SHELTER ANIMAL

OUTCOMES

Feuyang Tekeu Vanessa

Motis 8

04/05/2017

About The Competition

Every year, approximately 7.6 million companion animals end up in US shelters. Many animals are given up as unwanted by their owners, while others are picked up after getting lost or taken out of cruelty situations. Many of these animals find forever families to take them home, but just as many are not so lucky. 2.7 million dogs and cats are euthanized in the US every year.

Data Introduction

The data comes from Austin Animal Center from October 1st, 2013 to March, 2016. Outcomes represent the status of animals as they leave the Animal Center. All animals receive a unique Animal ID during intake.

In this competition, using a dataset of intake information including breed, color, sex, and age from the Austin Animal Center, we are asked to predict the outcome of the animal as they leave the Animal Center. These outcomes include **Adoption, Died, Euthanasia, Return to owner,** and **Transfer.**

The train and test data are randomly split.

File descriptions

- train.csv - the training set
- test.csv - the test set
- sample_submission.csv - a sample submission file in the correct format

Procedure

Train Dataset

In order to make a model, I first decided to prepare my dataset. Initially, the train dataset was made of the following column: AnimalID, Name, DateTime, OutcomeType, OutcomeSubtype, AnimalType, SexuponOutcome, AgeuponOutcome, Breed, and Color. I went through a Visual Analysis recipe to clean my dataset and keep only the data that I thought will be useful to achieve my goal of predicting the outcome of Animals.

In the Train Visual Analysis Recipe, I took the following steps:

- ✓ I deleted the column AnimalID, Name, OutcomeSubtype since they would not have help me find Animals' outcome.

- ✓ I split the SexuponOutcome (as it was a mixture of animals' sex and animal's status) column into 2 different columns and I renamed the columns as Sex and Status
- ✓ I split the Breed column as some animals were mix(/) and I used the trunk option to keep only the first Breed of animals who were mix. I did this step in order to reduce the type of animals' Breed, as it was too vast.
- ✓ I split the Color column as some animals were having more than one color (/) and I used the trunk option to keep only the first Color of animals who were mix. I did this step in order to reduce the type of animals' colors, as it was also too vast like in the Breed column.
- ✓ I parsed DateTime so that I could extract information from it(Year, Month, DayofWeek, Hour)
- ✓ Using the extracted Hour from the DateTime, I was able to create the column TimeofDay who give us an insight into the period when animals are mostly either adopted, transfer…(morning, midday', lateday', 'night)
- ✓ I split the AgeuponOutcome (as it was a mixture of an Integer, the value and the period) column into two different columns and I renamed the columns as TimeValue and UnitValue. I replace the "S" In some of the UnitValue field by nothing to have a singular value in every rows.
- ✓ I created a Mutiplier column using the field in the Unitvalue considering that year=365, Month=30, Week=7 and Day=1
- ✓ I then used the TimeValue and Multiplier value that I previously created to the find Age of the animals. This data will help us in knowing or example if younger animals have more chances to be adopted than older one for example.

So After deploying my recipe my new train dataset (trainShelter_Prepared) is made of the following column: Year, Month, DayofWeek, Hour, TimeofDay, OutcomeType, AnimalType, Status, Sex, TimeValue, UnitValue, Mutiplier, Age, Breed and Color.

Test Dataset

In order to test my model, I had to reproduce the same step that I did on the training dataset on the test one. Initially, the test dataset was made of the following column: ID, Name, DateTime, AnimalType, SexuponOutcome, AgeuponOutcome, Breed, and Color. I went through a Visual Analysis recipe to clean my dataset and keep only the data that I thought will be useful to achieve my goal of predicting the outcome of Animals.

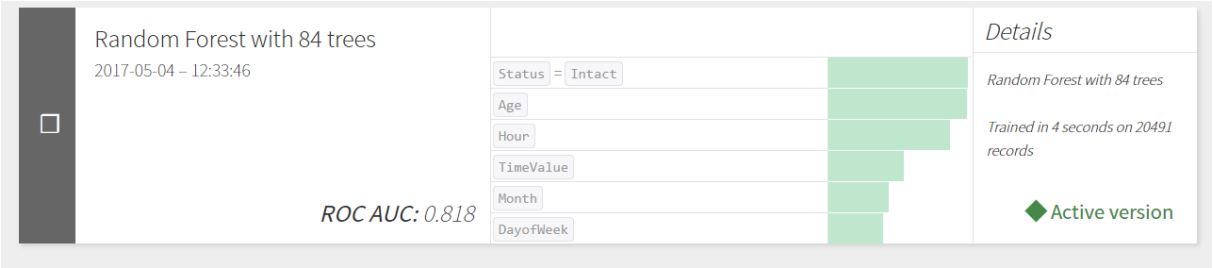
In the Test Visual Analysis Recipe, I took the following steps:

- ✓ I deleted the column ID, Name since they would not have help me find Animals' outcome.
- ✓ I split the SexuponOutcome (as it was a mixture of animals' sex and animal's status) column into 2 different columns and I renamed the columns as Sex and Status
- ✓ I split the Breed column as some animals were mix(/) and I used the trunk option to keep only the first Breed of animals who were mix. I did this step in order to reduce the type of animals' Breed, as it was too vast.
- ✓ I split the Color column as some animals were having more than one color (/) and I used the trunk option to keep only the first Color of animals who were mix. I did this step in order to reduce the type of animals' colors, as it was also too vast like in the Breed column.
- ✓ I parse DateTime son that I could extract information from it(Year, Month, DayofWeek, Hour)
- ✓ Using the extracted Hour from the DateTime, I was able to create the column TimeofDay who give us an insight into the period when animals are mostly either adopted, transfer…(morning, midday, lateday, 'night)
- ✓ I split the AgeuponOutcome (as it was a mixture of an Integer, the value and the period) column into two different columns and I renamed the columns as TimeValue and UnitValue. I replace the "S" In some of the UnitValue field by nothing to have a singular value in every rows.
- ✓ I created a Mutiplier column using the field in the Unitvalue considering that year=365, Month=30, Week=7 and Day=1
- ✓ I then used the TimeValue and Multiplier value that I previously created to the find Age of the animals. This data will help us in knowing or example if younger animals have more chances to be adopted than older one for example.

So After deploying my recipe my new test dataset (test_Prepared) is made of the following column: Year, Month, DayofWeek, Hour, TimeofDay, OutcomeType, AnimalType, Status, Sex, TimeValue, UnitValue, Mutiplier, Age, Breed and Color.

Model and Prediction

Since my two (trainShelter_Prepared and test_Prepared) datasets were ready, I decided to train my models using the trainShelter_Prepared dataset. I predicted the model on the OutcomeType column using the multiclass classification. I used three different Algorithms: Random tree forest, XGBOST and the Decision tree. The Random tree Forest had the best score so I decided to deploy it.



Then, I apply my Shelter Prediction Random Forest model on the trainShelter_Prepare Test dataset and I obtained the test_prepared_scored dataset in which you can find all the relevant probabilities and prediction. Finally based on the test_prepares_scored.csv document that I exported from Kaggle, I submitted my result on kaggle where I obtained the following score