Shelter Animal Outcome Classification Using Decision Tree, Random Forest, Logistic Regression and Naive Bayes

Individual Final Report – Lydia Shia

Project Introduction:

This project is to explore the life expectancy of shelter animals. Dataset was collected in 2016 from an Austin Animal Shelter. The primary types of animals in the shelter are dogs and cats, which are the most common domestic pets. The goal of this project is to explore the connection between animal characteristic to its likelihood to survive in a shelter.

Each analyst was assigned with work to understand and to implement models for prediction by stages of analysis. The study carried through in the order of data exploration, preprocessing and model selection and analysis.

Individual and Shared Work:

Upon learning that the response variable is in category format we selected Naïve Bayes, Decision Tree, Logistic Regression and Random Forest as our models. As a team member, I developed and was responsible for:

- Proposal with Yuke
- Abstract
- Introduction
- Study Design/ Structure of the project
  - Defined final project focus items/sections by common Machine Learning report criteria
- Exploratory analysis (data visualization)
  - Figures by variables and explanations
- Preprocessing part 2 (most variables were overlaps with 1st preprocessing by Yuke, except feature "Main Breed")
- Naïve Bayes model
- Github creation, uploads and maintenance
- Exploration of Outcome Subtype to animal life expectance. (Refer to readme.md under my individual folder)

Result Discussion:

The dataset consists mainly of categorical variables. After preprocessing, some variables were removed and some has been expanded to increase dimensionality. To efficient model process we designed 2 types of label variables as of the following:

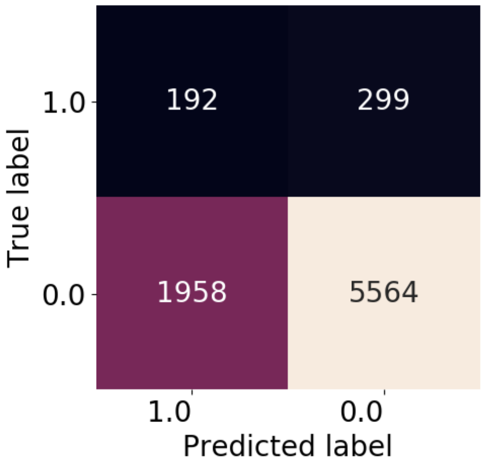
Original outcome variable levels(Outcome):

Level	Label	Count
Adoption	0	10769
Transfer	1	9406
Return_to_owner	2	4785
Euthanasia	3	1553
Died	4	197

Target	Level	Label	Count
Survived	Adoption + Transfer + Return_to_owner	1	24960
Died	Euthanasia + Died	0	1750

I started with exploratory analysis by examining distribution per old and new features to have a sense of which features will be most important. However, from the aggregated outcome the dataset is inevitably unbalanced.

We first ventured into Naïve Bayes before One Hot encoding, which we anticipated will further increase dimensionality. Naïve Bayes' capability in making predictions based on event probability seems to fit the criteria of our data structure because most variables are recorded in counts. I then chose multinomial as the model class to account for the categorically dominated dataset. The result is promising on binary target outcome. However, accuracy reduced tremendously when multi level outcome is tested. It is obvious that the assumed class conditional independence caused the loss of accuracy.



Classification Report when *Target* was target

target	Precision	Recall	F1-score	support
0	0.09	0.95	0.15	491
1	0.95	0.74	0.83	7522
avg/total	0.90	0.72	0.79	8013
Accuracy	71.83%			
ROC_AUC	59.24%			

The group then moved on to tree models. Starting from Decision Tree to include all created features since Decision Tree allows for high dimensionality . It is used commonly for large dataset, allowing for feature reduction. The model is optimized when label variable only has 2 levels, and was able to give insight on the most important features such as the feature ageC_0 (young) and HaveName_1 (animal with names). These are the variables have the most impact on model predictivity. We then moved on to Random Forest as a way to ensemble decision tree. The yielded outcome/accuracy were similar. However, after tuning the most important feature appears to be HaveName and fertility.

Aware of our binary outcome, we again tested the data with logistic regression model, and obtained the most optimal AUC. This is largely due to how we grouped our data into binary form. When we attempted with multilevel classes the outcomes are way less desirable. Therefore, it also suggest we may have overfitted our model by assigned a generalized binary outcome level.

What I have learned from this project is that knowing the data structure and the theory behind model algorithm is the key for training. As a starter, it is easy to always probe for an answers with trials and errors. However, just implementing the method without understanding will not help the knowledge to stay. It may require a considerable time and persistence to fully grasp the material, but the reward will be everlasting.

Code Evaluation

Nb.py about 30% self contribution, the code was downloaded from professor's example.

Preprocessing2.py about 60% self contribution, 40% of code was a combination of Kernal codes.

Code for exploration on Preprocessing and README.md is about 80% self contribution.

Reference

Various sites from : https://medium.com/
https://stackoverflow.com/
https://www.kaggle.com/apapiu/visualizing-breeds-and-ages-by-outcome
https://www.kaggle.com/zoupet/clean-feat-fit-and-submit
https://www.kaggle.com/apurvanaik/feature-engineering-rf-predict-outcome
https://machinelearningmastery.com/