The results below are generated from an R script.

```
# Assignment: ASSIGNMENT 6
# Name: Jordan, Andrew
# Date: 2022-05-11

## Set the working directory to the root of your DSC 520 directory
setwd("/Users/Andrew/StatClass/dsc520/")

## Load the `data/r4ds/heights.csv` to
heights_df <- read.csv("data/r4ds/heights.csv")
heights_df
```

```
##      earn   height    sex ed age     race
## 1   50000 74.42444   male 16  45    white
## 2   60000 65.53754 female 16  58    white
## 3   30000 63.62920 female 16  29    white
## 4   50000 63.10856 female 16  91    other
## 5   51000 63.40248 female 17  39    white
## 6    9000 64.39951 female 15  26    white
## 7   29000 61.65633 female 12  49    white
## 8   32000 72.69854   male 17  46    white
## 9    2000 72.03947   male 15  21 hispanic
## 10  27000 72.23493   male 12  26    white
## 11   6530 69.51215   male 16  65    white
## 12  30000 68.03161   male 11  34    white
## 13  12000 67.55693   male 12  27    white
## 14  12000 65.43059 female 12  51    white
## 15  22000 65.66285 female 16  35    white
## 16  17000 67.75877   male 12  58    white
## 17  40000 68.35184 female 14  29    white
## 18  44000 69.60957   male 13  44    white
## 19   7000 64.18457 female 12  55    black
## 20  53000 73.07461   male 13  35    black
## 21   5000 62.37553 female 13  51    white
## 22  14000 63.02393 female 14  21    white
## 23   5500 67.22990   male 14  22    white
## 24  40000 65.55111 female 12  41    white
## 25  34000 72.07965   male 12  45    white
## 26  10000 63.09113 female 12  35    black
## 27  27000 64.32355 female 16  60    white
## 28  50000 71.64285   male 16  38    white
## 29  41000 76.79309   male 16  33    white
## 30  15000 63.89391 female 14  25    white
## 31  25000 63.80262 female 12  33    white
## 32  75000 71.59223   male 17  39    white
## 33  27000 67.52196   male 17  31    white
## 34  12000 64.39435 female 12  26    white
## 35   7500 61.17822 female 14  78    white
## 36  30000 66.98388 female 14  31    black
## 37  21000 65.31646 female 12  57    white
## 38  27000 63.57419 female 14  26    white
## 39   3000 66.61100 female 15  65    white
## 40  25000 64.91176 female 12  30    white
## 41  24000 64.78968 female 12  41    white
```

```
## 42  32000 66.93769 female 18  29    white
## 43  10000 68.17281 female 17  30    white
## 44  11000 60.45066 female 12  21 hispanic
## 45  18700 64.79325 female 13  32    white
## 46  20000 61.81492 female 12  29    white
## 47   3500 71.57215   male 10  18    white
## 48  13000 67.31441   male  8  56    black
## 49  25000 69.89987   male 12  65    white
## 50  21000 69.76170   male 17  41    white
## 51  34000 67.74647 female 17  49    white
## 52   6000 60.19022 female 12  65    white
## 53  17000 71.00650   male 12  28    white
## 54  35000 71.16680   male 12  32    white
## 55   4000 72.73563   male 13  18    white
## 56  14000 68.13822 female 14  55    white
## 57  10000 66.37981 female 12  57    white
## 58  25000 69.23278   male 16  29    white
## 59  16000 63.27394 female 14  27    white
## 60  16000 61.82776   male 14  28 hispanic
## 61  16500 64.22121 female 14  43    white
## 62   4000 63.84127 female  9  68    white
## 63   3840 66.97477 female  9  52    white
## 64  22000 71.45149   male 12  39    white
## 65    200 59.61265 female 16  53    white
## 66  26000 65.79939 female 16  27    white
## 67   2500 66.45804 female 15  21    white
## 68  17000 64.60288 female 14  39    white
## 69   8000 70.44048 female 13  22    white
## 70  12000 65.92281 female 13  68    white
## 71  10000 61.85683 female 12  47    white
## 72  10000 65.78444 female 15  67    white
## 73  15000 71.83128   male 12  39    white
## 74   2400 67.04533 female  8  39 hispanic
## 75  30000 68.30551   male 12  32 hispanic
## 76  30000 70.02546   male 12  33    white
## 77  10000 61.81039 female 12  38    white
## 78   5000 62.95107 female 13  26    white
## 79  12000 65.82114 female 13  63    white
## 80  20000 70.39755 female 10  61    white
## 81  20000 68.37778 female 12  36    white
## 82  20000 69.93270   male 14  23    white
## 83   1200 66.17181 female 12  20    white
## 84    700 68.45636 female 16  32    white
## 85  20000 69.90386   male 16  27    white
## 86  10000 61.14966 female 12  22 hispanic
## 87  30000 63.36335 female 12  73    white
## 88  40000 64.14708 female 14  56    white
## 89  25000 67.31839   male 12  89    white
## 90  10000 60.67494 female 17  79    white
## 91  60000 68.84090 female 18  63    white
## 92  18000 67.68273 female 12  66    white
## 93  16040 64.49677 female 12  33    white
## 94  15000 66.81240 female 14  30    black
## 95  10000 68.74644   male 17  23    white
```

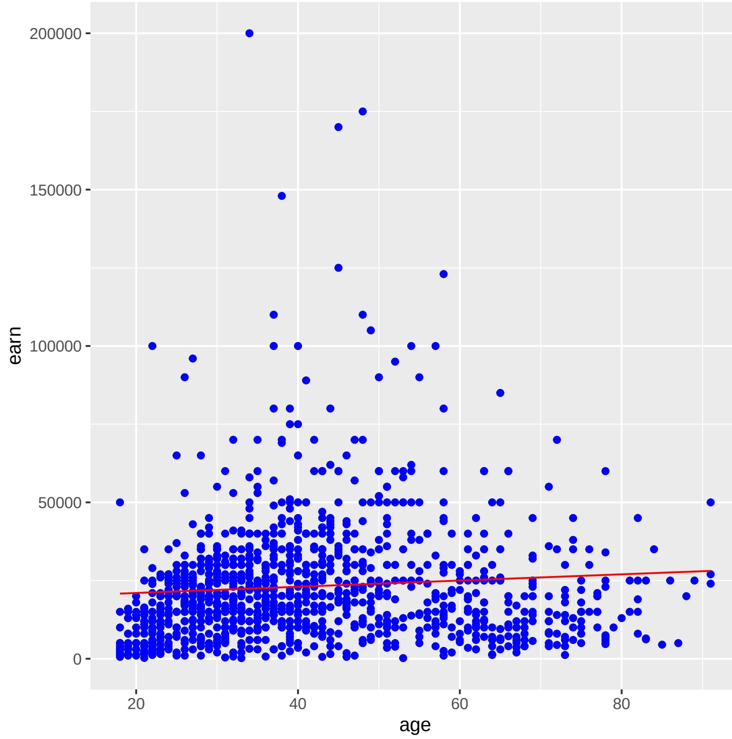
```
## 96  33000 67.06765 female 13  43    white
## 97  18000 68.13799 female 12  30    white
## 98  15000 63.34290 female 12  37    white
## 99  21000 71.38667   male 12  22    white
## 100 21000 63.98834 female 17  43    black
## 101 37000 68.48639   male 11  37    white
## 102 38000 67.51614 female 17  44    white
## 103 17000 65.60084 female 14  43 hispanic
## 104 32000 76.80019   male 16  30    white
## 105 27500 67.10538 female 12  58    white
## 106 16500 62.15164 female 12  44    white
## 107 25000 66.86762 female 18  35    white
## 108 27000 61.04220 female 18  43    white
## 109  5000 64.12329 female 12  28    white
## 110 70000 61.54482 female 16  38    white
## 111  5000 62.55624 female 12  40    white
## 112  5000 68.16377   male 16  24    white
## 113 20000 63.65513 female 15  26    white
## 114  4000 72.37352   male 15  21    white
## 115 60000 64.14708 female 16  35    white
## 116  5000 61.32670 female 13  31    white
## 117 30000 74.36640   male 12  38    white
## 118 70000 70.21016   male 14  35    white
## 119 50000 71.10619   male 16  41    white
## 120 44000 62.59484 female 12  39    white
## 121 30000 64.05496 female 14  43    white
## 122 10000 61.57362 female 16  40    white
## 123 23000 70.48020 female 17  42    white
## 124 45000 71.18591   male 17  62    white
## 125 15000 71.43364   male 14  31    white
## 126  4000 70.22885 female 14  71    white
## 127 17000 67.28086   male 14  31    white
## 128 30000 63.75869 female 12  32    white
## 129 27500 67.08652 female 12  30    white
## 130  5688 61.67960 female  8  69    white
## 131 18000 62.28600 female 13  56 hispanic
## 132 43000 68.29248   male 13  44    black
## 133 32000 61.58948 female 14  44    black
## 134 10000 68.41774 female 18  56    black
## 135 60000 73.99126   male 13  45    white
## 136 21000 67.56107 female 12  50    other
## 137  2400 62.33793 female 16  22    white
## 138  1000 66.24001 female 15  28    white
## 139 27000 68.09847   male 12  27    white
## 140  6600 59.77087 female 14  28 hispanic
## 141 16000 68.06338   male  8  43    white
## 142 90000 71.68015   male 12  26    white
## 143  8000 66.35971 female 12  42    white
## 144 20000 68.35626   male 10  32    white
## 145 15000 68.45654 female 12  18    white
## 146 12000 68.78610 female 12  60    white
## 147 24000 64.10224 female 16  46    white
## 148 20000 65.11349 female 14  39    white
## 149 19000 60.64919 female 12  46    white
```

```
## 150 10000 72.12570   male 12  49    white
## 151 40000 65.51073 female 16  34    white
## 152 25000 67.93190   male 14  64    white
## 153 25000 70.44492   male 12  24    white
## 154 25000 71.36585   male 14  32    white
## 155 19000 71.12507   male 16  61    white
## 156 44000 68.16014   male 16  48    white
## 157 15000 60.11333 female 14  49    white
## 158 17000 62.78820 female 12  36    white
## 159 24000 68.07772   male 12  56    white
## 160 23000 64.05084 female 12  37    white
## 161 13000 69.71580   male 12  74    white
## 162 65000 68.22067   male 16  46    white
## 163  7000 60.88386 female 12  63    white
## 164 40000 68.40754   male 18  63    white
## 165 15000 66.00198 female 17  43    white
## 166 20000 69.79789   male 16  25    white
##  [ reached 'max' / getOption("max.print") -- omitted 1026 rows ]
```

```r
## Load the ggplot2 library
library(ggplot2)

## Fit a linear model using the `age` variable as the predictor and `earn` as the outcome
age_lm <-  lm(earn~age, data = heights_df)

## View the summary of your model using `summary()`
summary(age_lm)
```

```
##
## 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
```

```r
## Creating predictions using `predict()`
age_predict_df <- data.frame(earn = predict(age_lm, heights_df), age=heights_df$age)

## Plot the predictions against the original data
ggplot(data = heights_df, aes(y = earn, x = age)) +
  geom_point(color='blue') +
  geom_line(color='red',data = age_predict_df, aes(y=earn, x=age))
```




```
mean_earn <- mean(heights_df$earn)
## Corrected Sum of Squares Total
sst <- sum((mean_earn - heights_df$earn)^2)
## Corrected Sum of Squares for Model
ssm <- sum((mean_earn - age_predict_df$earn)^2)
## Residuals
residuals <- heights_df$earn - age_predict_df$earn
## Sum of Squares for Error
sse <- sum(residuals^2)
## R Squared R^2 = SSM\SST
r_squared <- ssm/sst

## Number of observations
n <- sum(complete.cases(heights_df))
## Number of regression parameters
p <- 2
## Corrected Degrees of Freedom for Model (p-1)
dfm <- p-1
## Degrees of Freedom for Error (n-p)
dfe <- n-p
## Corrected Degrees of Freedom Total:   DFT = n - 1
dft <- n-1

## Mean of Squares for Model:   MSM = SSM / DFM
msm <- ssm/dfm
## Mean of Squares for Error:   MSE = SSE / DFE
mse <- sse/dfe
## Mean of Squares Total:   MST = SST / DFT
mst <- sst/dft
```

```r
## F Statistic F = MSM/MSE
f_score <- msm/mse

## Adjusted R Squared R2 = 1 - (1 - R2)(n - 1) / (n - p)
adjusted_r_squared <- 1-(1-r_squared)*(n-1)/(n-p)

## Calculate the p-value from the F distribution
p_value <- pf(f_score, dfm, dft, lower.tail=F)
```

The R session information (including the OS info, R version and all packages used):

```r
sessionInfo()

## R version 4.2.0 (2022-04-22 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19044)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.utf8  LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8 LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] car_3.0-13    carData_3.0-5 lm.beta_1.6-2 ggplot2_3.3.6 readxl_1.4.0
##
## loaded via a namespace (and not attached):
##  [1] highr_0.9        cellranger_1.1.0 pillar_1.7.0     compiler_4.2.0   tools_4.2.0
##  [6] digest_0.6.29    evaluate_0.15    lifecycle_1.0.1  tibble_3.1.7     gtable_0.3.0
## [11] pkgconfig_2.0.3  rlang_1.0.2      cli_3.3.0        yaml_2.3.5       xfun_0.30
## [16] fastmap_1.1.0    stringr_1.4.0    withr_2.5.0      dplyr_1.0.9      knitr_1.39
## [21] generics_0.1.2   vctrs_0.4.1      grid_4.2.0       tidyselect_1.1.2 glue_1.6.2
## [26] R6_2.5.1         fansi_1.0.3      rmarkdown_2.14   farver_2.1.0     purrr_0.3.4
## [31] magrittr_2.0.3   scales_1.2.0     ellipsis_0.3.2   htmltools_0.5.2  abind_1.4-5
## [36] colorspace_2.0-3 labeling_0.4.2   utf8_1.2.2       stringi_1.7.6    munsell_0.5.0
## [41] crayon_1.5.1

Sys.time()

## [1] "2022-05-15 13:01:12 EDT"
```