Bike Sharing Demand Prediction

(Regression)

Sudhanshu Kumar

Data science trainees, AlmaBetter, Bangalore

Abstract:

Presently, rental bikes are available in many urban areas to promote commuting convenience and minimize pollution. Regular availability of rental bikes for the municipality is becoming a key concern because it is crucial to make them accessible and available to the general public at the appropriate time to reduce latency. Predicting the number of bikes needed to keep the right number of rental bikes every hour is essential.

Regression prescriptive analytics is used in this project to forecast demand for bike sharing. In this project, we are provided with a seoulbikedata CSV file which contains 8760 rows and 14 features, we aim to explore the data using various pandas manipulation techniques, for that first of all we have to do data cleaning (remove the duplicates, missing value, outlier, and incorrect data types) then perform feature engineering. then we check multicollinearity between the features (if exist, then we have to remove that feature). After that we perform one hot encoding for categorical columns, then we split the data set into test and train, and finally using various machine learning algorithms we can find the evaluation matrix.

1. Problem statement:

This project explores the correlation between the rental bike usage per hour and the numerous predictors, including meteorological and time information. The dataset provides the number of public bikes rented in the Seoul Bike Sharing System at each hour, together with the related weather reports and date information.

The main aim of this project is to build a predictive model which can predict the number of bikes required each hour for the stable supply of rental bikes.

2. Introduction:

Bike sharing is the statutory requirement of a fleet of bicycles across a channel of well-placed "bike rental stops", generally dispersed throughout an urban area, that can be used by various user types (such as daily visitors or irregular users) for short-term rentals enabling point-to-point journeys. In the past few years, bike sharing has attracted considerable attention as part of initiatives to promote biking, strengthen the very first connectivity to certain other means of transport, and lessen the impact of transportation on the climate. The expansion of the biking culture, increases in the usage of vehicles, the lowering of greenhouse gas footprints, the preservation of community wellness, and traffic problems all are greatly influenced by bike sharing. Because of this steady increase in customers, it is essential to quantify how often rental bikes must be available for the smooth functioning bike sharing system. This project uses a machine learning algorithm to forecast the number of rental bikes anticipated at each hour.

3. **Feature Description:**

1. Rented Bike Count

   total number of bikes rented

2. Time information

   a) Hour

      data is given for the period of 24 hours (i.e., 0 hours to 23 hours)

   b) Date

      365 unique values for each day

3. Weather information

   a) Temperature (^{O}C)

   b) Humidity (%)

   c) Wind speed (m/s)

   d) Visibility (10m)

   e) Dew point temperature (^{O}C)

   f) Solar Radiation (MJ/m2)

   g) Rainfall(mm)

   h) Snowfall (cm)

   i) Seasons

      Here data is given for all four seasons ('Winter' 'Spring' 'Summer' 'Autumn')

4. Holiday

   Two unique value is given 'No Holiday' 'Holiday'

5. Functioning Day

   Two unique value is given i.e. ('Yes' and 'No').

4. **Data information**

In this case, we apply certain pandas’ tools to get insight from data

a) **shape ()** method to find the number of features and rows in data sets

We observe that:

our dataset has 8760 rows and 14 features

Based on the preliminary evaluation, it is discovered that the data was broadly decent, except for a few columns with incorrect data types

b) **head ()** method to preview the top five rows

c) **.info ()** method to preview null data and data types

We observe the following:

Few columns have inappropriate datatype, hence we have to correct it, because it results in a decrease in memory usage

d) **. duplicate ()** method to check duplicate values

We observe that:

There are no duplicate rows

e) We are plotting a boxplot for a column to detect the outlier for particular columns

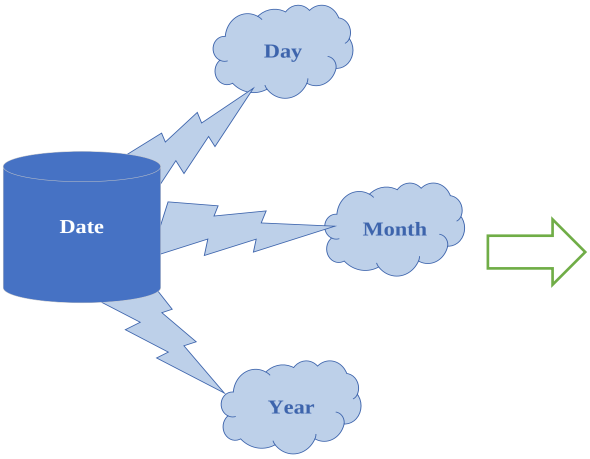
5. Data cleaning

Correcting incorrect datatypes for date columns:

```
bike_df['Date'] = pd.to_datetime(df['Date'], format='%d/%m/%Y')
```

6. Feature engineering

(a) For date columns

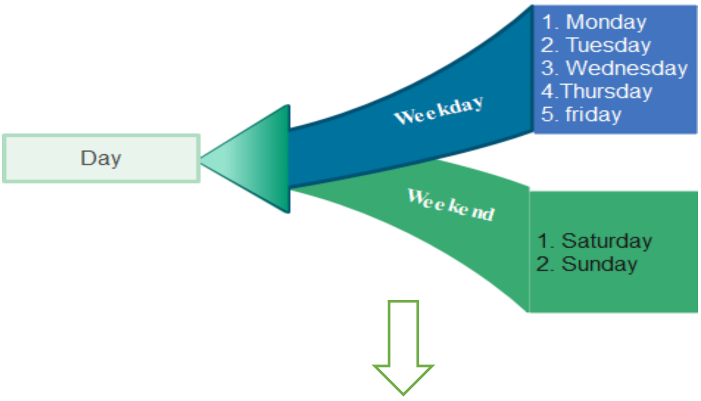


```
bike_df['Year']=bike_df['Date'].dt.year

bike_df['Month']=bike_df['Date'].dt.month_name()

bike_df['Day']=bike_df['Date'].dt.day_name()
```

(b) Creating new categorical columns type of day



```
bike_df['type of day']=bike_df['Day'].apply(lambda x: "weekend" if x=='Saturday' or x=='Sunday' else "weekday")
```

7. Exploratory data analysis:

(a) Number of rented bikes on weekdays and weekends:

- ✓ 73% of bikes booked on a weekday and remaining 23% are book on weekend
- ✓ the possible reason for this is people have the compulsion to go their office, hence they book bike so that they reach office on time.
- ✓ On the weekend people booked bikes only for their pending work.

(b) Rented bike count for each hour

- ✓ When we plot rented bike count for 24 hours by taking categorical variable as a hue parameter, we found that:
- ✓ Rented bike count follows a similar pattern for each of the four seasons
- ✓ On office stating time (08:00 for the morning shift and 18:00 for night shift) and office leaving time (08:00 for evening shift and 18:00 for night shift) a maximum number of bookings.
- ✓ Booking count on a working day is more than a holiday.
- ✓ Bikes are only booked on the functioning day, which means that there is no emergency provision.
- ✓ For each month, the peak time is 08:00 hours in the morning and 16::00 in the evening, the possible reason for this timing is office hours.
- ✓ December, January and February have a smaller number of bike bookings as compared to other months, the possible reason for this temperature is very less in these months.
- ✓ Saturday and Sunday (weekend)do not show a peak on 08:00 hours and 16:00 hours, a possible reason for this is the holiday.

(c) Bike count for a different month

- ✓ December, January and February have negative mean temperatures, which hampers the rented bike count
- ✓ Month which has negative mean temperature having less number of the booking.
- ✓ hence, it can conclude that due to lower temperatures people prefer to travel with other mediums of transportation instead of using bikes.

(d) Rented bike count for different seasons

- ✓ for the summer and winter season, a very less number of the bike are booked on Monday and Sunday
- ✓ for the spring and summer season, Thursday has the maximum number of bike booking
- ✓ for the Autumn season, Monday has the maximum number of bike booking and Tuesday have the least.

8. Data Preparation

(a) Correlation among variables

It indicates that the correlation between two variables

It is classified into three major categories:

a) Positive correlation

It indicates an increase in one variable results in an increase in another variable.

b) Negative correlation

It indicates an increase in one variable results in a decrease in another variable.

c) Zero correlation

It indicates an increase in one variable results in no change in the other variable (No correlation).

Following conclusion drawn from the correlation plot

- ✓ Temperature shows high collinearity with dew point temperature
- ✓ Humidity and visibility are moderately correlated.
- ✓ temperature and hour show a positive correlation with rented bike count
- ✓ As per domain knowledge, their dew point temperature column is irrelevant to finding the rented bike count. Hence, we will drop dew point temperature columns.

(b) Measures of central tendency

- ✓ Target variable i.e., "Rented Bike Count" is positively skewed and we know that as per the assumption of linear regression, the target variable must be normally distributed, hence using "sqrt transform" or "log transform" we can convert it into normal form.

(c) Check for multicollinearity

In a regression model, multicollinearity develops when independent variables are correlated. Because control variables should be independent, this is a concern. If the correlation between the variables is strong enough, it may be difficult to fit the model and understand the data.

- ✓ In our project we calculate VIF to find the multicollinearity variable.

9. Machine learning algorithm

(a) Linear regression:

The linear relationship between a dependent variable and a group of independent variables is investigated by the statistical model known as linear regression. When there is a linear relationship between two variables, then the change in the value of the independent variable changes the value of an independent variable.

Assumption:

a) Multi-collinearity: It is very little or no multi-collinearity in the data which means that independent variables or features have a dependency on them.

b) Auto-correlation: There is very little or no auto-correlation in the data. It occurs when there is a dependency between residual errors.

c) Relationship between variables: It is assumed that there is a linear relationship between response and feature variables.

(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.

- It shrinks the parameters. Therefore, it is used to prevent multicollinearity
- It reduces the model complexity by coefficient shrinkage

(c) Lasso Regression

The word "LASSO" stands for Least Absolute Shrinkage and Selection Operator. It is a statistical formula for the regularisation of data models and feature selection.

Lasso regression is a regularization technique. It is used over regression methods for a more accurate prediction. This model uses shrinkage. Shrinkage is where data values are shrunk towards a central point as the mean.

- Performs L1 regularization, i.e. adds penalty equivalent to the absolute value of the magnitude of coefficients
- Minimization objective= 'least squares objective'+ α * (sum of the absolute value of coefficients)

'Least squares objective'=the linear regression objective without regularization

(d) Ridge with GridSearchCV

It is a method for fine-tuning parameters. The key to how this method works is that it painstakingly constructs and evaluates the model for every possible combination of algorithm parameter combinations that are given in a grid. As a result, we may claim that this algorithm is search-based.

(e) Polynomial Regression

To transform linear regression into polynomial regression, certain polynomial terms are added to it due to the non-linear relationship between the dependent and independent variables.

(f) Elastic Net

Elastic net linear regression regularises regression models by using the penalties from the lasso and ridge algorithms. To improve the regularisation of statistical models, the strategy combines the lasso and ridge regression approaches.

(g) Decision Tree Regressor

A decision tree uses a tree-like structure to generate regression models. It incrementally develops an associated decision tree while segmenting a dataset into smaller and smaller sections. The outcome is a tree containing leaf nodes and decision nodes. Two or more branches, one for each value of the characteristic under test, make up a decision node. A choice regarding the numerical aim is represented by a leaf node. The root node is the topmost decision node in a tree and corresponds to the best predictor. Both category and numerical data can be handled by decision trees.

(h) Random Forest

A decision tree bagging algorithm known as Random Forest builds several decision trees from a randomly chosen subset of the training set, gathers the labels from these subsets, and then averages the final prediction based on how often a given label has been correctly predicted across all of the decision trees.

(i) Gradient Boosting

Gradient boosting decision trees integrate several weak learners into a single strong learner. Individual decision trees are the poor learners in this circumstance. Every tree attempts to reduce the imperfection of the one before it, and all of the trees are series-connected. Boosting algorithms are typically slow to train, but also extremely accurate, because of this successive association. Slower learning models surpass faster learning ones in statistical learning.

10. Conclusions are drawn from the project

The following conclusions are drawn from the project:

1) Elastic net model understands data properly and hence shows very poor performance with data.
2) linear regression and lasso and ridge with or without hyperparameter tuning show very poor performance with the data (all having almost the same R2_score).
3) By using polynomial regression with degree 2, R2_score improve to 0.8 for train data and 0.77 for test data. Model performance is improved as compared to the earlier model.
4) By using DecisionTreeRegressor with GridSearchCV, training R2_score is 0.99 and for the test, data R2_score is 0.89. Model performance is improved but slightly increase in overfitting.
5) By using RandomForest with GridSearchCV, training R2_score is 0.93 and test R2_score is 0.99. The model is slightly underfitting.
6) By using Gradient Boosting without hyperparameter tuning training R2_score 0.89 and test R2_score 0.90, the model is generalized very well.
7) By using Gradient Boosting with GridSearchCV training R2_score is 0.99 and test R2_score is 0.95 model performance is improved with an accuracy of 0.95.