Shelter Animal Outcomes (Kaggle)

using R

By:
Anish Anand

Shelter Animal Outcomes

Business Problem/Objective

Help Improve Outcomes for Shelter Animals

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.
Using a dataset of intake information including breed, color, sex, and age from the Austin Animal Center, we're asking you to predict the outcome for each animal.
We also believe this dataset can help us understand trends in animal outcomes. These insights could help shelters focus their energy on specific animals who need a little extra help finding a new home.
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, you are going to predict the outcome of the animal as they leave the Animal Center.
These outcomes include:
Adoption, Died, Euthanasia, Return to owner, and Transfer.

Submissions are evaluated using the multi-class logarithmic loss.

Shelter Animal Outcomes

Data Availability

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. The train and test data are randomly split. The analysis has to be performed using the following file format given:

1. **train.csv :** This csv(comma separated value) file is the given training set on which we have to train our machine learning model. The dimension of file is **[26729rows*10columns]**. The file contains following fields :

VARIABLE DESCRIPTIONS:

- *AnimalID* – unique id of each animal
- *Name*
- *DateTime* – Time at which animal was taken in the center
- *OutcomeType* – The outcome is a categorical variable which shows what wasexactly done with the animal.
  It takes 5 factors namely :
  Adoption, Died, Euthanasia, Return to owner, and Transfer

train.csv

AnimalID	Name	DateTime	OutcomeType	OutcomeS	AnimalType	SexuponOutcome	AgeuponOutcome	Breed	Color
A671945	Hambone	2/12/2014 18:22	Return_to_owner		Dog	Neutered Male	1 year	Shetland Sheepdog Mix	Brown/White
A656520	Emily	10/13/2013 12:44	Euthanasia	Suffering	Cat	Spayed Female	1 year	Domestic Shorthair Mix	Cream Tabby
A686464	Pearce	1/31/2015 12:28	Adoption	Foster	Dog	Neutered Male	2 years	Pit Bull Mix	Blue/White
A683430		7/11/2014 19:09	Transfer	Partner	Cat	Intact Male	3 weeks	Domestic Shorthair Mix	Blue Cream
A667013		11/15/2013 12:52	Transfer	Partner	Dog	Neutered Male	2 years	Lhasa Apso/Miniature Poodle	Tan
A677334	Elsa	4/25/2014 13:04	Transfer	Partner	Dog	Intact Female	1 month	Cairn Terrier/Chihuahua Shorthair	Black/Tan
A699218	Jimmy	3/28/2015 13:11	Transfer	Partner	Cat	Intact Male	3 weeks	Domestic Shorthair Mix	Blue Tabby
A701489		4/30/2015 17:02	Transfer	Partner	Cat	Unknown	3 weeks	Domestic Shorthair Mix	Brown Tabby
A671784	Lucy	2/4/2014 17:17	Adoption		Dog	Spayed Female	5 months	American Pit Bull Terrier Mix	Red/White
A677747		5/3/2014 7:48	Adoption	Offsite	Dog	Spayed Female	1 year	Cairn Terrier	White
A668402		12/5/2013 15:50	Transfer	SCRP	Cat	Unknown	2 years	Domestic Shorthair Mix	Black
A666320		11/4/2013 14:48	Adoption		Dog	Spayed Female	2 years	Miniature Schnauzer Mix	Silver
A684601	Rocket	2/3/2016 11:27	Adoption	Foster	Dog	Neutered Male	4 years	Pit Bull Mix	Brown
A704702	Scooter	6/8/2015 16:30	Return_to_owner		Dog	Neutered Male	2 years	Yorkshire Terrier Mix	Black/Red
A688584	Preston	11/25/2015 15:00	Return_to_owner		Dog	Neutered Male	1 year	Great Pyrenees Mix	White/Cream
A678825	Oliver	7/12/2014 12:10	Adoption	Foster	Cat	Neutered Male	3 months	Domestic Shorthair Mix	Orange Tabby/White
A678050		5/3/2014 16:15	Transfer	Partner	Cat	Intact Male	3 weeks	Domestic Shorthair Mix	Brown Tabby
A680028		6/7/2014 12:54	Transfer	Partner	Dog	Intact Female	2 weeks	Pit Bull Mix	Brown/White
A679010	Chrissy	5/17/2014 11:32	Transfer	Partner	Cat	Intact Female	2 years	Angora Mix	White

Shelter Animal Outcomes

- *OutcomeSubtype* : subtype of outcome.
- *AnimalType* : Type of animal admitted- either Dog or Cat
- *SexuponOutcome* : It denotes sex of the animal and whether it is neutered or not.
- *AgeuponOutcome* : age of animal till the date
- *Breed*
- *Color*

2. test.csv : The testing set is a CSV(comma separated value) file and this file is, on which the machine learning model trained on training set has to be tested. The dimension of file is **[11456rows*8columns].**However, both sets contain almost same fields, but the set does not have *OutcomeType and OutcomeSubtype* in the set .
This is what, has to be found using the model.

3. sample_submission.csv : The submission format is given on the adjacent figure. This is how the problem demands the output to be.

test.csv

ID	Name	DateTime	AnimalType	SexuponOutcome	AgeuponOutcome	Breed	Color
1	Summer	10/12/2015 12:15	Dog	Intact Female	10 months	Labrador Retriever Mix	Red/White
2	Cheyenne	7/26/2014 17:59	Dog	Spayed Female	2 years	German Shepherd/Siberian Husky	Black/Tan
3	Gus	1/13/2016 12:20	Cat	Neutered Male	1 year	Domestic Shorthair Mix	Brown Tabby
4	Pongo	12/28/2013 18:12	Dog	Intact Male	4 months	Collie Smooth Mix	Tricolor
5	Skooter	9/24/2015 17:59	Dog	Neutered Male	2 years	Miniature Poodle Mix	White
6	Beau	6/23/2015 11:17	Dog	Neutered Male	3 years	Beagle Mix	Brown/White
7	Bobo	3/12/2014 9:45	Cat	Neutered Male	13 years	Domestic Medium Hair Mix	Brown Tabby/White
8	Abby	6/25/2014 8:27	Cat	Spayed Female	6 months	Domestic Shorthair Mix	Brown Tabby
9	Ruby Grace	11/12/2014 18:05	Dog	Spayed Female	3 months	Cairn Terrier	Black/Cream
10	Ruby	4/7/2014 17:41	Dog	Spayed Female	1 year	Pit Bull Mix	Brown/White
11	Marlee	1/1/2014 17:03	Dog	Spayed Female	5 months	Labrador Retriever Mix	White/Black
12	Lala	9/19/2014 15:40	Dog	Intact Female	2 years	Pit Bull Mix	Blue/White
13	Chloe	1/25/2015 15:13	Dog	Spayed Female	6 months	Pit Bull Mix	Brown Brindle
14		9/22/2015 18:17	Dog	Neutered Male	2 years	Miniature Poodle Mix	White
15	Duffy	7/20/2014 19:24	Dog	Neutered Male	1 year	Flat Coat Retriever/German Shepherd	Black/Brown
16	Jiggy	3/26/2014 18:33	Dog	Spayed Female	4 months	Pointer/Labrador Retriever	White
17	Ghengis	5/29/2015 17:04	Dog	Neutered Male	5 years	Doberman Pinsch Mix	Black/Tan
18	Decker	1/19/2016 19:31	Dog	Neutered Male	3 years	English Bulldog	White/Tricolor
19	Freckles	6/16/2014 12:49	Cat	Spayed Female	3 years	Domestic Shorthair Mix	Torbie
20		2/11/2014 19:27	Cat	Intact Female	4 years	Domestic Shorthair Mix	Torbie
21	Pumpkin	11/14/2014 19:32	Cat	Spayed Female	2 years	Domestic Shorthair Mix	Black

sample_submission.csv

ID	Adoption	Died	Euthanasi	Return_to	Transfer
1	1	0	0	0	0
2	1	0	0	0	0
3	1	0	0	0	0
4	1	0	0	0	0
5	1	0	0	0	0
6	1	0	0	0	0
7	1	0	0	0	0
8	1	0	0	0	0
9	1	0	0	0	0
10	1	0	0	0	0
11	1	0	0	0	0
12	1	0	0	0	0
13	1	0	0	0	0
14	1	0	0	0	0
15	1	0	0	0	0
16	1	0	0	0	0

Prepared by : Anish Anand

Shelter Animal Outcomes

Approach

Overview

In this problem, the goal is to use a dataset on shelter animals to do two things:

- gain insights that can potentially improve their outcome, and
- to develop a classification model which predicts the outcome of animals (adoption, died, euthanasia, return to owner, transport.

Understanding what our goal is?

The aim is to develop a model which is useful for the shelter. What I believe is that the shelter would use the model when a new animal arrives. This is the time when the shelter wants to know the likely outcome of the new animal. A classifier that predicts the outcome of animals at the time of their intake can help in several ways:

- If the adoption probability of an animal is low, the shelter employees can focus on those animals and give them a little extra help finding a new home. The extra help could be neutering/spaying the animal. The shelter could also lower the adoption fee of such animals.
- With a slight modification, the model can help with logistic planning. If the model predicts how long the animals stay in the shelter (instead of predicting the likely outcome), one could better plan and allocate resources for animals that are likely to stay long.

Shelter Animal Outcomes

Gaining Insight and Collecting features

From seeing the dataset, what I can perceive is that, upon the arrival of a new animal, the shelter has these types of available information:

- most people like to adopt young puppies and kitties, and the adoption probability of older animals is lower.
- People also have preferences for certain breeds. Not many people like specific dog breeds that are perceived aggressive (e.g. pitbull, rottweiler, bulldog).
- the properties of the animals which are currently in the shelter. If there are many young animals currently in the shelter, it probably lowers the adoption probability of old animals.

Some feature engineering on the dataset

Some new features can be added that might be helpful

- The first feature describes whether the animal has a name (1) or not (0).
- The second feature describes how many characters are there in the name.
- Finally, the third feature describe how frequently that name occurred in the training set.
- The features of breed and colour have around 100 categories, so we have to devise something to include that in our model, i.e by including same breeds or same colour types.
- The intake time features describe the year, month, day, and hour of the intake, which weekday was the intake (e.g. Monday or Tuesday), and which quarter of the year was the intake.
- The age of the animal can be converted into days(for uniformity).

Shelter Animal Outcomes

Packages Information

1. **lubridate –**

Functions to work with date-times and timespans: fast and user friendly parsing of date-time data, extraction and updating of components of a date-time (years, months, days, hours, minutes, and seconds), algebraic manipulation on date-time and timespan objects. The 'lubridate' package has a consistent and memorable syntax that makes working with dates easy and fun

Parsing dates

Lubridate's parsing functions read strings into R as POSIXct date-time objects. Users should choose the function whose name models the order in which the year ('y'), month ('m') and day ('d') elements appear the string to be parsed:
(dmy, myd, ymd, ydm, dym, mdy, ymd_hms).
Lubridate can also parse partial dates from strings into Period-class objects with the functions **hm, hms and ms**

Manipulating dates

Lubridate distinguishes between moments in time (known as instants) and spans of time (known as time spans). Time spans are further separated into
Duration-class, Period-class and Interval-class objects

Shelter Animal Outcomes

Time zones

Lubridate provides two helper functions for working with time zones. **with_tz** changes the time zone in which an instant is displayed. The clock time displayed for the instant changes, but the moment of time described remains the same. **force_tz** changes only the time zone element of an instant.

Durations

Durations measure the exact amount of time that occurs between two instants. This can create unexpected results in relation to clock times if a leap second, leap year, or change in daylight savings time (DST) occurs in the interval. Functions for working with durations include **is.duration, as.duration** and **duration**. **dseconds, dminutes, dhours, ddays, dweeks and dyears** convenient lengths

2. ggplot2

ggplot2 is a plotting system for R for data exploration and producing plots, based on the grammar of graphics, which tries to take some of the good parts of base and lattice graphics. It takes care of many of the fiddly details that make plotting a hassle (like drawing legends) as well as providing a powerful model of graphics that makes it easy to produce complex multi-layered graphics. It produces fantastic-looking graphics and allows one to slice and dice one's data in many different ways.

There are two major functions that we will use in ggplot2

Shelter Animal Outcomes

- **qplot() –** for quick plots. Works much like plot() function in base graphics system. Looks for data in a data frame, similar to lattice. Moreover, the plots are made up of aesthetics(size, shape, color) and geoms(points, lines)
- **ggplot() –** It is the core function and is very flexible for doing things that qplot() cannot do . It is used for fine, granular control of everything.

Let's quickly understand the structure of ggplot code:

- **aes –** refers to aesthetics. It includes variable names used to create plots.
- **geom_point –** ggplot offers many 'geoms' which are used to represent data. Since, we are interested here in scatter plot, we used geom_points.
- **scale_x_continuous –** x variable is continuous. This parameter is used to alter information represented at x axis.
- **scale_y_continuous –** It performs the same task as scale_x_continuous but for y axis.
- **theme_bw –** It refers to setting the background of plots. I used the grid version.

The trend is to start with normal plot function(or similar), i.e include the variable names on the plot and then use annotation functions to keep on adding new layers on the plot(text, line, points, axis, colors). Moreover, it has to be noted that we cant go back when the plot has started(i.e to adjust margins). Hence, we need a plan in advance, as to how are we going to add layers on the plot.

Installation of package

install.packages(ggplot2)

Shelter Animal Outcomes

Basic syntax

✓ qplot(x, y, ..., data, facets, margins = FALSE, geom = "auto", xlim = c(NA, NA), ylim = c(NA, NA), xlab, ylab)

- **x, y, ... -** Aesthetics passed into each layer
- **data** - Data frame to use (optional). If not specified, will create one, extracting vectors from the current environment.
- **facets** - like subplot in matlab. Multiple plots on same window corresponding to categorical var.
- **margins** -either a logical value or a character vector. Margins are additional facets which contain all the data for each of the possible values of the faceting variables. If FALSE, no additional facets are included (the default). If TRUE, margins are included for all faceting variables
- **Geom** - Character vector specifying geom(s) to draw. Defaults to "point" if x and y are specified, and "histogram" if only x is specified.

3. rpart (Recursive Partitioning)

The rpart programs build classication or regression models of a very general structure using a two stage procedure; the resulting models can be represented as binary trees. The types of endpoints that rpart handles includes

- classications (such as yes/no),
- continuous values (such as bone mineral density),
- poisson counts (such as the number of fractures in Medicare patients), and
- survival information (time to death/last known contact).

Shelter Animal Outcomes

The rpart library includes tools to model, plot, and summarize the end results.
Attach the library so that the functions can be found.
library(rpart)
Decide what type of endpoint you have

- Categorical ==> method == "class“
- Continuous ==> method == "anova“
- Poisson Process/Count ==> method == "poisson“
- Survival ==> method == "exp"

Fit the model
fit <- rpart(y ~ x1 + x2 +x3 …, data=data1, method='poisson')
Print a text version of the tree
print(fit)
Print a summary which examines each node in depth
summary(fit)

5. stringr
Strings are not glamorous, high-profile components of R, but they do play a big role in many data cleaning and preparations tasks.

Shelter Animal Outcomes

The stringr provides two basic families of functions:

- basic string operations, and
- pattern matching functions which use regular expressions to detect, locate, match, replace, extract, and split strings.

Basic string operations

There are three string functions that are closely related to their base R equivalents, but with a few enhancements:

- str_c() is equivalent to paste(), but it uses the empty string ("") as the default separator and silently removes NULL inputs.
- str_length() is equivalent to nchar(), but it preserves NA's (rather than giving them length 2) and converts factors to characters (not integers).
- str_sub() is equivalent to substr() but it returns a zero length vector if any of its inputs are zero length, and otherwise expands each argument to match the longest. It also accepts negative positions, which are calculated from the left of the last character. The end position defaults to -1, which corresponds to the last character.
- str_sub<- is equivalent to substr<-, but like str_sub it understands negative indices, and replacement strings not do need to be the same length as the string they are replacing.

Three functions add new functionality:

- str_dup() to duplicate the characters within a string.
- str_trim() to remove leading and trailing whitespace.
- str_pad() to pad a string with extra whitespace on the left, right, or both sides.

Shelter Animal Outcomes

5. xgboost (Extreme Gradient Boosting)

XGBoost is a library designed and optimized for boosting trees algorithms. The underlying algorithm of XGBoost is similar, specifically it is an extension of the classic gbm algorithm. By employing multi-threads and imposing regularization, XGBoost is able to utilize more computational power and get more accurate prediction.

There're various high-level interfaces. Currently there are interfaces of XGBoost in C++, R, python, Julia, Java and Scala. The core functions in XGBoost are implemented in C++.

The overall parameters have been divided into 3 categories by XGBoost authors:

I. **General Parameters:** Guide the overall functioning

- **booster** -Select the type of model to run at each iteration. (gbtree: tree-based models)or(gblinear: linear models)
- **silent -**Silent mode is activated is set to 1, i.e. no running messages will be printed.
- **nthread [default to maximum number of threads available if not set]-**used for parallel processing

II. **Booster Parameters:** Guide the individual booster (tree/regression) at each step

- **eta [default=0.3]**Analogous to learning rate in GBM
- **min_child_weight [default=1]**Defines the minimum sum of weights of all observations required in a child
- **max_depth [default=6]**The maximum depth of a tree, same as GBM
- **max_leaf_nodes**The maximum number of terminal nodes or leaves in a tree
- **alpha [default=0]**L1 regularization term on weight (analogous to Lasso regression)
- **lambda [default=1]**L2 regularization term on weights (analogous to Ridge regression)
- **max_delta_step [default=0]**In maximum delta step we allow each tree's weight estimation to be. If the value is set to 0, it means there is no constraint. If it is set to a positive value, it can help making the update step more conservative.

Shelter Animal Outcomes

III. Learning Task Parameters: These parameters are used to define the optimization objective the metric to be calculated at each step

- **objective [default=reg:linear]**This defines the loss function to be minimized. Mostly used values are:
  - **binary:logistic** –logistic regression for binary classification, returns predicted probability (not class)
  - **multi:softmax** –multiclass classification using the softmax objective, returns predicted class (not probabilities) you also need to set an additional **num_class** (number of classes) parameter defining the number of unique classes
  - **multi:softprob** –same as softmax, but returns predicted probability of each data point belonging to each class.
- **eval_metric [default according to objective]**The metric to be used for validation data.

  Typical values are:**rmse** – root mean square error
  - **mae** – mean absolute error
  - **logloss** – negative log-likelihood
  - **error** – Binary classification error rate (0.5 threshold)
  - **merror** – Multiclass classification error rate
  - **mlogloss** – Multiclass logloss
  - **auc:** Area under the curve
- **seed [default=0]**The random number seed

Shelter Animal Outcomes

6. Dplyr-

When working with data you must:

- Figure out what you want to do.
- Describe those tasks in the form of a computer program.
- Execute the program.

The dplyr package makes these steps fast and easy:

- By constraining your options, it simplifies how you can think about common data manipulation tasks.
- It provides simple “verbs”, functions that correspond to the most common data manipulation tasks, to help you translate those thoughts into code.
- It uses efficient data storage backends, so you spend less time waiting for the computer.

Dplyr aims to provide a function for each basic verb of data manipulation:

- filter() (and slice()) : filter() allows you to select a subset of rows in a data frame.
- arrange() : arrange() works similarly to filter() except that instead of filtering or selecting rows, it reorders them
- select() (and rename()):select()allows you to rapidly zoom in on a useful subset using operations that usually only work on numeric variable positions:

Shelter Animal Outcomes

•distinct() : Use distinct()to find unique values in a table:
•mutate() (and transmute()) : Besides selecting sets of existing columns, it's often useful to add new columns that are functions of existing columns. This is the job of mutate().
•summarise() : The last verb is summarise(). It collapses a data frame to a single row (this is exactly equivalent to plyr::summarise()):
•sample_n() (and sample_frac()) : You can use sample_n() and sample_frac() to take a random sample of rows: use sample_n() for a fixed number and sample_frac() for a fixed fraction

Shelter Animal Outcomes

Steps

1. Install and load the relevant packages.
2. Getting the Data Into R
   the data for the Animal shelter project is divided into two csv-format files:

- **train.csv** (data containing attributes and known outcomes for animals)
- **test.csv** (data containing attributes *without* outcomes for animals)

We import the above files in R using *read.csv(),* make the fields same and further bind the datasets by columns using *rbind()*, so that the changes or new variables creation in training set are created simultaneously in testing set.

3. Then we can also see relation of other fields by plotting using histogram, boxplot, etc , to get a clear picture of how the fields affect the outcome of animal.

4.Feature Engineering

It is the process of using domain knowledge of the data to create features that make machine learning algorithms work. We will use feature engineering in our set to create new features, i.e has name or not, male or female, intact or neutered,baby or adult,mix breed, time of day, how long is the name, and many more.

Shelter Animal Outcomes

5. If we see clearly, we can also see that there are missing values in age of animals. There are basically 3 approaches to deal with these types of problem :

- If number of missing values entries are very less as compared to that of size of data, we can remove them using *na.omit().*
- In other cases, we can replace the missing values using either mean, median or mode (seeing which of them could possibly relate to the distribution)
- However, the third case which we can use is to deploy an individual machine learning algorithm for finding missing values of the age of passenger using the relevant features. This will help in reducing error to great extent.

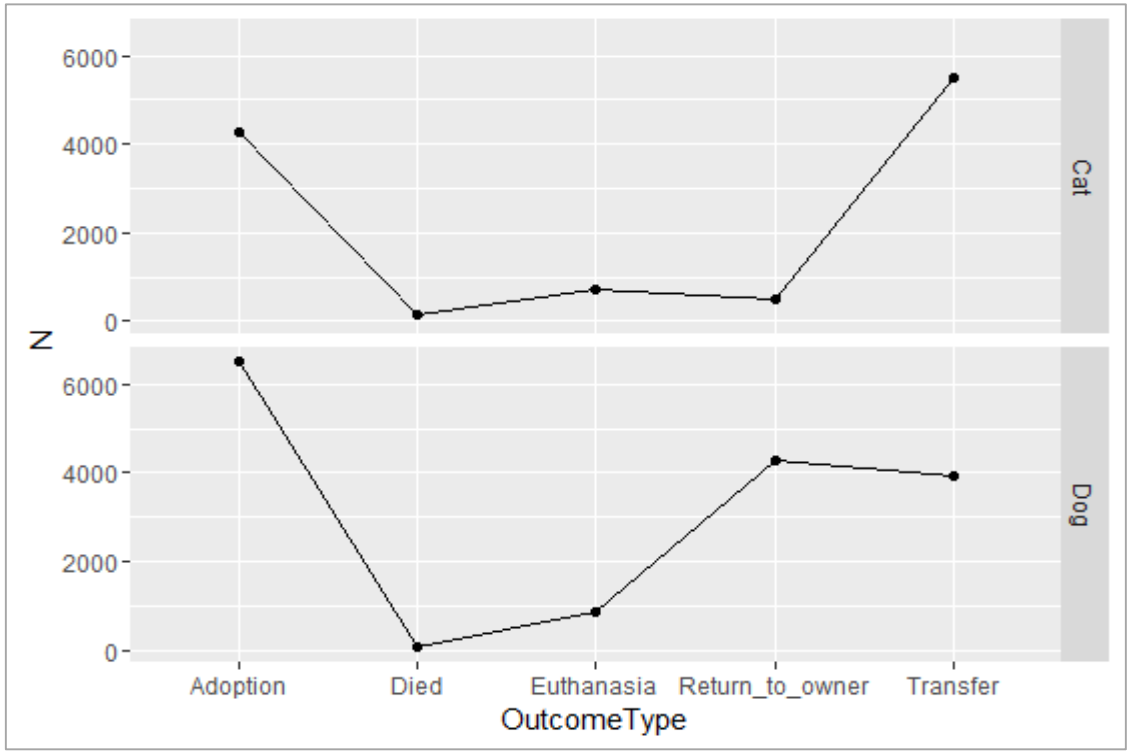
6. Fitting a Model

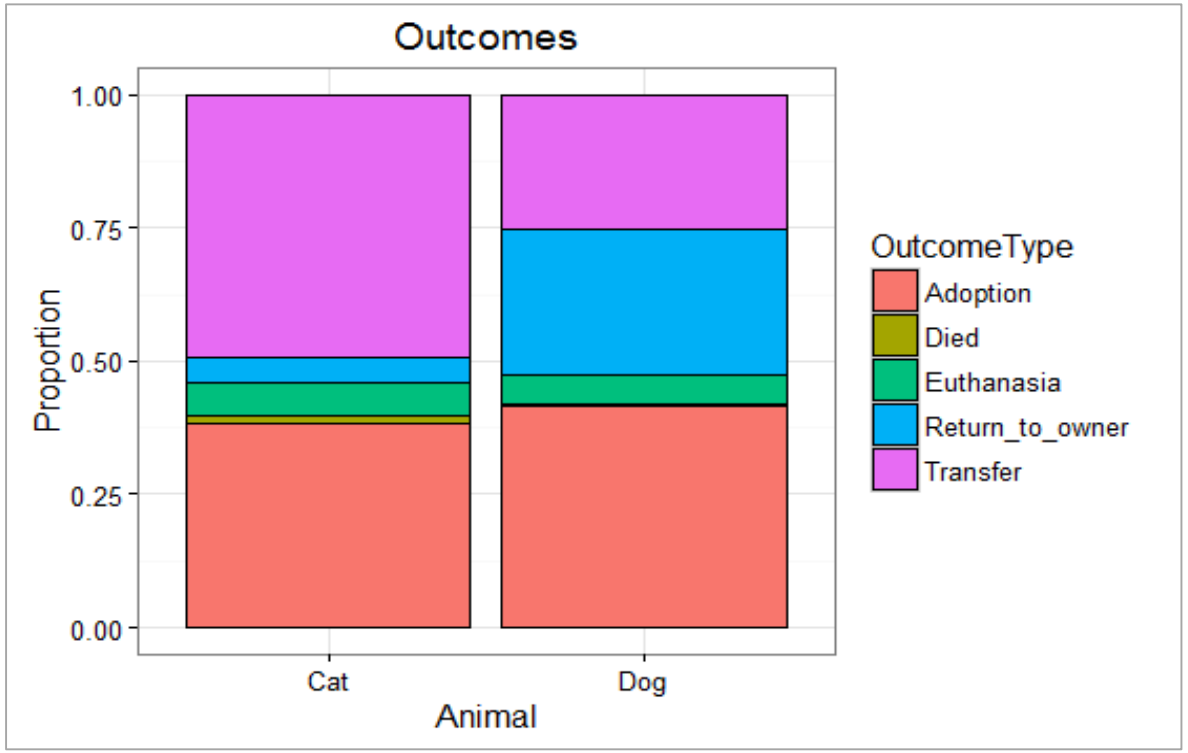
At the end, we will set a seed value and divide the training and testing set to their original self(number of rows). Then we will use a model for our data. There can be number of models and ensemble methods which can be deployed on a model.

Initially, we can use one of the simplest classification methods ,Logistic regression for this problem too. We can start simple by passing essentially the features provided in the raw training data through the R function for fitting general linearized models(glm). In my model I started with randomForest(to get a score of 2.7), however, using the xgboost with tuned parameters boosted up my score to 0.72(since the submissions are evaluated using the multi-class logarithmic loss.).

Shelter Animal Outcomes

Visualizations

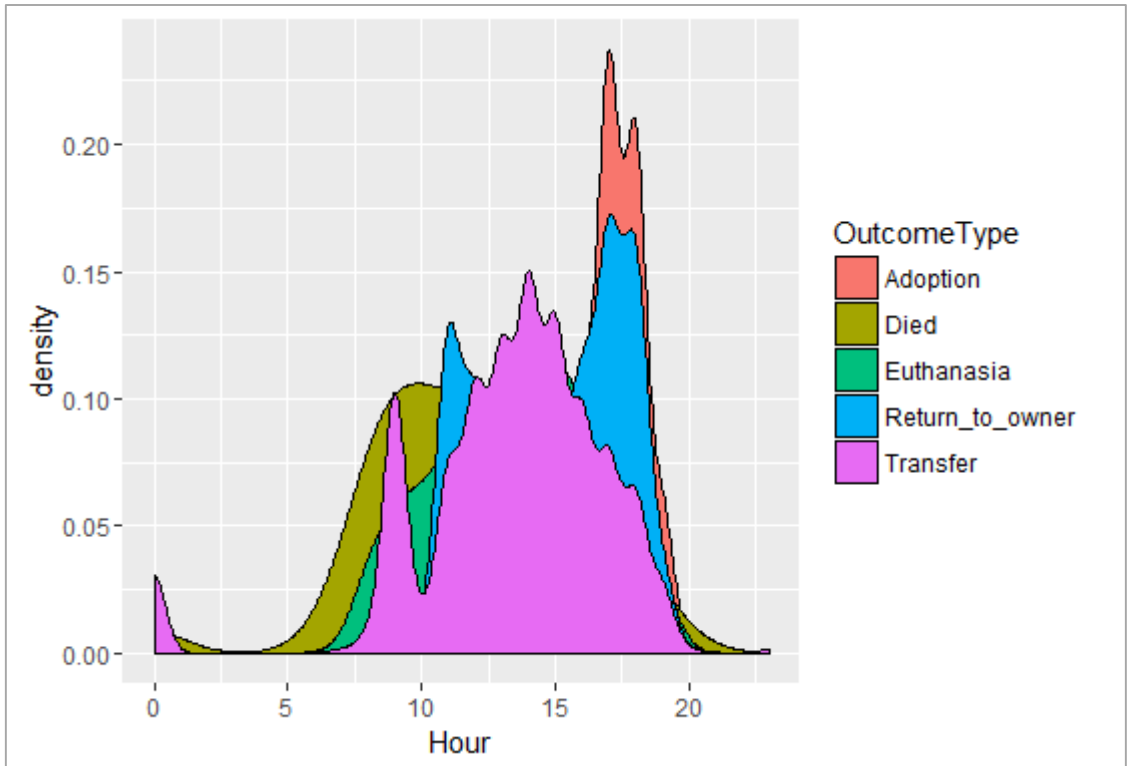




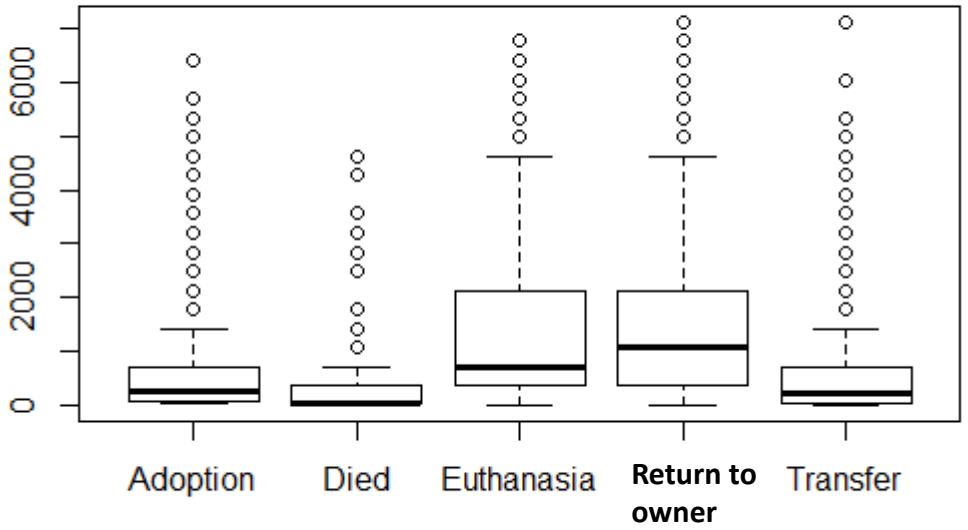
The above plots shows that dogs and cats are majorly adopted or transferred.
Moreover, when it comes to adoption, dogs are preferred over cats and in case of transfer, the situation is opposite.
Very few cases of died and euthanasia are seen, which in turn shows the effectiveness of firm in dealing with animals.

Prepared by : Anish Anand

Shelter Animal Outcomes



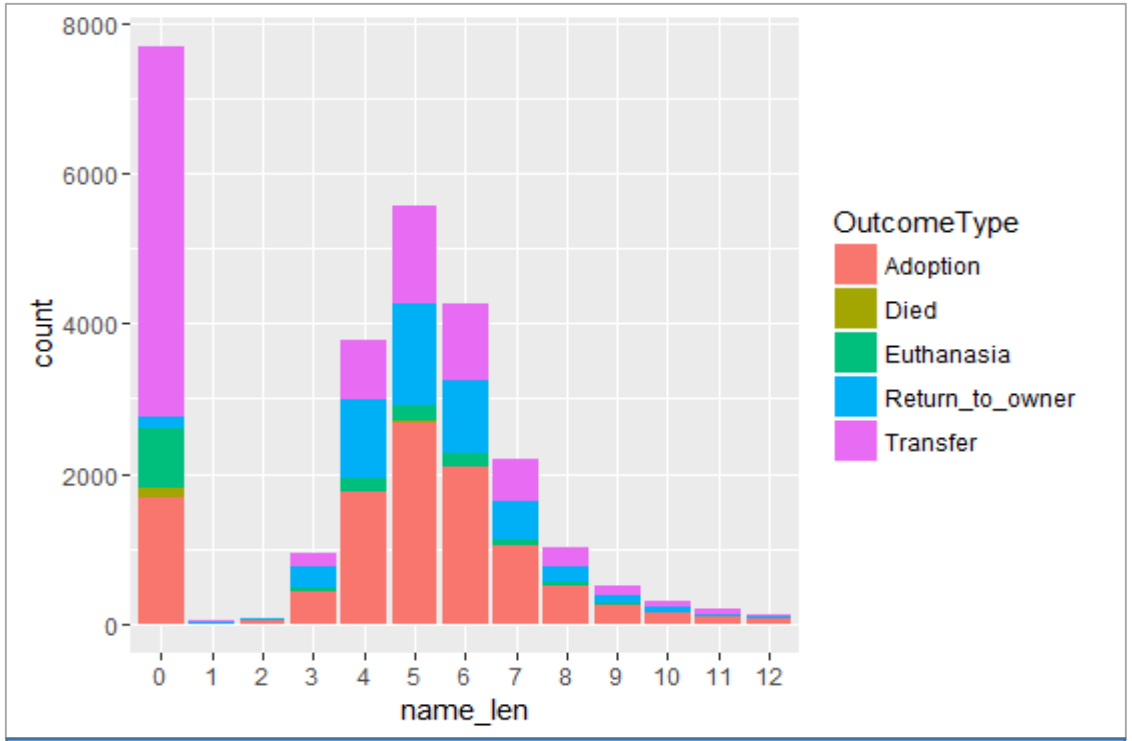
- Most cases of transfer are done between the day hours of 12:00-17:00.
- The adoption cases are registered mostly in the evening time(probably when people are free from work) between 16:00-19:00.



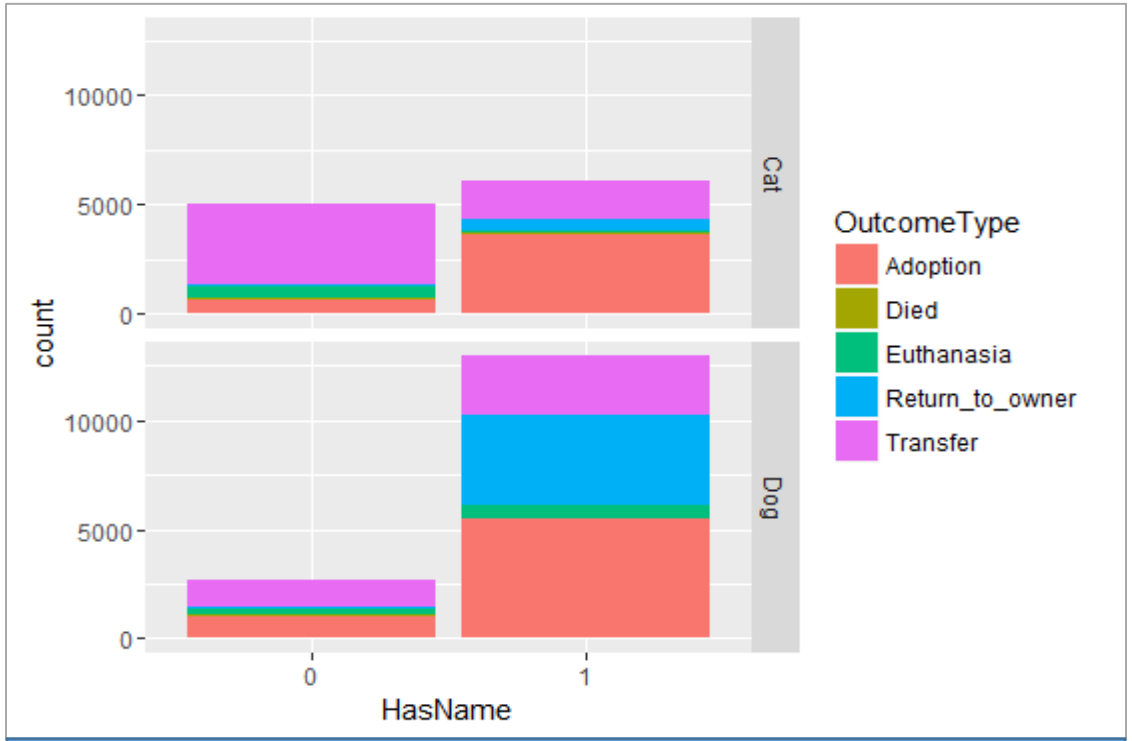
Most adoption cases are registered for young animals and many die at that age too.
Most adult animals are usually pets and are successfully returned to their owner

Prepared by : Anish Anand

Shelter Animal Outcomes



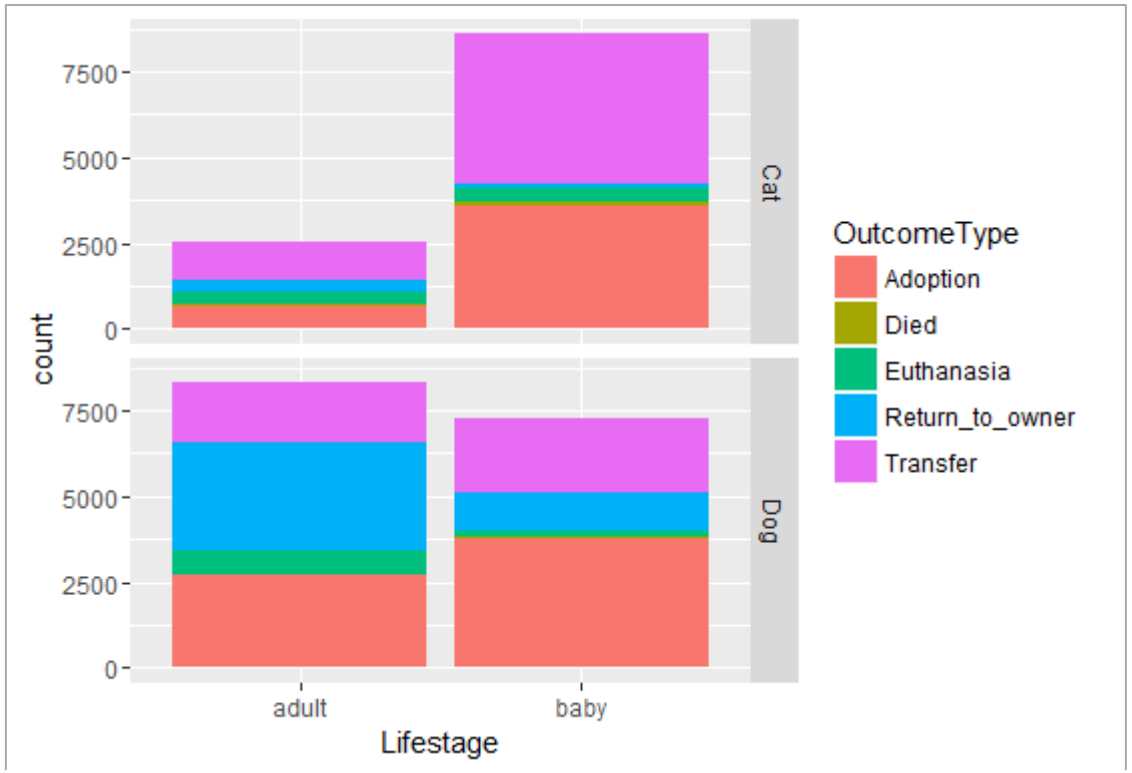
As it can be seen, length of the name of the animal does have an impact on the outcome.
The nameless are normally transferred and bigger names are not much in demand
However, the animals with moderate name length[4-7characters] are usually preferred by the people.



Most cats are seen nameless are compared to dogs.
Moreover, the cases of cats are normally registered for transfer(if it is nameless) ,whereas in case of dogs, the probability of transfer are less.
In most cases of dogs, if found, are usually adopted or returned to owner

Prepared by : Anish Anand

Shelter Animal Outcomes



Kittens are most preferred as compared to puppies. However, in terms of adult animals, the dogs are much preferred over cats.
Most cases of kittens and puppies are seen to be adoption or transfer.

The neutered animals have high chances of adoption. In case of transfer scenarios, there are mostly intact cats.
The cases of Died and euthanasia are pretty rare and mostly consist of intact animals

Prepared by : Anish Anand