Medical Statistics:Assignment 1

Stephen Brownsey

Question 1

The dataset chosen for this analysis was `"blocks"` from the package *GLMsData*. It was chosen in particular for analysis due its high number of observations, $n = 100$ and mix of both continuous and categoric predictor variables. These include a response variable `Number`, that is, the number of blocks the child could successfully stack, and predictor variables: `Time` (numeric vector), `Trial` (2 Levels), `Shape`, the type of blocks stacked (2 Levels) and `Age` (numeric vector), given in years of the child. Since the response variable `Number` models count data, a suitable generalised linear model for our analysis is a Poisson generalised linear model with a log link function, the default option in R.

Question 2

The response variable `Number` will be considered as Y_i for the purpose of the equations below. The distribution of Y_i, from $i = 1, ..., n$ can be modelled by the Poisson distribution with mean μ_i:

$$Y_i \sim Poisson(\mu_i)$$

which has the probability mass function $f(y) = e^{-\mu}\frac{\mu^y}{y!}$

One common link function used for the Poisson regression is the log link function. That is:

$$log(\mu_i) = X_i^T\beta$$

where X_i is a predictor vector of the i^{th} observation and β is an unknown coefficients vector. The link function implies that

$$\mu_i = exp(X_i^T\beta)$$

Then the Likelihood function of β is defined as:

$$\mathcal{L}(\beta) = P(Y, \beta) = \prod_{i=1}^{n} e^{-\mu_i}\frac{\mu_i^{Y_i}}{Y_i!}$$

The score function is the derivitive of the log-likelihood function with respect to the parameter β. Hence it is first necessary to state the Log-likelyhood function:

$$\begin{aligned} l(\beta) &= log(\mathcal{L}(\beta)) \\ &= \sum_{i=1}^{n} Y_i \log(\mu_i) - \sum_{i=1}^{n} \mu_i - \sum_{i=1}^{n} \log(Y_i)! \\ &= \sum_{i=1}^{n} Y_i X_i^T\beta - \sum_{i=1}^{n} \exp(X_i^T\beta) - \sum_{i=1}^{n} \log(Y_i)! \end{aligned}$$

Then the Score function can be defined as:

$$\begin{aligned} s(\beta) &= \frac{\partial \log \mathcal{L}(\beta)}{\partial\beta} \\ &= \sum_{i=1}^{n} (Y_i - \exp(X_i^T\beta))X_i \end{aligned}$$

The MLE $\hat{\beta}$ is the solution of $s(\beta) = 0$ or equivalently

$$\hat{\beta} = argmax_\beta \, \mathcal{L}(\beta|X) = argmax_\beta \, [\sum_{i=1}^{n} Y_i X_i^T \beta - \sum_{i=1}^{n} \exp(X_i^T \beta)]$$

If this were to be solved by hand, the Newton-Raphson method could be utilised, which involves iteratively solving:
$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$
In R the MLE estimators for $\hat{\beta}$ are given as the coefficients of the estimate variable in the output of a GLM function call.

The Hessian function is a square matrix of second-order partial deritatives of the log-likelihood function.

$$\begin{aligned} H &= \frac{\partial^2}{\partial\beta\partial\beta^T} log(\mathcal{L}(\beta)) \\ &= -\sum_{i=1}^{n} X_i X_i^T exp(X_i^T \beta) \\ &= -X^T V X \end{aligned}$$

where $X = (X_1..., X_n)^T$ is a n x p design matrix and $V = diag(exp(X_1^T), ..., exp(X_n^T))$

From lectures it is known that the Poisson regression model has a limiting distribution in the form of either a Normal (Z) or T depending on whether the variance is known. In the scenario given, although variance is unknown, there are 100 observations, this means that by the central limit theorem, its limiting distribution can be summarised as either a normal Z distribution or T distribution. The decision was made to use a Z distribution, though it is worth noting that this does increase the chance of type I error slightly as there is a greater chance of rejecting a true null hypothesis. This is because the Z 5% critical value is 1.6449, whereas the T 5% critical value is 1.6602, both given to 4 decimal places.

Question 3

In this section the three candidate models will be compared using the Bayesian Information Criterion (BIC). This was chosen over Akaike Information Criterion (AIC) as it is slightly harsher on adding predictors to a model and in general, it is best to be as parsimonious as possible. The predicted value for observation Y_i based on the model is denoted by $g(\mathbb{E}[y_i])$. The coefficient of the corresponding predictor variable is denoted by β_i. For ease of model understanding the predictor variable will be renamed from $X_{1i}, ..., X_{ni}$ to: T_i for time, S_i is a binary indicator as to whether the shape is a cylinder, Tr_i for trial and A_i for age and β_0 denotes the intercept.
The first model considered is the full model:
$g(\mathbb{E}[y_i]) = \beta_0 + \beta_1 Tr_i + \beta_2 S_i + \beta_3 T_i + \beta_4 A_i$

The second model considered is the full model with the `Trial` predictor variable removed (Least significant value from the full model):
$g(\mathbb{E}[y_i]) = \beta_0 + \beta_1 S_i + \beta_2 T_i + \beta_3 A_i$

The final model considered is model2 with the `Age` predictor variable removed (Least significant value from model2):
$g(\mathbb{E}[y_i]) = \beta_0 + \beta_1 S_i + \beta_2 T_i$

```
#Load the dataset
data(blocks)
```

```
blocks <- blocks %>%
  #set categorical factors as factors
 mutate(Trial = factor(Trial)) %>%
 mutate(Shape = factor(Shape))

model1 <- glm(Number ~ Trial + Shape + Time + Age,data = blocks , family = poisson)
model1 %>%
  tidy() %>%
  #Need to specify dyplr select as otherwise R env tries to use Mass:select.
  dplyr::select(term, estimate, p.value)
```

```
## # A tibble: 5 x 3
##   term          estimate       p.value
##   <chr>            <dbl>         <dbl>
## 1 (Intercept)    1.38    0.00000000278
## 2 Trial2         0.0294  0.701
## 3 ShapeCylinder -0.334   0.0000370
## 4 Time           0.00410 0.00122
## 5 Age            0.135   0.0115
```

```
#Gives "best" other models as  Number ~  Shape + time + age
model2 <- glm(Number ~ Shape + Time + Age,data = blocks , family = poisson)
model2 %>%
  tidy() %>%
  dplyr::select(term, estimate, p.value)
```

```
## # A tibble: 4 x 3
##   term          estimate       p.value
##   <chr>            <dbl>         <dbl>
## 1 (Intercept)    1.39    0.00000000103
## 2 ShapeCylinder -0.334   0.0000371
## 3 Time           0.00408 0.00125
## 4 Age            0.135   0.0115
```

```
model3 <- glm(Number ~  Shape + Time,data = blocks , family = poisson)
model3 %>%
  tidy() %>%
  dplyr::select(term, estimate, p.value)
```

```
## # A tibble: 3 x 3
##   term          estimate   p.value
##   <chr>            <dbl>     <dbl>
## 1 (Intercept)    1.93    1.11e-157
## 2 ShapeCylinder -0.331   4.30e-  5
## 3 Time           0.00429 8.24e-  4
```

```
#Selecting the best model using BIC
summary <- data.frame(Model = c("Model 1", "Model 2", "Model 3"),
       BIC = c(BIC(model1), BIC(model2), BIC(model3)))

#Outputting the summary as a "nicer" to read table
kable(summary, "latex", booktabs = T) %>%
kable_styling(latex_options = "striped")
```

Model	BIC
Model 1	422.6346
Model 2	418.1764
Model 3	420.1000

BIC is calculated as a penalty term on the number of parameters k - the maximised value of the likelihood function: $BIC = log(n)k - 2log(\hat{\mathcal{L}})$, which demonstrates the lower the BIC value, the better the model fits the data. Since Model 2 has the lowest BIC value, it can be seen that this model is the best fit for the data. With the coefficients $\beta_0, ..., \beta_3$ demonstrated in the following model formula:
$g(\mathbb{E}[y_i]) = 1.391 - 0.334S_i + 0.004T_i + 0.135A_i$

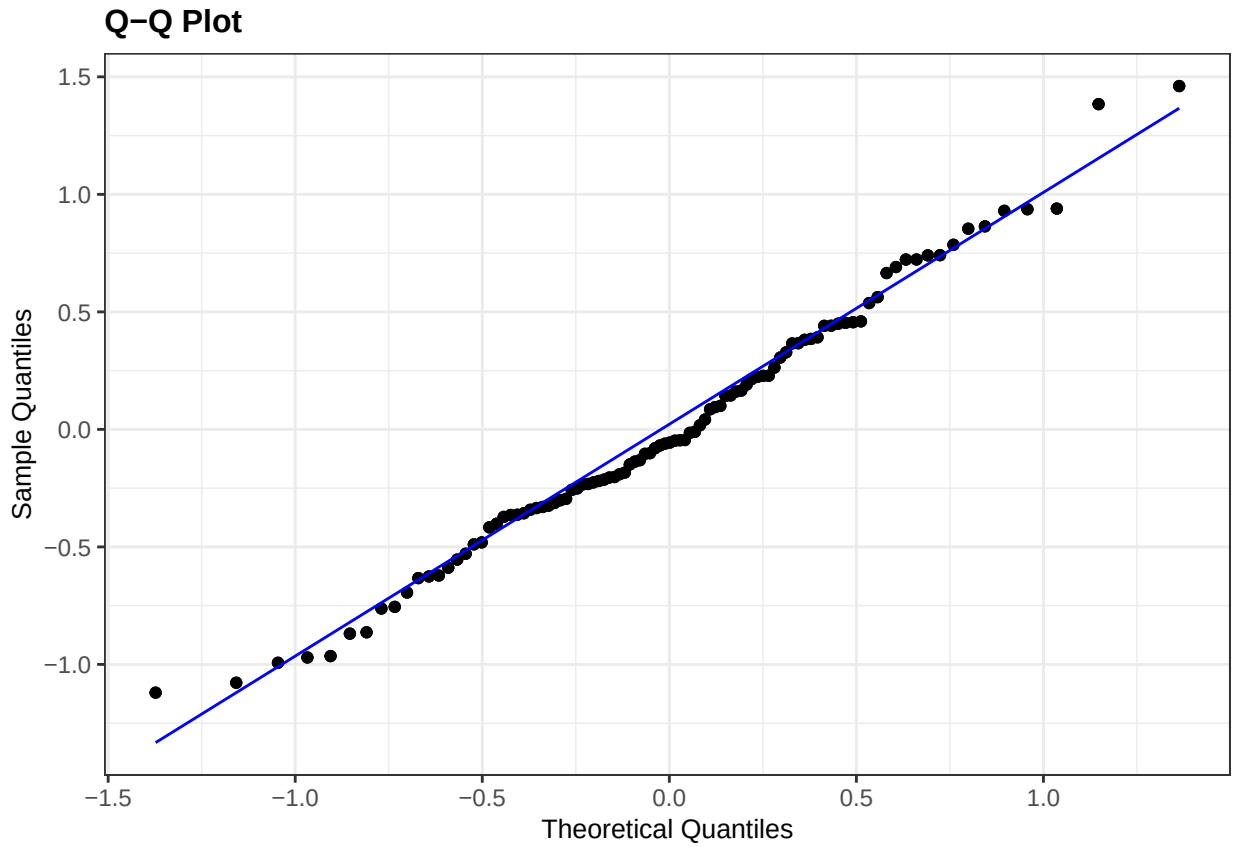
Question 4

To understand the residuals of the model and see how well the model fits the data, residual plots can be used to visualise certain aspects. The first aspect the residuals will be checked from is the a QQ-plot. This shows whether the residuals follow a normal distribution as it compares the theoretical quantiles of a normal distribution with those observed from the samples. If the points roughly follow a $y = x$ line then it demonstrates a good fit for the model.

The second aspect to be considered is the plot of residuals vs fitted (predicted values), this should be normally distributed around $y = 0$ as x increases and there should be no distinguishable trend in the plot.

The final part of analysis will showcase some of the benefits of the library `ggResidpanel`, demonstrating some other ways of looking at residuals which go beyond the standard four plots generated using plot(model_name) as this package contains functions for all of these as well as others.

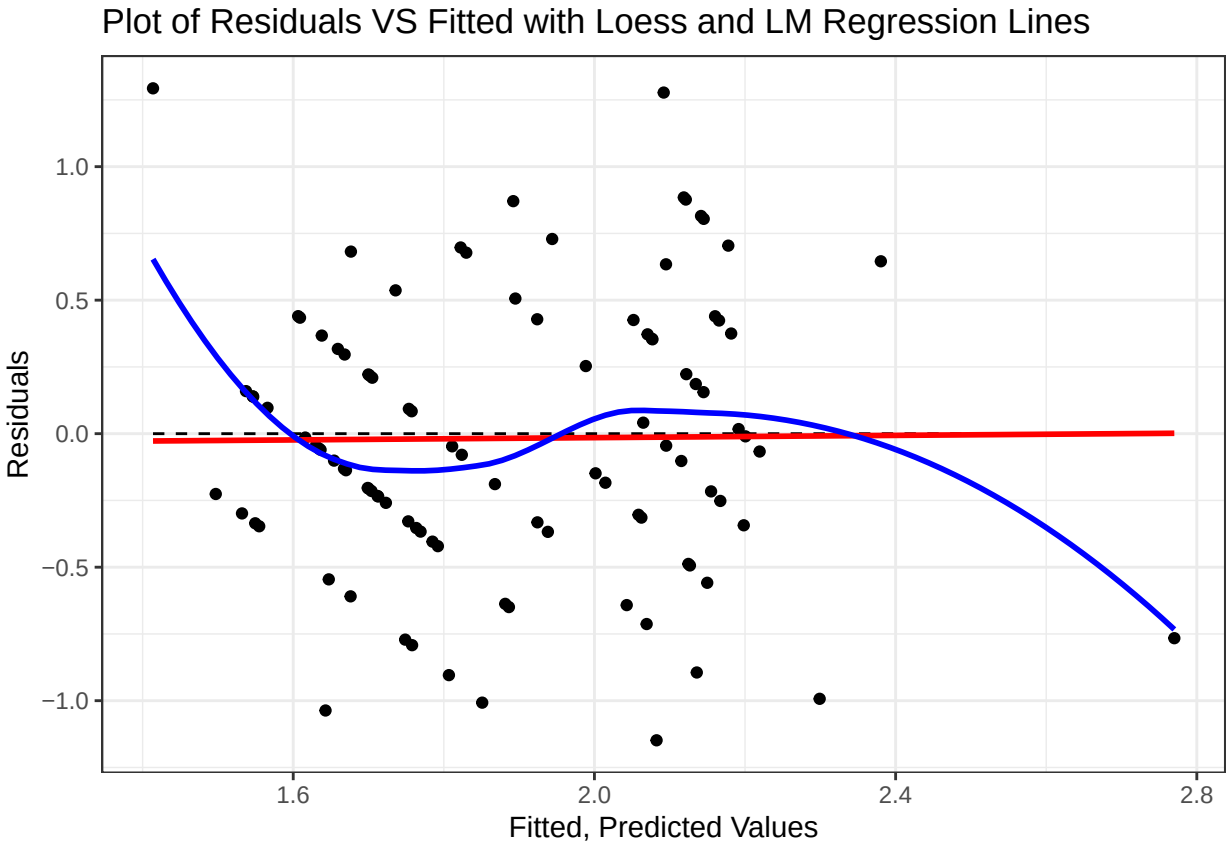
```
#QQPLOT - using the package ggResidpanle does this nicely in 1 line with a good range of options
resid_panel(model2, plots = "qq", type = "stand.pearson")
```



The QQ plot demonstrates the points lie close to the line $y = x$ which demonstrates that they tend to follow a normal distribution, even though they drift slightly at the tails, this is also demonstrated by the loess function in the next plot.

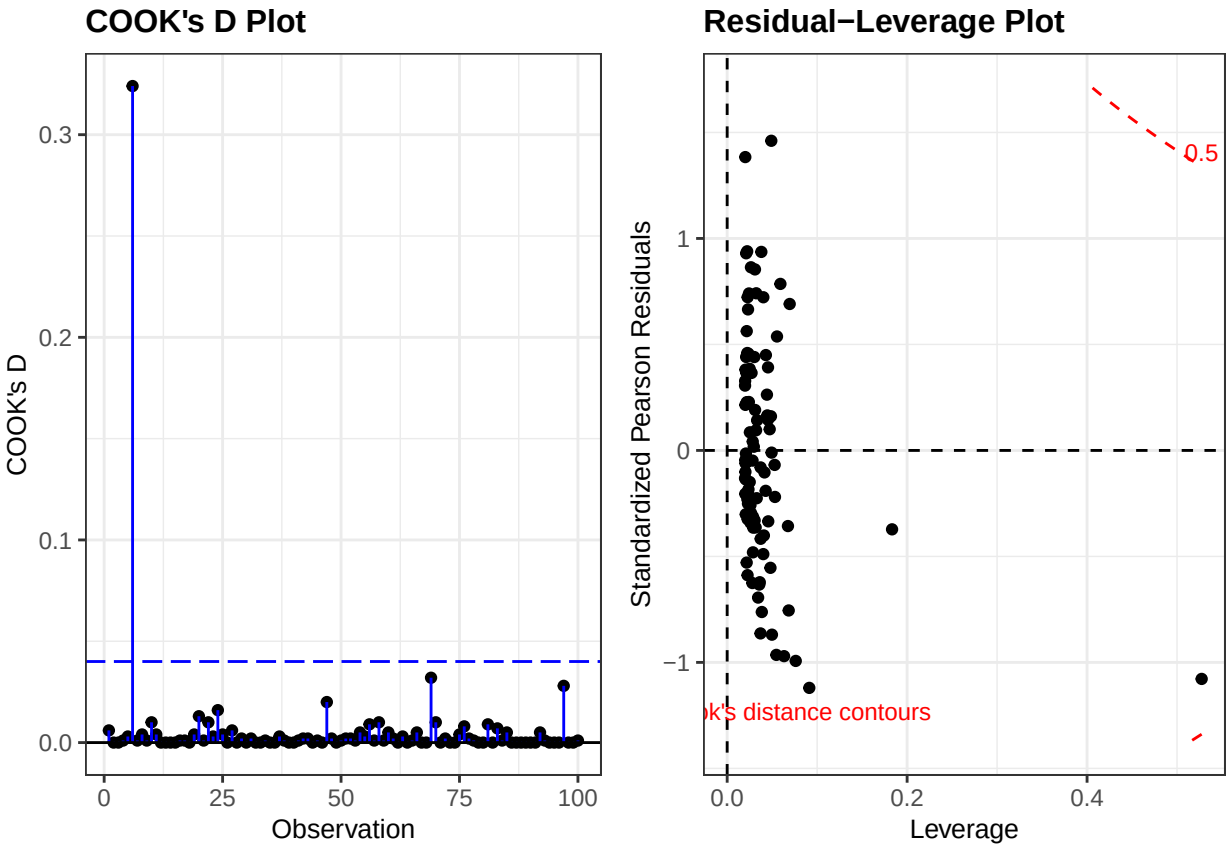
```
##Residuals VS Fitted, ggplot2 implementation
#could also use ggResidpanel with plots = "yvp"
#Main reason for ggplot is overlay two stats models
modf <- augment(model2)
ggplot(modf, aes(x = .fitted, y = .resid)) +
  geom_point() +
  geom_line(aes(y = 0), linetype = 2) +
  #Linear model
  stat_smooth(method = "lm", colour = "red",se = FALSE) +
  #Loess model overlayed
  stat_smooth(colour = "blue", se = FALSE) +
  labs(title = "Plot of Residuals VS Fitted with Loess and LM Regression Lines", y = "Residuals", x ="F
  theme_bw()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



In the residuals VS fitted plot there seems to be little variation in the middle part of the plot where the majority of the points are, this is skewed slightly at the x limits of the graph due to the few number of data points for the Loess regression (in blue). In general, as demonstrated by the linear regression line (in red), it shows no obvious correlation, which emphasises the model is a good fit for the data.

```
#Using the standardised Pearson Residuals to look at anomalous/important points
resid_panel(model2, plots = c("cookd", "lev"), type = "stand.pearson")
```



Finally, both the Cook's Distance and Residuals VS Leverage plot can be used to show both outliers and influencial points. The plots demonstrate the occurence of a point which seems anomalous and a couple of points with high influence. In general however, it demonstrates a good fit for our data. Going forwards the anomalous point could be removed for future model fitting on the data.