Project Report

Tianwei Mo, tm3929

Ziwei Wang, zw3296

# Quad chart



# 1. Introduction

As the crisis of species extinction becomes more and more serious, knowledge of biogeography and macroecology is needed to help relieve the dilemma. Our goal of the project is to develop a machine learning method that can automatically determine how at risk a species is. Our models take information about the features of species as input (e.g. features related to their reproduction, diet. life habits, etc) and output a prediction indicating the threatened status. The models we used are SVM, BPNN(Back Propagation Neural Network), KNN(K-Neighbours), and RF(Random Forest)[[1]](https://www.cnblogs.com/wj-1314/p/9628303.html).

The data we mainly used came from COMBINE: a coalesced mammal database of intrinsic and extrinsic traits[[2]](https://esajournals.onlinelibrary.wiley.com/doi/full/10.1002/ecy.3344#support-information-section). The dataset contains an extensive review of published mammal trait data sources between 1999 and May 2020. We used the latest version dataset (imputation\_phylo\_979.csv) for training.

IUCN Red List[[3]](https://www.iucnredlist.org/) records the current status of a wide range of species. According to the number of extant species, the website divides them into 7 species: least concern (LC), near threatened (NT), vulnerable (VU), endangered (EN), critically endangered (CR), extinct in the wild (EW), and extinct (EX). The website provides labels of the species from the COMBINE database.

# 2. Contribution

## (1) Generate the dataset

We first download the dataset. While the dataset contains species and their feature, it does not include the threatened status, which is the labels. We wrote a python script (copy\_data.py) to scrap the class labels from the IUCN website with the species names. However, the script performed quite slowly, eventually we searched for 4296 labels. Next, we joined the labels and features to form a complete dataset.

## (2) Preprocessing

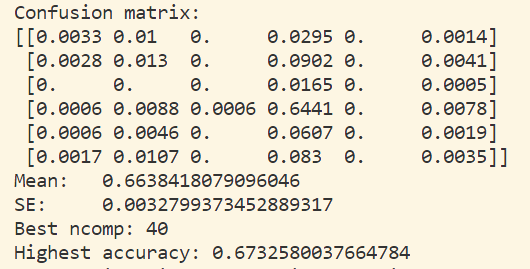
The raw dataset needs to be preprocessed. The preprocessing function first filled na value with the mean of the feature. There were two useless labels for training to be removed, “dd” for data deficient and “Remove” for no result. In addition, the function removed data with too few corresponding labels. We also made category features into one-hot features for better training results. This led to too many features, so we performed a PCA to select the main features, last the dataset was split into a training set and a testing set.

The total models we used are SVM, BPNN(Back Propagation Neural Network) , KNN(K-Neighbours) and RF(Random Forest). With the 2 methods we improved , we increased the average accuracy from 57% to 67%.

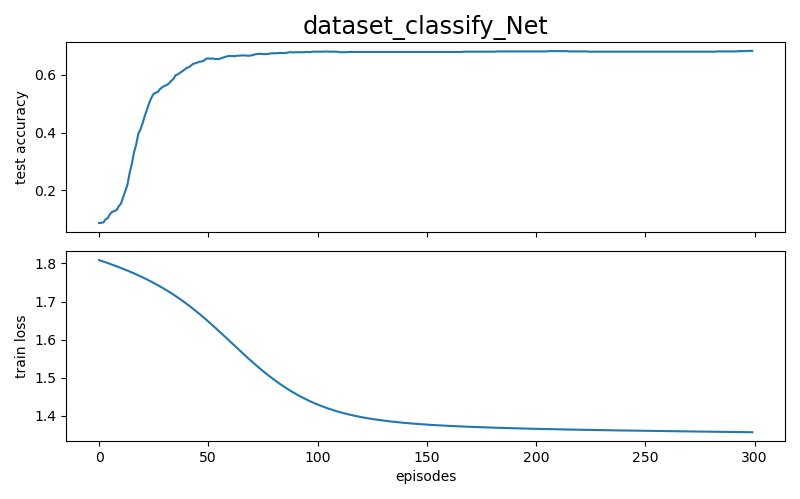
# 3. Training

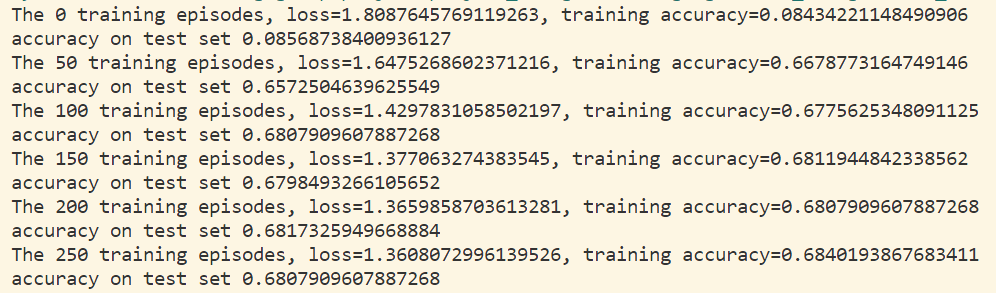
Before training, we set a threshold of 80% for our models. We hope they could reach to on average 80% of accuracy.

For training, we used a back propagation neural network (BPNN). as the main model. Before training, we performed a cross-validating using logistic regression to find the optimal PC number. The total feature number is 57, and we select [10, 20, 25, 30, 40, 57] as the testing PC numbers. When PC number = 40, we got the best result:



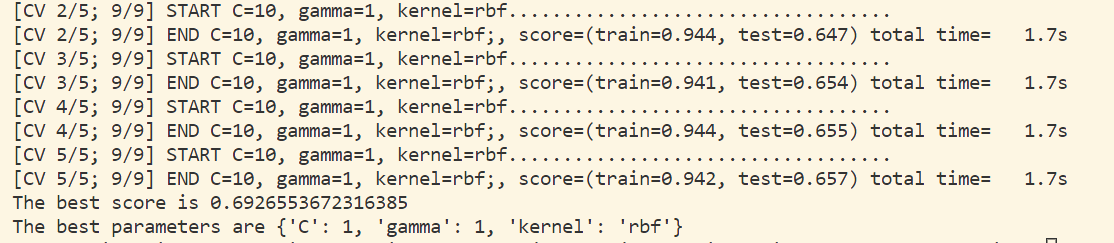
To train our BPNN, we first created a BPNN class. The BPNN had one hidden layer and an output layer with 30 hidden units in each hidden layer. The hidden layer used the RELU activation function and since this is a multi-class classification task, we used the softmax activation function for the output. In terms of the optimizer, after testing, we found that the Adam optimizer had the best performance. The training process contained 300 episodes and 50 episodes in each epoch. For each epoch, the training and testing accuracy and loss were printed on the console. At last, we got the highest accuracy of 68.07%.



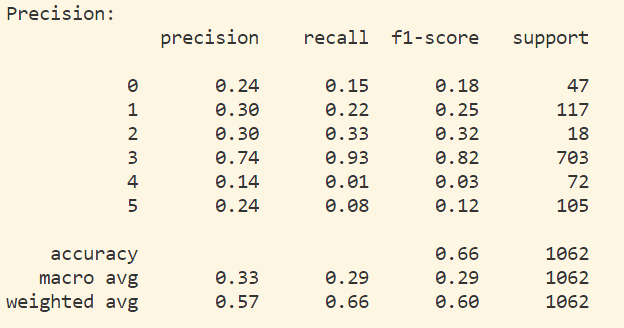


Apart from logistic regression and BPNN, to get the best result, we tried several other classification models, such as SVM, K-nearest neighbors (KNN), and random forest (RF). Generally speaking, they have similar performances.

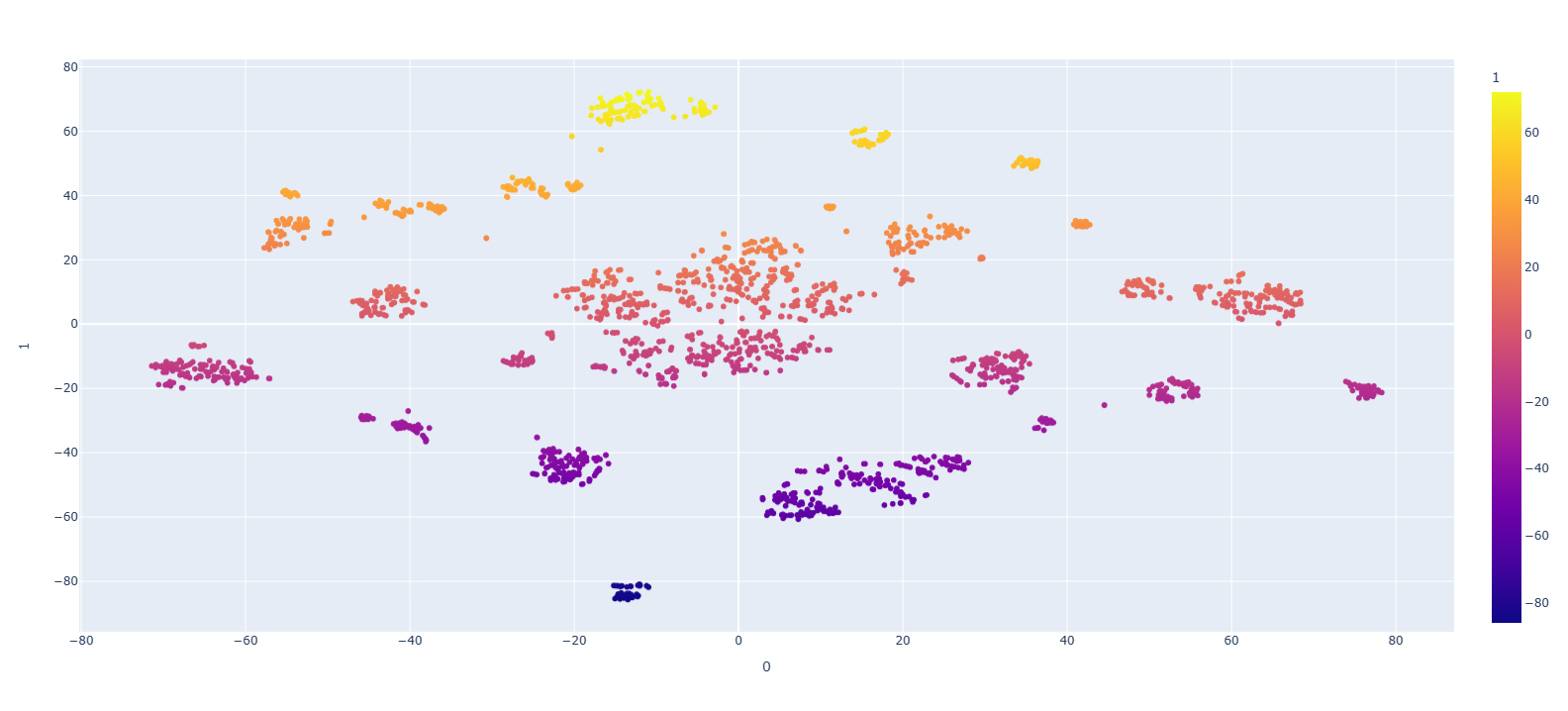
We performed a simple SVM classifier to do the prediction. To select the best C and gamma, we did a cross-validation and found the best test score. It was the highest accuracy we had gotten (69.3%).



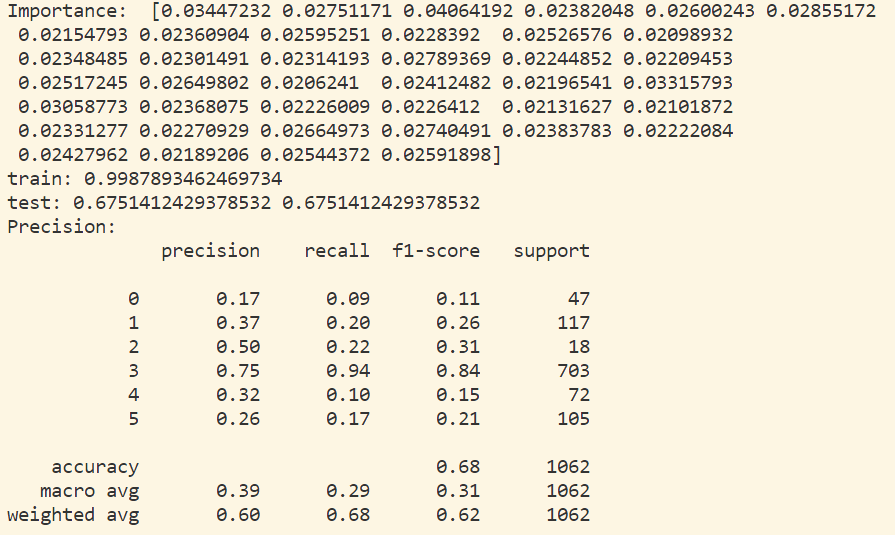
Similarly, we built a simple KNN classifier with 6 neighbors since our samples were predicted to be one of the six classes. The plotted classification metrics showed that it did not perform better but similarly.



Before performing RF, we did a T-distributed stochastic neighbor embedding (TSNE) to reduce and visualize the data samples. Our code plotted a RF scatter graph to help us to visualize the clusters.



The feature importance matrix showed that all features were not quite relevant. Like KNN, printed the classification metrics. Their performance was quite similar.

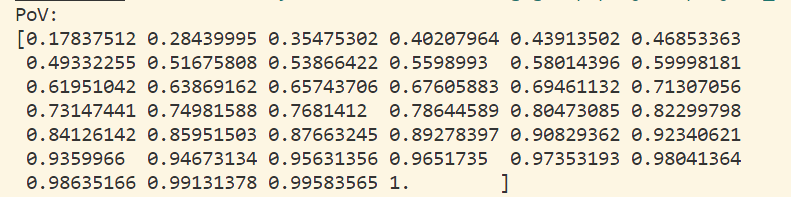


# 4. Performance Analysis

None of our models achieved our expectations. In general, they all have 67% accuracy and are not higher than 70%.

One of the potential problems could be the imbalance of data. We checked that among 3500 valid samples, 2300 of them had “lc” labels, and most of the other samples have about 100-200 samples. This resulted in high accuracy in predicting species with “lc” labels, and wrong results on other species. This can be visualized from the performance metrics, where label 3 always had an outstanding prediction accuracy.

Meanwhile, the low relevance of features affected the result. From the PoV table, we can see even the PC with the highest variance made little difference in samples. Besides, the importance matrix shows that all features have low importance, which means PCA did not help. From BPNN’s accuracy plot we know that the network converged quickly. This indicates that the features did not provide much information to train.



Lastly, we used a limited number of samples since label fetching is time-consuming. There are about ⅓ samples that haven’t been used.

# 5. Conclusion

To sustain human life and protect the ecosystem, we start the project to use machine learning models to predict species’ endangered level. We improved the preprocessing process and tried several machine learning models and saw a performance enhancement from 57% accuracy to 69% accuracy. We discussed multiple issues. I hope our project made some contribution to protecting animal species diversity.

# Reference:

[1]: [Python机器学习笔记：随机森林算法 - 战争热诚 - 博客园 (cnblogs.com)](https://www.cnblogs.com/wj-1314/p/9628303.html)

[2]: [COMBINE: a coalesced mammal database of intrinsic and extrinsic traits - Soria - 2021 - Ecology - Wiley Online Library](https://esajournals.onlinelibrary.wiley.com/doi/full/10.1002/ecy.3344#support-information-section)

[3]: [IUCN Red List of Threatened Species](https://www.iucnredlist.org/)