Executive Summary

Business Problem
LP3: Recently, Banglore Food Assist has seen a fall in quality of service. It has been identified that this fall has been as a result of poor information quality related to customer feedback for similar restaurants. This information further forms the basis for inferences and insights related to other characteristics of the restaurants such as the types of meals offered at the establishment and the estimation of the ratings for the restaurants. Especially for restaurants that do not have a rating profile on the Zomato database. For such restaurants to be evaluated in comparison with other restaurants, BFA must estimate their ratings.

Solution to Business Problem
LP4: The recommended solutions revolve around deployment of analytical estimation models that will allow BFA to increase its effectiveness in estimating ratings for restaurants that do not have a rating profile on the Zomato database. The meticulously selected estimation model, of Gradient Boosted Tree (refer to page 7), will estimate the ratings of such restaurants and make them comparable with other restaurants. These ratings can be the basis of making consultancy choices for BFA as well. When approached by a prospective client, BFA can use this metric to show where the restaurant stands in comparison to the other businesses and how can it improve.

Extension
LP3: The analytical solution will be sufficient to provide a competitive advantage through improved information resources. BFA can offer better quality services to individual and groups of customers. This solution will allow BFA to derive insights about estimation of rating, association between types of meals offered and different types of restaurants grouped on the basis of customer opinions.

LP4: The implementation of this solution will support BFA business decisions in terms of expansion and maintaining its current client base. The initial results from exploration give insight into the ratings of the restaurants (refer to page 9). BFA's business decisions will involve full fledge deployment of these solutions on larger datasets. This solution will allow BFA to interact with clients that lack an extensive online presence, significantly increasing the prospective client's cohort. Increased investments will be required to set up a team of analysts to execute this extensive plan. It is recommended that this plan is first rolled out gradually; a beta run on a small group of restaurants and eventually executed on a large scale when the analytical teams and management teams have gained confidence in the performance measures of the analytical models. The management must show keen interest in the monetary benefits that can be possible reaped from such analytical practices. It not only allows BFA to gain a competitive advantage in the short run, an opportunity for increased market share in the medium run but also an elaborate platform for market research. Increased returns in the form of profits and revenues can be expected in the medium run. Careful roll out of this this business solution can potentially benefit BFA Not just in monetary terms but also in terms of market standings and market share.

References

- (Dietterich, T.G., Ensemble Learning, 2002)
- (Breslow, N. E. & Clayton, D. G., Approximate Inference in Generalized Linear Mixed Models)
- (Kotu, V., Deshpande, B.,Data Science: Concepts and Practice , 2019)

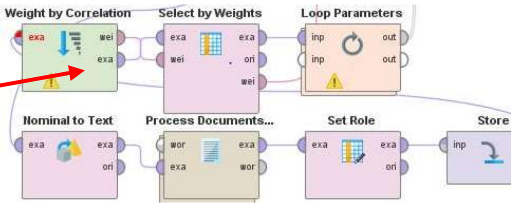
Data exploration and relationships - Clustering in RapidMiner

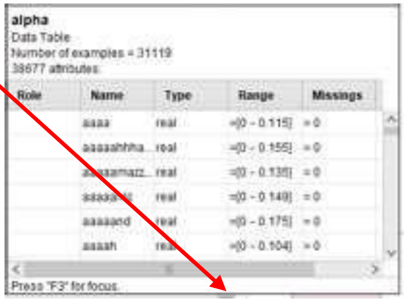
Aim

To demonstrate your understanding of text processing and interpretation.

Expectation

Local random Seed: 1996.



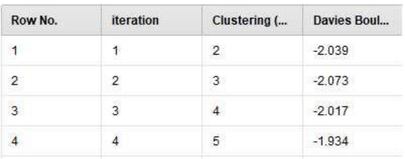


alpha
Data Table
Number of examples = 31119
38677 attributes

Role	Name	Type	Range	Missings
	aaaa	real	=[0 - 0.115]	= 0
	aaaaahhha...	real	=[0 - 0.155]	= 0
	aaaaamazz...	real	=[0 - 0.135]	= 0
	[illegible]	real	=[0 - 0.149]	= 0
	aaaaaand	real	=[0 - 0.175]	= 0
	aaaah	real	=[0 - 0.104]	= 0

Press "F3" for focus.

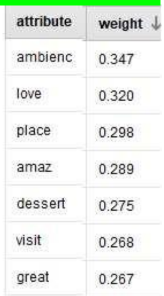
SOLUTION A The nominal attribute '*review_text*' was converted to text using the *nominal to text* operator_and was then parsed using the *process documents from data* operator This process was saved in a local data store names alpha using the store operator. Then the dimensionality was reduced down to 50 based on correlation using the *select by weights* operator. The attributes with seven highest weights have been illustrated.



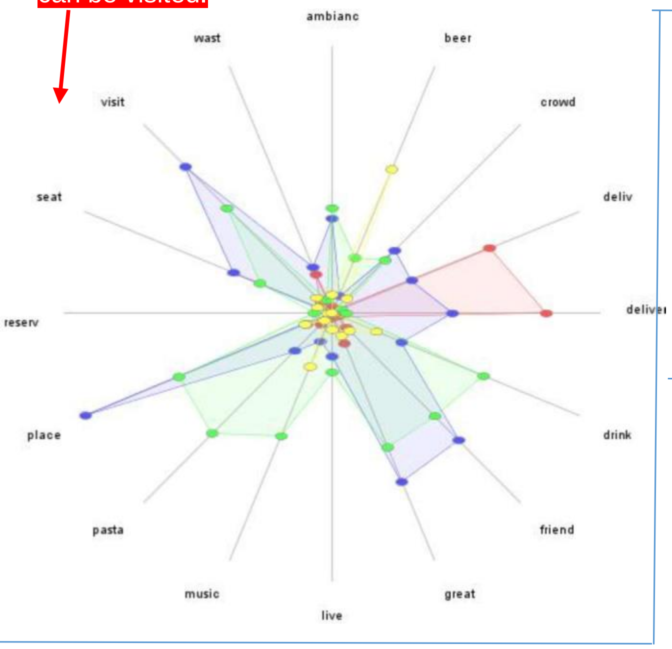
Row No.	iteration	Clustering (...	Davies Boul...
1	1	2	-2.039
2	2	3	-2.073
3	3	4	-2.017
4	4	5	-1.934

Then the ***k*** for clustering was optimized using the loop parameter operator and the resulting k value of 4 (based on davies bouldin estimate with **k** range min 2-max 22 in loop parameter) was used to cluster the data. The following results were obtained:

The available dataset allows analysis of 31,119 data entries (due to missing values of rating and reviews). The data is clustered into four clusters. On overall analysis of clusters with all the *word*-attributes, it was observed that the clusters can be distinguished on the basis of the following attributes; ambiance, beer, crowd, delivering, drink, friend, great, live, music, pasta, place reserve, visit and waste. On segmentation analysis the following can be said about the four clusters; CLUSTER0: Based on the cust feedback this cluster comprises of fine dining restaurants that offer seatings with an ambiance and music, it is a place is visited and offer drinks. CLUSTER1: This cluster represents BARS&CLUBS that offer beverages like beer and have a a lot of mentions of music. CLUSTER2:This is the biggest cluster and it represents CONVENTIONAL EATERIES that offer a mix services and dishes. These restaurants are describes as places and are visited. CLUSTER3: This cluster represents food joints that DELIVER FOOD to its customers and do not offer seating or can be visited.

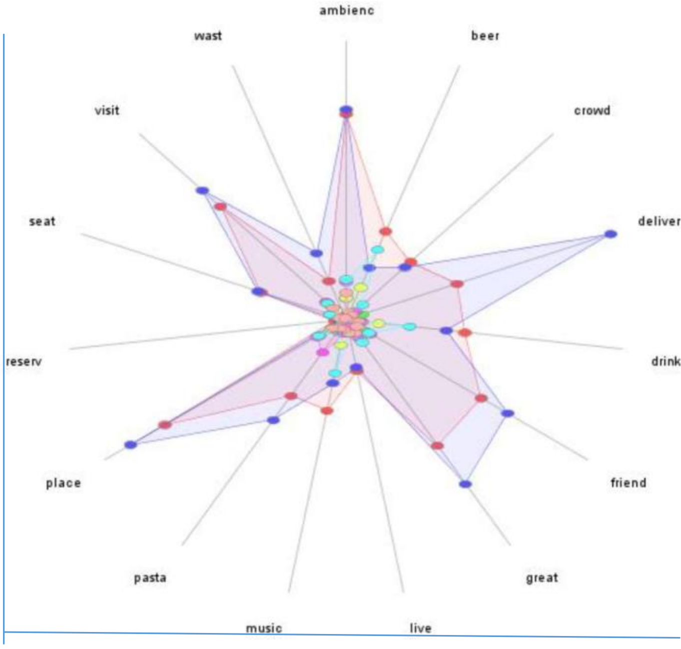


attribute	weight ↓
ambienc	0.347
love	0.320
place	0.298
amaz	0.289
dessert	0.275
visit	0.268
great	0.267



Extension: SOLUTION B The relation between review_text and meal_type can be seen through the chart below, the seven meal types can be distinguished with corresponding *term-attributes*. The delivery meal type has maximum mentions of the term c friend, place. Drinks and nightlife along with dine-out meal type restaurants h large mentions of beer, drink and music. The café and the desserts meal typ relatively less mentions of most terms and is therefore can be distinguished u the selected terms.

● Dine-out ● Delivery ● Desserts ● Pubs and bars ● Cafes ● Drinks & nightlife ● Buffet



ANAMOLY DETECTION:

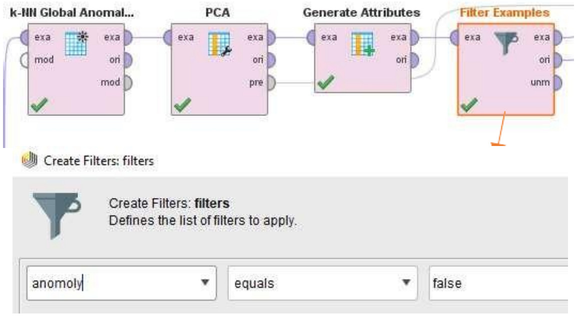
The threshold for anomaly detection was set as 0.6. Any data entry with an outlier greater than 0.6 is regarded as an anomaly.

Data exploration and cleanup - Anomalies in RapidMiner

Aim

To demonstrate your understanding of anomaly detection in a mix of text and structured data.

Expectation



Create Filters: filters

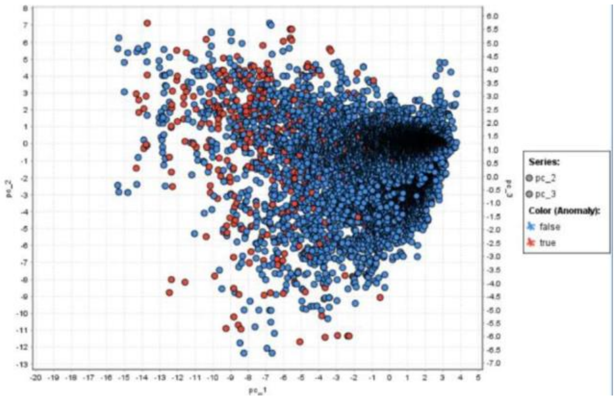
Create Filters: **filters**
Defines the list of filters to apply.

anomoly | equals | false

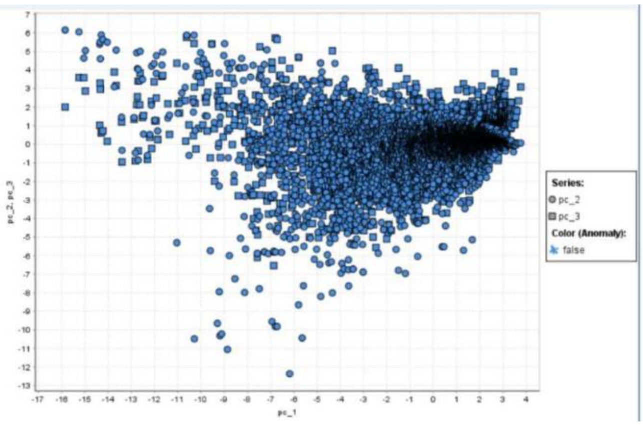
The KNN global anomaly operator is used to identify outliers in the data. The threshold is set in the generate attributes operator as ANOMALY. An entry is classified as an anomaly if the outlier value I greater than 0.6. The anomalies are then eliminated using the filter examples operator.

Extension

The adjacent scatter chart shows the relation between principal component 1 and principal component 2. The anomalies have been highlighted on red. This analysis involves text attributes and the structured attributes.



The adjacent scatter plot shows the relation between principal component 1 and principal component 2. The anomalies have been highlighted in red.



The adjacent scatter plot shows the relation between PC2, PC3 and PC1 after the removal of anomalies.

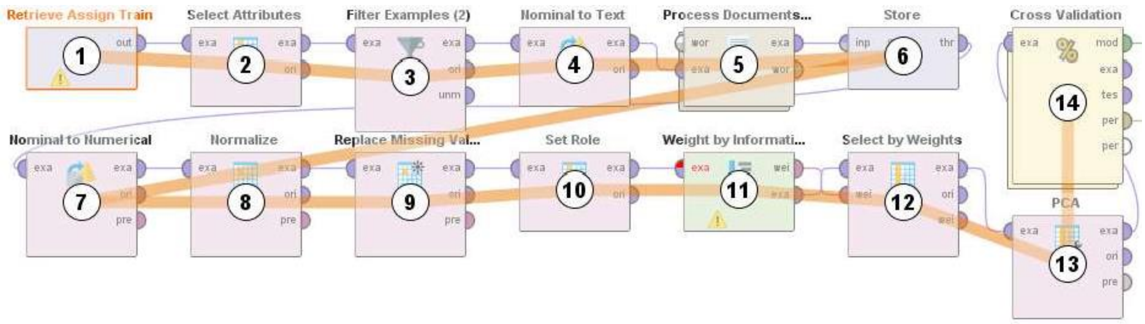
Create a Model(s) in RapidMiner (two pages / page 5)

Aim
To explain details of developed estimation models and selected methods for data preparation.
The purpose of this section is to explain your analytic processes.
Do not include here any process runs or their results!

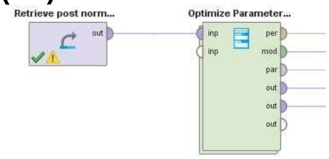
Expectation

Regression

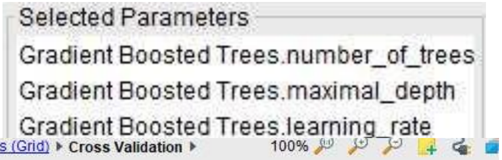
The data was prepared for the regression model by selecting the relevant attributes (using select attributes);(2) all attributes were selected except dish_like and menu_item (these attributes had excessive missing values). (3)The entries with missing rate and review_text were filtered out of the dataset. (4) &(5) the reviews_text attribute was converted to text from polynomial and then the text was parsed and evaluated based on TF-ID. The process was then stored so that it can be replicated and used for other processes. The remaining numerical attributes were converted to numerical and then normalized so that the regression model can process them. (9) The remaining missing values replaced with the average values. So that the regression model can run. (10) The RATE attribute was set as the label. (11) & (12) The attributes were weighted as per correlation and then selected to reduce dimensionality (top 500). (13) PCA was run to further reduce the dimensionality. (14) The regression model was nested in the cross validation (10 folds) operator. The model was applied using apply model and performance operator was used to measure the performance of the model using the metrics (root mean squared error, absolute error and correlation)

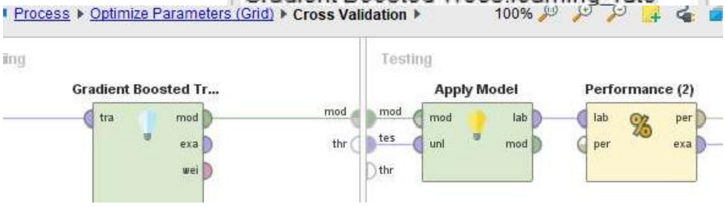


(M2) GBTs

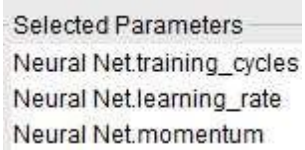


The data was prepared and executed for the gradient boosted tree model. The processed data stored in ‘post normalization’ was retrieved and connected to GRID parameter operator (the parameters put under optimization are; number of trees, maximum depth and learning rate). The cross validation parameter is nested inside the optimize parameter operator with the gradient boosted tree, apply model and performance operators further nested inside cross validation. The metrics selected for performance measurement are correlation, absolute error and root mean squared.

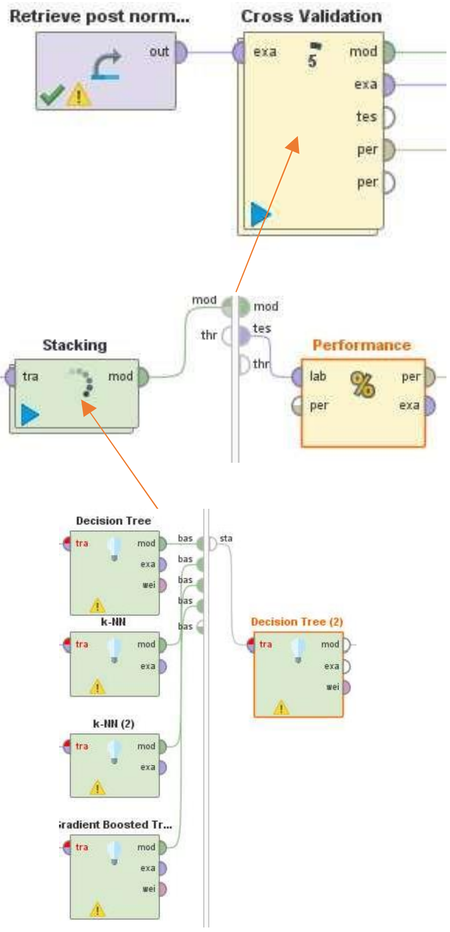

Selected Parameters
Gradient Boosted Trees.number_of_trees
Gradient Boosted Trees.maximal_depth
Gradient Boosted Trees.learning_rate



(M3) neural nets The neural net model is executed in the same stencil, except the parameters put under optimization are training_cycles, learning_rate and net.momentum. These are the most important parameters for a neural net model and the grid parameter runs 1331 combinations to evaluate the optimized combination of parameter values.


Selected Parameters
Neural Net.training_cycles
Neural Net.learning_rate
Neural Net.momentum

Create a Model(s) in RapidMiner



Extension

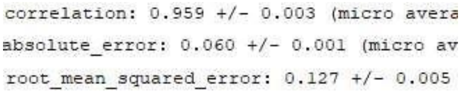
Ensemble

The ‘post normalization’ store was retrieved>the data was cross validated with 10 folds and the ‘stacked’ operator was nested inside the cross-validation operator. Deep learning and linear regression models were nested inside the stacking operator and generalized linear model was placed as the stacking model learner. The performance operator was placed and the performance measures were set as root mean squared error, absolute error and correlation.

Evaluate and Improve the Model(s) in RapidMiner

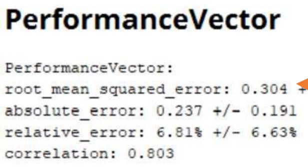
Aim

To report and explain the performance of developed estimation models.
Here report all process runs, optimisation, their results, performance analysis and interpretation.



```
correlation: 0.959 +/- 0.003 (micro avera
absolute_error: 0.060 +/- 0.001 (micro av
root_mean_squared_error: 0.127 +/- 0.005
```

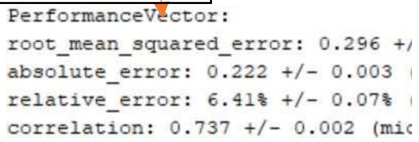
Expectation The following models were run with cross validation and the results have been recorded and compared for model selection:



PerformanceVector

```
PerformanceVector:
root_mean_squared_error: 0.304 +
absolute_error: 0.237 +/- 0.191
relative_error: 6.81% +/- 6.63%
correlation: 0.803
```

MODEL	Correlation	Absolute_error	Root_mean_squared_error
GBTree	0.959	0.060	0.127
Linear Reg.	0.636	7.54	0.337
Neural Net.	0.737	0.222	6.41
Ensemble	0.849	0.1410	0.232

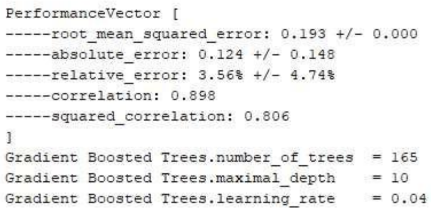


```
PerformanceVector:
root_mean_squared_error: 0.296 +/
absolute_error: 0.222 +/- 0.003 (
relative_error: 6.41% +/- 0.07% (
correlation: 0.737 +/- 0.002 (mic
```

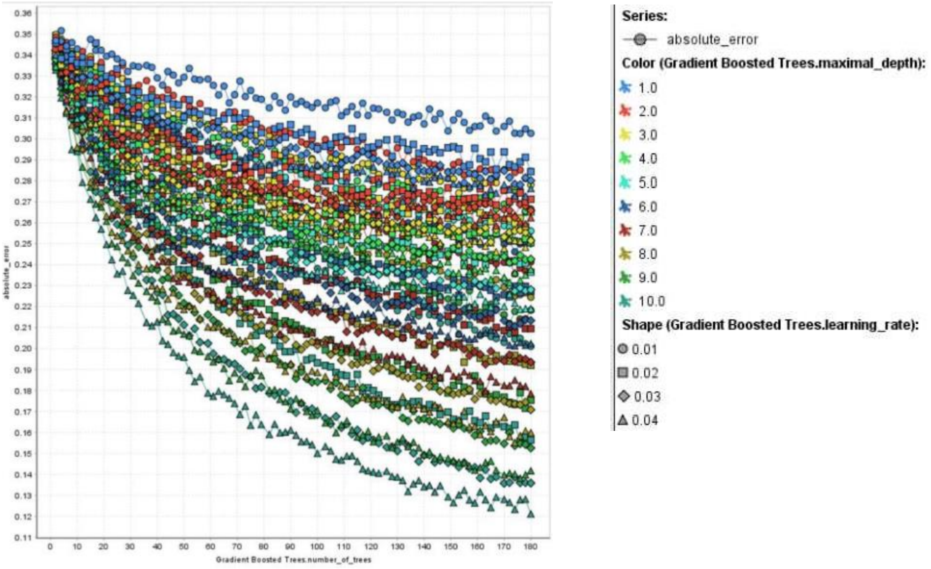
The Gradient Tree Model with parameters, no. of trees: 70, Maximal Depth 30 and learning rate 0.1 has the maximum performance amongst the executed models. With a high correlation of 0.959, a small absolute_error of 0.60 and root_mean_squared_error of 0.127. These results can be retrieved by executing the process 'RATE GBT.RMP'.

Extension

The following illustrations visualize the optimization results for 1) GBT and 2) Neural Networks
The parameters optimized for number of trees maximal depth and learning rate. The resulting optimized values for a minimized absolute error of 0.124 have also been attached. The visualization agrees with the resulting parameter values, with gradient boosted trees.learning rate at 0.04.

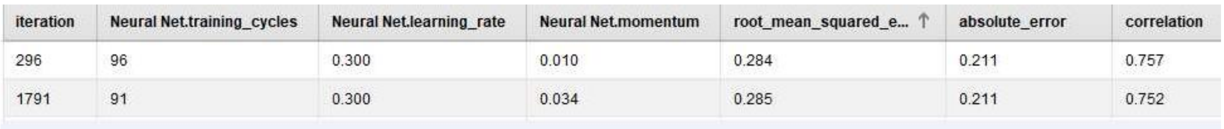


```
PerformanceVector [
-----root_mean_squared_error: 0.193 +/- 0.000
-----absolute_error: 0.124 +/- 0.148
-----relative_error: 3.56% +/- 4.74%
-----correlation: 0.898
-----squared_correlation: 0.806
]
Gradient Boosted Trees.number_of_trees   = 165
Gradient Boosted Trees.maximal_depth     = 10
Gradient Boosted Trees.learning_rate     = 0.04
```

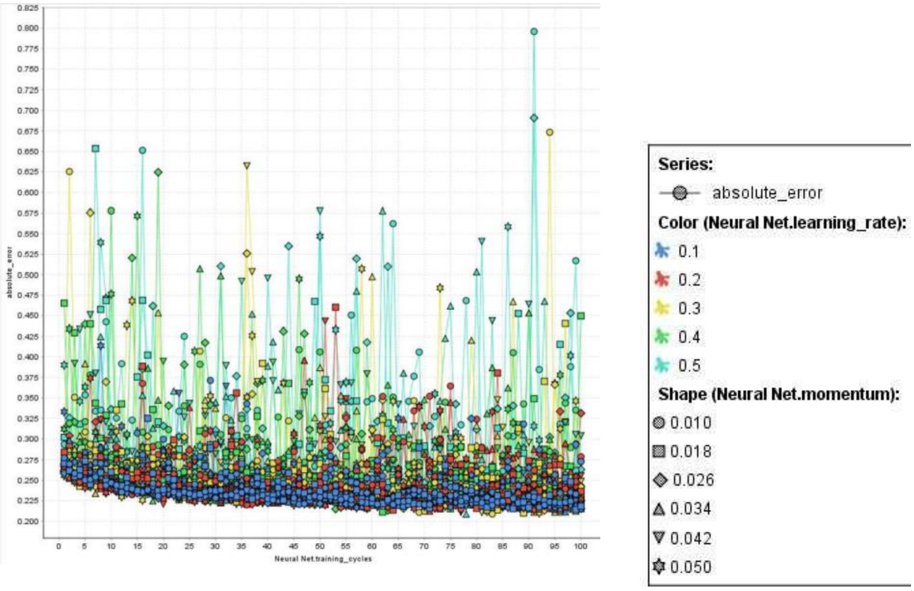


Evaluate and Improve the Model(s) in RapidMiner

The parameters optimized for neural net are learning rate, net momentum and training cycles. The optimized parameter values for a minimized absolute error of 0.211 have also been attached. The visualization shows that the optimized value for learning rate and momentum are 0.300 and 0.034.



iteration	Neural Net.training_cycles	Neural Net.learning_rate	Neural Net.momentum	root_mean_squared_e... ↑	absolute_error	correlation
296	96	0.300	0.010	0.284	0.211	0.757
1791	91	0.300	0.034	0.285	0.211	0.752



Provide an Integrated Solution in RapidMiner

Aim

To report the final results with justification.

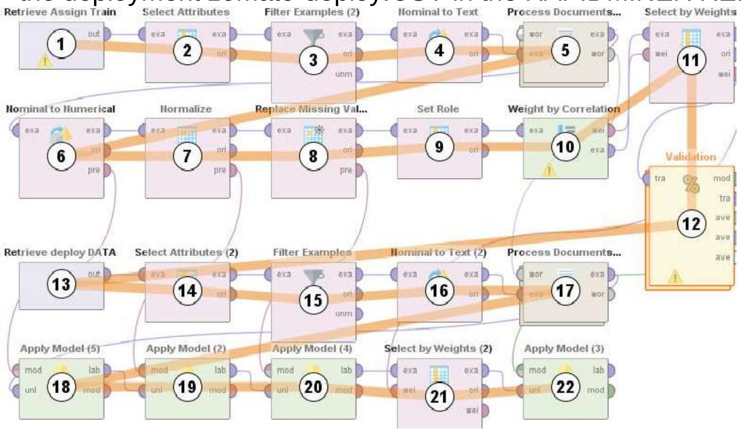
To explain how to execute the developed process(es), either to replicate the results or to apply it to new data.

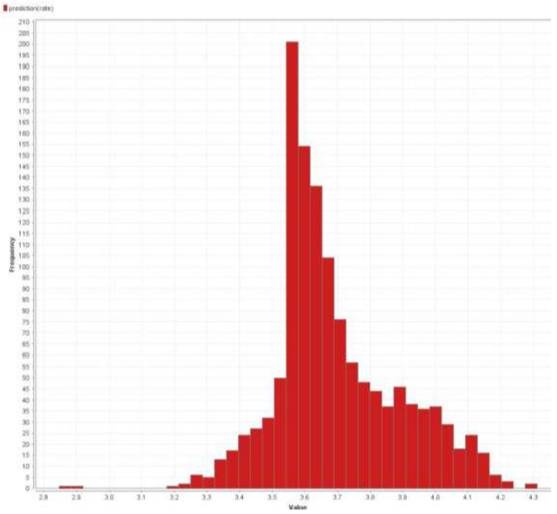
Also learn to create analytic processes that can be deployed to their operating environment, with all the relevant pre-processing, clustering, predictive and transformation models.

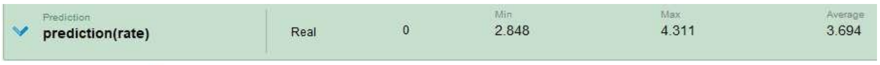
Expectation

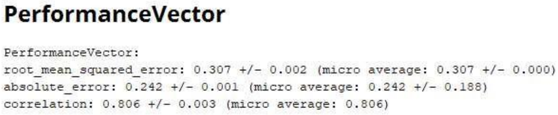
SOLUTION C: Reference: DEPLOYMENT EXPECTATION.RMP

***Through the above analysis, *the 'gradient boosted tree' had the highest performance, (correlation 0.95). hence, the model has been chosen for deployment.** EXECUTION of process: The intended user must follow the following steps;1> place the deployment zomato-deploy.CSV in the RAPIDMINER REPOSITORY.2> Then access the rapidminer software and use the repository window to run the process named "deployment expectation.RMP". The user will see a process with the first 12 steps in the adjacent illustration. 3> use the operator window to access the read CSV operator.4> In the parameter section access the repository and select the zomato-deploy.CSV. Followed by a store operator, in the store operator's parameters select C:DRIVE/DOCS/rapidminer repository as the source of the data and name the dataset as 'DEPLOY DATA'. Now this dataset can be retrieved easily by dragging and dropping the dataset from the repository window on the left. 5> We now begin with prepping the data for deployment. Select the operators 2, 3, 4 & 5 and copy, paste them beside the retrieve ass2train operator and connect it. These operators prep the data by selecting a subset of attributes, filtering the data entries that have missing values, converts the reviews_text to a text attribute, and then converts the text data into TF-IDs scores. 6> Drag and drop THE APPLY MODEL operator and join a connection from the process documents that goes through all the apply model operators, use the uni as the input port and lab as the output port. And connect these apply model operators with nominal to numerical operator, replace missing values, normalize use the 'pre' output and use the 'mod' port for the input in the apply model operators. 7> for the last apply model we join it with the validation operator. The last apply model is joined with the final connection. On execution the results for the process give us the SOLUTION C: The user can read the table in the RESULTS(//) tab and see the prediction(rate) column for the predicted values of rating for respective reviews. The user can see the overall stats from the stats tab in the results section and and see results like the minimum predicted rating of 2.645 and a maximum of 4.311. The user can visualize the predicted rating column and make inferences about the distribution of the predicted rating variable by selecting histogram in the simple chart section.






Prediction prediction(rate) Real 0 Min 2.848 Max 4.311 Average 3.694


PerformanceVector

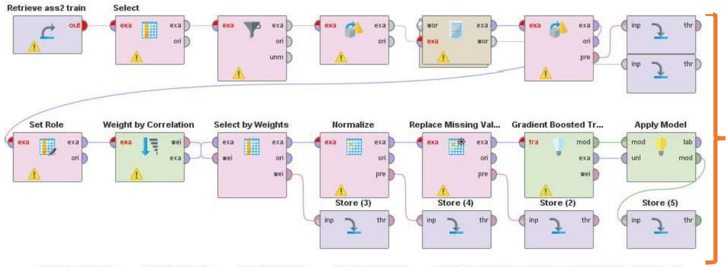
```
PerformanceVector:
root_mean_squared_error: 0.307 +/- 0.002 (micro average: 0.307 +/- 0.000)
absolute_error: 0.242 +/- 0.001 (micro average: 0.242 +/- 0.188)
correlation: 0.806 +/- 0.003 (micro average: 0.806)
```

Extension

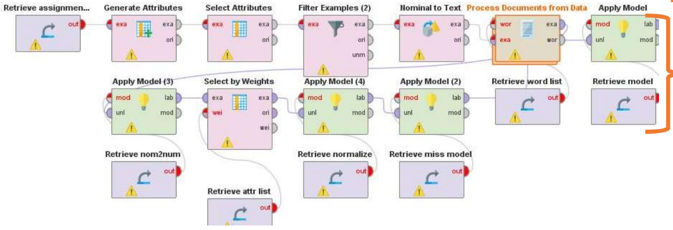
Reference: DEPLOYMENT EXTENSION.RMP

DEPLOYMENT PRE PROCSSES

The various pre-processes have been stored into different stores, so that the results can be retrieved and used in the deployment process. **Gradient Boosted Tree** is the model with the highest performance.



The stored pre-processes are then retrieved and used for data prep for the deployment data set. And are used as input for various operators in the form of word list, weights for words, application of the created model.



Retrieve assignmen...
Generate Attributes
Select Attributes
Filter Examples (2)
Nominal to Text
Process Documents from Data
Apply Model
Apply Model (3)
Select by Weights
Apply Model (4)
Apply Model (2)
Retrieve word list
Retrieve model
Retrieve nom2num
Retrieve normalize
Retrieve miss model
Retrieve attr list