Assignment 2

Adam Sulak, Onder Akacik, Arda Ergin

Exercise 1

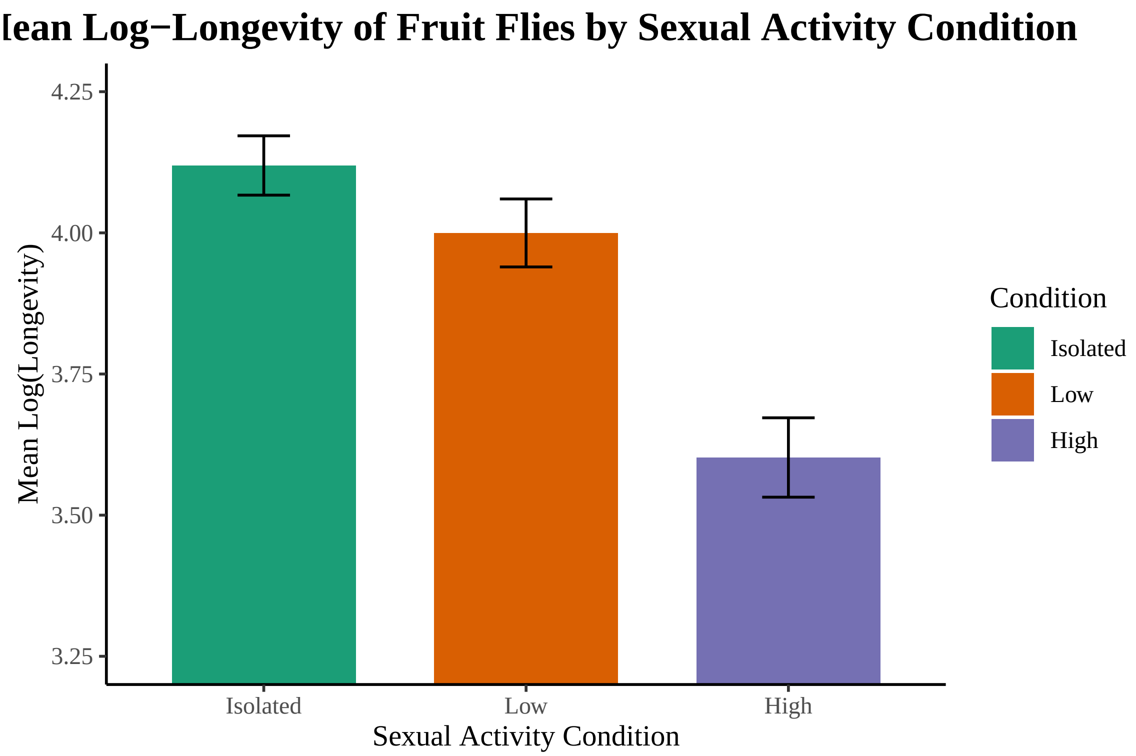
```r
data_raw <- read.table("fruitflies.txt", header = TRUE)
```

```r
data_raw <- data_raw %>% mutate(Activity = factor(
    dplyr::recode(activity, "isolated" = 0, "low" = 1, "high" = 2),
    levels = 0:2, labels = c("Isolated","Low","High")))
data_raw$loglongevity <- log(data_raw$longevity)
data <- subset(data_raw, select = -c(activity))
```

In addition to adding a `loglongevity` column through log-transforming the `longevity` variable, we have also properly factorized the `activity` variable. In the raw data set, “low” category was the baseline with 0, however it made more sense to have the “isolated” category as a baseline comparison with the categories “low” and “high”.

Question 1a)

```r
# Summary
data_summary <- data %>% group_by(Activity) %>% summarise(
    Mean_loglongevity = mean(loglongevity), SE_loglongevity = sd(loglongevity) / sqrt(n()),
    Mean_thorax = mean(thorax), SE_thorax = sd(thorax) / sqrt(n()))
# Plotting
data_summary %>% ggplot(aes(x = Activity, y = Mean_loglongevity, fill = Activity)) +
  geom_bar(stat = "identity", position = "dodge",width = 0.7) +
  geom_errorbar(aes(ymin = Mean_loglongevity - SE_loglongevity,
                    ymax = Mean_loglongevity + SE_loglongevity), width = 0.2) +
  labs(x = "Sexual Activity Condition",y = "Mean Log(Longevity)",fill = "Condition",
    title = "Mean Log-Longevity of Fruit Flies by Sexual Activity Condition") +
  coord_cartesian(ylim = c(3.25, 4.25)) + theme_classic(base_family = "Times") +
  theme(plot.title = element_text(face = "bold", hjust = 0.5, size = 15)) +
  scale_fill_brewer(palette = color_choice)
```



Considering our main interest is the condition variable of `activity`, which is a categorical variable, we have decided to use a bar plot to visualize the data. When looking at the bar plot with the confidence intervals, we can see that the individuals in the "high" activity condition, on average, seems to be living shorter lives compared to both "isolated" and "low" activity conditions. However, we can see that the difference between "isolated" and "low" activity conditions seem to be relatively little, probably not significant. The estimated longevity can be found by looking at the means for all groups: 4.12 (isolated), 4.00 (low), 3.60 (high).

```
model_simple <- lm(loglongevity ~ Activity, data = data)
results_model <- anova(model_simple)
summary_model <- summary(model_simple)
```

The appropriate statistical test here is a **one-way ANOVA**. The results of this analysis shows that the activity condition has a significant effect on (log-transformed) longevity, $F(2, 72) = 19.4$, $p < .001$. The longevity variable explained 35% of the total variance in the data.

Post-hoc analyses using `summary()` shows that while the "high" activity condition significantly differed from the baseline "isolated" condition, $t(72) = -5.95$, $p < .001$, there appears to be no significant difference between the "low" activity condition and the baseline "isolated" condition, $t(72) = -1.38$, $p = 0.17$. These results are in line with the visualization in the bar plot.

Question 1b)

```
model_with_cov <- lm(loglongevity ~ Activity + thorax, data = data)
model_comparison <- anova(model_simple, model_with_cov)
results_cov <- anova(model_with_cov)
```

When we include the `thorax` variable in our model as a covariate, and run a model comparison with just `activity` condition as our predictor, we can see that the addition of the covariate of thorax length to the model results in a significant improvement, $F(1, 71) = 94.4$, $p < .001$.

We can further see this when we run an **ANCOVA** with thorax length, finding that the effect of thorax length on longevity is significant, $F(1, 71) = 94.4$, $p < .001$, alongside the effect of activity $F(2, 71) = 44.6$, $p < .001$.

```
model_with_cov$coefficients
```

```
##  (Intercept)  ActivityLow ActivityHigh       thorax 
##        1.629       -0.124       -0.410        2.979
```

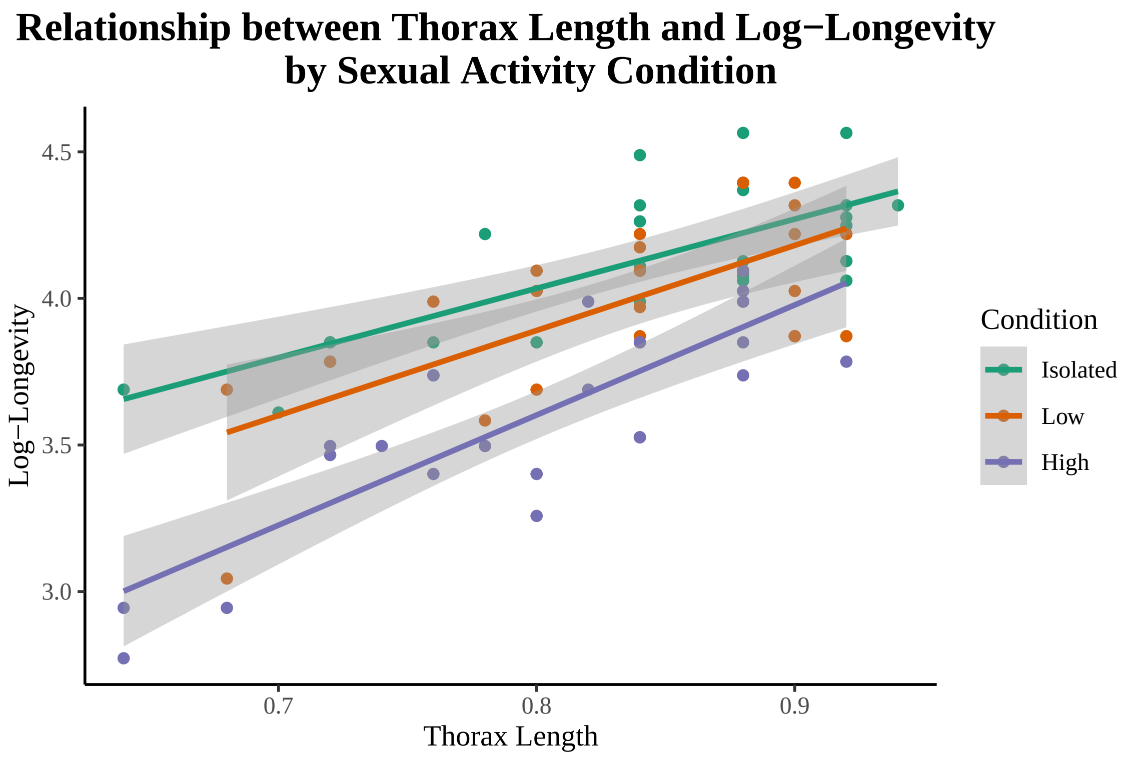
When we investigate the `lm()` output, we can see the estimated longevities for the three activity groups, using their coefficients and the thorax length coefficient (as an average).

For the "isolated" activity condition (derived from the intercept), the longevity is $1.629 + 2.979 = 4.6$, for "low" activity condition $-0.124 + 2.979 = 2.855$, and "high" activity condition $-0.410 + 2.979 = 2.569$.

Question 1c)

```
data %>% ggplot(aes(x = thorax, y = loglongevity, color = Activity)) +
  geom_point() + geom_smooth(method = "lm", se = TRUE) +
  labs(x = "Thorax Length", y = "Log-Longevity", color = "Condition",
    title = "Relationship between Thorax Length and Log-Longevity
    by Sexual Activity Condition") +
  theme_classic(base_family = "Times") +
  theme(plot.title = element_text(face = "bold", hjust = 0.5, size = 15)) +
  scale_color_brewer(palette = color_choice)
```

```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```



In a line plot, we can see the overall effect of thorax length through the slope, and the effect of condition by difference between the three lines. The graph clearly shows two main effects. **Thorax length positively influences longevity**. but since there seem to exist no slope differences between the three lines, it appears that there is no interaction between the two variables. This thinking of ours is further confirmed when we run the ANOVA with an interaction term.

```
model_with_interact = lm(loglongevity ~ Activity*thorax, data = data)
results_interact <- anova(model_with_interact)
```

The analysis results show that `activity:thorax` interaction is not significant, $F(2, 69) = 1.93$, $p = 0.15$. Hence, we can conclude that this influence of thorax length on longevity is similar under all three conditions of sexual activity.

Question 1d)

There is no 'wrong analysis' per se in the beginning. However, once we find thatthorax length was a significant factor, the correct thing to do is to include it in our model. Considering the model comparison, the model with the covariate is significantly more explanatory than the model without the covariate. Hence, thorax length significantly explains some part of the variance in the data, we should be including it in the model. Although we should not include the interaction term, as it is not significant.

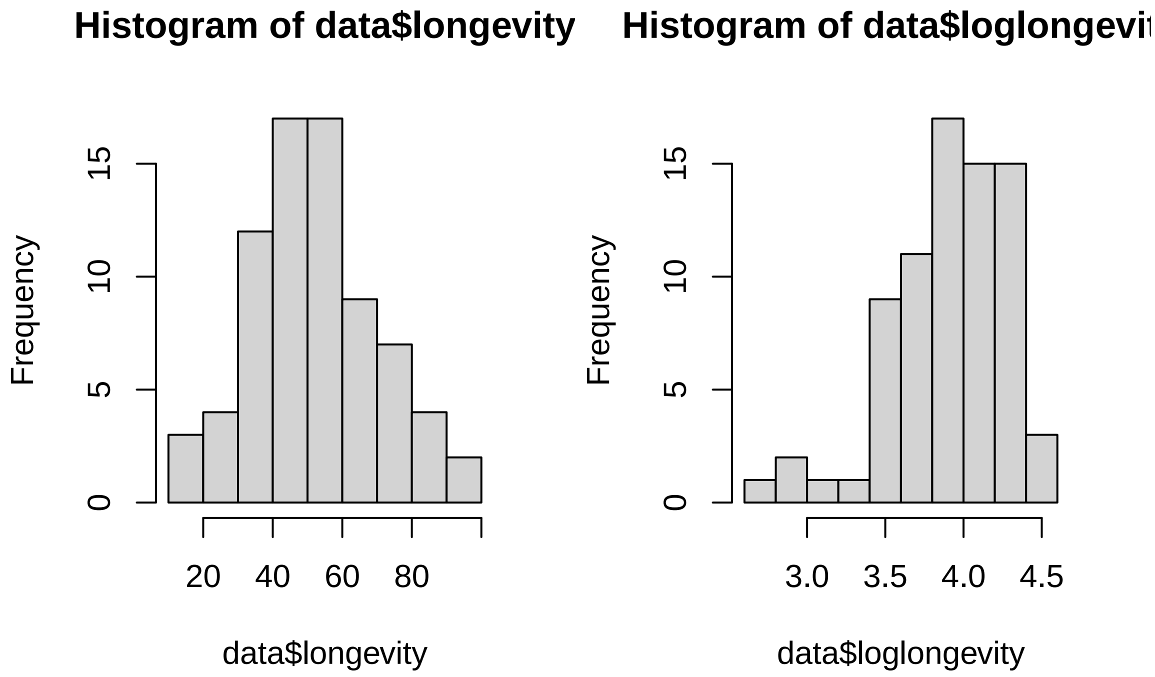
Question 1e)

```
model_with_cov_nolog = lm(longevity ~ Activity+thorax, data = data)
results_new <- anova(model_with_cov_nolog)
```

When we use "the number of days as the response, rather than its logarithm", there seems to be no difference in the results of our model, as both of the factors remain significant.

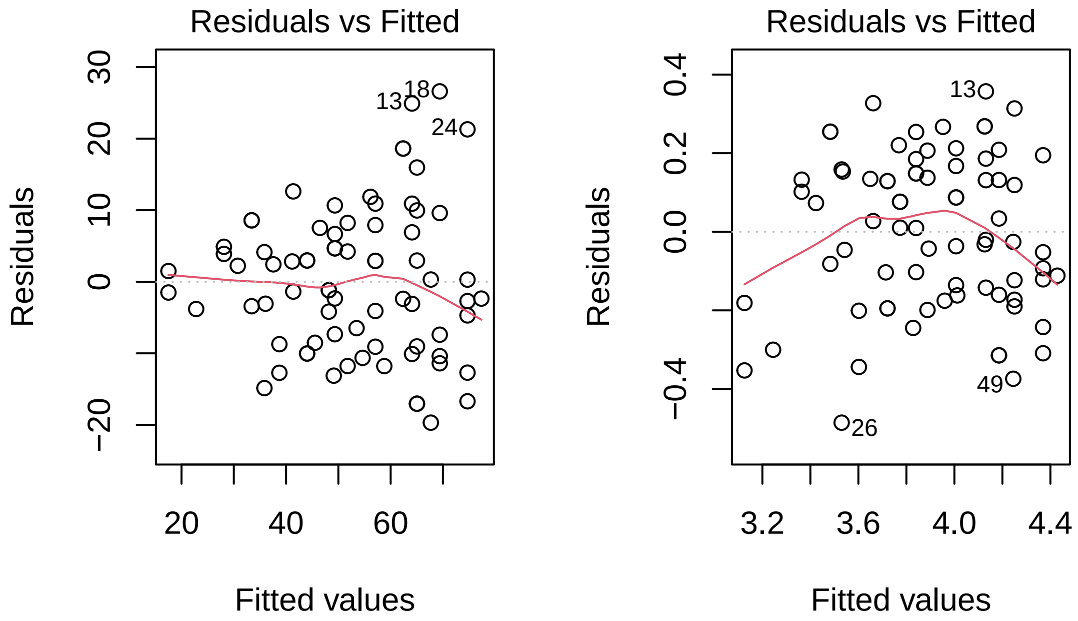
However, we need to consider the nature of the data and the model assumptions when deciding whether it is appropriate to long-transform a variable. Log-transformation makes sense when we have skewed data that violates normality, we have an issue with heteroskedasticity, or that we have issues in regards to outliers.

```
par(mfrow = c(1,2))
hist(data$longevity);hist(data$loglongevity)
```



When we investigate the histograms, it actually appears that the log-transformation skews the actually normally distributed variable, which is not a good thing considering ANOVA is relatively sensitive to normality violations.

```
par(mfrow = c(1,2))
plot(model_with_cov_nolog, which = 1)
plot(model_with_cov, which = 1)
```



In terms of checking heteroskedasticity through the residuals vs. fitted values, we can see that there seems to be a systematic issue with the non-log-transformed variable, as we can see a clear difference in error variances across the fitted values. We can see this when looking into the funnel shape of the points. The error variances are more equally distributed for the log-transformed data.

Overall, to make a conclusion, we can say that while log-transformation causes an issue with normality, it helps with heteroskedasticity. We can consider the issue with normality a little bit less serious than heteroskedasticity, and say that the log-transformation was indeed a good decision.

Exercise 2

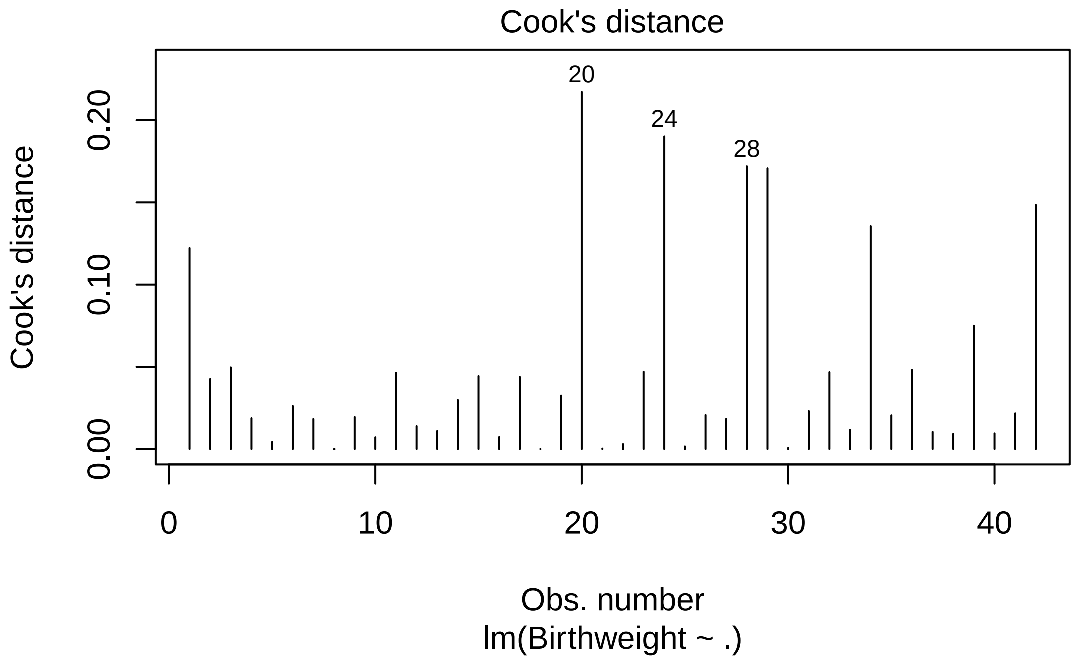
```r
data <- read.csv("Birthweight.csv")
```

Question 2a)

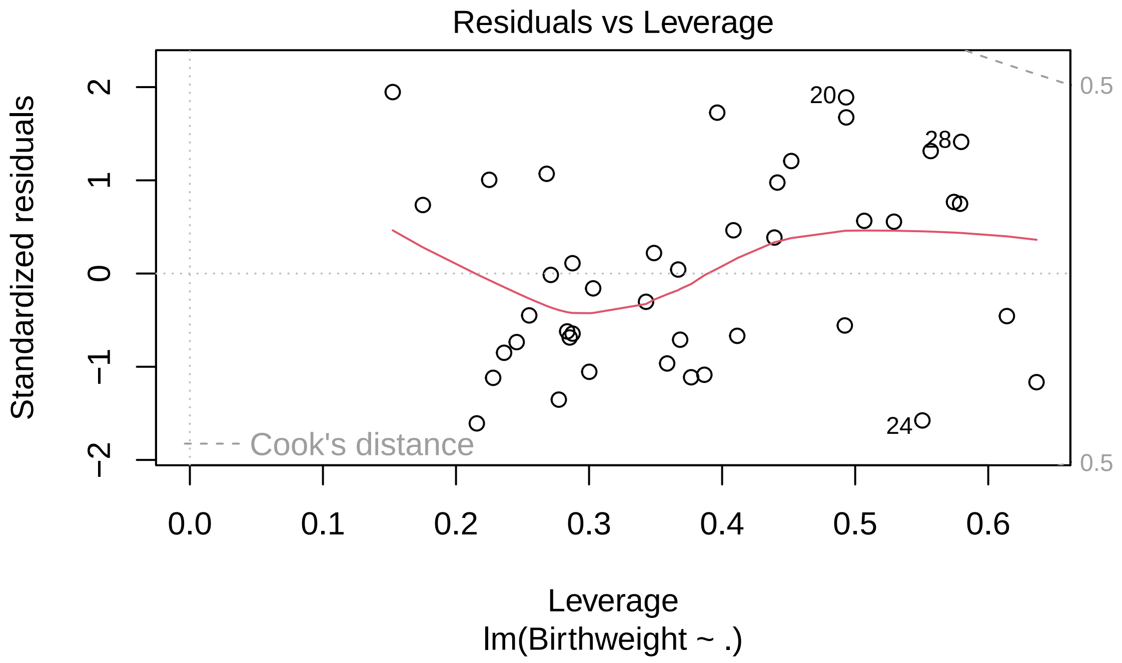
```r
model_full <- lm(Birthweight ~ ., data = data)
```

In order to investigate "the problem of potential and influence points", we can first take a look at the diagnostic plots.

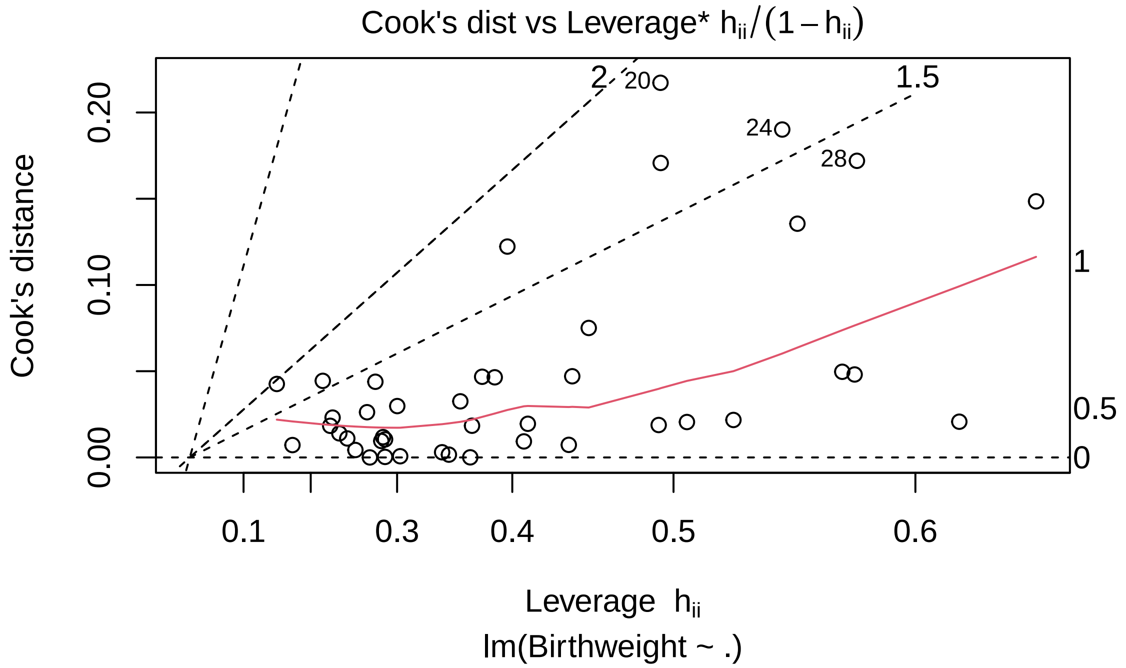
```r
plot(model_full, which=4)
```



```r
plot(model_full, which=5)
```

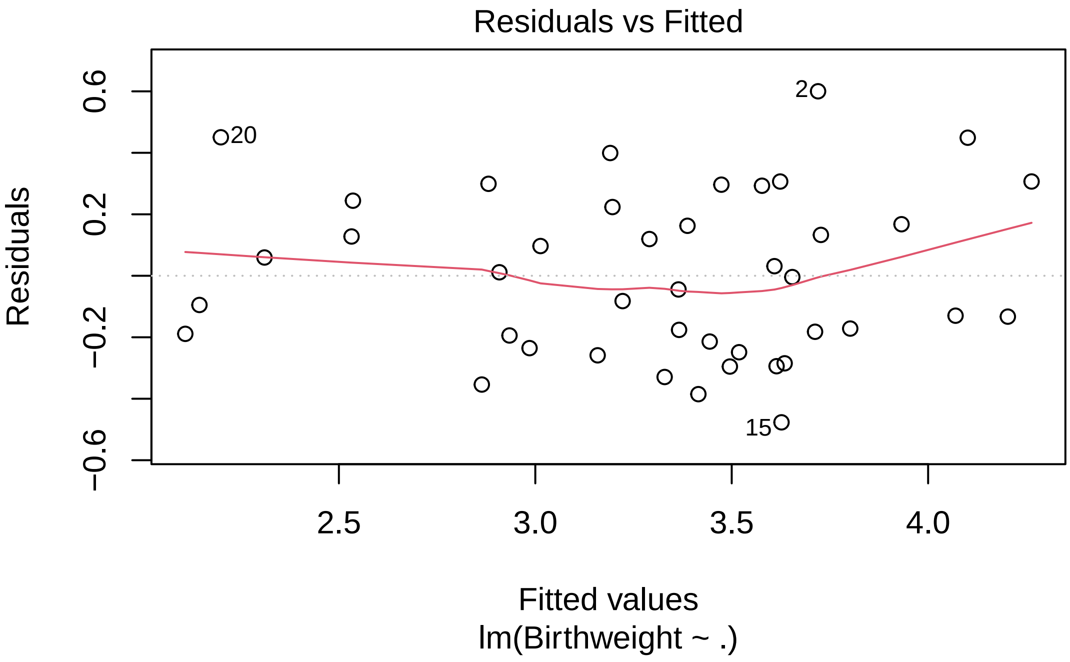


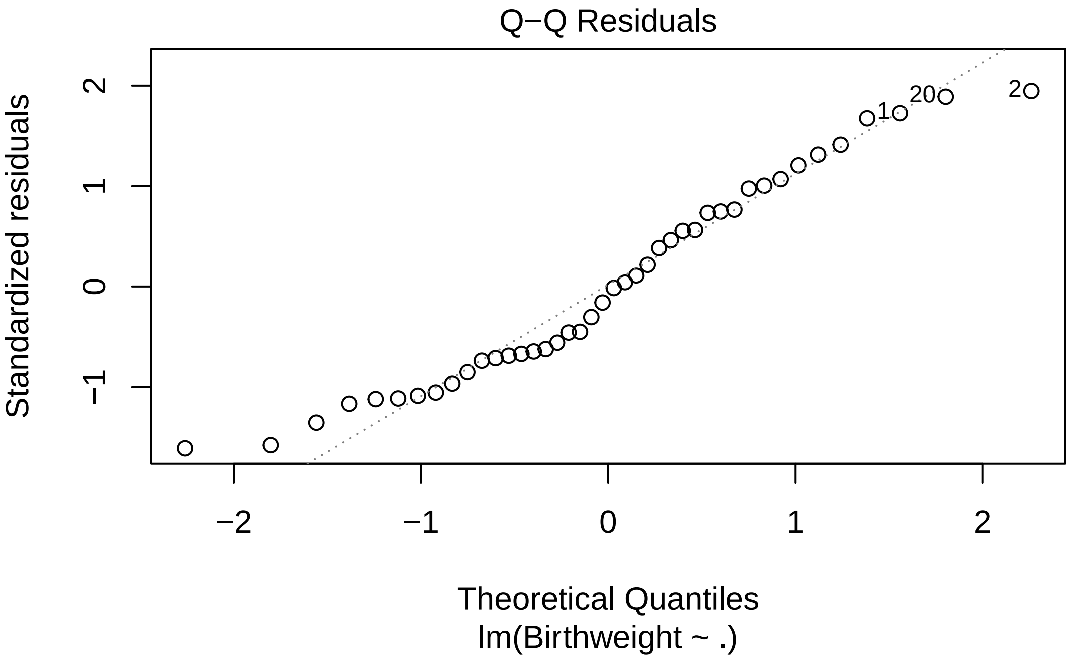
```r
plot(model_full, which=6)
```

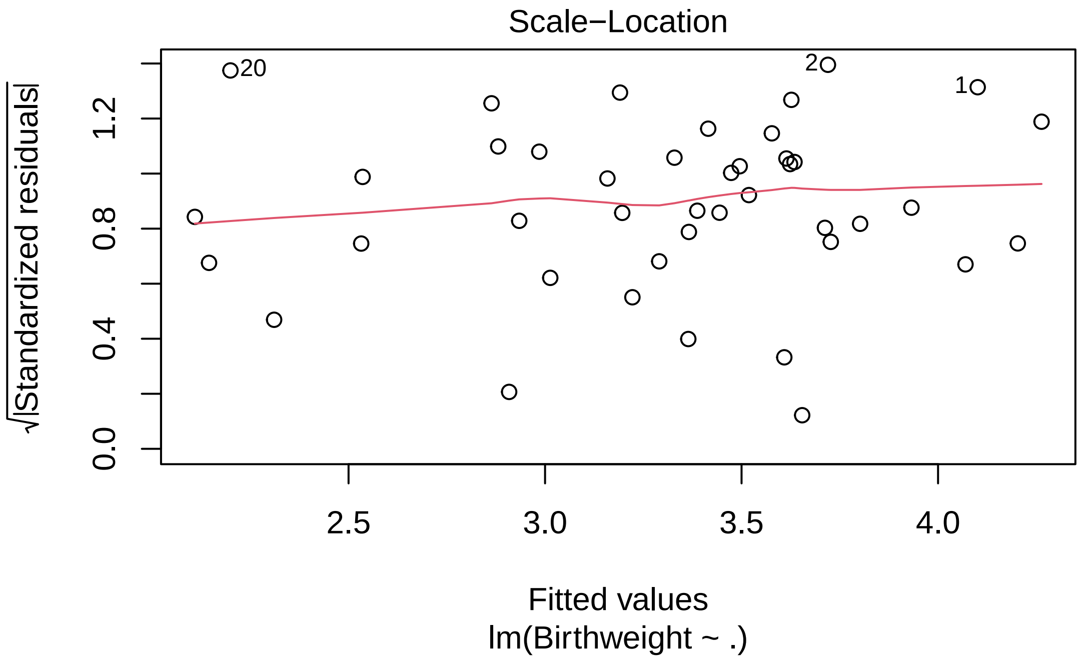


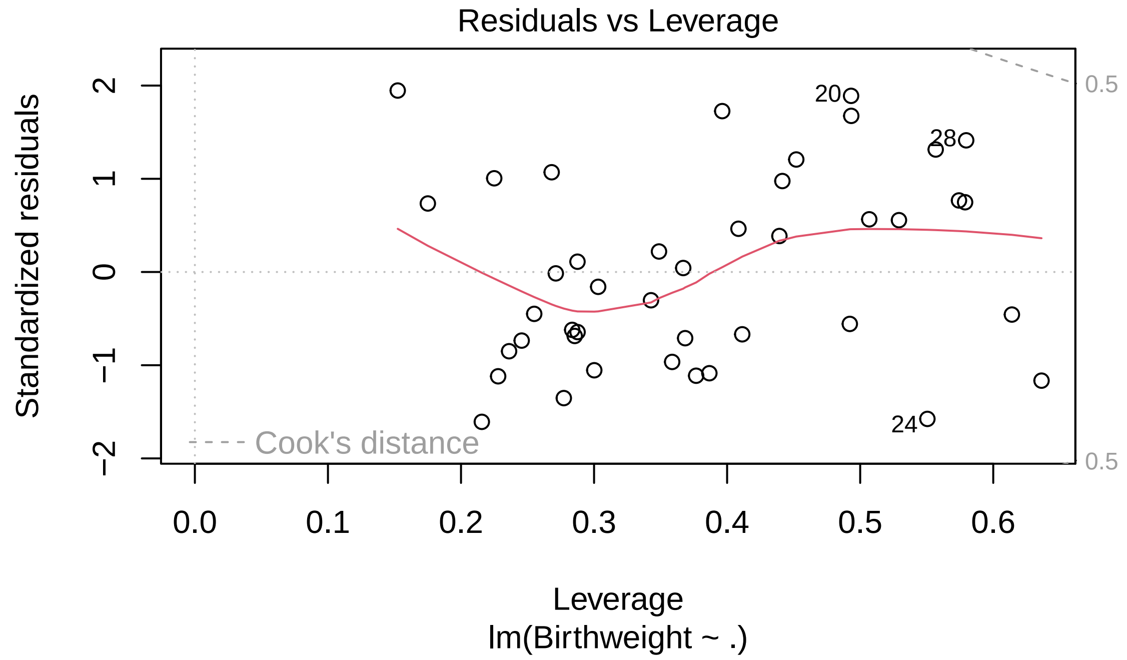
These plots reveal

```r
# par(mfrow=c(2,2))
plot(model_full)[1]
```


Residuals vs Fitted
Residuals
0.6
0.2
−0.2
−0.6
2.5
3.0
3.5
4.0
20
2
15
Fitted values
lm(Birthweight ~ .)


Q−Q Residuals
Standardized residuals
2
1
0
−1
−2
−1
0
1
2
1
20
2
Theoretical Quantiles
lm(Birthweight ~ .)





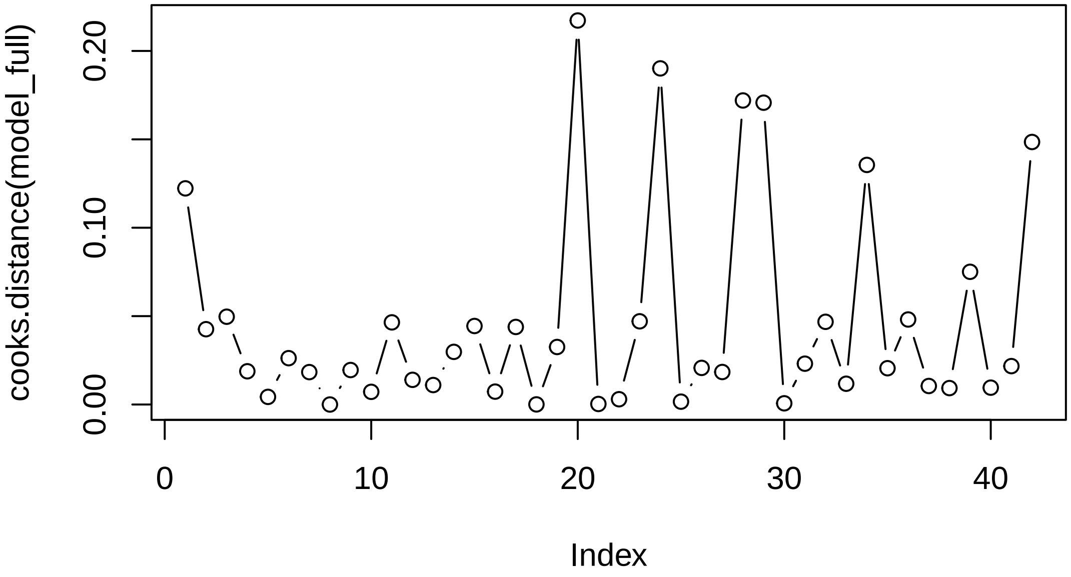
```
## NULL
```

```
#
resid_ordered <- order(abs(residuals(model_full)))
```

```
u = rep(0,length(resid_ordered))
u[length(resid_ordered)] = 1
#
cooks.distance(model_full)
```

```
##        1        2        3        4        5        6        7        8
## 1.22e-01 4.26e-02 4.97e-02 1.88e-02 4.32e-03 2.62e-02 1.83e-02 5.17e-06
##        9       10       11       12       13       14       15       16
## 1.95e-02 7.17e-03 4.65e-02 1.40e-02 1.10e-02 2.98e-02 4.44e-02 7.29e-03
##       17       18       19       20       21       22       23       24
## 4.39e-02 6.63e-05 3.25e-02 2.17e-01 3.08e-04 3.00e-03 4.71e-02 1.90e-01
##       25       26       27       28       29       30       31       32
## 1.62e-03 2.07e-02 1.84e-02 1.72e-01 1.71e-01 6.89e-04 2.31e-02 4.68e-02
##       33       34       35       36       37       38       39       40
## 1.18e-02 1.36e-01 2.05e-02 4.81e-02 1.05e-02 9.29e-03 7.51e-02 9.54e-03
##       41       42
## 2.17e-02 1.49e-01
```

```
plot(cooks.distance(model_full), type="b")
```



```
#forbeslm_42 = lm(y~x+u11); summary(forbeslm11)
```

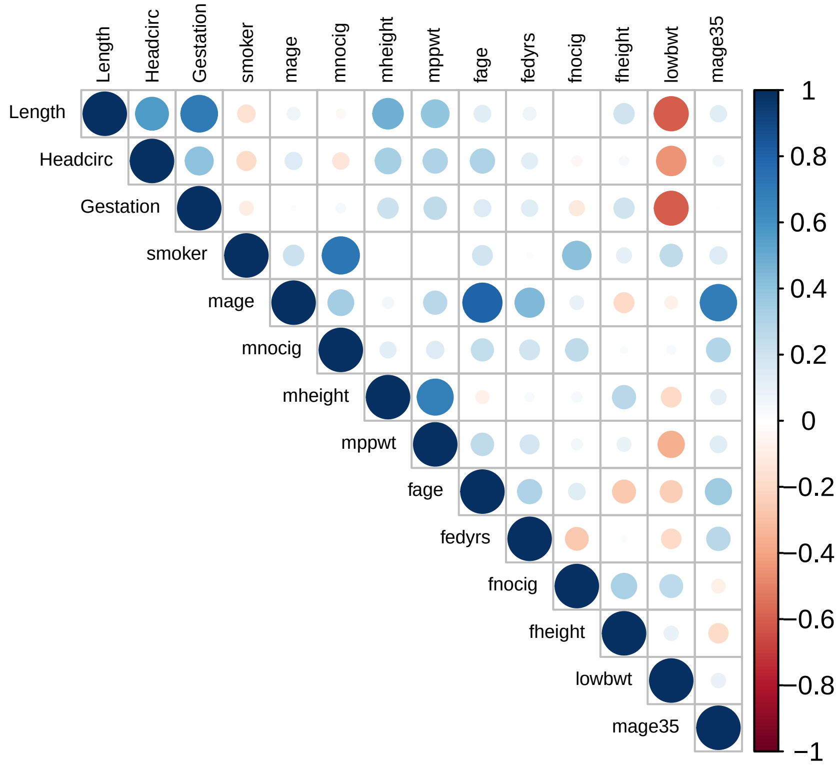
Investigating multi-collinearity:

```
car::vif(model_full)
```

```
##         ID    Length  Headcirc Gestation    smoker      mage    mnocig   mheight
##      1.55      4.76      1.89      2.65      2.82      9.90      2.87      3.35
##     mppwt      fage    fedyrs    fnocig   fheight    lowbwt    mage35
##      2.53      6.74      1.73      2.04      1.76      3.00      4.34
```

When we look into `car::vif()`, if we follow the rule of thumb, we see that no variable has a VIF value that is higher than 5. Hence, we do not seem to have any issues regarding multi-collinearity.

```r
# Calculate correlation matrix
cor_matrix <- cor(data[, -c(1,3)]) # Exclude response variable Birthweight

# Plot the correlation matrix
corrplot(cor_matrix,
         method = "circle", type = "upper", tl.col = "black", tl.cex = 0.6)
```



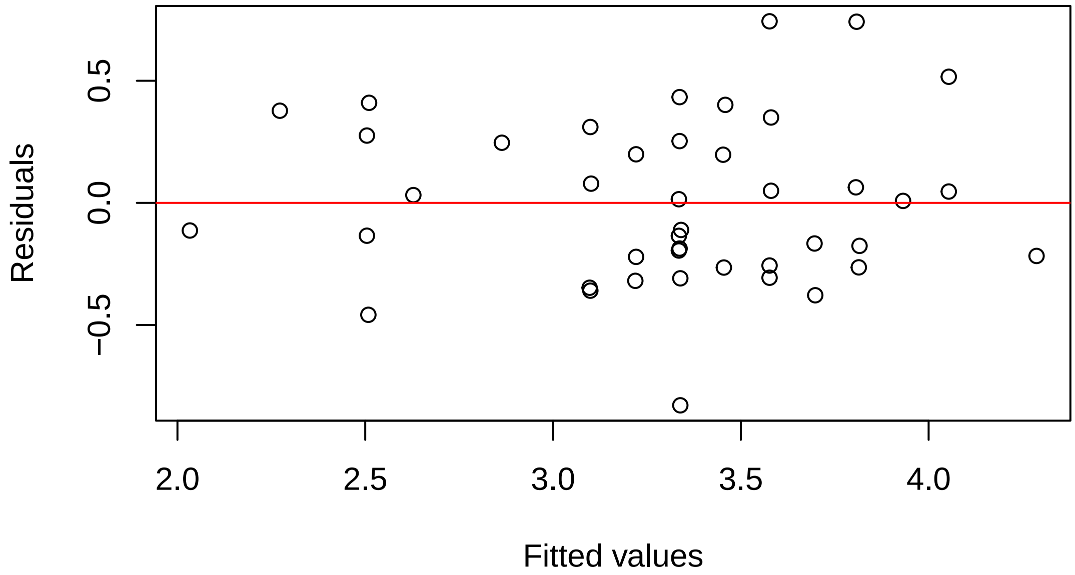
Question 2b)

```r
# step_down_model <- step(model, direction = 'backward')
model <- lm(
    Birthweight ~ Length + Headcirc + Gestation + mage + mnocig + mheight + mppwt + fage + fedyrs + fnoc
    data = data
)
# remove fage
model_1 <- lm(
    Birthweight ~ Length + Headcirc + Gestation + mage + mnocig + mheight + mppwt + fedyrs + fnocig + fl
    data = data
)
```

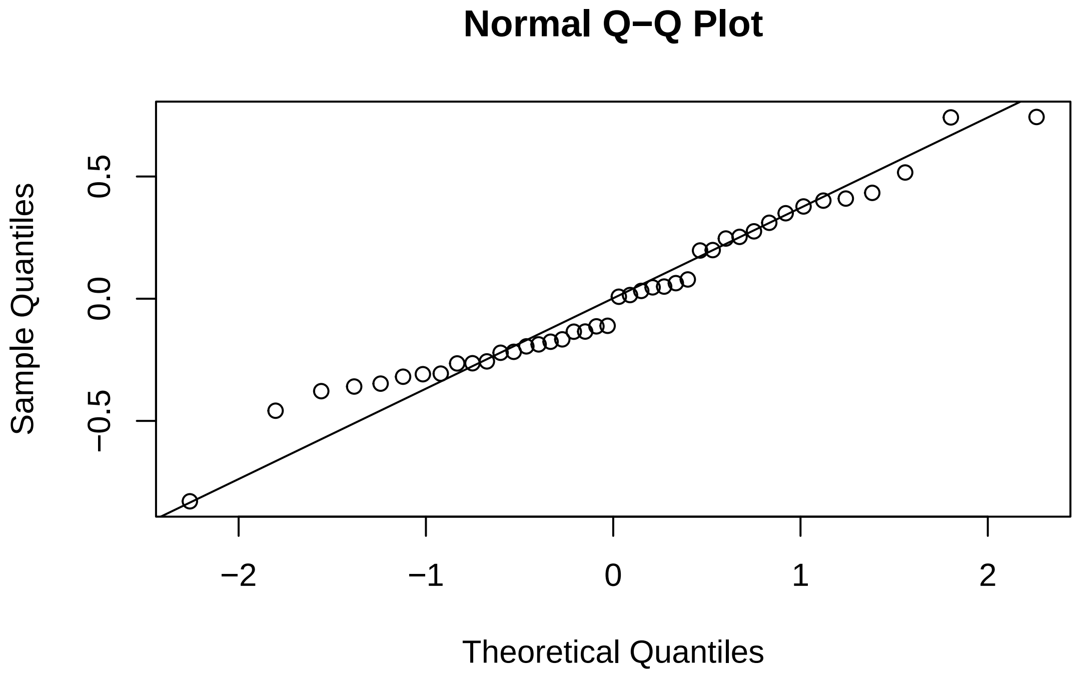
```r
# remove mheight
model_2 <- lm(
    Birthweight ~ Length + Headcirc + Gestation + mage + mnocig + mppwt + fedyrs + fnocig + fheight,
    data = data
)
# remove fedyrs
model_3 <- lm(
    Birthweight ~ Length + Headcirc + Gestation + mage + mnocig + mppwt + fnocig + fheight,
    data = data
)
# remove fnocig as it's not significant and Adjusted R-squared is comparable
model_4 <- lm(
    Birthweight ~ Length + Headcirc + Gestation + mage + mnocig + mppwt + fheight,
    data = data
)

# remove mnocig as it's not significant and Adjusted R-squared is comparable
model_5 <- lm(
    Birthweight ~ Length + Headcirc + Gestation + mage + mppwt + fheight,
    data = data
)

# remove length as it's not significant and Adjusted R-squared is comparable
model_6 <- lm(
    Birthweight ~ Headcirc + Gestation + mage + mppwt + fheight,
    data = data
)

# remove fheight as it's not significant and Adjusted R-squared is comparable
model_7 <- lm(
    Birthweight ~ Headcirc + Gestation + mage + mppwt,
    data = data
)

# remove mage as it's not significant and Adjusted R-squared is comparable
model_8 <- lm(
    Birthweight ~ Headcirc + Gestation + mppwt,
    data = data
)


# remove mppwt as it's not significant and Adjusted R-squared is comparable
step_down_model <- lm(
    Birthweight ~ Headcirc + Gestation,
    data = data
)
```

- Tests below are done for the model assumption..

```r
# linearity
plot(predict(step_down_model), residuals(step_down_model), xlab = "Fitted values", ylab = "Residuals")
abline(h = 0, col = "red")
```



```r
# normality of residuals
qqnorm(residuals(step_down_model))
qqline(residuals(step_down_model))
```



Question 2c)

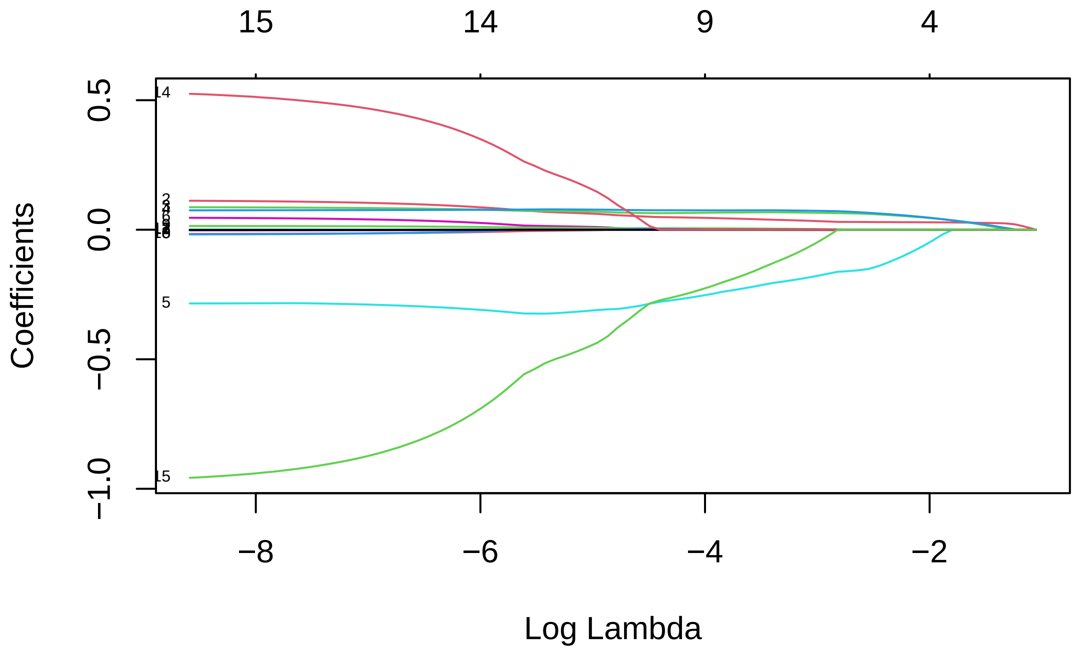
```
head_mean <- mean(data$Gestation)
gest_mean <- mean(data$Headcirc)

df <- data.frame(Gestation = gest_mean, Headcirc = head_mean)

ci <- predict(step_down_model, newdata = df, interval = 'confidence')
pi <- predict(step_down_model, newdata = df, interval = 'prediction')

ci; pi
```

```
##    fit  lwr upr
## 1 3.32 2.94 3.7

##    fit  lwr  upr
## 1 3.32 2.53 4.11
```

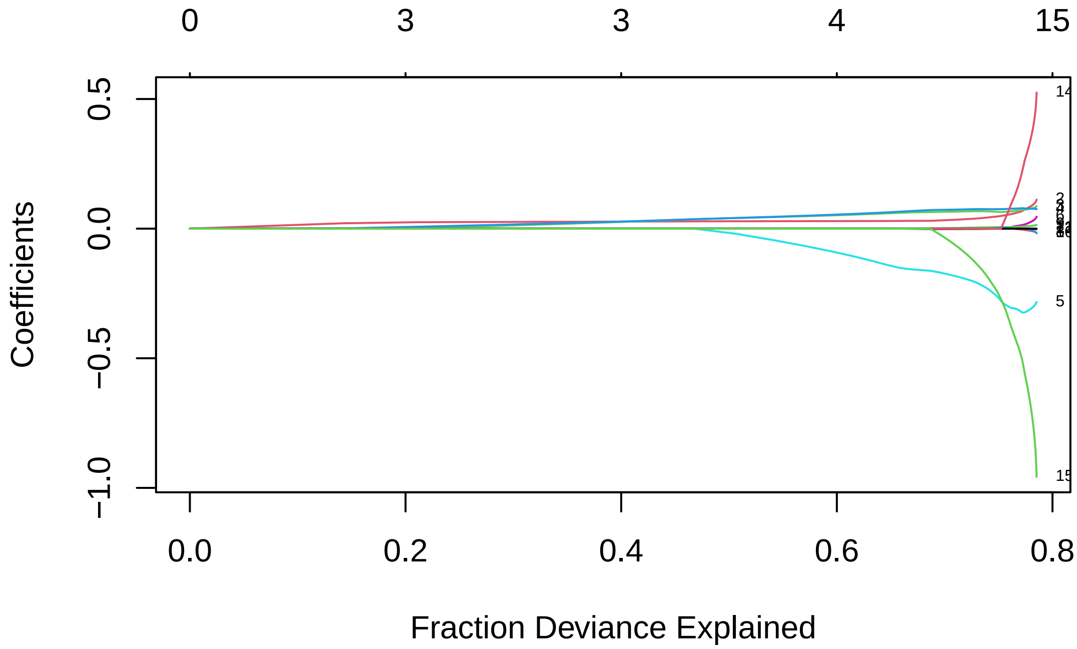
Question 2d)

In order to asses accuracy of LASSO method we create training and testing data sets. Then we use training data to fit the model using LASSO method (alpha = 1) and then determine optimal lambda value by performing cross-validation. We then compare resulting MSE values to determine if model is better than the one obtained by step-down method.

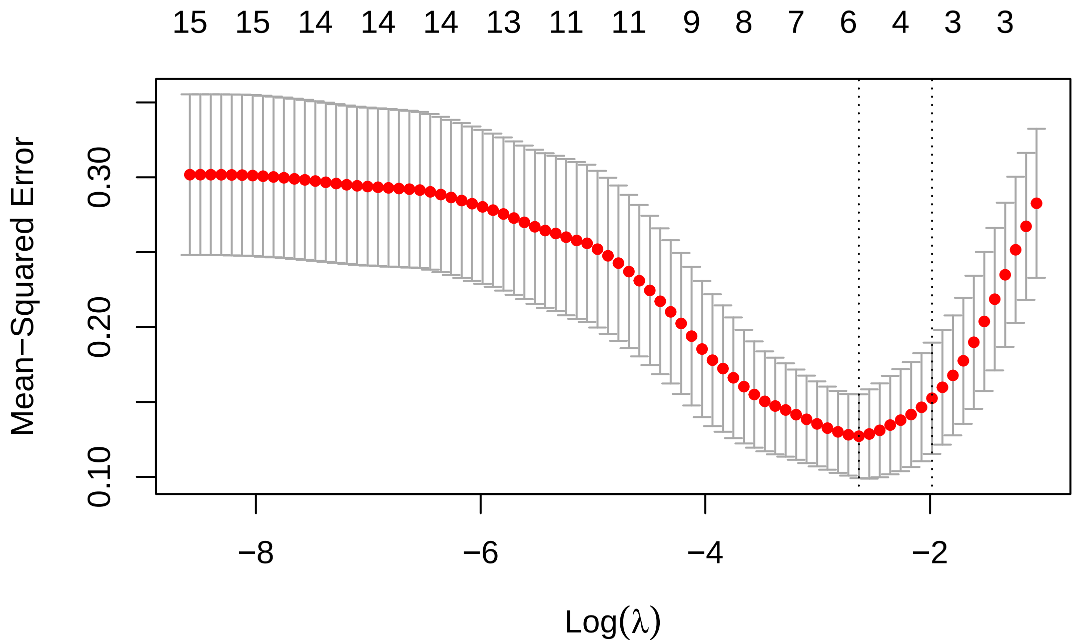
```
x <- subset(data, select = -Birthweight)
y <- data$Birthweight

train=sample(1:nrow(x),0.67*nrow(x))
x_train=x[train,]; y_train=y[train]
x_test=x[-train,]; y_test = y[-train]

y_predict_lm=predict(step_down_model,newdata=data[-train,]) # predict for the test rows
mse_lm=mean((y_test-y_predict_lm)^2)# prediction quality by the linear model

lasso_model <- glmnet(x_train, y_train, alpha = 1)

lasso_cv <- cv.glmnet(
    as.matrix(x_train),
    y_train,
    alpha = 1,
    type.measure="mse",
    nfolds=5
)

lasso_pred1=predict(lasso_model,s=lasso_cv$lambda.min,newx=as.matrix(x_test))
lasso_pred2=predict(lasso_model,s=lasso_cv$lambda.1se,newx=as.matrix(x_test))
mse1_lasso=mean((y_test-lasso_pred1)^2)
mse2_lasso=mean((y_test-lasso_pred2)^2)


plot(lasso_model, xvar="lambda", label = TRUE)
```



```r
plot(lasso_model, xvar="dev", label = TRUE)
```



```
plot(lasso_cv)
```



```
coef(lasso_cv, s= lasso_cv$lambda.1se)
```

```
## 16 x 1 sparse Matrix of class "dgCMatrix"
##                   s1
## (Intercept) -1.3608
## ID            .
## Length       0.0287
## Headcirc     0.0444
## Gestation    0.0456
## smoker      -0.0443
## mage          .
## mnocig        .
## mheight       .
## mppwt         .
## fage          .
## fedyrs        .
## fnocig        .
## fheight       .
## lowbwt        .
## mage35        .
```

```
# coef(lasso_cv_model, s= lasso_cv_model$lambda.min)

mse_lm; mse1_lasso; mse2_lasso
```

```
## [1] 0.119
## [1] 0.178
```

```
## [1] 0.254
```

After running experiment multiple times we determine that step-down model performs better with average MSE value of 0.15 compared to LASSO model with lambda = lambda.1se which had average MSE value of 0.2.

Question 2e)

```
data_new <- data[,c("lowbwt","Gestation","smoker","mage35")]
data_new$Gestation <- as.numeric(data_new$Gestation)
data_new$smoker <- factor(data_new$smoker, levels = 0:1, labels = c("No","Yes"))
data_new$mage35 <- factor(data_new$mage35, levels = 0:1, labels = c("No","Yes"))
```

To investigate "Do *smoking mothers* seem to have lighter babies?" and "Do *older mothers* seem to have lighter babies?", we can check the **Crosstabs**:

```
xtabs(lowbwt~smoker+mage35,data=data_new) / xtabs(~smoker+mage35,data=data_new)
```

```
##       mage35
## smoker     No    Yes
##    No  0.0526 0.0000
##    Yes 0.2105 0.3333
```

```
# Aggregate over mage35 and smoker
xtabs(lowbwt~smoker,data=data_new) / xtabs(~smoker,data=data_new)
```

```
## smoker
##    No   Yes
## 0.050 0.227
```

```
xtabs(lowbwt~mage35,data=data_new) / xtabs(~mage35,data=data_new)
```

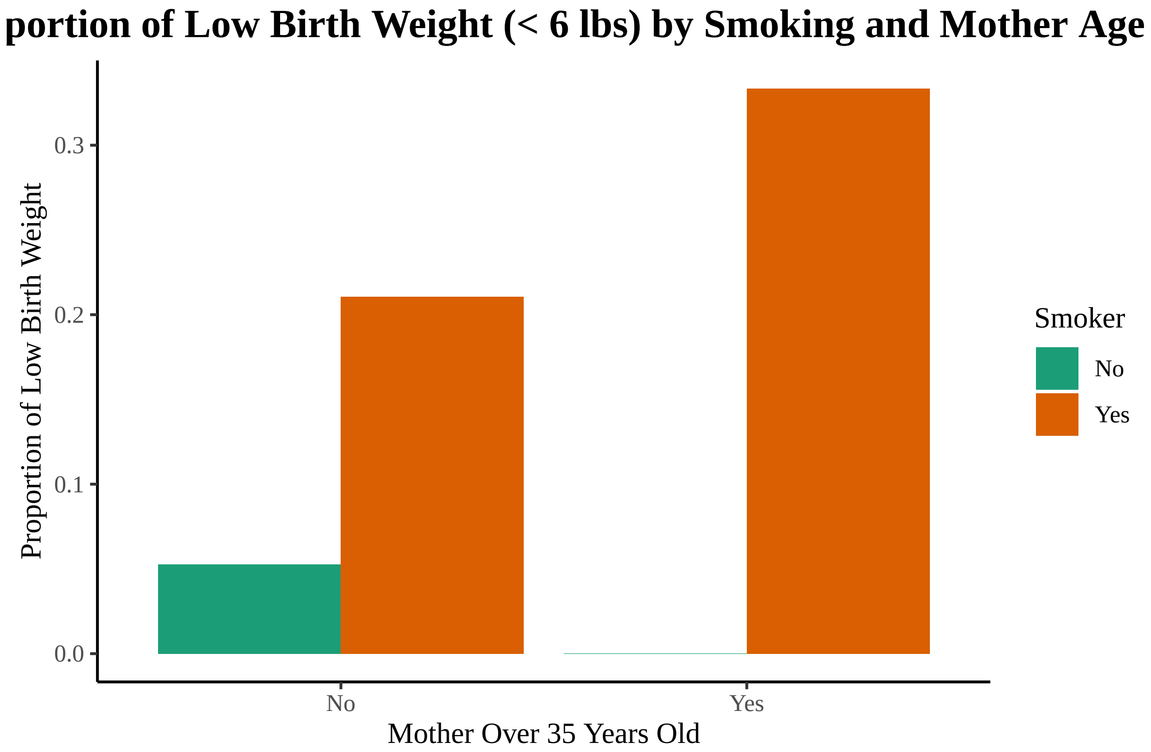
```
## mage35
##    No   Yes
## 0.132 0.250
```

```
# Factorizing the DV after the crosstabs:
data_new$lowbwt <- factor(data_new$lowbwt, levels = 0:1, labels = c("No","Yes"))
```

To get an even more visual look, we can further plot these proportion of low birth weight by *Smoking* and *Mother Age*.

```
aggregated_data <- data %>%
  group_by(smoker, mage35) %>%
  summarise(proportion_dv = mean(lowbwt), .groups = 'drop')
aggregated_data$smoker <- factor(aggregated_data$smoker, levels = 0:1, labels = c("No","Yes"))
```

```r
aggregated_data$mage35 <- factor(aggregated_data$mage35, levels = 0:1, labels = c("No","Yes"))

ggplot(aggregated_data, aes(x = mage35, y = proportion_dv, fill = as.factor(smoker))) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Mother Over 35 Years Old", y = "Proportion of Low Birth Weight", fill = "Smoker",
       title = "Proportion of Low Birth Weight (< 6 lbs) by Smoking and Mother Age") +
  theme_classic(base_family = "Times") + scale_fill_brewer(palette = color_choice) +
  theme(plot.title = element_text(face = "bold", hjust = 0.5, size = 15))
```



Based on the crosstabs and the bar plot:
1) It is immediately clear that there is *a higher proportion of low birth weight* in babies given birth by smoking mothers, compared to babies given birth by non-smoker mothers. If we aggregate over `mage35` variable in crosstabs, we can see that the difference between the proportions of 0.05 (non-smoker) and 0.23 (smoker) appears to be quite considerable. Hence, smoking mothers indeed seem to have lighter babies.
2) It appears "older mothers seem to have lighter babies", however the *main effect* is not as apparent as being a smoker.
- **Within the smoker mothers**, age of the mother seems to influence low birth weight: compared to smoker mothers that are younger than 35, smoker mothers that are 35 or older seem to have a higher proportion of Low Birth Weight in babies.
- **Within the non-smoker mothers**, age of the mother does not seem to influence low birth weight: there appears to be no difference between non-smoker mothers that are younger than 35 and non-smoker mothers that are 35 or older in Low Birth Weight in babies (since the difference in proportions of 0.05 and 0 seem negligible). - If we aggregate over `smoker` variable in crosstabs, we can see that there seems to be a difference between the proportions of 0.13 (under 35 y.o.) and 0.25 (over 35 y.o.). Yet, we need further testing to see whether this difference is actually significant. Although, considering the above points, it is possible that there is a `smoker*mage35` interaction.

Question 2f)

```r
model_glm_0 <- stats::glm(lowbwt ~ 1, family=binomial("logit"), data=data_new)
model_glm_smoker <- stats::glm(lowbwt ~ smoker, family=binomial("logit"), data=data_new)
model_glm_mage35 <- stats::glm(lowbwt ~ mage35, family=binomial("logit"), data=data_new)
model_glm_both <- stats::glm(lowbwt ~ smoker + mage35, family=binomial("logit"), data=data_new)
# Testing the Predictors through model comparison:
results_1 <- anova(model_glm_0, model_glm_smoker, test="Chisq")
results_2 <- anova(model_glm_0, model_glm_mage35, test="Chisq")
# Odds
exp(model_glm_smoker$coefficients[2])
```

```
## smokerYes
##      5.59
```

```r
exp(model_glm_mage35$coefficients[2])
```

```
## mage35Yes
##       2.2
```

The results of the binomial logistic regression model, with `glm`, shows that neither of the predictors have a significant main effect. We can see this through testing each of the predictors with making model comparison with `anova(model_1, model_2, test="Chisq")`.

We compared a base model, which includes only the intercept, to models including whether the mother is a smoker (`smoker`), age of the mother (`mage35`), and both of these predictors predictors (`smoker + mage35`) to assess the significance of each predictor. The analysis of deviance using Chi-square tests revealed that neither of the predictors are significant in their effect on low birth weight for newborns (`lowbwt`).

For the comparison between the base model and the model including `smoker`, there was no significant improvement in model prediction, $Deviance = 2.93$, $p = .087$. Similarly, for the comparison between the base model and the model including `mage35`, there was no significant improvement in model prediction, $Deviance = 0.35$, $p = .55$.

Although, it needs to be noted that the predictor of `smoker` is very close to significance with $p = .087$, which is in line with the graphs we have produced for 2e, although we would have expect it to be significant. Regarding `mage35`, based on the graphs, we had doubts whether this was a significant predictor, and now we can see that it is not.

When we investigate the **odds**, we can take the exponential of the coefficients in the model. The results of the models show that the odds of having a low birth weight baby for smoker mothers are about 5.6 times the odds for non-smoker mothers. And, the odds of having a low birth weight baby for mothers older than 35 are about 2.2 times the odds for mothers younger than 35. Although, as stated above, neither of these predictors seem to be significant with the default alpha level of 0.05.

Question 2g)

Investigate the interaction of predictor *Gestation* with *smoker*, and the interaction of *Gestation* with *mage35* (one interaction at a time). From this and f), choose a resulting model.

```r
model_glm_Gestation <- stats::glm(lowbwt ~ Gestation, family=binomial("logit"), data=data_new)
model_glm_int_smoker <- stats::glm(lowbwt ~ smoker*Gestation, family=binomial("logit"), data=data_new)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
model_glm_int_mage35 <- stats::glm(lowbwt ~ mage35*Gestation, family=binomial("logit"), data=data_new)

# Testing with ANOVA
results_1 <- anova(model_glm_0, model_glm_Gestation, test="Chisq")
results_2 <- anova(model_glm_Gestation, model_glm_int_smoker, test = "Chisq")
results_3 <- anova(model_glm_Gestation, model_glm_int_mage35, test = "Chisq")
```

We followed a similar model comparison procedure as 2f. As a first step, we can see that the model with `Gestation` as a predictor, compared to only the model with the intercept is significantly better in terms of model prediction, $Deviance = 16.4$, $p < .001$.

Given that the models with `smoker` and `mage35` predictors are not significantly better than the model with only the intercept, when we add the interaction terms (seperately to two models), we can deduce whether the interaction terms `smoker*Gestation` and `mage35*Gestation` provide significant improvement to model prediction.

We can see that the `smoker*Gestation` model is significantly better than just having `Gestation` as a predictor, $Deviance = 7.87$, $p = .02$. However, the model with `mage35*Gestation` model is not a significant improvement than just having `Gestation` as a predictor.

To note, running this `glm` model gives a `Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred`. Upon some research, we have discovered that this warning means that the model almost perfectly predict the outcome variable. We interpret this as a good sign.

However, we need to consider that the model improvement for `lowbwt ~ smoker * Gestation` might simply depend on the near-significant effect (see our answer for 2f) of `smoker` as a predictor. So, we can further test this by running a Chi-square test for the interaction model.

```
results <- anova(model_glm_int_smoker, test = "Chisq")
```

We can see that the interaction term `smoker:Gestation` is not significant, $p = 0.198$. We also further confirm this result through a model comparison with `smoker + Gestation` and `smoker*Gestation`:

```
model_glm_noint_smoker <- stats::glm(lowbwt ~ smoker + Gestation, family=binomial("logit"), data=data_n
anova(model_glm_noint_smoker, model_glm_int_smoker, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: lowbwt ~ smoker + Gestation
## Model 2: lowbwt ~ smoker * Gestation
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1        39       11.8
## 2        38       10.2  1     1.66      0.2
```

Hence, even though `smoker` variable is nearly non-significant, due to the model comparison results, we choose our final model as `lowbwt ~ smoker + Gestation`.

Question 2h)

h) For the resulting model from g), estimate the probability of low baby weight for each combination of levels of the involved factors and the gestation of 40 weeks.

```r
new_data <- data.frame(Gestation = 40, smoker = factor(c("No", "Yes"), levels = c("No", "Yes")))
probabilities <- predict(model_glm_noint_smoker, newdata = new_data, type = "response")
new_data$probability_lowbwt <- probabilities
print(new_data)
```

```
##   Gestation smoker probability_lowbwt
## 1        40     No           6.97e-05
## 2        40    Yes           1.65e-02
```

Question 2i)

i) Another approach to address the questions in e) would be to apply a contingency table test. Implement the relevant test(s). Is this approach wrong? Name both an advantage and a disadvantage of this approach as compared to the one from f).

```r
## Chi-squared test for smoking mothers
smoker_table <- table(data_new$smoker, data_new$lowbwt)
chisq_test_smoker <- chisq.test(smoker_table)
```

```
## Warning in chisq.test(smoker_table): Chi-squared approximation may be incorrect
```

```r
chisq_test_smoker
```

```
## 
##  Pearson's Chi-squared test with Yates' continuity correction
## 
## data:  smoker_table
## X-squared = 1, df = 1, p-value = 0.2
```

```r
## Chi-squared test for mothers over 35
mage35_table <- table(data_new$mage35, data_new$lowbwt)
chisq_test_mage35 <- chisq.test(mage35_table)
```

```
## Warning in chisq.test(mage35_table): Chi-squared approximation may be incorrect
```

```r
chisq_test_mage35
```

```
## 
##  Pearson's Chi-squared test with Yates' continuity correction
## 
## data:  mage35_table
## X-squared = 3e-31, df = 1, p-value = 1
```

Exercise 3

```
data <- read.table('awards.txt', header=TRUE)
```

Question 3a)

a) Investigate whether the type of program influences the number of awards by performing a Poisson regression, without taking variable *math* into account. Estimate the numbers of awards for all the three types of program. Which program type is the best for the number of awards for this model?

```
data$prog = as.factor(data$prog);
mod1 = glm(num_awards ~ prog, family="poisson", data=data);
summary(mod1);
```

```
##
## Call:
## glm(formula = num_awards ~ prog, family = "poisson", data = data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -0.549      0.196   -2.80   0.0052 **
## prog2          0.707      0.216    3.27   0.0011 **
## prog3          0.443      0.246    1.80   0.0720 .
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 228.83  on 199  degrees of freedom
## Residual deviance: 216.10  on 197  degrees of freedom
## AIC: 512.4
##
## Number of Fisher Scoring iterations: 5
```

```
# Exponentiate the coefficients to get the relative multipliers
exp_coefs <- exp(coef(mod1))

# The intercept gives the expected count for prog1
expected_prog1 = exp(coef(mod1)["(Intercept)"])

# For prog2 and prog3, multiply the expected count for prog1 by their relative multipliers
expected_prog2 = expected_prog1 * exp_coefs["prog2"]
expected_prog3 = expected_prog1 * exp_coefs["prog3"]

# Output the expected counts
expected_prog1
```

```
## (Intercept)
##       0.578
```

```
expected_prog2
```

```
## (Intercept)
##        1.17
```

```
expected_prog3
```

```
## (Intercept)
##         0.9
```

Answer:

- The type of program **significantly influences** the number of awards students receive. This is evidenced by the Poisson regression model, where the coefficient for the general program (prog2) is statistically significant, indicating a higher expected number of awards compared to the vocational program (prog1/reference). However, the difference between the academic program (prog3) and the vocational program (prog1/reference) is not statistically significant at the 0.05 level.
- The expected number of awards for vocational (prog1), general (prog2), and academic (prog3) programs are approximately **0.578, 1.17, and 0.9** awards respectively. This estimation is based on the exponentiated coefficients from the Poisson regression model.
- The general program **(prog2) is the best** for maximizing the number of awards, as students in this program are expected to receive the highest number of awards compared to the other programs, based on the model's estimates.

Question 3b)

b) For the situation in a), can the Kruskall-Wallis test also be used? If yes, apply the test and comment on the results; of no, explain why this test cannot be used.

```
kruskal.test(num_awards ~ prog, data = data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  num_awards by prog
## Kruskal-Wallis chi-squared = 11, df = 2, p-value = 0.005
```

Answer:

- Yes, the Kruskal-Wallis test can be used as a non-parametric alternative to investigate whether the type of program influences the number of awards students receive. This test is suitable for comparing the distributions of a continuous or ordinal dependent variable (in this case, the number of awards) across more than two groups (the program types) without assuming a normal distribution of the data.
- The p-value obtained from the Kruskal-Wallis test is 0.00462, which is less than the significance level of 0.05. This indicates that there are statistically significant differences in the distribution of the number of awards received by students across the three types of programs (vocational, general, and academic).

Question 3c)

c) Now include predictor *math* into analysis and investigate the influence of the explanatory variables *prog* and *math* (and their interaction) on the numbers of awards. Which program type is the best for the number of awards? Comments on your findings. Use the resulting model to predict the numbers of awards for all three programs and math score 56.

```
mod2 = glm(num_awards ~ prog * math , family="poisson", data=data);
summary(mod2);
```

```
##
## Call:
## glm(formula = num_awards ~ prog * math, family = "poisson", data = data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.57844    1.39143   -1.13     0.26
## prog2       -1.06123    1.53452   -0.69     0.49
## prog3        0.96214    1.63597    0.59     0.56
## math         0.02036    0.02695    0.76     0.45
## prog2:math   0.02744    0.02897    0.95     0.34
## prog3:math  -0.00944    0.03240   -0.29     0.77
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 228.83  on 199  degrees of freedom
## Residual deviance: 194.36  on 194  degrees of freedom
## AIC: 496.7
##
## Number of Fisher Scoring iterations: 5
```

```
mod3 = glm(num_awards ~ prog + math , family="poisson", data=data);
summary(mod3);
```

```
##
## Call:
## glm(formula = num_awards ~ prog + math, family = "poisson", data = data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.37258    0.47552   -4.99  6.1e-07 ***
## prog2        0.45262    0.22475    2.01    0.044 *
## prog3        0.56172    0.24748    2.27    0.023 *
## math         0.03578    0.00834    4.29  1.8e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 228.83  on 199  degrees of freedom
## Residual deviance: 198.05  on 196  degrees of freedom
## AIC: 496.4
##
## Number of Fisher Scoring iterations: 5
```

- After including the predictor *math* and investigating the influence of *prog*, *math*, and their interaction on the number of awards, the coefficients in model 2 (mod2) were not statistically significant. This led to the creation of a simplified model 3 (mod3) without the interaction terms. The results from mod3 indicate that both program type and math scores significantly influence the number of awards, with math scores showing a strong positive effect. This suggests that higher math scores are associated with a higher number of awards, independent of program type.
- Based on the comparison of coefficients from the model, **program type 3** is identified as the best for maximizing the number of awards. With a coefficient of **0.56172**, it surpasses program type 2, which has a coefficient of **0.45262**. This indicates that, holding math scores constant, students in program type 3 are expected to receive the highest number of awards due to its more substantial positive impact on the award count.
- Below, the resulting model is used to predict the numbers of awards for all three programs and math score being 56.

```r
# Create a new data frame for prediction
new_data <- expand.grid(prog = factor(levels(data$prog)), math = 56)

# Predict the number of awards using the model
new_data$predicted_awards <- predict(mod3, newdata = new_data, type = "response")

print(new_data)
```

```
##   prog math predicted_awards
## 1    1   56            0.691
## 2    2   56            1.087
## 3    3   56            1.213
```