

311 Service Request

Group Members:-

1. Deepak Kadam (F16111035)
2. Arisha Chamadia (F16111049)
3. Swati Patra (F16111065)

Guided By:-

Prof R.A.Khan

Dataset Description: Data on 311 service requests in Pittsburgh. Data is related to every call made on 311. 311 is a non-emergency phone number that people can call in many cities to find information about services, make complaints, or report problems like graffiti or road damage. Even in cities where a different phone number is used, 311 is the generally recognized moniker for non-emergency phone systems.

<u>Attributes</u>	<u>Type</u>
REQUEST_ID	Numerical
CREATED_ON	Numerical
REQUEST_TYPE	Categorical
REQUEST_ORIGIN	Categorical
STATUS	Binary
DEPARTMENT	Categorical
NEIGHBORHOOD	Categorical
COUNCIL_DISTRICT	Numerical
WARD	Numerical
PUBLIC_WORKS_DIVISION	Numerical
PLI_DIVISION	Numerical
POLICE_ZONE	Numerical
FIRE_ZONE	Numerical

X	Numerical
Y	Numerical
GEO_ACCURACY	Categorical

Import Dataset: Getting the dataset into Rapid Miner for preprocessing.

The screenshot displays the Rapid Miner Studio interface with a workflow designed for data preprocessing. The workflow consists of the following steps:

- Read Excel:** The initial step to load the dataset into the process.
- Replace Missing Val...:** An operator used to handle missing values in the data.
- Remove Duplicates:** An operator used to eliminate duplicate entries from the dataset.
- Store:** The final step to save the processed data.

The interface includes several panels:

- Repository:** Shows the local repository with various operators like crossValidation2, DecisionTree, k-nn, naive, Preprocessing, rule_induction, and updated.
- Operators:** A search bar and a list of operators categorized by function (Data Access, Blending, Cleansing, Modeling, Scoring, Validation, Utility).
- Parameters:** A panel for configuring the selected operator, showing settings like logverbosity, logfile, resultfile, random seed, send mail, and encoding.
- Help:** A panel providing information about the selected operator, including a synopsis and description.

The workflow is currently in the 'Design' view, and the 'Process' panel shows the sequence of operators connected by arrows. The 'Data Editor' panel at the bottom is also visible.

Cleaning Process: The dataset is not perfect i.e it is not in the form we need. The classification algorithms work on the basis of Arithmetic and Statistical rules. Therefore numeric entries are to be fed into the dataset and not other types like binary and categorical. For this we have to convert them into numerical values.

Furthermore, some attributes have missing values so we need to fill these missing values, we need to remove any duplicate entries if present using statistical methodologies which ever promises to yield better results in the future.

Views: Design Results Turbo Prep Auto Model Deployments

Find data, operators...etc All Studio

Result History ExampleSet (/311Service/Process/updated)

Open in Turbo Prep Auto Model Filter (146 / 146 examples): all

Row No.	STATUS	REQUEST_ID	CREATED_ON	REQUEST_T...	REQUEST_O...	DEPARTMENT	NEIGHBORH...	COUNCIL_DI...	WARD	PUBLIC_WO...	PLI_DIVISIC
1	1	168714	2017-07-10T...	Fire Safety Sy...	Control Panel	Permits, Lice...	Carrick	4	29	3	29
2	1	251339	2018-05-22T...	Illegal Dumpi...	Website	DOMI - Permits	Garfield	4	10	2	10
3	1	168709	2017-07-10T...	Potholes	Call Center	DPW - Street ...	Polish Hill	7	6	6	6
4	0	168710	2017-07-10T...	Abandoned V...	Call Center	Police - AVU	Marshall-Sha...	1	27	1	27
5	0	189977	2017-09-27T...	Manhole Cov...	Call Center	DOMI - Permits	Central Busin...	6	2	6	2
6	1	214948	2018-02-01T...	Crossing Gu...	Website	School Guards	Perry North	1	26	1	26
7	1	284780	2018-08-30T...	Board Up (PL...	Control Panel	DPW - Constr...	Perry South	6	25	1	25
8	1	214949	2018-02-01T...	Potholes	Website	DPW - Street ...	Mt. Oliver	3	16	3	16
9	1	284767	2018-08-30T...	Building Main...	Control Panel	Permits, Lice...	Sheraden	2	20	5	20
10	0	195917	2017-10-27T...	Manhole Cov...	Call Center	DOMI - Permits	Bluff	6	1	3	1
11	0	200171	2017-11-22T...	Water Main B...	Call Center	Pittsburgh W...	Banksville	2	20	5	20
12	1	6973	2015-06-01T...	Sinkhole	Call Center	DPW - Street ...	Beltzhoover	3	18	5	18
13	1	193043	2017-10-12T...	Guide Rail	Call Center	DPW - Constr...	East Hills	4	13	2	13
14	1	200800	2017-11-29T...	Graffiti	Control Panel	Police - Zone...	South Side FL...	3	16	3	16

ExampleSet (146 examples, 1 special attribute, 15 regular attributes)

Indexing Academy content

Splitting the Result: Now after data pre-processing we need to split the data into train and test set. This data splitting is necessary for training and testing of the model. The train set will be used to train the model and the test set to evaluate the model. the training data set should be relatively larger than the test set as it will define the model performance.

<new process> - RapidMiner Studio Free 9.3.001 @ Ashley

File Edit Process View Connections Settings Extensions Help

Views: Design Results Turbo Prep Auto Model

Find data, operators...etc All Studio

Repository

- Import Data
- Samples
- DB (Legacy)
- Local Repository (10/17/19 3:39 PM - 174 KB)
 - Connections (10/17/19 3:39 PM - 174 KB)
 - data (10/17/19 3:34 PM - 174 KB)
 - processes (10/17/19 3:34 PM - 174 KB)
 - Cleaning Process (10/17/19 3:34 PM - 174 KB)
 - output (10/17/19 3:34 PM - 174 KB)

Operators

Search for Operators

- Data Access (53)
- Blending (79)
 - Attributes (46)
 - Examples (11)
 - Table (11)
 - Values (11)
- Cleansing (26)
- Modeling (156)
- Scoring (12)
- Validation (29)

[Get more operators from the Marketplace](#)

Process

Process

100%

Retrieve output

Split Data

Parameters

Process

logverbosity: init

logfile:

resultfile:

random seed: 2001

send mail: never

encoding: SYSTEM

[Hide advanced parameters](#)

[Change compatibility \(9.3.001\)](#)

Help

Process

RapidMiner Studio Core

Synopsis

The root operator which is the outer most operator of every process.

Description

Each process must contain exactly one operator of this class, and it must be the root operator of the

Leverage the Wisdom of Crowds to get operator recommendations based on your process design!

Activate Wisdom of Crowds

Type here to search

15:58 17-10-2019

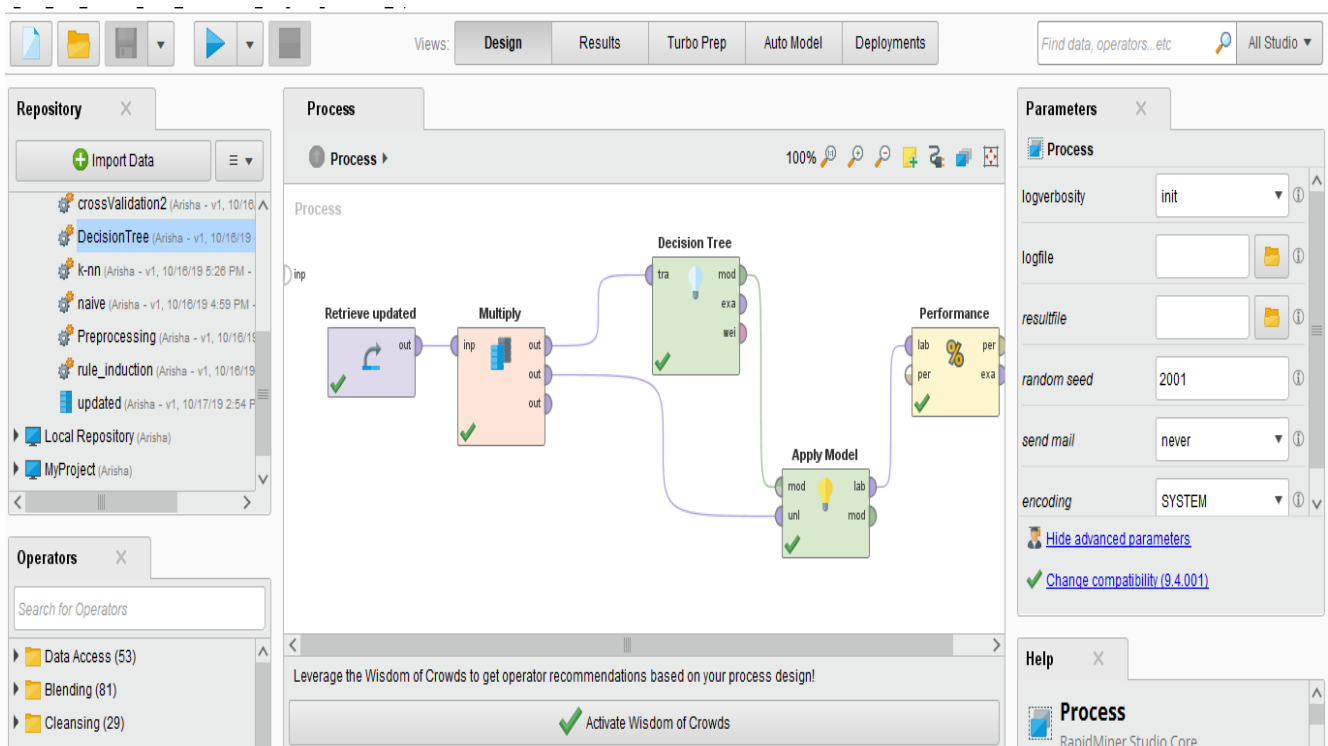
Training the Model: Training of the model here is done by using 3 classification algorithms :

1. Decision Tree
2. K-NN
3. Naïve Bayes

1. Decision Tree:

Decision tree is the most powerful and popular tool for classification and prediction. A Decision Tree is a flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.

A tree can be learned by splitting the source set into subsets based on an attribute value test. This process is repeated on each derived subset in a recursive manner called *recursive partitioning*. The recursion is completed when the subset at a node all has the same value of the target variable, or when splitting no longer adds value to the predictions. The construction of decision tree classifier does not require any domain knowledge or parameter setting, and therefore is appropriate for exploratory knowledge discovery. Decision trees can handle high dimensional data. In general decision tree classifier has good accuracy. Decision tree induction is a typical inductive approach to learn knowledge on classification.



Decision Tree Accuracy/ Confusion Matrix:

eVector (Performance)ExampleSet (/311Service/Process/updated)

Table View

Plot View

accuracy: 70.55%

	true 1	true 0	class precision
pred. 1	103	43	70.55%
pred. 0	0	0	0.00%
class recall	100.00%	0.00%	

2. K Nearest Neighbour

K-Nearest Neighbours is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection.

It is widely disposable in real-life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data

The screenshot displays the RapidMiner Studio Educational 9.4.001 interface. The main workspace shows a process design with the following operators: Retrieve updated, Multiply, k-NN (2), Apply Model (2), Cross Validation, and Performance (2). The process flow starts with 'Retrieve updated', followed by 'Multiply', then 'k-NN (2)'. The output of 'k-NN (2)' is connected to 'Apply Model (2)', which then feeds into 'Cross Validation'. Finally, the output of 'Cross Validation' is connected to 'Performance (2)'. The 'Parameters' panel on the right shows settings for the 'Process' operator, including 'logverbosity' (init), 'logfile', 'resultfile', 'random seed' (2001), 'send mail' (never), and 'encoding' (SYSTEM). The 'Operators' panel on the left lists various operators available in the repository, such as 'crossValidation2', 'DecisionTree', 'k-nn', 'naive', 'Preprocessing', 'rule_induction', and 'updated'. The 'Help' panel at the bottom right provides a synopsis of the 'Process' operator, stating it is the root operator of every process.

KNN Model Accuracy/ Confusion Matrix:

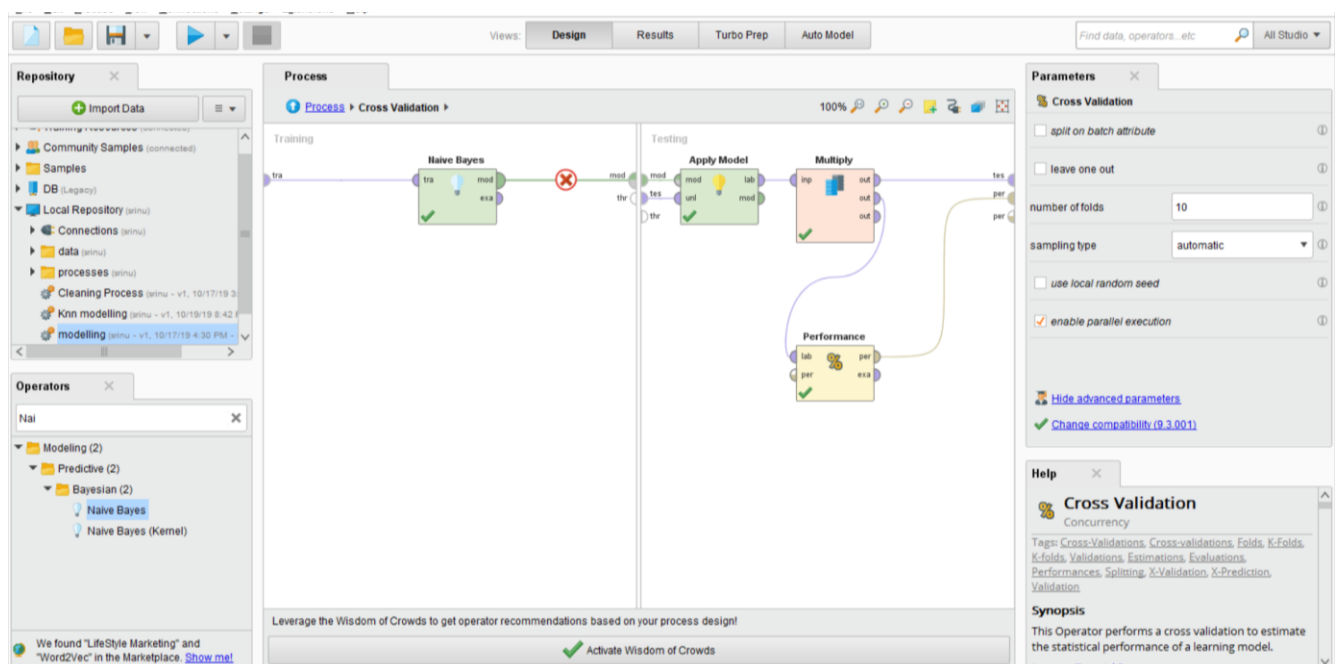
Criterion	Table View Plot View		
accuracy	accuracy: 63.67% +/- 7.37% (micro average: 63.70%)		
precision			
recall			
AUC (optimistic)			
AUC			
AUC (pessimistic)			
	true 1	true 0	class precision
pred. 1	85	35	70.83%
pred. 0	18	8	30.77%
class recall	82.52%	18.60%	

3. Naïve Bayes:

In machine learning, naïve Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong independence assumptions between the features.

Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

Naïve Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem. Maximum-likelihood training can be done by evaluating a closed-form expression,[4]:718 which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers.



Naive Bayes Accuracy / Confusion Matrix:

Criterion: **accuracy**

☒ Table View ☐ Plot View

accuracy: 75.00%

	true 1	true 0	class precision
pred. 1	26	6	81.25%
pred. 0	5	7	58.33%
class recall	83.87%	53.85%	

Cross Validation ROC Comparison :

//311Service/Process/crossValidation2 - RapidMiner Studio Educational 9.4.001 @ LAPTOP-ITULT9B

File Edit Process View Connections Settings Extensions Help

Views: Design Results Turbo Prep Auto Model Deployments

Find data, operators, etc. All Studio

Repository

- crossValidation2 (Arisha - v1, 10/22/19 4:17 PM - 4 KB)
- crossValidation2 (Arisha - v1, 10/22/19 2:53 PM - 4 KB)
- DecisionTree (Arisha - v1, 10/16/19 4:18 PM - 2 KB)
- k-nn (Arisha - v1, 10/16/19 5:26 PM - 8 KB)
- naive (Arisha - v1, 10/16/19 4:59 PM - 5 KB)
- Preprocessing (Arisha - v1, 10/16/19 3:44 PM - 5 KB)
- rule_induction (Arisha - v1, 10/16/19 5:20 PM - 3 KB)
- updated (Arisha - v1, 10/22/19 2:54 PM - 16 KB)
- Local Repository (Arisha)

Operators

Search for Operators

- Data Access (53)
- Blending (81)
- Cleansing (29)
- Modeling (160)
- Scoring (14)
- Validation (30)
- Utility (85)

[Get more operators from the Marketplace](#)

Process

Process > Compare ROCs

100%

Decision Tree (2)

k-NN

Random Forest

Naive Bayes

Rule Induction

Leverage the Wisdom of Crowds to get operator recommendations based on your process design!

Activate Wisdom of Crowds

Parameters

Compare ROCs

- number of... 10
- split ratio 0.7
- sampling t... stratif...
- ☐ use local / random se
- ☒ use example weight
- [Hide advanced parameters](#)
- [Change compatibility \(9.4.001\)](#)

Help

Compare ROCs

RapidMiner Studio

Tags: Roc, Curves, Compari, Comparisons, Validations, E, Performances, Sensitivity, V

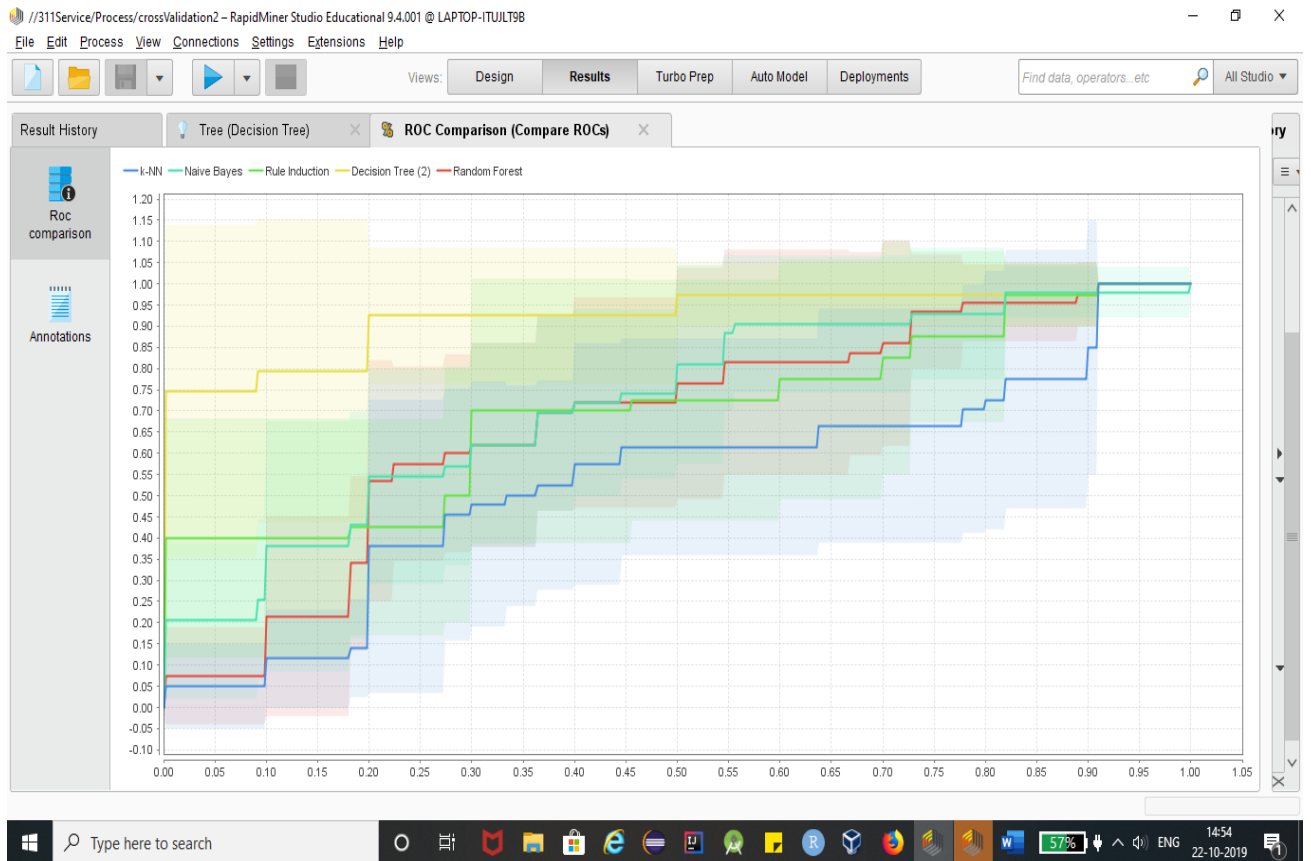
Synopsis

This operator generates R

for the models created by

Windows taskbar: Type here to search, 58%, 14:55, 22-10-2019

Roc Results:



Conclusion: By the given ROC graph and the confusion matrix, we can conclude that Naïvie Bayes yields better results with the accuracy of 75%.