# Week 2 Assignment

By: Zach Adair

MSDS 680 – Machine Learning

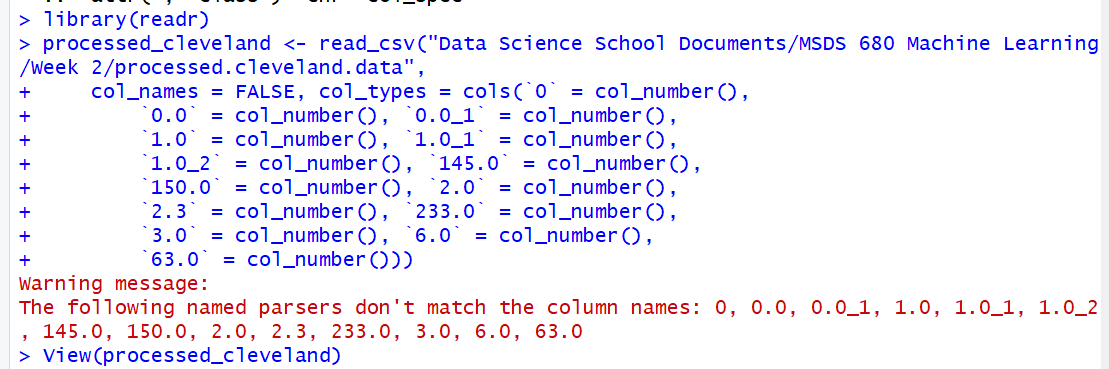
Regis University

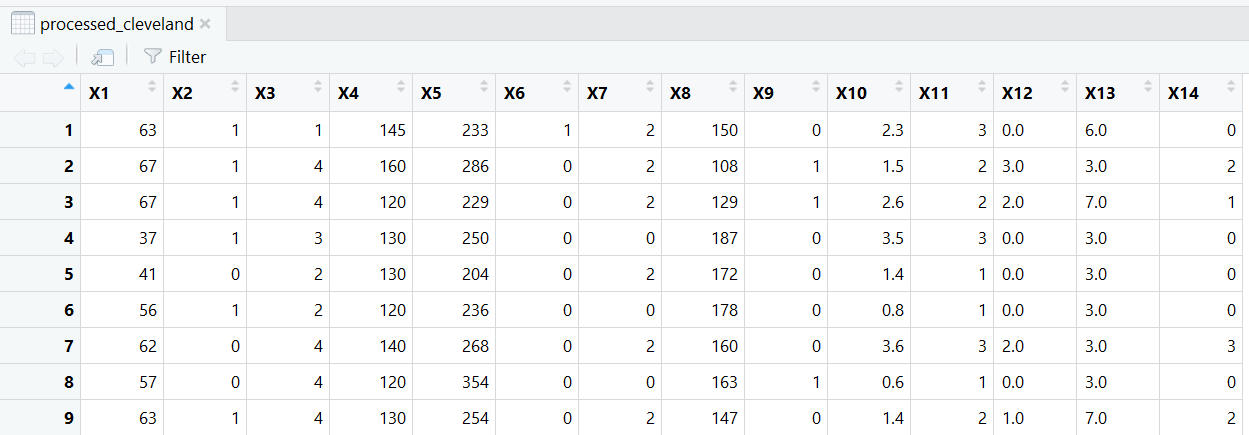
## Introduction

For this assignment, we will be working Cleveland heart data and attempting to predict heart disease in their patients. I will start by doing some data exploration and seeing what the data is like and try to find anything that may look like a trend within the data. Next I will divide the data into a training set and a test set randomly. The training:test ratio will be at 70:30 and prediction will be based on 1-nearest neighbor. From there I will calculate the error rate and report the confusion matrix and accuracy of my results.

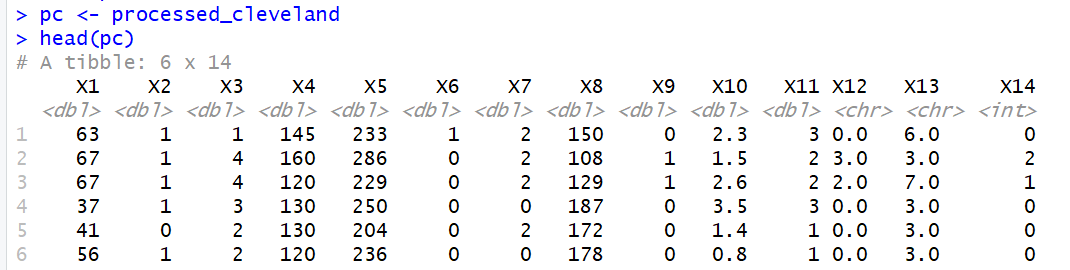
## Bringing data into RStudio and Exploring It

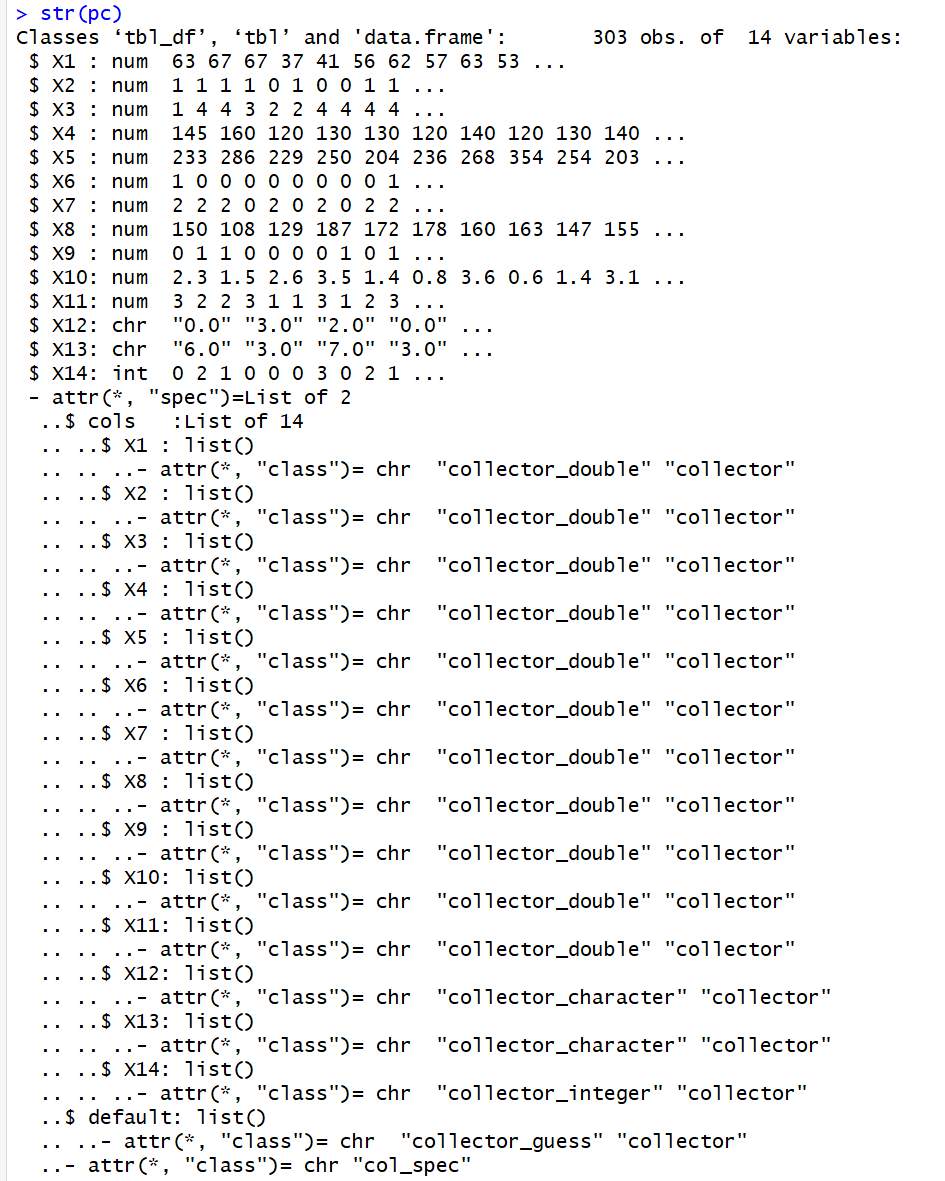
We will start by importing the data file into RStudio.

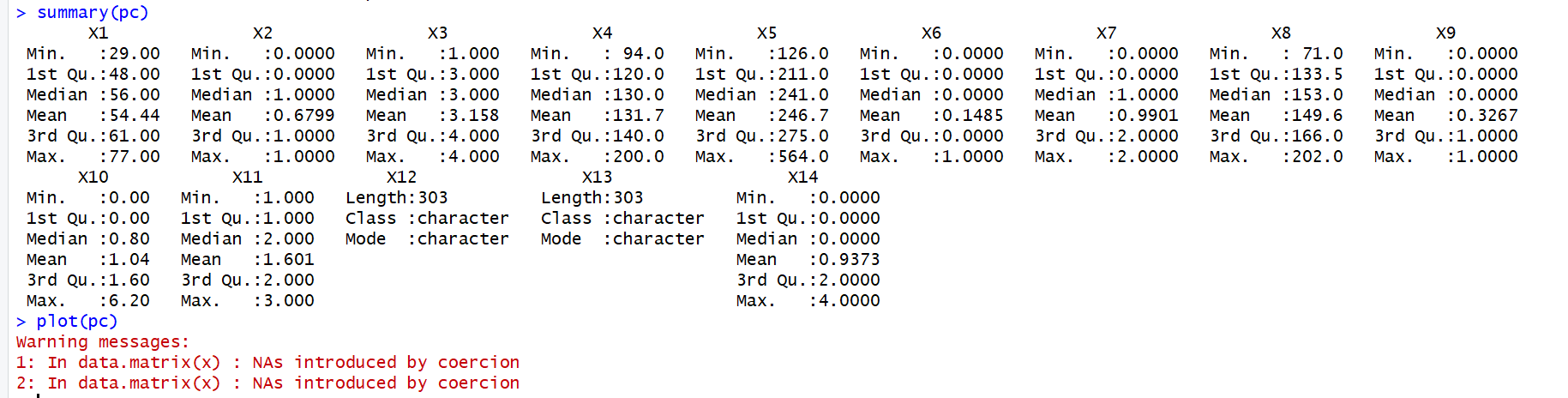


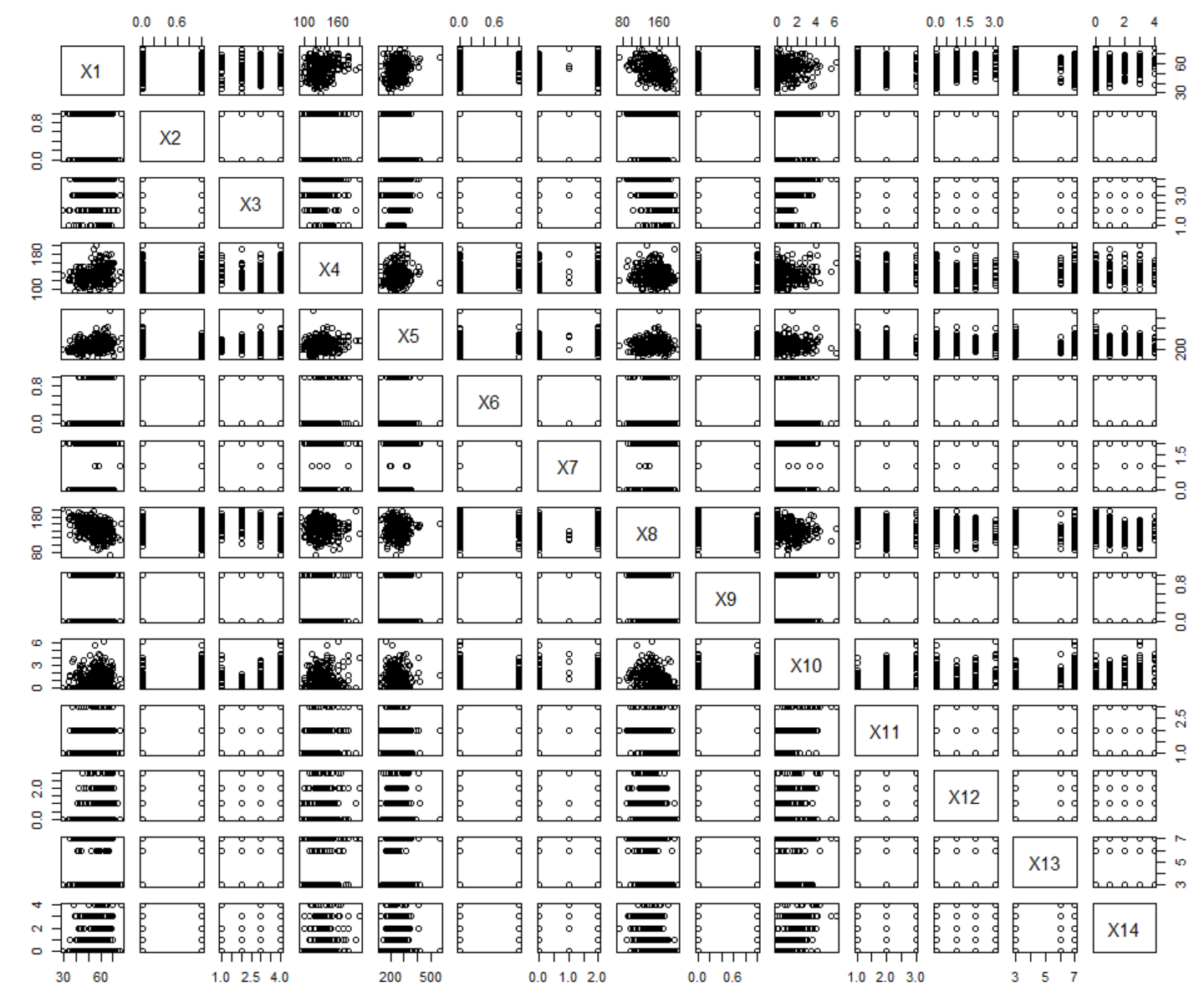


Now that I’ve pulled the data into RStudio, let’s take a look and explore the data.



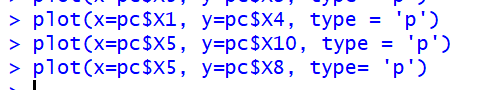


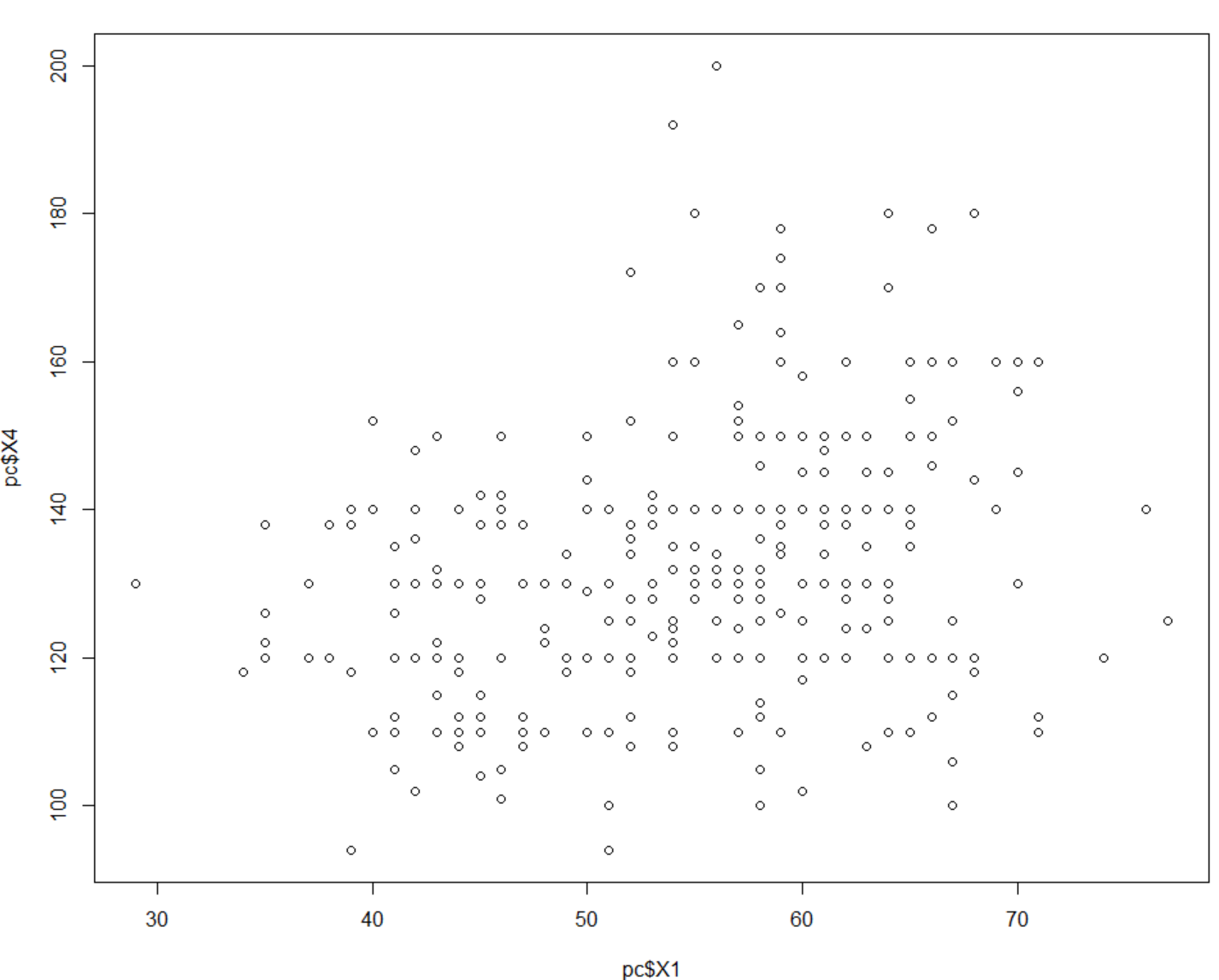


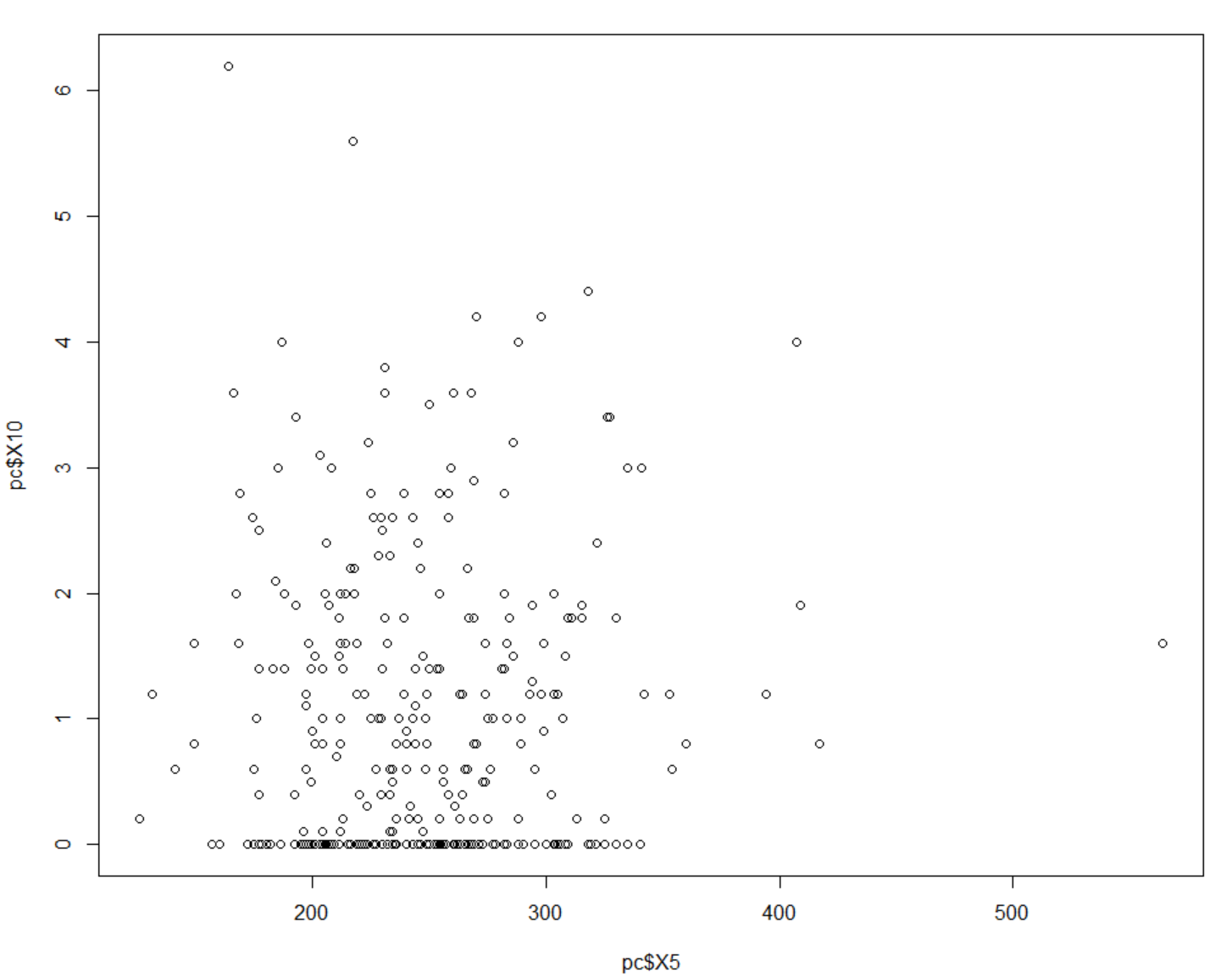


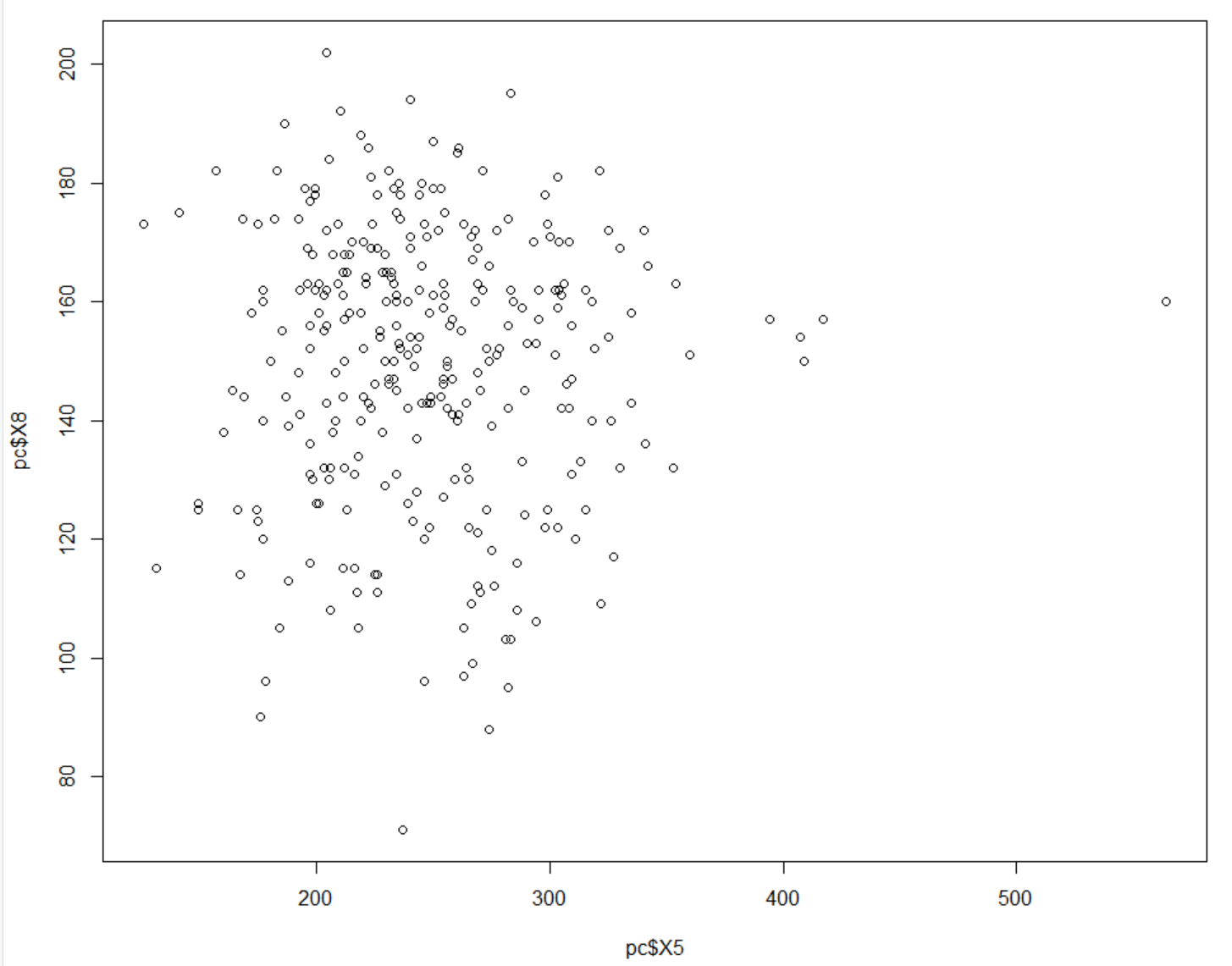
Now, I’ll take a look at some of the more interesting plots, some that look like that both variables are either 0 or 1 and others with a slew of different values for other variables. Next I’m going to get a closer look at the two variables against one another.

The variables I used for three different scatter plots were X1 vs. X4, X5 vs. X10 and X5 vs. X8.









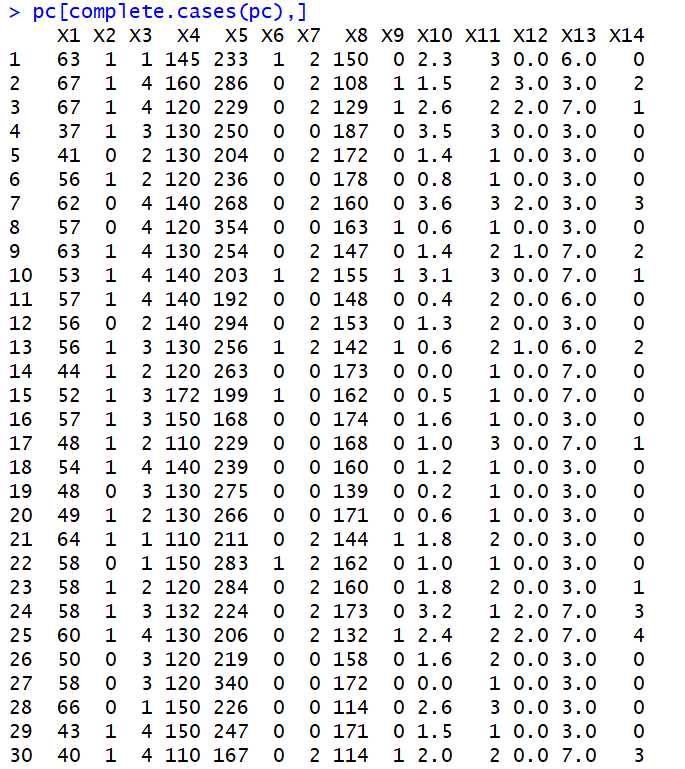
None of these graphs give us any further insight on these variables and how they really interact with one another.

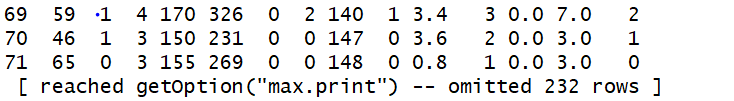
## Check Data for Missing Values

At this stage we will run some commands that will check our data for any missing values, if there are missing values we will go about the business of getting rid of them and then checking our mean and median without those values.

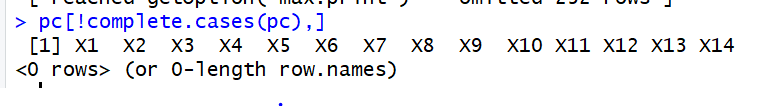
The next thing I’m going to do is look through the data for any missing values and see how many rows there are without data.

We will start by looking at what is there.





If we do some simple addition we can see that the output the command gave us was 71 rows, and it gave us a notification stating that it had omitted 232 rows. 71 + 232 equals 303 which is also the same number of records we have in the dataset, so we can deduce that all the rows have data within it. But just to make sure we will run the !complete.cases command to validate this.



With no rows in our output we can deduce that this data is free of any missing data points. This means we will not have to manipulate our data to scrub our set of missing data.

Now I will Normalize and Standardize my data

## Normalization and Standardization

With there being varying scales at which the data is being measured being a problem for measuring the distance and potentially affecting the distance computation and classification it is crucial that each variable should have the same to the distance.

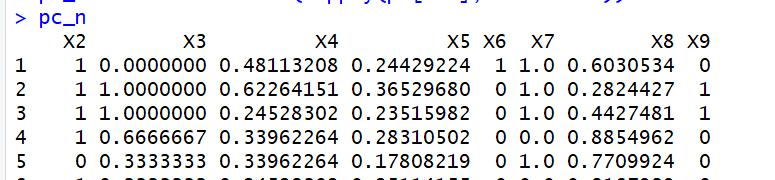
So with that being said, I will start by normalizing the data. To do this we will change the scale of all our variables so that the scale of each is all the same. We will do that by creating a normalize function where we will take the value subtracted by the minimum value, and divide it by the max value by the minimum value. It will look something like this below:



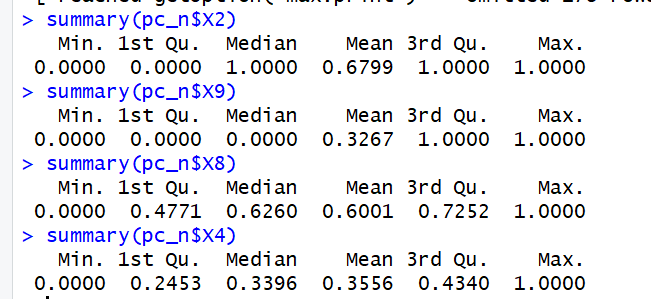
From here we apply it to our data like so:



Which will give us outputs like so:



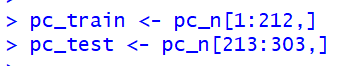
And we can now see different variables and how they are now different from what they once were.



As you can see from the summary of four of our variables, all the variables have a range of between 0 and 1. This was a crucial step and allows us to more accurately measure these values.

## KNN

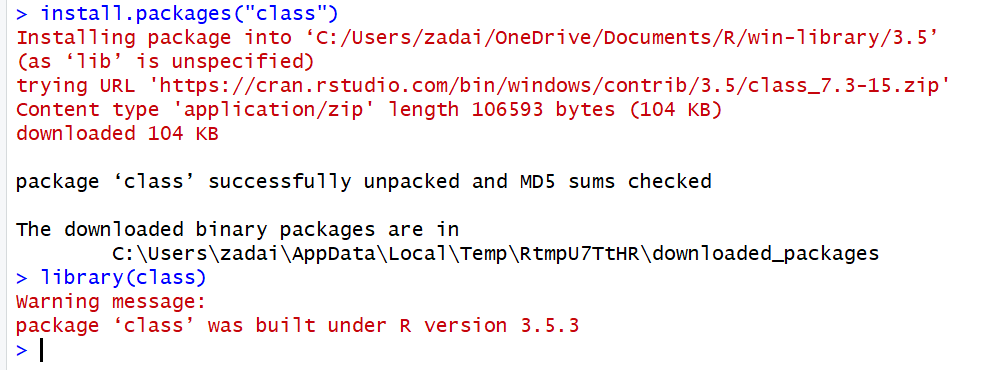
To start, we need to create a training and test data set. The ratio that we are going for is 70 Training 30 Testing for our model. We will create this ratio with our new normalized data set from the last step. 70% of 303 values is 212 and our test data will be 91 observations.



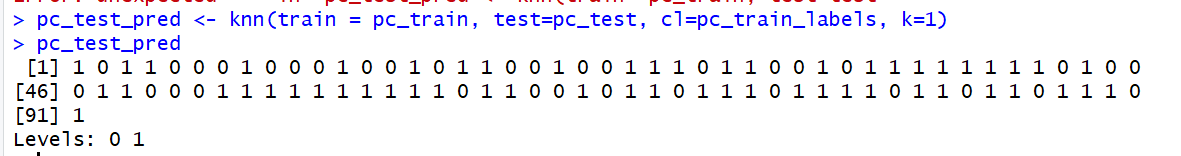
Let’s also set some labels for our new test and training data sets.



From here I can train the model. I will need to download the class package into RStudio to accomplish this goal.



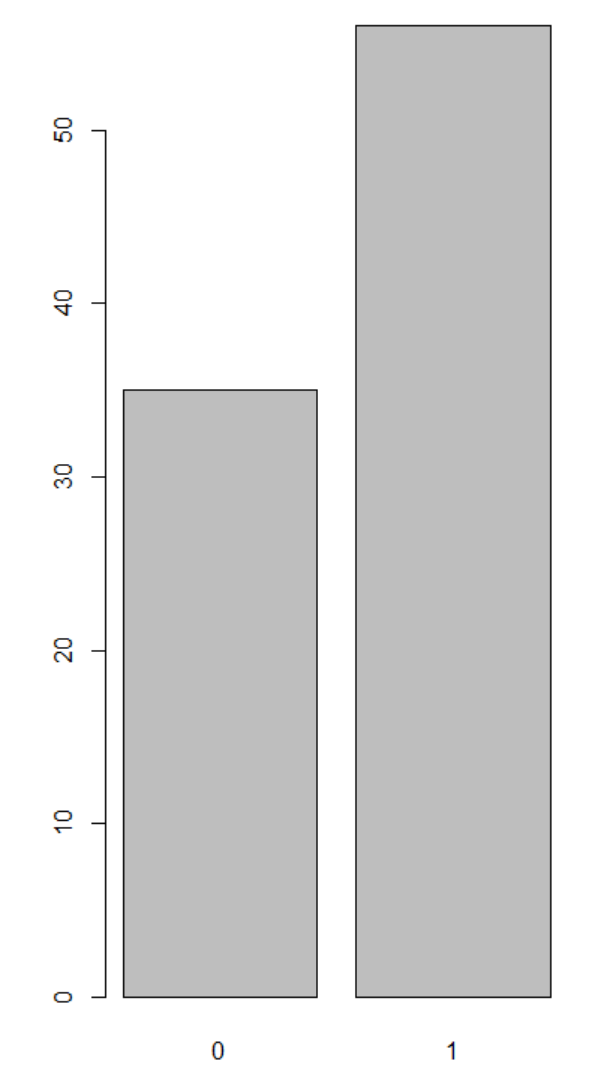
Now we will create the predictor model from our train and test datasets.



I also decided to display the values of pc\_test\_pred because I wanted to see the results. As you can see, the data comes back as only 0’s and 1’s.

I’m also going to plot the distribution between the 0’s and 1’s to see the frequency of them within the data.

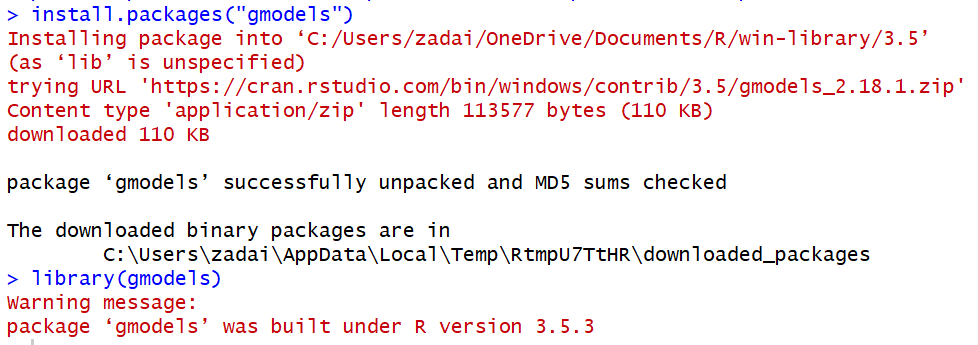




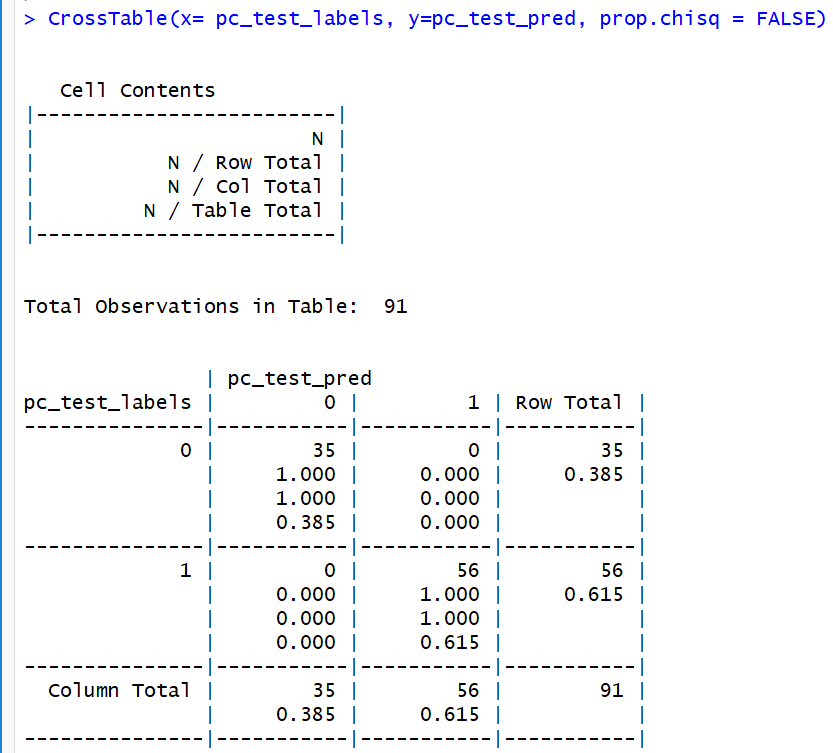
As you can see, there is quite a large disparity between the 0 values and the 1’s.

Now that the model is created I need to now test and evaluate the model.

I will start by using the package gmodels to test the model.

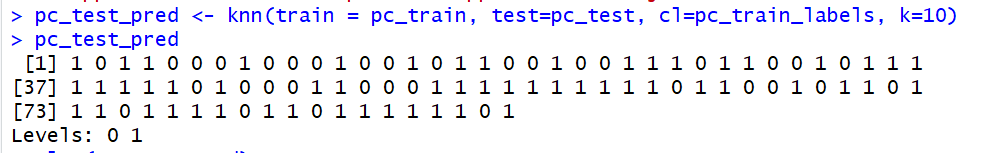


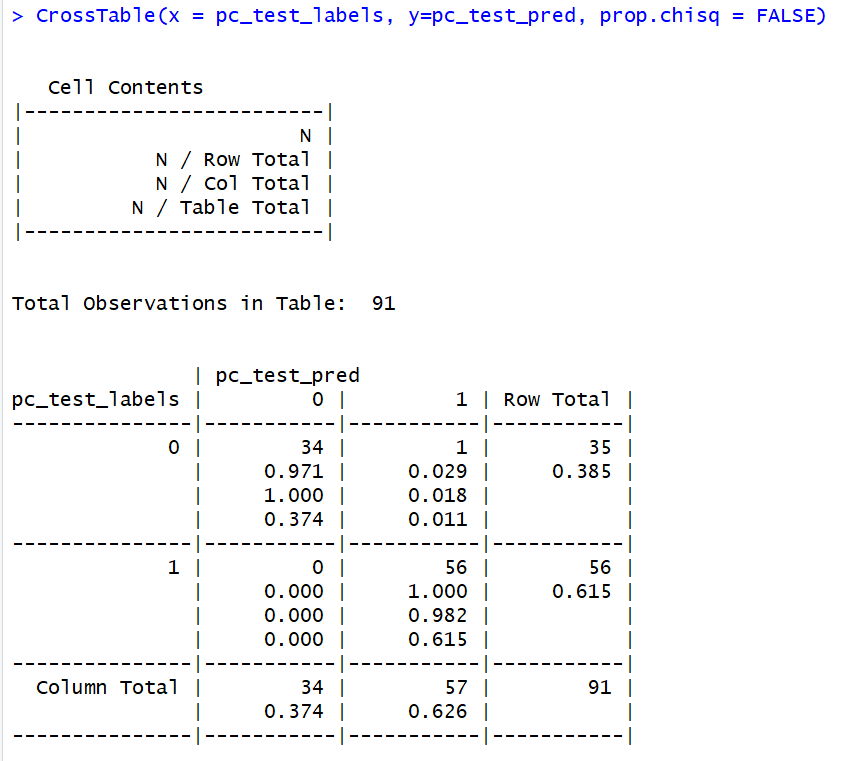
With that loaded we will start by using a cross table to evaluate how many of the test data set values were predicted correctly.



From the 91 values in the test data, all but one ended up being correctly predicted. 35 were predicted correctly as false when the value was false, and 56 were positive when the result was positive. Only one value was predicted incorrectly from the model which leads to 100% accuracy.

So because I’m a bit uncomfortable with having a model with 100% accuracy I decided to change up pc\_test\_pred and see if that would make the accuracy of my model drop some. I just changed the k value from 1 to 10. Here are the results below.





The only difference in the model that I’m seeing after changing that k value is that the Cross Table is now only 99%, which isn’t a huge drop but the model did wrongly predict 1 when the value should have been 0. I don’t think I could make very many changes to the structure of the model without changing the data itself or the number of entries in the test and train sets to make the model less accurate so I will leave it there and I will conclude my studies with my wrap up.

## Conclusion

For this week’s assignment I got to start and finish a KNN model and I found it intriguing the different steps it took to go through the process. Starting with the standard data exploration I couldn’t make ends meet about the data or any trends in it, from what I downloaded from the processed data file, the data was all numerical and it was hard to see if any two variables could possibly have any correlation. I enjoyed being able to check and see if there were any missing data values within the data, I believed there was at first but I was quickly proven wrong when I checked which made it easy for me to continue forward because that cut out the steps for if there were missing values within the data. I enjoyed also normalizing the data, I thought it was really unique the result of doing so and seeing all the values of the data set on the same sort of even ground when it came to the range of 0 to 1. Finally getting into the the KNN model and setting up the test and train data sets and then finally running the model and checking its accuracy was a real fun exercise, I didn’t expect my model to be so accurate so I did play around a little bit and bumped up my k value from 1 to 10. The result wasn’t much difference because instead of getting zero observations wrong the model did get one value wrong when k=10. I believe if I could have somehow changed the data in anyway or if I would not have normalized the data set the model would have been much less accurate. The assignment in its entirety was a good exercise, I believe I learned a good deal about how these types of models are built and maintained in RStudio because of it, and it only gets me more excited for what other types of machine learning models await us in the coming weeks.