Problem Sheet 2

Theory of Inference

Dom Hutchinson

```r
set.seed(16111998)
```

Question 1

Part a)

```r
head(cars);
```

```
##   speed dist
## 1     4    2
## 2     4   10
## 3     7    4
## 4     7   22
## 5     8   16
## 6     9   10
```

Part b)

```r
cars.model<-lm(dist~speed+I(speed^2)-1,data=cars) # response vars ~ expected predictors form # I() ensu
cars.model
```

```
##
## Call:
## lm(formula = dist ~ speed + I(speed^2) - 1, data = cars)
##
## Coefficients:
##      speed  I(speed^2)
##    1.23903     0.09014
```

$\hat{\beta}_1 = 1.23903$ & $\hat{\beta}_2 = 0.0901388$.

Part c)

```r
summary(cars.model)
```

```
##
## Call:
## lm(formula = dist ~ speed + I(speed^2) - 1, data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -28.836  -9.071  -3.152   4.570  44.986
##
## Coefficients:
##            Estimate Std. Error t value Pr(>|t|)
```

```
## speed       1.23903    0.55997   2.213  0.03171 *
## I(speed^2)  0.09014    0.02939   3.067  0.00355 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.02 on 48 degrees of freedom
## Multiple R-squared:  0.9133, Adjusted R-squared:  0.9097
## F-statistic: 252.8 on 2 and 48 DF,  p-value: < 2.2e-16
```

```
coefs<-summary(cars.model)$coefficients
coefs["speed","Estimate"]
```

```
## [1] 1.23903
```

$\hat{\beta}_1 = 1.23903,\ \hat{\beta}_2 = 0.0901388,\ \hat{\sigma}_{\hat{\beta}_1} = 0.5599707$ & $\hat{\sigma}_{\hat{\beta}_2} = 0.0293892$.

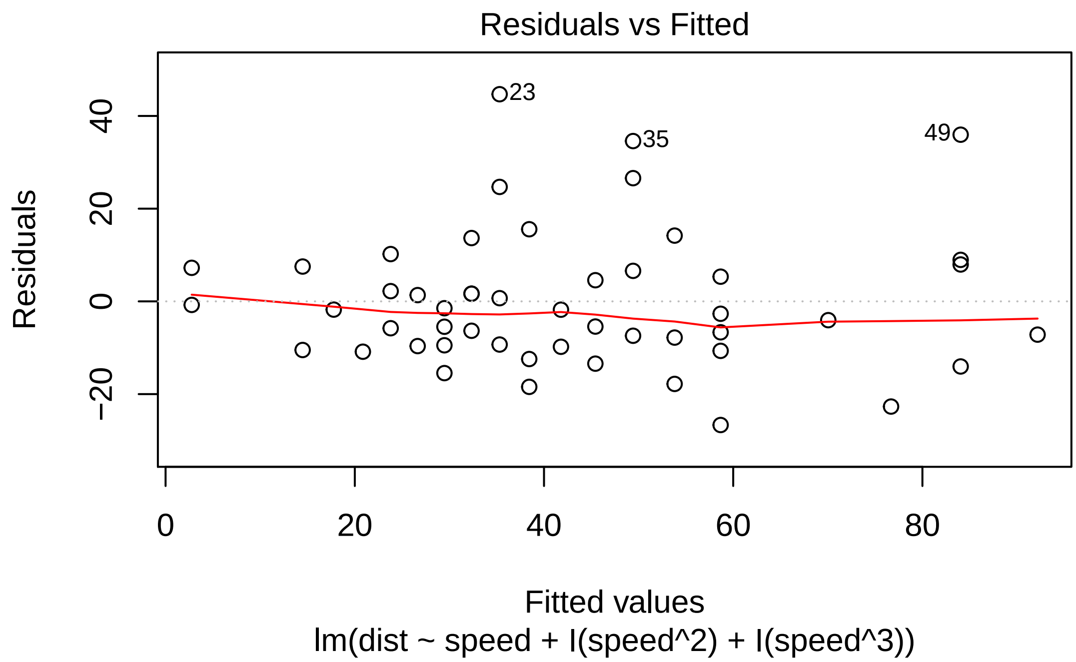
Pard d)

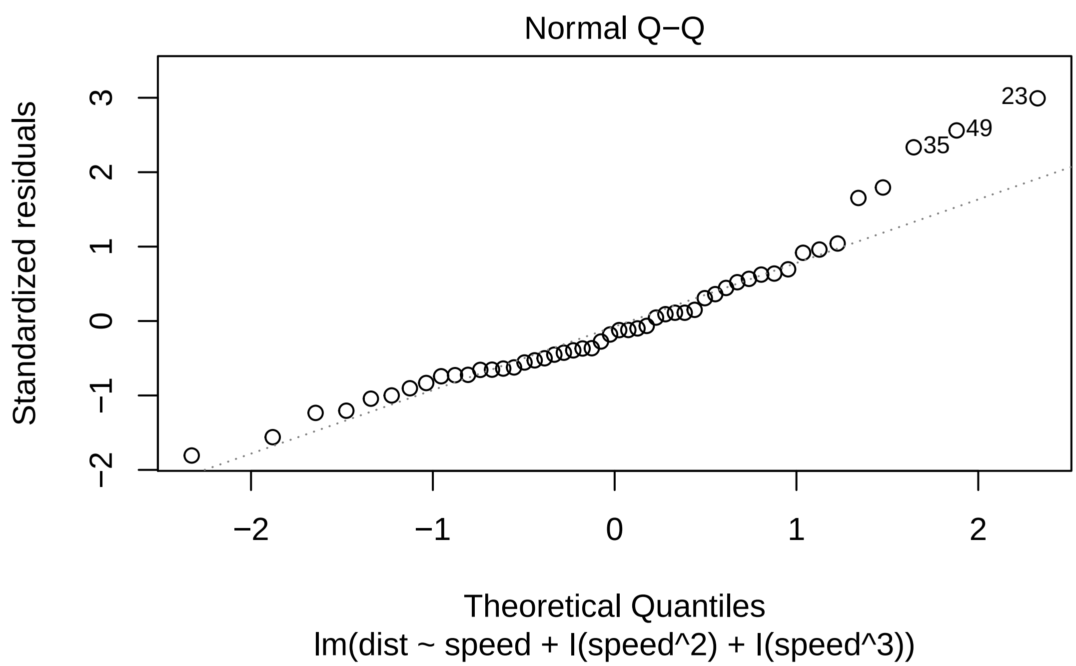
```
head(model.matrix(cars.model))
```

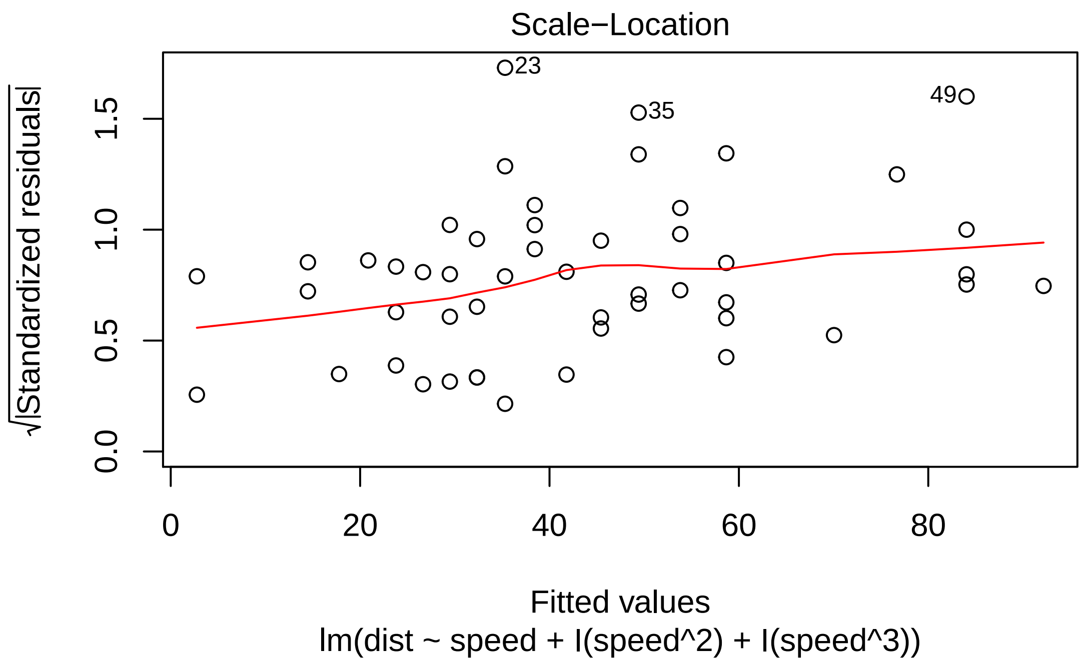
```
##   speed I(speed^2)
## 1     4         16
## 2     4         16
## 3     7         49
## 4     7         49
## 5     8         64
## 6     9         81
```

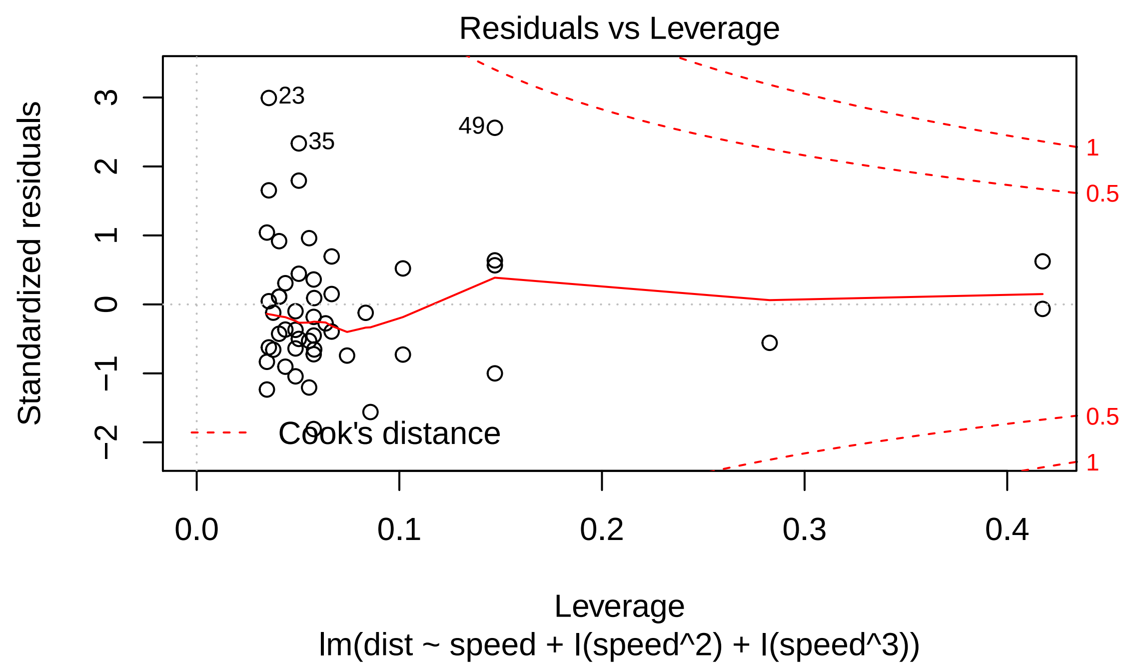
Part e)

```
cm1<-lm(dist~speed+I(speed^2)+I(speed^3),data=cars)
plot(cm1)
```


Residuals vs Fitted
Residuals
Fitted values
lm(dist ~ speed + I(speed^2) + I(speed^3))
23
35
49
40
20
0
−20
0
20
40
60
80


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(dist ~ speed + I(speed^2) + I(speed^3))
23
35
49
3
2
1
0
−1
−2
−2
−1
0
1
2





The mean of the residuals deviates further from the mean as the fitted value increases, indicating it is less accurate for high values. The variability seems fairly consistent, possibly increasing as the fidded values increase. The mean of the residuals is less zero meaning the predicted valeus are consistently less than the true values.

Part f)

```
summary(cm1)
```

```
##
## Call:
## lm(formula = dist ~ speed + I(speed^2) + I(speed^3), data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -26.670  -9.601  -2.231   7.075  44.691
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -19.50505   28.40530  -0.687    0.496
## speed         6.80111    6.80113   1.000    0.323
## I(speed^2)   -0.34966    0.49988  -0.699    0.488
## I(speed^3)    0.01025    0.01130   0.907    0.369
##
## Residual standard error: 15.2 on 46 degrees of freedom
## Multiple R-squared:  0.6732, Adjusted R-squared:  0.6519
## F-statistic: 31.58 on 3 and 46 DF,  p-value: 3.074e-11
```

```
cm2<-lm(dist~speed+I(speed^2)+I(speed^3)-1,data=cars)
summary(cm2)
```

```
##
## Call:
## lm(formula = dist ~ speed + I(speed^2) + I(speed^3) - 1, data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -27.741  -8.755  -4.049   5.435  45.345
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## speed       2.299945   1.802672   1.276    0.208
## I(speed^2) -0.038399   0.209557  -0.183    0.855
## I(speed^3)  0.003638   0.005871   0.620    0.539
##
## Residual standard error: 15.12 on 47 degrees of freedom
## Multiple R-squared:  0.914,  Adjusted R-squared:  0.9085
## F-statistic: 166.5 on 3 and 47 DF,  p-value: < 2.2e-16
```

```
cm3<-lm(dist~speed+I(speed^3)-1,data=cars)
summary(cm3)
```

```
##
## Call:
## lm(formula = dist ~ speed + I(speed^3) - 1, data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -28.084  -9.058  -4.116   5.062  45.289
##
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## speed      1.9751471  0.3250123   6.077 1.91e-07 ***
## I(speed^3) 0.0025727  0.0008204   3.136  0.00292 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.97 on 48 degrees of freedom
## Multiple R-squared:  0.9139, Adjusted R-squared:  0.9103
## F-statistic: 254.8 on 2 and 48 DF,  p-value: < 2.2e-16
```

```
coefs3<-summary(cm3)$coefficients
```

By dropping the least significant term until all terms have p-valeus less than 0.05 we get the suggestion that the following is the best model

$$\texttt{dist}_i=\beta_1\texttt{speed}_i+\beta_2\texttt{speed}_i^3+\varepsilon_i$$

where $\hat{\beta}_1=1.9751471$ & $\hat{\beta}_2=0.0025727$.

Part g)

$\texttt{time}_i=\frac{\texttt{dist}_i}{\beta_1\texttt{speed}_i}$.
We obtain a 95% confidence interval for β_1 using $\hat{\beta}_1\pm t_{n-p}(.975)\hat{\sigma}_{\hat{\beta}_1}$.
In this scenario $n=50,\ p=2,\ \hat{\beta}_1=1.9751471$ & $\hat{\sigma}_{\hat{\beta}_1}=0.3250123$. Note that $t_{48}(.975)=2.0106348$.
Producing the following confidence interval for β_1.

$$[1.3216661, 2.6286281]$$

If we now take the sample means `speed` & `dist` of we can produce a confidence interval for `time` (accounting for speed being in miles/hour & distance being in feet).

$$\frac{1}{5280\times 60\times 60}\left[\frac{1.3216661\times 42.98}{15.4},\frac{2.6286281\times 42.98}{15.4}\right]=\left[1.9405776\times 10^{-7},3.8595654\times 10^{-7}\right]$$

The final confidence interval is for the reaction speed in seconds.
Question 2

Part a)

```
n<-100                     # Sample size
beta.true<-c(.5,1,10)      # True parameter values
ct<-qt(.975,n-3)           # Critical points for CIs
cp<-beta.true*0            # Coverage probability array
n.rep<-1000                # Number of replicates to run
for (i in 1:n.rep) {
  x<-runif(n)                                                      # simulated covariate
  mu<-beta.true[1]+beta.true[2]*x+beta.true[3]*x^2
  y<-mu+rnorm(n)*.3                                                # Simulated data
  m1<-lm(y~x+I(x^2))                                               # fit model to this replicate
  b<-coef(m1)                                                      # extract parameter estimates
  sig.b<-diag(vcov(m1))^.5
  cp<-cp+as.numeric(b-ct*sig.b<=beta.true & b+ct*sig.b>=beta.true) # Count whether coefficients in inte
}
cp/n.rep
```

```
## [1] 0.950 0.959 0.951
```

Observed coverage is close to the nomial coverage of .95.

Part b)

```
n<-100                  # Sample size
beta.true<-c(.5,1,10) # True parameter values
ct<-qt(.975,n-3)      # Critical points for CIs
cp<-beta.true*0       # Coverage probability array
n.rep<-1000           # Number of replicates to run
for (i in 1:n.rep) {
  x<-runif(n)                                             # simulated covariate
  mu<-beta.true[1]+beta.true[2]*x+beta.true[3]*x^2
  y<-rpois(n,mu)                                        # Simulated data
  m1<-lm(y~x+I(x^2))                                      # fit model to this replicate
  b<-coef(m1)                                             # extract parameter estimates
  sig.b<-diag(vcov(m1))^.5
  cp<-cp+as.numeric(b-ct*sig.b<=beta.true & b+ct*sig.b>=beta.true) # Count whether coefficients in inte
}
cp/n.rep
```

```
## [1] 0.999 0.972 0.936
```

The coverage increases for the constant & linear `speed` terms but decreases for quadratic `speed`

Part c)

```
# Generate data
n<-50;
x<-runif(n)
mu<-beta.true[1]+beta.true[2]*x+beta.true[3]*x^2
y<-rpois(n,mu)

# Bootstrap
cp<-beta.true*0
n.rep<-1000
for (i in 1:n.rep) {
  bi<-sample(1:n,n,replace=TRUE)
  yb<-y[bi]
  xb<-x[bi]
  m1<-lm(yb~xb+I(xb^2))
  b<-coef(m1)
  sig.b<-diag(vcov(m1))^.5
  cp<-cp+as.numeric(b-ct*sig.b<=beta.true & b+ct*sig.b>=beta.true) # Count whether coefficients in inte
}
cp/n.rep
```

```
## [1] 0.994 0.942 0.861
```