# **MAT 303 Module Six Problem Set Report**

Decision Trees

Nicholas Kreuziger

nicholas.kreuziger@snhu.edu

Southern New Hampshire University

## 1. Introduction

We will be creating two Decision Trees, a Classification Decision Tree, and a Regression Decision Tree. They will be using different datasets.

A classification decision tree is used to classify an observation as likely belonging to a particular category, in this instance it will be determining whether a person is likely to default on their credit. The dataset being explored is the “default of credit card clients Data Set” hosted by The University of California Irvine. This dataset is intended for training a machine earning model to estimate the probability of someone defaulting on their credit commitment. The response variable is whether someone defaults it is a binary measure. Results from analyses such as these can be used to assess the likelihood a credit applicant would default on their credit cards.

A Regression Decision Tree is used to predict a value of a quantitative variable, with the Regression Tree we will predict the value of wage growth. The dataset being used will be a simulated economic dataset. The theoretical scenario is that you are an analyst working for the government studying wage growth of the labor force. In this scenario the analysis would aid in setting economic policies. Results from analyses such as these can be used to assess the likely direction of wage growth. While theoretical, it could very well be applied to a real-time dataset.

## 2. Data Preparation

Within the classification tree dataset there are 24 columns and 600 rows. For our purposes we will utilize 8 columns and the full 600 rows. The variables we are interested in are represented by the table below:

Text, email

Description automatically generated

Within the regression tree dataset there are 6 columns and 99 rows. The 6 columns are detailed below:

Graphical user interface

Description automatically generated with medium confidence

## 3. Classification Decision Tree

### Reporting Results

Firstly, we will be splitting the data into training and validation sets. A training set of data will be used to train our Classification Decision Tree. This fits it to the data, which in turn allows for more accurate predictions. The Validation set of data will be used to validate how well the model was trained with the training data. The total size of the data set is 600 rows. 420 rows will account for the Training Set and 180 rows will account for the Validation set. A 70/30 split of the original 600 rows.

Next, we will create a classification decision tree for the default variable using missed payment, credit utilization and assets as predictors. Below is the cost-complexity table associated with this:

Text

Description automatically generated

The initial decision tree produced from this approach is,

Diagram, timeline

Description automatically generated

Note how this tree has 5 leaf nodes, these are the nodes on the bottom of the tree. Internal Nodes are nodes such as “credit\_utilize < 0.35”, as an example. The purpose of pruning a model is to find the optimal balance between the number of internal nodes without introducing so much fit that it lacks the robustness to accurately predict the variability of real-world data.

Pruning a model can be aided by Plotting Validation against Cost-Complexity with the dataset. The validation values are the predicted values vs the validation dataset values that we split earlier. This gives a better indication of the best Cost Complexity (CP) value to use. Below is the Plot:

Chart, line chart

Description automatically generated

The horizontal red line is one standard error above the minimum error. Minimum Error is the amount of error being allowed when the tree is making predictions on the test data. Typically, you want to select the point that is closest to the red line while not rising above it. In this instance, 0.028 Cost Complexity Parameter is selected. Below is the resulting decision tree.   
Timeline

Description automatically generated

The resulting pruned decision tree has 4 leaves. The only prevalent variables being used to indicate default likelihood are assets and credit utilization.

### Evaluating Utility of Model

A confusion matrix is used as a diagnostic for Logistic Models. A confusion matrix is a table output as:

|  |  |  |
| --- | --- | --- |
|  | Prediction = 0 | Prediction =1 |
| Actual = 0 | True Negatives | False Positives |
| Actual = 1 | False Negatives | True Positives |

Using a confusion matrix we can assess Accuracy, Precision and Recall. Defined as,

**Accuracy** is the ratio of the number of correct predictions to the total number of observations.

**Precision** is the ratio of correct positive predictions to the total predicted positives.

**Recall** is the ratio of correct positive predictions to the total positives’ examples.

We will use the Confusion Matrix to assist with evaluating the utility of this classification decision tree. A Confusion Matrix plots a count of the actual and predicted answers that the model outputs.

Graphical user interface, application

Description automatically generated

For simplicity, True Negative = 87, False Negative = 4, True Positive = 84, False Positive = 5.

### Making Predictions Using Model

Now that we have tested the utility of this model, we can make predictions. We will test this model on two scenarios.

* Likelihood of defaulting on credit for an individual who has not missed payments, owns a car and a house, and has a 30% credit utilization.
  + The model predicts no, the person will not default.
* Likelihood of defaulting on credit for an individual who has missed payments, does not have any assets, and has a 30% credit utilization.
  + The model predicts yes, the person will default. 

## 4. Regression Decision Tree

### Reporting Results

Firstly, we will be splitting the data into training and validation sets. A training set of data will be used to train our Classification Decision Tree. This fits it to the data, which in turn allows for more accurate predictions. The Validation set of data will be used to validate how well the model was trained with the training data. The total size of the data set is 99 rows. 79 rows will account for the Training Set and 20 rows will account for the Validation set. An 80/20 split of the original 99 rows.

Next, we will create a regression decision tree to predict wage growth using economy, unemployment and gdp as predictors. Below is the cost-complexity table associated with this:

Text, table

Description automatically generated

The initial decision tree produced from this approach is,

Diagram

Description automatically generated

Note how this tree has 6 leaf nodes, there’s a very real possibility this initial model is overfitted.

Pruning a model can be aided by Plotting Validation against Cost-Complexity with the dataset. The validation values are the predicted values vs the validation dataset values that we split earlier. This gives a better indication of the best Cost Complexity (CP) value to use. Below is the Plot:

Chart, line chart

Description automatically generated

The horizontal red line is one standard error above the minimum error. Minimum Error is the amount of error being allowed when the tree is making predictions on the test data. Typically, you want to select the point that is closest to the red line while not rising above it. In this instance, 0.035 Cost Complexity.

Diagram, timeline

Description automatically generated

The resulting pruned decision tree has 4 leaves. The only prevalent factor in this decision tree is unemployment as an indicator of wage growth.

### Evaluating Utility of Model

The root mean squared error is the standard deviation of prediction errors. That is to say, the root mean squared error can tell you how tightly the data is clustered around the model’s fit. This model gives us a RMSE of 0.8386.

Logo

Description automatically generated with low confidence

Considering that the GDP predictions range from 2.6-9.6 in the fitted decision tree, one could reason those productions obtaining a standard deviation of 0.8386 from the true value is promising.

### Making Predictions Using Model

Now that we have tested the utility of this model, we can make predictions. We will test this model on two scenarios.

* What is the predicted wage growth of an economy not in recession with an unemployment of 3.4% and a GDP growth rate of 3.5%?
  + The predicted wage growth is 7.7924.Chart

    Description automatically generated
* What is the predicted wage growth of an economy not in recession with an unemployment of 7.4% and a GDP growth rate of 1.5%?
  + The predicted wage growth is 2.6364.A picture containing text

    Description automatically generated

## 5. Conclusion

Using the Classification tree on a credit card dataset we were able to programmatically isolate the major junctions in the variables credit utilization, assets and missed payments that result in predicting the likelihood of credit default. We looked at two versions of this classification tree. 6 leaf nodes were overfitted, while the 4-node version below fit the model in a manner more realistic to be applied to a real dataset. Only Credit Utilization and Assets were included in this pruned model, missed payments were not considered influential enough on credit default. Timeline

Description automatically generatedUsing the Regression tree on an economic dataset we were able to programmatically isolate the major junctions in variables Economy, Unemployment and GDP to help to predict wage growth. Looking at two versions of the Regression tree, we went from 6 nodes to 4 nodes to fit the model. In this model Unemployment was the most important factor in predicting wage growth. Economy and GDP were not included.Diagram, timeline

Description automatically generated

Both models demonstrated realistic use cases of both models. Classification and Regression trees can be used in real world scenarios to provide visually easy to understand models of data.