Week 5 Assignment – SVM vs ANN

Zach Adair

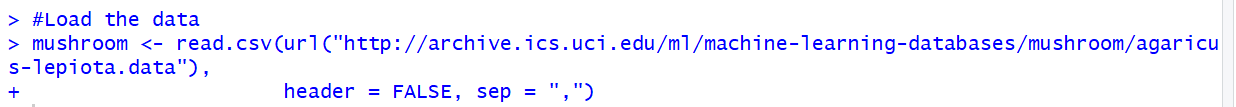
Regis University

Abstract

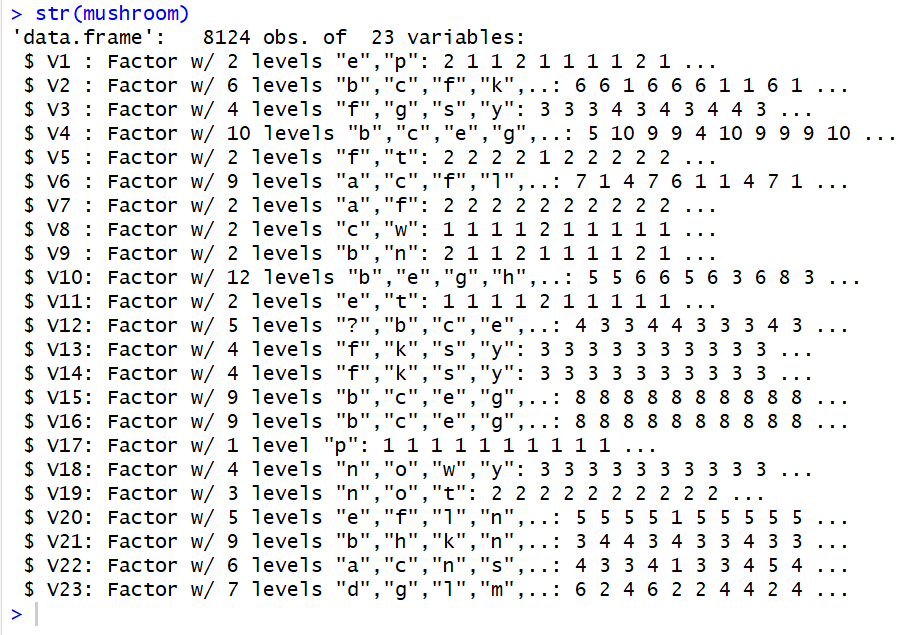
For this week’s assignment I will be taking a mushroom data set and running and first running a support vector machine on the data and summarizing that analysis, next I will run an artificial neural network on the same mushroom data set and summarizing my findings there. Then to finish up I will compare the results of both tests and state any concerns I have from the results.

Week 5 Assignment – SVM vs ANN

To start the analysis I first need some data, so I will pull the data from the website and begin the analysis of the mushroom data from there.



Lets also check out the structure of the dataset now.



As you can see each variable is of type factor and all but one has multiple levels. Now I need to put names on the columns and prepare the data for analysis.

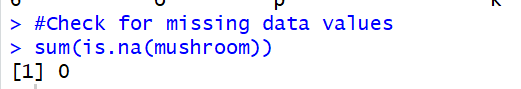
I have ran the head function after as well, not to see the data but to see the column names and make sure they were correctly put onto the dataset.

# Data Preparation.

I will start with the names which are provided on the website but will need to be assigned manually.

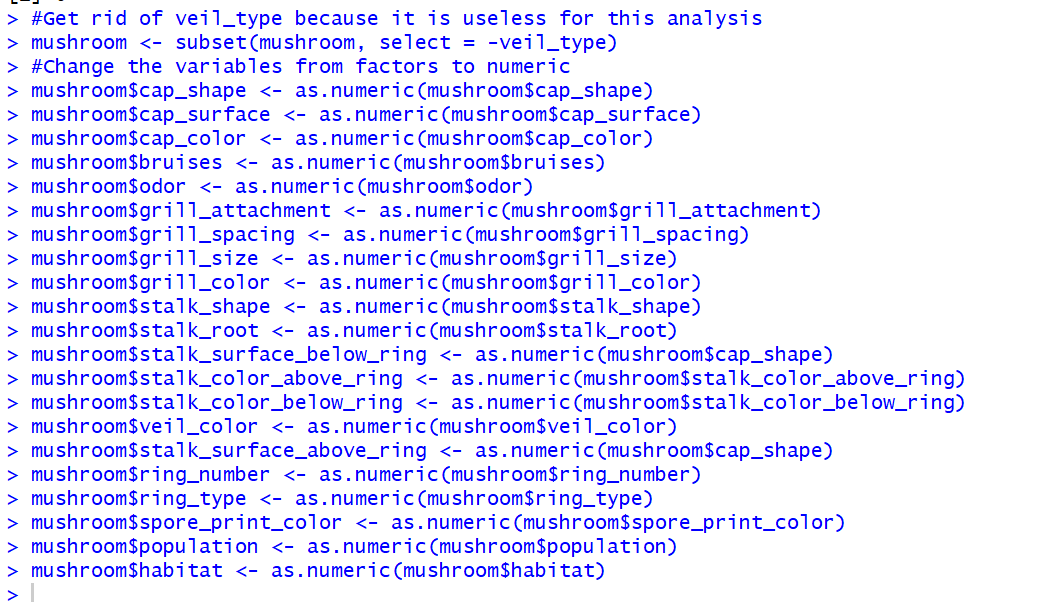


Next I’ll check the data for any missing values.



There was no missing values within the data, but from checking out the structure of the dataset we know one of the variables only has one level. That variable is veil type, this variable is pointless for the coming analysis so I will get rid of that variable.  

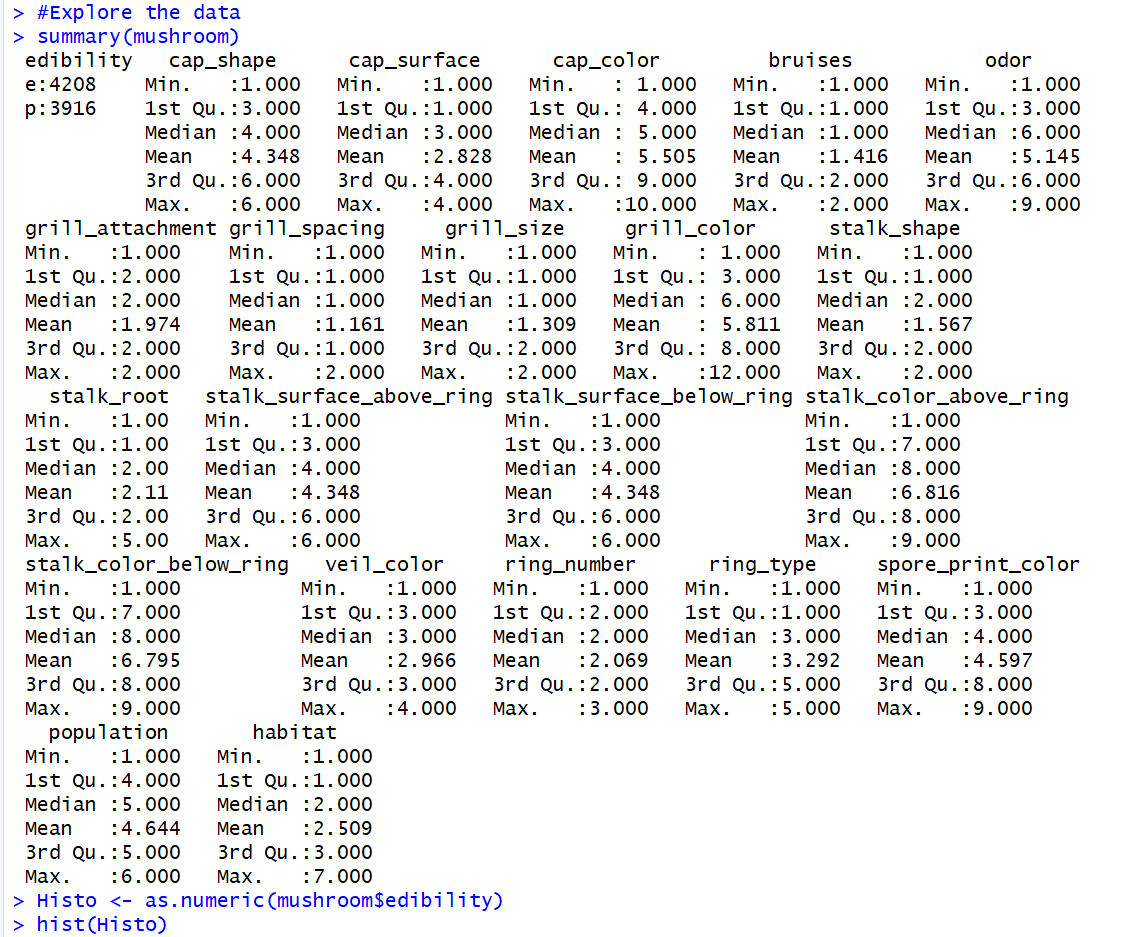

Now we have our remaining variables and only one, edibility needs to be a factor, the rest need to be numeric so I will go about the business of changing that next.

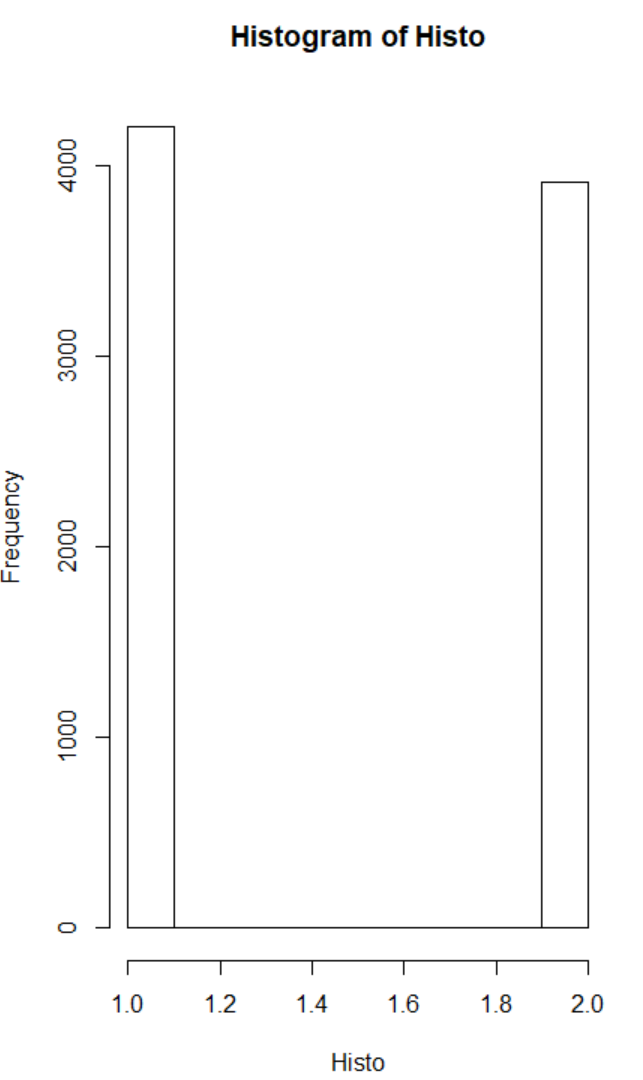


The data is essentially prepped and no longer needs any manipulation by me, so now I’ll explore the dataset.

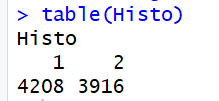
## Explore the Data

To start the data exploration process, I’m going to simply look at a summary of the variables within the mushrooms dataset, then I want to see a distribution between poisonous and non-poisonous mushrooms so I will map the distribution between the two levels and switch the edibility variable to numeric for that one graph.



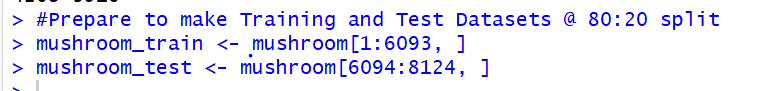


### From the summary and distribution we can see some good information about each of the variables in their numeric form. The edibility is obviously skewed slightly with a little less than 300 mushrooms difference between edible and poisonous but nothing to be concerned about. Now I’ll show a table of the edibility table to show off that fact about the first variable again.



### **SVM Model.**

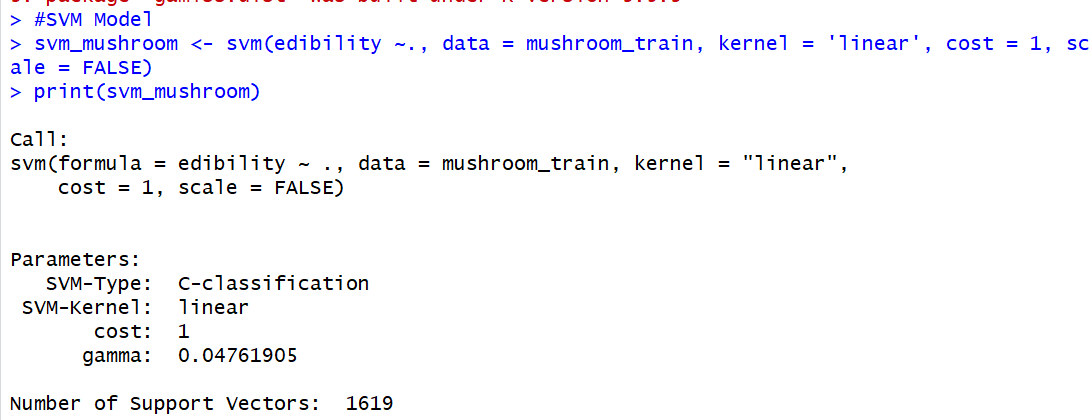
Now I will start on building the SVM model. To begin I will quickly make the training and testing data sets and I will split them into 80% train 20% test.



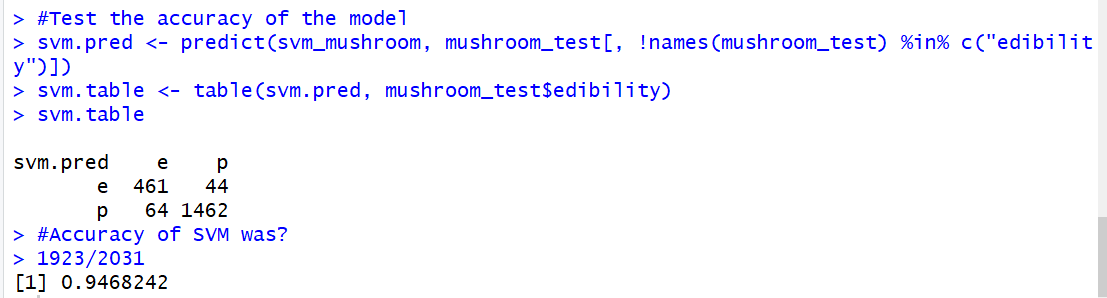
I need to bring in some RStudio packages to further my analysis and construct this SVM. So now I will do that. The packages I’m downloading in are the e1071 package and the gamlss package.



Now it is time to build the support vector machine with the mushroom data.

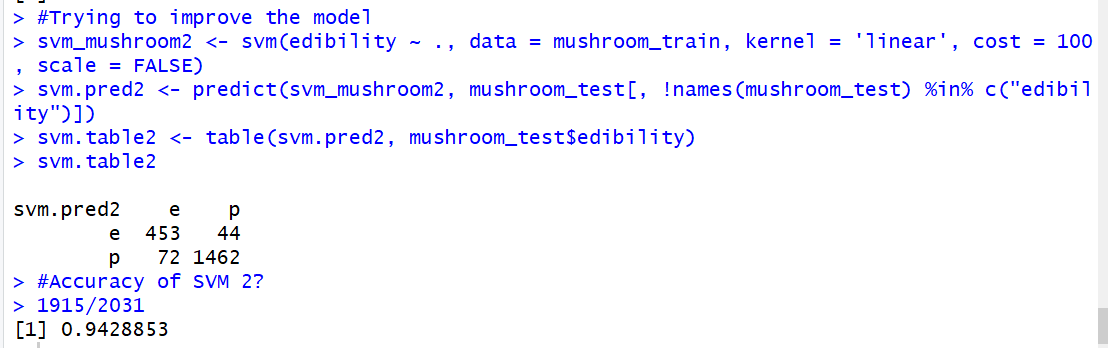


I set up the model to test edibility of the mushrooms obviously, I set the kernel to linear and kept the cost only to 1. After printing the model I know the gamma value is less than 0.05 and the number of support vectors is 1619. With model set up it is now time to test the SVM and see how accurate it is with the test dataset.

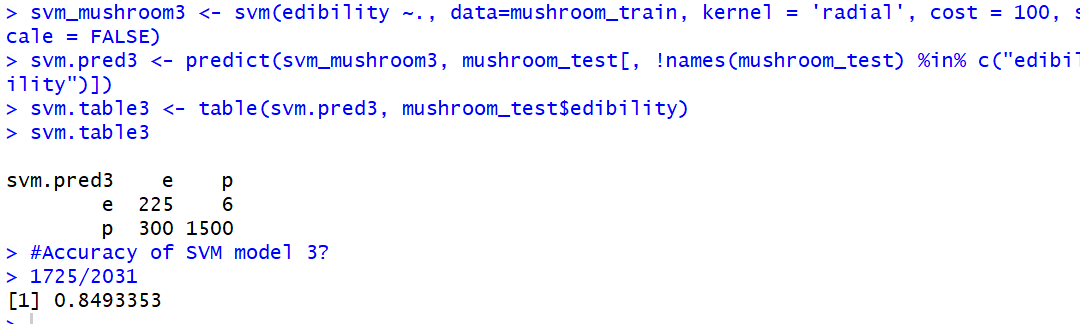


As you can see, the result of the SVM model is just under 94% which is a pretty good start to testing our data. Out of the 525 that were edible, the model predicted 461 correctly and out of the poisonous mushrooms the model predicted 1462 out of 1506 total mushrooms. This is a pretty good start to the model, but I want to try to alter it and see if I can improve the accuracy of the predictions.

The changes I’ll make to the second SVM model is I will run the model with the cost now at 100 instead of at 1, and see how the model fluctuates from a change so drastic to the cost.



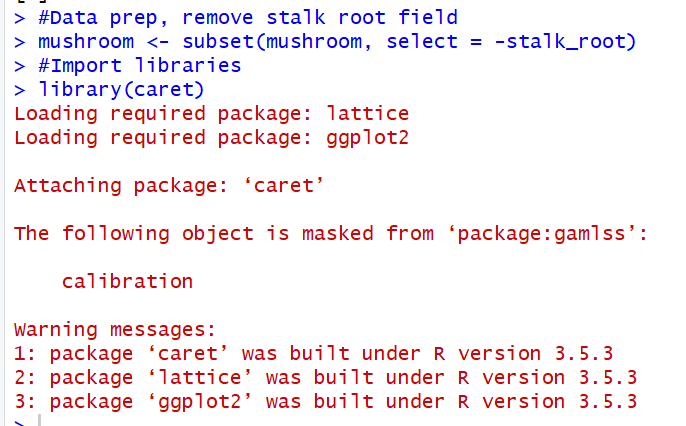
#### It wasn’t a big decrease but the second model unfortunately did worse than the first. I would like to try a third time to improve the model, however, this time I will change the kernel type. I will change the kernel type from linear to radial and hopefully see an improved result.



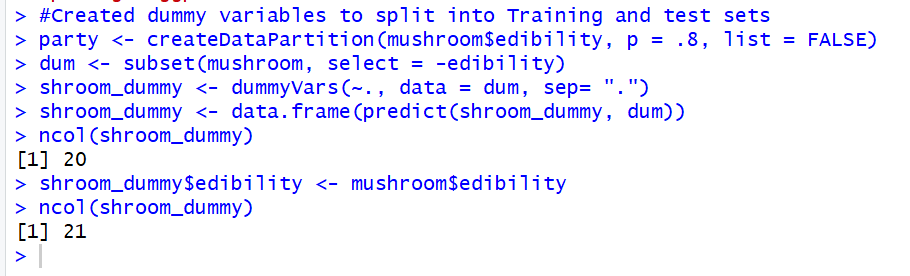
Changing the kernel and putting the cost back down to 1 drastically worsened my SVM. The third iteration was just under 85% so it may be smart to just stick with the first model and move onto creating a neural network model to compare the first SVM model to, so that is what I’ll do.

#### ***Neural Network***

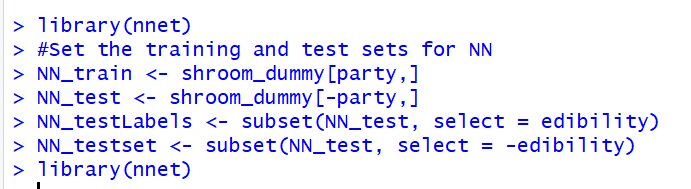
##### A neural network is similar to how a support vector machine so they make for good models to compare against one another. To start my NN I will first need to remove the stalk root variable and import the caret library.



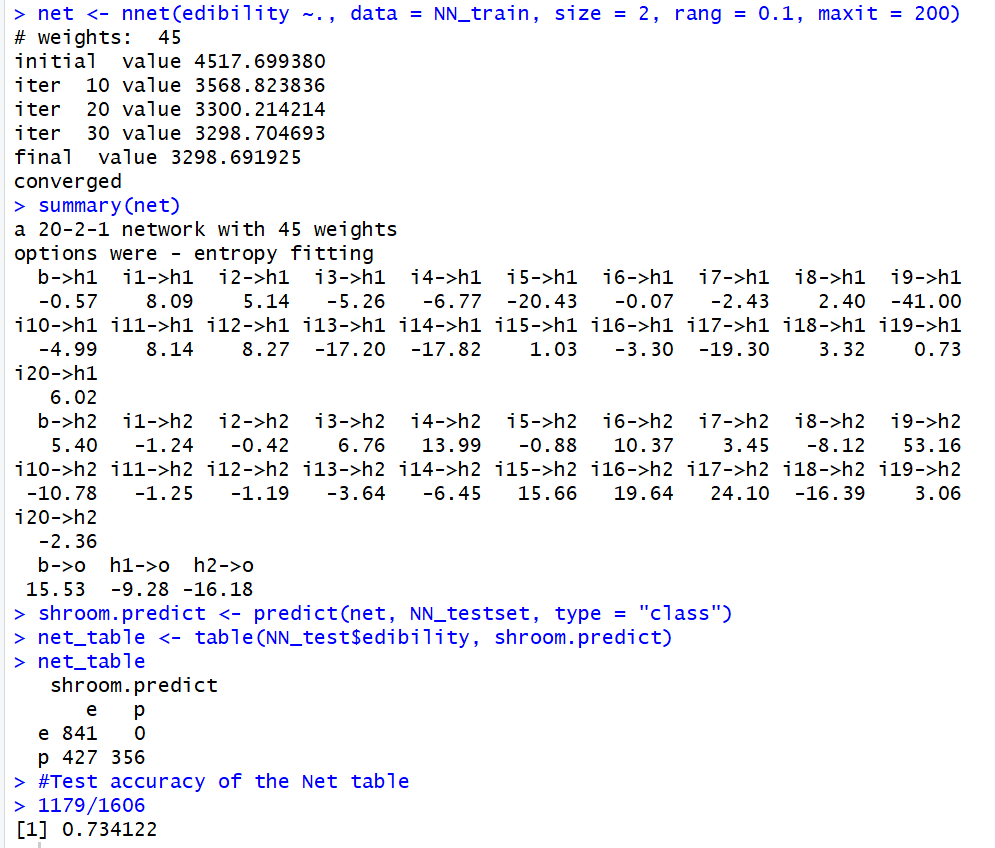
Next I’ll set up the dummy variables needed for the training and test sets.



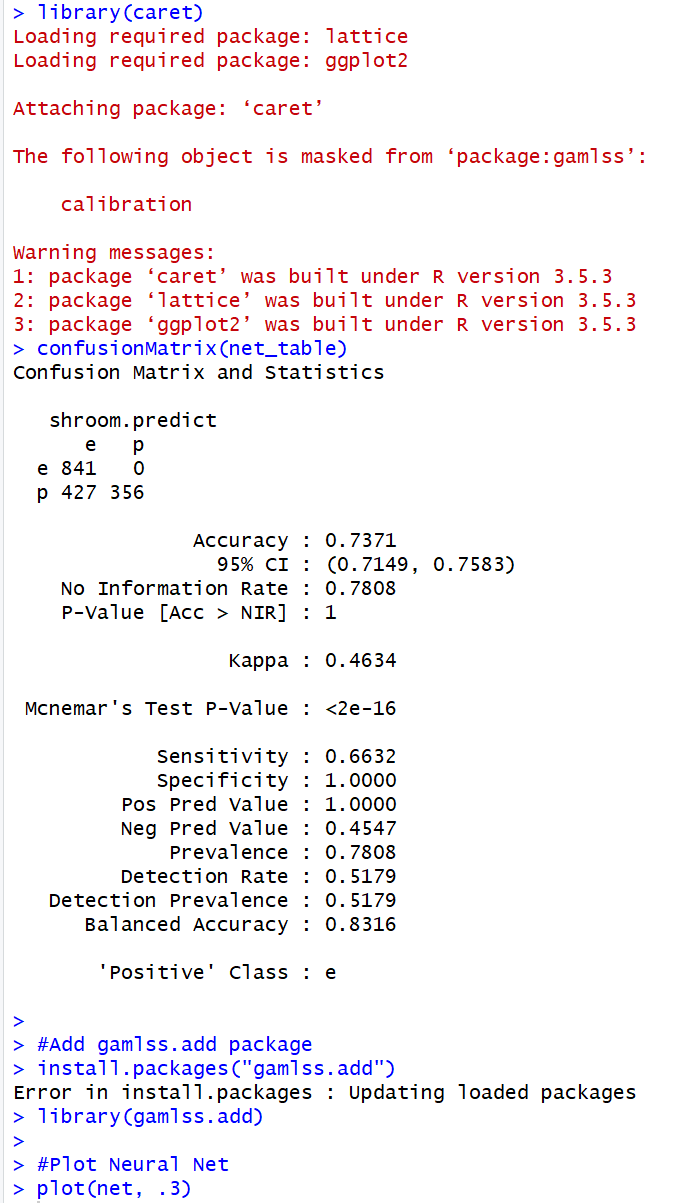
The dummy variables are set up, and I also did a column count to see how many variables are in the data with and without the variable edibility. Next I will set the training and testing sets for NN as well as bring in the nnet RStudio package.

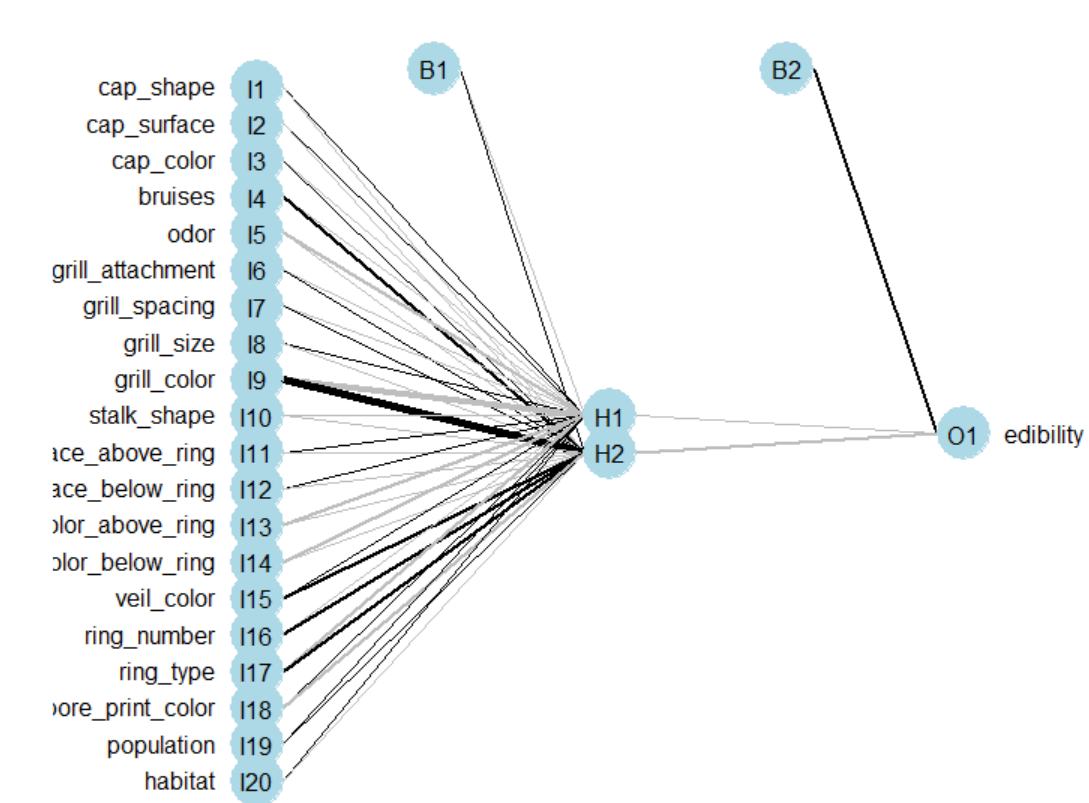


Now that the NN model is set, it is time to test the mushroom data upon it.



The accuracy of the net table is discouraging because the accuracy is only at a 73.4% which is even lower than the 3rd SVM model which was just under 85%. I want to look at a confusion matrix of the net table as well and then plot the net model. Before I run the confusion matrix I need to bring in the caret package.





From the confusion matrix and the plot it is clear to see how the model was set up and how it played out over the test data. Unfortunately the NN didn’t perform as anticipated and was just below 75%.

##### Comparing the performance of the SVM vs. the NN*.*

The Support Vector Machine surprisingly outperformed the neural network model by a margin that really surprised me. My guess is there could possibly have been some degree of overfitting of the SVM and that just was not the case for the NN. I know typically SVMs are more prone to that happening then NN so that could have happened. My other guess is the kernel and cost variables that you can assign in the SVM were able to help the model fit better upon the data, but besides that I’m not certain to why the NN got outperformed by all three SVM models.

*Summary*.

So this week I was able to learn about support vector machines and neural networks, how to run them within RStudio and what are their similarities and differences. Both seem to be quite effect as well as popular machine learning algorithms but are constructed differently. An SVM uses a kernel and a cost variable which help the model fit to the data, but the neural network doesn’t have that. The NN instead sets the size of the model and then does a range as well as a max which maxes out how many connections you have and then goes through the iterations of the model at every allotted value. From my studies I thought I would have a NN model that would perform better than the SVM but in this situation that was not the case. Perhaps I need to make adjustments to my commands but from what I’m seeing and reading it just appears the NN predictor model just didn’t do as well with the mushroom data.