STK-IN4300
Statistical Learning Methods in Data Science

OBLIG 1

Egil Furnes
Student: 693784

Problem 1

a)

I have chosen the dataset `HistData::PearsonLee` which is inspired by a similar dataset from Galton on heights of parents and their offspring. From the description of the dataset:

> Wachsmuth et. al (2003) noticed that a loess smooth through Galton's data on heights of mid-parents and their offspring exhibited a slightly non-linear trend, and asked whether this might be due to Galton having pooled the heights of fathers and mothers and sons and daughters in constructing his tables and graphs.

This is an expantion on Galtons data to find if there are also gender-effects exhibited on the relation between parent-child height.

```r
df <- HistData::PearsonLee %>% as_tibble()
df %>% summary()
```

```
     child           parent         frequency       gp
 Min.   :52.50   Min.   :52.50   Min.   : 0.250   fd:206
 1st Qu.:62.50   1st Qu.:61.50   1st Qu.: 0.750   fs:179
 Median :65.50   Median :65.50   Median : 2.500   md:185
 Mean   :66.05   Mean   :65.01   Mean   : 6.558   ms:176
 3rd Qu.:69.50   3rd Qu.:68.50   3rd Qu.: 9.250
 Max.   :79.50   Max.   :75.50   Max.   :46.500
     par            chl
 Father:385   Daughter:391
 Mother:361   Son     :355
```

b)

In this task I design a few bad and good charts. First we look at the distribution of the categorical variable `par`, which is a binary variable with two levels, `Father` and `Mother`. To make it even 'better' for our readers we have plotted this distribution using polar coordinates, such that one must read the distribution by angle rather than height or length. Furthermore we use blue for `Mother` and red for `Father` which is probably against convention and can contribute to making the plot more difficult to read.

```r
ggplot(df, aes(x = par, weight = frequency, fill = par)) +
  geom_bar() +
  coord_polar(theta = "y") +
  labs(title = "Parent sex distribution",
       x = NULL,
       y = NULL) +
  theme_minimal()
```

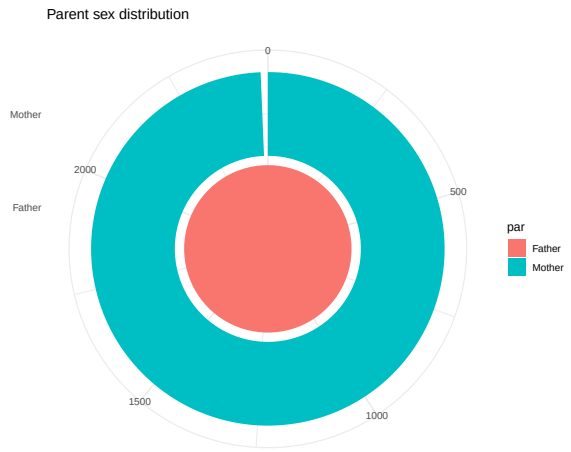

Figure 1: Parent sex distribution

Next we plot a distribution of the height of the children in the variable `child`, which is a continuos variable. It does make sense to plot a distribution of a continuos variable – but in our plot we use way too few `bins` in `geom_histogram` with 3, which only divides the height of the children into 3 categories. Furthermore, using `coord_flip` in this plot makes it have an unfortunate visual representation.

```
ggplot(df, aes(x = child, weight = frequency)) +
  geom_histogram(bins = 3,
                 fill = "#D2B48C",
                 color = "black") +
  ggtitle("Distribution of child height") +
  coord_flip() +
  theme_minimal()
```

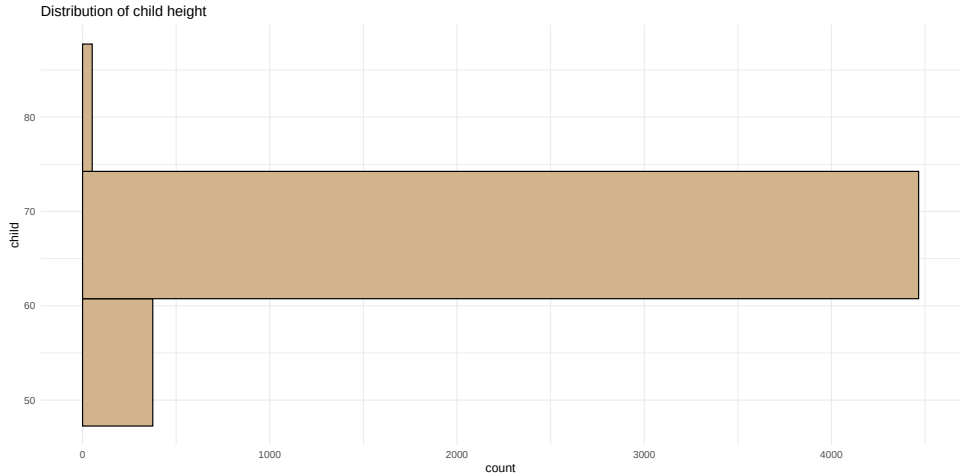

Figure 2: Distribution of child height

c)

In this task we are set to improve the plots or maybe even make them 'good'. Regarding the distribution of children height in inches I have now removed the `coord_flip` command and furthermore increased the bins to show an approximated normal distribution in height.

```
ggplot(df, aes(x = child, weight = frequency)) +
  geom_histogram(bins = 29,
                 fill = "#D2B48C",
                 color = "black") +
  ggtitle("Distribution of child height") +
  xlab("Height of children in inches") +
  ylab("Count") +
  theme_minimal()
```

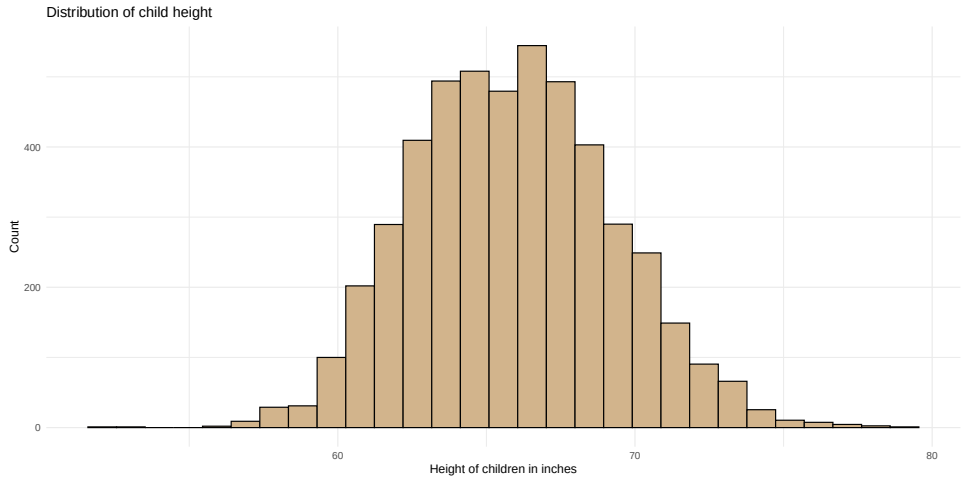

Figure 3: Caption

Regarding the distribution of parents, I found that just plotting the count for `Father` 2454 and `Mother` 2438 would be two boring bars. Therefore I added the distribution of the parent heights as density plots and split them by sex. As such we find some cool insights, such as a left-shift for mothers and right-shift for father indicating fathers on average are taller than mothers. Furthermore the distribution of mothers look 'taller' and for fathers 'wider'.

```
ggplot(df, aes(
  x = parent,
  weight = frequency,
  fill = par,
  colour = par
)) +
  geom_density(alpha = 0.35, adjust = 1.1) +
  scale_fill_manual(values = c(Father = "#4C78A8", Mother = "#F56598
     ")) +
  scale_colour_manual(values = c(Father = "#4C78A8", Mother = "#
     F56598")) +
```

```
  labs(
    title = "Parent height distribution by sex",
    x = "Parent height in inches",
    y = "Density",
    fill = NULL,
    colour = NULL
  ) +
  theme_minimal()
```

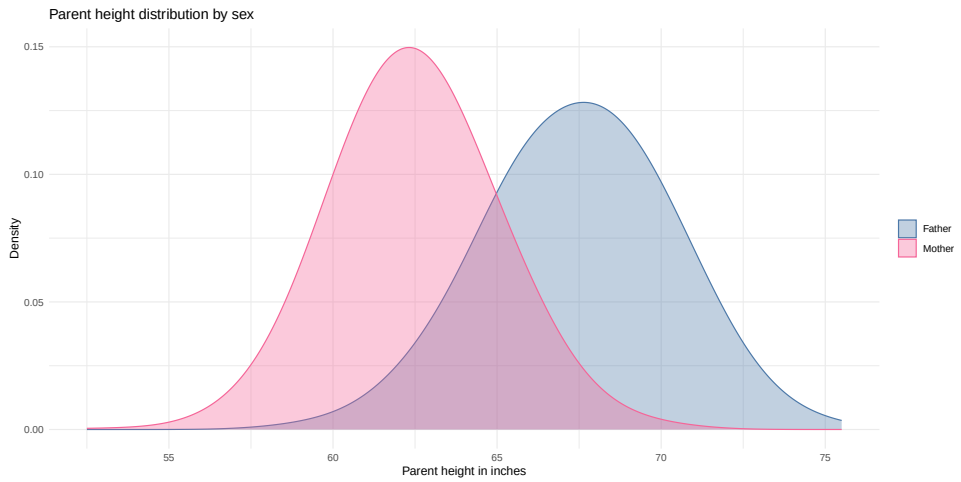

Figure 4: Parent height distribution in inches by sex

Problem 2

In this dataset we initially have 6 variables `child`, `parent`, `frequency`, gp, `par`, and `chl`.

However, looking at values for gp this is directly colinear with the pair of values from `par` and `chl`. As such I first remove the gp and rahter just keep the two others. Furthermore the `frequency` variable is an aggregated count of families from methodology by Pearson and Lee, and as such is not relevant to explain the height of the children, its rather just a useful data marker. We therefore remove this one too.

```
> df %>% head(3)
# A tibble: 3 × 6
  child parent frequency gp    par    chl
  <dbl>  <dbl>     <dbl> <fct> <fct>  <fct>
1  59.5   62.5       0.5 fs    Father Son
2  59.5   63.5       0.5 fs    Father Son
3  59.5   64.5       1   fs    Father Son
```

Now fitting the data to the model we just use a simple linear regression `lm()` model and try out a stepwise model selection using both 'forward' and 'backward' selection with the argument `both` in the function `stepAIC()` from the `MASS` package.

```r
library(MASS)
df <- HistData::PearsonLee %>% as_tibble()
lm1 <- lm(child~parent+par+chl, data = df, weights = frequency)
lm2 <- stepAIC(lm1, direction = "both", trace = FALSE)
```

From the regression output we actually find that both the `lm1()` simple linear regression and the `lm2()` stepwise model selection are equal. This is explained by all the used variables in the model being significant and so at an `0 ***` level.

```
> lm1

Call:
lm(formula = child ~ parent + par + chl, data = df, weights =
    frequency)

Coefficients:
(Intercept)       parent    parMother       chlSon
    28.4825       0.5216       2.8293       4.6985

> lm2

Call:
lm(formula = child ~ parent + par + chl, data = df, weights =
    frequency)

Coefficients:
(Intercept)       parent    parMother       chlSon
    28.4825       0.5216       2.8293       4.6985
```

Now looking at the regression output of the model we can assess how it is at explaining the height of children using the other variables. First off, we see that all input variables are statistically significant `***` including the `Intercept`.

Although this doesn't necessarily mean that the variables explain the output well. Now looking at the explanatory power of this model, we read the R-statistic, either the `Multiple R-squared` or the `Adjusted R-squared` which are `0.5658` and `0.5641` respectively, both being higher than the sort of threshold at `0.3`, and with a statistically significant `p-value` of `< 2.2e-16`.

```
> lm1 %>% summary()

Call:
lm(formula = child ~ parent + par + chl, data = df, weights =
    frequency)

Weighted Residuals:
     Min       1Q   Median       3Q      Max
-14.0597  -4.5536   0.2846   4.6882  12.7085

```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  28.4825     2.2150   12.86   <2e-16 ***
parent        0.5216     0.0327   15.95   <2e-16 ***
parMother     2.8293     0.2356   12.01   <2e-16 ***
chlSon        4.6985     0.1708   27.50   <2e-16 ***
---
Signif. codes:  0 ‘’*** 0.001 ‘’** 0.01 ‘’* 0.05 ‘’. 0.1 ‘’  1

Residual standard error: 5.917 on 742 degrees of freedom
Multiple R-squared:  0.5658,    Adjusted R-squared:  0.5641
F-statistic: 322.3 on 3 and 742 DF,  p-value: < 2.2e-16
```

In sum, we can probably say that the linear model appears to be sufficient in explaining this scenario.