**Section B**

**Jupyter Notebook Section**

Step **0** create at least 6-7 exploratory questions.

Exploratory questions: (after loading the dataset) (must be in relation to the target)

(for bar create a new data frame of the data to be plotted) (specify the plotted data frame in the code)

1. Does remaining lease affect resale price (Scatter)

* *pyspark.pandas.DataFrame.plot.scatter(x=’remaining\_lease’, y=’resale\_price’)*

1. Relationship between floor area and resale price (Scatter)
2. Relationship between town and average resale price (bar) (groupby town) (create avg)

* *pyspark.pandas.DataFrame.plot.bar(x= ‘resale\_price’, y=‘town’)*

1. Relationship between flat type and average resale price (bar) (groupby flat type) (create avg)

* *pyspark.pandas.DataFrame.plot.barh(x= ‘flat\_type’, y=‘resale\_price’)*

1. Average resale prices of flat models (bar) (groupby flat model) (create avg)

* *pyspark.pandas.DataFrame.plot.barh(x= ‘flat\_model’ , y=‘avg\_resale\_price’)*

1. Average resale price by year (line) (groupby year) (create new avg column)

* *pyspark.pandas.DataFrame.plot.line(x= ‘year’ , y=‘avg\_resale\_price’)*

1. Relationship of block and resale price (scatter) (cfm no relationship) (just to justify dropping) (no point)

* *pyspark.pandas.DataFrame.plot.scatter(x=’block’ , y=’resale price’)*

1. Distribution of numerical values (highlight the target variable) (floor area, remaining lease, resale price) \*specify the column in the dataframe (convert to pandas)

* *pyspark.pandas.DataFrame.hist(bin = 20)*

Step 2 create tables based on the question.

Step 3 if tables cannot be made, use (scatter/histogram).

Step 4 for the visuals with table it will be line or bar chart or pie.

For most of the numerical values do histogram and scatter (floor\_area\_sqm, remaining\_lease, resale\_price)

AT LEAST 3-4 are 2 dimensions (must be relevant to target)

Irrelevant columns

**Report Section (REFER TO DW ASSIGNMENTS AND DEA)**

Intro (Explain MLlib) (Explain Spark) (introduce assignment and problem statement)

**1**

This report aims to cover and explain my findings and the processes involved in building a machine learning model.

Before modelling, we first have to formulate a value-based prediction problem statement to help us transform the data and build a model to evaluate the dataset. To form the problem statement, pyspark is employed to load and convert the csv file into a dataframe “df\_pyspark” for exploration.

For exploration, functions such as .printSchema(), .show(), and .describe() was used to give an overview of the features present, the data types for each feature, and a brief statistical summary of the numerical features in the dataset. To display the dimensions of the dataset, a combination of .count and len() functions was used. Lastly, the unique values in the categorical features were identified using .distinct() and .collect().

Based on the initial exploration of the dataset, the problem statement formed is “Construct a model that predicts the resale prices of any given HDB resale transaction based on its characteristics”, where characteristics can refer to flat model, storey range, flat type, etc.

This model will be helpful to give potential buyers or sellers a rough estimate on the resale price of their HDB flat. (198)

**2**

Exploratory Data Analysis (EDA) is conducted to better understand the trends, relationships, and potential errors within the dataset. Exploratory Data Analysis is done using pyspark tables using aggregate functions as well as functions such as .groupby() and .select().

Analysis of the features will be done in relation to the target, “resale\_price”.

When exploring the features, a common trend observed is, flats are priced higher when remaining lease, floor area and storey range is higher. These are features that will be useful for model prediction. Another observation is that there is a downward trend in average resale price from 2017 to 2019.

It can also be observed that the top 3 towns with the highest resale value is Bukit Timah, Bishan, and the Central Area. This may indicate that these are high value locations.

No connection was observed between the block number and resale price. Therefore, we can safely drop this feature later.

There is an anomaly in remaining lease as the values are in the hundreds despite the max lease being 99 years. After investigating, it was found that remaining lease was stored in months instead of years.

Missing values were also found in “floor\_area\_sqm”. There was a total of 50 missing rows in the dataset. Considering we have 64247 rows; it is preferable to drop the rows as we would still have sufficient data for prediction. Dropping is preferred as imputation may introduce falsified data which can affect the accuracy of and bias of the final model. (248)

**3**

Before transformation, the data was further cleansed by removing the unnecessary features identified during that data exploration process. These features are,

* “block”, which has little correlation with the target.
* “lease\_commence\_date”, which has no purpose as “remaining\_lease” is derived it.
* “street\_name” and “month” adds insignificant value in relation to the target.

For the categorical features, one-hot encoding was used to encode the categorical features into a distinct value readable by the model. To perform One-Hot encoding with pyspark, the features were encoded using string indexing with “StringIndexer”, transformed into a binary vector with One-hot Encoder. The Pipeline function was used to perform both steps on the categorical features, "town","flat\_type","flat\_model","storey\_range". Finally, the “string\_encoded” and original columns are dropped.

For numerical transformation, Vector Assembler and Feature Scaling was utilized. Vector Assembler was used to combine the columns, apart from “resale\_price”, into a single vector column. This is done as Spark MLlib models only accept vectorized columns to maximize efficiency and scaling.

Then scaling was applied onto the vectorized columns. Feature scaling is done to reduce the varying magnitude between features, to ensure that the weights of the model is not skewed. The following scalers were tested,

* Standard Scaler uses Z-score to perform scaling.
* Min-Max Scaler rescales each feature to a common range [min, max] linearly.
* Robust Scaler removes the median and scales the data based on quantile range (Default IQR).
* Max-Abs Scaler rescales each feature to range [-1, 1] by dividing through the largest maximum absolute value in each feature. (250)

**4**

Before inputting the train data into the model, the number of rows and columns of the train and test datasets were checked with “.count()” and “len()”. This is to validate the data and ensure that the predicted results are not trivial or unrealistic.

Train:

Number of rows: 51337

Number of columns: 2

Shape of the Dataframe: (51337, 2)

Test

Number of rows: 12860

Number of columns: 2

Shape of the Dataframe: (12860, 2)

These are the final features of the data that will be used for building the machine learning model.

Predictor Features: "year", "floor\_area\_sqm", "remaining\_lease", "town\_one\_hot", "flat\_type\_one\_hot", "flat\_model\_one\_hot", "storey\_range\_one\_hot".

Xcols\_scaled: a vectorized column that consists of all the predictor features scaled using feature scaling. Used as the “featuresCol”.

Target Variable: "resale\_price". Used as the “labelCol”.

The model used will be a Linear Regression model as it is the model best suited for the given scenario. This is because we are building a predictive model which predicts the numerical value for a target variable, “resale\_price”, based on its predictor features, “Xcols\_scaled”. The model is built with the “LinearRegression” function imported from Spark MLlib. The model is then trained and fitted with the train dataset. (194)

**5**

To evaluate the model, “regressor.evaluate()” and “.predictions.show()” was used on both the train and test data to obtain the model’s predicted values. This allows us to obtain metrics that will be useful for evaluating the model’s performance. Since we are building a linear regression model, we will be using Mean Squared Error, Mean Absolute Error, and R-Squared as the metrics to evaluate performance.

Mean Squared Error and Mean Absolute Error are based on the squared and absolute error between the observed and predicted resale price. Thus, the closer these values are to 0, the more accurate the model.

R-Squared determines the measure of variance between the dependent variable that is explained by the independent variables. It measures the goodness-of-fit for linear regression model from a range of 1 to 0, where 1 indicates a perfect model.

The models were tested with four different feature scalers as well as transforming “remaining\_lease” from year to month to determine the best performing model based on the metrics.

Comparing the results, we can see that the model using “remaining\_lease” in month and scaled using Standard Scaler has the best overall performance among the models tested.

**6**

Overall, the performance of the machine learning model built is satisfactory as it was able to achieve a R-squared value greater than 0.75. Through data exploration and analysis, we were able to identify key trends as patterns that helped identify features that were useful for predicting the target variable based on the prediction problem statement.

Based on the accuracy of the final model, there is definitely room for improvement as I believe the error present in the model can be further reduced to improve the performance.

Some possible improvements that can be made are increasing the complexity of the model by adding new features derived from the original features. This will allow our model to train on more data which may potentially increase the model’s performance. Enriching the dataset with data and features from other sources, such as “distance to nearest CC or MRT” or “last renovation date” can also help to further refine the model.

Lastly, model accuracy can be further optimized by testing other transformation methods such as outlier handling, normalizing, binning, and other numerical transformers such as log transformer, box-cox transformer, etc. To compare and find functions that best suit the data. (195)

Explain what Data Exploration is. Why it is needed. Explain the context. Explain the Examples. Identify any errors to be corrected. Explain the trends

Problem Statement is: Predict the resale prices of any given HDB resale transaction. (Based on certain features, predict the resale price of a given HDB)

Target would be resale price. It is a Linear Regression Model because we a predicting the value of a target variable which is Resale Price. (Can explain other models briefly and why we used this)

After explaining all the data exploration, explain which features were dropped and why. (These features will be helpful in the problem statement) (based on our findings from the exploratory analysis and looking at the relationship between the features, we can form a problem statement for machine learning modelling.) The value-based problem statement is to predict the resale price of a given HDB transaction.

Transformation will use numerical transformation to ensure the model is not affected by the changes in magnitude of the values. One-hot encoding for the categorical values (works the best) look into change column types

Visuals will be in the second section with alight explanations.

Talk about OHE briefly

Explain each scaler

Lease commencement is basically the year that the 99 year lease begins. Since our dataframe already has remaining lease in months, I see no reason to keep this column in the model.

Conclusion: There does not seem to be any relationship between the block number and the resale price of a flat. Therefore, we can safely drop this feature later.

The table above seems to indicate that the average resale price of flats has decreased slightly from 2017 to 2019. This is quite interesting as, despite numerous public sources stating that HDB prices are on the rise due to an increasing demand for housing, the table above seems to indicate a downward trend in resale pricing.

From the table, it can be observed that there is a trend where the flat is priced higher if it is located on a higher storey range.

By comparing the 2 tables, we are able to confirm that the resale price of the flat is proportional to the floor area of the flat. (more area = more expensive)

It can be observed that the top 3 towns with the highest resale value is Bukit Timah, Bishan, and the Central Area. It may indicate that these are high value locations in terms of housing.

It can be observed that a higher remaining lease tends to mean a higher resale price based on the comparison between the two tables above.

(MAKE SURE TO JUSTIFY ALL THE METHODS USED IN THE MODELING)