Springboard Data Science Career Track - Capstone Project 2

Quality Measurement of NYC 311 Calls

Jeffrey Ma 12/31/2019

Project Objectives

This project has two main objectives, a learning objective and a business objective.

Learning Objective:

- This capstone project summarizes the knowledge and skills of program participants in the data science career track as demonstrated in the project workflow from data gathering, wrangling, analysis, modeling, summarizing, and recommendation for action plans.

Business Objective:

- Through the use of real business data, findings and conclusions from this project are useful for NYC 311 public agencies to improve their service request resolutions

Methodology

The following methodology was adapted to assess performance of agencies resolving service requests from NYC 311:

- **Dataset:** Identify where to find the required data and data retrieval
- **Data Wrangling:** Once the data is identified and retrieved, basic data preprocessing and quality control are conducted
- **Data Exploration:** When the data is ready for exploration, apply statistical methods to identify data clusters, correlations, and trends
- **Data Modeling with Machine Learning for Prediction:** With basic understanding of the data, more advanced data analytics of machine learning is applied to the data to build predictive models
- **Summary:** conclusions and recommendations

Dataset

- The dataset that was used is the “NYC Open Data 311 Service Requests from 2010 to Present”.
- The data was acquired by using the Socrata Open Data API (SODA) which provides programmatic access to the dataset.
- The original dataset contains 22.1 million rows and 41 columns, a total of 906.1 million data points.
- To make this training project more manageable in Python as well as more relevant to today’s status, a 50,000 row subset of the data from January 1, 2019 to January 15, 2019 was taken.

Data Problem

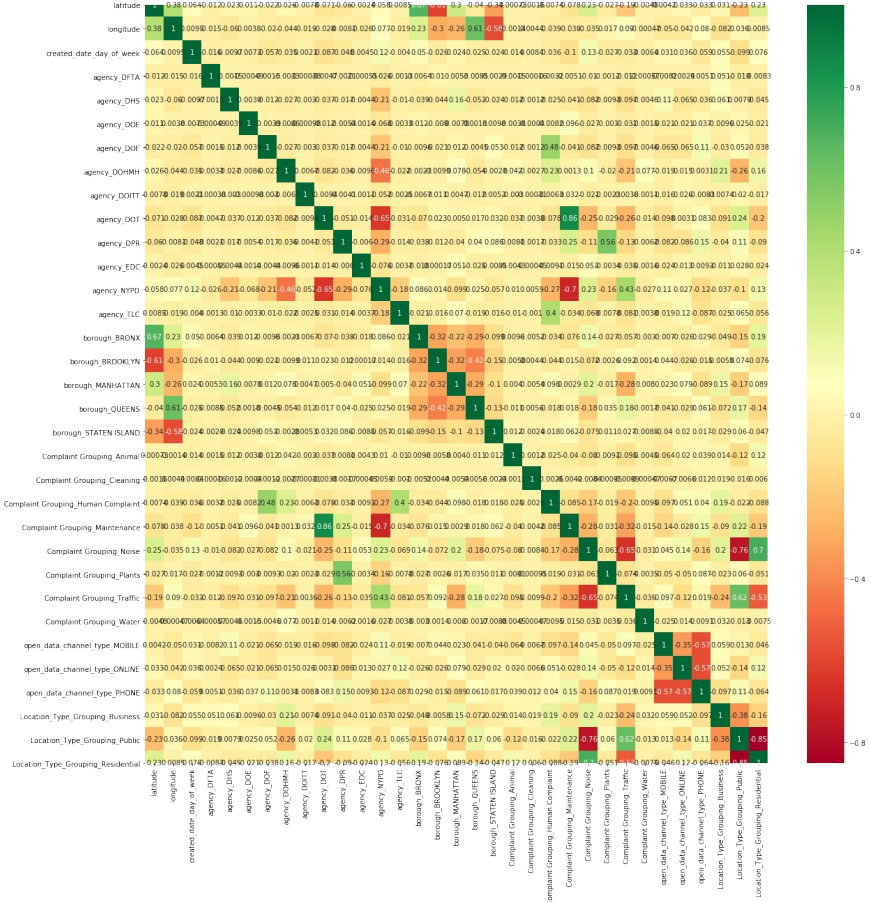
- When people require non-emergency municipal services, the number to call is 311.
- A potential problem that can be addressed is the number of late incident resolutions.
- A timeframe known as a service level agreement is created based on the type of request made, which informs the customer how much time it will take to respond to the request.

Some questions this dataset brought up:

- Can we predict the likelihood a service request will be resolved after the estimated due date?
- Can we improve estimates for service request resolutions?

Data Wrangling

- Any extraneous data was removed
  - Columns containing only redundant data
  - Columns that held only descriptive information of the complaint
  - Columns that were specific to a certain complaint type like bridges and taxis
- Features for Data Exploration and Machine Learning were identified and prepared
  - Converting dates to datetime
  - Grouping complaint, location, and open data channel types in order to reduce features
  - Encoding nominal values as dummy variables
  - Preparing the target column “late” by defining it as all cases closed after the estimated due date



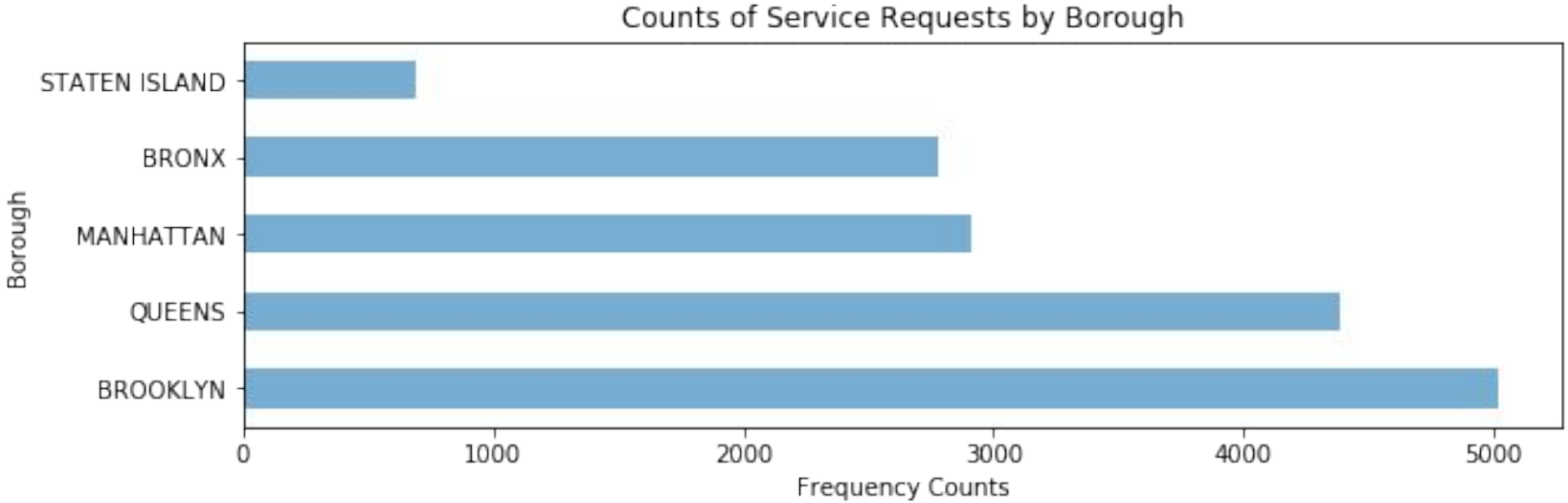
37 independent variables correlated with themselves

Data Exploration: Correlation Matrix

- A few features seem to be correlated (the green ones)
- Overall the feature variables appear to be independent
- This leads to a data question:
  - Can we predict if a service request will be resolved on time or not based on these independent variables?

Data Exploration - Findings

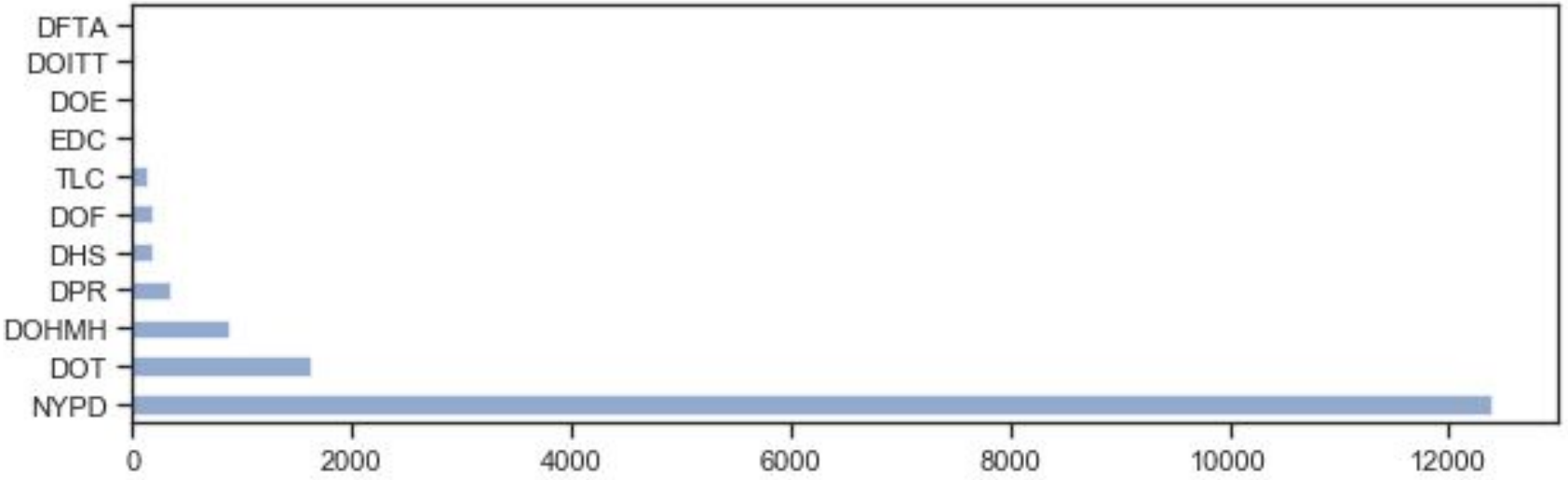
- Service requests are well represented by each borough



Graph displaying counts of service requests by borough

Data Exploration - Findings

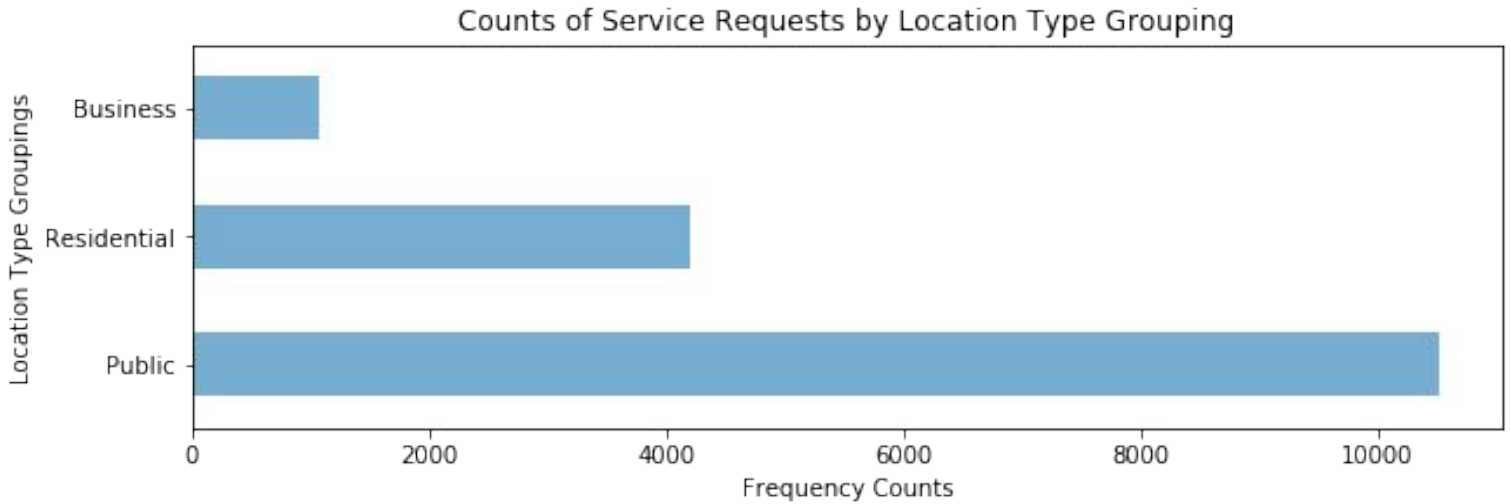
- A few agencies appear to not have many service requests, could affect the outcome of the study.



Graph displaying counts of service requests by NYC Agency

Data Exploration - Findings

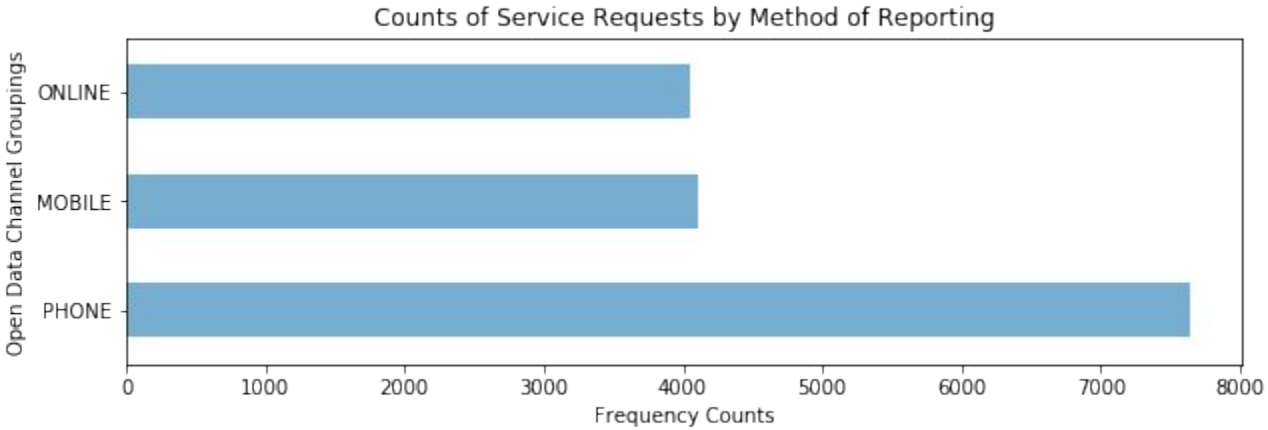
The location type of the service requests in the dataset seem well represented



Graph displaying counts of service requests by location type.

Data Exploration - Findings

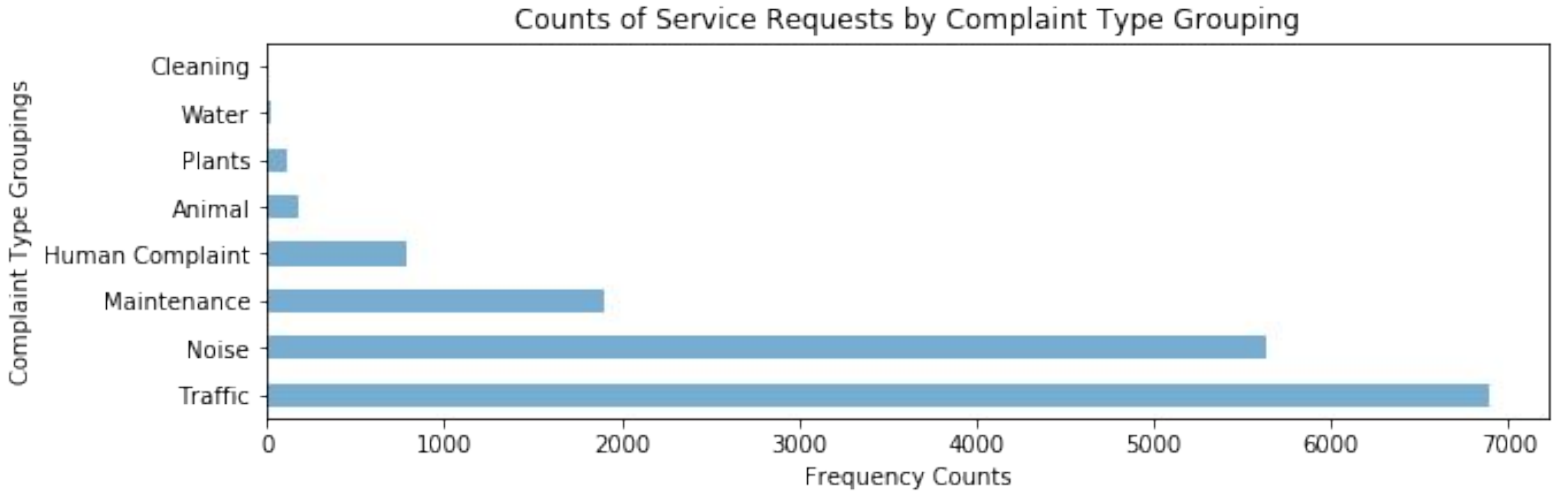
The methods by which service requests are reported appears to be relatively evenly distributed



Graph displaying counts of service requests by method of reporting.

Data Exploration- Findings

Cleaning and Water labels seem underrepresented in the dataset. Could affect interpretation of results.



Graph displaying counts of service requests by complaint type.

Data Exploration - Findings

- 16% of the cases in the subset were resolved after the estimated resolution date of the request

```
#create target variable: late

api_df['late']=api_df['closed_date']>api_df['due_date']
api_df['late'].describe()
```

```
count     16587
unique        2
top       False
freq      13873
Name: late, dtype: object
```

```
#Percentage of cases that are late

print("Late Percentage:", (16586-13872)/16586*100, "%")
```

```
Late Percentage: 16.363197877728204 %
```

Machine Learning [ML] - Data Preparation

To prepare for machine learning, a one hot encoding was done to create binary variables out of the scalar variables for modeling.

```
api_df= pd.get_dummies(api_df, columns=['agency','borough',
                                        'Complaint Grouping','open_data_channel_type','Location_Type_Grouping'])
```

```
api_df.head()
```

Complaint Grouping_Water	open_data_channel_type_MOBILE	open_data_channel_type_ONLINE	open_data_channel_type_PHONE	Location_Type_Grouping_Business	Locatio
0	0	1	0	0	
0	0	1	0	0	
0	0	0	1	0	
0	0	1	0	0	
0	0	0	1	0	

ML - Algorithms used

- As the goal of the data analysis is to classify service request resolutions as late or not, a supervised machine learning classification method seems appropriate.
- A Random Forest Model and a Logistic Regression Model were built on the dataset.
- The score of the random forest was low, so more emphasis was placed on the logistic regression model.

```
#Score of model

random_forest.score(X_test,y_test)
```

```
0.14334279168987
```

- Score of random forest model

ML - Logistic Regression

After choosing the logistic regression model built on the training set. We used the test set to check to see how our model was performing. Accuracy of 0.87 was noted.

```
#split again into training and test set due to changed columns and then fit

logreg = LogisticRegression()
X_train, X_test , y_train, y_test = train_test_split(X,y, test_size = .5, random_state = 2)
result=logreg.fit(X_train, y_train)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear_model\logistic.py:432: FutureWarning: Default solver will be changed
to 'lbfgs' in 0.22. Specify a solver to silence this warning.
  FutureWarning)
```

```
#Accuracy

y_pred = logreg.predict(X_test)
print('Accuracy of logistic regression classifier on test set: {:.2f}'.format(logreg.score(X_test, y_test)))
```

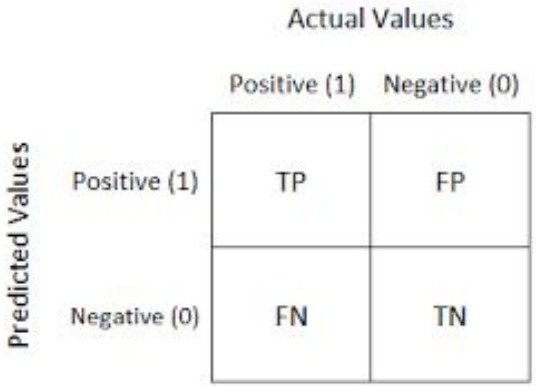
```
Accuracy of logistic regression classifier on test set: 0.87
```

ML - Confusion Matrix for Logistic Regression Model

The confusion matrix shows how the model performed on the test set.

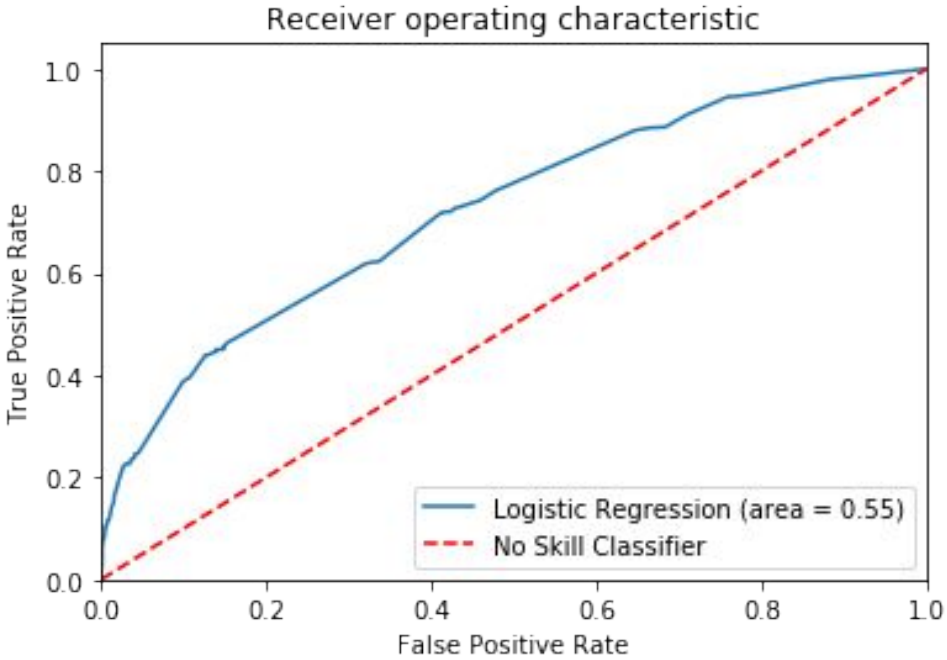
```
#Confusion Matrix

confusion_matrix = confusion_matrix(y_test, y_pred)
print(confusion_matrix)
```

```
[[6726   56]
 [ 989  129]]
```



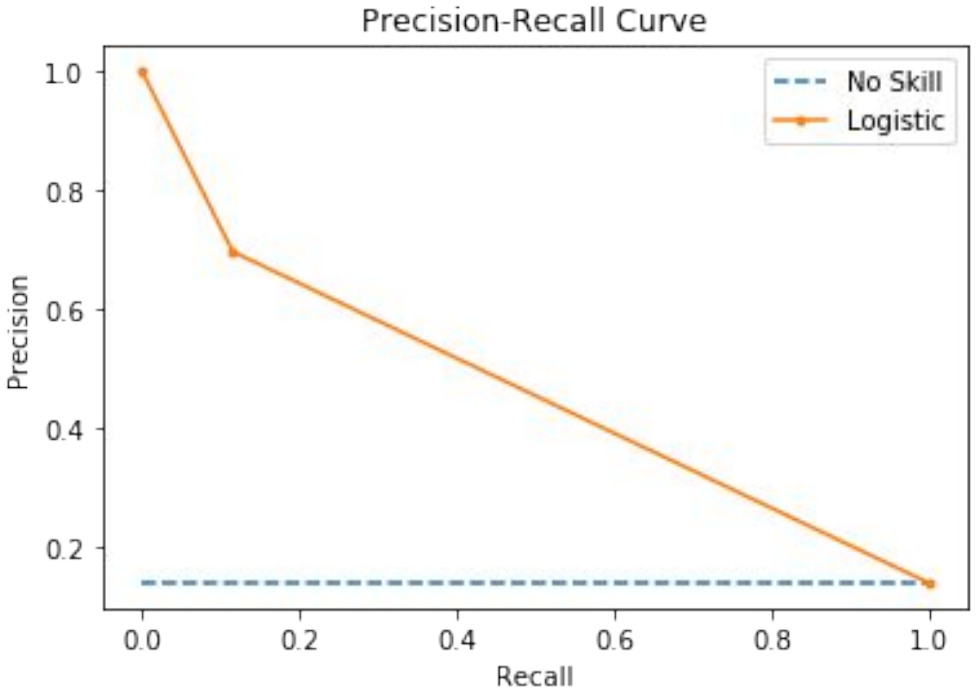
ML - Logistic Regression AUC-ROC Curve

- ROC is a probability curve and AUC represents degree or measure of separability.
- Tells how much the model is capable of distinguishing between classes.
- The dotted line represents a no-skill classifier that cannot discriminate between the classes and predicts labels randomly or as a constant class.
- With a large AUC of 0.55, the model is strong at predicting late calls as late.



ML- Logistic Regression Precision Recall Curve

- Precision describes how good a model is at predicting the positive class
- Recall is the number of true positives divided by the sum of the true positives and the false negatives.
- For our dataset, the curve depicted above represents a skillful model because the curve bows towards (1,1) (a model with perfect skill) above the flatline of no skill.



Results

- The coefficients of logistic regression and odds ratios for each feature in the dataset.
- The odds ratio shows the increase or decrease in likelihood of a service request being resolved if all other features are kept constant.
- The largest odds ratio seen belonged to calls handled by the Taxi and Limousine Commission with a 849% increase in probability of a service request being labeled as late according to the model.
- The feature with the lowest odds ratio belong to calls handled by the New York Department of Transportation. These cases were 68% less likely to result in a label of “late” according to the model.

features	coefficients	odds ratio
object	float64	float64
agency_DFTA	-0.4890501427947647	0.6132085782311333
agency_DHS	0.7905214250876688	2.204545632016986
agency_DOE	0.7771162692006365	2.175190548481822
agency_DOHMH	-0.6124181891947023	0.5420385312540468
agency_DOITT	-0.1976192736550583	0.8206822490184139
agency_DOT	-1.1494340475693037	0.3168160214476185
agency_EDC	1.6669727864811212	5.296111046878813
agency_NYPD	-0.8540228941578774	0.42569894085552573
agency_TLC	2.1387588088341096	8.488894746206027
borough_BRONX	0.6556384958614397	1.926372105408051
borough_BROOKLYN	-0.4115813203032325	0.6626016358375854
borough_MANHATTAN	-0.30382587293139574	0.7379893591870111
borough_STATEN ISLAND	0.4763546105937086	1.6101939065477158
Complaint Grouping_Animal	-0.2509088880773264	0.7780932619025418
Complaint Grouping_Human Complaint	0.9912242857268226	2.6945313300373996
Complaint Grouping_Maintenance	-0.4340824675104814	0.6478588259777599
Complaint Grouping_Noise	-0.7471392687660418	0.4737198012095078
Complaint Grouping_Plants	0.3615000433971755	1.4354810842801455
Complaint Grouping_Water	0.31667372023654555	1.3725546620470068
Location_Type_Grouping_Business	0.7035743904383929	2.0209635253553206

Conclusion

With the machine learning model built in this project, features which contributed to predicting whether or not a service request would be resolved on time or not were identified.

This information is useful for New York City agencies to improve their service and performance. The model suggests that the Bronx and Staten Island require more attention from public agencies, whereas service requests originating in Manhattan and Brooklyn were resolved much quicker.

An improvement to the model created in this project would be to train the model on a much bigger subset of the data. 50,000 rows were taken from the NYC Open Data 311 dataset, but 22.1 million rows of data exist.

Acknowledgements

- I would like to thank my mentor Nathan Sutton for all his help and advice throughout the Springboard program, especially the capstone projects
- I would also like to thank Springboard management and subject matter experts for their help