DATA ANALYTICS

IS 665

“DATA MINING PROJECT”

On

“Classification of Iris”

- ***Rohan Uday Khambekar***

Description:

This is perhaps the best known database to be found in the pattern recognition literature. Fisher's paper is a classic in the field and is referenced frequently to this day. (See Duda & Hart, for example.) The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other 2; the latter are NOT linearly separable from each other.

Question:

We are trying to predict the class of the Iris plant.

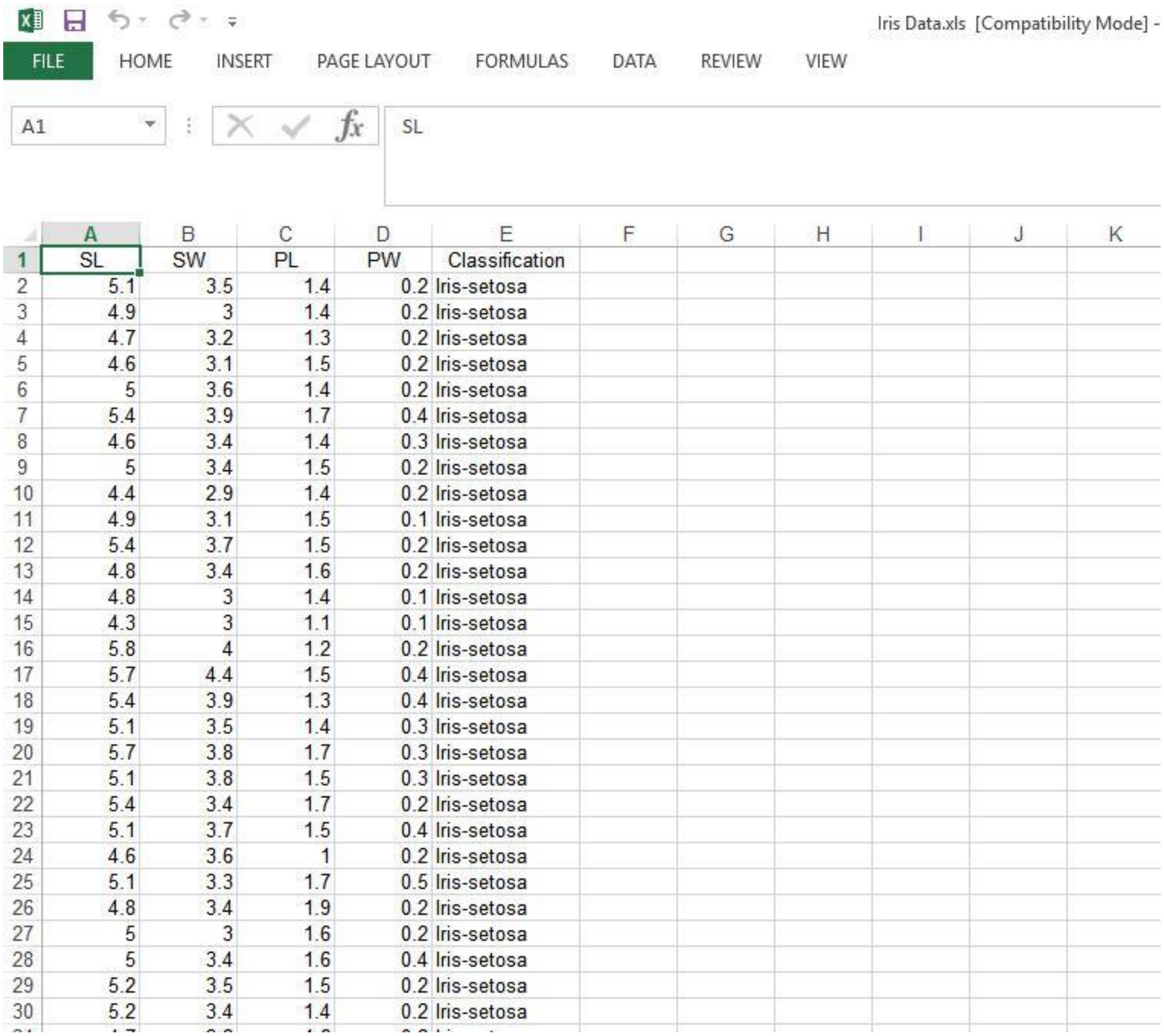
Data Source:

http://mercury.webster.edu/aleshunas/Data%20Sets/Supplemental%20Excel%20Data%20Sets.htm

Data Dictionary:

Attribute	Data Type	Range of values
Sepal Length in cm (SL)	Numeric	4.3 – 7.9
Sepal Width in cm (SW)	Numeric	2.0 – 4.4
Petal Length in cm (PL)	Numeric	1.0 – 6.9
Petal Width in cm (PW)	Numeric	0.1 – 2.5
Classification	Varchar	Iris Setosa, Iris Virginica, Iris Versicolor

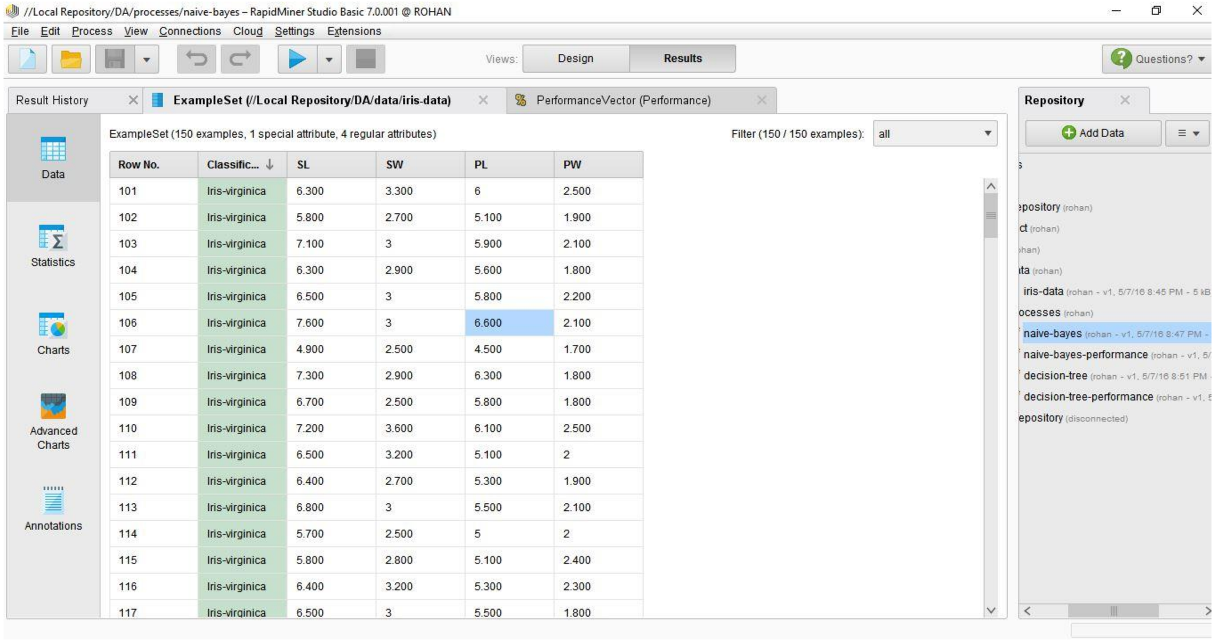
Sample Data:



	A	B	C	D	E
1	SL	SW	PL	PW	Classification
2	5.1	3.5	1.4	0.2	Iris-setosa
3	4.9	3	1.4	0.2	Iris-setosa
4	4.7	3.2	1.3	0.2	Iris-setosa
5	4.6	3.1	1.5	0.2	Iris-setosa
6	5	3.6	1.4	0.2	Iris-setosa
7	5.4	3.9	1.7	0.4	Iris-setosa
8	4.6	3.4	1.4	0.3	Iris-setosa
9	5	3.4	1.5	0.2	Iris-setosa
10	4.4	2.9	1.4	0.2	Iris-setosa
11	4.9	3.1	1.5	0.1	Iris-setosa
12	5.4	3.7	1.5	0.2	Iris-setosa
13	4.8	3.4	1.6	0.2	Iris-setosa
14	4.8	3	1.4	0.1	Iris-setosa
15	4.3	3	1.1	0.1	Iris-setosa
16	5.8	4	1.2	0.2	Iris-setosa
17	5.7	4.4	1.5	0.4	Iris-setosa
18	5.4	3.9	1.3	0.4	Iris-setosa
19	5.1	3.5	1.4	0.3	Iris-setosa
20	5.7	3.8	1.7	0.3	Iris-setosa
21	5.1	3.8	1.5	0.3	Iris-setosa
22	5.4	3.4	1.7	0.2	Iris-setosa
23	5.1	3.7	1.5	0.4	Iris-setosa
24	4.6	3.6	1	0.2	Iris-setosa
25	5.1	3.3	1.7	0.5	Iris-setosa
26	4.8	3.4	1.9	0.2	Iris-setosa
27	5	3	1.6	0.2	Iris-setosa
28	5	3.4	1.6	0.4	Iris-setosa
29	5.2	3.5	1.5	0.2	Iris-setosa
30	5.2	3.4	1.4	0.2	Iris-setosa

Example Set: (on adding the data)

- ### Data Screen



//Local Repository/DA/processes/naive-bayes – RapidMiner Studio Basic 7.0.001 @ ROHAN

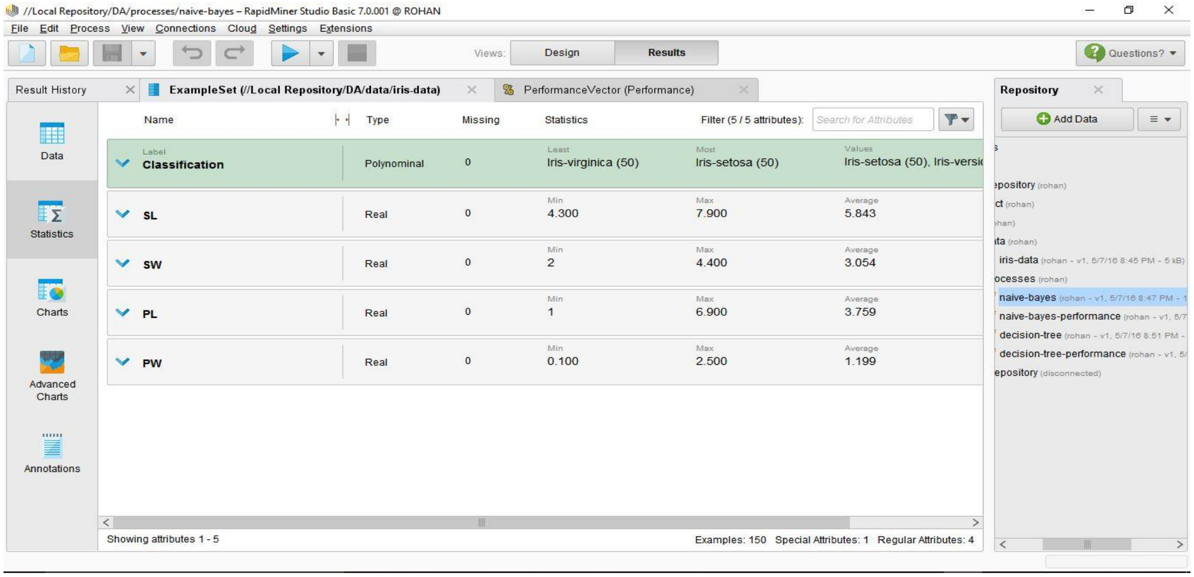
File Edit Process View Connections Cloud Settings Extensions
Views: Design Results
Questions?
Result History
ExampleSet (//Local Repository/DA/data/iris-data)
PerformanceVector (Performance)
Repository
Add Data
Data
Statistics
Charts
Advanced Charts
Annotations
epository (rohan)
ct (rohan)
han)
ta (rohan)
iris-data (rohan - v1, 5/7/16 8:45 PM - 5 kB
ocesses (rohan)
naive-bayes (rohan - v1, 5/7/16 8:47 PM -
naive-bayes-performance (rohan - v1, 5/
decision-tree (rohan - v1, 5/7/16 8:51 PM -
decision-tree-performance (rohan - v1, 5
epository (disconnected)

ExampleSet (150 examples, 1 special attribute, 4 regular attributes)

Filter (150 / 150 examples): all

Row No.	Classific...	SL	SW	PL	PW
101	Iris-virginica	6.300	3.300	6	2.500
102	Iris-virginica	5.800	2.700	5.100	1.900
103	Iris-virginica	7.100	3	5.900	2.100
104	Iris-virginica	6.300	2.900	5.600	1.800
105	Iris-virginica	6.500	3	5.800	2.200
106	Iris-virginica	7.600	3	6.600	2.100
107	Iris-virginica	4.900	2.500	4.500	1.700
108	Iris-virginica	7.300	2.900	6.300	1.800
109	Iris-virginica	6.700	2.500	5.800	1.800
110	Iris-virginica	7.200	3.600	6.100	2.500
111	Iris-virginica	6.500	3.200	5.100	2
112	Iris-virginica	6.400	2.700	5.300	1.900
113	Iris-virginica	6.800	3	5.500	2.100
114	Iris-virginica	5.700	2.500	5	2
115	Iris-virginica	5.800	2.800	5.100	2.400
116	Iris-virginica	6.400	3.200	5.300	2.300
117	Iris-virginica	6.500	3	5.500	1.800

- ### Statistics Screen



//Local Repository/DA/processes/naive-bayes – RapidMiner Studio Basic 7.0.001 @ ROHAN

File Edit Process View Connections Cloud Settings Extensions
Views: Design Results
Questions?
Result History
ExampleSet (//Local Repository/DA/data/iris-data)
PerformanceVector (Performance)
Repository
Add Data
Data
Statistics
Charts
Advanced Charts
Annotations
Search for Attributes
epository (rohan)
ct (rohan)
han)
ta (rohan)
iris-data (rohan - v1, 5/7/16 8:45 PM - 5 kB)
ocesses (rohan)
naive-bayes (rohan - v1, 5/7/16 8:47 PM - 1
naive-bayes-performance (rohan - v1, 5/7
decision-tree (rohan - v1, 5/7/16 8:51 PM -
decision-tree-performance (rohan - v1, 5/
epository (disconnected)

Filter (5 / 5 attributes):

Name	Type	Missing	Statistics		
Label Classification	Polynominal	0	Least Iris-virginica (50)	Most Iris-setosa (50)	Values Iris-setosa (50), Iris-versic
SL	Real	0	Min 4.300	Max 7.900	Average 5.843
SW	Real	0	Min 2	Max 4.400	Average 3.054
PL	Real	0	Min 1	Max 6.900	Average 3.759
PW	Real	0	Min 0.100	Max 2.500	Average 1.199

Showing attributes 1 - 5

Examples: 150 Special Attributes: 1 Regular Attributes: 4

1. Naïve Bayes:

- **Conditional Probability**

$P(A,B) = P(A|B)P(B) = P(B|A)P(A)$

Bayes rule

$P(B|A)P(A) = P(A|B)P(B)$

- **Naïve Bayes Classifier**

Applying Bayes rule $P(Yi)$

$P(X1\ldots Xn|Yi)$ $P(Yi|X1\ldots Xn)$

$= P(X1\ldots Xn)$

$P(Yi)$ $P(X1|Yi)P(X2|Yi)\ldots P(Xn|Yi) = P(X1\ldots Xn)$

Y ⍰ arg max $P(Yi)$ $P(X1|Yi)P(X2|Yi)\ldots P(Xn|Yi)$

$P(Y|\mathbf{X})$: posterior probability for Y $P(Y)$: prior probability

$P(\mathbf{X}|Y)$: class-conditional probability

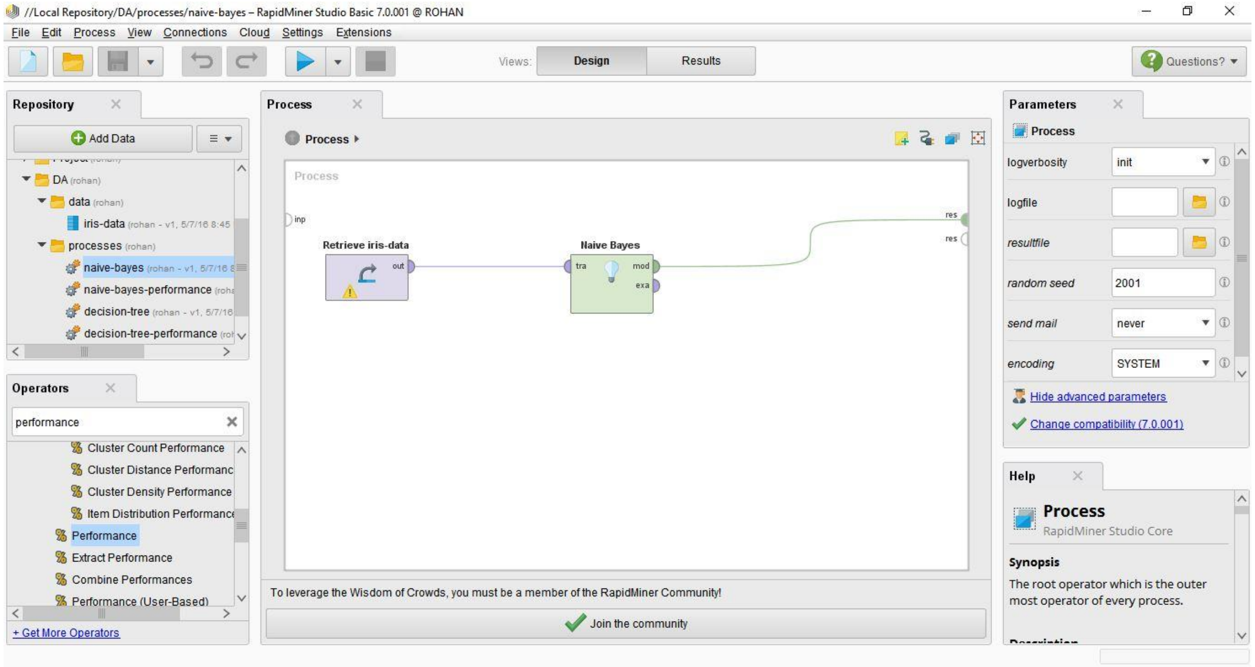
$P(\mathbf{X})$: evidence

Bayes theorem (Bayes rule) allows us to calculate the posterior probability $P(Y|\mathbf{X})$ using the prior probability $P(Y)$, the class-conditional probability $P(\mathbf{X}|Y)$ and the evidence $P(\mathbf{X})$

(Which is constant and ignored).

Source: wikipedia

Process Screen:



//Local Repository/DA/processes/naive-bayes – RapidMiner Studio Basic 7.0.001 @ ROHAN

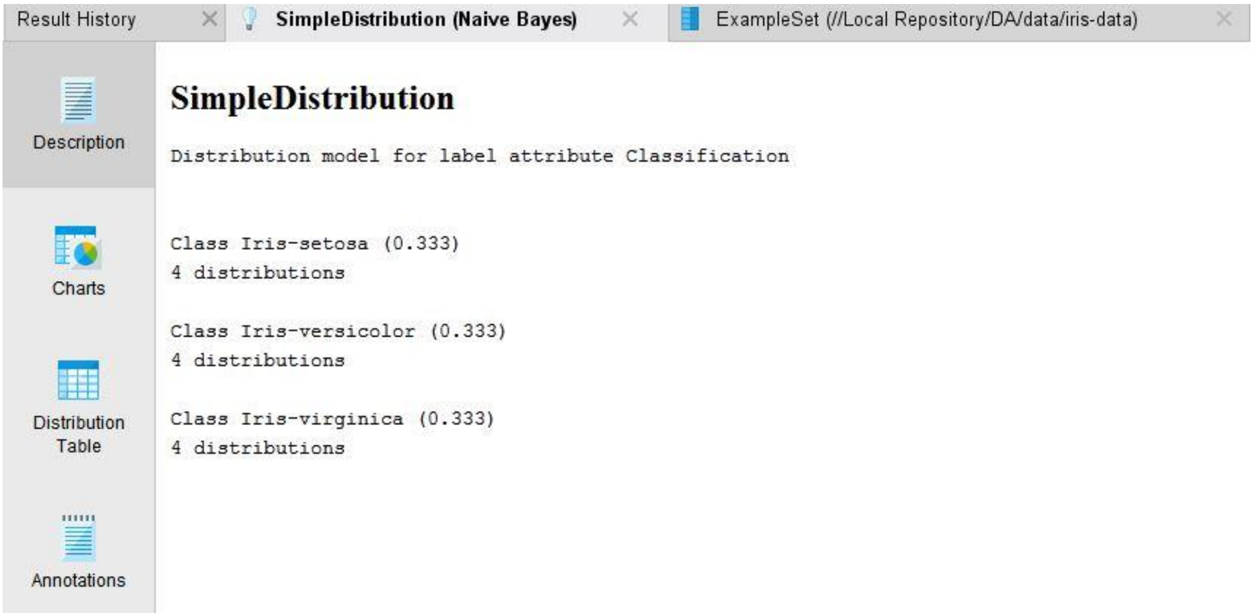
logverbosity	init
logfile	
resultfile	
random seed	2001
send mail	never
encoding	SYSTEM

To leverage the Wisdom of Crowds, you must be a member of the RapidMiner Community!

Process
RapidMiner Studio Core

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

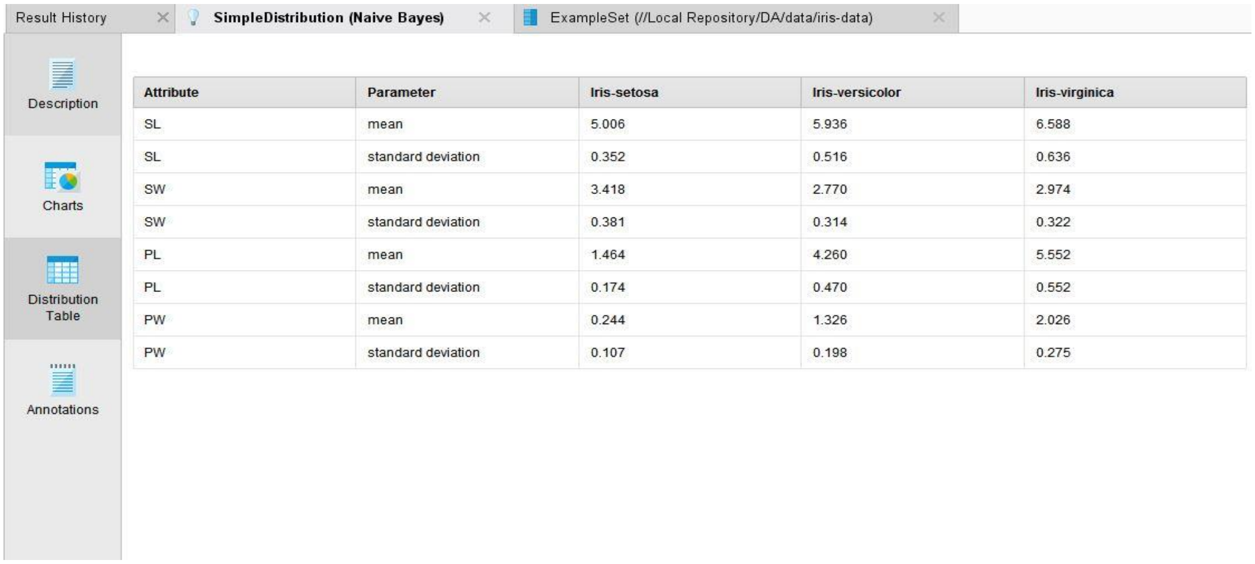
Description:



Result History
SimpleDistribution (Naive Bayes)
ExampleSet (//Local Repository/DA/data/iris-data)
Description
Charts
Distribution Table
Annotations

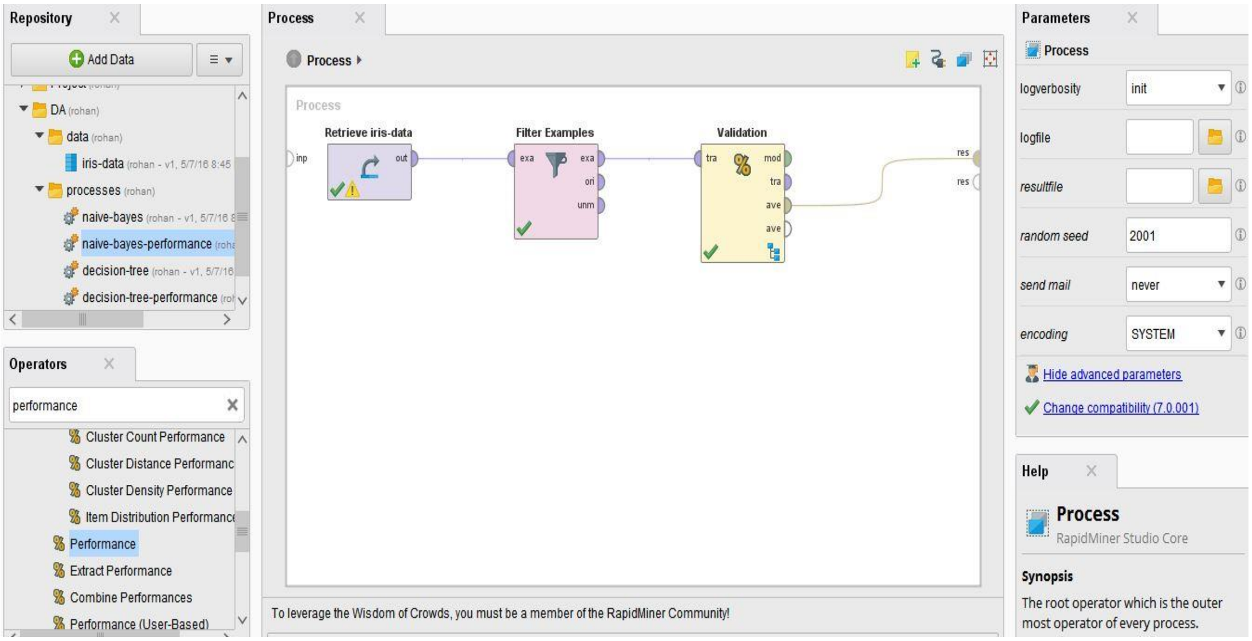
```
SimpleDistribution

Distribution model for label attribute Classification


Class Iris-setosa (0.333)
4 distributions

Class Iris-versicolor (0.333)
4 distributions

Class Iris-virginica (0.333)
4 distributions
```

Distribution Table:


Result History
SimpleDistribution (Naive Bayes)
ExampleSet (//Local Repository/DA/data/iris-data)
Description
Charts
Distribution Table
Annotations

Attribute	Parameter	Iris-setosa	Iris-versicolor	Iris-virginica
SL	mean	5.006	5.936	6.588
SL	standard deviation	0.352	0.516	0.636
SW	mean	3.418	2.770	2.974
SW	standard deviation	0.381	0.314	0.322
PL	mean	1.464	4.260	5.552
PL	standard deviation	0.174	0.470	0.552
PW	mean	0.244	1.326	2.026
PW	standard deviation	0.107	0.198	0.275

Validation:



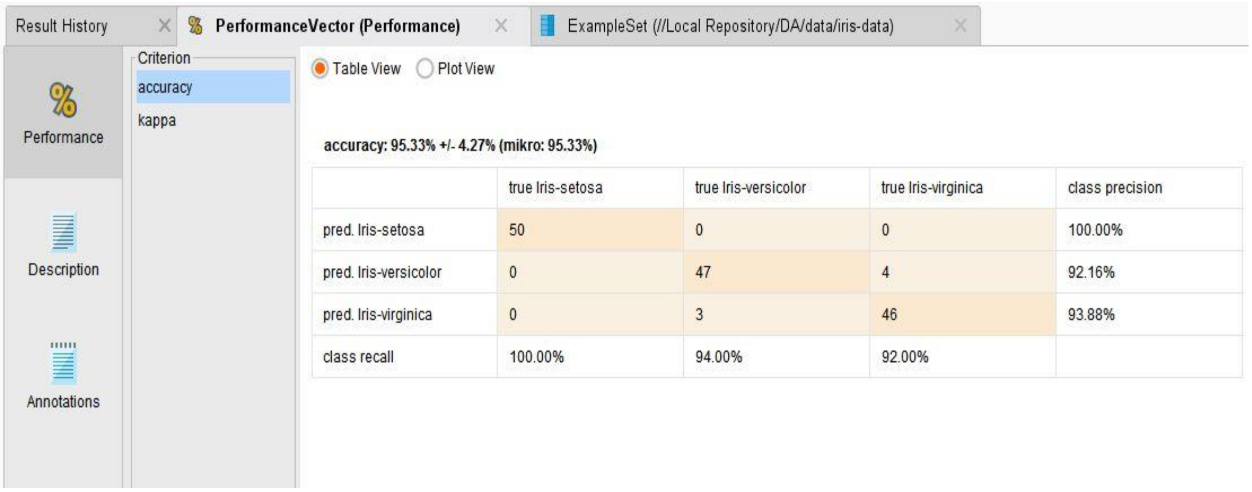
logverbosity	init
logfile	
resultfile	
random seed	2001
send mail	never
encoding	SYSTEM

To leverage the Wisdom of Crowds, you must be a member of the RapidMiner Community!

Synopsis

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

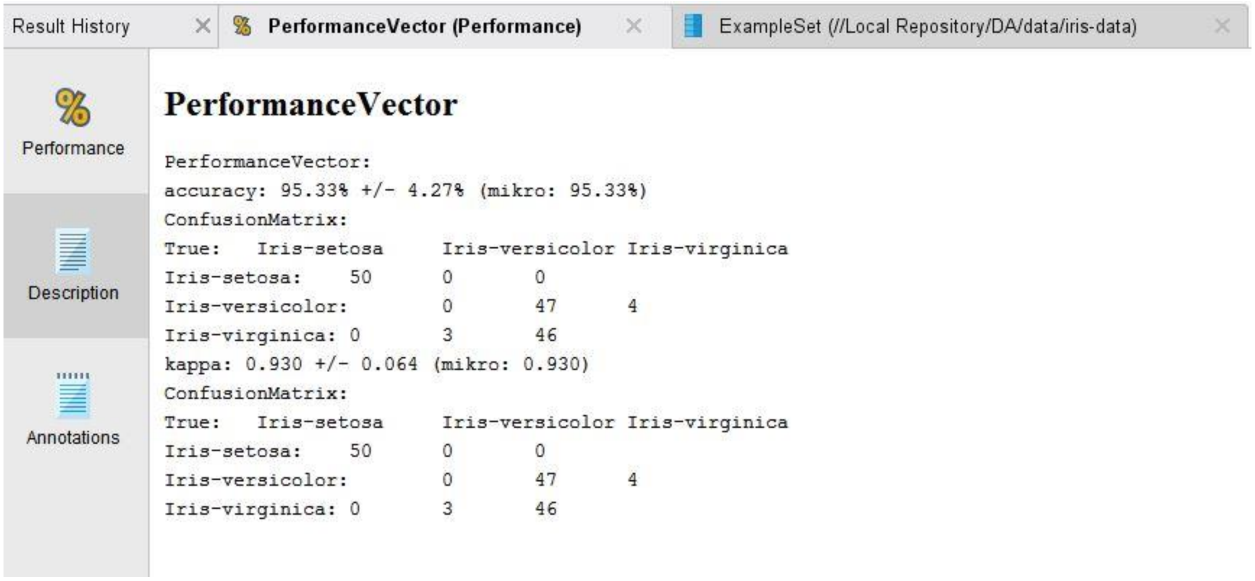
Performance Result:



accuracy: 95.33% +/- 4.27% (mikro: 95.33%)

	true Iris-setosa	true Iris-versicolor	true Iris-virginica	class precision
pred. Iris-setosa	50	0	0	100.00%
pred. Iris-versicolor	0	47	4	92.16%
pred. Iris-virginica	0	3	46	93.88%
class recall	100.00%	94.00%	92.00%	

Performance Vector:


Result History
PerformanceVector (Performance)
ExampleSet (//Local Repository/DA/data/iris-data)
Performance
Description
Annotations

PerformanceVector

```
PerformanceVector:
accuracy: 95.33% +/- 4.27% (mikro: 95.33%)
ConfusionMatrix:
True:   Iris-setosa     Iris-versicolor Iris-virginica
Iris-setosa:    50      0       0
Iris-versicolor:        0       47      4
Iris-virginica: 0       3       46
kappa: 0.930 +/- 0.064 (mikro: 0.930)
ConfusionMatrix:
True:   Iris-setosa     Iris-versicolor Iris-virginica
Iris-setosa:    50      0       0
Iris-versicolor:        0       47      4
Iris-virginica: 0       3       46
```

2. Decision Tree

- Decision tree learning is a method commonly used in data mining.[1] The goal is to create a model that predicts the value of a target variable based on several input variables. An example is shown below. Each interior node corresponds to one of the input variables; there are edges to children for each of the possible values of that input variable. Each leaf represents a value of the target variable given the values of the input variables represented by the path from the root to the leaf.

- A decision tree is a simple representation for classifying examples. For this section, assume that all of the features have finite discrete domains, and there is a single target feature called the classification. Each element of the domain of the classification is called a class. A decision tree or a classification tree is a tree in which each internal (non-leaf) node is labeled with an input feature. The arcs coming from a node labeled with a feature are labeled with each of the possible values of the feature. Each leaf of the tree is labeled with a class or a probability distribution over the classes.

- A tree can be "learned" by splitting the source set into subsets based on an attribute value test. This process is repeated on each derived subset in a recursive manner called recursive partitioning. The recursion is completed when the subset at a node has all the same value of the target variable, or when splitting no longer adds value to the predictions. This process of *top-down induction of decision trees* (TDIDT) is an example of a greedy algorithm, and it is by far the most common strategy for learning decision trees from data.

- In data mining, decision trees can be described also as the combination of mathematical and computational techniques to aid the description, categorization and generalization of a given set of data.

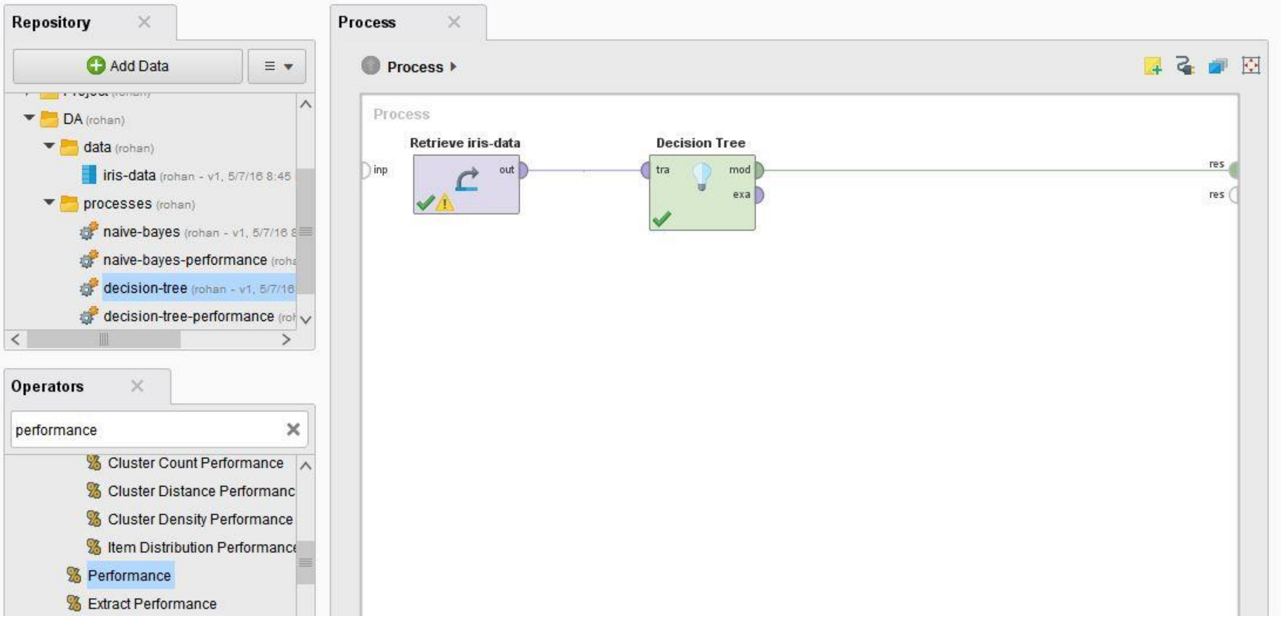
- Data comes in records of the form:

$$(\mathbf{x}, Y) = (x_1, x_2, x_3, ..., x_k, Y)$$

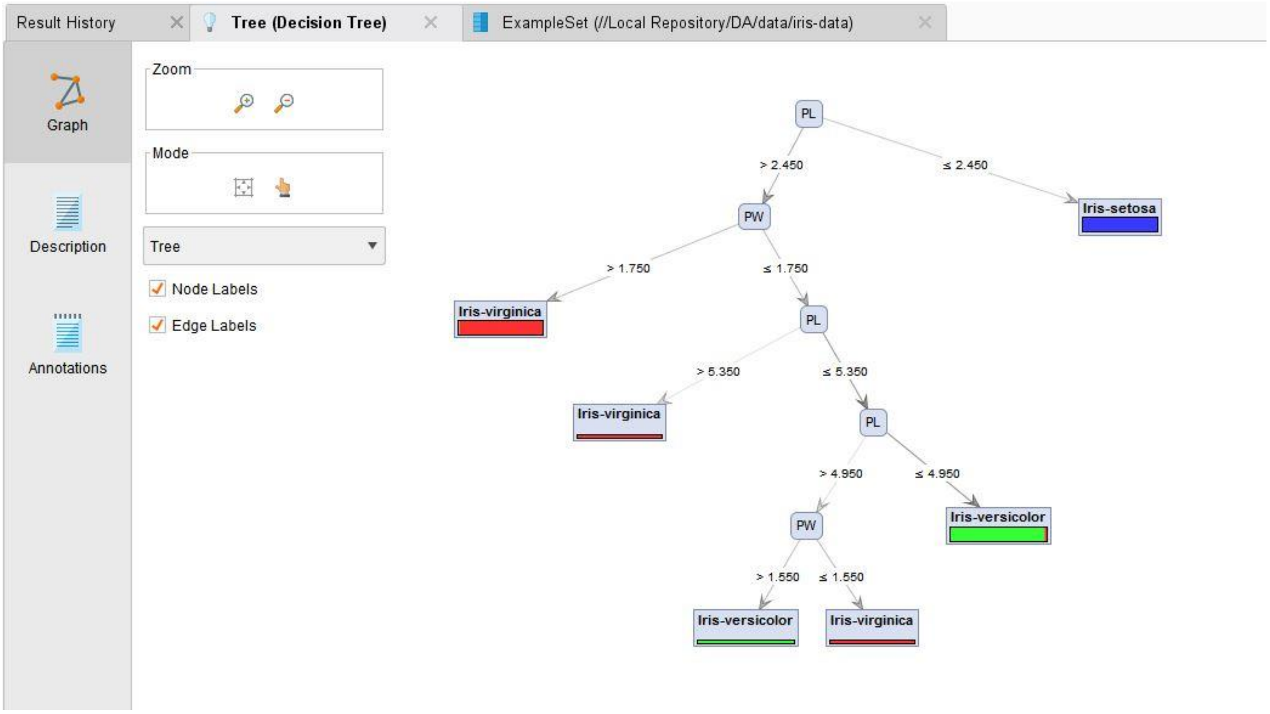
- The dependent variable, Y, is the target variable that we are trying to understand, classify or generalize. The vector **x** is composed of the input variables, x_1, x_2, x_3 etc., that are used for that task.

Source: Wikipedia

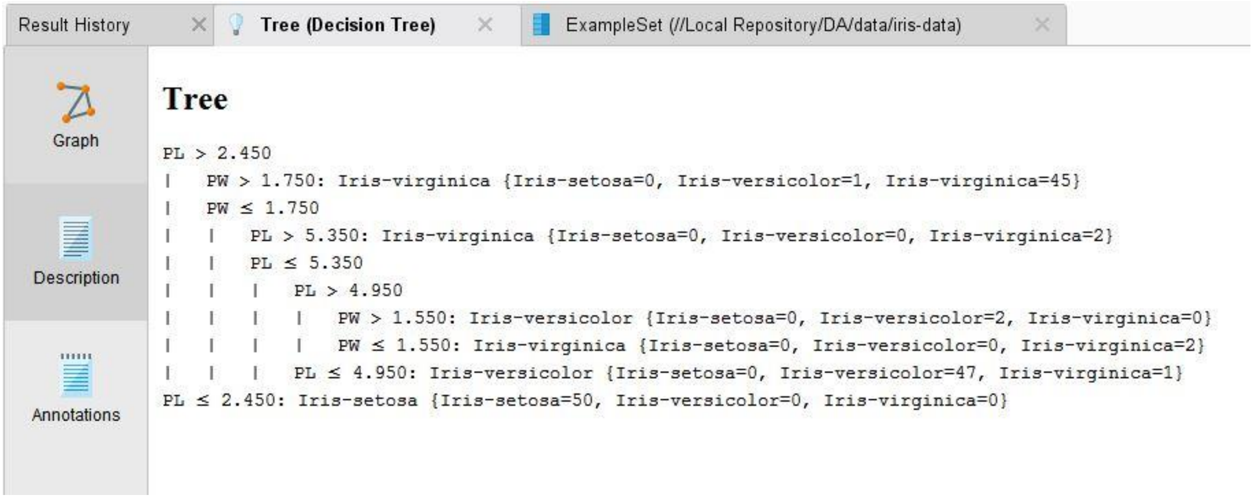
Process Screen:


Repository
Add Data
DA (rohan)
data (rohan)
iris-data (rohan - v1, 5/7/16 8:45
processes (rohan)
naive-bayes (rohan - v1, 5/7/16
naive-bayes-performance
decision-tree (rohan - v1, 5/7/16
decision-tree-performance
Operators
performance
Cluster Count Performance
Cluster Distance Performanc
Cluster Density Performance
Item Distribution Performance
Performance
Extract Performance
Process
Process
Process
Retrieve iris-data
inp
out
Decision Tree
tra
mod
exa
res
res

Decision Graph:



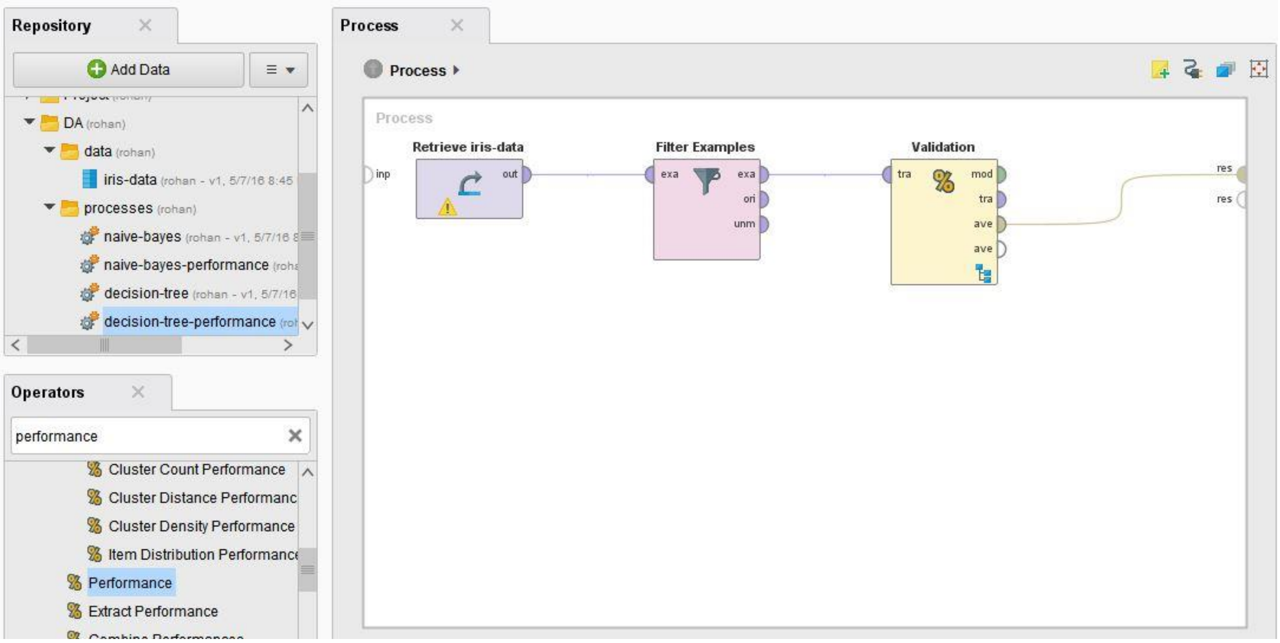
Tree:



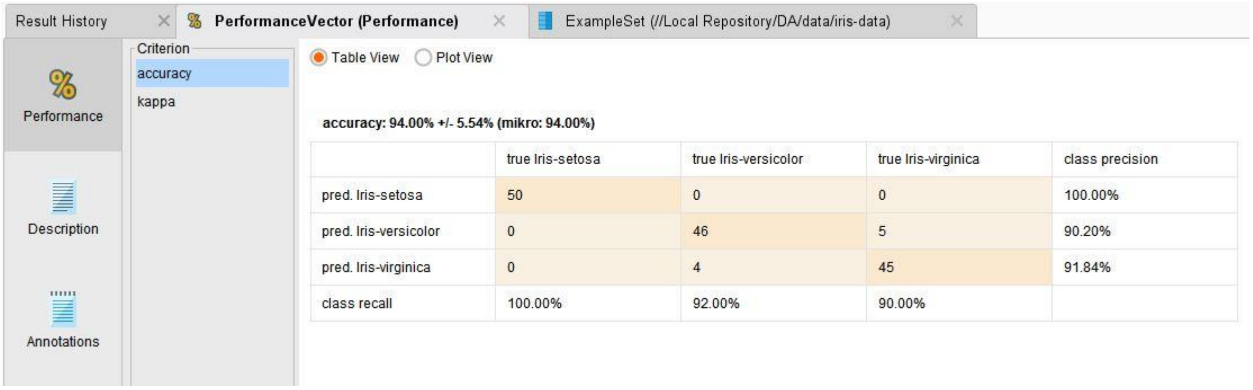
Tree

```
PL > 2.450
|   PW > 1.750: Iris-virginica {Iris-setosa=0, Iris-versicolor=1, Iris-virginica=45}
|   PW ≤ 1.750
|   |   PL > 5.350: Iris-virginica {Iris-setosa=0, Iris-versicolor=0, Iris-virginica=2}
|   |   PL ≤ 5.350
|   |   |   PL > 4.950
|   |   |   |   PW > 1.550: Iris-versicolor {Iris-setosa=0, Iris-versicolor=2, Iris-virginica=0}
|   |   |   |   PW ≤ 1.550: Iris-virginica {Iris-setosa=0, Iris-versicolor=0, Iris-virginica=2}
|   |   |   PL ≤ 4.950: Iris-versicolor {Iris-setosa=0, Iris-versicolor=47, Iris-virginica=1}
PL ≤ 2.450: Iris-setosa {Iris-setosa=50, Iris-versicolor=0, Iris-virginica=0}
```

Validation:



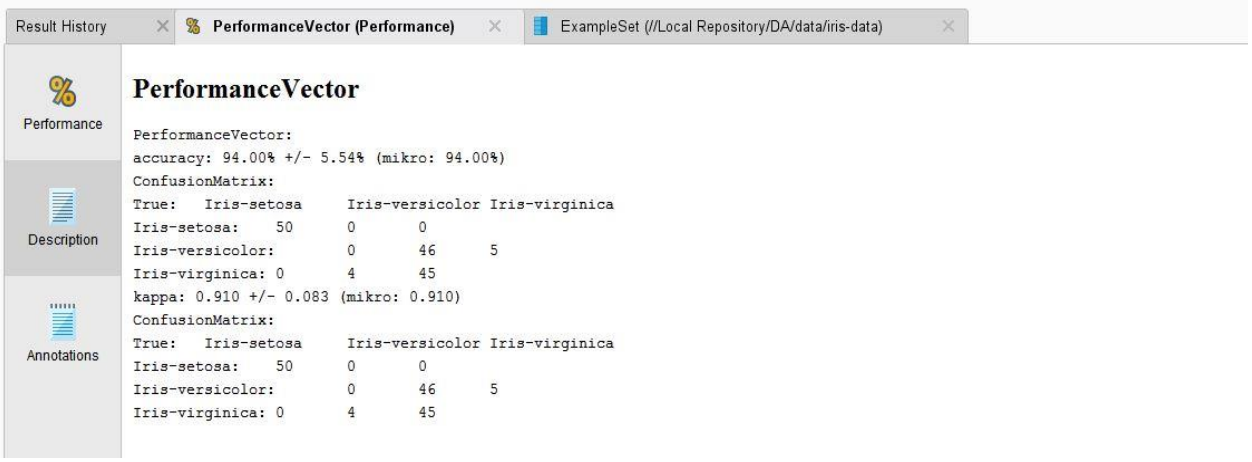
Performance:



accuracy: 94.00% +/- 5.54% (mikro: 94.00%)

	true Iris-setosa	true Iris-versicolor	true Iris-virginica	class precision
pred. Iris-setosa	50	0	0	100.00%
pred. Iris-versicolor	0	46	5	90.20%
pred. Iris-virginica	0	4	45	91.84%
class recall	100.00%	92.00%	90.00%	

Performance Vector:


Result History
PerformanceVector (Performance)
ExampleSet (//Local Repository/DA/data/iris-data)
Performance
Description
Annotations

PerformanceVector

```
PerformanceVector:
accuracy: 94.00% +/- 5.54% (mikro: 94.00%)
ConfusionMatrix:
True:   Iris-setosa     Iris-versicolor Iris-virginica
Iris-setosa:    50      0       0
Iris-versicolor:        0       46      5
Iris-virginica: 0       4       45
kappa: 0.910 +/- 0.083 (mikro: 0.910)
ConfusionMatrix:
True:   Iris-setosa     Iris-versicolor Iris-virginica
Iris-setosa:    50      0       0
Iris-versicolor:        0       46      5
Iris-virginica: 0       4       45
```

Performance Comparison:

Naïve Bayes vs Decision Trees

Attribute	Classification	Naïve Bayes	Decision Trees
Precision	Iris Setosa	100%	100%
	Iris Versicolor	92.16%	90.20%
	Iris Virginica	93.88%	91.84%
Recall	Iris Setosa	100%	100%
	Iris Versicolor	94%	92%
	Iris Virginica	92%	90%

We can clearly see from the above table that Naïve Bayes slightly edges out Decision Trees in terms of performance if we are to compare their Precision and Recall values.

Thus for the Iris Data Set, Naïve Bayes is better than Decision Trees.

END OF REPORT