London Bike-sharing Usage Analysis

07-Cabbage: https://github.com/yumengl720/stat480_project.git

Yumeng Li
yumengl4

Xiao Zhang
xiaoz16

Kexin Zhu
Kexinz8

***Abstract*—With the promotion of sustainable travel alternatives and the development of the sharing economy, topics about bike sharing usage such as exploratory display, analysis of influencing factors and demand forecasting are attracting scholars to discuss. This paper focuses on the spatiotemporal evolution analysis of the demand for shared bikes in hourly-level in London in the three years from 2017 to 2020. From the time perspective, utilize prophet to implement a multiplicative decomposition model. For data augmentation, call google place search API to make the geographic variables of each station more informative (count the places of interest around). Use the augmented explanatory variables to cluster the stations with K-Means, build a Naïve Bayes model and a CNN classification model, and use accuracy on test set and confusion matrix to evaluate their performances.**

***Index Terms*—bike-sharing, Prophet, Google API, K-Means, Naïve Bayes,CNN**

I. Introduction

Bike sharing facilitates people to travel short distances where public transportation is not available and can also promote low-carbon living. As this system is more widely adopted around the world, the issue of how to dynamically adjust the supply of stations according to demand is starting to attract research attention. In product environment, an official vehicle shuttling within suburbs to balance the distribution of bike, transporting excess bikes in less busy areas to higher demand stations. Before being too late, we need to predict the future rental and return difference to improve the quality of service.

This project focuses on the future demand difference of bike-sharing in London from both temporal and spatial perspectives. For data, we perform exploratory data analysis with summary statistics and visual figures. Implement data augmentation by transforming station_id and hour into more informative ones: total counts of each place of interest category (Google Search API), weekday vs weekend (dummy variable) and hour in a day (One-Hot Encoding).

For model, we take the difference between the number of returned bike and the number of bike rented out as our outcome variable modeled by all input variables obtained after data augmentation period. We first build a time series model to carve out the bike-sharing demand difference in hourly basis by station. Next, break down the goal into a classification problem (the sign of demand difference is used to determine the need to redistribute the bikes at a specific hour). In response to that, we build Naïve Bayes model and Convolutional Neural Network. Evaluating the performances of two classifiers by accuracy on test set and a confusion matrix. In the end, we summarize our main findings and propose future expectations for this project.

II. Related Work

According to current literature related to Bicycle-Sharing System (BSS), we can divide them into three topics: the benefit of BSS, the factors affecting renting and returning behavior and the predictive models.

A. Research Value

A lot of studies prove the benefit of BSS to traffic, environment and health. Paul DeMaio discussed the history and concluded bike-sharing has had profound effects on creating a larger cycling population, decreasing greenhouse gasses, and improving public health [1]. Sakari Jäppinenet al. concluded BBS complementing the traditional public transport system could potentially promote sustainable daily mobility based on model [2]. Other studies also show the helpful influence of BSS, so analyzing how to improve the service quality of BSS is very meaningful.

B. Influencing Factors

Researchers consider weather, time, spatial distribution and potential demand as the factors affecting Bicycle-sharing usage. Huthaifa I. Ashqar et al. considered the effect of weather conditions on bike counts in their model [3] Wafic El-Assi et al. revealed a significant correlation between temperature, land use and bike share trip activity through studying Canada's second largest BSS [4]. Boniphace and Hualiang also included campus characteristics for prediction [5]. Eric Hsueh-Chan Lu et al. used sine and cosine transformation to get periodic temporal features [6] Sakari Jäppinen et al. use the time from station to points of interest as the spatial features [2]and Eric et al. also define a method to get a spatial vector to quantify spatial features [6]. Based on what we discussed above and the variables in our data set, we consider rental features, temporal features and spatial features as our factors. And different from these papers, we use dummy variables as temporal features: we use 24 hours and divide a week to weekday and weekend we use the counts of points of interest as the spatial features instead of the distance.

C. Predictive Models

Most of the studies use statistical models to predict. Fournier et al. developed a sinusoidal model to predict the pattern of seasonal sharing bicycle demand [7] Yajun et al. fitted a prediction method based on the Markov chain model [8] The rest of studies use data mining and deep learning. Sathishkumar used five Statistical regression models and Gradient Boosting Machine got the best results [9] Eric Hsueh-Chan Lu et al. used Recurrent Neural Network (RNN) to predict the rental of users [6] Leonardo et al. proposed Decision Support System (DSS) based on Artificial Neural Networks (ANN) and Fuzzy Logic to predict the demand [10]. To summarize, statistical models usually have low cost, but others may perform better prediction. Because of the difference of the data set, we cannot conclude which model is the best to predict the numbers of renting and returning bicycles. Therefore, first we use a time series model to see the trend and prediction. Then we try to find some models like KNN and Naive Bayes, CNN to predict the difference demand.

III. Data

The dataset "London and Taipei Bike-Share Data" is retrieved from Kaggle [11], where the London dataset is what we focus on. It is 5GB of unzipped multi-dimensional time-series data about bike-sharing in London. There are two CSV files describing the data from London, one contains records of each rental action with 38215560 records and 9 features, including rental id, duration, bike id, start and end rental date time, and start and end station. There are 68282 records missing values and because of the large amount of dataset, we can directly remove these records. The first 10 records of the original data are as shown in Fig 1:

rental_id	duration	bike_id	end_rental_date_time	end_station_id	end_station_name	start_rental_date_time	start_station_id	start_station_name
61343322	60.0	12871.0	2016-12-28 00:01:00	660.0	West Kensington &...	2016-12-28 00:00:00	633	Wrenker Road Nort...
61343321	300.0	2837.0	2016-12-28 00:05:00	763.0	Mile End Park Lei...	2016-12-28 00:00:00	531	Twig Folly Bridge...
61343323	360.0	1269.0	2016-12-28 00:06:00	99.0	Old Quebec Street...	2016-12-28 00:00:00	116	Little Argyll Str...
61343325	1140.0	4238.0	2016-12-28 00:20:00	468.0	Cantrell Road, Bow	2016-12-28 00:01:00	443	Philpot Street, W...
61343324	null	1496.0		null	null	2016-12-28 00:01:00	319	Baldwin Street, S...
61343326	1380.0	11485.0	2016-12-28 00:25:00	219.0	Bramham Gardens, ...	2016-12-28 00:02:00	116	Little Argyll Str...
61343327	960.0	7686.0	2016-12-28 00:20:00	710.0	Albert Bridge Roa...	2016-12-28 00:04:00	74	Vauxhall Cross, V...
61343328	720.0	4192.0	2016-12-28 00:16:00	260.0	Broadwick Street,...	2016-12-28 00:04:00	188	Hatford Place, Ma...
61343329	780.0	10153.0	2016-12-28 00:18:00	233.0	Pall Mall East, W...	2016-12-28 00:05:00	68	Theobald's Road ,...
61343330	180.0	14189.0	2016-12-28 00:08:00	275.0	Barbican Centre, ...	2016-12-28 00:05:00	95	Aldersgate Street...

Fig. 1. First 10 of London Dataset

The other contains geographic information about stations in London with 802 stations and 4 features, including station id, station names, longitude and latitude. The first 10 records of this data are shown in Fig 2:

station_id	station_name	longitude	latitude
1	River Street, Cle...	-0.109971	51.5292
2	Phillimore Garden...	-0.197574	51.4996
3	Christopher Stree...	-0.0846057	51.5213
4	St. Chad's Street...	-0.120974	51.5301
5	Sedding Street, S...	-0.156876	51.4931
6	Broadcasting Hous...	-0.144229	51.5181
7	Charlbert Street,...	-0.168074	51.5343
8	Maida Vale, Maida...	-0.18348599999999998	51.529857
9	New Globe Walk, B...	-0.09644080000000001	51.5074
10	Park Street, Bank...	-0.0927542	51.506

Fig. 2. First 10 of London Stations Dataset

To prepare for our model, we calculate the demand difference for each station every hour, which is the difference between the number of bikes returned and the number of rental bikes per hour. First, we group by start rental date time and start station id. Then count the bilk id as the numbers of rental bikes and create a new dataset named start. The same as returned bikes with end time and end station id. Third, combine the two datasets by time and station is and make difference of end count and start count.

IV. Exploratory Data Analysis

Table 1 shows the summary statistics of the outcome variable: demand difference. The mean value is around 0, but the standard deviation is large compared to the mean. The min and max values are also very different than other position statistics. The distribution of the demand difference is very concentrated at 0 while having some outliers. The histogram of the demand difference also supports this conclusion, it can be found in Fig 3.

TABLE I
Summary Table of y

Summary	y
count	8499193
mean	-0.0744
stddev	3.2779
min	-58
25%	-1
50%	0
75%	1
max	57

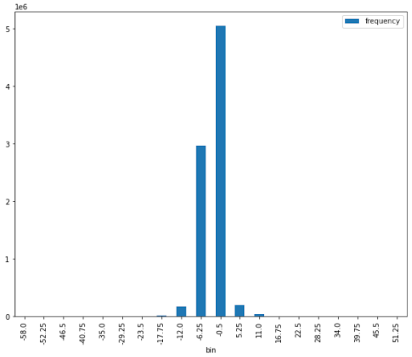


Fig. 3. Histogram of y

Fig 4 shows a distribution map of all stations in our dataset. There are 802 stations in total with clear clustering patterns based on distance.

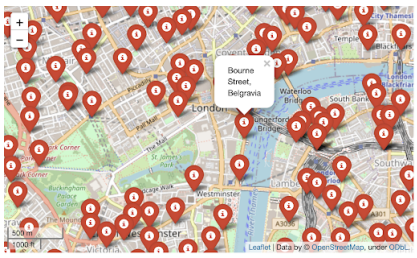


Fig. 4. Map of London Station

Fig 5 shows the status of bike demand on 2016-12-28 12:00:00. The red color indicates the station is currently in excess demand. The green color indicates the station is currently in shortage demand.

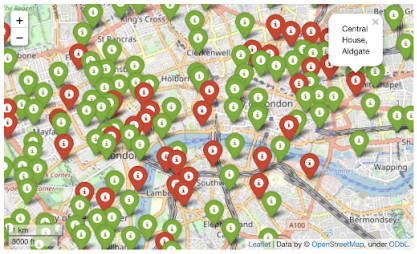


Fig. 5. Map of London Station, Colored by Demand

Fig 6 shows the demand difference from 2017-01-01 to 2017-01-10 of some selected stations. Station 14 displays a distinct pattern from other stations.

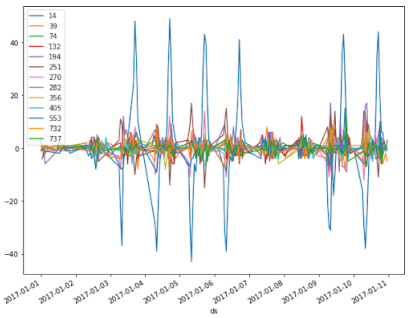


Fig. 6. Diff v.s Time by Station

Overall, the time series dataset shows different patterns from different stations. While the station shows some clustering patterns on distance. And the outcome variable may suffer from outlier problems.

V. Methods and Results

A. Time Series Model

In order to investigate the pattern between date time and the demand difference, we utilize an open-source industry-level timing prediction library fdprophet to empower our implementation. Prophet is a time series forecasting model developed by Facebook in 2017 which can effectively deal with multiple seasonalities, such as yearly, weekly and daily trends. With our time series data grouped by station, we use a pandas User-Defined Function (UDF) to establish a Prophet model, which allows us to apply it to each station in our dataset.

To decrease the running time of Prophet model, we choose a subset of total 802 stations in our dataset by examining the frequency of each station and choosing the stations whose frequency exceeds 20,000. The station ID of this subset includes 132, 194, 14, 74, 39, 251, 553, 732, 270, 356, 405, 737, and 282. Splitting the training and testing set is based on the date "2020-03-09". After splitting, the training set includes all data before 2020-03-09, which counts for 88% of the whole dataset.

For each individual model fit, we use RMSE (Root Mean Square Error) to evaluate the effectiveness of the models. The definition of RMSE is as below. Lower values of RMSE indicate the higher performance of the model.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} \|y(t) - y\hat{(t)}\|^2}{n}}$$

The overall RMSE of each station is demonstrated as Table 2. Station 14 has the biggest value of RMSE at 9.71, while other stations have similar values around 3. In the broad picture, the Prophet model did a good job of predicting the demand difference.

TABLE II
Overall RMSE Comparison

Station ID	RMSE	Station ID	RMSE	Station ID	RMSE
14	9.71	39	2.84	74	2.43
132	3.10	194	3.80	251	3.93
270	2.93	282	2.44	356	2.56
405	2.25	553	3.23	732	2.95
737	2.72				

Further, we take a closer step to an individual station to get detailed information about the model results. We take Station 132, which has the largest frequency as an example to illustrate. Fig 7 shows the model prediction on the training set, where y denotes the true value of demand difference, and yhat denotes the prediction of Prophet. Specifically, the model captured the daily peaks and weekly seasonality. Fig 8 shows the contrast between true values and prediction values in the testing set. Although it does not capture the extreme values in the testing set, the seasonality is still matched. Table 3 represents the RMSE values for the training and testing set. RMSE for the training set is slightly higher than that for the testing set, which may be an indication of overfitting.

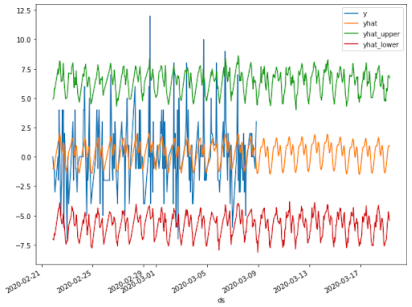


Fig. 7. True v.s Prediction in Train

B. Classification

In the next part of the machine learning model, instead of caring about specific values as in the time series part, we break down the problem into a classification prediction. We want to release a signal through the demand difference: a demand

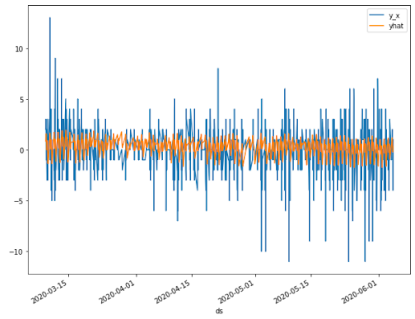


Fig. 8. True v.s Prediction in Test

TABLE III
RMSE OF STATION 132

	RMSE
Training	3.12
Testing	2.91

difference less than zero indicates that the number of bikes at the station in the current period is in short supply, alerting the need for manual dispatching of vehicles, and vice versa (a demand difference greater than or equal to zero).

1) Data Augmentation: Before building the machine learning model, we implement data augmentation to enrich the few available temporal and geographic information in the original dataset.

a) Augmentation on temporal side: We carve out the time characteristics on week and daily level. We create three dummy variables to denote that four intervals and weekday vs weekend. Conduct One-Hot Encoding to represent the specific hour within a day.

b) Augmentation on spatio side: It is reasonable to think that the surrounding sites and the environment will affect the demand difference for bike-sharing station. Out of this idea, we categorize all places of interests into 11 categories. For each category and station, we determine the total counts of places of interest around by calling the Google places search API based on stations' latitude and longitude. (Here, we take 0.06 miles as the radius threshold to determine the surrounding area)

After the data augmentation, the original variables are transformed into more informative ones: apartment, business, entertainment, food, government, hospital, locality, mall, park, sport, transportation, weekend, and code representing the specific hour.

2) Clustering: Learning from the previous section that significant difference between stations, we perform a clustering analysis of the stations, establishing following models within clusters rather than treating them as a whole. To determine the cluster size, we tune the parameter ranging from 5 to 20 and obtained scree plots as shown as Fig 9.

According to the heuristic elbow method, we take 16 as our desired number of clusters. We implement the K-means based on all features obtained after data augmentation and visualize

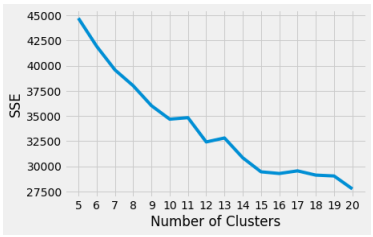


Fig. 9. scree plot

it on the first two principal components axis. The visualization could be found in Fig 10.

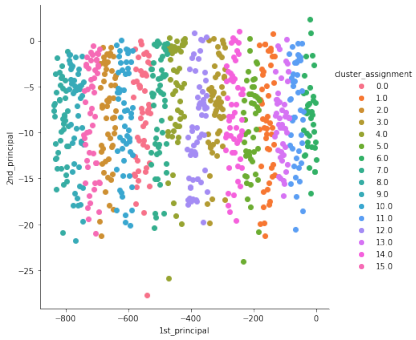


Fig. 10. PCA plot

3) Clarifications:

- We randomly split the original dataset into train (90%) and test(10%) sets.
- Since we discuss classification on the station cluster level, we only take cluster 4 (which shown in figure to be a clear cluster without overlapping with others) as example to help illustrate result.
- All the classification evaluation metrics are performances from test set.

4) Naïve Bayes Classification Model: Naïve Bayesian is a simple but surprisingly powerful predictive modeling algorithm. We take it to fit our demand difference classification model because of its stability and satisfying efficiency on large data training.

For one station at a specific hour, the classification result is obtained by the following formula:

$$\hat{y} = \prod_{i=1}^{n} P(X_j = x_j | C = i) P(C = i), i = 0 \text{ or } 1$$

In which, $P(X_j = x_j|C = i)$ is the class-conditional probability and we take $P(C = i) = \frac{1}{K}$ as the prior probability. We implement NaïveBayes on pyspark library ml. The reported prediction accuracy in test set is 0.534 and the confusion matrix is shown in Table 4.

From the prediction accuracy and the confusion matrix we could tell that the Naïve Bayes classifier does not perform as desired especially in classify class 1 (bikes in short supply). Next, we will resort to neural network.

TABLE IV
CONFUSION MATRIX OF NAÏVE BAYESIAN

	0	1
0	32563	2192
1	18777	2028

5) Convolutional Neural Network: We try to use deep learning method for demand prediction. Based on related work, we fit Conventional Neural Network (CNN).

By increasing the number of convolutional layers in the CNN, the model will be able to detect more complex features in an image. However, with more layers, it'll take more time to train the model and increase the likelihood of overfitting. We input 35 features, choose two intermediate of size 20 and 10, and output 2 neurons since we have 2 possible classes to predict. To avoid overfitting and improve the accuracy, we set blockSize as 128 and MaxIter as 100.

Then, we train the model with training dataset and test it with testing dataset. The confusion matrix could be found in Table 5. The accuracy of CNN is about 62.77%, and the recall is just 5.43%, which is not a very good result. The confusion matrix is shown as Table 5. We also set more parameters like tol, stepSize, blockSize to be smaller than default and more layers, but the results are not better than before, this may because the model is overfitted.

TABLE V
CONFUSION MATRIX OF CNN

	0	1
0	33745	1010
1	19676	1129

VI. DISCUSSION AND FUTURE WORK

In this paper, we utilize three methods to examine the demand status of London Bike Sharing System. To predict the specific demand difference, Prophet model by each station gives a good prediction with low RMSE values. To predict if the supply of the sharing-bikes meets the demand of the sharing-bikes, Naïve Bayes and CNN have similar results. However, the accuracy and recall rate are not good enough to distinguish these two classes.

Our work has several shortcomings. In the Prophet model, the training RMSE exceeds the testing RMSE, which could be an indication of overfitting. In the classification model, we create the spatio variables to replace the information contains in the station id and station position. By doing so, our original aim is to expand the information of each station, and get rid of existing station frameworks. Even if a new station comes up whose id is not in the training set of the model, we still could use our model to make a prediction based on these spatio variables. Unfortunately, the models we proposed in this project could not figure out the patterns.

For future work, more complex models could be considered to address this problem. And more possible predictors could be added to increase the variance of the design matrix.

REFERENCES

[1] DeMaio, Paul. "Bike-sharing: History, impacts, models of provision, and future." Journal of public transportation 12.4 (2009): 3.

[2] Jäppinen, Sakari, Tuuli Toivonen, and Maria Salonen. "Modelling the potential effect of shared bicycles on public transport travel times in Greater Helsinki: An open data approach." Applied Geography 43 (2013): 13-24.

[3] Ashqar, Huthaifa I., Mohammed Elhenawy, and Hesham A. Rakha. "Modeling bike counts in a bike-sharing system considering the effect of weather conditions." Case studies on transport policy 7.2 (2019): 261-268.

[4] El-Assi, Wafic, Mohamed Salah Mahmoud, and Khandker Nurul Habib. "Effects of built environment and weather on bike sharing demand: a station level analysis of commercial bike sharing in Toronto." Transportation 44.3 (2017): 589-613.

[5] Kutela, Boniphace, and Hualiang Teng. "The influence of campus characteristics, temporal factors, and weather events on campuses-related daily bike-share trips." Journal of Transport Geography 78 (2019): 160-169.

[6] Lu, Eric Hsueh-Chan, and Zhan-Qing Lin. "Rental prediction in bicycle sharing system using recurrent neural network." IEEE Access 8 (2020): 92262-92274.

[7] Fournier, Nicholas, Eleni Christofa, and Michael A. Knodler Jr. "A sinusoidal model for seasonal bicycle demand estimation." Transportation research part D: transport and environment 50 (2017): 154-169.

[8] Zhou, Yajun, et al. "A Markov chain based demand prediction model for stations in bike sharing systems." Mathematical problems in engineering 2018 (2018).

[9] Sathishkumar, V. E., Jangwoo Park, and Yongyun Cho. "Using data mining techniques for bike sharing demand prediction in metropolitan city." Computer Communications 153 (2020): 353-366.

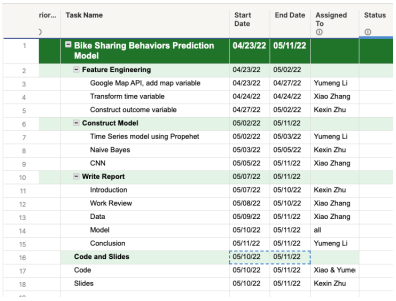
[10] Caggiani, Leonardo, and Michele Ottomanelli. "A modular soft computing based method for vehicles repositioning in bike-sharing systems." Procedia-Social and Behavioral Sciences 54 (2012): 675-684.

[11] Bike-Share Usage in London and Taipei Network https://www.kaggle.com/datasets/ajohrn/bikeshare-usage-in-london-and-taipei-network

VII. CONTRIBUTION

- Kexin Zhu 33.3Final report: Data Augmentation; Clustering + PCA; Naive Bayes Classifier Presentation: Present the work done in report and record the video
- Yumeng Li 33.3Final report: Exploratory data analysis; Time series model Presentation: Present the work done in report
- Final report: Related work; data processing; CNN Presentation: Present the work done in report

VIII. GANTT PLOT



	rior...	Task Name	Start Date	End Date	Assigned To	Status
1		**Bike Sharing Behaviors Prediction Model**	**04/23/22**	**05/11/22**		
2		**Feature Engineering**	04/23/22	05/02/22		
3		Google Map API, add map variable	04/23/22	04/27/22	Yumeng Li	
4		Transform time variable	04/24/22	04/24/22	Xiao Zhang	
5		Construct outcome variable	04/27/22	05/02/22	Kexin Zhu	
6		**Construct Model**	05/02/22	05/11/22		
7		Time Series model using Propehet	05/02/22	05/03/22	Yumeng Li	
8		Naive Bayes	05/03/22	05/05/22	Kexin Zhu	
9		CNN	05/05/22	05/11/22	Xiao Zhang	
10		**Write Report**	05/07/22	05/11/22		
11		Introduction	05/07/22	05/10/22	Kexin Zhu	
12		Work Review	05/08/22	05/10/22	Xiao Zhang	
13		Data	05/09/22	05/11/22	Xiao Zhang	
14		Model	05/10/22	05/11/22	all	
15		Conclusion	05/11/22	05/11/22	Yumeng Li	
16		**Code and Slides**	05/10/22	05/11/22		
17		Code	05/10/22	05/11/22	Xiao & Yume	
18		Slides	05/10/22	05/11/22	Kexin Zhu	