NYC Taxi Travel Time Prediction

APMAE 4990 - Introduction to Data Science
Final Project

Camille Ruppli
Diana Mojahed
Pawan Lapborisuth

Gathering Data from BigQuery

- 2016 yellow taxi data
  - Pickup date/time
  - Pickup lat/lang
  - Dropoff date/time
  - Dropoff lat/lang
  - Haversine distance calculated from pickup and dropoff lat/lang
- 2016 NOAA weather data
  - Temperature
  - Visibility
  - Snow depth
  - Precipitation
- ~3 million samples saved into csv for ease of storage

Removing Outliers and Filter Data

- Remove entry with 0 or NaNs
- Lat/Lang: Filter those only within NYC perimeter
- Travel time: Filter between 0 to 1 hour
- Haversine distance: Filter between 0 to 20 km
- Fix entry with missing weather data (prcp,sndp) to be 0 instead of 999.99

Feature Processing

- Day of week
  - Obtain day of week (Mon-Sun) from pickup datetime
- Hour of day
  - Obtain hour of day from pickup datetime
- Sine/Cosine conversion
  - Hour (0-24) and day of week (1-7) are converted to sine/cosine variables in order to avoid abrupt changes between 0/24 and 1/7
- K-means clustering for lat/lang for linear models
  - Relatively accurate for small ranges of lat/lang (such as in NYC perimeter)

Model Selection: Linear Models

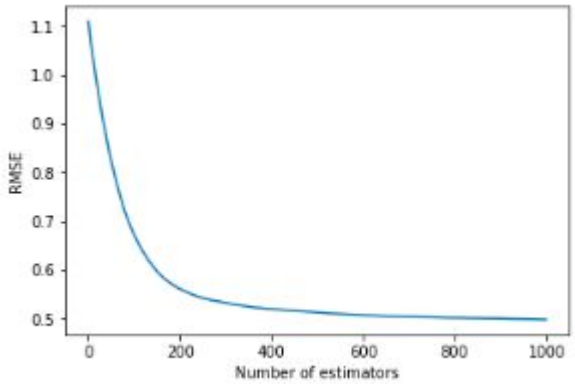
- Linear regression
  - Final R^2 result = 0.5814
- Lasso
  - Scale data
  - GridsearchCV to find the optimal alpha value -> Alpha = 0.001
  - Final R^2 result = 0.5875
- Ridge
  - Scale data
  - GridsearchCV to find the optimal alpha value -> Alpha = 0.9
  - Final R^2 result = 0.5875

Model Selection: K-Nearest Neighbors

- Scale data
- Use GridsearchCV to search for the optimal K value between 1-100
  - 3 fold CV
  - Optimal K = 10
- Final R^2 result = 0.7462
- Final RMSE result = 0.5207
- Final RMSLE result = 1.0210

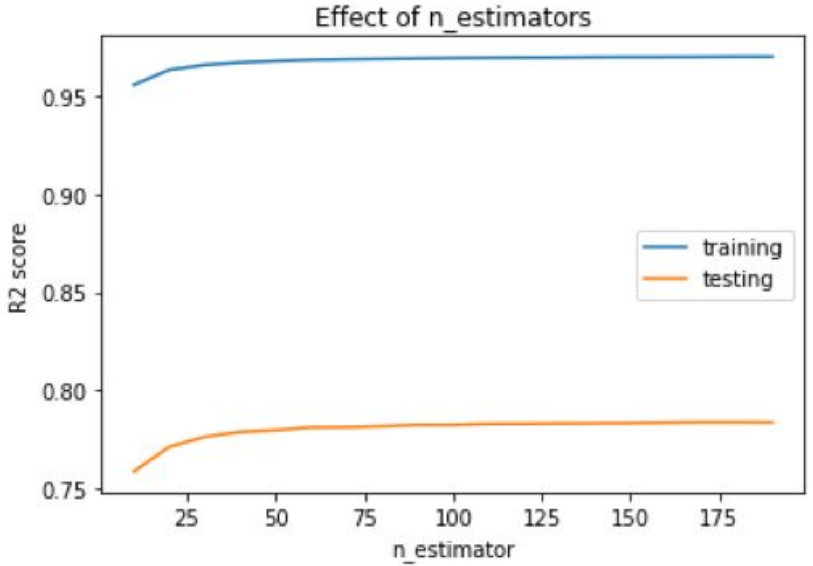
Model Selection: AdaBoost and XGBoost

- Grid search parameters for AdaBoost : learning rate and n_estimators
- AdaBoost performances worst than default XGBoost
- Randomized then Grid Search to optimize : max_depth, min_child_weigth, subsample, colsample_by_tree, gamma, regularization parameters
- Further tuning to reduce overfitting
- **Score for XGBoost on our test set**
- RMSE 0.50
- RMSLE 0.43
- R2 0.72



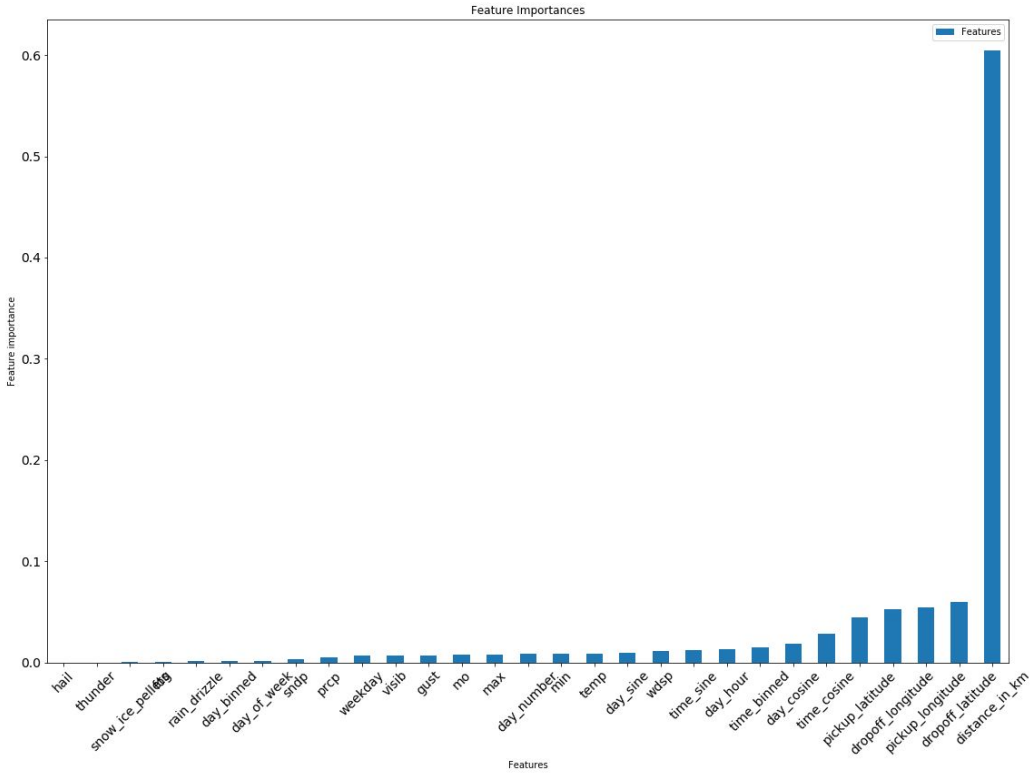
Model Selection: Random Forest

- Parameters were tuned using GridSearch
- Final parameters: 50 estimators, 50 maximum tree depth, automatic maximum features
- Five-fold cross-validation used to evaluate performance



Random Forest: Feature Importance

- Most important features are:
- Location: Haversine distance, drop-off latitude, pickup longitude, drop-off longitude, pickup latitude
- Time: time (cosine), day (cosine), time (binned), hour of the day
- Weather was not significantly important



Random Forest: Results

- Final random forest model was trained on the entire dataset (3 million data points) and 29 features

	Training	Testing
R2	0.97	0.80
RMSE	0.17	0.47
RMSLE	0.21	0.40

Prediction using Test Dataset

- Test dataset
  - Calculate haversine distance using pickup/dropoff lat/lang
  - Calculate additional time features used in the training dataset
  - Query 2015 weather data from bigquery
  - Merge taxi and weather data using date_of_year feature
  - Drop unnecessary/repeated columns
  - Fix missing weather data entry
- Perform final prediction using Random Forest Regressor with the final parameters after tuning
  - 50 estimators
  - 50 maximum tree depth
  - Automatic maximum features