Part I

Business Case :

Bike sharing systems are a means of renting bicycles where the process of obtaining membership, rental, and bike return is automated via a network of kiosk locations throughout a city.

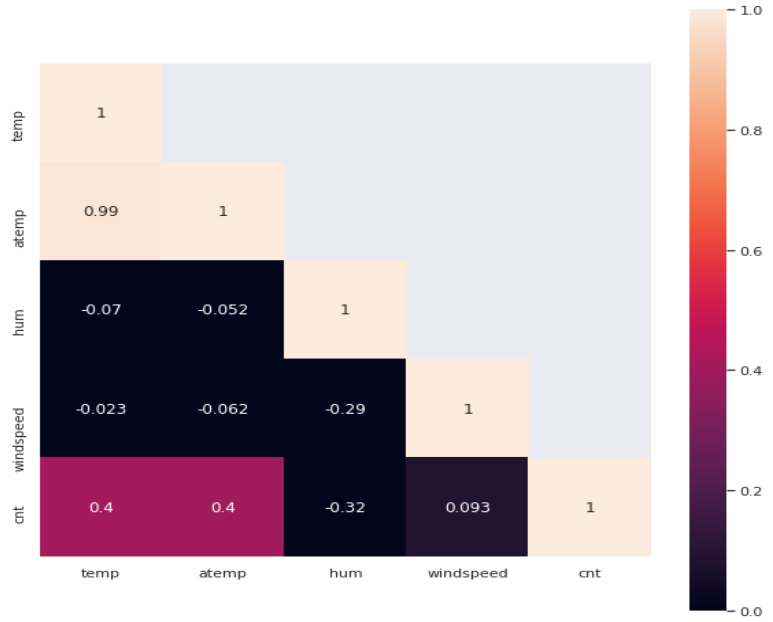
Prediction model for hourly utilization of total count of bike rentals. This estimates the expected revenue of bike sharing systems. It also gives a idea of number of bikes to be distributed on to each stations. Allowing maximum usage and good revenue. Here, Investigation of prediction of the hourly bike count based on features. For example, The total count variable is directly proportional to the casual and registered variable. Identifying the appropriate features contribute to the better prediction of total count and devising a predictive model.

DataSource:

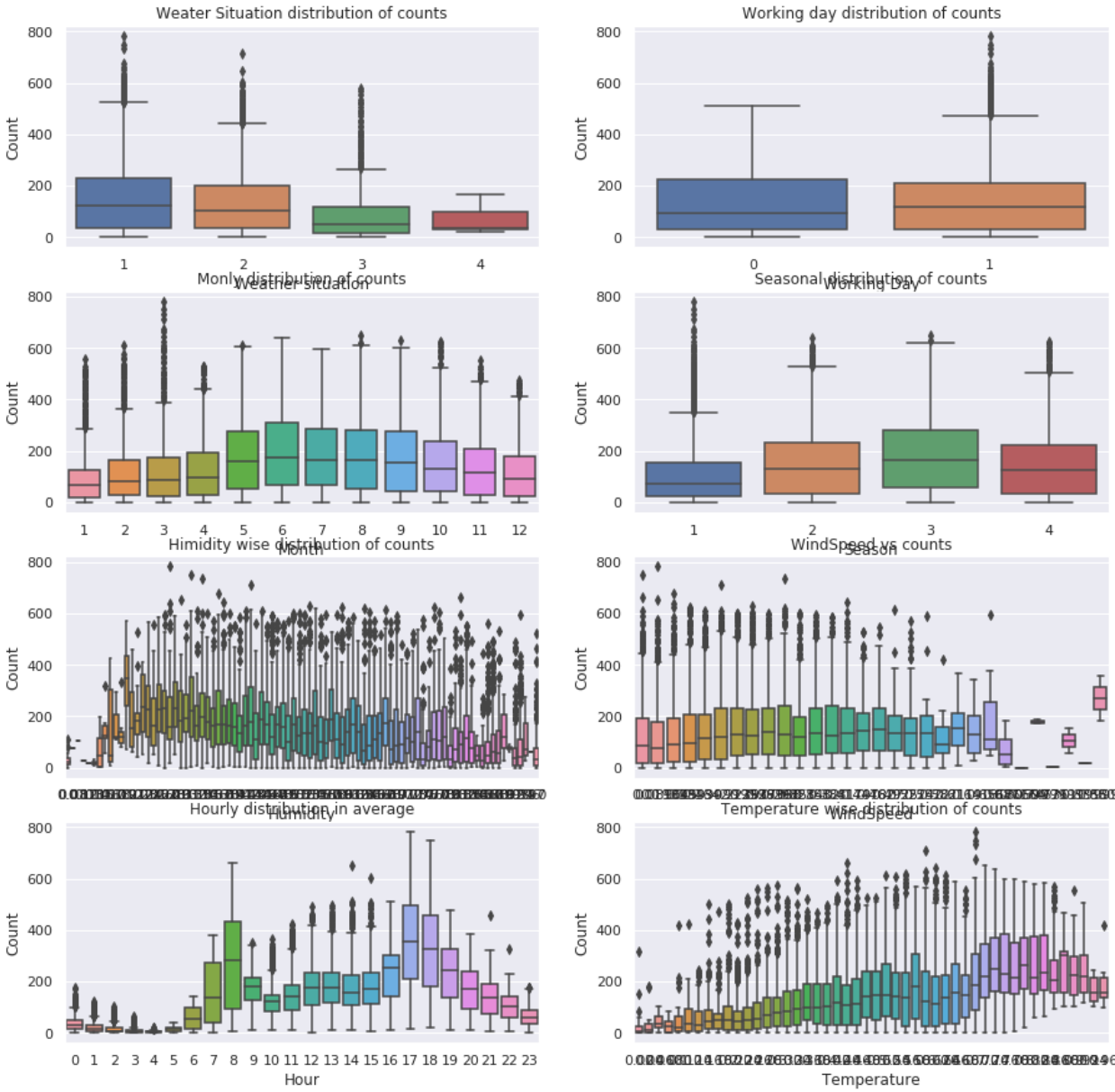
The programming task given was modeling the Bike sharing dataset from the UCI repository https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset. The description and attribute information are mentioned the link.

Data Exploration:

The Jupyter notebook file JDA_BikesharingAnalysistask.ipynb has exploratory data Analysis, Missing Value Analysis, Outlier Analysis, Correlation Analysis presented in detail. The Missing value analysis identifies the data does not have any null values. From the box plot some extreme values were presented. This led to outlier analysis and remove of outliers using median and Inter-Quartile Rage(IQR), since it did not fit the normal distribution. The sample data were reduce removing the outliers. ‘Casual’ and ‘registered’ variables are removed as it constitutes direct proportionality with ‘count’ variable, thus leading to data leakage.



Correlation matrix(numerical_features)



BoxPlots revealing outliers

***Perception**:*

1) The working day plots shows that more bicycles are rent during normal working days than on weekends/holidays. On holidays the bikes are rented mostly during mid day (10 Am and 4 Pm) . The hourly boxplots show maximum of bike rental at 8 am and one at 5 pm which indicates that most users of the renters use the bikes to get to work or school(registered users). Most of the outlier points are mainly contributed by Working Day.

2) Temperature plots show: higher temperatures lead to an increasing number of bike rents and lower temperatures not only decrease the average number of rents but also shows more outliers in the data. Weather situation plot shows renting more bikes in clear weather. Seasonal plots shows the fall has higher bike rentals

3) The count variable has got little dependency on hour, temperature and humidity variables

4) The variables temp and atemp are strongly co-ordinated. Hence to reduce dimensionality "atemp" is not taken. Month and season has high-correlation effect as well.

Model Selection:

The characteristics of given problem

1) Target variable: 'cnt' which a continuous variable thus **Regression** method
2) Small to medium dataset - 100 K samples
3) Correlation matrix and scatter plots identified the importance of few features and to discard of few features for analysis
4) Metrics taken are MAE(Mean Absolute Error), MSE(Mean Squared Error), RMSE(Root Mean Squared error), RMSLE(Root Mean Squared Log Error), R^2 and Adjusted R^2.

Feature Selection:

Hour has the largest effect on the prediction, followed by temperature, weather situation, Humidity and Working day. Holiday feature was of the least importance. Working day parameter when not taken does not give any change in results.

Selected Business Model : Random Forest Regressor

Reason :

Ensemble models combine the weak learners and improve stability and predictability of model. It improves the performance by reducing the variance, avoids over-fitting and averages out the bias.
Choosing of random forest was based on metrics calculated based on RMSLE, RMSE, Adjusted r^2 value. The random forest regressor gives an improvement compared to other scores. Random forest gives the importance to features as of model fitting. It is especially useful in understanding how features contribute in model building and also in feature selection. It is shown from the figure below that hour attribute plays a significant role in model prediction.

Assuming booting algorithms give good results when compared to bagging. GradientBoostingRegressor, ExtraTreesRegressor (Un-interpretable output each time thus ignored) were also taken for analysis as it gave similar results to random forest. Hence working on both algorithms to identify the parameters to model the prediction. After identifying the some of the best parameters for the both the algorithms. The model was validated to identify with RMSLE 0.49 (Gradient Boosting) and 0.476 (Random Forest).

The selected model was executed with GridSeachCV to identify the best parameters, with the K-fold cross validation. It was split in 10 folds to identify the mean absolute Error(MAE) value and RMSLE value to be improved with the mean value of 0.40. Please find the table attached table below.

Model	Root Mean Squared Error	Mean Absolute Error	RMSLE	R² score
RandomForestRegressor	63.951	42.169	0.403	0.870
RandomForestRegressor	64.302	42.228	0.406	0.877
RandomForestRegressor	66.927	42.398	0.399	0.864
RandomForestRegressor	71.536	45.453	0.404	0.846
RandomForestRegressor	66.424	43.347	0.387	0.868
RandomForestRegressor	67.444	44.135	0.402	0.856
RandomForestRegressor	66.128	43.277	0.413	0.860
RandomForestRegressor	65.927	43.435	0.382	0.874
RandomForestRegressor	65.182	42.985	0.406	0.875
RandomForestRegressor	64.042	42.121	0.412	0.874
RandomForestRegressor	66.186	43.155	0.401	0.866

Part 2

For large scale datasets used sklearn python implementation of random forests will be slow, it could lead to memory problems.

The solution for large scale dataset is to use CART model for segmentation of dataset along with Random Forest and using framework like Apache spark ML for cluster computing

CART model is highly suitable for large scale datasets and it generates trees that can be traversed to predict the outcome. Gradient boosting/Random Forest algorithms improve the accuracy of the CART model. Apache Spark ML is a framework optimized for distributed cluster computing. The Apache spark ML has transformer and estimator which allows the data-frames to be inputed and resulting in training the model in real time data too. Data can be accessed from Cassandra, PostgreSQL etc. In addition, it can run on Hadoop, Kubernetes etc.

The drawbacks of the mentioned technologies depends on Pre-processing of data, Computation power and sometimes decision trees can be unstable, this aggregates over the results of many trees, does not produce a single, easily interpretable tree diagram.

I have worked with Apache spark ML and CART for classification of words for self learning. I do not have any hands-on experience. But I have obtained theoretical knowledge through courses.

Running the code:

Create a python environment using conda and the JDA_Analysistask_env.yaml file using the following command from the source folder:

conda env create –f JDA_Analysistask_env.yaml

To activate the environment:

conda activate JDA_Analysistask_env

All analysis steps are provided in jupyter notebook. Run it with the following command from the source folder:

jupyter notebook

The results of analysis and prediction model execution is in JDA_BikesharingAnalysistask.ipynb.