**Problem Statement**

* This assignment is a programming assignment wherein you have to build a multiple linear regression model for the prediction of demand for shared bikes. You will need to submit a Jupyter notebook for the same.
* A bike-sharing system is a service in which bikes are made available for shared use to individuals on a short term basis for a price or free. Many bike share systems allow people to borrow a bike from a "dock" which is usually computer-controlled wherein the user enters the payment information, and the system unlocks it. This bike can then be returned to another dock belonging to the same system.
* A US bike-sharing provider BoomBikes has recently suffered considerable dips in their revenues due to the ongoing Corona pandemic. The company is finding it very difficult to sustain in the current market scenario. So, it has decided to come up with a mindful business plan to be able to accelerate its revenue as soon as the ongoing lockdown comes to an end, and the economy restores to a healthy state.
* In such an attempt, BoomBikes aspires to understand the demand for shared bikes among the people after this ongoing quarantine situation ends across the nation due to Covid-19. They have planned this to prepare themselves to cater to the people's needs once the situation gets better all around and stand out from other service providers and make huge profits.
* They have contracted a consulting company to understand the factors on which the demand for these shared bikes depends. Specifically, they want to understand the factors affecting the demand for these shared bikes in the American market. The company wants to know:
  + Which variables are significant in predicting the demand for shared bikes.
  + How well those variables describe the bike demands
  + Based on various meteorological surveys and people's styles, the service provider firm has gathered a large dataset on daily bike demands across the American market based on some factors.

**Business Goal**

You are required to model the demand for shared bikes with the available independent variables. It will be used by the management to understand how exactly the demands vary with different features. They can accordingly manipulate the business strategy to meet the demand levels and meet the customer's expectations. Further, the model will be a good way for management to understand the demand dynamics of a new market.

**Data Preparation**

* You can observe in the dataset that some of the variables like 'weathersit' and 'season' have values as 1, 2, 3, 4 which have specific labels associated with them (as can be seen in the data dictionary). These numeric values associated with the labels may indicate that there is some order to them - which is actually not the case (Check the data dictionary and think why). So, it is advisable to convert such feature values into categorical string values before proceeding with model building. Please refer the data dictionary to get a better understanding of all the independent variables.
* You might notice the column 'yr' with two values 0 and 1 indicating the years 2018 and 2019 respectively. At the first instinct, you might think it is a good idea to drop this column as it only has two values so it might not be a value-add to the model. But in reality, since these bike-sharing systems are slowly gaining popularity, the demand for these bikes is increasing every year proving that the column 'yr' might be a good variable for prediction. So think twice before dropping it.

**Model Building**

* In the dataset provided, you will notice that there are three columns named 'casual', 'registered', and 'cnt'. The variable 'casual' indicates the number casual users who have made a rental. The variable 'registered' on the other hand shows the total number of registered users who have made a booking on a given day. Finally, the 'cnt' variable indicates the total number of bike rentals, including both casual and registered. The model should be built taking this 'cnt' as the target variable.

**Model Evaluation**

* When you're done with model building and residual analysis and have made predictions on the test set, just make sure you use the following two lines of code to calculate the R-squared score on the test set.

from sklearn.metrics import r2\_score

r2\_score(y\_test, y\_pred)

where y\_test is the test data set for the target variable, and y\_pred is the variable containing the predicted values of the target variable on the test set.

Please don't forget to perform this step as the R-squared score on the test set holds some marks. The variable names inside the 'r2\_score' function can be different based on the variable names you have chosen.

**Submissions Expected**

* Python Notebook: One Python notebook with the whole linear model, predictions, and evaluation.
* Subjective Questions PDF: Apart from the Python notebook, you also need to answer some subjective questions related to linear regression which can be downloaded from the file below. Answer these questions and submit it as a PDF.
* Note: There are some questions in the subjective questions doc that you might not be familiar with. So you're expected to research these questions and give an appropriate answer in order to expand your learnings of this topic.
* You need to submit a GitHub repository link that contains the above files. You need to write a README.md for the GitHub repository. Make sure your GitHub link is public.

**Rubrics**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Meets expectations** | **Does not meet expectations** |
| **Data understanding, preparation and EDA (25%)** | Data quality checks, if any, are performed, and all data quality issues are addressed in the right way. Explanations for data quality issues are clearly mentioned in comments.    Categorical variables are handled appropriately.        Dummy variables are created properly wherever applicable.        New metrics are derived if applicable and are used for analysis and modelling.        The data is converted to a clean format suitable for analysis. | Data quality checks, if any, are not done, data quality issues are not addressed correctly to an appropriate level.    Categorical variables are not handled appropriately where required.    Dummy variables are not created properly.      New metrics are not derived or are not used for analysis.    The data is not converted to a clean format which is suitable for analysis or is not cleaned using commands. |
| **Model building and evaluation (50%)** | Model parameters are tuned using correct principles and the approach is explained clearly. Both the technical and business aspects are considered while building the model.      Correct variable selection techniques are used. A reasonable number of different models are attempted and the best one is chosen based on key performance metrics.    Residual analysis is performed after model building and the assumptions are validated.    Model evaluation is done using the correct principles and appropriate evaluation metrics are chosen.      The results are on par with the best possible model on the dataset.    The model is interpreted and explained correctly. The commented code includes a brief explanation of the important variables and the model in simple terms. | Parameters are not tuned enough or tuned incorrectly. Relevant business aspects are not considered while model building.    Variable selection techniques are used incorrectly / not conducted. A variety of models are not considered or a sub-optimal one is finalised.    Residual analysis is not performed after model building and the assumptions are not validated.    The evaluation process deviates from correct model selection principles, inappropriate metrics are evaluated or are incorrectly evaluated.      The results are not on par with the best possible model on the dataset.    The model is not interpreted and explained correctly. |
| **Coding Guidelines (5%)** | Appropriate comments are written wherever applicable.        If new variables are created, the names are descriptive and unambiguous.      The code is written concisely wherever possible.        Overall, code readability is good with appropriate indentations.    The GitHub repository link contains a python file, PDF file and a README.md file. README.md file should describe the project briefly.    If any assumptions are made, they are stated clearly. | Comments are not written rendering the code difficult to understand.      Variables are poorly or ambiguously named.      The code is more complex than what is required by the problem.    Code readability is poor because of poor indentation / other reasons.    The GitHub repository should not be private.    The GitHub repository link which does not contain python file, PDF file and a README.md file.    README.md file which has no proper brief description of the project.  Contains unnecessary details or lacks the important ones.    Assumptions made, if any, are not stated clearly. |
| **Subjective Questions (20%)** | The answers are correct, concise, and to the point.    Examples are provided wherever necessary. | The answers are incorrect or unnecessarily long.    Appropriate examples are not provided in places where required. |