assignment_07_Koby-HercskyTheodore

Theodore Koby-Hercsky

5/8/2021

html_document: https://rpubs.com/theoKoby/767092

Set the working directory to the root of your DSC 520 directory

setwd("~/Documents/Bellevue University Classes/DSC520/assignments/assignment07")

The heights.csv file

```
## I am importing readr from the library so I can use the read_csv function t
o create my student survey data frame.
library(readr)
## Creating the student survey data frame by using the read_csv function to p
ull my student survey data.
heights_df <- read_csv("data/r4ds/heights.csv")
heights_df
```

```
## # A tibble: 1,192 x 6
##     earn height sex       ed   age race
##    <dbl>  <dbl> <chr>  <dbl> <dbl> <chr>
##  1 50000   74.4 male      16    45 white
##  2 60000   65.5 female    16    58 white
##  3 30000   63.6 female    16    29 white
##  4 50000   63.1 female    16    91 other
##  5 51000   63.4 female    17    39 white
##  6  9000   64.4 female    15    26 white
##  7 29000   61.7 female    12    49 white
##  8 32000   72.7 male      17    46 white
##  9  2000   72.0 male      15    21 hispanic
## 10 27000   72.2 male      12    26 white
## # … with 1,182 more rows
```

Fit a linear model

```
## Seen below is a linear model using the `height`, `age`, `sex`, `race`, and
`ed` variables as the predictors and `earn` as the outcome
earn_lm <-  lm(earn ~ height + age + sex + race + ed, data = heights_df)
## When we fit our linear model by using our variables as the predictor and e
arn as the outcome with the heights_df as our data we see coefficients for in
tercept and height, age, sex, race, and ed which is the slope for the predict
ors.
earn_lm
```

```
##
## Call:
## lm(formula = earn ~ height + age + sex + race + ed, data = heights_df)
##
## Coefficients:
##  (Intercept)        height           age       sexmale  racehispanic
##     -41478.5         202.5         178.3       10325.6       -1414.3
##    raceother     racewhite            ed
##        371.0        2432.5        2768.4
```

```
## I will view the full report by using summary of my earn_lm model
## As seen below we see that the max residual to be at 158723 with a median a
t -2208.
## While we see three stars against regression coefficients that implies the
independent variables are highly correlated with dependent variable earn with
a linear relationship.
summary(earn_lm)
```

```
##
## Call:
## lm(formula = earn ~ height + age + sex + race + ed, data = heights_df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -39423  -9827  -2208   6157 158723
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -41478.4    12409.4  -3.342 0.000856 ***
## height          202.5      185.6   1.091 0.275420
## age             178.3       32.2   5.537 3.78e-08 ***
## sexmale       10325.6     1424.5   7.249 7.57e-13 ***
## racehispanic  -1414.3     2685.2  -0.527 0.598507
## raceother       371.0     3837.0   0.097 0.922983
## racewhite      2432.5     1723.9   1.411 0.158489
## ed             2768.4      209.9  13.190  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17250 on 1184 degrees of freedom
## Multiple R-squared:  0.2199, Adjusted R-squared:  0.2153
## F-statistic: 47.68 on 7 and 1184 DF,  p-value: < 2.2e-16
```

Prediction data frame

```
## Created a new heights test data frame to incorporate different heights
newheight_test_df <- data.frame(height = c(71.2,62.5,75.5,74.8,76.8,61.1,74.2
,75.9,77.3,73.5))
newheight_test_df
```

```
##    height
## 1    71.2
## 2    62.5
## 3    75.5
## 4    74.8
## 5    76.8
## 6    61.1
## 7    74.2
## 8    75.9
## 9    77.3
## 10   73.5
```

```
## Created a new age test data frame to incorporate different ages
newage_test_df <- data.frame(age = c(21,16,28,39,50,43,69,49,76,36))
newage_test_df
```

```
##    age
## 1   21
## 2   16
## 3   28
## 4   39
## 5   50
## 6   43
## 7   69
## 8   49
## 9   76
## 10  36
```

```
## Created a new sex test data frame to incorporate different sex
newsex_test_df <- data.frame(sex = c('male','female','male','male','male','fe
male','male','male','male','male'))
newsex_test_df
```

```
##       sex
## 1    male
## 2  female
## 3    male
## 4    male
## 5    male
## 6  female
## 7    male
## 8    male
## 9    male
## 10   male
```

```
## Created a new race test data frame to incorporate different race
newrace_test_df <- data.frame(race = c('other','other','hispanic','white','ot
her','hispanic','white','hispanic','other','hispanic'))
newrace_test_df
```

```
##        race
## 1     other
## 2     other
## 3  hispanic
## 4     white
## 5     other
## 6  hispanic
## 7     white
## 8  hispanic
## 9     other
## 10 hispanic
```

```
## Created a new ed test data frame to incorporate different ed
newed_test_df <- data.frame(ed = c(17,16,17,18,19,15,16,18,15,19))
newed_test_df
```

```
##    ed
## 1  17
## 2  16
## 3  17
## 4  18
## 5  19
## 6  15
## 7  16
## 8  18
## 9  15
## 10 19
```

```
## Last we create a new test data frame that combines all the new test data f
rames we just created.
new_test_df <- data.frame(height = c(71.2,62.5,75.5,74.8,76.8,61.1,74.2,75.9,
77.3,73.5), age = c(21,16,28,39,50,43,69,49,76,36), sex = c('male','female','
male','male','male','female','male','male','male','male'), race = c('other','
other','hispanic','white','other','hispanic','white','hispanic','other','hisp
anic'), ed = c(17,16,17,18,19,15,16,18,15,19))
new_test_df
```

```
##    height age    sex     race ed
## 1    71.2  21   male    other 17
## 2    62.5  16 female    other 16
## 3    75.5  28   male hispanic 17
## 4    74.8  39   male    white 18
## 5    76.8  50   male    other 19
## 6    61.1  43 female hispanic 15
## 7    74.2  69   male    white 16
## 8    75.9  49   male hispanic 18
## 9    77.3  76   male    other 15
## 10   73.5  36   male hispanic 19
```

```
## Next we will be creating a predictions using `predict()` function by using
all the new test data frames we created.
```

```
## In this prediction I use the my new test data frames set to equal to the c
orrect variables and earn set to predict from our earn_lm and the new data th
at was created in the data frame new_test_df.

predicted_df <- data.frame(ed = newed_test_df, race = newrace_test_df, height
= newheight_test_df, age = newage_test_df, sex = newsex_test_df, earn = predi
ct(earn_lm, newdata = new_test_df))

## The predicted_df shows 10  earnings predictions that take into account var
iable that we created which we see much higher earnings for males then we do
females as seen below.

predicted_df

##    ed     race height age    sex     earn
## 1  17    other   71.2  21   male 34440.87
## 2  16    other   62.5  16 female 18693.87
## 3  17 hispanic   75.5  28   male 34774.35
## 4  18    white   74.8  39   male 43209.05
## 5  19    other   76.8  50   male 46282.24
## 6  15 hispanic   61.1  43 female 18670.95
## 7  16    white   74.2  69   male 42899.95
## 8  18 hispanic   75.9  49   male 41368.07
## 9  15    other   77.3  76   male 39945.93
## 10 19 hispanic   73.5  36   male 41332.55

## Compute deviation (i.e. residuals)
## The mean earn for our heights data frame shows a $ 23,154.77 mean for our
ten earnings for our variables.
mean_earn <- mean(heights_df$earn)
mean_earn

## [1] 23154.77

## Corrected Sum of Squares Total
sst <- sum((mean_earn - heights_df$earn)^2)
## As seen below we receive a value of 653257157128 which is the sum of squar
es total for the mean of earn from our heights data set. Which is known as th
e total amount of differences presented in a basic model that represents the
how good the mean is as a model of the observed data.
sst

## [1] 451591883937

## Corrected Sum of Squares for Model
ssm <- sum((mean_earn - predicted_df$earn)^2)
## As seen below we calculate the model sum of squares by taking the mean_ear
n that we calculated and minus it by our predicted earnings to the second. Wh
ich measures the deviation of data points away from the mean value. As our re
```

```
sult indicates a large degree of variability within the data set.
ssm

## [1] 2573418960

## Residuals
## We can see our residuals for each and every earnings in the heights data f
rame.
residuals <- heights_df$earn - predicted_df$earn
residuals

##    [1]  15559.13313  41306.13110  -4774.35134   6790.94899   4717.75753
##    [6]  -9670.95215 -13899.94510  -9368.07071 -37945.92578 -14332.54788
##   [11] -27910.86687  11306.13110 -22774.35134 -31209.05101 -24282.24247
##   [16]  -1670.95215  -2899.94510   2631.92929 -32945.92578  11667.45212
##   [21] -29440.86687  -4693.86890 -29274.35134  -3209.05101 -12282.24247
##   [26]  -8670.95215 -15899.94510   8631.92929   1054.07422 -26332.54788
##   [31]  -9440.86687  56306.13110  -7774.35134 -31209.05101 -38782.24247
##   [36]  11329.04785 -21899.94510 -14368.07071 -36945.92578 -16332.54788
##   [41] -10440.86687  13306.13110 -24774.35134 -32209.05101 -27582.24247
##   [46]   1329.04785 -39399.94510 -28368.07071 -14945.92578 -20332.54788
##   [51]   -440.86687 -12693.86890 -17774.35134  -8209.05101 -42282.24247
##   [56]  -4670.95215 -32899.94510 -16368.07071 -23945.92578 -25332.54788
##   [61] -17940.86687 -14693.86890 -30934.35134 -21209.05101 -46082.24247
##   [66]   7329.04785 -40399.94510 -24368.07071 -31945.92578 -29332.54788
##   [71] -24440.86687  -8693.86890 -19774.35134 -40809.05101 -16282.24247
##   [76]  11329.04785 -32899.94510 -36368.07071 -27945.92578 -21332.54788
##   [81] -14440.86687   1306.13110 -33574.35134 -42509.05101 -26282.24247
##   [86]  -8670.95215 -12899.94510  -1368.07071 -14945.92578 -31332.54788
##   [91]  25559.13313   -693.86890 -18734.35134 -28209.05101 -36282.24247
##   [96]  14329.04785 -24899.94510 -26368.07071 -18945.92578 -20332.54788
##  [101]   2559.13313  19306.13110 -17774.35134 -11209.05101 -18782.24247
##  [106]  -2170.95215 -17899.94510 -14368.07071 -34945.92578  28667.45212
##  [111] -29440.86687 -13693.86890 -14774.35134 -39209.05101  13717.75753
##  [116] -13670.95215 -12899.94510  28631.92929  10054.07422   2667.45212
##  [121]  -4440.86687  -8693.86890 -11774.35134   1790.94899 -31282.24247
##  [126] -14670.95215 -25899.94510 -11368.07071 -12445.92578 -35644.54788
##  [131] -16440.86687  24306.13110  -2774.35134 -33209.05101  13717.75753
##  [136]   2329.04785 -40499.94510 -40368.07071 -12945.92578 -34732.54788
##  [141] -18440.86687  71306.13110 -26774.35134 -23209.05101 -31282.24247
##  [146]  -6670.95215 -18899.94510 -21368.07071 -20945.92578 -31332.54788
##  [151]   5559.13313   6306.13110  -9774.35134 -18209.05101 -27282.24247
##  [156]  25329.04785 -27899.94510 -24368.07071 -15945.92578 -18332.54788
##  [161] -21440.86687  46306.13110 -27774.35134  -3209.05101 -31282.24247
##  [166]   1329.04785 -22899.94510 -21368.07071 -14945.92578   7667.45212
##  [171]  -9440.86687  -7693.86890 -18774.35134  -8209.05101  78717.75753
##  [176]   4329.04785 -25899.94510 -14368.07071  30054.07422  -6332.54788
##  [181] -24440.86687  16306.13110 -19774.35134 -31209.05101 -38282.24247
##  [186] -10670.95215  -7899.94510   3631.92929 -24945.92578 -26332.54788
##  [191] -10440.86687   6306.13110  -9774.35134 -23209.05101 -22282.24247
```

```
##  [196]  25329.04785  26100.05490  20631.92929  -7945.92578 -21332.54788
##  [201]  -2440.86687   6306.13110 135225.64866  -8209.05101  -6282.24247
##  [206]  14329.04785 -24899.94510 -11368.07071 -13945.92578 -36332.54788
##  [211] -14440.86687  -1693.86890  -2774.35134 -28209.05101   3717.75753
##  [216] -10670.95215  -2899.94510  -1368.07071  -7195.92578 -21332.54788
##  [221]   1559.13313 -12693.86890 -22774.35134  16790.94899  -6282.24247
##  [226]  24329.04785   2100.05490 -35368.07071 -31945.92578 -21332.54788
##  [231] -17440.86687 -16693.86890  30225.64866   6790.94899 -35282.24247
##  [236]  16329.04785 -15899.94510 -37868.07071   2054.07422 -21332.54788
##  [241] -19440.86687  -8693.86890 -28774.35134 -26209.05101 -11282.24247
##  [246]   9329.04785 -27899.94510 -21368.07071 -19945.92578 -30832.54788
##  [251] -21440.86687  -8693.86890 -31774.35134 -19209.05101 -29282.24247
##  [256]  -7670.95215 -10899.94510 -24368.07071 -36945.92578 -39232.54788
##  [261] -31248.86687  -1693.86890  -4774.35134 -28209.05101 -22282.24247
##  [266]  -8670.95215   7100.05490   8631.92929 -19945.92578 -11332.54788
##  [271] -12440.86687   8306.13110 -30774.35134 -25709.05101 -29782.24247
##  [276]   9329.04785   9100.05490 -26368.07071 -20945.92578 -14332.54788
##  [281] -19440.86687  -3693.86890 -20274.35134 -19209.05101 -28282.24247
##  [286] -14670.95215 -38899.94510 -40668.07071 -15945.92578 -14332.54788
##  [291] -22440.86687   3306.13110  -4774.35134  -8209.05101 -26282.24247
##  [296]  13329.04785 -36899.94510 -29368.07071 -29945.92578 -40332.54788
##  [301] -22440.86687  -2693.86890  -9774.35134 -18209.05101 -14282.24247
##  [306]  16329.04785 -36899.94510  38631.92929 -19945.92578 -40332.54788
##  [311]  -6440.86687   7306.13110  -7774.35134 -25209.05101 -44282.24247
##  [316]   1329.04785 -36899.94510 -16368.07071 -27945.92578 -29332.54788
##  [321]  -4440.86687 -14693.86890  -8774.35134 -20209.05101 -25282.24247
##  [326]   6329.04785 -24899.94510 -37368.07071 -16045.92578  -6332.54788
##  [331]  -8440.86687   6306.13110 -11274.35134 -31209.05101 -31282.24247
##  [336] -15670.95215 -36899.94510 -27368.07071 -24945.92578 133667.45212
##  [341] -33440.86687  -8693.86890  10225.64866 -28209.05101 -26282.24247
##  [346]  16329.04785  -2899.94510  -6368.07071  10054.07422  58667.45212
##  [351]    559.13313   5306.13110    225.64866  -5209.05101 -16282.24247
##  [356] -13670.95215 105100.05490 -11368.07071 -33445.92578 -38332.54788
##  [361] -11440.86687  21306.13110  -4774.35134 -29209.05101 -31282.24247
##  [366]   7329.04785 -34899.94510 -17368.07071 -34821.92578 -31332.54788
##  [371] -22440.86687  31306.13110 -11774.35134  -3209.05101 -40282.24247
##  [376] -13670.95215  67100.05490   -368.07071 -16945.92578 -20332.54788
##  [381] -30440.86687   6306.13110  -4774.35134 -29209.05101 -40282.24247
##  [386]  -4670.95215    100.05490 -16368.07071     54.07422  23667.45212
##  [391] -18440.86687 -10693.86890 -14774.35134 -29209.05101  -1282.24247
##  [396] -10670.95215 -30899.94510 -33368.07071 -23945.92578 -35332.54788
##  [401] -15440.86687   2306.13110   8225.64866  -8209.05101 -38282.24247
##  [406]   2329.04785 -37099.94510 -24368.07071 -15945.92578 -36332.54788
##  [411] -23440.86687  -8693.86890   5225.64866  -3209.05101 -22282.24247
##  [416]   5329.04785 -22899.94510  18631.92929 -28945.92578 -36332.54788
##  [421]  -7440.86687  70306.13110   5225.64866 -28209.05101 -45282.24247
##  [426]   3329.04785 -30899.94510 -19368.07071     54.07422  -1332.54788
##  [431]    559.13313 -11693.86890 -20774.35134 -28209.05101 -40282.24247
##  [436]  -7670.95215 -32899.94510 -21368.07071 -32945.92578  -3332.54788
##  [441]  -5440.86687   1306.13110 -26774.35134  -7209.05101 -44782.24247
```

```
##  [446]  23329.04785 -17899.94510 -28368.07071  -9945.92578 -34332.54788
##  [451]  -3440.86687  -3693.86890  15225.64866 -38209.05101 -45682.24247
##  [456]  -4670.95215  -2899.94510 -17368.07071 -17945.92578 -21332.54788
##  [461] -14328.86687  -6693.86890   6225.64866 -30209.05101  13717.75753
##  [466]   5329.04785  -4899.94510 -38368.07071 -26945.92578 -36332.54788
##  [471] -14440.86687   3306.13110  -6774.35134 -21209.05101 -16282.24247
##  [476]  11329.04785 -12899.94510  -9368.07071 -13945.92578 -19332.54788
##  [481]  -9440.86687  -7693.86890 -21774.35134 -33209.05101 -34282.24247
##  [486] -11670.95215  14100.05490  -5368.07071 -10945.92578 -33332.54788
##  [491] -31440.86687   1306.13110  -1774.35134 -28209.05101 -30282.24247
##  [496] -13670.95215 -34899.94510  -9368.07071 -25445.92578 -28332.54788
##  [501] -27440.86687  31306.13110 -22774.35134 -24209.05101 -37282.24247
##  [506]   1329.04785  12100.05490 -31368.07071 -26945.92578 -21332.54788
##  [511]  -4440.86687  -6693.86890  -8774.35134 -30709.05101 -38282.24247
##  [516]  -3670.95215  22100.05490 -16368.07071 -31945.92578 -26332.54788
##  [521] -22440.86687  16306.13110   5225.64866 -14209.05101 -31282.24247
##  [526]   1329.04785 -39899.94510 -40368.07071 -31945.92578 -36332.54788
##  [531]  -9440.86687   1306.13110  -4774.35134 -31209.05101 -36282.24247
##  [536]  -2670.95215   2100.05490  -1368.07071 -14945.92578 -22332.54788
##  [541] -16840.86687 -11693.86890  -4774.35134 -25209.05101 -21282.24247
##  [546] -12670.95215  27100.05490 -26368.07071 -14945.92578  -6332.54788
##  [551] -16440.86687   9306.13110 -19774.35134 -21209.05101 -11282.24247
##  [556]   -670.95215  -7899.94510 -24968.07071 -17945.92578 -11332.54788
##  [561] -17440.86687   6306.13110 -24774.35134  -3209.05101 -33782.24247
##  [566]  16329.04785  62100.05490  58631.92929 -18945.92578 -31332.54788
##  [571]  -1440.86687   7306.13110  25225.64866 -25209.05101 -31282.24247
##  [576] -13670.95215  17100.05490 -29368.07071 -11945.92578    667.45212
##  [581] -32440.86687   1306.13110  -6774.35134 -33209.05101 -15282.24247
##  [586]  -1670.95215 -26899.94510  -6368.07071 -36945.92578 -25332.54788
##  [591]    559.13313   8306.13110  -9774.35134  -3209.05101  -1282.24247
##  [596]  -4670.95215 -17899.94510  -1368.07071  -5945.92578  -1332.54788
##  [601]    559.13313  -6693.86890  -4774.35134 -20209.05101 -44282.24247
##  [606]   1329.04785 -32899.94510  18631.92929 -27945.92578 -16332.54788
##  [611] -12440.86687 -10693.86890 -32774.35134 -24609.05101 -26282.24247
##  [616]  81329.04785   7100.05490 -14368.07071 -38945.92578 -23332.54788
##  [621] -17440.86687  -8693.86890 -20774.35134 -36009.05101 -13282.24247
##  [626]   4329.04785 -32899.94510 -19368.07071 -26945.92578  -8332.54788
##  [631]   1559.13313  -8693.86890 -28774.35134 -31209.05101 -36282.24247
##  [636]   2329.04785 -16899.94510 -19368.07071 -16945.92578 -21332.54788
##  [641] -24440.86687  -3693.86890 -27774.35134 -28209.05101 -20282.24247
##  [646]  -8670.95215 -39899.94510 -14368.07071 -11945.92578 -26332.54788
##  [651]  20559.13313   1306.13110 -26274.35134 -13209.05101 -26282.24247
##  [656]   3329.04785 -17899.94510 -38368.07071 -23945.92578 -31332.54788
##  [661] -19440.86687   9306.13110  -3274.35134  51790.94899  -8282.24247
##  [666]  11329.04785  -6999.94510 -29368.07071 -29945.92578   3667.45212
##  [671]   5559.13313   2806.13110 -20774.35134  -5209.05101 -45282.24247
##  [676]  -4670.95215 -27899.94510 -17368.07071 -35445.92578 -23332.54788
##  [681] -20440.86687  -8693.86890 -25274.35134 -32209.05101 -29282.24247
##  [686] -12670.95215 -40399.94510  -1368.07071 -14945.92578 -13332.54788
##  [691]  -1440.86687 -14693.86890 -30674.35134 -18209.05101 -14282.24247
```

```
##  [696]  -5670.95215 -34899.94510 -24368.07071 -24945.92578 -23332.54788
##  [701] -22440.86687 -11693.86890 -33574.35134 -29209.05101 -29282.24247
##  [706]  11329.04785 -17899.94510 -19368.07071 -24945.92578 -14332.54788
##  [711] -14440.86687  -9693.86890 -17774.35134 -41409.05101 -28282.24247
##  [716]   6329.04785 -22899.94510  33631.92929 -36445.92578  -5332.54788
##  [721]  -4440.86687   6306.13110 -33774.35134  16790.94899 -11282.24247
##  [726]  -9670.95215  -7899.94510 -30368.07071 -27945.92578 -11332.54788
##  [731] -15440.86687  -9693.86890  -4774.35134 -34209.05101 -31282.24247
##  [736] -10670.95215 -32899.94510  -8368.07071 -27945.92578 -23332.54788
##  [741]  -4440.86687  -6193.86890 -24774.35134 -34709.05101 -25282.24247
##  [746]   2329.04785 -21899.94510 -36368.07071 -17945.92578 -11332.54788
##  [751]  88559.13313 -10693.86890 -32774.35134 -35609.05101 -36282.24247
##  [756]  16329.04785 -27899.94510 -37368.07071 -23945.92578 -41132.54788
##  [761] -11440.86687  -2693.86890 -28774.35134 -42609.05101 -18282.24247
##  [766] -14670.95215 -27899.94510 -26368.07071 -14945.92578 -26332.54788
##  [771]   5559.13313  11306.13110 -29774.35134  -8209.05101 -30282.24247
##  [776]   9329.04785 -22899.94510 -20368.07071 -22945.92578 -17332.54788
##  [781]  -9440.86687  21306.13110 -26774.35134 -24209.05101  -1282.24247
##  [786] -13670.95215 -18899.94510 -36368.07071  -6945.92578 -17332.54788
##  [791] -28440.86687  71306.13110  22225.64866 -29409.05101 -16282.24247
##  [796]  -3670.95215 -27899.94510 -10368.07071  -6945.92578 -11332.54788
##  [801] -22440.86687 -13693.86890 -13774.35134 -15209.05101 -34282.24247
##  [806]    329.04785 -36899.94510 -36068.07071 -24945.92578 -24332.54788
##  [811] -29740.86687  -9193.86890 -33074.35134 -37809.05101 -36282.24247
##  [816]   6329.04785  -7899.94510 -13368.07071   2054.07422   3667.45212
##  [821]  20559.13313  17306.13110  -9774.35134 -24209.05101  -8282.24247
##  [826]  -3670.95215 -38399.94510  54631.92929 -28945.92578 -32332.54788
##  [831] -29440.86687  31306.13110 -28774.35134  56790.94899 -29782.24247
##  [836]  13329.04785 -12899.94510 -31368.07071 -24945.92578 -34832.54788
##  [841]  -8440.86687  -1693.86890 -12774.35134 -33209.05101   3717.75753
##  [846]   1329.04785 -42499.94510 -25368.07071 -29945.92578 -27332.54788
##  [851] -29440.86687  21306.13110 -29774.35134 -18209.05101 -34282.24247
##  [856]  39329.04785 -24899.94510 -40368.07071 -18945.92578 -30332.54788
##  [861] -21440.86687   1306.13110  -9774.35134 -42609.05101 -21282.24247
##  [866]  17329.04785 -27899.94510 -37868.07071 -20940.92578  20667.45212
##  [871]  -9440.86687  31306.13110   5225.64866 -17209.05101 -24282.24247
##  [876]   2329.04785 -36791.94510 -11368.07071 -33945.92578 -14332.54788
##  [881] -30440.86687 -14277.86890 -29774.35134   1790.94899 -28282.24247
##  [886] -12670.95215 -28899.94510  -6368.07071 -24945.92578 -29332.54788
##  [891] -33440.86687  13306.13110 -33274.35134 -40209.05101 -11282.24247
##  [896]  51329.04785 -21899.94510   8631.92929 -24945.92578 -32332.54788
##  [901] -24440.86687  11306.13110 -18774.35134 -23209.05101  53717.75753
##  [906]  -8670.95215  -7899.94510 -34368.07071 -31945.92578 -30332.54788
##  [911]  -9440.86687   6306.13110  -4774.35134   4790.94899 -29282.24247
##  [916]   7329.04785 -32899.94510  18631.92929 -17945.92578 -35132.54788
##  [921] -10440.86687   1306.13110 -27774.35134 -13209.05101 -23282.24247
##  [926]   1329.04785 -39899.94510 -11368.07071 -27945.92578  -1332.54788
##  [931] -19440.86687  17306.13110 -24774.35134 -30209.05101 -36282.24247
##  [936]  11329.04785 -19899.94510 -21368.07071 -20945.92578 -21332.54788
##  [941]  -6440.86687  -5693.86890 -22774.35134 -23209.05101 -21282.24247
```

```
##  [946]   6329.04785 -24899.94510 -35368.07071 -23945.92578   3667.45212
##  [951]  -9440.86687  -6693.86890 -32774.35134 -33209.05101 -32282.24247
##  [956]  -3670.95215 -32899.94510 -40168.07071 -24945.92578 -31332.54788
##  [961] -32440.86687 -14693.86890 -24774.35134 -41709.05101 -28282.24247
##  [966] -14670.95215 -14899.94510 -31368.07071 -36945.92578 -29332.54788
##  [971] -15440.86687  17306.13110 -27774.35134  -8209.05101  -6282.24247
##  [976]  -3670.95215 -34899.94510 -16368.07071 -11945.92578 -19332.54788
##  [981] -12940.86687 -15693.86890 -11774.35134 -25209.05101 -26282.24247
##  [986]  -2670.95215 -27899.94510  11631.92929 -13945.92578 -37332.54788
##  [991] -24440.86687  24306.13110 -17774.35134 -39209.05101   3717.75753
##  [996]  36329.04785 -14899.94510 -36368.07071 -21945.92578 -20332.54788
## [1001] -22440.86687  14306.13110  15225.64866 -37209.05101 -30282.24247
## [1006]  16329.04785 -17899.94510 -21368.07071 -21945.92578  -7332.54788
## [1011]  -9440.86687   1306.13110  -1774.35134 -39209.05101 -26282.24247
## [1016]  21329.04785 -28899.94510 -29368.07071 -14945.92578  28667.45212
## [1021]   3559.13313  -4693.86890 -19774.35134 -26209.05101  13717.75753
## [1026]  -3670.95215 -17899.94510 -23368.07071  30054.07422 -11332.54788
## [1031] -26440.86687  11306.13110  -7774.35134   3790.94899 -34282.24247
## [1036]   5329.04785 -29899.94510  38631.92929 -34945.92578 -16332.54788
## [1041]  -6440.86687 -10693.86890 -24774.35134 -34209.05101 -26282.24247
## [1046]  -6670.95215 -27899.94510 -21368.07071 -11945.92578 -41067.54788
## [1051]    559.13313   7306.13110  -7774.35134 -29209.05101 -14282.24247
## [1056]  -8670.95215 -35899.94510 -13368.07071  20054.07422 -29332.54788
## [1061] -19440.86687  -3693.86890  -8774.35134 -17209.05101 -27562.24247
## [1066]  -6670.95215  37100.05490   6631.92929 -21945.92578  43667.45212
## [1071]    559.13313 -13693.86890 -25774.35134 -13209.05101  33717.75753
## [1076]  16329.04785 -30899.94510 158631.92929 -35945.92578 -39332.54788
## [1081] -10440.86687 -16693.86890 -19774.35134 -37209.05101 -41282.24247
## [1086] -17170.95215 -14899.94510 -32368.07071  20054.07422 -40132.54788
## [1091] -33040.86687 -11693.86890 -13774.35134    790.94899 -17282.24247
## [1096] -15670.95215 -30899.94510 -21368.07071 -38745.92578  -4332.54788
## [1101] -26440.86687  16306.13110  -9774.35134 -25209.05101 -16282.24247
## [1106]   1329.04785 -30899.94510 -31368.07071 -21945.92578 -21332.54788
## [1111]  -3440.86687 -12693.86890 -22774.35134 -30209.05101 -20282.24247
## [1116]  -3670.95215 -12899.94510 -39368.07071 -34945.92578  11667.45212
## [1121] -32440.86687   6306.13110 -29774.35134 -22209.05101 -29282.24247
## [1126]  -6670.95215   -899.94510 -20368.07071 -33945.92578 -16332.54788
## [1131] -31440.86687 -15093.86890 -30774.35134 -23209.05101   5717.75753
## [1136]  41329.04785 -27899.94510  48631.92929 -13945.92578 -39332.54788
## [1141] -18440.86687   5306.13110 -19774.35134  -1209.05101 -36282.24247
## [1146]  -1670.95215 -12899.94510  -9368.07071  -3945.92578 -31332.54788
## [1151] -14440.86687   5306.13110 -17774.35134 -20209.05101 -21282.24247
## [1156]  31329.04785 -22899.94510 -11368.07071 -32945.92578 -34332.54788
## [1161]  -4440.86687 -15693.86890 -14774.35134  -3209.05101 -36282.24247
## [1166]  -2670.95215 -31899.94510 -25368.07071 -21945.92578 -29332.54788
## [1171] -30440.86687  41306.13110   8225.64866 -12209.05101   3717.75753
## [1176]   8329.04785 -12899.94510 -29368.07071 -19945.92578 -26332.54788
## [1181]  -9440.86687 -16693.86890 -31774.35134  66790.94899   8717.75753
## [1186]  39329.04785 -32899.94510 -22368.07071 -24945.92578 -33332.54788
## [1191]  25559.13313 -12693.86890
```

```r
## Sum of Squares for Error
## Next we calculate the Sum of Squares for Error which is the deviations predicted from actual empirical values of data. As the error is the difference between the observed value and the predicted value which we want to be as minimal as possible.
sse <- sum(residuals^2)
sse
```

```
## [1] 736834662129
```

```r
## R Squared
## In R Squared we take the Corrected Sum of Squares for Model (SSM) and divide it by Corrected Sum of Squares Total (sst) Which is a statistical measure of how well the regression predictions approximate the real data points as seen below
r_squared <- ssm / sst
r_squared
```

```
## [1] 0.00569855
```

```r
## Number of observations
## I use the nrow function to determine the number of observations in our heights data set
n <- NROW(heights_df$earn)
n
```

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

```r
## Number of regression paramaters
## As seen below we set the number of regression parameters to 8
p <- 8
p
```

```
## [1] 8
```

```r
## Corrected Degrees of Freedom for Model
## To get the Corrected Degrees of Freedom for Model we take the Number of regression parameters and minus it by three.
dfm <- p - 3
dfm
```

```
## [1] 5
```

```r
## Degrees of Freedom for Error
## Calculate the Degrees of Freedom for Error we take the Number of observations and minus it by the Number of regression parameters.As the Corrected Degrees of Freedom Total shows the number of independent values that can vary in an analysis without breaking any constraints
dfe <- n - p
dfe
```

```
## [1] 1184
```

```
## Corrected Degrees of Freedom Total:   DFT = n - 1
## When calculating the Corrected Degrees of Freedom Total we take the Number of observations and minus it by one As the Corrected Degrees of Freedom Total shows the number of independent values that can vary in an analysis without breaking any constraints that have been corrected.
dft <- n -1
dft
```

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

```
## Mean of Squares for Model:   MSM = SSM / DFM
## When calculating the Mean of Squares for Model we take the Corrected Sum of Squares for Model and divide it by the Corrected Degrees of Freedom for Model which gives us 514683792
msm <- ssm / dfm
msm
```

```
## [1] 514683792
```

```
## Mean of Squares for Error:   MSE = SSE / DFE
## When we calculate the Mean of Squares for Error we are taking the Sum of Squares for Error and dividing it by the Degrees of Freedom for Error which gives us 622326573
mse <- sse /dfe
mse
```

```
## [1] 622326573
```

```
## Mean of Squares Total:   MST = SST / DFT
## When we calculate the Mean of Squares Total we take the Corrected Sum of Squares Total and divide it by the Corrected Degrees of Freedom Total which gives us 548494674
mst <- sst / dft
mst
```

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

```
## F Statistic
## When we calculate the F Statistic we take the Mean of Squares for Model and divide by the Mean of Squares Total to get 0.8270317 This shows that the coefficients that was used in the model to improved the models fit by a large percent seen as 82.70%.
f_score <- msm / mse
f_score
```

```
## [1] 0.8270317
```

```
## Adjusted R Squared R2 = 1 - (1 - R2)(n - 1) / (n - p)
## When we calculate the Adjusted R Squared we take one and minus it by (one - R Squared R^2)then times it by (Number of observations - one) then divide it by (Number of observations - Number of regression parameters) which gives us -0.001949506. As Adjusted R2 indicates how well terms fit a curve or line a
```

```r
s we see it is a negative percentage meaning it does not fit well with the cu
rve
adjusted_r_squared <- 1 - (1 - r_squared)*(n-1) / (n-p)
adjusted_r_squared
```

```
## [1] -0.0001799213
```