ASSIGNMENT 6

Amelia Farrell

October 18th 2021

Making predictions in earning potential

We will first Fit a linear model using the `age` variable as the predictor and `earn` as the outcome.

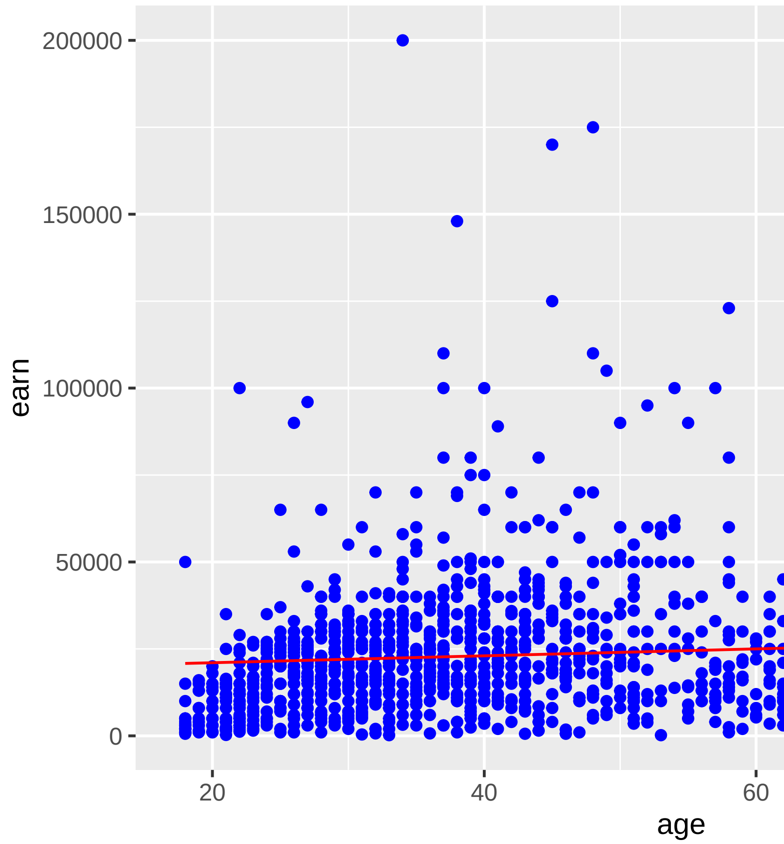
```
age_lm <- lm(earn ~ age, data = heights_df)
```

Viewing the summary of your model

```
##
## Call:
## lm(formula = earn ~ age, data = heights_df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -25098 -12622  -3667   6883 177579
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 19041.53    1571.26  12.119  < 2e-16 ***
## age            99.41      35.46   2.804  0.00514 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19420 on 1190 degrees of freedom
## Multiple R-squared:  0.006561,   Adjusted R-squared:  0.005727
## F-statistic:  7.86 on 1 and 1190 DF,  p-value: 0.005137
```

Now we can create predictions whith our model using the `predict()` function. The neww data frame ("age_predict_df") will have the same age values, but it will have the predicted earning values, using the original age data to make the predictions. The predictions are being put into the new data frame (age_predict_df).

```
age_predict_df <- data.frame(earn = predict(age_lm, heights_df), age=heights_df$age)
```



Lets plot the predictions against the original data below.

Corrected Sum of Squares Total

```
mean_earn <- mean(heights_df$earn)
sst <- sum((mean_earn - heights_df$earn)^2)
mean_earn
```

```
## [1] 23154.77
```

```
sst
```

```
## [1] 451591883937
```

Corrected Sum of Squares for Model

```
ssm <- sum((mean_earn - age_predict_df$earn)^2)
ssm
```

```
## [1] 2963111900
```

Residuals

```
residuals <- heights_df$earn - age_predict_df$earn
```

Sum of Squares for Error

```
sse <- sum(residuals^2)
```

R Squared R^2 = SSM/SST

```
r_squared <- ssm/sst
r_squared
```

```
## [1] 0.006561482
```

Number of observations

```
n <- nrow(heights_df)
n
```

```
## [1] 1192
```

Number of regression parameters

```
p <- 2
p
```

```
## [1] 2
```

Corrected Degrees of Freedom for Model (p-1)

```
dfm <- p-1
dfm
```

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

Degrees of Freedom for Error (n-p)

```
dfe <- n-p
dfe
```

```
## [1] 1190
```

Corrected Degrees of Freedom Total: DFT = n - 1

```
dft <- n-1
dft
```

```
## [1] 1191
```

Mean of Squares for Model: MSM = SSM / DFM

```
msm <- ssm/dfm
msm
```

```
## [1] 2963111900
```

Mean of Squares for Error: MSE = SSE / DFE

```
mse <- sse/dfe
mse
```

```
## [1] 376998968
```

Mean of Squares Total: MST = SST / DFT

```
mst <- sst/dft
mst
```

```
## [1] 379170348
```

F Statistic F = MSM/MSE

```
f_score <- msm/mse
f_score
```

```
## [1] 7.859735
```

Adjusted R Squared R2 = 1 - (1 - R2)(n - 1) / (n - p)

```
adjusted_r_squared <- 1 - (1 - r_squared)*(n - 1) / (n - p)
adjusted_r_squared
```

```
## [1] 0.005726659
```

Calculate the p-value from the F distribution

```
p_value <- pf(f_score, dfm, dft, lower.tail=F)
p_value
```

```
## [1] 0.005136826
```

References

Field, A., J. Miles, and Z. Field. 2012. Discovering Statistics Using R. SAGE Publications. https://books.google.com/books?id=wd2K2zC3swIC.

Lander, J. P. 2014. R for Everyone: Advanced Analytics and Graphics. Addison-Wesley Data and Analytics Series. Addison-Wesley. https://books.google.com/books?id=3eBVAgAAQBAJ.