WEKA

BIOINFORMATICS AND BIG DATA ANALYTICS

Ing. Antonio Brunetti
Prof. Vitoantonio Bevilacqua

Indice

Cosa è weka
- Tecnologie

Hands On Weka
- Weka Explorer
  - KnowledgeFlow /Simple CLI
- Caricare il dataset
- Il file arff
- Integrazione e valutazione dei dati
- Classificatori
  - Multilayer Perceptron
  - Random Forest
  - J48
- Matrice di Confusione e interpretazione dei risultati
- Esportazione delle reti addestrate
- Esempi pratici

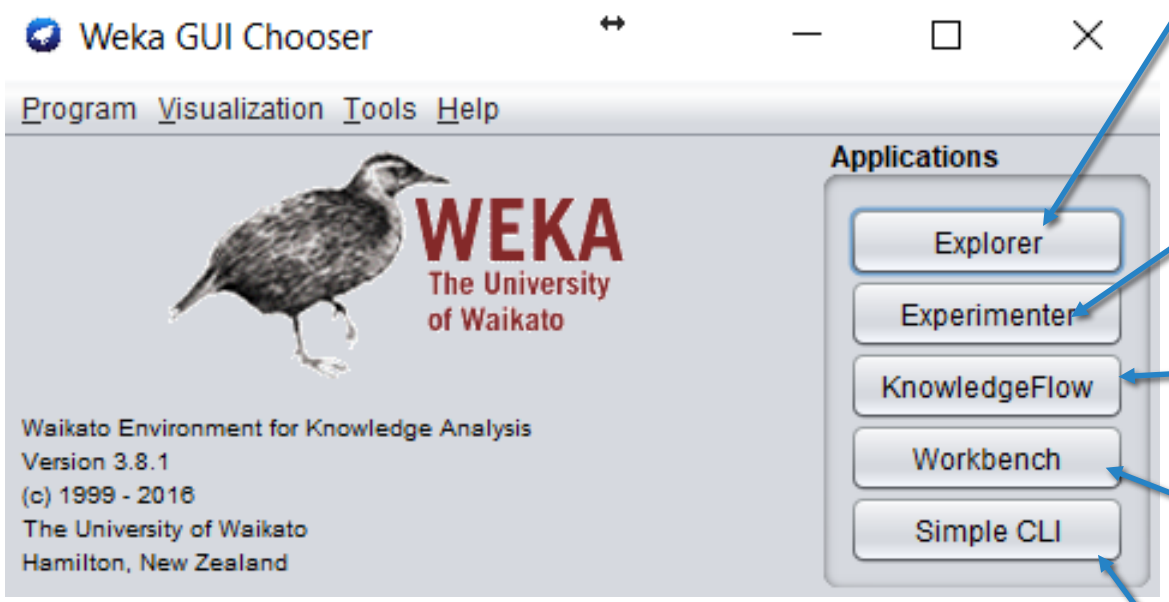
Cosa è WEKA

- Sviluppato dall'Università Waikato in Nuova Zelanda.
- Acronimo per **W**aikato **E**nvironment for **K**nowledge **A**nalysis.
- È una collezione di algoritmi e tecniche di apprendimento automatico (**machine learning**).
- È stato disegnato per testare velocemente metodologie esistenti su insiemi di dati diversi, in modo flessibile.
- Set completo di strumenti per il pre-processing, algoritmi di apprendimento e metodi di valutazione.
- Disponibile gratuitamente all'indirizzo http://www.cs.waikato.ac.nz/ml/weka/

Tecnologie

- Weka è realizzato in Java
- Implementa un vastissimo numero di algoritmi di classificazione e clustering
  - Reti Bayesiane
  - Funzioni di regressione e EBP
  - Classificatori basati su regole
  - Classificatori basati su alberi
- È possibile utilizzare in applicativi JAVA i motori di Weka ed importare direttamente i classificatori addestrati

Hands on Weka



EXPLORER
Funzionalità completa di clustering e classificazione

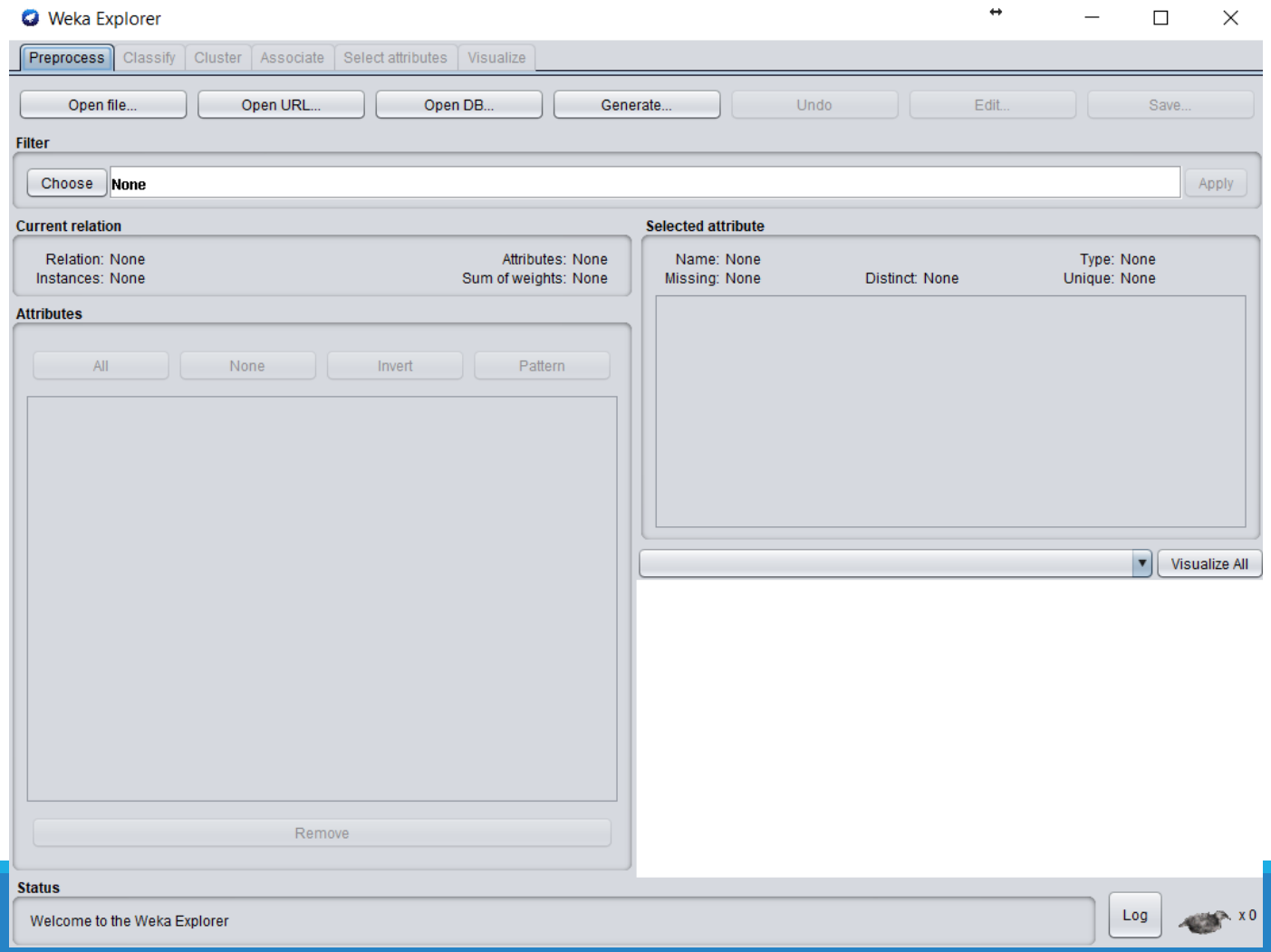
EXPERIMENTER
Permette un approccio di aggregazione dati per cercare l'algoritmo più corretto

KnowledgeFlow
Possibilità di modellare la classificazione in un ambiente simulink-like

Wokbench
Integra tutti i controlli Explorer, KnowledgeFlow E CLI

Simple CLI
Controllo a riga di comando

Weka Explorer



Il file ARFF

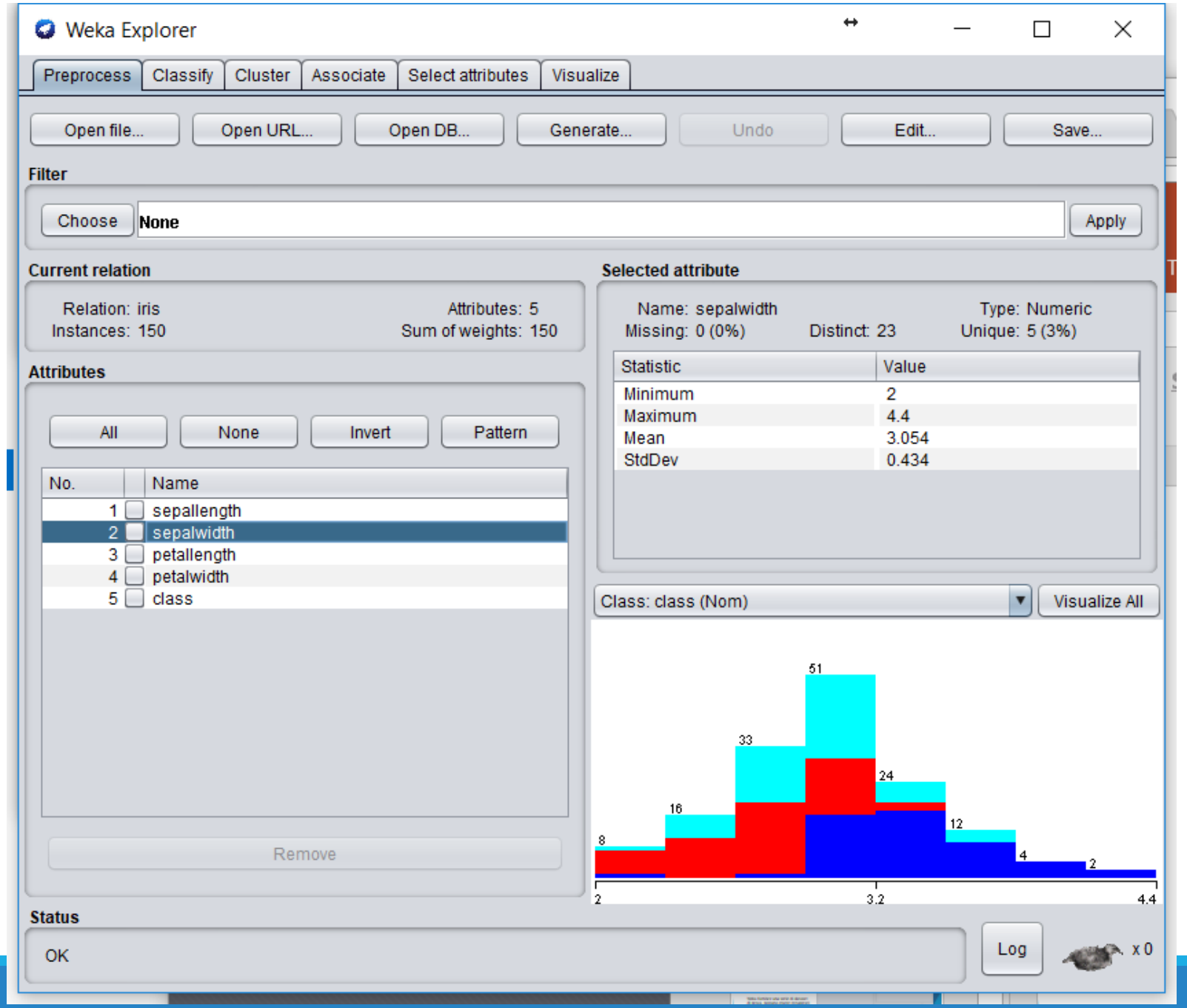
È un file testuale composto da due sezioni

- Header
- Dataset

Nella dichiarazione vengono presentati e categorizzati i dati che saranno poi presenti nella sezione @DATA


```
@RELATION iris
@ATTRIBUTE sepallength        REAL
@ATTRIBUTE sepalwidth         REAL
@ATTRIBUTE petallength        REAL
@ATTRIBUTE petalwidth         REAL
@ATTRIBUTE class {Iris-setosa,Iris-versicolor,Iris-virginica}

@DATA
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
[…]
```


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
None
Apply
Current relation
Relation: iris
Instances: 150
Attributes: 5
Sum of weights: 150
Attributes
All
None
Invert
Pattern
No.
Name
1 sepallength
2 sepalwidth
3 petallength
4 petalwidth
5 class
Remove
Selected attribute
Name: sepalwidth
Missing: 0 (0%)
Distinct: 23
Type: Numeric
Unique: 5 (3%)
Statistic
Value
Minimum 2
Maximum 4.4
Mean 3.054
StdDev 0.434
Class: class (Nom)
Visualize All
51
33
24
16
12
8
4
2
2
3.2
4.4
Status
OK
Log
x 0

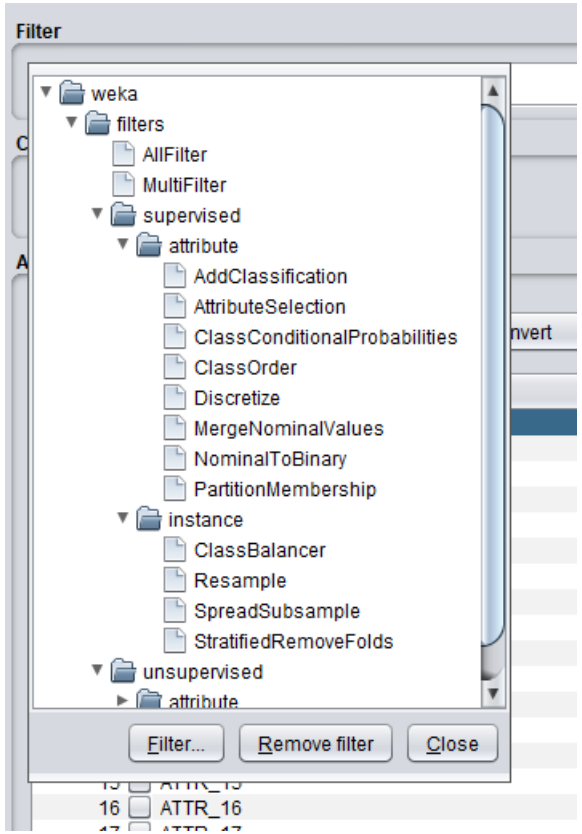
Integrazione e valutazione del dataset

Le funzioni di gestione dei dati (filtri) possono essere

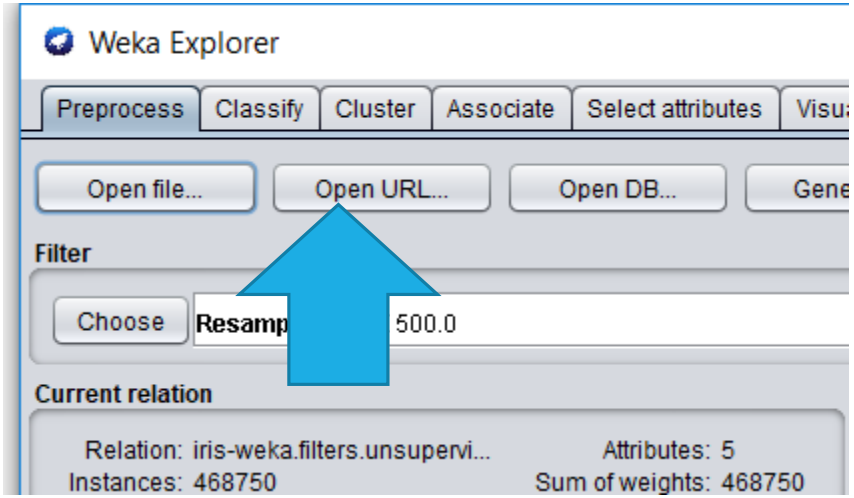
- Supervised: usano informazioni sulla classe per determinare gli attributi/istanze per effettuare l'operazione
- Unsupervised: non usano le informazioni sulla classe

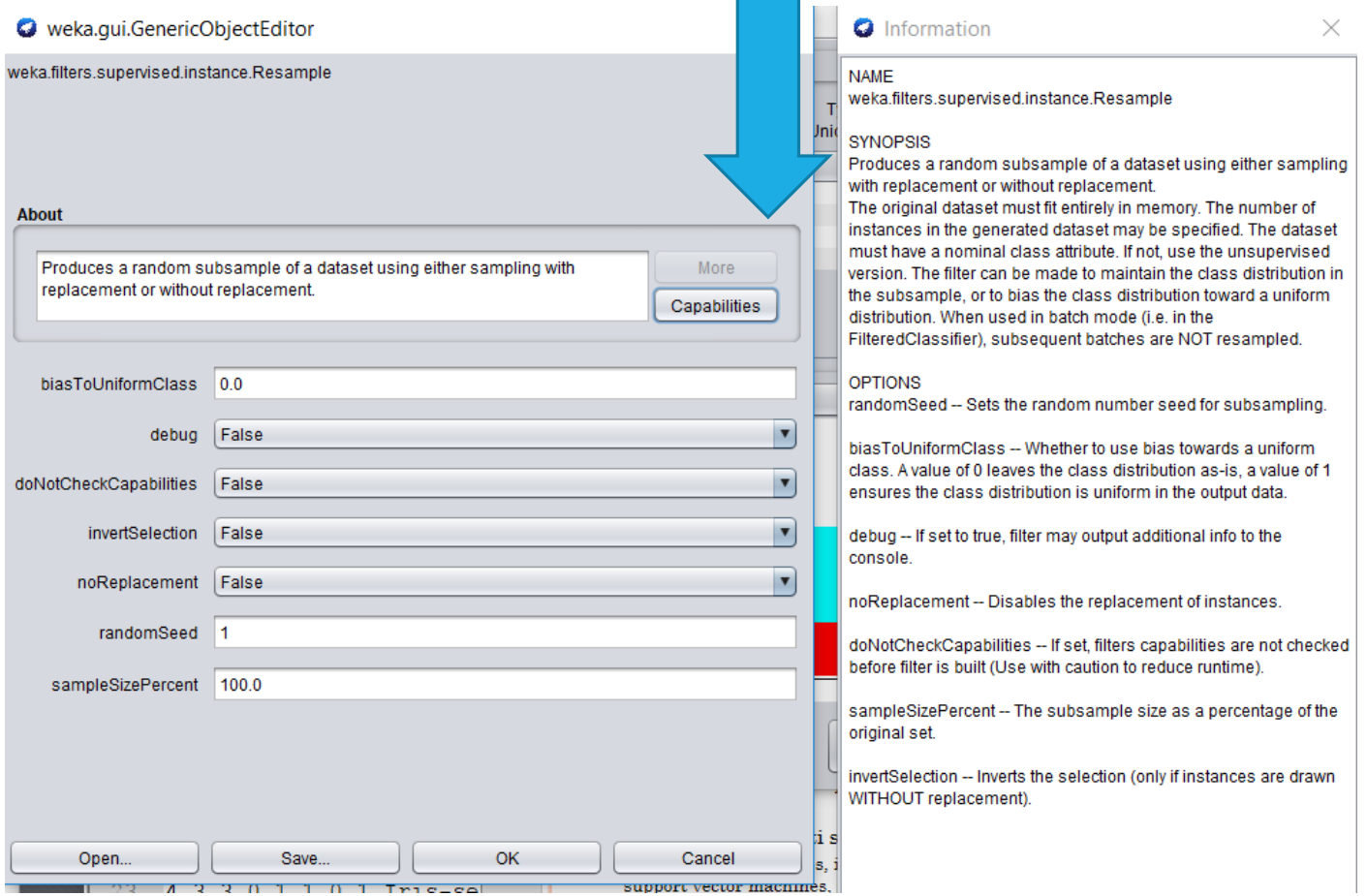
Inoltre, si distingue tra:

- Attribute filters: effettua un'operazione di attributo
- Instance filters: effettua un'operazione sulle istanze di un determinato attributo



Informazioni dettagliate

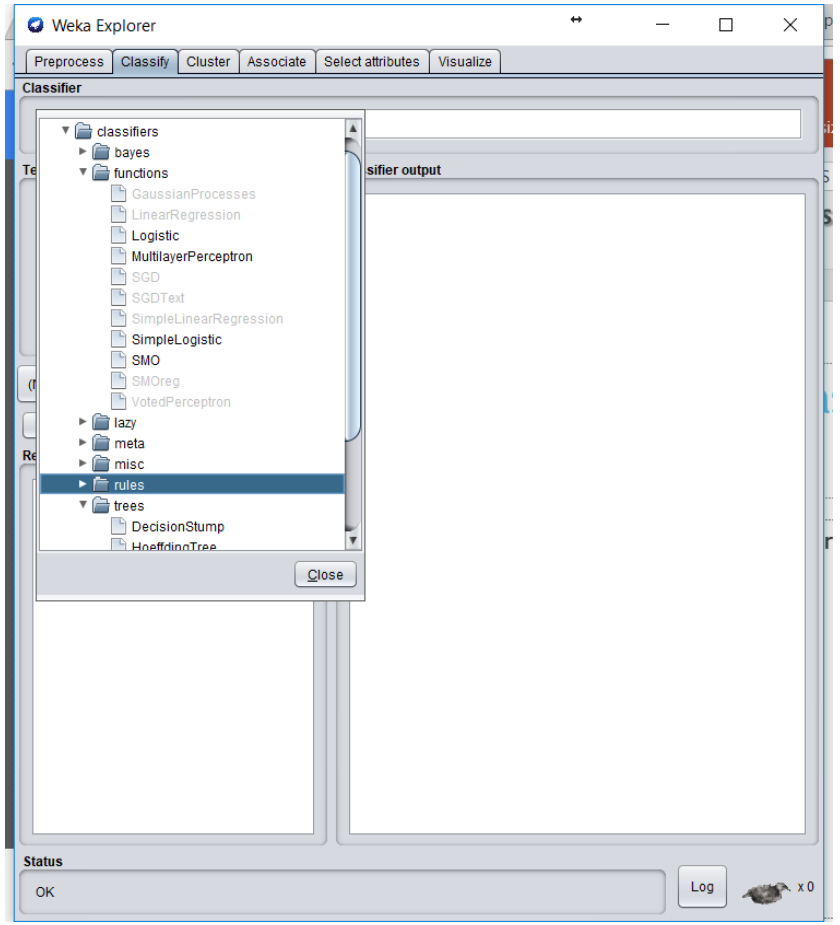




Classify

Contiene i metodi per effettuare operazioni di classificazione

- Test Options:
  - Permette di specificare le opzioni per il training e il test del classificatore
    - Use training set
    - Supplied test set
    - Cross validation
    - Percentage Split
  - Specificare l'attributo da usare come classe



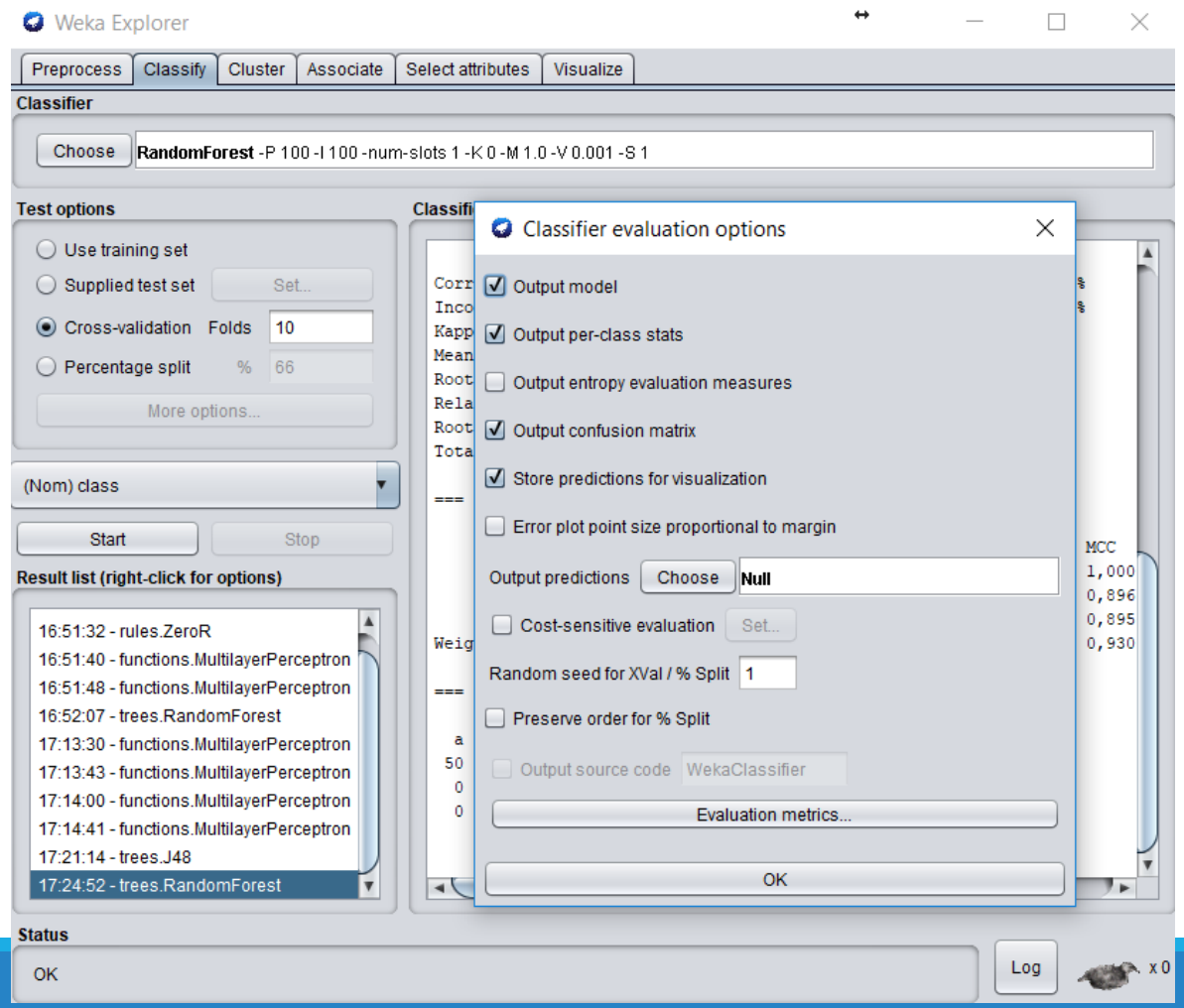
Output di Classificazione

More Options

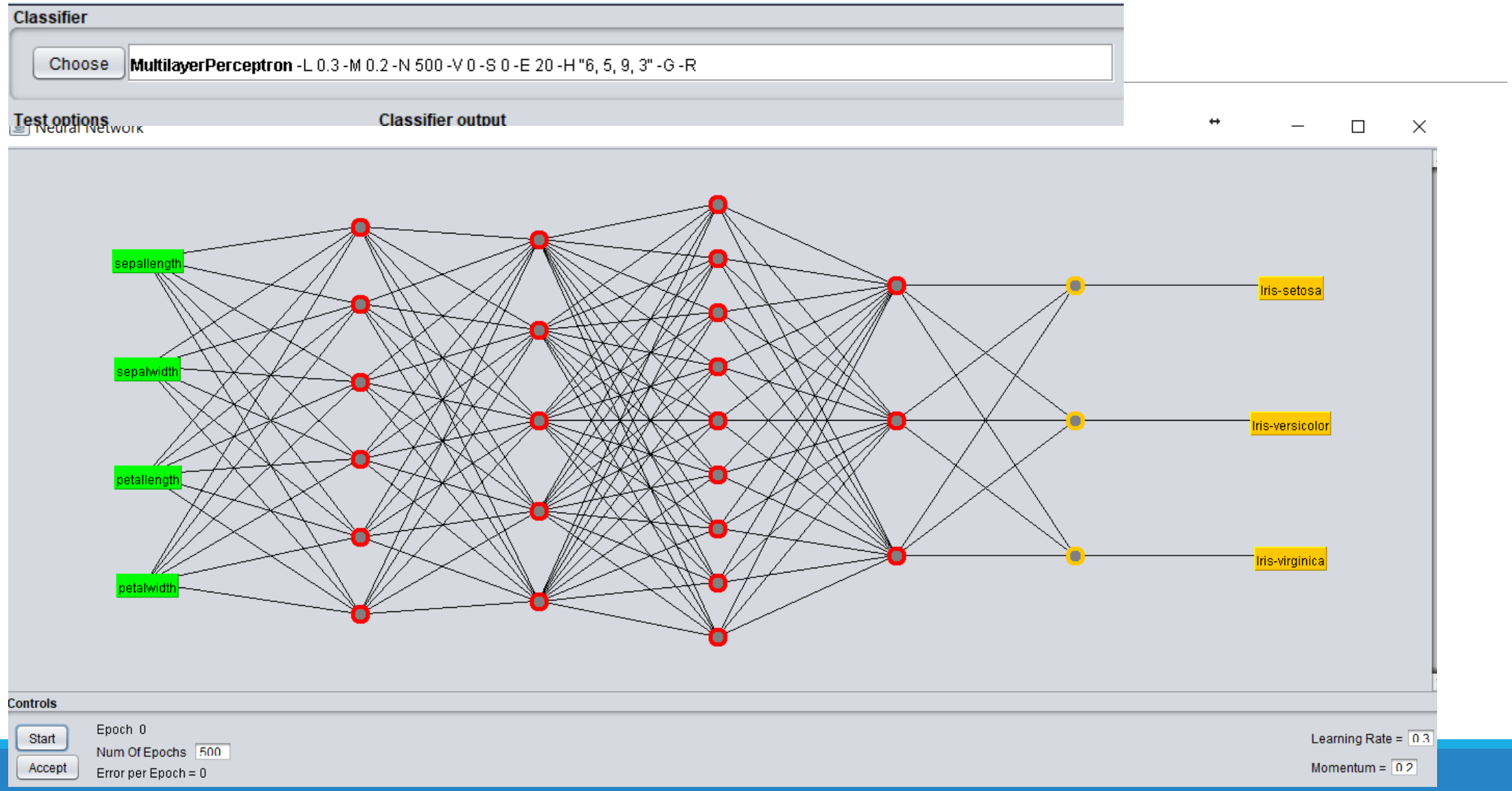
Result List

- Show Tree
- Roc Curve
- Model Export

Classifier output



Esempio di Multilayer Perceptron



Impostazioni Principali

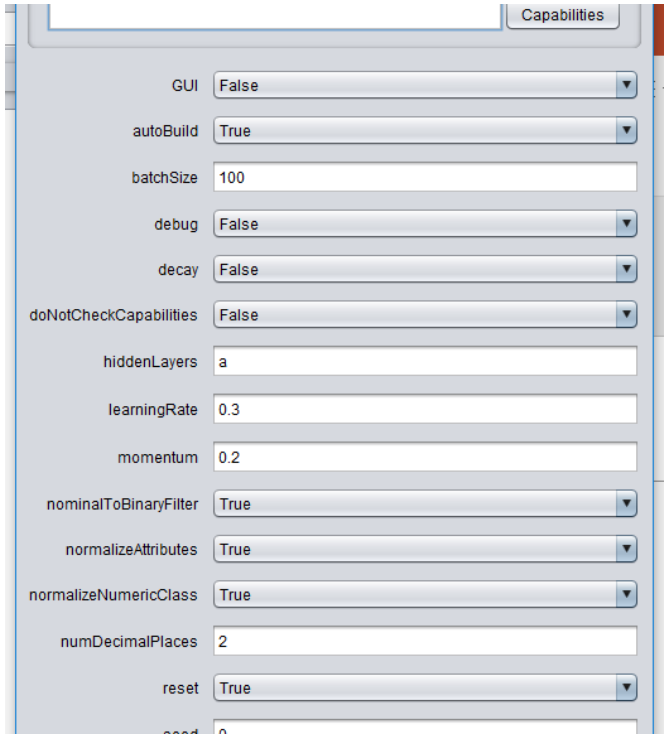
GUI: Mostra il layout della rete e permette di modificarlo

Learning Rate: di quanto vengono aggiornati i pesi, accelara il processo inizialmente ma rende piu difficile il fine tunig

Hidden layers, numero di neuroni per layer, implementa la topologia della rete Es: 6,7,3,2

Seed: Configurazione iniziale dei pesi

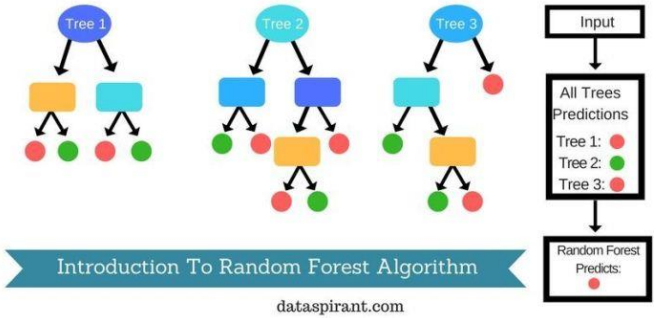
Training Time: Numero di epoche di addestramento prima che l'algoritmo si fermi

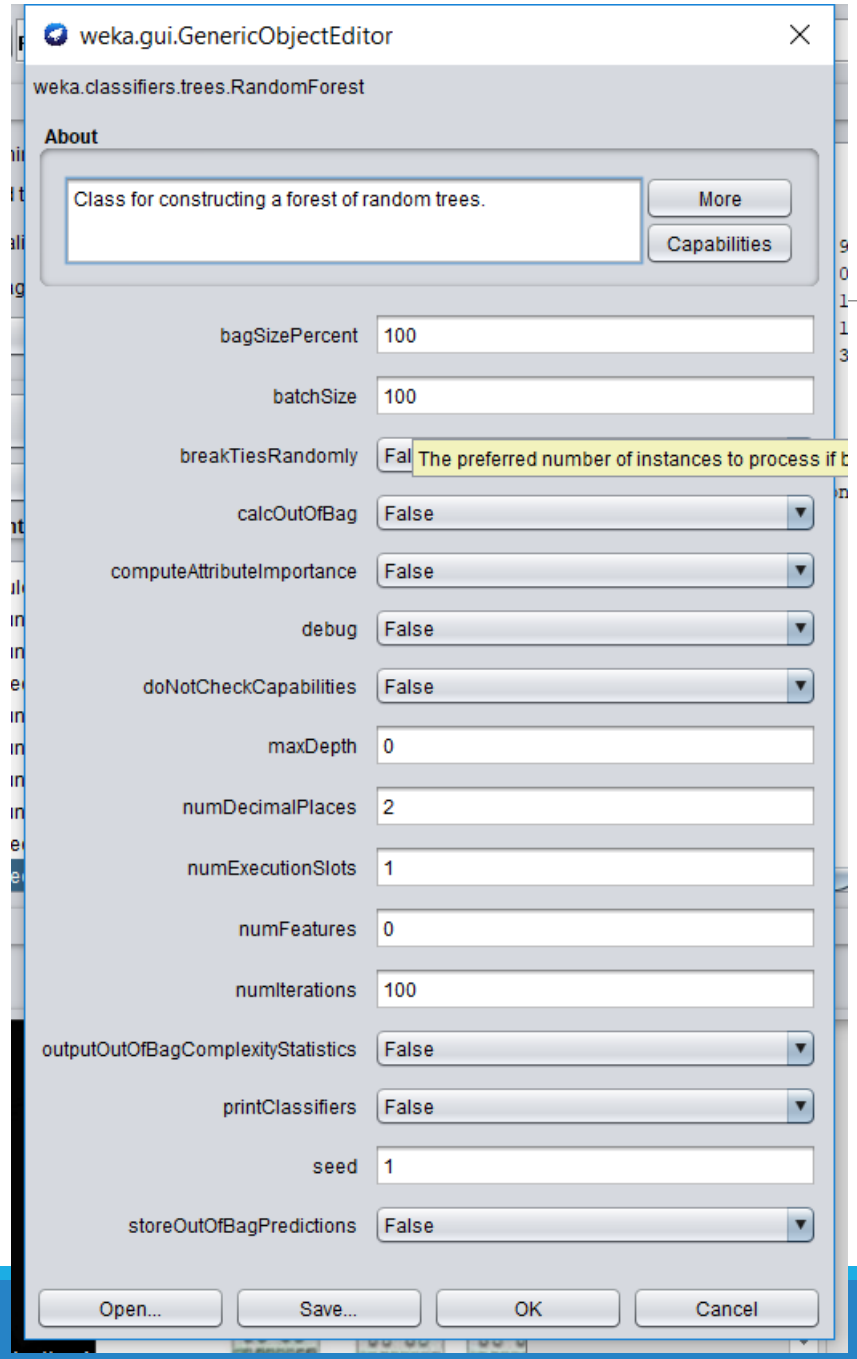


Random Forest

Le random forest sono uno strumento di classificazione introdotto per la prima volta nel 2001 da Leo Breiman, nate da lavori precedenti dello stesso autore volti a migliorare le prestazioni ottenibili da singoli alberi di decisione, introducendo alcune componenti aleatorie nella costruzione degli stessi e un meccanismo di voto per la determinazione del risultato della classificazione.

Oltre alla migliorata accuratezza, le foreste di decisione sono interessanti per la loro efficienza, dovuta al parallelismo sia in fase di costruzione sia in quella di classificazione.





Impostazioni Principali RF

batchSize – Numero di alberi nella foresta

Print Classifiers - mostra ogni singolo albero

maxDepth - Massima dimensione dell'albero

numFeatures – Numero di features utilizzate

J48

J48 è l'implementazone Java dell algoritmo C4.5.

C4.5 è un algoritmo utilizzato per generare un albero decisionale sviluppato da Ross Quinlan, è un'estensione del precedente algoritmo ID3 di Quinlan .

Gli alberi decisionali generati dal C4.5 possono essere utilizzati per la classificazione, e per questo motivo, C4.5 viene spesso definito classificatore statistico

PRO:

- Espressivi quanto un Decision Tree
- Facili da interpretare
- Facili da generare
- Rapidi nella classificazione di nuove istanze

CONTRO:

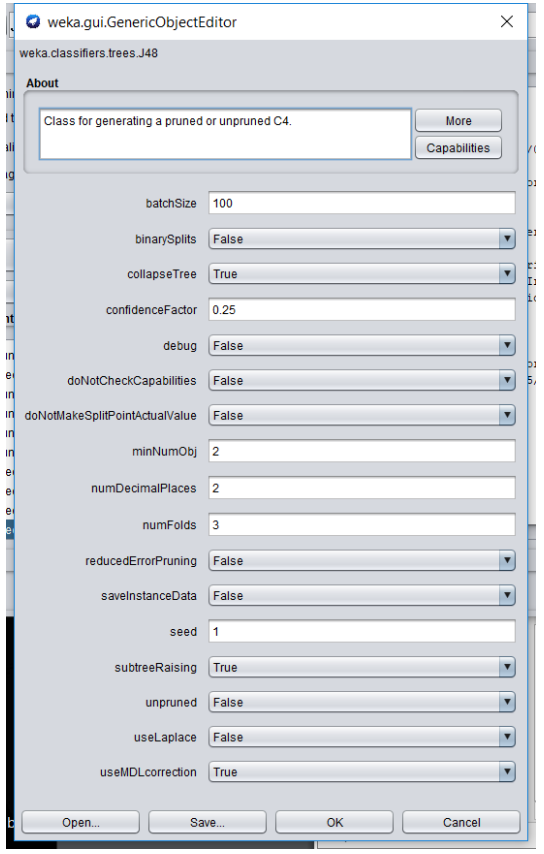
- Il costo di costruzione non scala al crescere del training set
- Risentono fortemente del rumore sui dati

Impostazioni Principali J48

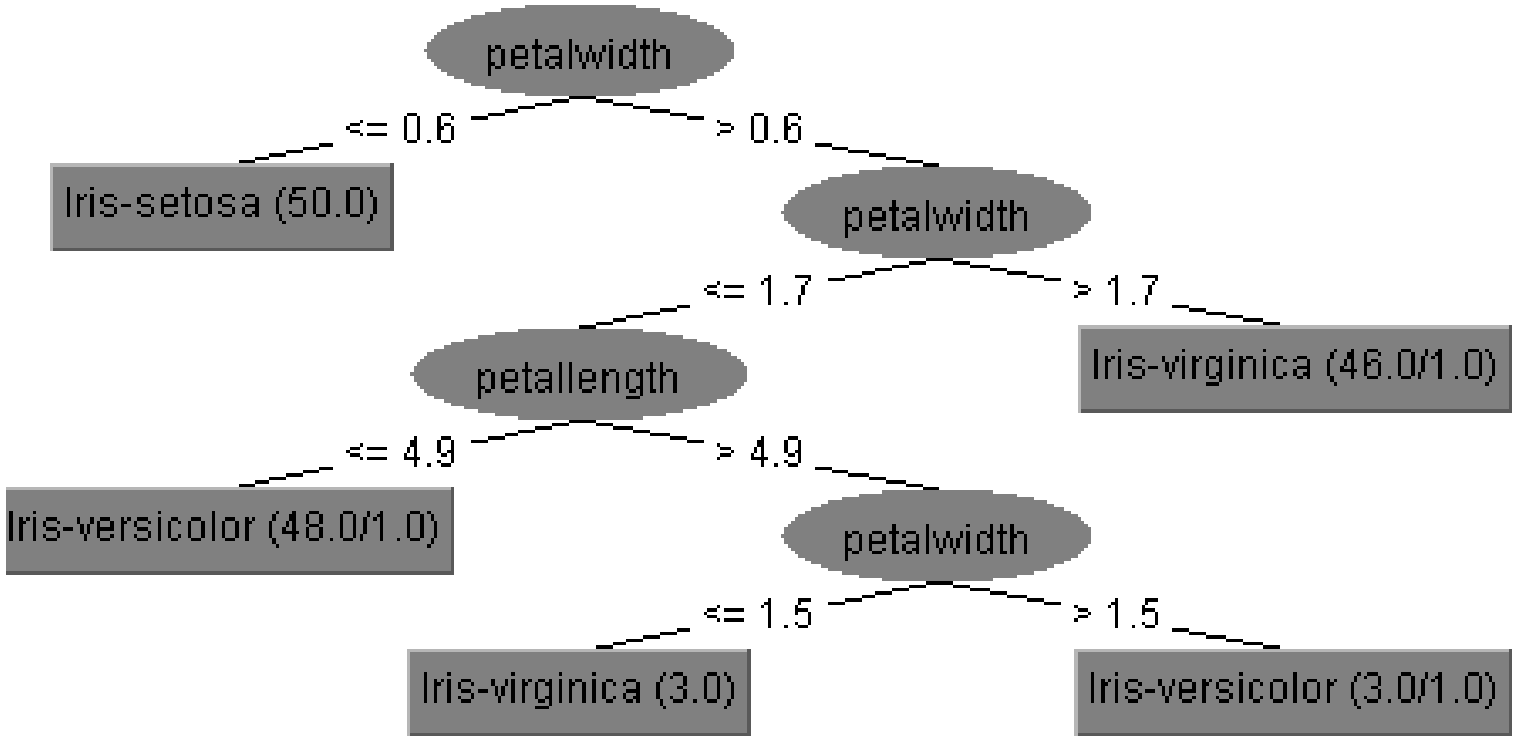
Il pruning è una attività fondamentale nella creazione dell'albero, molte impostazioni sono volte a migliorare questo aspetto

- Confidence Factor
- Num Folds
- reduceErrorPruning: utilizza un altro algoritmo rispetto a quello standard
- Unpruned

binarySplits – se false permette la creazione di split non binari



Visualizzazione dell'albero generato dal J48



Classifier Output

=== Run information ===

=== Classifier model (full training set) ===

=== Stratified cross-validation ===

=== Summary ===

=== Detailed Accuracy By Class ===

=== Confusion Matrix ===

La schermata Classifier Output

=== Summary ===

Correctly Classified Instances		144	96 %
Incorrectly Classified Instances		6	4 %
Kappa statistic		0.94	
Mean absolute error	0.035		
Root mean squared error		0.1586	
Relative absolute error	7.8705 %		
Root relative squared error		33.6353 %	
Total Number of Instances		150	

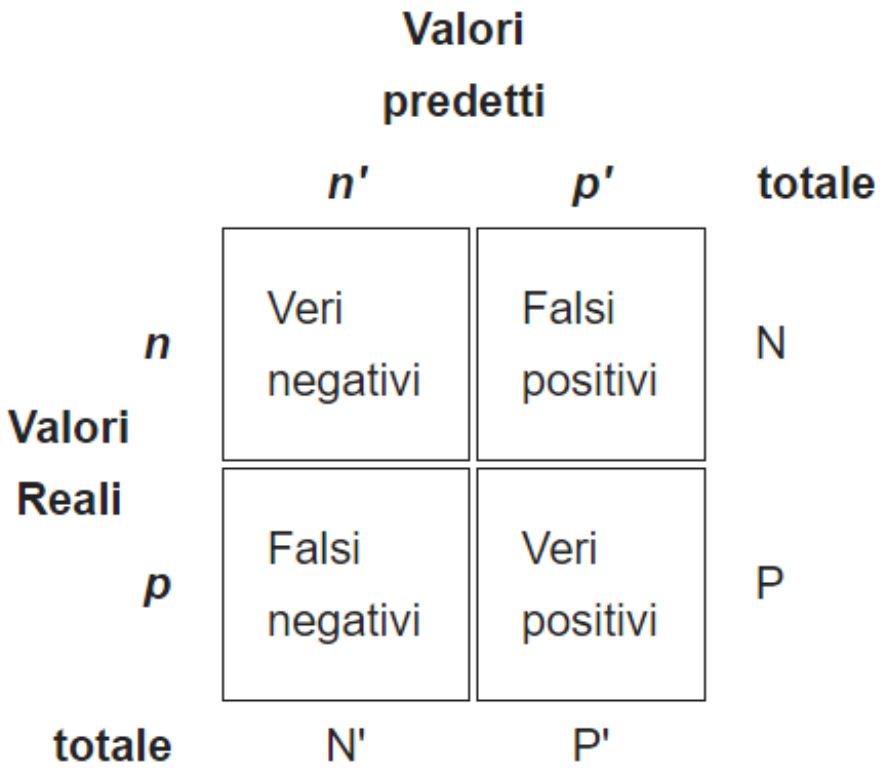
=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,980	0,000	1,000	0,980	0,990	0,985	0,990	0,987	Iris-setosa
	0,940	0,030	0,940	0,940	0,940	0,910	0,952	0,880	Iris-versicolor
	0,960	0,030	0,941	0,960	0,950	0,925	0,961	0,905	Iris-virginica
Weighted Avg.	0,960	0,020	0,960	0,960	0,960	0,940	0,968	0,924	

=== Confusion Matrix ===

```
 a  b  c   <-- classified as
49  1  0 |  a = Iris-setosa
 0 47  3 |  b = Iris-versicolor
 0  2 48 |  c = Iris-virginica
```

Confusion Matrix

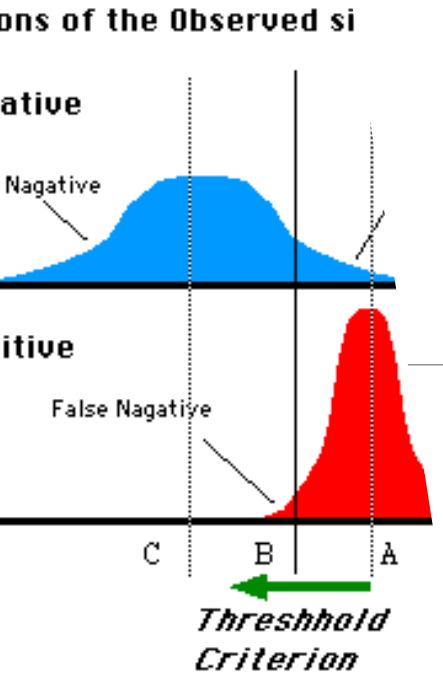


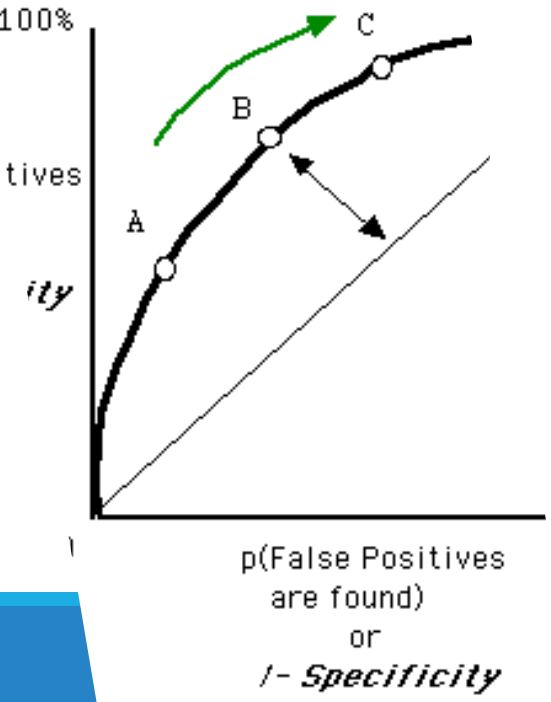

```
  a   b   c    <-- classified as
 49   1   0  |   a = setosa
  0  47   3  |   b = versicolor
  0   2  48  |   c = virginica
```

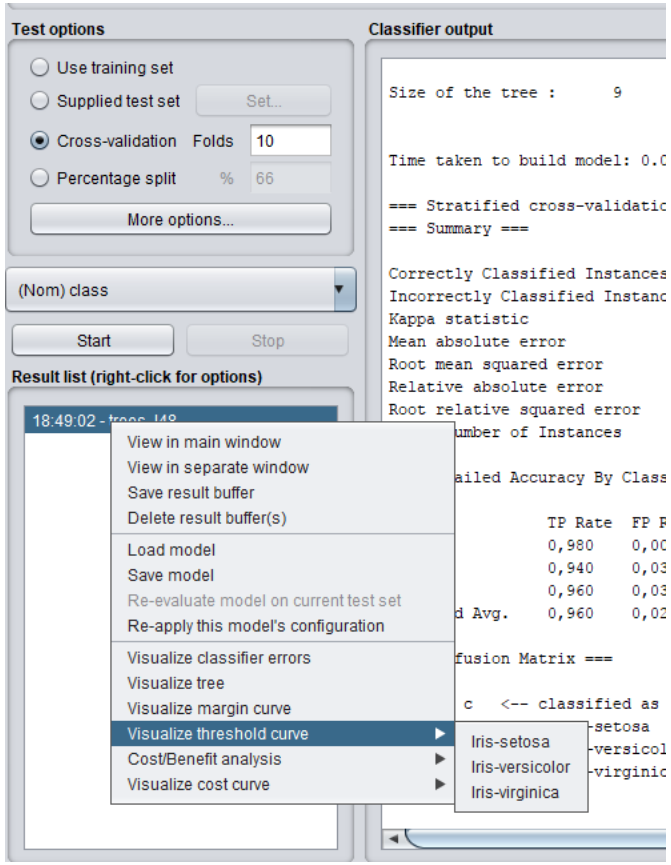
Indica che 3 campioni di Versicolor sono stati riconosciuti come Virginica

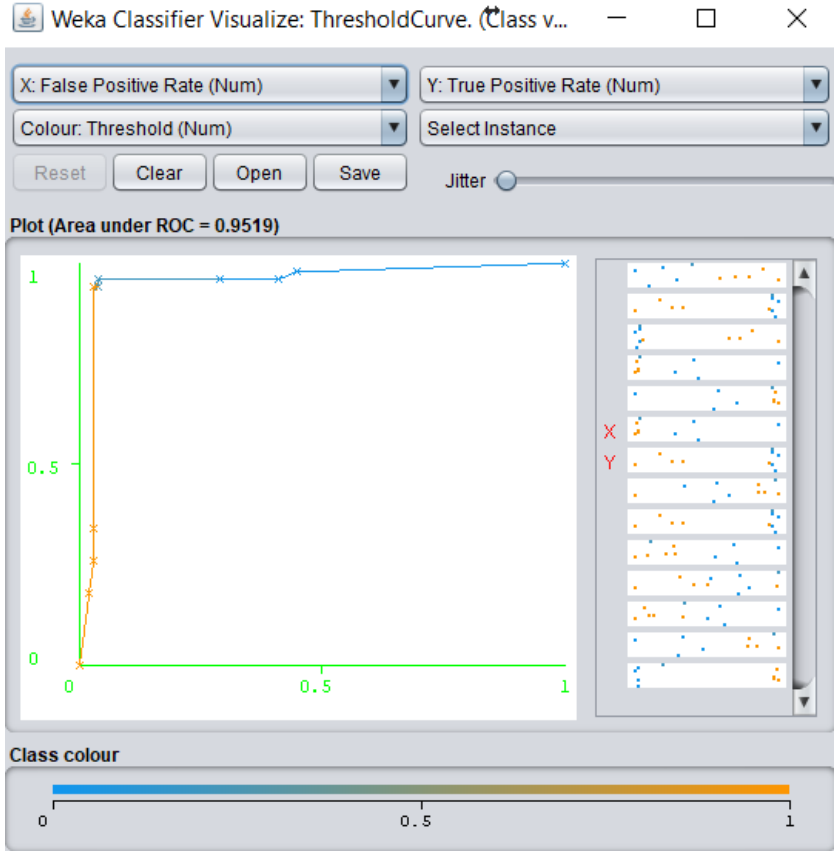
Indica che 2 campioni di `virginica` sono stati riconosciuti come `versicolor`

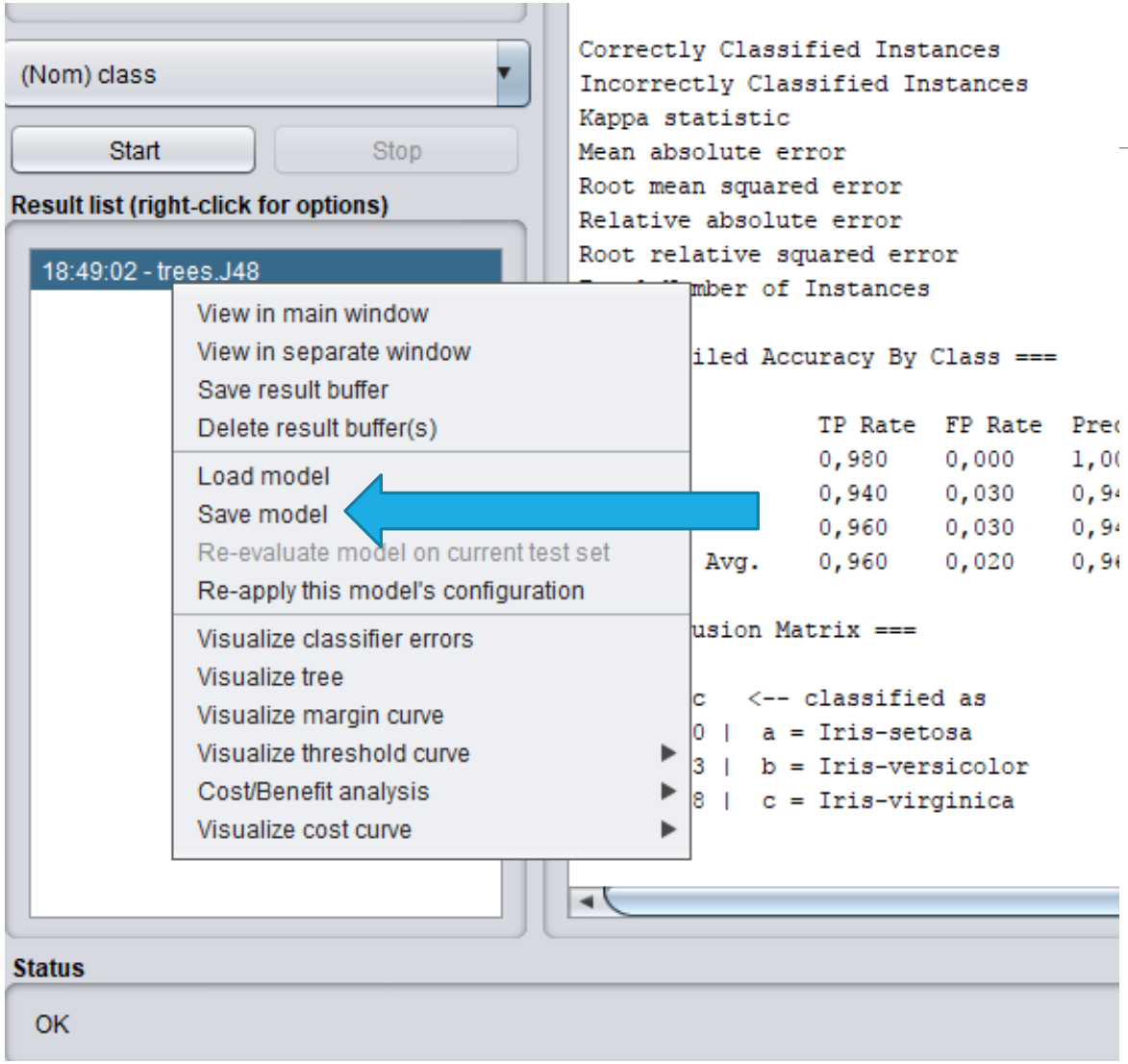
Curve ROC


ons of the Observed si
ative
Nagative
itive
False Nagative
C
B
A
Threshhold
Criterion


100%
C
B
A
tives
ity
p(False Positives
are found)
or
/- Specificity


Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
(Nom) class
Start
Stop
Result list (right-click for options)
Classifier output
Size of the tree : 9
Time taken to build model: 0.0
=== Stratified cross-validatic
=== Summary ===
Correctly Classified Instances
Incorrectly Classified Instanc
Kappa statistic
Mean absolute error
Root mean squared error
Relative absolute error
Root relative squared error
umber of Instances
ailed Accuracy By Class
TP Rate FP R
0,980 0,00
0,940 0,03
0,960 0,03
d Avg. 0,960 0,02
fusion Matrix ===
c <-- classified as
-setosa
-versicol
-virginic
View in main window
View in separate window
Save result buffer
Delete result buffer(s)
Load model
Save model
Re-evaluate model on current test set
Re-apply this model's configuration
Visualize classifier errors
Visualize tree
Visualize margin curve
Visualize threshold curve
Cost/Benefit analysis
Visualize cost curve
Iris-setosa
Iris-versicolor
Iris-virginica


Weka Classifier Visualize: ThresholdCurve. (Class v...
X: False Positive Rate (Num)
Y: True Positive Rate (Num)
Colour: Threshold (Num)
Select Instance
Reset
Clear
Open
Save
Jitter
Plot (Area under ROC = 0.9519)
1
0.5
0
0
0.5
1
X
Y
Class colour
0
0.5
1

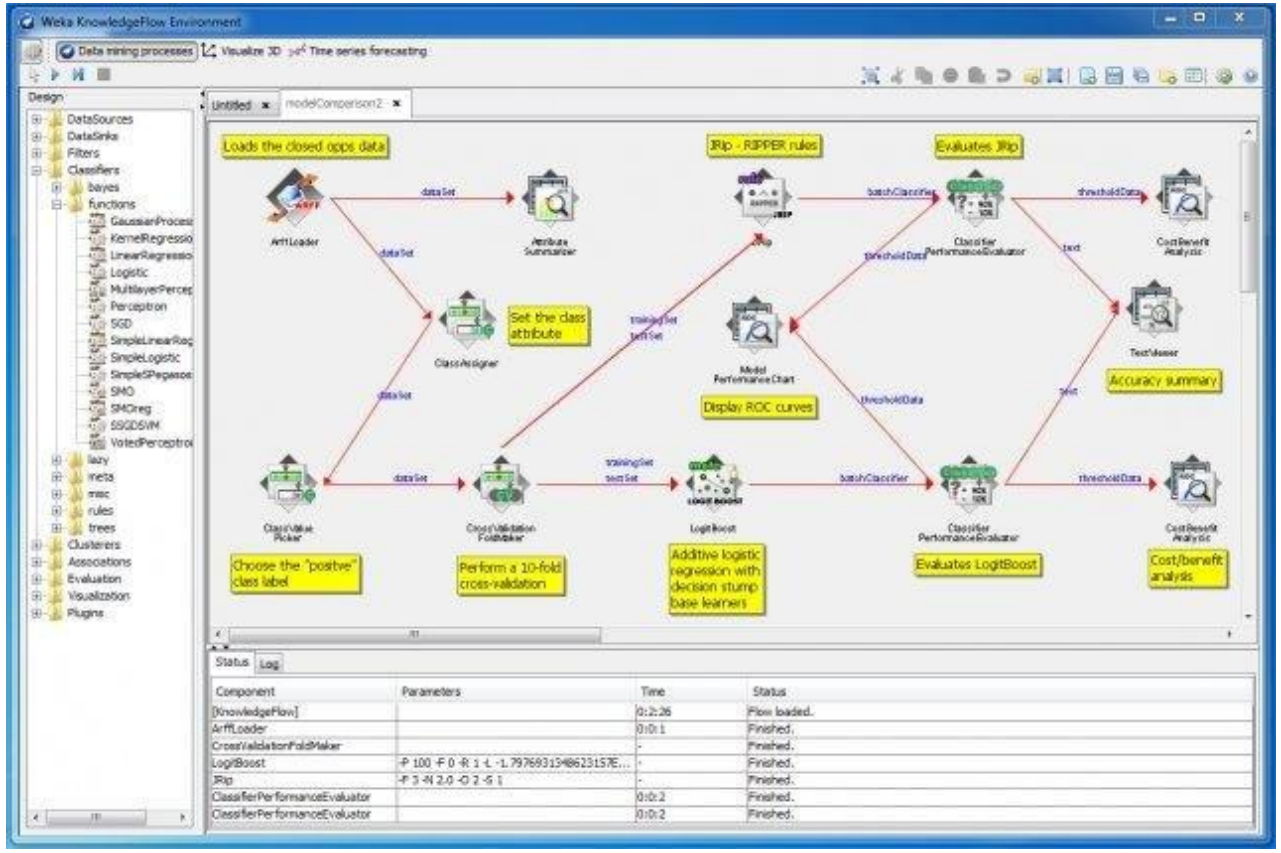


Esportazione dei motori addestrati

E’ possibile salvare I motori addestrati da weka in file .model

Questi file possono essere richiamati per valutare nuovi dataset senza dover riaddestrare la rete, operazione che spesso può richiedere molto tempo

Knowledge Flow



Simple CLI

weka.classifiers.bayes.BayesNet

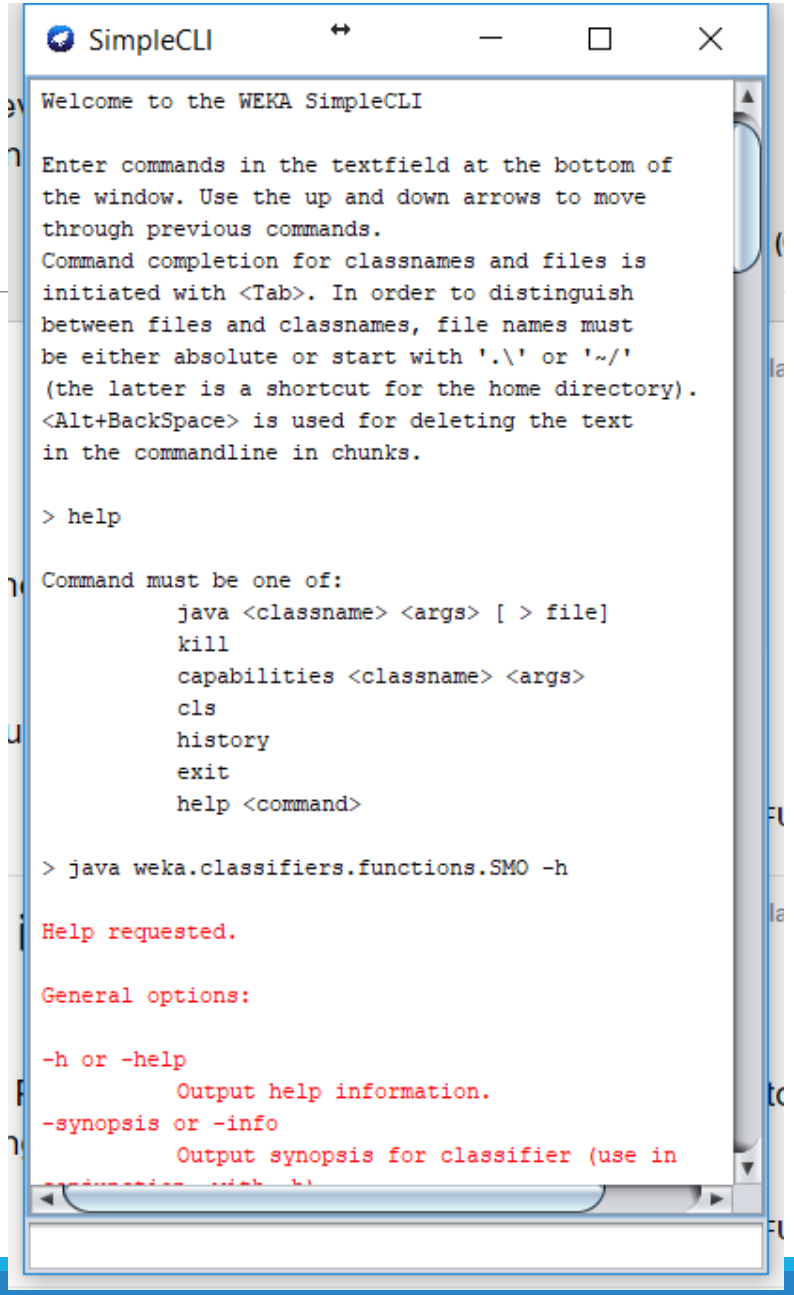
weka.classifiers.trees.J48

weka.classifiers.bayes.NaiveBayesMultinomial

weka.classifiers.RandomizableClassifier

weka.classifiers.functions.LinearRegression

weka.classifiers.trees.RandomTree



Esempio pratico