Bike Share Usage Prediction in London

Ford Rylander
fordr@stanford.edu

Bo Peng
bpeng@stanford.edu

Jeff Wheeler
jeffsw@stanford.edu

***Abstract*—The rapid growth in recent years of bike sharing around the world has led to real challenges balancing supply and demand. To provide a possible solution for this situation, bike sharing usage prediction is critical. In this study, a model based on K-mean clustering and polynomial regression is applied to predict the bike sharing usage in a given time period and station. London bike sharing data are used for training and testing our model.**

I. Introduction

Bike sharing is a short-distance bike rental service that allows customers to pick up a bike at one station, use it for transportation as needed, and return it to another station. This transportation model delivers customers the conveniences of a bike without the headaches of bike ownership including theft, parking, storage and routine maintenance.

Demand for bike sharing services has exploded in recent times; as of June 2014, public bike-sharing systems can be found in 712 cities on five continents, with approximately 806,200 bicycles in circulation at a total of 37,500 stations [1].

Demand prediction at bicycle sharing stations is crucial to ensure that there are neither too many bikes (and hence not enough bikes at some other station that needs them) nor too few bikes (and thus lost profits). This paper presents the findings of an algorithm that utilizes historical usage and weather data to accurately predict the number of bikes required at a given station in London.

An examination of prior arts reveals several proposed methods to predict bike demand including an autoregressive moving average (ARMA) algorithm [2], linear regression, and support vector machine (SVM) regression [3]. The ARMA model leverages time-domain autocorrelation to highlight underlying periodicity in the bike demand data, linear regression uses a computationally efficient yet powerful model to fit the time-series data, and SVM regression is more computationally intensive. Taking the pros and cons of these prior art approaches into consideration, this paper presents the performance of a 2nd-order polynomial model that is fit after clustering the training set using the geographical coordinates of the start stations.

II. Data

Historical bike share station demand and weather data are publicly available for download as a list of rides and weather data in a given day. The bike rental raw data includes time and location information for each rental. These raw data are then processed with a binning algorithm to separate the data in the required time-segment granularity and then time-correlating this with the weather data. The ride counts are then shifted by several time intervals and joined back with the original data to generate the historic features.

The list of all bike rides is available directly from Transport for London [4].

III. Features

We designed a pipeline structure that transforms our basic feature set into a 2nd-order polynomial feature set, and then reduces it to only the best features.

X — Original Feature Set

We have aggregated these data into a single design matrix, where each row represents an hour-long segment of time for any given station. Each row in X is given by

$$\begin{aligned} x^{(i)} = [&d_1\ d_2\ d_3\ d_7\ weather\ mean\_temp\ \ldots \\ &weekday\ is\_weekend\ \ldots \\ &h_1\ \ldots\ h_5]^T \in \mathbb{R}^{14}. \end{aligned}$$

The values in this vector are given by

1-15	$Cluster_{11} \rightarrow LR$	$Cluster_{12} \rightarrow LR$	$Cluster_{13} \rightarrow LR$	$Cluster_{14} \rightarrow LR$		
16-30	$Cluster_{21} \rightarrow LR$	$Cluster_{22} \rightarrow LR$	$Cluster_{23} \rightarrow LR$	$Cluster_{24} \rightarrow LR$		
31-50	$Cluster_{31} \rightarrow LR$	$Cluster_{32} \rightarrow LR$	$Cluster_{33} \rightarrow LR$	$Cluster_{34} \rightarrow LR$	$Cluster_{35} \rightarrow LR$	$Cluster_{36} \rightarrow LR$

Figure 1. Geographic-Based K-Means on Stations Grouped by Overall Popularity

d_1, d_2, d_3, d_7
: d_n is the number of rides during the same hour n days prior.

$weather$
: is a binary representation of the weather events from the day of $x^{(i)}$.

$mean\_temp$
: is the mean temperature on the day of $x^{(i)}$.

$weekday$
: is an integer representation of the day of the week, with $weekday \in [0, 6]$.

$is\_weekend$
: is a single bit that encodes whether $x^{(i)}$ is on a weekend or not.

$h_1, \ldots, h_5$
: h_n is the demand n hours prior to the hour of $x^{(i)}$.

$$\Downarrow$$

X' — Polynomial Feature Set

We create X' by applying a 2nd-order polynomial transformation to X. Each row is given by

$$x'^{(i)} = [1 \quad x_1^{(i)} \quad x_2^{(i)} \quad \ldots \quad x_{14}^{(i)} \quad \ldots \quad x_1^{(i)}x_1^{(i)} \quad x_1^{(i)}x_2^{(i)} \quad \ldots \quad x_{14}^{(i)}x_{14}^{(i)}]^T \in \mathbb{R}^{120}.$$

$$\Downarrow$$

X'' — Reduced to Best Features

Finally, we perform feature selection to limit the variance problems and reduce regression complexity by selecting the best 75 features:

$$x''^{(i)} = feature\_selection(x'^{(i)}) \in \mathbb{R}^{75}.$$

We chose 75 features experimentally by optimizing the test error, as shown in Figure 2. With large numbers of features, the test error degrades because we overfit the training samples.

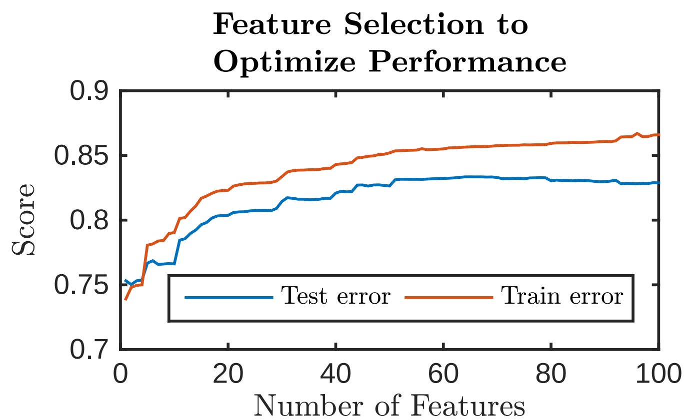

Figure 2. Selecting number of features

IV. Models

We divide the dataset into stations by overall popularity, and then cluster these divided data based on GPS coordinates of each starting station. In each cluster (denoted as $Cluster_{rs}$ in Figure 1), we fit a 2nd-degree polynomial model using linear regression. When predicting, we find the cluster closest to the station whose demand we're trying to predict and use the fitted polynomial for that cluster. We use more data for the less popular stations (31-50), while reducing the number of stations in each cluster by increasing the number of clusters. This is necessary because the less popular stations have data which is more sparse, and thus we need more data to train on.

A 2nd-order polynomial was chosen experimentally to optimize the performance of the model. The performance of the model with varying degrees is shown in Figure 3.

The K-Means within each set of stations uses the GPS coordinates to find stations that are located near to each other. We train on features $g^{(i)}$ given by

$$g^{(i)} = \begin{bmatrix} latitude \\ longitude \end{bmatrix} \in \mathbb{R}^2,$$

and then perform K-Means by optimizing the

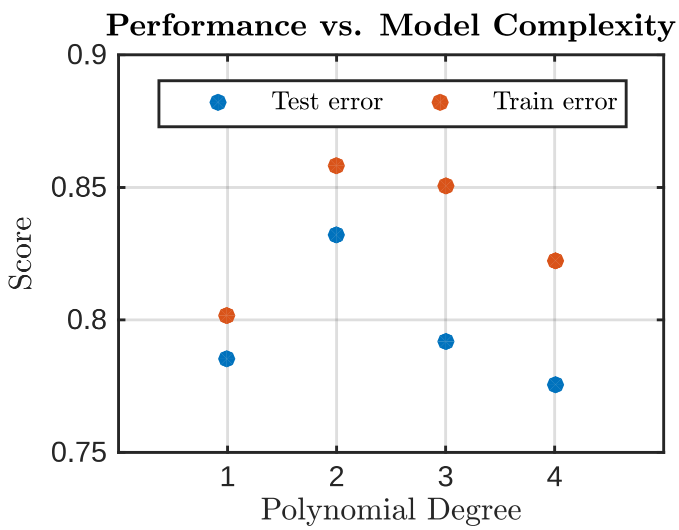

Figure 3. Performance vs. Model Complexity

cluster centers according to

$$\arg\min_{\mu,c} \sum_{i}^{m} \|g^{(i)} - \mu_c^{(i)}\|^2 \ .$$

The use of K-Means was inspired by [3]. The preliminary K-Means clustering algorithm used Euclidean K-Means in the original feature space; however, we found that K-Means clustering on the geographic coordinates was more effective at predicting the bike demand. The geographic-based K-Means linked stations that were likely to have similar traffic patterns. Furthermore, segmenting stations by total demand prior to K-Means, as shown in Figure 1, helped separate the stations that performed differently. Further work in [5] could also be useful for clustering the stations according to their traffic profiles.

Each regressor cell performs regression on the 2nd-order polynomial feature set according to

$$h_\theta(x''^{(i)}) = \theta^T x''^{(i)} = \sum_{j=0}^{k} \theta_j x_j''^{(i)} \ .$$

V. Results

We tried several regression algorithms inside each cluster, and have plotted the overall performance of each of them in Figure 4. We found that a 2nd-order polynomial regressor performed best among all the algorithms we tried.

The performance metric that we used was explained variance, which computes the normalized variance of the difference between the actual and predicted bike demands in a given hour. In particular, explained variance is defined as

$$Explained\_Variance = 1 - \frac{Var(y - \hat{y})}{Var(y)} \ .$$

The performance results for each of our models are summarized in Table I.

The learning curve shows, satisfyingly, that our training and test scores converge to ≈ 0.83. This suggests that we have sufficient data; to improve our learning algorithm, we required additional features that convey unique information about the problem.

The polynomial regression (linear regression over our polynomial feature set) performs very similarly to ridge regression, which adds a penalty term for outlying data. Because the penalty term was not useful for our application, it was rarely significant and therefore ridge and simple polynomial regression performed almost identically. Because the extra complexity of ridge came without benefit, we chose the simpler model.

VI. Discussion

Our learning algorithm performs very well on high traffic stations. Performance degrades on less popular stations as their demand is not as predictable because their usage is sporadic in nature. That is, the number of bikes taken each hour from less popular stations is much more difficult to predict because only a few bikes are taken in an hour and they do not follow regular patterns. Weekend traffic is also difficult to predict at the high-traffic stations because most of the traffic at those stations occurs on weekdays (e.g. during commutes or lunch breaks). Their weekday traffic likely corresponds to

Table I
Model Performance

1st-order polynomial regression	0.801	0.786
2nd-order polynomial regression	0.858	0.832
3rd-order polynomial regression	0.851	0.791
4th-order polynomial regression	0.823	0.775
SGD (L2 regularization)	0.637	0.620
Ridge, $\alpha = 0.5$	0.856	0.831
Ridge, $\alpha = 2.0$	0.865	0.829

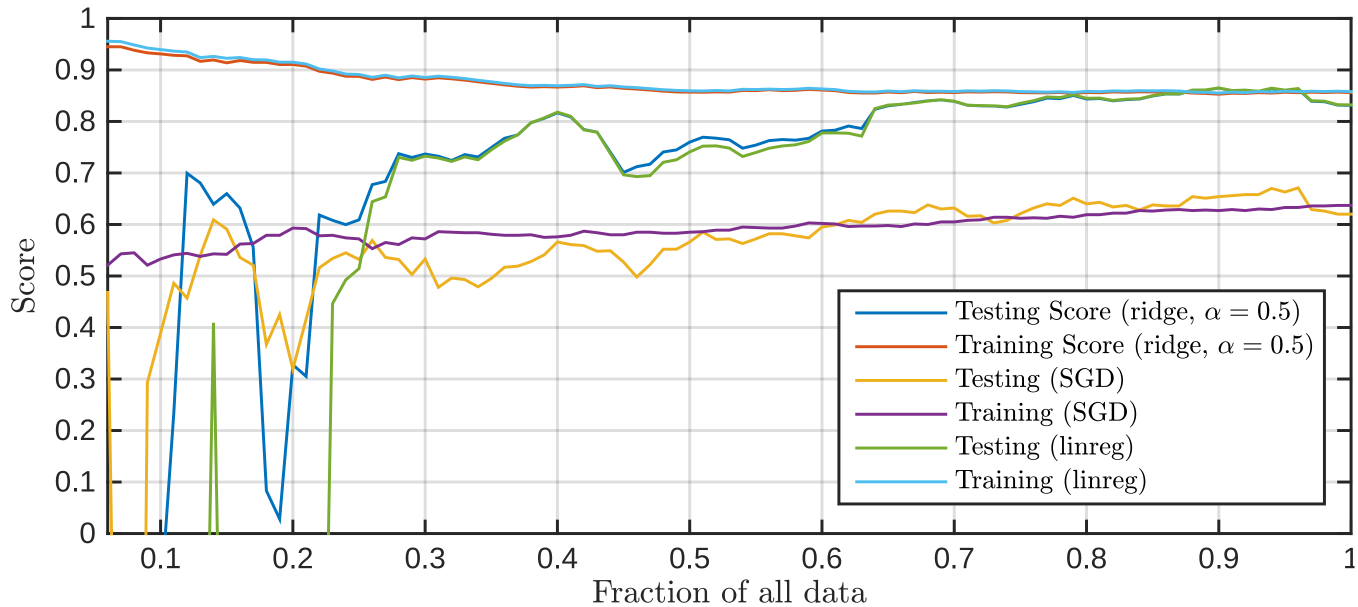

Figure 4. Learning Curve

local events happening nearby, which we did not attempt to capture in our algorithm.

We expected to be able to predict the total usage for all stations, but were surprised by our ability to predict per-station traffic quite well, to which we credit our usage of K-Means clustering on the geographic station coordinates.

Figure 5 shows our predicted bike demand vs. actual bike demand for a particular week in January 2013. This particular station is relatively high-traffic and our algorithm does a very good job of predicting demand. Many other high-traffic stations share a similar profile; some others have two peaks each weekday corresponding to morning and evening commutes or morning and lunch commutes.

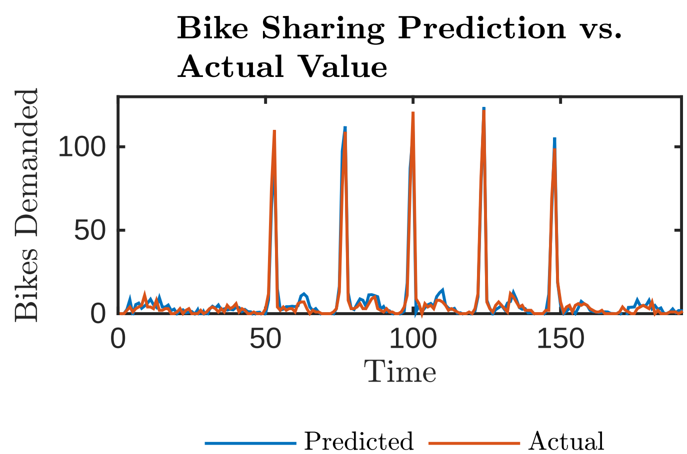

Figure 5. Bike Sharing Prediction vs. Actual Value

VII. Future

The fundamental limit of our algorithm was determined by the features that were available. Given more time, we would like to incorporate other features such as subway ridership which we believe would be an accurate predictor of bike demand. We would also like to implement geographic visualizations of the data to better understand how geography affects our learning algorithm's performance.

References

[1] S. A. S. et al., "Public bikesharing in north america during a period of rapid expansion: Understanding business models, industry trends and user impacts," in *Mineta Transportation Institute (MTI)*, 2015, p. pp.5.

[2] A. Kaltenbrunner, R. Meza, J. Grivolla, J. Codina, and R. Banchs, "Urban cycles and mobility patterns: Exploring and predicting trends in a bicycle-based public transport system," *Pervasive and Mobile Computing*, vol. 6, no. 4, pp. 455–466, 2010.

[3] H. Xu, J. Ying, H. Wu, and F. Lin, "Public bicycle traffic flow prediction based on a hybrid model," *Appl. Math. Inf. Sci*, vol. 7, pp. 667–674, 2013.

[4] Transport for London. City of London. [Online]. Available: https://www.tfl.gov.uk/

[5] H. Wu, X. Fang, and H. Xu, "Station segmentation of hangzhou public free-bicycle system based on improved randomized algorithm," in *Machine Learning and Cybernetics (ICMLC), 2011 International Conference on*, vol. 4. IEEE, 2011, pp. 1588–1592.