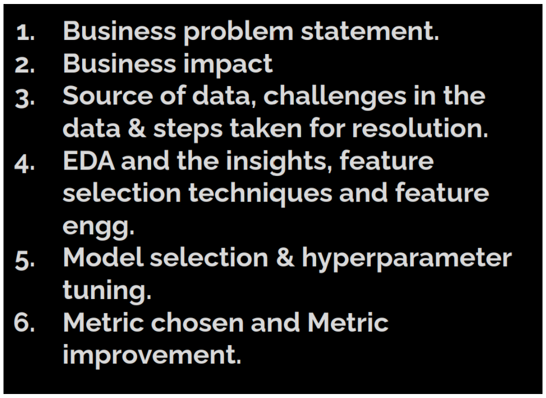
CAPSTONE - 2 Seoul Bike Sharing Demand Prediction



1. Business problem statement.
2. Business impact
3. Source of data, challenges in the data & steps taken for resolution.
4. EDA and the insights, feature selection techniques and feature engg.
5. Model selection & hyperparameter tuning.
6. Metric chosen and Metric improvement.

so the first project is of regression in which we have to predict the number of bikes that can be rented per hour by the company

💡 **Problem Description-**

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 reduces waiting time. Eventually, providing the city with a stable supply of rental bikes becomes a major concern. The crucial part is the prediction of the bike count required at each hour for the stable supply of rental bikes.

Our objective is **to predict the number of bikes that can be rented per hour by the company.**

coulumns = 14
rows =

- Date : year-month-day
- Rented Bike count - Count of bikes rented at each hour
- Hour - Hour of the 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), Func(Functional hours)

💡 Source of data, challenges in the data, and steps are taken for resolution

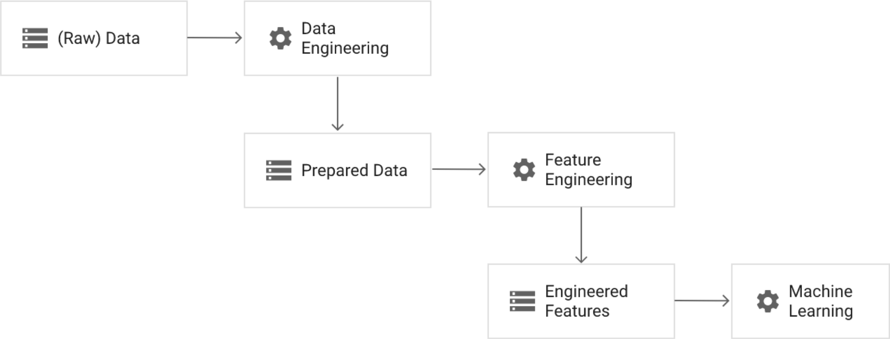
- source of data - dataset has been provided by alma-better.
- challenges we faced in the dataset is that
  - the data is provided in the Unicode form and we have to decode it first for that we used the `unicode_escape` encoding technique to read the data from CSV.
  - Unicode is an international character encoding standard that provides a unique number for every character across languages and scripts, making almost all characters accessible across platforms, programs, and devices.
  - After that, we extracted Day, month, and year from the date column to gain more clarity regarding the dataset.

💡 Steps followed

▼ Step 1 - **Collecting Data**

▼ Step 2 - **Preparing the Data / Data Preprocessing**

Data Preprocessing involves both data engineering and feature engineering. Data engineering is the process of converting *raw data* into *prepared data*. Feature engineering then tunes the prepared data to create the features that are expected by the ML model



- **Data cleansing:** removing or correcting records that have corrupted or invalid values from raw data, Removing duplicate rows and correcting data type conversion.
- Before performing exploratory data analysis (EDA) on the dataset, we extracted the day, month, and year from the date column so that we could use these features to gain useful insights

▼ In the exploratory data analysis (EDA) phase of our analysis, we divided the dataset into two parts: numerical features and categorical features. For the numerical features, we further divided them into discrete and continuous features.

To visualize the distribution and skewness of the continuous features, we used histograms. A histogram is a graphical representation of the distribution of a continuous variable. By looking at the histogram, we can get an idea of the shape of the distribution and whether it is skewed to the left or right.

We found that our dependent variable, rented bike count, was right skewed, meaning that it had a longer tail on the right side of the distribution. To make the distribution more normal, we applied a log-square transformation to the data. A log-square transformation is a mathematical function that maps the data onto a new scale, typically to reduce skewness and improve the normality of the distribution.

- For discrete and categorical features, we used bar plots to visualize their distribution and understand their relationships with other variables.
- Finally, we used a heat map to plot the correlation between the dependent variable and a set of independent variables.
- Overall, our exploratory data analysis helped us gain a better understanding of the dataset and identify potential issues that needed to be addressed before moving on to modeling.
- we checked for multicollinearity between the independent features by calculating the variance inflation factor (VIF) for each feature, We removed features that had a VIF score greater than 10, as they were considered highly correlated and potentially irrelevant to our dataset. By removing these features, we were able to improve the stability and reliability of our model.
- **Feature tuning:** improving the quality of a feature for ML, which includes scaling and normalizing numeric values, and adjusting values that have skewed distributions.
- **Feature transformation:** we converted our categorical features into a numeric representation using one-hot encoding. By converting our categorical features to a numeric representation, we were able to use them in our machine learning models and gain insights from them. One-hot encoding also helped us avoid issues such as the ordinality of the data, where the categories may be treated as having an inherent order.

▼ why u used standardscalar over minmax scalar ?

- for scaling we used standardscalar,it scales the data with mean=0 and standard deviation=1, we used standardscalar here because the data is normally distributed, and standardscalar makes the algorithm less sensitive to outliers.
- on the other hand we used minmax scalar for sparse dataset and it transforms each value in the column within the range of [0,1] . generally used in image processing, neural networks. because their we expect values between 0 to 1.

▼ why u used one hot encoding over label encoding?

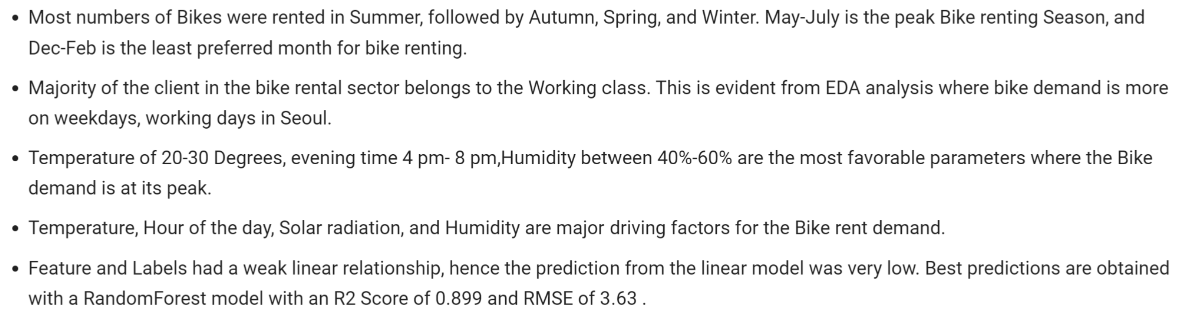
- One hot encoding and label encoding are two methods used to represent categorical variables in a dataset as numerical values.
- One hot encoding is a method that converts each category in a categorical variable into a separate binary column, with a value of 1 indicating that the sample belongs to that category and a value of 0 indicating that it does not. This is often used when the categories are not ordinal, meaning that there is no inherent order or ranking between them.
- Label encoding is a method that assigns a unique integer value to each category in a categorical variable. This is often used when the categories are ordinal, meaning that there is an inherent order or ranking between them.
- In general, one hot encoding is preferred over label encoding when working with categorical variables because it is more efficient and less prone to error. This is because one hot encoding creates separate columns for each category, which allows the model to learn the relationship between the categories and the target variable more

effectively. In contrast, label encoding assigns integer values to the categories, which can lead to errors if the model interprets the integer values as having a meaningful order or ranking.

- After completing the data preprocessing and feature transformation steps, we split our dataset into a training set and a validation set. The training set was used to train our machine learning models, while the validation set was used to evaluate the performance of the trained models.

▼ Step 3 - **Training the Model**

- During the training phase, we trained several different machine learning models on the dataset, including linear, lasso, ridge, decision tree, random forest, and gradient boosting algorithms. After comparing their performance, we selected a random forest as the final model for the task.
- To evaluate the performance of the models, we used two evaluation metrics: the R2 score and root mean squared error (RMSE). The R2 score is a measure of the amount of variance in the data that is explained by the model, while RMSE is a measure of the average difference between the predicted values and the true values.
- After evaluating the models using these metrics, we found that the linear, lasso, and ridge models did not perform well, while the decision tree model performed well but had high variance and low bias, indicating that it was overfitted. To reduce the variance and improve the performance of the model, we used the random forest algorithm.
- However, even after applying random forest, there was still some overfitting in the model. To further reduce overfitting and improve the performance of the model, we performed hyperparameter tuning using grid search cross-validation (GridSearchCV). This helped us find the optimal combination of hyperparameters for the model and improve the R2 score by approximately 1%.



- Most numbers of Bikes were rented in Summer, followed by Autumn, Spring, and Winter. May-July is the peak Bike renting Season, and Dec-Feb is the least preferred month for bike renting.
- Majority of the client in the bike rental sector belongs to the Working class. This is evident from EDA analysis where bike demand is more on weekdays, working days in Seoul.
- Temperature of 20-30 Degrees, evening time 4 pm- 8 pm,Humidity between 40%-60% are the most favorable parameters where the Bike demand is at its peak.
- Temperature, Hour of the day, Solar radiation, and Humidity are major driving factors for the Bike rent demand.
- Feature and Labels had a weak linear relationship, hence the prediction from the linear model was very low. Best predictions are obtained with a RandomForest model with an R2 Score of 0.899 and RMSE of 3.63 .