Daily Trip Analysis and Prediction

Dataset: SF Bay Area Bike Share

Zhipeng Yu[2] Jianzhong Cheng[1] Jin Han[2]

[1]Department of Civil and Environmental Engineering
[2]Department of Electrical Engineering and Computer Science

University of Michigan

April 2018

Outline

Outline

Motivation

Why bike-sharing system?

- **Wide coverage**:
  Over 50 countries, 712 cities with 806,200 bicycles operating at 37,500 stations.
- **Eco-friendliness and cost-effectiveness**:
  Bike-sharing system is green and of low carbon, and each bike can be used by several people per day
- **Convenience**:
  Bikes are usually more flexible in crowded areas than cars or buses

Why daily trip prediction?

- **Make the system more efficient**:
  Having a better distribution of bikes across different stations and cities
  Minimizing shortage or idle bikes at a particular station or city.

Outline

Literature review

- **Statistics Method**: Time series models (e.g. ARMA)
- **Machine Learning Method**: Linear Regression, Neural Network, Clustering and etc.
- In this project, however, we are trying to make daily demand predictions with a popular tree-based model - Xgboost (Extremely Gradient Boost), and compare it with classic models like baseline model, linear models. Furthermore, we implement stacked model.

Outline

Dataset Overview

Tables in dataset

Station:
Data that represents a station where users can pickup or return bikes, which include the station name, location, dock count, city and installation data

Status:
Data about the number of bikes and docks available for given station and time.

Trip:
Data about individual bike trips, which includes start/end time/location and duration.

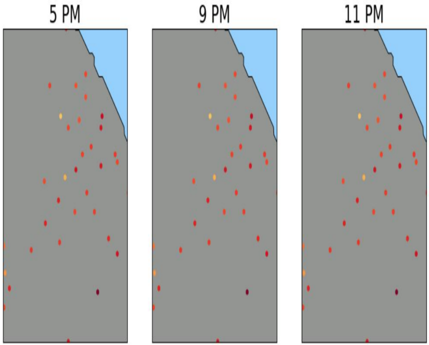
Weather:
Data about the weather on a specific day for certain zip codes.

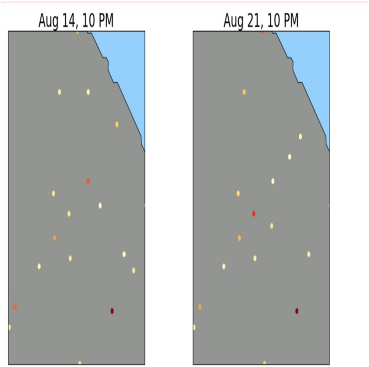
- 5 cities, 70 stations, nearly 700,000 trips.
- 733 days, from Aug.,2013 to Aug.,2015

Outline

Flow Visualization

San Francisco one hour visualization





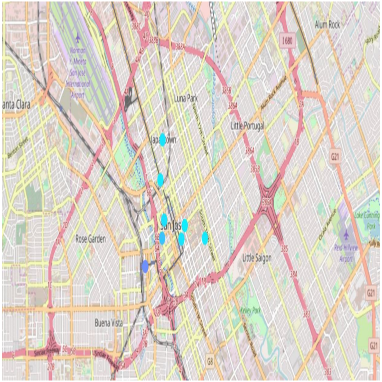
More dynamic: Video
https://www.youtube.com/watch?v=xOaHupdXA9sfeature=youtu.be

Clustering to Reduce Spatial Data Set Size

- implementation of DBSCAN clustering algorithm
  - why?
  - rendering a JavaScript web map (like Leaflet) with millions of data points on a mobile device can swamp the processor and be unresponsive.
- k-means vs DBSCAN
  - for spatial latitude-longitude data, the DBSCAN algorithm is far superior
  - k-means minimizes variance
  - DBSCAN minimizes physical distances from each point and cluster size

Flow Visualization

San Jose pick station and drop station.



https://youtu.be/DIzK3VrZzGE

Outline

What interests us from these tables

- Output extraction
  - Merge station data and trip data based on start station id.
  - Extract the data from a certain city.
  - Count the number of trips for each day
- Station dataset: we can get the station number based on the installation time.
- Trip dataset: we can get the number of trips for each city.
- Weather: we can get the weather, like humidity, temperature.. every day for each city by using the zip code.
- Date feature: beside the datasets provided, date is really important for riding bikes. So we get the date information, which includes
  - Year,Month,Day in a week
  - Holiday/Business day/Weekday

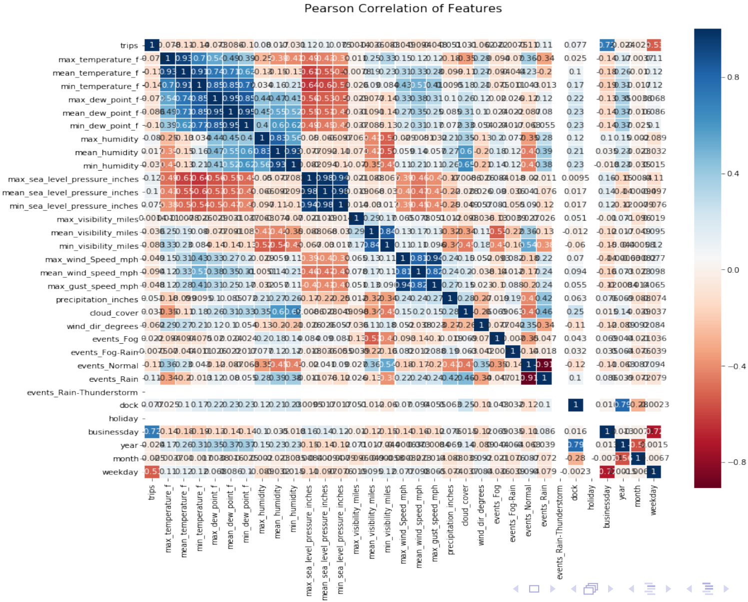
Data Preparation

Feature engineering and preprocessing

- Extracting relevant features
- Feature transformation
- Adding and combining features (Holiday, Weekday, Business day, Year, Month, Day of week)
- Filling missing data
- Normalization
- One-hot encoding for weather event
- Splitting the whole dataset into 90% training set and 10% test set randomly. Then pick 10% of training data as validation set.

```
1 max_temperature_f
2 mean_temperature_f
3 min_temperature_f
4 max_dew_point_f
5 mean_dew_point_f
6 min_dew_point_f
7 max_humidity
8 mean_humidity
9 min_humidity
10 max_sea_level_pressure_inches
11 mean_sea_level_pressure_inches
12 min_sea_level_pressure_inches
13 max_visibility_miles
14 mean_visibility_miles
15 min_visibility_miles
16 max_wind_Speed_mph
17 mean_wind_speed_mph
18 max_gust_speed_mph
19 precipitation_inches
20 cloud_cover
21 wind_dir_degrees
22 events_Fog
23 events_Fog-Rain
24 events_Normal
25 events_Rain
26 events_Rain-Thunderstorm
27 dock
28 holiday
29 businessday
30 year
31 month
32 weekday
```

Feature correlation



Outline

Model Selection

Baseline Model

- Use the mean value as prediction

Linear Model

- Do data normalization with pipeline
- Use cross-validation to find hyperparameters
- Apply linear regression with l1 norm, which could sparse features.

Ensemble Model

- Use Grid Search to find hyperparameters
- Xgboost (an updated version of GBDT)

Ensemble Method

- Model Stacking

Model stacking

- The output of first layer would be input of seconde layer.
- Highlight each base model where it performs best and discredit each base model where it performs poorly.
- Ensemble of strong models
- Two layers of models for this problem
  - First Layer: Adaboost, GBDT, Random Forest
  - Second Layer: Linear regression with l1 norm.

Outline

1. Introduction
   - Motivation
   - Literature review
2. Analysis
   - Dataset Overview
   - Flow Visualization
3. Model and Results
   - Data Preparation
   - Model Selection
   - Result Comparison
   - Feature exploration

Result Comparison

- **San Jose**

The RMSE result is from the prediction on the test set.

Baseline Model

- Baseline RMSE: 21.69

Linear Model

- Lasso Regression RMSE: 12.69

Ensemble Model

- Xgboost RMSE: 10.83

Ensemble Method

- Stack RMSE: 10.20

Result Analysis

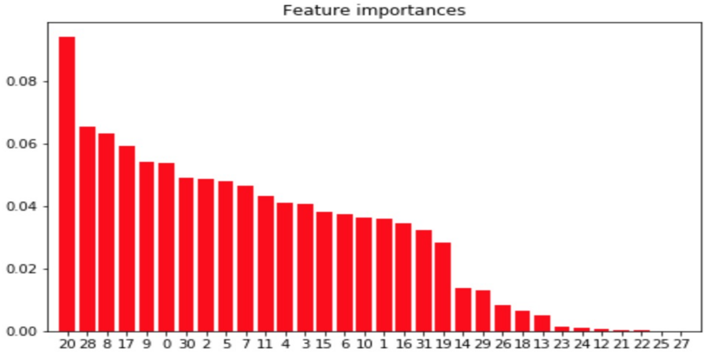
- **San Jose**

Comparison

- The machine learning model are quite better than base model.
- Xgboost and stack performs very well and stack method is a little better than xgboost.

Outline

Feature importance from tree model



The corresponding features are presented in slide 15

Top5 important features:
20.cloud_cover;28.holiday;8.mean_humidity;17.mean_wind_speed_mph;
9.min_humidity

Summary

- We merge the datasets, extract trip features and do clustering for the future exploration.
- We visualize the trip numbers for each station.
- We explore the performance of different models for predicting daily trips for a certain city and feature importance

Thanks!