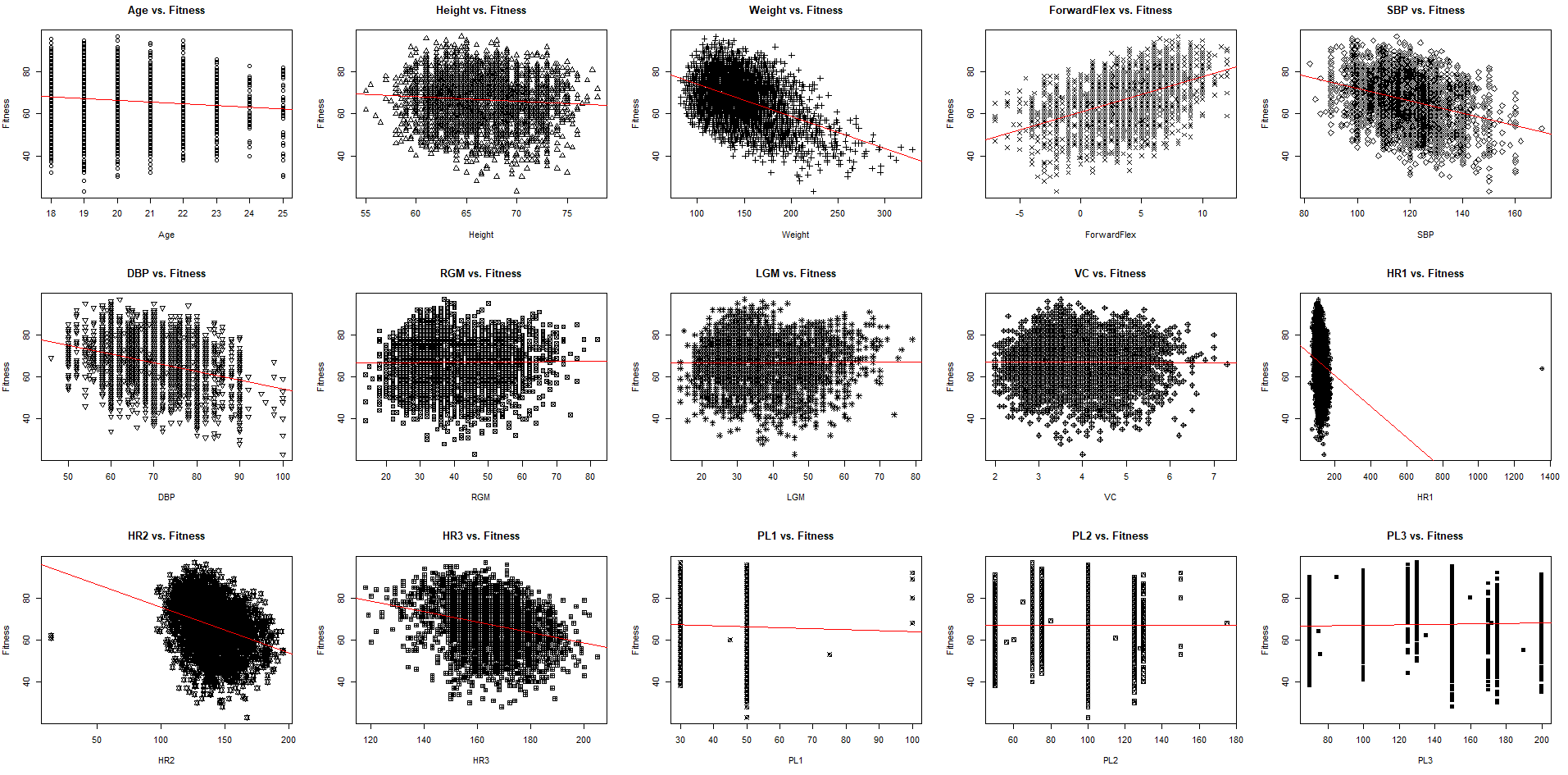
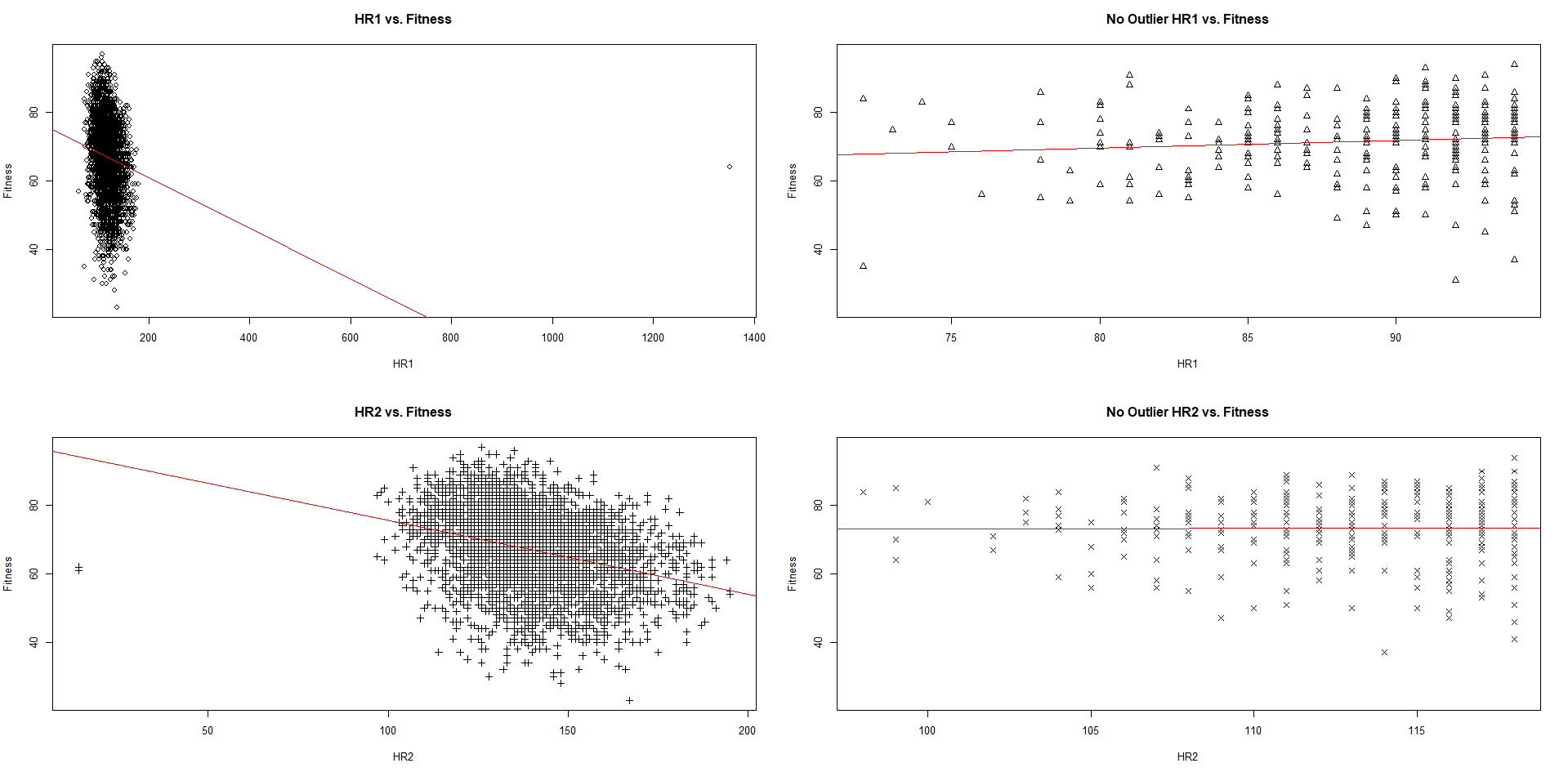
CPSC 375: Project 1

1. Exploratory Data Analysis

With the data given in the CSV file, the first step is to create scatterplots of the data on the most important data columns. This is to determine what the actual trends are for the data with respect to the FitnessScore. Looking at the CSV file containing 40 individual variables that measure fitness score, the 15 most important ones with data that has some correlation with fitness score are chosen. The list includes age, height, weight, forward flex, SBP, DBP, RGM, LGM, VA, HR1-3, and PL1-3. Each variable was graphed against the fitness score value obtained from the CSV file.



In looking at the data, a lot of the data is centralized around the red trendline in all the graphs. For most of the variables, many of the data points lie in a cluster with no real outliers in the data. The variables that have no visible outliers include age, weight, height, forward flex, SBP, DBP RGM, LGM, VC, HR3, PL2, and PL3. In looking at the graphs of HR1, HR2, and PL1, there are outliers that are far away from the center of the data. Thus these data points need to be filtered out as outliers to obtain a more centralized data.



Interpreting the 15 data column, the variables all have a large cluster of points around a centralized area. The shape of the graph defines a region rather than a consistent line of points that can be marked. Finding a line or function that represents a region of information that is not linear or non-linear means that finding a good model to relate FitnessScore with the data and have a good correlation value, R2, will be difficult to find.

The summary of the data was computed for each of the 15 columns given above to see what are the range of values given are (min, median, max, and R2 values). The values themselves have been normalized around the mean using the function summary() for each of the 15 columns is given below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Variable** | **Min** | **Median** | **Max** | **R2** |
| Age | -44.479 | 0.521 | 30.366 | 0.012880 |
| Height | -43.103 | 0.682 | 29.713 | 0.005483 |
| Weight | -32.226 | 0.002 | 30.487 | 0.213700 |
| ForwardFlex | -34.512 | 0.724 | 24.261 | 0.217800 |
| SBP | -35.338 | 0.489 | 29.171 | 0.106300 |
| DBP | -37.132 | 0.299 | 30.387 | 0.110200 |
| RGM | -43.990 | 0.938 | 30.117 | 0.000151 |
| LGM | -43.990 | 0.975 | 30.132 | 0.000079 |
| VC | -43.897 | 1.019 | 30.065 | 0.000040 |
| HR1 | -42.497 | 0.725 | 88.423 | 0.031000 |
| HR2 | -39.601 | 0.769 | 27.922 | 0.101300 |
| HR3 | -39.167 | 0.845 | 30.354 | 0.076090 |
| PL1 | -43.479 | 0.588 | 29.588 | 0.001797 |
| PL2 | -43.887 | 1.055 | 30.078 | 0.000001 |
| PL3 | -39.504 | 0.994 | 29.697 | 0.001848 |

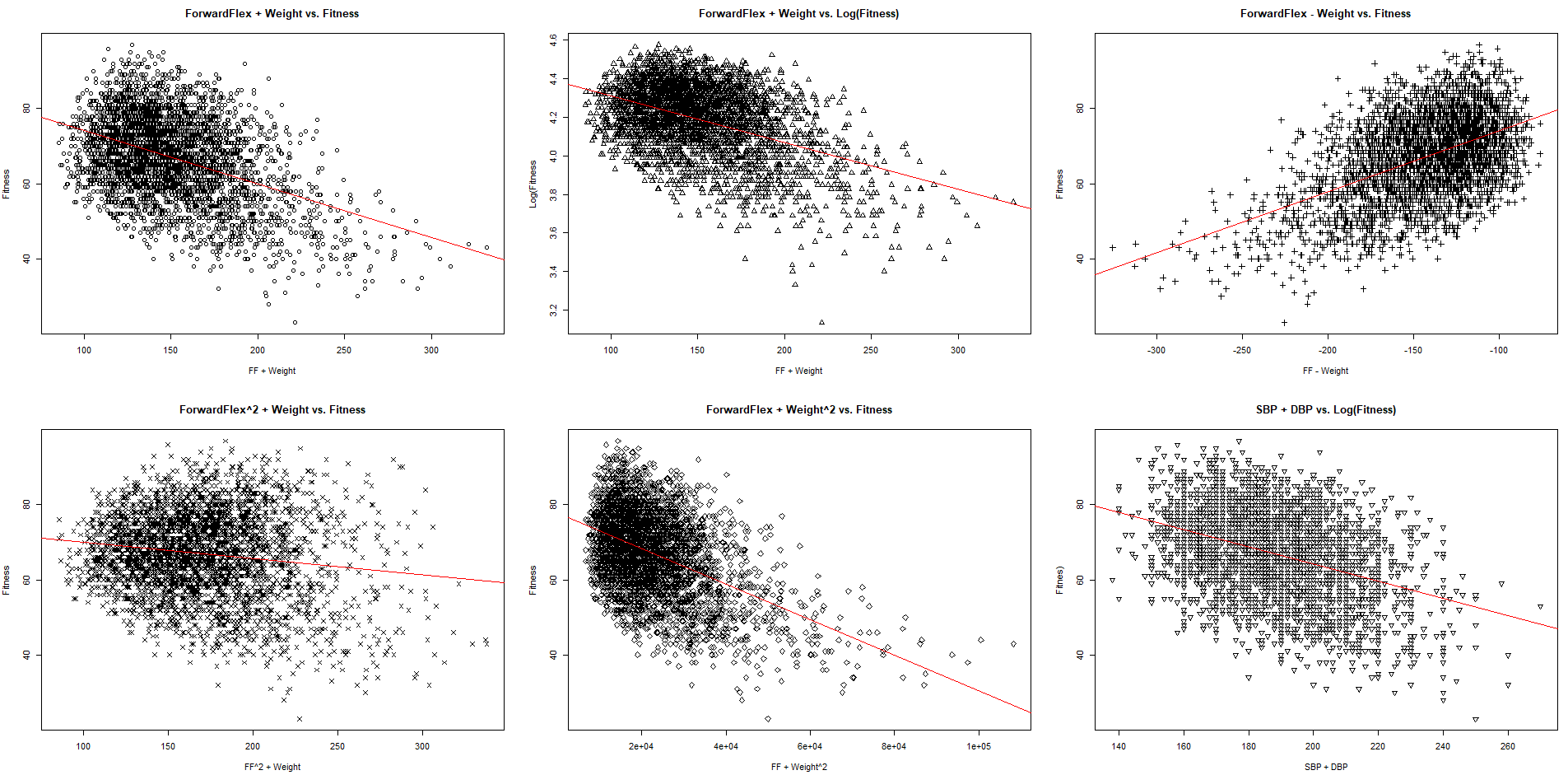
In looking at the R2 (correlation) of the data to the fitness scores, the highest value R2 means that the data fits the red trend line the best. With ForwardFlex and Weight being the 2 highest, and SBP and DBP being the next 2 highest, these are the variables to consider in the model.

1. Models Estimating Fitness Score

The best way to find a model is to interpret different combinations of these variables. The following 5 combinations of variables are considered for this assignment, most of which is related to Weight and ForwardFlex because they have the highest correlation constant.

1. SBP + DBP
2. ForwardFlex + Weight
3. ForwardFlex – Weight
4. ForwardFlex2 + Weight
5. ForwardFlex + Weight2

Each of the previous 5 models were tested to see how the data aligns with the actual trendline of the program. Each of these sets of variables contribute as a value of X in the linear regression equation Y = mX + b.



All 5 linear combinations of variables make the graphs more centralized in the chart and continues to show a weak relationship with the trendline. The first two graphs from top to bottom, left to right are the differences when the Y value (Fitness Score) is not logarithmic and when it is (Log Y). After the transformation of Y, little change in the actual shape of the graph occurs. Thus, for the duration of this analysis, Y is always considered to be itself with no other functional changes. After completing the analysis of these new functional models, the m (slope of the X), b (intercept of the X), and R2 values after applying the function comes out to be:

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **m** | **b** | **R2** |
| SBP + DBP | -0.229262 | 110.055390 | 0.1423 |
| ForwardFlex + Weight | -0.143060 | 88.571652 | 0.1800 |
| ForwardFlex – Weight | 0.161940 | 90.251095 | 0.2455 |
| ForwardFlex2 + Weight | -0.043570 | 74.340369 | 0.0219 |
| ForwardFlex + Weight2 | 77.79 | -0.000474 | 0.2250 |

After looking at the 5 models, the best model with the highest R2 value is Y = 0.161940(ForwardFlex – Weight) + 90.251095. This was used in the function calculateFitness() and computed for the final MSE value. After running the code, the MSE for this final model is 9.721688 which is decent for the calculation of fitness from ForwardFlex and Weight. The number of N/A values in this group is 0 as well because there were no non-applicable data points in either of these two variables.

In looking at these different combinations, it seems like FitnessScore is a complex value to determine from just one or two variables alone. Based on the correlation constant R2, two variables seem like not enough information to determine an accurate fitness score. This mirrors real life because a person’s measure of fitness is not just simply on height, weight, age, etc., but rather a combination of the terms together. Several contributing factors, in different combinations, determine the actual fitness score.

1. Code

varPlot <- function(xvar, yvar, title, xlabel, ylabel, winnum, func) {

plot(xvar, yvar, main=title, xlab=xlabel, ylab=ylabel, pch=winnum)

abline(func, col="red")

cor(xvar, yvar)

cor.test(xvar, yvar)

summary(func)

}

elimOutliers <- function(vals, ...) {

quant <- quantile(vals, probs=c(0.25,0.75), na.rm = TRUE)

rng <- 1.5 \* IQR(vals, na.rm = TRUE)

newVals <- vals

newVals[vals < (quant[1] - rng)] <- NA

newVals[vals > (quant[2] - rng)] <- NA

return(newVals)

}

fitness\_score <- function(input) {

weight <- input$Wt

forwFlex <- input$FF

return(0.161940 \* (forwFlex - weight) + 90.251095)

}

setwd("~")

fitnessData <- read.csv(file = 'Fitness and Obesity - F20.csv')

age <- fitnessData$Age

height <- fitnessData$Ht

weight <- fitnessData$Wt

forwFlex <- fitnessData$FF

sbp <- fitnessData$SBP

dbp <- fitnessData$DBP

rgm <- fitnessData$RGM

lgm <- fitnessData$LGM

vc <- fitnessData$VC

hr1 <- fitnessData$HR.1

hr2 <- fitnessData$HR.2

hr3 <- fitnessData$HR.3

pl1 <- fitnessData$PL.1

pl2 <- fitnessData$PL.2

pl3 <- fitnessData$PL.3

fitness <- fitnessData$FitnessScore

newHR1 <- elimOutliers(fitnessData$HR.1)

newHR2 <- elimOutliers(fitnessData$HR.2)

dev.new()

par(mfrow=c(2,2))

varPlot(hr1, fitness, "HR1 vs. Fitness", "HR1", "Fitness", 1, lm(fitness~hr1))

varPlot(newHR1, fitness, "No Outlier HR1 vs. Fitness", "HR1", "Fitness", 2, lm(fitness~newHR1))

varPlot(hr2, fitness, "HR2 vs. Fitness", "HR2", "Fitness", 3, lm(fitness~hr2))

varPlot(newHR2, fitness, "No Outlier HR2 vs. Fitness", "HR2", "Fitness", 4, lm(fitness~newHR2))

dev.new()

par(mfrow=c(3,5))

varPlot(age, fitness, "Age vs. Fitness", "Age", "Fitness", 1, lm(fitness~age))

varPlot(height, fitness, "Height vs. Fitness", "Height", "Fitness", 2, lm(fitness~height))

varPlot(weight, fitness, "Weight vs. Fitness", "Weight", "Fitness", 3, lm(fitness~weight))

varPlot(forwFlex, fitness, "ForwardFlex vs. Fitness", "ForwardFlex", "Fitness", 4, lm(fitness~forwFlex))

varPlot(sbp, fitness, "SBP vs. Fitness", "SBP", "Fitness", 5, lm(fitness~sbp))

varPlot(dbp, fitness, "DBP vs. Fitness", "DBP", "Fitness", 6, lm(fitness~dbp))

varPlot(rgm, fitness, "RGM vs. Fitness", "RGM", "Fitness", 7, lm(fitness~rgm))

varPlot(lgm, fitness, "LGM vs. Fitness", "LGM", "Fitness", 8, lm(fitness~lgm))

varPlot(vc, fitness, "VC vs. Fitness", "VC", "Fitness", 9, lm(fitness~vc))

varPlot(hr1, fitness, "HR1 vs. Fitness", "HR1", "Fitness", 10, lm(fitness~hr1))

varPlot(hr2, fitness, "HR2 vs. Fitness", "HR2", "Fitness", 11, lm(fitness~hr2))

varPlot(hr3, fitness, "HR3 vs. Fitness", "HR3", "Fitness", 12, lm(fitness~hr3))

varPlot(pl1, fitness, "PL1 vs. Fitness", "PL1", "Fitness", 13, lm(fitness~pl1))

varPlot(pl2, fitness, "PL2 vs. Fitness", "PL2", "Fitness", 14, lm(fitness~pl2))

varPlot(pl3, fitness, "PL3 vs. Fitness", "PL3", "Fitness", 15, lm(fitness~pl3))

dev.new()

par(mfrow=c(2,3))

fw1 = forwFlex + weight

fw2 = forwFlex - weight

fw3 = forwFlex^2 + weight

fw4 = forwFlex + weight^2

fw5 = sbp + dbp

lfit = log(fitness)

varPlot(fw1, fitness, "ForwardFlex + Weight vs. Fitness", "FF + Weight", "Fitness", 1, lm(fitness~fw1))

varPlot(fw1, lfit, "ForwardFlex + Weight vs. Log(Fitness)", "FF + Weight", "Log(Fitness", 2, lm(lfit~fw1))

varPlot(fw2, fitness, "ForwardFlex - Weight vs. Fitness", "FF - Weight", "Fitness", 3, lm(fitness~fw2))

varPlot(fw3, fitness, "ForwardFlex^2 + Weight vs. Fitness", "FF^2 + Weight", "Fitness", 4, lm(fitness~fw3))

varPlot(fw4, fitness, "ForwardFlex + Weight^2 vs. Fitness", "FF + Weight^2", "Fitness", 5, lm(fitness~fw4))

varPlot(fw5, fitness, "SBP + DBP vs. Log(Fitness)", "SBP + DBP", "Fitnes)", 6, lm(fitness~fw5))

preds <- fitness\_score(fitnessData)

sqrt(mean((fitnessData$FitnessScore - preds)^2, na.rm = TRUE)) # RMSE

sum(is.na(preds))