Bike share Demand Prediction

Chandrashekhar Awate

Data science Trainee

Almabetter, Banglore

Problem Statement: **Currently Rental bikes are introduced in many urban cities for the enhancement of mobility comfort. It is important to make the rental bike available and accessible to the public at the right time as it lessens the waiting time. Eventually, providing the city with a stable supply of rental bikes becomes a major concern. The crucial part is the prediction of bike count required at each hour for the stable supply of rental bikes.**

Data Description

The dataset contains weather information (Temperature, Humidity, Windspeed, Visibility, Dewpoint, Solar radiation, Snowfall, Rainfall), the number of bikes rented per hour and date information.

Attribute Information:

- **Date : year-month-day**
- **Rented Bike count - Count of bikes rented at each hour**
- **Hour - Hour of he day**
- **Temperature-Temperature in Celsius**
- **Humidity - %**
- **Windspeed - m/s**
- **Visibility - 10m**
- **Dew point temperature - Celsius**
- **Solar radiation - MJ/m2**
- **Rainfall - mm**
- **Snowfall - cm**
- **Seasons - Winter, Spring, Summer, Autumn**
- **Holiday - Holiday/No holiday**
- **Functional Day - NoFunc(Non Functional Hours), Fun(Functional hours)**

Exploratory Data Analysis

Exploratory data analysis was done to know the insights of the data and which features have there more influence on Rented bike count.

Null value Treatment

Our dataset was not having any null values and duplicates too.

Changes in data formats

We changed some of our data formats for computational convenience. And from date column we extracted day as weekday or weekend day. We also extracted the month as it have influence on our target variable.

Transformation of Data

We have used power transformation as there is heteroskedasticity in data.

Fitting different models

1)Linear Regression

2)polynomial regression

3)Random forest regressor

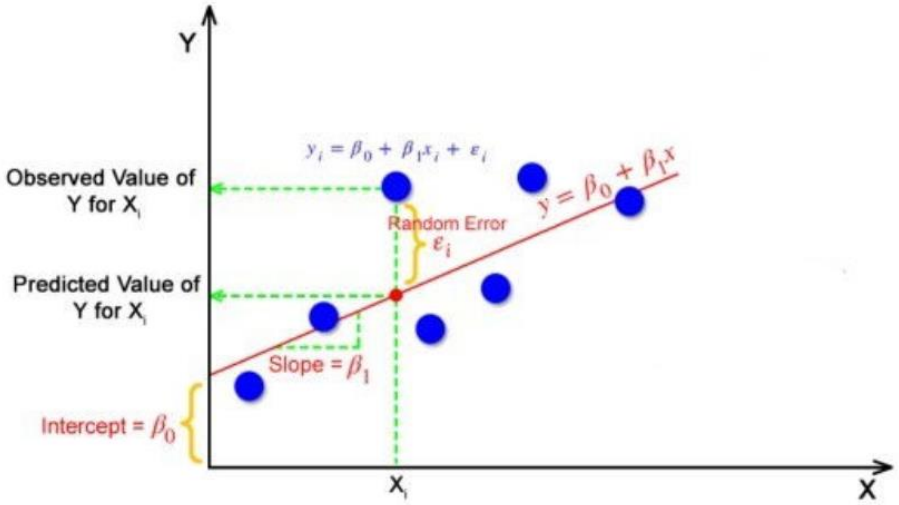
Hyperparameter Tuning and regularization

Hyperparameter tuning and regularization is important for regression problems. It help us find the optimal regression model which is free from errors and can be used efficiently for prediction.

Algorithms

1)Linear Regression

Linear regression is a quiet and the simplest statistical regression method used for predictive analysis in machine learning. Linear regression shows the linear relationship between the independent(predictor) variable i.e. X-axis and the dependent(output) variable i.e. Y-axis, called linear regression. If there is a single input variable X(dependent variable), such linear regression is called *simple linear regression*.



But how the linear regression finds out which is the best fit line?

The goal of the linear regression algorithm is to get the best values for B_0 and B_1 to find the best fit line. The best fit line is a line that has the least error which means the error between predicted values and actual values should be minimum.

2)Polynomial regression

Polynomial regression is a special case of linear regression where we fit a polynomial equation on the data with a curvilinear relationship between the target variable and the independent variables.

In a curvilinear relationship, the value of the target variable changes in a non-uniform manner with respect to the predictor (s).

In Linear Regression, with a single predictor, we have the following equation:

$$Y = \theta_0 + \theta_1 x$$

where,

***Y* is the target,**

***x* is the predictor,**

θ0 is the bias,

and θ1 is the weight in the regression equation

This linear equation can be used to represent a linear relationship. But, in polynomial regression, we have a polynomial equation of degree *n* represented as:

$$Y = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \ldots + \theta_n x^n$$

Here:

θ0 is the bias,

θ1, θ2, ..., θn are the weights in the equation of the polynomial regression,

and *n* is the degree of the polynomial

The number of higher-order terms increases with the increasing value of *n*, and hence the equation becomes more complicated.

3)Random Forest Regressor

Steps involved in Random Forest Algorithm

Step-1 – We first make subsets of our original data. We will do row sampling and feature sampling that means we'll select rows and columns with replacement and create subsets of the training dataset

Step- 2 – We create an individual decision tree for each subset we take

Step-3 – Each decision tree will give an output

Step 4 – Final output is considered based on Majority Voting if it's a classification problem and average if it's a regression problem.

Model performance

1) Mean Absolute Error(MAE)

MAE is a very simple metric which calculates the absolute difference between actual and predicted values.

To better understand, let's take an example you have input data and output data and use Linear Regression, which draws a best-fit line.

Now you have to find the MAE of your model which is basically a mistake made by the model known as an error. Now find the difference between the actual value and predicted value that is an absolute error but we have to find the mean absolute of the complete dataset.

so, sum all the errors and divide them by a total number of observations And this is MAE. And we aim to get a minimum MAE because this is a loss.

2) Mean Squared Error(MSE)

MSE is a most used and very simple metric with a little bit of change in mean absolute error. Mean squared error states that finding the squared difference between actual and predicted value.

So, above we are finding the absolute difference and here we are finding the squared difference.

What actually the MSE represents?

It represents the squared distance between actual and predicted values. we perform squared to avoid the cancellation of negative terms and it is the benefit of MSE.

3) Root Mean Squared Error(RMSE)

As RMSE is clear by the name itself, that it is a simple square root of mean squared error.

4) R Squared (R2)

R2 score is a metric that tells the performance of your model, not the loss in an absolute sense that how many wells did your model perform.

In contrast, MAE and MSE depend on the context as we have seen whereas the R2 score is independent of context.

So, with help of R squared we have a baseline model to compare a model which none of the other metrics provides. The same we have in classification problems which we call a threshold which is fixed at 0.5. So basically R2 squared calculates how must regression line is better than a mean line.

Hence, R2 squared is also known as Coefficient of Determination or sometimes also known as Goodness of fit

Hyper parameter tuning:

Hyperparameters are sets of information that are used to control the way of learning an algorithm. Their definitions impact parameters of the models, seen as a way of learning, change from the new hyperparameters. This set of values affects performance, stability and interpretation of a model. Each algorithm requires a specific hyperparameters grid that can be adjusted according to the business problem. Hyperparameters alter the way a model learns to trigger this training algorithm after parameters to generate outputs.

We used gridsearch cv.

Grid Search CV-Grid Search combines a selection of hyperparameters established by the scientist and runs through all of them to evaluate the model's performance. Its advantage is that it is a simple technique that will go through all the programmed combinations. The biggest disadvantage is that it traverses a specific region of the parameter space and cannot understand which movement or which region of the space is important to optimize the model.

Conclusion:

That's it! We reached the end of our exercise.

Starting with loading the data so far we have done EDA ,finding patterns in data , feature selection and then model building.

In all of these models our accuracy revolves in the range of 77 to 88%.

And there is no such improvement in accuracy score even after hyperparameter tuning.

So the accuracy of our best model is polynomial regression as it have least mean absolute error and 88% R-squared value.