**Exploratory Data Analysis (EDA)**

The goal of this exploratory data analysis is to be able to predict a specific cover type given the attributes of a cover. The data set I am provided with contains 12 measures of data, 10 of which are different quantitative variables, 4 binary wilderness areas, and 40 binary soil type variables that total to 54 attributes of data. Nothing is missing from this dataset so we do not need to deal directly with the problem of missing data but it will be discussed.

First, I sorted the data by the number of collected samples that belonged to each cover group to develop an understanding of the distribution of the data (Fig. 1)

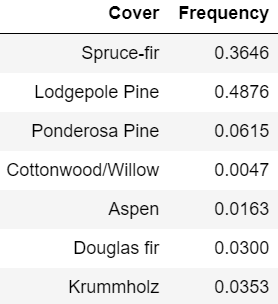
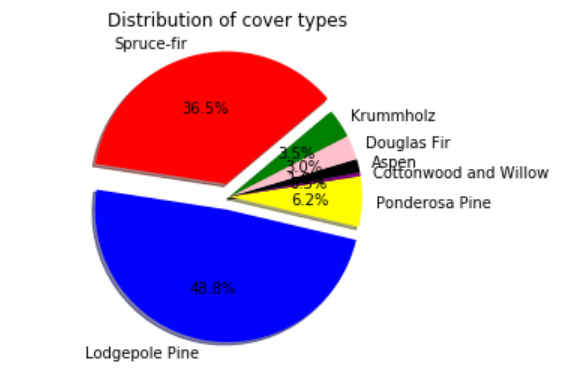


Figure 1. The distribution of the covers of every sample in the dataset

Viewing the descriptive statistics via Python’s panda's library can quickly allow me to see which variables may be most predictive of which specific cover type. For example, by grouping the data frame by “Cover” and taking the mean of each of every attribute one can clearly see that the means for the attributes vary greatly between covers (Fig. 3).

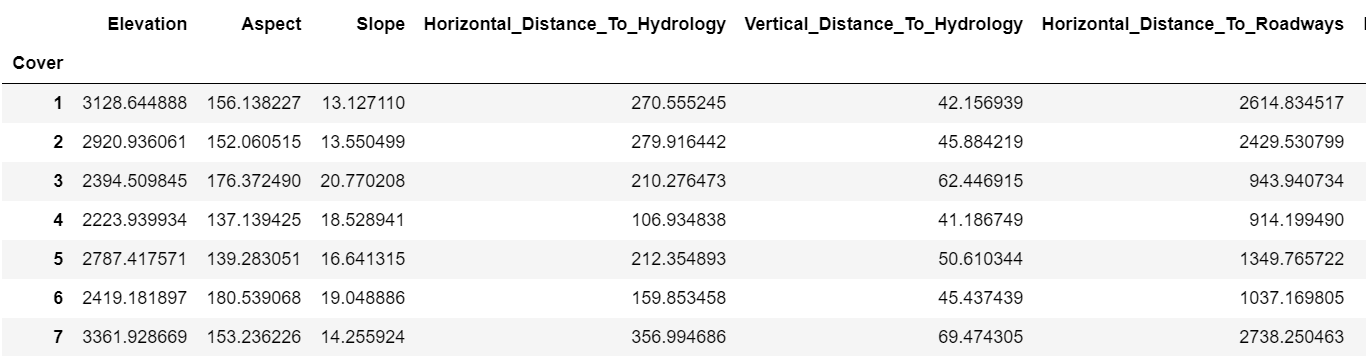
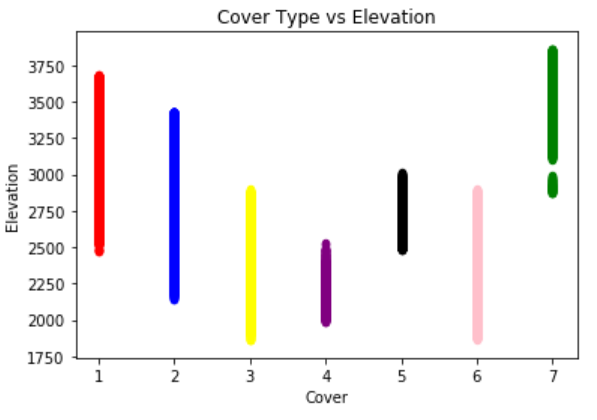


Figure 2. (Left) The range of the elevations for each cover type

Figure 3. (Right) Data table of the mean of the first 4 attributes for each cover

The first 10 variables all appear to be predictive of the different cover types due to changing means and high standard deviations seen across different cover types.

**Key Insights**

A couple of key insights were found. I found something by displaying a scatterplot between each cover type with their respective aspects and elevations. The scatterplots show that each cover lacks some variation and is much more likely to stay in their respective domains (Fig. 4). Another key insight is that the binary soil data is quite cluttered, and can be condensed into a single column representing the soil type in integer form for each sample given (Fig. 5).

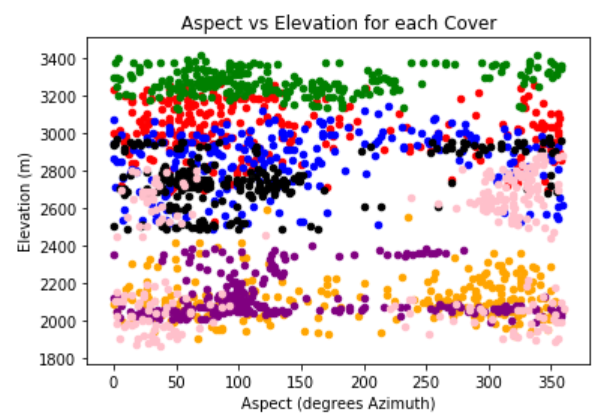


Figure 4. Shows the Aspect and Elevation levels for 100 samples of each cover type

Figure 5. Shows the condensed binary soil measurements into a single column of type of soil

**Potential Challenges**

A potential challenge that I have not found to be the case in this data set but could be the case in similar data sets is the problem of missing data. One thing one could do is throw away the entire sample related to that specific piece of missing data. The problem with this is that if there is not a lot of data to begin with then this can prove troublesome as we have thrown away many other variables that could have been useful for analysis. In this example throwing out small amounts of data where data is missing would likely not be a problem because of the 500,000+ samples in the dataset. Another thing one could do is implement a prediction model to predict the value of the missing piece of data based on the values of data that are similar to our one specific piece of data. This is useful because we don’t have to throw any of the useful data out and the process could be as simple as taking the median/mean of the surrounding instances of the variable. Another potential challenge in our data set is that the soil type is based on a binary system to show absence or presence. To mitigate this challenge I condensed the 40 columns for soil type to one soil type column that contains the integer value for soil type of a specific cover. This can make visualizing the data much easier as there is less clutter in the data frame.

**Modeling Strategy**

The model I am using is a shallow neural network created via MATLAB’s deep learning toolbox. This neural network was crafted using 70% of the dataset for training, 15% for validation, and 15% for testing. The inputs for the neural network designed are the quantitative measurements of the data set for one test, and the quantitative measurements plus the single-column representation of soil type for another test. The output is the specific cover type. The optimal number of hidden nodes and layers was determined to be 7nodes and 1 hidden layer (Fig. 6). Weights, distribution of data for training, validation, and testing, and biases were randomized at the start of every iteration of the experiment. The cost function used was cross-entropy checking, and the updating method used is a scaled conjugate backpropagation.

**Results of Modeling Strategy**

The model developed proved to be effective as shown by Fig. 5, a confusion matrix that shows the efficiency of the neural network designed. The confusion matrix shows that the network had difficulty in correctly classifying covers 4-7. This makes sense because the dataset contains much less information about these covers (~7.5% of the dataset describes these covers) than covers 1-3, which the neural network performed best with. This model can be improved because the network is predicting Cover 5 correctly only 8.7% of the time in this iteration.

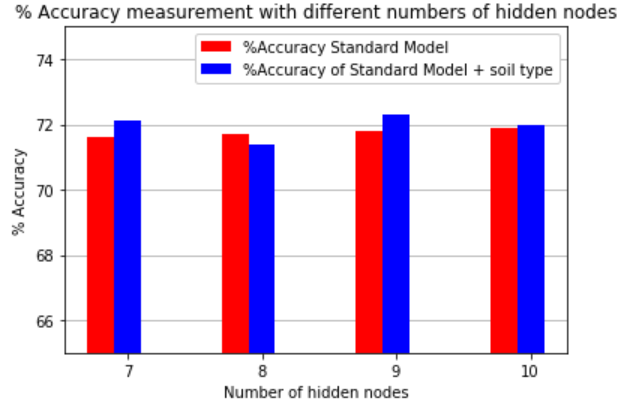
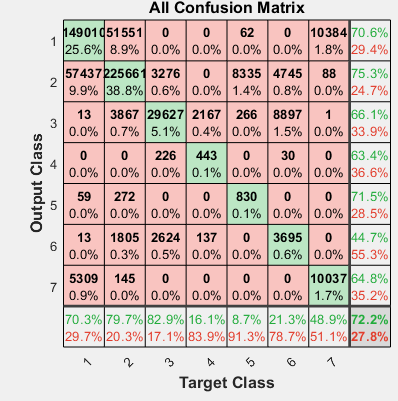


Figure 5. (Left) Shows the outputted confusion matrix

Figure 6. (Right) Shows %accuracy comparing a different number of hidden nodes

**Future Analysis**

If more time were provided for this case study then there a few other things that I would like to look more into. I would perform other modeling strategies such as discriminant analysis. Only performing discriminant analysis would likely give worse results than the current neural network because it is not as powerful, however, I would use it to help visually display the relationships between the variables collected and the cover prediction. Another strategy I would pursue is using more of the data for prediction, for example, I would make some graphs to measure the relationship between unused variables like wilderness type and relate it to the specific cover types. Perhaps I could engineer a new feature that combines soil type and wilderness area and analyze how this affects my model’s performance. This would improve the prediction of my model because it would allow for more data to become usable.