Seol bike sharing demand prediction

Shiyas Ali T M

Data science trainee

AlmaBetter, Bangalore

Abstract: Prediction of demand for number of bikes at a given hour of a particular day in a bike sharing system is attempted. Feature selection is done by analyzing correlation coefficient between variables. Various regression techniques such as linear regression, regularized linear regression, Random Forest and Gradient Boosting are used. Methods of grid search and random search along with cross validation is used for hyperparameter tuning. Models are evaluated against a number of evaluation techniques like Rsquared, mean square error, root mean square error, mean absolute error and mean absolute percentage error. Performance of different models are compared.

Keywords: Linear regression, Lasso, Ridge, Random Forest, Gradient Boosting, hyperparameter tuning, grid search, random search.

1. Problem statement

Bike renting has become so popular among people, especially in big cities, in recent times. Demand for bikes in such systems fluctuates depending upon various factors. Knowing in advance, the number of bikes needed at a particular time, is crucial for the seamless running of the business. This can help not only to prepared with the right number of bikes at the right time, but also to enforce dynamic pricing depending on the demand.

2. Introduction

We are given a data set of a particular bike sharing company which contain bike rental data from ‘2017-January-1' to ‘2018-December-11'. The data set contains following features.

- Date: Date of booking.
- Rented Bike Count: Number of bikes booked.
- Hour: Hour of booking.
- Temperature(°C): Temperature at the time of booking.
- Humidity(%): Humidity at the time of booking.
- Wind speed (m/s): Wind speed at the time of booking.
- Visibility (10m): Visibility at the time of booking.
- Dew point temperature(°C): Dew point temperature at the time of booking.

- Solar Radiation (MJ/m2): Solar radiation at the time of booking.
- Rainfall(mm): Amount of rainfall at the time of booking.
- Snowfall (cm): Amount of snowfall at the time of booking.
- Seasons: Season of booking.
- Holiday: Whether or not a Holiday.
- Functioning Day: Whether or not a functioning day.

Our target feature is 'Rented Bike Count', which gives the number of bikes rented at each hour of a day. This value changes according to a number of external features. We are trying to build a machine learning model to predict the required count of bikes, by training it using past data. We will use a number of regression techniques to achieve this goal.

3. Approach

3.1. Exploratory Data Analysis

Exploratory data analysis is done to identify general properties of different features. The distribution of different variables can be understood by performing EDA. Patterns in the data, if there is any, can be seen from EDA. Outliers in the data set can be identified using EDA. This helps us in obtaining an overall idea about the data in hand.

3.2. Null values and outlier treatment.

Null values are always undesired in a data set. It can cause various problems to our work here. They are always removed from the data set, either by dropping the observation altogether or by replacing them with appropriate values. Our data set doesn't contain any null values. Outliers can also be as bad as null values sometimes. But dropping or replacing them blindly might adversely affect our purpose. Some features of our data set contain large number of outliers. Some of them are transformed into binary variables. Target variable is used both with and without outliers in some models.

3.3. Feature selection

Feature selection plays a crucial role in any model building. They can have a strong impact on the overall quality of the model. Finding the relevant features is a major step towards our goal. Features which have the potential to impact target variable must be chosen. Increasing the number of features will make the model more complex and can lead to overfitting. So, hitting the right balance is really important. We employ the technique of correlation coefficient analysis in order to choose the right features. Features with good correlation with the target variable are chosen. Choosing features with high correlation between themselves can introduce redundant information.

3.4. Encoding categorical variables

Machine learning model can only take numerical values as its inputs. Incomprehensible categorical variables should be converted into numerical variables before giving it as input to the model. Dichotomous categorical variables in our data set are converted into binary variables. Hour variable is transformed into a cyclic variable, in order to incorporate the cyclic nature of time. One hot encoding is used to encode multi-level categorical variable ‘Season’.

3.5. Splitting into train and test values

The whole data set is divided into train and test set. Training set is used exclusively for the purpose of training the model. Test set is set aside for validating the model after training. This helps in observing how the model performs on unseen data. In our case, 80 percent of the data is used for training and the remaining 20 percent for testing.

3.6. Standardizing features

The range of values varies considerably from feature to feature. In order to fully understand how target varies with different features, it is important to bring all the features into same scale. This also helps in achieving faster convergence in the model. We use the method of normalizing or min-max scaling. The values are brought into the range of 0 to 1.

3.7. Training the model

Once the data set is cleaned and pre-processed, it is fed to a machine learning algorithm, to find patterns in the data set and learn to predict the target from feature values. There are various techniques for doing this in the machine learning domain. We are employing supervised regression techniques for our purpose.

3.8. Hyperparameter tuning

Hyperparameters are properties or values associated with a model. We are free to try different values for these parameters. Finding the values that give best model performance is called hyperparameter tuning. This is often performed as a trial and error with different combination of values.

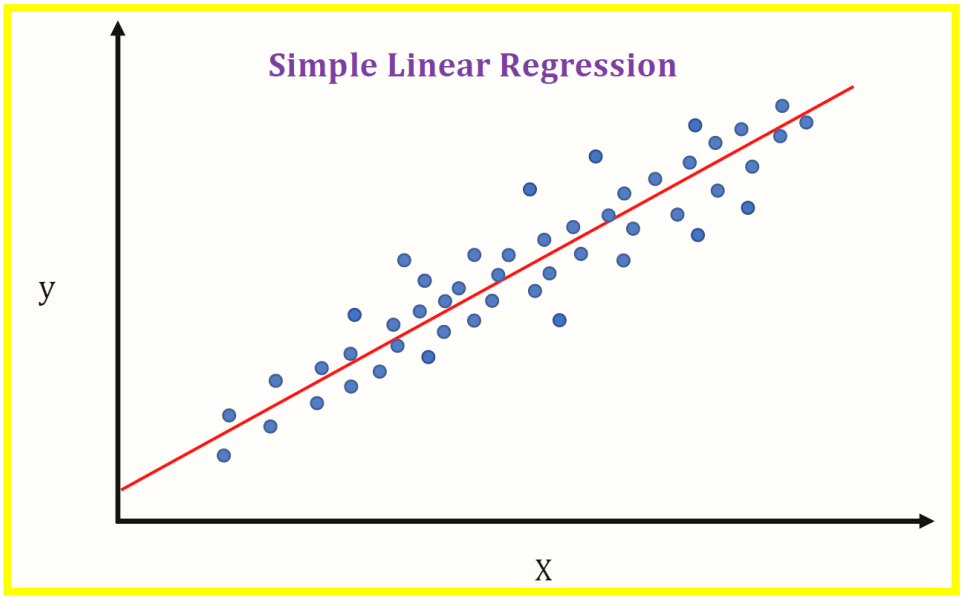
3.9. Model validation

Model validation is the process of checking how good is the model. The model is evaluated against some evaluation metrics. This gives us a measure of how well the model is performing on both the training and test data.

4. Algorithms

4.1. Linear regression

Linear regression fits a straight line or surface that minimizes the discrepancies between predicted and actual output values.



$\hat{Y} = b_0 + b_1 X_1 + b_2 X_2 + \ldots + b_p X_p$, The output is predicted based on this equation. b0 through bp are the parameters of the linear regression. Best set of parameters is obtained by minimizing a loss function. The loss function used is mean squared error.

4.2. Lasso regression

Regularization is a concept used in machine learning to avoid overfitting the data. One of the regularization techniques used in linear regression is Lasso. The loss function is slightly modified to penalize the parameters from getting large. Lasso regression is also called L1 regularization.

$$\sum_{i=1}^{n}(y_i - \sum_{j} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

The second term in the summation is the modification made to the loss function. Lambda is the regularization parameter.

4.3. Ridge regression

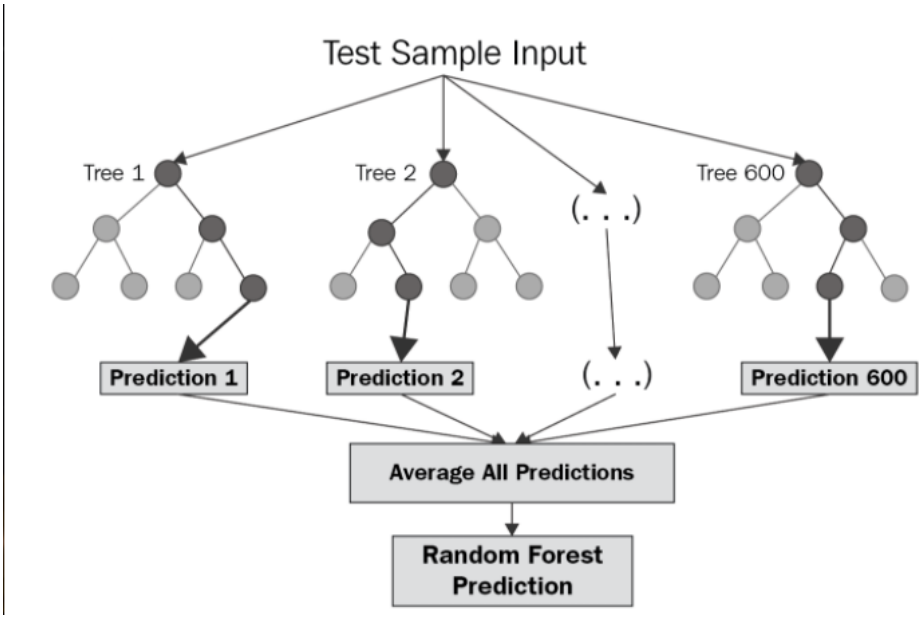
Ridge regression is another regularization technique used in linear model. It is also called L2 regularization technique. Unlike lasso regression, it doesn't shrink the coefficients all the way down to zero. The loss function for Ridge regression is

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda \sum_{j=1}^{p}\beta_j^2$$

Lambda here as well is the regularization parameter.

4.4. Random Forest regressor

Random Forest is an ensemble technique of decision trees, which uses the concept of bagging. It also incorporates bootstrapping. A number of trees are fit on a subset of the data using a selected number of features at a time. Each tree of the forest gives a prediction, and the average of each prediction is taken as the final prediction.



4.5. Gradient Boosting.

Gradient Boost is another ensemble technique of decision trees. This uses the concept of boosting. A number of weak learners contribute together to form a strong learner. Gradient Boosting involves 3 elements.

1. A loss function to optimize
2. A weak learner to make prediction
3. An additive model to add to weak learners to minimize loss function.

The loss function may be defined depending upon the problem being solved. We use squared error as loss function.

5. Evaluation metrics

5.1. R-squared

It gives the proportion of variance of the target variable explained by the independent variables. The range of this measure is 0 to 1.

R-squared = 1-(RSS/TSS)

Where, RSS is the sum of squares of residual and TSS is the total sum of squares

5.2. Mean squared error

Mean squared error (MSE) is the average of the squared errors in a model.

5.3. Root mean squared error

Root mean squared error (RMSE) is the square root of MSE

5.4. Mean absolute error

Mean absolute error (MAE) is the mean of the absolute value of individual errors

5.5. Mean absolute percentage error

The mean absolute percentage error (MAPE) is the mean or average of the absolute percentage errors of forecasts.

6.Hyperparameter tuning

6.1. Grid search

Grid search is a technique used for hyperparameter tuning. A set of values for each hyperparameter is taken. Every combination of these values are tested using cross-validation. The best combination is chosen to build the model.

6.2. Random search

Similar to the grid search, a set of values for each hyperparameter is taken. But instead of trying every combination, a defined number of combinations are randomly tested using cross-validation. The best combination out these is used for model building.

7. Conclusion

So far, we have discussed various techniques and methods which are used towards achieving our goal of predicting bike count. We started with EDA, then moved on to data cleaning through null values and outlier treatment. We did model training after feature selection and data pre-processing.

Linear models invariably poor on both training and testing data. The highest Rsquared value that could be achieved was 56%. This could be due to the lack of linear relationship between dependent and independent variables. Ensemble models performed well on both training and testing data. While training data gave a Rsquared value of 97%, testing data shows an Rsquared value of 87%. There was some overfitting in this case.