# PSTAT127_HW6

*Celine Mol*

*February 27, 2017*

## 1.

```r
library(faraway)
data("worldcup")
#Remove goalkeepers from the dataset
worldcup <- worldcup[!(worldcup$Position=="Goalkeeper"),]
#Due to substitution and varying success in the tournament, number of minutes
#played is quite variable by each player. Compute new variables for the number
#of tackles and passes made per 90-minute game
newTackles <- (worldcup$Tackles * 90.0)/worldcup$Time
newPasses <- (worldcup$Passes * 90.0) / worldcup$Time
```

### a) Fit a Poisson model with the number of shots as the response and team, position, tackles and passes per game as predictor. Note that time played is a rate variable and should be accounted for as described in Section 5.3. Interpret the effect of tackles and passes on shots.

```r
#Since time played is a rate variable, we should model the count response, so to
#create a rate model, we need to use the log of Time.
model <- glm(worldcup$Shots ~ offset(log(worldcup$Time)) + worldcup$Team +
               worldcup$Position + newTackles + newPasses, family=poisson)
summary(model)
```

```
##
## Call:
## glm(formula = worldcup$Shots ~ offset(log(worldcup$Time)) + worldcup$Team +
##     worldcup$Position + newTackles + newPasses, family = poisson)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.9982  -1.0468  -0.3104   0.5641   4.3556
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -5.0968308  0.2063050 -24.705  < 2e-16 ***
## worldcup$TeamArgentina      0.0643970  0.2253098   0.286 0.775020
## worldcup$TeamAustralia     -0.2082580  0.2588007  -0.805 0.420990
## worldcup$TeamBrazil         0.2200378  0.2185350   1.007 0.313994
## worldcup$TeamCameroon      -0.0460037  0.2493812  -0.184 0.853644
## worldcup$TeamChile          0.0316334  0.2360436   0.134 0.893391
## worldcup$TeamDenmark       -0.1792117  0.2523051  -0.710 0.477520
## worldcup$TeamEngland        0.0033321  0.2310205   0.014 0.988492
## worldcup$TeamFrance        -0.1508411  0.2589482  -0.583 0.560220
## worldcup$TeamGermany       -0.1303409  0.2220481  -0.587 0.557208
## worldcup$TeamGhana           0.1566204  0.2143928   0.731 0.465066
## worldcup$TeamGreece         -0.0646529  0.2529095  -0.256 0.798231
## worldcup$TeamHonduras       -1.0306478  0.3308377  -3.115 0.001838 **
## worldcup$TeamItaly          -0.0857020  0.2485438  -0.345 0.730232
## worldcup$TeamIvory Coast    -0.0224531  0.2424748  -0.093 0.926222
## worldcup$TeamJapan          -0.2408761  0.2424290  -0.994 0.320420
## worldcup$TeamMexico          0.0144959  0.2377421   0.061 0.951381
## worldcup$TeamNetherlands    -0.1339307  0.2202002  -0.608 0.543040
## worldcup$TeamNew Zealand    -0.9088003  0.3157115  -2.879 0.003995 **
## worldcup$TeamNigeria        -0.2660574  0.2591583  -1.027 0.304599
## worldcup$TeamNorth Korea    -0.1349975  0.2528652  -0.534 0.593430
## worldcup$TeamParaguay       -0.3180785  0.2312488  -1.375 0.168982
## worldcup$TeamPortugal        0.0894685  0.2307862   0.388 0.698262
## worldcup$TeamSerbia          0.0727482  0.2426215   0.300 0.764298
## worldcup$TeamSlovakia       -0.3819174  0.2475044  -1.543 0.122813
## worldcup$TeamSlovenia       -0.7925716  0.3022278  -2.622 0.008730 **
## worldcup$TeamSouth Africa    0.0043928  0.2491021   0.018 0.985930
## worldcup$TeamSouth Korea     0.0084120  0.2347408   0.036 0.971414
## worldcup$TeamSpain           0.1234560  0.2202803   0.560 0.575173
## worldcup$TeamSwitzerland    -0.5055568  0.2830170  -1.786 0.074049 .
## worldcup$TeamUSA            -0.0342582  0.2259684  -0.152 0.879498
## worldcup$TeamUruguay        -0.1784583  0.2172783  -0.821 0.411455
## worldcup$PositionForward     1.6065065  0.0869572  18.475  < 2e-16 ***
## worldcup$PositionMidfielder  0.9814239  0.0826426  11.876  < 2e-16 ***
## newTackles                  -0.0873870  0.0258248  -3.384 0.000715 ***
## newPasses                    0.0007916  0.0023943   0.331 0.740930
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 1474.38  on 558  degrees of freedom
## Residual deviance:  865.76  on 523  degrees of freedom
## AIC: 2036.1
##
## Number of Fisher Scoring iterations: 5
```

```r
anova(model, test="Chi")
```

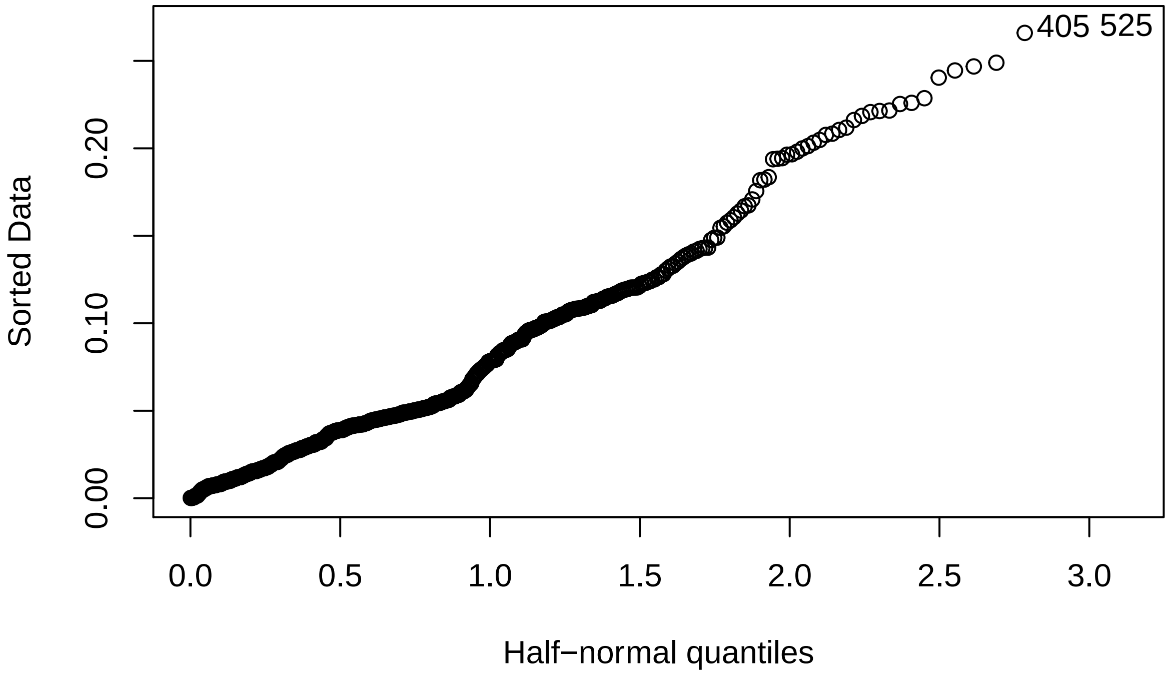
```
## Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: worldcup$Shots
##
## Terms added sequentially (first to last)
##
##
##                   Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                                558    1474.38
## worldcup$Team     31    67.35       527    1407.03 0.0001682 ***
## worldcup$Position  2   529.42       525     877.61 < 2.2e-16 ***
## newTackles         1    11.74       524     865.87 0.0006117 ***
## newPasses          1     0.11       523     865.76 0.7411660
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Here, Yi = number of shots within time i and Timei = Time each player is on the floor. Our random component is Yi ~ independent Poisson(Ui) where Ui depends on Timei. Since Ui depends on Timei, we can put log(Timei) as a predictor and offset it so that it continues to have a ratio with coefficient of 1. This would give us a systematic component of ln(Ui) = ln(Timei) + xi'B and a link function g(Ui) = ln(Ui).

Shown through the above model statistics, tackles seems to have a significant effect on shots, but the effect of passes do not seem to be significant.

### b) Calculate the leverages for the current model. Report which player has the highest leverage and suggest why this might be so. Make an appropriate plot of the leverages and comment on whether any leverage is exceptional.

```
levg <- influence(model)
#For a GLM, we do not expect the residuals to be normally distributed, but we are
#still interested in detecting outliers. For this purpose, it is better to use a
#half-normal plot that compares the sorted absolute residuals and the quantiles
#of the half-normal distribution
halfnorm(levg$hat)
```



```
worldcup[405,]
```

```
##                          Team Position Time Shots Passes Tackles Saves
## Park Chu-Young South Korea  Forward  347    14     96       0     0
```

```
worldcup[525,]
```

```
##          Team Position Time Shots Passes Tackles Saves
## Villa Spain  Forward  529    22    169       2     0
```

```
newTackles[405]
```

```
## [1] 0
```

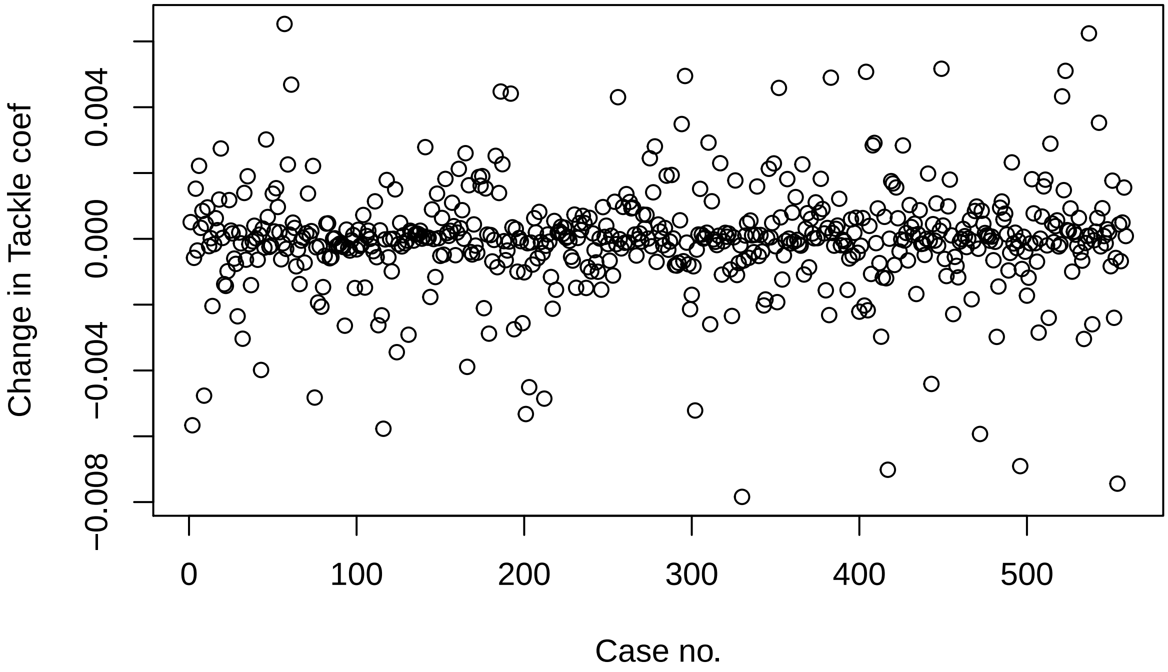
```
newTackles[525]
```

```
## [1] 0.3402647
```

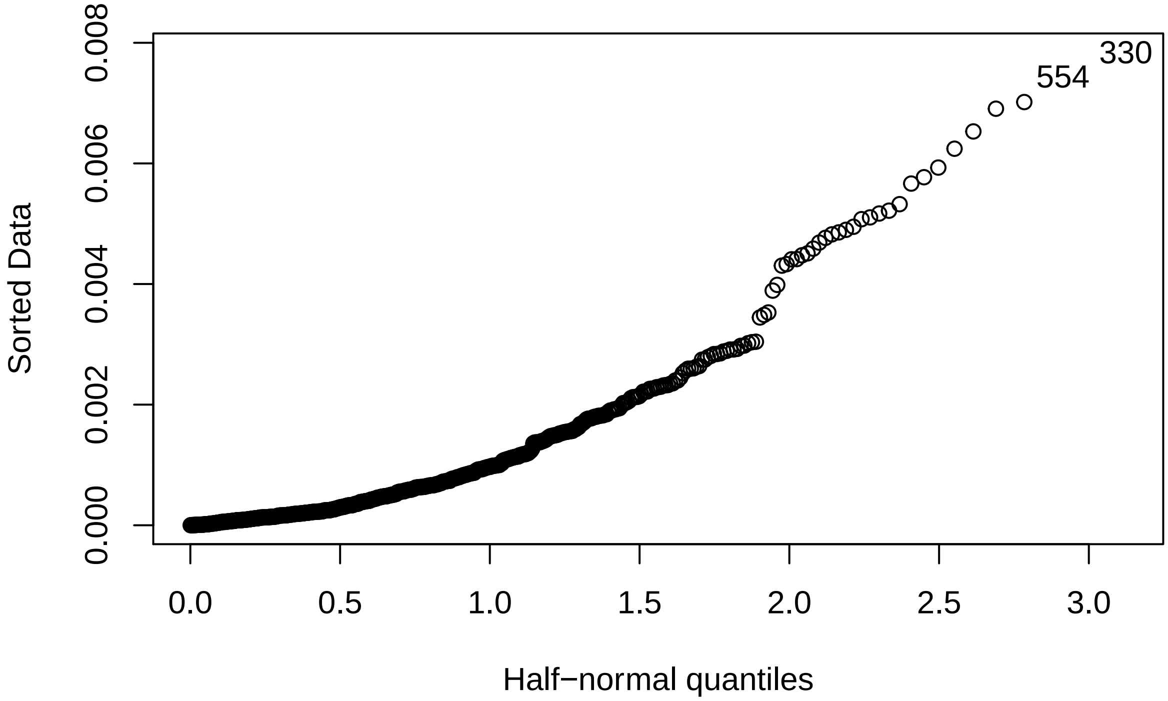
There is some indication that case 405 and 525 may have some leverage, which are players Park Chu-Young from South Korea and Villa from Spain. Leverage is a measure of how far away the independent variable values of an observation are from those of the other observations. These players may have high leverage because their Tackles are low (0 and 2 respectively), even though their playing times are quite high (347 and 529 respectively), which gives them an overall low value for newTackles (0 and 0.340). This can give these players high leverage because Tackles has a significant effect on Shots, and their Shot values are high (14 and 22 respectively), allowing them to be outlying values of the independent variables (depending on both the Xs and the Ys).

### c) Compute the change in the regression coefficients when each case is dropped. In particular, examine the change in the tackle coefficient identifying the player which causes the greatest absolute change in this value. What is unusual about this player? Plot the change in the tackle coefficient and determine if any of the values is particularly large.

```
#We can examine the change in the fit from ommitting a case by looking at the
#changes in coefficients in an index plot of the change in the tackle coefficient
plot(levg$coefficients[,35], ylab="Change in Tackle coef", xlab="Case no.")
```



```
#If we take a look at the halfnorm plot, we can see which case number has a
#substantial change
halfnorm(levg$coefficients[,35])
```



```
worldcup[330,]
```

```
##                   Team   Position Time Shots Passes Tackles Saves
## Mascherano Argentina Midfielder  360     0    237      19     0
```

```
newTackles[330]
```

```
## [1] 4.75
```

We can identify that player number 330 causes the greatest absolute change in this value. What we know about this player is that it has a high number of Tackles and a relatively high amount of playing Time (giving it a newTackles value of 4.75, which is above the 3rd Quadrant), but 0 shots, which could cause the greatest change in the value of the Tackle coefficient since currently the Tackle coefficient is significant in its representation of the number of shots that will be made. If we compare the full fit to a model without this case, we find:

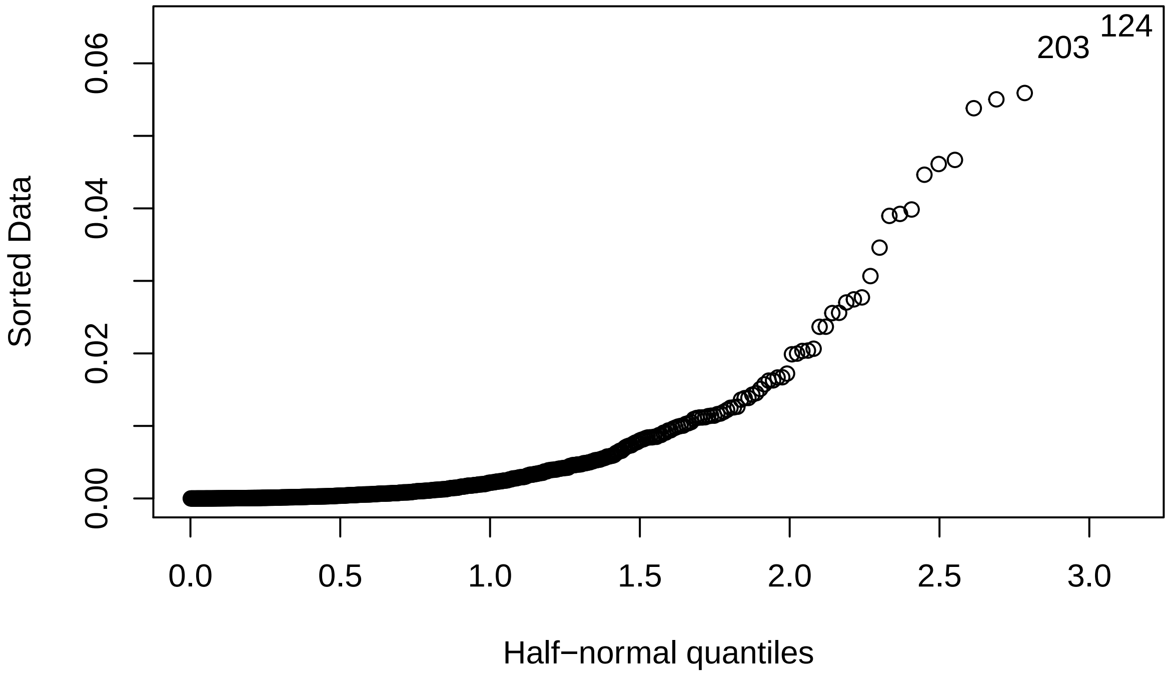
```
modplr <- glm(worldcup$Shots ~ offset(log(worldcup$Time)) + worldcup$Team + worldcup$Position + newTackl
cbind(coef(model), coef(modplr))
```

```
##                                     [,1]          [,2]
## (Intercept)                -5.0968308191 -5.1073090942
## worldcup$TeamArgentina      0.0643969863  0.1324400080
## worldcup$TeamAustralia     -0.2082580005 -0.2056698734
## worldcup$TeamBrazil         0.2200378187  0.2230912622
## worldcup$TeamCameroon      -0.0460036524 -0.0412371306
## worldcup$TeamChile          0.0316334126  0.0304818523
## worldcup$TeamDenmark       -0.1792117060 -0.1770809889
## worldcup$TeamEngland        0.0033321414  0.0068388385
## worldcup$TeamFrance        -0.1508410806 -0.1502624162
## worldcup$TeamGermany       -0.1303409197 -0.1278980555
## worldcup$TeamGhana          0.1566204062  0.1593928963
## worldcup$TeamGreece        -0.0646529366 -0.0629727391
## worldcup$TeamHonduras      -1.0306478201 -1.0281149371
## worldcup$TeamItaly         -0.0857020175 -0.0831727657
## worldcup$TeamIvory Coast    -0.0224530904 -0.0192222316
## worldcup$TeamJapan          -0.2408760952 -0.2405249139
## worldcup$TeamMexico          0.0144958847  0.0159581224
## worldcup$TeamNetherlands    -0.1339307257 -0.1302945423
## worldcup$TeamNew Zealand    -0.9088002599 -0.9060867116
## worldcup$TeamNigeria        -0.2660573509 -0.2637028314
## worldcup$TeamNorth Korea    -0.1349975276 -0.1333713863
## worldcup$TeamParaguay       -0.3180784502 -0.3177473797
## worldcup$TeamPortugal        0.0894684795  0.0883183976
## worldcup$TeamSerbia          0.0727481530  0.0767957925
## worldcup$TeamSlovakia       -0.3819174264 -0.3770467517
## worldcup$TeamSlovenia       -0.7925715532 -0.7888773584
## worldcup$TeamSouth Africa    0.0043928008  0.0063649039
## worldcup$TeamSouth Korea     0.0084120031  0.0095696349
## worldcup$TeamSpain           0.1234560152  0.1301634376
## worldcup$TeamSwitzerland    -0.5055567902 -0.5036960427
## worldcup$TeamUSA            -0.0342581754 -0.0303704762
## worldcup$TeamUruguay        -0.1784583249 -0.1781049042
## worldcup$PositionForward     1.6065065256  1.6086769163
## worldcup$PositionMidfielder  0.9814239175  0.9897025936
## newTackles                  -0.0873869727 -0.0818845870
## newPasses                    0.0007915914  0.0006459424
```

It seems to me that there is no unusual change without this player… all the coeffients in our new model are about the same as our old model, and looking specifically at newTackles, the value barely changes. This may mean that although this player causes the greatest absolute change in the Tackle coefficient, it may not change the model significantly if removed.

### d) Calculate the Cooks statistics. Which player has the largest such statistic and what is unusual about him?

```
halfnorm(cooks.distance(model))
```



```
worldcup[124,]
```

```
##          Team   Position Time Shots Passes Tackles Saves
## Dempsey  USA Midfielder  390    15    137       6     0
```

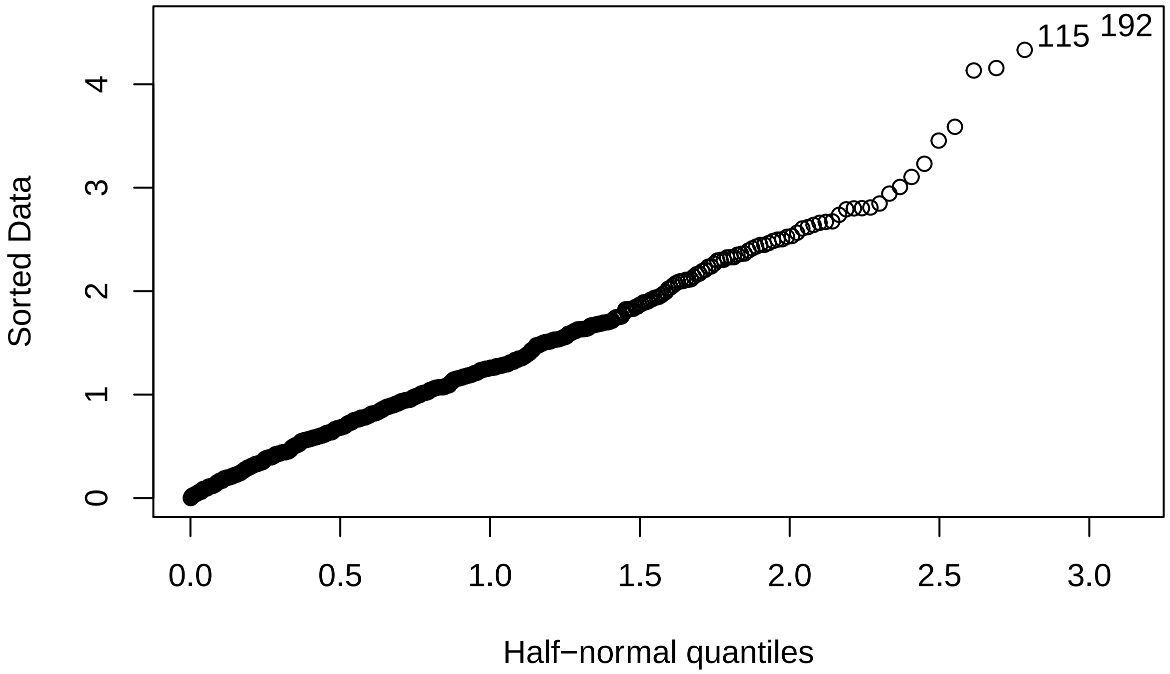
```
worldcup[203,]
```

```
##        Team Position Time Shots Passes Tackles Saves
## Gyan Ghana  Forward  501    27    151       1     0
```

The players with the largest such statistics are 124 and 203, which are players Dempset from USA and Gyan from Ghana. Cook's distance is used to estimate the influence of a data point when performing a least squares regression and measures the effect of deleting a data point with large residuals or high leverage values. The cook's statistics values here are different from the one recommended in looking at the change in the Tackle coefficient. This means that they may have a lot of influence in other coefficient values, possibly with regard to their Team or Position value, or since they both have a high number of passes (137 and 151 respectively), more likely the influence of Passes on Shots (15 and 27). In any case, the cook's statistics value does not go any higher than 0.06, so these data points may not have too much influence on the model anyways (Cook's statistics values greater than 1 are suggested to determine cut off values for spotting highly influential points).

### e) Find the jacknife residuals. Find the player with the largest absolute residual of this kind. How did he come to be the largest?

```
halfnorm(rstudent(model))
```



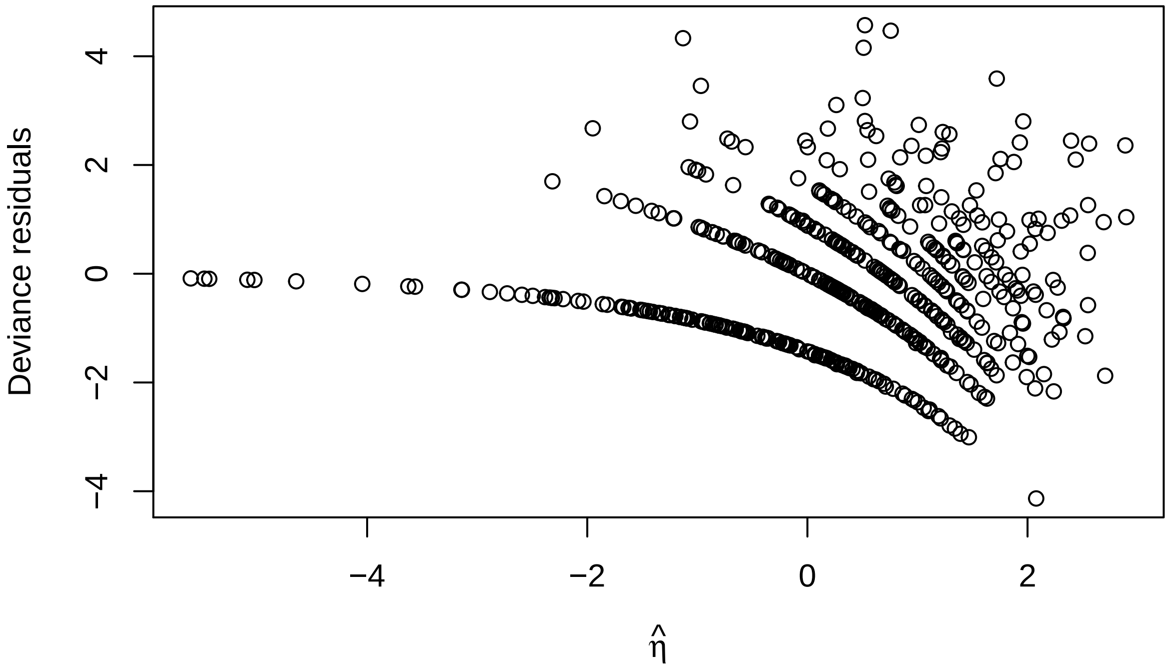
```
#worldcup[115,]
worldcup[192,]
```

```
##            Team   Position Time Shots Passes Tackles Saves
## GonzalezC Chile Midfielder  138    10     41       6     0
```

The player with the largest jacknife residuals is 192, which is player Gonzalez from Chile. Jacknife residuals are also used to indicate outliers. Gonzalez has a relatively high number of Tackles, Passes, and Shots, with a relatively high amount of time. This player is an outlier because hes not an average player, all of his characteristics are high.

### f) Plot the residuals against the appropriate fitted values. Explain the source of the lines of points appearing on the plot. What does this plot indicate?

```
jacknife <- rstudent(model)
plot(jacknife ~ predict(model, type="link"), xlab=expression(hat(eta)),
     ylab="Deviance residuals")
```



We get different curves because our Yi's are discrete, so this plot indicates that we are dealing with discrete variables. For example, if we are dealing with bernoulli variables, we would be dealing with 2 counts, 0 and 1, so our residual plots would have two curves. Since we are dealing with poisson, we should have more than 2 curves, based on how many counts our response variable Yi has. Since our Yi represents shots, I would assume the first (bottom) curve is when shots = 0, then the second where shots = 1, and as we go higher up in the count of shots, we have more diversity in how many counts have shots = i (i.e., only one player has shots = 27). This means that the plot of the residuals is not particularly helpful, because it shows curved lines of points corresponding to the counts of observed responses, which prevents us from checking whether there is a nonlinear relationship between the predicted values and the residuals, so we cannot indicate whether or not there is a lack of fit (obscures the main purpose of the plot).

## g) Plot:

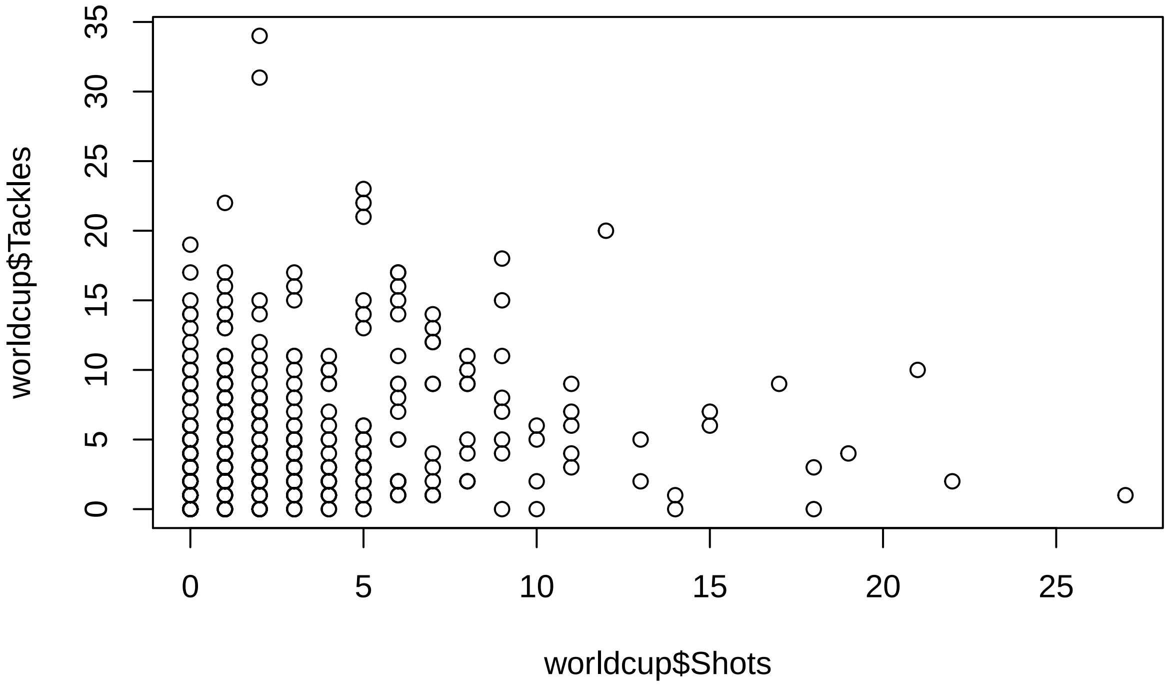
**- Raw shots against tackles**

**- Shots per game against tackles per game**

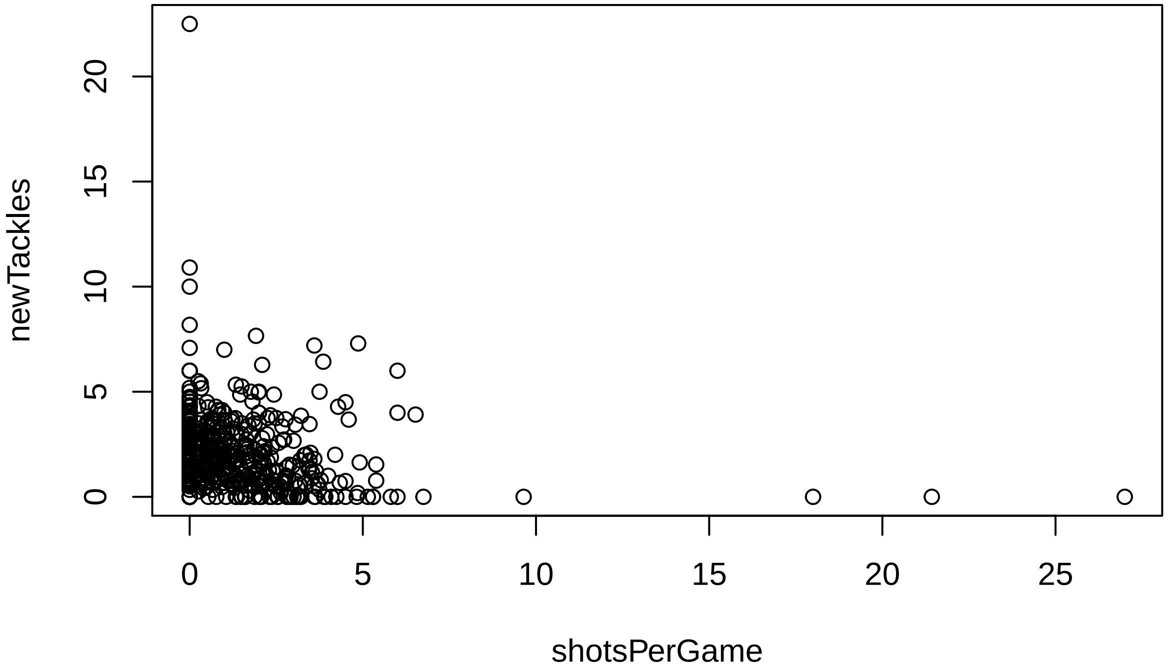
**- Linearized response against tackles per game**

### Make an interpretation of each plot and choose the best one for discovering the relationship between this predictor and the response. Justify your choice.

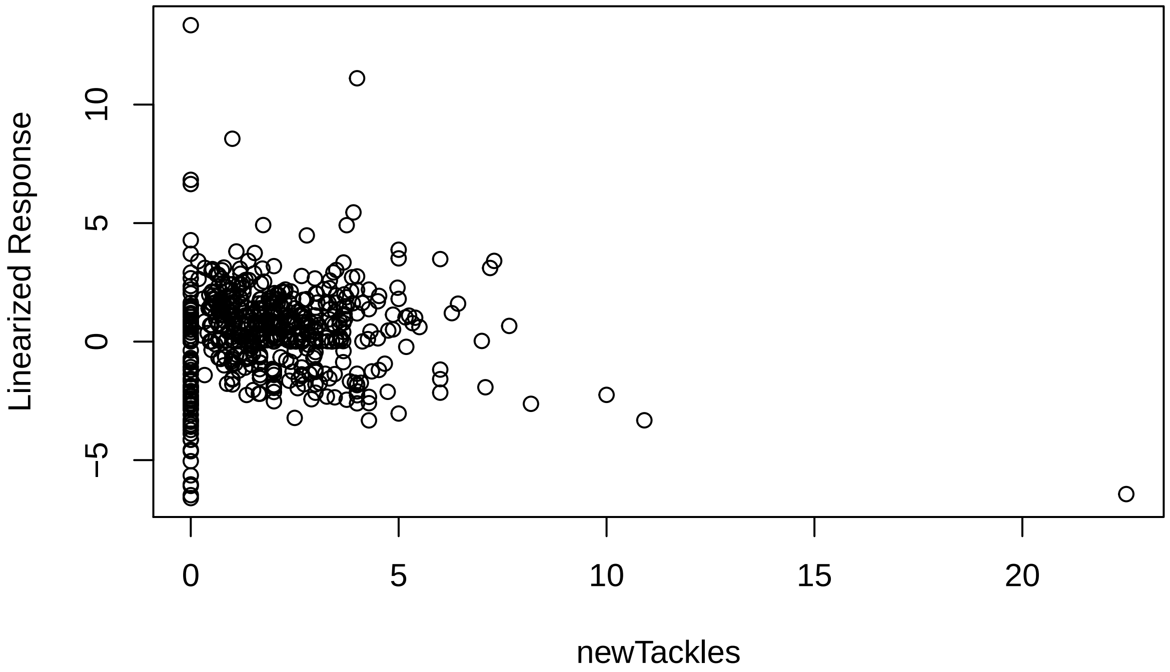
```
#Raw shots against tackles
plot(worldcup$Tackles ~ worldcup$Shots)
```



```
#Shots per game against tackles per game
shotsPerGame <- (worldcup$Shots * 90.0) / worldcup$Time
plot(newTackles ~ shotsPerGame)
```



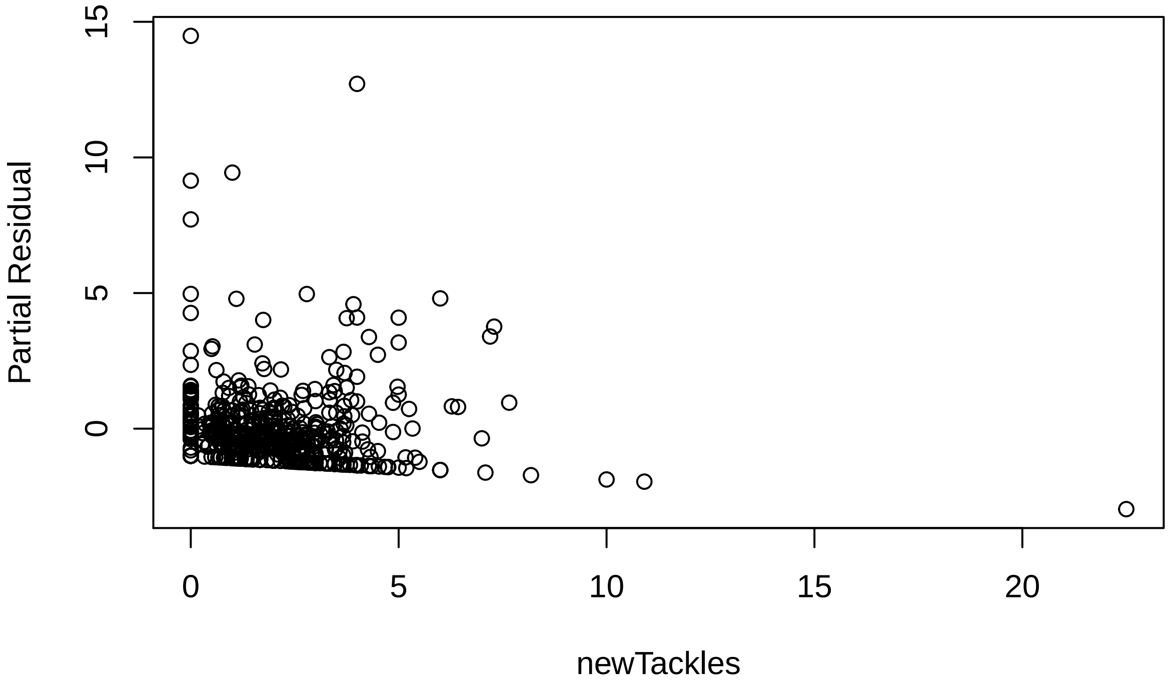
```
#Linearized response against tackles per game
mu <- predict(model, type="response")
z <- predict(model) + (worldcup$Shots-mu)/mu
plot(z ~ newTackles, ylab="Linearized Response")
```



We want to investigate the nature of the relationship between the predictors and the response. In the first plot, we can see that more Tackles does not necessarily mean more Shots, and again we can recognize the discreteness in the response variable Shots. In the second plot, we can clearly see that both variables have skewed distributions, and probably should perform a transformation on the predictor to get a more adequate representation of our data. This skewedness is also recognized in our linearized response plot, because we cannot recognize a linear relationship in our data. Maybe this suggests that our data does not follow a linear relationship, or a transformation of newTackles is necessary. I think the first plot is the best at discovering the relationship between the the predictor and response because you can clearly see that we are dealing with count data and there is no skewedness in the plot.

### h) Construct the partial residual plot for tackles and interpret. Is the point on the far right really influential?

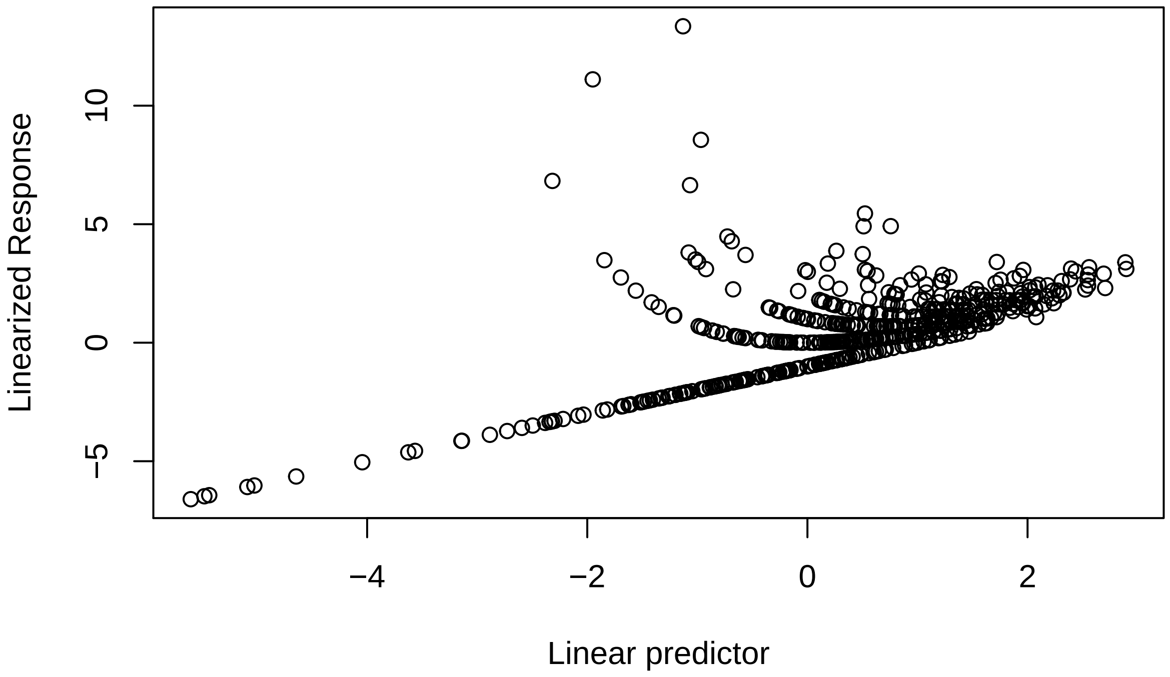
```
#Partial residuals for tackles per game (newTackles)
u <- (worldcup$Shots-mu)/mu + coef(model)[35]*newTackles
plot(u ~ newTackles, ylab="Partial Residual")
```



We want to check the partial residuals plot because we want to take into account the effects of other predictors, in this case newTackles. Partial residual plots are used for linear models to make allowance for the effect of the other predictors while focusing on the relationship of interest. Again, there is some reason for concern, since we see nonlinearity, indicating a need to transform, and there are also obvious outliers and influential points.

### i) Make a diagnostic plot to check the choice of the (default) link function.

```r
plot(z ~ predict(model), xlab="Linear predictor", ylab="Linearized Response")
```



To adequately look at the choice of the link function, it is important to eliminate other simpler violations of the assumptions such as outliers or transformations. We did not do that here, but our link function still shows somewhat of a linear trend, incorporating our discrete poisson counts for the response variable Shots. Thus, we can make an assumption that our link function is adequate.

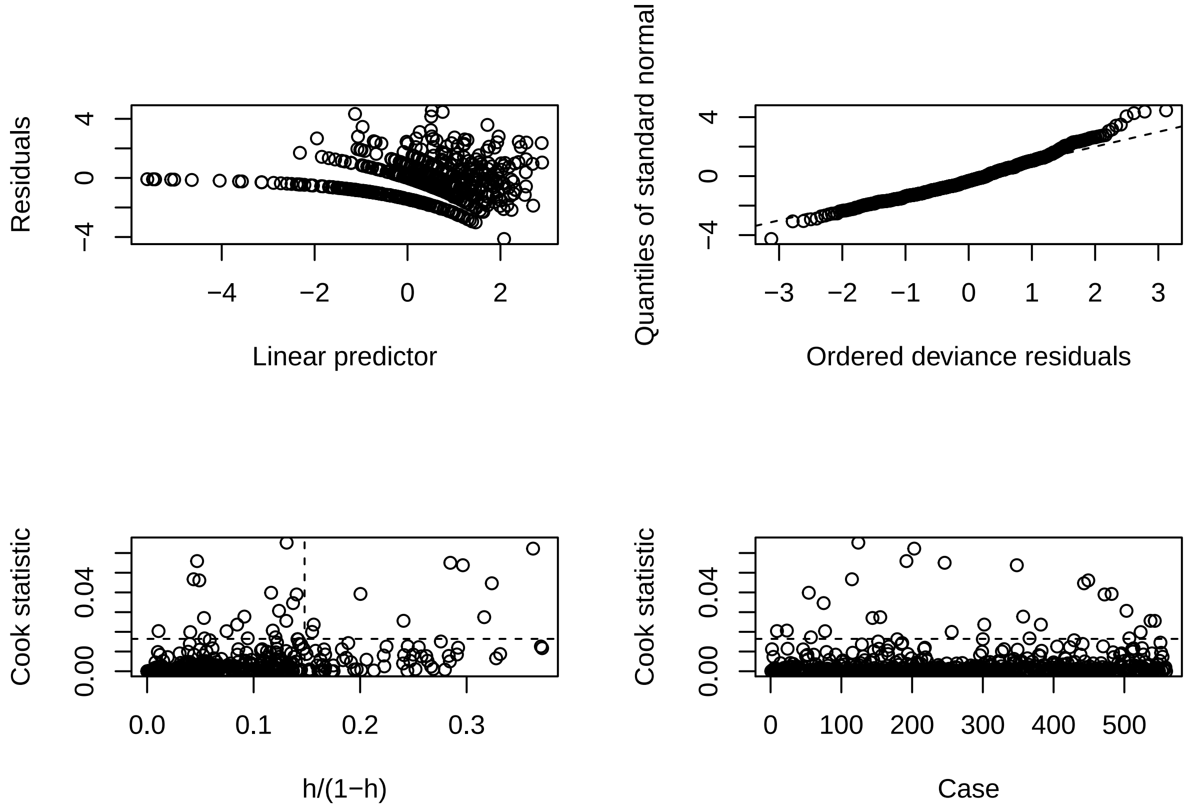
### In addition, also examine the plots produced by glm.diag.plots in library(”boot”) for your data analysis.

```
library(boot)
```

```
##
## Attaching package: 'boot'

## The following objects are masked from 'package:faraway':
##
##     logit, melanoma
```

```
glm.diag.plots(model)
```



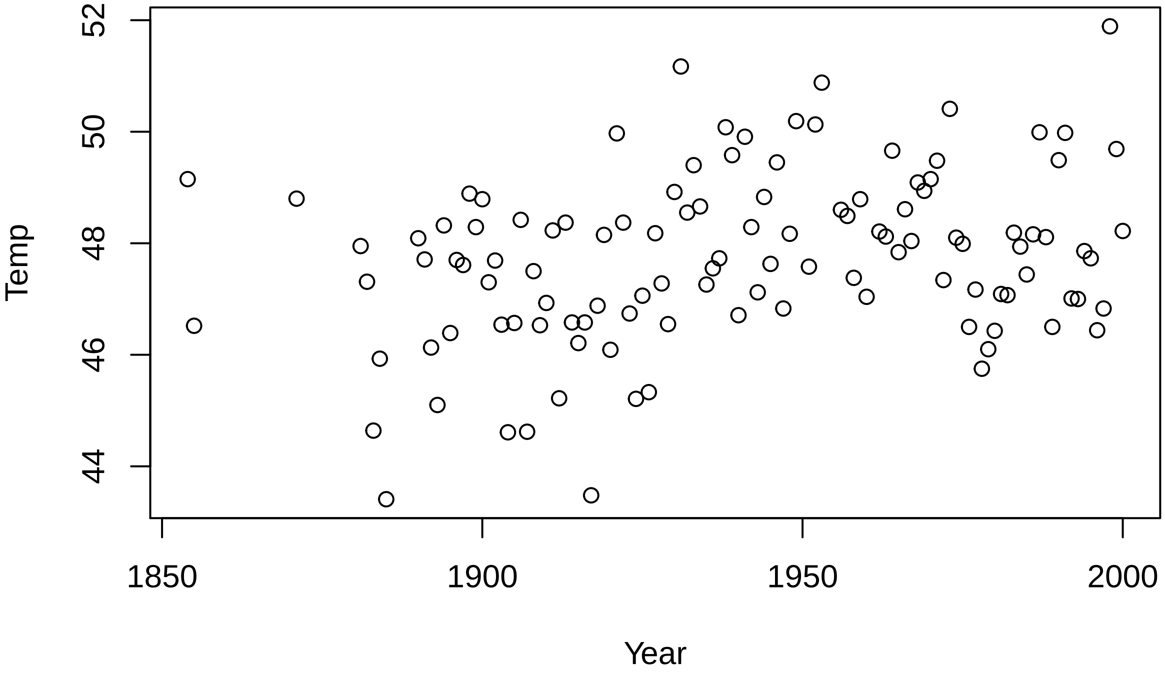
The plot on the top left is one that we have already discussed. The plot on the top right is a normal QQ plot of the standardized deviance residuals and shows us that our standard errors follow a normal distribution, which validates our assumption. The plot on the bottom left is a plot of the Cooks statistics against the standardized leverages. Points above the dotted horizontal are points that have high influence on the model, and points to the right of the vertical line have high leverage compared to the variance of the raw residual at that point. The plot on the bottom right shows the Cooks statistic plotted against case number, enabling us to find which observations are influential. We can see here that we have quite a few points that are influential, or above the dotted horizontal line.

## 2.

```r
data("aatemp")
```

### 5.(a) Plot the temperature as a function of time and comment on the underlying trend.

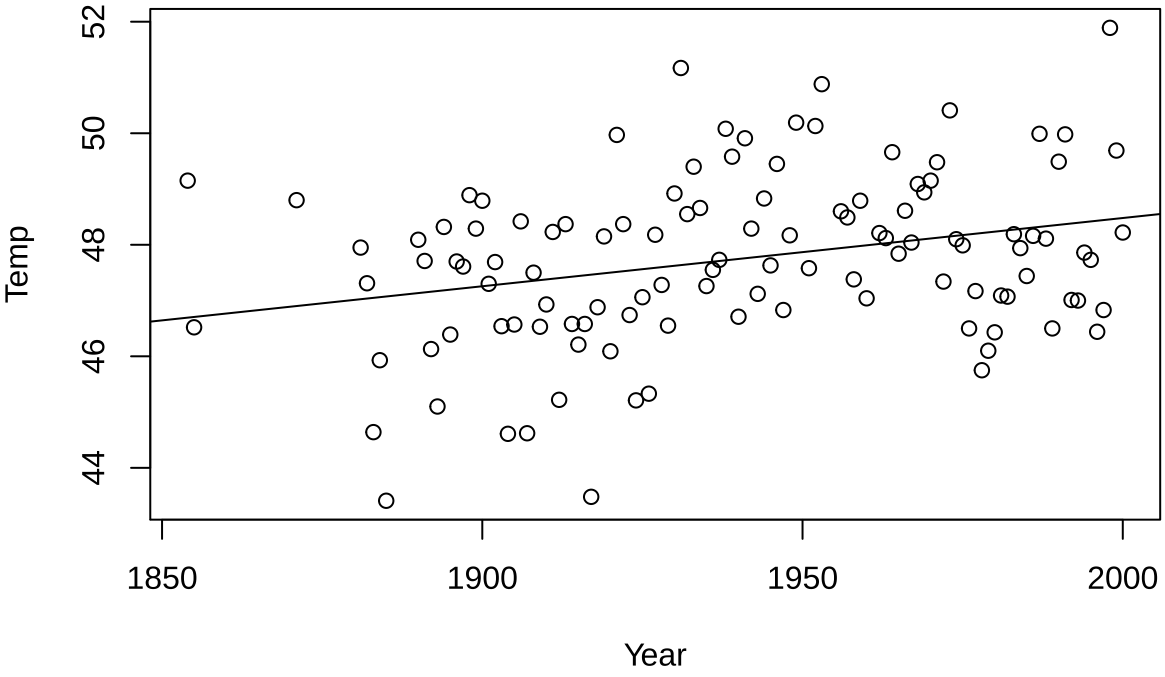
```r
plot(aatemp$temp ~ aatemp$year, xlab="Year", ylab="Temp")
```



The trend here looks somewhat linear, so we can assume that the temperature increases as the year increases.

### b.) Fit a least squares line to the data and test whether the slope of the line is different from zero. What is the main drawback of this modeling approach?

```r
fit2 <- lm(aatemp$temp ~ aatemp$year)
plot(aatemp$year, aatemp$temp, xlab="Year", ylab="Temp")
abline(fit2)
```



```
summary(fit2)
```

```
##
## Call:
## lm(formula = aatemp$temp ~ aatemp$year)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9843 -0.9113 -0.0820  0.9946  3.5343
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 24.005510   7.310781   3.284  0.00136 **
## aatemp$year  0.012237   0.003768   3.247  0.00153 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.466 on 113 degrees of freedom
## Multiple R-squared:  0.08536,    Adjusted R-squared:  0.07727
## F-statistic: 10.55 on 1 and 113 DF,  p-value: 0.001533
```

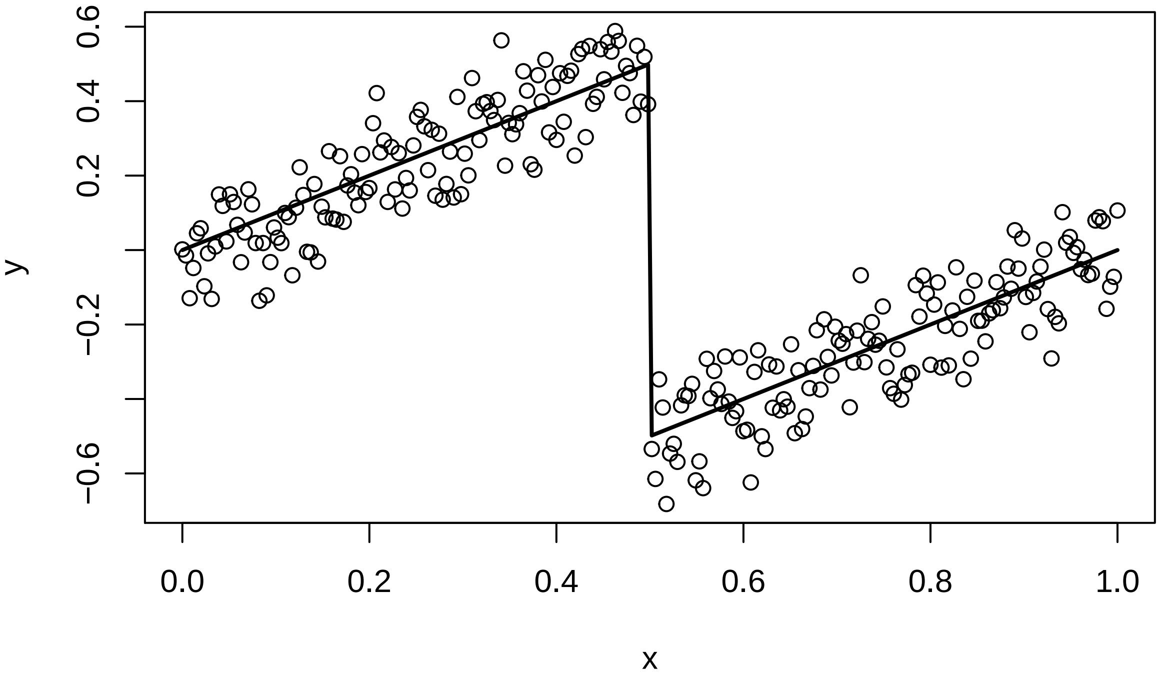
Here, our model is: Temp = Bo + B1(Year) + e and our assumptions are e ~ N(0,sigma^2).
We can use a least squares regression to see what linear fit our data represents. For our test, our Ho: B1 = 0 (no slope) and Ha: B1 not equal to 0 (significant slope). We perform a t test to determine whether the slope is significant, where the t value is the Estimate / Std. Error and follows a t distribution with (n-1) degrees of freedom. The rejection region is { |t| > t(crit),(n-1)df,alpha/2 } Shown through a summary of our fit, the slope of the line is 0.01224, which is significant at alpha = 0.05 shown through a p-value of 0.00153, so we reject our hypothesis that the slope is 0 and conclude that the slope is significant, or not = 0.

The main drawback of this modeling approach is that it only looks at the means, and removes a lot of the variance from the data, which is misleading.

## 3.

**6.(a) Plot the data along with the true function f.**

```
set.seed(10)
xv <- c() #to create 256 evenly spaced points on 0,1
f <- function(x){
  if(x > (1/2)) {
    return(x-1)
  } else {
    return(x)
  }
}
e <- rnorm(n=256, mean=0, sd=0.1)
y <- c()
p <- 1
for (x in seq(0,1,(1/255))) {
  xv[p] <- x
  y[p] <- f(x) + e[p]
  p <- p + 1
}
p <- 1
#m is the true function f(x)
m <- c()
for (x in seq(0,1,(1/255))){
  m[p] <- f(x)
  p <- p + 1
}
plot(y ~ xv, xlab="x")
lines(m ~ xv, lwd=2) #Represents the true function f
```


y
0.6
0.4
0.2
−0.2
−0.6
0.0
0.2
0.4
0.6
0.8
1.0
x