



UNIVERSITÀ DEGLI STUDI “Aldo Moro”

DIPARTIMENTO DI INFORMATICA

CORSO DI LAUREA IN INFORMATICA MAGISTRALE

**Progetto di Sperimentazione
Intelligenza Artificiale**

*Utilizzo di WEKA per confrontare
performance di due Classificatori:
J48 (algoritmo C4.5) e JRIP(algoritmo RIPPER)*

Professore:

N. Di Mauro

F. Esposito

Studente:

Alessandro Balestrucci

617937

ANNO ACCADEMICO 2014-15

Capitolo 1.

I Sistemi di Apprendimento.....	5
1.1 Introduzione.....	6
1.2 Sistema di Apprendimento Automatico	6
1.2.1 Apprendimento supervisionato.....	6
1.2.2 Apprendimento non supervisionato.....	7
1.3 Software per l'apprendimento	7

Capitolo 2.

Algoritmi di M.L. Dataset ed Addestramento	9
2.1 Introduzione.....	10
2.2 Alberi decisionali (algoritmo J48).....	10
2.3 Regole di decisione (algoritmo JRIP)	11
2.4 Dataset	12
2.4.1 Iris.....	12
2.4.2 CMC	12
2.4.3 Wine	13
2.5 K-Fold Cross-Validation	14

Capitolo 3.

Sperimentazione.....	15
3.1 Introduzione.....	16
3.2 Pre-processing	16
3.3 Sperimentazione Iris.....	18
3.3.1 con J48.....	18
3.3.2 con JRIP	19
3.4 Sperimentazione CMC	20
3.4.1 con J48.....	20
3.4.2 con JRIP	21
3.5 Sperimentazione WINE.....	23
3.5.1 con J48.....	23
3.5.2 con JRIP	24
3.6 Tabelle comparative per Fold.....	26

3.7	Tabella comparativa generale.....	31
<i>Capitolo 4.</i>		
Ottimizzazione Parametri.....		32
4.1	Introduzione.....	33
4.2	CVParameterSelection	33
4.3	GridSearch.....	36
4.4	MultiSearch	38
4.5	Sperimentazione Iris CVParameterSelection	39
4.5.1	J48	39
4.5.2	JRIP	42
4.6	Sperimentazione CMC CVParameterSelection.....	45
4.6.1	J48	45
4.6.2	JRIP	49
4.7	Sperimentazione WINE CVParameterSelection	52
4.7.1	J48	52
4.7.2	JRIP	55
4.8	Tabella di confronto e considerazioni operative per l'esperimento	58
<i>Capitolo 5.</i>		
Confronto e Conclusioni		61
5.1	Introduzione.....	62
5.2	Test delle ipotesi.....	62
5.3	Experimenter	62
5.4	Risultati del Test.....	64
5.5	Conclusioni.....	65
Bibliografia.....		66

Premessa

Questo lavoro ha l'obiettivo di descrivere il processo di sperimentazione per la comparazione di sistemi di apprendimento sfruttando un software molto utilizzato in ambito scientifico, didattico, ma anche, in ambito applicativo.

In questo lavoro si utilizzerà il software WEKA che presenta un'architettura ed una interfaccia facilmente comprensibili, scritto in JAVA.

Per quanto riguarda la sperimentazione del sistema di apprendimento, saranno testati due dei diversi algoritmi direttamente implementati nel software, quali: J48 e JRIP.

La sperimentazione sarà condotta su tre dataset che saranno descritti nei prossimi capitoli.

I risultati della sperimentazione saranno opportunamente comparati, utilizzando dei test statistici, per poter trarre delle conclusioni.

Capitolo 1.

I Sistemi di Apprendimento

1.1 Introduzione

In questo capitolo saranno brevemente introdotti i concetti fondamentali relativi ai Sistemi di Apprendimento Automatico o Machine Learning, ovvero, la branca dell'Intelligenza Artificiale che, si occupa della teoria e delle tecniche di acquisizione della conoscenza, della sua organizzazione e dell'utilizzo per la scoperta di nuovi fatti e teorie in maniera automatica.

1.2 Sistema di Apprendimento Automatico

Un Sistema di Apprendimento Automatico è “*un dispositivo artificiale che ha abilità a migliorare le sue prestazioni basandosi sul suo funzionamento passato*”.

Utilizzando un'altra definizione sui “programmi che apprendono”: *un programma apprende dall'esperienza E, rispetto ad una classe di problemi T e alla misura di performance P, se la sua performance sui problemi in T, così come misurata da P migliora con l'esperienza in E.*

Pertanto, la capacità di apprendimento è fondamentale in un Sistema Intelligente perché da questa capacità dipendono le seguenti abilità a:

- migliorare la risoluzione di un problema (maggiore efficienza);
- non ripetere gli errori fatti in passato;
- adattarsi ai cambiamenti dell'ambiente autonomamente, senza ricorrere alla conoscenza fornita dall'esperto del dominio;
- risolvere nuovi problemi.

Queste capacità portano il Sistema al cambiamento, (subisce una modifica della sua base di conoscenza che viene via via arricchita), al miglioramento delle sue azioni e alla loro generalizzazione (inferisce regole generali che possono essere adottate anche a casi non osservati precedentemente).

1.2.1 Apprendimento supervisionato

Una delle due grandi categorie di apprendimento è quello supervisionato (guidato da esempi) che si basa sull'osservazione di esempi che possono essere positivi e/o negativi (*training set*) relativi ad un certo concetto/classe/categoria da apprendere, da questi, si inducono regole più generali. Cioè, si cerca un classificatore in grado di predire se uno o più dati/esempi, non ancora osservati, appartengano o meno ad una certa classe. A questa categoria appartengono algoritmi

che, generalmente, hanno buone prestazioni però sono costosi a causa della loro necessita di addestramento.

1.2.2 Apprendimento non supervisionato

L'altra grande categoria è rappresentata dall'apprendimento non supervisionato (guidato dalle ipotesi) che ha l'obiettivo di formulare un modello per predire come osservazioni future appartengano o meno ad un certo concetto/classe/categoria. Pertanto, gli esempi (training set), non classificati a priori, saranno classificati ed organizzati sulla base di caratteristiche comuni per cercare di effettuare ragionamenti e previsioni sugli input successivi. L'obiettivo è raggruppare al meglio le osservazioni nei concetti/classi/categorie.

Gli algoritmi che appartengono a questa categoria sono meno costosi e molto efficienti grazie al forte uso della statistica a patto che i dati siano numerici.

1.3 Software per l'apprendimento

Il software utilizzato per l'apprendimento e la sperimentazione è WEKA (Waikato Environment for Knowledge Analysis). Nato come punto di riferimento del Machine Learning è anche largamente impiegato per il Data Mining, utilizza un ambiente grafico, è scritto in Java, pertanto, è multiplatforma ed è distribuito sotto la GNU Public License. La versione a cui si fa riferimento è l'ultima stabile cioè la 3.6.12, scaricata direttamente dal sito dell'Università del Waikato (NZ) <http://www.cs.waikato.ac.nz/ml/weka/downloading.html>.

Mentre la versione DEVELOPER è la 3.7.12

WEKA grazie alla sua interfaccia grafica molto intuitiva permette di effettuare diverse operazioni raggruppate in una serie di interfacce.

Explorer permette le seguenti attività:

- Preprocess: pre-processing dei dati tramite caricamento e modifica per le varie applicazioni;
- Classify: permette di effettuare e valutare operazioni di classificazione e regressione di dati utilizzando appositi algoritmi di apprendimento direttamente implementati nel software;
- Cluster: sezione dedicata all'applicazione degli algoritmi di clustering;

-
- Associate: elaborazione e valutazione di regole di associazione;
 - Select attributes: tecniche di selezione degli aspetti più rilevanti di un insieme di dati;
 - Visualize: visualizzazione di diversi grafici bidimensionali sui dati in elaborazione.

Esperimenter è l'altra interfaccia utile per l'addestramento e test con l'utilizzo di test statistici per il confronto dei dati ottenuti dai diversi algoritmi di Machine Learning o Data Mining.

Nello specifico, saranno presi in considerazione due algoritmi per l'apprendimento il j48 ed il jrip.

Capitolo 2.

Algoritmi di M.L. Dataset ed Addestramento

2.1 Introduzione

In questo capitolo sarà riportata una breve descrizione delle prime due categorie di algoritmi esistenti in letteratura ed utilizzati per le tecniche di classificazione nel Machine Learning, cioè:

- Alberi decisionali o Decision Tree;
- Regole di decisione;
- Nearest-neighbor;
- Reti Bayesiane;
- Reti neurali;
- Support Vector Machines.

Per la sperimentazione seguente è stato scelto di mettere a confronto due algoritmi di categorie differenti:

- Alberi decisionali tramite l'algoritmo j48;
- Regole di decisione tramite l'algoritmo jrip.

La tecnica di addestramento, validazione e test che sarà utilizzata per poter mettere a confronto gli algoritmi sarà la K-Fold Cross Validation. Questa tecnica statistica prevede un procedimento per evitare che l'addestramento e quindi i risultati ottenuti, possano essere falsati dalla scelta del training set e del test set.

Infine, sarà riportata una descrizione dei dataset utilizzati per la sperimentazione, prelevati dal sito UCI Machine Learning Repository <http://archive.ics.uci.edu/ml/index.html>.

2.2 Alberi decisionali (algoritmo J48)

È una delle tecniche di classificazione maggiormente utilizzate che permette di rappresentare con un albero un insieme di regole di classificazione.

Struttura gerarchica che consiste di un insieme di nodi, correlati da archi (rami) orientati ed etichettati. Si hanno due tipi di nodi:

- i nodi foglia che definiscono le classi
- i rimanenti nodi che sono etichettati in base all'attributo che partiziona i record.

Il criterio di partizionamento rappresenta l'etichetta degli archi.

L'algoritmo J48 è una implementazione open source dell'algoritmo C4.5 (Quinlan 1993) che a sua volta è una estensione del più generale algoritmo ID3 (Quinlan 1986).

C4.5 è uno degli algoritmi più diffusi per la costruzione di classificatori mediante alberi di decisione. Si basa sostanzialmente sull'algoritmo CART (Breiman et al. 1984), quest'ultimo si articola in due fasi principali. La generazione dell'albero completo e il pruning dell'albero. Un albero completo, infatti, è spesso costituito da un numero molto elevato di regole estremamente complesse. Per questo motivo, affinché l'albero possa essere effettivamente utile nel classificare e fornire regole efficaci, è necessario sfoltire la ridondanza dell'albero, eliminando i rami meno significativi.

Le differenze sostanziali tra C4.5 rispetto a CART sono:

- L'albero non è binario. Da ogni nodo partono un numero di rami pari a tutti i possibili valori che possono essere assunti dal campo *splitter* per quel nodo.
- Il criterio utilizzato per decidere quale campo utilizzare come splitter in ogni nodo si basa sul concetto di *information gain*.
- La procedura di pruning si basa sull'attribuzione di un errore previsto, ad ogni foglia. In pratica questo errore previsto viene assegnato ipotizzando che l'errore commesso sul training set in corrispondenza di una data foglia, sia il massimo che si può ottenere.

2.3 Regole di decisione (algoritmo JRIP)

Questa tecnica (Rule Induction) mira a costruire un modello mediante l'identificazione di un insieme di regole del tipo “if ... then ...”, ovvero, (*condizione*) → *y* dove *condizione* è una congiunzione di attributi ed *y* è l'etichetta di classe.

Il sistema restituisce come risultato dell'apprendimento un insieme di regole riguardanti i casi particolari, ed una regola di default vera per le classi rimanenti. Ciascuna di queste regole prodotte da Ripper ha una determinata informazione relative alla “confidenza”: il numero di esempi abbinati (cioè quelle istanze che si conformano alla regola) ed il numero di esempi non abbinati (le istanze che sono in conflitto con la regola) all'interno dei dati di training.

L'algoritmo JRIP è una implementazione open source dell'algoritmo RIPPER (Cohen and Singer, 1999).

2.4 Dataset

Sono stati scelti 3 dataset diversi tra loro per numerosità del campione, numero e tipologia degli attributi. Tutti i dati sono classificati. Sono stati prelevati dalla UCI dataset Repository, preferendo la sezione MOST POPULAR

2.4.1 Iris

Famoso dataset per la classificazione di tre varietà dell'omonimo fiore.

Il dataset presenta 4 attributi numerici relativi alle misure del fiore:

1. Lunghezza in cm del sepalo;
2. Larghezza in cm del sepalo;
3. Lunghezza in cm del petalo;
4. Larghezza in cm del petalo;

e 3 classi di appartenenza:

1. Iris Setosa;
2. Iris Versicolour;
3. Iris Virginica.

Delle 150 istanze presenti nel dataset, 50 appartengono alla varietà Iris-Setosa, 50 all'Iris-Versicolour e 50 all' Iris-Virginica.

2.4.2 CMC

Il dataset Contraceptive Method Choice contiene interviste ad un campione di 1.473 donne sposate che non sono o non sanno di essere incinta all'atto del colloquio. Lo scopo è la predizione del metodo contraccettivo scelto basandosi sulle sue condizioni socio-economiche e sulle sue caratteristiche demografiche.

Sono presenti 9 attributi molti dei quali relativi a classi di appartenenza:

1. Età della moglie;
2. Istruzione della moglie;
3. Istruzione del marito;

-
4. Numero di aborti;
 5. Moglie di Religione islamica;
 6. Moglie lavoratrice;
 7. Occupazione del marito;
 8. Indice standard di tenore di vita;
 9. Esposizione mediatica;

e 3 classi di metodi contraccettivi utilizzati (Nessuno – Lungo termine – Breve termine).

La distribuzione delle classi di appartenenza in questo dataset è differente dal precedente, in particolare:

- 629 (1) Nessun metodo;
- 333 (2) Lungo termine;
- 511 (3) Breve termine.

2.4.3 Wine

Dataset contenente il risultato di un'analisi chimica dei vini di produzione nella stessa Regione italiana, ma derivate da tre diverse piantagioni. L'analisi ha determinato su 178 istanze i quantitativi di 13 elementi presenti in ciascuno dei tre tipi o categorie (1-2-3) di vini.

Pertanto, troveremo 13 attributi tutti numerici:

1. Alcol;
2. Acido malico;
3. Ceneri (Ash);
4. Alcalinità delle ceneri;
5. Magnesio;
6. Fenoli totali;
7. Flavonoidi;
8. Fenoli non flavonoidi;
9. Tannino (proantocianidine);

-
10. Colorazione;
 11. Tonalità;
 12. Proteine diluite;
 13. Prolina.

In questo dataset delle 178 istanze di vino selezionato, 58 appartengono alla Categoria 1, 71 alla Categoria 2 e 48 alla Categoria 3.

2.5 K-Fold Cross-Validation

Gli schemi di apprendimento, di solito, operano in due passi per costruire le strutture basilari e ottimizzare le impostazioni dei parametri. Le procedure usano tre insiemi scelti indipendentemente, training data, validation data e test data.

Avendo a disposizione un insieme di dati limitato, si adotta la procedura Holdout che è un metodo di splitting dei dati originali in dati di training e di testing. Per quanto riguarda la stratificazione ci affidiamo al software Weka che effettua automaticamente la suddivisione del dataset.

Per l'addestramento si è scelto di utilizzare il K-Fold Cross Validation tecnica che prevede la suddivisione di ogni dataset in K parti uguali (K può essere a piacere). In questo esperimento si usa il **ten-fold cross-validation** in quanto, esperimenti estensivi (e anche teorie) hanno dimostrato che dieci è il numero di migliore di fold per una stima accurata.

La tecnica prevede 10 iterazioni dove ogni volta si utilizza una parte di dataset (9/10) per l'apprendimento, allenamento dell'algoritmo ed 1/10 del dataset per il test. A rotazione per tutte le K parti. Alla fine di ogni iterazione si raccolgono i risultati dell'operazione per i calcoli e le comparazioni finali dei due algoritmi.

Le operazioni precedenti saranno effettuate per tutti i dataset e per entrambi gli algoritmi.

I dati finali saranno opportunamente comparati.

Capitolo 3.

Sperimentazione

3.1 Introduzione

In questo capitolo saranno presentati i dati generali ottenuti dalla sperimentazione condotta sui due algoritmi J48 e JRIP e sui tre dataset IRIS, CMC, WINE già descritti.

Si utilizzerà il software WEKA per condurre la preparazione dei dati, per le sperimentazioni e la raccolta dei risultati dell'apprendimento.

I tempi espressi in secondi varieranno anche per la pro cessazione dello stesso dataset per il semplice motivo che qualche volta si ha avuto la possibilità di poter usare una Macchina molto potente (non propria), altre volte invece si è utilizzato un pentium 4.

3.2 Pre-processing

I dataset scaricati dal sito UCI sono stati trasformati in formato ARFF (uno dei formati compatibili di Weka).

Il formato ARFF prevede che in un file di testo vengano memorizzate le informazioni con in testa il nome della relazione (@Relation <nome>) e degli attributi con relativa tipologia (@Attribute <nome> <Tipo>) seguiti da una lista dei dati (@Data).

Esempio:

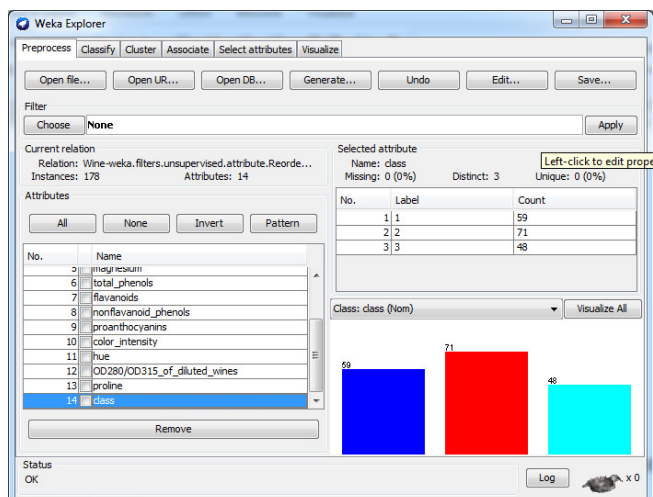
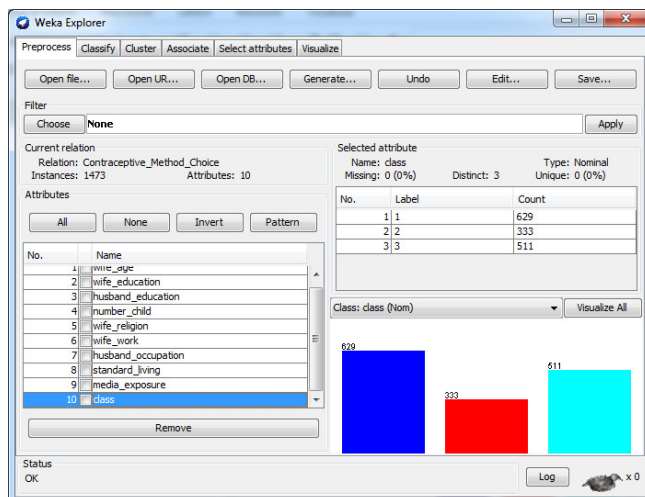
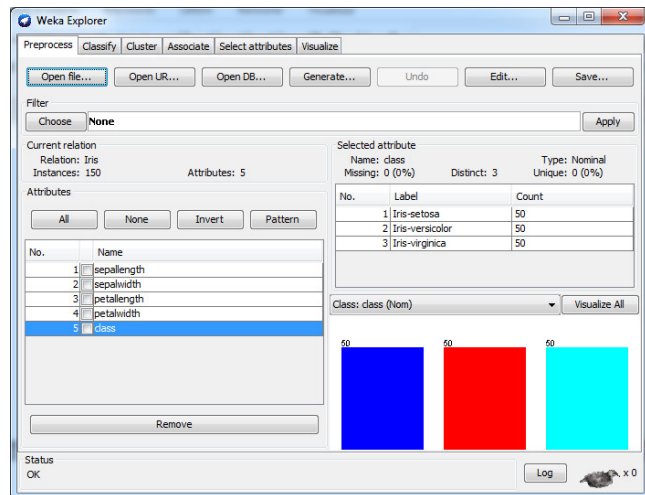
```
@RELATION Iris
@ATTRIBUTE sepallength NUMERIC
@ATTRIBUTE sepalwidth NUMERIC
@ATTRIBUTE petallength NUMERIC
@ATTRIBUTE petalwidth NUMERIC
@ATTRIBUTE class {Iris-setosa,Iris-versicolor,Iris-virginica}

@DATA
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
...
```

Questa operazione viene effettuata per tutti e tre dataset.

Tramite l'interfaccia Explorer si possono aprire e visualizzare i dataset trasformati, come ulteriore conferma della validità dell'operazione appena effettuata.

Come si può vedere dai risultati seguenti:

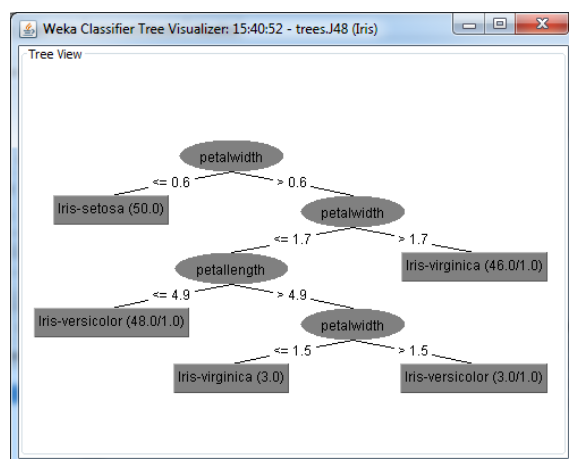


Le tre schermate precedenti confermano quanto riportato nella descrizione dei dataset.

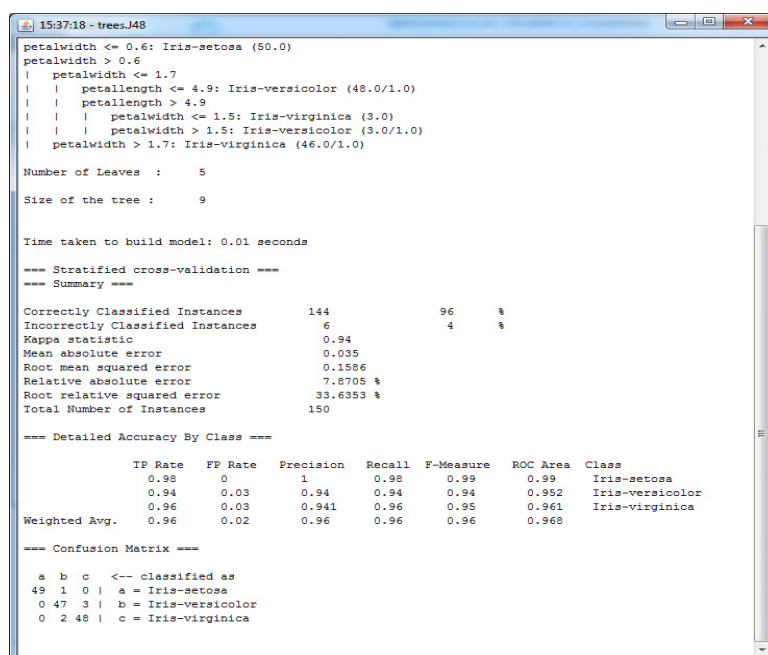
3.3 Sperimentazione Iris

3.3.1 con J48

Nella sperimentazione condotta con J48 sul dataset Iris, applicando la tecnica K-Fold Cross Validation, in un tempo di 0,01 secondi, è stato ottenuto il seguente albero formato da 5 nodi foglia e 4 nodi di partizionamento:



La seguente videata riporta tutte le informazioni generali ottenibili dall'esecuzione della classificazione con Weka – Explorer:



3.3.2 con JRIP

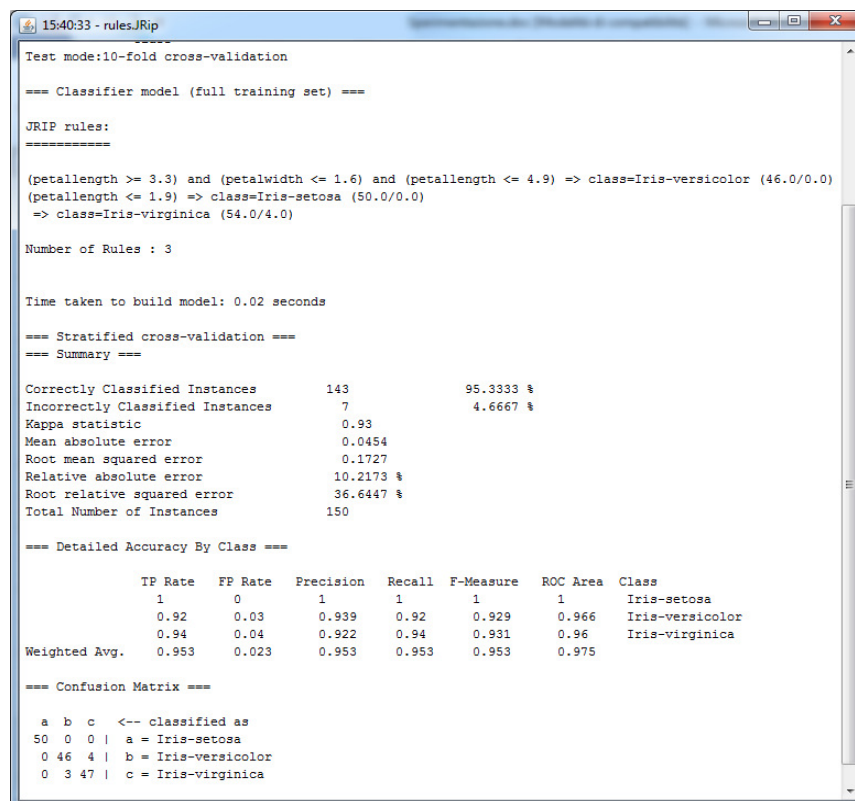
Nella sperimentazione condotta con JRIP, in un tempo di 0,02 secondi, sono state ottenute le seguenti 3 regole:

(petallength >= 3.3) and (petalwidth <= 1.6) and (petallength <= 4.9) → class=Iris-versicolor (46.0/0.0)

(petallength <= 1.9) → class=Iris-setosa (50.0/0.0)

→ class=Iris-virginica (54.0/4.0)

La seguente videata riporta tutte le informazioni generali ottenibili dall'esecuzione della classificazione con Weka – Explorer:



```
15:40:33 - rules.JRip
Test mode:10-fold cross-validation

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

JRIP rules:
=====
(petallength >= 3.3) and (petalwidth <= 1.6) and (petallength <= 4.9) => class=Iris-versicolor (46.0/0.0)
(petallength <= 1.9) => class=Iris-setosa (50.0/0.0)
=> class=Iris-virginica (54.0/4.0)

Number of Rules : 3

Time taken to build model: 0.02 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      143           95.3333 %
Incorrectly Classified Instances     7             4.6667 %
Kappa statistic                     0.93
Mean absolute error                  0.0454
Root mean squared error              0.1727
Relative absolute error              10.2173 %
Root relative squared error          36.6447 %
Total Number of Instances           150

=== Detailed Accuracy By Class ===

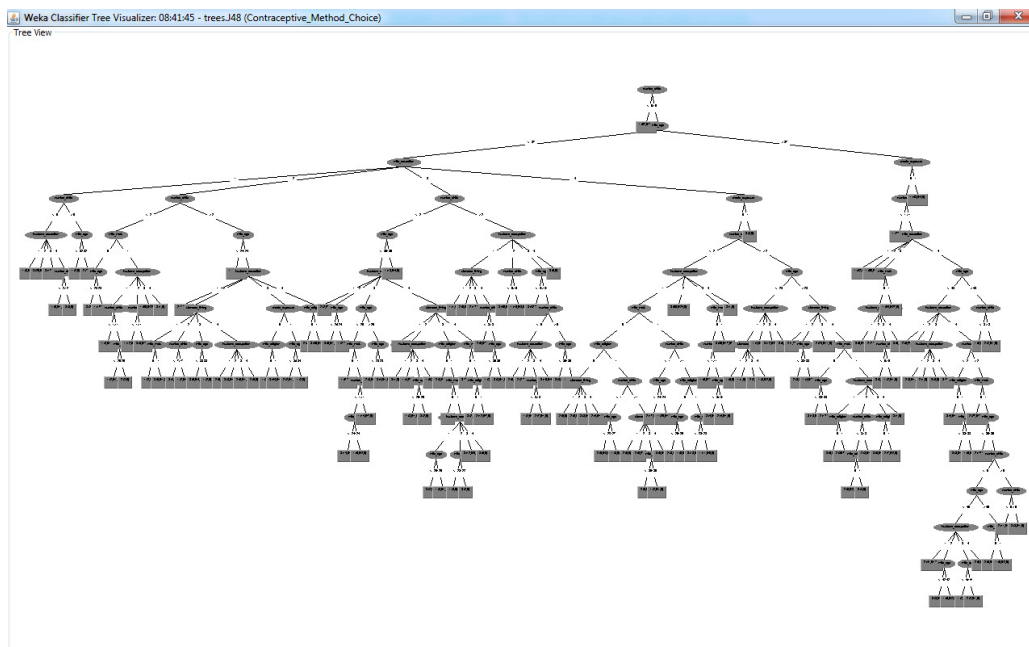
      TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
      1         0         1           1         1           1       Iris-setosa
      0.92      0.03      0.939      0.92      0.929      0.966     Iris-versicolor
      0.94      0.04      0.922      0.94      0.931      0.96     Iris-virginica
Weighted Avg.   0.953      0.023      0.953      0.953      0.953      0.975

=== Confusion Matrix ===
  a  b  c  <-- classified as
50  0  0 | a = Iris-setosa
 0 46  4 | b = Iris-versicolor
 0  3 47 | c = Iris-virginica
```

3.4 Sperimentazione CMC

3.4.1 con J48

Nella sperimentazione condotta con J48, in un tempo di 0,05 secondi, sul dataset Contraceptive Method Choice applicando sempre la tecnica K-Fold Cross Validation, è stato ottenuto un albero formato da 157 nodi foglia e 106 nodi di partizionamento.



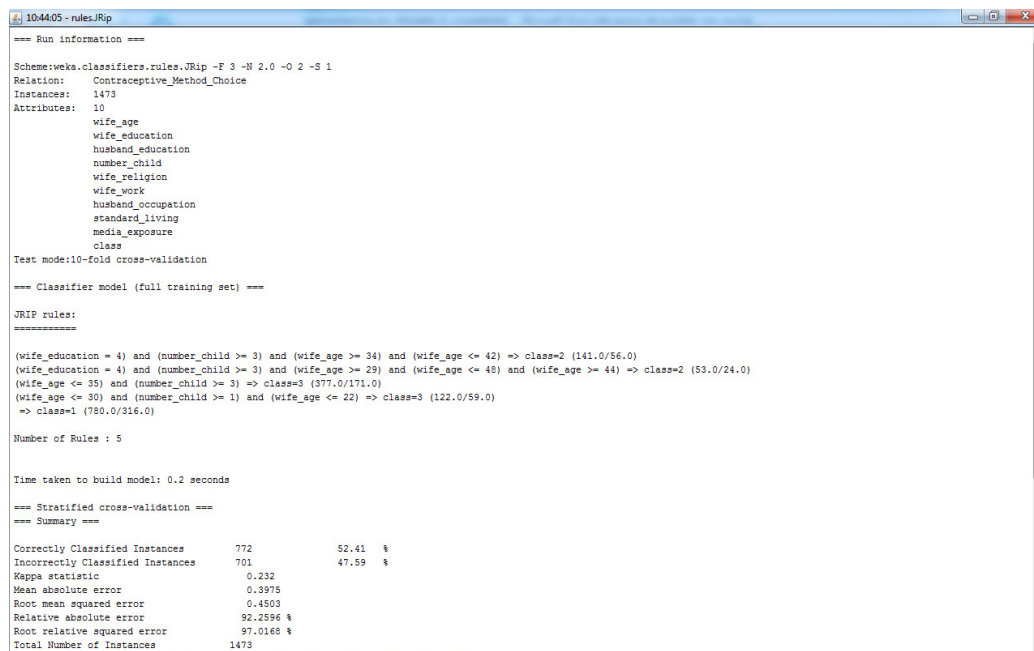
Viene riportata di seguito la rappresentazione grafica poco comprensibile data la dimensione dell'albero: Le seguenti videate riportano tutte le informazioni generali ottenibili dall'esecuzione della classificazione con Weka – Explorer

(wife_age <= 35) and (number_child >= 3) → class=3 (377.0/171.0)

(wife_age <= 30) and (number_child >= 1) and (wife_age <= 22) → class=3 (122.0/59.0)

→ class=1 (780.0/326.0)

Le seguenti videate riportano tutte le informazioni generali ottenibili dall'esecuzione della classificazione con Weka – Explorer:



```
10:44:05 - rules.JRip
=== Run information ===

Scheme: weka.classifiers.rules.JRip -F 3 -N 2.0 -O 2 -S 1
Relation: Contraceptive_Method_Choice
Instances: 1473
Attributes: 10
    wife_age
    wife_education
    husband_education
    number_child
    wife_religion
    wife_work
    husband_occupation
    standard_living
    media_exposure
    class

Test mode: 10-fold cross-validation

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

JRIP rules:
=====

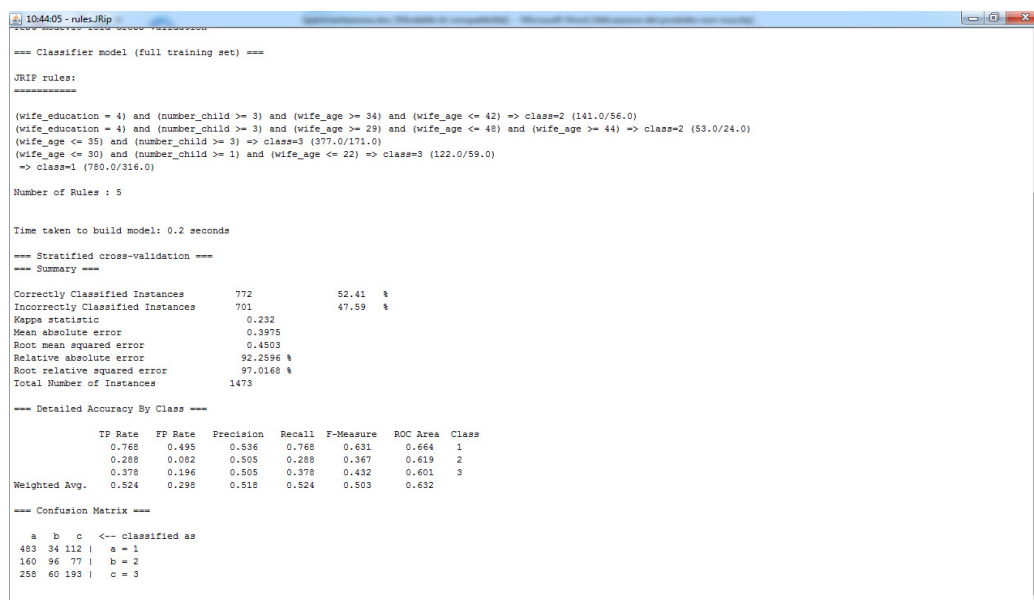
(wife_education = 4) and (number_child >= 3) and (wife_age >= 34) and (wife_age <= 42) => class=2 (141.0/56.0)
(wife_education = 4) and (number_child >= 3) and (wife_age >= 29) and (wife_age <= 48) and (wife_age >= 44) => class=2 (53.0/24.0)
(wife_age <= 35) and (number_child >= 3) => class=3 (377.0/171.0)
(wife_age <= 30) and (number_child >= 1) and (wife_age <= 22) => class=3 (122.0/59.0)
=> class=1 (780.0/316.0)

Number of Rules : 5

Time taken to build model: 0.2 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      772           52.41 %
Incorrectly Classified Instances    701           47.59 %
Kappa statistic                    0.232
Mean absolute error                0.3975
Root mean squared error            0.4503
Relative absolute error            92.2596 %
Root relative squared error        97.0168 %
Total Number of Instances         1473
```



```
10:44:05 - rules.JRip
=== Classifier model (full training set) ===

JRIP rules:
=====

(wife_education = 4) and (number_child >= 3) and (wife_age >= 34) and (wife_age <= 42) => class=2 (141.0/56.0)
(wife_education = 4) and (number_child >= 3) and (wife_age >= 29) and (wife_age <= 48) and (wife_age >= 44) => class=2 (53.0/24.0)
(wife_age <= 35) and (number_child >= 3) => class=3 (377.0/171.0)
(wife_age <= 30) and (number_child >= 1) and (wife_age <= 22) => class=3 (122.0/59.0)
=> class=1 (780.0/316.0)

Number of Rules : 5

Time taken to build model: 0.2 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      772           52.41 %
Incorrectly Classified Instances    701           47.59 %
Kappa statistic                    0.232
Mean absolute error                0.3975
Root mean squared error            0.4503
Relative absolute error            92.2596 %
Root relative squared error        97.0168 %
Total Number of Instances         1473

=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
      0.768    0.495    0.536    0.768    0.631    0.664    1
      0.288    0.082    0.505    0.288    0.367    0.619    2
      0.378    0.196    0.505    0.378    0.432    0.601    3
Weighted Avg.    0.524    0.298    0.518    0.524    0.503    0.632

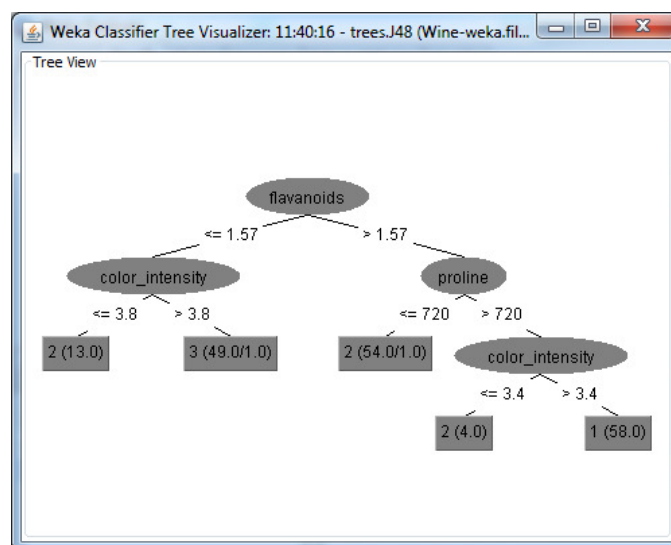
=== Confusion Matrix ===

  a  b  c  <-- classified as
483 34 112 |  a = 1
160 96 77 |  b = 2
258 60 193 |  c = 3
```

3.5 Sperimentazione WINE

3.5.1 con J48

Nella sperimentazione condotta con J48 sul dataset Wine applicando sempre la tecnica K-Fold Cross Validation, in un tempo di 0,06 secondi, è stato ottenuto il seguente albero formato da 5 nodi foglia e 4 nodi di partizionamento:



Le seguenti videate riportano tutte le informazioni generali ottenibili dall'esecuzione della classificazione con Weka – Explorer:

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation: Wine-weka.filters.unsupervised.attribute.Reorder-R2,3,4,5,6,7,8,9,10,11,12,13,14,1
Instances: 178
Attributes: 14
    alcohol
    malic_acid
    ash
    alcalinity_of_ash
    magnesium
    total_phenols
    flavanoids
    nonflavanoid_phenols
    proanthocyanins
    color_intensity
    hue
    OD280/OD315_of_diluted_wines
    proline
    class

Test mode:10-fold cross-validation

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

J48 pruned tree
-----

flavanoids <= 1.57
```

```

11:40:16 - trees.J48
|   color_intensity <= 3.8: 2 (13.0)
|   color_intensity > 3.8: 3 (49.0/1.0)
flavanoids > 1.57
|   proline <= 720: 2 (54.0/1.0)
|   proline > 720
|   |   color_intensity <= 3.4: 2 (4.0)
|   |   color_intensity > 3.4: 1 (58.0)

Number of Leaves :    5
Size of the tree :    9

Time taken to build model: 0.06 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      167           93.8202 %
Incorrectly Classified Instances    11           6.1798 %
Kappa statistic                    0.9058
Mean absolute error                 0.0486
Root mean squared error            0.2019
Relative absolute error            11.0723 %
Root relative squared error        43.0865 %
Total Number of Instances          178

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
               0.983    0.034    0.935    0.983    0.959    0.977    1
               0.944    0.056    0.918    0.944    0.931    0.937    2
               0.875    0.008    0.977    0.875    0.923    0.946    3
Weighted Avg.   0.938    0.036    0.94    0.938    0.938    0.953

=== Confusion Matrix ===

  a  b  c  <-- classified as
58  1  0 |  a = 1
 3 67  1 |  b = 2
 1  5 42 |  c = 3

```

3.5.2 con JRIP

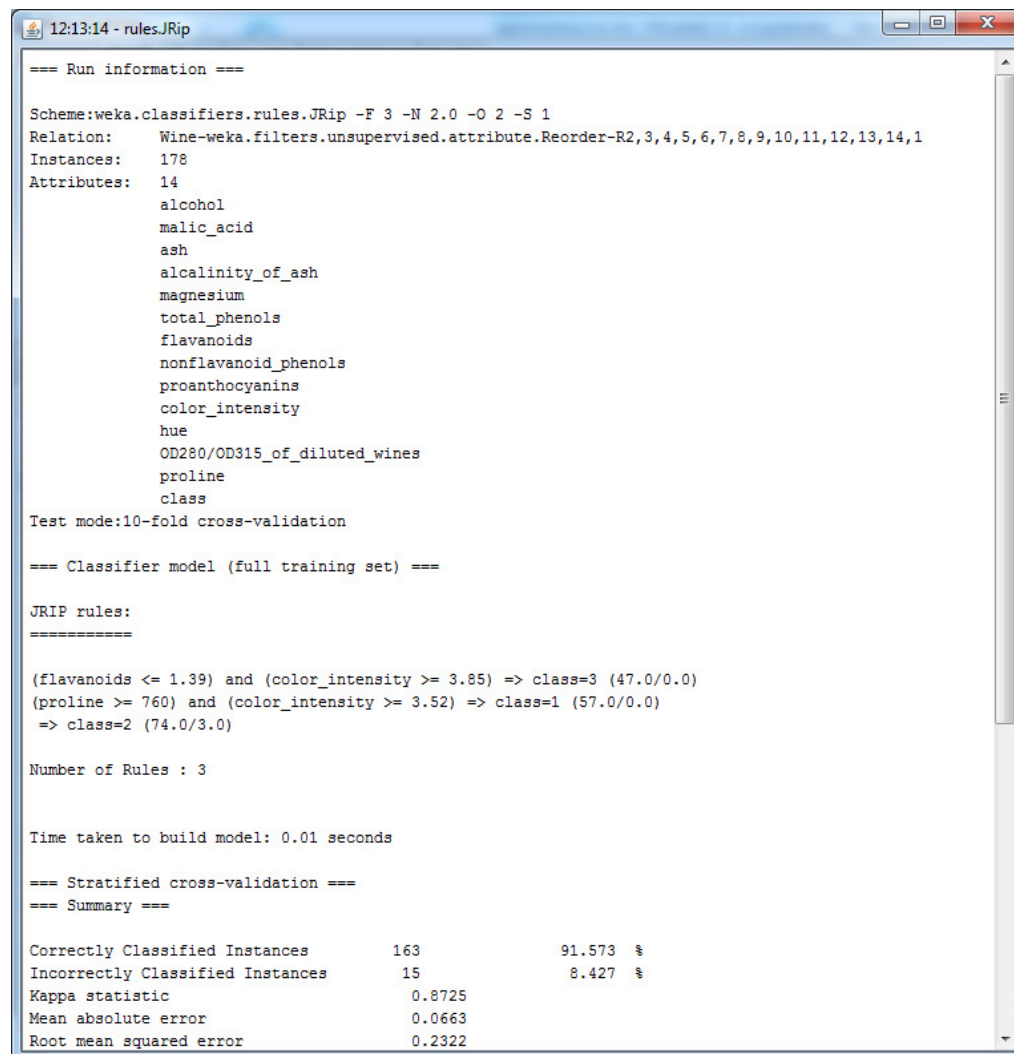
Nella sperimentazione condotta con JRIP, in un tempo di 0,01 secondi, sono state ottenute le seguenti 3 regole:

(flavanoids <= 1.39) and (color_intensity >= 3.85) → class=3 (47.0/0.0)

(proline >= 760) and (color_intensity >= 3.52) → class=1 (57.0/0.0)

→ class=2 (74.0/3.0)

Le seguenti videate riportano tutte le informazioni generali ottenibili dall'esecuzione della classificazione con Weka – Explorer:



```
12:13:14 - rules.JRip

=== Run information ===

Scheme:weka.classifiers.rules.JRip -F 3 -N 2.0 -O 2 -S 1
Relation:      Wine-weka.filters.unsupervised.attribute.Reorder-R2,3,4,5,6,7,8,9,10,11,12,13,14,1
Instances:      178
Attributes:     14
                alcohol
                malic_acid
                ash
                alcalinity_of_ash
                magnesium
                total_phenols
                flavanoids
                nonflavanoid_phenols
                proanthocyanins
                color_intensity
                hue
                OD280/OD315_of_diluted_wines
                proline
                class
Test mode:10-fold cross-validation

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

JRIP rules:
=====

(flavanoids <= 1.39) and (color_intensity >= 3.85) => class=3 (47.0/0.0)
(proline >= 760) and (color_intensity >= 3.52) => class=1 (57.0/0.0)
=> class=2 (74.0/3.0)

Number of Rules : 3

Time taken to build model: 0.01 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      163           91.573 %
Incorrectly Classified Instances    15            8.427 %
Kappa statistic                    0.8725
Mean absolute error                 0.0663
Root mean squared error             0.2322
```

```
12:13:14 - rulesJRip
Test mode:10-fold cross-validation

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

JRIP rules:
=====

(flavanoids <= 1.39) and (color_intensity >= 3.85) => class=3 (47.0/0.0)
(proline >= 760) and (color_intensity >= 3.52) => class=1 (57.0/0.0)
=> class=2 (74.0/3.0)

Number of Rules : 3

Time taken to build model: 0.01 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      163           91.573 %
Incorrectly Classified Instances    15            8.427 %
Kappa statistic                     0.8725
Mean absolute error                 0.0663
Root mean squared error             0.2322
Relative absolute error             15.0955 %
Root relative squared error         49.5519 %
Total Number of Instances          178

=== Detailed Accuracy By Class ===

                TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
                0.915    0.05    0.9        0.915   0.908      0.96      1
                0.873    0.047   0.925     0.873   0.899      0.938     2
                0.979    0.031   0.922     0.979   0.949      0.973     3
Weighted Avg.   0.916    0.044   0.916     0.916   0.915      0.955

=== Confusion Matrix ===

  a  b  c  <-- classified as
54  5  0 | a = 1
 5 62  4 | b = 2
 1  0 47 | c = 3
```

3.6 Tabelle comparative per Fold

I dati delle sei tabelle riassuntive riportate di seguito, sono relative alle elaborazioni di ogni fold, dei tre dataset e per ogni algoritmo che Weka – Experimenter effettua. Questo aspetto della sperimentazione sarà meglio riportato nel capitolo successivo.

I diversi attributi generati dal software sono memorizzabili in file ARFF che contengono istanze per ogni fold creato durante l’elaborazione. Detti files sono presenti nella sottocartella Weka del Cd allegato al progetto. Un file per ogni dataset ed uno che contiene tutti i dati dell’intera sperimentazione.

I dati di seguito riportati, opportunamente, filtrati sono stati estrapolati per essere inseriti nelle tabelle riassuntive, essi si riferiscono ad una sola esecuzione dell’esperimento per ogni fold.

Di seguito viene riportata, a titolo di esempio, la videata relativa solo all'elaborazione del dataset Iris:

No.	Key_Fold	Key_Scheme	Percent_correct	Percent_incorrect	Percent_undclassified	IR_precision	IR_recall	F_measure
	Nominal	Nominal	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric
1	1	weka.classifiers.trees.J48	93.333333	6.666667	0.0	1.0	1.0	1.0
2	2	weka.classifiers.trees.J48	100.0	0.0	0.0	1.0	1.0	1.0
3	3	weka.classifiers.trees.J48	100.0	0.0	0.0	1.0	1.0	1.0
4	4	weka.classifiers.trees.J48	100.0	0.0	0.0	1.0	1.0	1.0
5	5	weka.classifiers.trees.J48	93.333333	6.666667	0.0	1.0	1.0	1.0
6	6	weka.classifiers.trees.J48	100.0	0.0	0.0	1.0	1.0	1.0
7	7	weka.classifiers.trees.J48	86.666667	13.333333	0.0	1.0	0.8	0.888889
8	8	weka.classifiers.trees.J48	86.666667	13.333333	0.0	1.0	1.0	1.0
9	9	weka.classifiers.trees.J48	100.0	0.0	0.0	1.0	1.0	1.0
10	10	weka.classifiers.trees.J48	100.0	0.0	0.0	1.0	1.0	1.0
11	1	weka.classifiers.rules.JRip	93.333333	6.666667	0.0	1.0	1.0	1.0
12	2	weka.classifiers.rules.JRip	100.0	0.0	0.0	1.0	1.0	1.0
13	3	weka.classifiers.rules.JRip	100.0	0.0	0.0	1.0	1.0	1.0
14	4	weka.classifiers.rules.JRip	100.0	0.0	0.0	1.0	1.0	1.0
15	5	weka.classifiers.rules.JRip	86.666667	13.333333	0.0	1.0	1.0	1.0
16	6	weka.classifiers.rules.JRip	100.0	0.0	0.0	1.0	1.0	1.0
17	7	weka.classifiers.rules.JRip	93.333333	6.666667	0.0	1.0	1.0	1.0
18	8	weka.classifiers.rules.JRip	93.333333	6.666667	0.0	1.0	1.0	1.0
19	9	weka.classifiers.rules.JRip	86.666667	13.333333	0.0	1.0	1.0	1.0
20	10	weka.classifiers.rules.JRip	100.0	0.0	0.0	1.0	1.0	1.0

Per la comparazione dei risultati dei due algoritmi si è scelto di utilizzare la metrica F-measure che riassume le metriche Precision e Recall. Queste ultime sono molto utilizzate nelle applicazioni in cui la corretta classificazione dei record della classe positiva riveste una maggiore importanza.

- Recall misura la frazione di record positivi correttamente classificati (valori elevati indicano che pochi record della classe negativa sono stati erroneamente classificati come positivi).
- Precision misura la frazione di record risultati effettivamente positivi tra tutti quelli che erano stati classificati come tali (valori elevati indicano che pochi record della classe positiva sono stati erroneamente classificati come negativi).

Classe effettiva	Classe prevista	
	Class=Yes	Class=No
	Class=Yes	Class=No
Class=Yes	TP	FN
Class=No	FP	TN

Per il calcolo di F-measure (f), Precision (p) e Recall (r) si utilizzano i dati della matrice di confusione secondo lo schema a due classi riportato a lato, dove:

- *TP (true positive): sono i record correttamente classificati come classe Yes*
- *FN (false negative): record incorrettamente classificati come classe No*

- *FP (false positive): record incorrettamente classificati come classe Yes*
- *TN (true negative) record correttamente classificati come classe No*

$$p = TP/(TP+FP), r = TP/(TP+FN), f = 2rp/(r+p)$$

Tabelle riassuntive

Le tabelle seguenti riportano una sintesi dei dati per ogni dataset – algoritmo:

Tabella Iris – J48:

N° Fold	% Corretti	% Errati	% non Classificati	Precision	Recall	F-measure
1	93,333	6,667	0	1	1	1
2	100	0	0	1	1	1
3	100	0	0	1	1	1
4	100	0	0	1	1	1
5	93,333	6,667	0	1	1	1
6	100	0	0	1	1	1
7	86,667	13,333	0	0,8	0,8	0,889
8	86,667	13,333	0	0,8	0,8	0,889
9	100	0	0	1	1	1
10	100	0	0	1	1	1
Medie	96	4	0	0,96	0,96	0,96

Tabella Iris – JRIP:

N° Fold	% Corretti	% Errati	% non Classificati	Precision	Recall	F-measure
1	93,333	6,667	0	1	1	1
2	100	0	0	1	1	1
3	100	0	0	1	1	1
4	100	0	0	1	1	1
5	86,667	13,333	0	0,8	0,8	0,889
6	100	0	0	1	1	1
7	93,333	6,667	0	1	1	1
8	93,333	6,667	0	1	1	1
9	86,667	13,333	0	0,8	0,8	0,889
10	100	0	0	1	1	1

<i>Medie</i>	<i>95,333</i>	<i>4,667</i>	<i>0</i>	<i>0,953</i>	<i>0,953</i>	<i>0,953</i>
--------------	---------------	--------------	----------	--------------	--------------	--------------

Tabella CMC – J48:

N° Fold	% Corretti	% Errati	% non Classificati	Precision	Recall	F-measure
<i>1</i>	59,459	40,541	0	0,690	0,635	0,661
<i>2</i>	54,054	45,946	0	0,6233	0,762	0,686
<i>3</i>	55,405	44,595	0	0,652	0,681	0,682
<i>4</i>	47,620	52,380	0	0,521	0,587	0,552
<i>5</i>	50,340	49,660	0	0,64	0,508	0,566
<i>6</i>	51,700	48,300	0	0,683	0,651	0,667
<i>7</i>	50,340	49,660	0	0,623	0,524	0,569
<i>8</i>	53,061	46,939	0	0,587	0,587	0,587
<i>9</i>	51,701	48,299	0	0,565	0,556	0,56
<i>10</i>	47,619	52,381	0	0,565	0,565	0,565
<i>Medie</i>	<i>52,138</i>	<i>47,861</i>	<i>0</i>	<i>0,521</i>	<i>0,521</i>	<i>0,52</i>

Tabella CMC – JRIP:

N° fold	% Corretti	% Errati	% non Classificati	Precision	Recall	F-measure
<i>1</i>	53,378	46,622	0	0,522	0,746	0,614
<i>2</i>	49,324	50,676	0	0,485	0,778	0,598
<i>3</i>	52,703	47,297	0	0,553	0,825	0,662
<i>4</i>	46,939	53,061	0	0,461	0,952	0,628
<i>5</i>	52,381	47,619	0	0,563	0,778	0,653
<i>6</i>	55,102	44,898	0	0,543	0,810	0,610
<i>7</i>	58,503	41,497	0	0,622	0,730	0,672
<i>8</i>	52,381	47,619	0	0,562	0,651	0,603
<i>9</i>	53,741	46,259	0	0,582	0,730	0,648
<i>10</i>	49,660	50,340	0	0,532	0,677	0,596
<i>Medie</i>	<i>52,41</i>	<i>47,59</i>	<i>0</i>	<i>0,518</i>	<i>0,524</i>	<i>0,503</i>

Tabella Wine – J48:

N° fold	% Corretti	% Errati	% non Classificati	Precision	Recall	F-measure
<i>1</i>	83,333	16,667	0	1	0,833	0,909
<i>2</i>	94,444	5,556	0	1	1	1
<i>3</i>	88,889	11,111	0	0,857	1	0,923
<i>4</i>	94,444	5,556	0	0,857	1	0,923
<i>5</i>	100	0	0	1	1	1
<i>6</i>	88,889	11,111	0	0,75	1	0,857
<i>7</i>	100	0	0	1	1	1
<i>8</i>	94,444	5,556	0	1	1	1
<i>9</i>	100	0	0	1	1	1
<i>10</i>	94,118	5,882	0	1	1	1
Medie	93,820	6,168	0	0,94	0,938	0,938

Tabella Wine – JRIP:

N° fold	% Corretti	% Errati	% non Classificati	Precision	Recall	F-measure
<i>1</i>	83,333	16,667	0	1	0,5	0,667
<i>2</i>	94,444	5,556	0	1	1	1
<i>3</i>	94,444	5,556	0	0,857	1	0,923
<i>4</i>	100	0	0	1	1	0,923
<i>5</i>	72,222	27,778	0	0,8	0,667	0,727
<i>6</i>	100	0	0	1	1	1
<i>7</i>	94,444	5,556	0	0,857	1	0,923
<i>8</i>	94,444	5,556	0	0,857	1	0,923
<i>9</i>	100	0	0	1	1	1
<i>10</i>	82,353	17,647	0	0,75	1	0,857
Medie	91,573	8,427	0	0,916	0,916	0,915

3.7 Tabella comparativa generale

Da una prima analisi informale del valore medio f-measure, relativo ad un solo run, si può evidenziare che il comportamento di entrambi gli algoritmi è molto vicino.

In particolare si evidenzia che entrambi gli algoritmi presentano buoni risultati (medie oltre il 90%) per i dataset Iris e Wine, probabilmente perché entrambi presentano solo attributi numerici continui.

Le prestazioni di entrambi gli algoritmi scadono notevolmente (risultati medi intorno al 50%) per il dataset CMC, scadimento, probabilmente, dovuto alla serie di valori discreti che definiscono le classi di appartenenza degli attributi.

Da notare la complessità dell'albero generato da J48 con il dataset CMC, anche se in pochissimo tempo, rispetto alle 5 regole, con al massimo 5 test per la regola più articolata, dell'algoritmo JRIP.

Questa breve analisi, che vuole essere solo un libero commento informale, ai dati riassuntivi riportati nella tabella sottostante, porterebbe a preferire l'algoritmo JRIP:

Algoritmo – Dataset	N° Istanze	N° Attributi	Tempo esecuzione	N° Nodi Foglia	N° Nodi Intermedi	N° Regole	F-measure
<i>J48 – Iris</i>	150	4	0,01	5	4	–	0,96
<i>JRIP – Iris</i>	150	4	0,01	–	–	3	0,953
<i>J48 – CMC</i>	1473	9	0,05	157	106	–	0,52
<i>JRIP – CMC</i>	1473	9	0,2	–	–	5	0,503
<i>J48 – Wine</i>	178	13	0,06	5	4	–	0,932
<i>JRIP – Wine</i>	178	13	0,01	–	–	3	0,915

Un'analisi formale con una sperimentazione composta da più run, sarà effettuata nel capitolo 5, dove saranno tratte una serie di conclusioni supportate da evidenza statistica.

Capitolo 4.

Ottimizzazione Parametri

4.1 Introduzione

Trovare i parametri ottimali per un classificatore può essere un processo piuttosto noioso e soprattutto molto oneroso dal punto di vista computazionale; Weka propone alcuni modi per automatizzare questo processo.

I seguenti meta-classificatori consentono di ottimizzare alcuni parametri del nostro classificatore di base:

- CVParameterSelection
- GridSearch
- MultiSearch (only developer version weka 3.7)

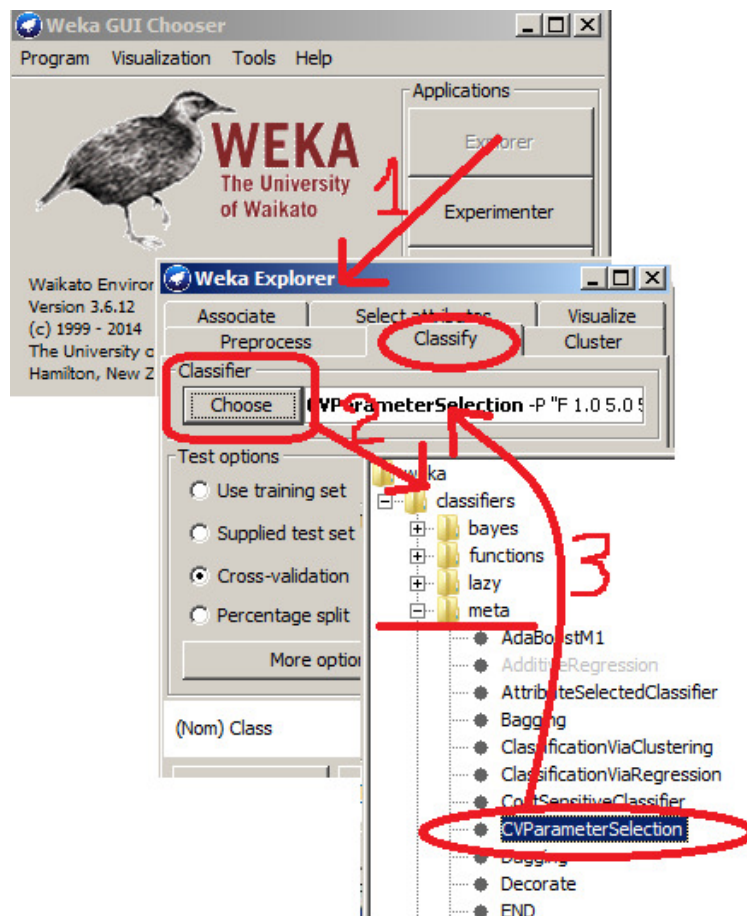
La computazione avviene in 2 fasi principali e tra loro successive:

- Ricerca del miglior set possibile di parametri
- Processo di training, impostando i parametri del classificatore scelto a quelli ritrovati, di cui al precedente punto.

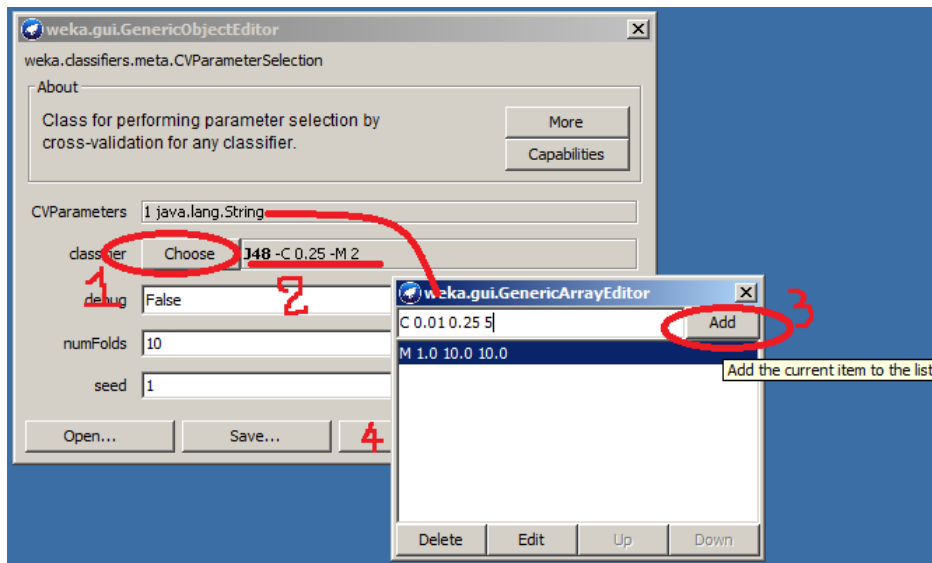
4.2 CVParameterSelection

Questa meta-classificatore può ottimizzare su un **numero arbitrario** di parametri, con un solo difetto (a parte l'esplosione evidente di possibili combinazioni di parametri): non si può ottimizzare sulle opzioni nidificate, solo opzioni dirette del classificatore di base.

Un esempio proposto (proprio sul J48) può essere l'ottimizzazione del parametro C, indicante l'intervallo di confidenza; nell'immagine in basso si specifica il tutto come avviene



Dopo aver scelto il `CVParameterSelection`, comparirà nella TextBox il nome del meta-classificatore; cliccandoci sopra si aprirà la schermata che ci permetterà di poter procedere alla scelta del classificatore da noi scelto su cui ricercare il set di parametri più conveniente.



Dopo aver premuto CHOOSE è possibile scegliere il classificatore, in modo analogo a come precedentemente abbiamo scelto il META-CLASSIFICATORE. In questo caso il J48, cliccando sopra la JLabel di CVParameters si aprirà una schermata dove è possibile inserirèi parametri da ottimizzare.

In questo caso i parametri da ottimizzare saranno 2:

- C: partirà da un valore di 0.01 e arriverà al massimo a 0.25 con un passo di 0.05 (5 step).
- M: partirà da 1 e arriverà a 10 con passo di 1 (10 step)

L'Output sarà questo:

```
Cross-validated Parameter selection.
Classifier: weka.classifiers.trees.J48
Cross-validation Parameter: '-C' ranged from 0.01 to 0.25 with 5.0 steps
Cross-validation Parameter: '-M' ranged from 1.0 to 10.0 with 10.0 steps
Classifier Options: -C 0.06999999999999999 -M 9
```

Dove i migliori valori dei parametri sono individuati da $C \cong 0.07$ e $M=9$.

4.3 GridSearch

È un meta-classificatore per esplorare solo 2 parametri, quindi dal nome *griglia* che indentifica per l'appunto 2 dimensioni. Il classificatore crea un output adatto per [gnuplot](#), vale a dire, le sezioni del registro conterranno sezioni di script e di dati.

È possibile specificare un classificatore di base e di un filtro, entrambi possono essere ottimizzate (un parametro ciascuno).

In contrasto `CVPParameterSelection`, `GridSearch` non è limitata ai parametri di primo livello del classificatore di base.

A causa di alcuni bugfix importanti, si dovrebbe utilizzare una versione di Weka 3.5.6 o superiore; oppure una [snapshot](#) più tardi del 11 settembre 2007.

Per ciascuno dei due assi, X e Y, si possono indicare i seguenti parametri:

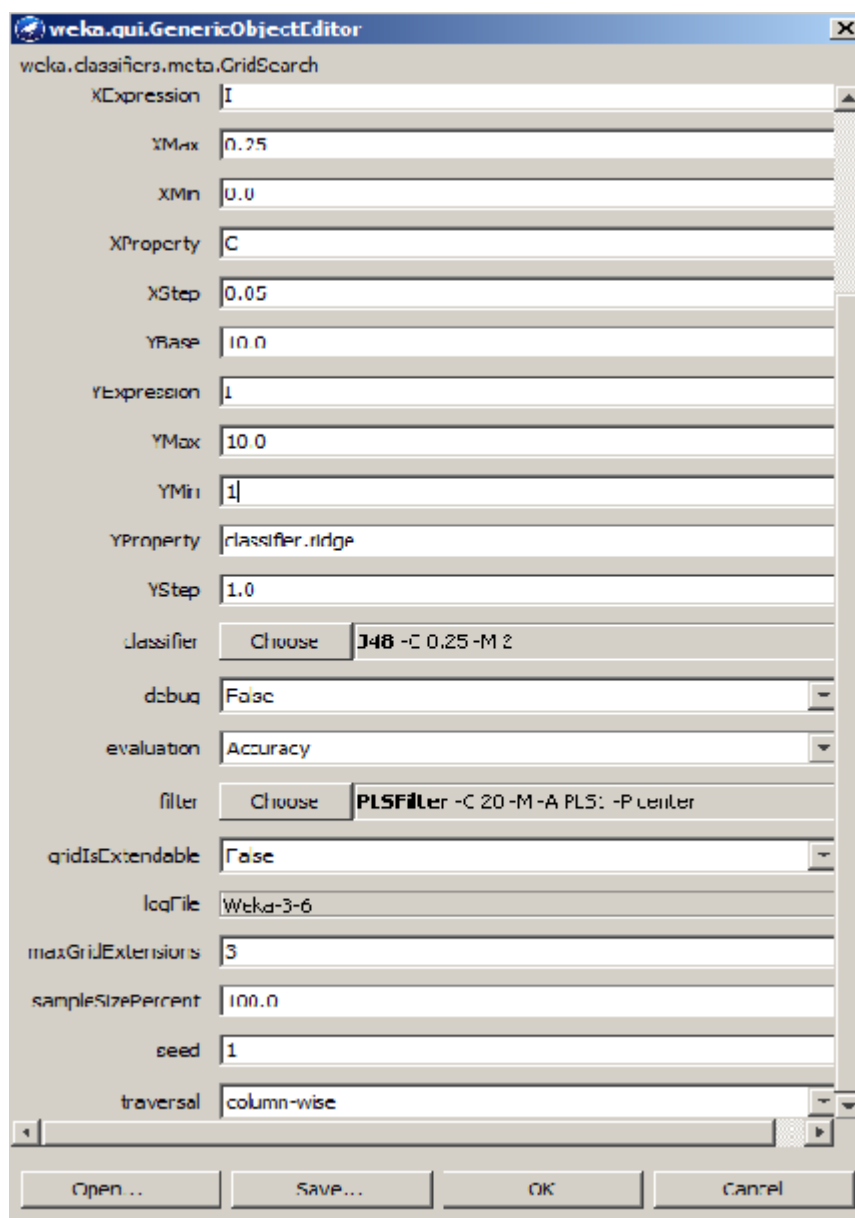
- **proprietà**
Il percorso (dot-notation) indicando la proprietà di da ottimizza (J48 basta scrivere C per intervallo di confidenza, come per il `CVPParameterSelection`).
- **espressione**
L'espressione matematica per generare il valore per la proprietà, elaborati con la classe `weka.core.MathematicalExpression`, che supporta le seguenti funzioni: `abs`, `sqrt`, `log`, `exp`, `sin`, `cos`, `tan`, `tampa`, `piano`, `pow`, `ceil`. Queste variabili sono disponibili nell'espressione: `BASE`, `DA`, `AL`, `STEP`, `I`; con `I` vanno da `FROM` a `TO`.
- **min**
Il valore minimo da cui partire.
- **max**
Il valore massimo a cui giungere.
- **passo**
La dimensione del passo usato per giungere dal *minimo* al *massimo*.
- **base**
Utilizzato in calcoli `pow()`.

`GridSearch` ottimizza in base alle seguenti misure:

- Coefficiente di correlazione (= CC)
- Scarto quadratico medio (RMSE =)
- Root relativo errore quadratico (= RRSE)
- Errore medio assoluto (= MAE)
- Root errore assoluto (= RAE)
- Combinato: $(1 - \text{abs}(\text{CC})) + \text{RRSE} + \text{RAE}$
- Precisione (= ACC)
- Kappa (= KAP) [solo quando si utilizzano pacchetti Weka]

Il *Coefficiente di correlazione* è disponibile solo per le classi a valori numerici e *precisione* solo per quelle classi a valori nominali.

Per quanto riguarda il modo di impostare il Meta-Classificatore la procedura è prevalentemente la stessa del CVPParameterSelection, ad eccezione della schermata di impostazione dei parametri di GridSearch, riportata in basso.



L'output che produce è di questo genere:

```
weka.classifiers.meta.GridSearch:  
Filter: weka.filters.AllFilter  
Classifier: weka.classifiers.trees.J48 -C 0.25 -M 2
```

```
X property: C  
Y property: M
```

```
Evaluation: Accuracy  
Coordinates: [0.25, 10.0]  
Values: 0.25 (X coordinate), 10.0 (Y coordinate)
```

4.4 MultiSearch

Questo meta classificatore è disponibile unicamente attraverso il download di un pacchetto, trovabile a questo indirizzo: <https://github.com/fracpete/multisearch-weka-package/releases>; inoltre per poter aver modo di utilizzarlo è necessario una versione di weka nuova la 3.7 disponibile in sezione DEVELOPER, e successivamente attraverso la GUI CHOOSER è necessario andare su Tool > Package Manager > Selezionare il Button File/URL nella sezione UNOFFICIAL > navigare nel FileSystem e selezionare il pacchetto.zip cui sopra.

Multisearch è simile a GridSearch, più generale e semplice allo stesso tempo.

Più in generale, perché permette l'ottimizzazione di un numero arbitrario di parametri, non solo due.

Più semplice, perché non offre di eventuali espansioni di spazio di ricerca o [gnuplot](#) uscita e meno opzioni.

Per ciascun parametro da ottimizzare, l'utente deve definire un parametro di ricerca *. *. Ci sono due tipi di parametri disponibili:

- `MathParameter` - ciò che usa `GridSearch`, con un'espressione per calcolare il valore effettivo con il min, max e parametri step
- `ListParameter` - l'elenco vuoto di valori separati è usato come ingresso per l'ottimizzazione (utile, se i valori non possono essere descritti da una funzione matematica)

Come per il `GridSearch`, `MultiSearch` può ottimizzare in base alle seguenti misure:

- Coefficiente di correlazione (= CC)
- Scarto quadratico medio (RMSE =)
- Root relativo errore quadratico (= RRSE)
- Errore medio assoluto (= MAE)
- Root errore assoluto (= RAE)
- Combinato: $(1 - \text{abs}(\text{CC})) + \text{RRSE} + \text{RAE}$
- Precisione (= ACC)

- Kappa (= KAP)

Il MultiSearch come anche il GridSearch hanno un costo computazionale davvero elevato, poco si adattano ad essere utilizzate con computer datati e poco performanti.

Il CVPParameterSelection contrariamente a questi 2 ha delle tempistiche molto accettabili anche per computer datati, ed è per questo che si è preferito utilizzarlo per la sperimentazione.

4.5 Sperimentazione Iris CVPParameterSelection

4.5.1 J48

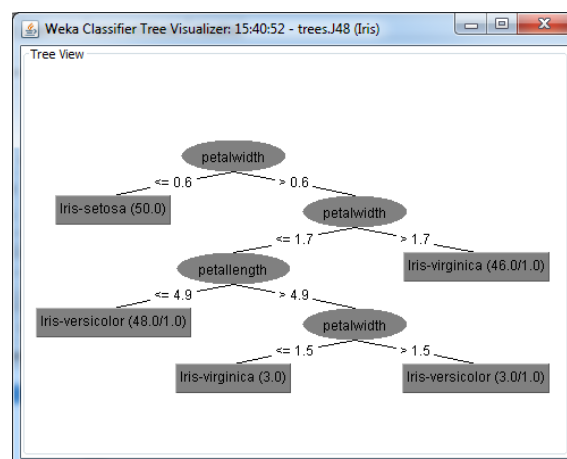
Dataset: Iris

Tecnica: K-Fold Cross Validation (10 fold)

Classificatore: J48

Algoritmo: C4.5

Nella sperimentazione condotta con J48 sul dataset Iris, applicando la tecnica K-Fold Cross Validation, in un tempo di 5,63 secondi, è stato ottenuto il seguente albero formato da 5 nodi foglia e 4 nodi di partizionamento:



La seguente riporta tutte le informazioni generali ottenuta dall'esecuzione della classificazione con Weka – Explorer:

=== Run information ===

Scheme: weka.classifiers.meta.CVParameterSelection -P "C 0.01 0.25 5.0"
-P "M 1.0 10.0 10.0" -X 10 -S 1 -W weka.classifiers.trees.J48 -- -C 0.25 -M
2

Relation: iris

Instances: 150

Attributes: 5

sepalength

sepalwidth

petallength

petalwidth

class

Test mode: 10-fold cross-validation

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

Cross-validated Parameter selection.

Classifier: weka.classifiers.trees.J48

Cross-validation Parameter: '-C' ranged from 0.01 to 0.25 with 5.0 steps

Cross-validation Parameter: '-M' ranged from 1.0 to 10.0 with 10.0 steps

Classifier Options: -C 0.19 -M 2

J48 pruned tree

petalwidth <= 0.6: Iris-setosa (50.0)

petalwidth > 0.6

| petalwidth <= 1.7

| | petallength <= 4.9: Iris-versicolor (48.0/1.0)

| | petallength > 4.9

| | | petalwidth <= 1.5: Iris-virginica (3.0)


```
|      |      |      petalwidth > 1.5: Iris-versicolor (3.0/1.0)
|      petalwidth > 1.7: Iris-virginica (46.0/1.0)
```

Number of Leaves : 5

Size of the tree : 9

Time taken to build model: 5.63 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances	140	93.3333 %
Incorrectly Classified Instances	10	6.6667 %
Kappa statistic	0.9	
Mean absolute error	0.053	
Root mean squared error	0.1943	
Relative absolute error	11.9141 %	
Root relative squared error	41.2231 %	
Coverage of cases (0.95 level)	96.6667 %	
Mean rel. region size (0.95 level)	37.7778 %	
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
	0,920	0,060	0,885	0,920	0,902	0,852	0,948	0,860	Iris-versicolor
	0,900	0,040	0,918	0,900	0,909	0,864	0,957	0,897	Iris-virginica
Weighted Avg.	0,933	0,033	0,934	0,933	0,934	0,900	0,965	0,915	

=== Confusion Matrix ===

```
a  b  c  <-- classified as
49  1  0  |  a = Iris-setosa
```

```
0 46 4 | b = Iris-versicolor
0 5 45 | c = Iris-virginica
```

4.5.2 JRIP

Dataset: Iris

Tecnica: K-Fold Cross Validation (10 fold)

Classificatore: JRip

Algoritmo: RIPPER

Nella sperimentazione condotta con JRIP, in un tempo di 26.5 secondi, sono state ottenute le seguenti 3 regole:

- 1) (petallength >= 3) and (petalwidth <= 1.6) and (petallength <= 4.9) => class=Iris-versicolor (47.0/0.0)
- 2) (petallength >= 4.5) => class=Iris-virginica (53.0/3.0)
- 3) => class=Iris-setosa (50.0/0.0)

In seguito sono riportate tutte le informazioni generali ottenibili dall'esecuzione della classificazione con Weka – Explorer:

```
=== Run information ===
```

```
Scheme:          weka.classifiers.meta.CVParameterSelection -P "F 1.0
5.0 5.0" -P "N 1.0 5.0 5.0" -P "O 1.0 5.0 5.0" -P "S 1.0 5.0 5.0" -X
10 -S 1 -W weka.classifiers.rules.JRip -- -F 3 -N 2.0 -O 2 -S 1
```

```
Relation:      iris
```

```
Instances:     150
```

```
Attributes:    5
```

```
    sepallength
```

```
    sepalwidth
```

```
    petallength
```

```
    petalwidth
```

```
    class
```

```
Test mode:     10-fold cross-validation
```

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

Cross-validated Parameter selection.

Classifier: weka.classifiers.rules.JRip

Cross-validation Parameter: '-F' ranged from 1.0 to 5.0 with 5.0 steps

Cross-validation Parameter: '-N' ranged from 1.0 to 5.0 with 5.0 steps

Cross-validation Parameter: '-O' ranged from 1.0 to 5.0 with 5.0 steps

Cross-validation Parameter: '-S' ranged from 1.0 to 5.0 with 5.0 steps

Classifier Options: -F 3 -N 3 -O 5 -S 5

JRIP rules:

=====

(petallength >= 3) and (petalwidth <= 1.6) and (petallength <= 4.9)

=> class=Iris-versicolor (47.0/0.0)

(petallength >= 4.5) => class=Iris-virginica (53.0/3.0)

=> class=Iris-setosa (50.0/0.0)

Number of Rules : 3

Time taken to build model: 26.5 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances	142	94.6667 %
Incorrectly Classified Instances	8	5.3333 %
Kappa statistic	0.92	
Mean absolute error	0.0507	

```

Root mean squared error          0.1848
Relative absolute error          11.4077 %
Root relative squared error      39.2119 %

Coverage of cases (0.95 level)   96      %
Mean rel. region size (0.95 level) 38.6667 %
Total Number of Instances        150

```

```
=== Detailed Accuracy By Class ===
```

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1,000	0,000	1,000	1,000	1,000	1,000	1,000	1,000	Iris-setosa
	0,880	0,020	0,957	0,880	0,917	0,879	0,948	0,923	Iris-versicolor
	0,960	0,060	0,889	0,960	0,923	0,884	0,948	0,836	Iris-virginica
Weighted Avg.	0,947	0,027	0,948	0,947	0,947	0,921	0,965	0,919	

```
=== Confusion Matrix ===
```

```

a  b  c  <-- classified as
50  0  0 |  a = Iris-setosa
 0 44  6 |  b = Iris-versicolor
 0  2 48 |  c = Iris-virginica

```

4.6 Sperimentazione CMC CVParameterSelection

4.6.1 J48

Dataset: Contraceptive Method Choice

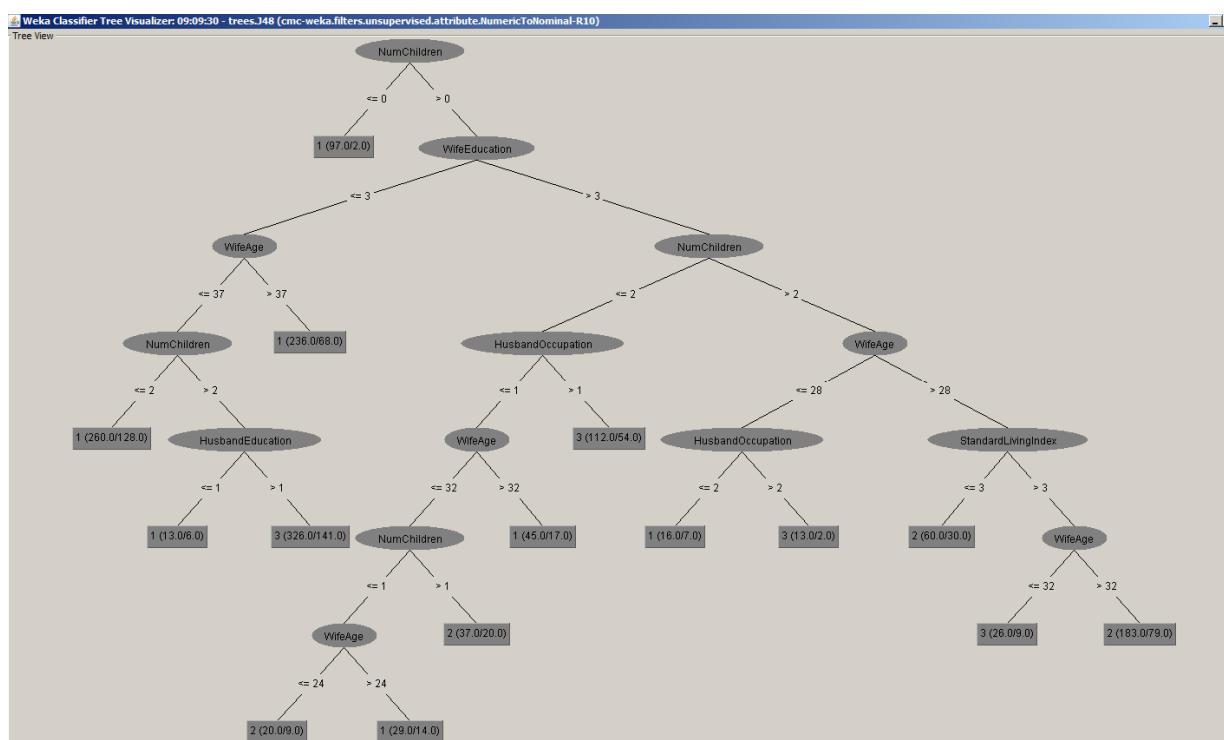
Tecnica: K-Fold Cross Validation (10 fold)

Classificatore: J48

Algoritmo: C4.5

Nella sperimentazione condotta con J48, in un tempo di 47,95 secondi, sul dataset Contraceptive Method Choice applicando sempre la tecnica K-Fold Cross Validation, è stato ottenuto un albero formato da 15 nodi foglia e 13 nodi di partizionamento.

Viene riportata di seguito la rappresentazione grafica poco comprensibile data la dimensione dell'albero:



Le seguenti videate riportano tutte le informazioni generali ottenibili dall'esecuzione della classificazione con Weka – Explorer:

=== Run information ===

Scheme: weka.classifiers.meta.CVParameterSelection -P "C 0.01 0.25 5.0" -P "M 1.0 10.0 10.0" -X 10 -S 1 -W weka.classifiers.trees.J48 -- -C 0.25 -M 2

Relation: cmc-weka.filters.unsupervised.attribute.NumericToNominal-R10

Instances: 1473

Attributes: 10

WifeAge

WifeEducation

HusbandEducation

NumChildren

WifeRreligion

WifeWorking

HusbandOccupation

StandardLivingIndex

MediaExposure

ClassContraceptiveMethodUsed

Test mode: 10-fold cross-validation

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

Cross-validated Parameter selection.

Classifier: weka.classifiers.trees.J48

Cross-validation Parameter: '-C' ranged from 0.01 to 0.25 with 5.0 steps

Cross-validation Parameter: '-M' ranged from 1.0 to 10.0 with 10.0 steps

Classifier Options: -C 0.06999999999999999 -M 9

J48 pruned tree

NumChildren <= 0: 1 (97.0/2.0)

NumChildren > 0

| WifeEducation <= 3

| | WifeAge <= 37

| | | NumChildren <= 2: 1 (260.0/128.0)

| | | NumChildren > 2

| | | | HusbandEducation <= 1: 1 (13.0/6.0)

| | | | HusbandEducation > 1: 3 (326.0/141.0)

| | WifeAge > 37: 1 (236.0/68.0)

| WifeEducation > 3

| | NumChildren <= 2

| | | HusbandOccupation <= 1

| | | | WifeAge <= 32

| | | | | NumChildren <= 1

| | | | | | WifeAge <= 24: 2 (20.0/9.0)

| | | | | | WifeAge > 24: 1 (29.0/14.0)

| | | | | NumChildren > 1: 2 (37.0/20.0)

| | | | WifeAge > 32: 1 (45.0/17.0)

| | | HusbandOccupation > 1: 3 (112.0/54.0)

| | NumChildren > 2

| | | WifeAge <= 28

| | | | HusbandOccupation <= 2: 1 (16.0/7.0)

| | | | HusbandOccupation > 2: 3 (13.0/2.0)

| | | WifeAge > 28

| | | | StandardLivingIndex <= 3: 2 (60.0/30.0)

| | | | StandardLivingIndex > 3

| | | | | WifeAge <= 32: 3 (26.0/9.0)

| | | | | WifeAge > 32: 2 (183.0/79.0)

Number of Leaves : 15

Size of the tree : 29

Time taken to build model: 47.95 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances	837	56.8228 %
Incorrectly Classified Instances	636	43.1772 %
Kappa statistic	0.3247	
Mean absolute error	0.3601	
Root mean squared error	0.4295	
Relative absolute error	83.5792 %	
Root relative squared error	92.5493 %	
Coverage of cases (0.95 level)	99.6606 %	
Mean rel. region size (0.95 level)	95.2704 %	
Total Number of Instances	1473	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,680	0,291	0,635	0,680	0,657	0,386	0,742	0,700	1
	0,423	0,122	0,504	0,423	0,460	0,321	0,697	0,417	2
	0,524	0,261	0,516	0,524	0,520	0,263	0,687	0,494	3
Weighted Avg.	0,568	0,243	0,564	0,568	0,565	0,329	0,713	0,565	

=== Confusion Matrix ===

```
a  b  c  <-- classified as
399 61 169 |  a = 1
 96 129 108 |  b = 2
158 97 256 |  c = 3
```

4.6.2 JRIP

Dataset: Contraceptive Method Choice

Tecnica: K-Fold Cross Validation (10 fold)

Classificatore: JRip

Algoritmo: RIPPER

Nella sperimentazione condotta con JRIP, in un tempo di 4219.14 secondi, sono state ottenute le seguenti 4 regole:

```
(wife_education = 4) and (number_child >= 3) and (wife_age >= 34) and (wife_age <= 42) → class=2 (141.0/56.0)
(wife_education = 4) and (number_child >= 3) and (wife_age >= 29) and (wife_age <= 48) and (wife_age >= 44) → class=2 (53.0/24.0)
(wife_age <= 35) and (number_child >= 3) → class=3 (377.0/171.0)
(wife_age <= 30) and (number_child >= 1) and (wife_age <= 22) → class=3 (122.0/59.0)
→ class=1 (780.0/326.0)
```

Le seguenti videate riportano tutte le informazioni generali ottenibili dall'esecuzione della classificazione con Weka – Explorer:

```
=== Run information ===
```

```
Scheme:          weka.classifiers.meta.CVParameterSelection -P "F 1.0 5.0
5.0" -P "N 1.0 5.0 5.0" -P "O 1.0 5.0 5.0" -P "S 1.0 5.0 5.0" -X 10 -S 1 -W
weka.classifiers.rules.JRip -- -F 3 -N 2.0 -O 2 -S 1
```

```
Relation:        cmc-weka.filters.unsupervised.attribute.NumericToNominal-
R10
```

```
Instances:       1473
```

```
Attributes:      10
```

```
WifeAge
```

```
WifeEducation
```

```
HusbandEducation
```

```
NumChildren
```

```
WifeReligion
```

```
WifeWorking
HusbandOccupation
StandardLivingIndex
MediaExposure
ClassContraceptiveMethodUsed
Test mode:    10-fold cross-validation
```

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

Cross-validated Parameter selection.

Classifier: weka.classifiers.rules.JRip

Cross-validation Parameter: '-F' ranged from 1.0 to 5.0 with 5.0 steps

Cross-validation Parameter: '-N' ranged from 1.0 to 5.0 with 5.0 steps

Cross-validation Parameter: '-O' ranged from 1.0 to 5.0 with 5.0 steps

Cross-validation Parameter: '-S' ranged from 1.0 to 5.0 with 5.0 steps

Classifier Options: -F 5 -N 1 -O 4 -S 5

JRIP rules:

=====

```
( WifeEducation  >=  4)  and ( NumChildren  >=  3)  and (WifeAge  >=  34)  =>
ClassContraceptiveMethodUsed=2 (214.0/96.0)
(WifeAge <= 35) and ( NumChildren >= 3) => ClassContraceptiveMethodUsed=3 (377.0/171.0)
(WifeAge <= 37) and ( NumChildren >= 1) and (WifeAge <= 22) and (WifeAge >= 21) and (
HusbandOccupation >= 2) => ClassContraceptiveMethodUsed=3 (63.0/24.0)
=> ClassContraceptiveMethodUsed=1 (819.0/342.0)
```

Number of Rules : 4

Time taken to build model: 4219.14 seconds

```
=== Stratified cross-validation ===
```

=== Summary ===

Correctly Classified Instances	790	53.632 %
Incorrectly Classified Instances	683	46.368 %
Kappa statistic	0.2595	
Mean absolute error	0.3927	
Root mean squared error	0.447	
Relative absolute error	91.1541 %	
Root relative squared error	96.3097 %	
Coverage of cases (0.95 level)	99.9321 %	
Mean rel. region size (0.95 level)	99.7058 %	
Total Number of Instances	1473	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,719	0,432	0,553	0,719	0,625	0,285	0,663	0,538	1
	0,339	0,088	0,531	0,339	0,414	0,299	0,654	0,366	2
	0,440	0,227	0,508	0,440	0,472	0,222	0,603	0,447	3
Weighted Avg.	0,536	0,283	0,532	0,536	0,524	0,266	0,640	0,467	

=== Confusion Matrix ===

a	b	c	<-- classified as
452	36	141	a = 1
143	113	77	b = 2
222	64	225	c = 3

4.7 Sperimentazione WINE CVPParameterSelection

4.7.1 J48

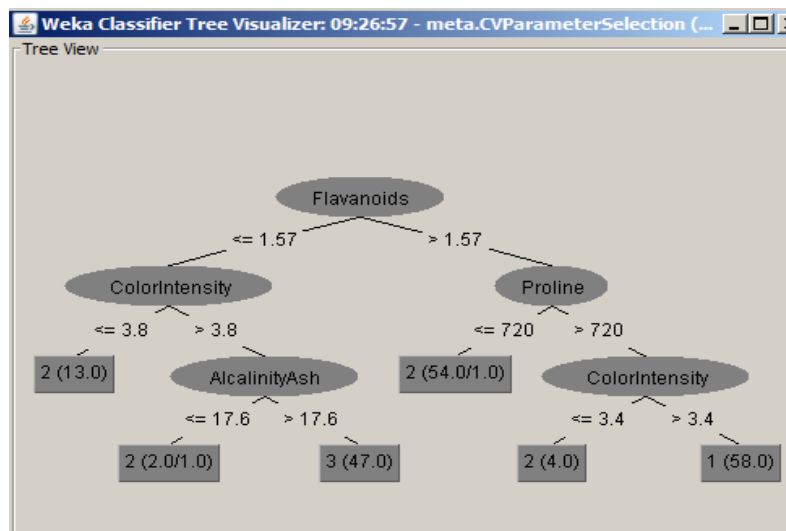
Dataset: Wine

Tecnica: K-Fold Cross Validation (10 fold)

Classificatore: J48

Algoritmo: C4.5

Nella sperimentazione condotta con J48 sul dataset Wine applicando sempre la tecnica K-Fold Cross Validation, in un tempo di 1,91 secondi, è stato ottenuto il seguente albero formato da 6 nodi foglia e 5 nodi di partizionamento:



Le seguenti videate riportano tutte le informazioni generali ottenibili dall'esecuzione della classificazione con Weka – Explorer:

=== Run information ===

Scheme: weka.classifiers.meta.CVPParameterSelection -P "C 0.01 0.25 5.0" -P "M 1.0 10.0 10.0" -X 10 -S 1 -W weka.classifiers.trees.J48 -- -C 0.25 -M 2

Relation: wine-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst-weka.filters.unsupervised.attribute.NumericToNominal-R1-weka.filters.unsupervised.attribute.Reorder-R2,3,4,5,6,7,8,9,10,11,12,13,14,1

Instances: 178

Attributes: 14

Alcohol
MalicAcid
Ash
AlcalinityAsh
Magnesium
TotalPhenols
Flavanoids
NonflavanoidPhenols
Proanthocyanins
ColorIntensity
Hue
DilutedWines
Proline
Class

Test mode: 10-fold cross-validation

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

Cross-validated Parameter selection.

Classifier: weka.classifiers.trees.J48

Cross-validation Parameter: '-C' ranged from 0.01 to 0.25 with 5.0 steps

Cross-validation Parameter: '-M' ranged from 1.0 to 10.0 with 10.0 steps

Classifier Options: -C 0.01 -M 1

J48 pruned tree

Flavanoids <= 1.57
| ColorIntensity <= 3.8: 2 (13.0)
| ColorIntensity > 3.8
| | AlcalinityAsh <= 17.6: 2 (2.0/1.0)

```
|      |      AlcalinityAsh > 17.6: 3 (47.0)
|      |      Flavanoids > 1.57
|      |      Proline <= 720: 2 (54.0/1.0)
|      |      Proline > 720
|      |      ColorIntensity <= 3.4: 2 (4.0)
|      |      ColorIntensity > 3.4: 1 (58.0)
```

Number of Leaves : 6

Size of the tree : 11

Time taken to build model: 1.91 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances	167	93.8202 %
Incorrectly Classified Instances	11	6.1798 %
Kappa statistic	0.9058	
Mean absolute error	0.05	
Root mean squared error	0.2021	
Relative absolute error	11.3797 %	
Root relative squared error	43.144 %	
Coverage of cases (0.95 level)	94.382 %	
Mean rel. region size (0.95 level)	35.0187 %	
Total Number of Instances	178	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,983	0,034	0,935	0,983	0,959	0,938	0,974	0,934	1
	0,944	0,056	0,918	0,944	0,931	0,884	0,936	0,884	2
	0,875	0,008	0,977	0,875	0,923	0,899	0,946	0,903	3
Weighted Avg.	0,938	0,036	0,940	0,938	0,938	0,906	0,952	0,905	

=== Confusion Matrix ===

```

      a   b   c   <-- classified as
58    1   0   |   a = 1
      3 67   1   |   b = 2
1     5 42   |   c = 3

```

4.7.2 JRIP CVParameterSelection

Dataset: Contraceptive Method Choice

Tecnica: K-Fold Cross Validation (10 fold)

Classificatore: JRip

Algoritmo: RIPPER

Nella sperimentazione condotta con JRIP, in un tempo di 0,01 secondi, sono state ottenute le seguenti 3 regole:

```

(flavanoids <= 1.39) and (color_intensity >= 3.85) → class=3 (47.0/0.0)
(proline >= 760) and (color_intensity >= 3.52) → class=1 (57.0/0.0)
→ class=2 (74.0/3.0)

```

Le seguenti videate riportano tutte le informazioni generali ottenibili dall'esecuzione della classificazione con Weka – Explorer:

```
=== Run information ===
```

Scheme:

```

weka.classifiers.meta.CVParameterSelection -P "F 1.0
5.0 5.0" -P "N 1.0 5.0 5.0" -P "O 1.0 5.0 5.0" -P "S
1.0 5.0 5.0" -X 10 -S 1 -W weka.classifiers.rules.JRip
-- -F 3 -N 2.0 -O 2 -S 1

```

```

Relation: wine-
weka.filters.unsupervised.attribute.NumericToNominal-
Rfirst-
weka.filters.unsupervised.attribute.NumericToNominal-

```

```
R1-weka.filters.unsupervised.attribute.Reorder-  
R2,3,4,5,6,7,8,9,10,11,12,13,14,1
```

```
Instances:      178
```

```
Attributes:     14
```

```
    Alcohol
```

```
    MalicAcid
```

```
    Ash
```

```
    AlcalinityAsh
```

```
    Magnesium
```

```
    TotalPhenols
```

```
    Flavanoids
```

```
    NonflavanoidPhenols
```

```
    Proanthocyanins
```

```
    ColorIntensity
```

```
    Hue
```

```
    DilutedWines
```

```
    Proline
```

```
    Class
```

```
Test mode:      10-fold cross-validation
```

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

```
Cross-validated Parameter selection.
```

Classifier: weka.classifiers.rules.JRip

Cross-validation Parameter: '-F' ranged from 1.0 to 5.0 with 5.0 steps

Cross-validation Parameter: '-N' ranged from 1.0 to 5.0 with 5.0 steps

Cross-validation Parameter: '-O' ranged from 1.0 to 5.0 with 5.0 steps

Cross-validation Parameter: '-S' ranged from 1.0 to 5.0 with 5.0 steps

Classifier Options: -F 2 -N 1 -O 5 -S 2

JRIP rules:

=====

(DilutedWines <= 2.11) and (ColorIntensity >= 3.85)
=> Class=3 (47.0/2.0)

(Flavanoids <= 0.6) => Class=3 (4.0/1.0)

(Proline >= 760) and (ColorIntensity >= 3.52) =>
Class=1 (57.0/0.0)

=> Class=2 (70.0/2.0)

Number of Rules : 4

Time taken to build model: 100.49 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances	162	91.0112 %
Incorrectly Classified Instances	16	8.9888 %
Kappa statistic	0.863	

Mean absolute error	0.0674
Root mean squared error	0.2407
Relative absolute error	15.3553 %
Root relative squared error	51.3639 %
Coverage of cases (0.95 level)	92.1348 %
Mean rel. region size (0.95 level)	35.5805 %
Total Number of Instances	178

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,881	0,025	0,945	0,881	0,912	0,872	0,960	0,930	1
	0,930	0,093	0,868	0,930	0,898	0,828	0,938	0,871	2
	0,917	0,023	0,936	0,917	0,926	0,900	0,949	0,924	3
Weighted Avg.	0,910	0,052	0,912	0,910	0,910	0,862	0,948	0,905	

=== Confusion Matrix ===

```

a  b  c  <-- classified as
52  7  0 |  a = 1
 2 66  3 |  b = 2
 1  3 44 |  c = 3

```

4.8 Tabella di confronto e considerazioni operative per l'esperimento

In questa sezione viene mostrata una tabella che permette di poter rapidamente visionare in una panoramica generale, i differenti risultati ottenuti permettendo un'analisi comparativa.

Dei 3 datasets verranno riportati gli indicatori utili alla valutazione degli algoritmi e anche i diversi parametri estratti (ottimizzati) e di default.

Dataset		CVParameterSelection		Default Parameter	
		J48	Jrip	J48	Jrip
Iris					
	Parametri	- C 0,19 -M 2	-F 3 -N 3 -O 5 -S 5	-C 0.25 -M 2	-F 3 -N 2 -O 2 -S 1
	Correctly Classified	93,3333	94,6667	96	95,3333
	InCorrectly Classified	6,6667	5,3333	4	4,6667
	Kappa	0,9	0,92	0,94	0,93
	Average TP rate	0,933	0,947	0,96	0,953
	Average FP rate	0,033	0,027	0,02	0,023
	Average Precision	0,934	0,948	0,96	0,953
	Average Recall	0,933	0,947	0,96	0,953
	Average F-measure	0,934	0,947	0,96	0,953
Contraceptive Method Choice					
	Parametri	-C 0,07 -M 9	-F 5 -N 1 -O 4 -S 5	-C 0.25 -M 2	-F 3 -N 2 -O 2 -S 1
	Correctly Classified	56,8228	53,632	52,1385	52,41
	InCorrectly Classified	43,1772	46,368	47,8615	47,59
	Kappa	0,3247	0,2595	0,2549	0,232
	Average TP rate	0,568	0,536	0,521	0,524
	Average FP rate	0,243	0,283	0,264	0,298
	Average Precision	0,564	0,532	0,521	0,518
	Average Recall	0,568	0,536	0,521	0,524
	Average F-measure	0,565	0,524	0,522	0,503
Wine					
	Parametri	-C 0,01 -M 1	-F 2 -N 1 -O 5 -S 2	-C 0.25 -M 2	-F 3 -N 2 -O 2 -S 1
	Correctly Classified	93,8202	91,0112	93,8202	91,573
	InCorrectly Classified	6,1798	8,9888	6,1798	8,427
	Kappa	0,9058	0,86	0,9058	0,8725
	Average TP rate	0,938	0,91	0,938	0,916
	Average FP rate	0,036	0,052	0,036	0,044
	Average Precision	0,94	0,912	0,94	0,916
	Average Recall	0,938	0,91	0,938	0,916
	Average F-measure	0,938	0,91	0,938	0,915

Come si può ben notare la differenza, in termini di risultati è pochissima (decimi o pochissimi punti in percentuale); a volte risulta però che i parametri ottimizzati producano un risultato

peggiore di quelli standard; ciò può essere spiegato dal fatto che dovendo scegliere un intervallo e il numero di step in cui ripartirlo, può succedere che il meta classificatore possa produrre una combinazione di parametri non molto performanti.

Ma si tratta comunque di differenze minime, che in nessun modo possono influire sulla valutazione della sperimentazione.

Per motivi operazionali, si è scelto di utilizzare nell'EXPERIMENT WEKA i parametri di DEFAULT, per il semplice fatto che non è possibile utilizzare un certo set di parametri su di un classificatore per uno SPECIFICO dataset, ma si dovrebbe inserire nella sezione algoritmi sempre lo stesso classificatore con i parametri cambiati, che andranno poi a essere computati con tutti i dataset. Computazionalmente l'operazione è troppo costosa. E tuttavia si ribadisce che la piccola differenza nei valori degli indici non è significativa a tal punto da richiedere una computazione di questo genere.

Capitolo 5.

Confronto e Conclusioni

5.1 Introduzione

In questo capitolo, dopo una breve introduzione sui test statistici, sarà riportata una descrizione delle operazioni di confronto vero e proprio dei due algoritmi e un'analisi dei risultati ottenuti.

5.2 Test delle ipotesi

Il test delle ipotesi mira a validare le ipotesi sperimentali formulate durante la pianificazione.

I test eseguiti possono essere test parametrici o non parametrici.

I test parametrici si basano su un modello che richiede una specifica distribuzione dei dati di tipo normale e misurati almeno su una scala intervallo. A parte queste restrizioni, essi godono di grande potenza statistica e possono essere applicati anche con pochi punti sperimentali.

I test non-parametrici hanno condizioni di applicabilità più generali, prescindono dalle assunzioni sulla distribuzione dei campioni e possono essere applicati anche a serie ordinate, ovvero aventi distribuzione con carattere qualitativo.

Paired T-Test o test di Student è un test parametrico frequentemente usato per confrontare statisticamente l'uguaglianza delle medie di due popolazioni (unpaired t-test) o di una stessa popolazione (paired t-test) rispetto ad un determinato parametro.

Weka – Experimenter prevede il paired t-test per effettuare i confronti dello stesso dataset sottoposto ai due algoritmi.

Le ipotesi formulate saranno le seguenti:

- H_0 : i due algoritmi hanno la stessa performance, quindi medie uguali.
- H_1 : i due algoritmi non hanno la stessa performance, quindi medie diverse.

Il livello di confidenza sarà $\alpha = 0,05$.

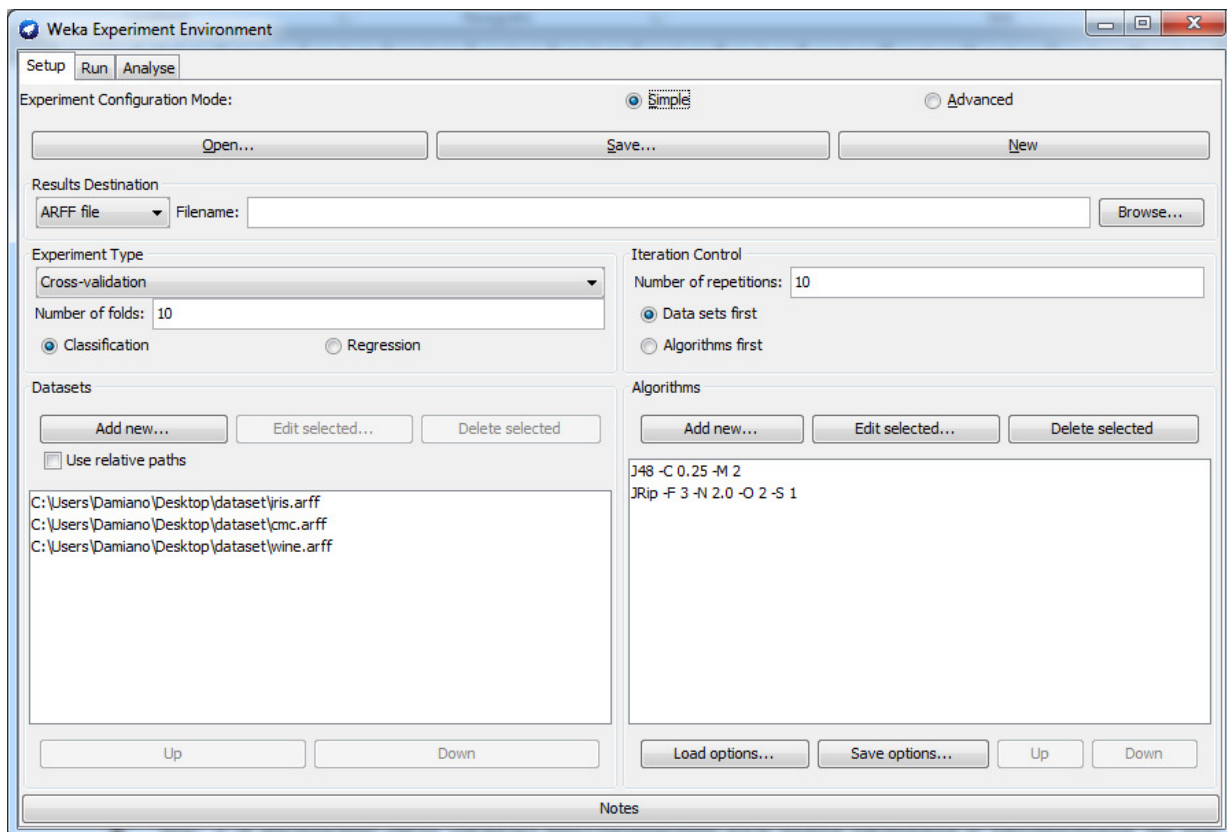
5.3 Experimenter

Per poter avviare la sperimentazione ed il confronto vero e proprio bisogna utilizzare l'interfaccia Experimenter ed effettuare le seguenti operazioni:

- indicare tutti i dataset da utilizzare;
- indicare quali algoritmi mettere a confronto;

- impostare il metodo di sperimentazione, numero di folds per il Cross-Validation e numero di iterazioni o run;

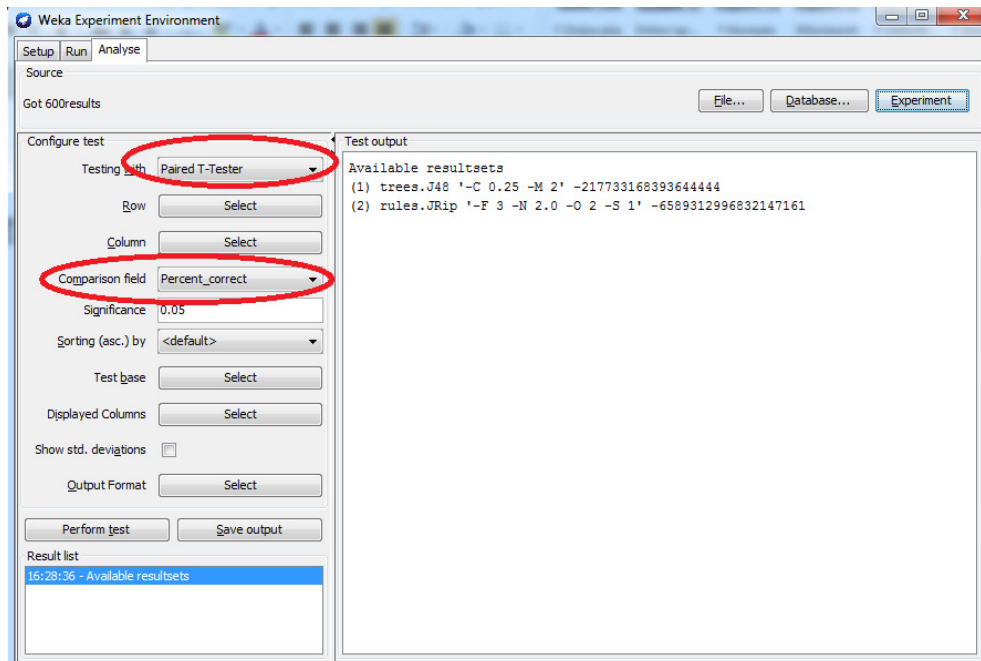
avviare la sperimentazione ed attendere i risultati.



Terminata l'esecuzione si passa all'analisi dei risultati, questi ultimi sono stati già riportati nelle tabelle precedenti.

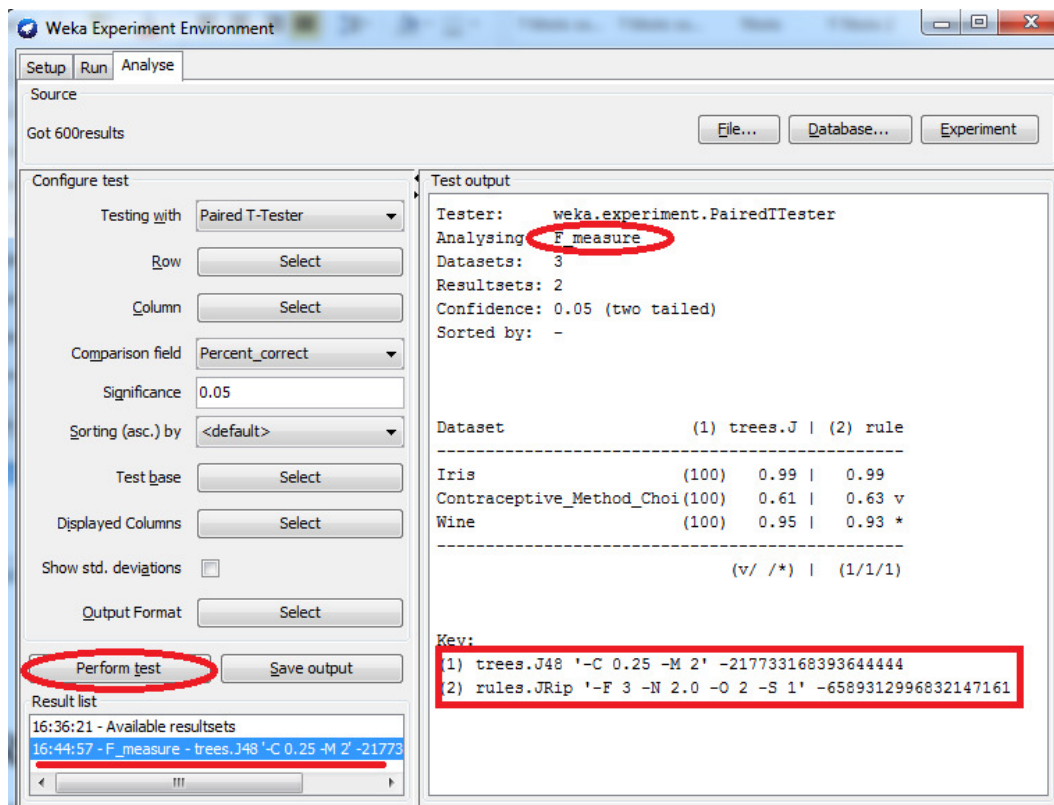
La videata successiva prepara la configurazione per la comparazione dei risultati ottenuti dall'esperimento. Il tasto Experiment ripropone i risultati relativi all'elaborazione precedente.

A questo punto bisogna configurare i parametri del test.



5.4 Risultati del Test

La videata seguente mostra i risultati ottenuti dal software:



I simboli riportati vicino ogni valore medio ed in corrispondenza del relativo algoritmo, sono da interpretare in questo modo:

- (v): indica una performance migliore;
- (*): indica una performance peggiore;
- (): indica la stessa performance.

Mentre sul lato destro dei simboli (v / $*$) tra parentesi viene riportato il numero di volte che sono stati ottenuti i diversi comportamenti ($I/I/I$).

Nello specifico, su una media di 100 valori per ogni dataset (10 fold per 10 run), sono stati ottenuti i seguenti risultati:

- con Iris, i due algoritmi si comportano allo stesso modo;
- con CMC, JRIP presenta performance migliori rispetto a J48;
- con Wine, J48 ha performance migliori rispetto a JRIP.

5.5 Conclusioni

Sulla base dei vari risultati ottenuti, durante le diverse fasi di classificazione e sperimentazione, possiamo asserire che:

- J48 ha prestazioni migliori in termini di tempo, però presenta alberi piuttosto complessi soprattutto in presenza di dataset con molte istanze e con attributi discreti;
- JRIP ha necessità di maggior tempo per estrapolare regole, queste ultime risultano comunque più compatte;
- dai risultati del test statistico possiamo accettare l'ipotesi H_0 quindi, possiamo asserire che i due algoritmi hanno le stesse performance;

quindi, i due algoritmi si equiparano.

Pertanto, potendo scegliere, possiamo preferire l'algoritmo JRIP data la compattezza e la semplicità di interpretazione delle regole che, in tutti e tre i casi, ha estrapolato durante la sperimentazione.

Bibliografija

- Mitchell, T. (1997), *Machine Learning*, McGraw Hill. [ISBN 0-07-042807-7](#)
- Quinlan, J. R. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, 1993.
- J. R. Quinlan. Improved use of continuous attributes in c4.5. *Journal of Artificial Intelligence Research*, 4:77-90, 1996
- J.R. Quinlan, *Induction of decision trees - Machine learning*, 1986 - Springer
- Valentin Koblar, Bogdan Filipič - optimizing parameters of machine learning algorithms
- R.S. Michalski , *A theory and methodology of inductive learning* - 1983 - Springer
- Weka Manual:
https://sourceforge.net/projects/weka/files/documentation/3.6.x/WekaManual-3-6-12.pdf/download?use_mirror=garr&download=
- UCI repository: <http://archive.ics.uci.edu/ml/>
Iris: <http://archive.ics.uci.edu/ml/datasets/Iris>
Wine: <http://archive.ics.uci.edu/ml/datasets/Wine>
Cmc: <https://archive.ics.uci.edu/ml/datasets/Contraceptive+Method+Choice>
- Metaclassifier Parameter Optimization:
<https://weka.wikispaces.com/Optimizing+parameters#MultiSearch>