# Week 7 – Assignment 1 – Supervised Learning

By: Zach Adair

Regis University

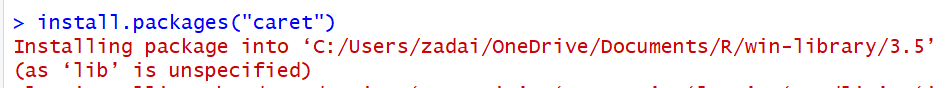
MSDS 650 – Data Analytics

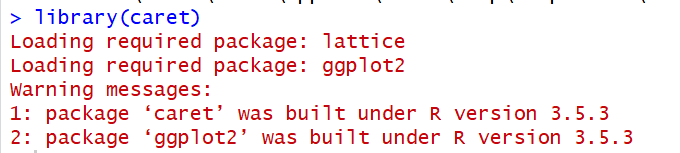
## Introduction

For the first assignment of Week 7 I will be going through the Supervised Learning assignment. Within my write up I will be going through the steps I did to go through the machine learning template that was attached to the assignment and then I will be going through the actual assignment as well as answering the questions throughout my write up. I’m very excited about this week because machine learning is a topic that I have probably the most interest in when it comes to the scope of Data Science so I look forward to what I can learn from this assignment.

## First Machine Learning Project in R

Start by installing the packages required, the package required is the caret package.

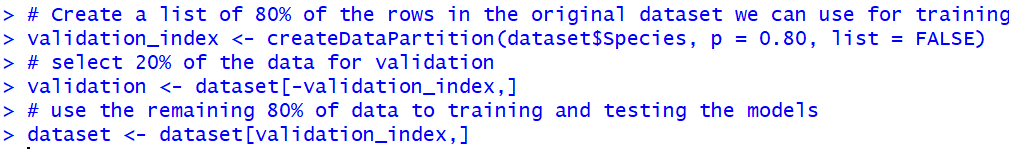




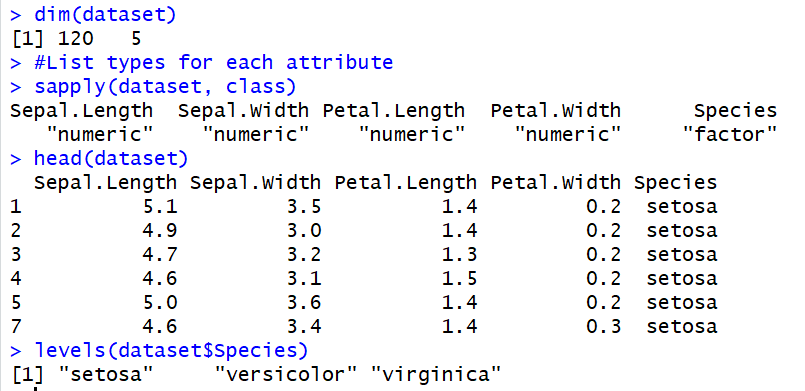
Next, load the data, the dataset the tutorial uses is the iris dataset.

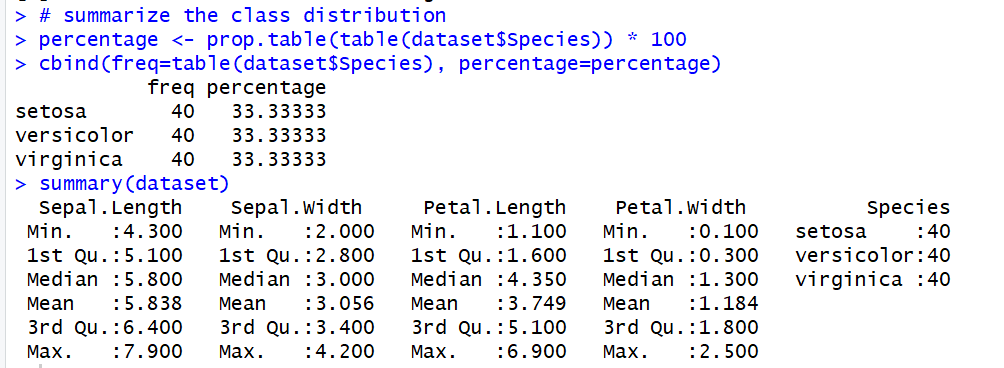


Now, create a validation set which will be used later on to test the accuracy of our models.



Summarize the dataset, find out some information about our data.



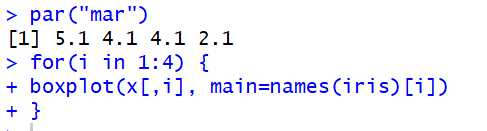


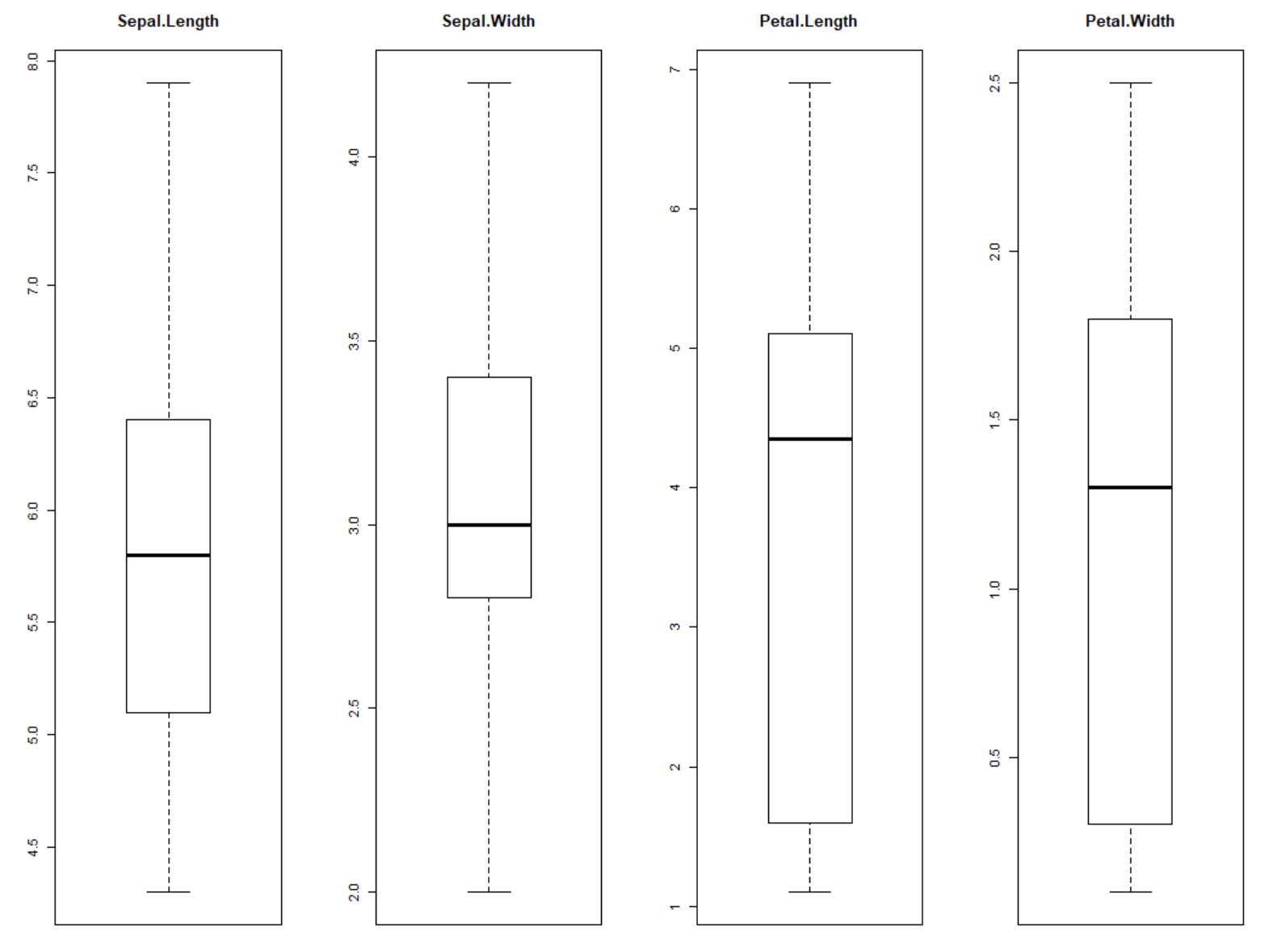
The commands above show a lot about the data, it tells us the dimension, the class of the fields, the levels, what the first several rows look like, the distribution of the Species and some summary statistics.

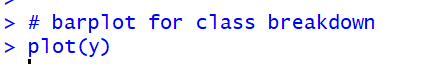
What we do next is visualize the data set with univariate and multivariate plots.

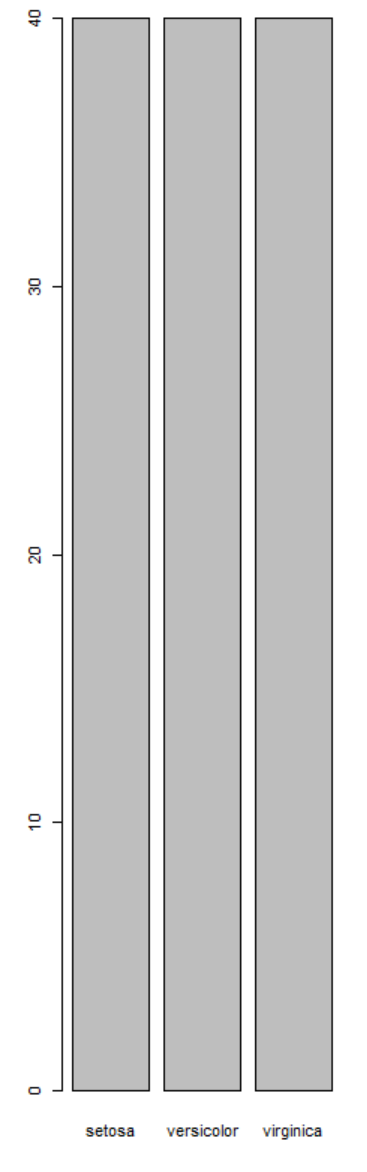


Was originally struggling with an error stating my margins were too large. But I did some research and was able to plot the boxplots.

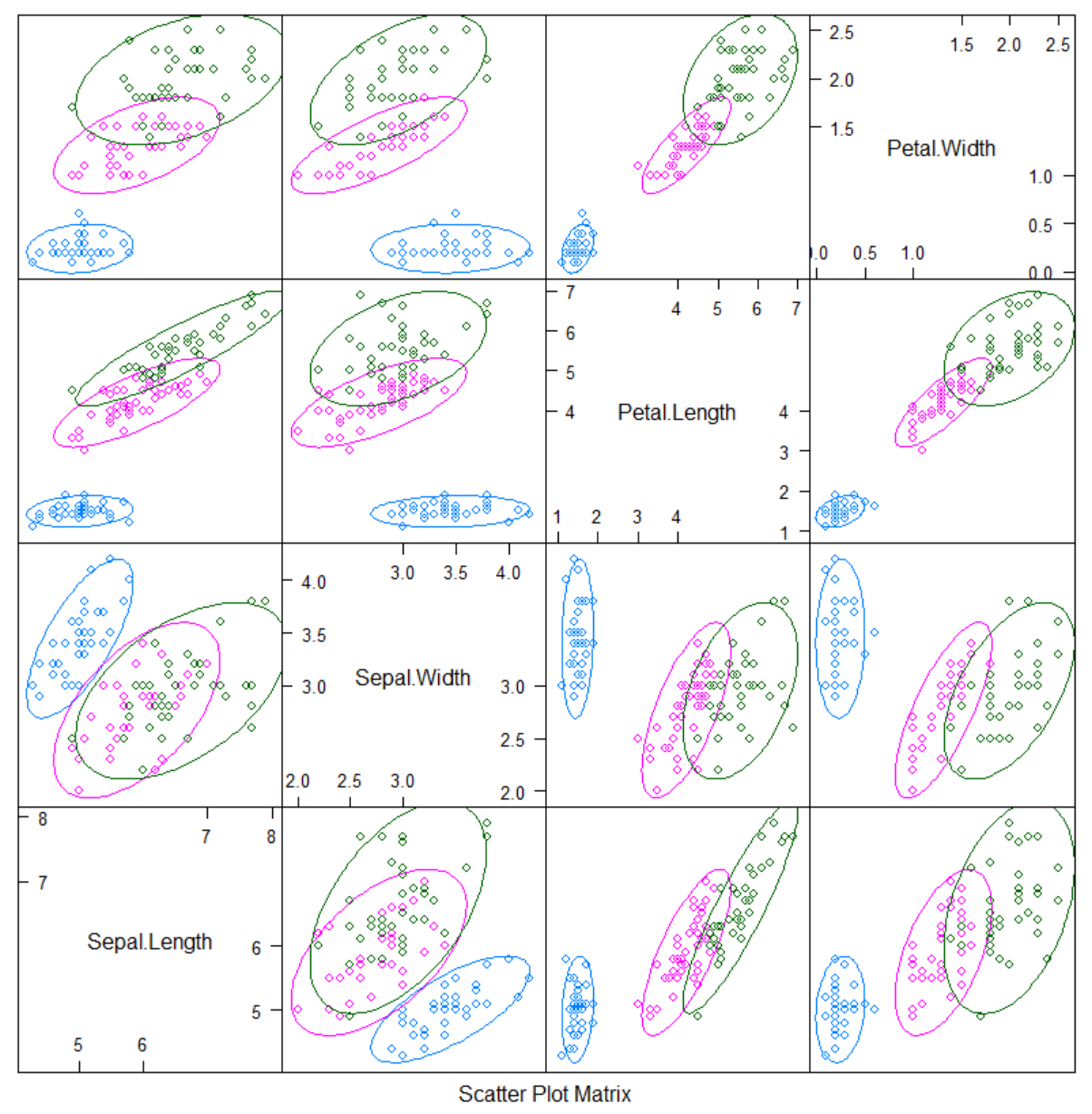




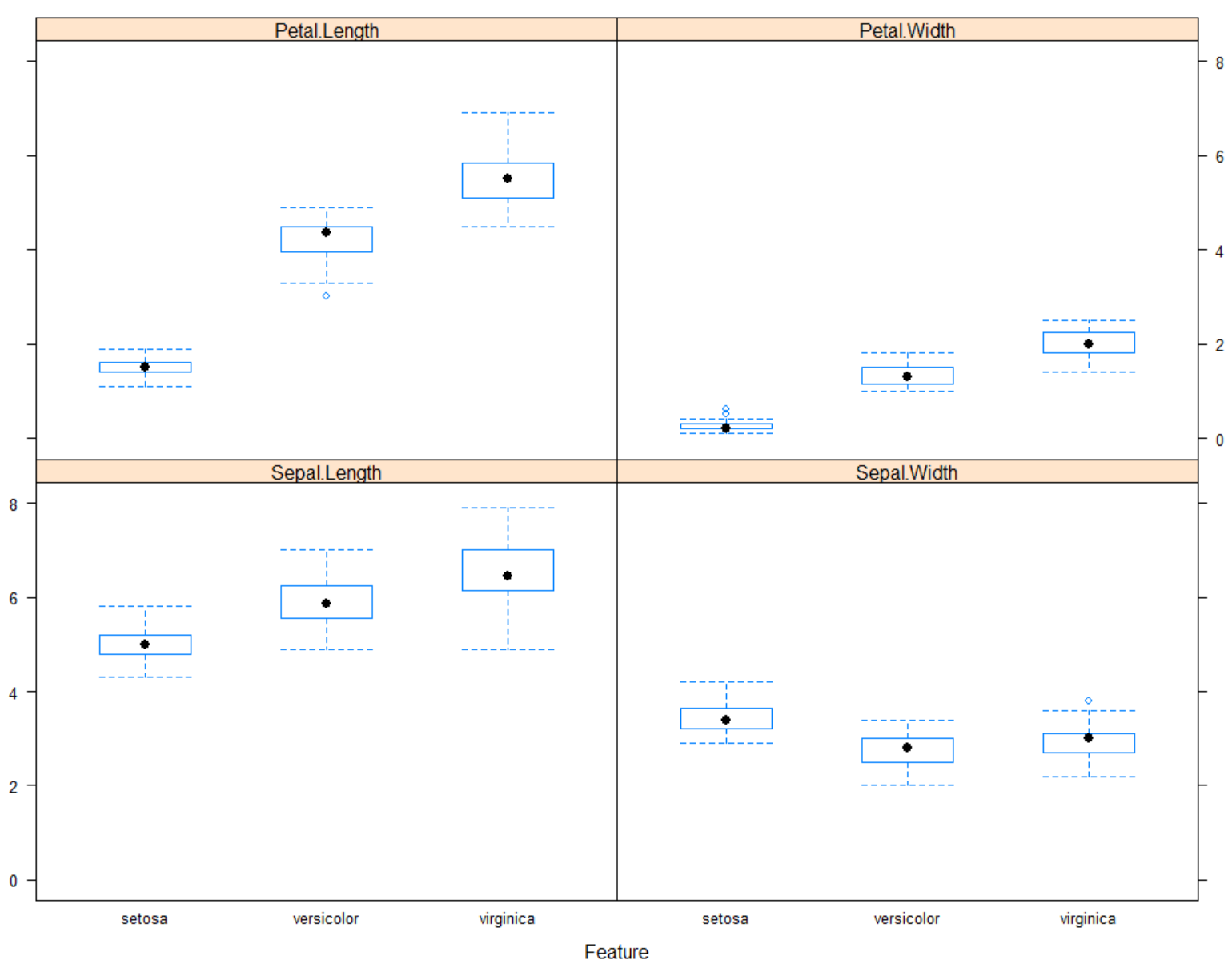


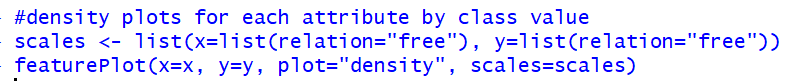


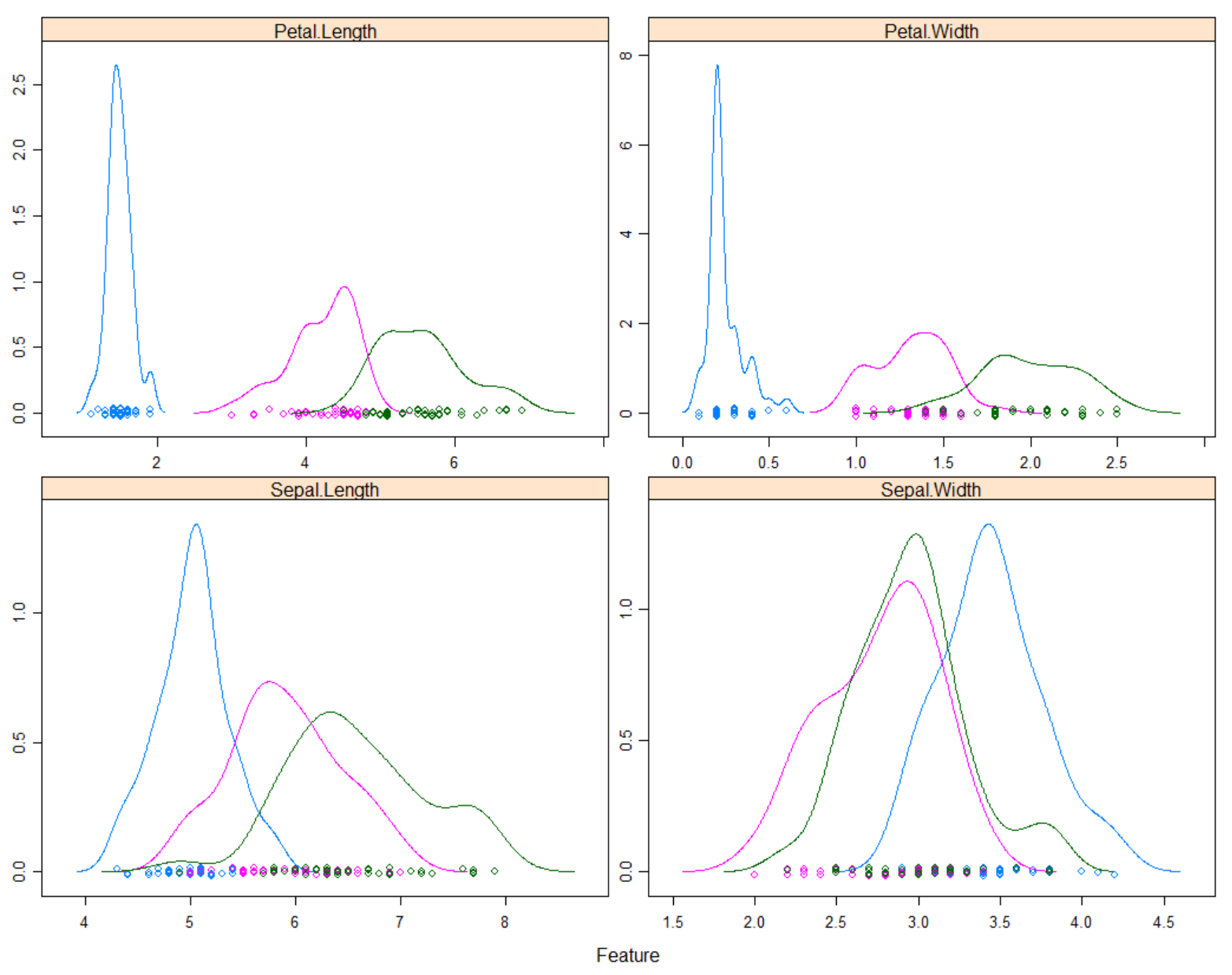






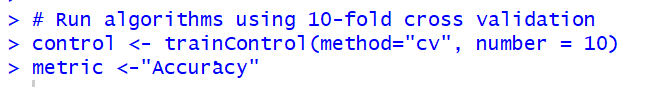






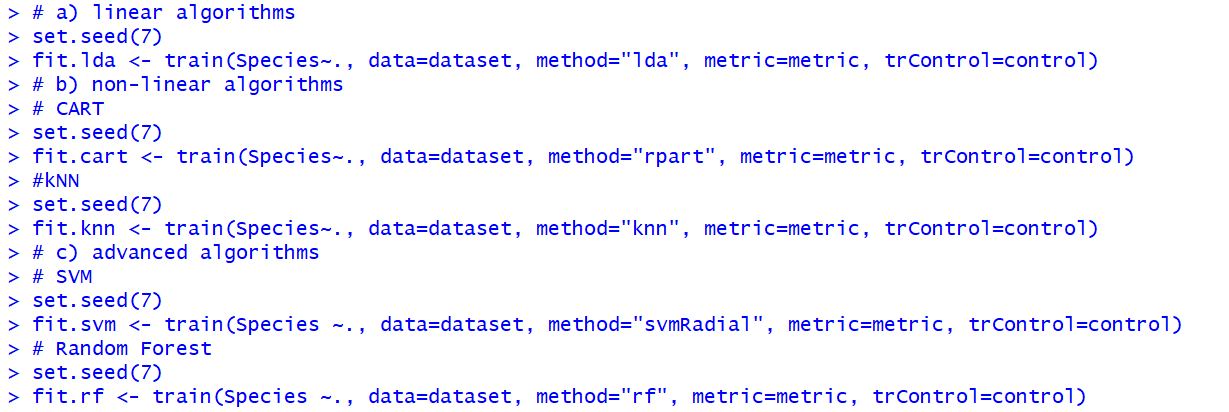
Those are some pretty cool charts! I’ve never seen these done before in R so I’m glad I have got to learn to do them, will come in very valuable in my future I’m sure.

The next step is to evaluate algorithms, and test the data against unseen data.

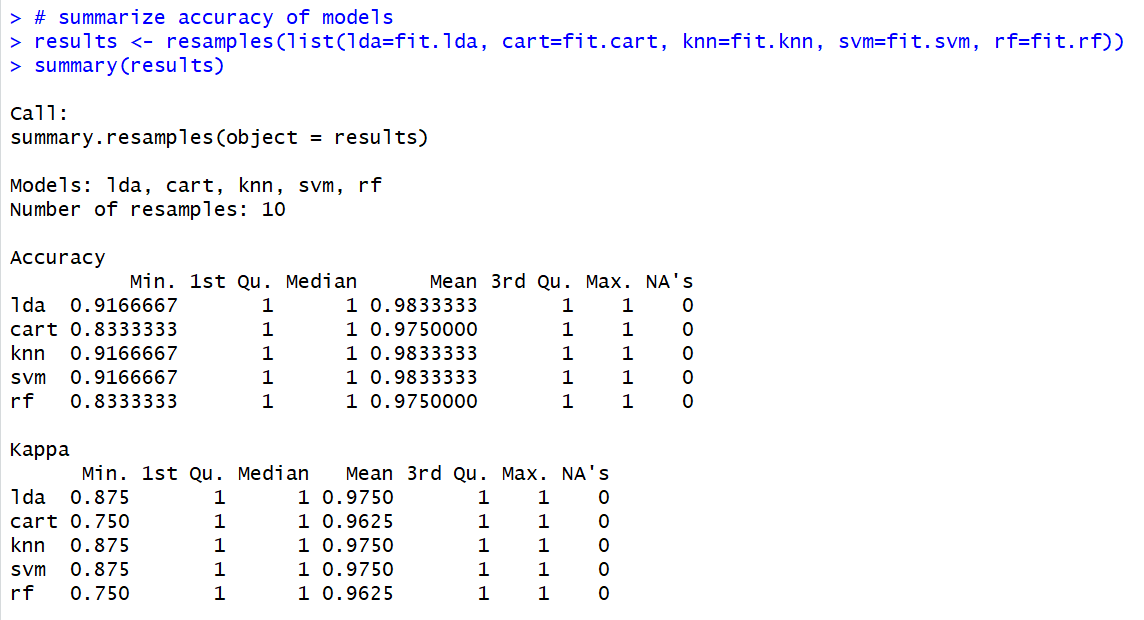


The metric of accuracy means that this is a ratio of the number of correctly predicted instances in divided by the total number of instances in the dataset by 100 to give a percentage of 95%.

Now we will build the 5 models we are going to test. Those models are: Linear Discriminant Analysis (LDA), Classification and Regression Trees (CART), k-Nearest Neighbors (kNN), Support Vector Machines (SVM) and Random Forest (RF).

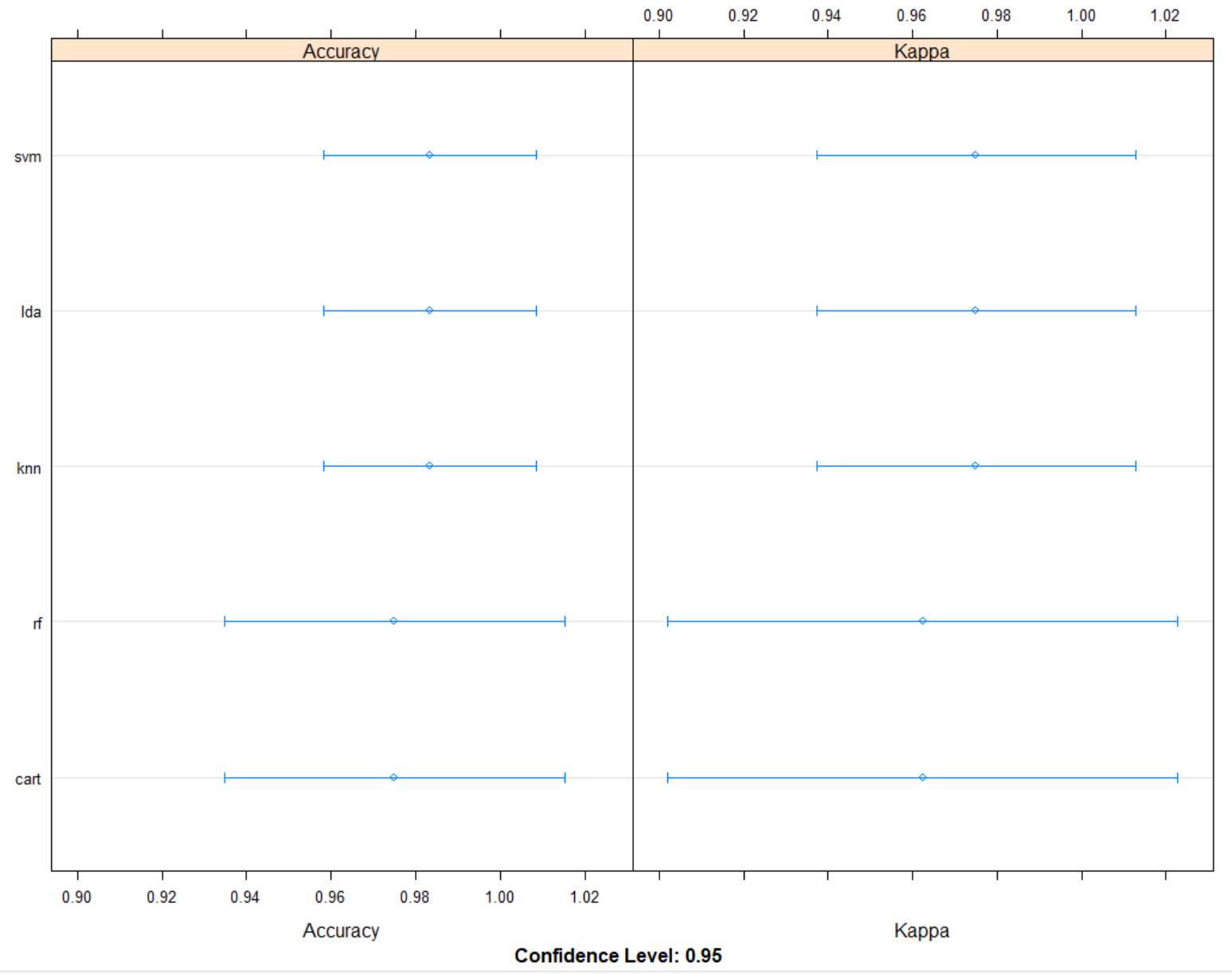


Now it is time to use the summary function on our newly created models.

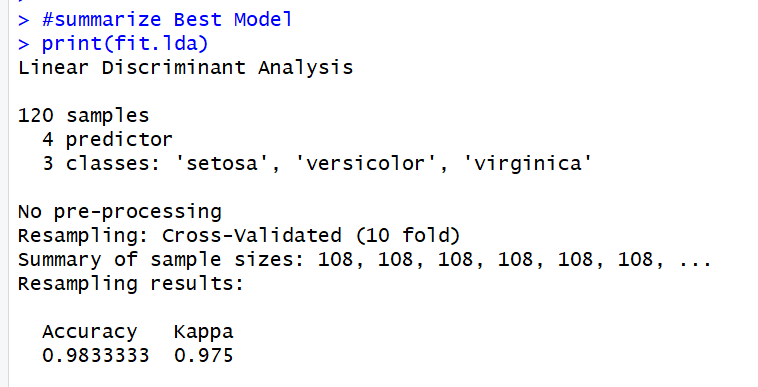


We can also create a plot of the model evaluation results and compare the spread and the mean accuracy of each model.

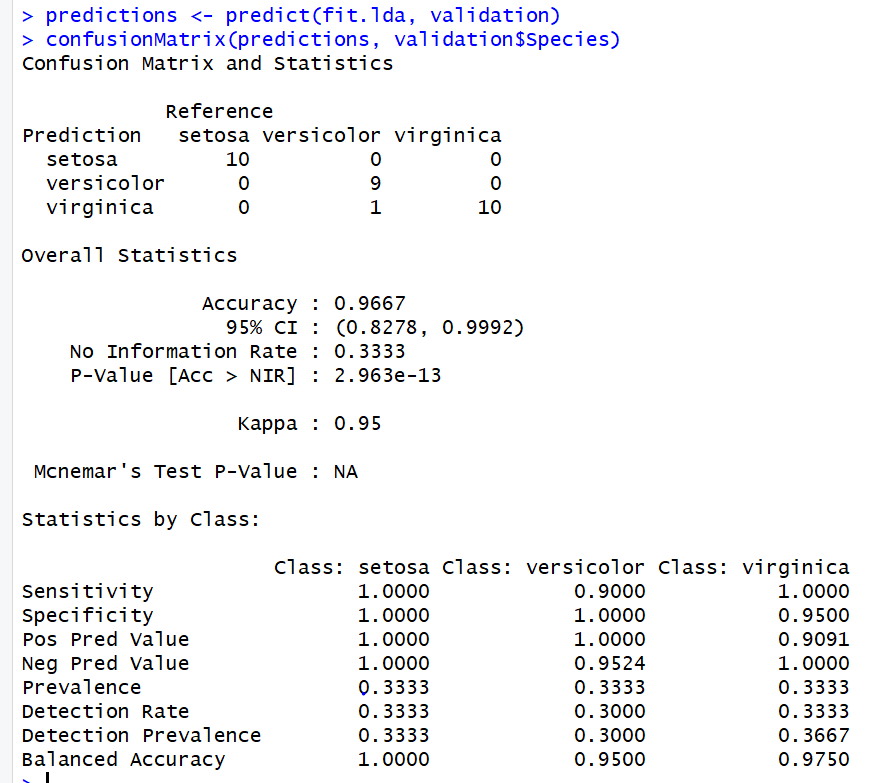




I can also summarize just one model, like below the LDA model summarized:



Next, I’ll take my most accurate model, LDA, and try to get an idea of the accuracy of the model on our validation set. It is valuable to keep a validation set just in case you overfitted to the training set or have a data leak. In those cases the results will be overly optimistic.



Though it is small at only 20%, we can assume our model is pretty accurate with an expected margin of 98%.

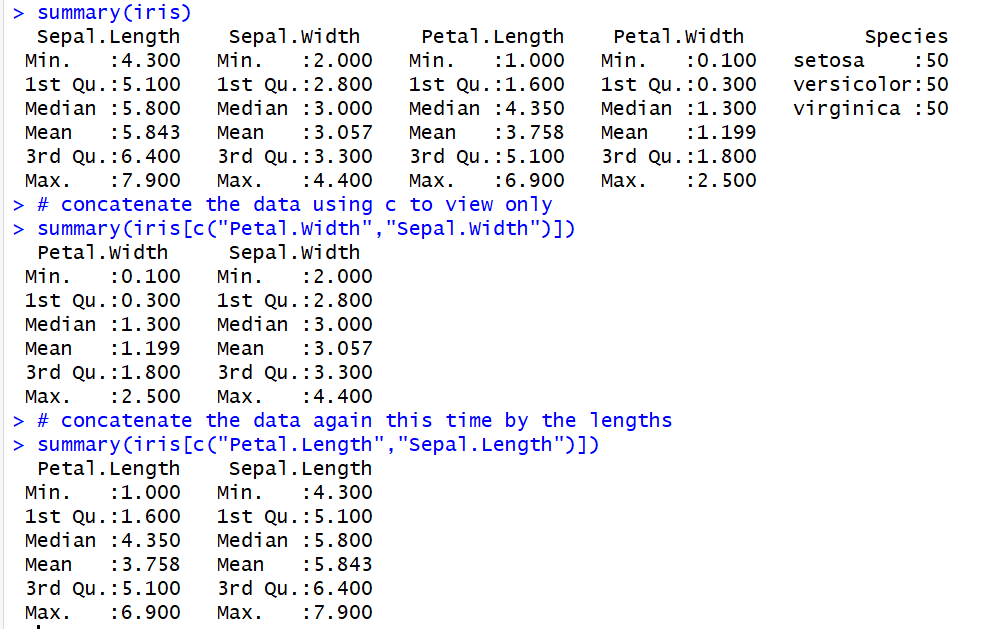
## Assignment – Supervised Learning

After using the online tutorial to get some sense of how to use supervised learning in R I will now apply that to the steps in the assignment. After going through these steps I will answer the questions assigned to this lesson.

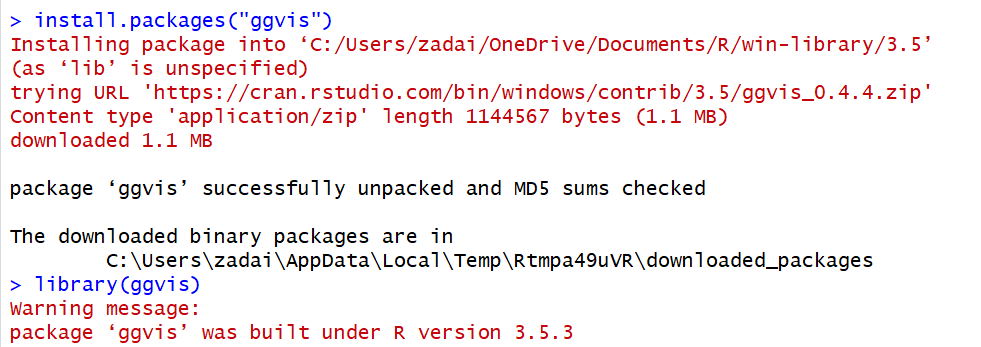
Like the first part of this assignment I will use the Iris dataset which is preloaded into RStudio.



Now I’m going to explore the data and take a look at what I can find out about it.

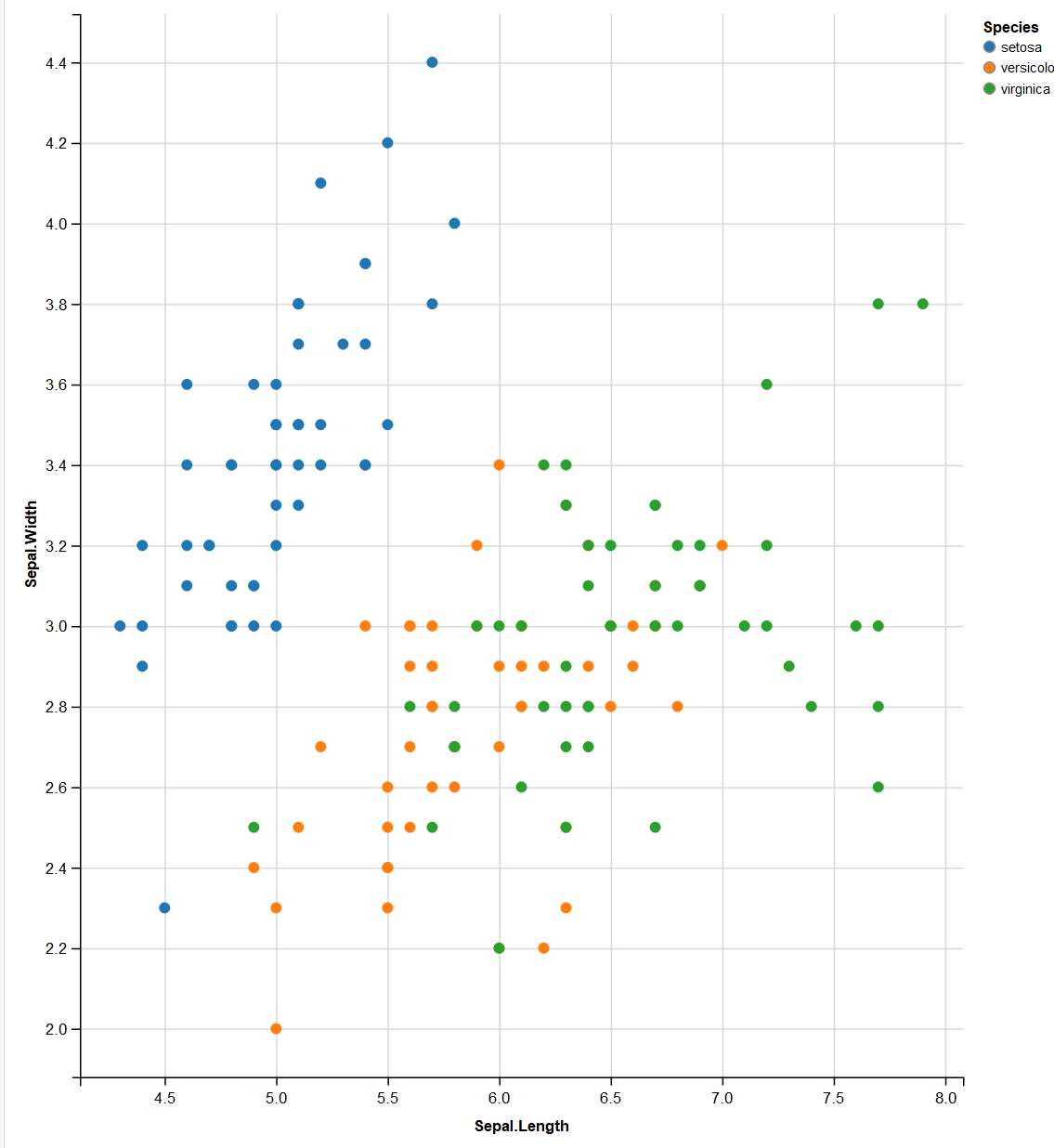


Now we will load the ggvis package to bring some data visualizations into our analysis.



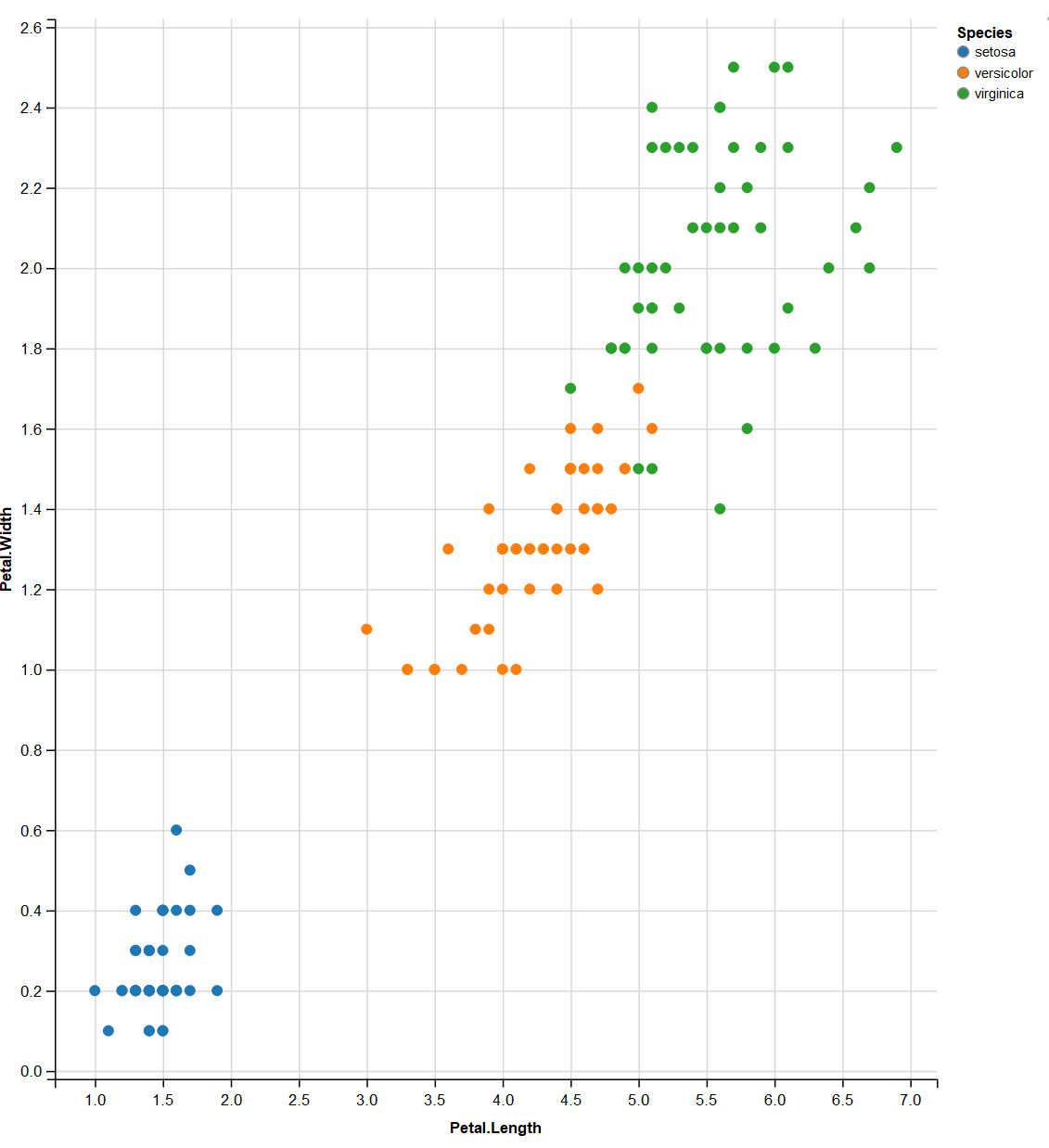
The first plot I will output is a scatterplot.



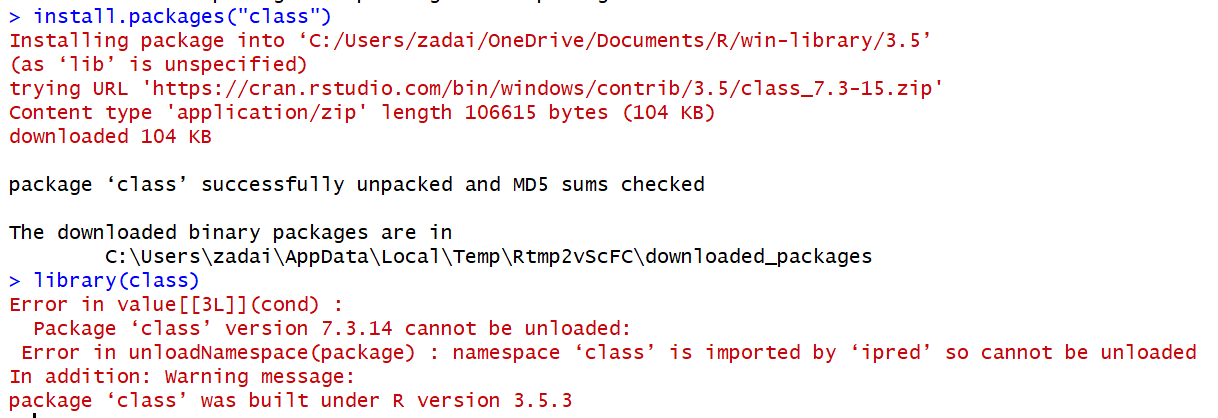


Now I’ll do the same thing but for petal length and width.

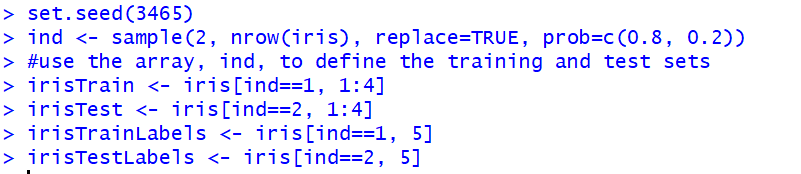




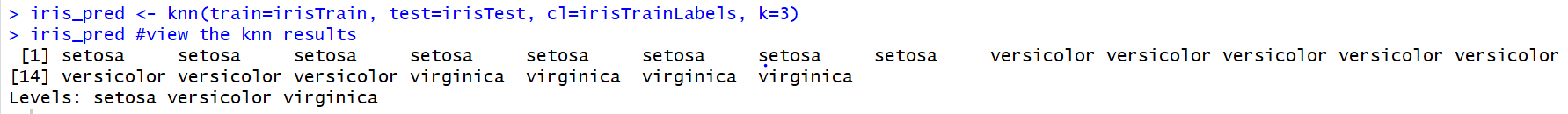
Now I will download the “class” package so I can use the KNN algorithm.



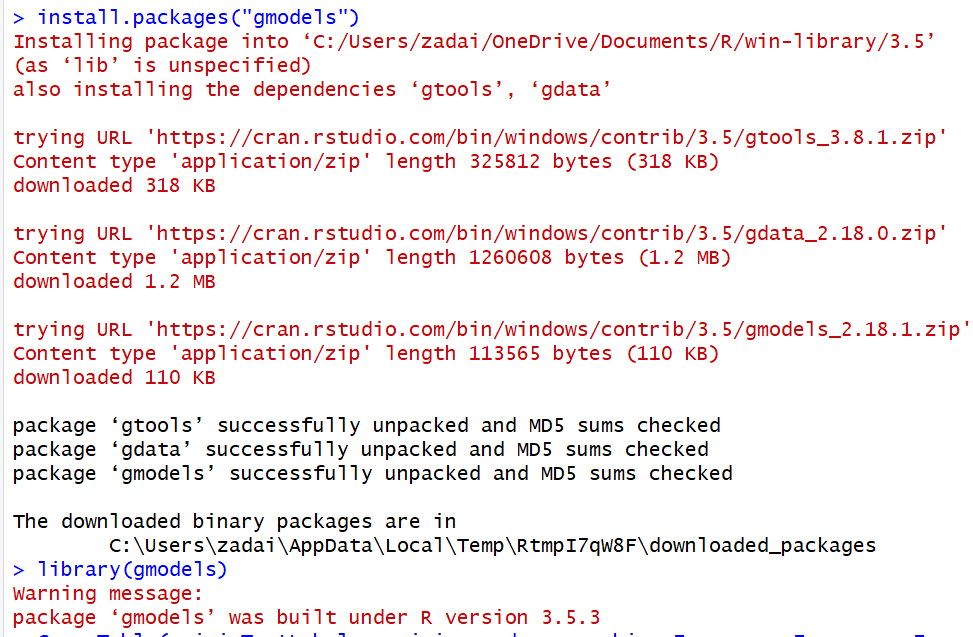
Now I will make the training and test sets of data, We will start by seeding the data and then a split of our data.



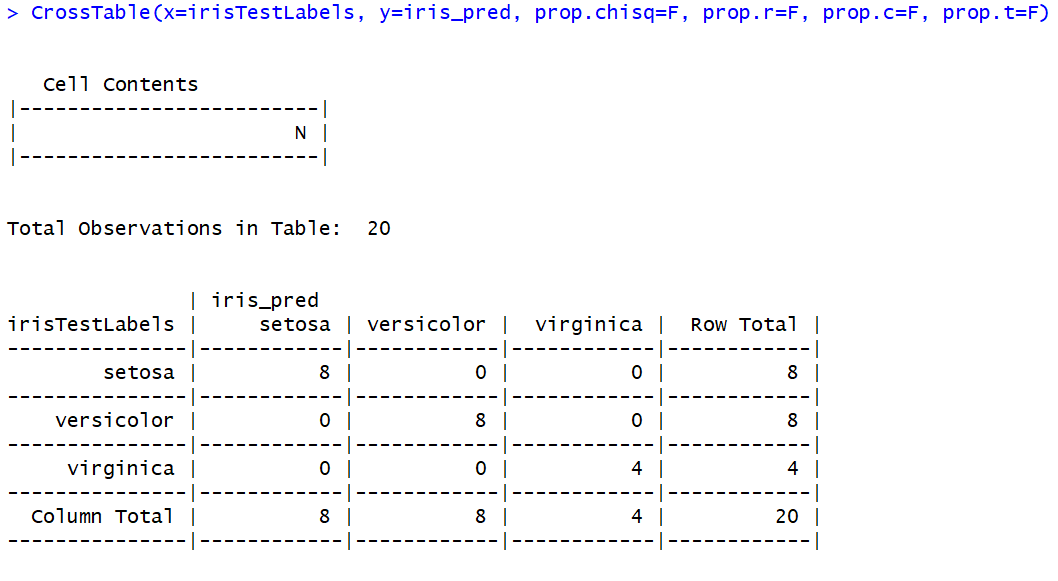
Next we will find the k Nearest Neighbors of the training set.



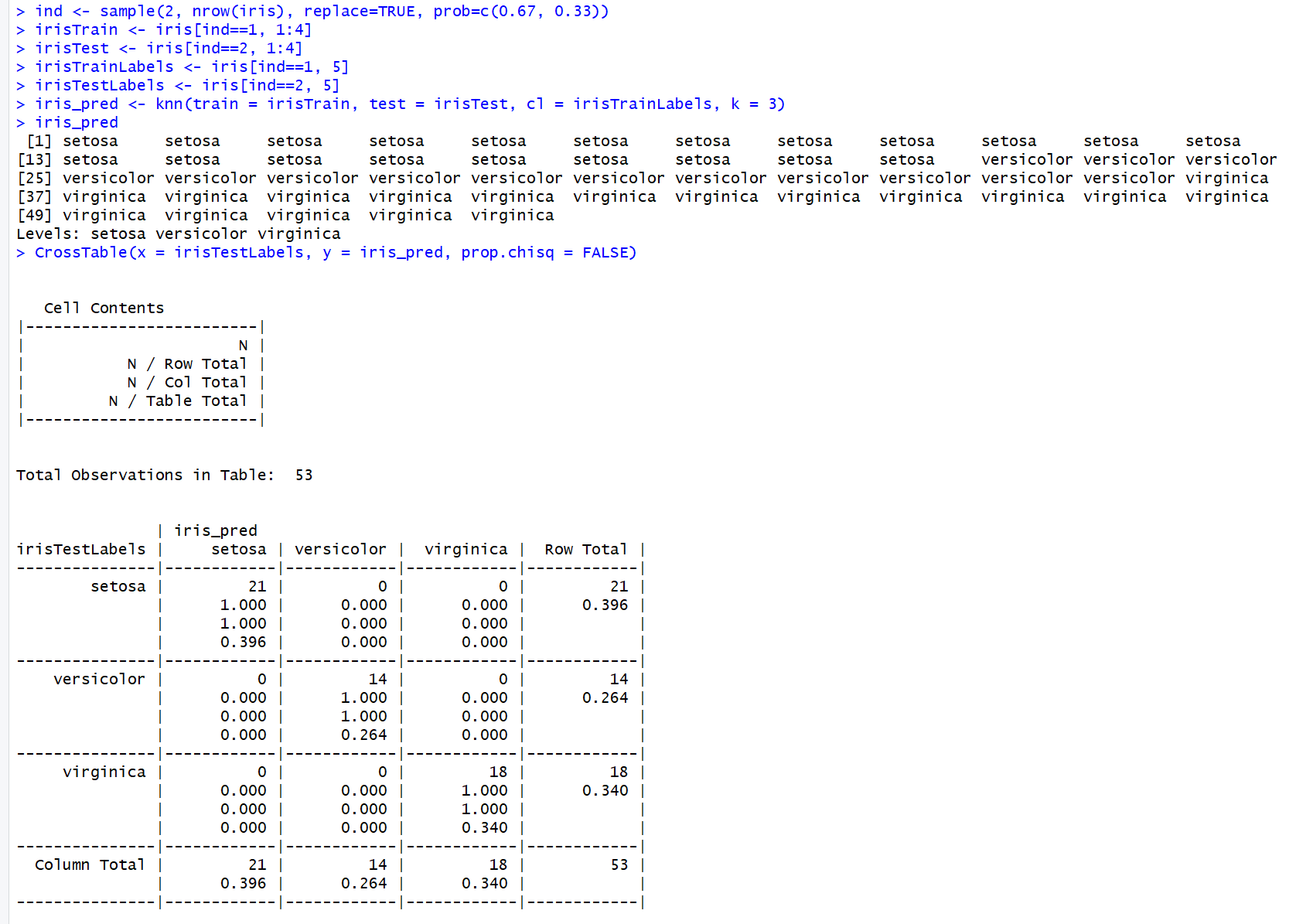
With our results in hand now its time to evaluate them, to do that I need to install the gmodels package.



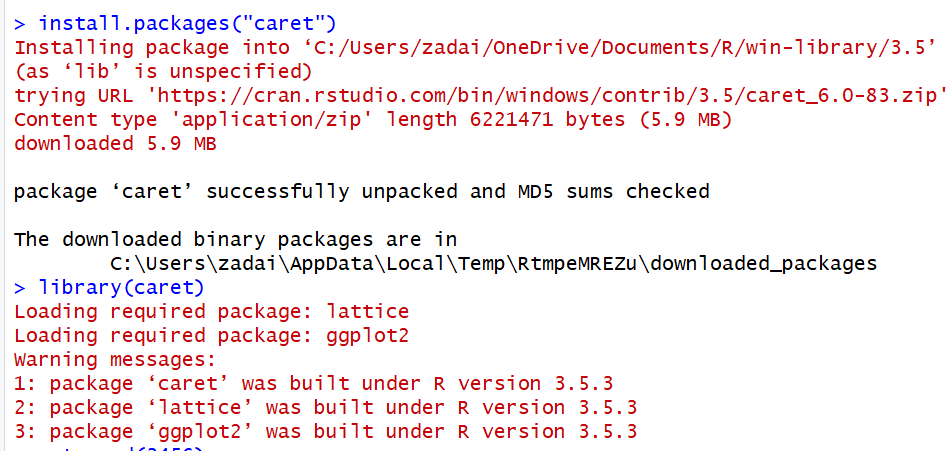
With the gmodel installed we are going to start by creating a contingency table of the Iris Test Labels and the Iris Predictor while setting chi squared and a few other properties to false since we don’t need to see them and we don’t care about their output.



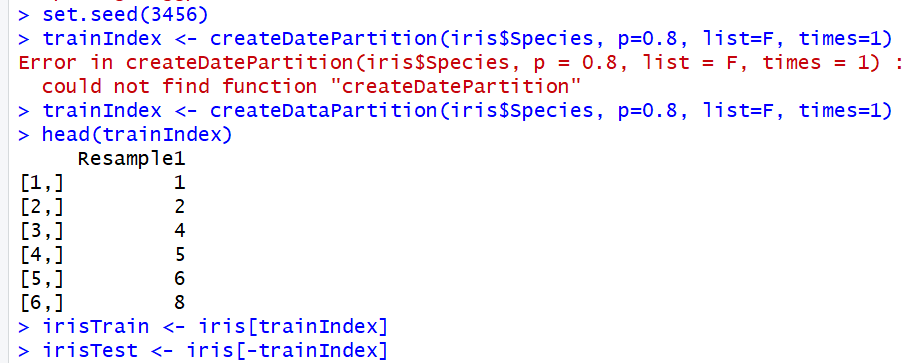
With the 80/20 split I created that should be my output between correct/incorrect predictions. I’m now going to construct the same thing but this time with a 67/33 split, this will output the same result but this time the correct/incorrect predictions should be split as stated.



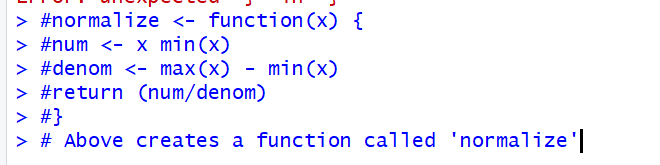
My results yielded that my model didn’t miss any of the labels that it should have hit. It got each and every record right.



Now I’ll create a data partition on the 80/20 data and the 67/33 data.



The final step we will do for this assignment is normalize the data. KNN is able to predict more efficiently when the dataset is normalized because it is a distanced based algorithm, so if one of he features has a huge value than the rest will be small, the feature with huge values will outweigh the smaller ones. Since the Iris data set is normal as is there isn’t anything that needs to be done, but to normalize a data set you would do the following:



Then from there you would create an x function and test it against your dataset, so for the Iris dataset you would create the x function for the Iris dataset by:



And then you would set a summary function on iris and the new iris\_x and check the results.

## Assignment Questions

1. **How do supervised learning algorithms solve regression and classification problems?**

Supervised learning algorithms solve regression and classification problems by trying to figure out the connection between the input and the output variables. The ultimate goal is to map the function so well that when you have new inputs that it can correctly predict the outputs. So when solving a regression problem the supervised learning algorithm will construct will best fit the data upon a hyper plane through the mapped out points, the better the model does getting to those mapped out points the better of a fit it is to the regression. For a classification problem the supervised learning algorithm will try and draw a conclusion based on the different output categories it has available to it. For example, it will try and map a group of 1st graders choices for favorite color based on different factors that student has provided.

1. **What packages in R and Python do supervised learning?**

RStudio: Mice, rpart, Party, Caret, RandomForest, nnet, kernlab

Python: StatsModels, Scikit-learn, XGBoost, Eli5, TensorFlow

1. **What measures of quality of the learning algorithm might you expect to see?**

The quality of the learning algorithm I might expect to see is the percentage of accuracy when the model is tested. For example when going through the exercise we were able to take a look at the actual result and the tested result and compare and see what the model predicted and if it was accurate or not and that accuracy can be measured and judged based on what the actual model did. Another is taking the log loss based on whatever the algorithm got incorrectly and take a logarithmic function, based on the probability of each guess. A confusion matrix which basically takes the true positives and true negatives and rewards those and penalizes false positives and false negatives. Another one is find the area under a curve, which can be used for binary classification problems, so something like logistic regression, and finding the True Positive Rate and the False Positive Rate which are two ends of the spectrum for a guess for a Supervised Learning Algorithm.

## Summary

This was my first foray into supervised learning within RStudio and I really enjoyed it. I found it really informative to walk through the exercises on the web page and in the hand out, and being able to not only read what each step was and the goal of it but also being able to do it myself I found really informative and I felt like I got a lot out of the whole assignment. I also liked that I was able to have points where I had to research and learn more about certain things with RStudio or the topic in general. The questions too I thought were good, they made me think a little deeper about the topic of supervised learning as well as do some research to help me fully put my thoughts together and answer the question properly.