Response to Teacher Question 3:

"With such a large sample size, you will find a lot of significant predictors. Discuss why."

Our sample size is over 17000 (17898). This can result in many of our predictors being statistically significant because a large sample size leads to an even more precise discovery of relationships between variables, reducing variance. Thus, with a very large sample size, even if the predictor's impact is smaller, it could still be considered statistically significant. If we think in terms of confidence intervals, they become extremely narrow because the margin of error decreases. While a predictor could be statistically significant, it may not be an impactful predictor of the output in reality.

One way we can tackle this problem is by using corrections such as Bonferroni, Holm, or Benjamini Hochberg to lower false positive rates. Also, we used training and testing split in predictions, which can reduce the number of samples during training and prevent overfitting to just the training data (and performing poorly on new data), and used AIC to choose models as well.

Research Question 1:

Which factors are the biggest contributors to overall FIFA ratings?

Here is how we set up the model

```
fifa <- read.csv("~/Desktop/School/UW/STAT 423/fifa_players.csv")
new_fifa <- fifa[c("overall_rating", "weak_foot.1.5.", "skill_moves.1.5.", "sprint_speed", "dribbling",
                   "strength", "stamina", "value_euro", "wage_euro")]
new_fifa <- na.omit(new_fifa)

basic <- lm(overall_rating ~ weak_foot.1.5. + skill_moves.1.5. + sprint_speed + dribbling + strength +
              stamina + log(value_euro) + log(wage_euro), data=new_fifa)
summary <- summary(basic)
summary
```

```
##
## Call:
## lm(formula = overall_rating ~ weak_foot.1.5. + skill_moves.1.5. +
##     sprint_speed + dribbling + strength + stamina + log(value_euro) +
##     log(wage_euro), data = new_fifa)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.1292 -1.3209 -0.1277  1.0716 14.2907
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.722617   0.182908  14.885  < 2e-16 ***
## weak_foot.1.5.    0.131624   0.026116   5.040  4.7e-07 ***
## skill_moves.1.5. -0.028740   0.040406  -0.711   0.4769
## sprint_speed     -0.039284   0.001721 -22.827  < 2e-16 ***
## dribbling        -0.004953   0.001943  -2.549   0.0108 *
## strength          0.043846   0.001507  29.088  < 2e-16 ***
## stamina           0.015620   0.001630   9.583  < 2e-16 ***
## log(value_euro)   4.139579   0.020907 197.998  < 2e-16 ***
## log(wage_euro)    0.690922   0.022343  30.924  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 2.138 on 17690 degrees of freedom
## Multiple R-squared:  0.906,  Adjusted R-squared:  0.906
## F-statistic: 2.132e+04 on 8 and 17690 DF,  p-value: < 2.2e-16
```

```
cor_matrix <- cor(new_fifa[, sapply(new_fifa, is.numeric)])
heatmap(cor_matrix, symm = TRUE, col = colorRampPalette(c("blue", "white", "red"))(100))
```

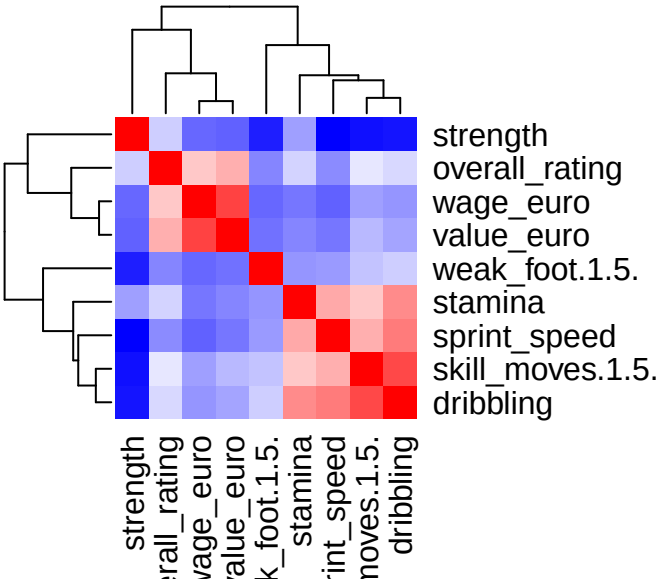

Fig. 8

So we started by looking at the relationship between each of our chosen predictors and the output variable of overall_rating

We saw roughly linear relationships for most cases. However, for value_euro and wage_euro, our scatter plots look very similar to a logarithmic curve.

```
par(mfrow = c(1, 2))
plot(new_fifa$value_euro, new_fifa$overall_rating, main = "Overall Rating vs. Value Euro", xlab="Value
plot(new_fifa$wage_euro, new_fifa$overall_rating, main = "Overall Rating vs. Wage Euro", xlab="Wage Eur
```

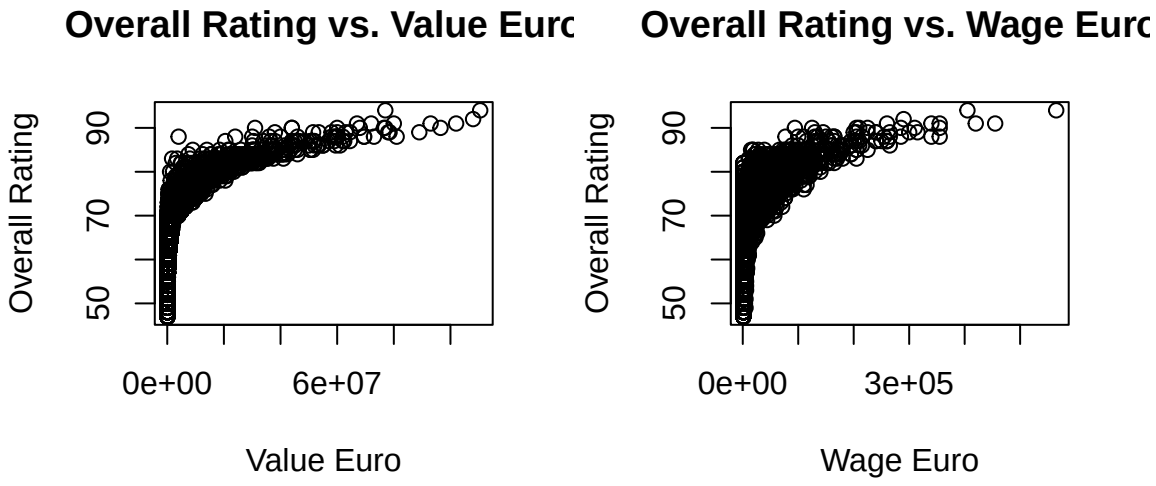

Fig. 9

Thus, we need to apply a transformation to our data and instead use ln(value_euro) and ln(wage_euro)

```
par(mfrow = c(1, 2))
plot(log(new_fifa$value_euro), new_fifa$overall_rating, main = "Overall Rating vs. Log Value Euro", xla
plot(log(new_fifa$wage_euro), new_fifa$overall_rating, main = "Overall Rating vs. Log Wage Euro", xlab=
```

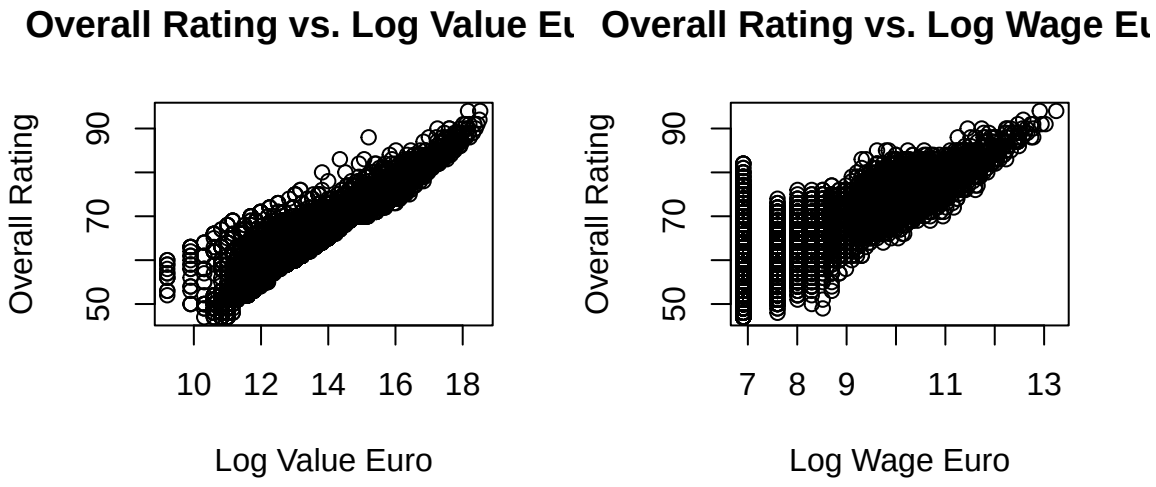

Fig. 10

This shows a much better linear relationship.

Now let's look at the R value or correlation coefficient of each variable to overall_rating:

```
correlations <- data.frame(
  Variable = c("Weak Foot", "Skill Moves", "Sprint Speed", "Dribbling", "Strength", "Stamina", "Log(Valu
  Correlation = c(
    cor(new_fifa$weak_foot.1.5., new_fifa$overall_rating),
    cor(new_fifa$skill_moves.1.5., new_fifa$overall_rating),
    cor(new_fifa$sprint_speed, new_fifa$overall_rating),
    cor(new_fifa$dribbling, new_fifa$overall_rating),
    cor(new_fifa$strength, new_fifa$overall_rating),
    cor(new_fifa$stamina, new_fifa$overall_rating),
    cor(log(new_fifa$value_euro), new_fifa$overall_rating),
    cor(log(new_fifa$wage_euro), new_fifa$overall_rating)
  )
)
print(correlations)
```

```
##             Variable Correlation
## 1          Weak Foot   0.2161278
## 2        Skill Moves   0.4194500
## 3       Sprint Speed   0.2191865
## 4          Dribbling   0.3799389
## 5           Strength   0.3625752
## 6            Stamina   0.3704942
## 7 Log(Value in Euro)   0.9406430
## 8  Log(Wage in Euro)   0.8087885
```

Table 5

We can see we found particularly high values for log(value_euro), log(wage_euro).

Now, let us form our linear regression model and utilize an F test to see our results.

```
model1 <- lm(overall_rating ~ weak_foot.1.5. + skill_moves.1.5. + sprint_speed + dribbling + strength +


summary <- summary(model1)
summary
```

```
##
## Call:
## lm(formula = overall_rating ~ weak_foot.1.5. + skill_moves.1.5. +
##     sprint_speed + dribbling + strength + stamina + log(value_euro) +
##     log(wage_euro), data = new_fifa)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.1292 -1.3209 -0.1277  1.0716 14.2907
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.722617   0.182908  14.885  < 2e-16 ***
## weak_foot.1.5.    0.131624   0.026116   5.040  4.7e-07 ***
## skill_moves.1.5. -0.028740   0.040406  -0.711   0.4769
## sprint_speed     -0.039284   0.001721 -22.827  < 2e-16 ***
## dribbling        -0.004953   0.001943  -2.549   0.0108 *
## strength          0.043846   0.001507  29.088  < 2e-16 ***
## stamina           0.015620   0.001630   9.583  < 2e-16 ***
## log(value_euro)   4.139579   0.020907 197.998  < 2e-16 ***
## log(wage_euro)    0.690922   0.022343  30.924  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.138 on 17690 degrees of freedom
## Multiple R-squared:  0.906,  Adjusted R-squared:  0.906
## F-statistic: 2.132e+04 on 8 and 17690 DF,  p-value: < 2.2e-16
```

So, we found p-values < 0.05 (our alpha/significance level) for the following predictors: weak_foot, sprint_speed, dribbling, strength, stamina, log(value_euro), log(wage_euro) meaning that these predictors were found to have a statistically significant impact on overall_rating. Because skill_moves had a p-value > 0.05 (0.4769), it means it is not statistically. Thus, we will create a new model without this predictor and compare results. This likely means the other variables also account for the impact made by skill moves. Also, in this model we had a very high Adjusted R^2 of 0.906 which is much higher than our base model's R^2 of 0.5495 and means our model does is effective at explaining the variance in overall_rating.

```
model2 <- lm(overall_rating ~ weak_foot.1.5. + sprint_speed + dribbling + strength + stamina + log(valu

summary2 <- summary(model2)
summary2
```

```
##
## Call:
## lm(formula = overall_rating ~ weak_foot.1.5. + sprint_speed +
##     dribbling + strength + stamina + log(value_euro) + log(wage_euro),
##     data = new_fifa)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -8.1412 -1.3210 -0.1272  1.0698 14.2894
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      2.733831   0.182224  15.003  < 2e-16 ***
## weak_foot.1.5.   0.130439   0.026063   5.005 5.65e-07 ***
## sprint_speed    -0.039334   0.001719 -22.876  < 2e-16 ***
## dribbling       -0.005806   0.001530  -3.796 0.000148 ***
## strength         0.043944   0.001501  29.278  < 2e-16 ***
## stamina          0.015644   0.001630   9.600  < 2e-16 ***
## log(value_euro)  4.137359   0.020673 200.137  < 2e-16 ***
## log(wage_euro)   0.690572   0.022337  30.916  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.138 on 17691 degrees of freedom
## Multiple R-squared:  0.906,  Adjusted R-squared:  0.906
## F-statistic: 2.437e+04 on 7 and 17691 DF,  p-value: < 2.2e-16
```

Here we can see we only have significant predictors and the same R^2

Running an ANOVA to compare our two models (model 2 has one less predictor):

```
anova(model1, model2)
```

```
## Analysis of Variance Table
##
## Model 1: overall_rating ~ weak_foot.1.5. + skill_moves.1.5. + sprint_speed +
##     dribbling + strength + stamina + log(value_euro) + log(wage_euro)
## Model 2: overall_rating ~ weak_foot.1.5. + sprint_speed + dribbling +
##     strength + stamina + log(value_euro) + log(wage_euro)
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1  17690 80885
## 2  17691 80888 -1   -2.3133 0.5059 0.4769
```

As we can see from the ANOVA, we had a p-value of 0.2091. Remember, in the ANOVA test, our null hypothesis is that there is no difference between our models. Our p-value being > alpha 0.05 proves that we fail to reject the null and do not have significantly significant evidence there is a difference between the two models. We can interpret this as meaning that our smaller model is as effective as the larger one and is a better choice for us to use since it is more condensed.