Taxi Trajectory Data ML Applications

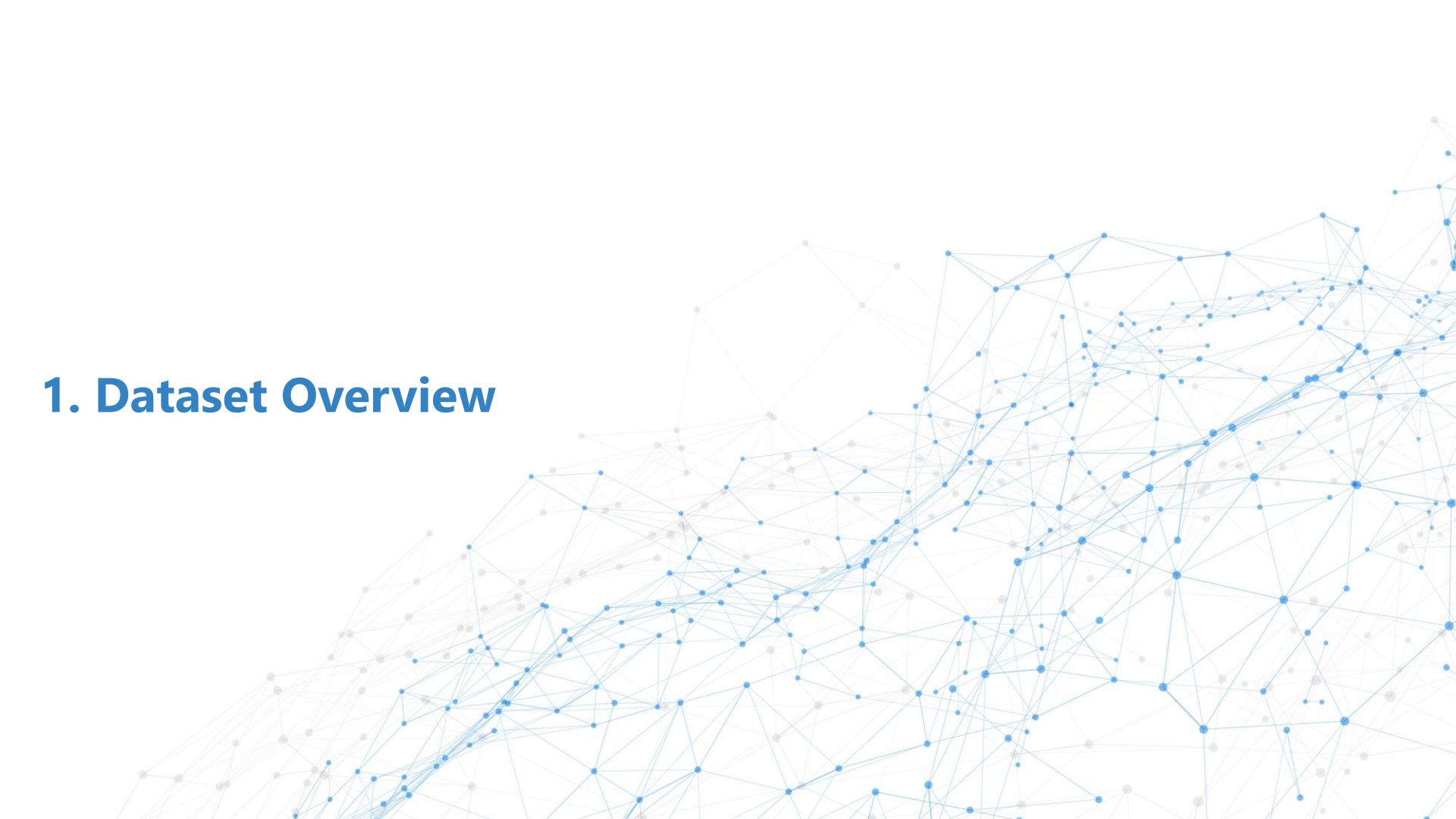
Nikolaos Chiotis | MTN2221

Feb ‘23

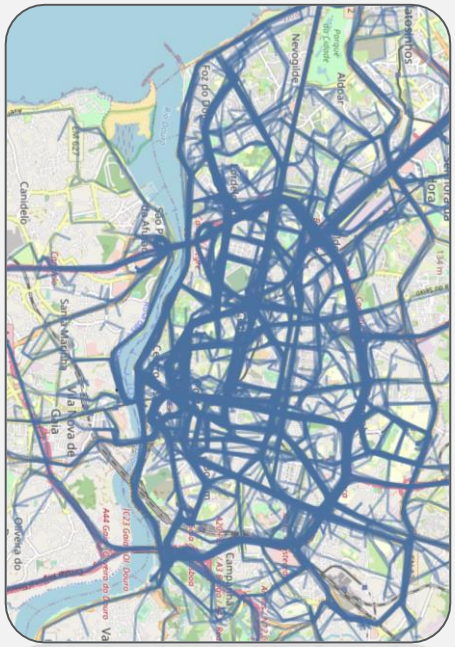
Agenda

1. Dataset Overview
2. Data Preprocessing
3. Travel Duration Prediction
4. Travel Value Prediction

1. Dataset Overview



1. Dataset Overview



0.4K
Taxis

- ✓ Trajectory data for **442 Taxis**
- ✓ City of **Porto, Portugal**
- ✓ **1 year of data** from July 2013 tp June 2014

1.7M
Rows

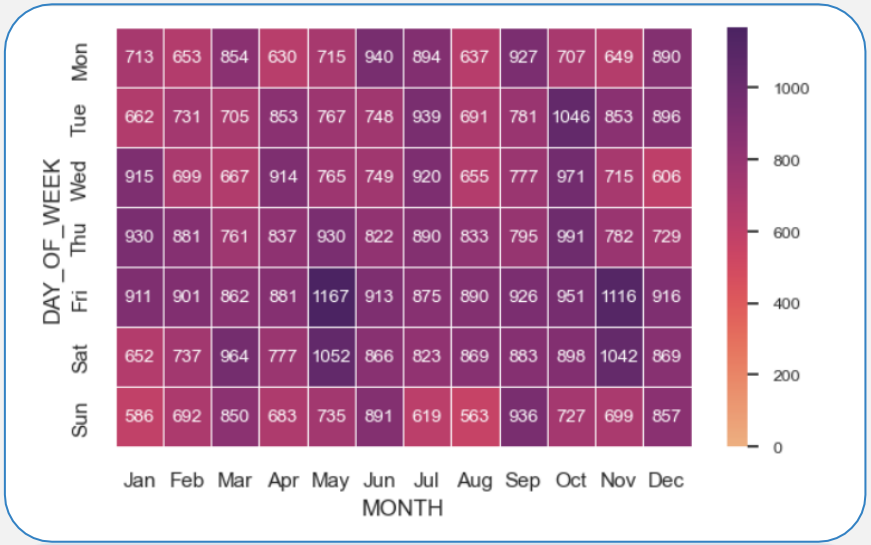
- ✓ Each row contains a **POLYLINE** for each taxi trip
- ✓ Trajectory has taxi's **coordinates every 15 seconds**
- ✓ **Undersampling to 100K rows**

9
Columns

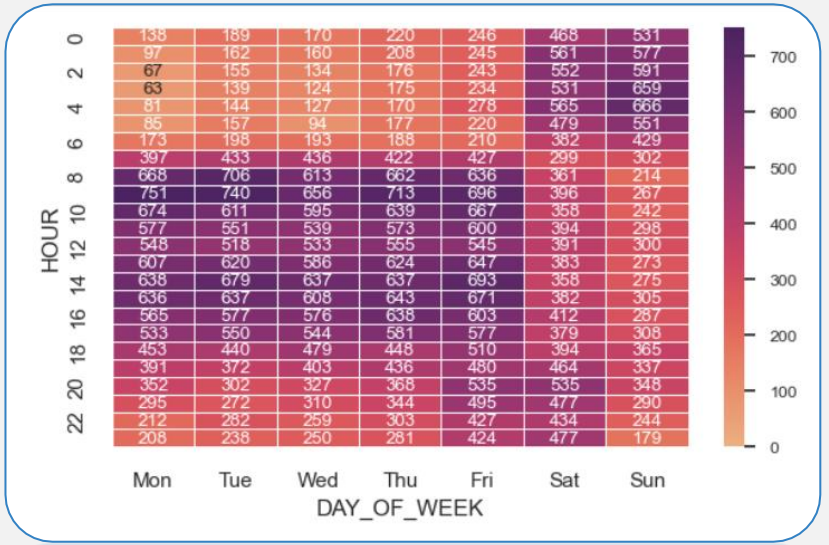
- ✓ **TRIP_ID**
- ✓ **CALL_TYPE**
- ✓ **ORIGIN_CALL**
- ✓ **ORIGIN_STAND**
- ✓ **TAXI_ID**
- ✓ **TIMESTAMP**
- ✓ **DAYTYPE**
- ✓ **MISSING_DATA**
- ✓ **POLYLINE**

1. Dataset Overview | Dates

Day of week/Month Distribution

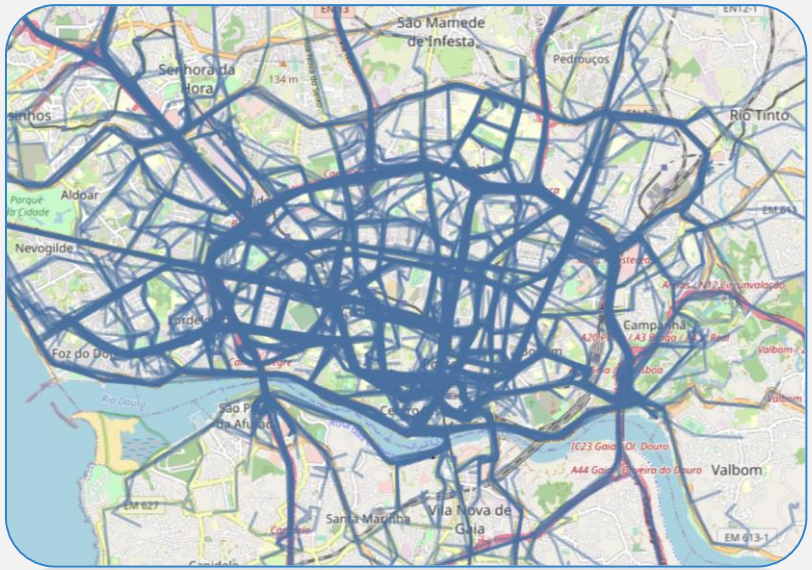


Hour/Day of week Distribution

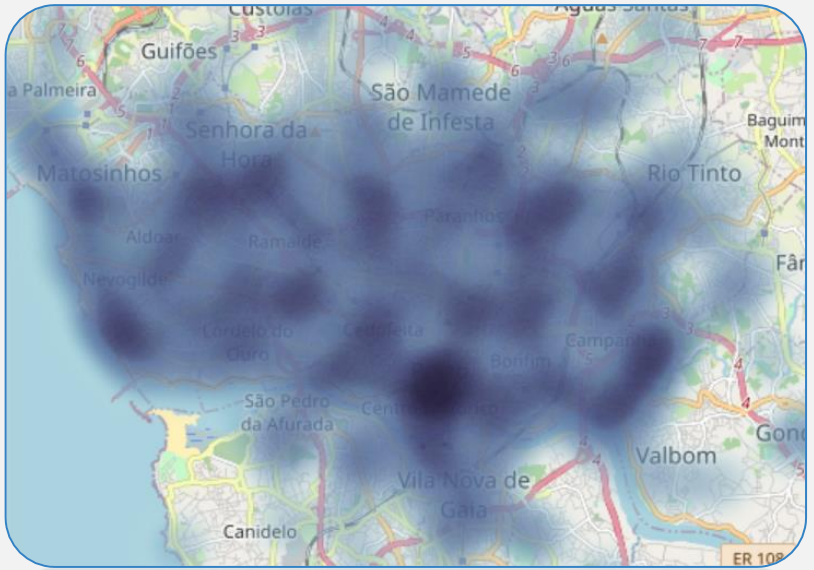


1. Dataset Overview | Routes & Starting Points

Routes Sample



Starting Point Heatmap



2. Data Preprocessing

2. Data Preprocessing | Geolocation Preprocessing

Parse Polyline

Polyline column is a list of coordinates in String.
We need to parse it and convert it to GeoPandas POLYLINE

Travel Duration

Extract travel duration based on trajectory.
By multiplying number of points by 15 seconds.

Departure & Arrival

Extract starting and ending point from POLYLINE.
Also calculate arrival time.

Outliers & Clean up

Identify outliers in trajectory data, based on speed & acceleration.
Exclude trips outside of city.
Exclude outliers from trajectories.

OSM Enhancement

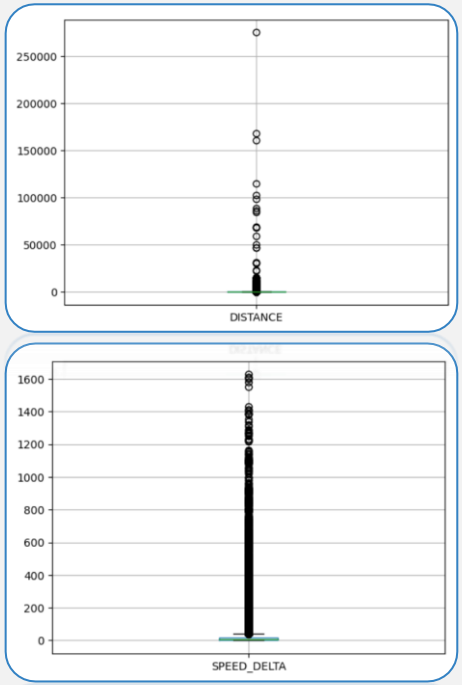
Map trajectories to actual roads.
Calculate distance on map based on roads
Find road type coverage

Feature Preprocessing

One-hot or label encoding
Check nulls and single-valued variables
Check correlations

2. Data Preprocessing | Outliers & Clean up

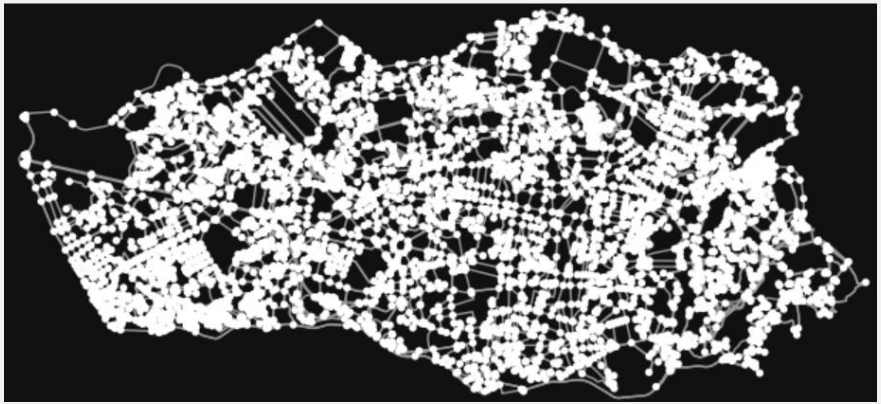
Boxplot



Outlier Example

2. Data Preprocessing | Open Street Map

Load Open Street Map – Road Network



Actions

- ✓ Load **road network**
- ✓ **Map trajectory** starting and ending points on **actual streets**
- ✓ Calculate **shortest paths**
- ✓ Extract extra feature:
  - **Route length**
  - **Type of road and number of lanes** at starting and ending point
  - **Percentage** of route coverage for each **road type** (roadway, residential etc)
  - **Number of turns**

3. Travel Duration Prediction

3. Travel Duration Prediction | Regression

Evaluation metric

Mean square error
Mean root square error

of

Travel duration

Algorithms

K-Neighbors Regressor
RandomForest Regressor

with

GridSearchCV

Results

RandomForest Regressor

with MRSE

of

6.37 minutes

Outcomes

Important

Features

were

Route Distance & Hour

4. Travel Value Classification

4. Travel Value Prediction | Classification

Target Variable

Taxi Route Value Factor

HIGH VALUE: 0/1

Based on the distance and time combination

Algorithms

AdaBoost Classifier
GradientBoosting Classifier

in

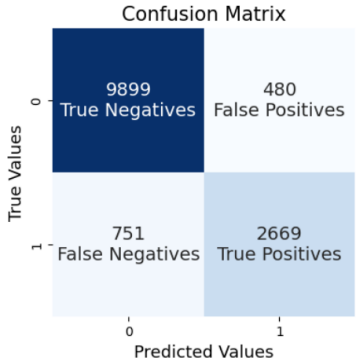
Voting Classifier

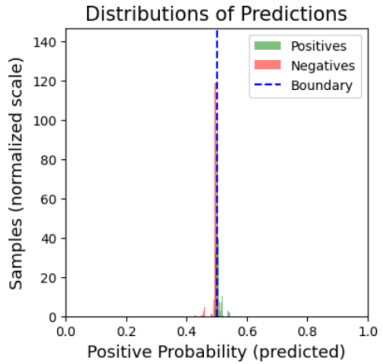
4. Travel Value Prediction | AdaBoost

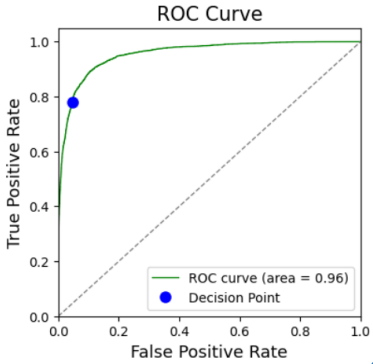
AdaBoost Classification Results

```
evalBinaryClassifier(ada_clf, X_test, y_test)
```

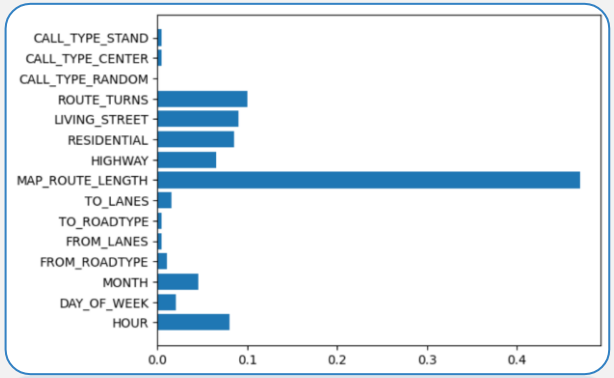
	precision	recall	f1-score	support
0	0.93	0.95	0.94	10379
1	0.85	0.78	0.81	3420
accuracy			0.91	13799
macro avg	0.89	0.87	0.88	13799
weighted avg	0.91	0.91	0.91	13799







Feature Importance

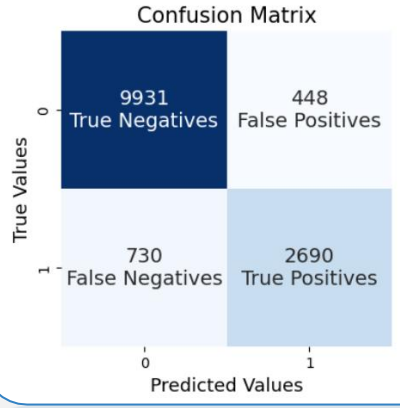


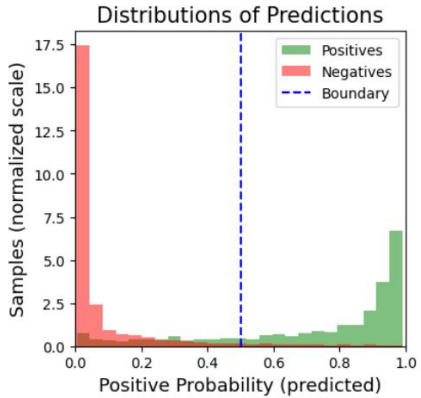
4. Travel Value Prediction | GradientBoosting

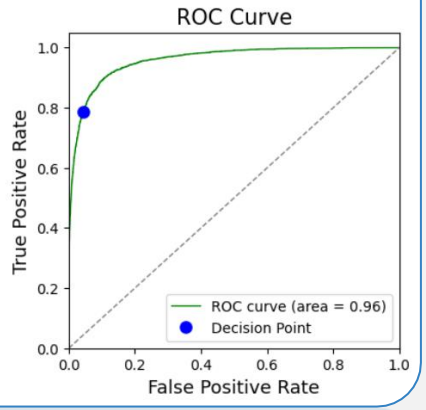
GradientBoosting Classification Results

```
evalBinaryClassifier(hgb, X_test, y_test)
```

	precision	recall	f1-score	support
0	0.93	0.96	0.94	10379
1	0.86	0.79	0.82	3420
accuracy			0.91	13799
macro avg	0.89	0.87	0.88	13799
weighted avg	0.91	0.91	0.91	13799





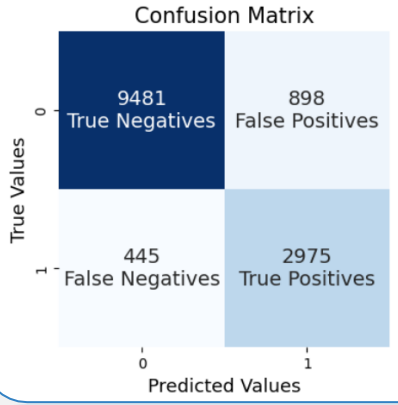


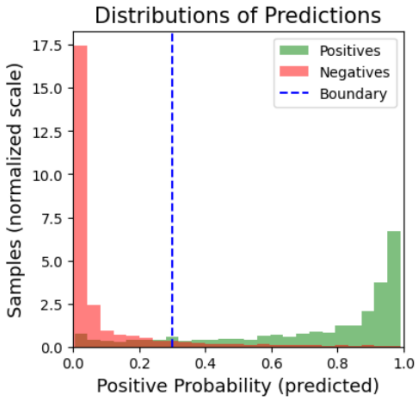
4. Travel Value Prediction | GradientBoosting (thres=0.3)

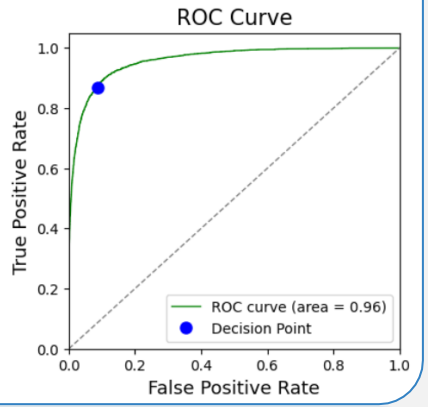
GradientBoosting Classification Results

```
evalBinaryClassifier(hgb, X_test, y_test, thres=0.30)
```

	precision	recall	f1-score	support
0	0.96	0.91	0.93	10379
1	0.77	0.87	0.82	3420
accuracy			0.90	13799
macro avg	0.86	0.89	0.87	13799
weighted avg	0.91	0.90	0.90	13799







4. Travel Value Prediction | Voting Classifier

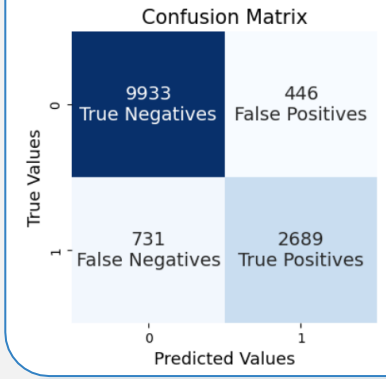
Voting Classification Results

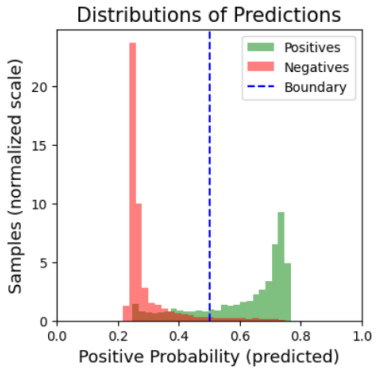
```
clf = VotingClassifier(estimators=[('ada_clf', ada_clf),('hgb', hgb)], voting='soft')
clf.fit(X_train, y_train)
evalBinaryClassifier(clf,X_test,y_test)
```

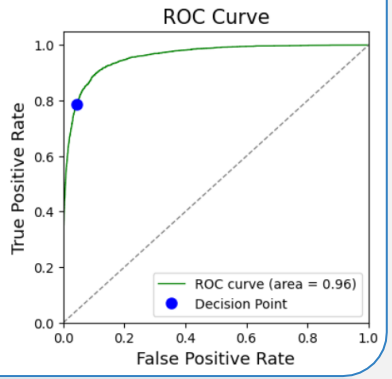
```
              precision    recall  f1-score   support

           0       0.93      0.96      0.94     10379
           1       0.86      0.79      0.82      3420

    accuracy                           0.91     13799
   macro avg       0.89      0.87      0.88     13799
weighted avg       0.91      0.91      0.91     13799
```







Taxi Trajectory Data – ML | Extensions

Possible Enhancements

- Graph ML to predict road traffic flow
- Outlier detection using ML on trajectories
- Apply node embeddings for route predictions

Thank you!

Questions?