Predicting Conservation Status of Bat Species with Ensemble Models: Feature Analysis and Model Evaluation

Kaelyn Long

December 2023

Contents

---

1 Introduction

Caves and karsts are fascinating ecosystems, inhabited by unique mixtures of taxa and protected from many outside influences. Bats are perhaps their most widely known inhabitants, serving as keystone species and facilitating nutrient flow into these isolated environments. As a result, bat population status is often a useful indicator of cave conservation necessity. As cave conservation funds are often lacking, however, decisions must be made in order to effectively prioritize certain caves. The DarkCideS 1.0 database was created as a tool to assist in making these decisions, with information surrounding over 6000 occurrences of cave-dwelling bats across over 2000 cave sites, 46 countries, and 12 biomes (Tanalgo et al., 2022b).

This project is an exploration of the DarkCideS 1.0 database in an attempt to predict conservation status for individual bat species based on ecological metadata. The DarkCideS database contains four sets of data concerning cave-dwelling bat species, bat-inhabited caves, and bat parasites and hyperparasites. I will be using datasets 1 and 4 in this project. Dataset 1 contains species-specific metadata such as habitat preference,

feeding groups, distribution range, generation length, ecological status, and exposure to various potential threats. Dataset 4 contains species of parasitic bat flies, their *Laboulbeniales* fungal hyperparasites, and their associations with different bat species.

For this project, I will be focusing on the **Conservation.status** variable as a target. This variable classifies each bat species present in dataset 1 into one of five levels of conservation status, or as data deficient. The data deficient records will be separated out, models will be trained and validated on the remainder of the data, and a heterogeneous ensemble model will be used to predict possible values for the data deficient record.

The incorporation of dataset 4 will come through the creation of novel parasitemia features for dataset 1. Three features that capture different levels of parasitemia will be created and assessed for a relationship with the target feature, **Conservation.status**. The specifics of the features are as follows:

- Parasitemia indicator: Does the bat species have one or more recorded parasitemia events?
- Number of unique batfly parasites: How many species of batfly are known to parasitize the bat species?
- Number of unique *Laboulbeniales* parasites: How many species of *Laboulbeniales* hyperparasites are known to be associated with the bat species?

After this feature creation, both the original and new features will be evaluated for association with the target feature, **Conservation.status**. Various metrics, including Chi-square, ANOVA, and Pearson correlation analysis, will be used to evaluate interactions between variables with different data types. Additionally, Factor Analysis of Mixed Data (FAMD) will be used to visualize feature contributions and potential clustering.

Pre-processing steps for the data will include handling of outliers and missing data, scaling and normalization, and class balancing. These steps will be carried out for training and validation sets separately in the interest of simulating model performance with novel data.

The base models to be trained on the data include the following:

- Naive Bayes: the Naive Bayes model is suitable for multiclass classification as well as a high number of categorical variables
- Decision Tree: the Decision tree model is suitable for diverse features and a mix of categorical and continuous variables
- Support Vector Machine (SVM): the SVM model is suitable for a high number of features and is resistant to overfitting

These models will undergo tuning of hyperparameters as appropriate. They will then be incorporated into two ensemble models: the Random Forest classifier, and a majority vote heterogeneous ensemble model.

The performance of each model, both separately and in ensembles, will then be assessed via k-fold cross-validation. The main evaluation metric used will be the F1 score, which combines the precision and recall metrics for each class. As the target feature, **Conservation.status**, has multiple levels, the F1 scores for each class will be summed for comparison against other models. Additionally, the accuracy of each model will be considered as a secondary metric.

Finally, our heterogeneous ensemble model will be used to generate predictions for the **Conservation.status** value of each record originally classed as "Data Deficient."

In the interest of transparency, I also want to bring attention to previous analysis efforts. The DarkCideS 1.0 database has been utilized to perform an assessment of conservation priorities of specific cave sites using a binomial generalized linear model to assess variables as predictors of binary threat status (Tanalgo et al., 2022a).

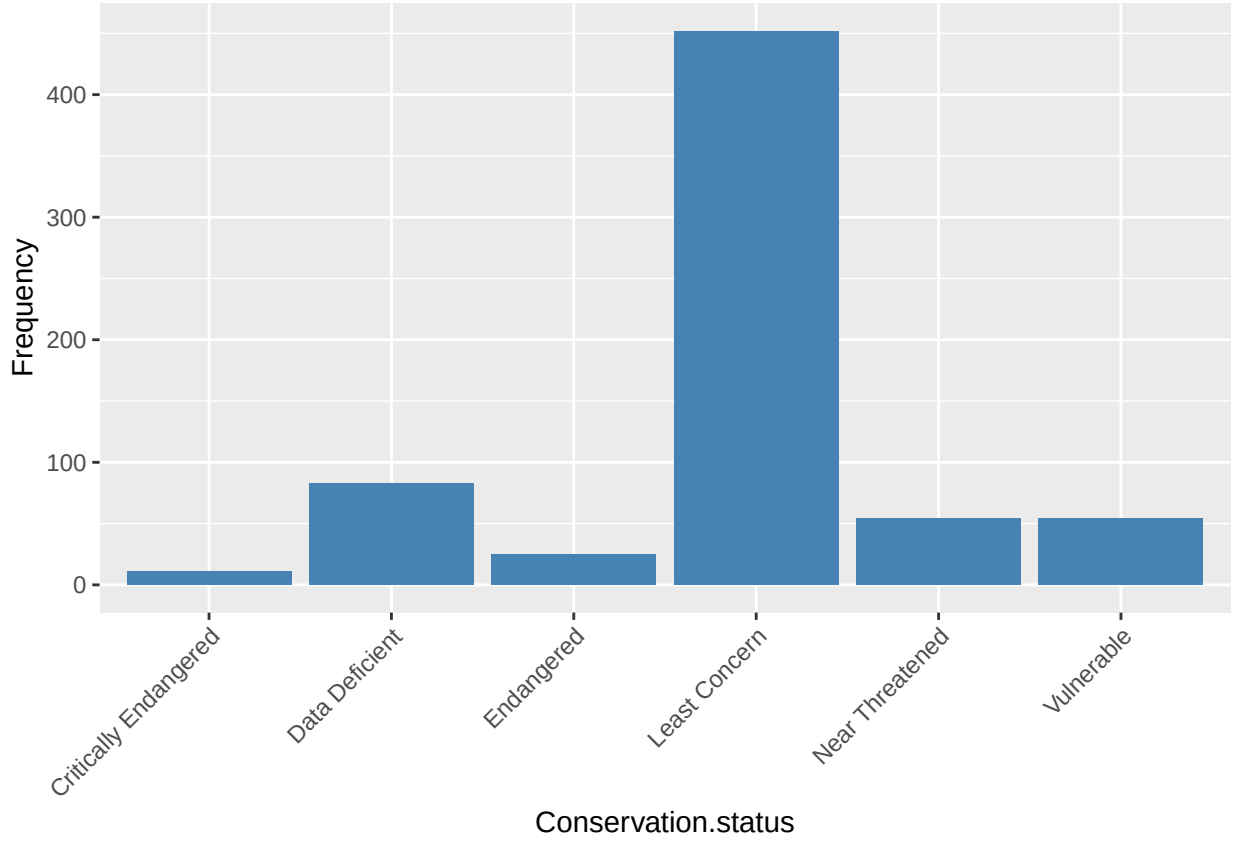
---

2 Data Handling

2.1 Data Acquisition

The two datasets used in this analysis can be retrieved from Figshare through the DarkCideS 1.0 official website. Both are retrieved as `.csv` files. Dataset 1 is focused on bat species and has 679 observations of 1 ID feature, 5 taxonomic features, 40 ecological metadata features including the target, and 6 additional reference features. Dataset 4 is focused on parasitemia by batflies and *Laboulbeniales* hyperparasites and has 126 observations of 1 ID feature, 5 parasitemia metadata features, 2 reference features, and 1 blank column.

As a first look at our prediction goal, we will examine the distribution of our target feature, **Conservation.status**.



We can first see that **Conservation.status** is extremely imbalanced, which could pose an issue for our models. In order to rectify this problem, we will later employ the Synthetic Minority Oversampling Technique (SMOTE) to strategically oversample minority classes. This balancing will be done for each training or validation dataset as it is created and the performance of each model with both balanced and unbalanced data will be assessed.

We can now proceed to our data acquisition and feature engineering. The six reference features in the bat dataset simply contain source information for other features. We will begin by removing these. We will also remove the record ID column. Additionally, taxonomic information on the suborder and family level will not be used to predict conservation status and will be removed. For the parasitic fly dataset, we only need the features **Bat.species**, **Batfly.parasite**, and **Laboulbeniales.hyperparasite** to create our parasitemia features for the bat dataset.

Before creating the parasitemia features, however, we need to clean up some of the values in the fly dataset. Multiple values have accessory information contained in brackets that needs to be removed.

Once the fly dataset is clean, we use it to derive additional features for the bat dataset. The features are

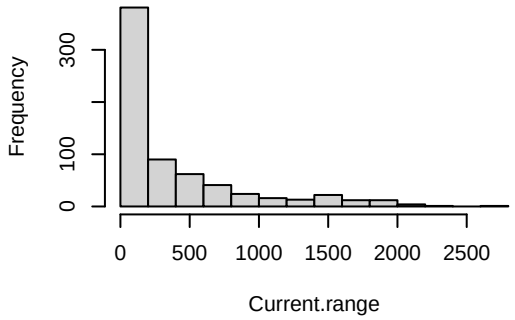
defined as follows:

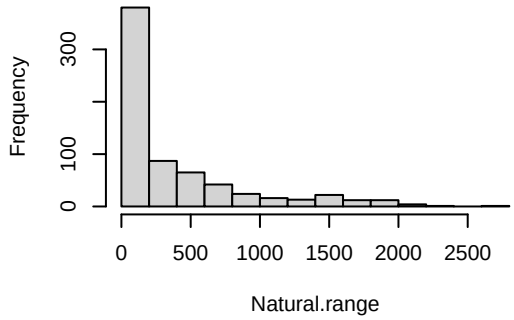
- Parasitemia indicator: This feature has a value of 1 if the bat species is recorded to have any parasitemia, and a value of 0 if not
- Number of unique fly parasites: This feature is an integer count of the number of unique fly species reported to parasitize the bat species
- Number of unique *Laboulbeniales* hyperparasites: This feature is an integer count of the number of unique *Laboulbeniales* species reported to be involved in a parasitic relationship with the bat species

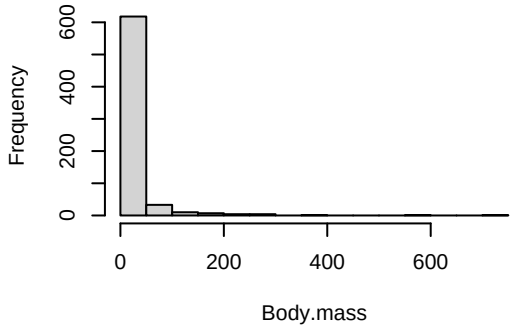
Once these features are created, we remove the taxonomic genus and species information from the bat dataset, as it does not have any direct bearing on conservation status.

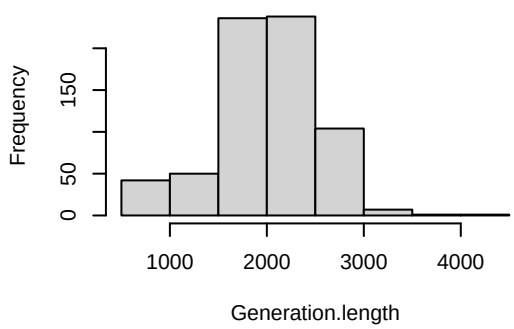
2.2 Data Cleaning & Shaping

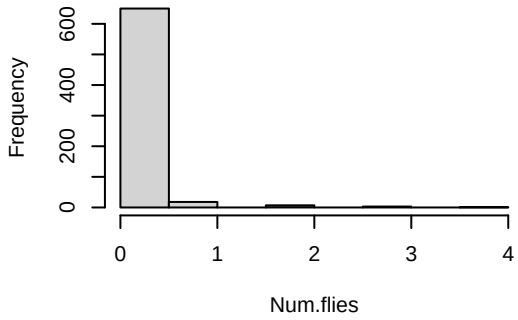
Now we move on to cleaning and shaping the dataset. We perform outlier and missing value detection. For outliers, we create a dataset that retains outliers as well as one that has outliers removed. Model performance for each of these will be evaluated further down the line. We also visualize each of our continuous features to assess outlier presence and distribution.

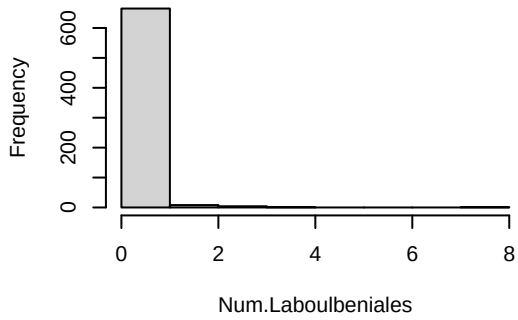












We found that though our dataset does not contain any true missing values, the target classification feature **Conservation.status** has a small number of records classified as "Data Deficient." Because these are not true missing values we will not impute them with the median value. Instead, we will move these records to their own dataset to serve as our final prediction goal after training the ensemble model.

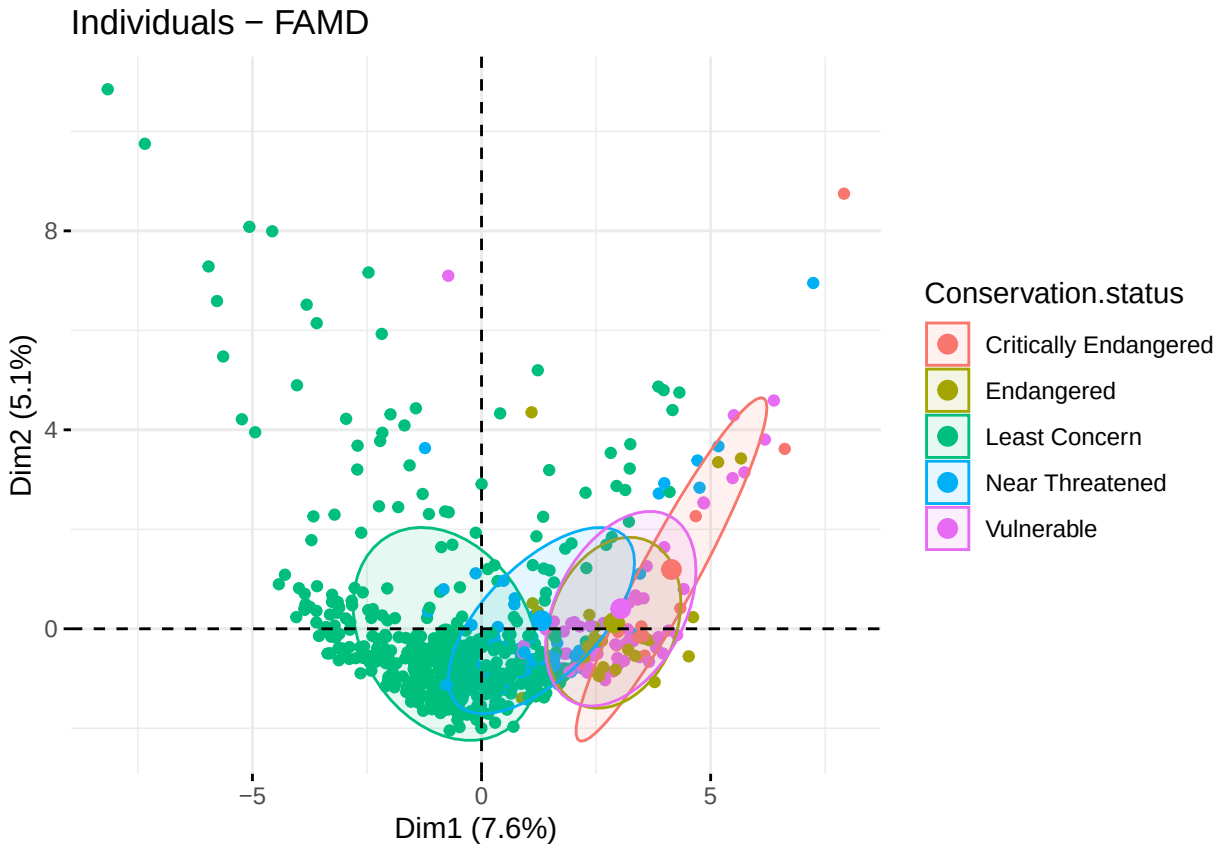
Because our dataset has no true missing values, we randomly remove some values in order to demonstrate imputation methods. Imputation methods for our data involve the replacement of missing continuous values with that feature's median value and the replacement of missing categorical values with that feature's statistical mode. The resulting datasets with both missing and imputed data are conserved separately from the original dataset for comparison later on.

We also create a function to log-normalize the datasets' continuous features. Scaling and normalization of these features will occur as part of the training of each model, as we will be mainly using k-fold cross-validation to assess our models and we want to preserve the independence of the training and validation partitions.

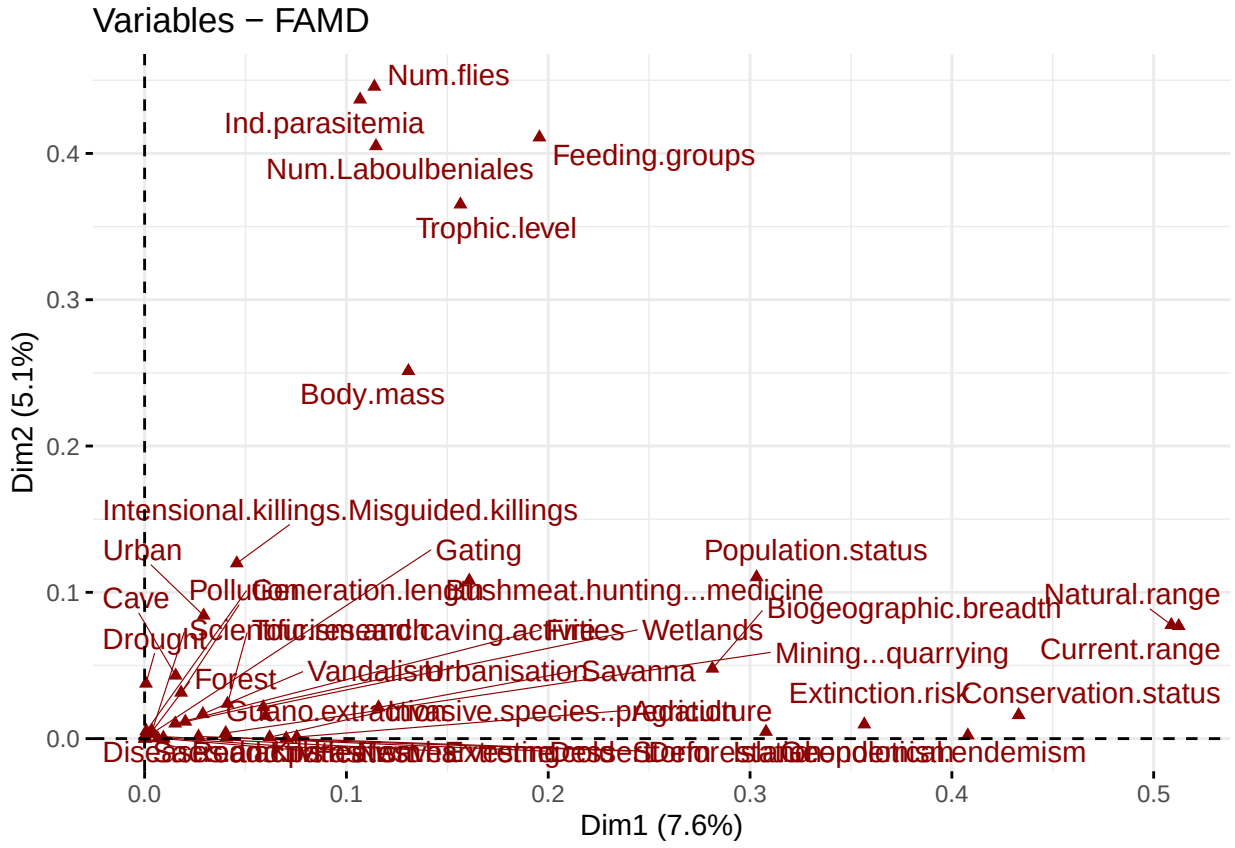
2.3 Data Exploration

We will then explore the relationships between features in our dataset. We begin with a visualization of the dataset through the lens of dimensionality reduction. As we have a mix of categorical and numeric features, we will employ Factor Analysis of Mixed Data (FAMD) rather than the typical Principal Component Analysis (PCA), which only takes continuous variables.

First, we see our data plotted as a function of the first two dimensions. There is substantial overlap between the levels in the target variable, **Conservation.status**, but we can see that dimensions 1 and 2 explain 7.6% and 5.1% of the dataset variance, respectively.

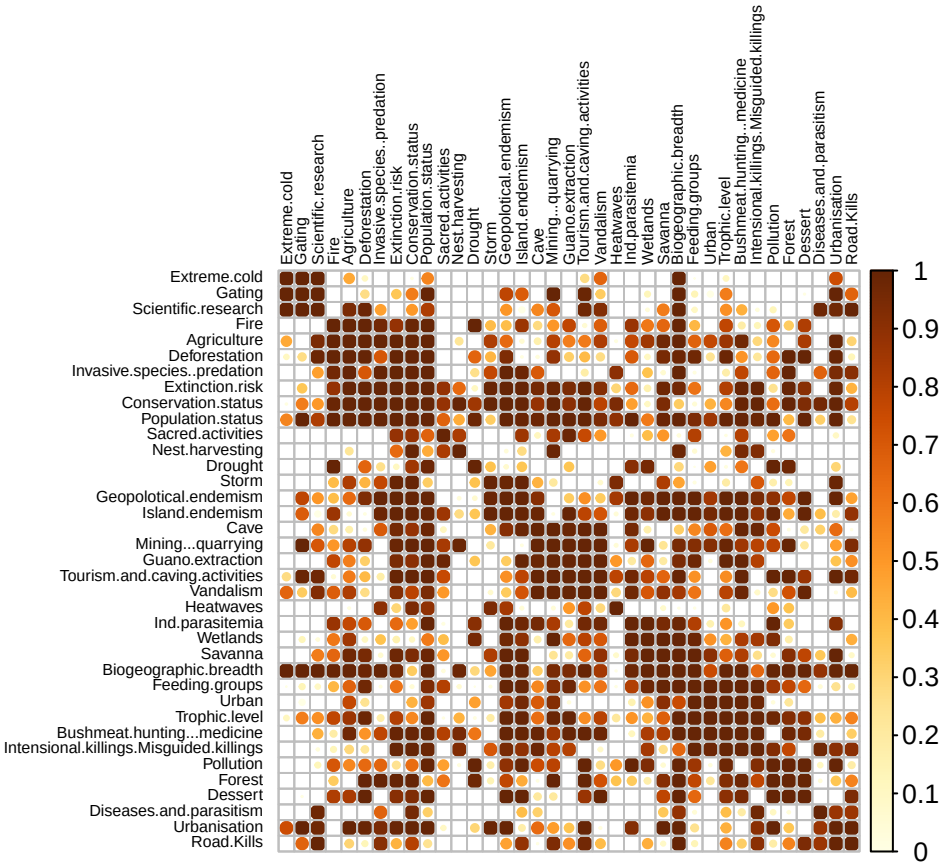


We now take a look at the contribution of each variable to each of the first two dimensions. We can see that the variable with the most contribution to the first dimension is **Current.range**, while the variable with the most contribution to the second dimenstion is **Num.flies**.

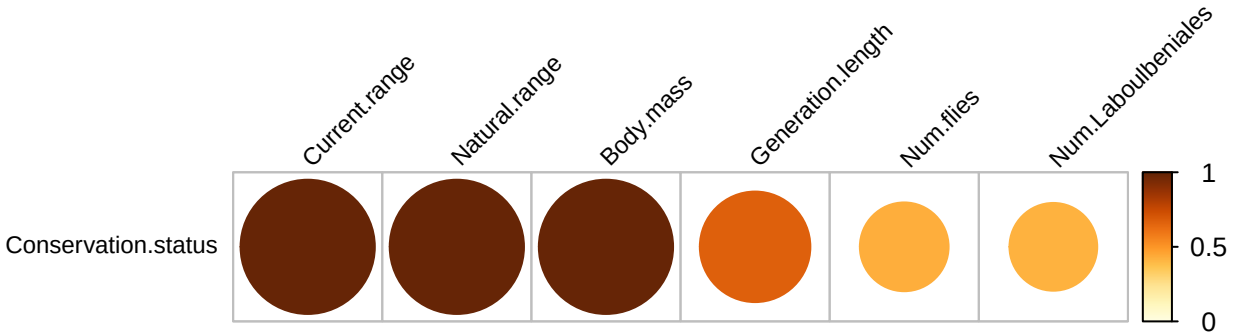


Now we will use Chi-square, ANOVA, and Pearson correlation analysis to quantitatively assess interactions between features. Our target variable, **Conservation.status**, is categorical, so we will use Chi-square analysis to detect associated categorical variables and ANOVA to detect associated continuous variables. We will also use Chi-square analysis and Pearson correlation to identify associations between non-target variables.

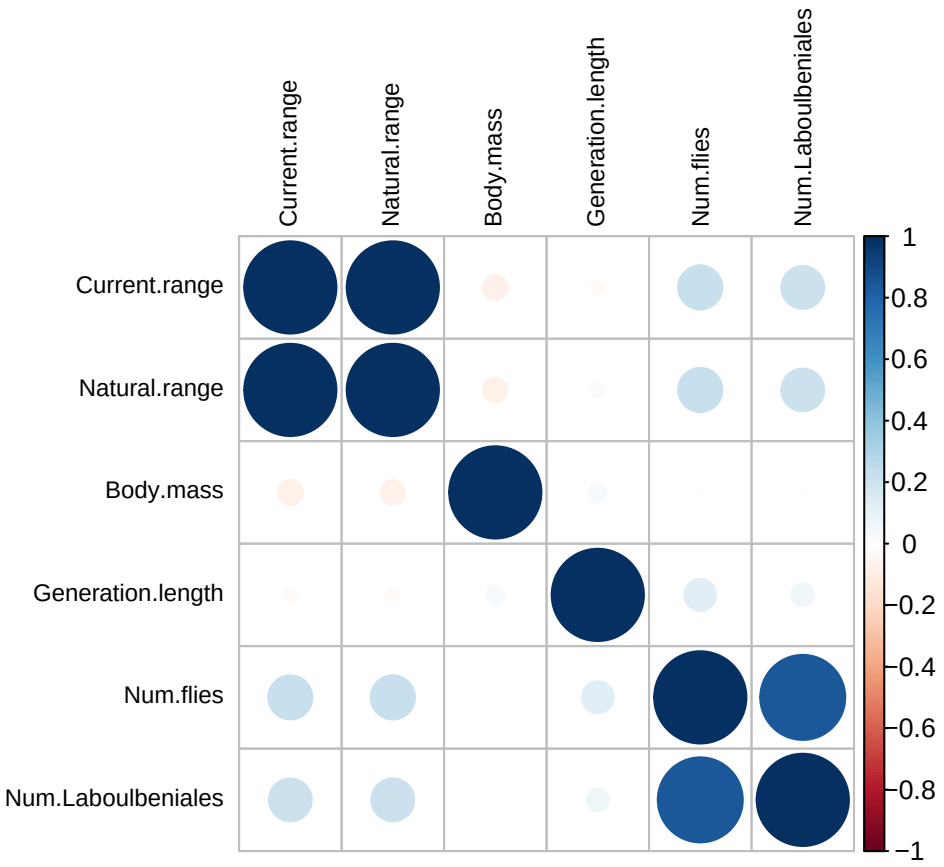
We first see the Chi-square-identified relationships between each of our categorical variables, including our target feature, **Conservation.status**. The color of each square is indicative of the value $1 - p$, meaning that dark squares indicate a greater degree of association between the two variables.



We then examine the ANOVA-identified relationships between our continuous variables and our target feature, **Conservation.status**. Here we again see the color reflecting the value of $1 - p$.



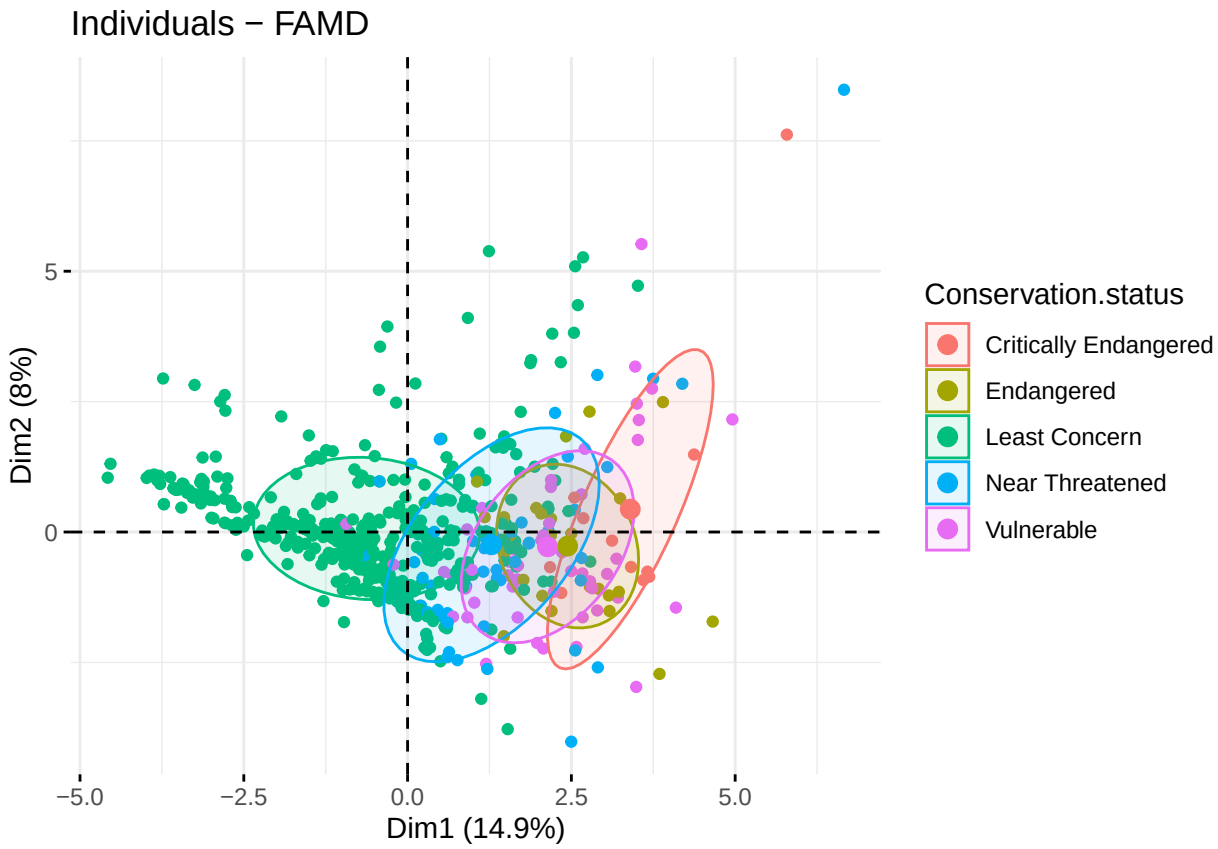
Finally, we use the Pearson correlation to view correlations between continuous variables. In this plot, the color corresponds to the value of the Pearson correlation coefficient, such that darker blue or red colors indicate a greater degree of positive or negative correlation, respectively.



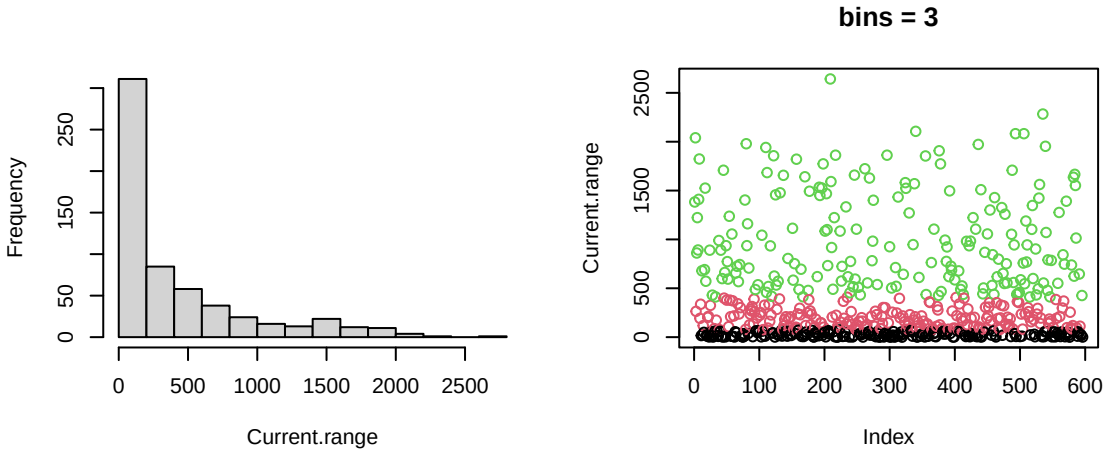
With these plots, we can see that there is a great deal of association between predictor features. Looking first at our categorical features, whose relationship is defined by the Chi-square p-value, two major trends of equivalent associations are seen. **Conservation.status** shows a high degree of association with **Extinction.risk** and **Population.status**, and **Geopolotical.endemism** shows a high degree of association with **Island.endemism**. For our continuous variables, we similarly see collinearity between **Current.range** and **Natural.range**. We will not remove these associated variables as their presence does not impact model performance, but we will keep these relationships in mind for interpretation down the line.

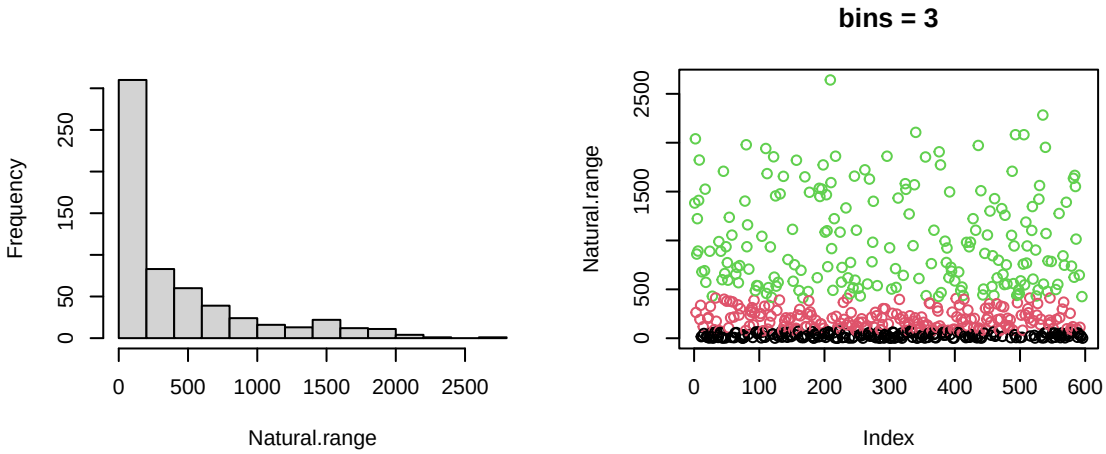
We will then remove all remaining features that do not show a significant correlation with our target feature, **Conservation.status**.

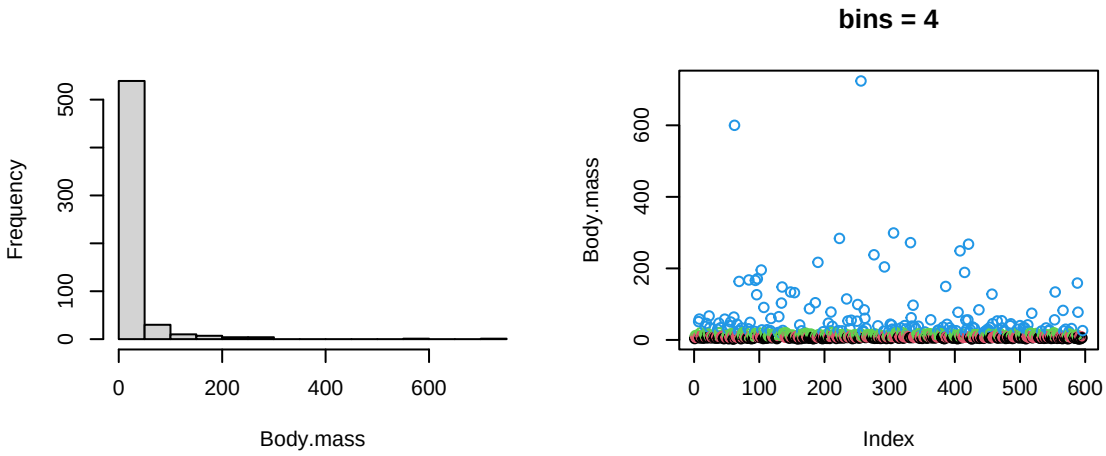
After feature removal, we again conduct FAMD to assess the difference. We can see that dimensions 1 and 2 now explain 14.9% and 8% of the dataset variance, respectively.



In preparation for model fitting, we now create some functions and variables to assist in balancing and discretizing our data. As our target feature, **Conservation.status** is fairly imbalanced, we will first create a balancing function that oversamples each minority class using the SMOTE algorithm. We will then examine our continuous features and define the number of bins to use for discretization. Below we can see the distribution of each of our remaining continuous features, and the membership of each point once binning is carried out. We chose to create sample bins by quantile to best represent the distribution of the data.







3 Base Model Construction, Tuning and Evaluation

3.1 Model Construction

We now move on to the construction of k-fold cross-validated base models in preparation for tuning. The parameters we will be tuning for each model are as follows:

- Naive Bayes: the main parameter we will be tuning is the LaPlace estimator.
- Decision Tree: the parameters we will be tuning include the minimum observations required for a node split, the minimum number of observations in a terminal node, and the maximum node depth.
- SVM: we will be selecting an appropriate kernel and tuning the regularization parameter C.

We also want to draw attention to the fact that k-fold cross-validation, complete with separate pre-processing of each selected training or validation set, is conducted entirely within each model function. The function takes in the complete raw dataset, excluding "Data Deficient" records, and creates 10 pairs of training and validation subsets with a 9:1 size ratio. Each subset then undergoes class balancing, missing value imputation, scaling, normalization, and continuous variable discretization as necessary for the base model.

3.2 Model Tuning & Performance Improvement

Also in preparation for model tuning, we will now set up our functions for performance evaluation. The main metric by which we will be assessing each model is the F1 score. The F1 score incorporates both precision and recall and is especially pertinent for a multiclass classification problem with inherently imbalanced class levels. Due to the multiple levels possessed by our target variable, **Conservation.status**, we will combine the F1 scores for each class into an overall **F1.Sum** statistic. This will take into account individual class prediction failure due to imbalanced data or an overfitted model and will be used in conjunction with accuracy to determine ideal model hyperparameter values.

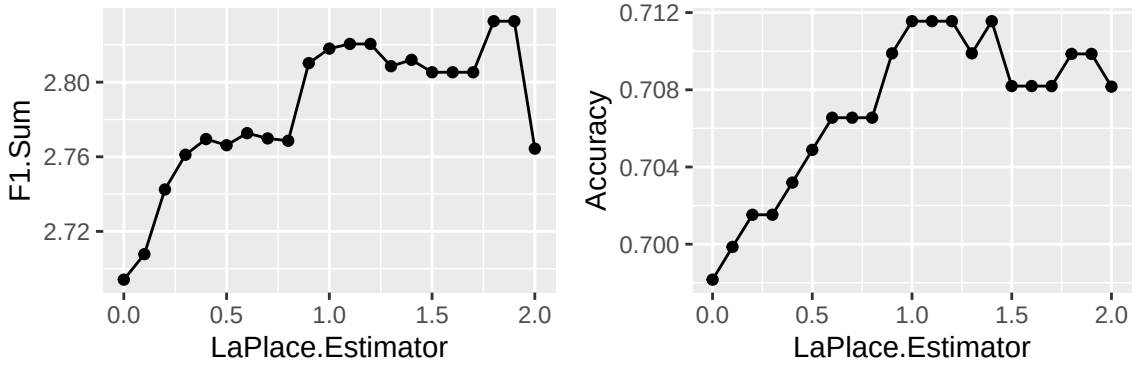
Now we will carry out model tuning by supplying a range of values for each tunable parameter and assessing the cumulative per-class F1 scores and overall accuracies at each value. Additionally, we will assess the effectiveness of class balancing via SMOTE for each model, as some are more prone to overfitting than others.

Model	Balanced_F1.Sum	Unbalanced_F1.Sum
Naive Bayes	1.556168	2.8326884
Decision Tree	1.478593	1.9139609
SVM	1.684023	0.8618101

After assessing the effect of SMOTE class balancing on each model, the balancing conducted is as follows:

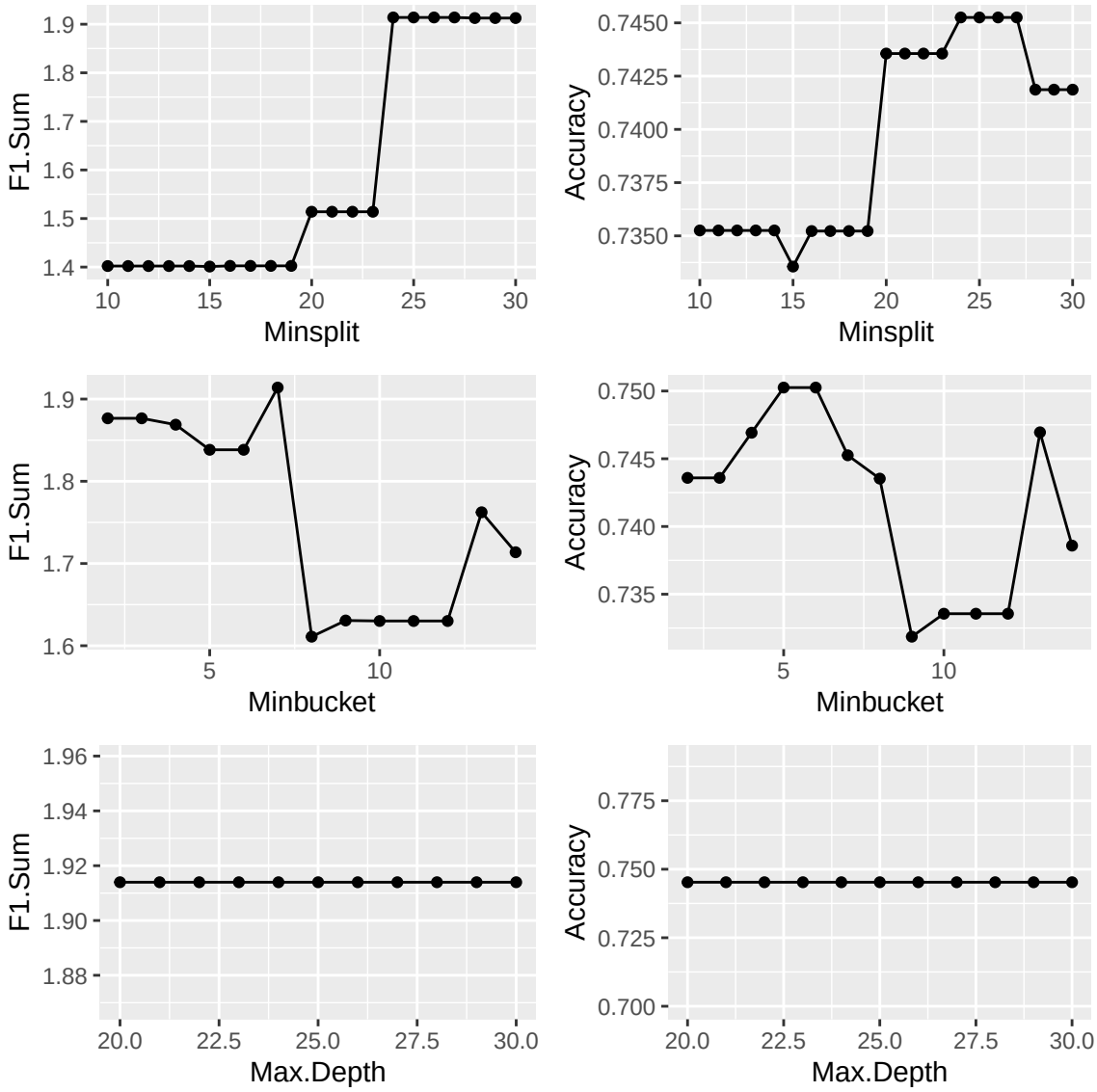
- Naive Bayes: unbalanced
- Decision Tree: unbalanced
- SVM: balanced

After Naive Bayes tuning, the ideal LaPlace estimator value is determined to be 1.8.



After tuning the Decision Tree model, the final model is as follows:

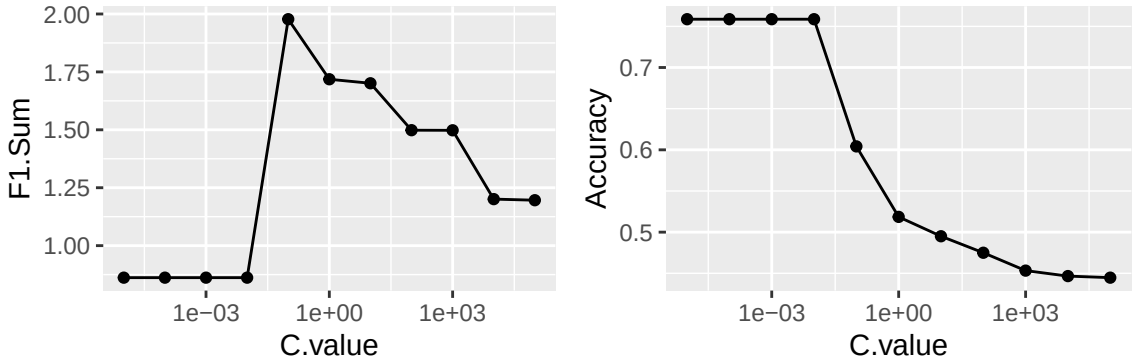
- Minimum Observations for Node Split: 24
- Minimum Observations in Terminal Node: 7
- Maximum Node Depth: 30



After tuning the SVM model, the final parameters are as follows:

- Kernel: Radial Basis "Gaussian" kernel
- C Regularization Parameter: 0.1

Kernel	F1.Sum	Accuracy
rbfdot	1.742489	0.5185593
polydot	1.595793	0.5019209
vanilladot	1.601848	0.4832203
tanhdot	1.277673	0.3779096
laplacedot	1.704180	0.5019209
besseldot	1.723830	0.5253672
anovadot	1.218548	0.2095480



3.3 Base Model Evaluation

We will now evaluate the performance of each individual base model after tuning. The model with the best performance, as assessed via F1 scores and accuracy, will serve as the tiebreaker for the majority vote heterogeneous ensemble model.

Model	F1.Sum	Accuracy
Naive Bayes	2.832688	0.7098588
Decision Tree	1.913961	0.7452542
SVM	1.973189	0.6008757

We can see above that the Naive Bayes model possesses the highest summed class-wise F1 scores, with only a small drop in accuracy when compared to the Decision Tree model. As the F1 score encompasses both precision and recall, we are prioritizing it over the simple accuracy metric. Additionally, we notice the large gap between the F1.Sum score of the Naive Bayes model and the other two models. This indicates that the Decision Tree and SVM models were unable to identify one or more minority classes from the **Conservation.status** target variable. We preemptively addressed this issue with SMOTE class balancing as described above, but it is hypothesized that the limited number of examples available for the minority classes simply do not provide enough information to appreciably improve results with balancing.

---

4 Ensemble Model Construction and Evaluation

4.1 Random Forest and Heterogeneous Ensemble Models

Now that our ideal model parameters have been determined and model performance has been assessed, we will use our base models to create two ensemble models. A bagging Decision Tree model will be implemented via the Random Forest classifier, and our three base models will also be combined into a heterogeneous ensemble model. For this model, the target class, **Conservation.status**, will be predicted by each base model and a majority vote will be used to determine the consensus prediction for each data point. In the event of a three-way tie, the Naive Bayes prediction will become the consensus, as that model was found to have the best performance after tuning.

4.2 Ensemble Model Evaluation

We will now assess our two ensemble models: the heterogeneous ensemble and our Random Forest implementation. These models will again be assessed via F1 score summation and accuracy, and we will also compare them to all three of our base models.

Model	F1.Sum	Accuracy
Naive Bayes	2.832688	0.7098588
Decision Tree	1.913961	0.7452542
SVM	1.973189	0.6008757
Random Forest	2.734688	0.7537853
Heterogeneous Ensemble	2.830592	0.7165254

It can be seen above that the Naive Bayes model continues to generate the best F1.Sum score, though it is closely followed by the our heterogeneous ensemble model. Additionally, the heterogeneous ensemble displays a slight improvement in accuracy over the Naive Bayes model. The Random Forest model also performs quite well, with a slight drop in F1.Sum and a slight increase in accuracy.

5 Data Assessment and Prediction

5.1 Effects of Preprocessing

As a last step before predicting **Conservation.status** values for the "Data Deficient" records, we will compare the effects of outlier removal and missing value imputation in the context of our heterogeneous ensemble model. The dataset pairs we will be comparing are as follows:

1. Outliers
   - Outliers are left in
   - Outliers are removed
2. Missing Values
   - Missing values are removed
   - Missing values are imputed with median or mode

Below we see the F1.Sum and accuracy metrics for each dataset.

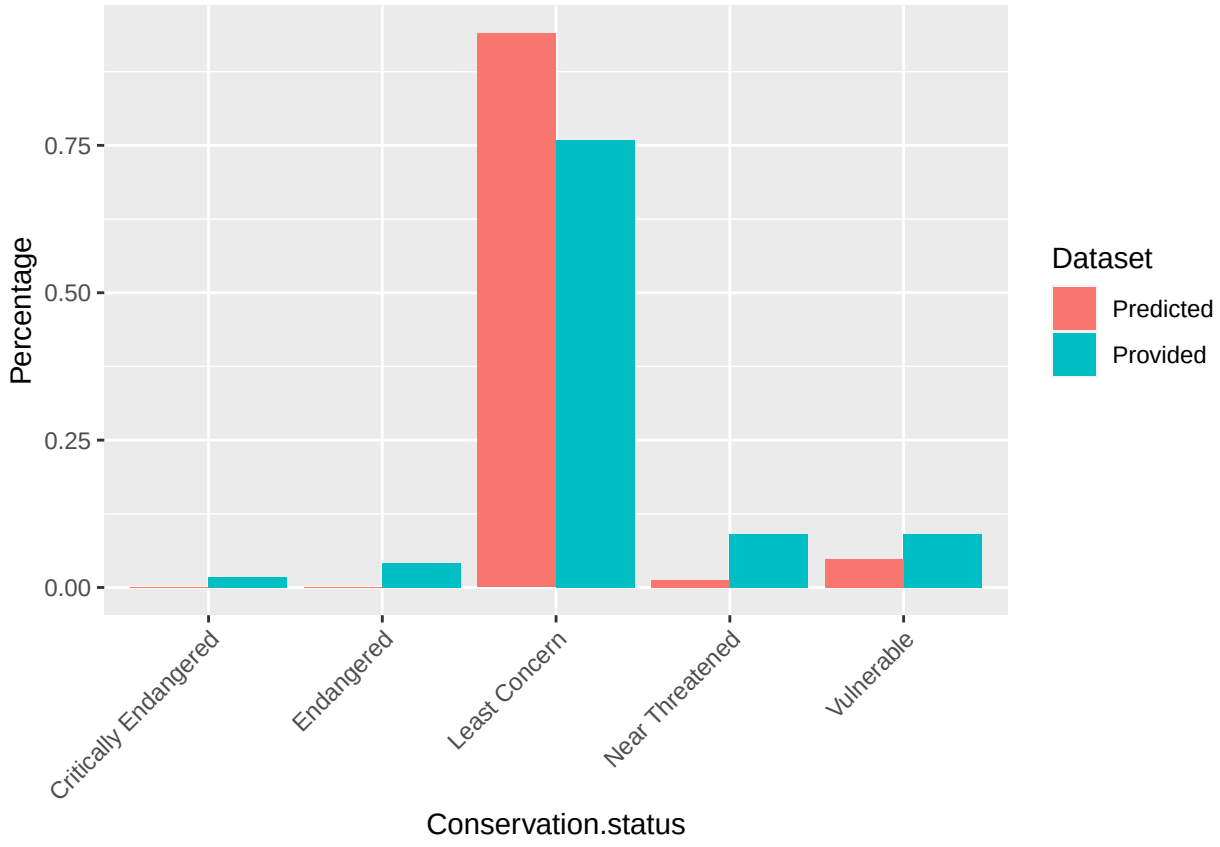
Dataset	F1.Sum	Accuracy
Has outliers	2.8305923	0.7165254
Removed outliers	1.4564187	0.5912917
Removed NAs	0.9199505	0.5732006
Imputed NAs	2.8359708	0.7181921

We can see here that the ideal outlier handling procedure for this dataset is to keep them in. This is likely due to the imbalance in **Conservation.status** levels causing minority levels to be detected as outliers. Additionally, we see that the ideal method of missing value handling is imputation with median or mode, according to value type.

5.2 Final Predictions

Finally, we use our heterogeneous ensemble model to predict **Conservation.status** values for those records that originally fell into the "Data Deficient" level. The frequency of each level within the predicted results as well as the level percentages compared to the percentages of those levels originally provided in the dataset can be seen below.

Conservation.status	Frequency
Critically Endangered	0
Endangered	0
Least Concern	78
Near Threatened	1
Vulnerable	4



6 Conclusion

Over the course of this project, we analyzed the utility of ecological metadata surrounding bats as predictors for conservation status. We combined two datasets to create novel features indicating parasitemia and assessed their relationship with conservation status. We also tested the effectiveness of various pre-processing techniques such as class balancing on our specific dataset and their utility in individual models. We applied three base models to the data and tuned their hyperparameters before constructing ensembles. Two ensemble models were then created and applied. The performance of each of these models was assessed throughout via k-fold cross-validation and the F1 score and accuracy metrics. Finally, our heterogeneous ensemble was applied to make predictions for a portion of the original data that lacked a definitive conservation status.

These resulting predictions followed the distribution of conservation status values in the original dataset, and our ensemble model assessment indicates a classification accuracy of 72%.

As we constructed the models, we also were able to assess the metadata features that contributed the most to conservation status. We learned that a bat species' current range shows the most association with conservation status, indicating that this could be a valuable metric for assessing conservation status in bat species for which less ecological metadata is available.

---

7 References

Tanalgo, K. C., Oliveira, H. F. M., & Hughes, A. C. (2022a). Mapping global conservation priorities and habitat vulnerabilities for cave-dwelling bats in a changing world. *The Science of the Total Environment, 843*, 156909. https://doi.org/10.1016/j.scitotenv.2022.156909

Tanalgo, K. C., Tabora, J. A. G., de Oliveira, H. F. M., Haelewaters, D., Beranek, C. T., Otálora-Ardila, A., Bernard, E., Gonçalves, F., Eriksson, A., Donnelly, M., González, J. M., Ramos, H. F., Rivas, A. C., Webala, P. W., Deleva, S., Dalhoumi, R., Maula, J., Lizarro, D., Aguirre, L. F., . . . Hughes, A. C. (2022b). DarkCideS 1.0, a global database for bats in karsts and caves. *Scientific Data, 9*(1), 155. https://doi.org/10.1038/s41597-022-01234-4