Mandatory exercise 1
STK2100 - Spring 2020

By Ida Johanne Austad

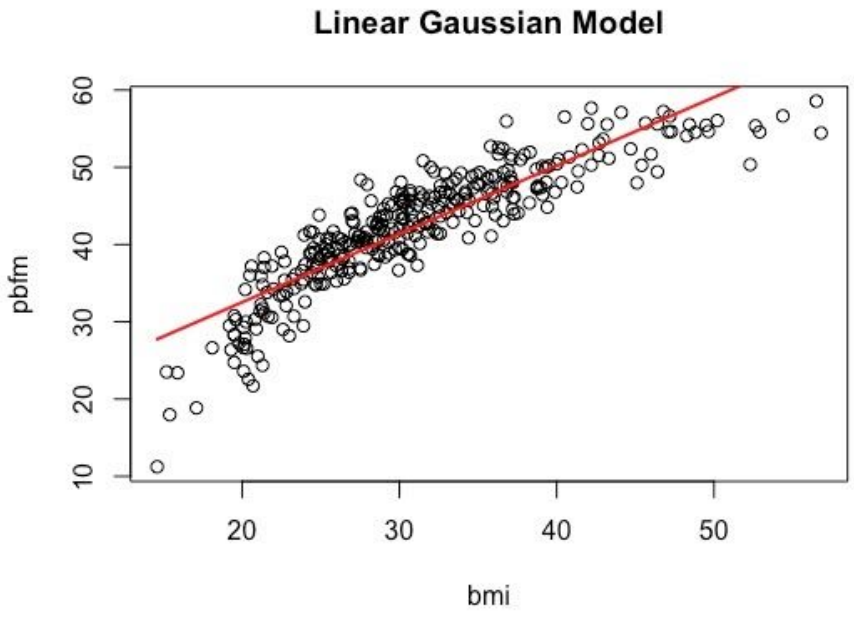
See appendix for R-code. The code output and figures in the answers are included in the main response below.

Problem 1

a) Result from fitting linear model:

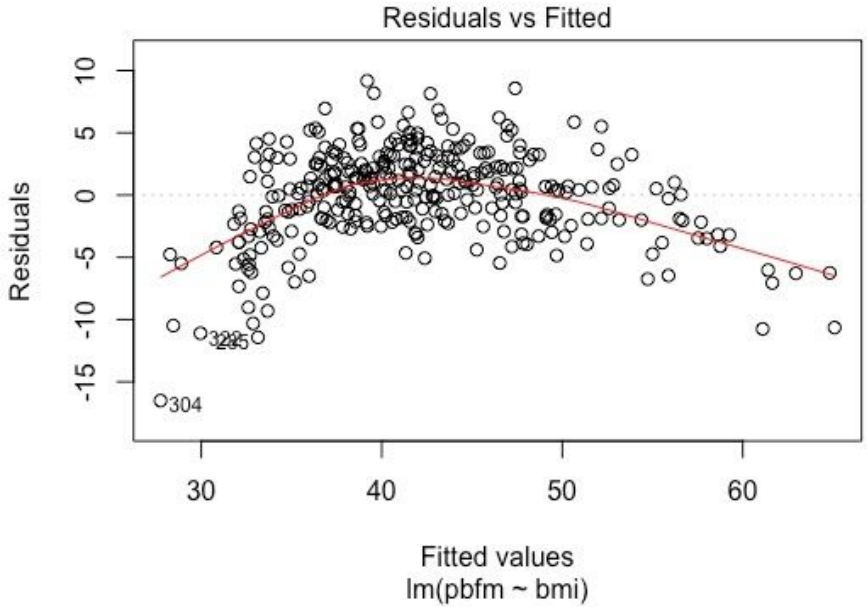
```
Call:
lm(formula = pbfm ~ bmi)

Residuals:
     Min       1Q   Median       3Q      Max
-16.5116  -2.0714   0.4083   2.4994   9.1758

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 14.82772    0.82671   17.94   <2e-16 ***
bmi          0.88481    0.02589   34.17   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.702 on 325 degrees of freedom
Multiple R-squared:  0.7823,  Adjusted R-squared:  0.7816
F-statistic:  1168 on 1 and 325 DF,  p-value: < 2.2e-16
```

Visualization of fitted model:

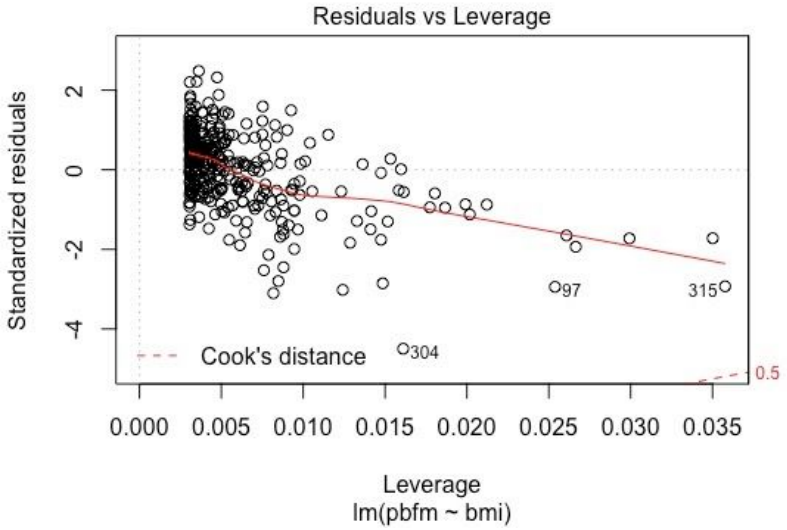


At a first glance, the linear model seems to fit the data points ok for bmi values between 20 and 45. However, outside this interval the model looks a bit off. Both the intercept and the bmi are strongly significant, indicating a strong correlation between the two variables.

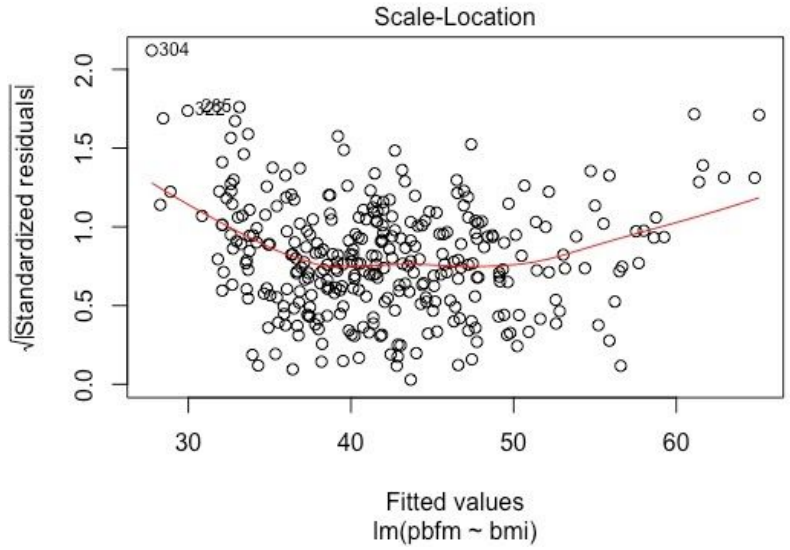
The four diagnostics plots:



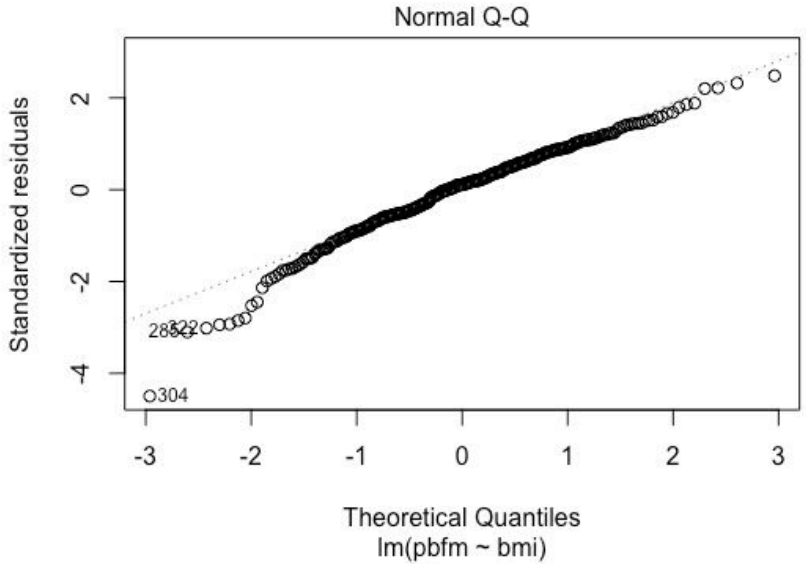
Residual vs fitted (Anscombe plot): This plot indicates whether there is a non-linear relationship between the residuals. If the relationship was linear, we would see the points evenly distributed around the dotted line. However, this is not the case for our model.



Residuals vs leverage: This plot would help us detect outliers of influence, which would be indicated by data points outside the red dotted lines. We do not see any such cases.



Scale-Location: This plot allows you to check the assumption of equal variance (homoscedasticity) - indicated by a horizontal line with equally (randomly) spread points. As we can see, this is not the case - indicating that the homoscedasticity assumption is not met.

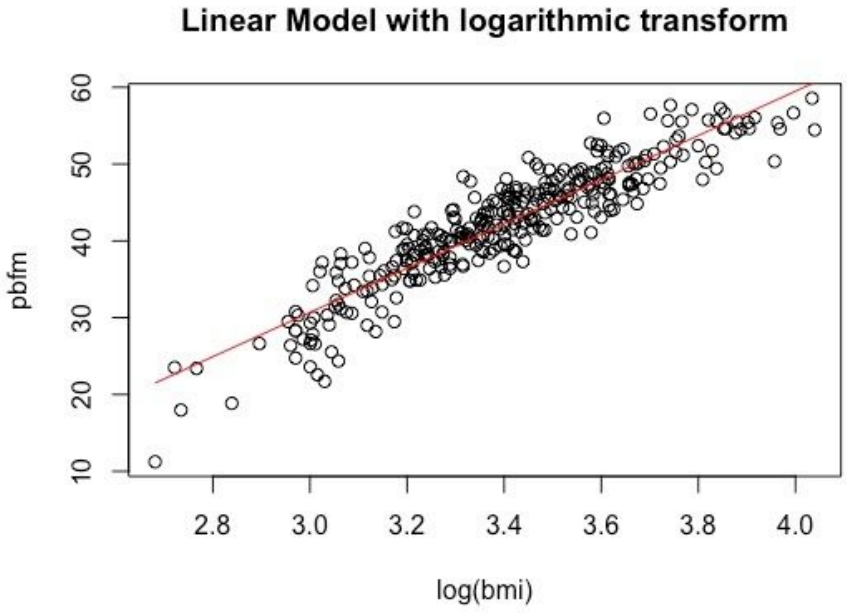


Normal Q-Q: This plot allows us to see if the residuals seem to follow a normal distribution, indicated by the residuals following the dotted line. However, we can see from the plot that the head and tails are off, and thus may not support the normality assumption.

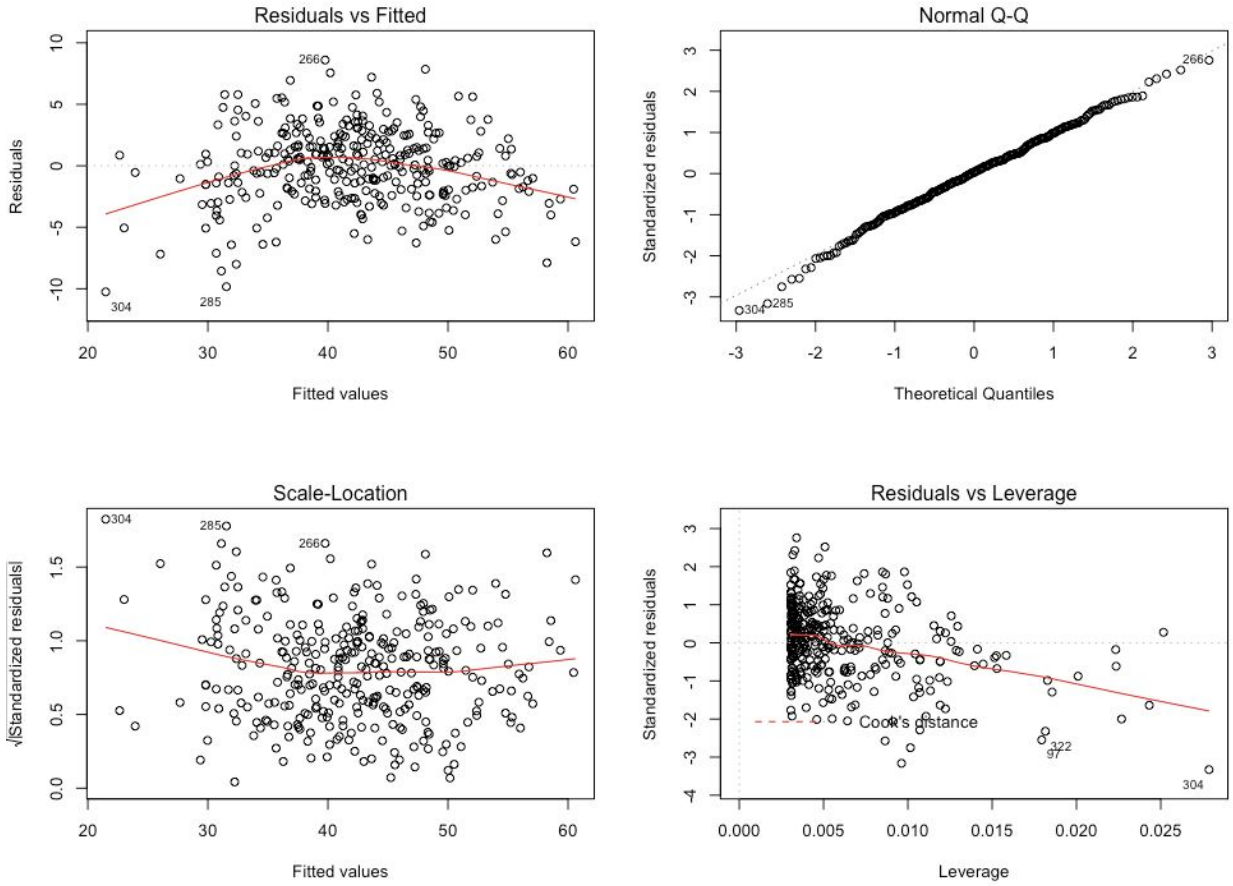
b)

Since we do not have any negative values, a logarithmic transformation may be a good choice.
Results of fitted model with logarithmic transform:

```
Call:
lm(formula = pbfm ~ log(bmi))

Residuals:
     Min       1Q   Median       3Q      Max
-10.2548  -2.0453   0.1026   2.1238   8.6029

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -55.7327     2.3337  -23.88   <2e-16 ***
log(bmi)     28.8031     0.6845   42.08   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.125 on 325 degrees of freedom
Multiple R-squared:  0.8449,  Adjusted R-squared:  0.8444
F-statistic:  1771 on 1 and 325 DF,  p-value: < 2.2e-16
```

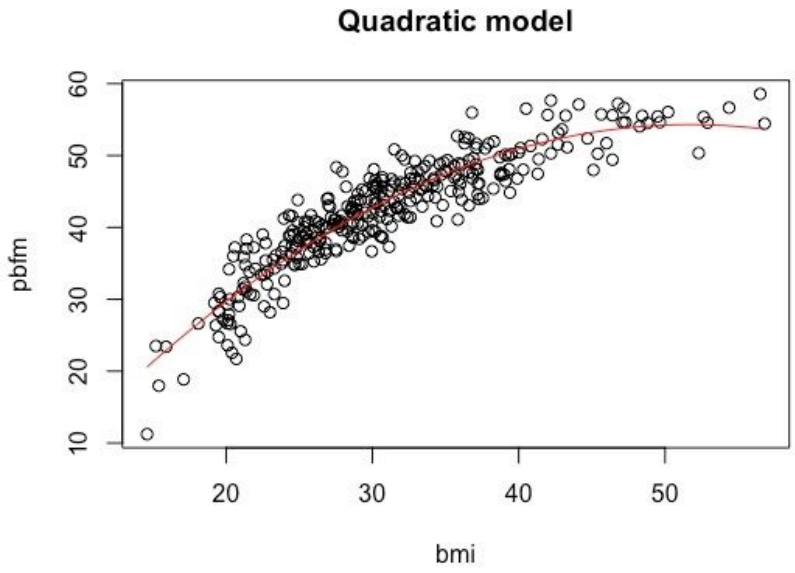


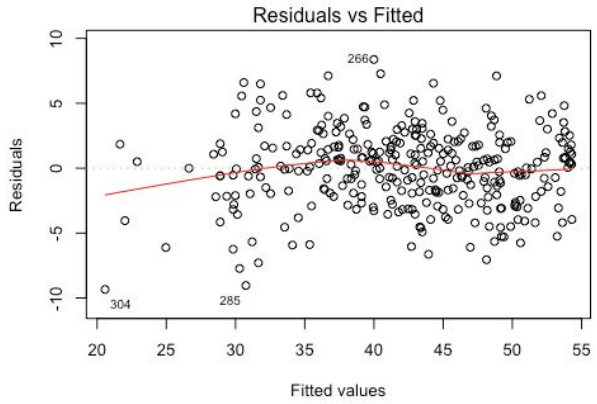
Diagnostics plot:

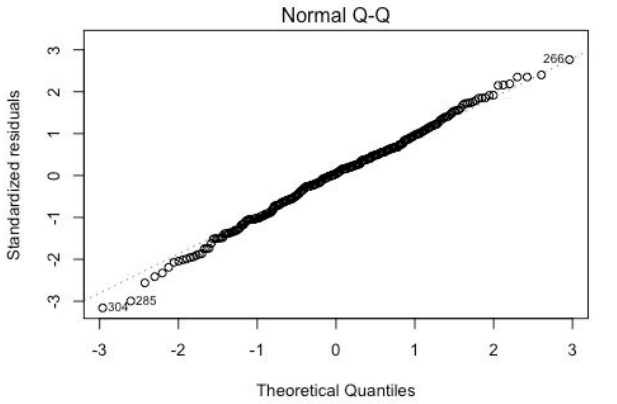


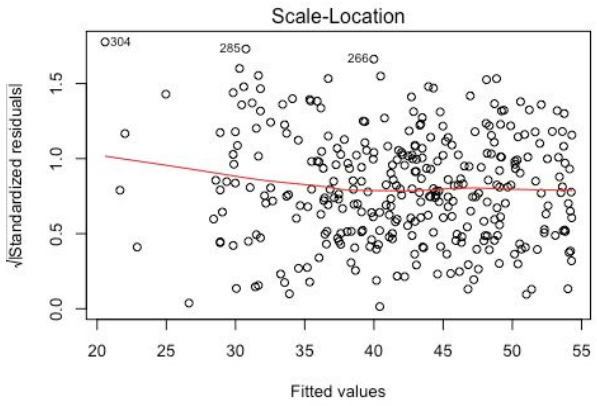
Results from fitted model with both linear and quadratic effect (no transform):

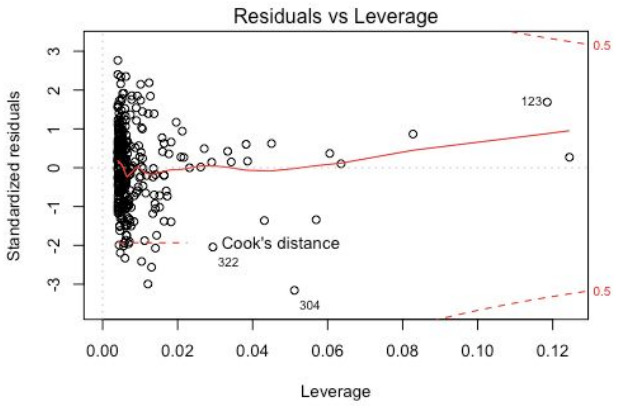
```
Call:
lm(formula = pbfm ~ bmi + I(bmi^2))

Residuals:
    Min      1Q  Median      3Q     Max
-9.3403 -1.9246  0.1433  1.8665  8.3780

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -11.17790    2.16977  -5.152 4.49e-07 ***
bmi           2.53223    0.13229  19.142  < 2e-16 ***
I(bmi^2)     -0.02448    0.00194 -12.617  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.036 on 324 degrees of freedom
Multiple R-squared:  0.854,   Adjusted R-squared:  0.8531
F-statistic: 947.8 on 2 and 324 DF,  p-value: < 2.2e-16
```











Based on the results of the two models, at first glance - it is hard to see a clear difference in terms of which is best based on the diagnostics plots. However, the R-squared values for the model with the quadratic terms is higher - suggesting that this model might be slightly better as it explains more of the variability in the data. Still, both make improvements over the model in a).

c) Running the attached script reveals that a polynomial model of fourth order provides the best result.

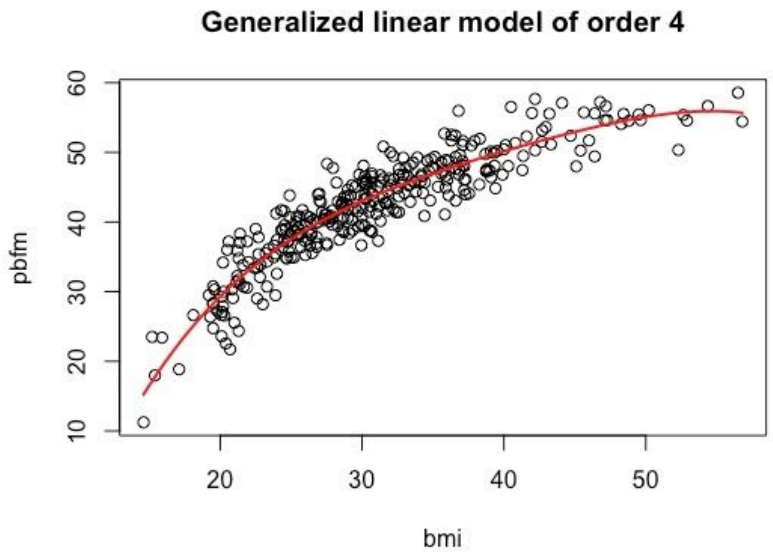
MSE of each order using 10-fold cross validation:

```
13.892645  9.494127  8.894479  8.820730  9.144608  9.089521  9.152449
9.237657  9.639235  9.641665
```

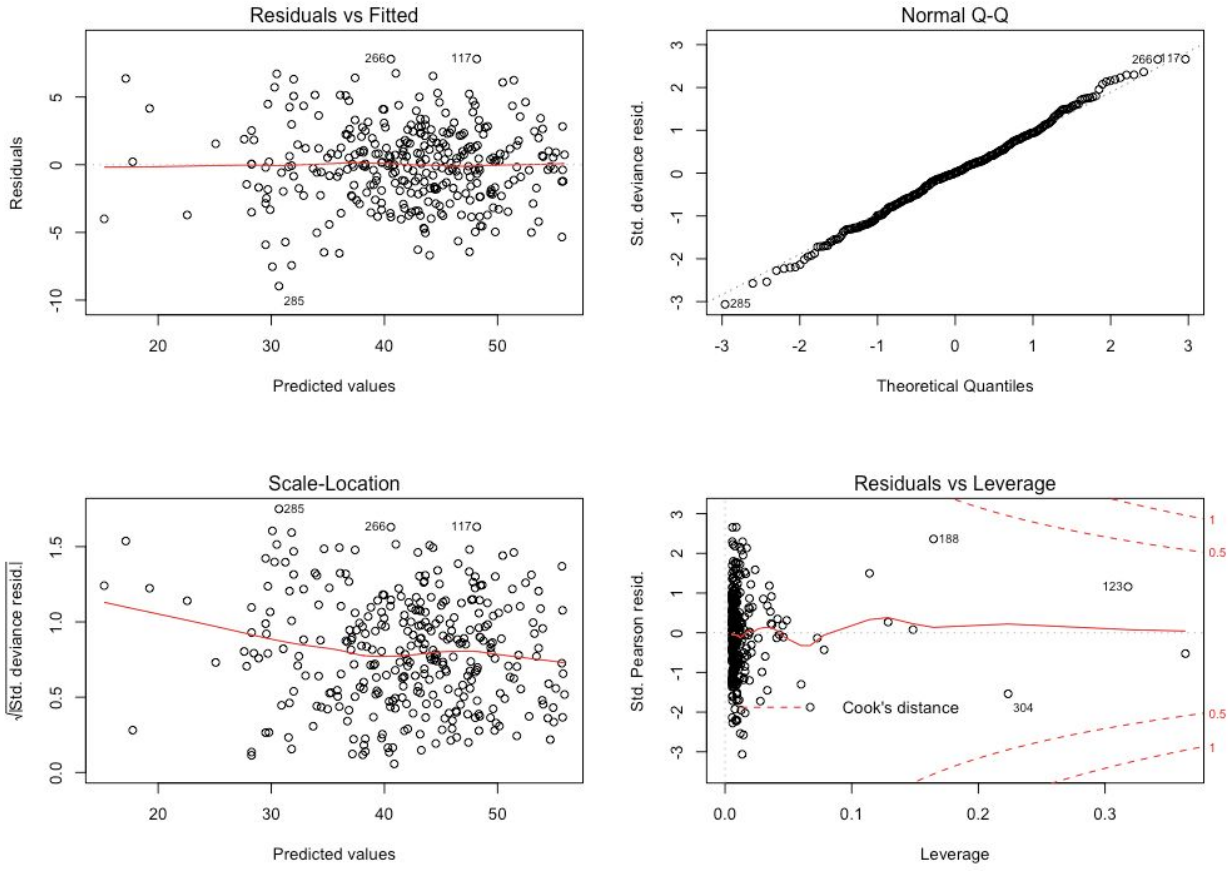
Summary of the best model (of fourth order):

```
Call:
glm(formula = pbfm ~ poly(bmi, 4), data = bodyfat)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-8.9670  -1.8763   0.0443   1.8563   7.8093

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)     42.200      0.163 258.956  < 2e-16 ***
poly(bmi, 4)1  126.529      2.947  42.937  < 2e-16 ***
poly(bmi, 4)2  -38.312      2.947 -13.001  < 2e-16 ***
poly(bmi, 4)3   12.181      2.947   4.134 4.56e-05 ***
poly(bmi, 4)4   -6.536      2.947  -2.218   0.0273 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 8.683972)

    Null deviance: 20464.8  on 326  degrees of freedom
Residual deviance:  2796.2  on 322  degrees of freedom
AIC: 1641.8

Number of Fisher Scoring iterations: 2
```

Visualization fo result:



Diagnostics plot:



The diagnostics plots for this model looks even better than the ones in a) and b). Firstly the residuals seem to be be more close to a normal distribution (Normal Q-Q). Also, the lines in the residual vs fitted plot and the scale-location plot are closer to linear with the residuals more evenly distributed around the center line. We still no not find any influential outliers in the residuals vs leverage plot.

d) Resulting AIC-score for models with an increasing number of polynomial order:

```
1788.062 1659.368 1644.708 1641.751 1643.617 1643.492 1645.330 1647.187
1648.974 1646.426
```

As we can see here, using AIC information score (negative log likelihood penalized for a number of parameters) also suggests that a polynomial model of fourth order is the best - as this gives the lowest score.

Problem 2

a) Resulting table from dichotomization:

```
ccstatus non-smoker smoker
       0         69    134
       1         22    172
```

Estimated probabilities and standard deviation:

- Overall: 0.489 and 0.025
- Smokers: 0.562 and 0.028
- Non Smokers: 0.242 and 0.045

b)

Hypothesis:

H0: probSmokers == probNonSmokers
H1: probSmokers != probNonSmokers

Computed the p-value to decide whether the null hypothesis can be rejected or not. As the answer is zero, we can conclude that the null hypothesis can be rejected, and as such that there is a significant different probability between the groups.

c) Summary from fitted binomial model:

```
Call:
glm(formula = y ~ x, family = "binomial")

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-1.285  -1.285  -0.744   1.073   1.685

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.1431     0.2448  -4.669 3.03e-06 ***
xTRUE         1.3927     0.2706   5.147 2.65e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
 (Dispersion parameter for binomial family taken to be 1)

     Null deviance: 550.15  on 396  degrees of freedom
 Residual deviance: 520.14  on 395  degrees of freedom
 AIC: 524.14

 Number of Fisher Scoring iterations: 4
```

Calculating the odds for smokers versus non-smokers gives approximately an answer of 4 to 1, indicating that it is clearly an increased risk of oral cancer in case of smoking.

d) Summary from binomial model with continuous variable for number of cigarettes:

```
Call:
glm(formula = y ~ x, family = "binomial")

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1923  -1.0237  -0.8228   1.1088   1.5796

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.909057   0.171766  -5.292 1.21e-07 ***
x            0.053624   0.008614   6.225 4.81e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 550.15  on 396  degrees of freedom
Residual deviance: 504.39  on 395  degrees of freedom
AIC: 508.39

Number of Fisher Scoring iterations: 4
```

The regression coefficient indicates that the probability of oral cancer increases with the number of cigarettes smoked per day. The intercept is different as, for instance, smoking one cigarette per day is closer to no cigarettes than 20. For the non-continuous model, all the smokers get the same odds - no matter how many cigarettes they smoke.

e) Results from binomial model including all features as predictors:

```
Call:
glm(formula = y ~ oral$drinks + oral$cigs + oral$age + oral$sex,
    family = "binomial")
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7185  -0.8589  -0.5832   0.9644   1.9776

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.966071   0.620756  -3.167  0.00154 **
oral$drinks  0.029623   0.004643   6.380 1.77e-10 ***
oral$cigs    0.035480   0.009571   3.707  0.00021 ***
oral$age     0.006529   0.009960   0.656  0.51213
oral$sex     0.594499   0.272752   2.180  0.02928 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 550.15  on 396  degrees of freedom
Residual deviance: 443.84  on 392  degrees of freedom
AIC: 453.84

Number of Fisher Scoring iterations: 5
```

Here we get 0.035 as log-odds related to an increasing number of cigarettes per day. This is a little lower than for the previous model. The reason for this could be that when we only included one feature, this feature “tried” to explain more of the variance. When including more features for explanation changes as the new features can also serve as predictors.

f) Summary of results from leaving age out:

```
Call:
glm(formula = y ~ oral$drinks + oral$cigs + oral$sex, family = "binomial")

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6943  -0.8596  -0.6084   0.9607   1.8857

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.592919   0.238787  -6.671 2.54e-11 ***
oral$drinks  0.029498   0.004638   6.360 2.01e-10 ***
oral$cigs    0.035536   0.009565   3.715 0.000203 ***
oral$sex     0.582183   0.271756   2.142 0.032169 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)
```

```
    Null deviance: 550.15  on 396  degrees of freedom
Residual deviance: 444.27  on 393  degrees of freedom
AIC: 452.27

Number of Fisher Scoring iterations: 5
```

As age showed a large p-value in the model in e), this indicates that it does not add significant explanatory value to the model. This informs us that the data does not show a clear relationship between age and oral cancer. The conclusion can then be to leave the age-feature out of the model.

g) Results from fitting a model with second order polynomial for drinks:

```
Call:
glm(formula = y ~ poly(oral$drinks, 2) + oral$cigs + oral$sex,
    family = "binomial")

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2192  -0.8379  -0.5405   0.8756   1.9980

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)           -0.735884   0.223006  -3.300 0.000967 ***
poly(oral$drinks, 2)1 20.132066   3.047200   6.607 3.93e-11 ***
poly(oral$drinks, 2)2 -7.430462   2.725478  -2.726 0.006405 **
oral$cigs              0.033010   0.009642   3.424 0.000618 ***
oral$sex               0.725644   0.285409   2.542 0.011007 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 550.15  on 396  degrees of freedom
Residual deviance: 437.62  on 392  degrees of freedom
AIC: 447.62

Number of Fisher Scoring iterations: 4
```

Adding a quadratic polynomial for drinks does not make a large improvement to the model, as it gets a larger p-value compared to the others. Still, it is significant.

h) Results from fitting a model with second order polynomial for cigs:

```
Call:
glm(formula = y ~ oral$drinks + poly(oral$cigs, 2) + oral$sex,
    family = "binomial")

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6870  -0.8764  -0.5808   0.9695   1.9303

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)         -1.015076   0.187761  -5.406 6.44e-08 ***
oral$drinks          0.028979   0.004635   6.253 4.04e-10 ***
poly(oral$cigs, 2)1  9.620966   2.614970   3.679 0.000234 ***
poly(oral$cigs, 2)2 -2.215989   2.581437  -0.858 0.390654
oral$sex             0.601901   0.274284   2.194 0.028204 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 550.15  on 396  degrees of freedom
Residual deviance: 443.55  on 392  degrees of freedom
AIC: 453.55

Number of Fisher Scoring iterations: 5
```

Adding a quadratic polynomial term for cigs did not improve the model. In fact, the p-value of the quadratic term is not significant - indicated by the large p-value.

i) The AIC-score for the model with quadratic term for drinks is a bit lower than the one without, while the AIC-score increases when we add a quadratic term for the cigs variable. But overall, adding the quadratic terms did not seem to have a drastic positive effect on the models' explanatory power.

j) If the model was improved a lot by including a quadratic part to a certain feature, then that would indicate that it had more than twice as large negative consequences - for instance by having two drinks - compared to just one. It does not necessarily make it much worse to drink 8, the coefficients will just change - depending on whether we use a linear or quadratic model for this feature.

Appendix: R-script to produce the results presented before

```
# MANDATORY EXERCISE 1 (Spring 2020)
# By Ida Austad

# PROBLEM 1

# read in, inspect and prep data
path <- file.path(getwd() , "res_bodyfat.csv")
bodyfat <- read.csv(path)
summary(bodyfat)
head(bodyfat)
attach(bodyfat)

#a)
plot(bmi, pbfm, main="Linear Gaussian Model")
fit_linear <- lm(pbfm ~ bmi)
print(summary(fit_linear))
x = (seq(min(bmi), max(bmi), length = 200))
beta = coef(fit_linear)
lines(x, beta[1] + beta[2]* x, col = 2, lty = 1, lwd = 2)

#get diagnostics plots (one at a time)
plot(fit_linear)

#b)
# fit model with log transform of exp.variable
fit_log <- lm(pbfm ~ log(bmi))
print(summary(fit_log))

# plot model
beta = coef(fit_log)
x = log(seq(min(bmi), max(bmi), length = 200))
plot(log(bmi), pbfm, main="Linear Model with logarithmic transform")
lines(x,(beta[1] + beta[2]*x), col = 2)

#plot diagnostics
par(mfrow=c(2,2))
plot(fit_log)

# fit model with both linear and quadratic effect of exp. var
fit_quad<- lm(pbfm ~ bmi + I(bmi^2))
print(summary(fit_quad))

#plot model
beta = coef(fit_quad)
x = (seq(min(bmi), max(bmi), length = 200))
par(mfrow=c(1,1))
plot(bmi, pbfm, main="Quadratic model")
```

```
lines(x, beta[1] + beta[2]*x + beta[3]*x^2, col=2)

#plot diagnostics
par(mfrow=c(2,2))
plot(fit_quad)

#c) find order of polynomial which best fits the data (using cv.glm #method from boot library
- #ref 5.5.3 in ILS)
#create list to hold errors of each fold
set.seed(1)
cv_errors = rep(0,10)
for (i in 1:10){
  glm_fit = glm(pbfm ~ poly(bmi,i), data=bodyfat)
  cv_errors[i] = cv.glm(bodyfat, glm_fit, K=10)$delta[1]
}
#print cv errors
cv_errors

#get index of smallest value and error
which.min(cv_errors)
min(cv_errors)

#plot best model (order 4)
par(mfrow=c(1,1))
plot(bmi, pbfm, main="Generalized linear model of order 4")
fit_best <- glm(pbfm ~ poly(bmi,4), data=bodyfat)
print(summary(fit_best))
x = (seq(min(bmi), max(bmi), length = 200))
y = predict(fit_best, list(bmi = x), type="response")
lines(x, y, col = 2, lty = 1, lwd = 2)

#plot diagnostics
par(mfrow=c(2,2))
plot(fit_best)

#d) using AIC
set.seed(1)
cv_AIC = rep(0,10)
for (i in 1:10){
  glm_fit = glm(pbfm ~ poly(bmi,i), data=bodyfat)
  cv_AIC[i] = AIC(glm_fit)
}
#print cv errors
cv_AIC

#get index of smallest value and error
which.min(cv_AIC)
min(cv_AIC)
```

```
#-----------------------------------------------------------
# PROBLEM 2

# read in, inspect and prep data
path <- file.path(getwd() , "oral_ca.csv")
oral <- read.csv(path)
summary(oral)
head(oral)

#create array saying false/true (non-smoker / smoker)
smoker_nonSmoker = cigs == 0

# a)
# create new column indicating smoker vs non-smoker
oral$smoker <- with(oral, ifelse(cigs==0, "non-smoker", "smoker"))
attach(oral)
smoker_table <- table(ccstatus, smoker, dnn = c('ccstatus', 'smoker'))

# calculation of binomial mean
mean_common = (smoker_table["1", "non-smoker"] + smoker_table["1", "smoker"] ) / nrow(oral)
mean_smoker =  smoker_table["1", "smoker"] / (smoker_table["0", "smoker"] + smoker_table["1",
"smoker"] )
mean_non_smoker = smoker_table["1", "non-smoker"] / (smoker_table["0", "non-smoker"] +
smoker_table["1", "non-smoker"] )

# calculation of binomial variance: Var(X) = n*p*(1-p)
sd_common = sqrt(mean_common * (1 - mean_common) / nrow(oral))
sd_smoker = sqrt(mean_smoker * (1 - mean_smoker) / (smoker_table["0", "smoker"] +
smoker_table["1", "smoker"]))
sd_non_smoker = sqrt(mean_non_smoker * (1 - mean_non_smoker) / (smoker_table["0",
"non-smoker"] + smoker_table["1", "non-smoker"] ))

# b)
# compute the likelihood ratio statistics test
llik_1 = sum(dbinom(smoker_nonSmoker[ smoker_nonSmoker == TRUE], 1, mean_smoker, log = TRUE))
+
  sum(dbinom(smoker_nonSmoker[ smoker_nonSmoker == FALSE], 1, mean_non_smoker, log = TRUE))
llik_pi_2 = sum(dbinom(smoker_nonSmoker, 1, mean_common, log=TRUE))

w = 2 * (llik_1 - llik_2)
p.val = 1 - pchisq(w, df = 1)
p.val #answer shows zero

# c)
# create x and y columns
y = oral$ccstatus == 1
x = oral$cigs > 0
```

```r
#fit binomial model and print results
bin_mod = glm(y ~ x, family = 'binomial')
print(summary(bin_mod))

# get coefficients
beta = bin_mod$coefficients
print(beta)

pi_nonSmoker = exp(beta[1]) / (1 + exp(beta[1]))
pi_smoker = exp(beta[1] + beta[2]) / (1 + exp(beta[1] + beta[2]))

#calculate odds and log odds ratio
odds_nonSmoker = mean_non_smoker / (1 - mean_non_smoker)
odds_smoker = mean_smoker / (1 - mean_smoker)

log_odds_ratio = (odds_smoker / odds_nonSmoker)
log_odds_ratio

# d)
# create the x and y variables, but with a continuous x
x = cigs
y = ccstatus
continuous_mod = glm(y ~ x, family = 'binomial')
print(summary(continuous_mod))

#e)
#include all features in the model
total_mod = glm(y ~  oral$drinks + oral$cigs + oral$age + oral$sex, family = 'binomial')
print(summary(total_mod))

#f)
#dropping age
no_age_mod = glm(y ~  oral$drinks + oral$cigs + oral$sex, family = 'binomial')
print(summary(no_age_mod))

#g)
poly_drinks_mod = glm(y ~  poly(oral$drinks,2) + oral$cigs + oral$sex, family = 'binomial')
print(summary(poly_drinks_mod))

#h)
poly_cigs_mod = glm(y ~  oral$drinks + poly(oral$cigs,2) + oral$sex, family = 'binomial')
print(summary(poly_cigs_mod))
```