MIS772 – Predictive Analytics

T2 2023

Assignment 2 – Individual
Student name: Sachin Bhat
Student number: 218676233

Executive summary (1 page)

AirBnb is world leader in providing accommodations to people (customers) around the globe who want to rent the spaces for several purposes such as vacations, work, travel etc. The firm also caters to people (property hosts) by helping them post their accommodations' advertisements in particular locations around the world using the houses and locality's images, descriptions about property, neighbourhood, people it can accommodate etc. Airbnb generates its earnings by taking commissions on the successful bookings that customers are able to make. It operates as an easy-to-use digital platform that links tourists with private property hosts.

Executive Problem Statement:

We are currently looking at Sydney (Australia) based listings. The company has provided us with 2 datasets that include approximately 37,000 rental listings and 5,49,000 associated customer reviews. Every day there are new hosts registering their properties along with new customers who are making bookings on their platform. Airbnb has approached us with a request to develop an advanced and effective method which would enable them to analyse and predict customer feedback about their stays particularly at their Sydney rental properties. It is extremely hard for the firm to assess each customer review and plan course of action on its basis.

Executive Solution Statement:

We are provided with negative and positive words which the hosts may use in their property descriptions, or the customers may use in their stay's reviews. Based on those words, we have devised a method to make use of hosts' property descriptions and customers' reviews to provide deductions. After carefully working around the data provided to us, we were able to derive a positive correlation between property hosts' description and customer reviews. It came to light that hosts who wrote more positive and well-oriented property descriptions (Figure 1) received more positive reviews from their guests (Figure 2). Key factors (Figure 3.1) such as accommodates, bathrooms, bedrooms, price, host responsiveness, cleanliness, location, accuracy, check-in, value, reviews per month and security deposit emerged as influential factors impacting review scores. For identifying meaningful property segments in the data, we made use of the influencing factors (Figure 4.2) such as accommodates, bathrooms, bedrooms, accuracy, check-in, cleanliness, host responsiveness, location, and value.

After performing the predictive modelling and applying techniques to increase their efficiency, we would recommend the following: (1) Linear Regression Model with the least root mean squared error of 4.534 +/- 0.000 and squared corelation of 0.746 (Figure 6). (2) Improved Data Clustering Model (k-Means) with clustering k = 5, Davies Bouldin = 1.220 and Example distribution = 0.319 (Figure 10,11).

After following our recommendations, Airbnb would be able to efficiently identify the dynamics of customer feedback, host property descriptions, property segmentation in their Sydney market. Insights generated using our methods would empower the hosts and Airbnb to optimise property descriptions, predict review ratings for new listings and modifying their offerings to cater to different guest preferences. Ultimately it would provide an excellent customer experience while supporting the host. Bringing in more users (hosts and customers) and in-turn revenue, this would further strengthen Airbnb's position in the competitive domestic tourism sector.

Limitation of our Linear Regression model is that it may suffer from multicollinearity wherein some independent factors might be highly correlated with each other. Limitation of our k-Means Data Clustering model is that it requires us to input the number of clusters beforehand and it is also sensitive towards outliers.

Data exploration, pattern discovery, and preparation (2 pages)

Data selection:

The provided datasets were imported into the RapidMiner environment and were worked upon as per the requirements. The listings_clean dataset has 30 regular attributes and reviews_clean has 6 regular attributes. Relevant utilization of these attributes as predictors and label helped us to move forward.

For Task A – Using the listings_clean dataset, from the **host's** perspective we utilised Description and ID as the **predictor** attributes. Using the reviews_clean dataset, from the **customers'** perspective we utilised Comments and ID as the **predictor** attributes. Using Set Role for both these datasets we set the role of ID as ID.

For Task B (Linear Regression) – The selected Predictor attributes must be related to the Label attribute. Using the listings_clean dataset, we selected the following attributes as the **predictors:** accommodates, bathrooms, bedrooms, price, review_scores_accuracy, review_scores_checkin, review_scores_cleanliness, review_scores_communication, review_scores_location, review_scores_value, review_scores_month, security_deposit. We selected review_scores_rating as the **label** attribute because based on it we can predict customer feedback about theirs stays at Sydney Airbnb rental properties.

For Task C (Data Clustering using k-Means) – We had the requirement of determining the most meaningful number of distinct property clusters and identifying ways for describing them. Using the listings_clean dataset, we selected the following attributes as **predictors**: accommodates, bathrooms, bedrooms, review_scores_accuracy, review_scores_checkin, review_scores_cleanliness, review_scores_communication, review_scores_location, review_scores_value. No label attribute is necessary.

Dealing with missing values:

Dealing with missing values is extremely crucial as it impacts the meaningfulness of our datasets. Various methods such as data transformation, imputation, taking mean or mode etc. can be used to deal with missing values.

For Task A – The listings_clean dataset had 1000's of missing values and reviews_clean had 100's of them among the attributes we are interested to utilise. To handle the missing values for our selected attributes in both the datasets, we made of use the Filter Examples operator. We chose the condition class as no_missing_attributes which would filter through the data and match us with examples without any missing values.

For Task B – We utilised the Replace Missing Values operator on listings_clean dataset to handle missing values in our selected attributes. We asked it to replace the missing values with their averages.

For Task C – Here we again made use of the Replace Missing Values operator on listings_clean dataset to handle missing values in our selected attributes and we asked it to substitute the missing values with their averages.

How we Transformed the Data:

Data transformation of relevant attributes must be performed if and how the business problem requires it.

For Task A – We had to perform the following transformations for sentiment analysis. For the listings_clean dataset, we utilised the Nominal to Text operator to transform the nominal attribute **description** to text form. For the reviews_clean dataset also, we utilised the Nominal to Text operator to transform the nominal attribute **comments** to text form.

For Task B – No transformations were done. The label attribute review_scores_rating from listings_clean dataset is of integer type suitable for Linear Regression.

For Task C – We used the Normalize operator on the selected attributes to scale values so that they could fit in a specific range.

Analysing the Data Distribution:

We utilised bar charts for data exploration to stress on any correlations existing between the raw sentiment scores of host's description of a property (Figure 1) and the customer's review of a property (Figure 2). Utilising the commonly used positive and negative words fed to our model the raw sentiment scores were calculated for host description and customer reviews (total positive words – total negative words). Majority of the hosts used positive words and very few used any negative words in their property descriptions as a result the sentiment value is quite high (Figure 1). Also, most of the customers used positive words and few used negative words in the property reviews due to which the sentiment value is moderately high (Figure 2). Majorly the sentiment scores between 0-10 are secured among host description and customer review. Hosts with high sentiment scores i.e., who wrote positive and well-written property descriptions are more likely to receive more positive reviews from their guests.

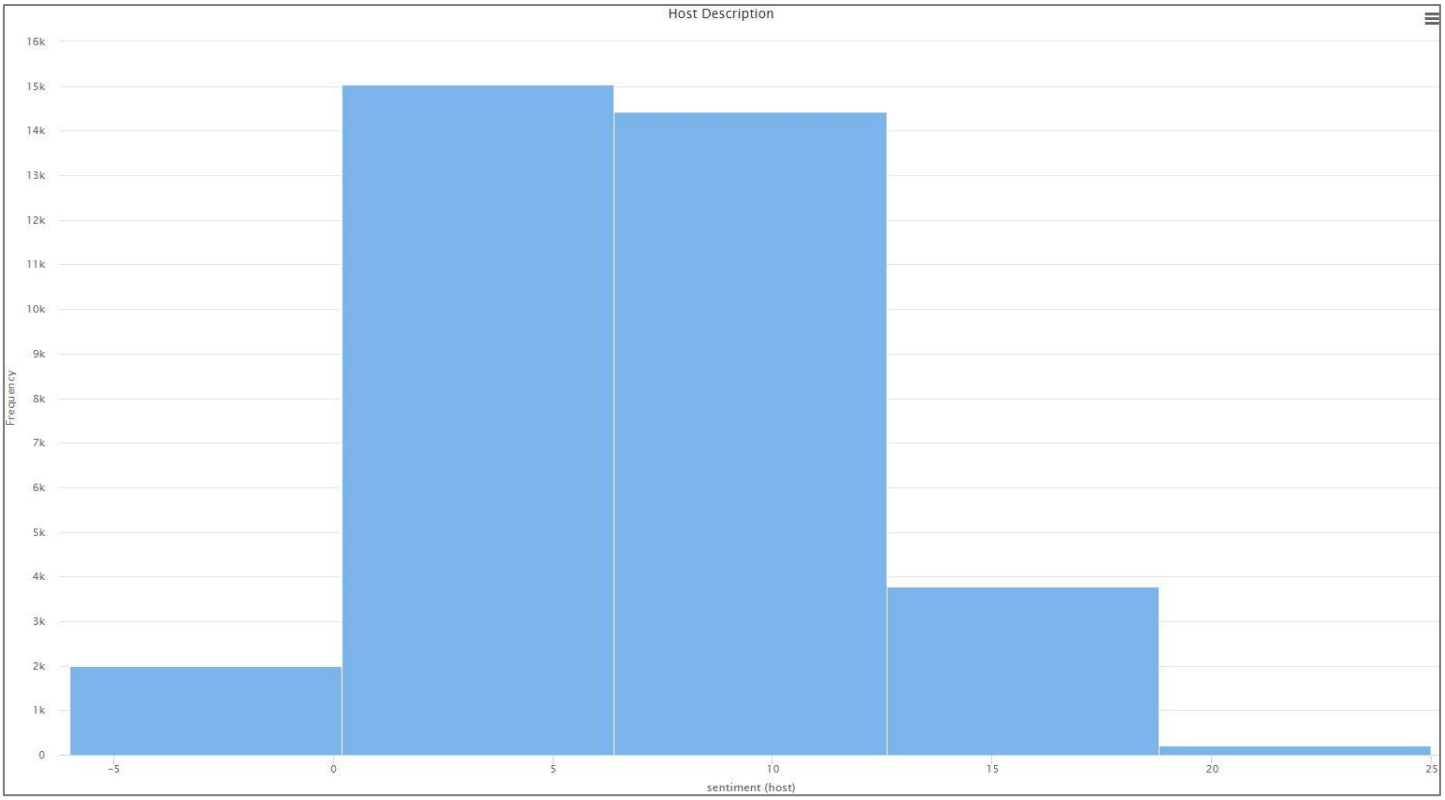

Figure (1). Host Description Sentiment

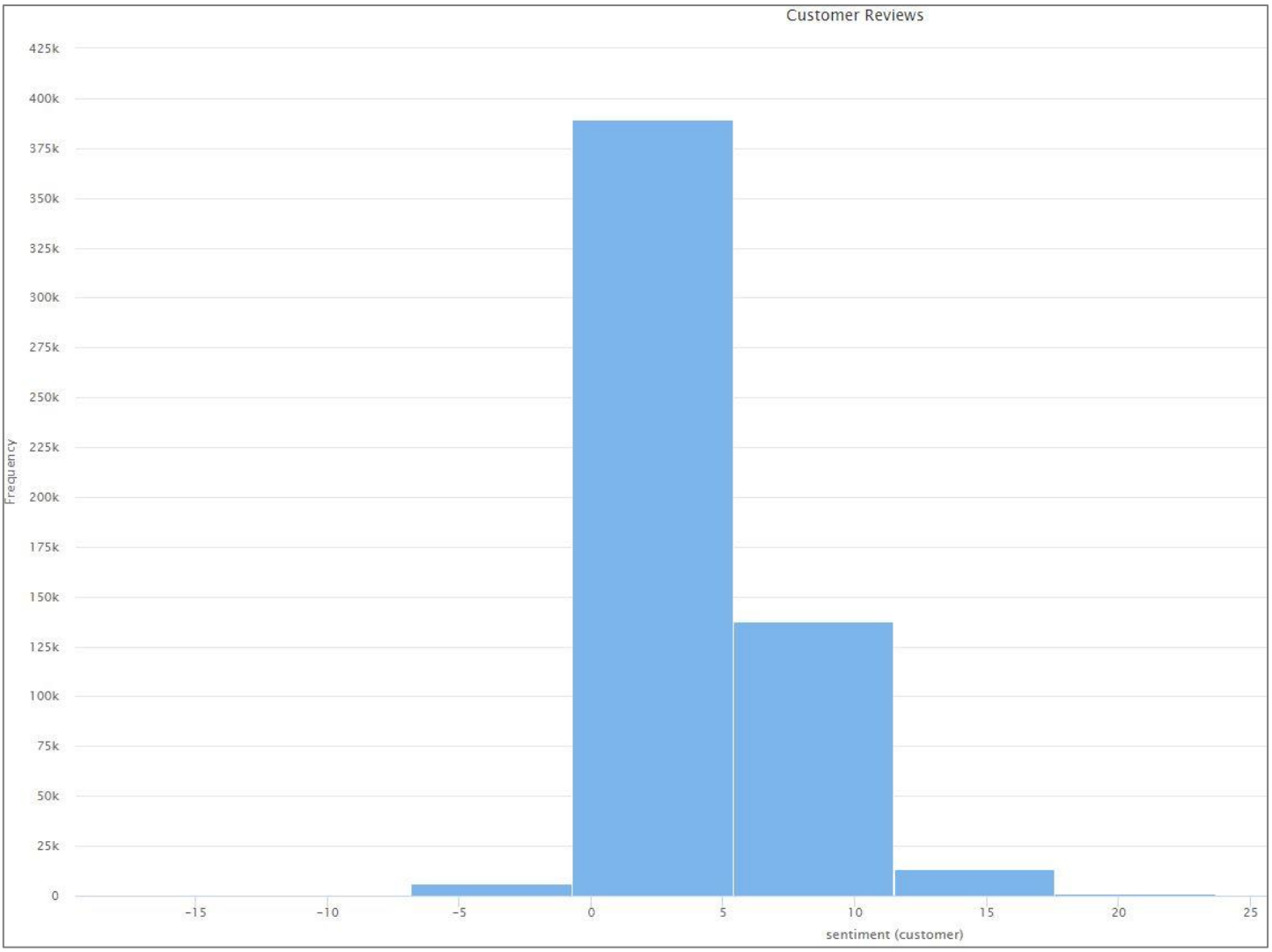

Figure (2). Customer Reviews Sentiment

Predictive modelling (2 pages)

After carefully exploring the data, patterns in it and preparing the data for modelling, we have acquired a meaningful dataset fit to be used. For our Predictive Modelling we have made use of RapidMiner's Linear Regression (Task B) and Data Clustering (using k-Means) models (Task C). Linear Regression goes well with our scenario as we have to perform numerical prediction. Also, because here, values of several predictor attributes influence the value of our label attribute. Secondly, Data clustering using k-Means goes well with our scenario as it groups similar examples together into meaningful clusters which is essential in our case.

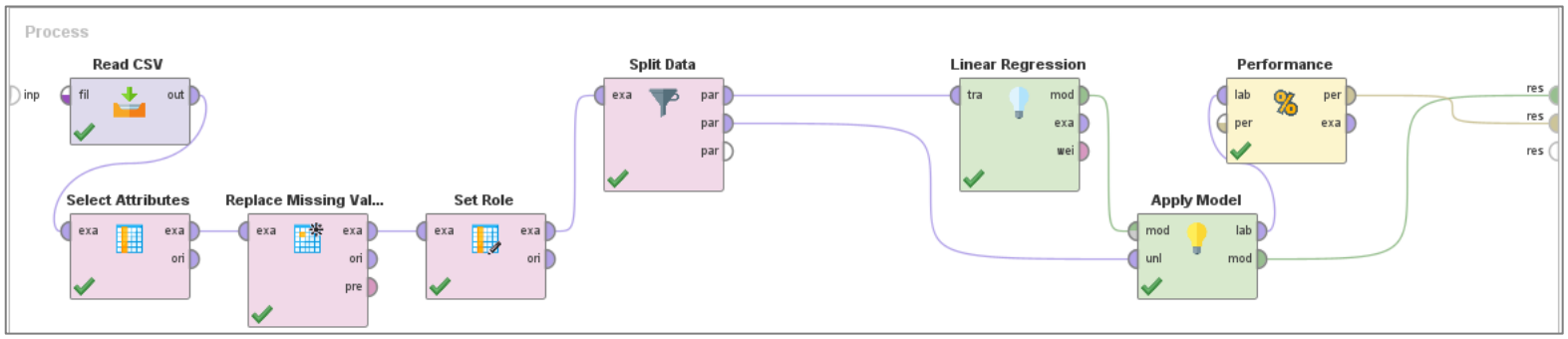

Figure (3). Linear Regression with M5 Prime

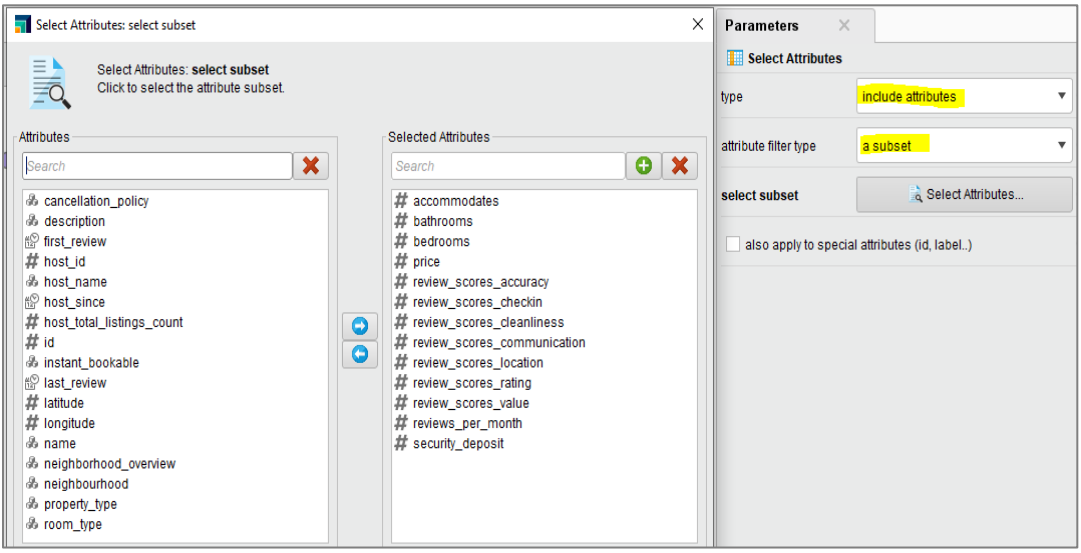

Figure (3.1). Select Attributes Parameters

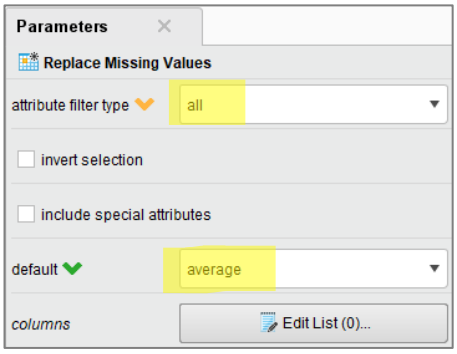

Figure (3.2). RMV Parameters

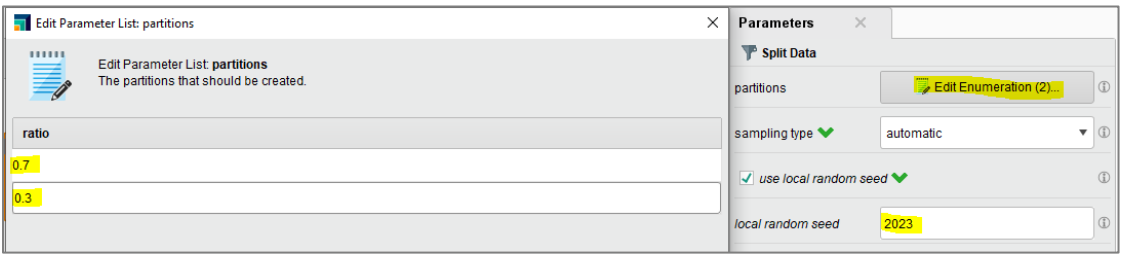

Figure (3.3). Select Attributes Parameters

As seen in Figure (3) the Linear Regression approach was utilised. We used the Read CSV operator to bring in the dataset into the RapidMiner environment. Using Select Attributes (Figure 3.1) we selected these attributes as predictors as they influence our label attribute i.e., review_scores_rating. Based on it we can predict customer feedback about theirs stays at Sydney Airbnb rental properties. Using Replace Missing Values operator (Figure 3.2) we substituted missing values with average values. Using Set Role, we set review_scores_rating as the label attribute. Using Split Data (Figure 3.3), we split the data into 2 parts – 70% of it was used to train the model and 30% was reserved to test the model exercising the Apply Model operator. The local random seed was set to **2023** i.e., a **constant**. As displayed in Figure (3), then we used Linear Regression to train 70% of the data. Then the trained output from the model was passed through the Apply Model operator which applied it on the test data. Then the trained output from Apply Model was passed through the Performance operator to record its performance based on root mean squared error and squared correlation. The feature selection is set to M5 prime with a minimum tolerance of 0.05 (neutral between its range of 0-1) to eliminate colinear features.

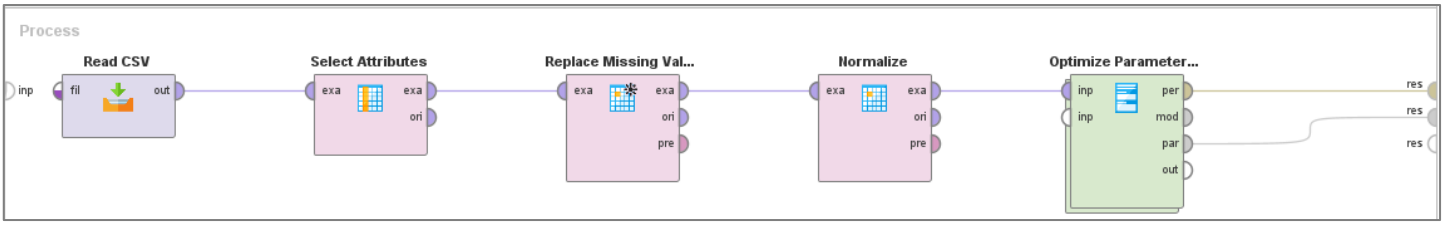

Figure (4). Data Clustering

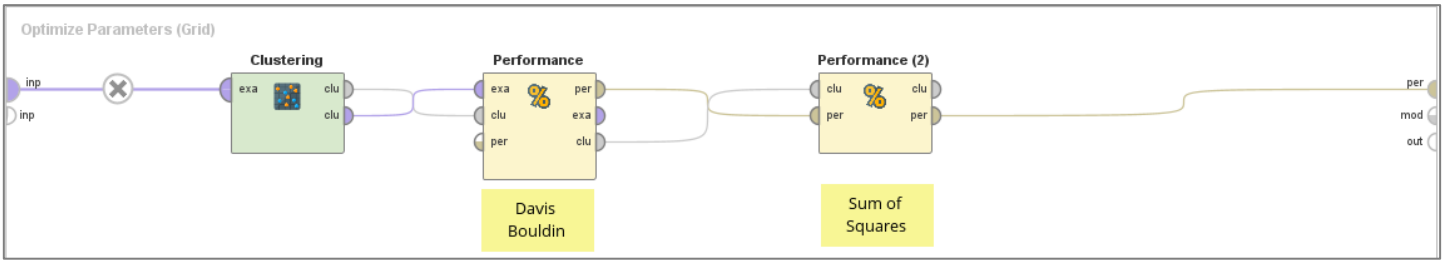

Figure (4.1). Inside Optimize Parameters (Grid) Nester Operator

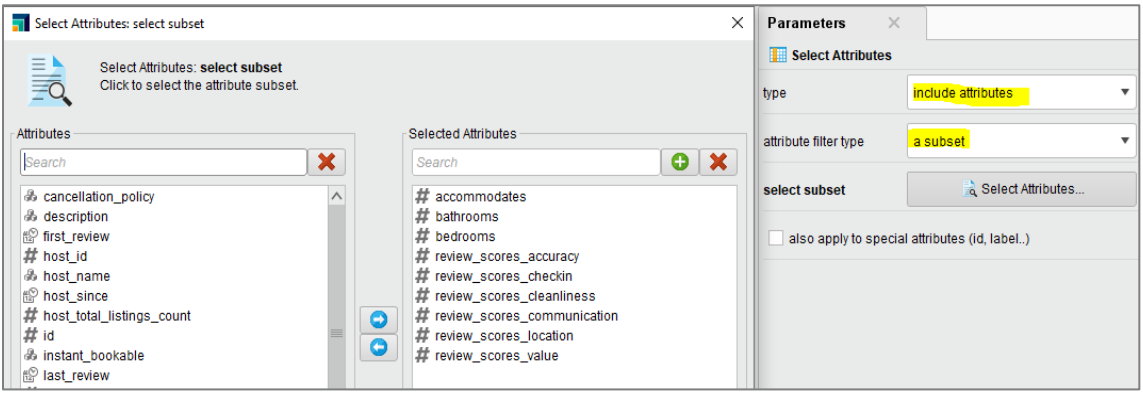

Figure (4.2). Select Attributes Parameters

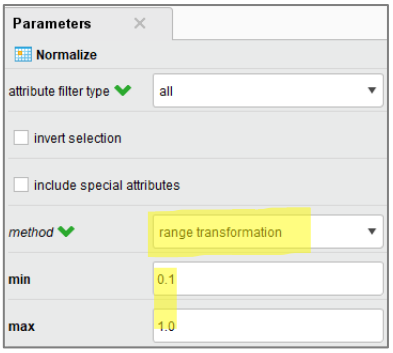

Figure (4.3). Normalize Parameters

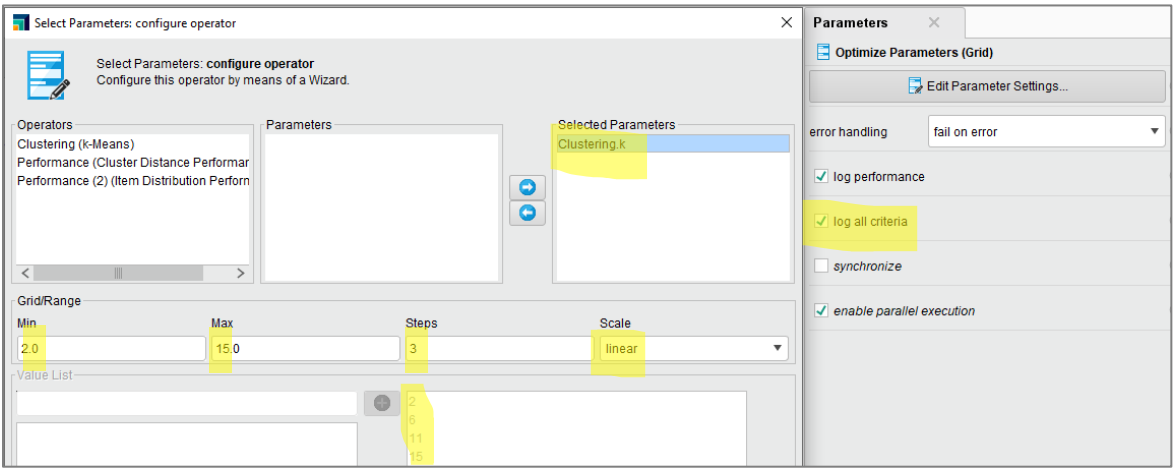

Figure (4.4) Grid Optimize Parameters

As seen in Figure (4) the Data Clustering approach was used. After importing the dataset using Read CSV, we selected the predictor attributes using Select Attributes operator (Figure 4.2). We selected these attributes as our predictors because these would be suitable for forming clusters which would include similar examples grouped together. It aligns with our requirement of determining the most meaningful number of distinct property clusters and ways to describe them. Replace Missing Values Operator was used to replace the missing values with average values (Figure 3.2). Then we used the Normalize operator to scale values so that they could fit in a specific range of 0.1-1 with the method of range transformation (Figure 4.3). Then we pass the data through Optimize Parameters (Grid) nested operator to log performance measures. We checked the log all criteria box to record all values i.e., Davies Bouldin, SumofSquares. From clustering (k-Means) operator, we selected Clustering.k as the selected parameter with minimum range as 2 and maximum range as 15 through 3 iterations (Figure 4.4). Scale was set to linear. This nested operator includes 3 operators (Figure 4.1) – 1) Clustering with value of k set to 2, max runs as 10, Numerical Measures as the measure type, Euclidean Distance as the numerical measure, local random seed set to 2023 i.e., constant. 2) Then Performance (Cluster Distance Performance) with Davies Bouldin as the main criterion and we tick main criterion. We also tick **maximize** to avoid the value's multiplication by -1 because Davies Bouldin must be reported as a positive value. 3) Then Performance (Item Distribution Performance) where we select the measure as sum of squares. Then the output from Optimize Parameters would give us the results of 3 iterations with 4 different k values along with 4 Davies Bouldin and 4 Example Distribution values.

Model evaluation and improvement (2 pages)

As have performed our predictive modelling using Linear Regression and Data clustering in the previous section, we would now look at the results for both the approaches below.

For Task B i.e., for Linear Regression we had performed hold-out sampling during the modelling where the data was split into 2 parts:70% of it was used for training and 30% for testing. We got the Linear regression coefficients (Figure 5) and output performance vector (Figure 6) as our results. The Linear Regression coefficients (Figure 5) were 0.086 for bathrooms, + 0.156 for bedrooms, + 0.000 for price, + 0.000 for security_deposit, - 0.222 for reviews_per_month, + 2.377 for review_scores_accuracy, + 2.630 for review_scores_cleanliness, + 0.914 for review_scores_checkin, + 1.519 for review_scores_communication, + 0.189 for review_scores_location, + 2.681 for review_scores_value and - 4.566 was the intercept. Here, we achieved the following performance criteria (Figure 6): an averaged root mean squared error of 4.534 +/- 0.000 and a squared correlation of 0.746 between the label and predictor attributes.

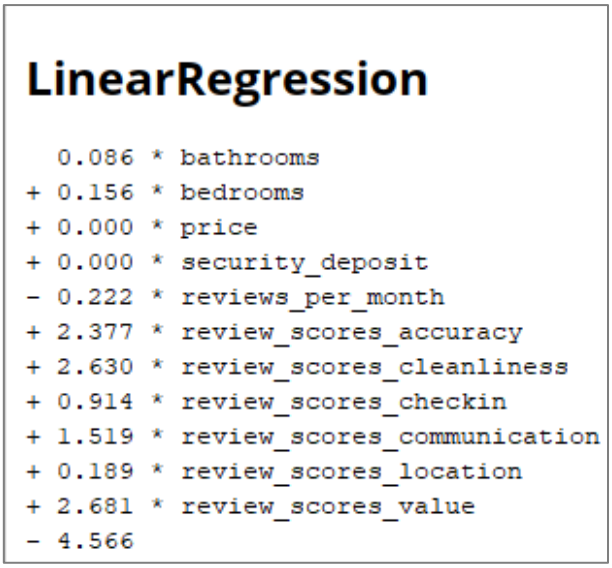

Figure (5).LR Coefficients

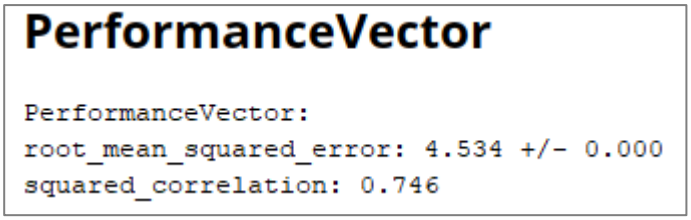

Figure (6).Performance Vector

To check for improvement, 2 methods were tried. In the first one, Replace Missing Value operator's default criteria was changed to 0. The Performance Vector obtained (Figure 7) gave us the RMSE of 6.254 +/- 0.000 and squared_correlation of 0.980. In the second one, Linear Regression Model's feature selection was changed to T-Test and the use bias box was unchecked. The Performance Vector obtained (Figure 8) gave us the RMSE of 4.548 +/- 0.000 and squared_correlation of 0.746.

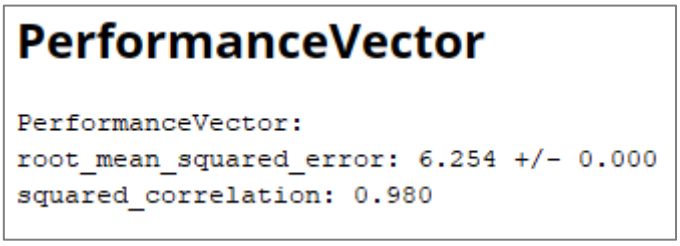

Figure (7).Performance Vector for RMV default as 0

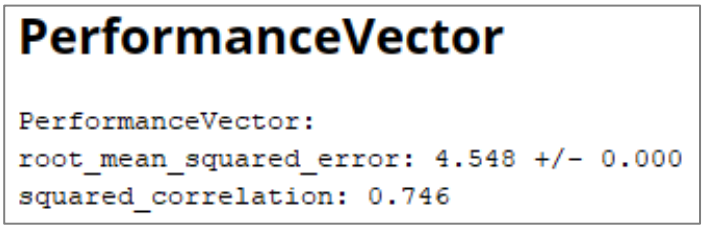

Figure (8).Performance Vector for T-Test without bias

For Linear Regression, our original model (Figure 3) performs effectively and provides the best results (Figure 6) as it has the least root mean squared error of 4.534 +/- 0.000 and shows high correlation. As closer the RMSE is to 0, the model performs more efficiently. We would recommend it.

For Task C i.e., Data Clustering using k-Means, we got the following results over 3 iterations. We had set the local random seed as 2023 i.e., a constant. With the value of k = 2, measure types for clustering was set to Numerical measures, and the numerical measure was set to Euclidean Distance. For Cluster Distance Performance the main criterion was changed to Davies Bouldin, and we checked the main criterion only and **maximize** (to prevent multiplication by -1 as Davies Bouldin is always reported as a positive value) boxes. As seen in Figure (9), for (1) clustering k = 2: Davies Bouldin=0.988, Example Distribution= 0.921, (2) clustering k = 6: Davies Bouldin=1.333, Example Distribution= 0.285, (3) clustering k = 15: Davies Bouldin=1.414, Example Distribution= 0.136, (4) clustering k = 11: Davies Bouldin=1.411, Example Distribution= 0.175. The closer Davies Bouldin value is to 0, the better clustering k is.

Optimize Parameters (Grid) (4 rows, 4 columns)

iteration	Clustering.k	Davies Bouldin	Example distribution
1	2	0.988	0.921
2	6	1.333	0.285
4	15	1.414	0.136
3	11	1.411	0.175

Figure (9).Output for Data Clustering (Original Model)

Optimize Parameters (Grid) (6 rows, 4 columns)

iteration	Clustering.k	Davies Bouldin	Example distribution
2	5	1.220	0.319
1	2	0.988	0.921
4	10	1.403	0.197
3	7	1.297	0.238
5	12	1.466	0.182
6	15	1.414	0.136

Figure (10). Data Clustering Output (Improvement)

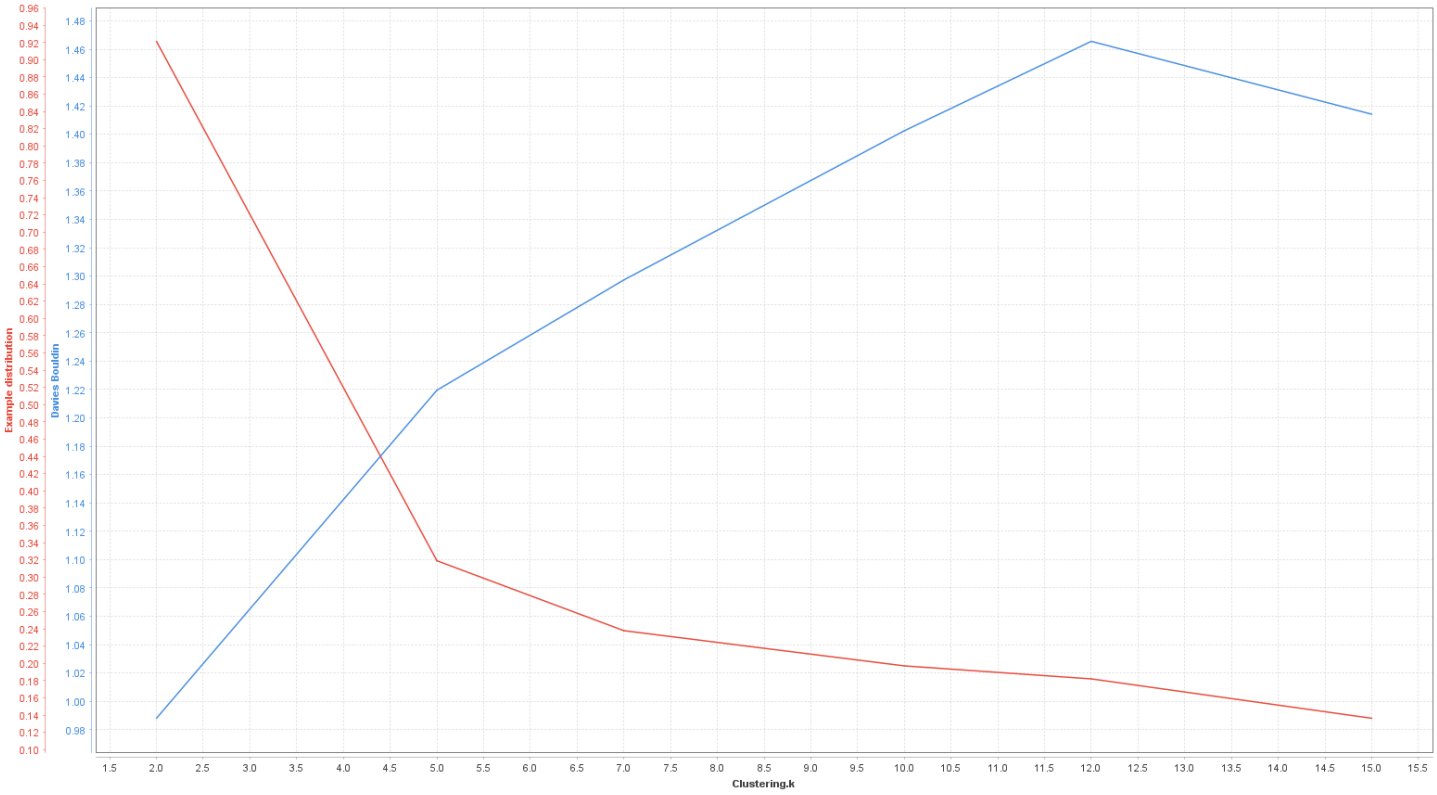

Figure (11). Improved Series Multiple Plot for Clustering k using Davies Bouldin (Blue) and Example Distribution (Red)

But here, we would consider Figure (10) and Figure (11) before making recommendations. After increasing the number of steps from 3 to 5 under the Grid Optimization's clustering k parameter we achieved better results. As the Sum of Squares forms an elbow like structure at clustering k = 5 (previously achieved at 6), we would consider that as the reporting value. Also, now we have a better Davies Bouldin value of 1.220 achieved at clustering k = 5 which is closer to 0. Our recommended Data Clustering model would have clustering k = 5, Davies Bouldin = 1.220 and Example distribution = 0.319. Also, PCA (Principal Component Analysis) with dimensionality reduction set to fixed number and number of components set to 5 was used for visualising clusters by reducing the dimensionality of the data. PC 1 explains 47.5% of variance in the data. PC 1 and 2 explain 71.7% of variance in the data. PC 1, 2 and 3 explain 81.7%. 90% is achieved at PC 5. 9 PC's put together give us 1 i.e., 100% of variance in the data is explained.