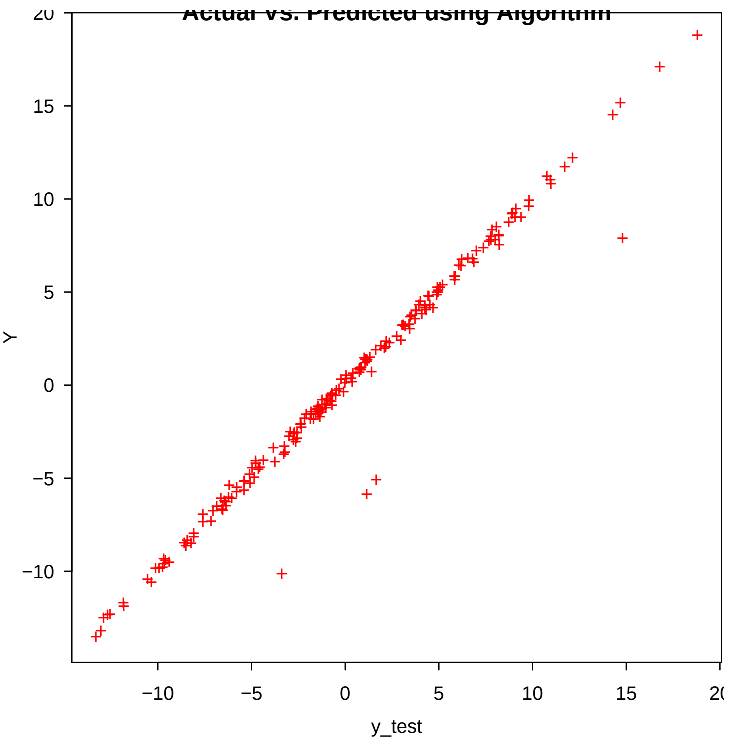
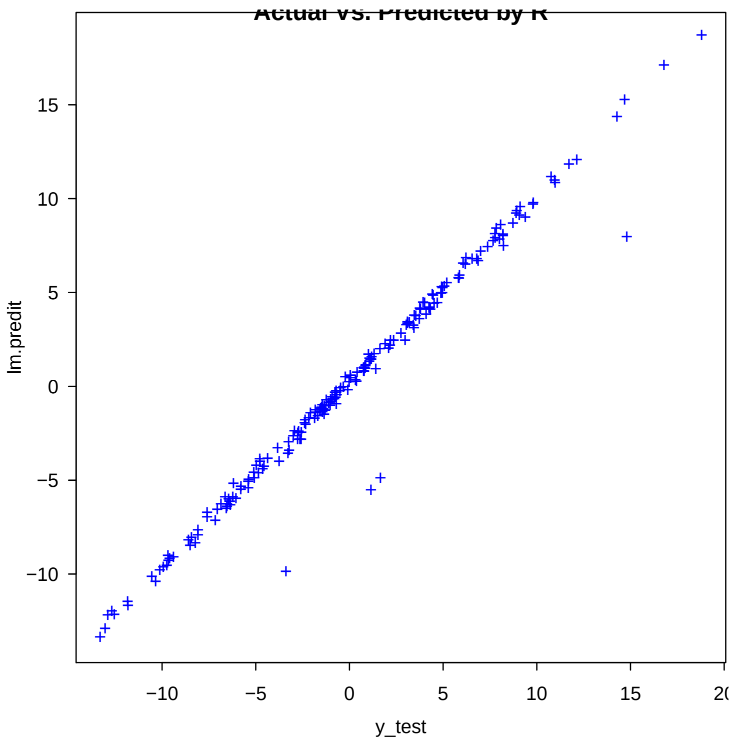
The results below are generated from an R script.

```
#Steven A Vasquez
#Date Created 2/18/2020
#Last Modified 3/5/2020

#Set working directory
setwd('~/R/ML')

## Error in setwd("~/R/ML"):  cannot change working directory

#Import needed libraries
library(tidyverse)

## - Attaching packages ------------------------- tidyverse 1.3.0 -
## v ggplot2 3.3.0     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.5
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## - Conflicts ---------------------------- tidyverse_conflicts() -
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

#Import data
df <- read_csv('TrainData_Group1.csv')

## Parsed with column specification:
## cols(
##   X1 = col_double(),
##   X2 = col_double(),
##   X3 = col_double(),
##   X4 = col_double(),
##   X5 = col_double(),
##   Y = col_double()
## )

#Create a column of 1
df$c <- 1

#Divide data into training and testing data
indexes <- sample(1:nrow(df),size = .8*nrow(df))
training <- df[indexes,]
testing <- df[-indexes,]
train <- df[-indexes,] #Needed for later

#Train the model using training data set
#Create X training Matrix. Seperate all values besides the Y into one variable
x_train <- training[,c(1,2,3,4,5,7)]

#Create Y training matrix with the remaining vector in training.
y_train <- training[,6]

#Change to matrices to do matrix multiplication
x_matrix <- as.matrix(x_train)
y_matrix <- as.matrix(y_train)
```

```r
#b <-  (X^t*X)^-1(X^t*Y)
#Setting up values to use in a solve function
#A is the same as (X^t*X)
#B is the same asX^t*Y)
A <- t(x_matrix)%*%x_matrix
B <- t(x_matrix)%*%y_matrix
#y_b <- ((t(x_matrix)%*%x_matrix)**-1)%*%t(x_matrix)%*%y_matrix

#The solve function is used for matrix multplication, solve(A,b) is essentially A^-1 %*% b
#Solve takes the dot matrix of a and x where x = b which is a matrix
b <- solve(A,B)

#Make prediction using test data set
x_test <- testing[,c(1,2,3,4,5,7)]
y_test <- testing[,6]

#Convert to Matrices
x_test <- as.matrix(x_test)
y_test <- as.matrix(y_test)

#Dot product %*% of b and x_test. this is the prediction
#Will give Y which is compared to y in test to calculate RSS
Y <- x_test%*%b

#Calculate R2 by making a fucntion that will calculate RSS, TSS,
errors <- function(y_test, Y) {
  RSS = sum((y_test - Y)^2)  #taking the sum of the difference between y calculated and y from the test
  TSS = sum((y_test - mean(Y))^2) #taking the sum of the difference between y calulcated the average of
  R2 <- 1 - RSS/TSS  #Coefficient of determination
  RMSE <- sqrt(mean((Y - y_test)^2))
  return(list(R2 = R2, RMSE = RMSE))
}

#Compare Predicted Y outcomes with the Y from test data
#X-axis has the y from the data set, y-axis has the values from the built model
plot(x=y_test, y=Y, pch = "+", col='red', main = "Actual Vs. Predicted using Algorithm")
```



```
#Use error function to print out R2
error <- errors(y_test,Y)
print(paste("R^2 from my algorithm",error$R2, sep = " "))

## [1] "R^2 from my algorithm 0.973623945784276"

#Now lets compare the built model to R's lm function
#Save the results of the function in a variable to use the predict function
lm.fit <-lm(formula=Y~., data=train)
#The predict function will predict
lm.predit <- predict(lm.fit, newdata = NULL, type='response')

#plot to display how accurate the actuql data is to the predicted values by R
#X-axis will have the actual values from the data set, y-axis will have the R predicted values
plot(x=y_test, y=lm.predit, pch = "+", col='blue', main = "Actual Vs. Predicted by R")
```



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

```
sessionInfo()
```

```
## R version 3.6.3 (2020-02-29)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Mojave 10.14.6
##
## Matrix products: default
## BLAS:   /System/Library/Frameworks/Accelerate.framework/Versions/A/Frameworks/vecLib.framework/Versio
## LAPACK: /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] forcats_0.5.0   stringr_1.4.0   dplyr_0.8.5     purrr_0.3.3     readr_1.3.1
##  [6] tidyr_1.0.2     tibble_2.1.3    ggplot2_3.3.0   tidyverse_1.3.0 knitr_1.28
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.3       cellranger_1.1.0 pillar_1.4.3     compiler_3.6.3   dbplyr_1.4.2
##  [6] highr_0.8        tools_3.6.3      lubridate_1.7.4  jsonlite_1.6.1   evaluate_0.14
## [11] lifecycle_0.2.0  nlme_3.1-144     gtable_0.3.0     lattice_0.20-38  pkgconfig_2.0.3
## [16] rlang_0.4.5      reprex_0.3.0     cli_2.0.2        rstudioapi_0.11  DBI_1.1.0
## [21] yaml_2.2.1       haven_2.2.0      xfun_0.12        withr_2.1.2      xml2_1.2.5
## [26] httr_1.4.1       fs_1.3.2         hms_0.5.3        generics_0.0.2   vctrs_0.2.4
## [31] grid_3.6.3       tidyselect_1.0.0 glue_1.3.2       R6_2.4.1         fansi_0.4.1
## [36] readxl_1.3.1     modelr_0.1.6     magrittr_1.5     backports_1.1.5  scales_1.1.0
## [41] rvest_0.3.5      assertthat_0.2.1 colorspace_1.4-1 stringi_1.4.6    munsell_0.5.0
## [46] broom_0.5.5      crayon_1.3.4
```

```
Sys.time()
```

```
## [1] "2020-04-03 19:11:45 EDT"
```