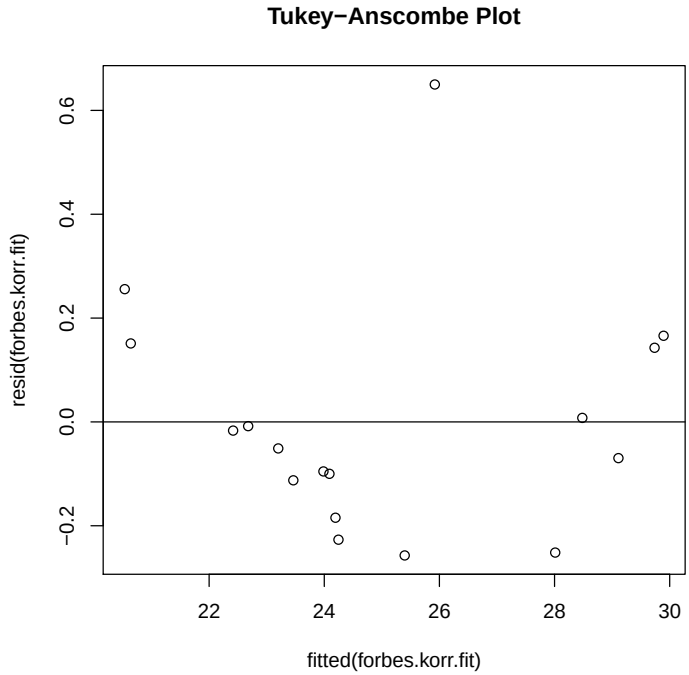
```
> plot(fitted(forbes.log.fit), resid(forbes.log.fit), main="Tukey-Anscombe Plot")
> abline(h=0)
> # identify(fitted(forbes.log.fit), resid(forbes.log.fit))
> forbes.korr=forbes[-12]
```



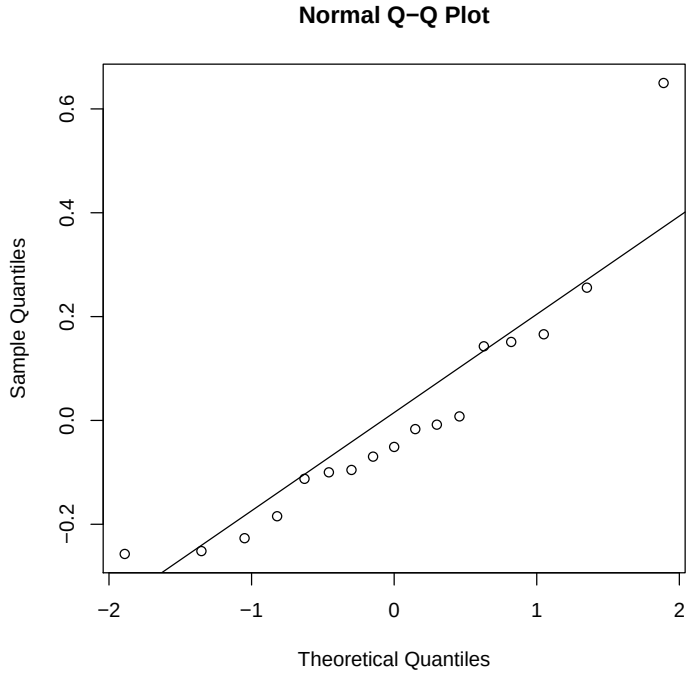

```
> forbes.korr.fit <- lm(Press ~ Temp, data = forbes.korr)
> # summary(forbes.korr.fit)
> plot(forbes.korr[,"Temp"], forbes.korr[,"Press"])
> abline(forbes.korr.fit)
```



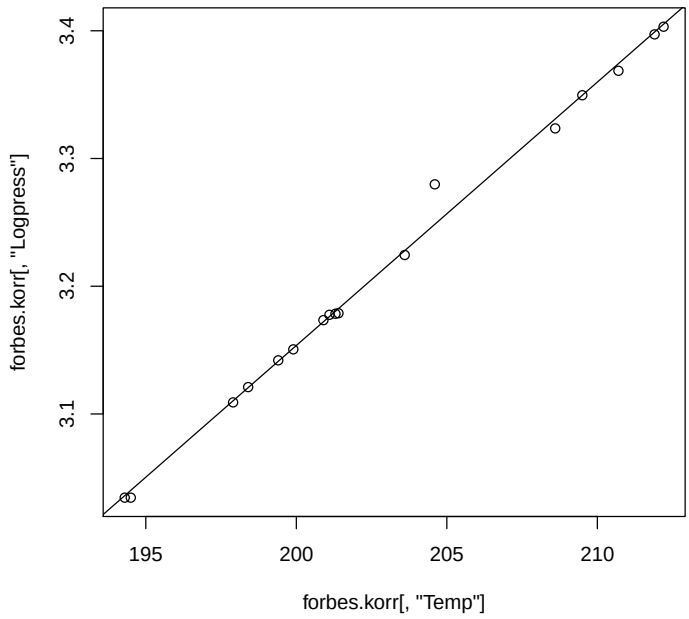

```
> plot(fitted(forbes.korr.fit), resid(forbes.korr.fit), main="Tukey-Anscombe Plot")
> abline(h=0)
```



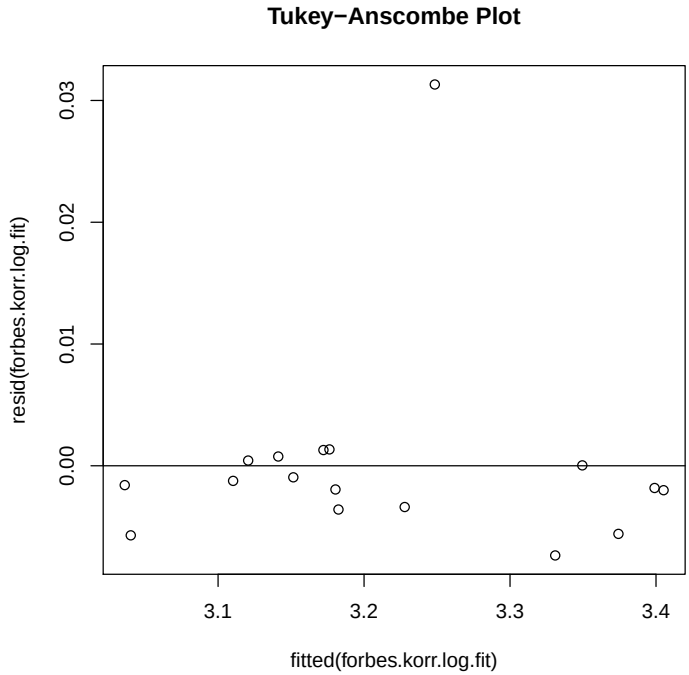
```
> qqnorm(resid(forbes.korr.fit))
> qqline(resid(forbes.korr.fit))
```



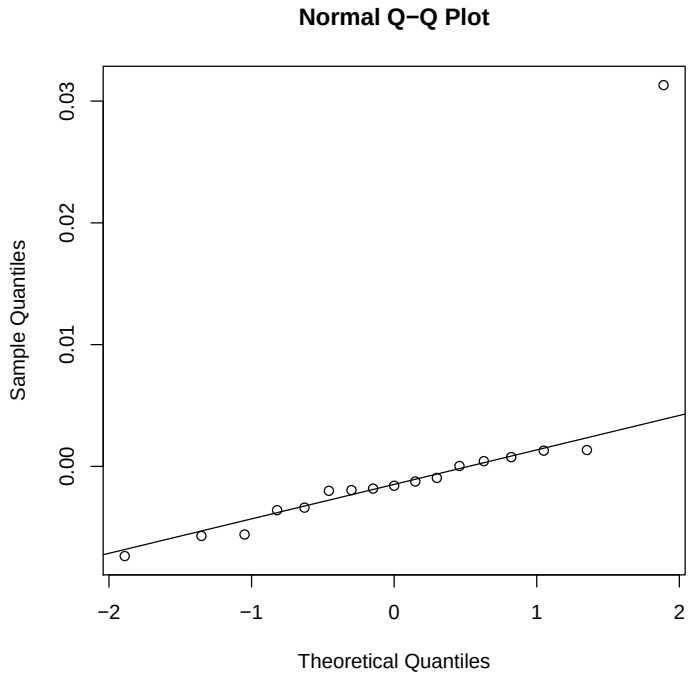

```
> forbes.korr[,"Logpress"] <- log(forbes.korr[,"Press"])
> plot(forbes.korr[,"Temp"], forbes.korr[,"Logpress"])
> forbes.korr.log.fit <- lm(Logpress ~ Temp, data = forbes.korr)
> abline(forbes.korr.log.fit)
```




```
> plot(fitted(forbes.korr.log.fit), resid(forbes.korr.log.fit), main="Tukey-Anscombe Plot"
> abline(h=0)
```



```
> qqnorm(resid(forbes.korr.log.fit))
> qqline(resid(forbes.korr.log.fit))
```




# 2 Aufgabe 2

```
> dist=c(100,200,400,800,1000,1500,2000,3000,5000,10000,20000,25000,30000)
> time=c(9.9,19.8,43.8,103.7,136,213.1,296.2,457.6,793.0,1650.8,3464.4,4495.6,5490.4)
> summary(lm(time ~ dist))

Call:
lm(formula = time ~ dist)

Residuals:
    Min      1Q  Median      3Q     Max
-106.95  -24.90   15.77   33.71  102.08

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -62.59296   21.81098   -2.87   0.0152 *
dist          0.18170    0.00173  105.05   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 62.68 on 11 degrees of freedom
Multiple R-squared: 0.999,        Adjusted R-squared: 0.9989
F-statistic: 1.103e+04 on 1 and 11 DF,  p-value: < 2.2e-16
```

## 2.1 a

Es gibt einen signifikanten Zusammenhang zwischen Distanz und Zeit, da Pr(>|t|) < 2e-16 ist.

## 2.2 b

```
> 0.18170-0.00173*qt(p=0.975,df=11)

[1] 0.1778923

> 0.18170+0.00173*qt(p=0.975,df=11)

[1] 0.1855077
```

ii

## 2.3 c

Mittels Formel:

```
> time[5]-(0.18170*dist[5]+(-62.59296))

[1] 16.89296
```

Mit R-Funktionen

```
> residuals(lm(time~dist))[5]

      5
16.8959
```

## 2.4 d

Nein

## 2.5 e

```
> sd(residuals(lm(time~dist)))

[1] 60.01135
```

Das Modell ist nur für grosse Werte von `time` brauchbar.

## 2.6 f

Wahrscheinlich wirkt zusätzlich ein quadratisch wirkender Teil mit.

## 2.7 g

$Zeit_i = \beta_0 + \beta_1 \cdot Distanz + \beta_2 \cdot Distanz^2 + \epsilon_i$

# 3 Aufgabe 3

```
> d.nebel <- read.table("hubble.dat",header=T,sep=",")
> nebel.v=d.nebel[,2]
> nebel.dist=d.nebel[,3]
> # summary(lm(nebel.dist ~ 0+nebel.v))
> lm(nebel.dist ~ 0+nebel.v)

Call:
lm(formula = nebel.dist ~ 0 + nebel.v)

Coefficients:
 nebel.v
0.001922

> alter=979.8*0.001922 # Alter in Milliarden Jahren
> alter

[1] 1.883176

> 0.3990982+qt(p=0.025,df=22)*0.1184697

[1] 0.1534071

> 0.3990982+qt(p=0.975,df=22)*0.1184697

[1] 0.6447893
```

# 4 Aufgabe 4

# 5 Aufgabe 5

# 6 Aufgabe 6