Assignment 2

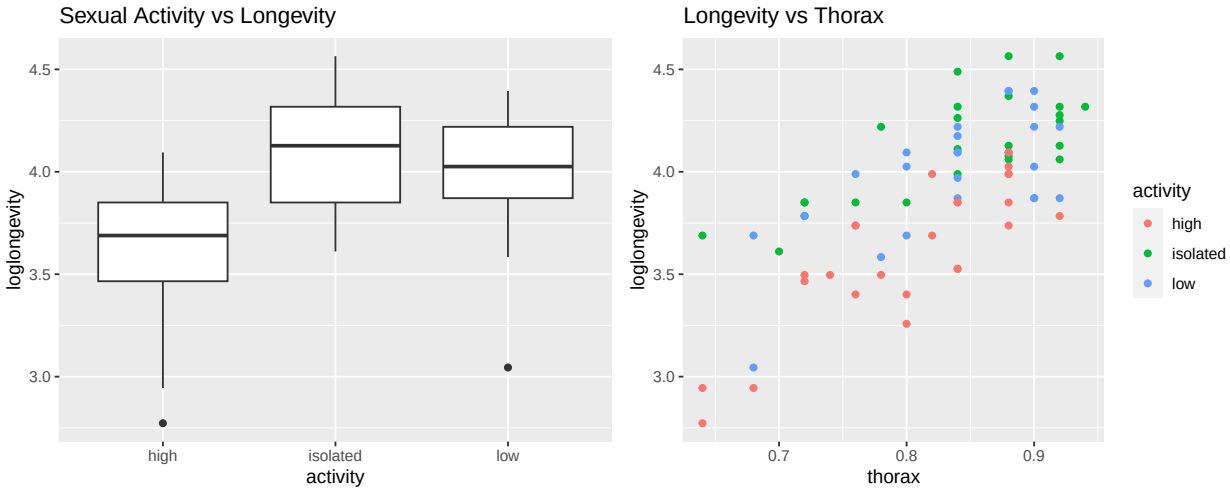
Martynas Vaznonis, Andrei Puchkov, Carlo Peron

2024-03-08

Exercice 1

a) After loading in the data and adding a column for the log of longevity we create the plots seen below. From the plot it seems like there may be a difference among the different levels of activity but it is not perfectly clear. On the other hand, a correlation between the log of longevity and thorax length is already likely from the plot. To clarify if sexual activity had an effect on the longevity, we ran a 1-way ANOVA test. The test showed that there is indeed a significant difference in the longevity of a fruit fly based on its sexual activity.

```
gridExtra::grid.arrange(
ggplot(data, aes(x = activity, y = loglongevity)) + geom_boxplot() +
  labs(title = 'Sexual Activity vs Longevity'),
ggplot(data, aes(x = thorax, y = loglongevity, color = activity)) + geom_point() +
  labs(title = 'Longevity vs Thorax'), ncol=2)
```




```
activityaov = lm(loglongevity ~ activity, data=data)
anova(activityaov)
```

```
## Analysis of Variance Table
##
```

```
## Response: loglongevity
##           Df Sum Sq Mean Sq F value  Pr(>F)
## activity   2   3.67   1.833    19.4 1.8e-07 ***
## Residuals 72   6.80   0.094
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Finally, we check what the mean log of longevity is per level of sexual activity and found that the virgin flies live longer. In other words, it seems that the more poontang a fly gets, the faster it dies.

```
tapply(data$loglongevity, data$activity, mean)
```

```
##     high isolated      low
##     3.60     4.12     4.00
```

b) The positive coefficient for thorax suggests that flies with longer thorax lengths tend to have a higher estimated longevity. The predicted results show variance in longevity given the average thorax for different activity levels. The highest expected longevity is seen in the isolated case while the lowest in the high case. Thus, under this model too, the longevity decreases with sexual activity.

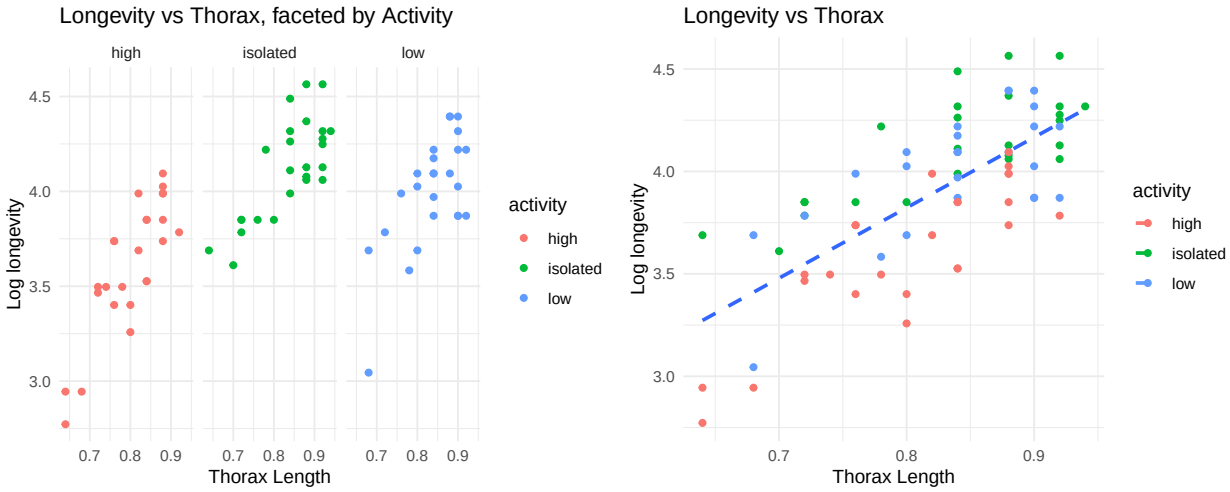
```
loglongaov = lm(loglongevity ~ thorax + activity, data=data); summary(loglongaov)
```

```
##
## Call:
## lm(formula = loglongevity ~ thorax + activity, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4858 -0.1612  0.0104  0.1510  0.3574
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        1.2189     0.2486    4.90  5.8e-06 ***
## thorax             2.9790     0.3067    9.71  1.1e-14 ***
## activityisolated   0.4100     0.0584    7.02  1.1e-09 ***
## activitylow        0.2857     0.0585    4.88  6.2e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.203 on 71 degrees of freedom
## Multiple R-squared:  0.721,  Adjusted R-squared:  0.709
## F-statistic: 61.2 on 3 and 71 DF,  p-value: <2e-16
```

```
average_thorax <- mean(data$thorax)
predictions <- data.frame(activity=unique(data$activity), thorax=average_thorax)
rbind(predictions$activity, round(predict(loglongaov, newdata=predictions), 3))
```

```
##      1          2       3
## [1,] "isolated" "low"   "high"
## [2,] "4.085"    "3.961" "3.675"
```

c) It is clear to see that the longevity is correlated with the thorax length from the plots below. Whether there is any difference in the influence of thorax among the levels of activity is not as clear. To test this, we run an ANOVA test with interaction between thorax length and sexual activity. If they interact, that means that the effect of thorax length is not the same across the groups of activity. Because we do not observe a significant contribution of the interaction term, we conclude that thorax length affects the longevity of all fruit flies the same.




```
interaction_model <- lm(loglongevity ~ thorax * activity, data=data)
anova(interaction_model)
```

```
## Analysis of Variance Table
##
## Response: loglongevity
##                 Df Sum Sq Mean Sq F value  Pr(>F)
## thorax           1   5.43    5.43  135.62 < 2e-16 ***
## activity         2   2.11    1.06   26.38 3.1e-09 ***
## thorax:activity  2   0.15    0.08    1.93    0.15
## Residuals       69   2.76    0.04
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

d) The analysis with thorax length is more complete. Because the thorax affects all levels of the activity factor identically, the results of both analyses coincide in terms of the significance of the activity factor but this may not always be the case. Therefore, including thorax leads to a more correct model. Furthermore, the test below shows that including thorax, significantly enhances the model.

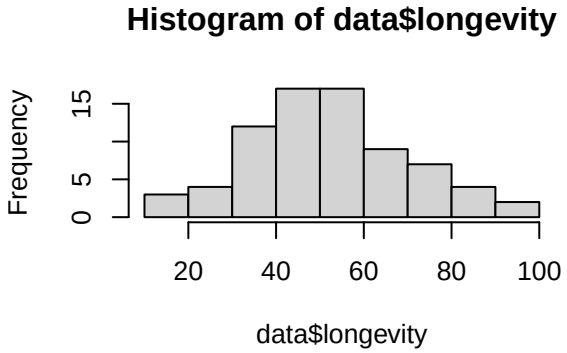
```r
anova(activityaov, loglongaov)
```

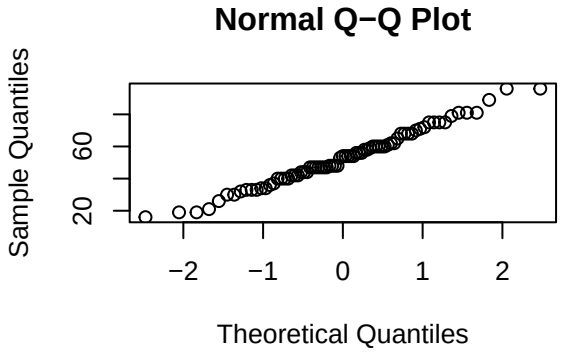
```
## Analysis of Variance Table
##
## Model 1: loglongevity ~ activity
## Model 2: loglongevity ~ thorax + activity
##   Res.Df  RSS Df Sum of Sq    F  Pr(>F)
## 1     72 6.80
## 2     71 2.92  1      3.88 94.4 1.1e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

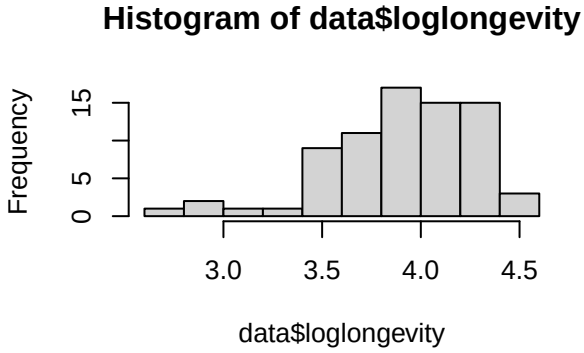
e) The plots below show that the longevity is likely normally distributed while the log of longevity is likely not. Further evidence for this is provided by the Shapiro-Wilk tests. The residuals for both models seem pretty good and likely normal. Since, ANOVA requires normally distributed samples, it is better to use the number of days as the response rather than the logarithm of it.

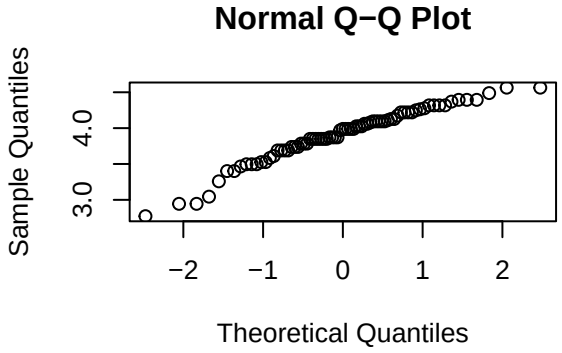
```r
longaov = lm(longevity ~ thorax + activity, data=data)

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







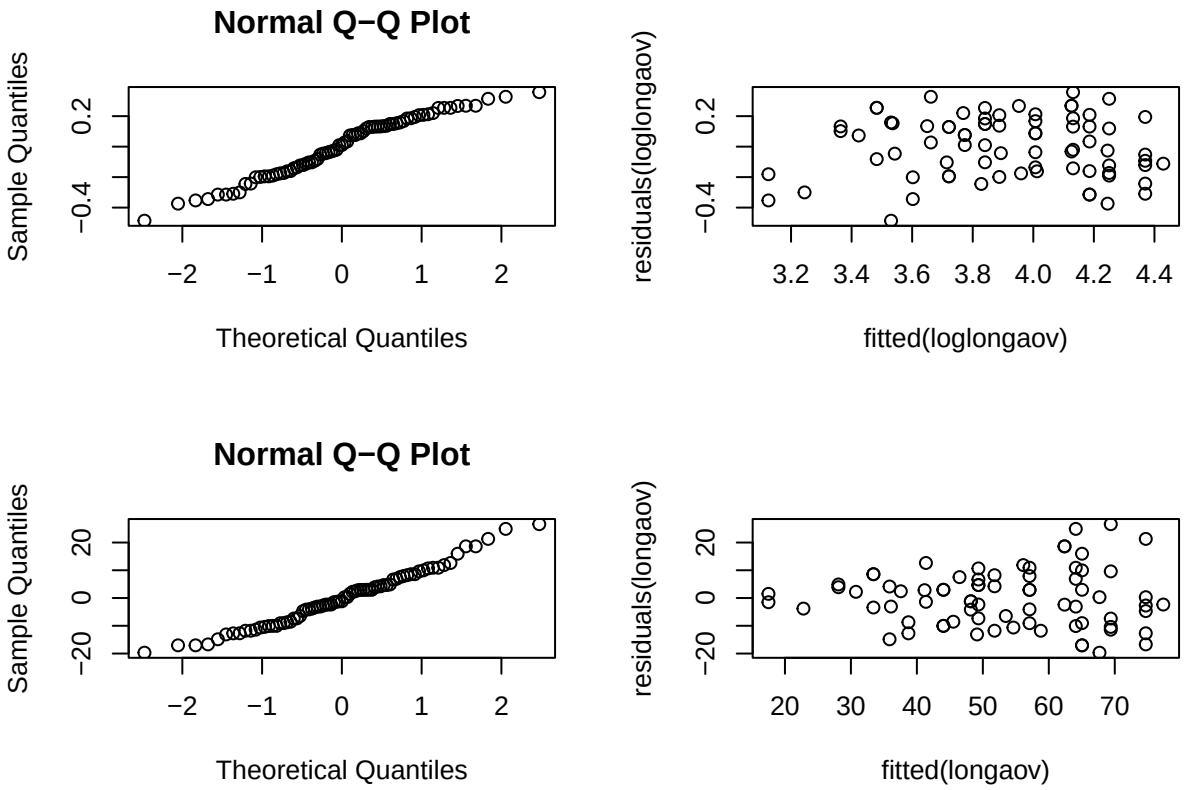


```r
shapiro.test(data$longevity)[2]; shapiro.test(data$loglongevity)[2]
```

```
## $p.value
## [1] 0.633
```

```
## $p.value
## [1] 0.00873
```

```r
qqnorm(residuals(loglongaov)); plot(fitted(loglongaov), residuals(loglongaov))
qqnorm(residuals(longaov)); plot(fitted(longaov), residuals(longaov))
```



Exercice 2

a) Spotting potential points is difficult visually in multidimensional space, but we only need to check for influence points as those are the only significant outliers. To analyze the problem of influence points, we're going to use Cook's distance. This method quantifies influence points in the dataset by measuring how much change that point contributes to in the final model. No influence points are present within the current model. This is unsurprising however, because of the high dimensionality of the data.

```r
# potential is a table with only the potential predictors
model_all = lm(Birthweight ~ ., data=potential)
cook <- as.data.frame(cooks.distance(model_all))
length(cook[cook>=1])
```

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

To check collinearity we can use a correlation matrix and pairwise correlation plots. But given the large number of potential predictors, these measures would likely be overwhelming to look at and so, bad for analysis. Instead we use VIF for a more concise description of collinearity. We can see relatively little collinearity with no predictor $\text{VIF}_j > 5$. The highest VIF value is for *fage*, which likely correlates with *mage* and potentially *fedyrs*.

```r
vif(model_all)
```

```
##    Length  Headcirc Gestation      mage    mnocig   mheight     mppwt      fage
##      3.12      1.84      2.51      4.03      1.42      3.11      2.38      4.52
##    fedyrs    fnocig   fheight
##      1.61      1.71      1.62
```

b) At the end of the loop, we can see every dropped predictor. The first was *fage* and the last *mppwt* with a total of 9 predictors dropped and 2 ones remaining. The variables left are *headcirc* and *gestation* which seem like very natural choices for estimating the birth weight of a child.

```r
reduced_model <- model_all
drop = NULL
repeat{
  p_values = summary(reduced_model)$coefficients[,c(0, 4)]
  max_p = max(p_values)
  if(max_p <= 0.05) break
  var = which.max(p_values)
  drop = c(drop, names(var))
  reduced_model <- update(reduced_model, as.formula(paste(". ~ . -", names(var))))
}
summary(reduced_model)
```

```
##
## Call:
## lm(formula = Birthweight ~ Headcirc + Gestation, data = potential)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8289 -0.2476 -0.0514  0.2514  0.7435
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -5.4480     0.9394   -5.80  9.8e-07 ***
## Headcirc      0.1198     0.0245    4.89  1.8e-05 ***
## Gestation     0.1178     0.0222    5.30  4.9e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.344 on 39 degrees of freedom
## Multiple R-squared:  0.691,  Adjusted R-squared:  0.675
## F-statistic: 43.6 on 2 and 39 DF,  p-value: 1.12e-10
```

```
drop
```

```
## [1] "fage"    "mheight" "fedyrs"  "fnocig"  "mnocig"  "Length"  "fheight"
## [8] "mage"    "mppwt"
```

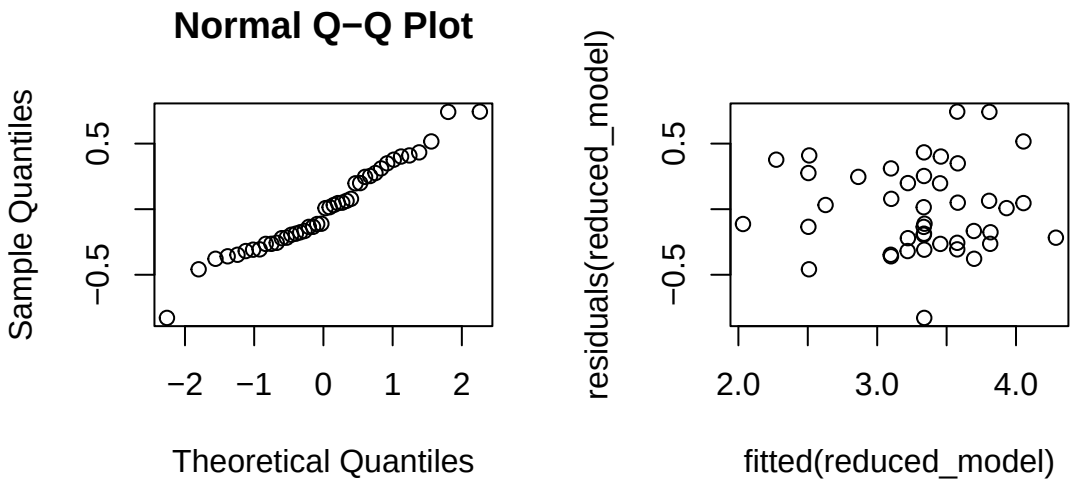
Checking for the model assumptions below we can see that there are no issues with collinearity nor outliers. The residuals seem a little off in the qq plot but a Shapiro-Wilk test (not shown here) does not reject normality.

```
vif(reduced_model); paste("# of outliers: ", sum(cooks.distance(reduced_model)>1))
```

```
##  Headcirc Gestation
##       1.2       1.2
```

```
## [1] "# of outliers:  0"
```

```
par(mfrow=c(1, 2)); qqnorm(residuals(reduced_model))
plot(fitted(reduced_model), residuals(reduced_model))
```



c) Below we can see the mean predicted birth weight value for the average baby with a 95% confidence and prediction interval respectively.

```
average_values <- data.frame(Headcirc = mean(data$Headcirc),
                             Gestation = mean(data$Gestation))
predict(reduced_model, newdata = average_values, interval = "confidence")[,]
```

```
##  fit  lwr  upr
## 3.31 3.21 3.42
```

```
predict(reduced_model, newdata = average_values, interval = "prediction")[,]
```

```
##  fit  lwr  upr
## 3.31 2.61 4.02
```

d) The LASSO method is a numerical optimization method rather than an analytical solution. Therefore, the MSE on the validation set is better modeled as a random variable. Moreover, the train-validation split is randomized as well, further contributing to the uncertainty not only of the LASSO model but also of the step down model. Thus, to compare the models derived with the LASSO method below with the model in **b)**, we need to train them a myriad of times. Similarly, we need to fit a different model each time for the predictors derived with the step down strategy to account for different train-validation splits. Then running ANOVA, we see that there is a significant difference among the conditions. Further investigations show that the worst model is indeed the one derived from the step down method. The best is the one that uses the 1sd λ value. The overall difference in the MSE however is still quite small.

The LASSO method, like the step down one, selected **Headcirc** and **Gestation** as significant predictors for the baby's body weight. But alongside those, it also selected **Length**.

```
mse <- function(pred, y){ # Shortcut to get the MSE
  return(mean(pred - y)^2)
}

x = as.matrix(potential[,names(potential) != "Birthweight"])
y = potential$Birthweight

df <- data.frame(matrix(ncol=3, nrow=0, dimnames=list(
  NULL, c("lambda.min", "lambda.1se", "step.down"))))

for(i in 1:1000){
  # New train-validation split
  train = sample(1:nrow(x), 0.67*nrow(x))
  x.train = x[train,]; y.train = y[train]
  x.val = x[-train,]; y.val = y[-train]

  # Fit a new LASSO and reduced model
```

```
  lasso.model = glmnet(x.train, y.train, alpha=1)
  lasso.cv = cv.glmnet(x.train, y.train, alpha=1, type.measure="mse", nfolds=5)
  red.mod = lm(reduced_model$call[[2]], data=potential[train,])

  # Append the MSEs to the dataframe
  df[nrow(df) + 1,] = c(
  mse(predict(lasso.model, s=lasso.cv$lambda.min, newx=x.val), y.val),
  mse(predict(lasso.model, s=lasso.cv$lambda.1se, newx=x.val), y.val),
  mse(predict(red.mod, newdata=potential[-train,]), potential[-train,]$Birthweight))
}

anova(lm(values ~ ind, data=stack(df)))
```

```
## Analysis of Variance Table
##
## Response: values
##             Df Sum Sq Mean Sq F value  Pr(>F)
## ind          2  0.011 0.00556    9.17 0.00011 ***
## Residuals 2997  1.816 0.00061
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

e) Below we can see how many babies were born underweight. First we discriminate by smoker and then by the mother's age. Both smoking and age seem to have a positive correlation with being born underweight. Notably, there are only four mothers over the age of 35 in the dataset which could misrepresent the actual proportion of the malnourished babies in this demographic. The bar plot indicates that smoking has a greater effect than age on low birth weight. Despite that the effect seems small and unlikely to be significant.

```
# Number of underweight babies
tapply(uwt$lowbwt, uwt$smoker, function(x) paste(sum(x), "out of", length(x)))
```

```
##    Non smoker        Smoker
## "1 out of 20" "5 out of 22"
```

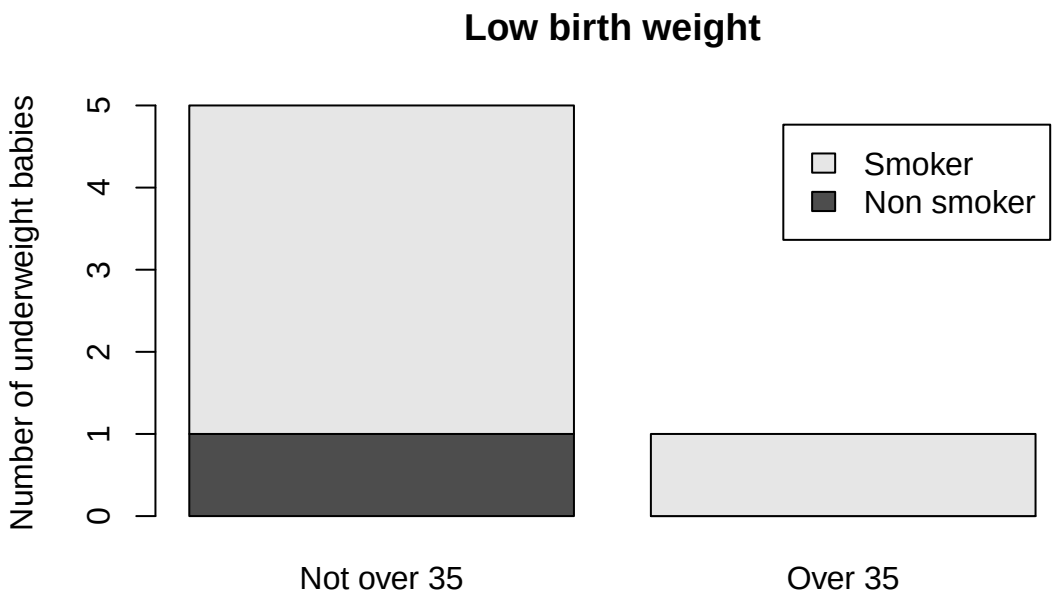
```
tapply(uwt$lowbwt, uwt$mage35, function(x) paste(sum(x), "out of", length(x)))
```

```
##   Not over 35       Over 35
## "5 out of 38"  "1 out of 4"
```

```
# Summary table
xtabs(lowbwt ~ smoker + mage35, data=uwt)
```

```
##             mage35
## smoker       Not over 35 Over 35
##   Non smoker           1       0
##   Smoker               4       1
```

```r
par(mfrow=c(1, 1))
barplot(xtabs(lowbwt ~ smoker + mage35, data=uwt), main="Low birth weight",
        ylab="Number of underweight babies", legend=unique(uwt$smoker))
```



f) The logistic regression model appears in line with what we have seen in **e)**, which is to say that smoking and old age do not significantly contribute to low birth weight. On the other hand, the explanatory variable gestation did have a significant effect. The way it can be interpreted is that one unit increase in gestation multiplies the odds of being born underweight by $e^{-1.463}$. In other words, gestation corresponds to a reduced likelihood of a low weight birth.

```r
logistic.model = glm(lowbwt ~ Gestation+smoker+mage35, data=data, family="binomial")
summary(logistic.model)
```

```
##
## Call:
## glm(formula = lowbwt ~ Gestation + smoker + mage35, family = "binomial",
##     data = data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   48.992     22.566    2.17    0.030 *
## Gestation     -1.463      0.670   -2.18    0.029 *
## smoker1        5.450      3.357    1.62    0.104
## mage351        0.322      4.375    0.07    0.941
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 34.450  on 41  degrees of freedom
## Residual deviance: 11.803  on 38  degrees of freedom
## AIC: 19.8
##
## Number of Fisher Scoring iterations: 8
```

g) In both cases, the interactions are insignificant, and in the case of the model with smoker:Gestation interaction, all predictors lost their significance. Thus, it is best to stick to the model in **f)** which was the simplest while still explaining the most data.

```
logistic.smoker = glm(lowbwt ~ Gestation*smoker+mage35, data=data, family="binomial")
logistic.mage35 = glm(lowbwt ~ Gestation*mage35+smoker, data=data, family="binomial")
```

h) As expected, the lowest probability of a low weight birth is when the mother is young and does not smoke. Smoking has a bigger influence than age on pushing the probability towards such a birth and the synergy of both leads to the highest odds of *lowbwt*. Regardless, both smoking and age are overshadowed by gestation leading to overall very slim chances of abnormally low birth weights across the board.

```
new = data.frame(Gestation=40, mage35=unique(data$mage35),
                 smoker=rep(unique(data$smoker), each=2))
new$pred = predict(logistic.model, newdata=new, type="response")
new
```

```
##   Gestation mage35 smoker     pred
## 1        40      0      0 7.20e-05
## 2        40      1      0 9.94e-05
## 3        40      0      1 1.65e-02
## 4        40      1      1 2.26e-02
```

i) We create contingency tables for *smoker*/*lowbwt* and *mage35*/*lowbwt* to examine the question outlined in **e)**. Then, we attempted to use the `chisq.test` to determine if these factors were related but because in both cases half of $E_{ij} < 5$, the χ^2 approximation is unreliable. Luckily, these are $2x2$ tables so we can use Fisher's exact test instead. Just as before, we find no significant relation between smoking/age and low birth weight.

```
fisher.test(table(data[,c("smoker", "lowbwt")]))
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  table(data[, c("smoker", "lowbwt")])
```

```
## p-value = 0.2
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##    0.526 277.497
## sample estimates:
## odds ratio
##       5.39
```

```r
fisher.test(table(data[,c("lowbwt", "mage35")]))
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  table(data[, c("lowbwt", "mage35")])
## p-value = 0.5
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##   0.035 33.957
## sample estimates:
## odds ratio
##       2.15
```

Exercice 3

a)

b)

c)