Correction GLM

Alix Zollinger (alix.zollinger@sib.swiss)
Bioinformatics Core Facility
Swiss Institute of Bioinformatics
Lausanne, Switzerland

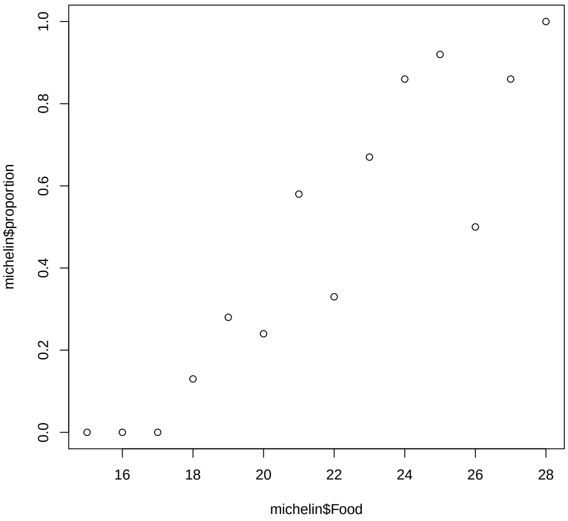
myDate

1 Michelin food

Question1

Start by graphically exploring the data

```
michelin <- read.delim("../MichelinFood.txt", header = TRUE, sep = "\t", as.is = TRUE)
michelin

##    Food InMichelin NotInMichelin mi proportion
## 1    15          0             1  1       0.00
## 2    16          0             1  1       0.00
## 3    17          0             8  8       0.00
## 4    18          2            13 15       0.13
## 5    19          5            13 18       0.28
## 6    20          8            25 33       0.24
## 7    21         15            11 26       0.58
## 8    22          4             8 12       0.33
## 9    23         12             6 18       0.67
## 10   24          6             1  7       0.86
## 11   25         11             1 12       0.92
## 12   26          1             1  2       0.50
## 13   27          6             1  7       0.86
## 14   28          4             0  4       1.00

plot(michelin$Food, michelin$proportion)
```



Question 2

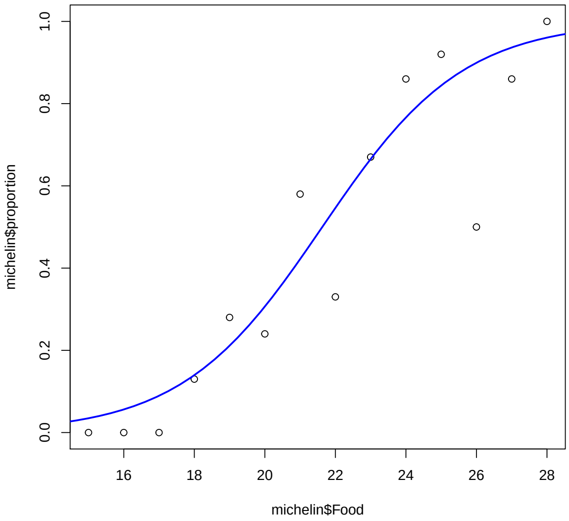
Fit a GLM using a binomial model for the response, using the food ranking as the predictor.

```
glm.mich <- glm(cbind(InMichelin, NotInMichelin) ~ Food, family = binomial(logit),
    data = michelin)
summary(glm.mich)

##
## Call:
## glm(formula = cbind(InMichelin, NotInMichelin) ~ Food, family = binomial(logit),
##     data = michelin)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4850  -0.7987  -0.1679   0.5913   1.5889
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -10.84154    1.86236  -5.821 5.84e-09 ***
## Food          0.50124    0.08768   5.717 1.08e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 61.427  on 13  degrees of freedom
## Residual deviance: 11.368  on 12  degrees of freedom
## AIC: 41.491
##
## Number of Fisher Scoring iterations: 4
```

Question 3

Predict the probabilities for a number of potential food rankings xnew, and plot a smooth function

```
# 3.
xnew <- data.frame(Food = seq(from = 14, to = 30, length.out = 50))
pred.prop <- predict(glm.mich, newdata = xnew, type = "response")
plot(michelin$Food, michelin$proportion)
lines(xnew$Food, pred.prop, col = "blue", lwd = 2)
```



Question 4

Check the model by looking at the residual deviance, other residuals and especially the quantile residuals

```
# 4.
summary(glm.mich)

##
## Call:
## glm(formula = cbind(InMichelin, NotInMichelin) ~ Food, family = binomial(logit),
##     data = michelin)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4850  -0.7987  -0.1679   0.5913   1.5889
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -10.84154    1.86236  -5.821 5.84e-09 ***
## Food          0.50124    0.08768   5.717 1.08e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 61.427  on 13  degrees of freedom
## Residual deviance: 11.368  on 12  degrees of freedom
## AIC: 41.491
##
## Number of Fisher Scoring iterations: 4
```

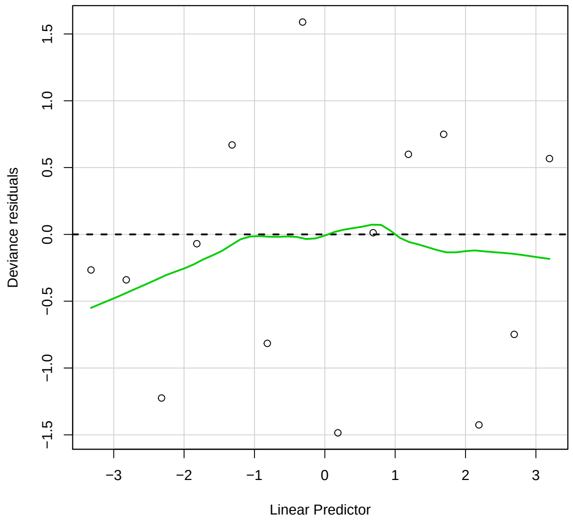
```
1 - pchisq(deviance(glm.mich), df.residual(glm.mich))
```

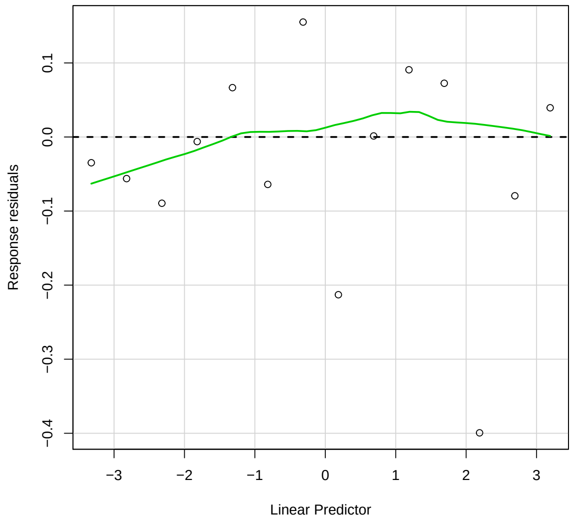
```
## [1] 0.4976357
```

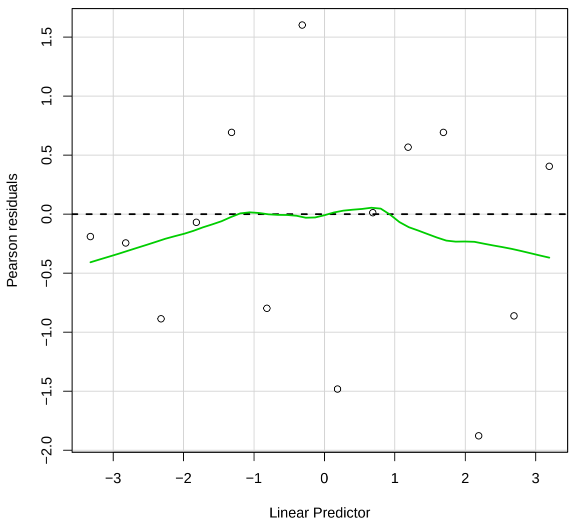
```
anova(glm.mich, test = "Chisq")
```

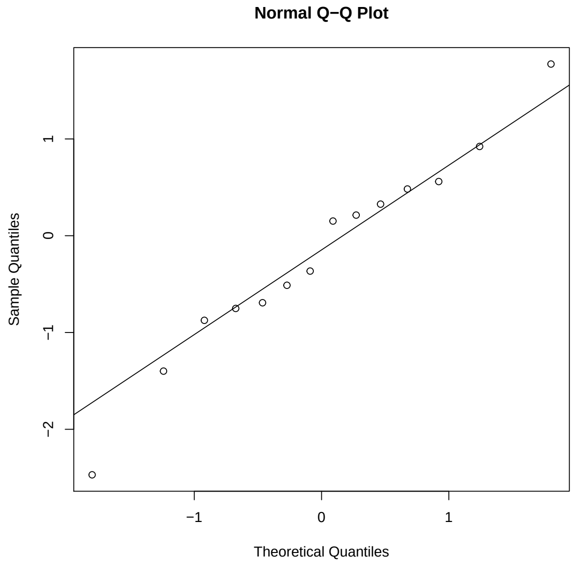
```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: cbind(InMichelin, NotInMichelin)
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                    13     61.427
## Food  1   50.059        12     11.368 1.492e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

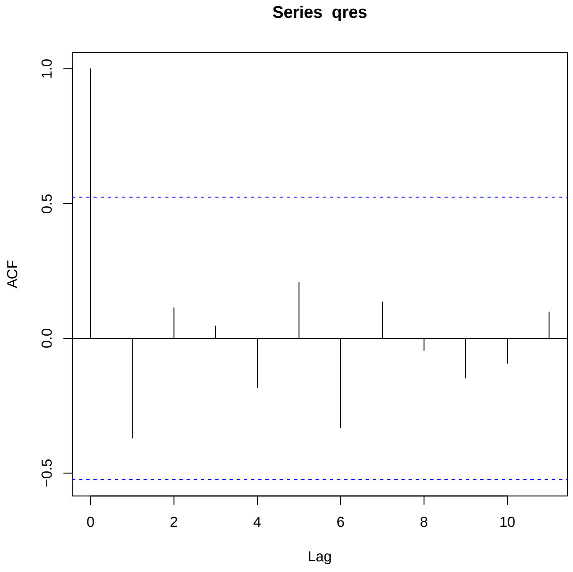
```
require(car)
residualPlot(glm.mich, type = "deviance")
residualPlot(glm.mich, type = "response")
residualPlot(glm.mich, type = "pearson")

library(statmod)
qres <- qresiduals(glm.mich)
qqnorm(qres)
qqline(qres)
acf(qres)
```


Deviance residuals
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−3
−2
−1
0
1
2
3
Linear Predictor


Response residuals
0.1
0.0
−0.1
−0.2
−0.3
−0.4
−3
−2
−1
0
1
2
3
Linear Predictor


Pearson residuals
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−3
−2
−1
0
1
2
3
Linear Predictor

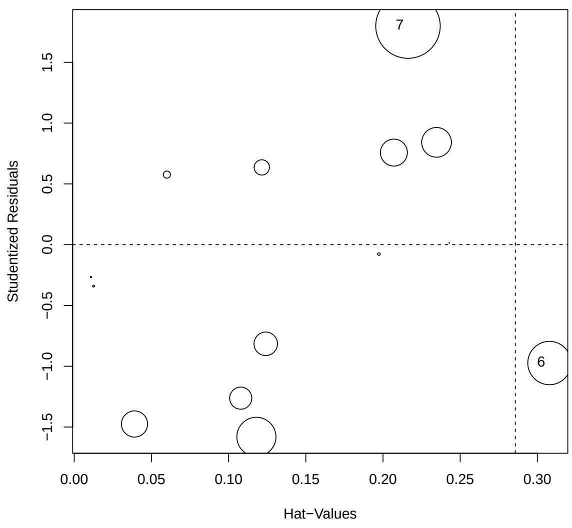

Normal Q−Q Plot
Sample Quantiles
1
0
−1
−2
−1
0
1
Theoretical Quantiles

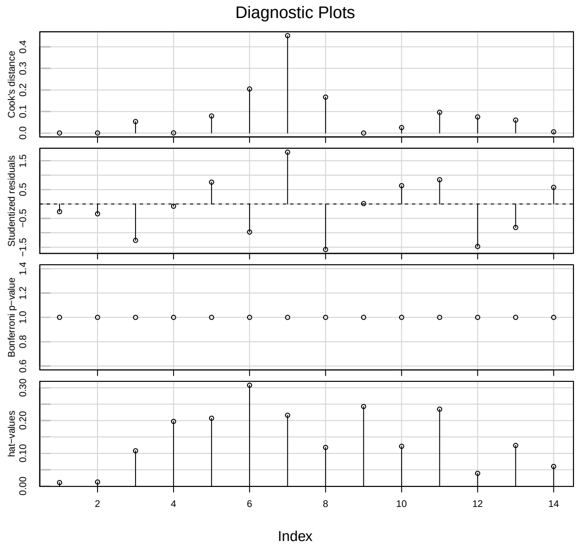

Series qres
ACF
1.0
0.5
0.0
−0.5
0
2
4
6
8
10
Lag

Question 5

Check the model for potential influencial observations.

```
influencePlot(glm.mich)

##       StudRes       Hat     CookD
## 6 -0.9736674 0.3078445 0.2044469
## 7  1.7979785 0.2161914 0.4516355

influenceIndexPlot(glm.mich)
```





2 Moth Death

Question1

Fit a GLM using the sex and dose as predictors. Include an interaction term in the model.

```
## 1.
moth <- data.frame(sex = rep(c("male", "female"), each = 6), dose = log2(rep(c(1,
    2, 4, 8, 16, 32), 2)), numdead = c(1, 4, 9, 13, 18, 20, 0, 2, 6, 10, 12,
    16))
moth$numalive <- 20 - moth$numdead
glm.moth <- glm(cbind(numalive, numdead) ~ sex * dose, data = moth, family = binomial)
# interaction is not significant
glm.moth <- glm(cbind(numalive, numdead) ~ sex + dose, data = moth, family = binomial)
summary(glm.moth)

##
## Call:
## glm(formula = cbind(numalive, numdead) ~ sex + dose, family = binomial,
##     data = moth)
##
## Deviance Residuals:
```

```
##       Min        1Q    Median        3Q       Max
## -1.42944  -0.48471   0.02225   0.65343   1.10540
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   3.4732     0.4685   7.413 1.23e-13 ***
## sexmale      -1.1007     0.3558  -3.093  0.00198 **
## dose         -1.0642     0.1311  -8.119 4.70e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 124.8756  on 11  degrees of freedom
## Residual deviance:   6.7571  on  9  degrees of freedom
## AIC: 42.867
##
## Number of Fisher Scoring iterations: 4
```

Question2

Does the model fit well? Perform an analysis of deviance

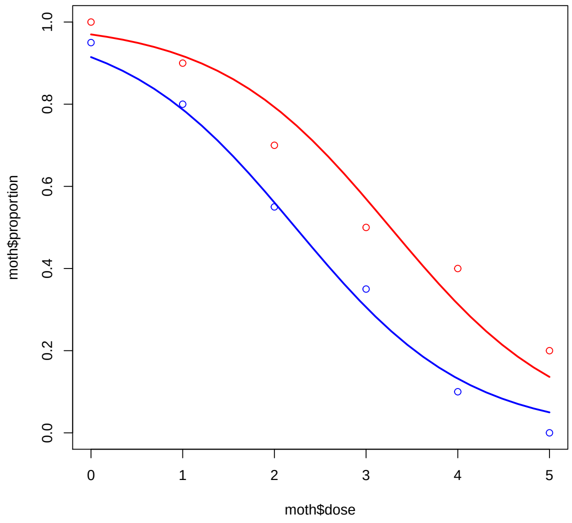
```
# 2.
1 - pchisq(deviance(glm.moth), df.residual(glm.moth))

## [1] 0.6623957

glm.null <- glm(cbind(numalive, numdead) ~ 1, data = moth, family = binomial)
anova(glm.null, glm.moth, test = "Chisq")

## Analysis of Deviance Table
##
## Model 1: cbind(numalive, numdead) ~ 1
## Model 2: cbind(numalive, numdead) ~ sex + dose
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1        11    124.876
## 2         9      6.757  2   118.12 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Question3

predict the probabilities for different doses and include a smooth line in the plots

```
# 3.
xnew <- data.frame(sex = rep(c("male", "female"), each = 30), dose = rep(seq(from = 0,
    to = 5, length.out = 30), 2))
pred.prop <- predict(glm.moth, newdata = xnew, type = "response")
moth$proportion <- moth$numalive/(moth$numdead + moth$numalive)
```

```
color <- moth$sex
levels(color) <- c("red", "blue")
plot(moth$proportion ~ moth$dose, col = as.character(color))
lines(xnew$dose[which(xnew$sex == "male")], pred.prop[which(xnew$sex == "male")],
    col = "blue", lwd = 2)
lines(xnew$dose[which(xnew$sex == "female")], pred.prop[which(xnew$sex == "female")],
    col = "red", lwd = 2)
```



3 Beetle data

Question 1

fit a logistic regression to the data using dose as a predictor

```
## 1
beetles <- data.frame(dose = c(1.6907, 1.7242, 1.7552, 1.7842, 1.8113, 1.8369,
    1.861, 1.8839), dead = c(6, 13, 18, 28, 52, 53, 61, 60), alive = c(51, 47,
    44, 28, 11, 6, 1, 0))
glm.beetles <- glm(cbind(alive, dead) ~ dose, beetles, family = "binomial")
```

Question 2

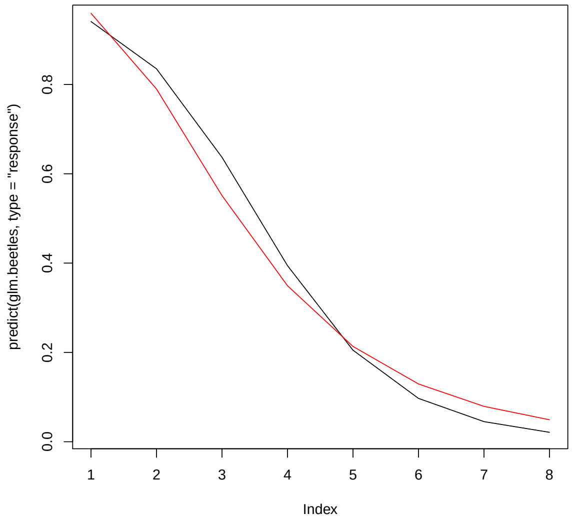
fit another logistic regression using the log-log link. Compare the two fits.

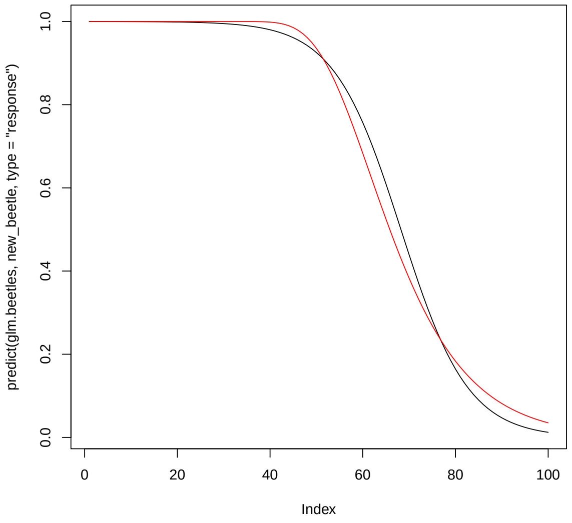
```
# 2.
glm.beetles_log <- glm(cbind(alive, dead) ~ dose, beetles, family = binomial(cloglog))
```

Question 3

Compare the prediction with each of the model.

```
# 3.

plot(predict(glm.beetles, type = "response"), type = "l")
lines(predict(glm.beetles_log, type = "response"), col = "red")

# or, on more observations
new_beetle <- data.frame(dose = seq(from = 1.5, to = 1.9, length.out = 100))

plot(predict(glm.beetles, new_beetle, type = "response"), type = "l")
lines(predict(glm.beetles_log, new_beetle, type = "response"), col = "red")
```





4 Pima data

Question 1

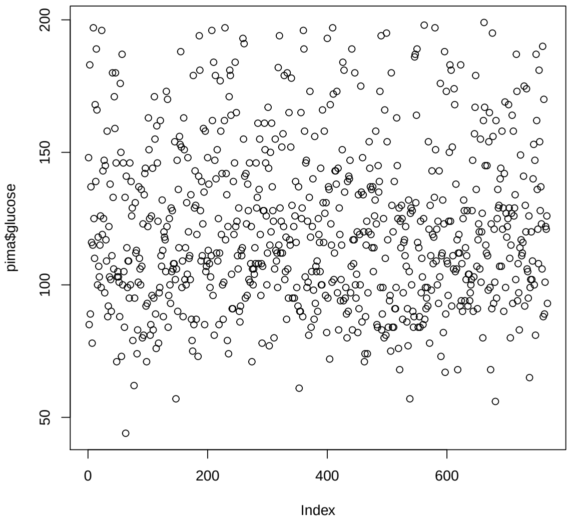
Perform simple graphical and numerical summaries of the data. Can you find any obvious irregularities in the data? If you do, take appropriate steps to correct the problems

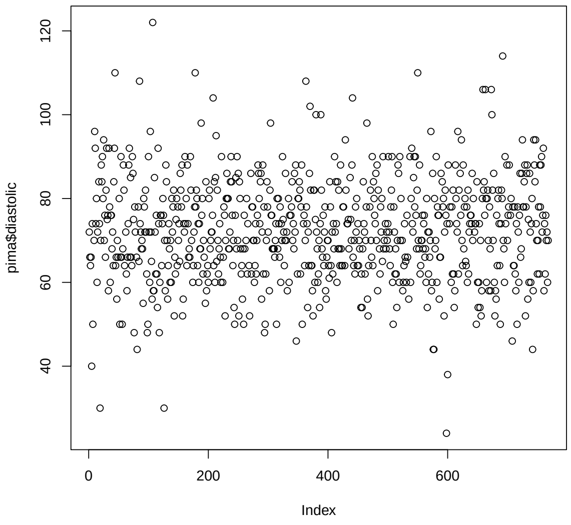
```
## 1.
library(faraway)
data("pima")
head(pima)

##   pregnant glucose diastolic triceps insulin  bmi diabetes age test
## 1        6     148        72      35       0 33.6    0.627  50    1
## 2        1      85        66      29       0 26.6    0.351  31    0
## 3        8     183        64       0       0 23.3    0.672  32    1
## 4        1      89        66      23      94 28.1    0.167  21    0
## 5        0     137        40      35     168 43.1    2.288  33    1
## 6        5     116        74       0       0 25.6    0.201  30    0

help(pima)
str(pima)
```

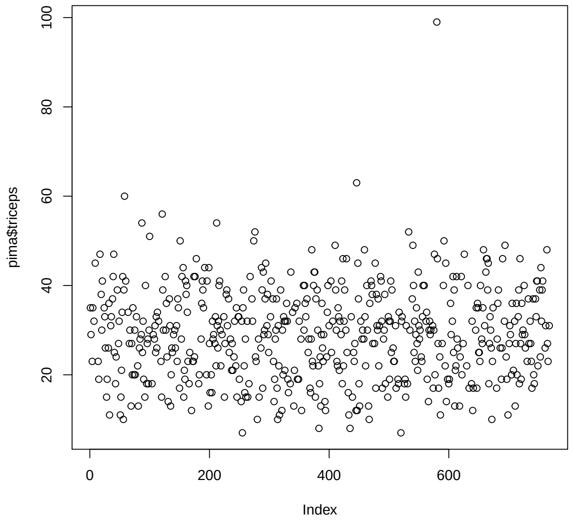
```
## 'data.frame': 768 obs. of  9 variables:
##  $ pregnant : int  6 1 8 1 0 5 3 10 2 8 ...
##  $ glucose  : int  148 85 183 89 137 116 78 115 197 125 ...
##  $ diastolic: int  72 66 64 66 40 74 50 0 70 96 ...
##  $ triceps  : int  35 29 0 23 35 0 32 0 45 0 ...
##  $ insulin  : int  0 0 0 94 168 0 88 0 543 0 ...
##  $ bmi      : num  33.6 26.6 23.3 28.1 43.1 25.6 31 35.3 30.5 0 ...
##  $ diabetes : num  0.627 0.351 0.672 0.167 2.288 ...
##  $ age      : int  50 31 32 21 33 30 26 29 53 54 ...
##  $ test     : int  1 0 1 0 1 0 1 0 1 1 ...

summary(pima)

##     pregnant         glucose        diastolic         triceps
##  Min.   : 0.000   Min.   :  0.0   Min.   :  0.00   Min.   : 0.00
##  1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 62.00   1st Qu.: 0.00
##  Median : 3.000   Median :117.0   Median : 72.00   Median :23.00
##  Mean   : 3.845   Mean   :120.9   Mean   : 69.11   Mean   :20.54
##  3rd Qu.: 6.000   3rd Qu.:140.2   3rd Qu.: 80.00   3rd Qu.:32.00
##  Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :99.00
##     insulin           bmi           diabetes           age
##  Min.   :  0.0   Min.   : 0.00   Min.   :0.0780   Min.   :21.00
##  1st Qu.:  0.0   1st Qu.:27.30   1st Qu.:0.2437   1st Qu.:24.00
##  Median : 30.5   Median :32.00   Median :0.3725   Median :29.00
##  Mean   : 79.8   Mean   :31.99   Mean   :0.4719   Mean   :33.24
##  3rd Qu.:127.2   3rd Qu.:36.60   3rd Qu.:0.6262   3rd Qu.:41.00
##  Max.   :846.0   Max.   :67.10   Max.   :2.4200   Max.   :81.00
##       test
##  Min.   :0.000
##  1st Qu.:0.000
##  Median :0.000
##  Mean   :0.349
##  3rd Qu.:1.000
##  Max.   :1.000

table(pima$pregnant)

##
##   0   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  17
## 111 135 103  75  68  57  50  45  38  28  24  11   9  10   2   1   1
```

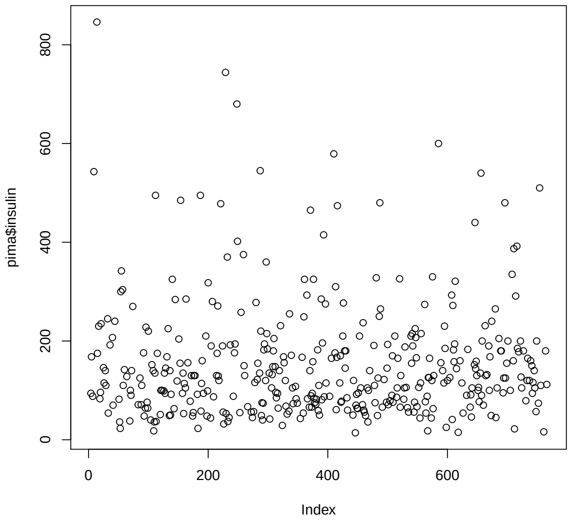
0 is likely to mean missing values so replace with NA also there are several potential outliers...

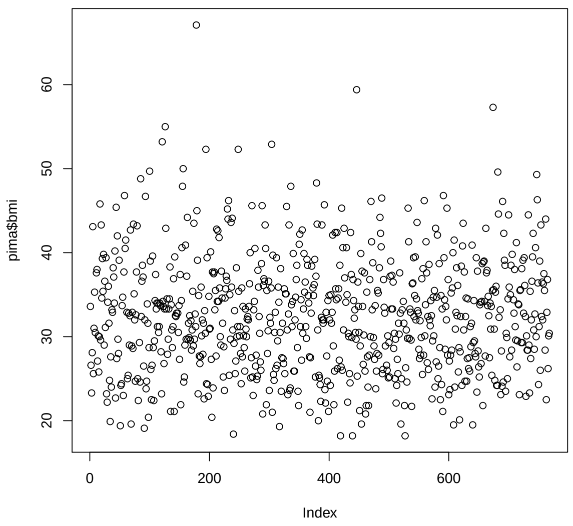
```
pima$glucose[which(pima$glucose == 0)] <- NA
pima$diastolic[which(pima$diastolic == 0)] <- NA
pima$triceps[which(pima$triceps == 0)] <- NA
pima$insulin[which(pima$insulin == 0)] <- NA
pima$bmi[pima$bmi == 0] <- NA

plot(pima$glucose)
plot(pima$diastolic)
plot(pima$triceps)
plot(pima$insulin)
plot(pima$bmi)
plot(pima$diabetes)
plot(pima$age)
# transform test to a factor
pima$test <- factor(pima$test)
levels(pima$test) <- c("negative", "positive")
table(pima$test)

##
## negative positive
##      500      268

boxplot(pima$diastolic ~ pima$test)
boxplot(pima$pregnant ~ pima$test)
boxplot(pima$glucose ~ pima$test)
boxplot(pima$triceps ~ pima$test)
boxplot(pima$insulin ~ pima$test)
boxplot(pima$bmi ~ pima$test)
boxplot(pima$diabetes ~ pima$test)
boxplot(pima$age ~ pima$test)
```

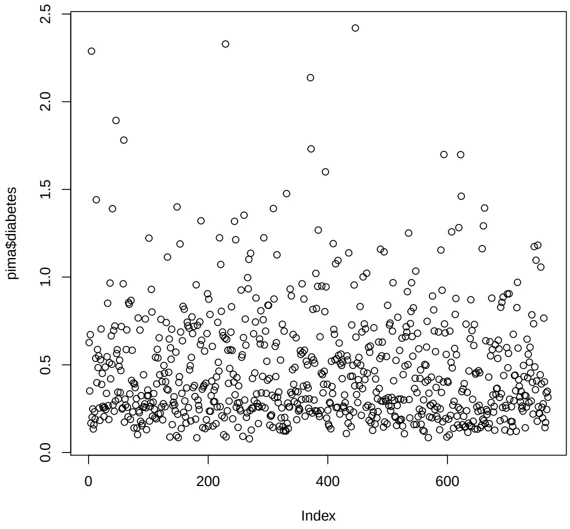


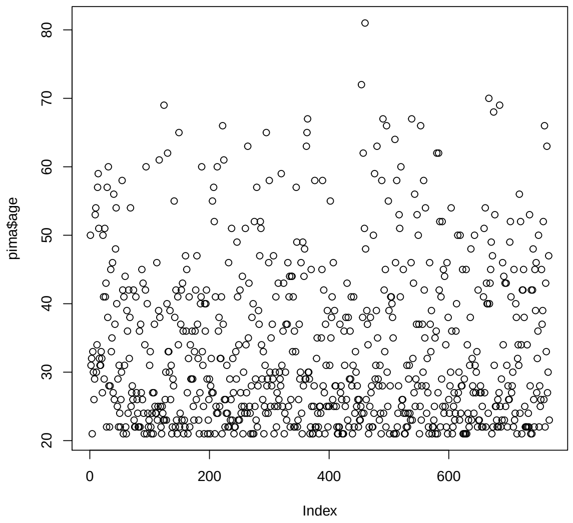


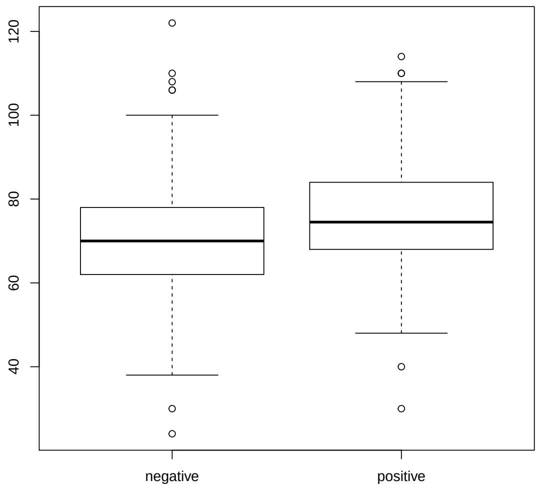

pima$triceps
100
80
60
40
20
0
200
400
600
Index

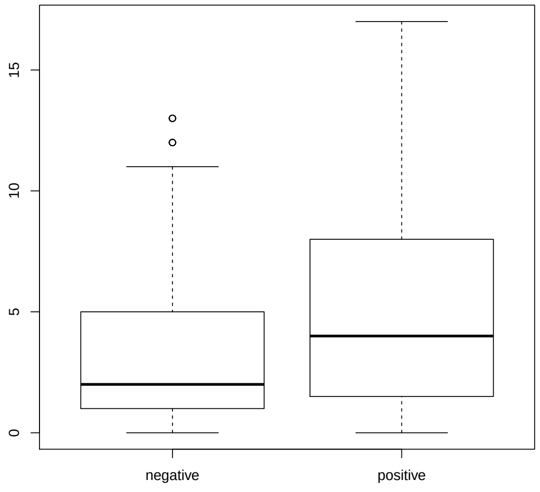

pima$insulin
800
600
400
200
0
0
200
400
600
Index

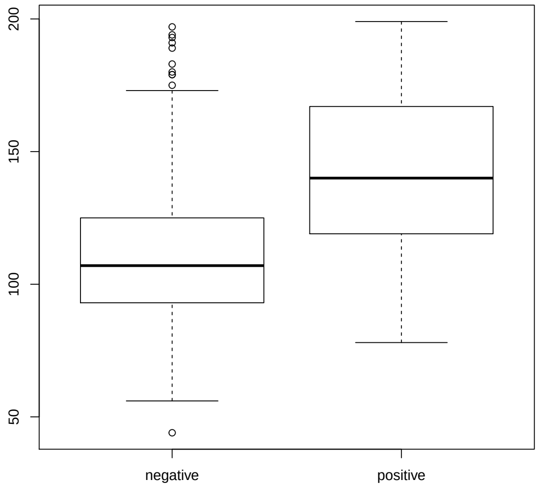

pima$bmi
60
50
40
30
20
0
200
400
600
Index

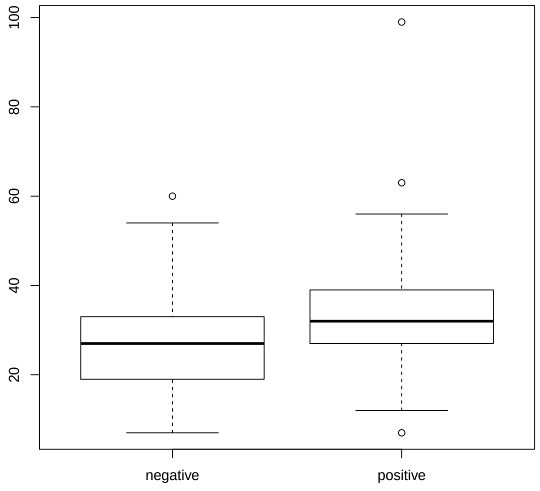

pima$diabetes
2.5
2.0
1.5
1.0
0.5
0.0
0
200
400
600
Index

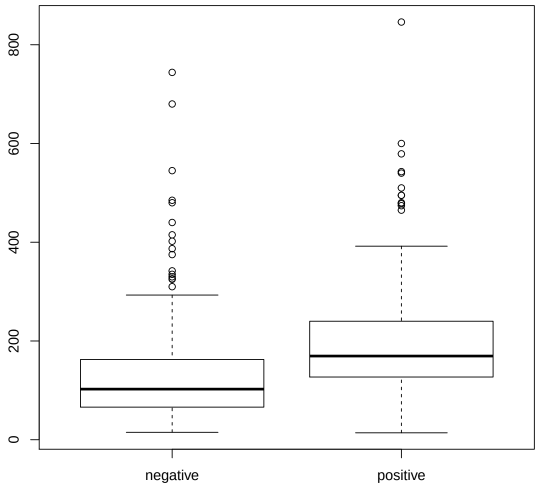

pima$age
80
70
60
50
40
30
20
0
200
400
600
Index

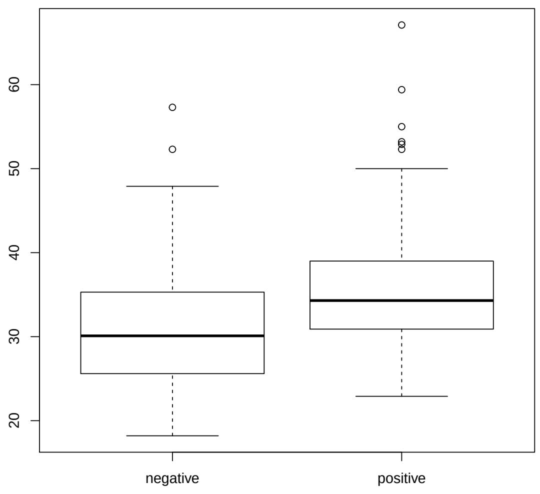

120
100
80
60
40
negative
positive

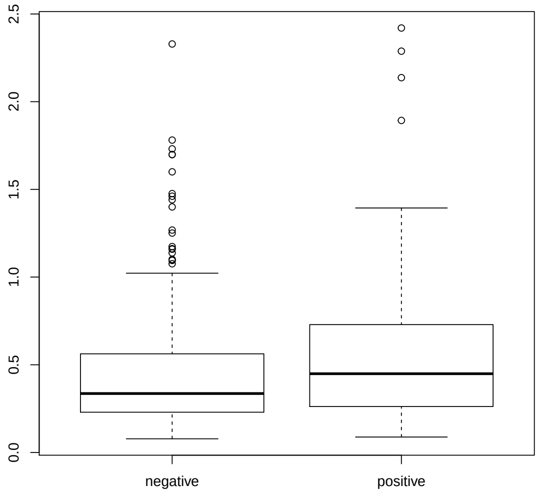

0
5
10
15
negative
positive

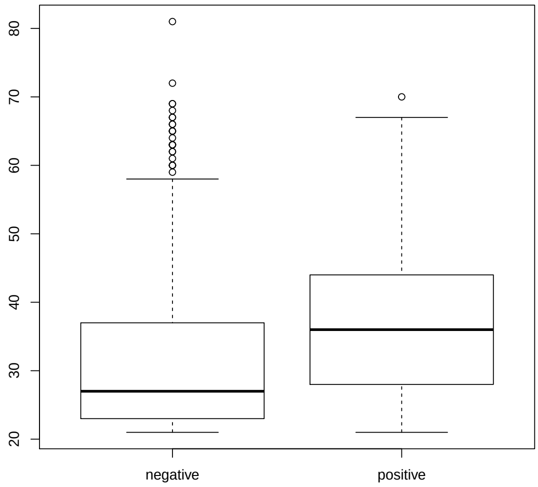

50
100
150
200
negative
positive


20
40
60
80
100
negative
positive


0
200
400
600
800
negative
positive


20
30
40
50
60
negative
positive


0.0
0.5
1.0
1.5
2.0
2.5
negative
positive



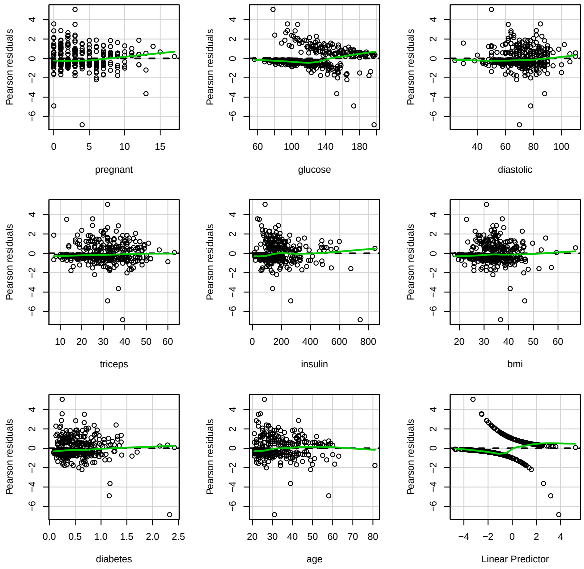
Question 2

Fit a model with the result of the diabetes test as the response and all the other variables as predictors. Can you tell whether this model fits the data?

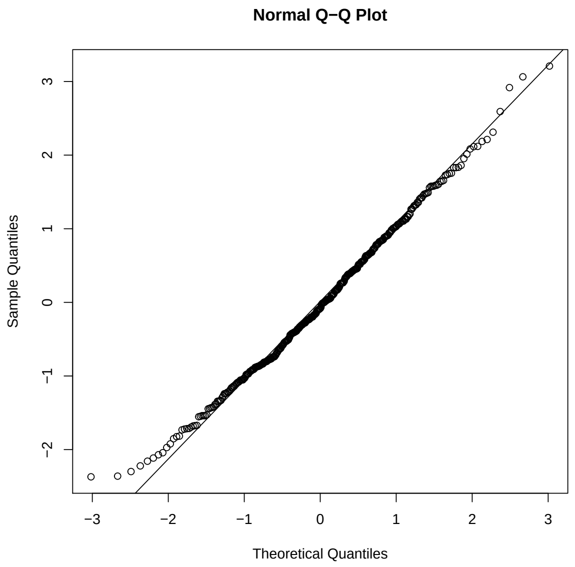
```
# 2.
glm_pima_full <- glm(test ~ ., pima, family = binomial)
summary(glm_pima_full)

##
## Call:
## glm(formula = test ~ ., family = binomial, data = pima)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7823  -0.6603  -0.3642   0.6409   2.5612
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.004e+01  1.218e+00  -8.246  < 2e-16 ***
## pregnant     8.216e-02  5.543e-02   1.482  0.13825
## glucose      3.827e-02  5.768e-03   6.635 3.24e-11 ***
## diastolic   -1.420e-03  1.183e-02  -0.120  0.90446
## triceps      1.122e-02  1.708e-02   0.657  0.51128
## insulin     -8.253e-04  1.306e-03  -0.632  0.52757
## bmi          7.054e-02  2.734e-02   2.580  0.00989 **
## diabetes     1.141e+00  4.274e-01   2.669  0.00760 **
## age          3.395e-02  1.838e-02   1.847  0.06474 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 498.10  on 391  degrees of freedom
## Residual deviance: 344.02  on 383  degrees of freedom
##   (376 observations deleted due to missingness)
## AIC: 362.02
##
## Number of Fisher Scoring iterations: 5
```

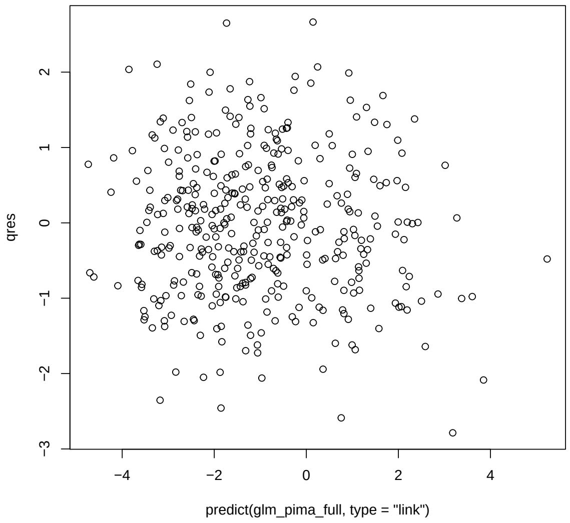
```
library(car)
residualPlots(glm_pima_full)
```

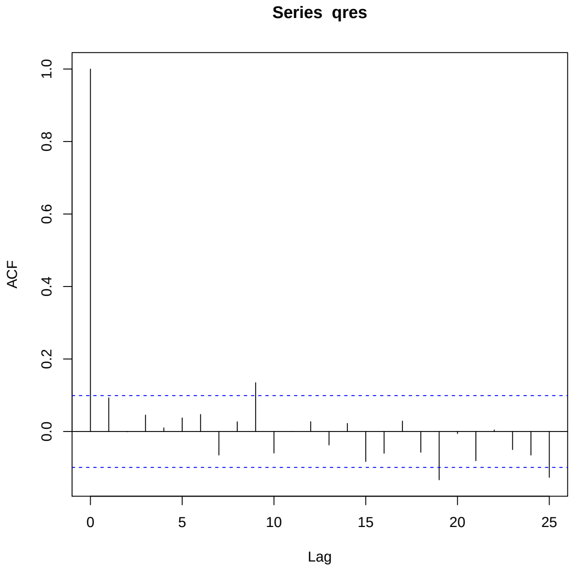
```
##           Test stat Pr(>|t|)
## pregnant      1.004    0.316
## glucose       0.000    0.985
## diastolic     0.765    0.382
## triceps       0.708    0.400
## insulin       2.661    0.103
## bmi           1.236    0.266
## diabetes      2.524    0.112
## age          10.143    0.001
```



```
library(statmod)
qqnorm(qresiduals(glm_pima_full))
qqline(qresiduals(glm_pima_full))
qres <- qresiduals(glm_pima_full)
plot(qres ~ predict(glm_pima_full, type = "link"))
acf(qres)
```







The fit is quite good. The residuals are good and the uniform residuals pass all the checks. There are many non-significant variables so we can remove them to have a better fit (parsimony principle!)

Question 3

What is the difference in the odds of testing positive for diabetes for a woman with a BMI at the first quartile compared with a woman at the third quartile, assuming that all other factors held constant? Give a confidence interval for this difference.

```
summary(glm_pima_full)

##
## Call:
## glm(formula = test ~ ., family = binomial, data = pima)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7823  -0.6603  -0.3642   0.6409   2.5612
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.004e+01  1.218e+00  -8.246  < 2e-16 ***
## pregnant     8.216e-02  5.543e-02   1.482  0.13825
## glucose      3.827e-02  5.768e-03   6.635 3.24e-11 ***
## diastolic   -1.420e-03  1.183e-02  -0.120  0.90446
## triceps      1.122e-02  1.708e-02   0.657  0.51128
## insulin     -8.253e-04  1.306e-03  -0.632  0.52757
## bmi          7.054e-02  2.734e-02   2.580  0.00989 **
## diabetes     1.141e+00  4.274e-01   2.669  0.00760 **
## age          3.395e-02  1.838e-02   1.847  0.06474 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 498.10  on 391  degrees of freedom
## Residual deviance: 344.02  on 383  degrees of freedom
##   (376 observations deleted due to missingness)
## AIC: 362.02
##
## Number of Fisher Scoring iterations: 5
```

The estimate for bmi (0.07) gives the log odds. So an increase on bmi by 1 unit increases the log odds by 0.07 . To get the odds, we use

```
exp(coefficients(glm_pima_full)["bmi"])

##      bmi
## 1.073085
```

so the odds of getting a positive test is increased by a factor 1.073 and the probability of having a positive test is

```
exp(coefficients(glm_pima_full)["bmi"])/(1 + exp(coefficients(glm_pima_full)["bmi"]))

##       bmi
## 0.5176271
```

let's compute the quartiles of bmi

```
diff_bmi <- with(pima, diff(quantile(bmi, prob = c(0.25, 0.75), na.rm = TRUE)))
logodds_diff_bmi <- diff_bmi * coefficients(glm_pima_full)["bmi"]
odds_bmi <- exp(logodds_diff_bmi)

# CI for odds of the difference
exp(confint(glm_pima_full)["bmi", ] * diff_bmi)

##    2.5 %   97.5 %
## 1.174445 3.128715
```

Question 4

Do women who test positive have higher diastolic blood pressures? Is the diastolic blood pressure signifcant in the regression model? Explain the distinction between the two questions and discuss why the answers are only apparently contradictory.

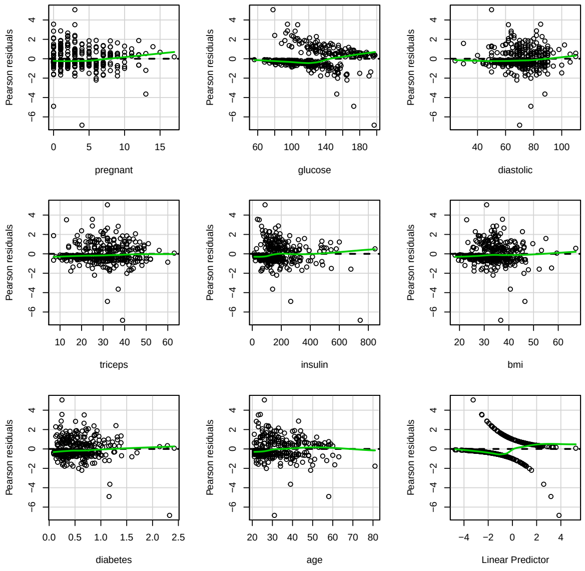
```
# 4. confonding factors
summary(glm_pima_full)

##
## Call:
## glm(formula = test ~ ., family = binomial, data = pima)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7823  -0.6603  -0.3642   0.6409   2.5612
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.004e+01  1.218e+00  -8.246  < 2e-16 ***
## pregnant     8.216e-02  5.543e-02   1.482  0.13825
## glucose      3.827e-02  5.768e-03   6.635 3.24e-11 ***
## diastolic   -1.420e-03  1.183e-02  -0.120  0.90446
## triceps      1.122e-02  1.708e-02   0.657  0.51128
## insulin     -8.253e-04  1.306e-03  -0.632  0.52757
## bmi          7.054e-02  2.734e-02   2.580  0.00989 **
## diabetes     1.141e+00  4.274e-01   2.669  0.00760 **
## age          3.395e-02  1.838e-02   1.847  0.06474 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 498.10  on 391  degrees of freedom
## Residual deviance: 344.02  on 383  degrees of freedom
##   (376 observations deleted due to missingness)
## AIC: 362.02
##
## Number of Fisher Scoring iterations: 5

summary(lm(diastolic ~ test, pima))

##
## Call:
## lm(formula = diastolic ~ test, data = pima)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -46.877  -7.321  -0.877   7.123  51.123
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    70.8773     0.5567 127.321  < 2e-16 ***
## testpositive    4.4441     0.9494   4.681 3.41e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.21 on 731 degrees of freedom
##   (35 observations deleted due to missingness)
## Multiple R-squared:  0.0291,Adjusted R-squared:  0.02777
## F-statistic: 21.91 on 1 and 731 DF,  p-value: 3.405e-06
```

diastolic blood pressure is not significant in the model.
Women with positive test tend to have higher diastolic blood pressure. However in this model the test factor may be confonded with another factor.

Question 5

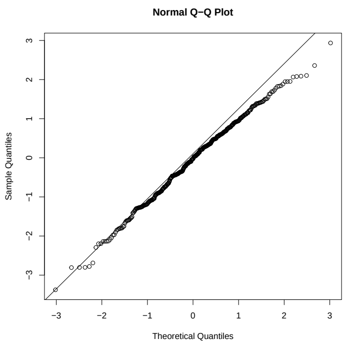
Perform diagnostics on the regression model, reporting any potential violations and any suggested improvements to the model Check the residuals

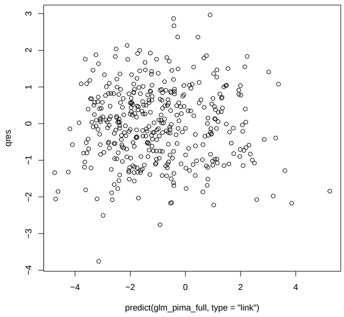
```
# 5.
library(car)
library(statmod)
residualPlots(glm_pima_full)

##            Test stat Pr(>|t|)
## pregnant       1.004    0.316
## glucose        0.000    0.985
## diastolic      0.765    0.382
## triceps        0.708    0.400
## insulin        2.661    0.103
## bmi            1.236    0.266
## diabetes       2.524    0.112
## age           10.143    0.001
```

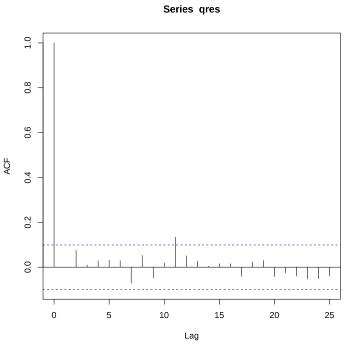


Check the quantile residuals

```
qqnorm(qresiduals(glm_pima_full))
qqline(qresiduals(glm_pima_full))
qres <- qresiduals(glm_pima_full)
plot(qres ~ predict(glm_pima_full, type = "link"))
acf(qres)
```

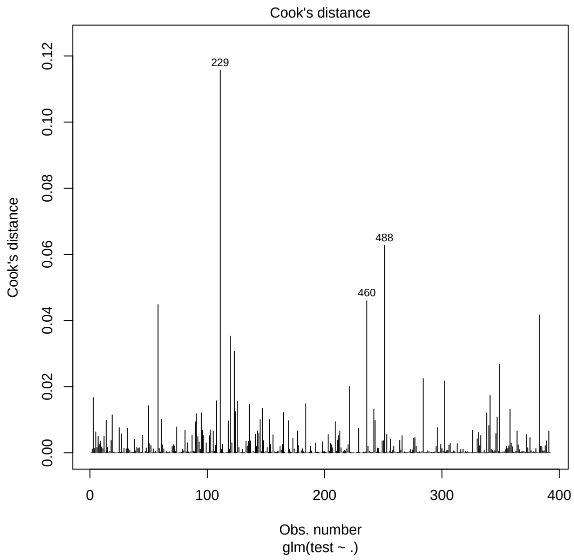


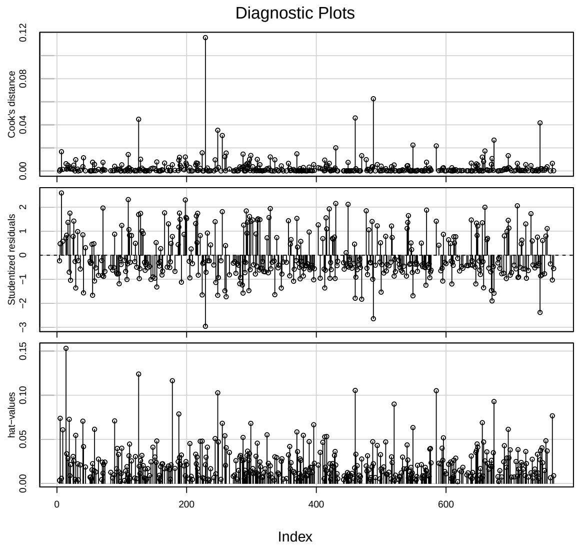




Check influence points

```
head(sort(cooks.distance(glm_pima_full), decreasing = TRUE))

##        229        488        460        126        745        248
## 0.11565136 0.06264414 0.04596353 0.04483756 0.04170002 0.03529009

plot(glm_pima_full, which = 4)
influenceIndexPlot(glm_pima_full, vars = c("Cook", "Studentized", "hat"))
```





check observations with large cooks distance

Question 6

Predict the outcome for a woman with the following predictor values:

```
new_pima <- data.frame(pregnant = 1, glucose = 99, diastolic = 64, triceps = 22,
    insulin = 76, bmi = 27, diabetes = 0.25, age = 25)
```

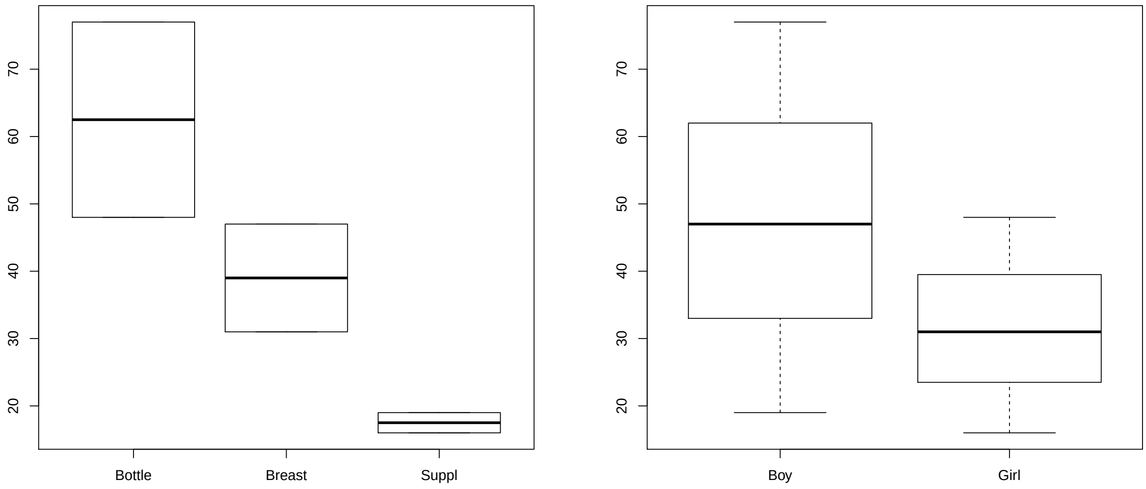
```
# 6.
new_pima <- data.frame(pregnant = 1, glucose = 99, diastolic = 64, triceps = 22,
    insulin = 76, bmi = 27, diabetes = 0.25, age = 25)
pred_prob <- predict(glm_pima_full, newdata = new_pima, type = "response", se = TRUE)
pred_prob$fit + 1.96 * pred_prob$se.fit

##          1
## 0.07298286

pred_logit <- predict(glm_pima_full, newdata = new_pima, se = TRUE)
ilogit(pred_logit$fit)  # gives the probability

##          1
## 0.04573331

# CI on the proba scale
ilogit(c(pred_logit$fit - 1.96 * pred_logit$se.fit, (pred_logit$fit + 1.96 *
    pred_logit$se.fit)))

##          1          1
## 0.02502570 0.08213208
```

5 Baby food data

Question 1

Explore the data

```
# 1.
library(faraway)
data(babyfood)
str(babyfood)

## 'data.frame': 6 obs. of  4 variables:
##  $ disease   : num  77 19 47 48 16 31
##  $ nondisease: num  381 128 447 336 111 433
##  $ sex       : Factor w/ 2 levels "Boy","Girl": 1 1 1 2 2 2
##  $ food      : Factor w/ 3 levels "Bottle","Breast",..: 1 3 2 1 3 2

summary(babyfood)

##     disease        nondisease      sex        food
##  Min.   :16.00   Min.   :111.0   Boy :3   Bottle:2
##  1st Qu.:22.00   1st Qu.:180.0   Girl:3   Breast:2
##  Median :39.00   Median :358.5            Suppl :2
##  Mean   :39.67   Mean   :306.0
##  3rd Qu.:47.75   3rd Qu.:420.0
##  Max.   :77.00   Max.   :447.0

boxplot(disease ˜ food, babyfood)
boxplot(disease ˜ sex, babyfood)
```



Question 2

What are the proportions of Boys/Girls in the different food categories?

```
# 2.
xtabs(disease/(disease + nondisease) ~ sex + food, babyfood)

##       food
## sex        Bottle     Breast      Suppl
##   Boy  0.16812227 0.09514170 0.12925170
##   Girl 0.12500000 0.06681034 0.12598425
```

Question 3

Fit a logistic regression to explain the probability of disease by sex and food.

```
# 3.
mdl <- glm(cbind(disease, nondisease) ~ sex + food, family = binomial, babyfood)
summary(mdl)

##
## Call:
## glm(formula = cbind(disease, nondisease) ~ sex + food, family = binomial,
##     data = babyfood)
##
## Deviance Residuals:
##       1        2        3        4        5        6
##  0.1096  -0.5052   0.1922  -0.1342   0.5896  -0.2284
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.6127     0.1124 -14.347  < 2e-16 ***
## sexGirl      -0.3126     0.1410  -2.216   0.0267 *
## foodBreast   -0.6693     0.1530  -4.374 1.22e-05 ***
## foodSuppl    -0.1725     0.2056  -0.839   0.4013
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 26.37529  on 5  degrees of freedom
## Residual deviance:  0.72192  on 2  degrees of freedom
## AIC: 40.24
##
## Number of Fisher Scoring iterations: 4
```

Question 4

What is the impact of breast feeding on the odds of respiratory disease compared to bottle feeding? Give a confidence interval for this value.

```
# 4.
exp(coefficients(mdl)[grep("food", names(coefficients(mdl)))])

## foodBreast  foodSuppl
##  0.5120696  0.8415226

# breast feeding reduces the odds of respiratory desease to 51% of that for
# bottle feeding.  CI
breast_food <- coefficients(summary(mdl))["foodBreast", ]
exp(c(breast_food["Estimate"] - 1.96 * breast_food["Std. Error"], breast_food["Estimate"] +
    1.96 * breast_food["Std. Error"]))

##  Estimate  Estimate
## 0.3793918 0.6911465
```

6 dvisits data

Question 1

Explore the dataset

```
library(faraway)
data("dvisits")

# 1.
str(dvisits)

## 'data.frame': 5190 obs. of  19 variables:
##  $ sex     : int  1 1 0 0 0 1 1 1 1 0 ...
##  $ age     : num  0.19 0.19 0.19 0.19 0.19 0.19 0.19 0.19 0.19 0.19 ...
##  $ agesq   : num  0.0361 0.0361 0.0361 0.0361 0.0361 0.0361 0.0361 0.0361 0.0361 0.0361
##  $ income  : num  0.55 0.45 0.9 0.15 0.45 0.35 0.55 0.15 0.65 0.15 ...
##  $ levyplus: int  1 1 0 0 0 0 0 0 1 1 ...
##  $ freepoor: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ freerepa: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ illness : int  1 1 3 1 2 5 4 3 2 1 ...
##  $ actdays : int  4 2 0 0 5 1 0 0 0 0 ...
##  $ hscore  : int  1 1 0 0 1 9 2 6 5 0 ...
##  $ chcond1 : int  0 0 0 0 1 1 0 0 0 0 ...
##  $ chcond2 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ doctorco: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ nondocco: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ hospadmi: int  0 0 1 0 0 0 0 0 0 0 ...
##  $ hospdays: int  0 0 4 0 0 0 0 0 0 0 ...
##  $ medicine: int  1 2 2 0 3 1 0 1 1 1 ...
##  $ prescrib: int  1 1 1 0 1 1 0 1 0 1 ...
##  $ nonpresc: int  0 1 1 0 2 0 0 0 1 0 ...
```

```
summary(dvisits)
```

```
##       sex              age             agesq            income
##  Min.   :0.0000   Min.   :0.1900   Min.   :0.0361   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.2200   1st Qu.:0.0484   1st Qu.:0.2500
##  Median :1.0000   Median :0.3200   Median :0.1024   Median :0.5500
##  Mean   :0.5206   Mean   :0.4064   Mean   :0.2071   Mean   :0.5832
##  3rd Qu.:1.0000   3rd Qu.:0.6200   3rd Qu.:0.3844   3rd Qu.:0.9000
##  Max.   :1.0000   Max.   :0.7200   Max.   :0.5184   Max.   :1.5000
##     levyplus         freepoor          freerepa         illness
##  Min.   :0.0000   Min.   :0.00000   Min.   :0.0000   Min.   :0.000
##  1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.000
##  Median :0.0000   Median :0.00000   Median :0.0000   Median :1.000
##  Mean   :0.4428   Mean   :0.04277   Mean   :0.2102   Mean   :1.432
##  3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:0.0000   3rd Qu.:2.000
##  Max.   :1.0000   Max.   :1.00000   Max.   :1.0000   Max.   :5.000
##     actdays            hscore           chcond1          chcond2
##  Min.   : 0.0000   Min.   : 0.000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.: 0.0000   1st Qu.: 0.000   1st Qu.:0.0000   1st Qu.:0.0000
##  Median : 0.0000   Median : 0.000   Median :0.0000   Median :0.0000
##  Mean   : 0.8619   Mean   : 1.218   Mean   :0.4031   Mean   :0.1166
##  3rd Qu.: 0.0000   3rd Qu.: 2.000   3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :14.0000   Max.   :12.000   Max.   :1.0000   Max.   :1.0000
##     doctorco         nondocco          hospadmi         hospdays
##  Min.   :0.0000   Min.   : 0.0000   Min.   :0.0000   Min.   : 0.000
##  1st Qu.:0.0000   1st Qu.: 0.0000   1st Qu.:0.0000   1st Qu.: 0.000
##  Median :0.0000   Median : 0.0000   Median :0.0000   Median : 0.000
##  Mean   :0.3017   Mean   : 0.2146   Mean   :0.1736   Mean   : 1.334
##  3rd Qu.:0.0000   3rd Qu.: 0.0000   3rd Qu.:0.0000   3rd Qu.: 0.000
##  Max.   :9.0000   Max.   :11.0000   Max.   :5.0000   Max.   :80.000
##     medicine        prescrib         nonpresc
##  Min.   :0.000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.000   1st Qu.:0.0000   1st Qu.:0.0000
##  Median :1.000   Median :0.0000   Median :0.0000
##  Mean   :1.218   Mean   :0.8626   Mean   :0.3557
##   3rd Qu.:2.000   3rd Qu.:1.0000   3rd Qu.:1.0000  
##   Max.   :8.000   Max.   :8.0000   Max.   :8.0000  

dvisits$sex <- factor(dvisits$sex)
levels(dvisits$sex) <- c("male", "female")

dvisits$levyplus <- factor(dvisits$levyplus)
levels(dvisits$levyplus) <- c("no", "private")

dvisits$freepoor <- factor(dvisits$freepoor)
levels(dvisits$freepoor) <- c("nofreepoor", "freepoor")

dvisits$freerepa <- factor(dvisits$freerepa)
levels(dvisits$freerepa) <- c("nofreerepa", "freerepa")

dvisits$chcond1 <- factor(dvisits$chcond1)
levels(dvisits$freerepa) <- c("notchronic", "chronic")

dvisits$chcond2 <- factor(dvisits$chcond2)
levels(dvisits$freerepa) <- c("notchronic", "chronic_limited")
```

Question 2

Build a Poisson regression model with doctorco as the response and sex, age, agesq, income, levyplus, freepoor, freerepa, illness, actdays, hscore, chcond1 and chcond2 as possible predictor variables. Considering the deviance of this model, does this model fit the data?

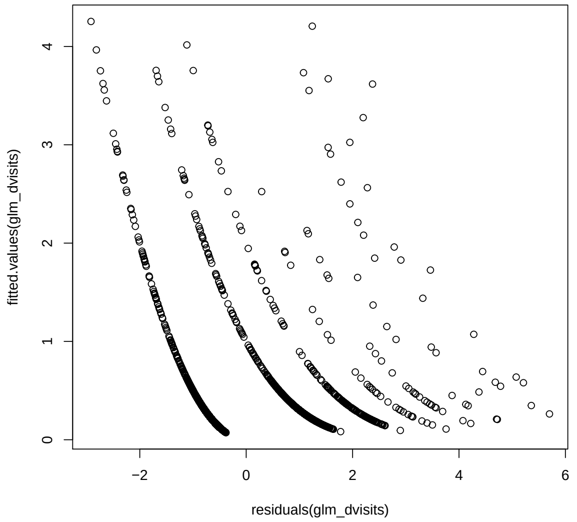
```
# 2.
glm_dvisits <- glm(doctorco ~ sex + age + agesq + income + levyplus + freepoor +
    freerepa + illness + actdays + hscore + chcond1 + chcond2, data = dvisits,
    family = poisson)
summary(glm_dvisits)

## 
## Call:
## glm(formula = doctorco ~ sex + age + agesq + income + levyplus + 
##     freepoor + freerepa + illness + actdays + hscore + chcond1 + 
##     chcond2, family = poisson, data = dvisits)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -2.9170  -0.6862  -0.5743  -0.4839   5.7005  
## 
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)    
## (Intercept)         -2.223848   0.189816 -11.716   <2e-16 ***
## sexfemale            0.156882   0.056137   2.795   0.0052 ** 
## age                  1.056299   1.000780   1.055   0.2912    
## agesq               -0.848704   1.077784  -0.787   0.4310    
## income              -0.205321   0.088379  -2.323   0.0202 *  
## levyplusprivate      0.123185   0.071640   1.720   0.0855 .  
## freepoorfreepoor        -0.440061   0.179811  -2.447   0.0144 *
## freerepachronic_limited  0.079798   0.092060   0.867   0.3860
## illness                  0.186948   0.018281  10.227   <2e-16 ***
## actdays                  0.126846   0.005034  25.198   <2e-16 ***
## hscore                   0.030081   0.010099   2.979   0.0029 **
## chcond11                 0.114085   0.066640   1.712   0.0869 .
## chcond21                 0.141158   0.083145   1.698   0.0896 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 5634.8  on 5189  degrees of freedom
## Residual deviance: 4379.5  on 5177  degrees of freedom
## AIC: 6737.1
##
## Number of Fisher Scoring iterations: 6
```

deviance seems not too bad (same range as the df)

Question 3

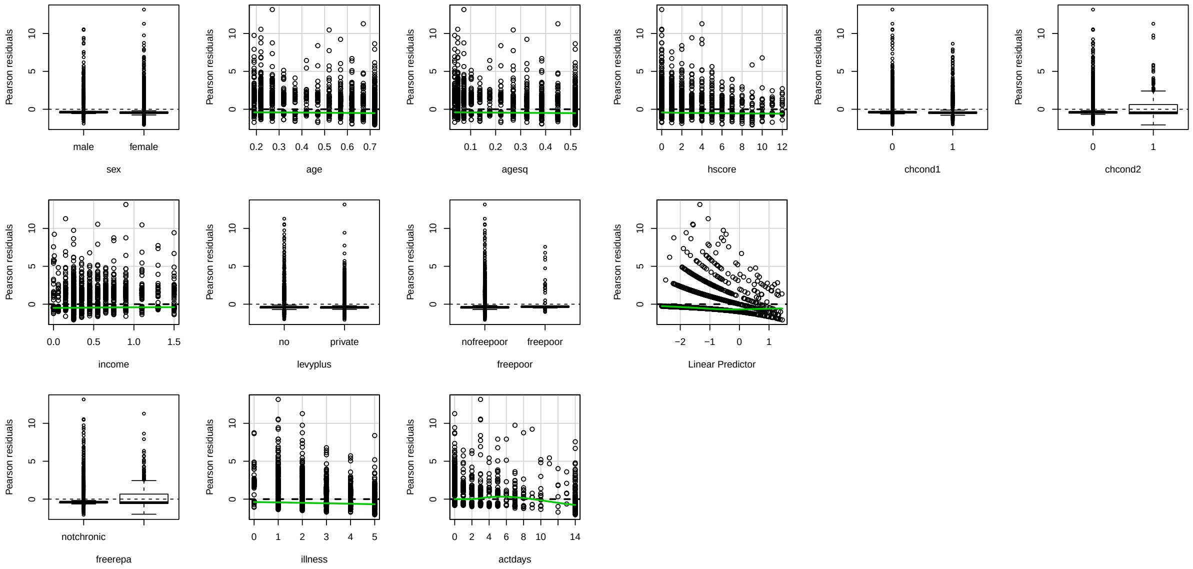
Plot the residuals and the fitted values. Why are there lines of observations on the plot?

```
# 3.
plot(residuals(glm_dvisits), fitted.values(glm_dvisits))
table(dvisits$doctorco)

##
##    0    1    2    3    4    5    6    7    8    9
## 4141  782  174   30   24    9   12   12    5    1
```



we have 9 levels of response and so the residuals also follow that. each line corresponds to a different possible value

```
residualPlots(glm_dvisits)

##          Test stat Pr(>|t|)
## sex             NA       NA
## age          0.000    1.000
## agesq        0.505    0.477
## income       5.881    0.015
## levyplus        NA       NA
## freepoor        NA       NA
## freerepa        NA       NA
## illness     62.407    0.000
## actdays    174.913    0.000
## hscore       1.299    0.254
## chcond1         NA       NA
## chcond2         NA       NA
```



Question 4

Use backward eliminiation with a critical p-value of 5% to reduce the model as much as possible. Report your model.

```
# 4. we remove each time the least significant variable
glm_dvisits <- update(glm_dvisits, . ~ . - agesq, data = dvisits)
glm_dvisits <- update(glm_dvisits, . ~ . - freerepa, data = dvisits)
glm_dvisits <- update(glm_dvisits, . ~ . - levyplus, data = dvisits)
glm_dvisits <- update(glm_dvisits, . ~ . - chcond1, data = dvisits)
glm_dvisits <- update(glm_dvisits, . ~ . - chcond2, data = dvisits)
summary(glm_dvisits)

##
## Call:
## glm(formula = doctorco ~ sex + age + income + freepoor + illness +
##     actdays + hscore, family = poisson, data = dvisits)
##
```

```
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -2.9258  -0.6829  -0.5752  -0.4945   5.6960  
## 
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)    
## (Intercept)      -2.051963   0.099522 -20.618  < 2e-16 ***
## sexfemale         0.175529   0.055433   3.167  0.00154 ** 
## age               0.433532   0.137140   3.161  0.00157 ** 
## income           -0.171053   0.081926  -2.088  0.03681 *  
## freepoorfreepoor -0.496325   0.175304  -2.831  0.00464 ** 
## illness           0.196008   0.017585  11.146  < 2e-16 ***
## actdays           0.127793   0.004899  26.088  < 2e-16 ***
## hscore            0.032433   0.009938   3.263  0.00110 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for poisson family taken to be 1)
## 
##     Null deviance: 5634.8  on 5189  degrees of freedom
## Residual deviance: 4388.1  on 5182  degrees of freedom
## AIC: 6735.7
## 
## Number of Fisher Scoring iterations: 6
```

Question 5

What kind of person would be predicted to visit the doctor the most under your selected model? Under the last model, the person who is the most probable to visit the doctor is a female, old, low income, freepoor, with illnesses in the past 2 weeks, with reduced activity in the past 2 weeks and with high score to Goldberg's questionnaire.

Question 6

For the last person in the dataset, compute the predicted probability distribution for their visits to the doctor, i.e., give the probability they visit 0,1,2,... times.

```
# 6.
new.data <- tail(dvisits, 1)
mu <- predict(glm_dvisits, newdata = new.data, type = "response")
mu <- exp(predict(glm_dvisits, newdata = new.data))
sapply(seq(0, 9, 1), function(x) dpois(x, lambda = mu))

##  [1] 8.451821e-01 1.421623e-01 1.195608e-02 6.703505e-04 2.818878e-05
##  [6] 9.482888e-07 2.658420e-08 6.387927e-10 1.343087e-11 2.510129e-13
```

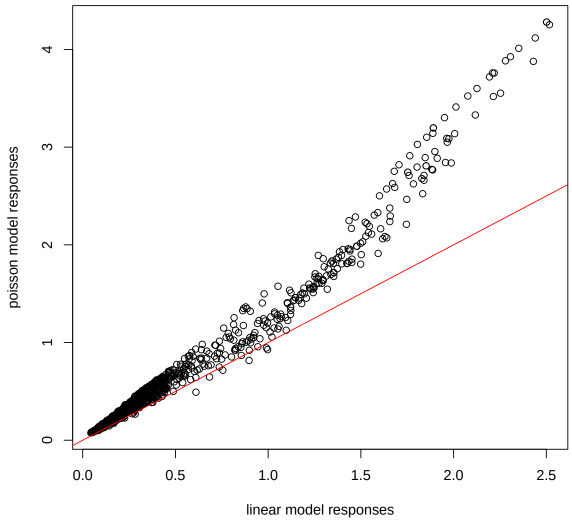
Question 7

fit a comparable (Gaussian) linear model and graphically compare the fits. Describe how they differ. We get better fit by taking the log of the response (with 0.1 offset to avoid taking the log of 0)

```
# 7.
lm_dvisits <- lm(log(doctorco + 0.1) ~ sex + age + income + freepoor + illness +
    actdays + hscore, data = dvisits)
summary(lm_dvisits)

##
## Call:
## lm(formula = log(doctorco + 0.1) ~ sex + age + income + freepoor +
##     illness + actdays + hscore, data = dvisits)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7777 -0.5103 -0.3019 -0.1190  3.6011
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -2.224692   0.048770 -45.616  < 2e-16 ***
## sexfemale         0.093124   0.029215   3.188  0.00144 **
## age               0.369413   0.073412   5.032 5.02e-07 ***
## income           -0.053272   0.040477  -1.316  0.18820
## freepoorfreepoor -0.229826   0.070087  -3.279  0.00105 **
## illness           0.116920   0.010912  10.715  < 2e-16 ***
## actdays           0.107279   0.004966  21.601  < 2e-16 ***
## hscore            0.029723   0.007086   4.195 2.78e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9808 on 5182 degrees of freedom
## Multiple R-squared:  0.1697,Adjusted R-squared:  0.1686
## F-statistic: 151.3 on 7 and 5182 DF,  p-value: < 2.2e-16
```

compare the mean under lm and glm: note that the mean of a log-normal random variable is $\exp(\mu + \sigma^2/2)$.
`predict(lm_dvisits)` gives the mean and `deviance/df.residual` the variance

```
sigma_lm <- deviance(lm_dvisits)/lm_dvisits$df.residual
plot(exp(predict(lm_dvisits) + sigma_lm/2) - 0.1, predict(glm_dvisits, type = "response"),
    xlab = "linear model responses", ylab = "poisson model responses")
abline(0, 1, col = "red")
```



All the fitted values under lm are lower than the corresponding fitted values under glm. The poisson model assumes that the mean=variance.
The normal model assumes that the variance of $\log(Y)$ is constant (therefore also that $var(Y)$ is constant)

7 Salmonella data

Question 1

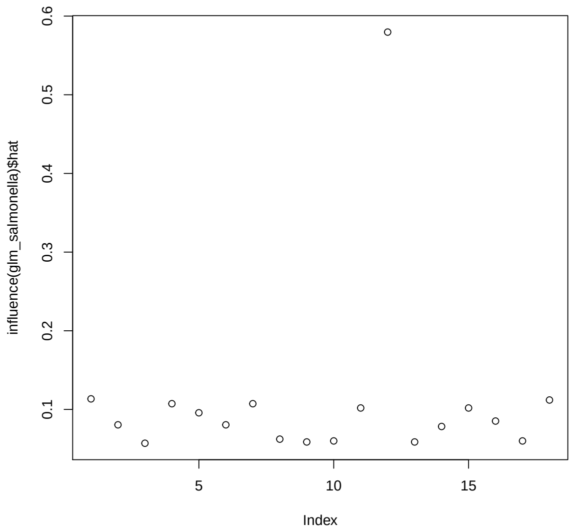
Show that a poisson GLM is inadequate and that some overdispersion must be allowed for. Do not forget to check out other reasons for a high deviance.

```
library(faraway)
data("salmonella")
salmonella

##    colonies dose
## 1        15    0
## 2        21    0
## 3        29    0
## 4        16   10
## 5        18   10
## 6        21   10
## 7        16   33
## 8        26   33
## 9        33   33
## 10       27  100
## 11       41  100
## 12       60  100
## 13       33  333
## 14       38  333
## 15       41  333
## 16       20 1000
## 17       27 1000
## 18       42 1000
```

```
glm_salmonella <- glm(dose ~ colonies, family = poisson, data = salmonella)
summary(glm_salmonella)

##
## Call:
## glm(formula = dose ~ colonies, family = poisson, data = salmonella)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -21.84  -17.97  -14.86    2.46   40.34
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 4.891047   0.040642  120.34   <2e-16 ***
## colonies    0.020105   0.001177   17.09   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 7889.6  on 17  degrees of freedom
## Residual deviance: 7615.1  on 16  degrees of freedom
## AIC: 7716.1
##
## Number of Fisher Scoring iterations: 6

glm_salmonella$deviance

## [1] 7615.106

glm_salmonella$df.residual

## [1] 16
```

the residual variance is much larger than the df other reason than overdispersion could be an outlier or high influence of a point

```
plot(influence(glm_salmonella)$hat)
```



one observation seems high

```
identify(influence(glm_salmonella)$hat)
```

click on the plot and press escape to finish identifying points it is observation 12

```
glm_salmonella2 <- glm(dose ~ colonies, family = poisson, data = salmonella[-12,
    ])
summary(glm_salmonella2)

##
## Call:
## glm(formula = dose ~ colonies, family = poisson, data = salmonella[-12,
##     ])
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -22.354  -16.108  -13.805   -2.992   44.578
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 4.080851   0.055223   73.90   <2e-16 ***
## colonies    0.049655   0.001688   29.42   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 7772.5  on 16  degrees of freedom
## Residual deviance: 6881.6  on 15  degrees of freedom
## AIC: 6976.2
##
## Number of Fisher Scoring iterations: 6
```

still high deviance.

```
dp <- sum(residuals(glm_salmonella, type = "pearson")^2/glm_salmonella$df.residual)
summary(glm_salmonella, dispersion = dp)

##
## Call:
## glm(formula = dose ~ colonies, family = poisson, data = salmonella)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -21.84  -17.97  -14.86    2.46   40.34
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.89105    0.98990   4.941 7.77e-07 ***
## colonies     0.02010    0.02866   0.702    0.483
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 593.2436)
##
##     Null deviance: 7889.6  on 17  degrees of freedom
## Residual deviance: 7615.1  on 16  degrees of freedom
## AIC: 7716.1
##
## Number of Fisher Scoring iterations: 6
```

can also fit a negative binomial

```
library(MASS)
glm_salmonella_negbin <- glm.nb(dose ~ colonies, salmonella)
summary(glm_salmonella_negbin)

##
## Call:
## glm.nb(formula = dose ~ colonies, data = salmonella, init.theta = 0.3165188115,
##     link = log)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.04456  -1.06899  -0.73845   0.01296   1.39384
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.45881    1.13474   3.929 8.52e-05 ***
## colonies     0.03424    0.03621   0.945    0.344
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.3165) family taken to be 1)
##
##     Null deviance: 22.872  on 17  degrees of freedom
## Residual deviance: 22.305  on 16  degrees of freedom
## AIC: 218.77
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.3165
##           Std. Err.:  0.0957
##
##  2 x log-likelihood:  -212.7680
```

8 Lung cancer data

Question 1

Model this count using the city and age category as predictors. Fit a Poisson GLM to the data. Is the fit appropriate?

```
library(ISwR)
data(eba1977)
eba1977

##          city   age  pop cases
## 1  Fredericia 40-54 3059    11
## 2     Horsens 40-54 2879    13
## 3     Kolding 40-54 3142     4
## 4       Vejle 40-54 2520     5
## 5  Fredericia 55-59  800    11
## 6     Horsens 55-59 1083     6
## 7     Kolding 55-59 1050     8
## 8       Vejle 55-59  878     7
## 9  Fredericia 60-64  710    11
## 10    Horsens 60-64  923    15
## 11    Kolding 60-64  895     7
## 12      Vejle 60-64  839    10
## 13 Fredericia 65-69  581    10
## 14    Horsens 65-69  834    10
## 15    Kolding 65-69  702    11
## 16      Vejle 65-69  631    14
## 17 Fredericia 70-74  509    11
## 18    Horsens 70-74  634    12
## 19    Kolding 70-74  535     9
## 20      Vejle 70-74  539     8
## 21 Fredericia   75+  605    10
## 22    Horsens   75+  782     2
## 23    Kolding   75+  659    12
## 24      Vejle   75+  619     7

# 1.
glm_cancer <- glm(cases ~ city + age, data = eba1977, family = poisson)
summary(glm_cancer)
```

```
##
## Call:
## glm(formula = cases ~ city + age, family = poisson, data = eba1977)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.54853  -0.57942  -0.02872   0.49797   1.68933
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.24374    0.20363  11.019   <2e-16 ***
## cityHorsens -0.09844    0.18129  -0.543    0.587
## cityKolding -0.22706    0.18770  -1.210    0.226
## cityVejle   -0.22706    0.18770  -1.210    0.226
## age55-59    -0.03077    0.24810  -0.124    0.901
## age60-64     0.26469    0.23143   1.144    0.253
## age65-69     0.31015    0.22918   1.353    0.176
## age70-74     0.19237    0.23517   0.818    0.413
## age75+      -0.06252    0.25012  -0.250    0.803
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 27.704  on 23  degrees of freedom
## Residual deviance: 20.673  on 15  degrees of freedom
## AIC: 135.06
##
## Number of Fisher Scoring iterations: 5
```

deviance seems ok

Question 2

In the previous model, we are not considering the number of potential cases in each group (ie the population size). Modify the model by using an offset which takes the population size into account.

```
# 2.
glm_cancer_off <- glm(cases ~ offset(log(pop)) + city + age, data = eba1977,
    family = poisson)
summary(glm_cancer_off)

##
## Call:
## glm(formula = cases ~ offset(log(pop)) + city + age, family = poisson,
##     data = eba1977)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.63573  -0.67296  -0.03436   0.37258   1.85267
```

```
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.6321     0.2003 -28.125  < 2e-16 ***
## cityHorsens  -0.3301     0.1815  -1.818   0.0690 .
## cityKolding  -0.3715     0.1878  -1.978   0.0479 *
## cityVejle    -0.2723     0.1879  -1.450   0.1472
## age55-59      1.1010     0.2483   4.434 9.23e-06 ***
## age60-64      1.5186     0.2316   6.556 5.53e-11 ***
## age65-69      1.7677     0.2294   7.704 1.31e-14 ***
## age70-74      1.8569     0.2353   7.891 3.00e-15 ***
## age75+        1.4197     0.2503   5.672 1.41e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 129.908  on 23  degrees of freedom
## Residual deviance:  23.447  on 15  degrees of freedom
## AIC: 137.84
##
## Number of Fisher Scoring iterations: 5
```

age effect is very significant.

Question 3

Fit a binomial model to the data by considering success as being lung cancer cases and failures as being $(populationsize - numberofcases)$.

```
# 3.success as being lung cancer and cases as failures
success <- eba1977$cases
failures <- eba1977$pop - eba1977$cases
glm_cancer_bin <- glm(cbind(success, failures) ~ city + age, family = "binomial",
    data = eba1977)
summary(glm_cancer_bin)
```

```
##
## Call:
## glm(formula = cbind(success, failures) ~ city + age, family = "binomial",
##     data = eba1977)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.64532  -0.67472  -0.03449   0.37480   1.85912
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.6262     0.2008 -28.021  < 2e-16 ***
## cityHorsens  -0.3345     0.1827  -1.830   0.0672 .
## cityKolding  -0.3764     0.1890  -1.991   0.0465 *
## cityVejle    -0.2760     0.1891  -1.459   0.1444
## age55-59      1.1070     0.2490   4.445 8.77e-06 ***
## age60-64      1.5291     0.2325   6.577 4.81e-11 ***
## age65-69      1.7819     0.2305   7.732 1.06e-14 ***
## age70-74      1.8727     0.2365   7.918 2.42e-15 ***
## age75+        1.4289     0.2512   5.688 1.29e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 130.999  on 23  degrees of freedom
## Residual deviance:  23.638  on 15  degrees of freedom
## AIC: 137.74
##
## Number of Fisher Scoring iterations: 5
```

We see that the results are very close to those obtained with the Poisson model with offset. This is because the number of cases is generally very low compared to the population size, in other words, the population size is “almost infinite” compared to the number of cases. In this situation, the Poisson distribution is closely related to the binomial distribution (sampling from a finite, large population of known size is almost the same as sampling from an infinite population).

9 Melanoma data

Question 1

```
mel <- matrix(c(22, 16, 19, 11, 2, 54, 33, 17, 10, 115, 73, 28), nrow = 4, ncol = 3)
colnames(mel) <- c("headneck", "trunk", "extrm")
rownames(mel) <- c("hutch", "superf", "nodular", "indet")
mel

##         headneck trunk extrm
## hutch         22     2    10
## superf        16    54   115
## nodular       19    33    73
## indet         11    17    28

chisq.test(mel)

##
##  Pearson's Chi-squared test
##
## data:  mel
## X-squared = 65.813, df = 6, p-value = 2.943e-12

require(reshape2)
mel.long <- melt(mel, varnames = c("tumtype", "site"), value.name = "freq")
mel.main <- glm(freq ~ tumtype + site, family = poisson, data = mel.long)
mel.int <- glm(freq ~ tumtype * site, family = poisson, data = mel.long)
AIC(mel.main, mel.int)
```

```
##          df      AIC
## mel.main  6 122.9064
## mel.int  12  83.1114

anova(mel.int, test = "Chisq")

## Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: freq
##
## Terms added sequentially (first to last)
##
##
##              Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                            11    295.203
## tumtype       3  145.106         8    150.097 < 2.2e-16 ***
## site          2   98.302         6     51.795 < 2.2e-16 ***
## tumtype:site  6   51.795         0      0.000  2.05e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(exp.count <- predict(mel.main, type = "response"))

##       1       2       3       4       5       6       7       8       9
##   5.780  31.450  21.250   9.520   9.010  49.025  33.125  14.840  19.210
##      10      11      12
## 104.525  70.625  31.640

(obs.count <- predict(mel.int, type = "response"))

##   1   2   3   4   5   6   7   8   9  10  11  12
##  22  16  19  11   2  54  33  17  10 115  73  28

sum((residuals(mel.main, type = "pearson"))^2)

## [1] 65.81293
```

10 Africa data

```
library(faraway)
data(africa)
summary(africa)

##     miltcoup       oligarchy          pollib         parties
##  Min.   :0.000   Min.   : 0.000   Min.   :0.000   Min.   : 0.00
##  1st Qu.:0.000   1st Qu.: 0.000   1st Qu.:1.000   1st Qu.:10.00
##  Median :1.000   Median : 1.000   Median :2.000   Median :13.00
##  Mean   :1.404   Mean   : 4.447   Mean   :1.667   Mean   :15.96
##  3rd Qu.:2.000   3rd Qu.: 9.000   3rd Qu.:2.000   3rd Qu.:19.00  
##  Max.   :6.000   Max.   :18.000   Max.   :2.000   Max.   :62.00  
##                                   NA's   :5                       
##     pctvote           popn              size           numelec      
##  Min.   : 0.00   Min.   :  0.067   Min.   :   0.5   Min.   : 0.000  
##  1st Qu.:18.90   1st Qu.:  1.450   1st Qu.:  33.0   1st Qu.: 4.000  
##  Median :28.95   Median :  5.600   Median : 274.0   Median : 6.000  
##  Mean   :31.88   Mean   : 10.953   Mean   : 516.7   Mean   : 6.191  
##  3rd Qu.:43.04   3rd Qu.: 11.450   3rd Qu.: 813.0   3rd Qu.: 8.500  
##  Max.   :77.40   Max.   :113.800   Max.   :2506.0   Max.   :14.000  
##  NA's   :6                                                          
##     numregim    
##  Min.   :1.000  
##  1st Qu.:2.000  
##  Median :3.000  
##  Mean   :2.511  
##  3rd Qu.:3.000  
##  Max.   :4.000  
## 
```

```
str(africa)
```

```
## 'data.frame': 47 obs. of  9 variables:
##  $ miltcoup : int  0 5 0 6 2 0 1 3 1 2 ...
##  $ oligarchy: int  0 7 0 13 13 0 0 14 15 0 ...
##  $ pollib   : int  2 1 NA 2 2 2 2 2 2 2 ...
##  $ parties  : int  38 34 7 62 10 34 5 14 27 4 ...
##  $ pctvote  : num  NA 45.7 20.3 17.5 34.4 ...
##  $ popn     : num  9.7 4.6 1.2 8.8 5.3 11.6 0.361 3 5.5 0.458 ...
##  $ size     : num  1247 113 582 274 28 ...
##  $ numelec  : int  0 8 5 5 3 14 2 6 4 6 ...
##  $ numregim : int  1 3 1 3 3 3 1 4 3 2 ...
```

```
# we notice that pollib should be a factor
africa$pollib <- factor(africa$pollib)
glm_africa <- glm(miltcoup ~ ., data = africa, family = poisson)
summary(glm_africa)
```

```
## 
## Call:
## glm(formula = miltcoup ~ ., family = poisson, data = africa)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -1.5075  -0.9533  -0.3100   0.4859   1.6459  
## 
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -0.2334274  0.9976112  -0.234  0.81500  
## oligarchy    0.0725658  0.0353457   2.053  0.04007 *
## pollib1     -1.1032439  0.6558114  -1.682  0.09252 .
## pollib2     -1.6903057  0.6766503  -2.498  0.01249 *
## parties      0.0312212  0.0111663   2.796  0.00517 **
## pctvote      0.0154413  0.0101027   1.528  0.12641
## popn         0.0109586  0.0071490   1.533  0.12531
## size        -0.0002651  0.0002690  -0.985  0.32444
## numelec     -0.0296185  0.0696248  -0.425  0.67054
## numregim     0.2109432  0.2339330   0.902  0.36720
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 65.945  on 35  degrees of freedom
## Residual deviance: 28.249  on 26  degrees of freedom
##   (11 observations deleted due to missingness)
## AIC: 113.06
##
## Number of Fisher Scoring iterations: 5
```

```
glm_africa <- update(glm_africa, . ~ . - numelec)
glm_africa <- update(glm_africa, . ~ . - numregim)
glm_africa <- update(glm_africa, . ~ . - size)
glm_africa <- update(glm_africa, . ~ . - popn)
glm_africa <- update(glm_africa, . ~ . - pctvote)
summary(glm_africa)
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties, family = poisson,
##     data = africa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4392  -1.0775  -0.3756   0.5738   1.7526
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.253231   0.443079   0.572   0.5676
## oligarchy    0.098412   0.020988   4.689 2.74e-06 ***
## pollib1     -0.480040   0.469087  -1.023   0.3061
## pollib2     -1.013746   0.448055  -2.263   0.0237 *
## parties      0.016554   0.008806   1.880   0.0601 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 79.124  on 41  degrees of freedom
## Residual deviance: 42.235  on 37  degrees of freedom
##   (5 observations deleted due to missingness)
## AIC: 125.92
##
```

```
## Number of Fisher Scoring iterations: 5
```

for each added year of oligarchy, the number of coups is increased by exp(0.09) while the number of coups is decreased if the pollib=2 compared to 0 (goes from no civil rights to full civil rights) increase in the number of parties also tend to increase the number of coups.