Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

Project Name:

Classification, Prediction, and Exploratory Data Analysis of Animal Population and Outcomes in the Austin, TX Animal Center

Participants:

- Jana Voege: (JanaVoege@my.unt.edu)
- Keith Santamaria: (KeithSantamaria@my.unt.edu)
- Farhan Almufleh: (FarhanAlmufleh@my.unt.edu)
- Sri Lakshmi Tammineni: (SriLakshmiTammineni@my.unt.edu)

Workflow:

- After cleaning and preparation, we saved a common data set to a shared folder created in Outlook OneDrive.
- We created individual jupyter notebooks and combined them into one final one containing the code for selected models
- We collaborated on the report and presentation drafts using Google Drive
- We met (virtually) several times weekly to share ideas, collaborate on code, and review results

Project Abstract:

This project applied our design and milestones to intake and outcomes data from the Austin Animal Center (a municipal animal shelter). We sought to predict variables such as adoption potential and length of stay at the shelter, based on characteristics of the animal and/or the intake, which the shelter could use for planning purposes. The shelter could also utilize both outcome predictions and more descriptive/exploratory results to

Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

better target potential donors, adopters, breed-specific rescue partners, and intervention efforts such as spay/neuter resources.

We combined two datasets retrieved from the City of Austin's Open Data Portal (https://data.austintexas.gov/browse?q=animal&sortBy=relevance). One dataset includes intakes from October 1, 2013 through June 5, 2021. The second dataset includes animal outcomes from October 1, 2013 through June 5, 2021. We performed data cleaning and preparation using both Excel and Python. We identified missing and/or erroneous data and documented our decisions on how to treat these data elements. We documented our decisions on how to use data for animals who have only intake data (i.e., joined the shelter recently and are still there) and data for animals who have only outcomes data (i.e., joined the shelter before October 1, 2013), as these observations were suitable for some but not all analyses (e.g.., not suitable for predictive analyses).

We performed exploratory data analysis and descriptive statistics on the data, such as the distribution of different animal types (such as domestic, livestock, and wildlife), outcomes, ages, lengths of stay at the shelter, etc. We attempted both Supervised and Unsupervised techniques. Specifically, we developed a kNN model to Classify animals into the correct outcome (adoption or euthanasia) based on features of the animal and features of the intake situation. We also developed a Logistic Regression model to predict which animals would be successfully placed in a home. We determined that unsupervised Cluster Analysis was not useful, due to the lack of appropriate numeric predictor variables.

Project Design/Project Milestones:

1. Data Cleaning and Preparation (Using Excel and Python):

Since our data came from two separate data sets (one for Animal Intake data and one for Outcomes data), we first combined them (in Excel) to a single data set. The data set included 126,956 samples.

Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

We identified that both Intake data and Outcomes data contained two date/time columns (i.e., two Intake data/time columns and two Outcome date/columns) and that they were redundant. We carefully verified that the two columns contained identical data, then deleted one Intake date/time column and one Outcomes date/time column. We then separated the dates and times into separate columns for both Intakes and Outcomes. We deleted the merged date/time columns, resulting in one remaining column each for Intake Date, Intake Time, Outcomes Date, and Outcome Time.

We created a new feature (Length of Stay), defined as the difference (Excel DATEDIF) measured in Days between Intake Date and Outcome Date.

We created a new binary Y/N feature (Altered at Shelter), which we defined as anytime Sex at Intake did not equal Sex at Outcome.

We deleted the original Age at Intake feature, because it appeared to be a free text field and had inconsistent units of measure. For example, some values were expressed in days, some in weeks, some in months, and some in years. Some very young animals' ages were expressed in days, weeks, or months, while others were just expressed as "0 years." We created new features for the animals' ages at intake (Intake Age in Days, Intake Age in Weeks, Intake Age in Months, and Intake Age in Years), defined as the difference between Date of Birth and Intake Date. We created the corresponding new features for the animals' ages at outcome (Outcome Age in Days, Outcome Age in Weeks, Outcome Age in Months, and Outcome Age in Years).

We created a new feature (Collapsed Breed), in which we combined all the mixes of a particular primary breed. For example, a Dachshund/Chihuahua Mix, a Dachshund/Schnauzer Mix, and a Dachshund/ChowChow Mix would all be Dachshund Mix in the new Collapsed Breed variable. However, this still yielded several hundred unique values, making this feature not useful.

We identified that this data set's data quality was poor; there were several types of erroneous data, some of which were identified via the feature engineering described above. First, there were animals with negative Length of Stay (i.e., indicating they were discharged before they were taken in). Second, there were animals with negative Intake and/or Outcome ages (i.e., indicating they were taken in and/or discharged before their

date of birth). Third, there were animals identified as Spayed or Neutered at intake but Intact at discharge/outcome. Fourth, there were a large number whose Outcome was "Died in Surgery," whose Sex at Intake was Intact, and whose Sex at Outcome was Spayed or Neutered. This would suggest the animal died during a spay/neuter procedure, but their age data indicated they were too young for a spay/neuter procedure. All these are unresolvable issues with the data's cleanliness, and throughout the report (and commented in the code), we address how those issues were dealt with for various analyses.

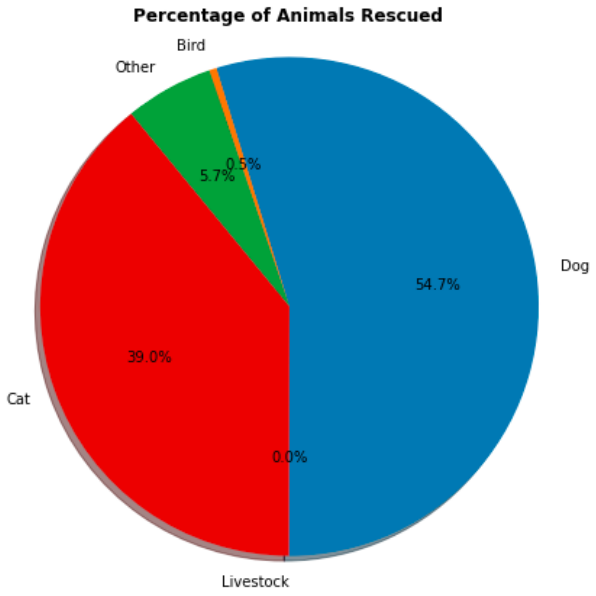
We had two other concerns with the data that were not errors but affected our analyses. One was that some animals were taken in before the start date of the data set (so they had only Outcomes data but no Intake data), and others were still at the shelter at the end date of the data set (so they had only Intake data but no Outcomes data). These animals' observations are useful for some but not all of the analyses. Second, we assume that many of the animals' Date of Birth values were estimates based on a veterinarian's impression of their approximate age, so all age data have a degree of uncertainty.

Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

2. Exploratory Data Analysis:

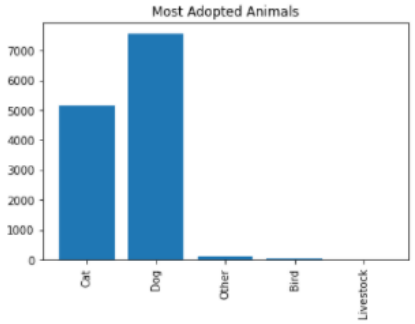
Although the Austin Animal Shelter took in a very interesting array of animals (including bats and "bat mixes," which are an Austin specialty), the vast majority (about 94%) were Dogs (54%+) and Cats (39%+).

In [10]:
```
#Visualization of Animal Types
from matplotlib import pyplot as plt
import matplotlib.pyplot as plt
import matplotlib.mlab as mlab
import numpy as np
animals = ['Dog', 'Bird', 'Other' ,'Cat', 'Livestock']
data = [16198,142,1697,11551,6]
fig = plt.figure(figsize = (10,7))
plt.pie(data,labels=animals, autopct='%1.1f%%',shadow=True,startangle=270)
plt.title('Percentage of Animals Rescued',fontweight='bold')
plt.axis('equal') # Equal aspect ratio ensures that pie is drawn as a circle.
plt.show()
```



Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

In [49]:
```
TransferedAnimalTypes = {}

AdoptedData = Data[Data["Outcome"]=="Adoption"]

for item in AdoptedData["Animal Type"]:
    if item in TransferedAnimalTypes:
        TransferedAnimalTypes[item] += 1
    else:
        TransferedAnimalTypes[item] = 1
#print(TransferedAnimalTypes)

Animalsvals = list(TransferedAnimalTypes.values())
Animalkeys = list(TransferedAnimalTypes.keys())


plt.bar(Animalkeys,Animalsvals,align = 'center')
plt.xticks(Animalkeys,rotation='90')
plt.title("Most Adopted Animals")
plt.show()
```



The vast majority of animals were picked up as Strays (88,000+), followed next by Owner Surrender (25,000+).

In [90]:
```
# Visualization of Intake Types
# Including All Animals with Intake Data, which is the entire dataset

Data['Intake Type'].value_counts()
print(Data['Intake Type'].value_counts())

objects = ("Abandoned", "Euthanasia Request", "Owner Surrender", "Public Assist", "Stray", "Wildlife")
ypos = np.arange(len(objects))
performance = [69, 66, 6160, 1842, 21936, 1243]

plt.bar(ypos, performance, align ='center', alpha=0.5, color ='teal')
plt.xticks(ypos, objects, rotation ='70')
plt.ylabel('Number of Intakes 10/13-6/21', fontweight='bold')
plt.xlabel('Intake Type', fontweight='bold')
plt.title('Intake Types Oct 2013 - June 2021', fontweight='bold', fontsize=12)
print(" ")
plt.show()
```

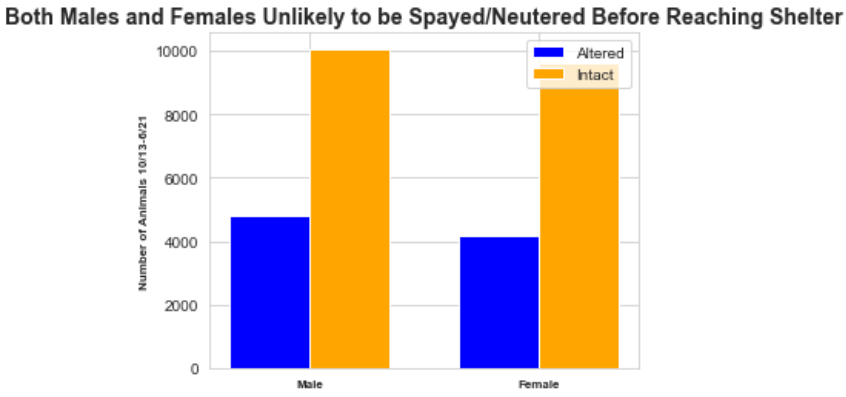
```
Stray                88202
Owner Surrender      25266
Public Assist         7772
Wildlife              5060
Abandoned              399
Euthanasia Request     257
Name: Intake Type, dtype: int64
```

Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

Two-thirds of both male and female animals were Intact (not spayed or neutered) when arriving at the shelter. The preponderance of dogs and cats, and of strays, is consistent with the large share of Intact animals.

In [91]:
```
# Create Side by Side Bar Graphs to see whether Males or Females
# Are More Likely to be Spayed/Neutered at Intake

Data['Sex at Intake'].value_counts()
print(Data['Sex at Intake'].value_counts())

Altered = (4826, 4152)
ind = np.arange(2)
width = 0.35
fig, ax = plt.subplots()
rects1 = ax.bar(ind, Altered, width, color='blue')

Intact = (10069, 9638)
rects2 = ax.bar(ind+width, Intact, width, color='orange')

ax.set_ylabel("Number of Animals 10/13-6/21", fontsize=8, fontweight='bold')
ax.set_title("Both Males and Females Unlikely to be Spayed/Neutered Before Reaching Shelter"
ax.set_xticks(ind+width/2)
ax.set_xticklabels(('Male', 'Female'), fontsize=8, fontweight='bold')
ax.legend((rects1[0], rects2[0]), ("Altered", "Intact"))
print(" ")
plt.show()
```

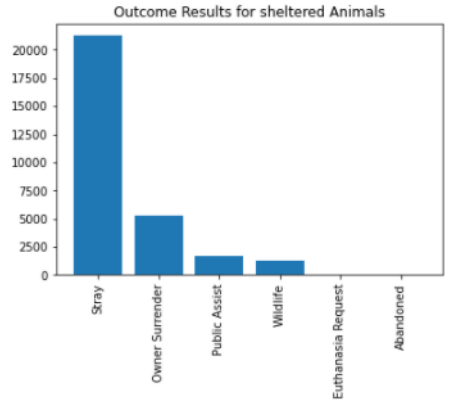
```
Intact Male      41335
Intact Female    39102
Neutered Male    19479
Spayed Female    16591
Unknown          10448
Name: Sex at Intake, dtype: int64
```



Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

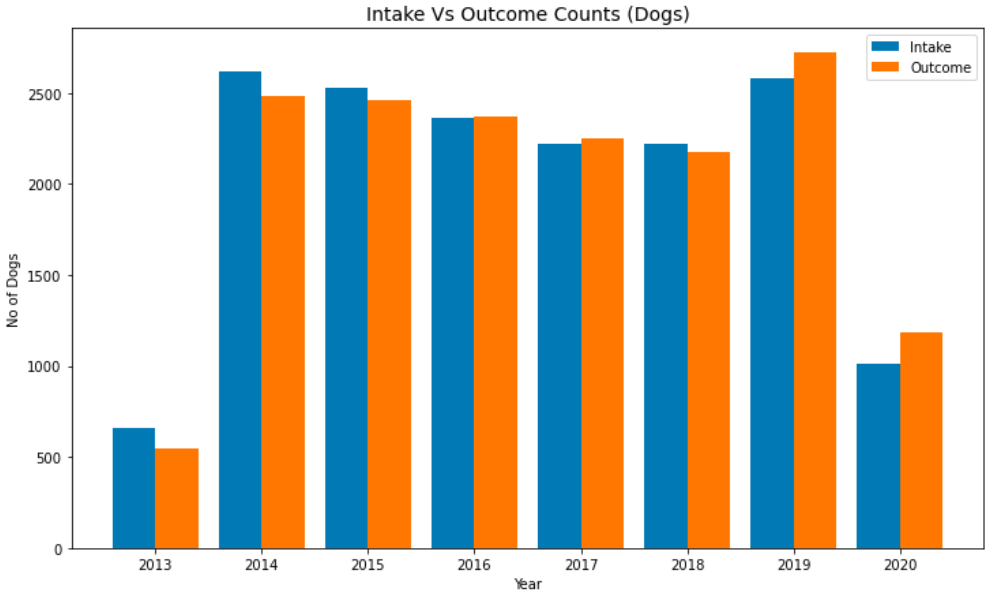
In [12]:

```
outcomeDict = {}

for item in Data['Intake Type']:
    if item in outcomeDict:
        outcomeDict[item] += 1
    else:
        outcomeDict[item] = 1

#print(outcomeDict)

outcomevals = list(outcomeDict.values())
outcomekeys = list(outcomeDict.keys())

plt.bar(outcomekeys,outcomevals,align = 'center')
plt.xticks(outcomekeys,rotation='90')
plt.title("Outcome Results for sheltered Animals")
plt.show()
```



Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

In [26]:
```
x = np.arange(len(year_list))
width = 0.4
# plot data in grouped manner of bar type
plt.rcParams["figure.figsize"] = (12,7)
plt.bar(x-0.2, dog_intakecnt_list, width)
plt.bar(x+0.2, dog_outcomecnt_list, width)
plt.xticks(x, year_list)
plt.title('Intake Vs Outcome Counts (Dogs)', fontsize=14)
plt.xlabel("Year")
plt.ylabel("No of Dogs")
plt.legend(["Intake", "Outcome"])
plt.show()
```



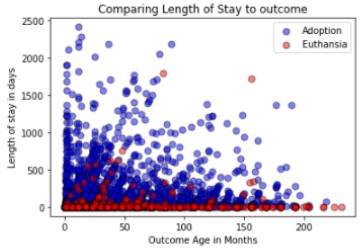
One very clear trend we saw was in the length of stay of the animal in the shelter. Most euthanized animals were in the shelter for a very short period of time. This feature later became a rather integral feature to our KNN model.

Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

Comparing Length of Stay to outcome

```
adopted_pets = pruned_data[ (pruned_data['Outcome'] == 'Adoption')]
euthanized_pets = pruned_data[ (pruned_data['Outcome'] == 'Euthanasia')]
```

```
plt.scatter(adopted_pets['Outcome Age in Months'] , adopted_pets['Length of Stay in Days'], s = 50, facecolors = 'blue', edgecolors= 'black', alpha = 0.5 )
plt.scatter(euthanized_pets['Outcome Age in Months'],  euthanized_pets['Length of Stay in Days'] ,s=50, facecolors = 'red', edgecolors= 'black', alpha = 0.5 )

plt.ylabel('Length of stay in days')
plt.xlabel('Outcome Age in Months')
plt.legend(['Adoption', 'Euthansia'], loc="best")
plt.title('Comparing Length of Stay to outcome')
plt.show()
```



As you can see here, most of the euthanized animals in the dataset had a very short stay at the shelter, while adopted animals' lengths of stay varied more.

Interestingly enough, the animal's age had no significant impact on the outcome.

3. Unsupervised Approaches (Python):

We were initially interested in whether any naturally occurring groupings existed among the animals, so we planned to perform Cluster Analysis techniques. However, after data cleaning/preparation and some research, it became clear the data did not include sufficient and reliable numeric variables suitable for Cluster Analysis.

4. Supervised Approaches (Python):

KNN. We chose a simple KNN classification, with hyperparameter k=20, for our model. We chose the precision of classifying animals with euthanasia outcome as the performance metric for this model because adoption was a much more common outcome. This meant if the model was badly tuned, the general accuracy would still be high because the model could just predict adoption for every sample.

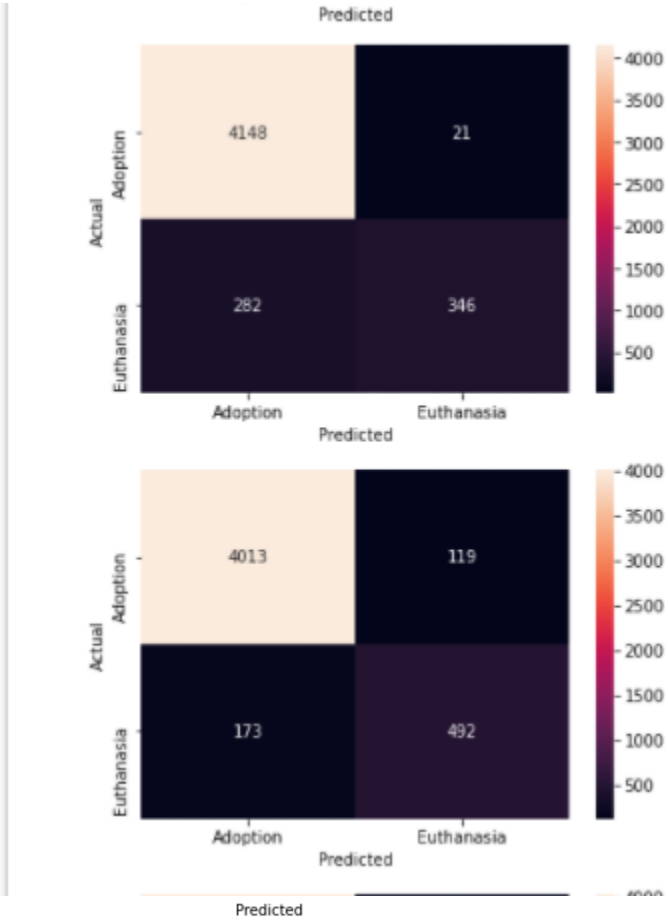
Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

Based on our exploratory analysis we started the model only using length of stay. Over 25 runs this gave an average accuracy of 65.

▾ Running model 25 times and storing euthanasia precision

```
accuracies = []
for runs in range( 0 , 25):
  x_train,x_test,y_train,y_test = train_test_split( pruned_features, pruned_target, test_size=0.3)
  model = KNeighborsClassifier(n_neighbors= 20)
  model.fit(x_train,y_train)
  pred = model.predict(x_test)
  accuracy_count = 0
  euthanasia_count = 0
  euthanasia_actual = 0

  y_test_as_list = y_test.to_list()

  for i in range(len(y_test_as_list)):
    if y_test_as_list[i] == 'Euthanasia':
      euthanasia_actual +=1


  for i in range(len(pred)):
    if pred[i] == y_test_as_list[i]:
      accuracy_count += 1
      if pred[i] == 'Euthanasia':
        euthanasia_count += 1
  accuracies.append(euthanasia_count/euthanasia_actual)
  confusion_data = {
      'y_Actual': y_test_as_list,
      'y_Predicted': pred
  }

  confusion_df = pd.DataFrame(confusion_data, columns=['y_Actual','y_Predicted'])

  confusion_matrix = pd.crosstab(confusion_df['y_Actual'], confusion_df['y_Predicted'], rownames=['Actual'], colnames=['Predicted'])

  sn.heatmap(confusion_matrix, annot=True, fmt='d')
  plt.show()

print("List of each run's precision: ",accuracies)
print("mean precision:", (sum(accuracies) / len(accuracies) ))
```

Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh




```
List of each run's precision:  [0.5619047619047619, 0.5102685624012638, 0.7218649517684887, 0.760655737704918,
mean precision: 0.6621520633163945
```

After adding other features like the intake type, and utilizing one-hot encoding, the overall euthanasia precision rose by 10%:

```
List of each run's precision:  [0.784251968503937, 0.7446153846153846, 0.771956
mean precision: 0.7691048077531961
```

However, other pairs of features like sex, spay/neuter status, and intake type did not improve classification accuracy:

```
List of each run's precision:  [0.5919003115264797, 0.598116169544741, 0.604
mean precision: 0.5989837713959586
```

Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

Logistic Regression1. Since we discovered the quality concerns with the Austin Animal Center's data, we decided to investigate whether the shelter (with cleaner data) could predict important outcomes from other variables collected. Specifically, we wondered whether (with cleaner data), the desired outcome of Placement in a home (i.e., either adoption or return to owner) could be predicted. To investigate this, we worked with a subset of the data that did not contain the errors and concerns we discovered. We removed the observations with one or more clearly erroneous values, such as a Date of Birth that was later than the Intake Date (shelter arrival date); Intake Date that was later than Outcome Date (the date at which the animal was adopted, returned to owner, etc.); missing Date of Birth; animals who were identified as Spayed/Neutered upon arrival but Intact at Outcome; etc. To minimize noise in the data, we also removed observations with an Intake Type of Euthanasia Request (only 0.2% of all intakes and often not a candidate for Placement) and observations with an Intake Type of Wildlife (of whom only four were placed in a home, and for whom Placement in a home is often not the goal). The remaining data set had 108,399 observations.

Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

In [41]:
```
LRData_df = pd.read_csv("Austin_Animal_Shelter_Dataset_LR.csv")

LRData_df.head()
LRData_df[0:10]
LRData_df.isnull().any()
print(LRData_df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 108399 entries, 0 to 108398
Data columns (total 16 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   Intake Age in Months  108399 non-null  int64
 1   Abandoned             108399 non-null  int64
 2   Owner Surrender       108399 non-null  int64
 3   Public Assist         108399 non-null  int64
 4   Stray                 108399 non-null  int64
 5   Normal                108399 non-null  int64
 6   SickInj               108399 non-null  int64
 7   PregNurs              108399 non-null  int64
 8   Feral                 108399 non-null  int64
 9   Bird                  108399 non-null  int64
 10  Cat                   108399 non-null  int64
 11  Dog                   108399 non-null  int64
 12  Livestock             108399 non-null  int64
 13  Other                 108399 non-null  int64
 14  Male                  108399 non-null  int64
 15  Placement             108399 non-null  int64
dtypes: int64(16)
memory usage: 13.2 MB
None
```

For this Logistic Regression, we also removed redundant, correlated, and unnecessary features. For example, we removed the redundant measures of age (Age in Days, Age in Weeks, and Age in Years). We removed Date of Birth, which would be correlated with age, and we removed Outcome, which would be correlated with Placement. We removed unnecessary features such as Name and Animal ID. We removed Breed and Color data because there were several hundred values of each, which would not be manageable in the model. We also removed Length of Stay, since from a practical standpoint it is more of an outcome than a predictor. Length of Stay would not

be known to the shelter in advance of outcome, so it does not belong in the regression model. The Logistic Regression model therefore included features:

- Intake Age
- Intake Type (Abandoned, Owner Surrender, Public Assist, and Stray)
- Intake Condition (Normal, Sick/Injured, Pregnant/Nursing, and Feral)
- Sex
- and Animal Type (Bird, Cat, Dog, Livestock, and Other).

The Target Variable was Placement (Yes/No, where Placement included either adoption or return to owner). We used 75% of the data as a Training Set and 25% as a Test Set, and the model yielded 68% accuracy.

```
In [44]: # Print Classification Report to look at model accuracy

         print(classification_report(y_test, y_pred))
```

```
              precision    recall  f1-score   support

           0       0.64      0.13      0.21      9250
           1       0.68      0.96      0.80     17850

    accuracy                           0.68     27100
   macro avg       0.66      0.55      0.51     27100
weighted avg       0.67      0.68      0.60     27100
```

Logistic Regression 2. We considered potential reasons why the first Logistic Regression model's accuracy was not excellent. One potential reason was that the variety of animals perhaps followed different patterns of placement, so the same model might not work for all of them. For example, the Animal Type "Other" was frequent and included rabbits, ferrets, rats, and guinea pigs but also snakes, bats, "bat mixes," and raccoons (even after removing those categorized as Wildlife). The Animal Type "Bird" also included a mix of birds that are typically pets and birds that are typically wild (even after removing those categorized as Wildlife). Therefore, we repeated the Logistic Regression using only Animal Types Cat and Dog to see if the available features would yield a more performant model when applied to a narrower set of animals limited to typical house pets.

Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

The Cats and Dogs data subset had 107,064 observations. This model used the same train/test split and same features as the first Logistic Regression. Perhaps because Cats and Dogs were the vast majority of the Animal Types to begin with, the results were nearly the same, again rounding to 68%. The model did not perform differently when limited to Cats and Dogs than it did when applied to all the non-Wildlife animal types.

In [10]:
```
LRCatDogData_df = pd.read_csv("Austin_Animal_Shelter_Dataset_LR_CatDog.csv")

LRCatDogData_df.head()
LRCatDogData_df[0:10]
LRCatDogData_df.isnull().any()
print(LRCatDogData_df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 107064 entries, 0 to 107063
Data columns (total 13 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   Intake Age in Months  107064 non-null  int64
 1   Abandoned             107064 non-null  int64
 2   Owner Surrender       107064 non-null  int64
 3   Public Assist         107064 non-null  int64
 4   Stray                 107064 non-null  int64
 5   Normal                107064 non-null  int64
 6   SickInj               107064 non-null  int64
 7   PregNurs              107064 non-null  int64
 8   Feral                 107064 non-null  int64
 9   Cat                   107064 non-null  int64
 10  Dog                   107064 non-null  int64
 11  Male                  107064 non-null  int64
 12  Placement             107064 non-null  int64
dtypes: int64(13)
memory usage: 10.6 MB
None
```

Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

In [13]:
```
# Print Classification Report to look at model accuracy

print(classification_report(y_test, y_pred))
```

```
              precision    recall  f1-score   support

           0       0.65      0.13      0.22      9008
           1       0.69      0.96      0.80     17758

    accuracy                           0.68     26766
   macro avg       0.67      0.55      0.51     26766
weighted avg       0.68      0.68      0.61     26766
```

In [14]:
```
# Generate a Confusion Matrix for another look at accuracy

y_train_pred = cross_val_predict(LogReg, x_train, y_train, cv=5)
confusion_matrix(y_train, y_train_pred)
precision_score(y_train, y_train_pred)
```

Out[14]:
```
0.6842885969718432
```

Conclusion:

We achieved several of the planned milestones. First, we cleaned and combined the datasets for intake and outcomes. Then we performed exploratory data analysis in Python, making graphs and plots to uncover patterns, suggest analyses, and provide distributions along various attributes. We planned one Unsupervised Approach (Cluster Analysis), but determined our dataset was a poor candidate for that approach. We performed two Supervised Approaches using Python. One was a kNN classification model to classify animals according to likely Adoption or Euthanasia outcomes. The other was a Logistic Regression model to predict whether or not an animal's outcome will be successful placement in a home.

Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

Reference Material:

1. Similar Projects:

Predicting Pet Adoption Speed Using Python — Part I | by Richa Vala | Medium

This project used the "petfinder.my" dataset from kaggle to predict pet adoption speed given certain attributes. Considering that this project has a similar goal, we can use this as a reference to build our models and also compare how our data differs from theirs. Petfinder is an internet-based pet adoption website that includes pets available at over 14,000 shelters and rescue organizations (petfinder.com).

Animals see the LightGBM: Predicting Shelter Outcomes for SoCo County Animals

This project is similar to ours, except it uses a different county as its data source and includes only cats and dogs. Since the structure and overall goal are similar, we can use this as a reference to help produce our model, give further insight and build off of their work, and compare/contrast our results to theirs.

Both of these projects are freely available and the published articles share the code snippets, making the code open-source. These projects provide clear metrics with which to compare our own results. We can then use this to see if the difference in our datasets changed how predictions were made to any significant degree. Our project differs from these two related projects in that the Petfinder project includes only pets and covers select affiliated shelters across North America, while the SoCo project only includes dogs and cats in a single county (Sonoma). Our dataset covers one city and includes not only pets but also livestock and wildlife species, including bats (a special feature of Austin's animal shelter, since Austin has North America's largest urban bat colony; Bats | AustinTexas.gov; Smithsonian Magazine on Austin Bats).

2. Data Source:

City of Austin Open Data Portal: https://data.austintexas.gov/browse

Jana Voege
Keith Santamaria
Sri Tammineni
Farhan Almufleh

3. Coding Help:

- Python Tutorial (w3schools.com)
- Logistic Regression for Machine Learning: complete Tutorial - Just into Data
- API Reference — scikit-learn 0.24.2 documentation
- Example of Confusion Matrix in Python - Data to Fish
- Python for Data Science Essential Training by Pierson