Aim
To clearly articulate understanding of the business problem and to present its solution to management.

Business Problem
There remains a great scope in improvement in the quality of service provided by BFA to its clients. The recommendations for various strategies regarding adoption of services such as Book_table and Online_order must be improved as the low quality has been affecting the overall revenue and the retention of clients. How can BFA provide effective consultancy to its clients (new and old restaurants in the Bangalore region) about what their strategy should be when formulating a service offering. Specifically, regarding offering secondary services of booking tables and online ordering to its patrons.

Solution: The proposed solution to the above business problem is to leverage the available data set to create predictive models that will allow BFA to predict the strategies for prospective and current clients, not based on instinct and human judgement but actual market trends backed by solid data. Interaction with the build analytical tools will allow management to offer confident suggestions to the restaurants. This will allow BFA to tackle the quality assurance issues, followed by which BFA can expect to see better client response, satisfaction, retention and financial returns in the medium to long run.

When adopted in the long run, BFA can expect to improve its service quality by multiple factors and increase its returns by multiple factors. Data driven solutions for quality assurance is a trusted approach, especially backed by the current success and trends of the market. IT is only wise to leverage these technologies and invest in the infrastructure to yield the untapped returns. The analytical process takes into account the detailed characteristics of the restaurants to predict what should be the behaviour of a restaurant that has the same or similar characteristics.
The proposed solutions offer recommendations for old as well new restaurants. The recommended plan of action is to beta run these solutions and see if the made recommendations in fact improve the consultancy quality or not. Once, the quality is assured the analytical processes can be deployed completely. This can also be approached by comparing satisfaction of clients who were given recommendations based on the results offered by this analytical solutions and the customer satisfaction of clients who were addressed with the previously adopted methodology. This will consolidate the value of this analytical solution to the company. Leveraging data for consultancy solution will allow BFA to be a industry leader in consultancy.

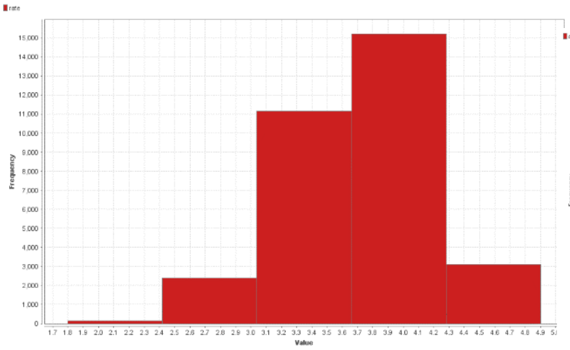
Data exploration and preparation in RapidMiner

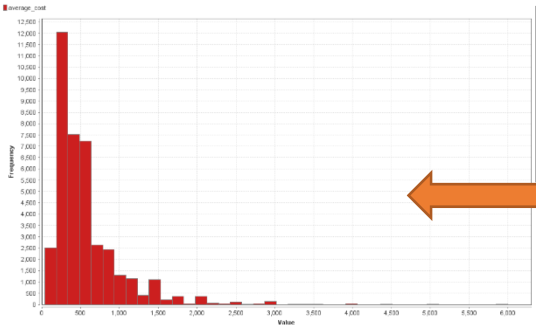
Aim

To demonstrate your understanding of data and its non-text attributes.

Expectation

LP1:

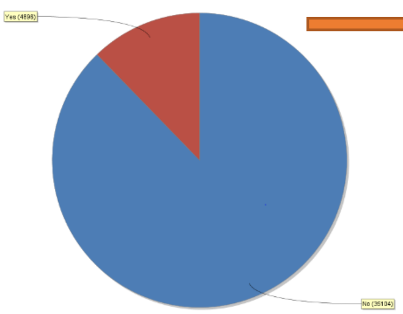




As per graphs A&B: The Distribution of 'rate' is skewed leftward and 'average_cost' is skewed to the right. The medians for these variables are better representatives.

Illustra.A: Distribution of Rate

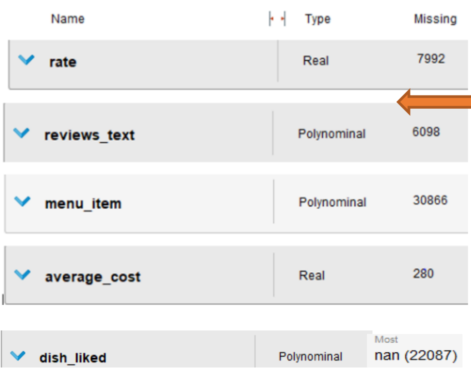
Illustra.B: Distribution of Average_cost



The red dipicts the minority class of **'YES'(4896).** It is an extremely unbalanced data. However, unbalanced data will be fixed if it allows higher acccuracy for models predicting book_table.

Illustra.C: Composition of '*book_table*'

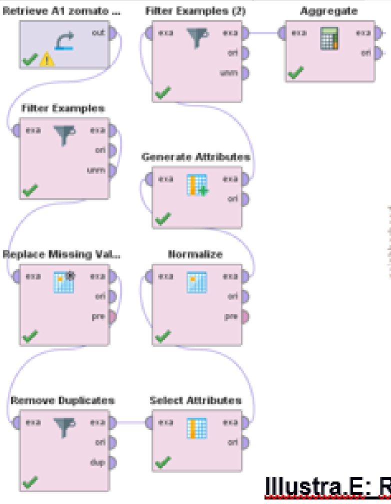
LP2:

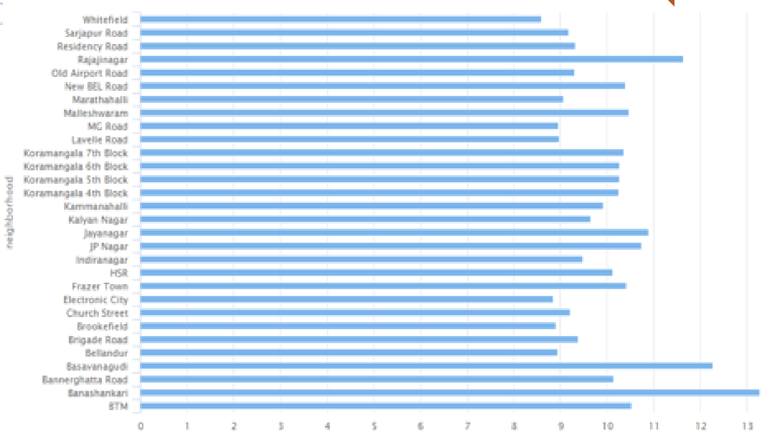


Attribute	No. of missing values; treatment
Rate	7,992; Most represent reviews for new restaurants. Other replaced with median.
Reviews_text	6,098; most represent reviews for new restaurants so no treatment as the entries will be filtered out
Menu_item	30,866; attribute will be excluded as most of the values are missing.
Average_cost	280; no treatment.
Dish_nan	22,087; attribute will be excluded as majority of the values are missing

Illustra.D: Table for missing values

Solution A:





Process: The data is retrieved>The new restaurants(without feedback) are filtered out> 'rate'missing value are replaced with median>reviews representing the same restaurants are removed to avoid inflation of new aggregate>average_cost and rate are normalized[0,1]> attribute; rate/averagecost is generated> examples with average cost=0 are filtered out> the rate/average_cost variable is grouped as per neighborhood and is represented as a bar graph. BANASHANKARI is the most attractive Neighborhood.

Illustra.E: RMProcess and bar chart showing most attractive neighbourhood

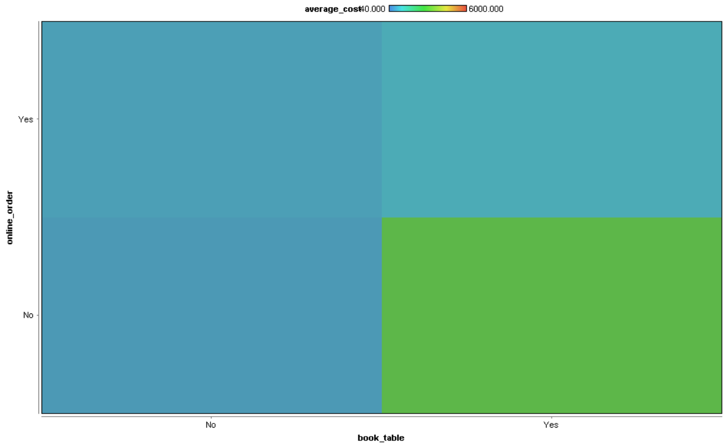
Discovering Relationships and Data Transformation in RapidMiner

Aim

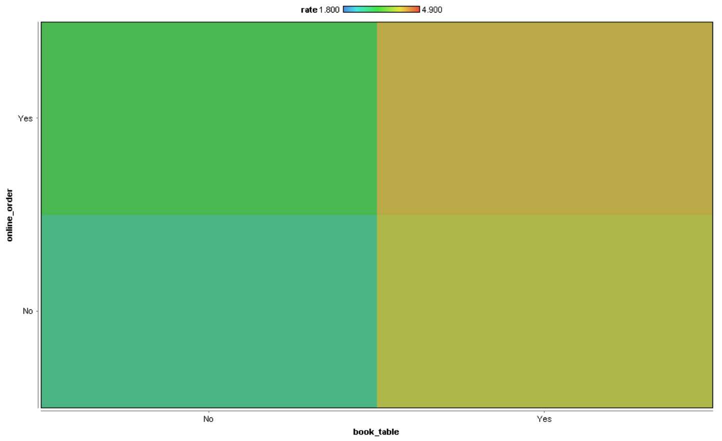
To demonstrate your understanding of data by describing complex relationships between non-text attributes.

Expectation

The labels: (A.)Book table (B.) Online_order
The predictors:COMMON for all models(new restaurants, old restaurants, book table, online order): **(A.) Average_cost (B.)Cuisines (C.) location (D.) meal_type (E.) neighbourhood (D.) rest_type.** For models concerned with established restaurants: **(A.) average_cost (B.) rate (C.) votes**



The adjacent box plot shows a relationship betwee online_order (Yaxis), book_table(Xaxis) and the average_cost(colour group). We can infer that reviews for restaurants with only the table booking service tend to have higher average_cost than othe restaurants. While, restaurants without any of the services tend to have lower average_cost. (this graph was executed as average_cost has high weight for both the labels; book_table, online_order.)



The adjacent box plot shows a relationship betwee online_order(Yaxis), book_table(Xaxis) and the ’*rate’ variable.* The reviews for restaurants with bot the services tend to have higher ‘*rate’ than other restaurants.* While the reviews for restaurants without any of the services has lower ‘rate’. Rate has been assigned a higher weight for the labels. This shows a relationship between the two service and the rate variable.

Attribut...	rate	votes	average...
rate	1	0.432	0.385
votes	0.432	1	0.381
average...	0.385	0.381	1

There is a positive/direct correlation between all th numeric variables. The metrics are not very high and do not exceed 0.5 for any combination of variables.

Extension

(A.)

attribute	weight
online_order	0.001
neighborhood	0.002
location	0.012
meal_type	0.037
menu_item	0.042
cuisines	0.044
rest_type	0.069
dish_liked	0.075
rate	0.100
votes	0.224
average_cost	0.317

(B)

attribute	weight
menu_item	0
book_table	0.002
neighborhood	0.004
location	0.007
rest_type	0.017
cuisines	0.038
meal_type	0.043
votes	0.056
dish_liked	0.063
rate	0.112
average_cost	0.180

The adjacent illustration (A.) shows that as per weight (information gain) votes, rate and average cost have the maximum influence over book_table. Further, the illustration (B.) shows rate, average_cost, dish_liked have the maximum influence over online_order.
However, since the number of attributes are limitec all the listed attributes could be used as predictors in the respective models.
The labels for the models would be online_order & book_table respectivelyas the business problem requires prediction of strategy related to table booking and online order for restaurants.

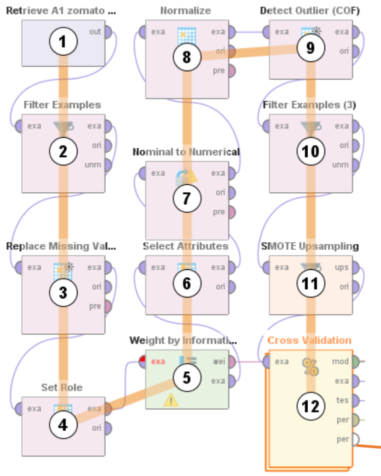
Create a Model(s) in RapidMiner

Aim

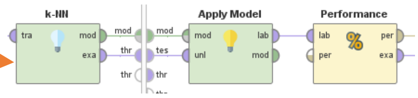
To explain details of developed classification models and selected methods for data preparation and reporting.

Expectation

Illustra.F: Process designed for building a KNN model to predict *Book_table for established restaurants*.

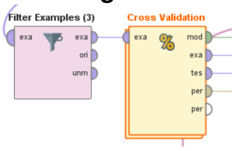


Process; 1>The Zomato_train.cvs is retrieved2>Entries representing new restaurants are filtered out (no feedback, refer **Illustra.G**) 3>Missing values for rate are replaced with the median (refer **Illustra.A**) of '3.7'4> The label is set as book_table.5>the weights for the predictors are calculated in relation to the label.6> The attributes are selected for the model. 7>The categorical data is converted to numerical as KNN operates on measuring distances between data points and such distances cannot be measured between *words* or *categories*. 8> The numeric data is normalized to within a range of [0,1] so that KNN can estimate the distances without being influence by one variable with a large range,9&10> The outliers are detected and then removed using filter '*outliers=false'* to avoid influence of outliers over the model.11>The SMOTE operator is used to avoid the influence of the unbalanced data over the model, since the data has an extremely unbalanced data (refer to Illustra.C) 2> Cross Validation operator is used to; train the KNN model, apply the model and then check the performance of the model using accuracy, Kappa, weighted mean precision and weighted mean recall.

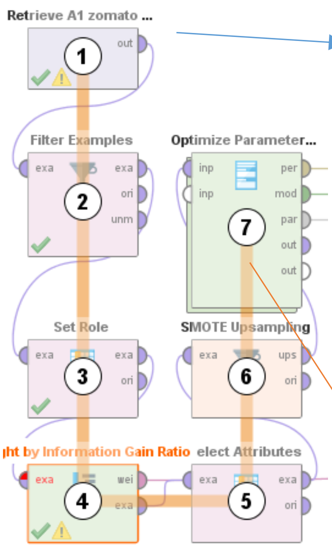


Illustra.G:

The same structure was used to build a process for Online_order(KNN), except without the SMOTE as the data was not imbalanced and the predictors were selected according to the weights estimated by information gain for this label.

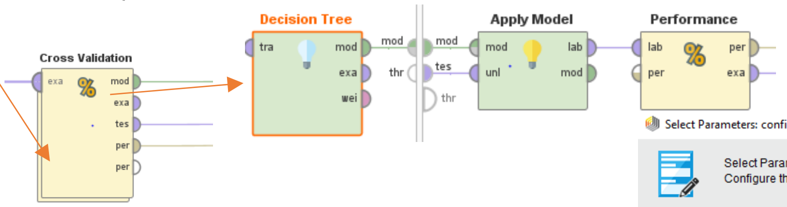


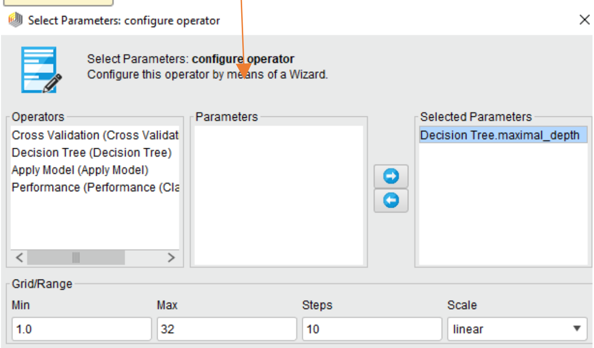
Extension



Illustra.H: Process designed for building a decision tree model to predict Book_table *for established restaurants*

Process; 1> Zomato_train.CVS is retrieved. 2>Entries representing new restaurants are filtered out(no feedback, refer Illustra.G) 3> The label is selected as Book_table4>The weights of the attributes are calculated by weights by information gain5>And the relevant attributes are selected 6> SMOTE is used to balance the unbalanced data (Refer Illustra.C)7> The optimize parameter operator is used to find the optimum value of '*Maximal_depth' so that the model can be prevented from overfitting. The upper limit for this parameter is set as 32 because a really high upper limit may give better accuracy, but it will cause overfitting of data. The cross validation operator is nested in this operator and the model, apply model and performance are further nested in cross validation.*





The same structure was used to build a process for Online_order(decision tree), except without the SMOTE as the data was not imbalanced and the predictors were selected according to the weights estimated by information gain for this label.

Evaluate and Improve the Model(s) in RapidMiner

Aim

To report and explain the performance of developed classification models.

Expectation

TableA: Table showing the accuracy and kappa for different models on training and testing data,

Training data

```
PerformanceVector:
accuracy: 93.77%
ConfusionMatrix:
True:   No      Yes
No:     8630    306
Yes:    345     1162
kappa: 0.745
ConfusionMatrix:
True:   No      Yes
No:     8630    306
Yes:    345     1162
```

```
PerformanceVector:
accuracy: 62.93%
ConfusionMatrix:
True:   No      Yes
No:     10      0
Yes:    3871    6562
kappa: 0.003
ConfusionMatrix:
True:   No      Yes
No:     10      0
Yes:    3871    6562
```

Testing Data

```
PerformanceVector:
accuracy: 93.33%
ConfusionMatrix:
True:   No      Yes
No:     5600    81
Yes:    386     934
kappa: 0.761
ConfusionMatrix:
True:   No      Yes
No:     5600    81
Yes:    386     934
```

```
PerformanceVector:
accuracy: 64.09%
ConfusionMatrix:
True:   No      Yes
No:     6       0
Yes:    2514    4481
kappa: 0.003
ConfusionMatrix:
True:   No      Yes
No:     6       0
Yes:    2514    4481
```

Label	*Training Data*		*Testing Data*	
Model type	*Accuracy*	*Kappa*	*Accuracy*	*Kappa*
Book_table;				
Decision Tree	*93.77*	*0.745*	*93.33*	*0.761*
KNN	*91.12*	*0.822*	*85.80*	*0.516*
Online_order;				
Decision Tree	*62.93*	*0.003*	*64.09*	*0.003*
KNN	*69.07*	*0.313*	*79.93*	*0.557*

Testing Data

```
PerformanceVector:
accuracy: 91.12%
ConfusionMatrix:
True:   No      Yes
No:     18031   811
Yes:    2910    2013
kappa: 0.822
ConfusionMatrix:
True:   No      Yes
No:     18031   811
Yes:    2910    2013
```

```
PerformanceVector:
accuracy: 69.07%
ConfusionMatrix:
True:   No      Yes
No:     1929    1283
Yes:    1944    5278
kappa: 0.313
ConfusionMatrix:
True:   No      Yes
No:     1929    1283
Yes:    1944    5278
```

Testing Data

```
PerformanceVector:
accuracy: 85.80%
ConfusionMatrix:
True:   No      Yes
No:     5264    272
Yes:    722     743
kappa: 0.516
ConfusionMatrix:
True:   No      Yes
No:     5264    272
Yes:    722     743
```

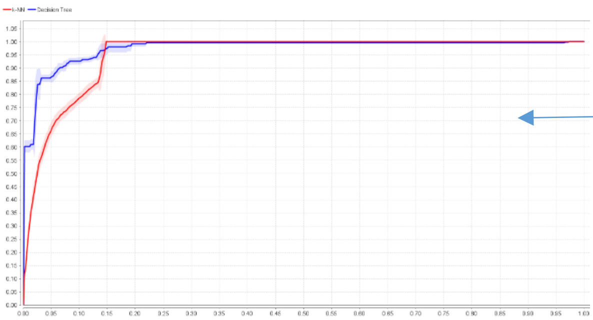
```
PerformanceVector:
accuracy: 79.93%
ConfusionMatrix:
True:   No      Yes
No:     8482    2533
Yes:    4455    19340
kappa: 0.557
ConfusionMatrix:
True:   No      Yes
No:     8482    2533
Yes:    4455    19340
```

Performance measurement using hold out validation, on comparing the accuracy and Kappa (0.822 &0.516) for the above models; The KNN model is selected to predict the BOOK_table variable as it has a higher Kappa. Both, KNN and DT, have good performances on training and testing data, this model can be used for application on new data sets. But KNN has a relatively superior metric, therefore it is a better model to use for new data sets. Similarly, in case for Online_order KNN is a better model as it has a higher Kappa (0.313 &0.557) for the training and test data, however, even though the metrics are relatively high both variants are not a good choice for deployment as they have low accuracy and kappa.

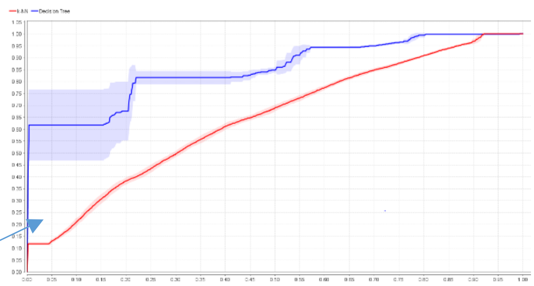
Extension:

The processes for decision trees and KNN for predicting book table and online order decisions were put through cross validation and honest tested using the ZOMATO.TEST data set. The results have been tabulated below. The metrics used are accuracy, Kappa, Tue positive rate and false positive rate. When comparing the honest testing and training data results, we have the same final result as hold out validation for the online_order label, which is that the KNN is a better model. However, these processes offer extremely low accuracy. Which may be a problem for future datasets. Furthermore for the book_table label the honest testing results show that decision tree is a better alternative than KNN. This means that there is a possibility that Decision tree may work better on unseen and new data than KNN even if it fails to perform better on the training data. The true positive rate for decision tree for the book table models see a drop. This means that the proportion of positive values correctly determined in relation to the total actual positive values drops when the model is applied to a new data set.

Label	*Training Data*				*Testing Data*			
Model type	*Accuracy*	*Kappa*	*TPR*	*FPR*	*Accuracy*	*Kappa*	*TPR*	*FPR*
Book_table;								
Decision Tree	*91.61*	*0.832*	*0.91*	*0.0825*	*90.99*	*0.692*	*0.63*	*0.15*
KNN	*93.90*	*0.878*	*0.90*	*0.0185*	*85*	*0.516*	*0.50*	*0.049*
Online_order;								
Decision Tree	*62*	*0.002*	*0.628*	*0.171*	*64*	*0.003*	*0.64*	*0*
KNN	*69*	*0.324*	*0.73*	*0.39*	*69.53*	*0.324*	*0.73*	*0.39*



*The blue curve represents the decision tree's ROC curve and the red curve represents the KNN's ROC. The shapes of the ROC suggest that Decision tree is a better alternative than KNN for building a predictive model; Book_table(label). *However the ROCS for online_order suggest that KNN is a better alternative for building a predictive model.



Deployment in RapidMiner

Aim

To explain how to execute the developed process(es), either to replicate the results or to apply it to new data.

Expectation

The deployment of the selected models to process new data requires the execution of the following steps:

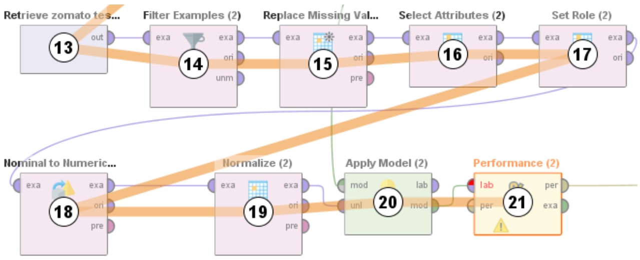
1.) The process of the desired model is accesses through the repository.(for ex: Process named: ‘A1 old book table KNN GO cross validation and honest testing’)

2.) The new data is read using ‘read CSV operator’ and using the parameter to reach the destination where the csv is stored on the system and is stored using the ‘’store’ operator, the destination for the store operator is changed to the file where the process is stored. If the data already exists on the RapidMiner;s local repository then the ‘retrieve’ operator is used to directly access the desired data.

3.) Then the data is put through basic preparation that it requires to match the dimensions of the data that was used to prepare the model, to avoid errors and inexecution. For example, to filter out missing values use filter examples, and add an entry that describes the desired constraint. Some of such operators are highlighted referring to the ILLUSTRA.H (like: rate is not missing (14), to replace missing values(15), ‘replace missing values’ is used, or to convert the nominal data to numerical(18) for KNN,

4.) The ‘set role’(17) operator is used and the parameter of this operator is changed to label and the relevant attribute that is to be predicted is selected, for instance: book_table.

5.) The ‘apply model’(20) operator is used, and the previous operator is connected to ‘apply model’ through the ‘unl’ port and the prepared model is input to the apply model through the ‘mod’ port. **ILLUSTRA.H:**

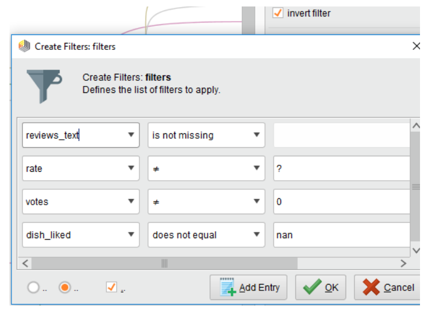


6.) The ‘apply model’ operator is followed by the ‘performance’ (21) operator.

7.) The ‘performance’ operator connected to the output port and the process is executed for results.

The interpretation of the results is done through the performance tab. The accuracy and kappa tell how fit the model is. And comparing these values to the performance measures of the trained model will tell if the model was overfitted for the train data or not. Accuracy, is based on the percentage of predictions done correctly, kappa gives an inference about the same but also takes into consideration the uneven size of the categories. SOLUTION B:

The strategy for new restaurants for booking_table and online_order is predicted using processes that factor in the attributes other than the customer feedback restaurants such as rate, vote, average_cost and review_text. The entries that do not have these values are included in the data set and are processed (using filter examples operator and then using the invert_selection parameter). The processes for this solution are under the name ‘A1 NEW restaurants book_table’ and ‘A1 NEW restaurants Online_order’.

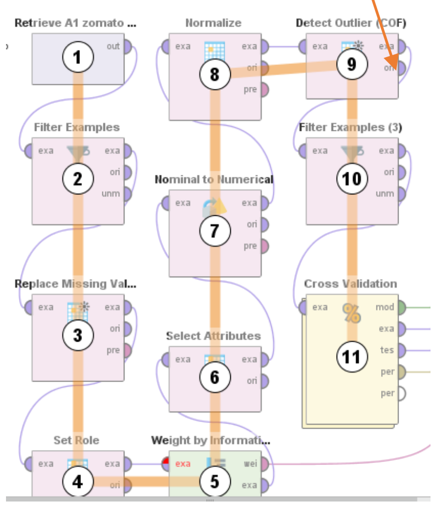


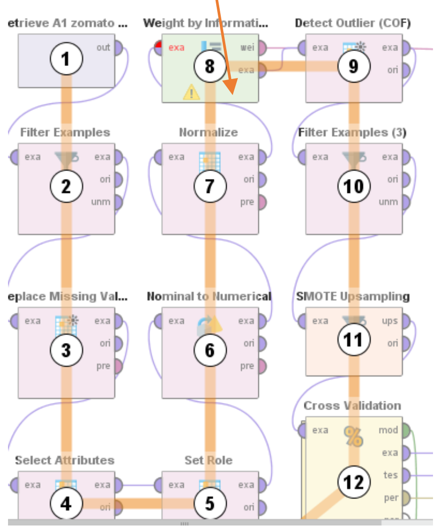
Extension: SOLUTION CNOTE:The strategy for established restaurants for booking table and online ordering can be accessed through the process files: ‘A1 old book table KNN GO cross validation’ and honest testing and ‘A1 old online order KNN GO cross validation and honest testing’

The processes can be accesped to the zip.file and can be used for new data. The processes have been tested and compared to other alternatives for quality assurance. These model will allow the management to tackle the business problem of providing better quality consultancy to its clients. These models take into consideration the various dimensions of the restaurants and allow the user to predict what should be the strategy of the user

Based on those considerations. The existing trends and booking_table and online_order strategies of the existing restaurants in the Bangalore region are used as the basis to make these recommendations.

These accurate recommendations will allow BFA to provide better services to its clients and ensure retention of clients in the short run and increase in revenue and clientele in the long run. Once the process is executed the cross valudation(example set) tab can be used to access the results; whether a certain restaurants should have booking table and online order as a strategy.





Further Research and Extensions in RM (one page)

Aim
To demonstrate your ability to seek new ways of solving analytic problems.

Expectation
On research, it was discovered how a decision tree and KNN operate and produce the desired results. While, decision trees work on entropy and information gain, which is estimated by various metrics such as GINI coefficient.

New analytic methods should be used (2-3) for your data analysis, modelling or visualisation - beyond what was covered in class (lectures, labs or demos up to the deadline).
You can use RapidMiner, but also R, Python, or some other tool (for this section only).

Extension
Vijay Kotu, Bala Deshpande, (2019) Data Science: Concept and Practices
Jacob Cybluski,(n.d.) 'Ironfrown', Youtube
Tutorials and help, Rapidminer.com