LEEDS BECKETT UNIVERSITY

School of Built Environment, Engineering and Computing

Intelligent Systems & Machine Learning

Dr. Gopal Jamnal

Week 3 – Decision Trees and Random Forest

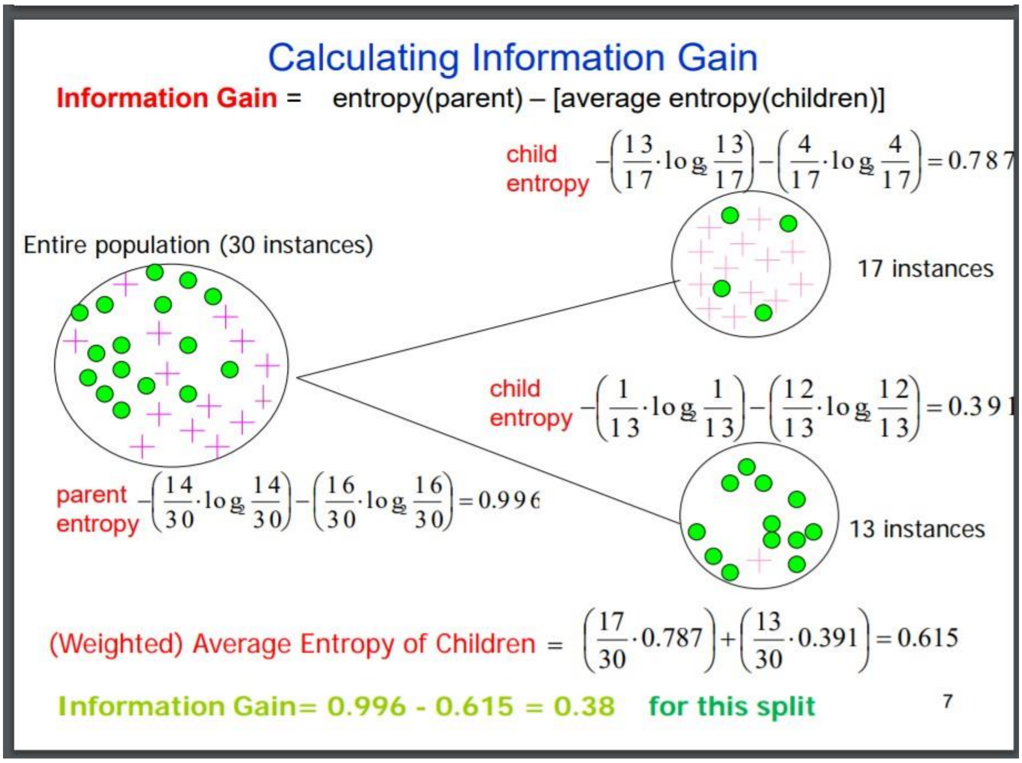
In this section, we'll be working on tree-based models such as decision tree and random forest. Open rapid miner and import example dataset to perform lab tasks.

Supporting Tasks

Read the RapidMiner Documentation on Decision Trees

Information Gain

Given the following dataset, calculate the Information Gain of the first node split.



Calculating Information Gain

Information Gain = entropy(parent) – [average entropy(children)]

child entropy $-\left(\frac{13}{17}\cdot\log_2\frac{13}{17}\right)-\left(\frac{4}{17}\cdot\log_2\frac{4}{17}\right)=0.787$

child entropy $-\left(\frac{1}{13}\cdot\log_2\frac{1}{13}\right)-\left(\frac{12}{13}\cdot\log_2\frac{12}{13}\right)=0.391$

parent entropy $-\left(\frac{14}{30}\cdot\log_2\frac{14}{30}\right)-\left(\frac{16}{30}\cdot\log_2\frac{16}{30}\right)=0.996$

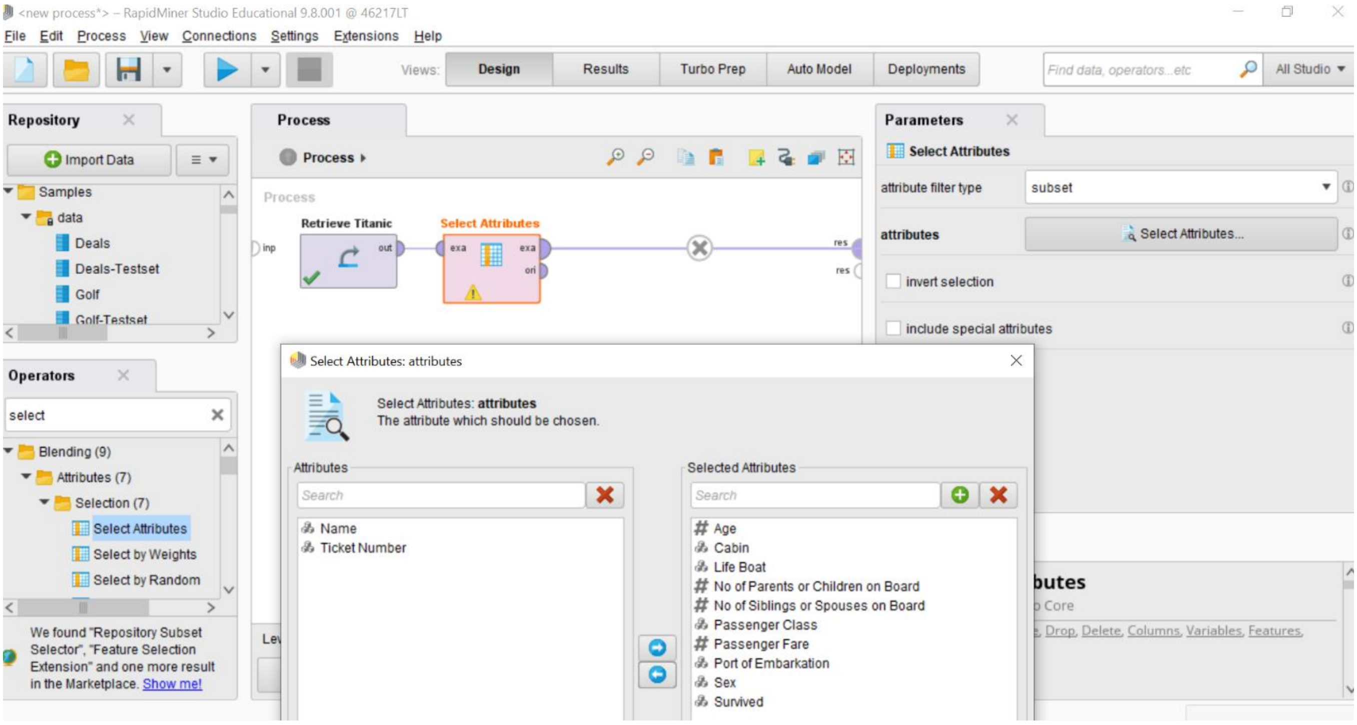
(Weighted) Average Entropy of Children $=\left(\frac{17}{30}\cdot0.787\right)+\left(\frac{13}{30}\cdot0.391\right)=0.615$

Information Gain= 0.996 - 0.615 = 0.38 for this split

Source is: https://homes.cs.washington.edu/~shapiro/EE596/notes/InfoGain.pdf

Decision Tree

Step 1: first import the sample dataset and use select attributes operator to filter data. As you can see, Name and Ticket Number are filtered out from selected attributes.


<new process*> – RapidMiner Studio Educational 9.8.001 @ 46217LT
File Edit Process View Connections Settings Extensions Help
Views: Design Results Turbo Prep Auto Model Deployments
Find data, operators...etc
All Studio
Repository
Import Data
Samples
data
Deals
Deals-Testset
Golf
Golf-Testset
Operators
select
Blending (9)
Attributes (7)
Selection (7)
Select Attributes
Select by Weights
Select by Random
We found "Repository Subset Selector", "Feature Selection Extension" and one more result in the Marketplace. Show me!
Process
Process ▸
Retrieve Titanic
Select Attributes
inp
out
exa
exa
ori
res
res
Parameters
Select Attributes

attribute filter type: subset

attributes: Select Attributes...

☐ invert selection

☐ include special attributes

Select Attributes: attributes

Select Attributes: **attributes**
The attribute which should be chosen.

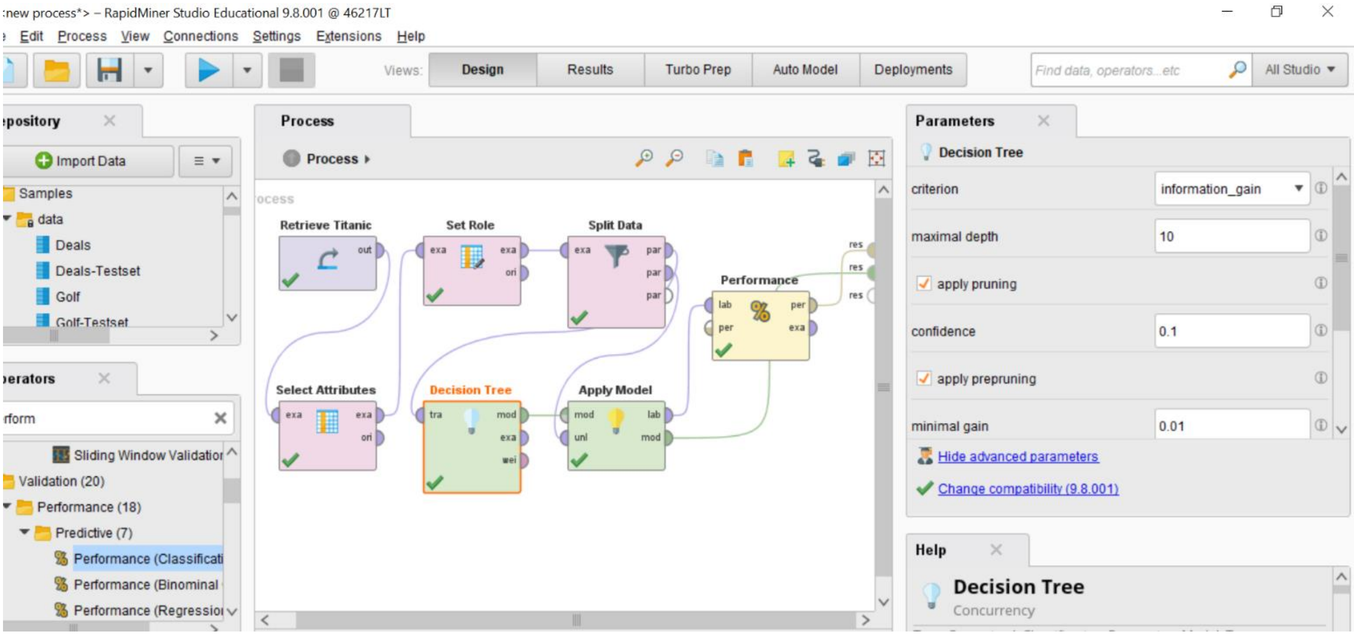
Attributes:
- Name
- Ticket Number

Selected Attributes:
- Age
- Cabin
- Life Boat
- No of Parents or Children on Board
- No of Siblings or Spouses on Board
- Passenger Class
- Passenger Fare
- Port of Embarkation
- Sex
- Survived

Step 2: For the target value, you'll set role to predict survived attribute. Once set role is defined, you'll need to split data into train and test set.

Task: you need to perform the task you did in week 2 tutorial - set role and split data. You can retrospectively revise the concept for these two operator by checking week 2 tutorial sheet.

Step 3: You'll be using decision tree as first classifier with parameters for split criterion>info_gain, pre-post pruning and maximum_depth = 10; as shown in the image below;



Parameters

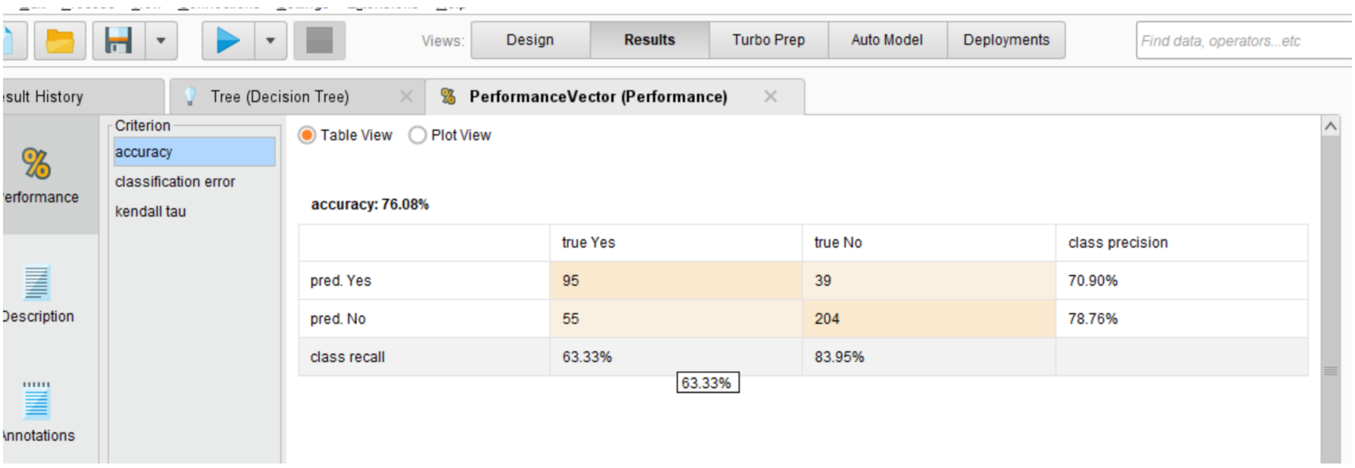
Decision Tree

criterion	information_gain
maximal depth	10
☑ apply pruning	
confidence	0.1
☑ apply prepruning	
minimal gain	0.01

Hide advanced parameters

Change compatibility (9.8.001)

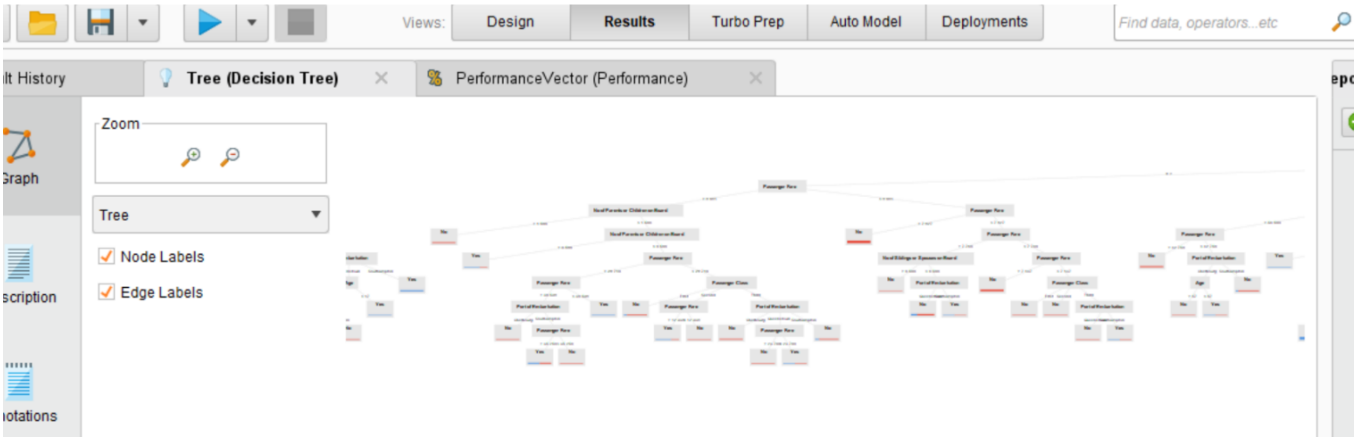
Step 4: Performance Vector analysis – you will be using performance(classification) to generate confusion matrix and accuracy score.



accuracy: 76.08%

	true Yes	true No	class precision
pred. Yes	95	39	70.90%
pred. No	55	204	78.76%
class recall	63.33%	83.95%	

63.33%


Views: Design Results Turbo Prep Auto Model Deployments
Find data, operators...etc
lt History
Tree (Decision Tree)
PerformanceVector (Performance)
Graph
scription
otations
Zoom
Tree
☑ Node Labels
☑ Edge Labels

Random Forest

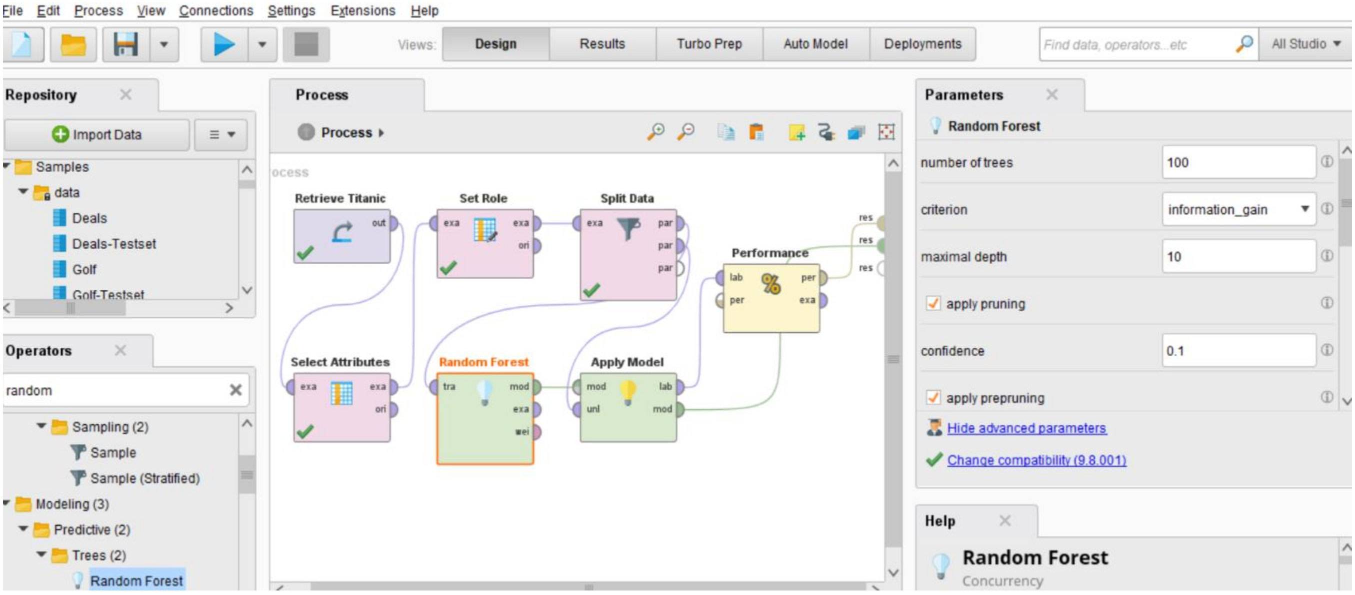
Note: same step for 1,2 and 4 will be used from random forest

Step 1: follow the same previous step from decision tree.

Step 2: follow the same previous step from decision tree.

Step 3: In this step, you'll be using random forest operator and would set parameters for number of tree=100, split criterion=info_gain, pre-post pruning abd maximum_depth=10 as shown in image below;

follow the same previous step from decision tree.



Parameters

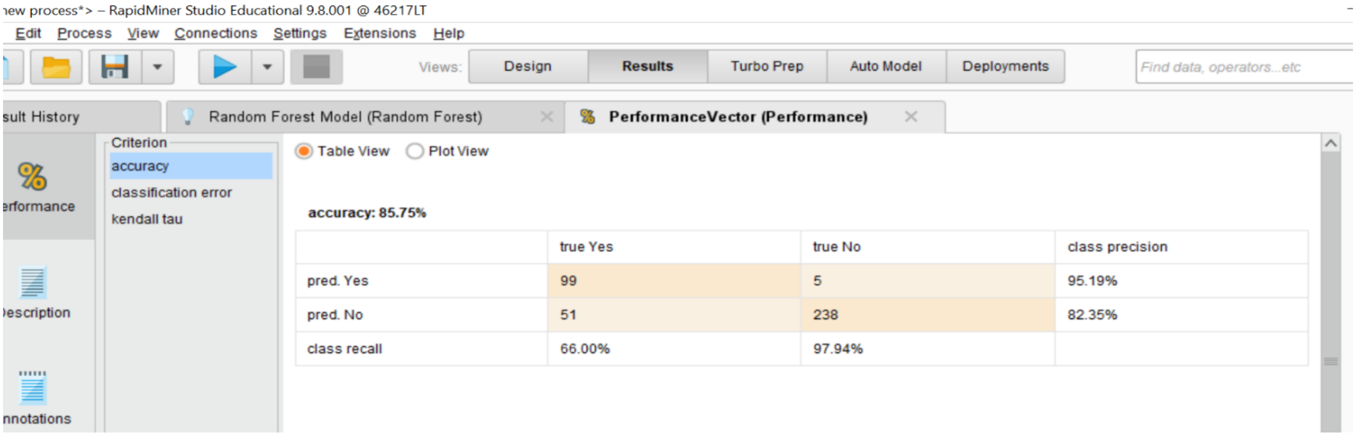
Random Forest

number of trees	100
criterion	information_gain
maximal depth	10
☑ apply pruning	
confidence	0.1
☑ apply prepruning	

Hide advanced parameters

Change compatibility (9.8.001)

Step 4: Analyze confusion matrix and accuracy score. Is it better than decision tree ?



new process*> – RapidMiner Studio Educational 9.8.001 @ 46217LT
Edit Process View Connections Settings Extensions Help
Views: Design Results Turbo Prep Auto Model Deployments
Find data, operators...etc
Result History
Random Forest Model (Random Forest)
PerformanceVector (Performance)
Performance
Description
Annotations
Criterion
accuracy
classification error
kendall tau
● Table View ○ Plot View

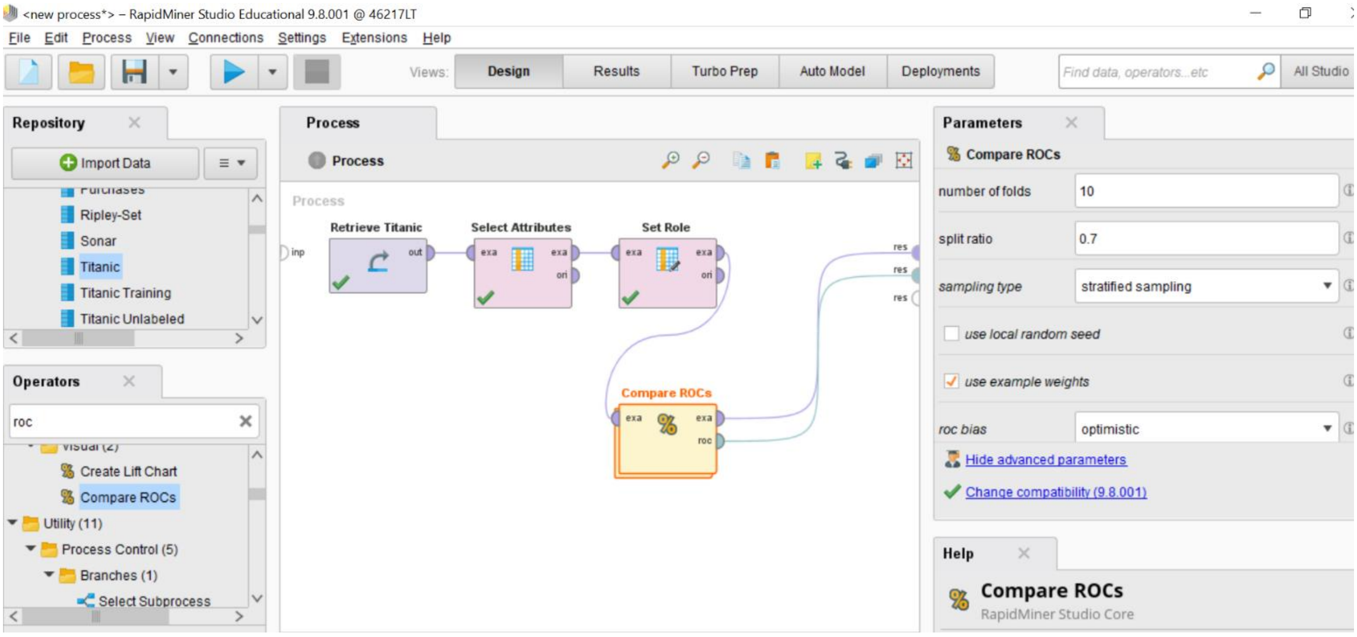
accuracy: 85.75%

	true Yes	true No	class precision
pred. Yes	99	5	95.19%
pred. No	51	238	82.35%
class recall	66.00%	97.94%	

Compare Decision Tree and Random Forest Performance with ROC

In the tutorial, we'll be using two different models named, decision tree and random forest. For the performance evaluation of both models, we'll compare roc curves. ROC curve is a graphical plot of the sensitivity, or true positive rate, vs. false positive rate (one minus the specificity or true negative rate), for a binary classifier system as its discrimination threshold is varied. The ROC can also be represented equivalently by plotting the fraction of true positives out of the positives (TPR = true positive rate) vs. the fraction of false positives out of the negatives (FPR = false positive rate). ROC curves are calculated by first ordering the classified examples by confidence. Afterwards all the examples are taken into account with decreasing confidence to plot the false positive rate on the x-axis and the true positive rate on the y-axis.

Step 3: you'll be using compare ROCs operator for decision tree and random forest comparison.



Parameters – Compare ROCs

number of folds: 10

split ratio: 0.7

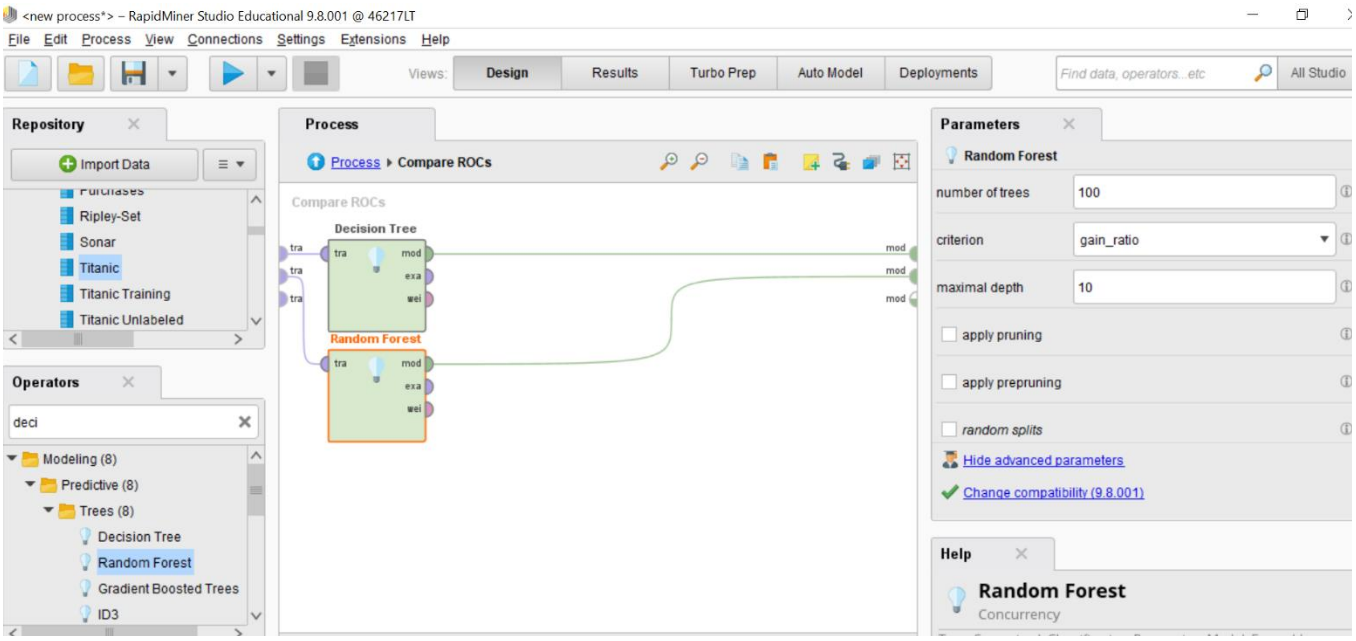
sampling type: stratified sampling

☐ use local random seed

☑ use example weights

roc bias: optimistic

Inside ROC, you will be using two classifiers as shown below;



number of trees: 100

criterion: gain_ratio

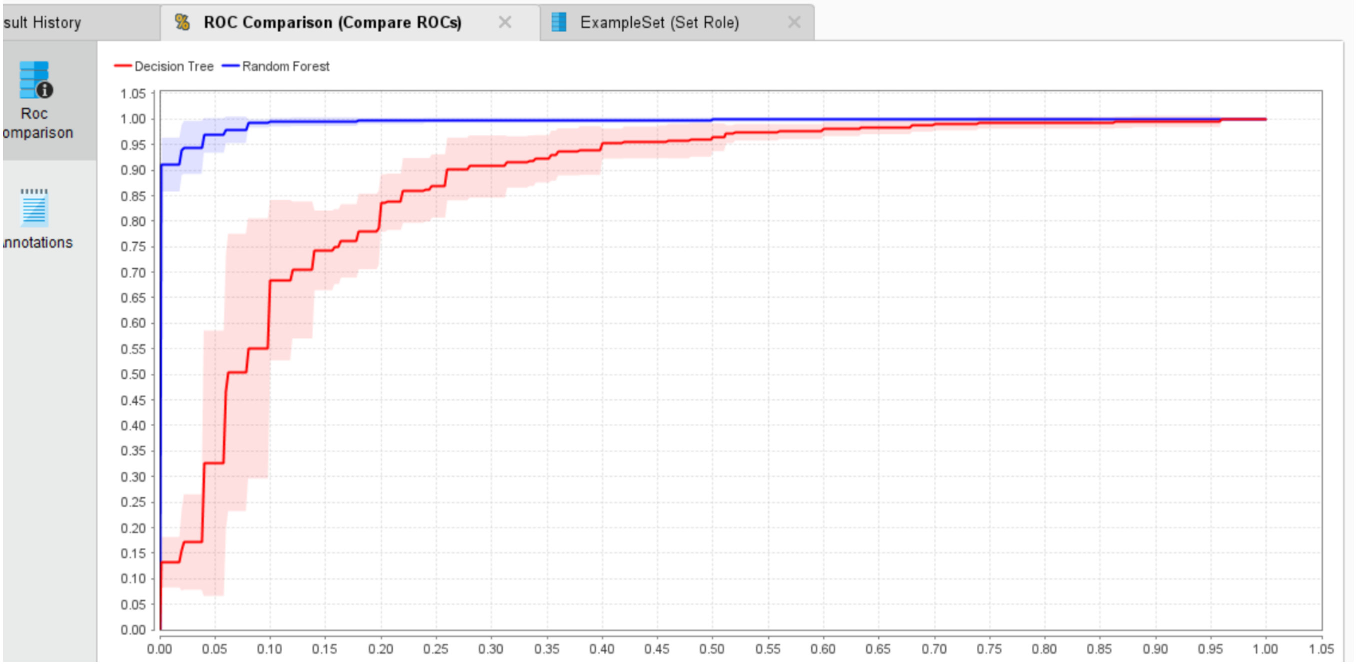
maximal depth: 10

☐ apply pruning

☐ apply prepruning

☐ random splits

Compare ROC curve and analyze which one is better to predict true positive(tp) cases correctly?



Exercise: for the practice and gain deeper understanding, look for other sample dataset and try to interpret their performance with confusion matrix and roc curve for true positive and false positive rates.