**Seoul Bike Sharing Demand Prediction**

**Anupam Mishra , Kartike, Animesh Chakraborty**

**Data science trainees,**

**AlmaBetter, Bangalore**

* **Abstract:**

Rental Bike Sharing is the process by which bicycles are procured on several basis- hourly, weekly etc. This phenomenon has seen its stock rise to considerable levels due to a global effort towards reducing the carbon footprint, leading to climate change, unprecedented natural disasters, ozone layer depletion, and other environmental anomalies.

In our project, we chose to analyze a dataset pertaining to Rental Bike Demand from South Korean city of Seoul, comprising of climatic variables like Temperature, Humidity, Rainfall, Snowfall, Dew Point Temperature, and others.

We applied machine learning models such as linear regression, regularized linear regression, Decision tree, random forest, and gradient boosting to predict consumer demand for bike sharing in Seoul. The data were retrieved from Seoul Public Data Park website, which records the counts of public bike rentals in Seoul of Korea from January 1 to December 31, 2020. We found that the two best models are the random forest and the Gradient boosting. Among the 17 features Temperature and Hour feature are the most important in model prediction. While almost all features are the least important, we found that they help enhance the performance of the models.

**Keywords:**-Machine learning, Linear Regression, Correlation Analysis, Bike Sharing Demand Prediction, Seoul.

**Introduction**

Currently rental bikes are introduced in many urban cities for the enhancement of mobility comfort. The purpose of this movement is to modernize cities and encourage people to head to a green world**.** The goal is to facilitate the commute in the Seoul and reduce the amount of cars and the pollution. Indeed, the development of the way to commute reduced the use of cars to go to work and visit the city.

It is important to make the rental bike available and accessible to the public, as it provides many alternatives to commuters in metropolises. There are a lot of advantages to bike rents, it is convenient because it permits people not to keep the bike all day long, whether it is at work or at school. Furthermore it is the healthiest way to travel and it has many environmental benefits.

This dataset contains features which are in the form of table:

|  |  |
| --- | --- |
| Features | Description |
| Date | Contains year-month-day of rented bike |
| Rented Bike count | Count of bikes rented at each hour |
| Hour | Hour of the day |
| Temperature | Temperature in Celsius |
| Humidity | Humidity in percentage |
| Windspeed | Wind speed in m/s |
| Visibility | Visibility |
| Dew point temperature | Dew point temperature |
| Solar radiation | Solar radiation |
| Rainfall | Rainfall in mm |
| Snowfall | Snowfall in cm |
| Seasons | Winter, Spring, Summer, Autumn |
| Holiday | Holiday/No holiday |
| Functional Day | NoFunc(Non Functional Hours), Fun(Functional hours) |

|  |  |
| --- | --- |
| dataset | We have saved the data as in data set variable. |
| dataset['Month'] | This variable gives the numerical value of the month |
| dataset['Weekdays\_or\_weekend'] | This variable gives the is in weekend or weekdays |
| numeric\_features | This variable gives the column name which is having numerical value. |
| categorical\_columns | This variable gives the column name which is having non numerical value. |
| correlation | In this variable we saved the data of correlation between the variables. |
| df | In this variable we saved the copy of dataset variable. |
| df['Spring'] | This variable gives the data of month which are in spring month. |
| df['Summer'] | This variable gives the data of month which are in summer month. |
| df['Autumn'] | This variable gives the data of month which are in autumn month. |
| df['Winter'] | This variable gives the data of month which are in winter month. |
| dependent\_variable | This variable gives the column which is dependable. |
| independent\_variables | This variable gives the column which is independable. |
| y | This variable gives the square root of dependent variable that is rennet bike count. |
| X\_train | It contains independent variables training data |
| X\_test | It contains independent variables test data |
| y\_pred\_train | This variable stores the values of predicted training data |
| y\_pred | This variable stores the values of predicted test data |
| MAE | This variable gives the mean\_absolute\_error. |
| MSE | This variable gives the mean\_squared\_error. |
| RMSE | This variable gives the square root of mean\_squared\_error. |
| R2 | This variable gives the of r2 score. |
| adj\_r2 | This variable gives the adjusted r2 score. |

**Problem Statement**

The project consists of one year data (from 1.12.2017 to 30.11.2018) of count of rented bikes combined with weather information and time. The main idea is to forecast the bike usage based on the past information.

The goal of the company Seoul Bike is providing the city with a stable supply of rental bikes. It becomes a major concern to keep user satisfied. The crucial part is the prediction of bike count rents at each hour for a stable supply of rental bikes. We can suppose that this study could be reported to the company 'Seoul Bikes'. We think it could help them knowing if yes or not they have to supply bikes stations in the city, in order to keep a good satisfaction of the customers.

**Steps involved:**

* **Exploratory Data Analysis**

Exploratory data analysis is an statistical way of understanding the data which is usually done in a visual way. The graphs plotted in exploratory data analysis are for better understanding of data to the analyst.

After loading the dataset we performed this method by comparing our target variable that is Rented\_Bike\_Count with other independent variables. This process helped us figuring out various aspects and relationships among the target and the independent variables. It gave us a better idea of which feature behaves in which manner compared to the target variable.

For the current data set , Since we have to predict the number of bikes that will be rented, the best way to begin is with the variable to predict, “count". We can stratify the "count" distribution as box plots for the categorical variables, and draw the "count" and numeric variables in another plot.

* **Null values Treatment**

After the data is loaded, The missing data is checked using is.na() or isnul() function .The output depicted that there was no missing values in our dataset.

So our dataset does not contain any missing values.

* **Encoding of Categorical features**

We used One Hot Encoding to produce binary integers of 0 and 1 to encode our categorical features because categorical features that are in string format cannot be understood by the machine and needs to be converted to numerical format.

Categorical variables- Seasons, Functioning Day, and Holiday- were converted coded into numerical depictions to fit our Model to predict Bike rented count.

* **Feature Engineering**

To make the data tenable for understanding and further analysis , the data set was analyzed for identifiable statistical trends and patterns. After preliminary analysis, the following steps were undertaken to transform the data into a systematically workable dataset:

i) Convert the data-time attributes in proper format and we separate day, month, year and hour into separate columns so that it is easy to perform operations on the data.

ii) Divide temperature, humidity and windspeed variables into categories by doing so we can better the accuracy in the model.

iii) Create dummy variables for season attribute, here season variable is broken down into 3 binary variables i.e. spring, summer and winter.

* **Standardization of features**

Our main motive through this step was to scale our data into a uniform format that would allow us to utilize the data in a better way while performing fitting and applying different algorithms to it.

The basic goal was to enforce a level of consistency or uniformity to certain practices or operations within the selected environment.

* **Feature Selection**

In these steps we used algorithms like Decision Tree, Random Forest and Gradient Boosting to check the results of each feature i.e. which feature is more important compared to our model and which is of less importance.

Among the 17 features Temperature and Hour feature are the most important in model prediction. While almost all features are the least important, we found that they help enhance the performance of the models.

* **Fitting different models**

For modeling we tried various algorithms like:

1. **Linear Regression**
2. **Regularized Linear regression**
3. **Lasso regression**
4. **Ridge regression**
5. **Elastic Net regression**
6. **Decision Tree**
7. **Random Forest**
8. **Gradient Boosting**

Hence as the data is understood properly, random forest predictive model and gradient boosting predictive model can be built for this data to predict the count variable

* **Algorithms:**

**1.Linear Regression**

Linear regression is the most widely used and simplest method to predict demand in various contexts. Due to its simplicity and straightforward economic intuition in explaining the relationship between predictors and the outcome, we use linear regression as a benchmark against which other more advanced models are compared for their predictive power. The linear regression model is given as

y=β0+∑inβixi+ε

Where βi is the coefficient of feature xi , β0 is the constant, and ε is the random error

**2. Regularised Linear Regression**

Regularization helps us to deal with the problem of overfitting by reducing the weight given to a particular feature x. This allows us to retain more features while not giving undue weight to one in particular. Regularisation is mediated by a parameter λ, as can be seen in the cost function:

J(θ)=12m(∑i=1m(hθ(x(i)−y(i))2)+λ2m(∑j=1nθ2j)

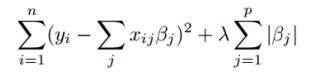
The first term is essentially the mean-squared-error term, whilst the additive term multiplies the sum of the square of the parameters (θ) by λ over 2m, where m is the number of training examples. Since the objective is to minimize J(θ) (minθJ(θ)) using a large λ will require small values of θj in order to achieve minima.

**a). Lasso Regression**

Lasso regression is a type of linear regression that uses shrinkage. Shrinkage is where data values are shrunk towards a central point, like the mean. The lasso procedure encourages simple, sparse models (i.e. models with fewer parameters). This particular type of regression is well-suited for models showing high levels of muticollinearity or when you want to automate certain parts of model selection, like variable selection/parameter elimination.

The acronym “LASSO” stands for Least Absolute Shrinkage and Selection Operator.

A tuning parameter, λ controls the strength of the L1 penalty. λ is basically the amount of shrinkage:



When λ = 0, no parameters are eliminated. The estimate is equal to the one found with linear regression.

As λ increases, more and more coefficients are set to zero and eliminated (theoretically, when λ = ∞, all coefficients are eliminated).

As λ increases, bias increases.

As λ decreases, variance increases.

**b)Ridge Regression**

Ridge regression is a model tuning method that is used to analyse any data that suffers from multicollinearity. This method performs L2 regularization. When the issue of multicollinearity occurs, least-squares are unbiased, and variances are large, this results in predicted values being far away from the actual values.

||y - Xw||^2\_2 + alpha \* ||w||^2\_2

c**) Elastic Net**

Elastic net is a penalized linear regression model that includes both the L1 and L2 penalties during training. Using the terminology from “The Elements of Statistical Learning,” a hyperparameter “alpha” is provided to assign how much weight is given to each of the L1 and L2 penalties

**3.Decision Tree**

Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. A tree can be seen as a piecewise constant approximation.

**4.Random Forest**

Random forest is a almighty tool which ensembles decision trees and bagging .The base learner of random forests is a binary tree constructed by recursive partitioning (RPART) and then developed using classification and regression trees. Binary splits of the parent node of a random forest splits data into two children’s nodes and increases homogeneity in children nodes compared to parent nodes. Note that a random forest does not split tree nodes based on all variables; instead, it chooses random variable subsets as candidates to find the optimal split at every node of every tree. Then the information from the n trees is aggregated for classification and prediction. Random forests also provide the importance of each feature by accumulated Gini gains of all splits in all trees representing the variable discrimination ability:

imporj=1#trees∑v∈xjGain(xj,v)

Where Gain(xj,v) is the gain of the Gini index of feature xj combined with node v.

**5. Gradient Boosting**

Gradient Boosting [9] like Random Forests is an ensemble learning method. Similar to latter, it uses multiple weak learners which are combined to form a strong learner. But unlike its Random Forests, Gradient Boosting as the name suggests uses boosting.

Boosting methods work iteratively to create a new learner at every stage; these new learners are then trained on the error residuals at a current iteration to produce new learners which are stronger than the previous stage. Applied to decision trees, every decision tree is works on the error residuals of the previous iteration to produce a better decision tree. The collection of these decision trees is then used as the overall model for predicting values.

**Model Performance :**

Model can be evaluated by various metrics such as:

**R-Square Value** is the goodness-of-fit and a statistical measure of how close the data are fitted to the regression line.

**Adjusted R-squared** compares the explanatory power of regression models that contain different numbers of predictors. It calculates R-Square of only Independent Variables those are statistically significant.

A minute difference between R-Square and Adjusted RSquare suggests all our Independent Variables being

significant, despite both values being on a relatively lower

side.

**Conclusion**

The best model is the random forest model and gradient boosting model in our study, and the most important features are temperature and Hour.

This study shows that the rents of bikes are influenced by a lot of features. In this study, we understood that many Koreans usually and mainly rent bikes during the week days, so we supposed that the main use is to go to school or work. There are also many conditions which contribute to the variation of number of rents like the day of the week, the moment of the day and weather conditions. Weather conditions are also very important because there are more rents during spring and summer. And as we expected more people are set to rent bikes when the weather is favorable.