Summer 2019

Prediction of High School Graduation with Decision Trees

Andrea M. Lee
Missouri State University, Lee317@live.missouristate.edu

As with any intellectual project, the content and views expressed in this thesis may be considered objectionable by some readers. However, this student-scholar's work has been judged to have academic value by the student's thesis committee members trained in the discipline. The content and views expressed in this thesis are those of the student-scholar and are not endorsed by Missouri State University, its Graduate College, or its employees.

PREDICTION OF HIGH SCHOOL GRADUATION WITH DECISION TREES

A Master's Thesis

Presented to

The Graduate College of

Missouri State University

In Partial Fulfillment

Of the Requirements for the Degree

Master of Science, Mathematics

By

Andrea Marie Lee

August 2019

PREDICTION OF HIGH SCHOOL GRADUATION WITH DECISION TREES

Mathematics

Missouri State University, August 2019

Master of Science

Andrea Marie Lee

ABSTRACT

While working as an educator for the past fourteen years, we are always looking at data and determining ways to help our students. Graduation status is one area of interest. I wanted to apply statistical methods to try and find early indicators of those students who may drop out, thus being able to provide early intervention to those students. With early intervention, we may be able to lower our dropout rate. While studying different methods of pattern recognition, I found that the decision tree method in machine learning was the best for the data that I had collected. Decision trees are suited for data that is numeric and categorical. It is a simplistic method of pattern recognition that is easy to interpret. Decision trees begin with a root node and attributes are tested to determine the branches that lead to the leaf node, which is where decisions or classifications are made for the target variable. Students state assessment scores and lunch status were used to find a pattern for those who graduate and those who drop out. The data was then re-run again using logistic regression. Running the data using logistic regression found some similarities with the decision tree. There was not a clear pattern that separated those students who graduate from those who dropped out. However, there are a few areas of testing that may provide a start for early intervention with our struggling students.

KEYWORDS: pattern recognition, supervised learning, decision trees, classification trees, nodes, target variable, pruning, graduation rate, proficiency, logistic regression

PREDICTION OF HIGH SCHOOL GRADUATION WITH DECISION TREES

By

Andrea Marie Lee

A Master's Thesis
Submitted to the Graduate College
Of Missouri State University
In Partial Fulfillment of the Requirements
For the Degree of Master of Science, Mathematics

August 2019

Approved:

Yingcai Su, Ph.D., Thesis Committee Chair

George Mathew, Ph.D., Committee Member

Songfeng Zheng, Ph.D., Committee Member

Julie Masterson, Ph.D., Dean of the Graduate College

In the interest of academic freedom and the principle of free speech, approval of this thesis indicates the format is acceptable and meets the academic criteria for the discipline as determined by the faculty that constitute the thesis committee. The content and views expressed in this thesis are those of the student-scholar and are not endorsed by Missouri State University, its Graduate College, or its employees.

ACKNOWLEDGEMENTS

I would like to thank the following people for their support during the course of my graduate studies. Thank you to Dr. Yingcai Su for providing direction and support while writing this thesis. Thank to Mr. Alvin Richardson for helping me collect data to use in my analysis. Finally, thank you to my family for your patience and support as I work on my degree.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

Pattern recognition, a field that began to develop in the 1960's, covers many disciplines of study including statistics, psychology, computer science as well as many others. This field has grown in recent years with the use of and improvements in technology. Pattern recognition is a scientific discipline where the goal is the classification of objects into a number of classes or categories (Theodoridis & Koutroumbas, 2006). Reliable and accurate pattern recognition is extremely useful, especially today (Duda, Hart, & Stork, 2001). Rokach and Maimon (2008) state the main objective of traditional statistical methods is model estimation, but the main objective of pattern recognition is model identification. In other words, with traditional statistics a known hypothesis is being tested, but with pattern recognition researchers are concerned with selecting a hypothesis.

According to Webb (2002), the stages of pattern recognition are as follows:

1. Formulation of the problem
2. Data Collection
3. Initial Examination of the Data
4. Feature Selection or Feature Extraction
5. Unsupervised pattern classification or clustering
6. Apply Discrimination or Regression Procedures
7. Assessment of Results
8. Interpretation

These stages may need to be stopped, reformulated, and repeated based on the findings of each stage.

Before data collection is done, the researcher needs to understand where they want to go within their investigation. Data collection could be a large portion of the cost of a study. Once the data collection has been completed, the data should be examined with summary statistics and graphs to get a better idea of the data that they have collected. In some problems the data that is available is limited. Which may cause difficulties in producing a classifier. Even when there is a significant amount of training data, the classifier could become complicated and may not do well on new patterns. The training set is a data set that is used to train the data. The researcher needs to select or extract central features from the samples. Choosing the features will rely on prior knowledge. This is to reduce the number of features and allows the design to become simpler. This may result in a poorer performance on the training sample, but a better performance on new patterns.

In supervised pattern classification, prior knowledge provides category labels or the cost for a pattern in the training set. However, unsupervised pattern classification also known as clustering is used when there is no *a priori* information known. A clustering algorithm is used to reveal groups. This may provide an end to the study or a start for supervised classification. Once groups are revealed, discrimination or regression procedures are used to design a classifier. A training set of data is used to produce the classifier. The key in creating a classifier is to not over-fit the data while at the same time not creating a classifier that is not complex enough. A classifier that over-fits the data will have a low fitting error but may not accurately classify the data in a test set of data. A classifier that is not complex enough will have a large fitting error and will not adequately model the variability of the data. The key is to find a way to optimize the tradeoff of over-fitting the data and having a model that will not appropriately model the data. The decisions made are cost specific (Duda et. al., 2001). Some of the variability that is seen may be due to not all the complexity, but also due to noise. According to Duda et. al. (2001), noise is any property of pattern which is not due to the true underlying model but is instead to due to randomness.

Once a classifier is created, it needs to be assessed. This may be done using a test set of patterns. Classifier performance may be assessed by classification error rate, the percentage of new patterns assigned to the wrong category (Duda et. al., 2001). Researchers may seek to minimize the error rate, but in many cases, it is better to minimize the expected cost or risk. Finally, the researcher is ready to interpret the results.

In Chapter 2, the decision tree methods will be reviewed. An example of a decision tree will also be given in this chapter. Then in Chapter 3, the decision tree methods will be applied to the data that was collected in trying to predict high school graduation. Next in Chapter 4, the data will be run again using logistic regression. This will give another look at the data to help verify the results and see if any new information may come to light. Chapter 5 will spend time discussing other methods of pattern classification. Finally, Chapter 6 will discuss the results of the data.

CHAPTER 2: TREE-BASED METHODS

With supervised learning the intent is to discover relationships. Relationships that are discovered are known as models. There are two main types of supervised models: classification models and regression models. Classification models are the most studied and have the greatest practical application. Decision or classification trees can be used with both classification and regression models.

Because it is natural to ask a sequence of questions in which your next questions depend on the previous answers, decision trees or classification trees have become a popular method for pattern recognition. Classification trees are a useful exploration technique that are especially beneficial when the data is nominal. This method is commonly used in marketing, finance, medicine, and engineering.

Decision Tree Description

According to Webb (2002), a classification tree is a multistage decision process. Instead of looking at the whole set of features, different subsets of features are used at different levels. Classification trees are predictive models that are used to classify an object to a predetermined set of classes based on certain attributes. For example, in the financial field, Underwriters may use a tree to determine if a loan should be approved or denied to an applicant. This is a very useful method of exploration, but it should not completely replace traditional statistical methods.

Rokach and Maimon (2008), state that decision trees are popular for their simplicity and transparency. Trees are easier for non-experts to understand especially when they are represented graphically. A non-expert would be able to follow the flow of the tree to determine a classification based on certain attributes. However, if the tree becomes too complex other methods should be used. Complexity influences the accuracy of the tree. The complexity is measured by one of the following: total number of nodes, total number of leaves, tree depth and number of attributes used (Rokach and Maimon, 2008). When a tree is too complex the simplicity is lost, and non-experts will struggle reading the graph.

Classification trees are recursive in nature. Trees consist of nodes and branches. There are several types of nodes including the root node, internal nodes, and leaf nodes. The root node is the initial node of the tree. The root node will use a specific property of the pattern and is the beginning of the classification. A root node will have no incoming edges or branches. This will then branch off to the internal or test nodes. These nodes will have an incoming edge/branch as well as outgoing edges/branches. There can be several outgoing edges from an internal node. The internal nodes are associated with variables and thresholds. These variables and thresholds are the attributes that split the object into sub-spaces. The attributes can be nominal or numerical values. When numerical values are used there will be conditions referred to in a range of values. For example, when using ages, they may be greater than twenty-five or less than or equal to twenty-five. The decisions made at the internal nodes will eventually lead to the leaf, which is considered the decision or terminal node. The leaf node is where the decisions are made to assign the pattern to a classification. The leaf will have no outgoing edge.

Objects are classified by navigating from the root down to the leaf according to the outcomes of tests along the way. The nodes are labeled with the attributes that are tested and the branches are labeled with corresponding values. Because the branches must be mutually distinct and exhaustive (Duda et. al., 2001), the conditions allow that only one set of branches will be followed, and you will only arrive at one terminal node or leaf with only one possible classification. The resulting terminal node or leaf represents the most appropriate target value for each situation. Sometimes it may result in the probability of having a certain value.

Classification trees can be used with a range of problems including determining approval for mortgages and medical decisions. They can be compactly stored, efficiently classify new samples, and have good generalization for a variety of problems (Webb, 2002). Another benefit of a classification tree is that it is a natural way of incorporating prior knowledge (Duda et. al., 2001). While ever process has its advantages, it also has its disadvantages. The design of the optimal tree is difficult. Large trees may be created with poor error rates. Also, if there is an additional it may require a redesign of the tree.

Decision Tree Creation Process

According to Webb (2002), there are three steps in the constructing of a classification tree:

1. Selecting a splitting rule for each internal node. This involves selected features and thresholds that will separate the data at each node.
2. Determining which nodes are terminal. At each node, a decision needs to be made to continue splitting or make that node a terminal node in which it will be assigned a class label.
3. Assigning class labels to the terminal nodes. Labels can be assigned to minimize the misclassification rate.

As with other methods, a training set of data is used to create a tree. Within these steps, the CART (classification and regression trees) methodology provides an outline that can be used in a variety of ways to produce different trees. The CART method is just one method used. The CART method was first introduced in 1984 by Leo Breiman, Jerome Friedman, Richard Olshen, and Charles Stone (Rao, 2013). Rokach and Maimon (2008), state that the CART method can consider misclassification costs as well as allowing users to provide prior probability distribution. However, further questions will arise along the way. Duda et. al. (2001) offers six questions that generally arise.

1. How many splits should occur at each node?
2. Which property should be tested at a node?
3. When should a node be declared a leaf?
4. If a tree is too large, how can it be pruned?
5. If a leaf is impure how should the category label be assigned?
6. How should missing data be handled? (p. 396-397)

The number of splits may vary throughout a tree. Many trees use binary splits. According to Duda et al. (2001), this is because of the expressive power and simplicity of the training. Decisions on which property test will be performed at a node will also vary within the tree. Decisions could be made by a threshold on a single variable as well as linear or nonlinear combination of variables (Webb, 2002). Duda et al. (2001) states the fundamental principle within tree creation is simplicity leading to simple trees with few nodes.

If a tree is grown fully the data is typically overfit. As with other models when a training set is overfit, the model is not accurate for new patterns. This could also result in having high Bayes error when there is noise in the patterns. However, if a tree is stopped too early performance may suffer again. Another downside of decision tree is a high variance (Theodoridis and Koutroumbas, 2006). High variance means that a slight change in the training data set could result in a very different tree (Theodoridis and Koutroumbas, 2006). Because of the top-down behavior of the tree, an error in the beginning of the tree will affect the tree all the way to the bottom, resulting in an entirely different result.

Validation is one approach to determine when to stop a tree. In this method, the tree is trained using a subset of the training data with the remaining used as a validation set. The tree is split until the error on the validation set is minimized. An improvement of the CART method over others is the application of validation (Ville & Neville, 2013).

Another method that is used to determine when to stop growing the tree is to set a small value in reduction in impurity. The splitting is stopped when a split reduces the impurity by less than present amount. This method will use all the data, but it is tough to know how to set the threshold. These are just two methods in determining when to stop growing the tree.

Pruning a tree is used because when the splitting is stopped benefits of future nodes may be taken away. Because of this it is suggested to grow the tree and then prune it. Pairs of neighboring leaf nodes are considered for elimination and any pair whose elimination results in a satisfactory increase in impurity is eliminated. This avoids the horizon effect and uses all of the information provided in the training set; however, it is a greater expense to prune the tree (Duda et al., 2001).

Decision Tree Construction

Using notation from Webb (2002), which is based on the CART method, a tree is defined to be set of T positive integers with functions $l(.)$ and $r(.)$ from T to $T \cup \{0\}$. The functions $l(t)$ and $r(t)$ represent the left and right nodes. The nodes of the tree correspond to a member of T. For each $t \in T$, the left and right nodes will either both be equal to zero or both be greater than zero. If both nodes are equal to zero, we have a terminal or leaf node. If both nodes are greater than zero, we have a non-terminal or internal node. A subtree is defined to be a non-empty subset T_1 of T along with the functions l_1 and r_1 such that

$$l_1(t) = \begin{cases} l(t), & \text{if } l(t) \in T_1 \\ 0, & \text{otherwise} \end{cases}$$

$$r_1(t) = \begin{cases} r(t), & \text{if } r(t) \in T_1 \\ 0, & \text{otherwise} \end{cases}$$

if T_1, $l_1(.)$ and $r(.)$ form a tree. A pruned subtree T_1 of T is a subtree with same root as T.

Therefore $T_1 \leq T$.

Terminal nodes are notated as $\tilde{T}$. If we let $\{u(t), t \in \tilde{T}\}$ where $u(t)$ is a subspace of $\mathbb{R}^p$ (the data space). As well if $t \neq s$ and $t, s \in \tilde{T}$, then $u(t) \cap u(s) = \emptyset$. In addition, $\bigcup_{t \in \tilde{T}} u(t) = \mathbb{R}^p$. The class labels are denoted with ω and we let $\omega_{j(t)} \in \{\omega_1, \cdots, \omega_C\}$. The classification trees consist of class labels $\left\{\omega_{j(t)}, t \in \tilde{T}\right\}$ and the partition$\{u(t), t \in \tilde{T}\}$.

According to Webb (2002), a labelled data set $\mathcal{L}$ is used to construct a classification tree, where $\mathcal{L} = \{(\boldsymbol{x}_i, y_i), i = 1, \cdots, n\}$. The data samples are represented by $\boldsymbol{x}_i$, and the corresponding class labels are y_i. $N(t)$ will represent the number of samples of $\mathcal{L}$ in which $\boldsymbol{x}_i \in u(t)$ and $N_j(t)$ will represent the number of samples for which $\boldsymbol{x}_i \in u(t)$ and $y_i = \omega_j$. Therefore, by definition, $\sum_j N_j(t) = N(t)$. Webb (2002) states that an estimate of $p(\boldsymbol{x} \in u(t))$ that is based on $\mathcal{L}$, may be defined as

$$p(t) = \frac{N(t)}{n} \qquad 1$$

and then an estimate of $p(y = \omega_j \middle| \boldsymbol{x} \in u(t))$ again based on $\mathcal{L}$ is given as

$$p(\omega_j | t) = \frac{N_j(t)}{N(t)} \qquad 2$$

and last, estimates of $p(\boldsymbol{x} \in u(t_L) | \boldsymbol{x} \in u(t))$ and $p(\boldsymbol{x} \in u(t_R) | \boldsymbol{x} \in u(t))$ based on $\mathcal{L}$, where $t_L = l(t)$, $t_R = r(t)$ will be respectively represented as

$$p_L = \frac{p(t_L)}{p(t)}, p_R = \frac{p(t_R)}{p(t)}$$

Labels will be assigned to each node, t, according to the proportions of samples from each class in $u(t)$. The label ω_j will be assigned to t if

$$p(\omega_j | t) = \max_i p(\omega_i | t)$$

Splitting rules are ways to determine which variables should be used at a certain node.

Splitting rules are also used to divide the sample into subgroups and to decide the threshold for a variable. Splits denoted s_p will consist of a condition on the elements of the vector $\boldsymbol{x} \in \mathbb{R}^p$. A split may be defined in different ways including a threshold on an individual feature or a threshold of a linear combination of features. Thus, if at a node $\boldsymbol{x} \in u(t)$ and $\boldsymbol{x} \in s_p$, then we will move to $l(t)$. If $\boldsymbol{x} \notin s_p$, then we move to $r(t)$.

The question then becomes what property should be tested at each node. The goal of tree classification is simplicity, creating a tree with minimal nodes. To help ensure a simplistic tree we search to test the properties that lower the impurity of the nodes (Duda et al., 2001). Webb (2002), defined the impurity function to be

$$I(t) = \phi(p(\omega_1|t), \dots , p(\omega_C|t))$$

where ϕ is a function defined on all C-tuples $(q_1, \dots , q_C)$ such that $q_j \geq 0$ and $\sum_j q_j = 1$. The node impurity function has three properties:

1. ϕ is a maximum only when $q_j = 1/C$ for all j.
2. It is a minimum when $q_j = 1$, $q_i = 0$, $i \neq j$, for all j.
3. It is symmetric function of $q_1, \dots , q_C$. (p. 232)

Theodoridis and Koutroumbas (2006), use the following formula for node impurity

$$I(t) = -\sum_{i=1}^{C} p(\omega_i|t) \log_2 p(\omega_i|t)$$

This is known as the entropy impurity. This function also has a maximum when the probabilities are equal to $1/c$, which is the highest impurity. The function will also become zero if all the data belongs to a single class (Theodoridis and Koutroumbas, 2006).

According to Webb (2002), the change in the impurity function is one measure of the validity of a split. The change in impurity can be represented as

$$\Delta I(s_p, t) \triangleq I(t) - (I(t_L)p_L + I(t_R)p_R)$$

over all splits s_p. Theodoridis and Koutroumbas (2006), state that the goal is to choose the split that leads to the highest decrease in impurity.

Several forms of the impurity function have been used, but both Duda et al. (2001) and Webb (2002) have used the *Gini impurity*. The formula is given by

$$I(t) = \sum_{i \neq j} p(\omega_i|t)p(\omega_j|t)$$

Duda et al. (2001) states that this is the expected error rate at the node, t, when a category is selected from the class distribution. Using the impurity function splits may be evaluated, however, there are many possible splits and searching through the possible splits may be exhaustive. To assist in dealing with this problem the researcher may use the following steps: first, only look at splits of a certain form. Looking at splits at a threshold on individual variables s_p is then defined to be $s_p = \{\boldsymbol{x}; x_k \leq \tau\}$, where $k = 1, \dots, p$ and τ ranges over the real numbers (Webb, 2002). To limit the number of splits that are inspected allow each variable x_k, τ to take one of a finite number of values within a range of possible values (Webb, 2002). Each variable has then been divided into a number of categories in which the computations should be kept at a reasonable number.

As the researcher grows the tree, the question then becomes when to stop. A researcher does not want to create a tree that over-fits the data. One way to determine when to stop is to create a stopping rule. This could be done by stating a predetermined threshold and then the node will not be split if the change in impurity is lower than the threshold (Theodoridis and Koutroumbas, 2006). Another method is to grow the true until the nodes are nearly pure and then prune the tree (Webb, 2002). Theodoridis and Koutroumbas (2006), state that the subset will be pure if all points in the subset belong to a single class. When pruning is used the tree is grown so that it overfits the data, then this tree is pruned by removing sub-branches that are not contributing to the generalization accuracy (Rokach and Maimon, 2008). According to Webb (2002), pruning will lead to better performance than will a stopping rule.

The following pruning algorithm comes from the CART method of classification (Webb, 2002).

$$R(t) = r(t)p(t)$$

Where $R(t)$ are the real numbers associated with each node t of a tree T. $R(t)$ may represent the proportion of misclassified samples if t is a terminal node. The misclassified samples are denoted as $M(t)$. Therefore,

$$R(t) = \frac{M(t)}{n}, t \in \tilde{T}$$

where n is the total number of data points. $\tilde{T}$ was defined earlier as terminal nodes. We then let $R_\alpha(t) = R(t) + \alpha$, where α is a real number. Then,

$$R(T) = \sum_{t \in \tilde{T}} R(t)$$

$$R_\alpha(T) = \sum_{t \in \tilde{T}} R_\alpha(t) = R(T) + \alpha|\tilde{T}|$$

The estimated misclassification rate is $R(T)$, $|\tilde{T}|$ is the cardinality of $\tilde{T}$, the estimated complexity-misclassification rate of a classification tree is $R_\alpha(T)$, and last α is constant which can be thought of as the cost of complexity per terminal node (Webb, 2002). As α increases, the minimizing subtree will have fewer terminal nodes.

Now to define $r(t)$.

$$r(t) = 1 - \max_{\omega_j} p(\omega_j|t)$$

We now have all the parts that make up the function $R(t) = r(t)p(t)$, where $p(t)$ was given earlier by equation 1 and $p(\omega_j|t)$ was given by equation 2. Therefore, if t is a terminal node, then $R(t)$ is the influence of that node to the total error (Webb, 2002).

If T_t is a subtree with root t and $R_\alpha(T_t) < R_\alpha(t)$, then the influence on the cost of complexity of the subtree is less than that for the node t (Webb, 2002). This will occur for small values of α. If α increases, then equality is found when

$$\alpha = g(t) = \frac{R(t) - R(T_t)}{N_d(t) - 1}$$

where $N_d(t)$ is the number of terminal nodes in the subtree T_t and termination of the tree at t is preferred (Webb, 2002). Thus, α or $g(t)$ is a measure of the strength of the link from the node t. In applying this process, the first step is to search for the node with the g value of $g(t)$. This is then made a terminal node and values of $g(t)$ are recalculated for its ancestors. This process is then repeated and will continue until only the root node is remaining. This pruning algorithm generates a succession of trees, where the tree at the *k*th stage is denoted by T^k. The values of $g(t)$ at each stage of the progression are denoted by α_k. The pruned tree T^k has all internal nodes with value of $g(t) > \alpha_k$.

The process presented by the CART method can be summarized with the following three steps (Webb, 2002). These steps are used when give a training set $\mathcal{L}_r$ and a test set $\mathcal{L}_s$.

1. Use the training set to generate a tree T. This is done by splitting all the nodes until the terminal nodes are pure. Where pure implies that all samples at each of the terminal nodes belong to the same class. If this not possible, an alternate approach is to stop when the number in a terminal node is less than a given threshold.
2. Use the CART pruning algorithm to generate a sequence of subtrees T^k using the test set.
3. Select the smallest subtree in which $R(T^k)$ is a minimum. (p. 236)
   An advantage of the decision tree method is that it is able to handle missing data.

Missing attributes may appear in the training or classification stage. They may also appear in both stages. Duda et al. (2001) states that you should first attempt to train the tree even though some of the patterns are missing attributes. Deleting the deficient patterns should only be considered if there are many complete patterns. Instead of deleting the patterns with missing data, Duda et al. (2001) suggests that researchers proceed as normal, however, when calculating impurities as a node, only use the attribute information that is available. When calculating the best split at a node where one of the patterns is missing an attribute, researchers will calculate the possible split of all points at the attributes that are complete and then use only those that are available at the missing attribute. Therefore, n points me be used at the first attribute x_1 and only $n-1$ points at attribute x_2. Reduction of impurities are still calculated even with different values of patterns. The desired split is still the one that produces the greatest reduction in impurity.

The process for when the test pattern is missing attributes is to create surrogate splits. During the training, the primary splits are created. But in addition, the nonterminal nodes are given ordered surrogate splits that consist of an attribute label and rule (Duda et al., 2001).

The CART algorithm uses surrogate splits to handle missing data. Webb (2002) explains this process. The best split of a node is $\int$ on the variable x_m, then the split $\int^*$ predicts $\int$ most accurately on a variable x_j which is a variable other than x_m is the known as the best surrogate for $\int$. Then a tree will be constructed following the normal algorithm, but at each node t, the best split $\int$ on a variable x_m is found by using only the samples where x_m is found. Objects are then assigned to t_L and t_R according to x_m. If these values are missing for the test data, then the split is made using the best surrogate, or if needed, the second-best surrogate. Duda et al. (2001) states that during classification of a test pattern that is missing attributes, the first split that does not involve the missing attributes is used.

Duda et al. (2001) also suggests another method known as virtual values where the missing attribute is assigned to its most likely value. Missing values may be just as informative as values that are present. A missing attribute in some cases may become a new feature and used in the classification. Duda et al. (2001) provides an example of this. In a medical diagnosis, a missing attribute may imply that a doctor had a reason not to measure that attribute.

Misclassification can occur within any data set. The goal may become to minimize the general cost. This is the cost when a pattern is classified as ω_i when it is actually ω_j. The misclassification rate is given as

$$R(T) = \sum_{ij} q(i|j)\pi(j)$$

where $q(i|j)$ is the proportion of samples of class ω_j assigned to class ω_i by the tree. The prior probabilities $\pi(j)$ are equal to N_j/n. Earlier λ_{ji} was defined as the cost of assigning a pattern to ω_i when it was an element of ω_j. Using this definition, the misclassification cost is then

$$R(T) = \sum_{ij} \lambda_{ji} q(i|j)\pi(j)$$

Advantages and Disadvantages

Rokach and Maimon (2008) site several disadvantages and advantages of using a decision tree for the purpose of classification. Disadvantages include the following:

1. Most of the algorithms require that the target attribute will have only discrete values.
2. As decision trees use the “divide and conquer” method, they tend to perform well if a few relevant attributes exist, but less so if many complex interactions are there.
3. The over-sensitivity to the training set, to irrelevant attributes, and to noise make decision trees unstable. A small change close to the root of the tree will result in changing the whole subtree below. Attributes that are not truly the best may be chosen.
4. The fragmentation problem causes portioning of the data into smaller fragments. If data splits are approximately equal on every split, then a univariate tree can only test a certain number of features, which is a disadvantage if there are many relevant features.
5. The effort needed to deal with missing values is another disadvantage. The CART method uses a complex scheme of surrogate features.
6. Last, the narrow-minded nature of most of the decision tree algorithms is shown by the fact that inducers look only on level ahead. For example, splitting criterion ranks possible attributes based on their immediate descendants. This may overlook combinations of attributes.

The advantages of decision trees include:

1. Trees are self-explanatory, and if they are compact, they are easy to follow.
2. Nominal and numeric input can be used with decision trees.
3. The representation of a decision tree is rich enough to represent a discrete-value classifier.
4. Trees can also handle datasets that have errors.
5. Decision trees can also deal with datasets with missing values.
6. Decision trees are a thought of as a nonparametric method; therefore, they do not include assumptions about the space distribution and on the classifier structure.
7. Because decision trees only ask for values of the feature along a single path from root to a leaf, they be more appealing when the classification cost is high. (p. 73-76)

Nisbet et al. states some additional advantages to the CART method:

1. CART is not significantly impacted by outliers in the input variables
2. Stopping rules may be relaxed to overgrow the tree and the prune the tree to an optimal size, thus set minimizing the probability that important structure will be overlooked by stopping too soon.
3. CART incorporates testing with test data and cross-validation to assess the goodness of fit more accurately.
4. CART can use the same variables more than once in different parts of the tree which may uncover complex interdependencies between set of variables.
5. CART can be used in conjunction with other models to select the input set of variables. (as cited in Rao, 2013, para. 13)

With all the disadvantages presented, decision trees may not be the desired method of classification. But the advantages that appear support that in many cases, it is the desired method. An attraction of the CART method is simplicity. Classifying a sample may be done in only a few simple tests. Binary splits on single variables are performed in a recursive manner. And with this simplicity, it still gives a superior performance to many traditional methods on complex data sets of many variables. Webb (2002), states that lack of interpretability is a major limitation of many of the other methods of classification but is an advantage of the univariate splitting.

Decision Tree Example

Figure 1 represents an example of a decision tree. In this tree, the root node is savings. The branches then lead to the internal and leaf nodes. The first branches relate to the root node. In this example, assets and income are internal nodes. The leaf node is the credit risk.

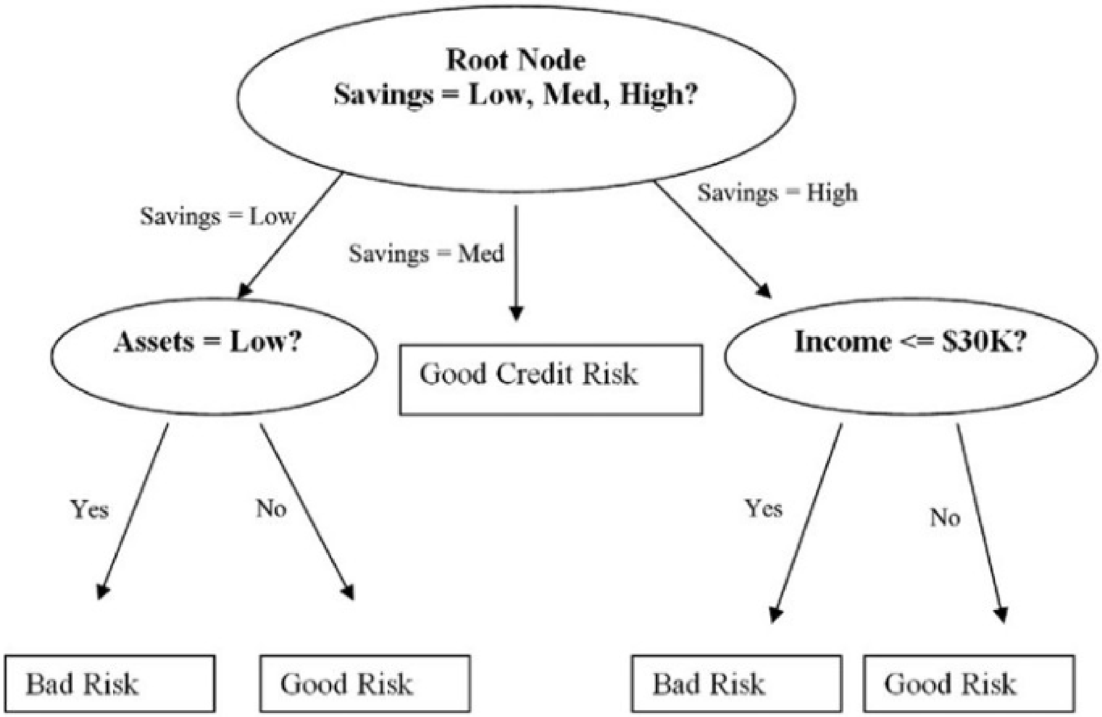

Figure 1. Decision Tree Example. Reprinted from "Decision trees – A simple way to visualize a decision," by R. S. Brid, 2018, *Medium.* Retrieved from https://medium.com/greyatom/decision-trees-a-simple-way-to-visualize-a-decision-dc506a403aeb

If a person's savings are low or high, more information is required is needed to determine if they are a good or bad risk. However, if a person's savings is medium, they are considered good risk. A person whose savings are high and income that is less than or equal to $30,000 is considered a bad credit risk. While a person with low savings and high assets is considered a good risk. This is just one example of how a decision tree is easy to interpret. Which is an advantage of the decision tree over other models.

CHAPTER 3: GRADUATION DATA ANALYSIS WITH TREE-BASED METHODS

Data Collection

In the field of education, state scores, attendance, standards, and graduation rate are major topics that are discussed. The goal for the state of Missouri is to be one of the top ten states by 2020. According to the MSIP-5 (Missouri School Improvement Program) comprehensive guide the Annual Performance Report (APR) is used to measure progress to the goal to be in the top ten by 2020. The MSIP-6 will begin to be implemented in 2020 or 2021. The APR is comprised of scores from the MSIP-5 performance standards which include: Academic Achievement, Subgroup Achievement, College and Career Readiness, Attendance Rate, and Graduation Rate. Status, progress, and growth are then used to calculate a score that will determine the accreditation level of the school district (*MSIP-5: Comprehensive Guide*, 2014, p. 5). Schools are expected to meet or exceed the standards in each of these areas. The performance standards are further explained by the MSIP-5 Comprehensive Guide:

1. Academic Achievement – districts administer assessments required by the Missouri Assessment Program (MAP) that measure academic achievement and demonstrates improvement in performance of its students over time.
2. Subgroup Achievement - the district demonstrates improvement in student performance for its subgroups. Subgroups include free/reduced price lunch, racial/ethnic background, English language learners, and students with disabilities.
3. College and Career Readiness -district provides adequate post-secondary preparation for all students. This is measured by the scores on measures of college and career readiness like the ACT, SAT, COMPASS, and Armed Services Vocational Aptitude Battery (ASVAB). It may also be measured by scores on an Advanced Placement (AP), International Baccalaureate (IP), Project Lead the Way (PLTW), or Technical Skills Attainment (TSA) assessments or grades through early college, dual enrollment, or approved dual credit courses. It also includes that the percent of graduates who attend post-secondary education/training or are in the military within six months meet the state standard or demonstrates required improvement. The percent of graduates who complete approved career education programs and are placed in occupations related to their

training, continue their education, or are in the military within six months of graduating meets the state standard or demonstrates required improvement.

4. Attendance Rate – district ensures all students regularly attend school.
5. Graduation Rate – district ensures all students successfully complete high school. (p. 6-7)

Schools are awarded points in these standards. Half of the points are given in the academic areas and the other half are given for the readiness categories. There are fifty-six points possible in the academic categories of Academic Achievement and fourteen points are possible in the Subgroup Achievement for a total of seventy points. The readiness categories of College and Career Readiness, Attendance Rate, and Graduation rate account for thirty, ten, and thirty points respectively again for a total of seventy points. There is a possibility of one-hundred and forty points. If a school receives seventy percent of the APR points, they are accredited, and ninety percent earns accreditation with distinction (with other criteria established by the State Board of Education). Graduation rate accounts for forty-three percent of the readiness points and twenty-one percent of the total points awarded to school for the APR.

Because of these performance standards, schools are always looking for ways to improve in these areas. One focus is to find ways to help our students stay in school and graduate. While collecting data, the initial goal was to look at the test scores and attendance of students throughout their school years and try to find a pattern for those who will drop out. The data was collected from the Marshfield R-1 School District. It was not possible to gather the elementary attendance data, but I was able to gather test scores, lunch status, and graduation status for the students. The data was from the students who were enrolled at Marshfield High School in the 2016-2017 and 2017-2018 school years. Looking at state test scores as well as their lunch status, the decision tree method will be used to determine a pattern for those who graduate and do not graduate from high school.

Data Explanation

Students are tested throughout their school career. The Missouri Assessment Program (MAP) is a grade level test given in the third through eighth grades. These tests are given in English language arts and mathematics for all these grade levels. Science is tested in the fifth and eighth grade only. The End of Course (EOC) is given at the high school level. The EOC is based on content rather than grade level. The EOC tests include tests in the following content areas American History, Government, Algebra 1, Algebra 2, Geometry, English 1, English 2, and Biology and Physical Science. All students take the MAP tests except for math 8. If a student takes Algebra 1 in the eighth grade, they will take the Algebra 1 EOC instead. The EOC's depend on what classes the students are enrolled in. There has been changes over the years. If MAP scores are missing for students, this may be due to the students not being enrolled at our school that year. Another reason that scores were missing is if took the MAP-A instead. The MAP-A is the alternative MAP for those students with severe cognitive disabilities. Scores were delayed at the end of last year due to changes in the testing, so some students are missing scores due to this.

The state of Missouri currently requires students to take the Algebra 1, English 2, Biology, and Government EOC's before they graduate. The Government EOC is given when the take US Government, which is either their junior or senior year. We rarely give the Geometry EOC. This is an optional test with the state. We give this test to students who have taken Algebra 1 as a seventh grader and are taking Geometry as an eighth grader. The Algebra 2 test is an optional test as well, but our school gives this test to all students enrolled in Algebra 2. If a student has taken the Algebra 1 EOC at the junior high level, they are required to take the Algebra 2 exam. Therefore, we choose to assess all students enrolled in Algebra 2. Not all students will take Algebra 2 before they graduate, therefore, we will not have Algebra 2 EOC scores for all our students. We no longer give the American History EOC, therefore, underclassmen will not have scores for this exam.

With changes in the format of the test and changes from the state the scaled scores are not an accurate way to compare scores over the years. So instead, we will look at the levels assigned to the students. Students are assigned to either below basic, basic, proficient, or advanced. When a new test is field tested, new cut scores are decided to help determine these levels. The states assess the schools based on many details including the proficiency of our students. Proficiency is determined by the percentage of students who receive proficient or advanced on the MAP or EOC tests.

Graduation rates are used to for state and federal accountability. The Missouri Department of Education has determined that graduation rates are one of the most important indicators of a school's success. According to the Missouri Department of Education the primary purpose of the Missouri School Improvement Program (MSIP) is to assess the accountability for school district accreditation. Accountability data is based on scores based on academic achievement, subgroup achievement, college and career readiness, attendance rate, and graduation rate.

The Otis-Lennon School Ability Test (OLSAT) is a test given to some students in the third or fifth grade. At the time that these students were in elementary school, not all students were given the OLSAT. This is an intelligence test that is given to determine students who are eligible for the gifted program. All students are now given the OLSAT throughout their education. The score that is given is the percentile of students of the same age. To qualify for the gifted program students must score in the ninety-fifth percentile. With so many students with missing scores, I made the decision to leave out this information.

I have also included data on the students' lunch status. There are three categories included: free, reduced, and standard. Qualification for free or reduced status depends on the family's income and size. Students may qualify for free or reduced lunch if their income falls within limits of a federal income chart. For example, according to the Missouri Department of Elementary and Secondary Education in 2019 for a family of five to qualify for free meals, there maximum household income may not exceed $38,246 annually. To qualify for reduced lunch prices the maximum annual gross income must be less than $54,427 for the household. Those who do not qualify for the free or reduced program are classified as standard. The state looks at the percentage of students who qualify for free or reduced lunches. According to the Missouri Department of Elementary and Secondary Education website the percentage of students who qualify for free or reduced lunches reflects poverty within the district. The state average for students who qualified for free or reduced lunches was 51.2% in 2017 and 50.7% in 2018.

The final category that is included within the data is the withdrawal status. I have labeled the students that are still in school as "IS." Students that have graduated are classified as "G," this includes students who graduated under normal circumstances, students who have graduated early, and students who graduated with alternate standards. The last category would include special education students who graduated based on their IEP goals. An IEP is an individualized education plan. Students who transferred to another school whether it is a private or public school are classified as "T." Students who have left the school system and are now homeschooled are classified as "THS." Last, the students who have dropped out are classified as "D."

SAS Enterprise Miner Process

A decision tree is an appropriate method with the type of data collected because it is able to handle both categorical and numeric information. To run the data that was collected and create the decision trees the program SAS Enterprise Miner was used. This program is able to perform various versions of decision trees including the CART and CHAID methods. I decided to use the CART method.

The first step in growing the tree was to preprocess the data (Ville & Neville, 2013). This involved looking at the data and its categories. Before importing the data into the SAS program, I decided to combine some of the categories. Originally the EOC and MAP scores were categorized as advanced, proficient, basic, and below basic. I then decided to combine the advanced and proficient scores into one category and basic and below basic together in a separate category. In the education field, this is a natural split. As educators, we are concerned with the number of students who earn proficient or advanced versus those who score basic or below basic. Then due to many missing scores, I decided to look at the proficiency percentage instead. This follows how the state of Missouri determines scores for schools. The state gives schools a score based on their proficiency percentage on certain tests. The state determines proficiency as those students who score proficient or advanced on the EOC or MAP. I also made the decision to combine lunch status data. I combined those who qualify for free or reduced lunch into one category. Again, I chose to do this based on how the state of Missouri looks at the school data.

After preprocessing the data, the next step was to set the input and target for the model (Ville & Neville, 2013). After combining the scores, I found percent proficiency in several categories: elementary (3-5), junior high (6-8), high school (9-12), English language arts, mathematics, science, and social sciences. A proficiency of 80% in the high school tests represented that the student earned proficient or advanced on 80% of the tests. I dropped out the social sciences at this point since some of the student would have only one test. At this point of time the government test is the only test that all of our students are required to take.

The percent proficiency and lunch status were set as input values. The percent proficiency scores were set as an interval level of measurement and lunch status classifications were set as binary. I set the target variable as withdrawal status. The data that I received included students who graduate, dropped out, transferred to public or private schools, transferred to home school, and those who are still in school. My goal was to look at the students who dropped out, so I made the decision to only look at students who graduated or dropped out. I took out those who were still in school, and because we do not know the status of those who transferred, I also took out these students. Therefore, the target variable is also binary. The input variables are the independent variables and the target variable is the dependent variable. Target variables are called dependent variables because they depend on the input or are a function of the input.

The third step is to select the growth parameters of the decision tree (Ville & Neville, 2013). The algorithm used to create the decision tree will pick the best input to form a split. Decisions concerning missing values were also determined at this time. SAS Enterprise Miner allows missing values to be included, put into their own category, distributed proportionally to the nodes, or use surrogates in the place of missing values. The number of surrogates that are allowed can also be controlled by Enterprise Miner. Other aspects that can be controlled by SAS include the size of nodes, the depth of the tree, and inputs can be constrained to single usage. I set missing values to be placed in the largest category and the maximum number of surrogates as five.

The next step is to cluster and process each branch (Ville & Neville, 2013). In this step the maximum number of branches may be set. The maximum number of branches was set to two so that the tree is a binary tree. Clustering algorithms that be used include variance reduction, entropy, Gini, and tests of significance. I chose to use the Gini impurity. The Gini impurity calculates node impurity and the split that is used is the one that reduces the node impurity the most.

The fifth step is to select the candidate decision tree branches (Ville & Neville, 2013). The CART method will grow the tree and then prune by cutting the branches that do not perform well. This is done using a validation approach. According to Ville and Neville (2013), the branches in the CART method can be selected by the number of leaves, best assessment value, most leaves, Gini, or variance reduction. Inputs are nominal or interval for CART. The splitting criteria will be variance reduction for interval targets or Gini for nominal or binary targets. The Gini method was used since the target variable in this case was binary. When creating splits, observations with a missing value in the splitting variable are omitted (Ville & Neville, 2013). Then surrogate splits are created to assign observations to a branch. However, if the main and surrogate variable is prevented by a missing variable to be used, then an observation is assigned the largest branch (Ville & Neville, 2013). In the settings for my data, those values were set for five surrogate splits and missing values to be assigned to the largest branch.

The CART tree is grown to overfit the data and then it is pruned. The best tree is found by assessment measures based on accuracy. Accuracy is based on the training sample, validation set, and a cross-validation approach. Retrospective pruning determines the best subtree using predictive accuracy. "This method is a kind of Ocean's razor, meaning that the subtree with the highest accuracy and fewest leaves is chosen over any other subtree that has a similar predicative accuracy" (Ville & Neville, 2013, p. 93).

The sixth and final step in the process is to complete the form and content of the final decision tree (Ville & Neville, 2013). A decision tree is recursively built. The first level of a tree is formed, and the nodes are then candidates for splitting. Making sure to stop the tree so that it does not overfit the data is important. When a tree overfits the data, the tree is not stable, and predictions based on the tree will not be valid, and also leads to a tree that may not be reproducible. The CART method uses validation tests to prune the trees, stop tree growth, and form the optimal tree. The CART method fully grows the tree, to a point where it is overfit, and then prunes because those who first used this method argued that "the right thresholds for stopping tree growth are not knowable in advance" (Ville & Neville, 2013, p. 95).

When assessing the tree, the CART method, picks the best subtree through the pruning process. The variance reduction approach is one method that the CART method uses to form the branches of the tree. Branches are tested against a validation sample to determine if it should be used in the decision tree. If the accuracy is high enough, then it is used. SAS Enterprise Miner picks the simplest model with the highest validation assessment ("*Applied Analytics*," 2015). One assessment measure that is available to use in SAS Enterprise Miner is average square error. Which was found by:

$$ASE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

where for the i^{th} case, y_i is the actual target value, and $\hat{y}_i$ is the predicted target value. SAS Enterprise Miner will search for the tree that has a lower average square error.

Data Results

After looking at my initial data, I made decision to organize the data differently than what was originally planned. Initially, I was going to look at their performance on the initial test, based on whether they scored below basic, basic, proficient, or advanced on each test. A portion of the results are shown in Table 1.

Table 1. Initial Data

Student	ELA 3	MA 3	ELA 4	MA 4	ELA 5	MA 5	SC 5
1			Basic	Basic	Basic	Proficient	Proficient
2	Advanced	Advanced	Advanced	Proficient	Advanced	Advanced	Advanced
3	Basic	Basic	Basic	Basic	Proficient	Basic	Basic

I then decided that it would be easier to deal with binary targets and grouped those who scored below basic and basic together as well as those who scored proficient or advanced. When schools analyze data, this is a natural grouping that is made. This is shown in Table 2. Not all tests are shown in this portion of the table. Student 1 had a 0 for the 4th grade MAP, which means that he or she earned basic or below basic on that test. This student had a 1 on the 5th grade MAP test representing a score of proficient or advanced on the test. I wanted to use this data to help pinpoint specific tests that may be of concern for those students who drop out. When looking out this data, I did reject several of the variables. American History, Geometry, Algebra 2, and English 1 were left out of the training. These scores were left out because several of them are no longer required to take. With changes at the state level not all students take these tests. The Geometry and Algebra 2 tests are also not taken by all of our students. The blank space by student 1 in Table 2 represents a missing score. This score may be missing because the student did not take the test that year or we did not receive their score from the school that they transferred from. Many students have missing scores because they have transferred from schools that are out of state which do not give the MAP or EOC tests.

Table 2. Binary Data

Student	CA 3	MA 3	CA 4	MA 4	CA 5	MA 5	SC 5
1			0	0	0	1	1
2	1	1	1	1	1	1	1
3	0	0	0	0	1	0	0

Due to the numbers of missing scores, I made the decision to look at percent proficiency. Proficiencies were found for the three grade levels of school: elementary, junior high (or middle school), and high school. Proficiencies were also found for the content areas: English language arts, mathematics, science, and social sciences. Finally, the overall proficiency was also found. Table 3 displays the percent proficiencies with all the categories that were created for the same three students as in the previous tables.

Percent proficiency represents the percentage of the tests that the individual scored either a proficient or advanced on the state test (MAP or EOC). For example, student 1 scored proficient or advanced on 40% of their elementary MAP tests and did not score proficient or advanced on any of their junior high MAP tests. The 40% represents that this student scored proficient or advanced on 40% of the tests that the took at the elementary level.

Table 3. Percent Proficiency

Student	Elem.	JH	HS	ELA	MA	Overall	Science	SS
1	40%	0%		0%	20%	17%	50%	
2	100%	100%	100%	100%	100%	100%	100%	100%
3	14%	57%	100%	57%	33%	44%	33%	

When looking at percent proficiency instead of the binary data, the number of missing values drastically reduced. However, as can be seen in Table 3, there are still missing values. Student 1 had a missing value for the high school category. This means that this student had not taken any EOC's at this time. Therefore, missing values will still be a factor in the data analysis.

My initial data from the two school years contained 1325 students. When I decided to make my target variable the graduation status, I decided to remove students who had transferred and those who were still in school. This left me with a much smaller data set of only 417 students and only twenty of these students dropped out. Eight input variables were used and one target variable. The input variables include lunch status and percent proficiency in elementary, junior high, high school, English language arts, mathematics, science, and their overall proficiency percent.

Before running the data and creating the decision tree, I explored the data that was available. Looking at the data in Figure 2, we can see how the overall percent proficiency align. For the students who graduated or who dropped out, the majority of these students had a percent proficiency between 90% and 100%.

Looking at all students who were enrolled during the time period selected the distribution only changes slightly. Figure 3 shows this new distribution. The most significant difference is the frequency for those in the category of those who scored below ten percent proficiency.

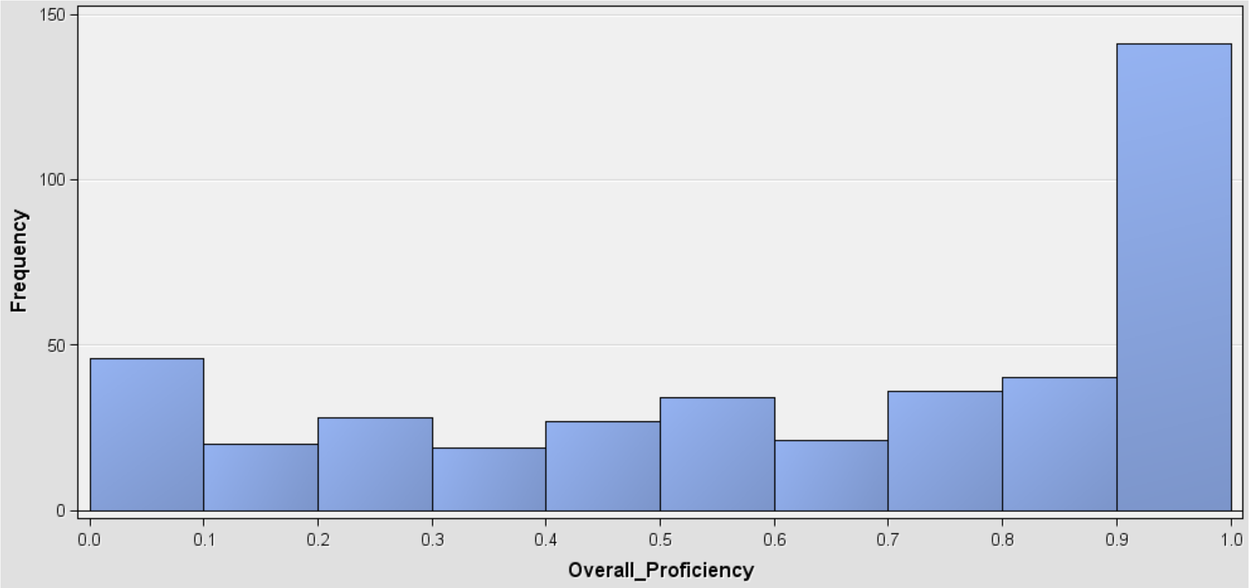

Figure 2. Overall Proficiency of Graduates and Dropouts

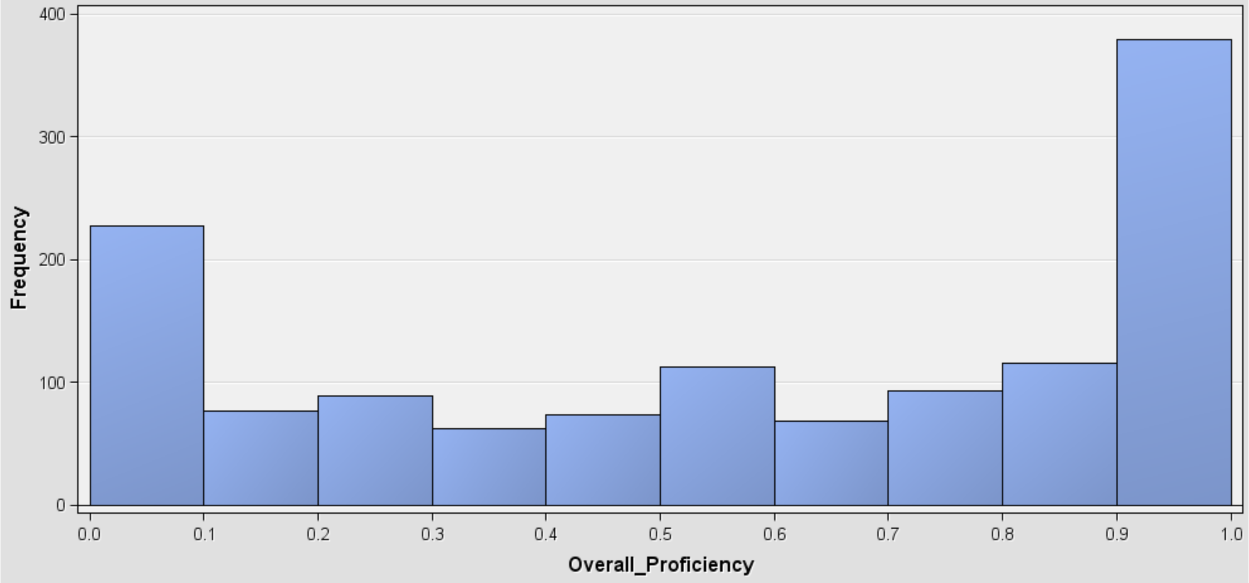

Figure 3. Overall Proficiency of All Students

Students are tested in the four core subject areas. However, mathematics and English language arts are testing every year starting in third grade through eighth grade. They are then tested at least once in these content areas throughout high school. It is interested that the distribution of the scores for mathematics and communication arts are very similar. In Figure 4, a bi-modal graph for each of these content areas is displayed. On the left, we have a histogram of the percent proficiency for the students in mathematics (MA Proficiency) and on the right English language arts (CA Proficiency). Each have modes at the extremes, with the majority of students scoring above 90% proficient in the content area tests. They both have a spike in the 50%-60% range as well.

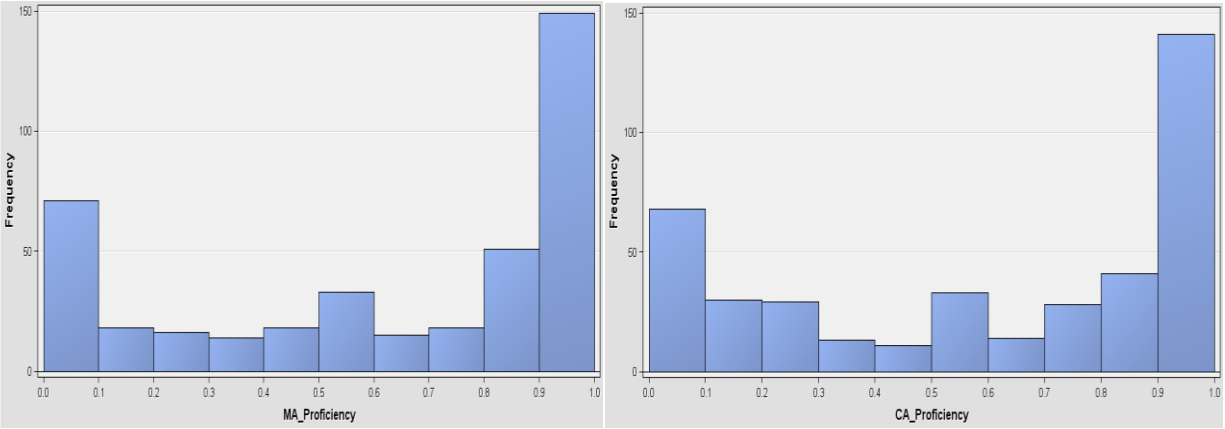

Figure 4. Mathematics and English Language Arts (CA) Proficiency

When looking at the variable worth with all students' proficiency on the high school EOC's has the highest worth. When only looking at students who graduated or dropped out, the mathematics proficiency scores has the highest worth, but with a much lower value. The low variable worth values help explain the small nature of the resulting tree. Mathematics, which has the highest worth was still less than a 0.01 as shown in Figure 5. Lunch status had the lowest worth which was surprising to me.

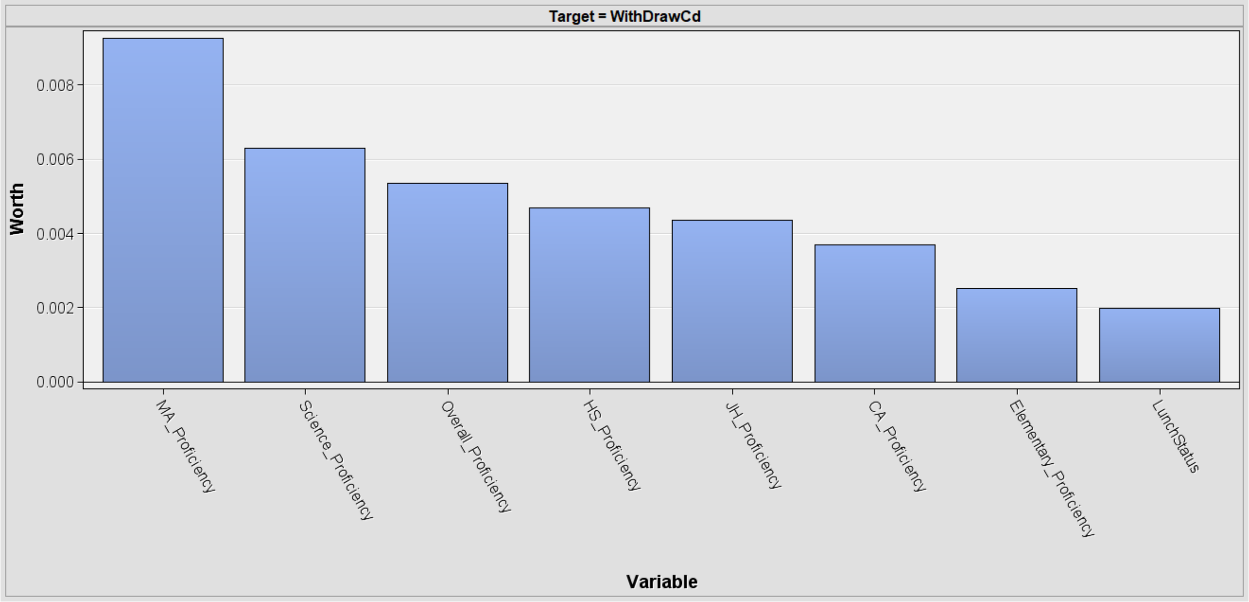

Figure 5. Variable Worth for Decision Tree with Percent Proficiency Input

After analyzing the data and choosing the settings to run a CART method decision tree, the resulting tree was formed is shown in Figure 6. The student's overall proficiency on the state assessments and the math proficiency were the only elements that appeared in the tree. With the training data, if a student's overall proficiency was less than 2.5%, then 73% were graduates and 27% were dropouts. This means that 30% of all the dropouts appeared in this node (2) with the training data.

When students earned greater than 2.5% overall on their EOC's but scored less than 59% on their mathematics EOC's, the result was 93% graduates and 7% dropouts. This resulted in 60% of the dropouts appearing in this node (4) when using the training data.

Overall proficiency appeared again within the tree. Surprisingly, scoring greater than 97.73% proficiency overall on their EOC's did not result in 100% graduates, there was a drop-out who appeared in this node (5).

When looking at the validation set. The eighth node for those who scored between 2.5%

and 97.73% proficiency overall and had greater than 58.57% proficiency in their mathematics assessments remained purely graduates. Node 2 remained the same with 30% of the dropouts overall. Node 4 dropped from 60% of the dropouts to 50%. Since the data set contained a very limited number of students who dropped out, this is not a significant change. Therefore, the validation set showed that the model was consistent.

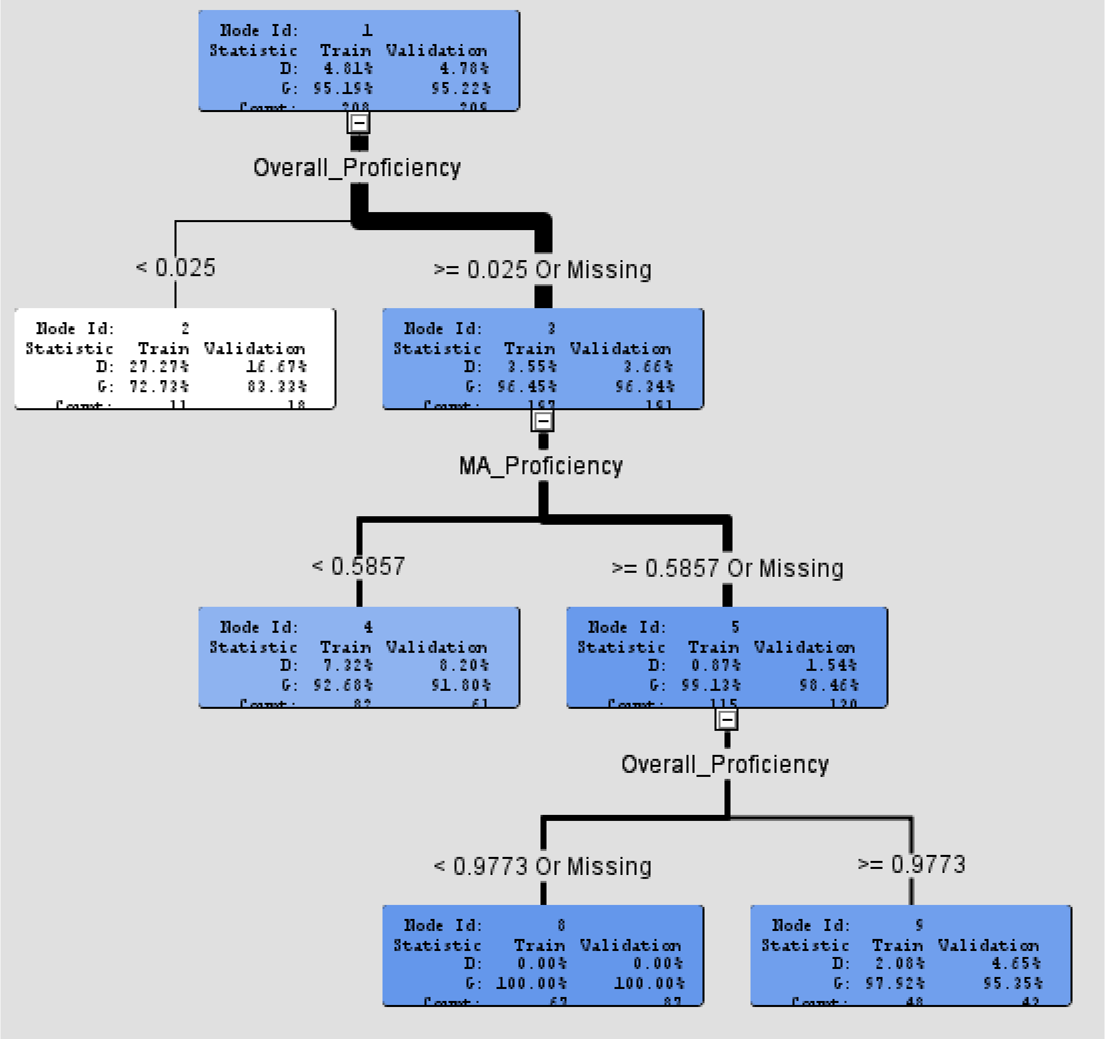

Figure 6. Decision Tree with Percent Proficiency Input

Science, which had the second highest worth, did not appear in this tree. However, mathematics and overall proficiency were in the top three. Science may have appeared in the overgrown tree but did not appear in the final pruned tree. I was surprised the lunch status had the lowest variable worth, so I looked at the data. The entire student population had 42.9% of the students qualifying for free or reduced lunch. This shifted to 38.3% when we only looked at those students who graduated or dropped out of school. Having a lower group of free or reduced lunch students may have been the reason that lunch status did not show up as a factor in the tree.

I wanted to see if there was maybe an individual test that had a factor with students who graduated or dropped-out, I decided to re-run the data with the scores set as those who were basic or below basic (0) and those who scored proficient or advanced (1). The target variable remained the same, but we now look at individual tests rather than a student's percent proficient or advanced in broader categories. The new tree appears in Figure 7. This tree was also created using the CART method. While pre-analyzing the data, the government and biology tests had the highest worth. However, neither appeared in the tree. The fourth-grade mathematics test, Algebra 1, lunch status, and English 2 were the only categories that remained in the final tree. If a student scored basic or below basic on the fourth-grade mathematics MAP test, then 92% of these students graduated and 8% dropped-out. The 8% represents 70% of the dropouts appeared in this node (2). Whenever free or reduced lunch status appeared in the tree, the students who dropped-out appeared in this category. But in the standard lunch branches, there are no students who dropped out. These results are consistent with the validation data set as well. Again, this shows a consistency with the model.

It is not surprising that two math tests appear in this tree. When looking at percentage proficiency it was mathematics that showed up in that tree as well. The lunch status did make an appearance in this tree, showing that students who dropped-out were in the free or reduced

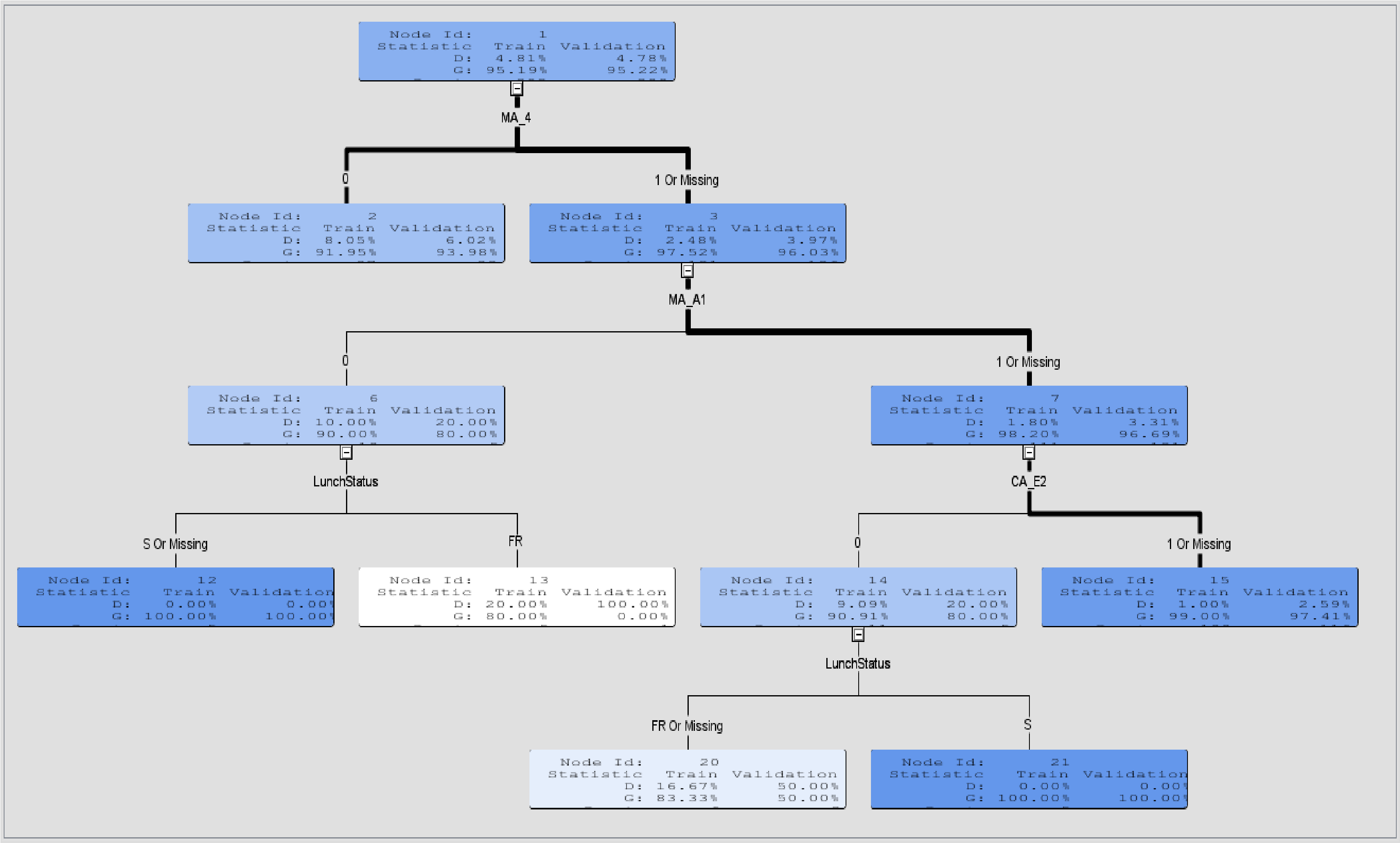

Figure 7. Decision Tree with Binary Input

category and not the standard lunch. Standard lunch means that students pay for full lunch prices, they do not qualify for a free or a reduced lunch.

CHAPTER 4: LOGISTIC REGRESSION ANALYSIS OF GRADUATION DATA

To help analyze the data with secondary methods, logistic regression could be used. Multiple-regression techniques would not work because with some of the data that has been collected there are only two levels such as success and failure. Logistic regression is able to help in this situation. Bock states that logistic regression may be better when trying to make conclusions based on what causes what but may be a poor choice when trying to describe the data or make a prediction (2018).

Using a multiple regression method, values could be obtained that may be impractical. The logistic regression uses transformations to force the equation to result in predicted values that between zero and one. The equation predicts the natural log of the odds for a person being in one category or another (Cody & Smith, 1997).

When dealing with data in this situation are success was graduation and are failure were those students who dropped out. When one was first trying to find a model to fit this situation a linear model would be the first thought. The dependent variable would be the probability of success, which is between zero and one inclusive, however, in a linear equation this value could take on values that are negative or greater than one. An exponential equation would allow the values to become non-negative, but they could still be greater than one. Therefore, the following logistic model is used:

$$p = \frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}}$$

where x is the independent variable (Pagano & Gauvreau, 2000). If the probability of an event occurring is p, then the odds in favor of that event are $\frac{p}{1-p}$. Solving this equation in terms of x results in $e^{\alpha+\beta x}$. Then when taking the natural log of each side, the natural log of the odds is equal to a linear regression model. Therefore, logistic regression assumes that the relationship between the natural log of the odds and x is linear, rather than assuming the relationship between p and x are linear (Pagano & Gauvreau, 2000).

Proficiency Results

Looking at the data and using SAS Enterprise Miner now with logistic regression instead of decision trees. First the data was run using the percent proficiencies. No selection process was chosen; therefore, all inputs were considered. The target was graduation status and the model is predicting the probability of graduation. When using regression with SAS, again a process is needed to deal with missing values. Therefore, before running the regression, a process of imputing was used. When imputing, a synthetics value is used for the missing value. The default for the SAS program was used in this case. Therefore, for the interval inputs the missing values were replaced with the mean of the non-missing values. For the categorical inputs, the most frequent category replaced any missing values ("*Applied Analytics*," 2015). Elementary proficiency had forty-three missing values in the training set which was by far the greatest. Junior high proficiency had only sixteen missing values in the training set and the rest were in the single digits. The elementary proficiency average was 54.34%, therefore, each missing value was replaced with this value during the impute process. Each of the other missing values were similarly replaced with their average values.

Table 4 shows the results for the Maximum Likelihood Estimators. Looking at the data that resulted, junior high proficiency and math proficiency are the two parameters that have a chi-squared value less than five percent. Lunch status, high school proficiency, and elementary proficiency also had p-values that can be considered significant. It is interesting that elementary, high school, and junior high proficiencies have negative estimates. A negative estimate in this case would result in a decrease in probability of graduating with an increase in proficiency. This negative estimate could be due to multicollinearity. Multicollinearity in a model occurs when parameters are correlated to the response variable and one other ("Enough is Enough," 2013). It also may be due to the fact that the log (odds of graduating) is not linearly dependent with these independent variables. The negative value for lunch status does make since though. If a student qualifies for free or reduced lunch a one is imputed into the equation which would lower the probability of graduation.

Table 4. Proficiency: Analysis of Maximum Likelihood Estimates

Parameter	Estimate	Pr > ChiSq
Intercept	3.3253	0.0002
ELA Proficiency	0.7109	0.7817
Elementary Proficiency	-3.5493	0.1108
HS Proficiency	-3.1847	0.1019
JH Proficiency	-4.5874	0.0402
MA Proficiency	4.6833	0.0433
Overall Proficiency	3.7154	0.5510
Science Proficiency	2.5297	0.2254
Lunch Status (FR)	-0.6609	0.0940

Looking at the odds ratio estimates, shown in Table 5, the math proficiency has a much greater impact on the odds of graduation rate than any other parameter. An increase in each unit results in an increase in the odds by the point estimate.

Table 5. Proficiency: Odds Ratio Estimates

Parameter	Point Estimate
ELA Proficiency	2.036
Elementary Proficiency	0.029
HS Proficiency	0.041
JH Proficiency	0.010
MA Proficiency	108.130
Overall Proficiency	41.077
Science Proficiency	12.550
Lunch Status (FR)	0.267

The data was then run again, but this time using a backward selection. SAS is able to make selection using a forward, backward, or stepwise approach. According to "*Applied Analytics,"* the backwards method creates models that decrease in complexity (2015). A model begins with all available inputs, then inputs are removed from the models. The input with the highest *p*-values is removed at each step. When all remaining inputs have *p*-values that are less then a predetermined cutoff, which was 15% in this case, the sequence is completed. The results using the backwards method are shown in Table 6.

Math proficiency, junior high proficiency, and lunch status are the only parameters in the model. The negative estimate value for the junior high proficiency is again intertesting. As discussed earlier, this could be due to multicollinearity. There will be overlap with math and junior high proficiencies and this may cause part of the issue as well. Junior high and math have a correlation coefficient of -0.7629 in this model, showing there may be a negative linear relationship between the two parameters. Also as discussed earlier, the negative estimate for lunch status is expected.

Table 6. Proficiency: Analysis of MLE with Imputed Values (Backwards)

Parameter	Estimate	Pr > ChiSq
Intercept	2.5924	<0.0001
JH Proficiency	-2.4120	0.0918
Math Proficiency	3.8371	0.0069
Lunch Status (FR)	-0.6184	0.0916

Using the values from this table, the equation for this model would become:

$$\log(odds\ of\ graduation) = 2.5924 - 2.4120 \times JH + 3.8371 \times MA - 0.6184 \times LS$$

Exponentiating each side of the equation will result in finding the odds of graduating. Odds is equal to $\frac{p}{1-p}$. Therefore, by solving for probability, results in probability of graduating is equal to $\frac{odds\ of\ graduating}{1+odds\ of\ graduating}$. The intercept would represent the log(odds of graduating) for a student who has a zero proficiency in each test and a student who does not qualify for free or reduced lunch. The probability of graduating for a student who fits this scenario is then 93.03%. A student who does not qualify for free or reduced lunch and is 80% proficiency in math and junior high MAP test will have a 97.66% chance of graduating. A student who scores 40% on math, but still qualifies for standard lunch prices and is 80% proficient in junior high will have a 90%

chance of graduating.

In deciding whether a student's lunch status has an effect on their probability of graduation is a concern of educators. Figure 8 was created using SAS Studio and looks at how this parameter may affect the probability of a student graduating. The graph was created using the imputed values. By looking at this graph, there is an obvious difference for a student who qualifies for standard lunch prices and those who qualify for free and reduced lunches and some intervention should be made to help these students.

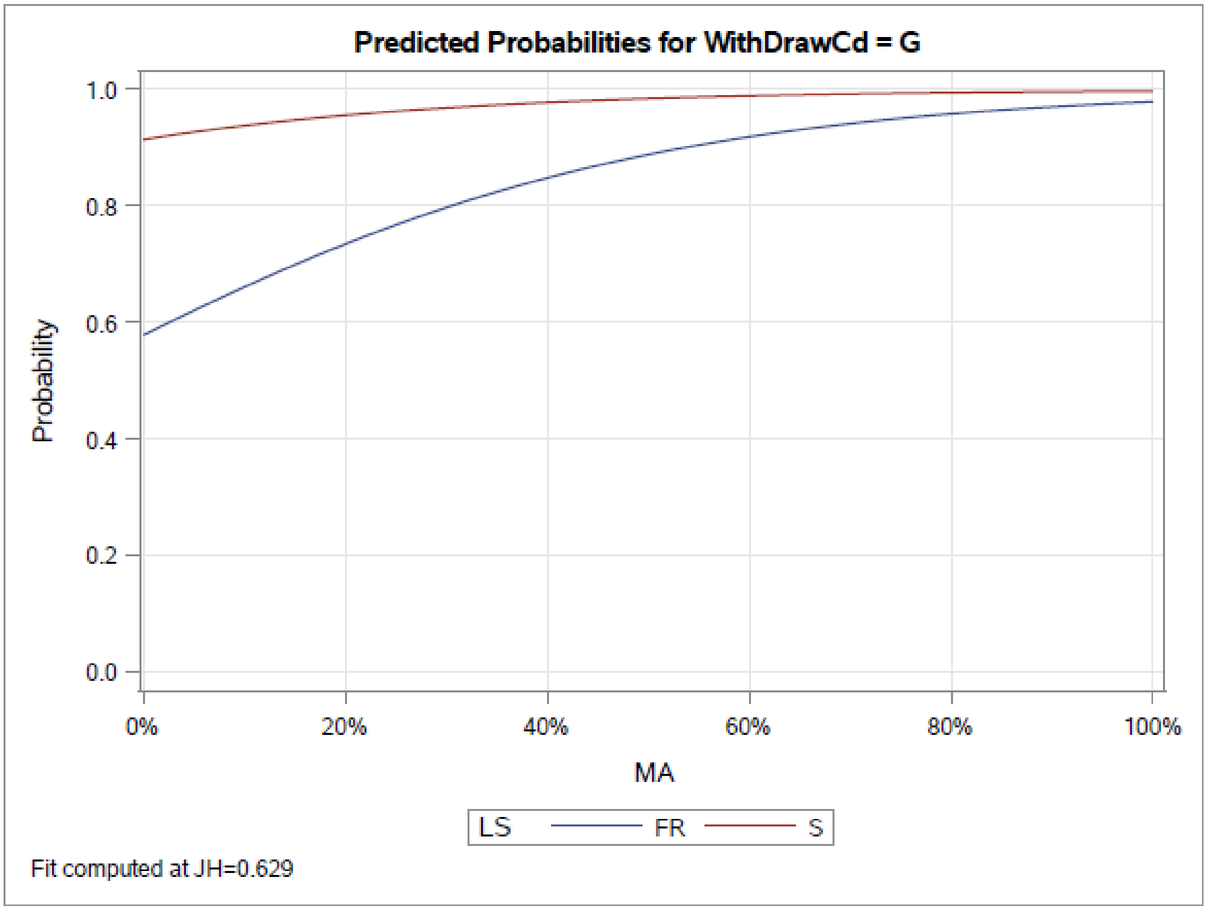

Figure 8. Predicted Probabilities

Looking at this data again, without imputing values for the missing values provides slightly different results. When values are not imputted, the students who had missing values were not considered in the model. Therefore, the size of the data was lowered. The imputed model had 208 values in the training set and this model dropped to 164 values in the training set. The backwards method with a *p*-value of 15% was used once again. Table 7 shows the results. All of the same variables remain from before, however, high school proficiency was also added to the model.

Table 7. Proficiency: Analysis of MLE without Imputed Values (Backwards)

Parameter	Estimate	Pr > ChiSq
Intercept	4.9884	0.0004
HS Proficiency	-3.8446	0.0532
JH Proficiency	-4.3135	0.0735
Lunch Status (FR)	-1.4668	0.0222
MA Proficiency	8.1838	0.0048

Again, junior proficiency has a negative estimate as well as high school proficiency. The math proficiency has an increased estimate from the imputted values model. This again could be due to multicollinearity. The negatve estimate for lunch status once again is expected in this situation. One would expect the probability of graduation to lower if a student qualifies for free or reduced lunch.

Whether imputing is used or missing values are excluded, the models are similar to one other. High school proficiency is the different parameter within the two methods. Junior high proficiency still maintains a negative estimate in the new model. What changes in the two models are the magnitudes of the estimates. Math proficency has the lowest p-value in each model. This may show that imputing values did not drastically alter the results of the model.

Creating a model with math proficiency and lunch status may be worth looking at. With all variable rejected except for mathematics and lunch status, each value had a p-value less than 15%. Mathematics had a p-value of 3.64% and lunch status had a p-value of 13.27%. Using the estimates the equation for this model is:

$$\log(odds\ of\ graduation) = 2.1361 + 1.9243 \times MA - 0.5380 \times LS$$

The model in this case is logical. As the math proficiency increases, the odds of graduating and the probability of graduating should increase. Also, if a student qualifies for free or reduced lunch it would be logical that their probability of graduation would decrease. A student who qualifies for standard lunch and is 80% proficient on their state math assessments would have a 97.53% probability of graduating. While a student who qualifies for standard lunch but is only 40% proficient would have a 94.81% probability of graduating.

Binary Results

Looking at the data in terms of zeros and ones instead of the overall proficiency gives another look at the data. As with the decision tree model, a zero represent a student who scored basic or below basic on the test and a one represents a student who scored advanced or proficient. When looking at the data in this manner, there are many more missing values than when looking at the data with proficiency. The third grade English language arts and mathematics each had fifty-five missing values for the training set. There were several other categories also missing forty to fifty values in the training set. As a result of all of the missing values, the imputing process was used again. Since all of the data was categorical the most frequent value was used to replace the missing values. For every data set except for the third grade English test they were replaced with ones. The third grade missing English scores were replaced with zeros.

When initially looking at the data, Table 8 shows the data that resulted. The students receiving below basic or basic would be inputted in the equation as a one while a student receiving proficient or advanced would be placed in as a zero. Similarly a student who qualifies as free or reduced lunch will be placed into the equation as one and a student who qualifies for standard lunch prices will be inputted as a zero. Therefore, for the parameters a negative estimate would be expected for the parameters. The intercept would be the only value that logically should be positive. Postive estimates may be due to multicollinearity or log (odds of graduating) is not linear dependent with the variables.

Next, the backwards model selection was used for this data as well. Table 9 shows the results from this method. Fifteen percent was once again used as the cut-off p-value. Using this method, 6th grade English language arts, English 2, 4th grade math, 6th grade math, and lunch status are the parameters that appear in the model.

Using these parameters and their estimates, the equation for this model is:

$$\begin{aligned}\log(odds\ of\ graduation) \\ &= 3.6575 + 2.5906 \times E6 - 1.1008 \times E2 - 0.8762 \times M4 - 1.0882 \times M6 \\ &- 0.7901 \times LS\end{aligned}$$

For a student who scores proficient or advanced on each test and qualifies for standard lunch has a 97.49% chance of graduating. While a student who scores basic or below basic and qualifies for free or reduced lunch will only have 91.63% probability of graduating.

Table 8. Binary Data: Analysis of Maximum Likelihood Estimates

Parameter	Estimate	Pr > ChiSq
Intercept	4.4132	<0.0001
ELA 3 (0)	-0.1312	0.7978
ELA 4 (0)	-0.3911	0.6102
ELA 5 (0)	-0.6036	0.4094
ELA 6 (0)	3.6356	0.0048
ELA 7 (0)	0.2290	0.7536
ELA 8 (0)	-0.3617	0.6287
English 2 (0)	-1.2410	0.0458
Math 3 (0)	0.3370	0.6001
Math 4 (0)	-0.6840	0.3903
Math 5 (0)	0.1405	0.8515
Math 6 (0)	-1.7921	0.0178
Math 7 (0)	-0.2678	0.6917
Math 8 (0)	0.1965	0.7399
Algebra 1 (0)	0.5038	0.4368
Science 5 (0)	0.7262	0.2848
Science 8 (0)	-0.4255	0.4765
Biology 1 (0)	-0.0612	0.9235
Government (0)	0.0509	0.9401
Lunch Status (FR)	-0.6222	0.2230

Table 9. Binary Data: Analysis of MLE (Backwards)

Parameter	Estimate	Pr > ChiSq
Intercept	3.6575	<0.0001
ELA 6 (0)	2.5906	0.0011
English 2 (0)	-1.1008	0.0320
MA 4 (0)	-0.8762	0.0732
MA 6 (0)	-1.0882	0.0456
Lunch Status (FR)	-0.7901	0.0709

There were too many missing values in this data set to run the test without imputing for the missing values. Without imputing there was only a data set of twenty values and no parameters were significant.

These results are interesting and worthwhile for the district to investigate. However, the amount of missing values and how those values are imputed may greatly affect the results. Therefore, before too much emphasis is put into these areas, it may warrant further investigation.

CHAPTER 5: OTHER METHODS OF PATTERN CLASSIFICATION

A set of measurements will be represented as a pattern vector $\boldsymbol{x}$. The pattern will be assigned to one of C possible classes $\omega_i, i = 1, \dots, C$. Then a decision rule will be used to divide the measurements into C regions $\Omega_i, i = 1, \dots, C$. According to Duda et. al. (2001), the true task in making a decision rule is to minimize the cost of our decisions. An observation vector in Ω_i is assumed to belong to class ω_i. Decision boundaries are the boundaries between the regions Ω_i. Misclassifications commonly occur in the regions near the boundaries. Some patterns may be rejected, or a decision made be withheld until more information is available. This produces a reject option which produces $C + 1$ outcomes in a problem. The reject option will be denoted ω_0 (Webb, 2002).

Bayes Decision Rule for Minimum Error

The Bayes decision rule for minimum error is used when the *a priori* probabilities are assumed to be known. The goal is to minimize the probability of making an error. The class probability distribution is known but no other information is known about the object. In this case, an object will be assigned to class ω_j if

$$p(\omega_j) > p(\omega_k) \;\; k = 1, \dots, C; k \neq j$$

The observation vector $\boldsymbol{x}$ also needs to be assigned to one of the C classes. The decision rule based on probabilities is to assign the vector to class ω_j if the probability of the class ω_j given the observation $\boldsymbol{x}$ is greatest over all classes $\omega_1, \dots, \omega_C$. Therefore, if

$$p(\omega_j|\boldsymbol{x}) > p(\omega_k|\boldsymbol{x}), \;\; k = 1, \dots, C; k \neq j \qquad 3$$

$\boldsymbol{x}$ will be assigned to class ω_j.

Using Bayes decision rule decision errors are unavoidable. Therefore, since researchers want to minimize the classification error. Bayes' theorem allows them to express *a posteriori* probabilities $p(\omega_j|\boldsymbol{x})$ in terms of the *a priori* probabilities and class-conditional density functions

$$p(\omega_i|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\omega_i)p(\omega_i)}{p(\boldsymbol{x})}$$

thus, with substitution the decision rule (equation 3) can be rewritten as assign $\boldsymbol{x}$ to class ω_j if

$$p(\boldsymbol{x}|\omega_j)p(\omega_j) > p(\boldsymbol{x}|\omega_k)p(\omega_k),\ k = 1, \dots, C; k \neq j \qquad 4$$

This is Bayes' rule for minimum error.

When manipulating the decision rule (equation 4) specifically for two cases it may be written as following

$$\ell(\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\omega_1)}{p(\boldsymbol{x}|\omega_2)} > \frac{p(\omega_2)}{p(\omega_1)} \text{ implies } \boldsymbol{x} \in \text{class } \omega_1$$

which is the likelihood ratio test.

The probability of making an error is

$$p(\text{error}) = \sum_{i=1}^{C} p(\text{error}|\omega_i)\, p(\omega_i)$$

where

$$p(\text{error}|\omega_i) = 1 - \int_{\Omega_i} p(\boldsymbol{x}|\omega_i)\, d\boldsymbol{x}$$

This is the integral of the class cumulative distribution function (cdf) over the region of space outside Ω_i. With the substitution the probability of making an error, or misclassifying a pattern, the probability of making error can be written as

$$p(\text{error}) = \sum_{i=1}^{C} \left(1 - \int_{\Omega_i} p(\boldsymbol{x}|\omega_i)\, d\boldsymbol{x}\right) p(\omega_i)$$

$$= 1 - \sum_{i=1}^{C} p(\omega_i) \int_{\Omega_i} p(\boldsymbol{x}|\omega_i)\, d\boldsymbol{x}$$

Minimizing the probability of making an error is equivalent to maximizing

$$\sum_{i=1}^{C} p(\omega_i) \int_{\Omega_i} p(\boldsymbol{x}|\omega_i)\, d\boldsymbol{x}$$

The regions Ω_i need to be chosen to make the previous integral a maximum. This may be achieved by selecting Ω_i to be the region in which $p(\omega_i)p(\boldsymbol{x}|\omega_i)$ is the largest of all the classes. Then the probability of correct classification, c, is

$$c = \int \max_i p(\omega_i)p(\boldsymbol{x}|\omega_i)d\boldsymbol{x}$$

and the Bayes decision rule error is

$$e_B = 1 - c$$

Bayes for Minimum Risk

Previously, a decision rule was selected to minimize the probability of making an error. Now the researcher considers a rule that will minimize expected loss or risk. This step is important since the cost of misclassifications can be severe, and some costs of misclassification are costlier than others.

A loss matrix $\boldsymbol{\Lambda}$ with components

$$\lambda_{ji} = \text{cost of assigning a pattern } \boldsymbol{x} \text{ to } \omega_i \text{ when } \boldsymbol{x} \in \omega_j$$

where λ may be difficult to assign. A loss matrix may be a combination of many factors such as time, money, life, etc. The conditional risk of assigning a pattern to ω_i may be defined as

$$l^i(\boldsymbol{x}) = \sum_{j=1}^{C} \lambda_{ji}\, p\big(\omega_j \big| \boldsymbol{x}\big)$$

The average risk over region Ω_i is then defined as

$$r^i = \int_{\Omega_i} l^i(\boldsymbol{x})\, p(\boldsymbol{x}) d\boldsymbol{x}$$

$$= \int_{\Omega_i} \sum_{j=1}^{C} \lambda_{ji}\, p\big(\omega_j \big| \boldsymbol{x}\big) p(\boldsymbol{x}) d\boldsymbol{x}$$

And then the overall risk is

$$r = \sum_{i=1}^{C} r^i = \sum_{i=1}^{C} \int_{\Omega_i} \sum_{j=1}^{C} \lambda_{ji}\, p\big(\omega_j \big| \boldsymbol{x}\big) p(\boldsymbol{x}) d\boldsymbol{x}$$

Again, the expected risk should be minimized. This may be achieved by choosing regions Ω_i that if

$$l^i(\boldsymbol{x}) \le l^k(\boldsymbol{x}), \qquad k = 1, \dots, C$$

then $\boldsymbol{x} \in \Omega_i$. Bayes risk may be given by

$$r^* = \int_{\boldsymbol{x}} \min_{i=1,\dots,C} \sum_{j=1}^{C} \lambda_{ji}\, p\big(\omega_j \big| \boldsymbol{x}\big) p(\boldsymbol{x}) d\boldsymbol{x}$$

Discriminant Functions

While applying Bayesian decision rules knowledge of class-conditional density functions must be known. When using discriminant functions, assumptions will be made of the forms of these functions.

A classification rule is found using a discriminant function which is a function of the pattern $\boldsymbol{x}$. When talking about a two-class problem, the discriminant function can be used as follows

$$h(\boldsymbol{x}) > k \Rightarrow \boldsymbol{x} \in \omega_1$$

$$h(\boldsymbol{x}) < k \Rightarrow \boldsymbol{x} \in \omega_2$$

for constant k. The optimal function for the two-case class is

$$\mathrm{h}(\mathbf{x}) = \frac{\mathrm{p}(\mathbf{x}|\omega_1)}{\mathrm{p}(\mathbf{x}|\omega_2)}$$

while

$$k = \frac{p(\omega_2)}{p(\omega_1)}$$

In case with C classes, there are C discriminant functions $g_i(\boldsymbol{x})$ so that

$$g_i(\boldsymbol{x}) > g_j(\boldsymbol{x}) \Rightarrow \boldsymbol{x} \in \omega_1 \;\; j = 1, \dots, C; j \neq i$$

A pattern will therefore be assigned to the class with the largest discriminant. If the optimal discriminant function is defined as

$$g_i(\boldsymbol{x}) = \ \mathrm{p}(\mathbf{x}|\omega_\mathrm{i})\, p(\omega_i)$$

it will lead to the Bayes decision rule.

Choosing a discriminant function may depend on prior knowledge of the patterns or its functional form may have its parameters adjusted by training data. There have been many forms of discriminant functions used that will vary in their complexity. For example, linear discriminant functions, piecewise linear discriminant functions, and generalized linear discriminant functions are a few.

According to Webb (2002) a linear discriminant function will be in the form

$$g(\boldsymbol{x}) = \boldsymbol{w^T x} + w_0 = \sum_{i=1}^{p} w_i + w_0$$

where $\boldsymbol{x} = \left(x_{1,\dots,}x_p\right)^T$, $\boldsymbol{w}$ is the weight vector, and the threshold weight is w_0.

Again, according to Webb (2002), in a piecewise linear discriminant function the function for class ω_i is

$$g_i(\boldsymbol{x}) = \max_{j=1,\dots,n_i} g_i^j(\boldsymbol{x})$$

Where g_i^j is a supplementary discriminant function, which is still linear and defined by

$$g_i^j(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{p}_i^j - \tfrac{1}{2}\boldsymbol{p}_i^{j^T}\boldsymbol{p}_i^j \quad j = 1, \dots, n_i; i = 1, \dots, C$$

There are many methods of analyzing pattern recognition. With density estimation parametric and nonparametric methods may be used. Nonparametric methods including the histogram, nearest neighbor, kernel methods, and tree-based methods will be examined.

Density Estimation – Nonparametric

When researchers have knowledge of the class-conditional probability density function they can use the likelihood ratio test to assign a pattern to a class. If researchers can make assumptions of the density function, then parametric methods will need to be used to estimate the parameters to describe the densities. However, if researchers are unable to make those assumptions, nonparametric methods will be used for the density estimation.

Knowledge of the properties of density estimators is needed to go forward. An estimator is a function of the sample. The properties of a good estimator are unbiasedness, consistency, and efficiency. If $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n$ are independent and identically distributed (iid) random variables with continuous density $p(\boldsymbol{x})$**,** then

$$p(\boldsymbol{x}) \geq 0 \text{ and } \int_{\mathbb{R}^p} p(\boldsymbol{x})d\boldsymbol{x} = 1$$

Histogram Method. One method of nonparametric estimation is the histogram method. This is the oldest used density estimator. Under this method a set of samples is used to create a probability density. In one dimension, the real line is sectioned into a number of equal-sized cells. In the histogram, the origin will be x_0 with a bin width of h. This is an easy concept to use, but there are several problems with this basic approach. There is an exponential growth in the number of cells which means to estimate the density, a large amount of data is needed (Webb, 2002). When histograms are used in cluster analysis and nonparametric discriminant analysis there is an inefficient use of the data (Silverman, 1986). The discontinuity of the histograms will present difficulties when derivatives of estimates are needed (Silverman, 1986). Because of this and other issues, there are other approaches that will alleviate these issues.

In the case of univariate data, histograms are useful for density estimates. However, the choice of origins can influence the perceptions by those observing them. Using histograms with multivariate data presents even more difficulties. Histograms are useful for the presentation, but other methods should be considered to find the density estimate.

Data-adaptive histograms and independence assumption are two methods used to overcome the issues of the histogram method. In the data-adaptive histogram, descriptors such as location, shape, and size are allowed to adapt to the data (Webb, 2002). As the title implies, in independence assumption, we assume that the variables are independent so that $p(x)$ may be written as

$$p(x) = \prod_{i=1}^{p} p(x_i)$$

This model is known by several names including the *naïve Bayes.*

Nearest Neighbor. The nearest neighbor method is another simple method of density estimation. The approach for this method is to fix the probability that centers the cell at a point $\boldsymbol{x}$ and let it grow till it contains k samples then calculate the volume which centered on the point $\boldsymbol{x}$.

These samples are the nearest neighbors of $\boldsymbol{x}$. This is different from the histogram method which fixes the cell size and then determines the number of points within the cell.

The parameter k can provide difficulties for the nearest-neighbor method. When the value of k is too large the estimate will be too smooth, and details will be lost. While if the value of k is too small the estimate will be spiky (Webb, 2002). One detail worth observing is that the density estimate is not a density because it violates the fact the integral under the curve is one, instead it is infinite. However, according to Webb (2002), the estimator is asymptotically unbiased and consistent, the qualities of a good estimator, if

$$\lim_{n\to\infty} k(n) = \infty$$

and

$$\lim_{n\to\infty} \frac{k(n)}{n} = 0$$

The first condition assures that $\frac{k(n)}{n}$ will be a good estimate of the probability that a point will fall in the cell (Duda et al., 2001). Duda et al. states that these two conditions are necessary and sufficient for $p_n(\boldsymbol{x})$ to converge to $p(\boldsymbol{x})$ in probability at all points where $p(\boldsymbol{x})$ is continuous.

The density can then be defined as

$$p(\boldsymbol{x}) = \frac{k/n}{V}.$$

According to Duda et al. (2001), to estimate *a posteriori* probabilities, we assume that a cell of volume V has been placed around $\boldsymbol{x}$ and captured k samples. Of these k samples, k_i are labeled ω_i. The joint distribution is then estimated by

$$p_n(\boldsymbol{x}, \omega_i) = \frac{k_i/n}{V}.$$

As a result, the *a posteriori* probability can be estimated by

$$p_n(\omega_i|\boldsymbol{x}) = \frac{p_n(\boldsymbol{x},\omega_i)}{\sum_{j=1}^{C} p_n(\boldsymbol{x},\omega_j)} = \frac{k_i}{k}.$$

The *a posteriori* probability is thus, a portion of the samples within the cell labeled ω_i.

Two disadvantages of the nearest -neighbor method includes the volume of storage needed since it requires the storage of all data samples. The second is the time needed to compute the nearest neighbors (Webb, 2000). Editing techniques have been studied to reduce the number of protypes and increase efficiency.

Kernel Methods. The kernel method also tries to fix the problems of the histogram method. In the histogram method, as the dimension of the data vectors increase, the number of cells increases exponentially. The kernel method approaches this problem by fixing the cell volume, finding the number of samples within the cell, and then using this information to estimate the density (Webb, 2002). The kernel method is of wide applicability. Its properties are best understood, and these properties will relate to other methods of density estimation. This is especially true for the univariate case. (Silverman, 1986).

CHAPTER 6: DISCUSSIONS

The results for finding a pattern for students who graduate or drop-out was not overwhelming. The data set contained a very small number of dropouts, twenty students which was 4.8% of the data. I believe that this may have had an impact on the results. With the emphasis that is put on helping students stay in school, I was assuming that there would be a larger number of dropouts, due to the emphasis that is put on graduation rate. I was pleasantly surprised that in two graduating classes, with over four-hundred students total, we only had twenty dropouts.

In my initial plans for this project, I planned on using elementary and junior high attendance as an input or independent variable. I would have liked to have been able to obtain that data to see if that also had an impact in the graduation status of our students. I wanted to see if early attendance red flags would be a target area for early intervention. Even though I was not able to obtain this data, I believe that there are interesting results within the data that was obtained.

I was surprised by the fact that the free or reduced lunch status did not have a bigger impact in the decision tree process. It is very positive that our students who come from a family that is struggling, that the home situation does not have an overwhelming effect on their graduation status. However, lunch status did appear in the logistic regression model. Looking at Figure 8 shows that we still need to provide intervention to our students who qualify for free and reduced lunch.

While comparing the results from the decision tree and regression methods, there were differences in the results. When looking at proficiencies, math proficiency appeared in both models. However, for the decision tree the overall proficiency appeared while in the regression model junior high proficiency and lunch status appeared. Imputing the average value for the missing value in the regression model may have had a greater impact.

The misclassification rate for the training and validation data for the decision tree were both 4.8%. The average square error for the training data was 4.2% and 4.4% for the validation set. With the regression model, the training and validation misclassification were both 4.8% again. The average square error for the training and validation set were 4.3% and 4.5%. In determining which model may be better, we will need to look at average square error. The decision tree was a slightly better model in terms of average square error.   There is not a huge difference in terms of performance, therefore, the decision tree may be preferred to be used because it is easier to read and interpret.

When looking at the data in terms of below basic and basic versus proficient and advanced there were again differences in the model. In the decision tree, 4th grade math, Algebra 1, English 2, and lunch status appeared in the model. However, the regression model had 6th grade English language arts, English 2, 4th and 6th grade math, and lunch status as its parameters. Again, the differences could be due to missing values. When looking at the data in terms of the two categories, there were many missing values. Lunch status is the only parameter with no missing values and appeared in both models. With the regression model imputing in the most frequent value for the missing value, this could drastically change the results.

With the decision tree while using binary data, the misclassification rate was 4.8% for both the training and validation sets. The average squared error for the decision tree was 4.4% for the training set and 4.3% for the validation set. With the regression model, the misclassification rate for the training and validations sets were 3.8% and 4.8% respectively. The average squared error was 3.5% for the training set and 4.6% for the validation set. The decision tree was a more consistent model. When choosing a model to use, the consistency in the decision tree model as well as the ease of reading the model, may help to determine to use the tree model over the regression model.

From the results that were found, at the school, those students who are consistently performing at the basic or below basic level, especially in mathematics and English should be watched. We need to continue to support our students in the core areas of mathematics and English language arts starting with early intervention at the elementary schools. If we provide these students, who score below basic or basic in these areas, especially for those who qualify for free or reduced lunch, we may continue to improve our graduation status.

REFERENCES

Applied analytics using SAS® Enterprise Miner™: Course notes. (2015). Cary, NC: SAS Institute.

Bock, T. (2018, October 25). Decision trees are usually better than logistic regression. Retrieved from https://www.displayr.com/decision-trees-are-usually-better-than-logistic-regression/

Cody, R, P., & Smith, J. K. (1997). *Applied statistics and the SAS® programming language* (4th ed.). Upper Saddle River, NJ: Prentice Hall.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification* (2nd ed.). New York, NY: John Wiley & Sons, Inc.

Enough is enough! Handling multicollinearity in regression analysis. (2013). Retrieved from https://blog.minitab.com/blog/understanding-statistics/handling-multicollinearity-in-regression-analysis

Missouri Department of Elementary and Secondary Education. (n.d.). Retrieved from https://dese.mo.gov/quality-schools/mo-school-improvement-program/msip-5

Missouri Department of Elementary and Secondary Education. (n.d.). Retrieved from https://dese.mo.gov/financial-admin-services/food-nutrition-services/free-and-reduced-price-information

MSIP-5: Comprehensive guide to the Missouri school improvement plan. (2014). Retrieved from https://dese.mo.gov/sites/default/files/MSIP-5-comprehensive-guide.pdf

Pagano, M., & Gauvreau, K. (2000). *Principles of biostatistics* (2nd ed.). Pacific Grove, CA: Duxbury Thomson Learning.

Rao, V. (2013, January 13). Introduction to classification & regression trees (CART). Retrieved from https://www.datasciencecentral.com/profiles/blogs/introduction-to-classification-regression-trees-cart

Rokach, L., & Maimon, O. (2008). *Data mining with decision trees: Theory and applications* (Vol. 69). Hackensack, NJ: World Scientific.

Silverman, B. W. (1986). *Density estimation for statistics and data analysis*. New York, NY: Chapman and Hall.

Theodoridis, S., & Koutroumbas, K. (2006). *Pattern recognition* (3rd ed.). San Diego, CA: Elsevier/Academic Press.

Ville, B. D., & Neville, P. (2013). *Decision trees for analytics: Using SAS Enterprise Miner*. Cary, NC: SAS Institute.

Webb, A. R. (2002). *Statistical pattern recognition* (2nd ed.). Chichester, England: John Wiley & Sons, Ltd.