



BizViz User Guide

Predictive Analysis

Release: 2.5

Date: Nov. 9, 2016



Table of Contents

1. About This Guide.....	6
1.1. Document History	6
1.2. Overview	6
1.3. Target Audience	6
2. Introducing BizViz Predictive Analysis Tool	6
2.1. Introduction to the BizViz Predictive Analysis	6
2.2. Prerequisites	6
2.2.1. Pre-requisites for Predictive Analysis	7
2.2.2. R Server Requirements	7
2.2.3. Predictive Spark Application Deployment Details	7
3. Getting Started with the BizViz Predictive Analysis	10
4. Predictive Analysis Home Page	13
4.1. Tree-node Menu	13
4.2. Header Menu- Options	14
4.3. Tabbed Menu Strip - Options.....	16
5. Acquiring Data from a Data Source	20
5.1. Acquiring Data from a CSV File	21
5.2. Acquiring Data from a Data Service	23
5.3. Acquiring Data from Cassandra Reader	26
5.4. Removing a Data Source from the Workspace	28
6. Data Preparation.....	29
6.1. Data Type Definition	29
6.2. Filter	31
6.3. Formula	34
6.4. Normalization.....	35
6.5. Sample.....	41
6.6. Spark Split Data	45
6.7. Spark Data Type Definition	48
7. Data Transformation.....	50
7.1. String Indexer	50



7.2.	Spark R Formula	52
8.	Algorithms.....	53
8.1.	Clustering	56
8.1.1.	R-K Means	56
8.1.2.	Spark-K- Means	59
8.2.	Forecasting.....	62
8.2.1.	Triple Exponential Smoothing.....	62
8.2.2.	Single Exponential Smoothing	67
8.2.3.	Double Exponential Smoothing	68
8.2.4.	R-Auto ARIMA	70
8.2.5.	R- Auto Forecasting.....	71
8.2.6.	Result View of Forecasting Algorithms when the selected output mode is 'Trend':	73
8.3.	Association.....	78
8.3.1.	Market Basket Analysis	78
8.4.	Regression Analysis.....	82
8.4.1.	R-Linear Regression.....	82
8.4.2.	R-Multiple Linear Regression	85
8.4.3.	R-Logistic Regression.....	87
8.5.	Outliers.....	88
8.5.1.	Interquartile Range	89
8.6.	Classification	92
8.6.1.	R-CNR Tree	92
8.6.2.	R-Naive Bayes.....	96
8.6.3.	Spark-Naive Bayes.....	98
8.7.	Correlation	102
8.7.1.	R- Correlation	102
9.	Apply Model.....	104
9.1.	Spark Apply Model	104
10.	Performance	106
10.1.	Binary Classification Model.....	108
10.2.	Multi Class Classification Model	108
11.	Data Writer(s)	109
11.1.	File Writer	109



11.1.1.	CSV Writer	109
11.1.2.	JSON Writer	110
11.2.	Database Writer	111
11.2.1.	Internal Data writer	111
11.2.2.	Cassandra Writer	114
12.	Custom R Script	119
12.1.	Creating a New R Script	119
12.2.	Saved R-Scripts	123
12.2.1.	Viewing a Saved R Script	123
12.2.2.	Editing a Saved R Script	123
12.2.3.	Deleting a Saved R Script	124
12.2.4.	Connecting Saved R Script with a Data Source	124
13.	Scheduler	126
13.1.	New Schedule	126
13.1.1.	Configuring General Tab	127
13.1.2.	Configuring Data Source	128
13.1.3.	Configuring a Data Writer	130
13.1.4.	Scheduling a New job	131
13.1.5.	Notification	135
13.2.	Status	137
14.	Live Job Status	138
15.	Saved Workflows	141
15.1.	Opening a Workflow	141
15.2.	Deleting a Workflow	142
15.2.1.	Delete Connection for a Workflow	142
15.3.	Renaming a Workflow	143
15.4.	Viewing Summary	143
15.5.	Sharing a Workflow	144
16.	Saved Models	146
16.1.	Saving a Model	146
16.2.	Reading a Model	147
16.3.	Renaming a Model	149
16.4.	Deleting a Model	150



16.5. Sharing a Model	151
17. Specific Options for a Spark Workflow	153
17.1. Force Start.....	153
17.2. Result of Each Component.....	154
17.3. Stop Button on the Progress Bar.....	154
17.4. Log Information Displayed under the Console Tab	155
18. Logging Out	155

1. About This Guide

1.1. Document History

The below table gives an overview of the most recent document changes:

Product Version	Date (Release date)	Description
BizViz Predictive Analysis 1.0	June 9 th , 2015	First Release of the document
BizViz Predictive Analysis 2.0	Feb 18 th , 2016	Updated document
BizViz Predictive Analysis 2.0	May 31 st , 2016	Minor Changes and Editing of the document
BizViz Predictive Analysis 2.5	November 9 th , 2016	Updated document

1.2. Overview

This guide covers steps to:

- Access the BizViz Predictive Analysis
- Designer Part of the BizViz Predictive Analysis
- Result or Analysis Part of the BizViz Predictive Analysis

1.3. Target Audience

This guide is aimed at business professionals, data analysts, data scientists, and statisticians who use BizViz Predictive Analysis tool to conduct various experimentations with data as in a Data Science Lab.

2. Introducing BizViz Predictive Analysis Tool

2.1. Introduction to the BizViz Predictive Analysis

BizViz Predictive Analysis is a statistical analytical tool that empowers its users by providing predictive models. These Predictive Models can be used to envision the future outcomes of business processes based on the past data. It is a user-friendly tool that shields users from the mathematical complexity and offers interactive graphical interface to provide an easy, intuitive experience. It enables the users to discover hidden insights and relationships in their data by applying various statistical algorithms provided by the popular R statistical language and Spark ML.

2.2. Prerequisites



2.2.1. Pre-requisites for Predictive Analysis

1. Predictive Analysis is a web based service so, only requirement is browser.
2. Predictive Analysis can be viewed only in desktops (mobile and tablet views are not supported).
3. R server and Predictive Spark App Settings should be configured from the Administration module.
4. User should be provided with all the necessary permissions to access and use the Predictive Analysis plugin from the User Management module of the BizViz Platform.
5. User should be permitted to access Data Management module from the BizViz Platform to use query service and Cassandra reader and writer for Predictive Analysis.
6. Limit of rows for data connectors need to be configured via the Administration module.

2.2.2. R Server Requirements

1. R server should be deployed publically.
2. Port should be open.
3. R server should be configured in Administration page of the BizViz platform.
4. Following packages should be installed in the R Server for predefined algorithms:
 - stringr
 - forecast
 - arules
 - arulesViz
 - rpart
 - e1071
5. In case of Custom R Script, script specific packages should be installed in the R Server.

2.2.3. Predictive Spark Application Deployment Details

1. Spark, Hadoop, Cassandra should be running in Cluster. For this application, Cluster should have free resources (Min 3 Core, 2 GB RAM in each executor according to application property).
2. Create a file with name spark_pa.properties in spark's configuration folder (cd \$SPARK_HOME/conf) and provide the following properties:

- spark.master <Spark master url:port> #Mandatory
- spark.app.name Spark Predictive Application #Mandatory.
- spark.scheduler.mode FAIR
- spark.eventLog.enabled true



- spark.eventLog.dir <log dir>
- spark.serializer org.apache.spark.serializer.KryoSerializer
- spark.extraListeners
org.apache.spark.ui.jobs.JobProgressListener,org.apache.spark.PASparkListener #Mandatory (Custom listener for the PA app)

3. Port Configuration: Any port series is fine provided they are exposed via the firewall. This is for the nodes within the Spark cluster.

- spark.ui.port 5003
- spark.history.ui.port 20080
- spark.driver.port 20081
- spark.executor.port 20082
- spark.filesystem.port 20083
- spark.broadcast.port 20084
- spark.replClassServer.port 20085
- spark.blockManager.port 20086

4. Cassandra Configuration

- spark.cassandra.input.split.size_in_mb 16
- spark.cassandra.input.fetch.size_in_rows 1000

5. Spark PA Configuration

- spark.pa.fs.default.name <HDFS host URL:port>
<hdfs://localhost:8020> #Mandatory
- spark.pa.process.queue.size 10 #Mandatory Default is 10. Queue size for PA app.
- spark.pa.process.pool.size 10 #Mandatory Default is 10. pool size for PA app.
- spark.pa.cache.size 100 #Mandatory Default is 100. Cache size for PA app.
- spark.pa.cache.timeout_sec 600 #Mandatory Default is 600 sec. Cache timeout for PA app
- spark.pa.hdfs.model.dir <hdfs://hostname:port/directory name> #Mandatory hdfs storage location for the models
<hdfs://localhost:8020/pa/model>
- spark.pa.hdfs.tmp.dir <hdfs://hostname:port/directory name> #Mandatory <hdfs://localhost:8020/pa/tmp>
- spark.pa.model.timeout_sec 86400 #Mandatory Default is 86400 (1 day). Time interval for deleting temporary model/s from the temporary hdfs location.



spark-pa.properties

6. Copy shade jar of pa_spark bundle in "spark/jars/" folder
 - Com.bdbizviz.pa.spark-shade-2.2.0.jar
7. Create a Script file named "start-pa.sh" in Spark's sbin folder to start application

If you need to execute in Kerberos mode, you need to generate the keytab file.

Script contents in Kerberos Mode:

```
#!/usr/bin/env bash

dir="$(cd "`dirname "$0"`"/..; pwd)"

nohup $dir/bin/spark-submit --keytab $dir/conf/hdfs.keytab \
--principal hdfs/<principlename> \
--executor-memory 3G --executor-cores 4 --num-executors 1 \
--verbose --properties-file $dir/conf/spark-pa.properties \
--driver-class-path $dir/jars/com.bdbizviz.pa.spark-shade
2.2.0.jar \
--class com.bdbizviz.pa.spark.executor.Executor --master yarn
deploy-mode client \
jars/com.bdbizviz.pa.spark-shade-2.2.0.jar 18786 >>
$dir/logs/spark-pa.log 2>&1&
```

please note that 18786 is a jetty port and can be changed to suite your needs

Script contents in Normal Mode:

```
#!/usr/bin/env bash

dir="$(cd "`dirname "$0"`"/..; pwd)"

nohup $dir/bin/spark-submit \
--executor-memory 3G --executor-cores 4 --num-executors 1 \
--verbose --properties-file $dir/conf/spark-pa.properties \
--driver-class-path $dir/jars/com.bdbizviz.pa.spark-shade
2.2.0.jar \
--class com.bdbizviz.pa.spark.executor.Executor --master yarn
deploy-mode client \
jars/com.bdbizviz.pa.spark-shade-2.2.0.jar 18786 >>
$dir/logs/spark-pa.log 2>&1&
```

please note that 18786 is a jetty port and can be changed to suite your needs



start-pa.txt

Save this file as a shell script (.sh)

8. Start Application with this command- **sbin/start-pa.sh**
9. Confirm the Spark PA Application is running in YARN:

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes
5	0	3	2	8	22 GB	25 GB	0 B	8	20	0	5	0	0	0	0

Scheduler Type		Scheduling Resource Type		Minimum Allocation		Maximum Allocation	
Capacity Scheduler	[MEMORY]	<memory:1024, vCores:1>	<memory:5120, vCores:4>				

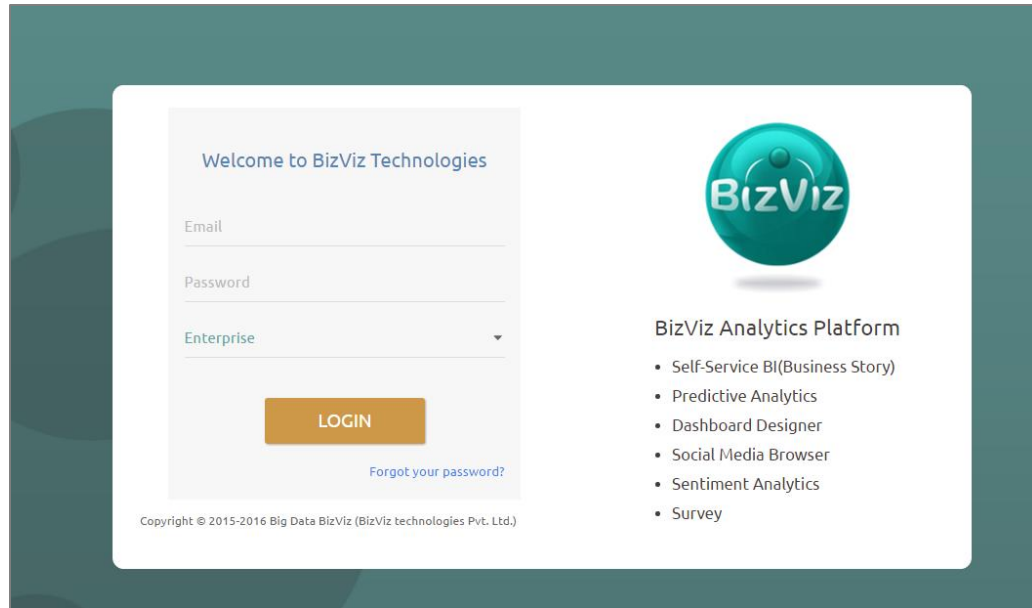
ID	User	Name	Application Type	Queue	StartTime	FinishTime	State	FinalStatus	Progress	Tracking UI	Blacklisted Nodes
application_1476353597736_0005	hdfs	Spark Predictive Application	SPARK	default	Tue Oct 18 14:52:02 +0550 2016	N/A	RUNNING	UNDEFINED	<input type="text"/>	ApplicationMaster	0
application_1476353597736_0004	hdfs	Spark Predictive Application	SPARK	default	Mon Oct 17 17:13:15 +0550 2016	Tue Oct 18 14:49:23 +0550 2016	FINISHED	SUCCEEDED	<input type="text"/>	History	N/A
application_1476353597736_0003	hdfs	Spark Predictive Application	SPARK	default	Thu Oct 13 16:11:09 +0550 2016	Mon Oct 17 17:11:56 +0550 2016	FINISHED	SUCCEEDED	<input type="text"/>	History	N/A
application_1476353597736_0002	hdfs	smb-analytics-17	SPARK	default	Thu Oct 13 15:53:04 +0550 2016	N/A	RUNNING	UNDEFINED	<input type="text"/>	ApplicationMaster	0
application_1476353597736_0001	hdfs	om-anache-spark-sol-hive-thriftServerHiveThriftServer2	SPARK	default	Thu Oct 13	N/A	RUNNING	UNDEFINED	<input type="text"/>	ApplicationMaster	0

Note: Confirm application have sufficient resources by the highlighted columns such as “Cores” and “Memory per Nodes”.

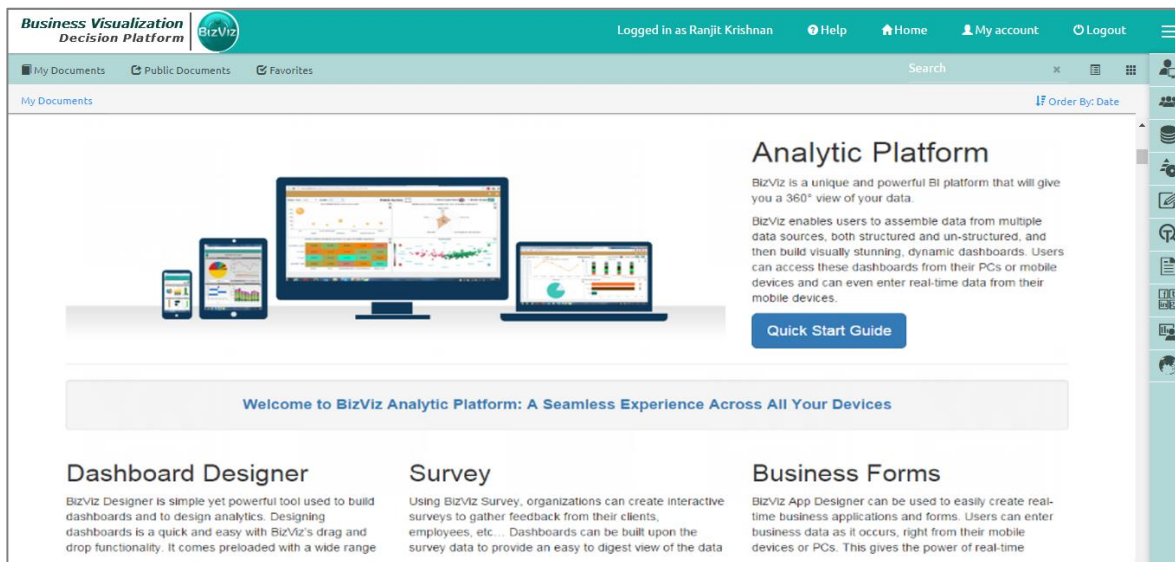
3. Getting Started with the BizViz Predictive Analysis

BizViz Predictive analysis is a plugin application provided under BizViz Platform.

- i) Open BizViz Enterprise Platform Link: <http://apps.bdbizviz.com/app/>
- ii) Enter your credentials to Login.
- iii) Click 'LOGIN'.

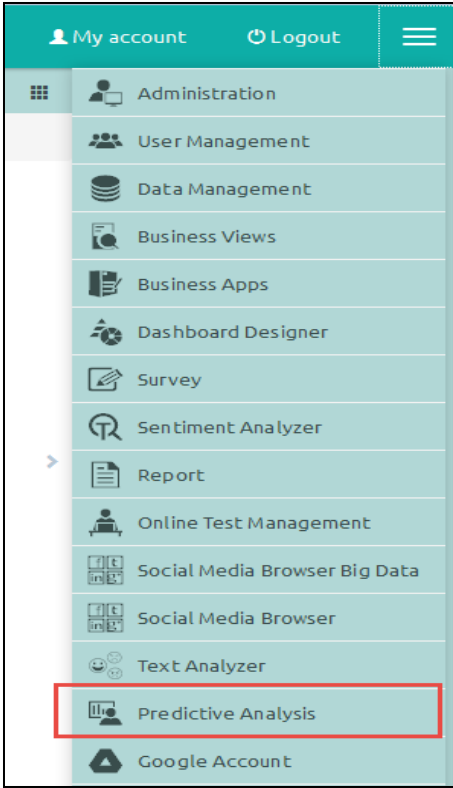


iv) Users will be redirected to the BizViz Platform home page.

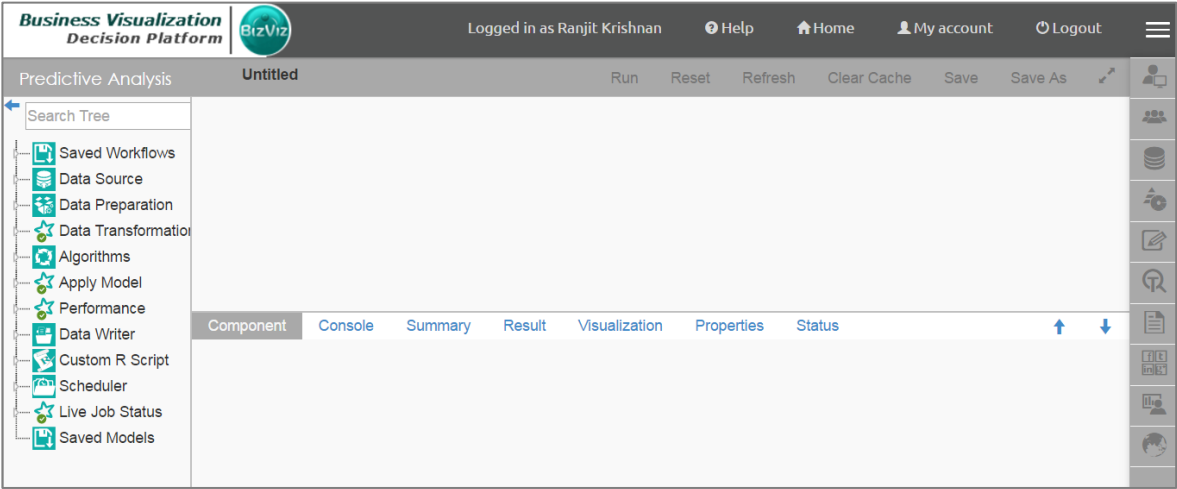


v) Click on the 'User Menu'  to display a list of all the available plugins.

vi) Select Predictive Analysis plugin from the list.



vii) Users will be redirected to the Predictive Analysis home page.



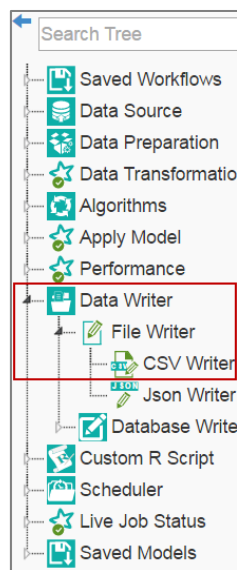
4. Predictive Analysis Home Page

This section describes all the options and icons provided on the Predictive Analysis home page. The Predictive Analysis home page can be described through the following Menus:


4.1. Tree-node Menu

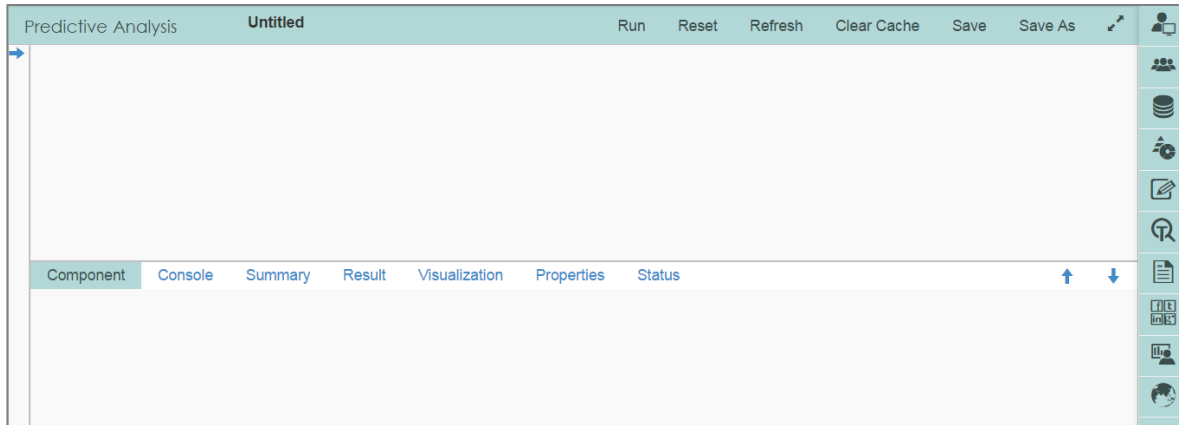
The Tree-node menu contains all the available component connectors to run a predictive execution. The components will be provided in the hierarchical order via a tree structure menu. All the main categories are included as tree-nodes and sub-categories are committed as petals to the respective tree-nodes.

E.g. '**Data Writer**' is a main category to which '**File Writer**' is committed as a sub-category and '**CSV Writer**' is displayed at the second level of hierarchy.



Note:

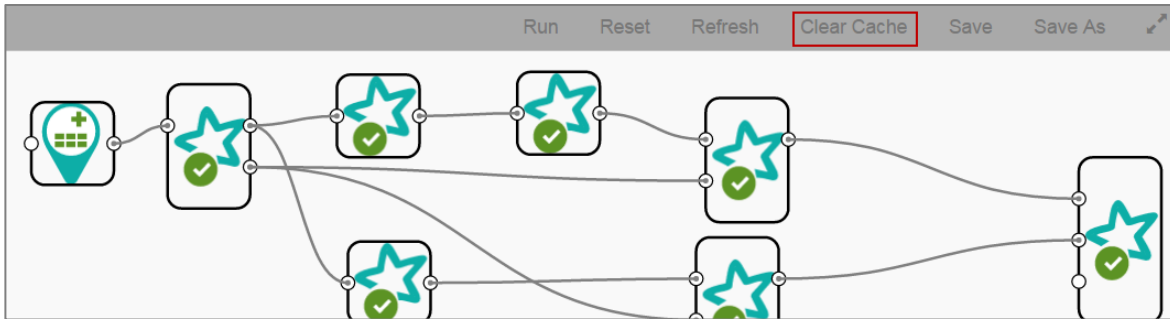
- a. The '**Search**' option has been provided for the entire tree structure menu.
- b. Click the '**Arrow**'  provided next to the '**Search**' box to collapse the tree structure menu from the home page.



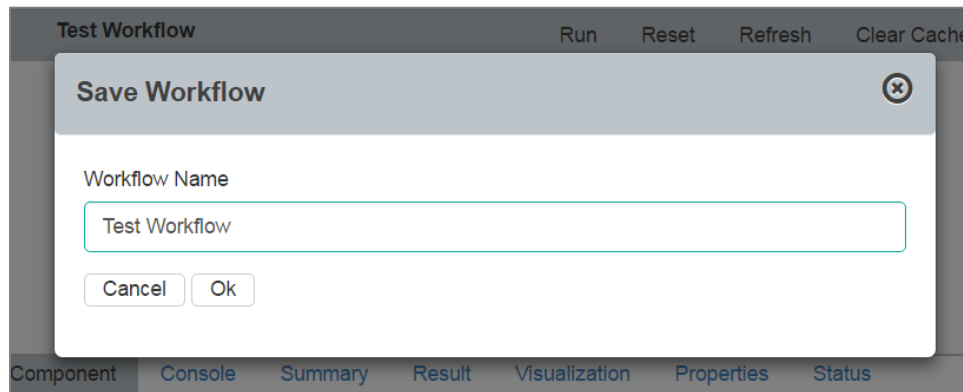
- c. This document is created focusing on each petal of the tree structure menu. All the available major and minor categories are described at length to understand a Predictive process.

4.2. Header Menu- Options

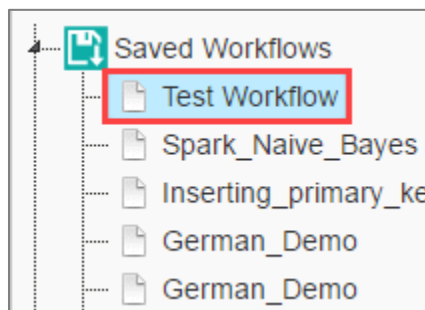
1. **Run:** Click 'Run' option to run the process and display the result set view. This option can be applied on data source, algorithms, and data preparation components.
2. **Reset:** The 'Reset' option to clean the workspace removing the current component connectors.
3. **Refresh:** The 'Refresh' option is provided on the menu row to fetch fresh data when adding a new component in the **Spark workflow**.
4. **Clear Cache:**
 - a. After using the 'Run' option, by default data will be cached in the server for the next 10 minutes. For latest results, users need to run workflow again.
 - b. Users need to click the 'Clear Cache' option to remove the cached data before running the workflow (again).
 - c. If users change any component parameter which is to be applied to fetch result then, 'Clear Cache' option must be clicked.




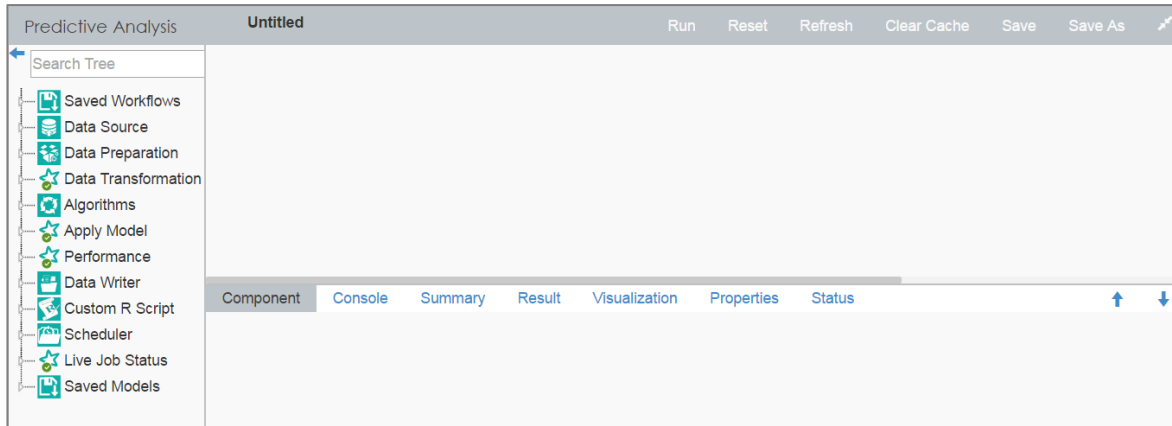
5. **Save:** Click the 'Save' option to save the created predictive workflow.
6. **Save As:** Click the 'Save As' option to copy a predictive workflow with a desired name.
 - i) Create a workflow by connecting various configured components.
 - ii) Click 'Save As'.
 - iii) A pop-up window will appear for confirmation.
 - iv) Click 'OK'.



- v) The workflow will be saved by the provided name in the 'Saved Workflows' list.



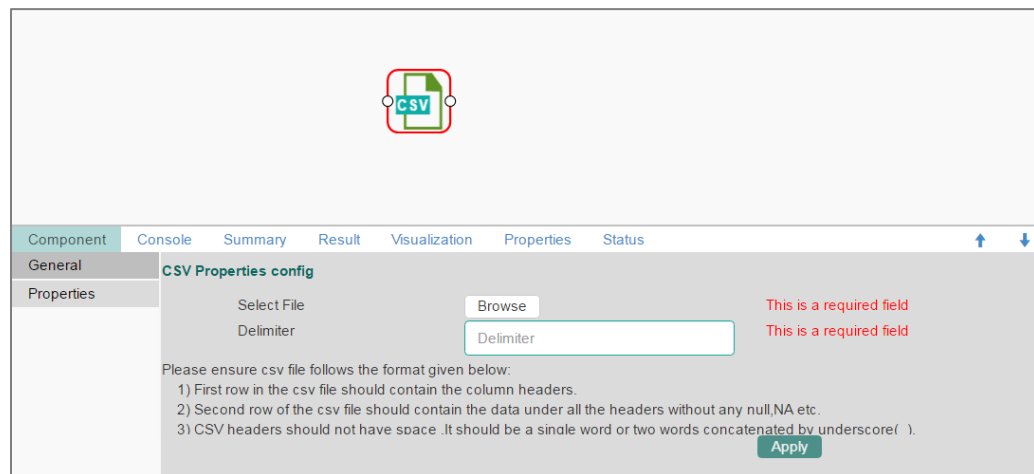
- Full Screen Icon:** Click 'Full Screen' icon  to provide full screen view of the Predictive Analysis home page. The platform menu row and plugin list will be removed to display a full screen view of the Predictive Analysis home page.



4.3. Tabbed Menu Strip - Options

1. Component

The 'Component' tab displays required configuration fields for the dragged components onto the workspace.



Note: The component tab may display various sub-tabs as per the selected components onto the workspace.

E.g. If the dragged data source is a CSV file then the component tab will display General and Properties fields while for the Cassandra Reader as a data source, the component tab will display General, Properties, and Column Selection.



2. Console

'Console' shows date and recorded time for the entire process.

- i) Click on '**Console**' option.
- ii) The below mentioned records will be displayed:
 - a. Process
 - b. Data Reader Process (starting and ending time)
 - c. R and Spark Process (starting and ending time)

```

Component  Console  Summary  Result  Visualization
15/10/2015 - 12:46:37 : Process started
15/10/2015 - 12:46:38 : Data Reader Process is started.
15/10/2015 - 12:46:38 : Data Reader Process is completed.
15/10/2015 - 12:46:38 : RProcess Process is started.
15/10/2015 - 12:46:39 : RProcess Process is completed.
    
```

3. Summary

Click the '**Summary**' tab to display R and Spark Server summary of the process.

```

Component  Console  Summary  Result  Visualization
----- Summary of the model -----
Column used in the algorithm :
  Airline_Passengers  (integer)
-----
      Length Class Mode
fitted   192 mts numeric
x         60 ts  numeric
alpha    1  -none- numeric
beta     1  -none- numeric
gamma    1  -none- numeric
coefficients 14 -none- numeric
seasonal  1  -none- character
SSE      1  -none- numeric
call     7  -none- call

The Model representation
Holt-Winters exponential smoothing with trend and additive seasonal component.

Call:
HoltWinters(x = tso, alpha = 0.3, beta = 0.1, gamma = 0.1, seasonal = c("additive"), start.periods = 2)

Smoothing parameters:
alpha: 0.3
beta : 0.1
gamma: 0.1

Coefficients:
    
```

4. Result

Click the **'Result'** tab to display a result list view based on the selected execution.

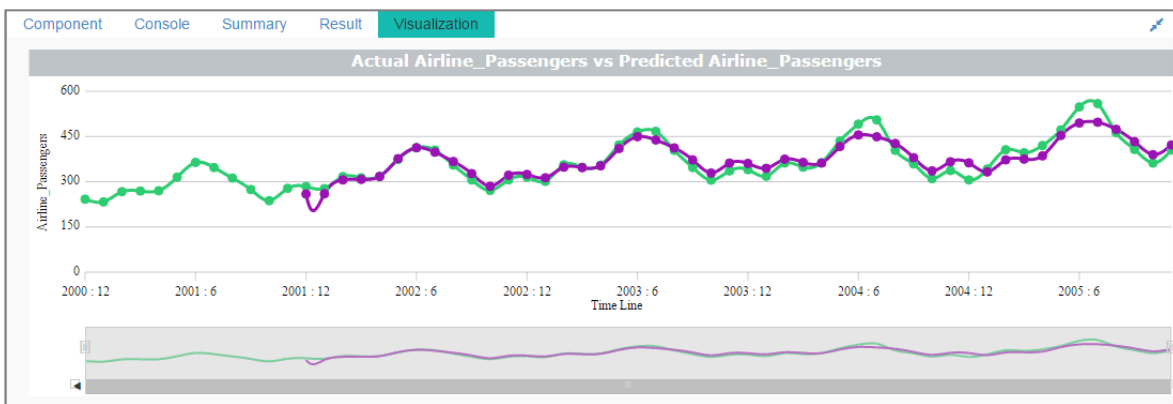
Year	Month	Date	Airline_Passengers	X_Axis	PredictedValues
2005	7	Aug-59	559	g2	497.358
2005	6	Jul-59	548	g1	494.752
2005	8	Sep-59	463	g3	473.775
2004	6	Jul-58	491	e7	455.19
2005	5	Jun-59	472	f9	453.289
2003	6	Jul-57	465	d4	449.297
2004	7	Aug-58	505	e8	449.006
2003	7	Aug-57	467	d5	438.427
2005	9	Oct-59	407	g4	432.908
2004	8	Sep-58	404	e9	426.938

Showing 1 to 10 of 60 entries

Note: The **'Result'** tab will be displayed for the given data only after data is configured and **'Run'** or **'Run Till Here'** option is selected. Upto 50000 cells can be displayed in the Result view.

5. Visualization

Click the **'Visualization'** tab to display graphical representation of the result data.



6. Properties:

Click the **'Properties'** tab to display properties for the current workflow on the Workspace.



Component	Console	Summary	Result	Visualization	Properties	Status
Created By			Ranjit Krishnan			
Created At			2016-10-03 15:28:28 +0530			
Last Modified By			Ranjit Krishnan			
Last Modified At			2016-10-03 15:28:28 +0530			
Version			2.2.0			

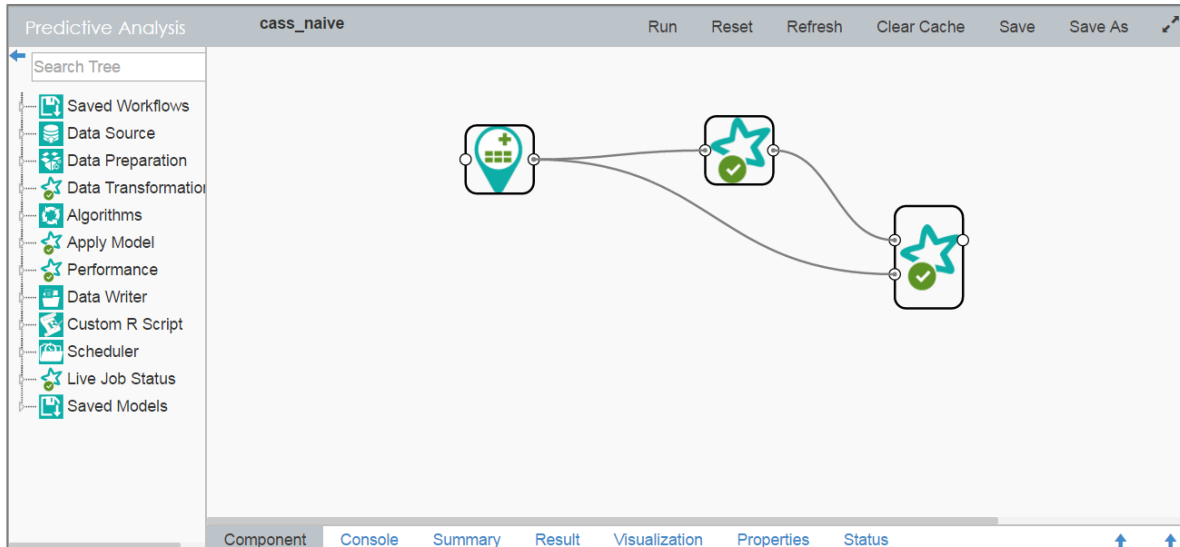
7. **Status:** Click the 'Status' tab to view the live job status of a running Spark job.


Workflow Name	Run by	Start time	End Time	Status	View Log	Live job status	Summary	Actions
CSS 22nd Sept	Ranjit Krishnan	Thu, 22 Sep 2016 09:18:07 GMT	NA	in progress				
CSS 22nd Sept	Ranjit Krishnan	Thu, 22 Sep 2016 09:11:41 GMT	NA	in progress				
css	Ranjit Krishnan	Wed, 21 Sep 2016 13:36:40 GMT	NA	in progress				

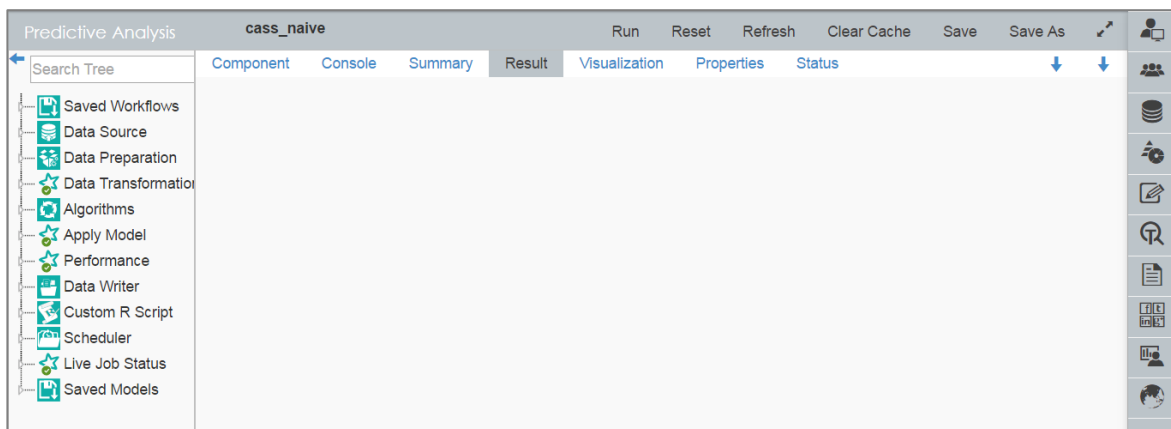
8. **Minimize Maximize Button**

The 'Minimize/Maximize' buttons have been provided to the view menu row to customize the workspace and view space as per the user requirement.

- a. Click icon to minimize the Tabbed Menu Strip on the Predictive Analysis home page.



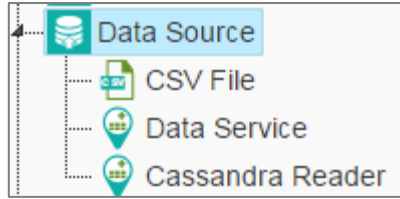
b. Click  icon to maximize the Tabbed Menu Strip on the Predictive Analysis home page.



5. Acquiring Data from a Data Source

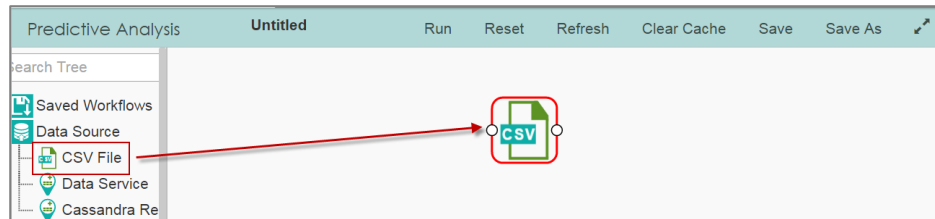
Acquiring data from a data source is the initial step for Predictive Analysis. The ‘**Data Source**’ tree-node offers 3 types of data connectors:

- a. CSV File
- b. Query Service
- c. Cassandra Reader

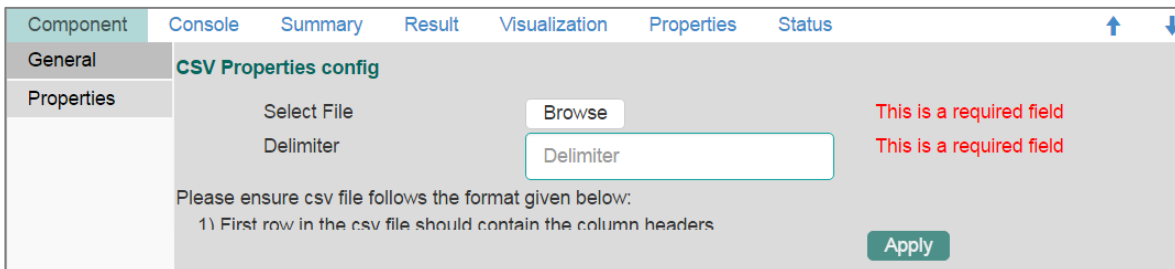


5.1. Acquiring Data from a CSV File

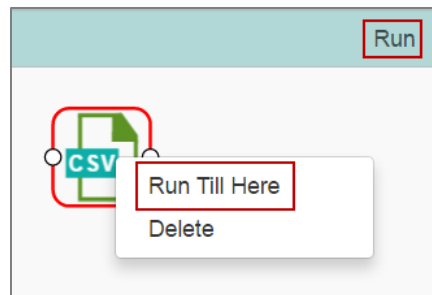
- i) Select and drag **'CSV File'** component onto workspace.
- ii) Click the **'CSV File'** component.



- iii) Configure the following **'CSV Properties Configuration'** fields:
 - a. **Select File:** Browse a CSV file
 - b. **Delimiter:** Mention the delimiter used in the CSV file
- iv) Click **'Apply'**.



- v) Click **'Run'** or **'Run Till Here'**.



- vi) The **'Result'** view or file data will be displayed.



SepalLength	SepalWidth	PetalLength	PetalWidth	Species
5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa

Showing 1 to 10 of 150 entries

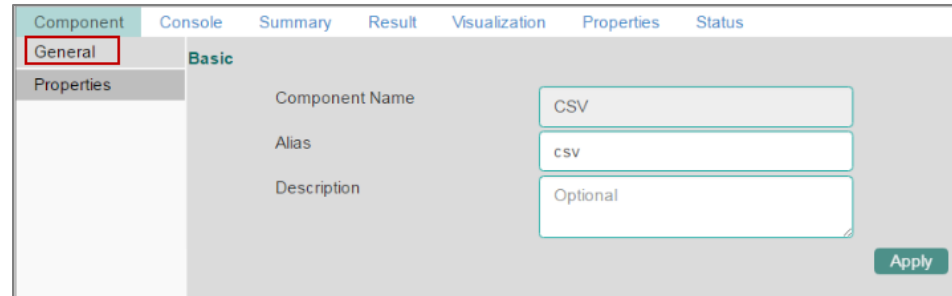
Previous 1 2 3 4 5 ... 15 Next

• **Rules to be Followed while Uploading a CSV File**

1. First row provided in the CSV file should contain the column headers.
2. Second row of the CSV file should contain the data under all the headers without any 'null' or 'NA'.
3. CSV headers should not have space. It should be a single word or two words concatenated by an underscore (_).
4. CSV headers should not contain any special characters. E.g. - %, #, \$, @, *, etc.
5. CSV headers should not contain single or double quotes, dot, brackets, and high-fen.
6. CSV headers should not contain merely numbers. Numerals should be used with at least one alphabet.
7. CSV header should not exceed 50 characters.
8. All rows in a column should have the same data type.

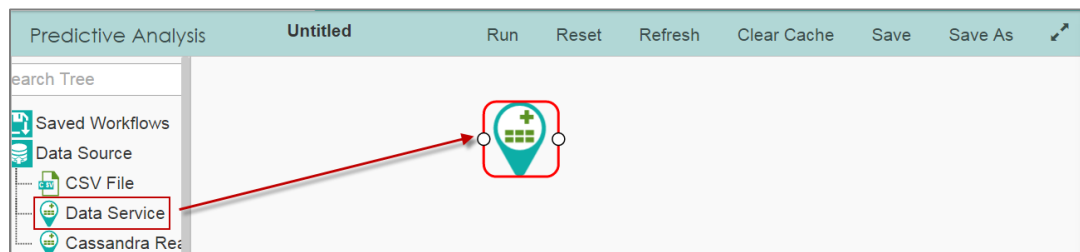
Note:

- a. The supported file types will be .csv, .tsv .
- b. **'General'** tab is provided to configure the following information for any tree-node component:
 - i. Alias Name
 - ii. Description (it is an optional field)
(E.g. the following image displays **'General'** tab for a CSV data source.)



5.2. Acquiring Data from a Data Service

- i) Select and drag '**Data Service**' connector onto workspace.
- ii) Click the '**Data Service**' connector.



- iii) Users will be redirected to the '**Properties**' fields provided under '**Components**' tab on the Tabbed Menu Strip.
- iv) Configure the '**Data Service Properties**':
 - a. **Select Data Connector**: Select a datasource from the drop-down menu
 - b. **Select Data Service**: Select a query service from the drop-down menu
 - c. **Fields**: The following tables will be displayed:
 - Column Header
 - Data Type
- v) Click '**Next**'.



Component	Console	Summary	Result	Visualization	Properties	Status
General	Data Service Properties					
Properties	Select Data Connector	QA_predictive ▾				
Conditions	Select Data Service	Employee_details ▾				
	Fields					
	Column Header	Data type				
	Employee_Id	int				
	First_Name	string				
	Last_Name	string				
	Salary	double				
	Joining_Date	timestamp				
	department	string				
						Next

- vi) Users will be redirected to the **'Conditions'** tab. (If the selected data service contains the filter values).
- vii) Configure the following information:
 - a. **Filter Type:** Available filter(s) in the data service will be displayed under this space.
 - b. **Control Type:** Users are provided with the following options to pass the filter values under this option:
 - **Text:** By selecting this option users can manually enter multiple filter values separated by coma.

Component	Console	Summary	Result	Visualization	Properties	Status
General	Filter Name		Control Type			
Properties	department		Text ▾			
Conditions						Apply

- **LOV:** By selecting this filter value option users will be directed to select another Data Connector and Data Service available in the space.
 - i. Once user selects a data service, a list of values will display for the user to select the filter values.

- ii. Users can select multiple values as filter values from the selected data service.

Component	Console	Summary	Result	Visualization	Properties	Status
General	Filter Name		Control Type			
Properties	department		LOV			
Conditions	Select Data Connector		QA_predictive			
	Select Data Service		Select			
						Apply

- viii) Click 'Apply'.
- ix) Click 'Run' or 'Run Till Here'.
- x) The 'Result' view or data from the data service will be displayed.

Employee_Id	First_Name	Last_Name	Salary	Joining_Date	department
1	John	Pinto	1000000	2013-01-01 12:30:00.0	Banking
2	Roy	Clarke	800000	2013-01-07 09:03:00.0	Insurance
3	Tom	Thomas	700000	2015-08-04 21:00:40.0	Services
4	Jerry	Jose	600000	2013-01-01 09:45:56.0	Banking
5	Philip	Mathew	650000	2013-08-01 10:10:40.0	Sales
6	Caren	Smith	750000	2013-02-01 09:47:01.0	Insurance
7	Abraham	Lida	650000	2013-07-06 11:06:30.0	Services
8	Richi	Margie	750000	2013-04-01 09:50:06.0	Services
9	Johny	Gill	650000	2013-01-01 10:18:32.0	Insurance
10	Sanky	Steve	850000	2013-02-01 10:05:00.0	Banking

Rules to be Followed while Creating a Data Service

1. Data service header should not have space. It should be a single word or two words concatenated by an underscore (_).
2. Data service header should not contain any special characters. E.g. - %, #, \$, @, *, etc.
3. Data service header should not contain single or double quotes, dot, brackets, and high-fen.
4. Data service header should not contain merely numbers. Numerals should be used with at least one alphabet.
5. Data service header should not exceed 50 characters.

Note:

- d. Users can develop a data service via the Data Management module of the BizViz Platform.
- e. **'Fields'** option under **'Properties'** tab will appear only after selecting the appropriate query service.
- f. LOV service provided under **'Conditions'** tab can contain only one column, in case of more than one column a warning message will appear.
- g. Users can configure the following information for a data service data source via **'General'** tab:
 - i. Alias Name
 - ii. Description (it is an optional field)

5.3. Acquiring Data from Cassandra Reader

- i) Select and drag **'Cassandra Reader'** connector onto workspace.
- ii) Click on the **'Cassandra Reader'** connector.
- iii) Users will be redirected to the **'Properties'** tab.
- iv) Configure the required properties:
 - a. Select Data Connector: Select a data connector using the drop-down menu
 - b. Host Name: Data connector specific hostname will be displayed
 - c. Port Number: Port number will be displayed
 - d. User Name: User name will be displayed
 - e. Password: Enter the password
 - f. Cluster Name: Enter a cluster name
 - g. Select Key Space: Select a key space from the drop-down menu
 - h. Select Table: Select a table from the drop-down menu
 - i. Limit by Row: Select an option using the drop-down menu. Two options will be provided as shown below:
 - a. Select all Rows
 - b. Limit By
 - j. Max. no. of Rows to be fetched: Enter a number to decide maximum fetched rows. (This option will appear only if 'Limit By' option has been selected using the 'Limit by Row' field. Default value for this field is 1000).
- v) Click **'Next'**.



Component Console Summary Result Visualization Properties Status

General **Data Service Properties**

Properties

Column Selection

Select Data Connector: Cassandra_BizVizQA ▾

Host Name: 192.168.1.17

Port Number: 9042

Username: smb

Password:

Cluster Name: AB

Select Key Space: UCI ▾

Select Table: pokerhandtrain ▾

Limit No: of rows to fetch: Limit by ▾

Max no: of rows to be fetched: 1000

Next

- vi) Users will be redirected to the 'Column Selection' tab.
- vii) Select the required columns from the list.
- viii) Click 'Apply'.

Component Console Summary Result Visualization Properties Status

General **Meta Data**

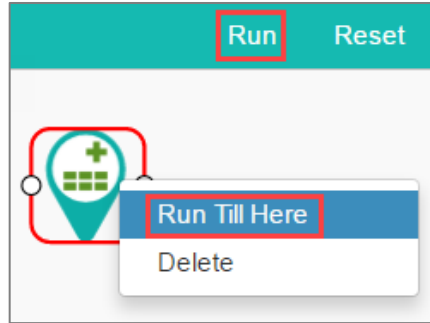
Properties

Column Selection

Headers	Type	Specify
ROWNUM	INT	
C1	DOUBLE	
C2	DOUBLE	
C3	DOUBLE	
C4	DOUBLE	
C5	DOUBLE	
CLASS	DOUBLE	
S1	DOUBLE	
S2	DOUBLE	
S3	DOUBLE	
S4	DOUBLE	
S5	DOUBLE	

Apply

- ix) Click 'Run' or 'Run Till Here'.



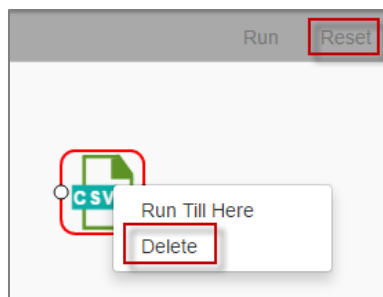
x) The Result view will be displayed.

Component		Console	Summary	Result	Visualization	Properties	Status
Show 10 entries		Search:					
C1	C2						
1.0	2.0						
2.0	4.0						
3.0	3.0						
1.0	3.0						
1.0	2.0						
1.0	3.0						
2.0	4.0						
3.0	2.0						
3.0	1.0						
3.0	2.0						
Showing 1 to 10 of 1,000 entries		Previous		1	2	3	4
				5	...	100	Next

Note: The Apache Spark predictive workflows require a 'Cassandra Reader' as a data source. The Cassandra Reader can be also used as a data source for the R Wrokflows.

5.4. Removing a Data Source from the Workspace

- i) Right click on the Data Source connector (on the workspace).
- ii) A context menu will appear.
- iii) Click 'Delete'.





iv) The selected Data Source connector will be removed from the workspace.

OR

Click on the **'Reset'** option to remove the connector(s) from the workspace.

Note: The same set of steps can be followed to remove a Data Service and Cassandra Reader data source from the workspace.

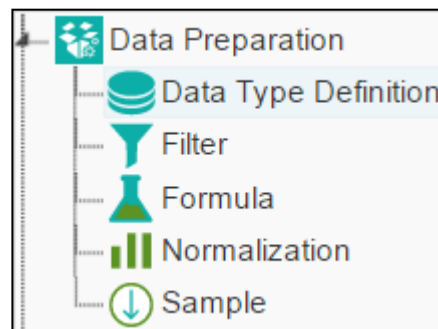
6. Data Preparation

Components provided under **'Data Preparation'** help in preparing the raw data from the data source and make it suitable for analysis. They organize data in order to gain accurate result out of it.

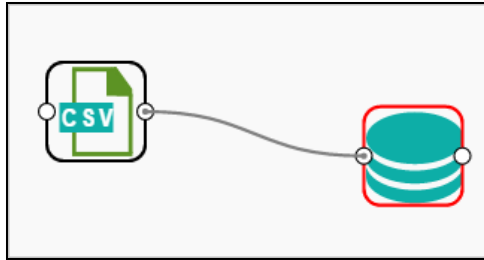
6.1. Data Type Definition


Data Type Definition can be used to change the name, data type of the data source column. This component helps users to prepare data and make it suitable for further analysis.

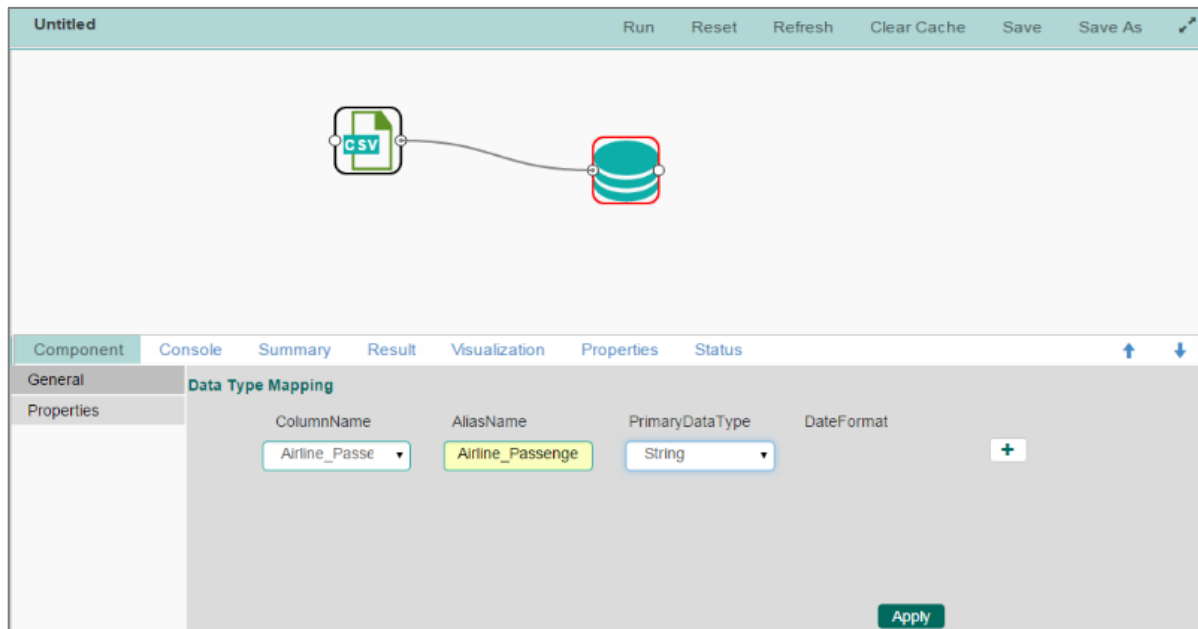
- i) Navigate to the Predictive home page.
- ii) Click **'Data Preparation'** tree-node.
- iii) A context menu will open.



- iv) Drag **'Data Type Definition'** component and connect it with a configured data source onto the workspace.
- v) Click the **'Data Type Definition'** component (on the workspace).



- vi) Users will be redirected to the **'Properties'** tab.
- vii) Configure the following **'Data Type Mapping'** details:
 - a. **Column Name:** Select a column name which you want to change
 - b. **Alias Name:** Enter an alias name for the required source column
 - c. **Primary Data Type:** Select a primary data type column that you want to change
 - d. **Date Format:** Select a date format that you want to display (Date format is optional for date Data Type)
 - e. **'Add' option**  : Click on this button to add one more row of the **'Data Type Mapping'** fields
- viii) Click **'Apply'**.



- ix) Click **'Run'** or **'Run Till Here'**.
- x) The **'Result'** view will be displayed.



Predictive Analysis **Untitled** Run Reset Refresh Clear Cache Save Save As

Saved Models
 Data Source
 CSV File
 Query Service
 Data Preparation
 Data Type Definition
 Filter
 Formula
 Normalization
 Sample
 Algorithms
 Data Writer
 Custom R Script
 Scheduler

Run Till Here
Delete

Component Console Summary **Result** Visualization Properties Status

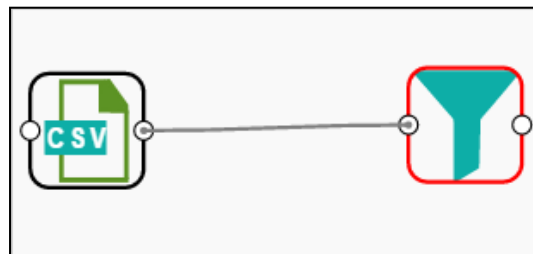
Show 10 entries Search:

Date	Airline_Passengers	X_Axis
Jan-55	242	a1
Feb-55	233	a2
Mar-55	267	a3
Apr-55	269	a4
May-55	270	a5
Jun-55	315	a6

6.2. Filter

This option is used to filter the data as per the business requirement.

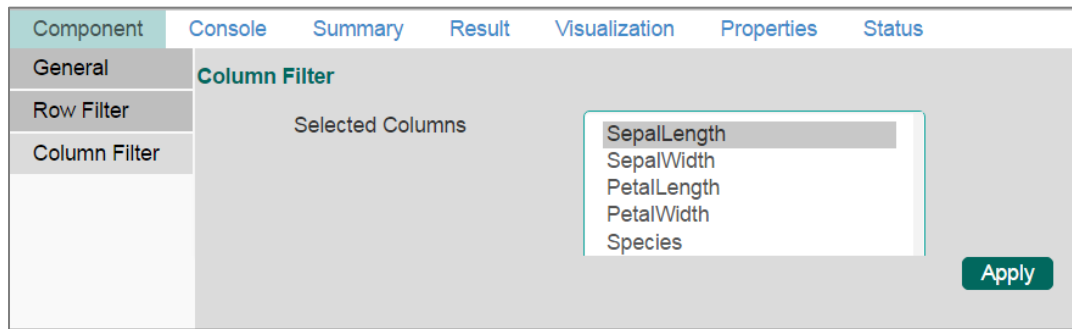
- i) Select and Drag **'Filter'** component onto the workspace.
- ii) Connect the **'Filter'** component to a configured datasource component.
- iii) Click the **'Filter'** component.



- iv) Configure the following component tabs:

Column Filter

- a. Select a column from the **'Selected Columns'** drop-down menu.
- b. Click **'Apply'** to configure the data.



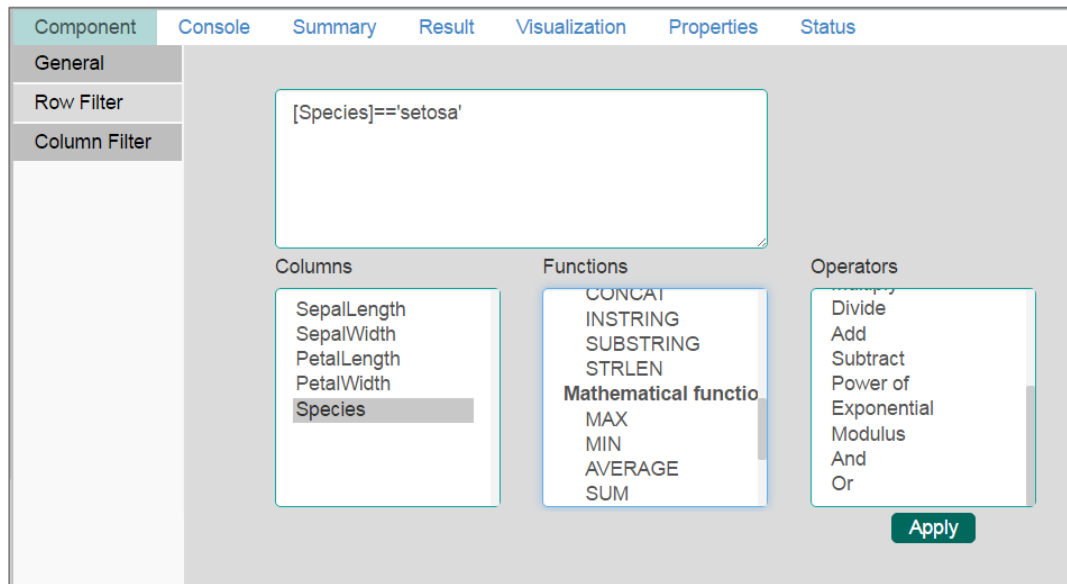
Result View (Column Filter):

- i) Click **'Run'** or **'Run Till Here'** option to display the **'Result'** view.
- ii) The filtered data will be displayed via the **'Result'** tab.

SepalLength
5.1
4.9
4.7
4.6
5
5.4
4.6
5
4.4
4.9

Row Filter

- i) Drag and connect the **'Filter'** component onto the workspace.
- ii) Connect the **'Filter'** component to a configured datasource.
- iii) Click the **'Filter'** component.
- iv) The **'Column Filter'** tab will be displayed (by default).
- v) Select **'Row Filter'** tab from the **'Component'** menu list.
- vi) Configure the required fields:
 - a. Double click on the components from **Columns, Functions,** and **Operators** list menus
 - b. A formula will be entered in the given box
 - c. Click **'Apply'**.



Result View (Row Filter):

- i) Click 'Run' or 'Run Till Here'.
- ii) The filtered data will be displayed via the 'Result' tab.

SepalLength	SepalWidth	PetalLength	PetalWidth	Species
5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa

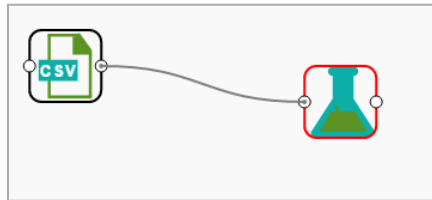
Note:

- a. Expression should retain Boolean output.
- b. User can not use Data manipulation functions.

6.3. Formula

User can create a calculated column using **'Formula'**. A formula can be created by using available columns, functions, and operators.

- i) Select and drag **'Formula'** component onto the workspace.
- ii) Connect the **'Formula'** component to a configure datasource.
- iii) Click on the **'Formula'** component.



- iv) Configure the required component fields:

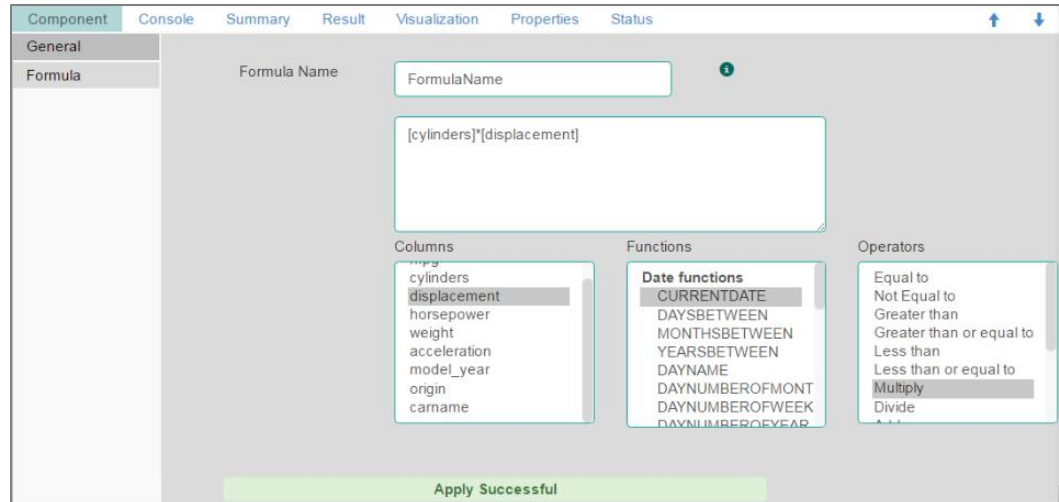
General

- a. **Component Name:** The default name for the component will be displayed
- b. **Alias Name:** Enter an appropriate name for the component (If required)
- c. **Description:** Describe about the component (It is an optional field)

The screenshot shows the configuration panel for the 'Formula' component. At the top, there are tabs for 'Console', 'Summary', 'Result', 'Visualization', 'Properties', and 'Status'. The 'Properties' tab is active, and the 'Basic' section is expanded. On the left, a sidebar lists 'General' and 'Formula', with 'Formula' selected. The main area contains three input fields: 'Component Name' with the value 'Formula', 'Alias' with the value 'Formula', and 'Description' with the value 'Optional'.

Formula

- a. **'Columns', 'Functions', and 'Operators'** : Double click on these lists will enter a formula in the given box.
- b. **Formula Name:** Enter a formula name in the given field.
- c. **Apply:** Click on this button to configure the formula.



- v) Click 'Run' or 'Run Till Here'.
- vi) The 'Result' view will be displayed.

mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	carname	FormulaName
8	8	307	130	3504	12	70	1	chevrolet chevelle malibu	2456.0
15	8	350	165	3693	11.5	70	1	buick skylark 320	2800.0
18	8	318	150	3436	11	70	1	plymouth satellite	2544.0
16	8	304	150	3433	12	70	1	amc rebel sst	2432.0
17	8	302	140	3449	10.5	70	1	ford torino	2416.0
15	8	429	198	4341	10	70	1	ford galaxie 500	3432.0
14	8	454	220	4354	9	70	1	chevrolet impala	3632.0
14	8	440	215	4312	8.5	70	1	plymouth fury iii	3520.0
14	8	455	225	4425	10	70	1	pontiac catalina	3640.0
15	8	390	190	3850	8.5	70	1	amc ambassador dpl	3120.0

Showing 1 to 10 of 398 entries

6.4. Normalization

This component controls the relevant data. It attempts to convert the available data from larger range to smaller range.

- **Normalization Methods**

Normalization contains 3 methods to normalize the vast amounts of data:

1. **Min-Max Normalization**

It implements a linear transformation on the original data values, and sets a new range for all the data values to fit in. User can fix New Maximum and New Minimum Value for the data from the new range. Consequently, each value “v”

from the original interval will be mapped into value “new_v” following the below given formula:

$$new_v = \frac{v - min_x}{max_x - min_x} \cdot (new_max_x - new_min_x) + new_min_x$$

2. Zero-Score

This normalization also known as ‘Zero Mean Normaliation’ is calculated on the ‘mean’ and ‘standard deviation’ for each attribute. It determines on whether a specific value is above or below average. It also signifies the exact proportion of the variance from the fixed limit of aver3age. After applying ‘Zero-Score’ normalization each feature will have mean value of zero (0). The unit of each value will be the number of (estimated) standard deviations away from the (estimated) mean. Zero score normalization may be sensitive to small values of ‘σ_x’. A new value ‘new_v’ can be found by using the following expression:

$$new_v = \frac{v - \mu_x}{\sigma_x}$$

3. Decimal-Scaling

The decimal point of the value of each element is moved in accord with its maximum absolute value. A modified value ‘new_v’ can be obtained using the following formula:

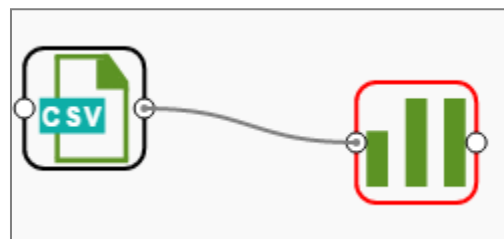
$$new_v = \frac{v}{10^c}$$

Note: In the decimal-scaling expression ‘c’ is the smallest integer so that max(new_v) < 1.

• Applying Normalization

1. Min-Max

- i) Select and drag ‘Normalization’ component onto the Workspace.
- ii) Connect the ‘Normalization’ component to a configured data source.
- iii) Click the ‘Normalization’ component.





iv) Configure the following component fields:

Properties

a. **Column Selection**

i. **Select a Column:** Select a column using drop-down menu (Only numerical column will be selected)

b. **Behavior**

i. **Normalization Type:** Select 'Min-Max' normalization type from the drop-down menu

ii. **New Maximum Value:** Set a new maximum value (Default value for this field is 1)

iii. **New Minimum Value:** Set a new minimum value (Default value for New Minimum field is 0)

v) Click 'Apply'.

Component	Console	Summary	Result	Visualization	Properties	Status
General	Column Selection					
Properties	Select a Column	cylinders		i		
	Behavior					
	Normalization Type	Min-Max				
	New Maximum	10				
	New Minimum	0				
						Apply

vi) Click 'Run' or 'Run Till Here'.

vii) Users will be directed to the 'Result' tab will be displaying the result data.



mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	carname
8	10.0	307	130	3504	12	70	1	chevrolet chevelle malibu
15	10.0	350	165	3693	11.5	70	1	buick skylark 320
18	10.0	318	150	3436	11	70	1	plymouth satellite
16	10.0	304	150	3433	12	70	1	amc rebel sst
17	10.0	302	140	3449	10.5	70	1	ford torino
15	10.0	429	198	4341	10	70	1	ford galaxie 500
14	10.0	454	220	4354	9	70	1	chevrolet impala
14	10.0	440	215	4312	8.5	70	1	plymouth fury iii
14	10.0	455	225	4425	10	70	1	pontiac catalina
15	10.0	390	190	3850	8.5	70	1	amc ambassador dpl

Showing 1 to 10 of 398 entries

Previous 1 2 3 4 5 ... 40 Next

2. Zero Score

- i) Select and drag **'Normalization'** component onto the Workspace.
- ii) Connect the **'Normalization'** component to a configured data source.
- iii) Click the **'Normalization'** Component.
- iv) Configure the required component fields:

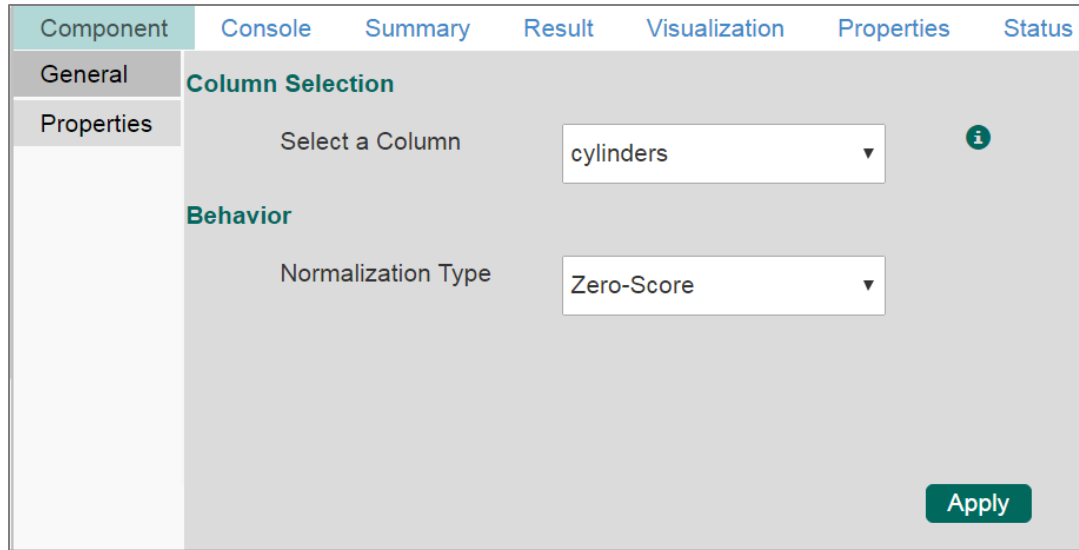
Properties

a. Column Selection

- i. **Select a Column:** Select a column using drop-down menu (Only numerical column will be selected).

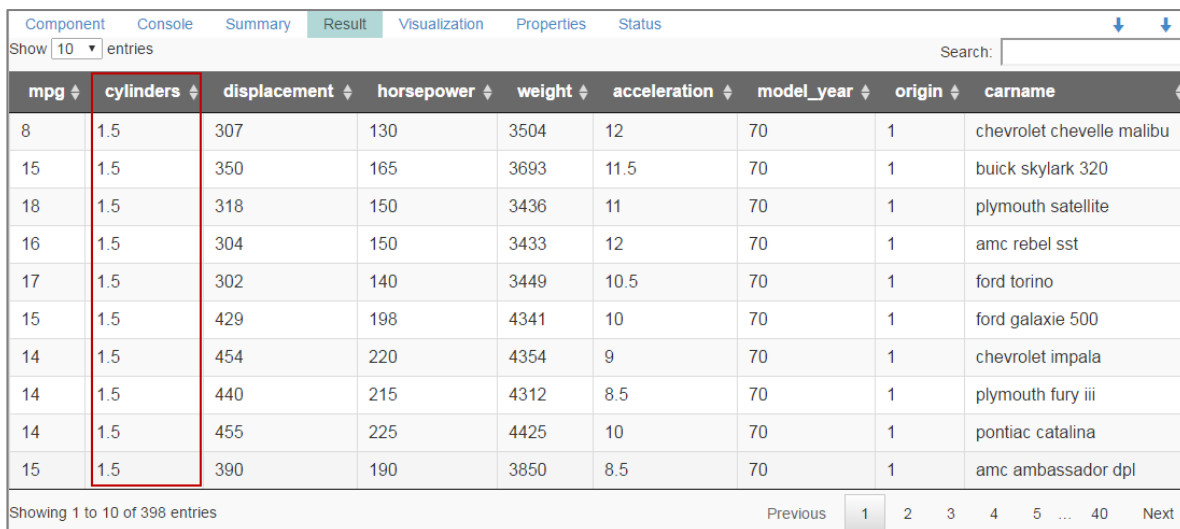
b. Behavior

- i. **Normalization Type:** Select **'Zero-Score'** normalization type from the drop-down menu.
- v) Click **'Apply'** to configure the fields.



The screenshot shows the 'Properties' tab of a component configuration interface. It features two main sections: 'Column Selection' and 'Behavior'. In the 'Column Selection' section, there is a label 'Select a Column' followed by a dropdown menu currently displaying 'cylinders'. In the 'Behavior' section, there is a label 'Normalization Type' followed by a dropdown menu currently displaying 'Zero-Score'. An information icon (i) is located to the right of the 'cylinders' dropdown. At the bottom right of the configuration area, there is a green 'Apply' button.

- vi) Click 'Run' or 'Run Till Here'.
- vii) Users will be directed to the 'Result' tab displaying the result list view.



The screenshot shows the 'Result' tab of the application, displaying a data table. The table has a search bar at the top right and a 'Show 10 entries' dropdown at the top left. The table columns are: mpg, cylinders, displacement, horsepower, weight, acceleration, model_year, origin, and carname. The 'cylinders' column is highlighted with a red box. The data rows are as follows:

mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	carname
8	1.5	307	130	3504	12	70	1	chevrolet chevelle malibu
15	1.5	350	165	3693	11.5	70	1	buick skylark 320
18	1.5	318	150	3436	11	70	1	plymouth satellite
16	1.5	304	150	3433	12	70	1	amc rebel sst
17	1.5	302	140	3449	10.5	70	1	ford torino
15	1.5	429	198	4341	10	70	1	ford galaxie 500
14	1.5	454	220	4354	9	70	1	chevrolet impala
14	1.5	440	215	4312	8.5	70	1	plymouth fury iii
14	1.5	455	225	4425	10	70	1	pontiac catalina
15	1.5	390	190	3850	8.5	70	1	amc ambassador dpl

At the bottom of the table, it says 'Showing 1 to 10 of 398 entries' and a pagination control showing 'Previous 1 2 3 4 5 ... 40 Next'.

3. Decimal Scaling

- i) Select and drag 'Normalization' component onto the Workspace.
- ii) Connect the 'Normalization' component to a configured data source.
- iii) Click the 'Normalization' Component.
- iv) Configure the required component fields:

Properties

a. Column Selection

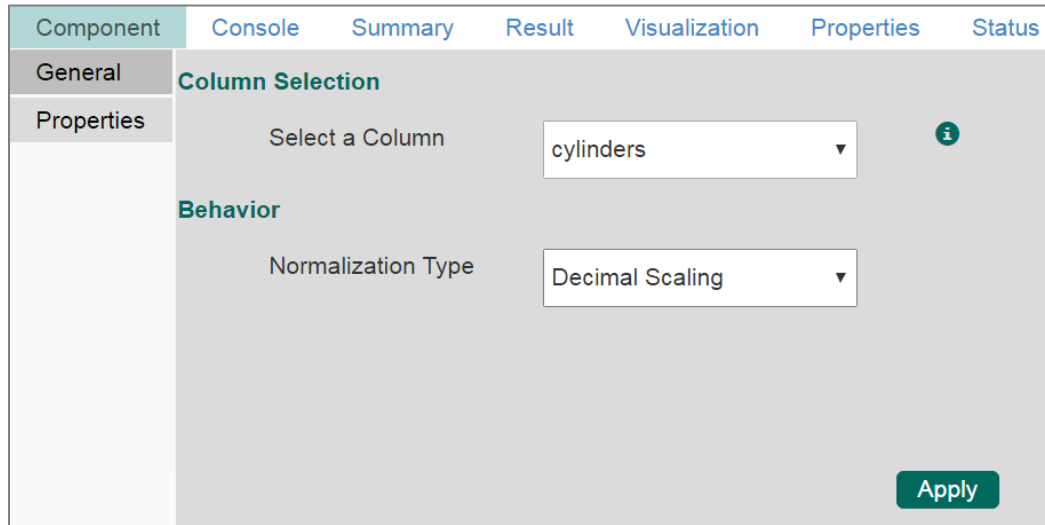
- i. **Select a Column:** Select a column using drop-down menu (Only

numerical column will be selected).

b. Behavior

- i. **Normalization Type:** Select **'Decimal Scaling'** normalization type from the drop-down menu.

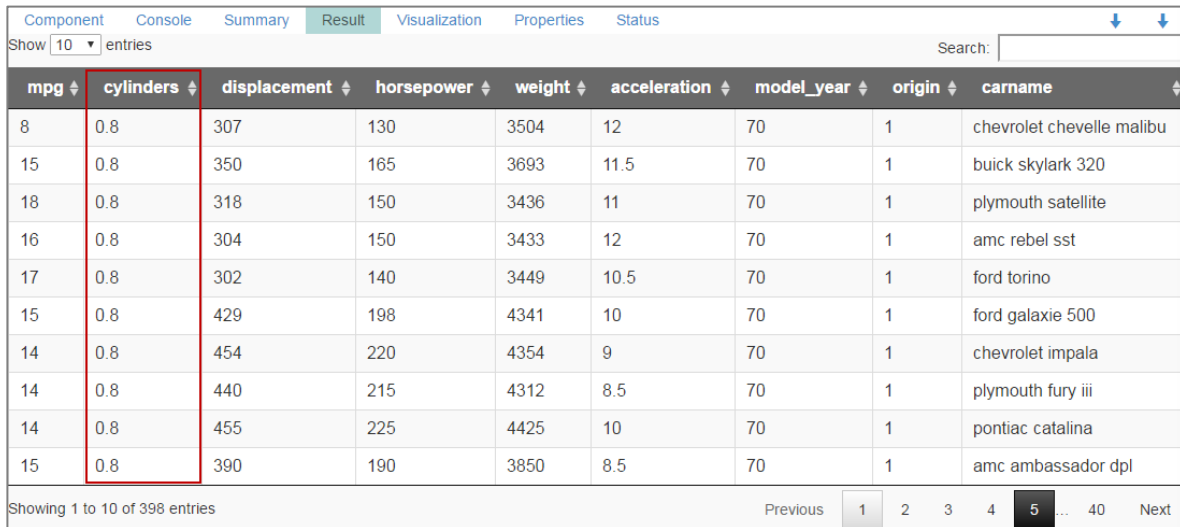
v) Click **'Apply'** to configure the fields:



The screenshot shows a configuration panel with tabs for Component, Console, Summary, Result, Visualization, Properties, and Status. The 'Properties' tab is active, showing 'Column Selection' set to 'cylinders' and 'Behavior' set to 'Decimal Scaling'. An 'Apply' button is visible at the bottom right.

vi) Click **'Run'** or **'Run Till Here'**.

vii) Users will be directed to the **'Result'** tab displaying the result list view.



The screenshot shows the 'Result' tab with a data table. The 'cylinders' column is highlighted with a red box, showing a value of 0.8 for all rows. The table includes a search bar and pagination controls at the bottom.

mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	carname
8	0.8	307	130	3504	12	70	1	chevrolet chevelle malibu
15	0.8	350	165	3693	11.5	70	1	buick skylark 320
18	0.8	318	150	3436	11	70	1	plymouth satellite
16	0.8	304	150	3433	12	70	1	amc rebel sst
17	0.8	302	140	3449	10.5	70	1	ford torino
15	0.8	429	198	4341	10	70	1	ford galaxie 500
14	0.8	454	220	4354	9	70	1	chevrolet impala
14	0.8	440	215	4312	8.5	70	1	plymouth fury iii
14	0.8	455	225	4425	10	70	1	pontiac catalina
15	0.8	390	190	3850	8.5	70	1	amc ambassador dpl

Note:

1. Normalization displays columns containing only numerical data.
2. **'New Maximum Value'** must be greater than **'New Minimum Value'**.

6.5. Sample

This component can be used to select subsection of data from a large data set.

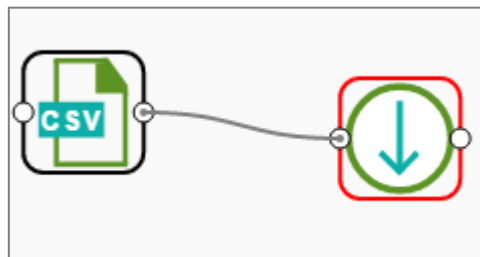
The following sample types are supported by the Sample component:

- **Sampling Methods**

1. **First N:** It will select first N records from the datasource. E.g. If selected value for “N” is 10, then it will select first 10 records from the data.
2. **Last N:** It will select last N records from the datasource. E.g. If selected value for “N” is 5, then it will select last 5 records from the data.
3. **Every Nth:** It will select every Nth record from the datasource, where in “N” indicates an interval. E.g. If N=3, then 3rd, 6th, and 9th records will be selected from the data.
4. **Simple Random:** It will select records randomly as per the value of “N” or percentage mentioned for “N” from the datasource. E.g. If selected value for “N” is 4 then, it will select randomly any 4 records from the datasource. If selected value for “N” is 4% then, it will select 4% records from the datasource.
5. **Systematic Random:** It will select data based on the bucket size. E.g. If selected value for bucket is 2 then, it will select 1st, 3rd, 5th records or 2nd, 4th, 6th records from the datasource.

- **Applying Sampling**

- i) Select and drag ‘**Sample**’ component onto the workspace.
- ii) Connect the ‘**Sample**’ component to a configured datasource.
- iii) Click the ‘**Sample**’ component.



- iv) Configure the required component fields:

Properties

a. Sampling Information

- i. **Sampling Type:** Select an option from the drop-down menu



- ii. **Limit Rows by:** Select an option from the drop-down menu. This field will offer two options as described below:
 - 1. **Numbers of Rows:** By selecting this option, it will display a new field 'Number of Rows'.
 - 2. **Percentage of Rows:** By selecting this option, it will display new field 'Percentage of Rows'.
- b. Sample Size Limit**
 - i. **Maximum Rows:** The maximum number of rows that can be viewed in the 'Result' tab (It is an optional field).
- v) Click 'Apply'.
- vi) Click 'Run' or 'Run Till Here'.
- vii) Users will be directed to the 'Result' tab displaying the result list view based on the selected Sampling Type.
- viii) Check out the following properties tab(s) and result list view(s) for various Sampling options:

1. First N (Where 'N' is 1 number of row)

Component	Console	Summary	Result	Visualization	Properties	Status
General	Sampling Information					
Properties	Sampling Type	First N				
	Limit Rows by	Number of Rows				
	Number of Rows	1				
	Sample Size Limit					
	Maximum Rows	optional				
						Apply

Component	Console	Summary	Result	Visualization	Properties	Status			
Show	10	entries	Search:						
mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	carname	
8	8	307	130	3504	12	70	1	chevrolet chevelle malibu	
Showing 1 to 1 of 1 entries							Previous	1	Next

2. Last N ('N' is 5% and maximum rows are 6)



Component	Console	Summary	Result	Visualization	Properties	Status
General	Sampling Information					
Properties	Sampling Type	First N				
	Limit Rows by	Percentage of Rows				
	Percentage of Rows	5				<i>i</i>
	Sample Size Limit					
	Maximum Rows	6				
						Apply

Component	Console	Summary	Result	Visualization	Properties	Status		
Show <input type="text" value="10"/> entries								
Search: <input type="text"/>								
mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	carname
27	4	151	90	2950	17.3	82	1	chevrolet camaro
27	4	140	86	2790	15.6	82	1	ford mustang gl
44	4	97	52	2130	24.6	82	2	vw pickup
32	4	135	84	2295	11.6	82	1	dodge rampage
28	4	120	79	2625	18.6	82	1	ford ranger
31	4	119	82	2720	19.4	82	1	chevy s-10
Showing 1 to 6 of 6 entries								
						Previous	<input type="text" value="1"/>	Next

3. Every Nth (Interval is 3 and maximum rows are 7)

Component	Console	Summary	Result	Visualization	Properties	Status
General	Sampling Information					
Properties	Sampling Type	Every Nth				
	Step Size	3				
	Sample Size Limit					
	Maximum Rows	7				
						Apply



Component	Console	Summary	Result	Visualization	Properties	Status			
Show <input type="text" value="10"/> entries							Search: <input type="text"/>		
mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	carname	
18	8	318	150	3436	11	70	1	plymouth satellite	
15	8	429	198	4341	10	70	1	ford galaxie 500	
14	8	455	225	4425	10	70	1	pontiac catalina	
14	8	340	160	3609	8	70	1	plymouth 'cuda 340	
24	4	113	95	2372	15	70	3	toyota corona mark ii	
21	6	200	85	2587	16	70	1	ford maverick	
25	4	110	87	2672	17.5	70	2	peugeot 504	
Showing 1 to 7 of 7 entries							Previous	1	Next

4. Simple Random (Number of Rows selected are 3). Randomly selected any 3 rows will be displayed.

Component	Console	Summary	Result	Visualization	Properties	Status
General	Sampling Information					
Properties	Sampling Type: <input type="text" value="Simple Random"/>					
	Limit Rows by: <input type="text" value="Number of Rows"/>					
	Number of Rows: <input type="text" value="3"/>					
	Sample Size Limit					
	Maximum Rows: <input type="text" value="optional"/>					
						<input type="button" value="Apply"/>

Component	Console	Summary	Result	Visualization	Properties	Status			
Show <input type="text" value="10"/> entries							Search: <input type="text"/>		
mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	carname	
32	4	71	65	1836	21	74	3	toyota corolla 1200	
17.7	6	231	165	3445	13.4	78	1	buick regal sport coupe (turbo)	
32	4	135	84	2295	11.6	82	1	dodge rampage	
Showing 1 to 3 of 3 entries							Previous	1	Next

5. Systematic Random (Bucket Size is 3).



Component	Console	Summary	Result	Visualization	Properties	Status
General	Sampling Information					
Properties	Sampling Type	Systematic Random ▾				
	Bucket Size	3				
	Sample Size Limit					
	Maximum Rows	optional				
						Apply

Component	Console	Summary	Result	Visualization	Properties	Status			
Show 10 entries									
Search: <input type="text"/>									
mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	carname	
32	4	71	65	1836	21	74	3	toyota corolla 1200	
17.7	6	231	165	3445	13.4	78	1	buick regal sport coupe (turbo)	
32	4	135	84	2295	11.6	82	1	dodge rampage	
Showing 1 to 3 of 3 entries							Previous	1	Next

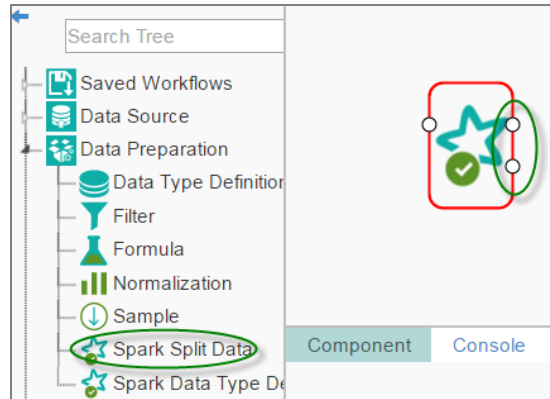
Note: Current document covers steps to deal with a CSV File data set. The similar steps can be followed for a Data Service data set.

6.6. Spark Split Data

The Spark Split Data component is used to split a dataset into training and testing data sets. Once the most suitable model is determined from the trained data, users can pass test data to that model.

Spark Split Data appears as a leaf node under the Data Preparation Tree node.

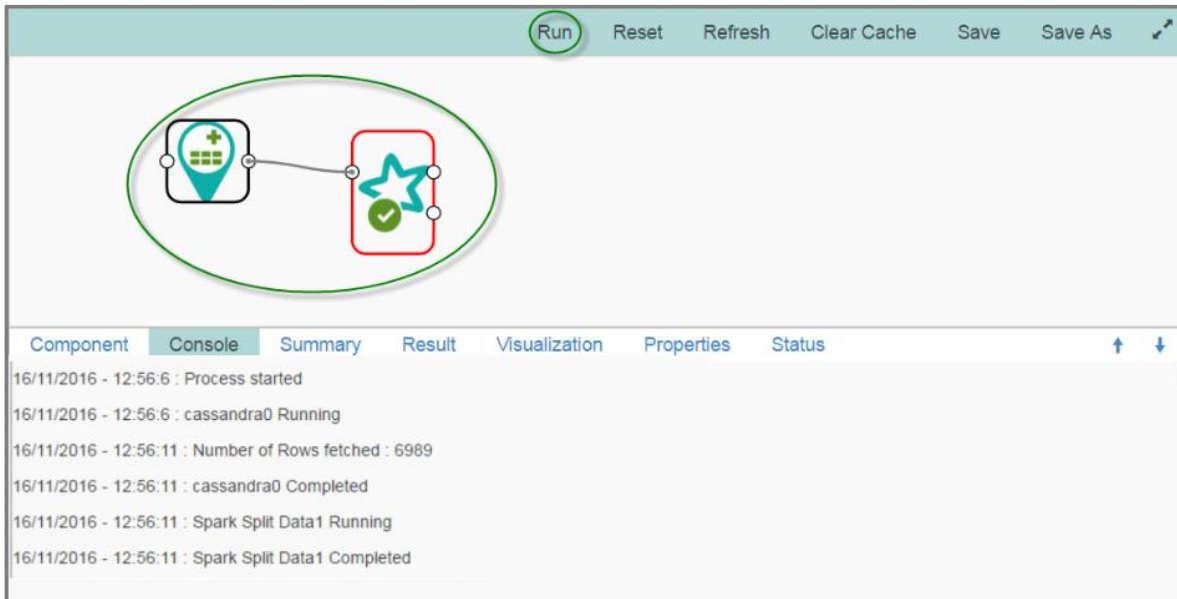
The Spark Split Data consists of two connector nodes: Upper node for the **training data set** and lower node for the **testing data set**.



- i) Select the '**Spark Split Data**' component and connect it with a valid data source (in this case, select Cassandra reader).
- ii) Click the '**Spark Split Data**' component on the workspace
- iii) Users will be directed to the Properties fields provided under the '**Components**' tab
- iv) Configure the following Properties:
 - a. Relative (Train): Enter value to decide ratio of train data out of the data set (Type: Decimal, Range: 0-1 and sum of train and test should be 1).
 - b. Relative (Test): Enter value to decide ratio of train data out of the data set (Type: Decimal, Range: 0-1 and sum of train and test should be 1).
 - c. Seeds: Enter a numerical value. Default Value: 10. It is an optional field. Set the seed of Spark's random number generator, which is useful for creating stimulations or random objects that can be reproduced. The random numbers are the same, and they would continue to be the same irrespective of how far in the sequence the users go. Use the seed function when running simulations to ensure all results, figures, etc. are reproducible.
- v) Click '**Apply**'.

Component	Console	Summary	Result	Visualization	Properties	Status
General	Choose size of first partition					
Properties	Relative(train)	<input type="text" value="0.3"/>				
	Relative(test)	<input type="text" value="0.7"/>				
	Seeds	<input type="text" value="1"/>				
						Apply

vi) Click 'Run' to view the console process.



vii) Click the 'Spark Split' component on the workspace

viii) Click the 'Result' tab.

ix) Users will be directed to the 'Result' tab to view the results.

The Result tab will contain two data sets separated by a sub-tab. As shown in the below given images:

a. Select the 'Split 1' tab to see one set of data (the training data set).

Component Console Summary **Result** Visualization Properties Status

Split 1 Split 2

Show 10 entries Search:

binarycolumn	PetalLength	PetalWidth	SepalLength	SepalWidth	Species
0	4.3	1.3	6.2	2.9	versicolor
0	4.9	1.8	6.3	2.7	virginica
0	6.3	1.8	7.3	2.9	virginica
1	1.5	0.4	5.7	4.4	setosa
1	1.9	0.2	4.8	3.4	setosa
0	4.2	1.3	5.6	2.7	versicolor
0	5.1	1.9	5.8	2.7	virginica
1	1.3	0.2	4.7	3.2	setosa
0	3.5	1	5.7	2.6	versicolor
0	4.7	1.4	7	3.2	versicolor

Showing 1 to 10 of 49 entries Previous 1 2 3 4 5 Next

b. Select the **'Split 2'** tab to see another set of data (the testing data set).

Component	Console	Summary	Result	Visualization	Properties	Status
Split 1	Split 2					
Show 10 entries		Search: <input type="text"/>				
binarycolumn	PetalLength	PetalWidth	SepalLength	SepalWidth	Species	
0	4.7	1.5	6.7	3.1	versicolor	
1	1.1	0.1	4.3	3	setosa	
1	1.4	0.2	5.1	3.5	setosa	
1	1.7	0.2	5.4	3.4	setosa	
1	1.6	0.2	4.7	3.2	setosa	
0	4.2	1.2	5.7	3	versicolor	
0	4.8	1.8	6	3	virginica	
0	5.2	2.3	6.7	3	virginica	
1	1.4	0.2	5.5	4.2	setosa	
1	1.9	0.4	5.1	3.8	setosa	
Showing 1 to 10 of 101 entries		Previous 1 2 3 4 5 ... 11 Next				

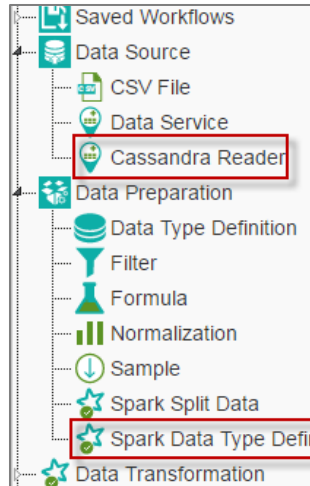
Note:


- a. Users need to click the Spark component and then click the **'Result'** tab to display result view for any Spark Component.
- b. Only Cassandra reader is supported as data source.

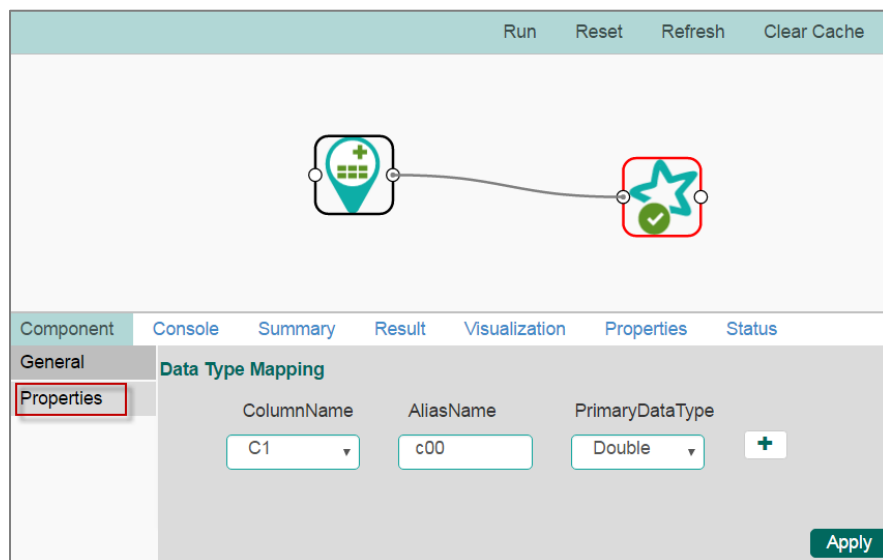
6.7. Spark Data Type Definition

This component can be used to type cast data into another form. Users can change the data type of a column, or change the alias name of the column using this component. Spark Data Type definition will appear as a leaf node under the Data Preparation tree node.

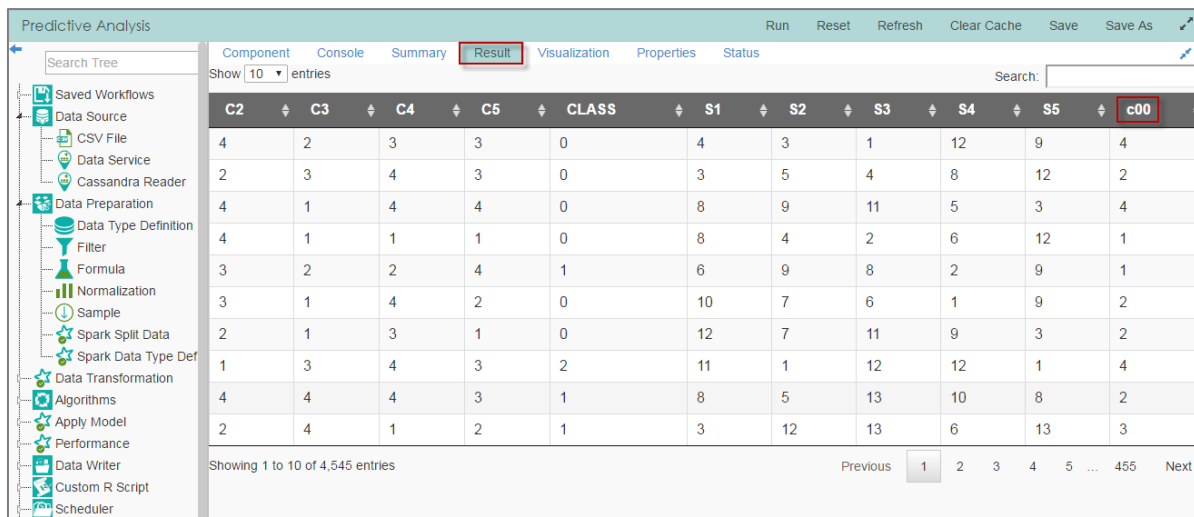
- i) Select the **'Spark Data Type Definition'** component and connect it with a valid data source (in this case, select Cassandra Reader as the data source).



- ii) Configure the Properties fields for the Spark Data Type Definition component
- iii) Configure the following ‘Data Type Transformation’ details:
 - a. **Column Name:** Select a column name which you want to change
 - b. **Alias Name:** Enter an alias name for the required source column
 - c. **Primary Data Type:** Select a primary data type column that you want to change.
 - d. **‘Add’ option**  : Click on this button to add more columns to be transformed.
- iv) Click ‘Apply’.
- v) Click ‘Run’.



vi) Select the 'Result' tab to view the results.



C2	C3	C4	C5	CLASS	S1	S2	S3	S4	S5	c00
4	2	3	3	0	4	3	1	12	9	4
2	3	4	3	0	3	5	4	8	12	2
4	1	4	4	0	8	9	11	5	3	4
4	1	1	1	0	8	4	2	6	12	1
3	2	2	4	1	6	9	8	2	9	1
3	1	4	2	0	10	7	6	1	9	2
2	1	3	1	0	12	7	11	9	3	2
1	3	4	3	2	11	1	12	12	1	4
4	4	4	3	1	8	5	13	10	8	2
2	4	1	2	1	3	12	13	6	13	3

Note:

- Users cannot typecast the advanced column types (E.g. map, list, UDT), UUID, and timestamp.
- Only Integer, Double, and String data types are supported by the Spark Data Type Definition.

7. Data Transformation

7.1. String Indexer

String Indexer converts a string column of labels to a column of label indices. The indices are in [0, numLabels), ordered by label frequencies, so the most frequent label gets index 0. If the input column is numeric, we will cast it to string and index the string values.

Users must set the input column of the component to this string-indexed column name, when pipeline components such as Estimator or Transformer make use of this string-indexed label. Users can set the input column with setInputCol.

- Users need to select the String Indexer component and connect it with a valid data source.
- Configure the required component fields for the String Indexer.
 - The Properties tab for Spark Indexer contains an option to select 'Label Column' from previous component headers on which a new column was created.
 - Users can rename the created label column using the 'Label Column



Name'.

The screenshot shows a workflow editor with a search tree on the left. The 'String Indexer' component is highlighted in red in the search tree and also in the workflow diagram. The properties panel for the String Indexer is open, showing the 'Advanced' tab. The 'Label Column' is set to 'Species' and the 'Label Column Name' is set to 'Labels'. There is an 'Apply' button at the bottom right of the properties panel.

- c. The String Indexer, when applied on one dataset, will handle unseen labels using either of the following methods:
- d. Users are provided with two options in the '**Advanced**' tab to handle the unseen labels.
 - i. Error: The unseen labels will be thrown as exception. (by default)
 - ii. Skip: The rows containing the unseen labels will be skipped.

The screenshot shows the 'Input Data Handling' properties panel for the String Indexer. The 'Missing values' dropdown is set to '1 checked'. There is an 'Apply' button at the bottom right of the properties panel.

Note:

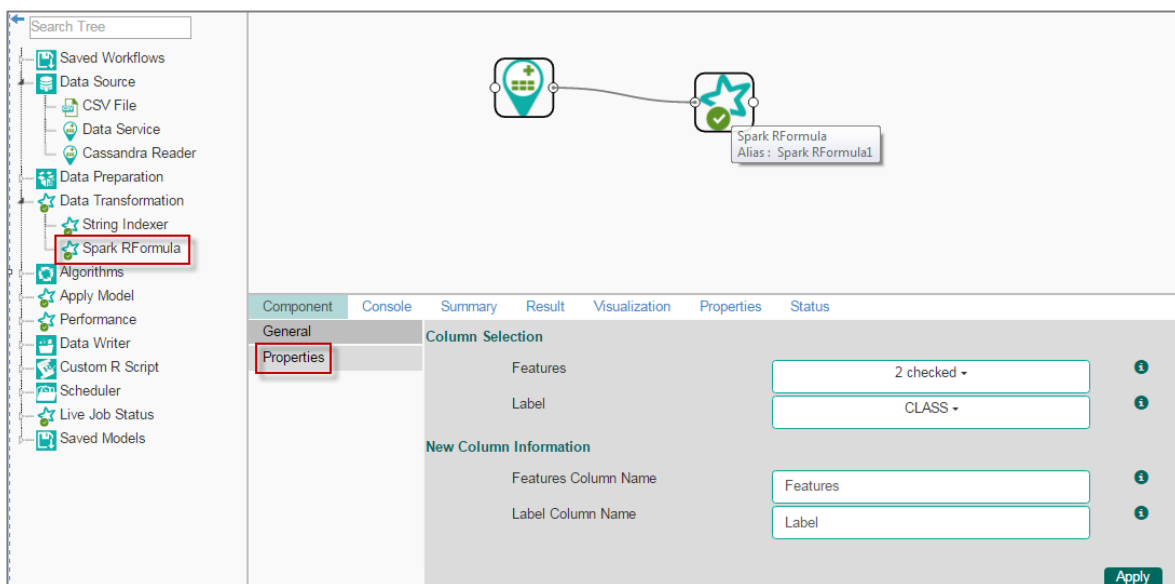
- a. The String Indexer can also connect to the Data Preparation components with the prefix '**Spark**'. (E.g. the Spark Data Type Definition and Spark Split Data).
- b. Since the String Indexer is a pipeline component, the result can be viewed only after connecting to an '**Apply Model**' component.

- c. The **'Data Preparation'** components cannot be added in between pipeline components in a workflow.
- d. End of the pipeline component should be an **'Apply Model'** component.
- e. A model can be saved from the context menu of an **'Apply Model'** component.

7.2. Spark R Formula

The Spark R Formula can be used to produce a vector column of features and a double column of labels.

- i) Users need to select the Spark R Formula component and connect it with a valid data source.
- ii) Select the desired features and labels from the column headers provided under the Properties tab.
- iii) Configure the **'New Column Information'** fields.
- iv) Click **'Apply'**.



Note:

- a. Spark R Formula can also connect to the Data Preparation component with the pre-fix **'Spark'** such as the Spark Split Data and Spark Data Type Definition.
- b. Users can change the column name by changing the New Column Information values.
- c. Since the Spark R Formula is a pipeline component, the results can be viewed only after running the R Formula with an **'Apply Model'** or another pipeline component.
- d. The **'Data Preparation'** components cannot be added in between

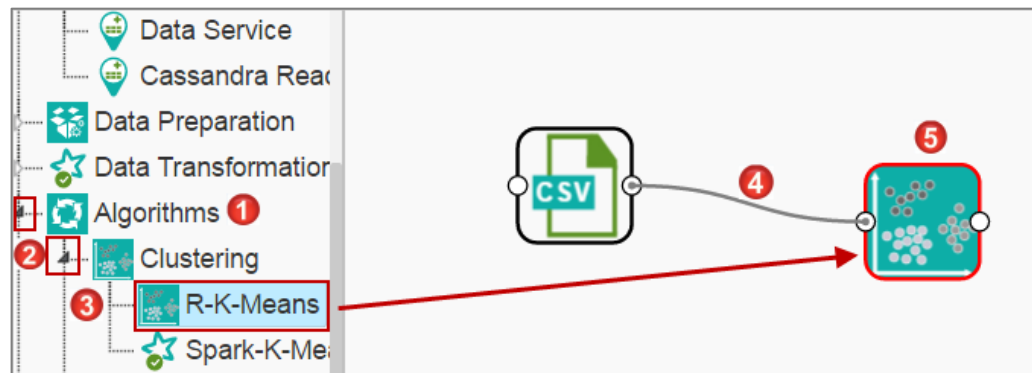
- pipeline components in a workflow.
- e. End of the pipeline component should be an **'Apply Model'** component.
- f. A model can be saved from the context menu of an **'Apply Model'** component.

8. Algorithms

Algorithms are statistical set of rules that help the user analyze large quantities of numerical data and extract appropriate information out of it. BizViz Predictive Analysis allows the user to apply more than one algorithm to manage vast amount of data.

- **Applying an Algorithm to a Data Source:**

- i) Click the **'Algorithms'** tree-node on the Predictive Analysis home page.
- ii) Click the Algorithm Category tree-node to display the available algorithm sub-categories.
- iii) Select and drag an algorithm component onto the workspace.
- iv) Connect the algorithm component to a configured data source.
- v) Click on the algorithm component.



- vi) Configure the required fields for the dragged algorithm component.
- vii) Click **'Apply'** to save the information.



Component Console Summary **Result** Visualization Properties Status

General **Output Information**

Properties Number Of Clusters ⓘ

Advanced **Column Selection**

Features ⓘ **Select atleast one**

New Column Information

Cluster Name ⓘ

Apply

viii) Click **'Run'** or **'Run Till Here'** to display the **'Result'** view .

Untitled **Run** Reset Refresh Clear Cache Save Save As ↗

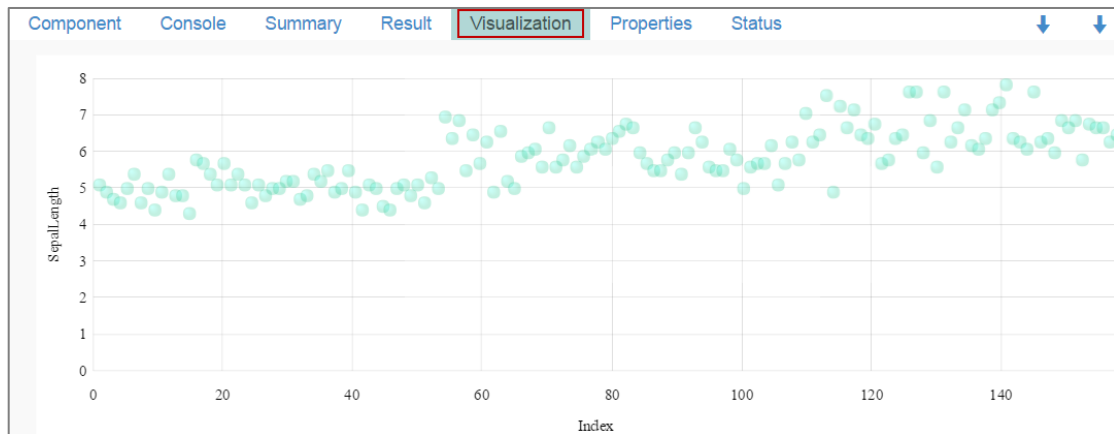
CSV → **Run Till Here** Delete

Component Console Summary **Result** Visualization Properties Status

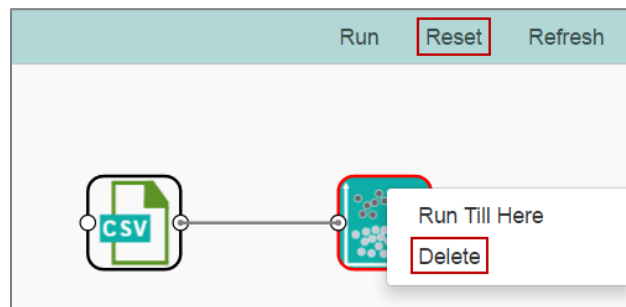
Show 10 entries Search:

SepalLength	SepalWidth	PetalLength	PetalWidth	Species	ClusterNumber
5.1	3.5	1.4	0.2	setosa	5
4.9	3	1.4	0.2	setosa	5
4.7	3.2	1.3	0.2	setosa	3
4.6	3.1	1.5	0.2	setosa	3

ix) Click the **'Visualization'** tab to see graphical representation of the result data.



- x) Click **'Delete'** or **'Reset'** option to remove the selected algorithm component from the workspace.



Note:

- a. Users can follow the above mentioned steps to configure all the available R- algorithms.
- b. Users can configure alias name for the algorithm component via the **'General'** tab.
- c. Basic configuration for all the algorithms is done through the **'Properties'** tab. Users are required to manually configure this tab while applying an algorithm component.
- d. Users can avail all the default values under **'Advanced'** tab. Users can manually set the **'Advanced'** tab, only if the advanced level configuration is required.
- e. After execution, users can click on the respective component to get data. Pipeline component will not have any result set, only summary will be available. Users need to connect the pipeline components with an apply model component and test data set to view result.

8.1. Clustering

Clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters).

8.1.1. R-K Means

K- means clustering is one of the most commonly used clustering method. It clusters data points into a predefined number of clusters. It first clusters observations into 'K' groups, wherein 'K' is an input parameter. The algorithm then assigns each observation to a cluster based on the proximity of the observation.

Applying R-K Means to a Data Source

Users will be redirected to the '**Component**' tabs when applying the '**R-K Means**' algorithm component to a configured data source.


- i) Drag the R-K Means to the Workspace and connect it to the configured data Source.
- ii) The Component tabs will be displayed on the Viewspace.
- iii) Configure the following fields in the '**Properties**' tab:
 - a. **Output Information**
 - i. **Number of Clusters:** Enter number of groups for clustering. The default value for this field is 5. Range should be between 1 and total number of clusters.
 - b. **Column Selections**
 - i. **Feature:** Select the input columns with which you want to perform the Analysis.
 - c. **New Column Information**
 - i. **Cluster Name:** Enter a name for the new column displaying cluster number.



Component	Console	Summary	Result	Visualization	Properties	Status	↓	↓
General	Output Information							
Properties	Number Of Clusters	<input type="text" value="3"/>						
Advanced	Column Selection							
	Features	<input type="text" value="4 checked"/>						
	New Column Information							
	Cluster Name	<input type="text" value="ClusterNumber"/>						
								<input type="button" value="Apply"/>

- **Rules for Naming a New Column**

- Do not use space in the name of a new column. It should be in a single word or two words should be connected by underscore (_). E.g. SampleData or Sample_Data.
- Do not use any special symbol alone or with any character as name of a new column. Eg. %, #, \$, @, * or Sample# are not acceptable.
- Do not use single or double quotes, dot, and brackets to name a new column.
- Do not use numbers alone to name a new column. Numbers can be used with atleast one character of alphabet and the name should not begin with numeral.
- Name given to a new column should not exceed 50 characters.

Note: Click the information icon  provided next to the **'New Column Information'** tab. A list of rules for naming a new column will be displayed.

- Click the **'Advanced'** tab.
 - Configure the required **'Behavior'** fields:
 - Maximum Iterations:** Enter the number of iterations allowed for discovering clusters. (The default value for this field is 100).
 - Number of Initial Centroids:** Enter the number of random initial centroid sets for clustering (The default value for this field is 1).
 - Algorithm type:** Select an algorithm type from the drop-down menu
 - Initial Cluster Center Seed:** Enter a number indicating initial cluster center seed (The default value for this field is 10).

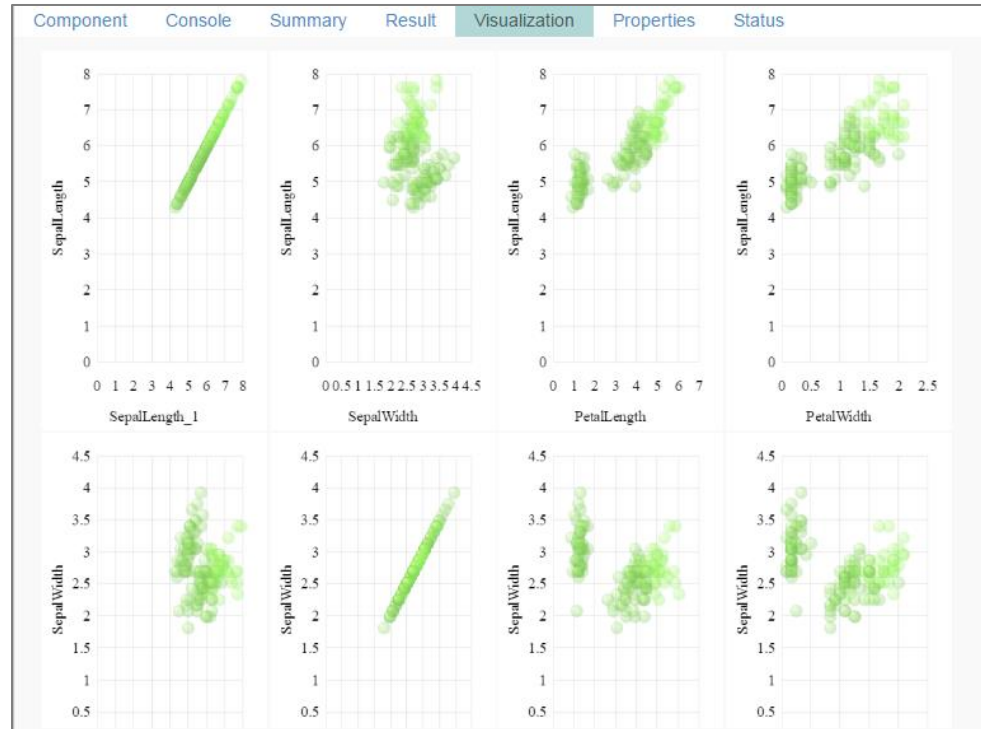


Component	Console	Summary	Result	Visualization	Properties	Status
General	Behavior					
Properties	Maximum Iterations <input type="text" value="100"/>					
Advanced	Number of initial centroids <input type="text" value="1"/>					
	Algorithm Type <input type="text" value="1 checked"/>					
	Initial Cluster Center Seed <input type="text" value="10"/>					
						Apply

- v) Click **'Apply'**.
- vi) Click **'Run'** or **'Run Till Here'**.
- vii) Users will be redirected to the **'Result'** tab.
- viii) A new column **'Cluster Number'** will be displayed in the result view.

SepalLength	SepalWidth	PetalLength	PetalWidth	Species	ClusterNumber
5.1	3.5	1.4	0.2	setosa	2
4.9	3	1.4	0.2	setosa	2
4.7	3.2	1.3	0.2	setosa	2
4.6	3.1	1.5	0.2	setosa	2
5	3.6	1.4	0.2	setosa	2
5.4	3.9	1.7	0.4	setosa	2
4.6	3.4	1.4	0.3	setosa	2
5	3.4	1.5	0.2	setosa	2
4.4	2.9	1.4	0.2	setosa	2
4.9	3.1	1.5	0.1	setosa	2

- ix) Click the **'Visualization'** tab.
- x) The result data will be displayed via the scatter plot matrix charts.

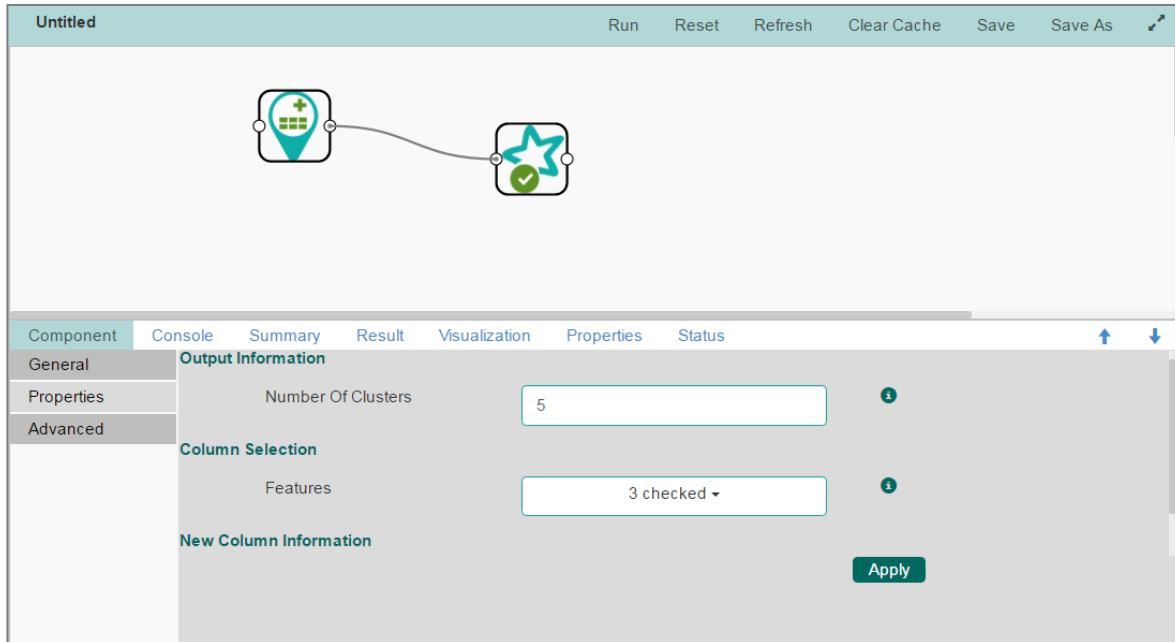


8.1.2. Spark-K- Means

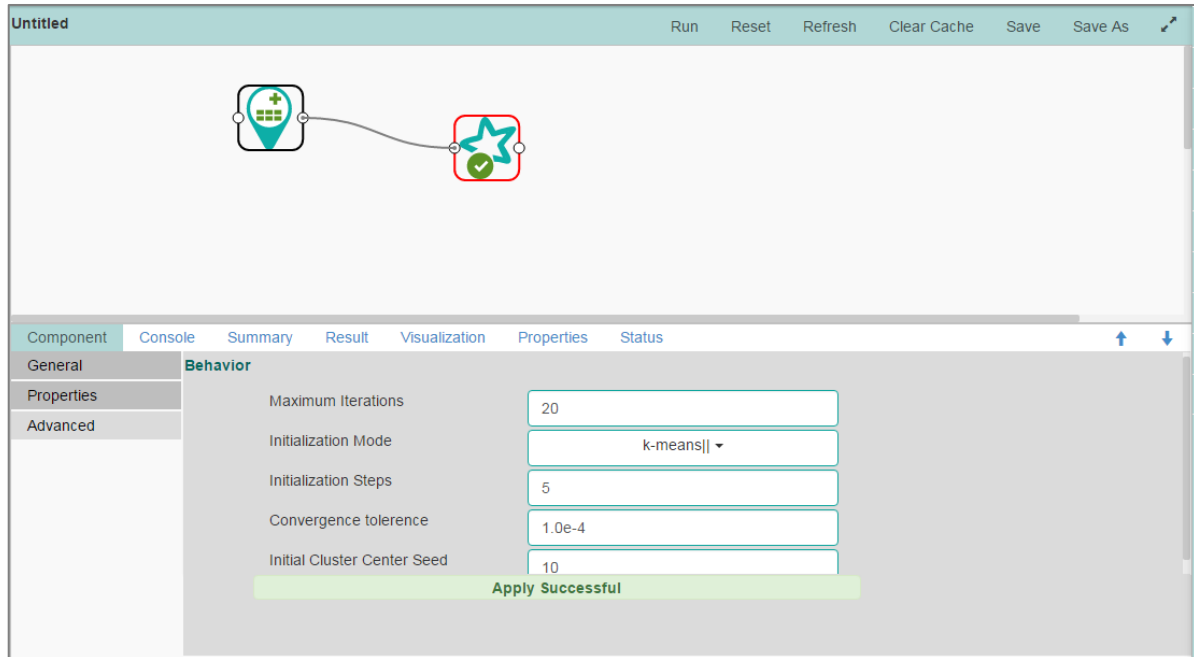
The Spark K-Means algorithm is provided as an option under the clustering algorithm category. The spark.ml implementation includes a parallelized variant of the k-means++ method called `k-means | |`.

Applying Spark-K-Means to a Data Source

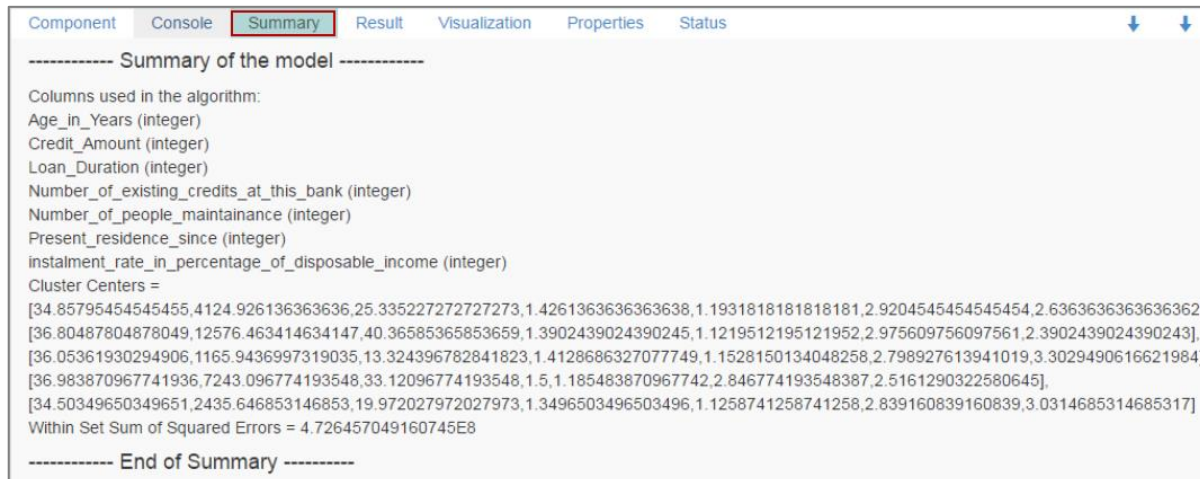
- i) Drag the Spark-K-Means to the workspace and connect to a configured data source.
- ii) Configure the following fields in the 'Properties' tab:
 - a. **Output Information**
 - i. **Number of Clusters:** Enter number of groups for clustering. The default value for this field is 5. Range should be between one and total number of clusters.
 - b. **Column Selections**
 - i. **Feature:** Select the input columns with which you want to perform the Analysis.
 - c. **New Column Information**
 - i. **Cluster Name:** Enter a name for the new column displaying cluster number.



- iii) Select the '**Advanced**' tab.
 - a. Configure the following '**Behavior**' fields:
 - i. **Maximum Iterations**: Enter the number of iterations allowed for discovering clusters (The default value for this field is 20).
 - ii. **Initialization Mode**: Select any one option at the beginning of the algorithm out of: '**Random**' or '**k-means||**' (default).
 - iii. **Initialization Steps**: Set number for the initialization mode as random (The default value for this field is 5).
 - iv. **Convergence Tolerance**: Set tolerance level to include clusters (The default value for this field is in exponential form. (the default value for this field is 1.0e-4).
 - v. **Initial Cluster Center Seed**: Enter a number indicating initial cluster center seed (The default value for this field is 10).



- iv) Click **'Apply'**.
- v) Click **'Run'** to run the execution.
- vi) Click the **'Summary'** tab to display summary of the model.

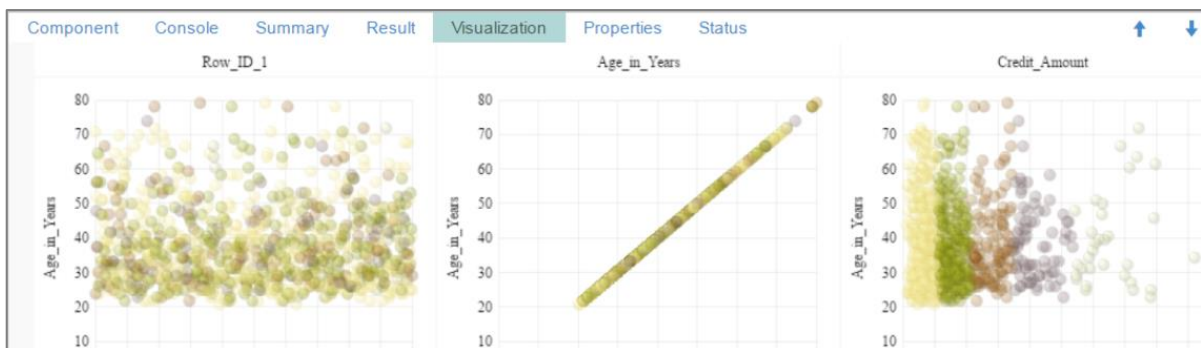


- vii) Click the dragged algorithm component.
- viii) Click the **'Result'** tab.
- ix) A new column **'ClusterNumber'** will be added in the displayed result data.



Savings_account_or_bonds	Telephone	instalment_rate_in_percentage_of_disposable_income	ClusterNumber
>= 1000 DM	yes, registered under the customers name	3	2
500 <= ... <1000 DM	yes, registered under the customers name	4	4
<100 DM	none	4	0
<100 DM	none	2	1
<100 DM	none	4	2

- x) Click the **'Visualization'** tab.
- xi) The result data will be displayed via the scatter plot matrix charts.



8.2. Forecasting

Forecasting is the process of making predictions of the future based on past and present data and analysis of trends. It uses smoothing as a statistical technique to spot trends in a disorderly data. It can also compare trends between two or more variable time series.

There are four sub types provided under **'Forecasting'**:

8.2.1. Triple Exponential Smoothing

- i) Configure the following fields in the **'Properties'** tab:
 - a. **Output Information**



- i. **Output Mode:** Select a mode in which you want to display output Data
 1. **Trend:** Selecting this option will display source data along with predicted values for the given dataset. A new column '**Predicted Values**' will be added in the result view when '**Trend**' output mode has been selected.
 2. **Forecast:** Selecting this option will display forecasted values for the given time period. Results will be appended to the target column, when '**Forecast**' output mode has been selected.
 - ii. **Period to Forecast:** This field appears only when the selected '**Output Mode**' option is '**Forecast**'
 - iii. **Select Output Columns:** Select a column that you want to display in output (Select at least one column using a tick mark)
- b. Column Selection**
- i. **Target Variable:** Select the target variable for which you want to apply forecasting analysis (First selected option gets selected by default. Only numerical columns are accepted.)
- c. Input Data Handling**
- i. **Period:** Select period of forecasting by choosing any one option from the drop-down menu.

Quarter
Month
✓ Custom

- ii. **Period Per Year:** This field appears only when selected '**Period**' option is '**Custom**'.
 - iii. **Start Period:** Enter a value between 1 and the value specified for the selected option for '**Period**' field
 - iv. **Start Year:** Enter a year from which you want the data entries to be considered. Enter four digit value for selecting a year (E.g. 2000)
- d. New Column Information**
- i. **Predicted Column Name:** Enter a name for the column containing predicted values
 - ii. **Year Values:** Enter a name for the column containing year value
 - iii. **Period Values:** Enter a name for the column containing period value (This field will change into '**Month Value**' , if the selected value for '**Period**' field is '**Month**'.)



Output Information

Output Mode: 1 checked ▾

Period To Forecast: 1

Select Output Columns: 2 checked ▾ ⓘ

Column Selection

Target Variable: 1 checked ▾ ⓘ

Input Data Handling

Period: 1 checked ▾

Periods per year: 1

Start Period: 1

Start Year: 2000

New Column Information

Predicted Column Name: PredictedValues ⓘ

Year Values: Year ⓘ

Period Values: Period ⓘ

Apply

Note:

- a. **'Output Information'** tab will display **'Period to Forecast'** field, only when **'Forecast'** option is selected from the **'Output Mode'** drop-down menu.
- b. **'New Column Information'** displays the below mentioned column names for the period value column based on the **'Period'** option selected from the **'Input Data Handling'** section.

Selected 'Period' option	'New Column Information' Field
Quarter	Quarter Values
Month	Month Values
Custom	Period Values

- ii) Click the **'Advanced'** tab and configure if required:
 - a. Configure the following **'Behavior'** fields:
 - i. **Alpha:** Enter a valid double value in the given field for smoothing



- observations. (Alpha Range: $0 < \alpha \leq 1$.)
 - ii. **Beta:** Enter a valid double value in the given field for finding trend parameters. (Beta Range: 0-1.)
 - iii. **Gamma:** Enter a valid double value in the given field for finding seasonal trend parameters. (Gamma Range: 0-1.)
 - iv. **Seasonal:** Select a smoothing algorithm type from the dropdown list (Holtwinter’s Exponential Smoothing algorithm)
 - v. **No. of Periodic Observation:** Enter the number of periodic observations required to start the calculation. The default value for this field is 2.
- b. Configure the following ‘Initial Values’ information:**
- i. **Level:** Enter the initial value for level. It is an optional field.
 - ii. **Trend:** Enter the initial value for finding trend parameters. It is an optional field.
 - iii. **Season:** Enter initial values for finding seasonal parameters. It will depend on the selected column. It is an optional field.
 - iv. **Optimizer Inputs:** Enter the initial values given for alpha, beta, gamma required for the optimizer. It is an optional field.

General	Behavior		
Properties	Alpha	<input type="text" value=".3"/>	
Advanced	Beta	<input type="text" value=".1"/>	
	Gamma	<input type="text" value=".1"/>	
	Seasonal	<input type="text" value="1 checked"/>	
	No: of Periodic Observation	<input type="text" value="2"/>	
	Initial Values		
	Level	<input type="text" value="Optional"/>	
	Trend	<input type="text" value="Optional"/>	
	Season	<input type="text" value="Optional"/>	
	Optimizer Inputs	<input type="text" value="Optional"/>	
			<input type="button" value="Apply"/>

- iii) Click **‘Apply’**.
- iv) Click **‘Run’** or **‘Run Till Here’**.

v) Users will be redirected to the **'Result'** tab. (In this case, the selected output mode is **'Forecasting'**.)

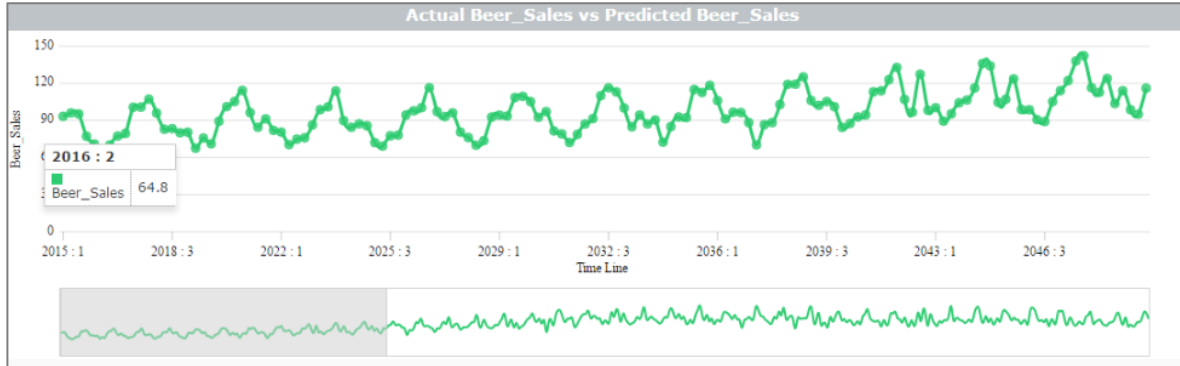
Year	Quarter	Beer_Sales
2015	1	93.2
2015	2	96
2015	3	95.2
2015	4	77.1
2016	1	70.9
2016	2	64.8
2016	3	70.1
2016	4	77.3
2017	1	79.5
2017	2	100.6

Showing 1 to 10 of 469 entries

Previous 1 2 3 4 5 ... 47 Next

vi) Click the **'Visualization'** tab.

vii) The result data will be displayed via the time series chart.



Note:

- 'Properties'** tab is displayed by default while opening all the provided algorithms types, but it actually appears after **'General'** tab. Hence, to maintain the sequence **'General'** tab is explained before **'Properties'** in this document.
- 'Properties'** and **'General'** sections remain the same for all the sub algorithms provided under **'Forecasting'**.
- Some fields provided under **'Advanced'** tab differs for algorithm sub-types. Hence, **'Advanced'** fields are explained below for all the sub-algorithms provided under **'Forecasting'**.

- d. Predicted values will be appended to the target column in the result view for all the **'Forecasting'** algorithms.

8.2.2. Single Exponential Smoothing

- i) Click the **'Advanced'** tab and configure if required:
 - a. Configure the following **'Behavior'** fields:
 - i. **Alpha:** Enter a valid double value in the given field for smoothing observations. Alpha Range: $0 < \alpha \leq 1$.
 - ii. **No. of Periodic Observation:** Enter the number of periodic observations required to start the calculation. The default value for this field is 2.
 - b. Configure the following **'Initial Values'** information:
 - i. **Level:** Enter the initial value for level. It is an optional field.

- ii) Click **'Apply'**.
- iii) Click **'Run'** or **'Run Till Here'**.
- iv) Users will be redirected to the **'Result'** tab.
- v) Predicted values will be appended to the target column in the result data (The selected output mode is **'Forecasting'**).

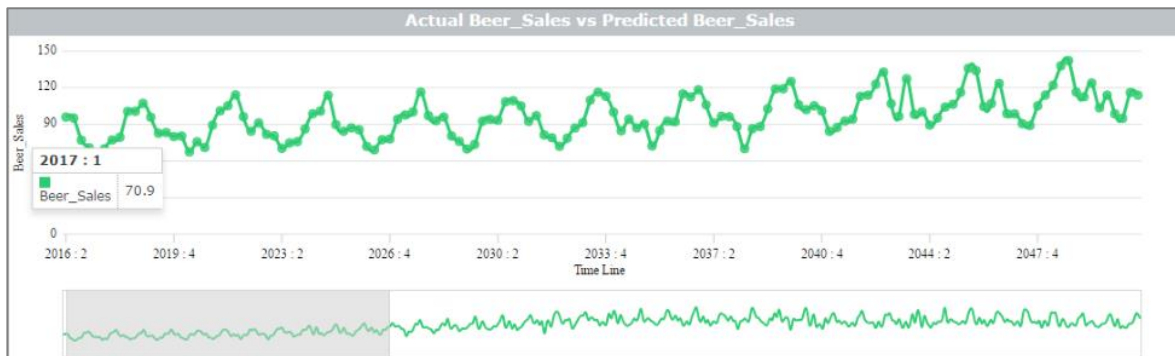


Show 10 entries Search:

Year	Month	Beer_Sales
2016	9	89.3
2016	10	101.1
2016	11	105.2
2016	12	114.1
2017	1	96.3
2017	2	84.4
2017	3	91.2
2017	4	81.9
2017	5	80.5
2017	6	70.4

Showing 21 to 30 of 469 entries Previous 1 2 3 4 5 ... 47 Next




- vi) Click the 'Visualization' tab.
- vii) The result data will be displayed via the time series chart.



8.2.3. Double Exponential Smoothing

- i) Click the 'Advanced' tab and configure if required:
 - a. Configure the following 'Behavior' fields:
 - i. **Alpha:** Enter a valid double value in the given field for smoothing observations. Alpha Range: $0 < \alpha \leq 1$.
 - ii. **Beta:** Enter a valid double value in the given field for smoothing observations. Beta Range: 0-1.
 - iii. **No. of Periodic Observation:** Enter the number of periodic observations required to start the calculation. The default value for this field is 2.
 - b. Configure the following 'Initial Values' information:
 - i. **Level:** Enter the initial value for level. (It is an optional field.)

- ii. **Trend:** Enter the initial value for finding trend parameters. (It is an optional field.)
- iii. **Optimizer Inputs:** Enter the initial values given for alpha and beta required for the optimizer. (It is an optional field.)

General	Behavior		
Properties	Alpha	<input type="text" value=".3"/>	
Advanced	Beta	<input type="text" value=".1"/>	
	No. of Periodic Observation	<input type="text" value="2"/>	
	Initial Values		
	Level	<input type="text" value="Optional"/>	
	Trend	<input type="text" value="Optional"/>	
	Optimizer Inputs	<input type="text" value="Optional"/>	

- ii) Click **'Apply'**.
- iii) Click **'Run'** or **'Run Till Here'**.
- iv) Users will be redirected to the **'Result'** tab.
- v) Predicted values will be appended to the target column in the result data (The selected output mode is **'Forecasting'**).

Show entries Search:

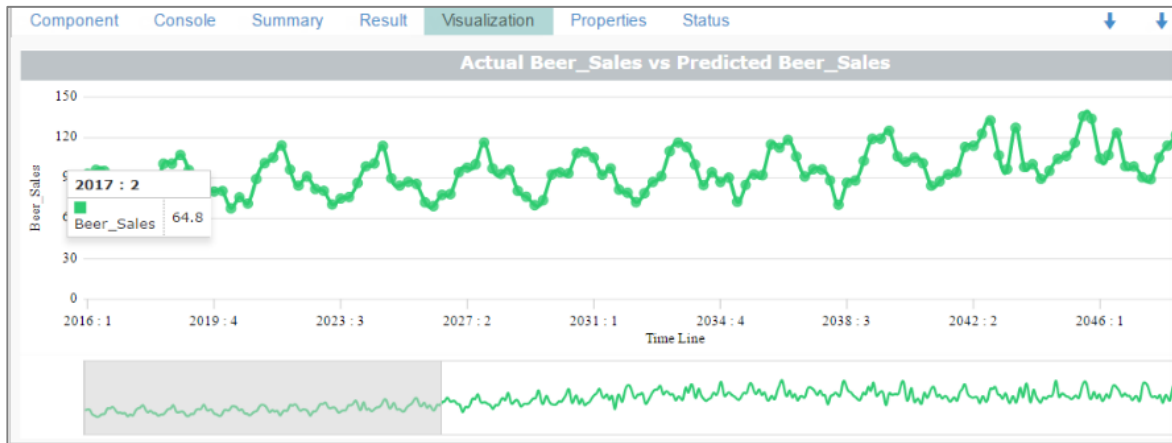
Year	Quarter	Beer_Sales
2016	1	93.2
2016	2	96
2016	3	95.2
2016	4	77.1
2017	1	70.9
2017	2	64.8
2017	3	70.1
2017	4	77.3
2018	1	79.5
2018	2	100.6

Showing 1 to 10 of 469 entries Previous 2 3 4 5 ... 47 Next

- vi) Click the **'Visualization'** tab.



vii) The result data will be displayed via the time series chart.



8.2.4. R-Auto ARIMA

- i) Click **'Apply'** to configure the required details.
- ii) Click **'Run'** or **'Run Till Here'**.
- iii) Users will be redirected to the **'Result'** tab.
- iv) Predicted values will be appended to the target column in the result data (the selected output mode is **'Forecasting'**).

Year	Month	Beer_Sales
2016	11	100.7
2016	12	107.1
2017	1	95.9
2017	2	82.8
2017	3	83.3
2017	4	80
2017	5	80.4
2017	6	67.5
2017	7	75.7
2017	8	71.1

Showing 11 to 20 of 469 entries

Previous 1 2 3 4 5 ... 47 Next

- v) Click the **'Visualization'** tab.
- vi) The result data will be displayed via the time series chart.



Note: The 'R-Auto ARIMA' does not contain the 'Advanced' tab .

8.2.5. R- Auto Forecasting

- i) Click the 'Advanced' tab and configure if required:
 - a. Configure the following 'Behavior' fields:
 - i. **Seasonal:** Select a smoothing algorithm type from the drop-down menu (Holtwinter's Exponential Smoothing algorithm)
 - ii. **No. of Periodic Observation:** Enter the number of periodic observations required to start the calculation. The default value for this field is 2.
 - b. Configure the following 'Initial Values' fields:
 - i. **Level:** Enter the initial value for level. (It is an optional field.)
 - ii. **Trend:** Enter the initial value for finding trend parameters. (It is an optional field.)
 - iii. **Season:** Enter initial values for finding seasonal parameters. It will depend on the selected column. It is an optional field.
 - iv. **Optimizer Inputs:** Enter the initial values given for alpha and beta required for the optimizer. (It is an optional field.)

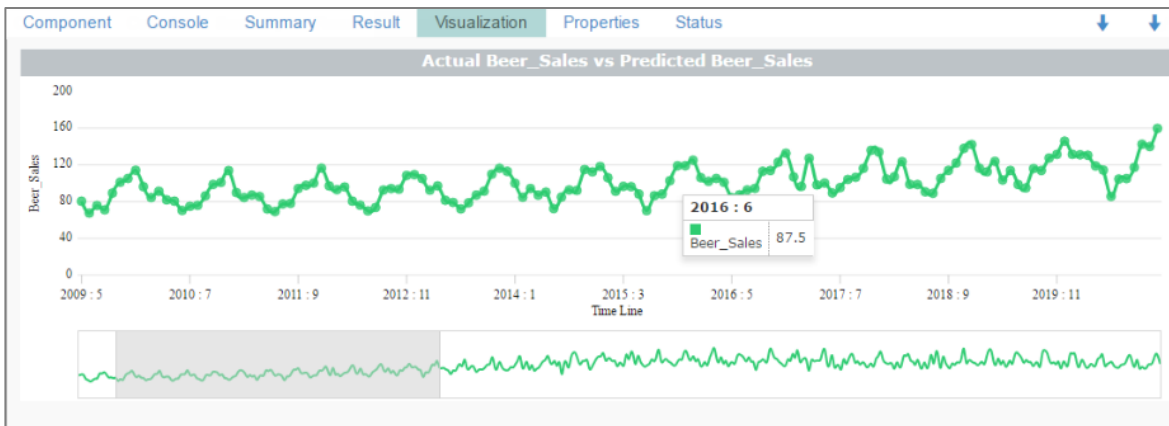


General	Behavior	
Properties	Seasonal	1 checked ▾
Advanced	No: of Periodic Observation	2
	Initial Values	
	Level	Optional
	Trend	Optional
	Season	Optional
	Optimizer Inputs	Optional
		Apply

- ii) Click **'Apply'**.
- iii) Click **'Run'** or **'Run Till Here'**.
- iv) Users will be redirected to the **'Result'** tab.
- v) Predicted values will be appended to the target column in the result data (The selected output mode is **'Forecasting'**).

Year	Month	Beer_Sales
2016	5	84.3
2016	6	87.5
2016	7	92.7
2016	8	94.4
2016	9	113
2016	10	113.9
2016	11	122.9
2016	12	132.7
2017	1	106.9
2017	2	96.6

- vi) Click the **'Visualization'** tab.
- vii) The result data will be displayed via the time series chart.



8.2.6. Result View of Forecasting Algorithms when the selected output mode is 'Trend':

A new column 'Predicted Values' will be added to the result view when 'Trend' is selected as output mode.

1. Triple Exponential Smoothing

- i) Select 'Trend' option from the 'Output Mode' drop-down menu.
- ii) Fill in the required fields.
- iii) Click 'Apply'.
- iv) Click 'Run' or 'Run Till Here'.
- v) Users will be redirected to the 'Result' tab.
- vi) A new column 'PredictedValues' will be added in the result data.

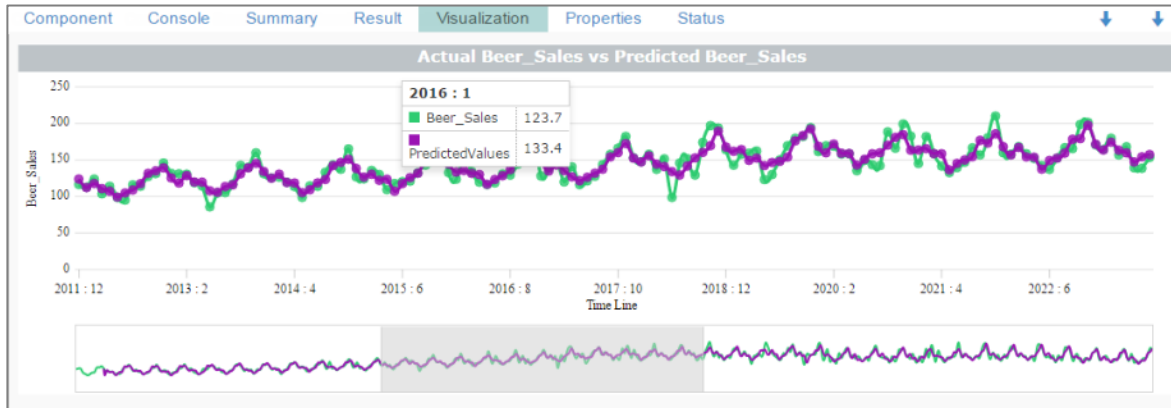
Component	Console	Summary	Result	Visualization	Properties	Status
Year	Month	Beer_Sales	PredictedValues			
2039	6	127	130.963			
2013	9	142.5	131.02			
2016	3	134	131.56			
2012	9	127.5	131.581			
2039	4	131	131.701			
2015	8	132.3	131.712			
2017	7	127.8	131.822			
2036	5	124.5	132.809			
2016	1	123.7	133.4			
2039	7	143	133.543			

Showing 181 to 190 of 468 entries

Previous 1 ... 18 19 20 ... 47 Next



- vii) Click the **'Visualization'** tab.
- viii) The result data will be displayed via the time series chart.



2. Single Exponential Smoothing

- i) Select **'Trend'** option from the **'Output Mode'** drop-down menu.
- ii) Fill in the required fields.
- iii) Click **'Apply'**.
- iv) Click **'Run'** or **'Run Till Here'**.
- v) Users will be redirected to the **'Result'** tab.
- vi) A new column **'PredictedValues'** will be added in the result data.

Year	Quarter	Beer_Sales	PredictedValues
2012	3	75.7	79.386
2021	4	92.6	79.407
2018	3	77.5	79.526
2015	4	75.9	79.833
2013	2	101.1	80.078
2016	2	98.7	80.947
2021	3	73.6	81.895
2015	3	74.8	81.99
2024	4	87.1	82.28
2019	2	97.7	83.361

- vii) Click the **'Visualization'** tab.
- viii) The result data will be displayed via the time series chart.



3. Double Exponential Smoothing

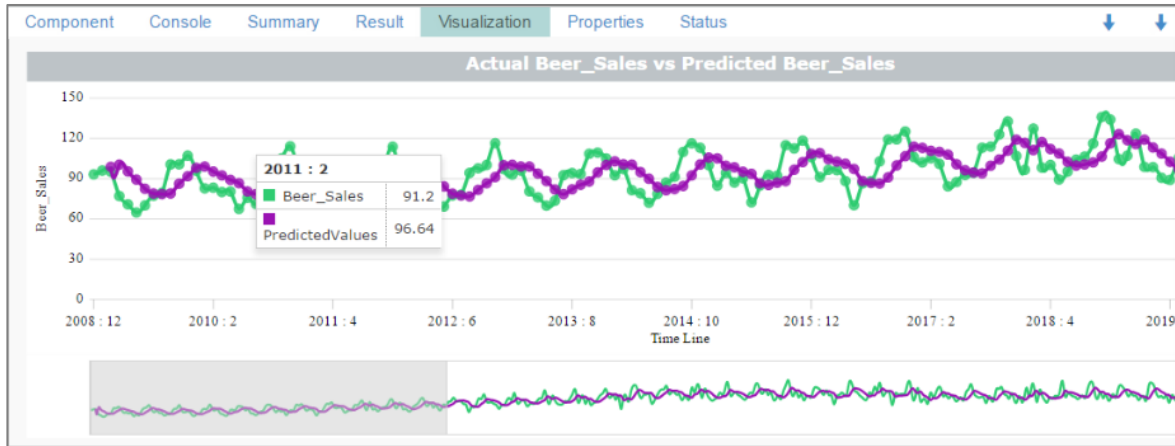
- i) Select 'Trend' option from the 'Output Mode' drop-down menu.
- ii) Fill in the other required fields.
- iii) Click 'Apply'.
- iv) Click 'Run' or 'Run Till Here'.
- v) Users will be redirected to the 'Result' tab.
- vi) A new column 'PredictedValues' will be added in the result data.

Year_1	Month_1	Year	Month	Beer_Sales	PredictedValues
2013	5	1969	June	69.9	88.245
2011	5	1967	June	70.4	88.407
2010	4	1966	May	80.4	89.102
2009	5	1965	June	64.8	89.315
2014	5	1970	June	72.1	89.922
2012	4	1968	May	72	90.106
2011	11	1967	December	113.8	90.255
2016	9	1972	October	119.1	90.938
2012	11	1968	December	116.4	91.152
2009	11	1965	December	107.1	91.784

Showing 41 to 50 of 468 entries

Previous 1 ... 4 5 6 ... 47 Next

- vii) Click the 'Visualization' tab.
- viii) The result data will be displayed via the time series chart.

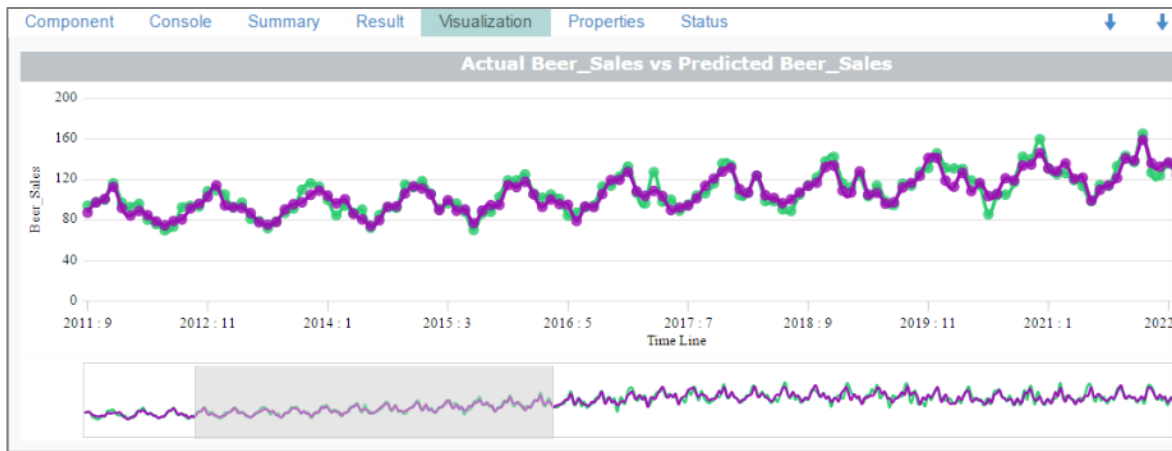


4. R-Auto ARIMA

- i) Select 'Trend' option from the 'Output Mode' drop-down menu.
- ii) Fill in the required fields.
- iii) Click 'Apply'.
- iv) Click 'Run' or 'Run Till Here'.
- v) Users will be redirected to the 'Result' tab.
- vi) A new column 'PredictedValues' will be added in the result data.

Year	Month	Beer_Sales	PredictedValues
2016	5	84.3	95.143
2016	6	87.5	79.225
2016	7	92.7	93.465
2016	8	94.4	92.797
2016	9	113	105.672
2016	10	113.9	119.508
2016	11	122.9	119.634
2016	12	132.7	127.383
2017	1	106.9	108.525
2017	2	96.6	103.917

- vii) Click the 'Visualization' tab.
- viii) The result data will be displayed via the time series chart.

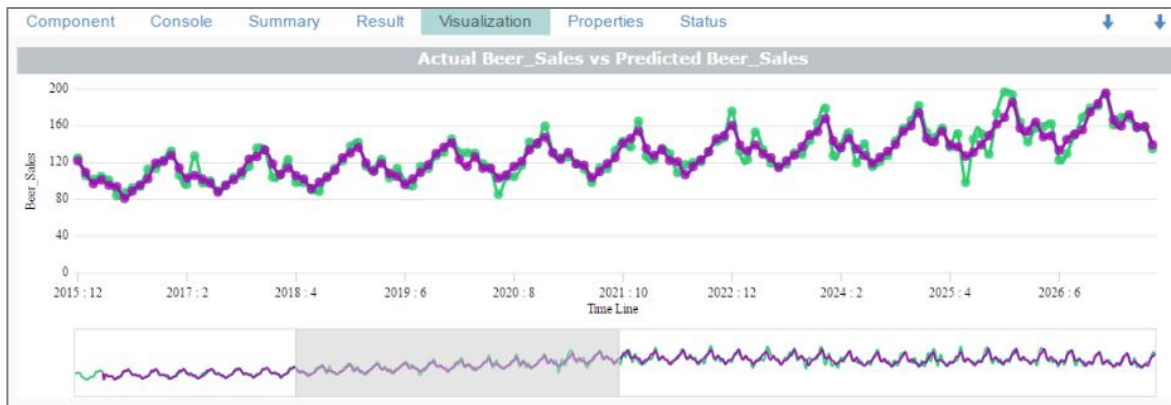


5. R-Auto Forecasting

- i) Select 'Trend' option from the 'Output Mode' drop-down menu.
- ii) Fill in the other required fields.
- iii) Click 'Apply'.
- iv) Click 'Run' or 'Run Till Here'.
- v) Users will be redirected to the 'Result' tab.
- vi) A new column 'PredictedValues' will be added in the result data.

Year	Month	Beer_Sales	PredictedValues
2020	9	117.3	121.59
2015	12	125.1	122.145
2023	7	118.6	122.242
2045	6	129	122.684
2022	4	130	122.707
2022	8	121	123.358
2020	1	131.5	123.628
2021	2	125.4	123.86
2017	10	116.2	124.017
2018	10	122.1	125.482

- vii) Click the 'Visualization' tab.
- viii) The result data will be displayed via the time series chart.



8.3. Association

This algorithm generates association rules discovering the recurrent patterns in large transactional datasets. It tries to understand future trends of customers based on their previous purchases and assists the vendors to associate items or services together.

8.3.1. Market Basket Analysis

i) Configure the following fields in the 'Properties' tab:

a. Output Information

- i. **Output Mode:** Select a mode of display for output data
 1. Selecting 'Rules' will display rules for the selected data set
 2. Selecting 'Transaction' will display the transaction IDs for the selected data set




b. Input Data Information

- i. **Input Data Format:** Select the format of the input data from the drop-down menu (out of the following choices):
 1. **Tabular**
 2. **Transactions**As per the selected 'Input Data Format', the result view will be of 2 types.
- ii. **Item Columns:** Select the item columns on which you want to apply association rules/analysis. Choose at least one option from the drop-down menu. This field displays only numerical and string columns. It can not display date columns.
- iii. **Transaction Id Column:** Select the column containing Transaction Ids to which you can apply the algorithm

Note: 'Transaction Id Column' field appears only when 'Transactions' option has been selected from the 'Input Data Format' drop-down menu.

c. Behavior

- i. **Support:** Enter a value for the minimum support of an item. The default value for this field is 0.1
- ii. **Confidence:** Select a value for the minimum confidence of the association. The default value for this field is 0.8

General	Output Information		
Properties	Output Mode	<input type="text" value="1 checked"/>	
Advanced	Input Data Information		
	Input Data Format	<input type="text" value="1 checked"/>	
	Item Column(s)	<input type="text" value="Select"/>	
	Behavior		
	Support	<input type="text" value="0.1"/>	
	Confidence	<input type="text" value="0.8"/>	

- ii) Click the '**Advanced**' tab and configure if required:

a. Output Appearance

- i. **Lhs Item(s):** Enter item tags separated by comma which should display on the left hand side of rules or itemsets.
- ii. **Rhs Item(s):** Enter item tags separated by comma which should display on the right hand side of rules or itemsets.
- iii. **Both Item(s):** Enter item tags separated by comma which should display on the both sides of rules or itemsets.
- iv. **None Item(s):** Enter item tags separated by comma which need not display in the rules or itemsets.
- v. **Default Appearance:** Select default appearance of the items out of the above given choices using a drop-down menu
- vi. **Min Length:** Set minimum length value. Default value for this field is 1.
- vii. **Max Length:** Set maximum length value. Default value for this field is 10.

b. Performance

- i. **Sort Type:** Select a sort type using the drop-down menu for sorting items based on their frequency.



- ii. **Filter Criteria:** Enter an indicating numerical value for filtering unused items from transactions. The default value for this field is 0.1.
- iii. **Use Tree Structure:** Selecting **'True'** option from the drop-down menu will organize transaction as a prefix tree.
- iv. **Use Heapsort:** Selecting **'True'** option from the drop-down menu will use heap sort against quick sort for sorting transaction.
- v. **Optimize Memory:** Selecting **'True'** option from the drop-down menu will minimize memory usage instead of maximizing speed.
- vi. **Load Transaction into Memory:** Selecting **'True'** from the drop down menu will load transactions into memory.

General	Output Appearance		
Properties	Lhs Item(s)	<input type="text" value="Optional"/>	
Advanced	Rhs Item(s)	<input type="text" value="Optional"/>	
	Both Item(s)	<input type="text" value="Optional"/>	
	None Item(s)	<input type="text" value="Optional"/>	
	Default Appearance	<input type="text" value="1 checked ▼"/>	
	Min Length	<input type="text" value="1"/>	
	Max Length	<input type="text" value="10"/>	

Performance	
Sort Type	<input type="text" value="1 checked ▼"/>
Filter Criteria	<input type="text" value="0.1"/>
Use Tree Structure	<input type="text" value="1 checked ▼"/>
Use Heapsort	<input type="text" value="1 checked ▼"/>
Optimize Memory	<input type="text" value="1 checked ▼"/>
Load Transaction into memory	<input type="text" value="1 checked ▼"/>
<input type="button" value="Apply"/>	



- iii) Click **'Apply'**.
- iv) Click **'Run'** or **'Run Till Here'**.
- v) Users will be redirected to the **'Result'** tab.
- vi) **'Rules'** will be displayed as first column in the result data (When the selected **'Output Mode'** option is **'Rules'**).

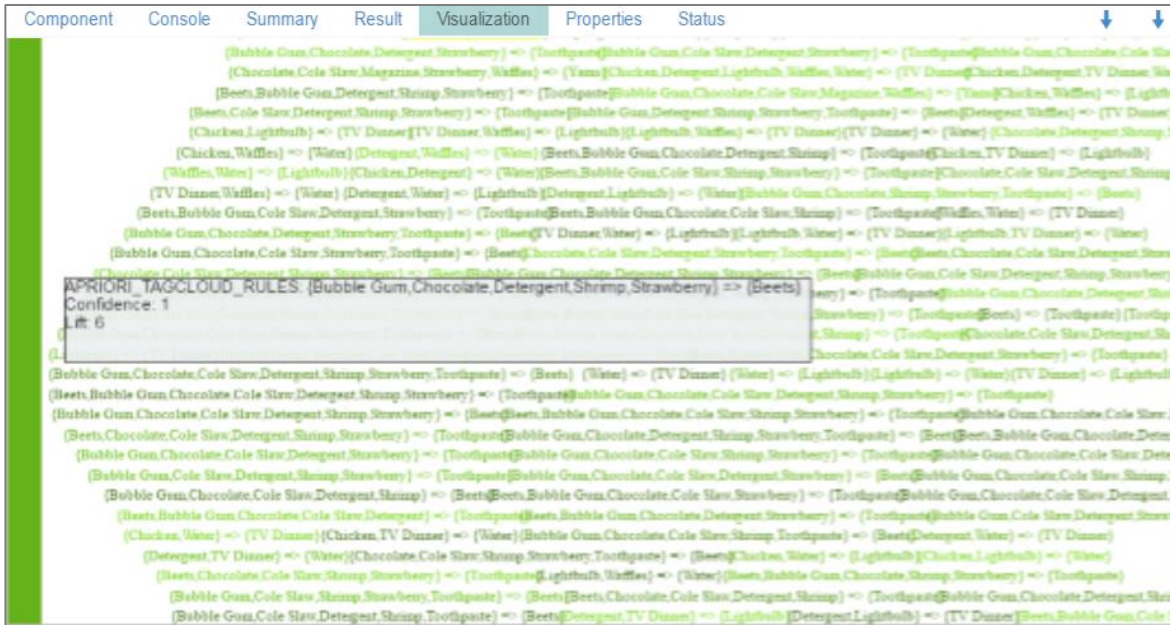
Rules	Support	Confidence	Lift
{Affluence=Low} => {MetroPolitan=Yes}	0.12	1	1.667
{Affluence=Low} => {SKYBox=Sky+HD 2TB}	0.12	1	1.515
{Affluence=Very Low} => {MetroPolitan=No}	0.1	0.833	2.083
{Affluence=Mid Low} => {MetroPolitan=Yes}	0.12	0.857	1.429
{Affluence=Mid Low} => {SKYBox=Sky+HD 2TB}	0.12	0.857	1.299
{Demographic:lifestyle=Liberal Opinion} => {HouseholdComposition=Men only HH}	0.12	0.857	2.521

- vii) **'Transaction_Id'** will be displayed as second column in the result data. (When the selected **'Output Mode'** option is **'Transaction'**).
- viii) The matching rules for the selected items will be displayed through the **'Matching_Rules'** column.

Items	Transaction_Id	Matching_Rules
{Chicken,Magazine,Oranges}	396	103,104,105
{Waffles}	434	
{Beets,Bubble Gum,Chocolate,Cole Slaw,Detergent,Shrimp,Strawberry,Toothpaste}	486	1455,1456,1457,1458,1459,1460,1461,1462
{Bubble Gum,Chocolate,Cole Slaw,Magazine,Strawberry,Waffles,Yams}	576	1392,1393,1394,1395,1396,1397,1398
{Chicken,Detergent,Lightbulb,TV Dinner,Waffles,Water}	664	1176,1177,1178,1179,1180,1181
{Chocolate,Cole Slaw,Oranges,Shrimp}	700	382,383,384

Showing 1 to 6 of 6 entries Previous **1** Next

- ix) Click the **'Visualization'** tab.
- x) The result data will be displayed via the word tag chart.



8.4. Regression Analysis

This algorithm is used to determine how an individual variable influences another variable using an exponential function. It finds trend in the dataset applying univariate regression analysis.

There are three sub types provided under ‘**Regression Analysis**’:

8.4.1. R-Linear Regression

i) Configure the following fields in the ‘**Properties**’ tab:

a. Output Information

- i. **Output Mode:** Select a mode of display for output data
 1. **Trend:** Selecting this option will predict the values for the dependent column and display them in output data through a new column
 2. **Fill:** Selecting this option will fill the missing values in the target column

b. Column Selection

- i. **Dependent Column:** Select the target column on which the regression analysis will be applied
- ii. **Independent Column:** Select the required input columns against which the regression analysis will be applied to the target column

c. New Column Information

- i. **Predicted Column Name:** Enter a name for the new column containing the predicted values.



General	Output Information	Output Mode	1 checked ▾	
Properties				
Advanced	Column selection	Dependent Column	1 checked ▾	
		Independent Column	1 checked ▾	
	New Column Information	Predicted Column Name	PredictedValues	

ii) Click the '**Advanced**' tab and configure if required:

a. Input Data Handling

- i. **Missing Values:** Select a method to deal with missing values from the drop-down menu
 1. **Ignore:** Selecting this option will skip the records containing missing values from the dependent and independent columns.
 2. **Keep:** Selecting this option will retain the records containing missing values while performing calculation.
 3. **Stop:** Selecting this option will stop application of the algorithm if a value is missing in any column.

b. Behavior

- i. **Allow Singular Fit:** Select an option for providing value to the Boolean Column
 1. **True:** Selecting this option will ignore aliased coefficients from the coefficient covariance matrix.
 2. **False:** Selecting this option will show an error in a model containing aliased coefficients
- ii. **Contrasts:** Selecting this option will display a list of contrast items that can be used for some variables in the model.
- iii. **Confidence Level:** Enter a value specifying accuracy (confidence level) of predictions for the algorithm. This field will take 0.95 as the default value.



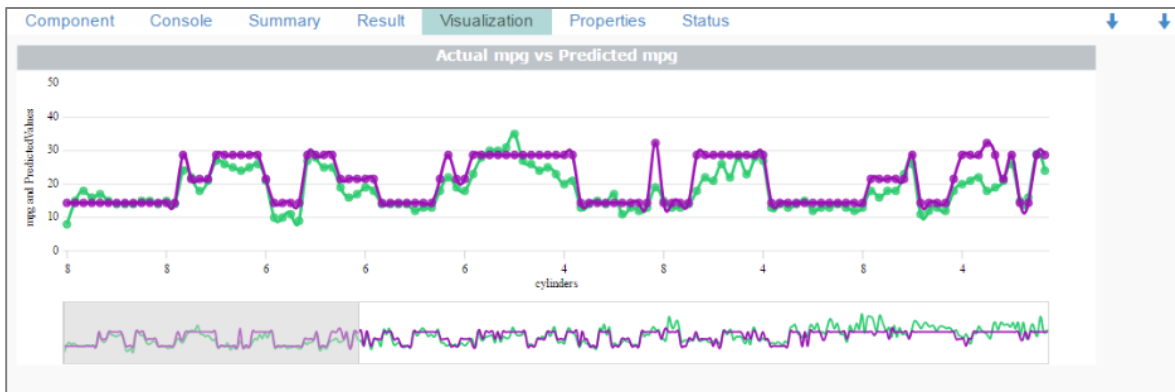
General	Input Data Handling	Missing values	1 checked ▾
Properties	Behavior	Allow Singular Fit	1 checked ▾
Advanced		Contrasts	1 checked ▾
		Confidence Level	0.95
			?
			Apply

Note: Model containing aliased coefficients signifies that the square matrix $x*x$ is singular.

- iii) Click **'Apply'**.
- iv) Click **'Run'** or **'Run Till Here'**.
- v) Users will be redirected to the **'Result'** tab.
- vi) A new column containing **'Predicted Values'** will be displayed in the result data.

Component	Console	Summary	Result	Visualization	Properties	Status	Search: <input type="text"/>							
Show 10 entries														
mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	carname	PredictedValues					
8	8	307	130	3504	12	70	1	chevrolet chevelle malibu	14.365					
15	8	350	165	3693	11.5	70	1	buick skylark 320	14.365					
18	8	318	150	3436	11	70	1	plymouth satellite	14.365					
16	8	304	150	3433	12	70	1	amc rebel sst	14.365					
17	8	302	140	3449	10.5	70	1	ford torino	14.365					
15	8	429	198	4341	10	70	1	ford galaxie 500	14.365					
14	8	454	220	4354	9	70	1	chevrolet impala	14.365					
14	8	440	215	4312	8.5	70	1	plymouth fury iii	14.365					
14	8	455	225	4425	10	70	1	pontiac catalina	14.365					
15	8	390	190	3850	8.5	70	1	amc ambassador dpi	14.365					
Showing 1 to 10 of 398 entries						Previous	1	2	3	4	5	...	40	Next

- vii) Click the **'Visualization'** tab.
- viii) The result data will be displayed via the time series chart.



Note: 'Behavior' fields provided under 'Advanced' section differs as per the algorithm sub-type. 'Input Data Handling' remains the same for all the provided Regression types. Hence, only 'Advanced' tab is explained below for all the sub-algorithms provided under 'Regression'.

8.4.2. R-Multiple Linear Regression

i) Click the 'Advanced' tab and configure if required:

a. Input Data Handling

- i. **Missing Values:** Select a method to deal with missing values (via the drop-down menu).
 1. **Ignore:** Selecting this option will skip the records containing missing values from the dependent and independent columns.
 2. **Keep:** Selecting this option will retain the records containing missing values while performing calculation.
 3. **Stop:** Selecting this option will stop application of the algorithm if a value is missing in any column.

a) **Behavior**

- **Confidence Level:** Enter a value specifying accuracy (confidence level) of predictions for the algorithm. This field will take 0.95 as the default value.



General	Input Data Handling
Properties	Missing values <input type="text" value="1 checked"/>
Advanced	Behavior
	Confidence Level <input type="text" value="0.95"/> i
<input type="button" value="Apply"/>	

- ii) Click **'Apply'**.
- iii) Click **'Run'** or **'Run Till Here'**.
- iv) Users will be redirected to the **'Result'** tab.
- v) A new column **'PredictedValues'** will be added in the result data.

mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	carname	PredictedValues
8	8	307	130	3504	12	70	1	chevrolet chevelle malibu	7.281
15	8	350	165	3693	11.5	70	1	buick skylark 320	7.876
18	8	318	150	3436	11	70	1	plymouth satellite	7.371
16	8	304	150	3433	12	70	1	amc rebel sst	7.176
17	8	302	140	3449	10.5	70	1	ford torino	7.138
15	8	429	198	4341	10	70	1	ford galaxie 500	9.065
14	8	454	220	4354	9	70	1	chevrolet impala	9.449
14	8	440	215	4312	8.5	70	1	plymouth fury iii	9.238
14	8	455	225	4425	10	70	1	pontiac catalina	9.464
15	8	390	190	3850	8.5	70	1	amc ambassador dpl	8.478

- vi) Click the **'Visualization'** tab.
- vii) The result data will be displayed via the time series chart.



8.4.3. R-Logistic Regression

i) Click the '**Advanced**' tab and configure if required:

a. Behavior

i. **Family:** Select an option from the drop down list

1. Binomial
2. Poisson
3. Gaussian
4. Gamma
5. Quasi
6. Quasipoisson
7. Quasibinomial

ii. **Maximum No. of Iterations:** Enter a valid integer value allowed to calculate the algorithm coefficient. The default values for this field is 25.

General	Input Data Handling	
Properties	Missing values	1 checked ▾
Advanced	Behavior	
	Family	1 checked ▾
	Maximum No Of Iterations	25
		Apply

ii) Click '**Apply**'.

iii) Click on '**Run**' or '**Run Till Here**'.

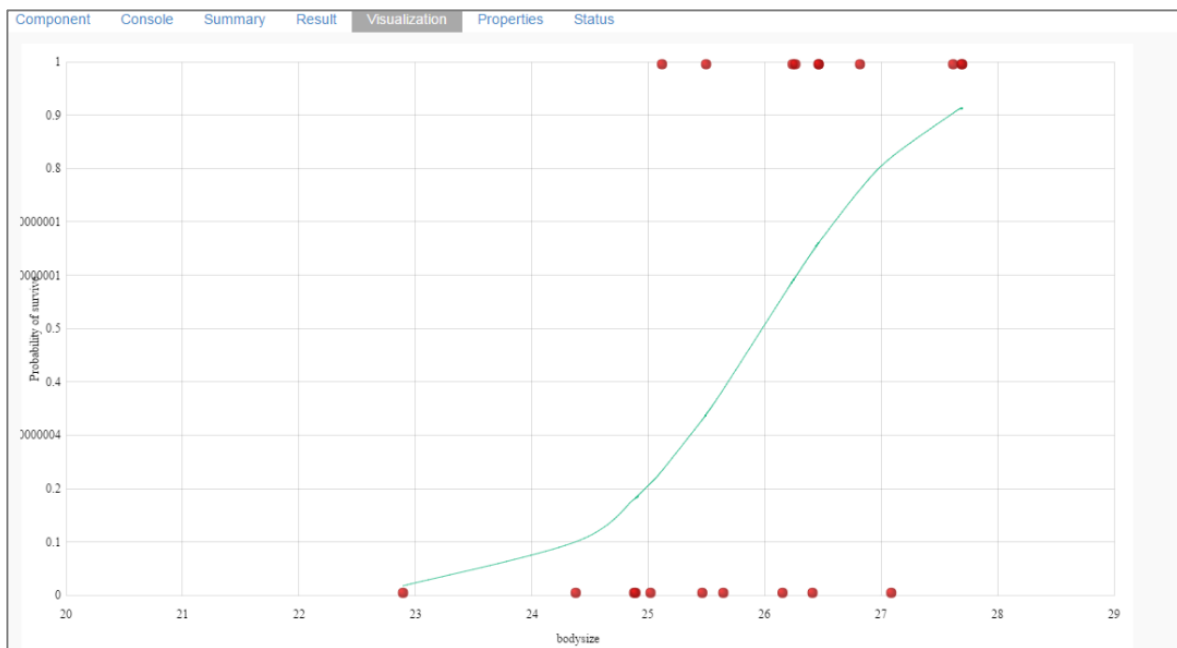
iv) Users will be redirected to the '**Result**' tab.

v) A new column containing '**PredictedValues**' will be added in the result data.



Component	Console	Summary	Result	Visualization	Properties	Status
Show 10 entries Search: <input type="text"/>						
bodysize	survive	PredictedValues				
27.1435042073117	0	0.01313191				
27.3698819529919	0	0.01909258				
27.4713319662842	0	0.02256035				
27.9561274911923	0	0.04953421				
28.6486888249175	0	0.14297599				
29.2359919584729	1	0.30914426				
29.2962754702829	0	0.33118131				
29.4650506154866	1	0.39668291				
29.7186205793043	0	0.50167263				
29.7719375103457	0	0.52404711				
Showing 1 to 10 of 20 entries						Previous 1 2 Next

- vi) Click the 'Visualization' tab.
- vii) The result data will be displayed via the scatter plot with regression line chart.



8.5. Outliers

This algorithm is used to discover patterns in dataset that do not follow the expected behavior. It lists the outlying values based on the statistical distribution between the first and third quartiles.

Interquartile Range has been provided as a sub algorithm type.

8.5.1. Interquartile Range

i) Configure the following fields in the 'Properties' tab:

a. Output Information

- i. **Output Mode:** Select a mode of display for output data.
 1. **Show Outlier:** Selecting this option will add a Boolean column to the input data identifying whether the resultant value is an outlier.
 2. **Remove Outlier:** Selecting this option will remove outlying values from the input data.

b. Column Selection




- i. **Feature:** Select an input column that can be used to perform the analysis.

c. Behavior

- i. **Fence Coefficient:** Enter the permissible deviation limit for values from the inter quartile range (The default value for this field is 1.5).

d. New Column Information

- i. **New Column Name:** Enter a name for the new column containing the predicted values.

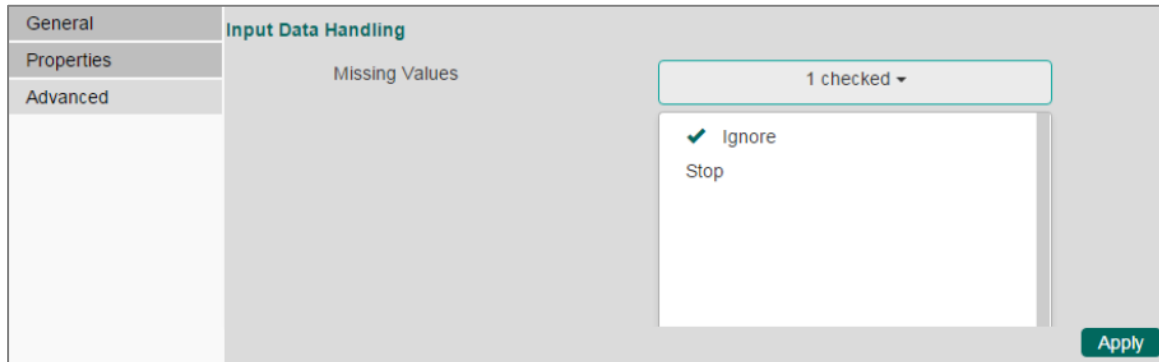
General	Output Information	Output Mode	1 checked ▾	
Properties	Column Selection	Feature	1 checked ▾	
Advanced	Behavior	Fence Coefficient	1.5	
	New Column Information	New Column Name	OutliersDetected	

ii) Click the 'Advanced' tab and configure if required:

a. Input Data Handling

- i. **Missing Values:** Select a method to deal with missing values from the drop-down menu.
 1. **Ignore:** Selecting this option will skip the records containing missing values in the columns.

2. **Stop:** Selecting this option will stop application of the algorithm if a value is missing in any column.



- iii) Click **'Apply'**.
- iv) Click **'Run'** or **'Run Till Here'**.
- v) Users will be redirected to the **'Result'** tab.
- vi) **'OutliersDetected'** column will be displayed in the result data (If **'Show Outliers'** option has been selected).

SepalLength	SepalWidth	PetalLength	PetalWidth	Species	OutliersDetected
4.8	3.1	1.6	0.2	setosa	0
5.4	3.4	1.5	0.4	setosa	0
5.2	4.1	1.5	0.1	setosa	1
5.5	4.2	1.4	0.2	setosa	1
4.9	3.1	1.5	0.2	setosa	0
5	3.2	1.2	0.2	setosa	0
5.5	3.5	1.3	0.2	setosa	0
4.9	3.6	1.4	0.1	setosa	0
4.4	3	1.3	0.2	setosa	0
5.1	3.4	1.5	0.2	setosa	0

OR

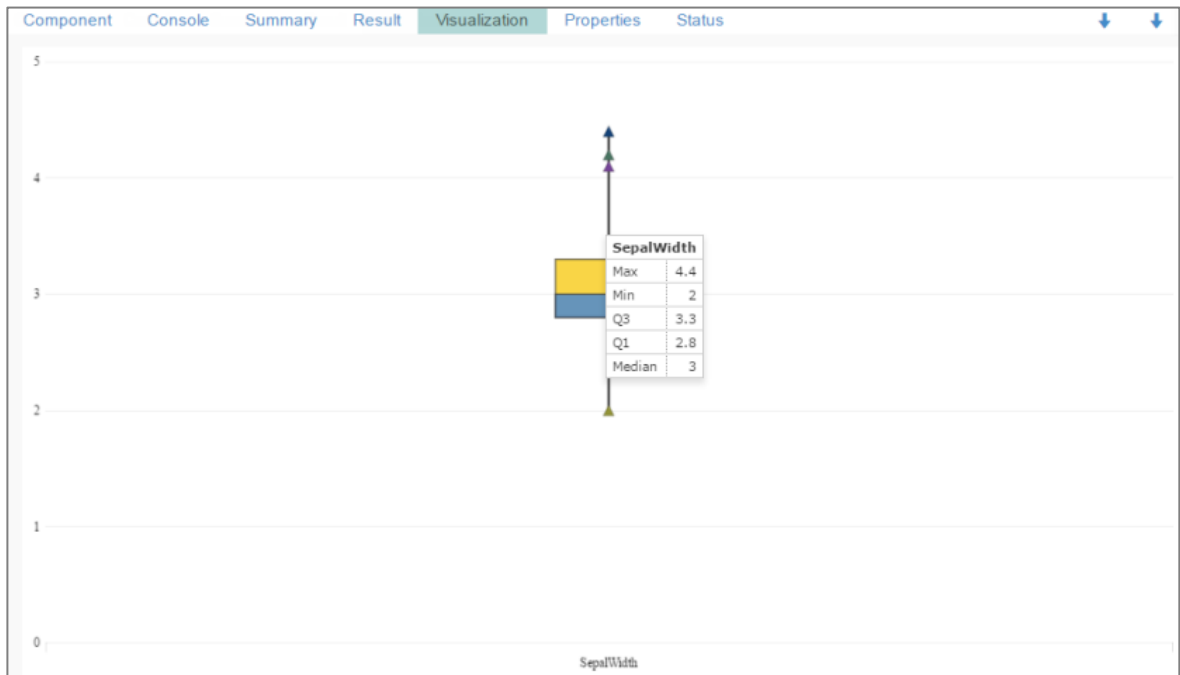
Outliers column will not be displayed in the result data (If **'Remove Outliers'** option has been selected).



SepalLength	SepalWidth	PetalLength	PetalWidth	Species
5.1	3.7	1.5	0.4	setosa
4.6	3.6	1	0.2	setosa
5.1	3.3	1.7	0.5	setosa
4.8	3.4	1.9	0.2	setosa
5	3	1.6	0.2	setosa
5	3.4	1.6	0.4	setosa
5.2	3.5	1.5	0.2	setosa
5.2	3.4	1.4	0.2	setosa
4.7	3.2	1.6	0.2	setosa
4.8	3.1	1.6	0.2	setosa

Showing 21 to 30 of 146 entries

- vii) Click the **'Visualization'** tab.
- viii) The result data will be displayed via the boxplot chart.



8.6. Classification

This algorithm categorizes a new observation on the basis of a trained set of data that contains observations from known category. It compares each new observation to previous observations using means of similarity or distance.

There are two subtypes provided under ‘**Classification**’:

8.6.1. R-CNR Tree

- i) Configure the following fields in the ‘**Properties**’ tab:
 - a. **Output Information**
 - i. **Output Mode:** Select a mode of display for output data.
 - 1. **Trend:** Selecting this option will predict the values for the dependent column and display them in output data through a new column.
 - 2. **Fill:** Selecting this option will fill the missing values in the target column.
 - ii. **Algorithm Type:** Select an algorithm type from the drop-down Menu.
 - 1. **Classification:** Select this option if users want to pass dependent column as the categorical values.
 - 2. **Reegression:** Select this option if users want to pass dependent column as numerical values.
 - iii. **Show Probability:** Select an option from the drop-down menu to create a new column for indicating the chance factor involved in the probability.
 - 1. **True:** Selecting this option will display a new column in the output data with probability values.
 - 2. **False:** Selecting this option will not display any probability value in the output data.
 - b. **Column Selection**
 - i. **Features:** Select input columns from the drop down list to which the target column can be compared for performing analysis.
 - ii. **Target Variable:** Select the target column for which the analysis is performed.
 - c. **New Column Information**
 - i. **Predicted Column Name:** Enter a name for the new column containing the predicted values.



General	Output Information	
Properties	Output Mode	1 checked ▾
Advanced	Algorithm Type	1 checked ▾
	Show Probability	1 checked ▾
	Column Selection	
	Features	1 checked ▾ ⓘ
	Target Variable	1 checked ▾ ⓘ
	New Column Information	
	Predicted Column Name	PredictedValues ⓘ

Note: The 'Show Probability' field will appear only if, 'Classification' option is selected via the 'Algorithm Type' drop-down menu.

ii) Click the 'Advanced' tab and configure if required:

a. Input Data Handling

- i. **Missing Values:** Select a method to deal with missing values from the drop-down list.
 1. **Rpart:** Selecting this option will try to estimate the missing values for the dependent column based on the independent columns.
 2. **Ignore:** Selecting this option will skip the records containing missing values in the columns.
 3. **Keep:** Selecting this option will retain the records containing missing values while performing calculation.
 4. **Stop:** Selecting this option will stop application of the algorithm if a value is missing in any column.

b. Tree Pruning

- i. **Minimum Split:** It indicates minimum number of observations within a single node for a split to be attempted. The default value for this field is 10.
- ii. **Complexity Parameter:** This parameter is primarily used to save the computing time by pruning off splits that are not worthwhile. Any split which does not improve the fit by a factor of complexity parameter is pruned off performing cross validation, hence the program will not pursue it. The default value for this field is 0.05.
- iii. **Maximum Depth:** It sets the maximum depth of any node of the

final tree keeping the depth count for root node 0. It is an optional field (It is recommended to set Maximum Depth value less than 30 rpart for 32 bit-machines.)

c. Behavior

- i. **Split Criteria:** It is an optional field that depends on the selected algorithm type from the **'Properties'**. (This field appears only when the selected algorithm type is **'Classification'**).

The splitting index can be:

1. **Gini:** Select this option to measure inequality among values of randomly chosen elements from a set.
 2. **Information:** Select this option to get information about the variables used in the algorithm.
- ii. **Cross Validation:** It indicates number of cross validations that were performed to check accuracy of the analysis method.
 - iii. **Prior Probability:** It is an optional field. This field is dependent on the prior data values mentioned in the selected dataset. (This field appears only when the selected algorithm type is **'Classification'**).

d. Surrogate Information

- i. **Use Surrogate:** Select one option from the drop-down menu.
 1. **Display Only:** Selecting this option will only display the observation, but not split it further.
 2. **Use Surrogate:** Selecting this option will search surrogate value for the missing values in order to split the observation. Two fields will be displayed:
 - a. **Surrogate Style:** Select a style using the drop-down menu.
 - b. **Maximum Surrogate:** Set the maximum surrogate value.
 3. **Stop if missing:** Selecting this option will choose an action based on the nature of majority observations. If values are missed for all the observations, then it will stop splitting further.

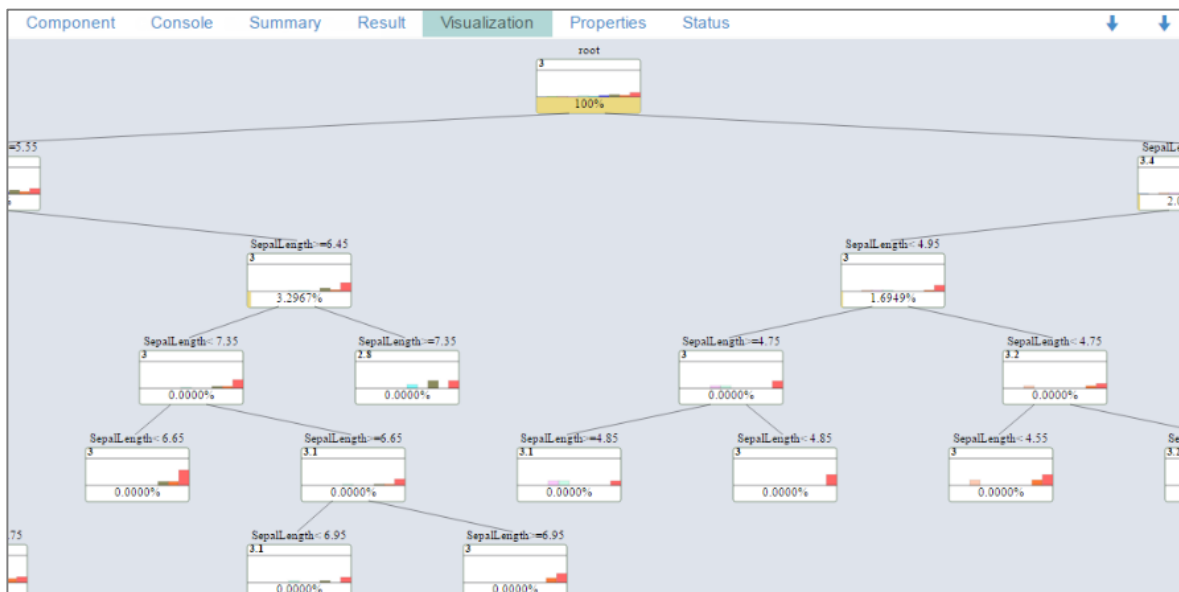


General	Input Data Handling	
Properties	Missing values	1 checked ▾
Advanced	Tree Pruning	
	Minimum Split	10
	Complexity Parameter	.005
	Maximum Depth	Optional
	Behavior	
	Split Criteria	1 checked ▾
	Cross Validation	Optional
	Prior Probability	Optional
	Surrogate Information	
	Use Surrogate	1 checked ▾
	Surrogate Style	1 checked ▾
	Maximum Surrogate	Optional
Apply		

- iii) Click **'Apply'**.
- iv) Click **'Run'** or **'Run Till Here'**.
- v) Users will be redirected to the **'Result'** tab.
- vi) A new column **'PredictedValues'** will be displayed in the result data.

SepalLength	SepalWidth	PetalLength	PetalWidth	Species	PredictedValues	Probability
5.1	3.5	1.4	0.2	setosa	3.5	0.2173913
4.9	3	1.4	0.2	setosa	3.1	0.3333333
4.7	3.2	1.3	0.2	setosa	3.2	0.5000000
4.6	3.1	1.5	0.2	setosa	3.2	0.5000000
5	3.6	1.4	0.2	setosa	3.5	0.2173913
5.4	3.9	1.7	0.4	setosa	3.4	0.2857143
4.6	3.4	1.4	0.3	setosa	3.2	0.5000000
5	3.4	1.5	0.2	setosa	3.5	0.2173913
4.4	2.9	1.4	0.2	setosa	3	0.4000000
4.9	3.1	1.5	0.1	setosa	3.1	0.3333333

- vii) Click the **'Visualization'** tab.
- viii) The result data will be displayed via the tree chart.



8.6.2. R-Naive Bayes

i) Configure the following fields in the **'Properties'** tab:

a. Output Information

- i. **Output Mode:** Select a mode of display for output data
 1. **Trend:** Selecting this option will predict the values for the dependent column and display them in output data through a new column.
 2. **Fill:** Selecting this option will fill the missing values in the target column.

b. Column Selection

- i. **Feature:** Select input columns from the drop-down menu to which the target variable can be compared for performing analysis.
- ii. **Target Variable:** Select the target column for which the analysis is Performed.

c. New Column Information

- i. **Predicted Column Name:** Enter a name for the new column containing the predicted values.



General	Output Information	
Properties	Output Mode	1 checked ▾
Advanced	Column Selection	
	Feature	1 checked ▾ i
	TargetVariable	1 checked ▾ i
	New Column Information	
	Predicted Column Name	PredictedValues i

- ii) Click the **'Advanced'** tab and configure if required:
 - a. **Input Data Handling**
 - i. **Missing Values:** Select a method to deal with missing values from the drop-down menu.
 1. **Ignore:** Selecting this option will skip the records containing missing values in the columns.
 2. **Keep:** Selecting this option will retain the records containing missing values while performing calculation.
 - ii. **Laplace Smoothing:** Enter the smoothing constant for smoothing observations. Smoothing constant must be a double value greater than 0. Entering 0 will disable Laplace smoothing.

General	Input Data Handling	
Properties	Missing values	1 checked ▾
Advanced	Laplace Smoothing	0
		Apply

- iii) Click **'Apply'**.
- iv) Click **'Run'** or **'Run Till Here'**.
- v) Users will be redirected to the **'Result'** tab.
- vi) A new column entitled **'Predicted Values'** will be added in the result data.



SepalLength	SepalWidth	PetalLength	PetalWidth	Species	PredictedValues
5.1	3.5	1.4	0.2	setosa	3.5
4.9	3	1.4	0.2	setosa	3.4
4.7	3.2	1.3	0.2	setosa	3.4
4.6	3.1	1.5	0.2	setosa	3.4
5	3.6	1.4	0.2	setosa	3.5
5.4	3.9	1.7	0.4	setosa	3.9
4.6	3.4	1.4	0.3	setosa	3.4
5	3.4	1.5	0.2	setosa	3.5
4.4	2.9	1.4	0.2	setosa	3
4.9	3.1	1.5	0.1	setosa	3.4

Note: The 'Visualization' tab does not display any graphical representation for the R Naïve Bayes results data.

8.6.3. Spark-Naive Bayes

The Naive Bayes is a simple multiclass classification algorithm with the assumption of independence between every pair of features. This algorithm can be trained to be very efficient. The user can set a threshold for each class. The algorithm will then classify values as per the set thresholds.

Spark Naive Bayes consists of two types of model selection methods:

1. Multinomial- If the data set is numerical
2. Bernouli- If the data set contains 0 and 1

- i) Configure the following fields in the 'Properties' tab:
 - a. **Feature:** Select from the drop-down menu
 - b. **Label:** Select from the drop-down menu
 - c. **Enable Validation:** Put a check mark in the box to enable the validation (It is an optional field).
- ii) Click 'Next'.



Component	Console	Summary	Result	Visualization	Properties	Status
General	Column Selection					
Properties						
Validation	Feature	2 checked ▾			<i>i</i>	
Advanced	Label	1 checked ▾			<i>i</i>	
	Enable Validation	<input checked="" type="checkbox"/>				
						Next

- iii) Users will be redirected to the Validation tab (When the validation has been enabled by putting a check mark in the box, ‘**Apply**’ will change to ‘**Next**’)

There are two types of validation methods:

- a. **Train Validation** – Train validation begins by splitting a data set into two parts, as training and testing data sets as per the train ratio. It also iterates through paramMapS. For each combination of parameters, the algorithm will iterate over it and select based on the evaluation metric.
- b. **Cross Validation** – Cross validation begins by splitting the data set into a set of folds which are used as separate training and test data sets. e.g., with k=3 folds, Cross Validator will generate 3 (training, test) dataset pairs, each of which uses 2/3 of the data for training and 1/3 for testing. It also iterates through paramMapS. The algorithm will iterate over each combination of parameters and folds to determine the best model using average of the k folds.

- iv) Configure the following ‘**Validation**’ information:

- a. **Model Selection Method:** Select any one validation method using the drop-down menu:
 - i. Train Validation
 - ii. Cross Validation
- b. **Evaluator:** Select any one option using the drop-down menu to define evaluator. Evaluator consist of two types:
 - i. Multi Class Classification – If the data set has multiple classes in label column
 - ii. Binary Class Classification- if the data set has two classes in label column
- c. **Train Ratio:** This field will be displayed if train validation has been selected by using the ‘**Model Selection Method**’ field.



Component	Console	Summary	Result	Visualization	Properties	Status
General	Model Selection					
Properties	Model Selection Method		<input type="text" value="1 checked"/>			
Validation	Evaluator		<input type="text" value="1 checked"/>			
Advanced	Train Ratio		<input type="text" value="0.7"/>			
						<input type="button" value="Apply"/>

OR

If **'Cross Validation'** is enabled, users will be provided with a field **'Number of folds'** from the input data to be taken as training data for the cross validation.

Component	Console	Summary	Result	Visualization	Properties	Status
General	Model Selection					
Properties	Model Selection Method		<input type="text" value="1 checked"/>			
Validation	Evaluator		<input type="text" value="1 checked"/>			
Advanced	Number of folds		<input type="text" value="3"/>			
						<input type="button" value="Apply"/>

- v) Configure the following **'Advanced'** information:
 - a. Model Type: Select an option from the drop-down list
Spark Naive Bayes consists of two types of model selection methods:
 - i. Multinomial- If the data set is numerical
 - ii. Bernoulli- If the data set contains 0 and 1
 - b. Thresholds: Enter multiple values separated by coma. Number of values entered as threshold should be same as that of number of classes in labels. Sum of values must be equal to 1. Enter at least two comma separated values in this field.
 - c. Parameter Grid: Enter a valid double value between 0 and 1 (1 included).
Users can enter single or comma separated valid double value.
- vi) Click **'Apply'**.



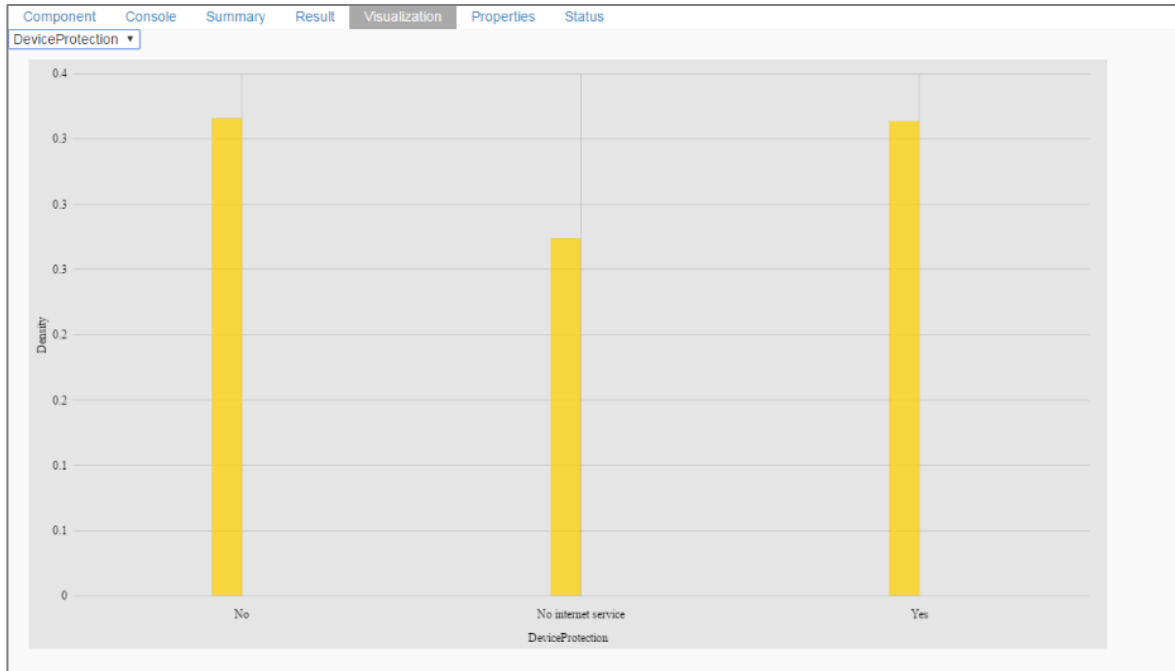
Component	Console	Summary	Result	Visualization	Properties	Status
General	Input Data Handling					
Properties	Model Type		<input type="text" value="1 checked"/>			
Validation	Thresholds		<input type="text" value="0.3,0.2, 0.5"/>			
Advanced	Parameter Grid (Additive Smoothing (λ)) Enter multiple values separated by Comma		<input type="text" value="1.0"/>			
						<input type="button" value="Apply"/>

Note: If validation is enabled, users can enter multiple comma separated values in the Parameter Grid in the Advanced tab and they will be taken as paraMapS.

- vii) Click 'Run' to run the process.
- viii) Users can click the 'Summary' tab to view summary of the model.

Component	Console	Summary	Result	Visualization	Properties	Status
----- Summary of the model -----						
Columns used as Feature:						
Features						
Label Column = Label						
Number of Classes = 2						
Number of Features = 30						
Model Types = multinomial						
Smoothing value = 1.0						
----- End of Summary -----						

- ix) Click the 'Visualization' tab.
- x) The graphical presentation of the data will be displayed via the conditional probability chart.



Note: Spark Naive Bayes supports only string data when cross validation is selected.

8.7. Correlation

The Correlation algorithm provides a method for clustering a set of objects into the optimal number of clusters without specifying the number in advance.

8.7.1. R- Correlation

- i) Configure the following fields in the **'Properties'** tab:
 - a. **Input Columns:** Select any two columns using the drop-down menu
 - b. **Method:** Select a method using the drop-down menu
 - c. **Missing Value Method:** Select the required option using the drop-down menu
- ii) Click **'Apply'**.

General	Column Selection
Properties	Input columns 2 checked ▾ Method 1 checked ▾ Missing value method 1 checked ▾
	Apply

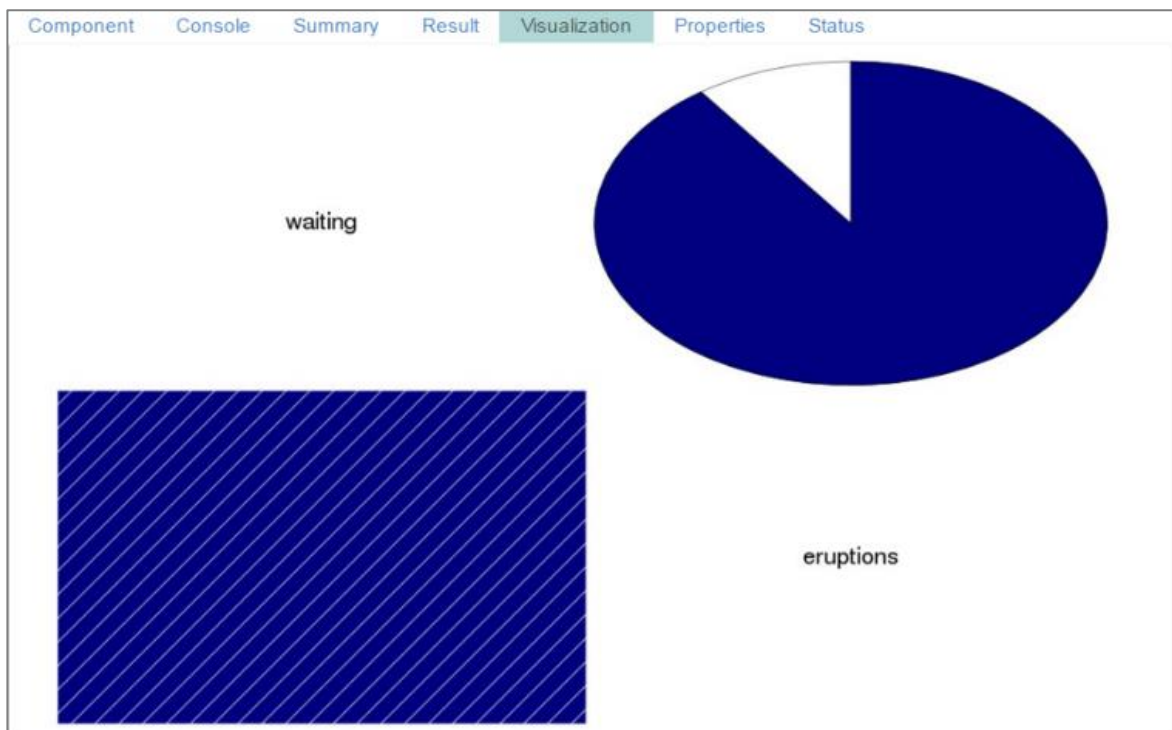


- iii) Click **'Run'** or **'Run Till Here'**.
- iv) Users will be redirected to the **'Result'** tab.
- v) Columns displaying **'Eruption'** and **'Waiting'** probable values will be added in the result data.

category	eruptions	waiting
eruptions	1.0	0.901
waiting	0.901	1.0

Showing 1 to 2 of 2 entries

- vi) Click the **'Visualization'** tab.
- vii) The probable values of the selected columns will be displayed via the correlogram chart.



9. Apply Model

9.1. Spark Apply Model

This component is provided to generate predictions based on trained classification model. Users can view predicted column value and probability of each label class by using the classification model.

The Spark Apply model has two input nodes:

1. The first node is for the Saved model component.
2. The second node is for the test data set.

The created columns will be based on the used algorithm.

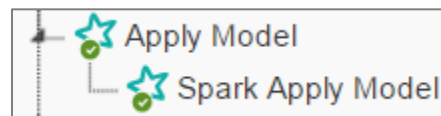
Users can create a model via the following ways:

- Generate a model using an algorithm
- Generate a model using the saved models

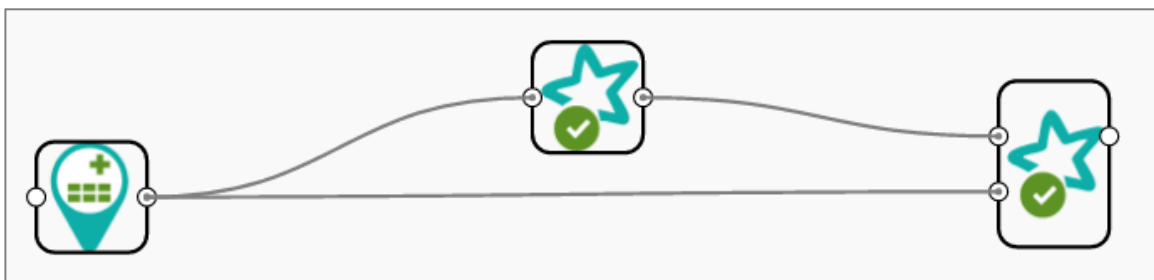
The Apply Model component consists of 2 input nodes and 1 output node.

- **Input Nodes**
 - Upper node – Model/Training data
 - Lower node – Testing data
- **Output Node**
 - Node – Result data

- i) Click the '**Apply Model**' tree-node.
- ii) The '**Spark Apply Model**' leaf-node will be displayed.



- iii) Drag a Spark Apply model component onto the workspace and connect it with a valid data set.
- iv) Click '**Spark Apply Model**' component.



- v) Basic component details will be displayed.
- vi) Click **'Apply'**.

Component	Console	Summary	Result	Visualization	Properties	Status
General	Basic					
	Component Name	<input type="text" value="Spark Apply Model"/>				
	Alias	<input type="text" value="Spark Apply Model2"/>				
	Description	<input type="text" value="Optional"/>				
						<input type="button" value="Apply"/>

- vii) Click **'Run'**.
- viii) Click the **'Apply Model'** Component on the workspace.
- ix) Click the **'Result'** tab to view the result data.

Component	Console	Summary	Result	Visualization	Properties	Status
Search: <input type="text"/>						
rawPrediction	probability	binarycolumn	prediction			
{"values": [-20.902468589073305,-24.957519611673458]}	{"values": [0.9829607724165242,0.01703922758347579]}	0	0			
{"values": [-20.528352738957153,-24.5292324212055]}	{"values": [0.9820293210205892,0.017970678979410865]}	0	0			
{"values": [-17.562146273204608,-20.053843038793048]}	{"values": [0.9235576785437413,0.07644232145625862]}	0	0			
{"values": [-23.73252349149471,-28.78898426447168]}	{"values": [0.9936722378759933,0.006327762124006625]}	0	0			
{"values": [-16.096269772540268,-18.398853127358507]}	{"values": [0.909090765439914,0.09090923456008618]}	0	0			
{"values": [-19.862575074442233,-23.105905486999283]}	{"values": [0.962432709475426,0.03756729052457394]}	0	0			
{"values": [-12.484543780343461,-11.725861771685647]}	{"values": [0.3189324847675862,0.6810675152324137]}	1	0			
{"values":	{"values":	0	0			

- x) Click the **'Properties'** tab to view the properties details (This Properties tab display workflow properties).



Component	Console	Summary	Result	Visualization	Properties	Status
	Created By			Ranjit Krishnan		
	Created At			2016-09-22 19:25:45 +0530		
	Last Modified By			Ranjit Krishnan		
	Last Modified At			2016-09-26 14:50:39 +0530		
	Version			2.2.0		

Note:

- a. Currently only '**Spark Apply Model**' is provided in the Tree-node menu.
- b. The result data set of the model can be written to a data base using the Cassandra Writer.
- c. Column header and data type of feature column for both saved model and testing data should match. If column headers and data types do not match, an alert message will be displayed.
- d. It is not mandatory for the testing data set to contain a label column.

10. Performance

The Spark Performance Components are used to evaluate model performance through a list of parameters. The Performance component can be attached to classification.

The Spark Performance component is provided as a leaf-node under the Performance tree-node. It contains 3 input nodes that can be used to compare up to 3 models. Each node has static name like model_0, model_1, and model_2. Based on connection to the node model summary can be viewed with respective names.

Connecting the Performance component to a model:

- i) Select and drag a Performance component onto the workspace.
- ii) Connect it with a valid workflow and configure the '**Properties**' tab.
 - i. **Performance Type**: Select an option out of 1. Binary Classification or 2. Multiclass classification (Default option is Multi-class Classification)
 - ii. **Beta Value**: Enter a numerical value (Optional)
- iii) Click '**Apply**'.



Component Console Summary Result Visualization Properties Status

General **Spark-Performance**

Properties

Performance Type

Beta Value

Apply

- iv) Click 'Run'.
- v) Click the 'Summary' tab.
- vi) The summary of the model will be displayed.

spark_nb_model Run Reset Refresh Clear Cache Save Save As

Component Console Summary Result Visualization Properties Status

----- Summary of MultiClass Metrics -----

Model Name	Accuracy	Weighted Precision	Weighted Recall	Weighted FMeasure	Weighted FMeasure(beta 1.0)	Weighted True Positive Rate
Model_0	0.5650793650793651	0.6065129809272264	0.5650793650793651	0.5809395617446835	0.5809395617446835	0.5650793650793651

----- Label Wise Model - 0 -----

Labels	Precision	Recall	FMeasure	FMeasure(beta 1.0)	TruePositiveRate	FalsePositiveRate
0.0	0.7268041237113402	0.6266666666666667	0.6730310262529834	0.6730310262529834	0.6266666666666667	0.5888888888888888
1.0	0.30578512396694213	0.4111111111111111	0.35071090047393366	0.35071090047393366	0.4111111111111111	0.3733333333333333

---- Confusion Matrix (Model - 0)----

	Predict_0.0	Predict_1.0
Actual_0.0	141.0	84.0
Actual_1.0	53.0	37.0

----- End of Summary -----

Performance components can be of the following formats:

1. Binary Classification: Used when the label has two classes
2. Multi Class Classification: Used when the label has 3 or more beta values.

In the case of multiple models, all the model statistics will come in the summary of performance (up to 3 models can be compared).



10.1. Binary Classification Model

- i) Each model is named as Model_0, Model_1, and Model_2
- ii) Each model is displayed in a separate tab under the 'Result' tab.
- iii) The model contains the following columns:
 - a. Threshold
 - b. Precision
 - c. Recall
 - d. F measure
 - e. F measure with theta
- iv) Rows can be created based on the number of threshold values.

Component	Console	Summary	Result	Visualization	Properties	Status
Model_0	Model_1	Model_2				
threshold	precision	recall	Fmeasure	FmeasurewithTheta		
0.82129495	0.200654993	0.27706118	0.691315328	0.422993115		
0.627022902	0.596710036	0.392648258	0.089319484	0.769040356		
0.003421089	0.386320956	0.038205996	0.436402286	0.707450743		
0.288700275	0.332891002	0.214962218	0.429561927	0.277724747		
0.606283985	0.489951939	0.912429898	0.028684714	0.053625012		
0.602992271	0.511832616	0.726989959	0.878639099	0.930925238		
0.048441691	0.336461558	0.930412735	0.147051489	0.815838378		

10.2. Multi Class Classification Model

The following statistics will be displayed for the Multi Class Classification Model via the 'Summary' tab.

- **Overall Statistics**
 - i) The overall statistics of each model can be viewed in tabular format.
 - ii) Each model will be displayed as rows.
 - iii) Each column displays the following statistical information:
 1. Precision
 2. Recall
 3. Accuracy
 4. F measure
 5. Weighted Precision
 6. Weighted Recall
 7. Weighted F measure
 8. Weighted F Measure (beta 4)
- **Label wise Statistics of each Model**
 - i) Number of classes will be number of rows

- ii) Statistics for each class(row) will be shown in corresponding columns.
- iii) Columns in label wise statistics of each model.

1. Precision
2. Recall
3. F Measure
4. F Measure (beta 4)
5. True Positive Rate
6. False Positive Rate

- **Confusion Matrix**

- i) The Confusion matrix of each model can be viewed under the confusion matrix header.
- ii) A column consists of Actual labels and a row consists of Predicted labels.

Model Name	Precision	Recall	Accuracy	FMeasure	Weighted Precision	Weighted Recall	Weighted FMeasure	Weighted FMeasure(beta .4)
Model 0	0.4996401439424230	0.4996401439424230	0.4996401439424230	0.4996401439424230	0.470678258039868570	0.4996401439424230	0.3337222168270502	0.2714897150423612

Labels	Precision	Recall	FMeasure	FMeasure(beta .4)	TruePositiveRate	FalsePositiveRate
0.0	0.49961980229719455	0.9992795965740815	0.6661686232657418	0.5366302673842765	0.9992795965740815	0.9988815211312615
1.0	0.5217391304347826	0.0011321822813472968	0.002259461495010356	0.008098485024784155	0.0011321822813472968	7.633058080632849E-4
2.0	0.0	0.0	0.0	0.0	0.0	0.0
3.0	0.0	0.0	0.0	0.0	0.0	0.0
4.0	0.0	0.0	0.0	0.0	0.0	0.0
5.0	0.0	0.0	0.0	0.0	0.0	0.0
6.0	0.0	0.0	0.0	0.0	0.0	0.0
7.0	0.0	0.0	0.0	0.0	0.0	0.0
8.0	0.0	0.0	0.0	0.0	0.0	0.0
9.0	0.0	0.0	0.0	0.0	0.0	0.0

	Actual 0.0	Actual 1.0	Actual 2.0	Actual 3.0	Actual 4.0	Actual 5.0	Actual 6.0	Actual 7.0	Actual 8.0	Actual 9.0
Predict 0.0	12484.0	9.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Predict 1.0	10587.0	12.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Predict 2.0	1206.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Predict 3.0	511.0	2.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Predict 4.0	83.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Predict 5.0	54.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Predict 6.0	38.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Predict 7.0	6.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Predict 8.0	5.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Predict 9.0	5.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

11. Data Writer(s)

Data Writers are provided to store the results of the predictive analysis in flat files or databases for further in-depth analysis.

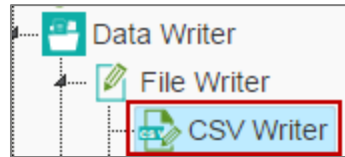
11.1. File Writer

Users can write output data to flat files like CSV, TEXT, and DAT files using the File Writer.

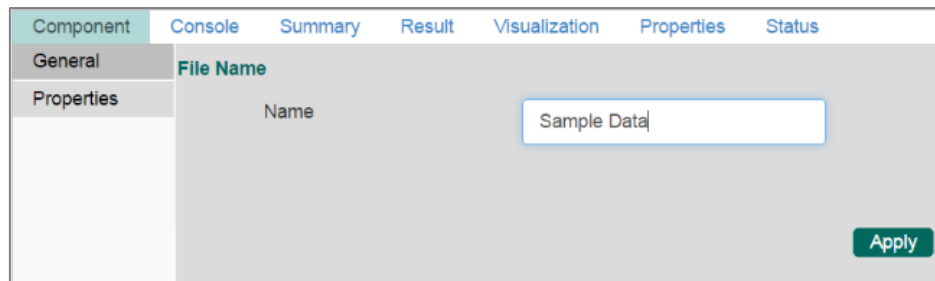
11.1.1. CSV Writer



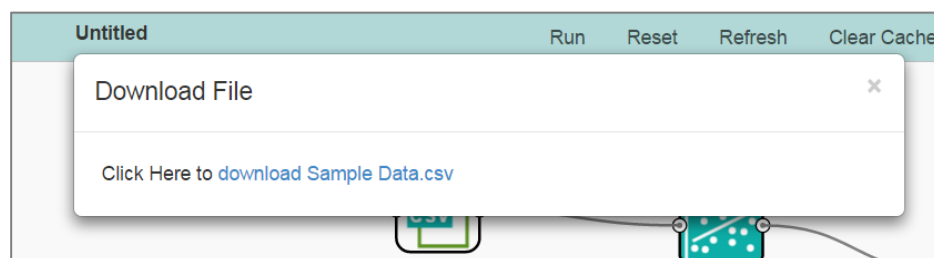
- i) Click **'TreeNode'** provided next to the **'Data Writer'** option.
- ii) Select **'File Writer'** option.
- iii) Select and drag **'CSV Writer'** component to the workspace.



- iv) Connect the **'CSV Writer'** to a configured data source.
- v) Click on CSV Writer component to access component properties.
- vi) Enter **'File Name'** in the displayed field.
- vii) Click **'Apply'**.



- viii) Click **'Run'** or **'Run Till Here'** option.
- ix) A pop-up message will appear with a link to download the CSV file.



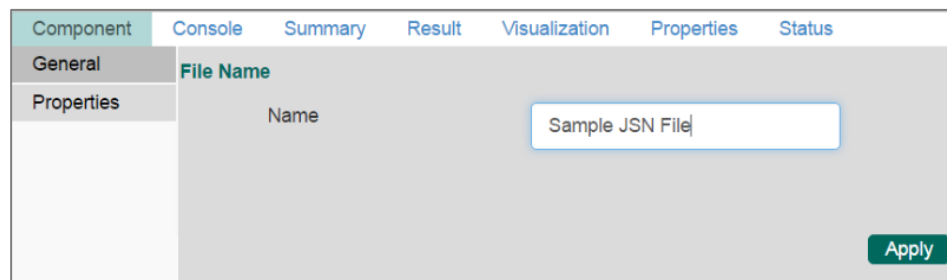
- x) Click the link to download the CSV file.

11.1.2. JSON Writer

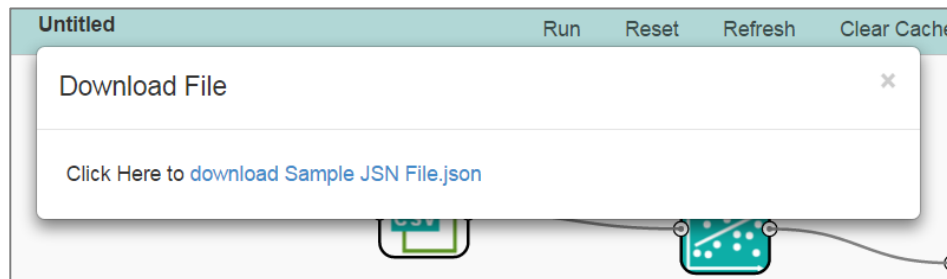
- i) Click on **'TreeNode'** provided next to the **'Data Writer'** option.
- ii) Select **'File Writer'** option.
- iii) Select and drag **'JsonWriter'** component to the workspace.



- iv) Connect the '**JsonWriter**' to a configured data source.
- v) Click on '**JsonWriter**' component to access component properties.
- vi) Enter '**File Name**' in the displayed field.
- vii) Click '**Apply**'.



- viii) Click on '**Run**' or '**Run Till Here**' option.
- ix) A Pop-up message will appear with a link to download the '**Json**' file.



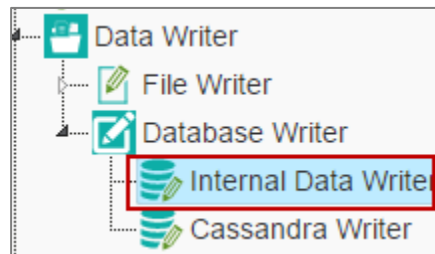
- x) Click the link to download the JSON file.

11.2. Database Writer

11.2.1. Internal Data writer

This data writer will store the data into databases like MySQL, MSSQL, and Oracle.

- i) Click '**TreeNode**' provided next to the '**Data Writer**' option.
- ii) Select '**Database Writer**' option.
- iii) Select and drag '**Internal Data Writer**' component to the workspace.



- iv) Connect the '**Internal Data Writer**' component to a configured data source.
- v) Click '**Internal Data Writer**' component to access the Component

Properties

Users will have different properties fields based on the selected table choice as described below:

a. **Selecting the 'Create a New Table' as Table Operation:**

- i. **Data Connector Name:** All the available data connectors in particular user id will be listed. Select a data connector from the drop-down menu.
- ii. **Type:** This field will be preselected based on the selected data Connector.
- iii. **Number of Rows in a batch:** Enter a number to limit the entries of rows for one batch
- iv. **Database Name:** Select a database name from the drop-down menu
- v. **Password:** Enter the database password
- vi. **Table Name:** Select '**Create New Table**' option from the list
- vii. **Create New Table:** It is an optional field. It appears only when the user selects '**Create New Table**' option from the '**Table Name**' drop down menu
- viii. **Column Selected from model:** Select columns that are needed to be written into the selected data base.



Component	Console	Summary	Result	Visualization	Properties	Status																								
General	Internal Data Writer Properties																													
Properties	<table> <tr> <td>Data Connector Name</td> <td>sample ▾</td> <td></td> </tr> <tr> <td>Type</td> <td>mysql</td> <td></td> </tr> <tr> <td>Number of Rows in a batch</td> <td>1000</td> <td>i</td> </tr> <tr> <td>Database Name</td> <td>school_data_mart ▾</td> <td></td> </tr> <tr> <td>Password</td> <td>*****</td> <td></td> </tr> <tr> <td>Table Name</td> <td>Create New Table ▾</td> <td></td> </tr> <tr> <td>Create New Table</td> <td>Sampletable</td> <td>i</td> </tr> <tr> <td>Column selected from model</td> <td>6 checked ▾</td> <td></td> </tr> </table>						Data Connector Name	sample ▾		Type	mysql		Number of Rows in a batch	1000	i	Database Name	school_data_mart ▾		Password	*****		Table Name	Create New Table ▾		Create New Table	Sampletable	i	Column selected from model	6 checked ▾	
Data Connector Name	sample ▾																													
Type	mysql																													
Number of Rows in a batch	1000	i																												
Database Name	school_data_mart ▾																													
Password	*****																													
Table Name	Create New Table ▾																													
Create New Table	Sampletable	i																												
Column selected from model	6 checked ▾																													
						Apply																								

b. Selecting an Existing Table as Table Operation:

- i. **Data Connector Name:** Select a data connector from the drop-down menu
- ii. **Type:** Displays a type based on the selected data connector
- iii. **Number of Rows in a batch:** Enter a number to limit the entries of rows for one batch
- iv. **Database Name:** Select a database name from the drop-down menu
- v. **Password:** Enter the database password
- vi. **Table Name:** Select an existing table name from the drop-down menu
- vii. **Table Operation:** Select an option using the drop-down menu. The following are the provided choices:
 1. Append Table
 2. Overwrite Table
- viii. **Column Selected from model:** Select columns that are needed to be written into the selected data base.
- ix. **Details of the Selected table:** Displays column headers from the selected table.



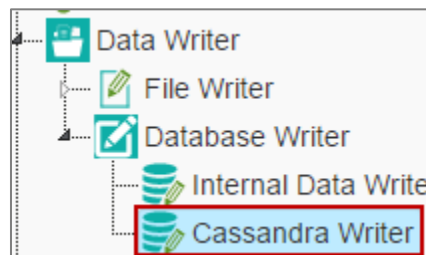
Component	Console	Summary	Result	Visualization	Properties	Status
General	Internal Data Writer Properties					
Properties	<p>Data Connector Name: QA_predictive ▾</p> <p>Type: mysql</p> <p>Number of Rows in a batch: 1000</p> <p>Database Name: predictive_analysis_v2 ▾</p> <p>Password:</p> <p>Table Name: Demo_churn ▾</p> <p>Table Operation: Overwrite Table ▾</p> <p>Column selected from model: 17 checked ▾</p> <p>Details of the selected table</p> <p>Year Employee Age Degree Salary</p>					
						Apply

- vi) Click **'Apply'**.
- vii) Click **'Run'** or **'Run Till Here'** option (by selecting the data writer component).
- viii) The data will be saved in the selected database.

11.2.2. Cassandra Writer

Cassandra Writer can be used to store predictive executions.

- i) Click **'TreeNode'** provided next to the **'Data Writer'** option.
- ii) Select **'Database Writer'**.
- iii) Select and drag **'Cassandra Writer'** component to the workspace.



- iv) Connect the **'Cassandra Writer'** to a configured data source.
- v) Click the **'Cassandra Writer'** component to access the component **Properties:**
 - a. **Selecting the 'Create a New Table' as Table Operation:**



- i. **Select Data Connector:** Select a data connector using the drop-down menu
 - ii. **Host Name:** Based on the selected data connector a host name will be displayed (Users cannot edit this field).
 - iii. **Port Name:** The server port number will be displayed (Users cannot edit this field).
 - iv. **Username:** Username of the selected connection appears by default. (Users cannot edit this field).
 - v. **Password:** the data base password
 - vi. **No. of rows in a batch:** Enter a numbe to limit the entieres of rows for one batch
 - vii. **Select Key Space:** Select a key space using the drop-down menu
 - viii. **Replication Factor:** The replication factor mentioned in the selected 'Key Space' will be displayed (Users cannot edit this field)
 - ix. **Select Table:** Select 'Create a New Table table from the drop-down menu
 - x. **Select Columns:** Select the columns that you want to write.
 - xi. **Consistency:** Select an option from the drop-down menu.
 - xii. **New Table:** Provide a name for the newly created table.
 - xiii. **New time uuid column name:** Enter a UUID column name.
- vi) Click 'Next'.

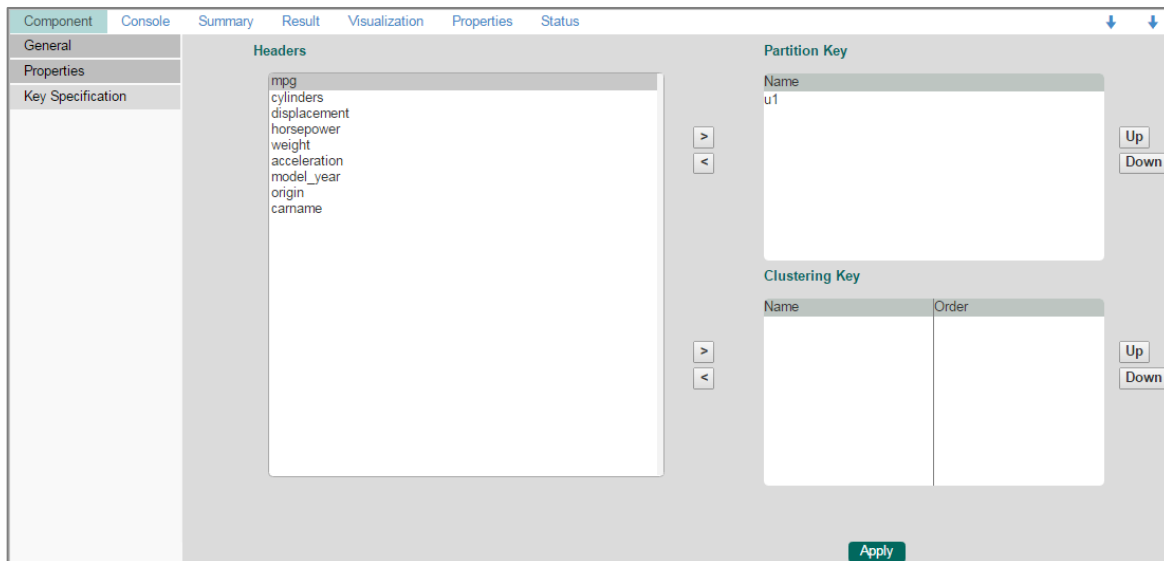


Component	Console	Summary	Result	Visualization	Properties	Status
General	Data Service Properties					
Properties						
Key Specification						
	Select Data Connector	cassandra ▾				
	Host name	192.168.1.17				
	Port Number	9042				
	Username	smb				
	Password	*****				
	No: of rows in a batch	100				
	Select Key Space	UCI ▾				
	Replication Factor	3				
	Select Table	Create new table ▾				
	Select columns	9 checked ▾				
	Consistency	1 checked ▾				
	New table	pok2				
	New time uuid column name	u1				
		Next				

vii) Users will be redirected to the **'Key Specification'** tab.

viii) Configure the following information:

- i. **Headers:** All the columns from the data set will be listed.
- ii. **Partition Key (Name):** The Partition Key determines which node stores the data. It is responsible for data distribution across the nodes.
 - The UUID Column name will be displayed under the **'Partition Key'** window.
 - Users can select and move any column from **'Header'** (Select Column) to **'Partition Key'** space.
 - The schquence of the columns listed under Partition Key can be arranged by using **'Up'** or **'Down'** options.
- iii. **Clustering Key:** The Clustering Key is a storage engine process that sorts data within the partition. It determines per-partition clustering.
 - The items listed under Clustering Key box can be arranged by using **'Up'** or **'Down'** options.
 - Users can select any column from **'Header'**(Select Column) to **'Clustering Key'** space.



Note: Users will be provided with some defined consistency level while designing the Key Space which can be overridden based on the selected replica nodes. Users are provided with the following consistency options:

- One
- Two
- Three
- Quorum

b. Selecting an Existing Table as Table Operation:

- i. **Select Data Connector:** Select a data connector from the drop-down menu
- ii. **Host Name:** Enter database server details (from where the user wants to fetch data)
- iii. **Port Name:** The server port number
- iv. **Username:** Username of the selected connection appears by default (Users cannot edit this field).
- v. **Password:** the data base password
- vi. **No. of rows in a batch:** Enter a number to limit the entire rows for one batch
- vii. **Select Key Space:** Select a key space using the drop-down menu
- viii. **Replication Factor:** Replication factor in the selected 'Key Space' will be displayed (Users cannot edit this field)
- ix. **Select Table:** Select a table from the drop-down menu



- x. **Select Columns:** Select columns from the drop-down menu that users want to be written in the data writer.
- xi. **Consistency:** Select an option using the drop-down menu
- xii. **Settings:** Select an option using the drop-down menu. The following choices will be the provided:
 1. Append Table
 2. Overwrite Table

Component	Console	Summary	Result	Visualization	Properties	Status																						
General	Data Service Properties																											
Properties																												
Key Specification																												
	Select Data Connector	cassandraqa ▾																										
	Host name	192.168.1.17																										
	Port Number	9042																										
	Username	smb																										
	Password	*****																										
	No. of rows in a batch	100																										
	Select Key Space	UCI ▾																										
	Replication Factor	3																										
	Select Table	pok1 ▾																										
	Select columns	Select ▾																										
	Consistency	1 checked ▾																										
	Settings	Overwrite ▾																										
	<table border="1"> <thead> <tr> <th>Headers</th> <th>Type</th> </tr> </thead> <tbody> <tr><td>u1</td><td>TIMEUUID</td></tr> <tr><td>acceleration</td><td>INT</td></tr> <tr><td>carname</td><td>TEXT</td></tr> <tr><td>cylinders</td><td>INT</td></tr> <tr><td>displacement</td><td>INT</td></tr> <tr><td>horsepower</td><td>INT</td></tr> <tr><td>model_year</td><td>INT</td></tr> <tr><td>mpg</td><td>INT</td></tr> <tr><td>origin</td><td>INT</td></tr> <tr><td>weight</td><td>INT</td></tr> </tbody> </table>		Headers	Type	u1	TIMEUUID	acceleration	INT	carname	TEXT	cylinders	INT	displacement	INT	horsepower	INT	model_year	INT	mpg	INT	origin	INT	weight	INT				
Headers	Type																											
u1	TIMEUUID																											
acceleration	INT																											
carname	TEXT																											
cylinders	INT																											
displacement	INT																											
horsepower	INT																											
model_year	INT																											
mpg	INT																											
origin	INT																											
weight	INT																											
	Apply																											

- ix) Click **'Apply'**.
- x) Click **'Run'** or **'Run Till Here'** option (by selecting the data writer component).
- xi) The list of column headers existing in table will be displayed once users select a table.



12. Custom R Script

Users can create and add customized algorithm components by using the 'Custom R-Script' component. The created scripts will be stored under the 'Saved Scripts' option.

12.1. Creating a New R Script

- i) Click 'Custom R Script' tree-node on the Predictive Analysis home page.
- ii) Click 'Create New Script'.
- iii) Users will be directed to the 'Component' tab.
- iv) Configure the following fields in the 'General' tab:
 - a. **Basic**
 - i. **Component Name:** Enter a name or title that you wish to be saved as a saved R script.
 - ii. **Component Type:** Default Component type will be displayed in this field.
 - iii. **Description:** Describe about the Component (It is an optional field).
- v) Click 'Next'.

Component	Console	Summary	Result	Visualization	Properties	Status
General	Basic					
Script		Component Name				
Settings		Component Type				
		Description				

- vi) Users will be directed to the 'Script' tab.
- vii) Provide the following information as required:
 - a. **Script Editor**
 - i. Paste the R-script in the given space under 'Script Editor'.
 - ii. Click the 'Validate' option.
 - iii. Use 'Primary Function Details' to embed the customized R-script into function.
 - iv. Set the function details as shown below:
 1. **Primary Function Name:** Select name of the created function from the drop-down menu.
 2. **Input Data Frame:** Select a dataset (that has been used above) from a drop-down menu.
 3. **Output Data Frame:** Enter an option to which the data will be passed.
 4. **Model Variable Name:** Enter the output model variable (This field will appear only when the model summary has been enabled).




- v. If you need a visualization chart for the ensuring data, tick the **‘Show Visualization’** check-box.
- vi. If you need to show the summary, tick the **‘Show Summary’** check-box.
- viii) Click **‘Next’**.


The screenshot displays the software interface with the following components:

- Component Tabs:** Console, Summary, Result, Visualization, Properties, Status.
- Script Editor:**
 - Buttons: Validate (with a green checkmark icon), R script has been validated successfully!
 - Code Snippet:

```
kmeansfunction<-function(dataFrame,independent,Clustersize,Iterations,algotype,numberofinitialdsets){
set.seed(4321);
kmeans_model<-kmeans(data.frame(dataFrame[,independent]),centers=Clustersize,iter.max=Iterations,
nstart=numberofinitialdsets,algorithm=algotype);
output<- cbind(dataFrame,kmeans_model$cluster);
plot(dataFrame,col=kmeans_model$cluster)
return(list(out=output,model=kmeans_model));
}
```
- Primary Function Details:**
 - Primary Function Name: kmeansfunct
 - Input DataFrame: dataFrame
 - Output DataFrame: out
 - Model Variable Name: kmeans_mo
 - Checkboxes: Show Visualization, Show Summary
 - Buttons: Previous, Next


- ix) Users will be directed to the **‘Settings’** tab.
- x) Configure the following fields:
 - a. **Output Table Definition:** This option will configure number of output columns, column headers, data types.
 - i. **Consider all columns from previous component:** To display all columns from previous component.
 - ii. **Consider None:** To display no column from previous component.
 - iii. **Data Type:** Select a data type for the newly created column using the drop down list.
 - iv. **New Predicted Column Name:** Enter an appropriate name for the new predicted column.
 - v. : To remove the an added row containing **‘Data Type’** and **‘New Predicted Column Name’**.



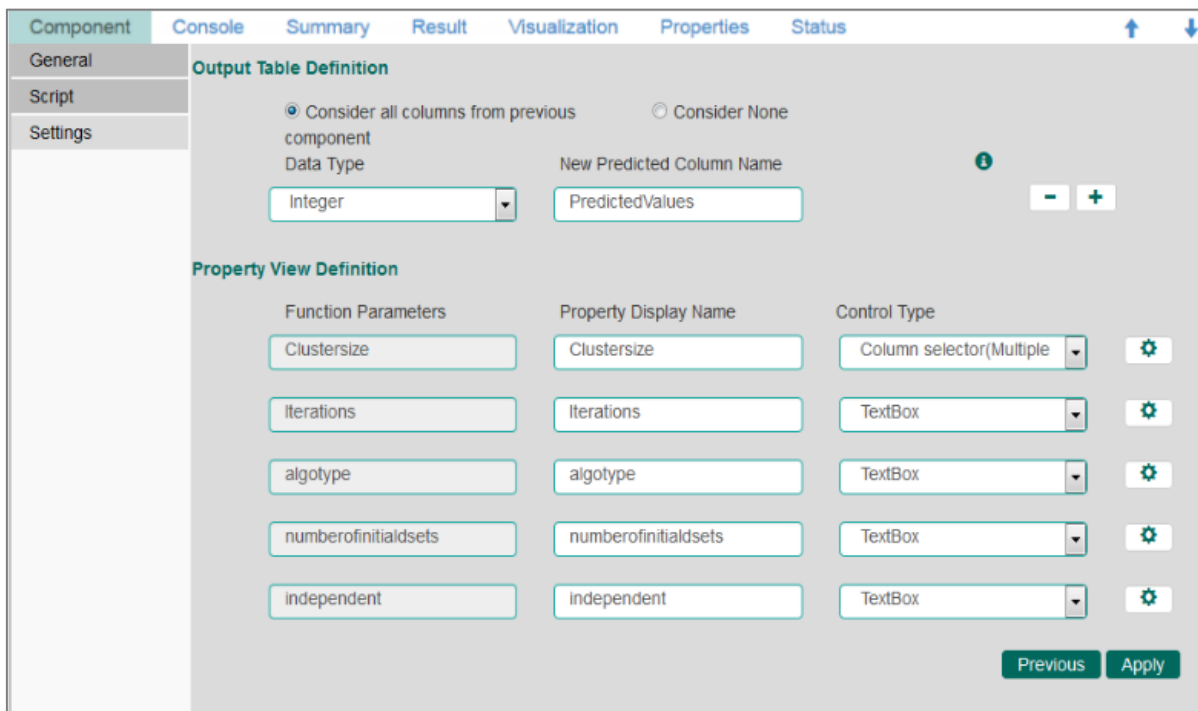
vi. : To add a new row containing ‘**Data Type**’ and ‘**New Predicted Column Name**’.

b. Property View Definition

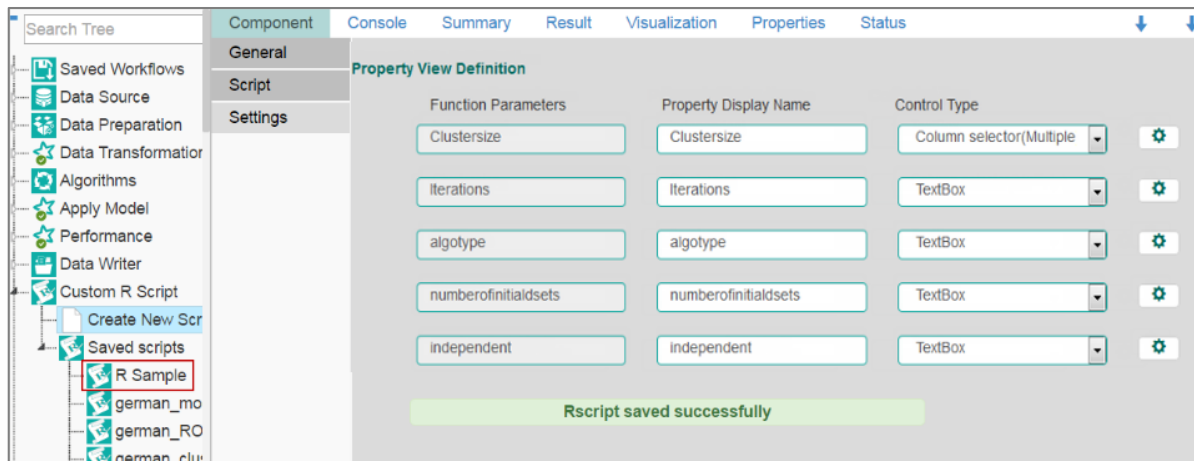
- i. **Function Parameters:** Actual names of parameters configured in the script.
- ii. **Property Display Name:** Parameter name to be displayed while configuring saved R script as a component.
- iii. **Control Type:** User can select out of the following options:
 1. Text box,
 2. Drop-down menu,
 3. Column Selector (single),
 4. Column Selector (multiple).

iv. **Settings option** : To set display for mandatory fields and validate data type for input column. This field is associated to function parameters.

xi) Click ‘**Apply**’.




xii) The newly created R Script will be saved in the ‘**Saved Scripts**’ list.



Guidelines to be followed while Writing R- Script

1. R- script needs to be written inside a valid R function. i.e. The entire code body should be inside the curly braces of the function.
2. The R-script should have at least one main function. Multiple functions are acceptable and one function can call another function, but it should be written above the calling function body. (If called function is an outer function) or above the calling statement (if called function is an inner function).
3. Any extra packages that are required to run your R script must be installed on the R-server and it should be loaded using library ('library_name') statement, before calling the associated function in your script.
4. The R-script should return data in the form of a list only, containing the data frame and model (if used).
5. In the return statement only a data frame can be assigned to the variable 'out'. This data frame supports all structures like list, string, vector, matrix, table.
6. If 'Show Visualization' field is marked as 'yes' during the creation of component, then there should be a plot created in the R-script and if 'Show Summary' field is marked as 'yes' then the structures list should have the 'model' variable.
7. Empty cells, (NULL), (null), NULL, null, /N, NA, N/A are considered as unwanted values and replaced by "NaN" in case of double, long, short, float, byte, integer, and "NA" in case of boolean, string, so instead of using these values in R code use "NaN" or "NA" according to data type of input data.

Note:

- a. Click the 'Information' button  to get the above mentioned list of rules for R-script.
- b. 'Model Variable Name' can be enabled only after selecting 'Show Summary' option.
- c. Select 'Show Summary' and 'Show Visualization' option only if, the R-script carries both the items.
- d. All the supported date data types are listed in date formats in data type definition, all other date formats are considered as string itself.
- e. Mssql data types are considered as string itself.

12.2. Saved R-Scripts

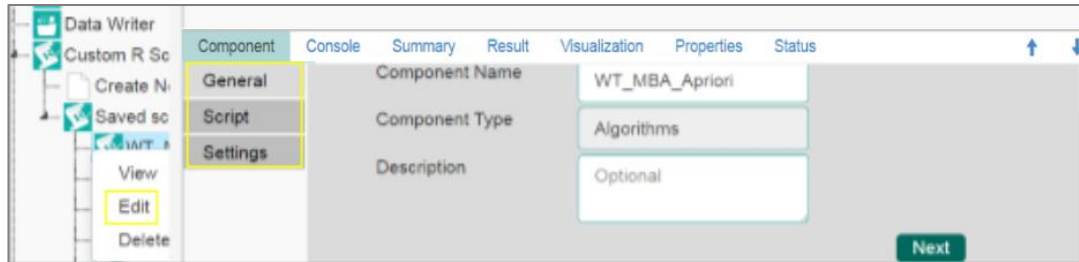
12.2.1. Viewing a Saved R Script

- i) Select an R Script from the list of 'Saved R-Script'.
- ii) Right click on the selected R Script.
- iii) A context menu will open.
- iv) Select 'View'.
- v) Users will be redirected to the 'Component' tab.



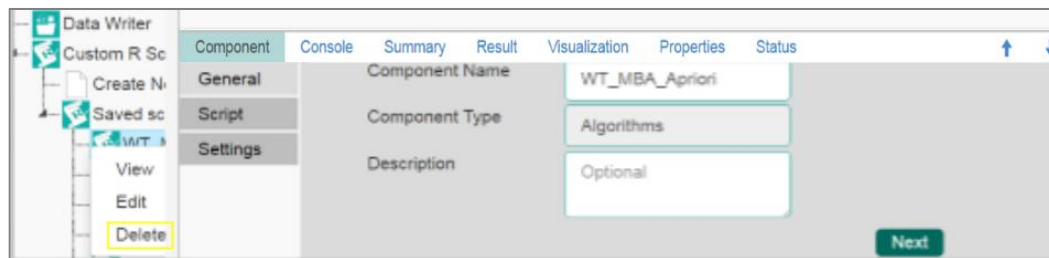
12.2.2. Editing a Saved R Script

- i) Select an R Script from the list of 'Saved R-Script'.
- ii) Right click on the selected R Script.
- iii) A context menu will open.
- iv) Select 'Edit'.
- v) Users will be redirected to the 'Component' tab.
- vi) Users can edit the required fields provided under **General**, **Script**, and **Settings** tabs.

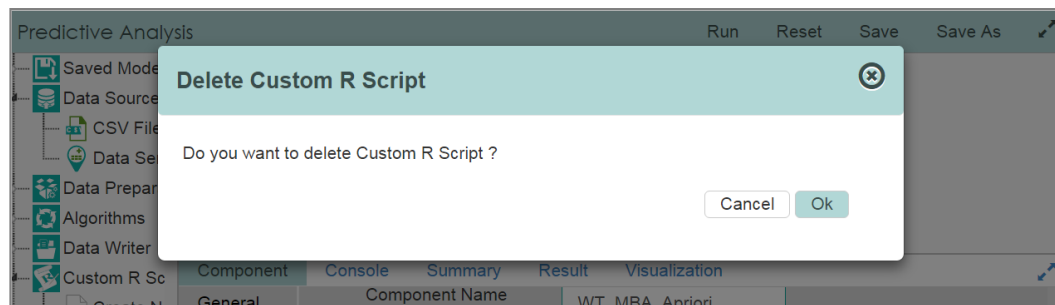


12.2.3. Deleting a Saved R Script

- i) Select an R Script from the list of **'Saved R-Script'**.
- ii) Right click on the selected R Script.
- iii) A context menu will open.
- iv) Select **'Delete'**.



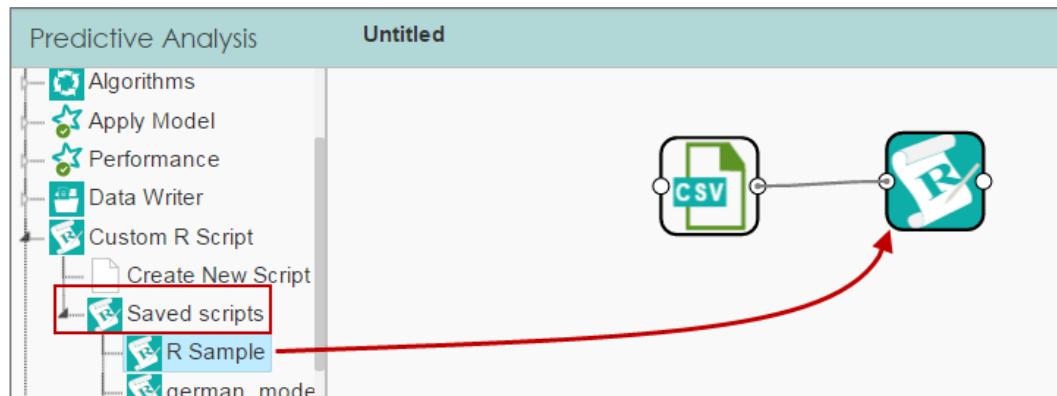
- v) A pop-up window will appear to assure the deletion.
- vi) Click **'OK'**.



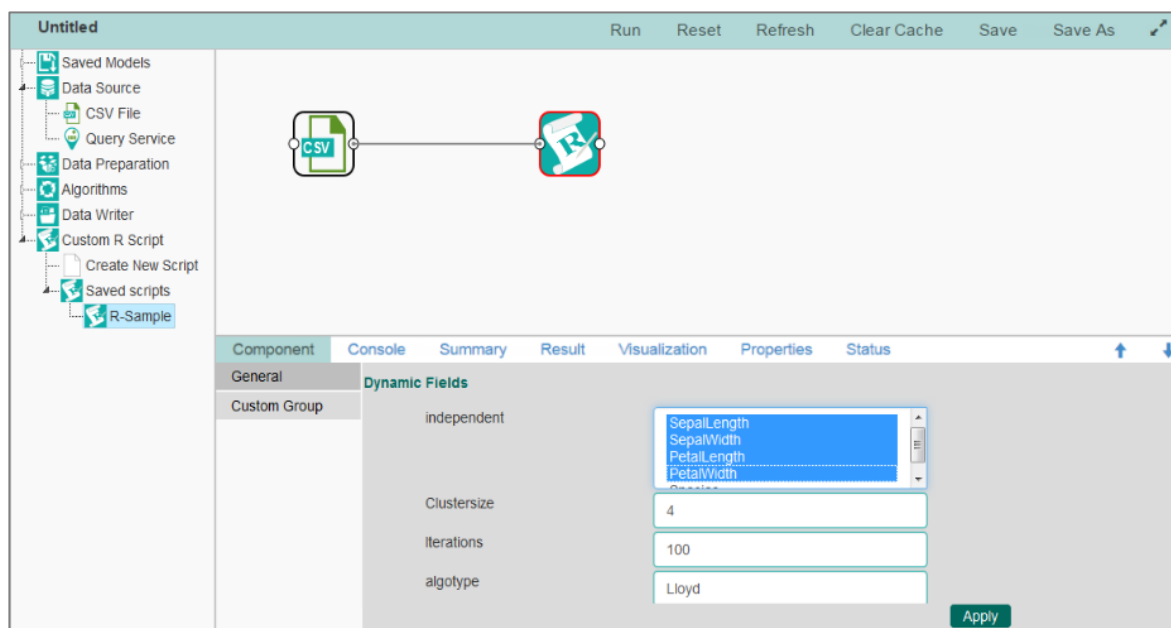
- vii) The selected R-Script will be deleted.

12.2.4. Connecting Saved R Script with a Data Source

- i) Click the **'Custom R Script'** treenode.
- ii) Select and drag a saved R-script to the workspace.
- iii) Connect the R-Script to a configured datasource component.



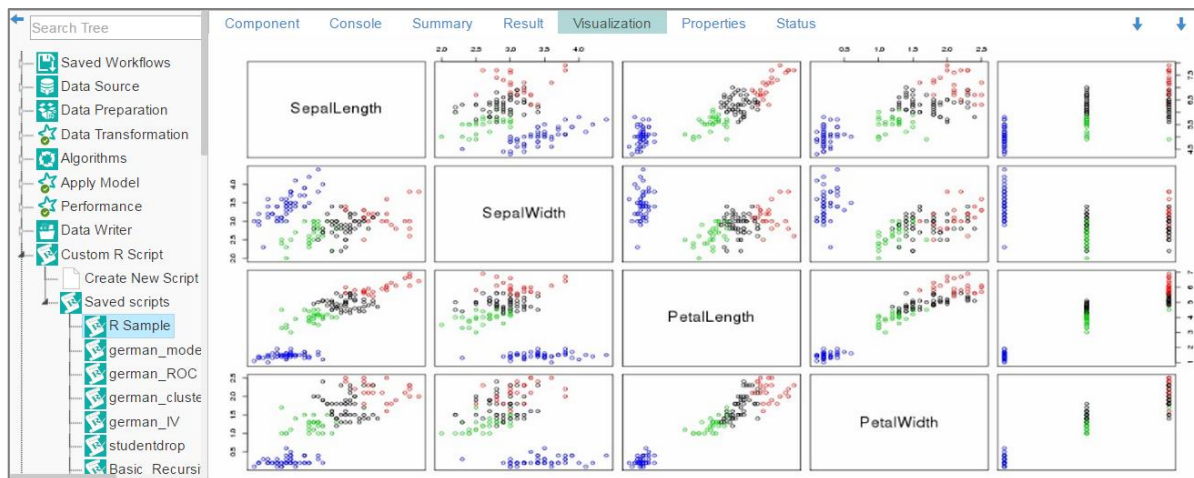
- iv) Click the **'R Script'** component.
- v) Configure the required component fields.
- vi) Click **'Apply'**.



- vii) Click **'Run'** or **'Run Till Here'**.
- viii) The **'Result'** view will be displayed.

SepalLength	SepalWidth	PetalLength	PetalWidth	Species	PredictedValues
5.1	3.5	1.4	0.2	setosa	4
4.9	3	1.4	0.2	setosa	4
4.7	3.2	1.3	0.2	setosa	4
4.6	3.1	1.5	0.2	setosa	4
5	3.6	1.4	0.2	setosa	4
5.4	3.9	1.7	0.4	setosa	4

- ix) Click the 'Visualization' tab
- x) The result data will be displayed through graphics.



Note: The above given process is displayed for a CSV data source. Similar set of steps can be followed for other datasource types.

13. Scheduler

Scheduler helps to schedule the Predictive Workflow as per the requirement.

13.1. New Schedule

This section explains steps to schedule a new job. Scheduling new job is a continuous step by step process as described below:



- i) Navigate to the Predictive home page.
- ii) Click the '**Scheduler**' tree node.
- iii) Two options will be displayed:
 - a. New Scheduler
 - b. Status
- iv) Select '**New Schedule**'.
- v) Users will be redirected to the '**General**' tab.

13.1.1. Configuring General Tab

- i) A '**General**' tab will open (by default).
- ii) Fill in the following fields:
 - a. **Model Name**: Select a model name using the drop-down menu
 - b. **Job Name**: Enter a job name
 - c. **Description**: Describe about the job (optional field)
 - d. **Use Existing Data Connector**: Use radio buttons to select an option
 - i. Select '**Yes**' to use an existing data connector
 - ii. Select '**No**' for not using an existing data connector
 - e. **Use Existing Datawriter**: Use radio buttons to select an option
 - i. Select '**Yes**' to use an existing data writer
 - ii. Select '**No**' for not using an existing data writer
- iii) Click '**Next**'.

Component	Console	Summary	Result	Visualization	Properties	Status
General	Basic					
Data Source	Model Name		modelschedule ▾			
Data Writer	Job Name		Modelschedule			
Schedule	Description		Optional			
Notification	Use Existing Data Connector		<input type="radio"/> Yes <input checked="" type="radio"/> No			
	Use Existing Datawriter		<input type="radio"/> Yes <input checked="" type="radio"/> No			
						Next

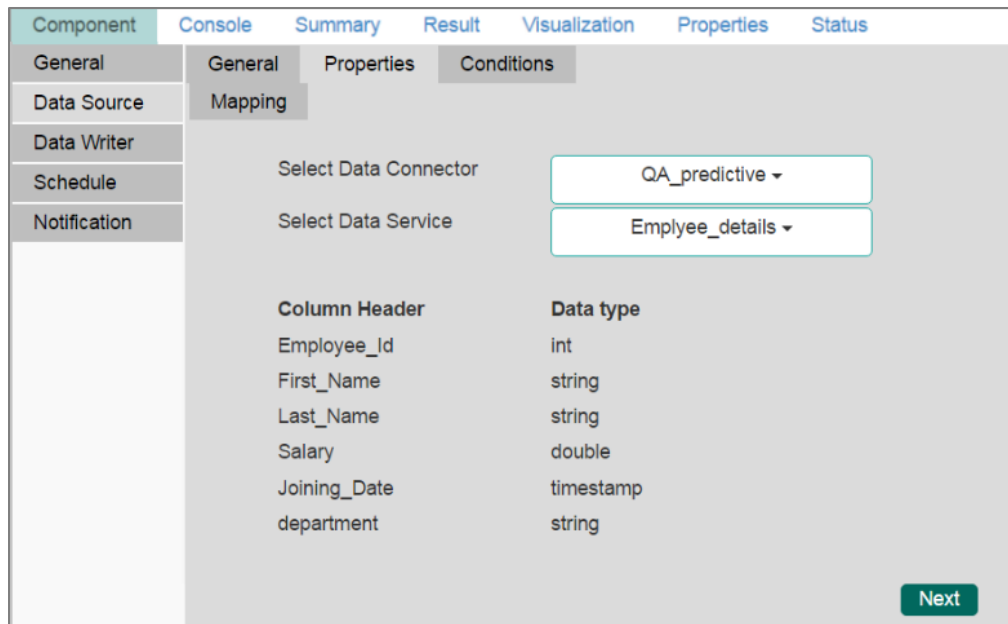


iv) Users will be redirected to the **'Data Source'** tab.

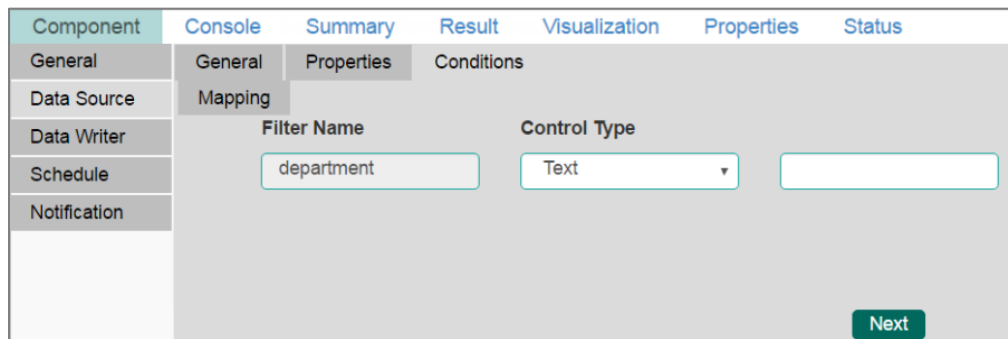
13.1.2. Configuring Data Source

- i) **'General'** fields will be displayed by default.
- ii) Users can fill in the required fields:
 - a. Component Name: A default name provided for the component.
 - b. Alias Name: User can enter a name for the component.
 - c. Description: Users can describe about the component (optional).
- iii) Click **'Next'**.

- iv) Users will be redirected to the **'Properties'** fields.
- v) Configure the following fields (to configure a new datasource):
 - a. **Select Data Connector:** Select a data connector from the drop-down menu
 - b. **Select Data Service:** Select a data service from the drop-down menu
 - c. Based on the selected data service the below given columns will be displayed
 - i. Column Header
 - ii. Data Type
- vi) Click **'Next'**.



- vii) Users will be redirected to the **‘Conditions’** tab. (If conditions are available, else the data source configuration will end at the previous step.)
- viii) Configure the required fields.
- ix) Click **‘Next’**.



- x) Users will be redirected to the **‘Mapping’** tab.
- xi) Configure the column header information from the data service that will be used for the selected model columns.
- xii) Click **‘Next’**.



Component	Console	Summary	Result	Visualization	Properties	Status
General	General	Properties	Conditions			
Data Source	Mapping					
Data Writer	Column selected from model			Column Header from data service		
Schedule	<input type="text" value="SepalLength"/> <input type="text" value="SepalWidth"/> <input type="text" value="PetalLength"/> <input type="text" value="PetalWidth"/> <input type="text" value="Species"/>			<input type="text" value="Employee_Id"/> ▾ <input type="text" value="First_Name"/> ▾ <input type="text" value="Last_Name"/> ▾ <input type="text" value="Salary"/> ▾ <input type="text" value="Joining_Date"/> ▾		
Notification	<input type="button" value="Next"/>					

xiii) Users will be redirected to the ‘**Data Writer**’ tab.

Note: Data source tab will be enabled only when an existing data connector is not selected from the Use Existing Data Connector field.

13.1.3. Configuring a Data Writer

- i) Fill in the required details to configure a data writer:
- ii) Click ‘**Next**’.

Component	Console	Summary	Result	Visualization	Properties	Status
General	Data Writer					
Data Source	Data Connector Name: <input type="text" value="QA_predictive"/> ▾					
Data Writer	Type: <input type="text" value="mysql"/>					
Schedule	Number of Rows in a batch: <input type="text" value="1000"/> ⓘ					
Notification	Database Name: <input type="text" value="23test"/> ▾					
	Password: <input type="text" value="....."/>					
	Table Name: <input type="text" value="Create New Table"/> ▾					
	Create New Table: <input type="text" value="Sampletable"/> ⓘ					
	Column Selected: <input type="text" value="7 checked"/> ▾					
	<input type="button" value="Next"/>					

iii) Users will be redirected to the '**Schedule**' tab

Note: Data Writer tab will be enabled only when existing data writer is not selected from the Use Existing Data Writer field.

13.1.4. Scheduling a New job

Users can select time to schedule a new job using this section. As per the selected scheduling time, refresh interval option will be provided.

i) **Start Date:** Select a start date and time for the scheduled job (It should be greater than the Current System Date and Time)

ii) **Select a Job Refresh Interval option:**

E.g. When selected time range is '**Hourly**', the selected interval option can be as described below:

Every_hour: Selecting this option will refresh the scheduled job after every selected interval.

OR

At: Selecting this option will refresh the scheduled job at the selected hour.

iii) **End Date:** Select an end date and time for the scheduled job. (It should be greater than the Start date and the Current System Date and Time)

iv) **Run Now:** Select this option to run the scheduled job on apply.

v) Click '**Next**'.

- **Hourly:** By selecting this option users can schedule the job on hourly basis.

Job Refresh Interval Details

1. Select a specific hour by using the below given options:

Every_hour: Selecting this option will refresh the scheduled job after selected hourly interval.

OR

At: Selecting this option will refresh the scheduled job at the selected hour.



Component	Console	Summary	Result	Visualization	Properties	Status
General	Hourly	Daily	Weekly	Monthly		
Data Source	Yearly					
Data Writer						
Schedule						
Notification						

Start Date

Every hour(s)

At

End Date

Run Now

[Next](#)

- **Daily:** By selecting this option users can schedule the job on daily basis.

Job Refresh Interval Details

1. Select a specific day by using the below given options:

Every_ Days: the scheduled job will be refreshed after every selected number of days.

OR

Every Week Day: the scheduled job will be refreshed daily till the end date.

2. Select Start time.



Component	Console	Summary	Result	Visualization	Properties	Status
General	Hourly	Daily	Weekly	Monthly		
Data Source	Yearly					
Data Writer						
Schedule						
Notification						

Start Date

Every Days
 Every Week Day

Start Time

End Date

Run Now

[Next](#)

- **Weekly:** By selecting this option users can schedule the job on weekly basis.

Job Refresh Interval Details

1. Select a day **or** days of week when the scheduled job can be refreshed.
2. Select a start time.

Component	Console	Summary	Result	Visualization	Properties	Status
General	Hourly	Daily	Weekly	Monthly		
Data Source	Yearly					
Data Writer						
Schedule						
Notification						

Start Date

Monday Tuesday Wednesday Thursday Friday
 Saturday Sunday

Start Time

End Date

Run Now

[Next](#)

- **Monthly:** By selecting this option users can schedule the job on monthly



basis. This time range is for more than one month.

Job Refresh Interval Details

1. Select a specific day of month by using the below given options:

E.g. 1st day of 1st month

OR

E.g. The First Monday of the 1st month

2. Select Start time

Component	Console	Summary	Result	Visualization	Properties	Status
General	Hourly	Daily	Weekly	Monthly		
Data Source	Yearly					
Data Writer						
Schedule						
Notification						

Start Date

Day of every month(s)
 The of every month(s)

Start Time

End Date

Run Now

[Next](#)

- **Yearly:** By selecting this option users can schedule the job on yearly basis. This time range is for more than one year.

Job Refresh Interval Details

1. Select a specific day of month by using the below given options:

Select Every 1st day of January month.

Or

Select the first Monday of January

2. Select Start time



Component	Console	Summary	Result	Visualization	Properties	Status
General	Hourly	Daily	Weekly	Monthly		
Data Source	Yearly					
Data Writer						
Schedule	Start Date <input type="text" value="Tue Jun 06 2016 01:00:00 G"/>					
Notification	<input checked="" type="radio"/> Every <input type="text" value="January"/> <input type="text" value="1"/>					
	<input type="radio"/> The <input type="text" value="First"/> <input type="text" value="Monday"/> of <input type="text" value="January"/>					
	Start Time <input type="text" value="12"/> <input type="text" value="00"/>					
	End Date <input type="text" value="Tue Jul 04 2016 02:00:00 GI"/>					
	<input checked="" type="checkbox"/> Run Now					
	Next					

Users will be redirected to the '**Notification**' tab.

Note: If users select 'Use Existing Data Connector' and 'Use Existing Data Writer', then Schedule tab will appear immediately after General tab.

13.1.5. Notification

- i) Configure the below given fields:
 - a. **Enable Email Notification:** Use a check mark in the box to enable email
 - b. **Email Address:** Enable this option by check marking the box
 - c. **Send Mail when R Server is not running:** Users can check mark in the box to enable this option. By enabling this option, users will get email when R server is not running.
 - d. **Send Mail when Process is Completed Successfully:** Users can check mark in the box to enable this option. By enabling this option users will get mail after the process is successfully completed.
 - e. **Send Mail when the Process is a Failure:** Users can check mark in the box to enable this option. By enabling this option users will get mail when the process fails.
- ii) Click '**Apply**' to save the details.



Component Console Summary Result Visualization Properties Status

General **Email Notification**

Data Source Enable Email Notification

Data Writer Email Address

Schedule

Notification

Send mails when R server is not running

Send mail when process is completed successfully

Send mails when the process is a failure

Apply

iii) A pop-up window will appear to assure that the job/process has been scheduled.

Success

Process has been scheduled.

Ok

Name	Frequency	Start Date	End Date	Next Run	Status	Scheduled By	Model Name
WT MBA	Hourly	9/Feb/2016-23:0:0	10/Feb/2016-23:0:0	NA	Stopped	Avin Jain	WT_MBA_Apriori

iv) The scheduled job/ process will be added to the scheduler list.

Component Console Summary Result Visualization Properties Status

Refresh

Search:

Task Name	Frequency	Start Date	End Date	Next Run	Status	Scheduled By	Model Name
WT MBA	Hourly	9/Feb/2016-23:0:0	10/Feb/2016-23:0:0	NA	Stopped	Avin Jain	WT_MBA_Apriori
Maruti Suzuki	Yearly	8/Apr/2016-0:0:0	30/Apr/2017-0:0:0	1/Jan/2017-12:0:0	Active	Avin Jain	Automobile_Maruti_Suzuki_India
Sample	Hourly	11/Apr/2016-16:0:0	30/Apr/2016-16:0:0	11/Apr/2016-16:0:0	Active	Avin Jain	Automobile_Maruti_Suzuki_India

Showing 1 to 3 of 3 entries

Previous 1 Next

Note:

















- a. The PDF summary will be sent through email for the scheduled workflows.
- b. Multiple email addresses can be entered in comma separated value.
- c. At present, Spark Workflows are not supported by Scheduler.

13.2. Status

This section will display detailed information for all the scheduled jobs.





- i) Click the **'Scheduler'** treenode.
- ii) Select **'Status'**.
- iii) A list containing all the scheduled jobs will be displayed.

Task Name	Frequency	Start Date	End Date	Next Run	Status	Scheduled By	Model Name
WT MBA	Hourly	9/Feb/2016-23:0:0	10/Feb/2016-23:0:0	NA	Stopped	Avin Jain	WT_MBA_Apriori
Maruti Suzuki	Yearly	8/Apr/2016-0:0:0	30/Apr/2017-0:0:0	1/Jan/2017-12:0:0	Active	Avin Jain	Automobile_Maruti_Suzuki_India
Sample	Hourly	11/Apr/2016-16:0:0	30/Apr/2016-16:0:0	11/Apr/2016-16:0:0	Active	Avin Jain	Automobile_Maruti_Suzuki_India
samplejob	Hourly	12/Apr/2016-12:0:0	30/Apr/2016-0:0:0	12/Apr/2016-12:0:0	Active	Avin Jain	WT_MBA_Apriori

Data Source	Data Base	Table	Logs	Actions
MBA_Input	wt_model	PA_MBA_Apriori	View Logs	   
Automobile_Forecasting_PA	bdi_demo_datamart	Automobile_Forecasting	View Logs	   
Automobile_Forecasting_PA	bdi_demo_datamart	Automobile_Forecasting	View Logs	   
MBA_Input	wt_model	PA_MBA_Apriori	View Logs	   

Click **'View Log'** to see the logs of the selected workflow under the **'Console'** tab.

Related Actions for a Scheduled Job:

Options	Name	Description
	Edit	To edit/update the scheduled job details
	Stop	To stop the scheduled job
	Remove	To remove the scheduled job from the list
	Start	To start the scheduled job

Note:

- a. **'Edit'** option will allow the user to update/ edit **'General'**, **'Schedule'**, and **'Notification'** tabs for the given job.
- b. Users can click **'Start'** button to restart the scheduler for a scheduled job until it reaches the end date.
- c. Users can enable **'Edit'** and **'Remove'** actions only after stopping the scheduler.

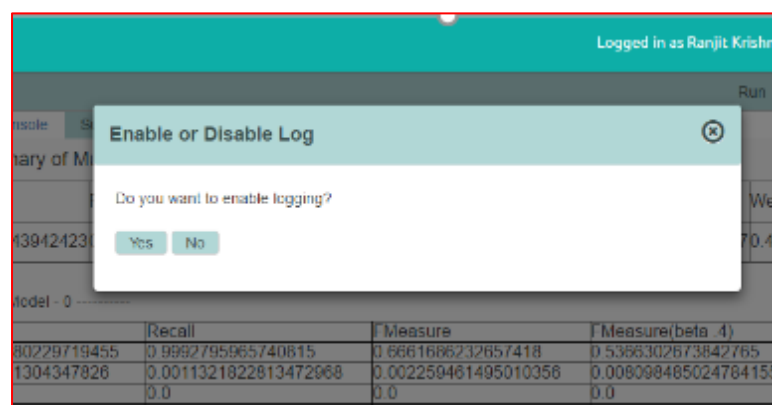
14. Live Job Status

Users can monitor spark processes using the **'Live job Status'** feature. The **'Live Job Status'** option will be a new tree node on the existing tree structure and Spark will be a leaf node to the new tree node.

a. Enable/Disable log

Users need to enable logging to view the log in live job status in Spark after running a workflow.

- i) Create a workflow in Spark.
- ii) Click **'Run'** on the menu row.
- iii) A pop-up window asking whether to enable or disable log will appear.
- iv) Click **'Yes'** to enable logging. (Selecting **'No'** will not log in live job status.)



- v) Click the **'Live Job Status'** tree node from the tree structure.
- vi) Click the **'Spark'** leaf node.
- vii) A data grid will appear in the **'Status'** tab.




Workflow Name	Run by	Start time	End Time	Status	View Log	Live job status	Summary	Actions
CSS 22nd Sept	Ranjit Krishnan	Thu, 22 Sep 2016 09:18:07 GMT	NA	in progress				
CSS 22nd Sept	Ranjit Krishnan	Thu, 22 Sep 2016 09:11:41 GMT	NA	in progress				
css	Ranjit Krishnan	Wed, 21 Sep 2016 13:36:40 GMT	NA	in progress				

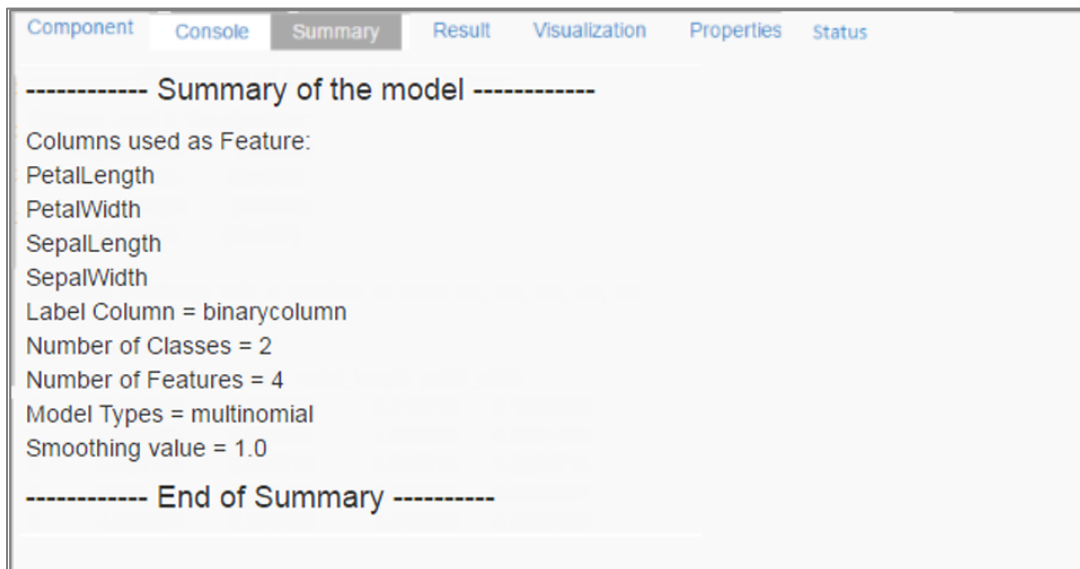
b. **View Log:** A log of the completed workflow can be viewed under the 'Console' tab by clicking the 'View Log' icon

Component	Console	Summary	Result	Visualization	Properties	Status
	02/Aug/2016 - 10:49:28					DataReader Cassandra Running
	02/Aug/2016 - 10:50:34					DataReader Cassandra Completed
	02/Aug/2016 - 10:50:34					Data Split Running
	02/Aug/2016 - 10:50:34					Data Split Completed
	02/Aug/2016 - 10:50:34					NavieBayes Running

c. **Live Job Status:** If the workflow execution is still in progress, users can view live action by clicking the 'Live Job Status' icon . Live jobs will be displayed under the 'Console' tab.

Component	Console	Summary	Result	Visualization	Properties	Status
	2/8/2016 - 10:54:12					Process started
	2/8/2016 - 10:54:15					Initiating Process
	2/8/2016 - 10:54:15					DataReader Cassandra Running
	2/8/2016 - 10:54:19					Job Id-0 : 0 task completed out of 4 with 0 failed task

- d. **Summary:** Click the ‘Summary’ icon  to view a consolidated summary of all the components in a workflow. It will be displayed under the ‘Summary’ tab.



e. **Actions**

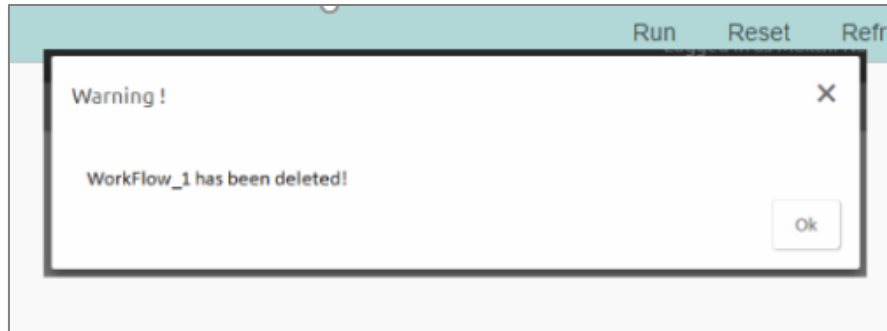
- i. **Stop:** Users can stop an execution at any time by clicking on the stop button. The status of the process will change to ‘Cancelled’ if the execution has been stopped.

EndTime	Status	ViewLog	LiveJobStatus	ViewSummary	Actions
20/July/2016-17:0:0	Cancelled				
20/July/2016-17:0:0	Success				
20/July/2016-17:0:0	Failed				

- ii. **Delete:** Click the ‘Delete’ icon to remove an execution.

EndTime	Status	ViewLog	LiveJobStatus	ViewSummary	Actions
20/July/2016-17:0:0	Success				
20/July/2016-17:0:0	Failed				

The selected workflow will be deleted from the ‘Live Job Status’ table and a warning pop-up message will be displayed.

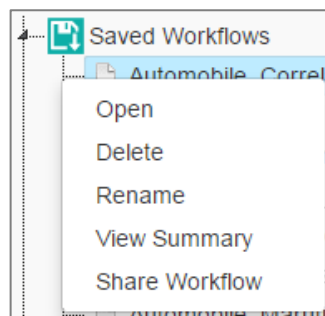
**Note:**

- a. Click '**Refresh**' to refresh the table for viewing a live job.
- b. Click '**Remove all jobs**' to delete all the jobs from the table.

15. Saved Workflows

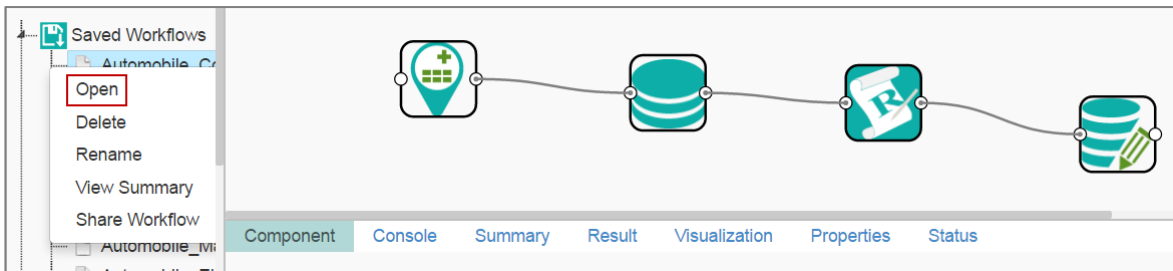
Users can save a workflow by clicking the '**Save**' button provided on the workspace menu row. All the saved workflows will be displayed under the '**Saved Workflow**' tree node. This section explains various options assigned to a saved workflow.

- i) Navigate to the Predictive home page.
- ii) Click '**Saved Workflow**' tree node.
- iii) A list of all the saved workflows will be displayed.
- iv) Right click on a work flow from the list of '**Saved Workflows**'.
- v) A context menu will open with various options (As shown below):

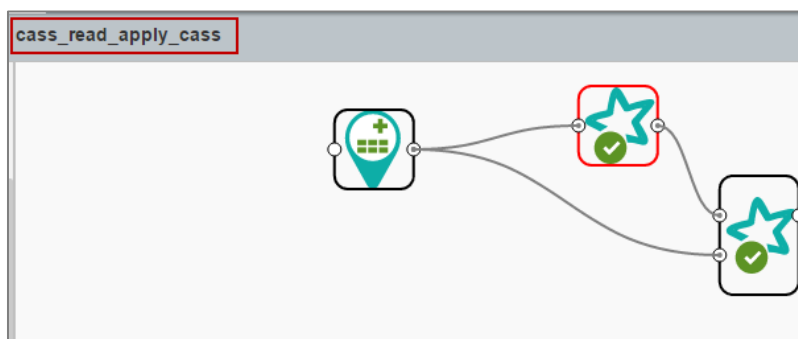


15.1. Opening a Workflow

- i) Right click on a work flow from the list of '**Saved Workflows**'.
- ii) Select '**Open**' from the context menu.
- iii) The selected workflow will be displayed on the right pane of the screen.

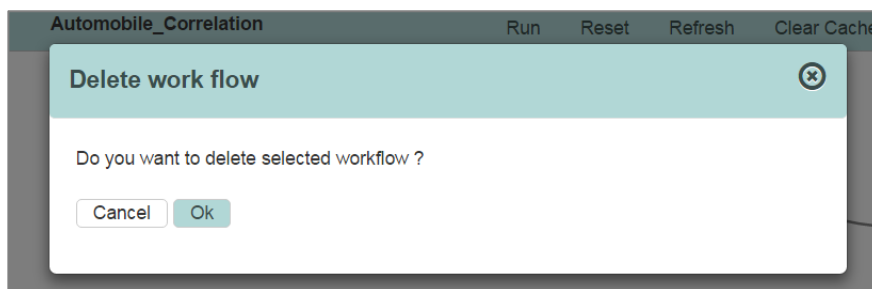


Note: When opening a saved workflow, the workflow name will be displayed on the left side of the workspace menu row.



15.2. Deleting a Workflow

- i) Right click on a work flow from the list of **'Saved Workflows'**.
- ii) Select **'Delete'** from the context menu.
- iii) A pop-up window will appear to confirm the deletion.
- iv) Click **'OK'**.

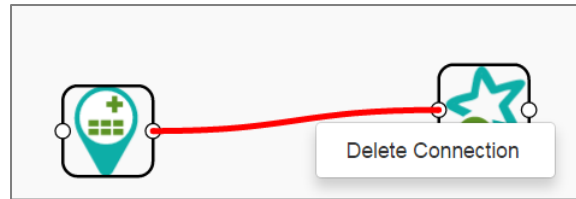


- v) The selected workflow will be deleted from the list.

15.2.1. Delete Connection for a Workflow

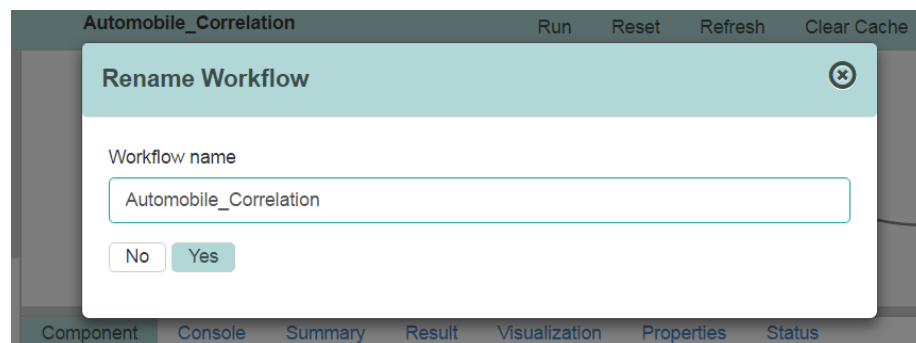
A Right click on the inter-node connection will display the **'Delete Connection'** option in a workflow.

Click the **'Delete Connection'** option to delete a connection.



15.3. Renaming a Workflow

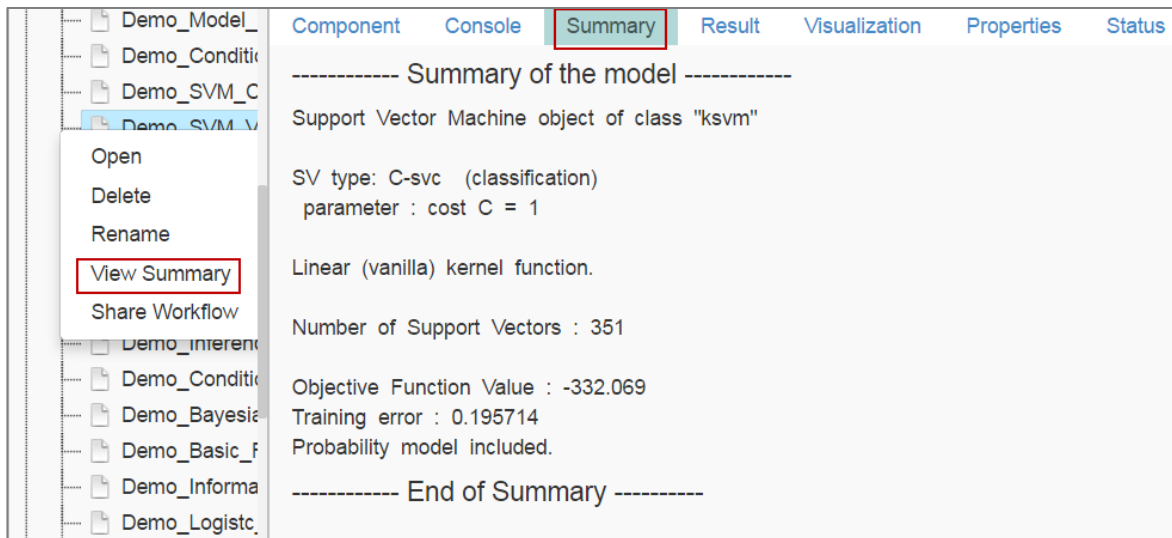
- i) Right click on a work flow from the list of '**Saved Workflows**'.
- ii) Select '**Rename**' from the context menu.
- iii) A pop-up window will appear.
- iv) Enter a new/modified name for the workflow.
- v) Click '**Yes**'.



- vi) The selected workflow will be renamed.

15.4. Viewing Summary

- i) Right click on a work flow from the list of '**Saved Workflows**'.
- ii) Select '**View Summary**' from the context menu.
- iii) The workflow summary will be displayed under the '**Summary**' option.

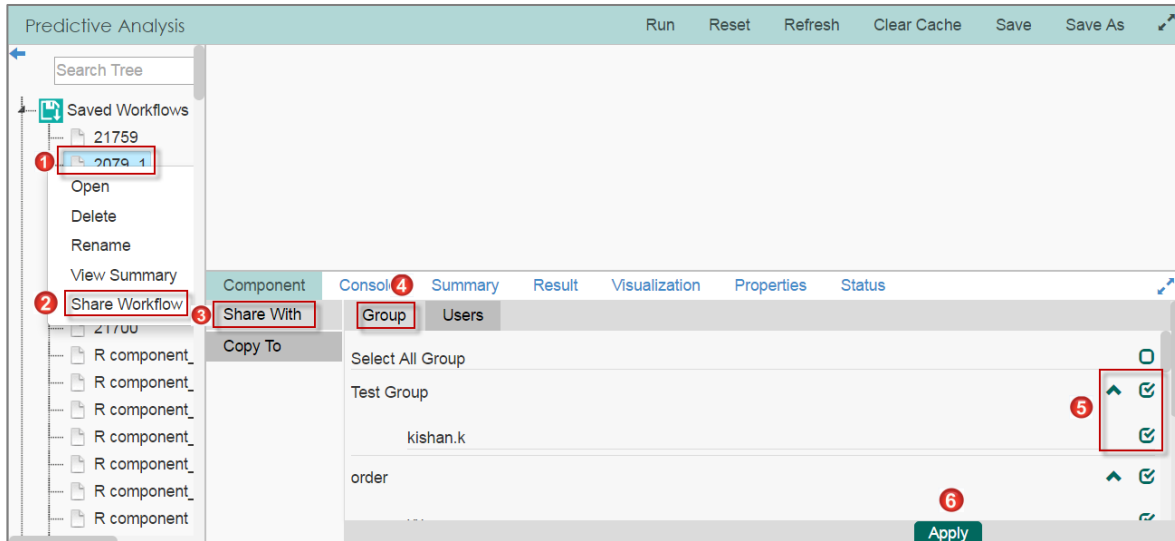


15.5. Sharing a Workflow

This feature gives users the ability to share saved workflows with other users and groups.

The following options are available to share a selected workflow:

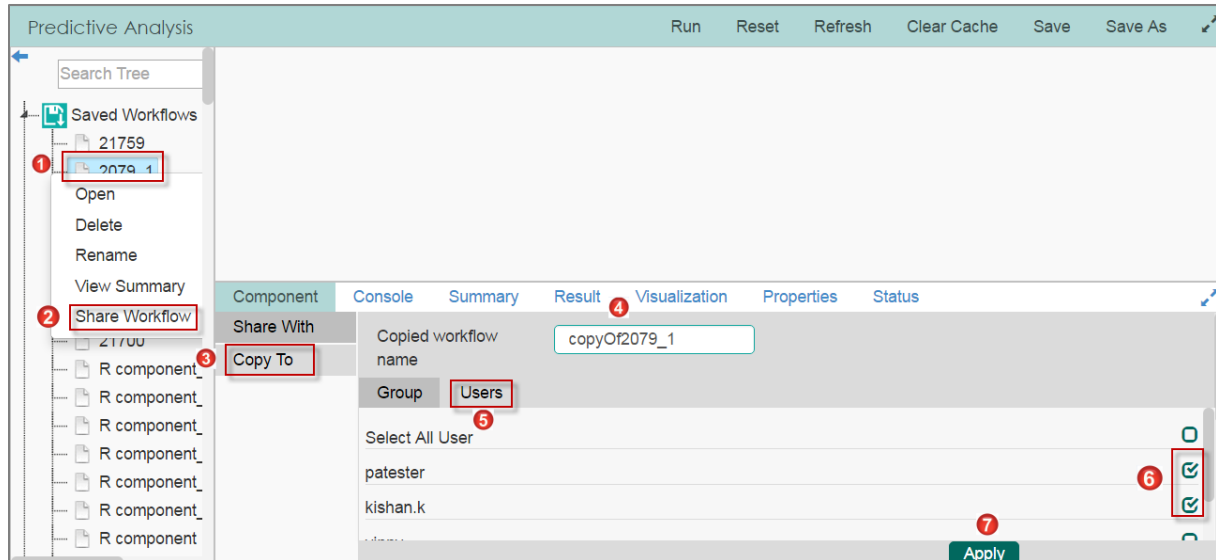
1. **Share With:** This option allows the user to share a file with the selected users or user groups. Any changes made to file will be transferred to all the users with whom the file has been shared.
 - i) Right click on a work flow from the list of **'Saved Workflows'**.
 - ii) Select **'Share Workflow'** from the context menu.
 - iii) The **'Share With'** option will be displayed (by default).
 - iv) Select either **'Group'** or **'Users'**.
 - a. By selecting a group all group members inside the group will be listed. Users can be excluded by not selecting them from the group.
 - b. Users can be excluded by not selecting a user name from the list when **'User'** option has been selected.
 - v) Select a specific group or user from the list by check marking the box.
 - vi) Click **'Apply'**.



vii) The selected workflow will be shared with the chosen users/groups.

2. **Copy To:** This option creates a copy and shares the copy with the selected users and user groups. Any changes to the original file after sharing will not show up for the users that received the shared file via the **'Copy To'** method.

- i) Right click on a work flow from the list of **'Saved Workflows'**.
- ii) Select **'Share Workflow'** from the context menu.
- iii) Select **'Copy To'**.
- iv) The copied workflow name will be displayed.
- v) Select either **'Group'** or **'Users'**.
 - a. By selecting a group all group members inside the group will be listed. Users can be excluded by not selecting them from the group.
 - b. Users can be excluded by not selecting a user name from the list when **'User'** option has been selected.
- vi) Select a specific group or user from the list by check marking the box.
- vii) Click **'Apply'**.



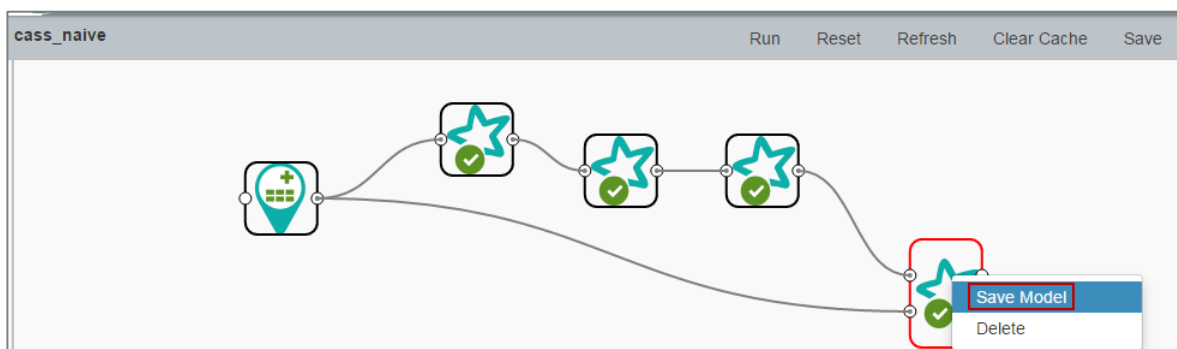
viii) The copied workflow will be shared with the chosen users/groups.

16. Saved Models

A model is a reusable component created by training an algorithm using historical data and saving the instance. The 'Saved Models' tree-node contains a list of all the saved predictive models.

16.1. Saving a Model

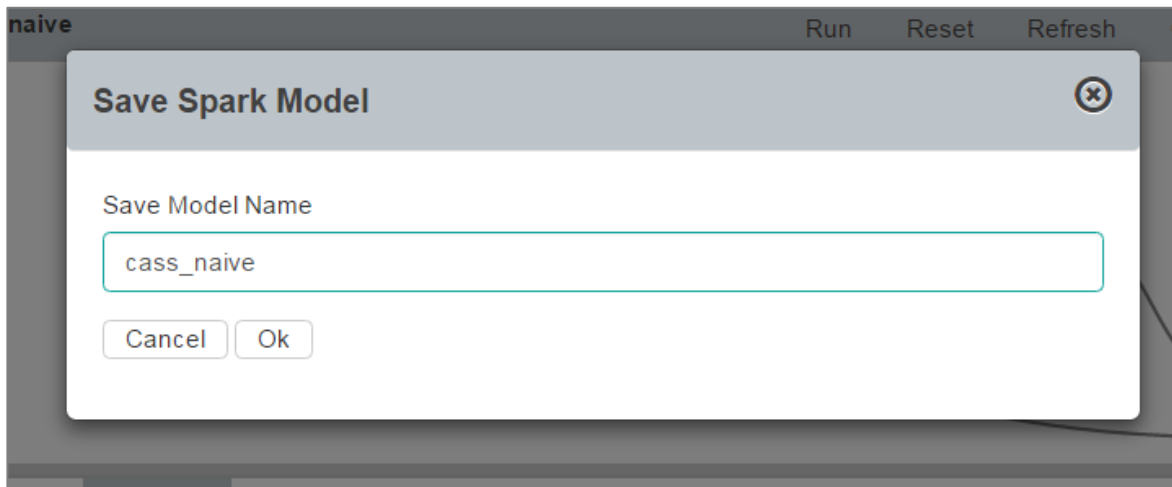
- i) Open a spark workflow.
- ii) Connect 'Apply Model' component with the workflow (as shown below).
- iii) Right click on the 'Apply Model' component.
- iv) A context menu will open.
- v) Select 'Save Model'.



vi) A pop-up window will appear.



- vii) Enter a name for the model that you wish to save.
- viii) Click 'OK'.



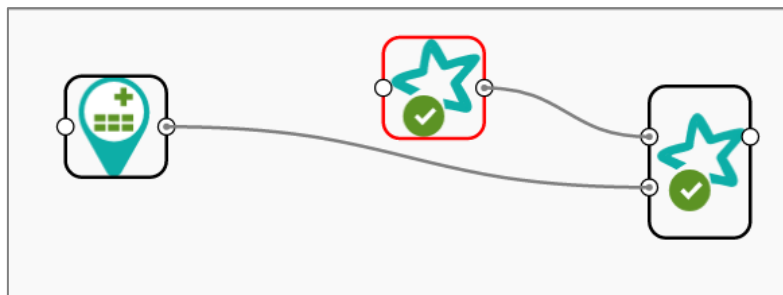
- ix) The created Predictive Model will be saved under the 'Saved Models' list.

Note: At present, the saved models support only Spark Naive Bayes algorithm.

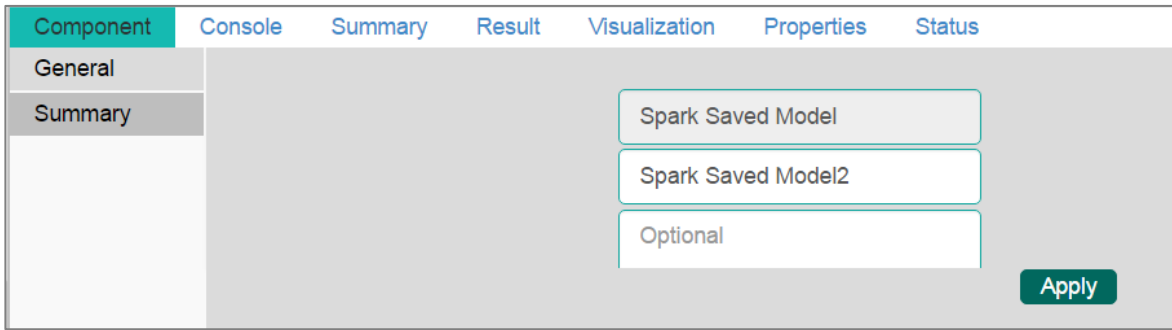
16.2. Reading a Model

Users can drag a saved model to workspace and reuse the model for a test data. A saved model can be connected to only Apply Model and new test data source.

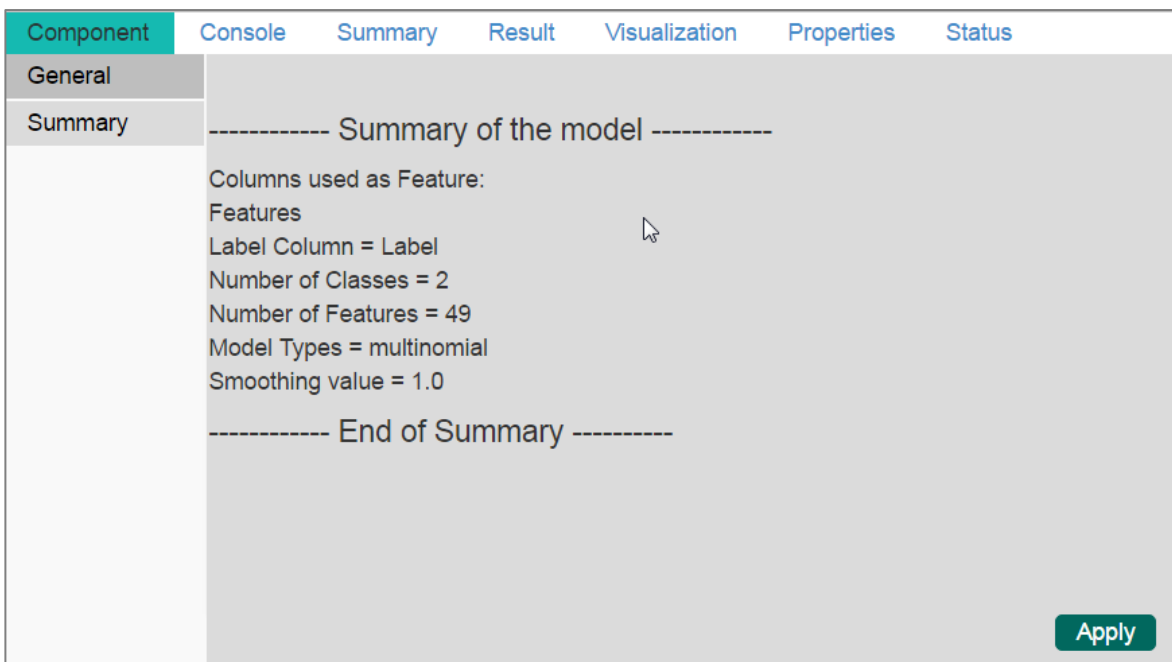
- i) Select and drag a saved model onto the workspace.
- ii) Connect the saved model with a configured data source and an apply model component (As shown in the following image).



- iii) Click on the dragged Saved Model component.
- iv) Users will be redirected to the component tab
- v) Configure the following fields in 'General':



vi) Click the **'Summary'** tab.



vii) Click **'Run'**.

viii) Users will be redirected to the **'Console'** tab.

ix) After the process gets completed under the Console tab, click the **'Result'** tab to see result view of data.



Component	Console	Summary	Result	Visualization	Properties	Status
Show	10	entries	Search: <input type="text"/>			
rownumber	petallength	petalwidth	sepalength	sepalwidth		
3	4.9	1.5	6.9	3.1		
23	3.7	1	5.5	2.4		
55	4.9	2	5.6	2.8		
24	1.5	0.2	5.3	3.7		
32	5.1	1.5	6.3	2.8		
12	5.6	2.1	6.4	2.8		
90	6.9	2.3	7.7	2.6		
77	5.6	1.4	6.1	2.6		
53	1.5	0.2	5	3.4		
91	4.6	1.3	6.6	2.9		

x) Click the **'Properties'** tab to display the model properties.

Component	Console	Summary	Result	Visualization	Properties	Status
Created By		Ranjit Krishnan				
Created At		2016-09-22 19:25:45 +0530				
Last Modified By		Ranjit Krishnan				
Last Modified At		2016-09-26 14:50:39 +0530				
Version		2.2.0				

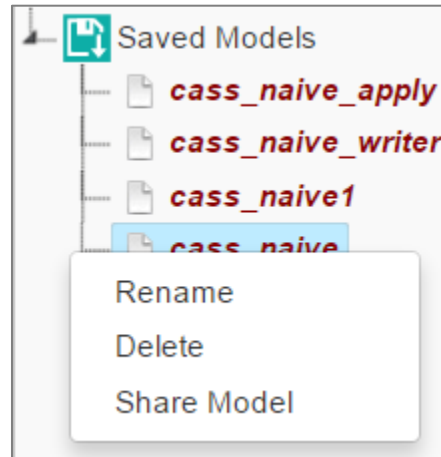
Note:

- a. To run the workflow with a **'Saved Model'** component it is mandatory that column headers and data type of the test data source should match with the selected saved model. Users will encounter error if validation fails while running the workflow.
- b. Users can connect a data writer to the **'Apply Model'** component in a workflow that contains a saved model.
- c. Currently only Spark trained Workflows can be saved under the **'Saved Models'** tree-node.

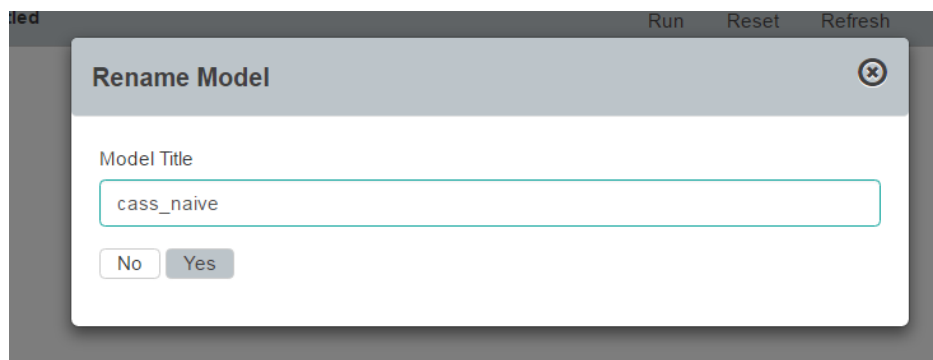
16.3. Renaming a Model

- i) Select a model from the **'Saved Models'** list.

- ii) Right click on the selected model.
- iii) A context menu will open.
- iv) Select '**Rename**'.



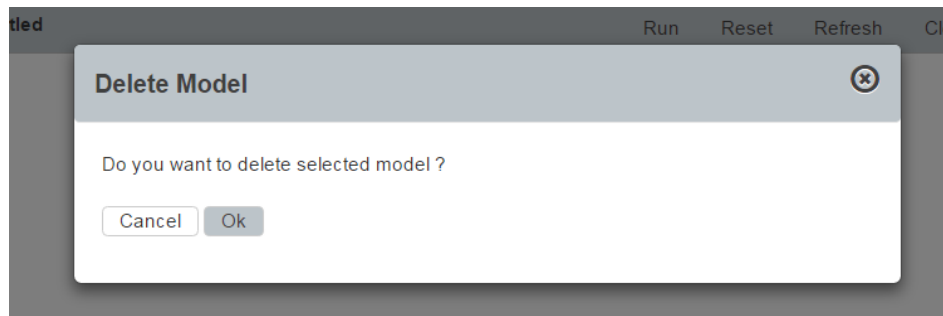
- v) A pop-up window will appear to rename the model.
- vi) Enter a new '**Model Title**' or modify the existing model title in the given field (if desired).
- vii) Click '**Yes**'.



- viii) The selected Predictive Model will be renamed.

16.4. Deleting a Model

- i) Select a model from the '**Saved Models**' list.
- ii) Right click on the selected model.
- iii) A context menu will open.
- iv) Select '**Delete**'.
- v) A pop-up window will appear to confirm deletion.
- vi) Click '**OK**'.

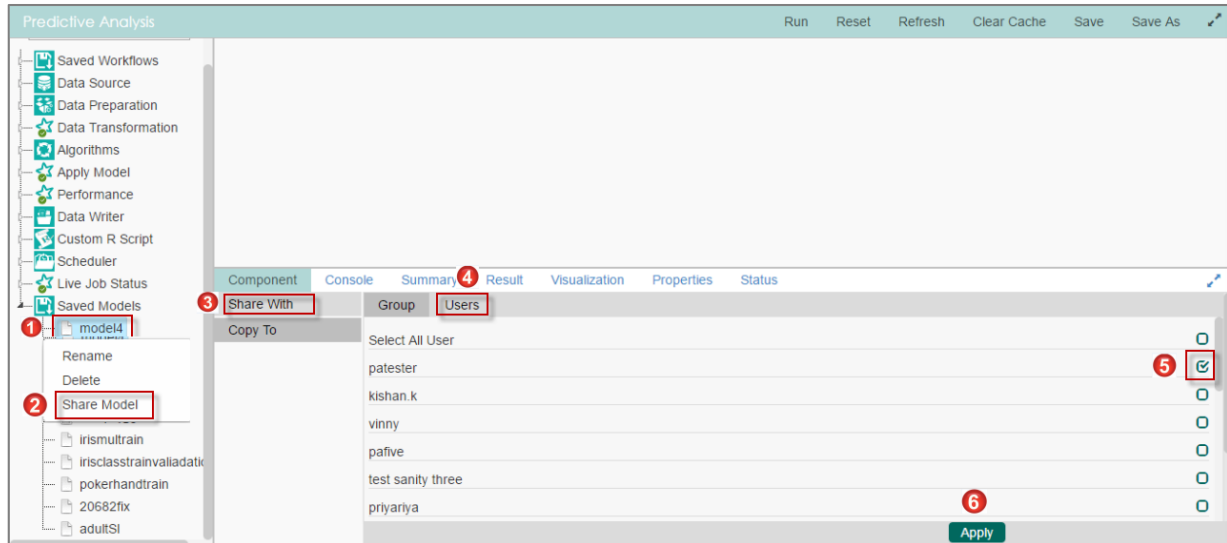


- vii) Selected predictive model will be deleted and removed from the list of **'Saved Models'**.

16.5. [Sharing a Model](#)

Users can share a saved model with other users or user groups. There are two options to share a selected model:

1. **Share With:** This option allows the user to share a file with the selected users or user groups. Any changes made to file will be transferred to all the users with whom the file has been shared.
 - i) Right click on a model from the list of **'Saved Models'**.
 - ii) Select **'Share Model'** from the context menu.
 - iii) The **'Share With'** option will be displayed (by default).
 - iv) Select either **'Group'** or **'Users'**.
 - a. By selecting a group all group members inside the group will be listed. Users can be excluded by not selecting them from the group.
 - b. Users can be excluded by not selecting a user name from the list when **'User'** option has been selected.
 - v) Select a specific group or user from the list by check marking the box.
 - vi) Click **'Apply'**.



2. **Copy To:** This option creates a copy and shares the copy with the selected users and user groups. Any changes to the original file after sharing will not show up for the users that received the shared file via the **'Copy To'** method.

ii) Right click on a work flow from the list of **'Saved Models'**.

iii) Select **'Share Model'** from the context menu.

iv) Select **'Copy To'** option.

v) The copied model name will be displayed.

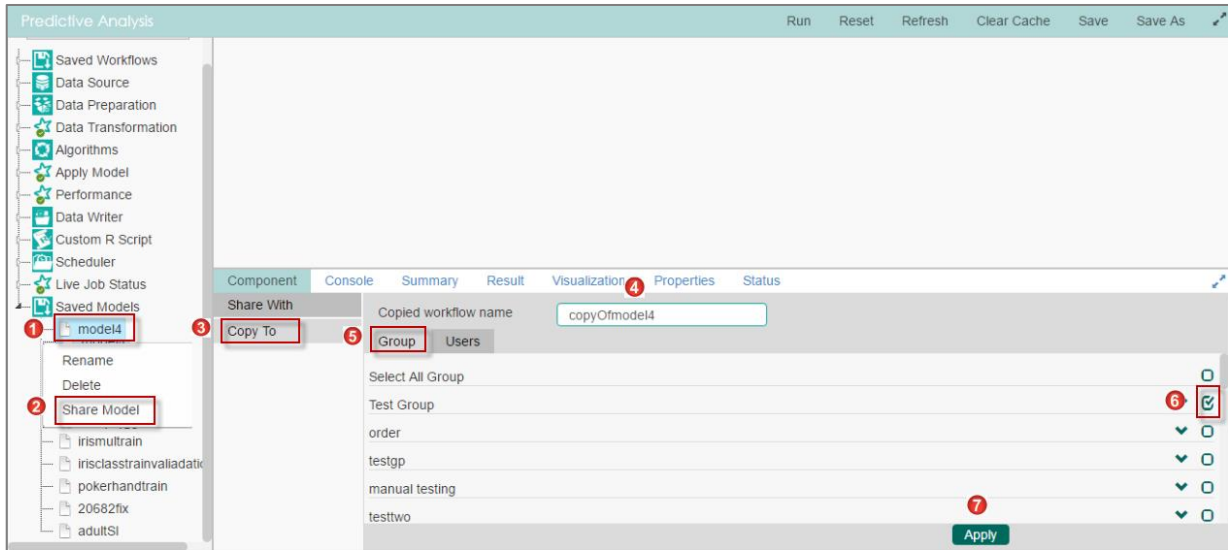
vi) Select either **'Group'** or **'Users'** option by a click.

a. By selecting a group all group members inside the group will be listed. Users can be excluded by not selecting them from the group.

b. Users can be excluded by not selecting a user name from the list when **'User'** option has been selected.

vii) Select a specific group or user from the list by check marking the box.

viii) Click **'Apply'**.



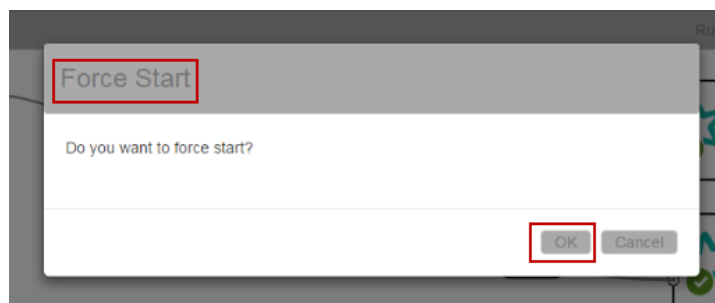
- A copy of the model will be shared with the selected user or group

17. Specific Options for a Spark Workflow

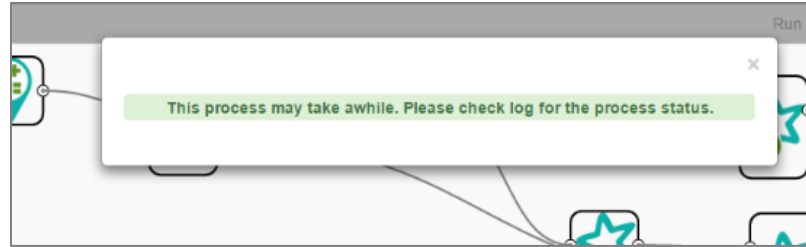
17.1. Force Start

This option can be used for Spark jobs if the Spark request queue becomes full.

- If the number of requests in spark is greater than 20, a dialogue box will be displayed prompting for a **'Force Start'**.
- Click **'OK'** to confirm.



- A message will pop-up asking the user to check the in the live job status for the log since the process may take some more time.



Note: Users can configure the number of spark requests while deploying spark application.

17.2. Result of Each Component

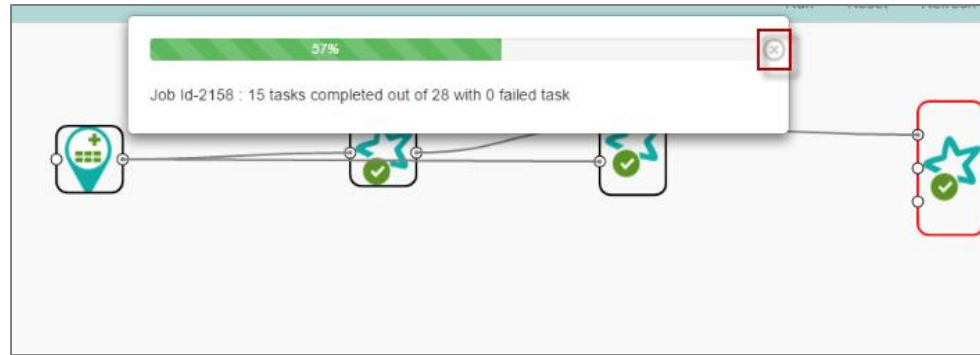
Users can view the result of each component in the spark workflow.

- Select a component from the spark workflow after the execution is completed.
- Click the **'Result'** tab.
- The result data of the selected component will be displayed.

ClusterNumber	PetalLength	PetalWidth	SepalLength	SepalWidth
1	5.8	2.2	6.5	3
3	4.6	1.3	6.6	2.9
3	4.7	1.2	6.1	2.8
3	5.1	1.9	5.8	2.7
5	1.5	0.2	5	3.4

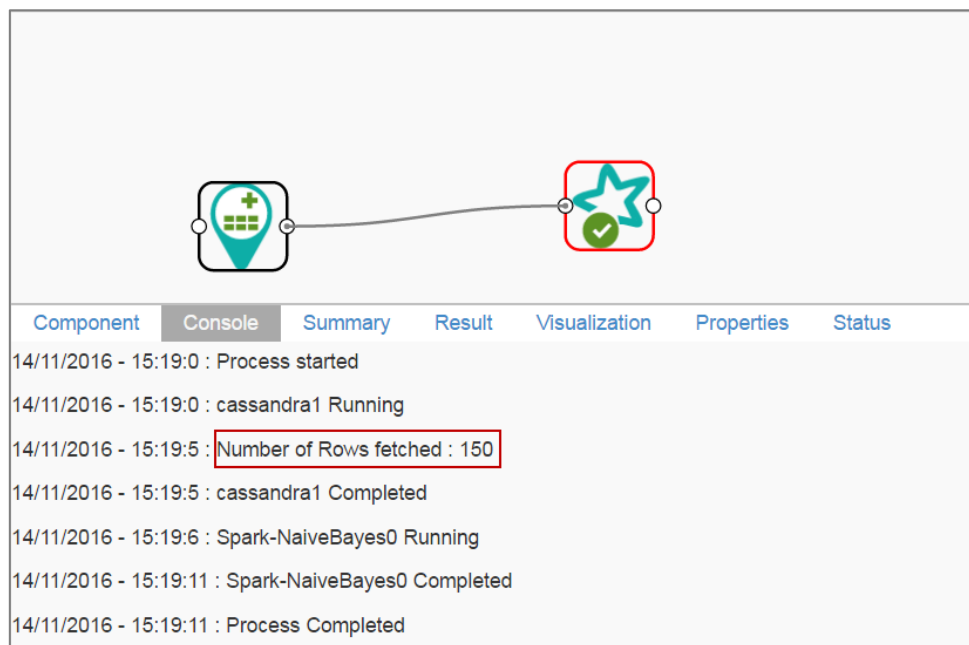
17.3. Stop Button on the Progress Bar

Users can stop an ongoing Spark workflow execution by clicking the **'Stop'** button on the progress bar.




17.4. Log Information Displayed under the Console Tab

- A log with the auto numbered alias name of component is displayed under the 'Console' tab when running a Spark workflow.
- The 'Number of Rows Fetched' during an execution will be provided in the log.



Component	Console	Summary	Result	Visualization	Properties	Status
	14/11/2016 - 15:19:0 : Process started					
	14/11/2016 - 15:19:0 : cassandra1 Running					
	14/11/2016 - 15:19:5 : Number of Rows fetched : 150					
	14/11/2016 - 15:19:5 : cassandra1 Completed					
	14/11/2016 - 15:19:6 : Spark-NaiveBayes0 Running					
	14/11/2016 - 15:19:11 : Spark-NaiveBayes0 Completed					
	14/11/2016 - 15:19:11 : Process Completed					

18. Logging Out

- Click on the  option from the Header Panel of the BizViz Platform.
- You will be successfully logged out from the Predictive Analysis and BizViz Platform.



Note: Clicking on '**Logout**' option will redirect the user to the Login screen of the BizViz Platform.









