# Final Report

Group 13

2021/12/16

## Introduction

Bike-Sharing systems are currently widely introduced in urban cities to solve the "last mile" problem, improve the link between other modes of transportation. Bike-sharing systems facilitate the use of public transportation, enhance traffic troubles, as well as minimize greenhouse gas emissions. However, the availability and accessibility of sharing bikes could be a problem since the demand and supply of bikes are not stable.

This project aims to use machine learning and data-mining-based algorithms to predict the demand for rental bikes in Seoul city at each hour in order to provide a stable bike supply. The project is based on the dataset downloaded at UC Irvine Machine Learning Repository. The dataset contains the hourly public rental number of Seoul bikes with date and weather information (Temperature, Humidity, Wind speed, Visibility, Dew point, Solar radiation, Snowfall, Rainfall) for one year from December 2017 to November 2018, 365 days. The number of rental bikes rented at every hour is determined from the bike rental history data collected from the Seoul Public Data Park website of South Korea.[1]

As all modes of transportation depend primarily on the weather conditions, including cycling, we would assume that the corresponding climate conditions have effects on the total number of rental bikes rented at each hour. And we would also assume that date parameters such as weekdays may enhance the performance of the prediction model. Thus, weather details and date parameters would be the covariates of the model.

## Methodology

### Linear Regression

Linear Regression Model (LM) is the simplest method to analyze the relationship of outcome Y and predictor X. The model is defined as below.

$$Y = \beta_0 + X^T\beta \tag{1}$$

In the regression, after we input the data of X and Y, the coefficients $\beta_0$ and $\beta$ could be estimated by solving the equation.

### LASSO

### Ridge

### Random Forest

### Support Vector Machine

Support Vector Machine(SVM) was designed for classification, and later the idea was generalized to regression. The optimization idea in SVM classification is to increase the amount of separation between two

classes. However, in SVM regression, we would like to form a "band" around the true regression function that contains most of the points. With this purpose, the loss function is defined as below.

$$\mu(x) = \beta_0 + x^T\beta \tag{2}$$

If the point (X,Y) = (x,y) is such that $|y - \mu(x)| \leq \epsilon$, then the loss is taken to be zero; if $|y - \mu(x)| \geq \epsilon$, then the loss is taken to be $|y - \mu(x)| - \epsilon$. The main goal is to find a $\mu(x)$ such that most points with the loss taken to be zero.

#### Neural Network

Neural networks (ANNs), also known as artificial neural networks (ANNs), are computing systems inspired by the way biological neural networks in the human brain process information. Building Neural networks is a widely adopted method in machine learning. The tasks to which neural networks are applied tend to fall within the following broad categories including regression analysis, classification, and data processing. Neural networks provide the current best solutions in solving many computer science problems such as image recognition, speech recognition, and natural language processing.

In this project, we are going to solve a regression problem and make a prediction using neural networks. Instead of applying Convolutional neural network or Recurrent neural network, we adopt the most basic one: *Feedforward Neural Network* (FFNN). Feedforward neural network was the first type of artificial neural network devised. Different from recurrent neural networks, the connections between nodes do not form a circle. It has only one direction from input layer formed by input nodes, through hidden layers and to out the output layer formed by output nodes. Specifically, we will adopt *Multilayer Perceptron* (MLP). A multilayer perceptron consists of at least three layers of nodes: an input layer, a hidden layer and an output layer. There are 8 hidden layers in our model.

The toolkit we used for building the neural network is *Keras*. Keras is a deep learning API (application programming interface) written in Python, running on top of the machine learning platform *TensorFlow*. The activation function we adopted are ReLU and Linear. Rectified Linear Unit (ReLU) is an activation function that defines the positive part of its arguement. It can be expressed as:

$$f(x) = x^+ = \max(0, x)$$

Linear is an activation function that simply returns the the argument itself. We also add regularizers that apply to a L1 and L2 penalty on the layer's output. L1 penalty is the penalty term in L1 regression or Lasso regression. L2 penalty is the penalty term in L2 regression or Ridge regression. Regularizers are used to avoid model overfitting and improve the robustness and generality.

## Dataset description

## Evaluation indices

## Model development

## Result

## Discussion

## Citation

E, S., Park, J., & Cho, Y. (2020). Using data mining techniques for bike sharing demand prediction in Metropolitan City. Computer Communications, 153, 353–366. https://doi.org/10.1016/j.comcom.2020.02.007