Zoo Animal Classification Using CART and Random Trees Models

Brittany Mross, Hailey Tyler, and Michelle Montevago

Marist College DATA 450L-111

TABLE OF CONTENTS

Abstract— The current study uses a dataset from Kaggle called Zoo Animal Classification with 101 cases and 16 predictor variables which were mostly flags, such as hair, feathers, eggs, milk, etc. Our goal was to build a multi-class classification model that would classify a species as one of seven basic categories based on their features. We focused on building a Random Trees model, but also built a CART model to demonstrate the functionality of decision trees. Our Random Trees model for classifying animals into seven basic categories had an accuracy of 96.77% for our testing dataset, compared to our CART model with an accuracy of 90.32%. Our results suggest that Random Trees is an algorithm that would be beneficial to utilize in animal classification. However, we used a dataset with simple features and basic categories, so this methodology should be applied to more complex data with larger sample sizes to build a model that could classify cases more specifically.

1 INTRODUCTION

1.1 Animal Classification

"Despite existing for hundreds of years, the science of classification is far from dead. Classification of many species, old and new, continues to be hotly disputed as scientists find new information or interpret facts in new ways." [1] Many people may not realize that animal classification is subjectively defined by scientists and is not clear cut. When finding a new species, or discovering something new about an already discovered species, scientists need to evaluate how its features are similar to the features of other documented species to classify it. We wanted to apply data mining techniques to make this classification easier and more consistent and reliable. By building classification models, new species can be classified based on previous data of the features of other species.

Data analysts have created many models to attempt to classify animals using images. Kumar and Divya [2] created models to automatically classify animals into 25 basic categories using KNN, neural network, and symbolic classifier. The models varied in accuracy, the best rate being 92.5%. This seems like a very good accuracy rate, but would scientists trust a model that could misclassify 8 out of every 100 species? In the current study, we theorized that accuracy may be better when using feature lists in the form of flag variables (e.g., bird: feathers, flies, lays eggs) to classify animals rather than through images which provide limited and varying information. In addition, using similar data mining techniques, scientists can avoid having to decide whether a feature of an animal category is absolutely necessary for membership (e.g., does a species have to fly to be a bird?) or is only a frequent characteristic by letting the algorithm decide. Our model uses data of features of species to classify animals into seven basic categories (that roughly correspond to classes in the official animal classification system) to test this methodology.

1.2 Data Set

The current study uses a dataset from Kaggle called Zoo Animal Classification with 101 cases and 16 predictor variables: hair, feathers, eggs, milk, airborne, aquatic, predator, toothed, backbone, breathes, fins, legs, tail, and catsize. All of these are flag variables (with values = 0, 1) except for legs which is continuous. Our goal was to build a multi-class classification model that would classify a species as one of seven categories based on their features: mammal, bird, reptile, fish, amphibian, bug, or invertebrate. This nominal target variable was called "class_type." [3]

1.3 Method Overview

We utilized the CART and Random Trees algorithms to build our single-label, multi-class classification models. This is multi-class because our model classifies a species into one of the seven possible categories, and it is single-label because each species can only belong to one of the categories. All data preparation, exploration, and modeling was done using IBM SPSS Modeler (see Figure 1 for the stream layout).

We chose to use random forest classification to provide a model as accurate as possible without overfitting the data as they use bagging and field sampling. These models are robust when dealing with large data sets and numbers of fields, so they would be useful in classifying lots of animals for scientific purposes. As a result of the tendency of the model to be much less prone to overfitting, our results are more likely to be repeated when new data is used. So, even though the data set that we used was rather small, similar results are likely to be repeated with a larger one. [4]

The Random Trees algorithm creates multiple decision trees and predicts the target variable as being the most frequent prediction among all of the models it created. So first, we built a single decision tree, our CART model, to demonstrate how decision trees function and are interpreted.

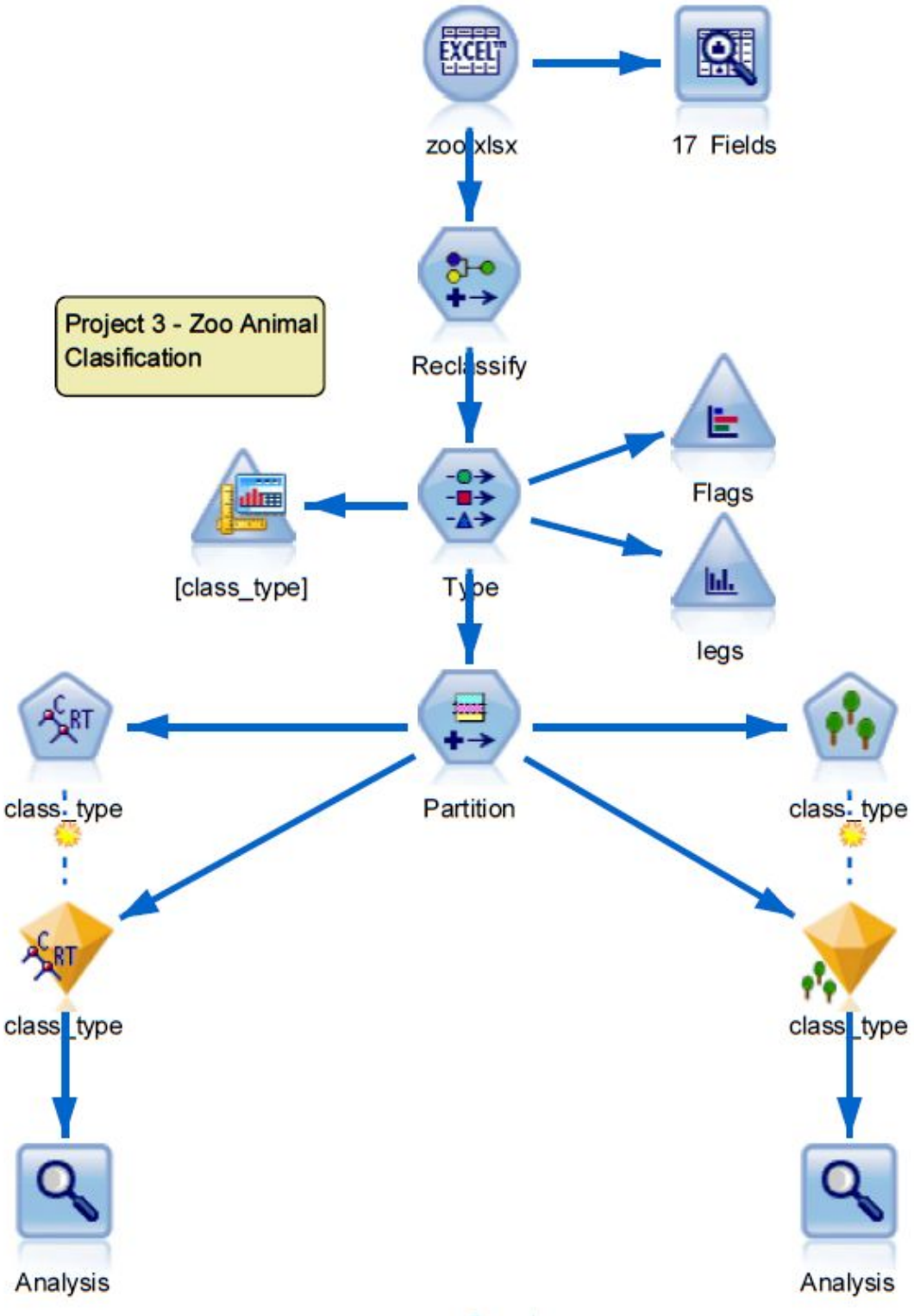

Figure 1: IBM SPSS Modeler stream

2 METHOD

2.1 Data Preprocessing

The data audit showed 100% of fields and records to be complete, so there were no missing values that needed to be replaced. The class_type variable used values in the range (1, 7) to represent the animal categories, so we used the reclassify node to rename these as the category names for readability based on the key provided (e.g., 1 → "mammal"). Min-max normalization of the variables was not necessary due to the majority being flag variables. We partitioned the data into 70% training and 30% testing sets prior to building the models.

2.2 Data Exploration

We examined the frequency distribution of the target variable class_type and the proportions of class_type in the distributions of legs and flag variables when true.

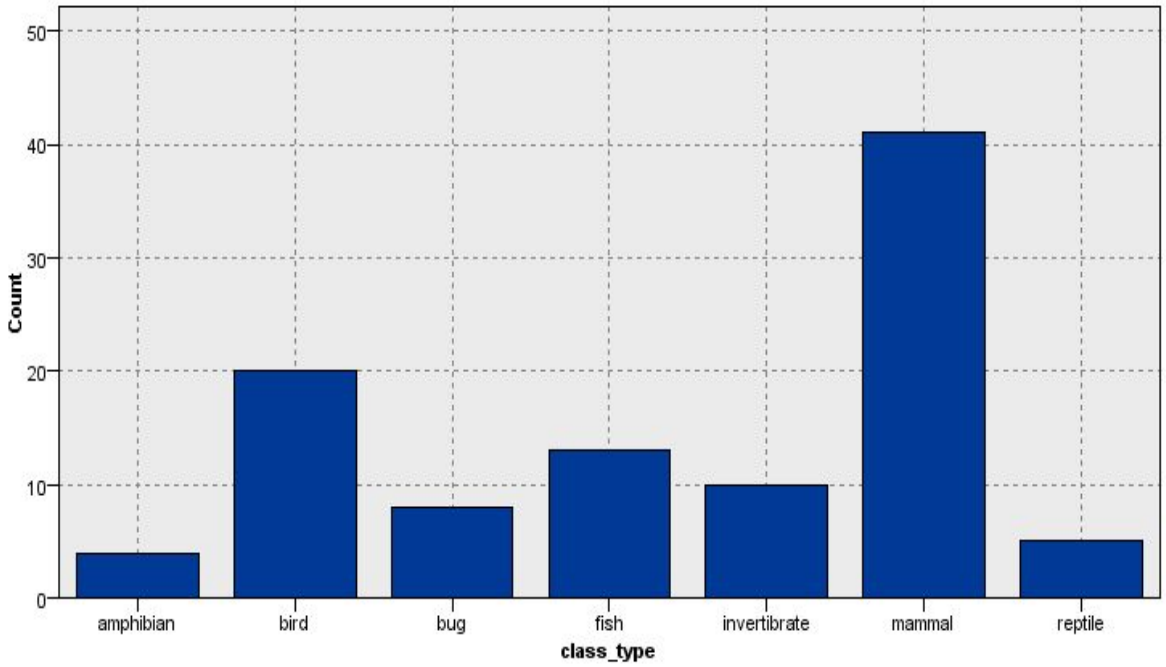

Figure 2: Frequency distribution of target variable class_type

The shape of the class_type graph is not meaningful since the variable is categorical, but it does allow us to see the distribution of the categories. Mammal is the most frequent category and amphibian is the least frequent among the cases (see Figure 2).

The proportion of the categories for legs and flags when true are going to be more heavily weighted for mammals due to the category being the most frequent in the dataset. Therefore, we cannot use Figures 3 and 4 as a way to compare the proportions of the categories, but we can use the legs distribution to tell us what the typical about of legs are for each group and we can use the flags when true to give us an idea of which categories have which features.

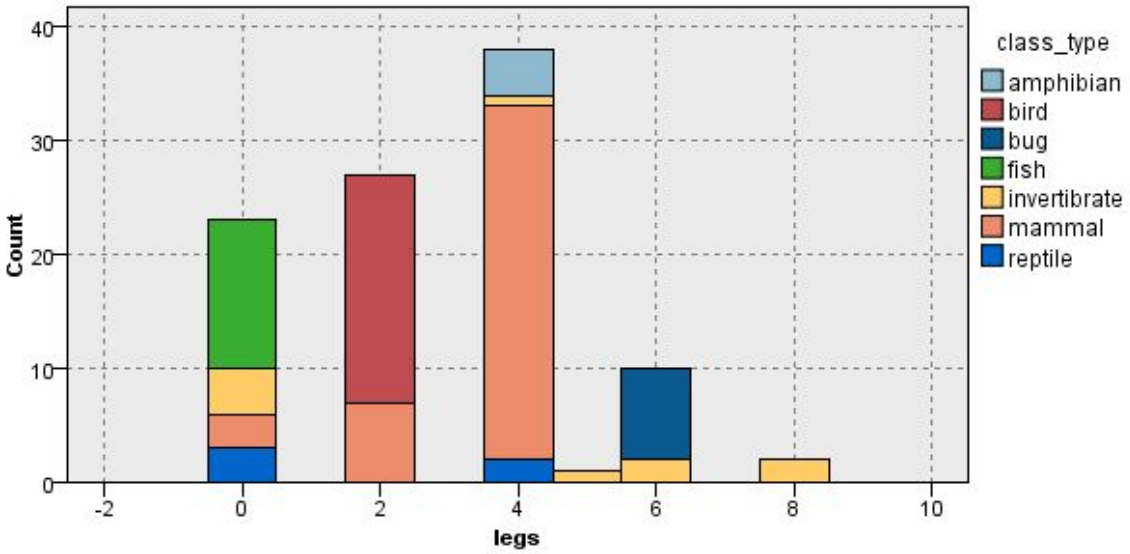

Figure 3: Distribution of continuous variable legs according to class_type

A few notable observations from Figure 3 include that amphibians always have 4 legs, birds always have 2, bugs always have 6, fish always have 0, invertebrates vary from 0 to 8, and mammals mostly have 4 but also vary. Some of this information is already obvious considering our prior knowledge about these animal categories, but this type of exploration may be useful when not having any background knowledge because it can help determine which features are

necessary for category membership and which are only frequent. For example, according to our dataset, having 2 legs is necessary for a species to belong to the bird category; in contrast, having 4 legs is a frequent occurrence for mammals but is not necessary.

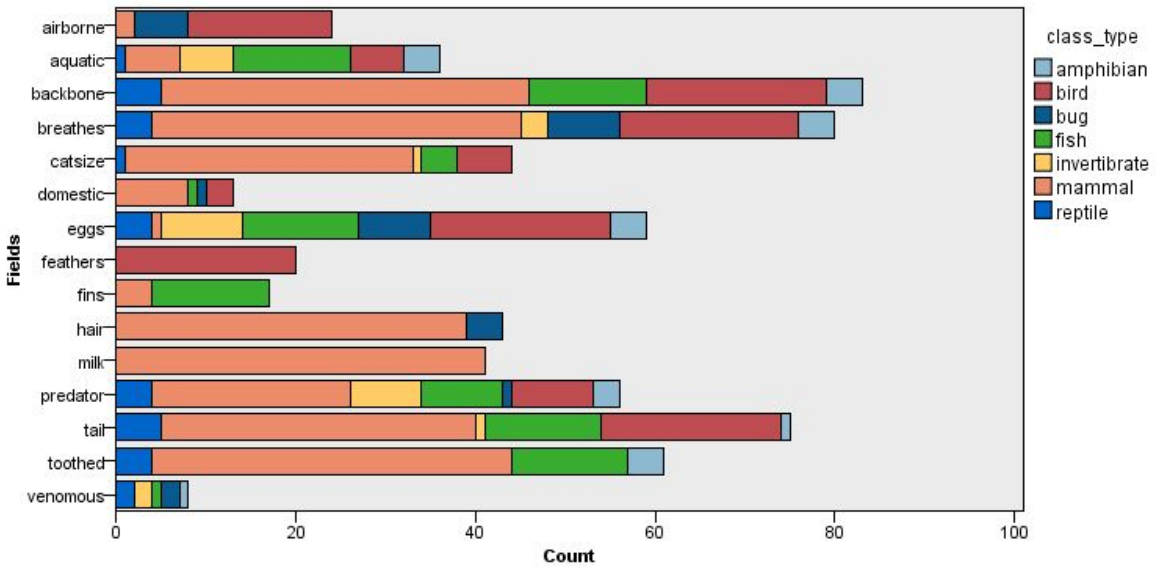

Figure 4: Frequency of true values for flag variables according to class_type

A similar exploratory analysis can be done for the flag variables when true (meaning the case has or does said variable), but we cannot conclude whether a feature is necessary for category membership because we do not see the proportions of the flag variables when false. However, we can still examine which categories have what features (see Figure 4).

2.3 Classification Models

2.3.1 CART - Decision Tree

To make our CART Decision Tree we used the Classification and Regression Tree node. We chose this method over a C5.0 model because it can accommodate continuous and categorical inputs; is easier to understand; and does not take a lot of time to process. The model takes multiple input fields to predict the one target field by using recursive partitioning to

binarily split the records into subgroups until a stopping criteria is triggered. We did not need to prune our tree because it was already simplified and easy to understand due to our data set. Only accuracy was calculated for the CART model as the Random Trees model produced a higher accuracy (See Table 1). [5]

Accuracy	
Training	0.9714
Testing	0.9032

Table 1: Percentages of correctly predicted class_type

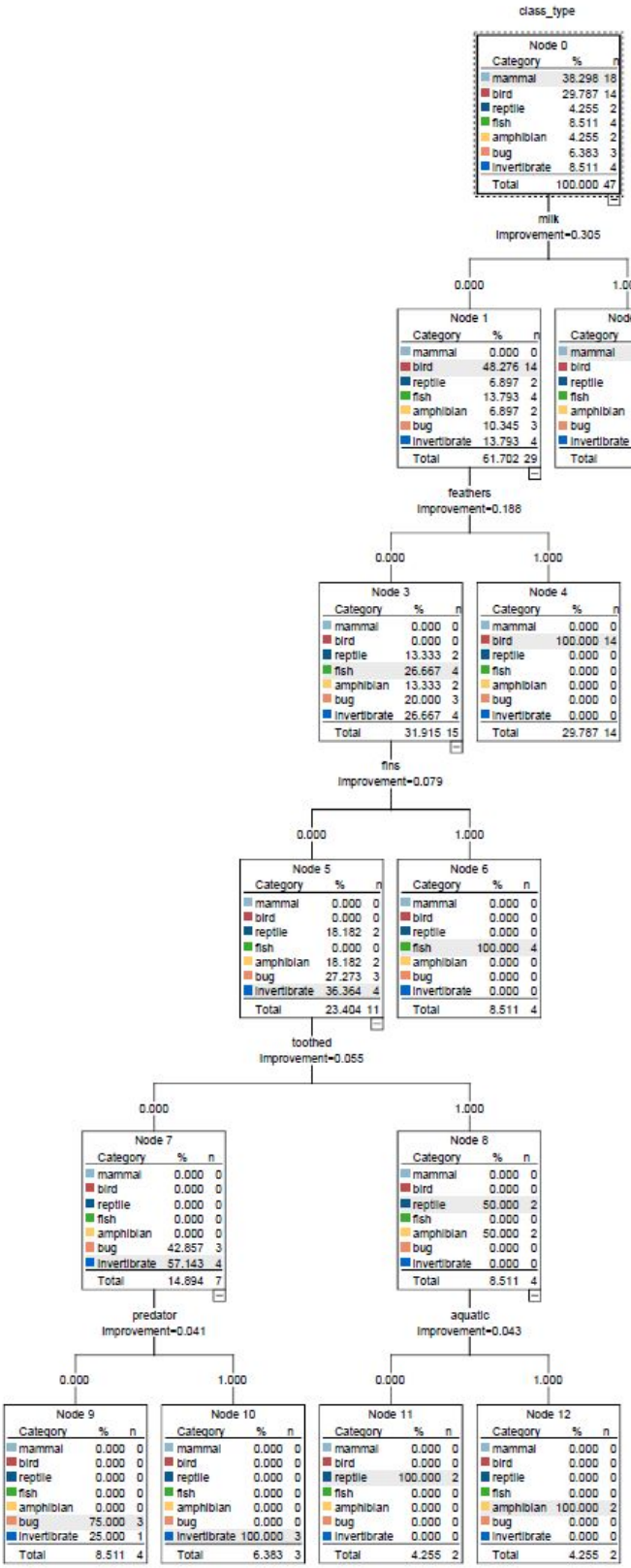

Figure 5: CART model for predicting class_type

Below we described the rules of the CART model and computed the support and confidence for each.

If (milk = 1) then mammal
    Support: 18/47 = .383
    Confidence: 18/18= 1
Elif (milk = 0)
    If (feather = 1) then bird
        Support: 14/47= .298
        Confidence: 14/14= 1
    Elif (feather = 0)
        If (fins = 1) then fish
            Support: 4/47= .085
            Confidence: 4/4= 1
        Elif (fins = 0)
            If (tooth = 1) then amphibian
                If (aquatic = 1)
                    Support: 2/47= .043
                    Confidence: 2/2=1
                Elif (aquatic = 0)
                    Support: 2/47= .043
                    Confidence: 2/2=1
            Elif (toothed = 0)
                If (predator = 1)
                    Support: 3/47= .064
                    Confidence: 3/3= 1
                Elif (predator = 0)
                    Support: 4/47= .085
                    Confidence: 3/4= .75

2.3.2 Random Trees

The Random Trees node creates an ensemble model that consists of multiple decision trees. It is a tree-based classification and prediction method that is built on Classification and Regression Tree methodology [6]. As opposed to our CART model, Random Trees uses bagging and creates bootstrap samples to form component models which form the entire ensemble model.

Additionally, at the split of each tree, only a sampling of the input fields is considered for the impurity measure which defines how well the two classes are separated. [6]

When setting up the Random Trees node, we changed the basic build options. We specified the maximum number of models for the tree to build at 50 and made the maximum tree depth, the number of times the sample is split recursively, at 8 which is half the number of predictor values.

For the predictive performance metrics, we calculated the accuracy, precision, recall, and specificity for each class and averaged them. Furthermore, we calculated the F-Measure which uses precision and recall to make the harmonic mean of the two. For all single label multi-class classifications, the higher the values of accuracy, precision, recall, specificity, and F-Measure, the better the performance of the learning algorithm [7]. We also determined the Hamming loss which is the prediction error and the missing error normalized over total number of classes and total number of examples. This follows the equation:

$HL = \frac{1}{kn} \sum_{i=1}^{n} \sum_{l=1}^{k} [I(l \in Z_i \wedge l \notin Y_i) + I(l \notin Z_i \wedge l \in Y_i)]$ where k is the number of labels, n is the number of instances, I is the indicator function, Z_i is the classifier predictions, and Y_i is the correct label vector. However, since $|Y_i| = 1$ and $|Z_i| = 1$, the equation simplifies to $\frac{2}{k}$ times the classification error which is $HL = \frac{2}{k} * \frac{f}{n}$ where f is the number of cases that were incorrectly identified and n is the total number of cases. This is because we are using a single label classifier and not a multi label one. A Hamming Loss of 0 implies a model with no error. These calculations are displayed in the chart below. [7]

	Accuracy (A) $A = \frac{1}{n}\sum_{i=1}^{n}\frac{\lvert Y_i \cap Z_i\rvert}{\lvert Y_i \cup Z_i\rvert}$	Precision (P) $P = \frac{1}{n}\sum_{i=1}^{n}\frac{\lvert Y_i \cap Z_i\rvert}{\lvert Z_i\rvert}$	Recall (R) $R = \frac{1}{n}\sum_{i=1}^{n}\frac{\lvert Y_i \cap Z_i\rvert}{\lvert Y_i\rvert}$	Specificity (S) $S = \frac{1}{n}\sum_{i=1}^{n}\frac{f}{f+\lvert Z_i\rvert}$	F-Measure (F_1) $F_1 = \frac{1}{n}\sum_{i=1}^{n}\frac{2\lvert Y_i \cap Z_i\rvert}{\lvert Y_i\rvert+\lvert Z_i\rvert}$	Hamming Loss (HL) $HL = \frac{2}{k} * \frac{f}{n}$
Training	1	1	1	1	1	0
Testing	.9677	.9881	.9287	.9932	.9575	.0092

Note: The Training set metrics are all one (except for Hamming Loss which is previously explained) because the model correctly identified all of the animals. The testing set is also nearly perfect, but misclassified a reptile as a mammal.
Table 2:Predictive Performance Metrics for Random Trees Model

3 DISCUSSION

3.1 Conclusions

Our Random Trees model for classifying animals into seven basic categories had an accuracy of 96.77% for our testing dataset, compared to our CART model with an accuracy of 90.32%. Table 3 shows the misclassification rates for both models and both partitions.

Misclassification Rate		
Model	Training	Testing

Random Trees	0/70 = 0%	1/31= 3.23%
CART	2/70 = 2.86%	3/31 = 9.68%

Table 3: The percentages of how many cases were wrongly predicted

Our Random Trees model misclassified zero cases in the training set and only one case in the testing set. It is possible that overfitting occurred with such a small sample, but these accuracy rates make us believe that if we had more data, we could refine this model to be extremely accurate without overfitting. Overall, our results suggest that Random Trees is an appropriate algorithm to utilize in animal classification.

3.2 Future Work

When building animal classification models in the future, the categories of the target variable should be more specific (e.g., genus, family) for the model to be practically useful to scientists. We built our Random Trees model to demonstrate how the algorithm can be utilized in animal classification by using feature lists, but we used a dataset with simple features and basic categories, which is a factor in why our model had such a high accuracy rate. This methodology should be applied to more complex data with larger sample sizes to build a model that could classify cases more specifically.

REFERENCES

[1] "Classification system," *Science Learning Hub,* 03-Sep-2018. [Online]. Available: https://www.sciencelearn.org.nz/resources/1438-classification-system.

[2] Y. H. Sharath Kumar and C. D. Divya, "Feature Selection Approach in Animal Classification," *Signal & Image Processing,* vol. 5, no. 4, Aug. 2014.

[3] UCI Machine Learning, *Zoo Animal Classification,* 2016. [Online]. Available: https://www.kaggle.com/uciml/zoo-animal-classification.

[4] A. Ruiz, "Random Trees algorithm in SPSS Modeler 17.1," SPSS Predictive Analytics, 12-Oct-2015. [Online]. Available: https://developer.ibm.com/predictiveanalytics/2015/10/11/random-trees-algorithm-in-spss-modeler-17-1/.

[5] "C&R Tree Node," IBM Knowledge Center. [Online]. Available: https://www.ibm.com/support/knowledgecenter/SS3RA7_18.1.0/modeler_mainhelp_client_ddita/clementine/cartnode_general.html. [Accessed: May-2019].

[6] "Random Trees node," IBM Knowledge Center. [Online]. Available: https://www.ibm.com/support/knowledgecenter/en/SS3RA7_17.1.0/modeler_mainhelp_client_ddita/clementine/rf_general.html. [Accessed: May-2019].

[7] M. S. Sorower, "A Literature Survey on Algorithms for Multi-label Learning," 2010. [Online]. Available: https://pdfs.semanticscholar.org/6b56/91db1e3a79af5e3c136d2dd322016a687a0b.pdf