Predizione del Divorzio tramite l'analisi di domande sulla vita coniugale

Studente: Massimiliano Sirgiovanni
Matricola: 7077251
Email: massimiliano.sirgiovanni@stud.unifi.it

Il Dataset scelto, *per la realizzazione del progetto per Data Mining and Organization*, è composto da una serie di **risposte a 54 domande**, le quali descrivono delle situazioni riguardanti la vita sentimentale di alcune coppie.
Ogni risposta ad una domanda è stata memorizzata sotto forma di **numero naturale**, compreso nell'intervallo [0,4], e rappresenta quanto gli individui interrogati ritengono la loro relazione sentimentale affine alla situazione descritta dalla domanda. Dunque, ad ogni domanda la risposta è stata scelta considerando le seguenti opzioni:

(0=Never, 1=Seldom, 2=Averagely, 3=Frequently, 4=Always)

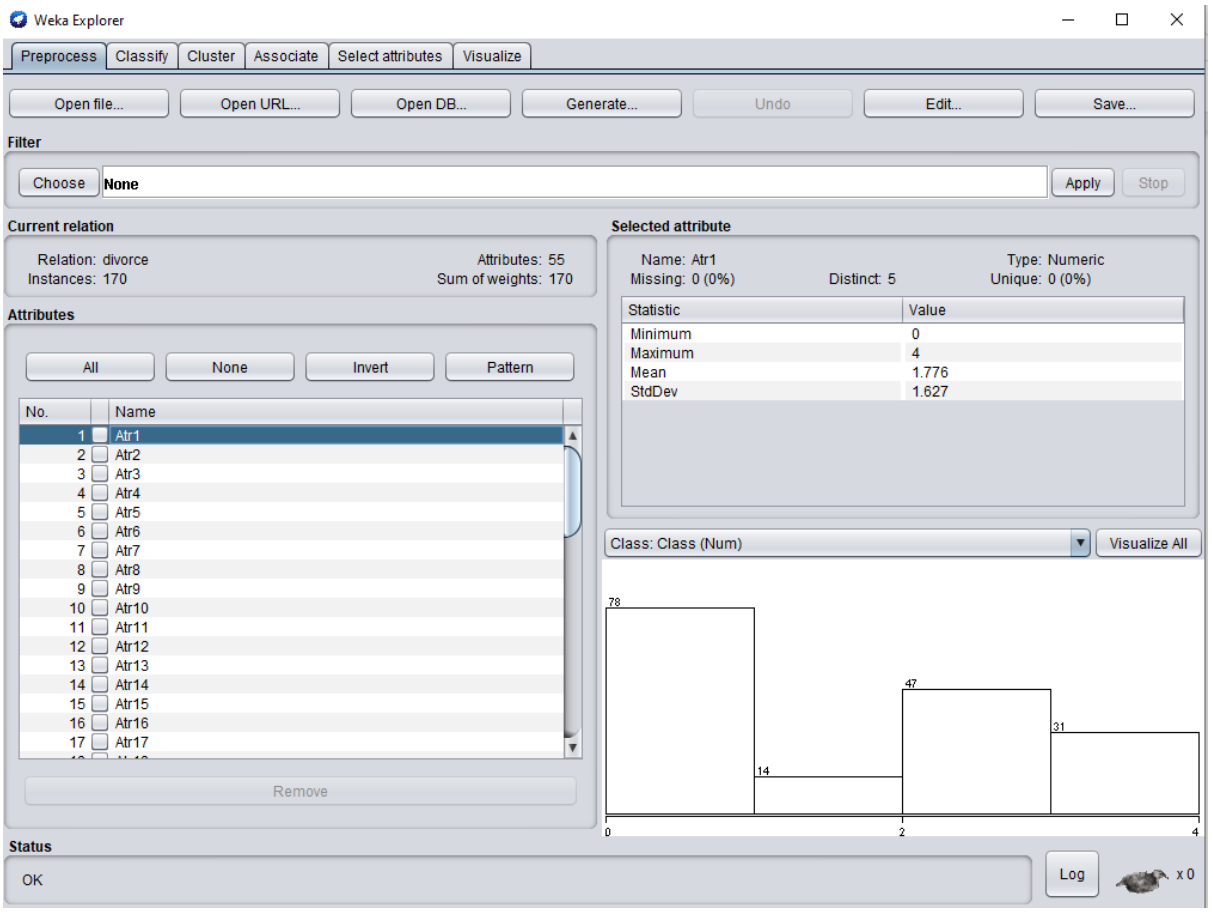
I dati sono stati raccolti da un gruppo di ricercatori dell'università **Hacı Bektaş Veli** nella città di **Nevşehir** (Turchia)**.** Per questo studio è stato selezionato un **campione di 170 candidati**, di cui 84 *(49%)* divorziati e 86 *(51%)* coppie sposate. Le persone divorziate hanno dovuto rispondere considerando i loro passati matrimoni.
D'altra parte, in rappresentanza del secondo gruppo, sono state selezionate solo coppie felicemente sposate, che non avessero alcuna intenzione di divorziare.
Il campione selezionato era composto sia da uomini che da donne, di età compresa tra i 20 e i 60 anni.
I partecipanti hanno compilato un modulo contenente le domande sopracitate.

Tutte le *"domande"* sono state numerate per essere poi ricollegabili alle etichette assegnati ai vari attributi del database. Viene, di sotto, riportato un **campione delle domande**, in modo da dare un'idea del tipo di situazioni proposte ai candidati *(tutte le 54 domande sono reperibili in un file allegato alla relazione)*.

1. If one of us apologizes when our discussion deteriorates, the discussion ends.
2. I know we can ignore our differences, even if things get hard sometimes.
3. When we need it, we can take our discussions with my spouse from the beginning and correct it.
4. When I discuss with my spouse, to contact him will eventually work.
5. The time I spent with my wife is special for us.

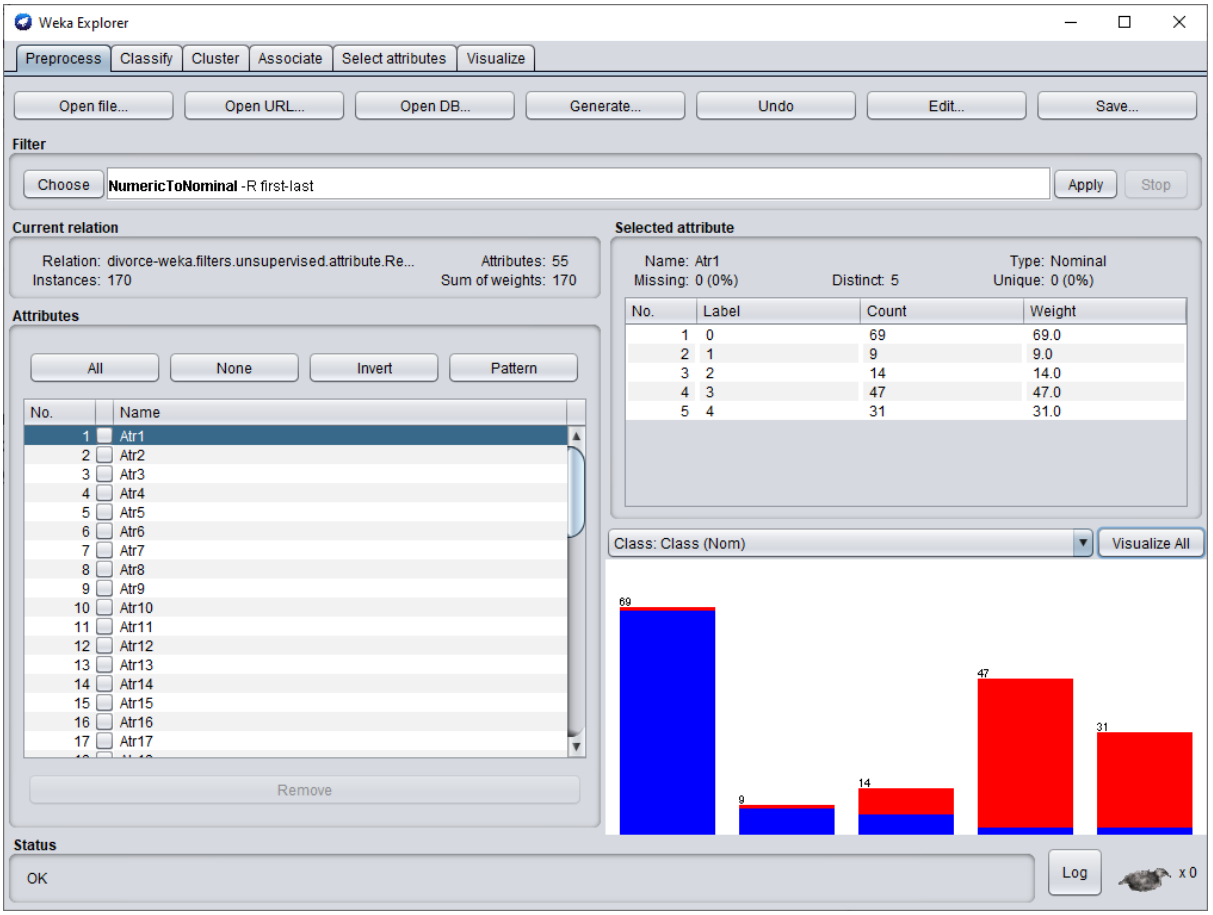
All'interno del dataset le etichette degli attributi assumono una forma del tipo: *"Atr(valore numerico)"*. Il valore numerico corrisponde al valore attribuito alla domanda. Inoltre, è presente un ultimo attributo che rappresenta il valore di classe.

Questo attributo può assumere valore uno, nel caso dei divorziati, o valore zero nel caso delle coppie sposate.
Il file contenente i dati è stato reso disponibile in due formati, xls e csv, tuttavia il contenuto dei due file è identico. Per comodità, è stato utilizzato il file in formato csv, tuttavia è risultato necessario effettuare una modifica prima che potesse essere importato nel software Weka.
Tale file, reso disponibile sul sito dell'UCI, contiene come separatore per i dati il carattere *";"*. Caricandolo in Weka, senza apportare alcuna modifica, l'intero dataset verrebbe considerato come avente un solo attributo, di etichetta *"Atr1;Atr2;...Atr54;Class"*.
Per correggere questo errore è stato sufficiente **sostituire**, all'interno del file csv, **il carattere *";"* con un carattere *","***. In questo modo il file sarà facilmente leggibile tramite Weka, riportando i 54 attributi come separati.
A questo punto il file può essere importato correttamente in Weka.



Si possono verificare, tramite la schermata di Weka, le caratteristiche del dataset e degli attributi, viste in precedenza (come il numero di istanze, il numero di attributi, il valore assunto dagli attributi e così via).
Considerato il tipo di dato, è improbabile che vi siano valori mancanti, tuttavia, è opportuno eseguire la funzionalità di Weka *"ReplaceMissingValues"* sul data set (nell'eventuale caso in cui un candidato avesse dimenticato di inserire una risposta).

Un'altra procedura necessaria da effettuare è la **trasformazione delle variabili**. Infatti, il fatto di avere attributi di tipo numerico preclude l'utilizzo di molti algoritmi di classificazione. Di conseguenza, è conveniente trasformare gli attributi numerici in attributi nominali, tramite l'apposito filtro *"NumericToNominal"*.
Si ottengono in questo modo anche nuove informazioni sui dati.



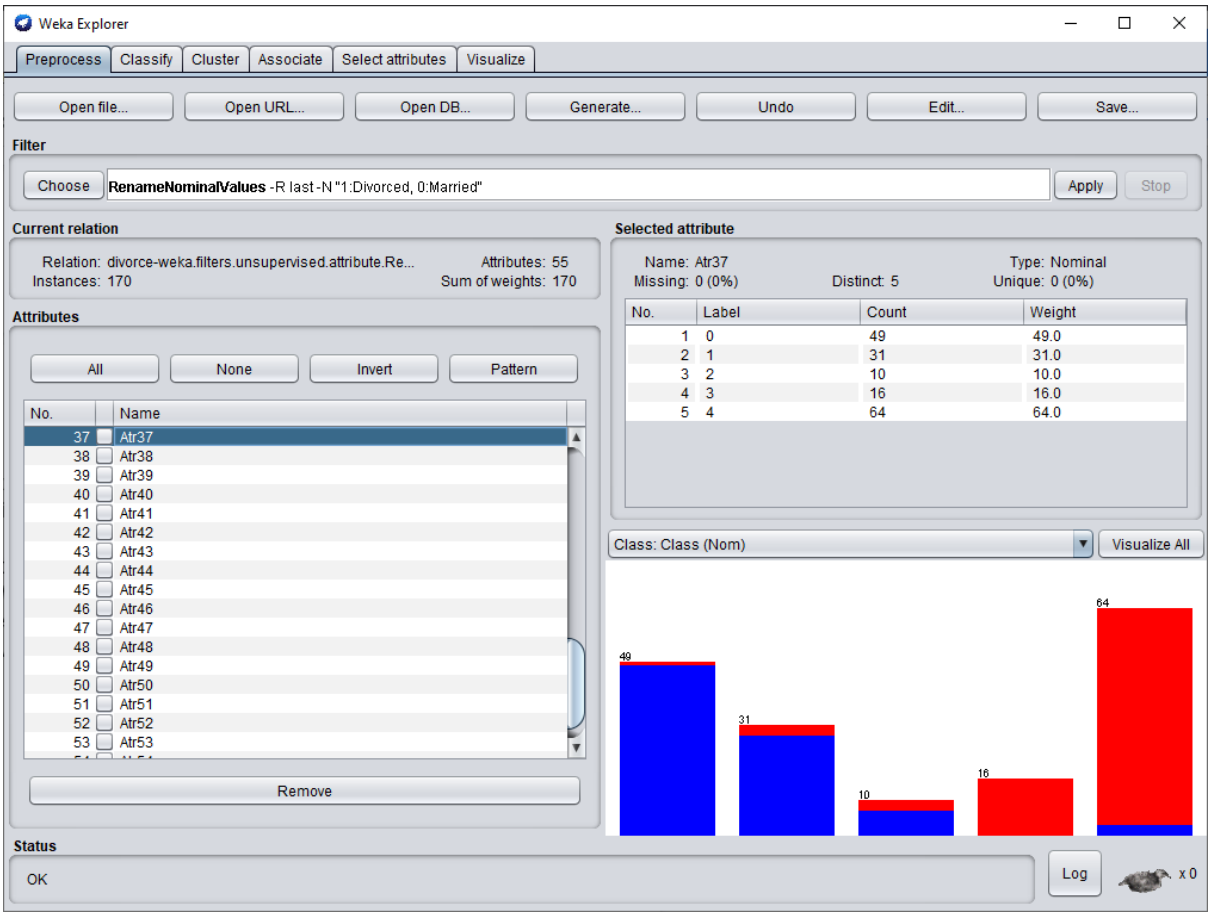
Si ottengono informazioni relative al numero di record in cui compare un determinato valore di un attributo. Nonché, delle informazioni grafiche che, specialmente viste nel loro insieme, restituiscono diverse informazioni che possono indurre ad importanti intuizioni.
Osservando l'attributo *Atr1*, contenente le risposte date alla domanda:

1. *If one of us apologizes when our discussion deteriorates, the discussion ends.*

Si può osservare un fatto curioso. Infatti, qualora la situazione sia molto familiare *(ovvero l'attributo assume un valore più alto)* le possibilità di divorzio *(identificate con il colore rosso)* tendono ad aumentare. D'altra parte, se la situazione risulta poco familiare, le possibilità di divorzio tendono a diminuire.
Sicuramente, appare come un fatto singolare, dato che, intuitivamente, si potrebbe immaginare una situazione inversa. In altre situazioni, invece, i dati restituiscono valori più prevedibili.
Ad esempio, nel caso dell'attributo numero 37, correlato alla domanda:

37. My discussion with my spouse is not calm.

Come ci si aspetterebbe, nel caso in cui l'affermazione sia tendenzialmente vera, si tratta di individui divorziati. Viceversa, quando la situazione non si verifica mai, si tratta di individui sposati.

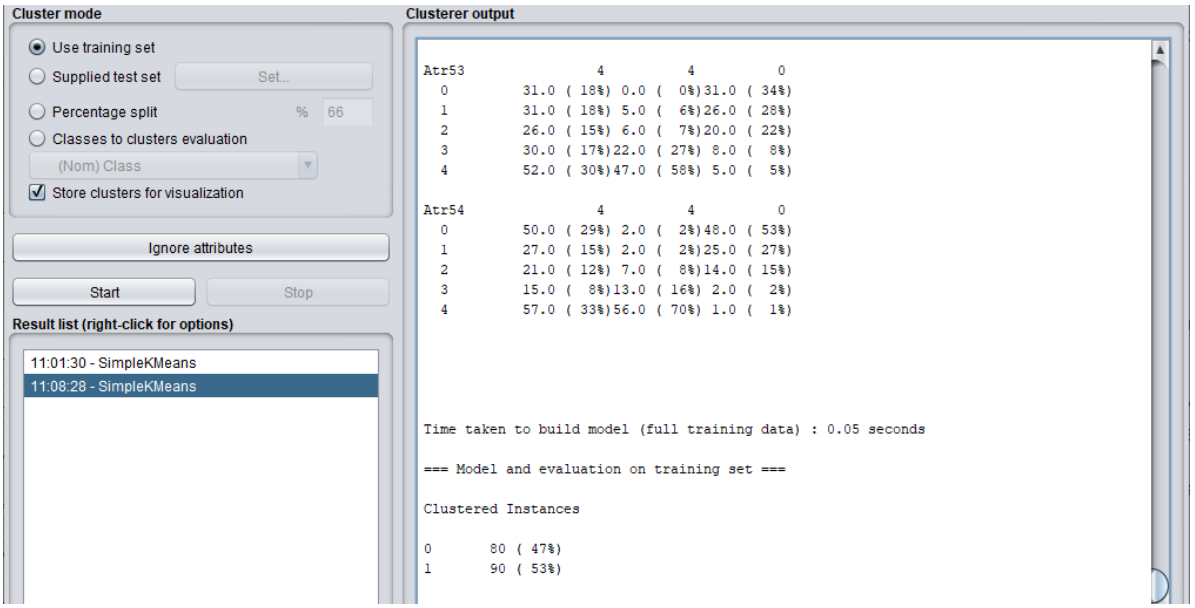


Per ottenere una migliore visualizzazione dei dati si è optato per modificare i valori dell'attributo di classe. Si è usato il filtro *RenameNominalValues*, effettuando le seguenti due sostituzioni:

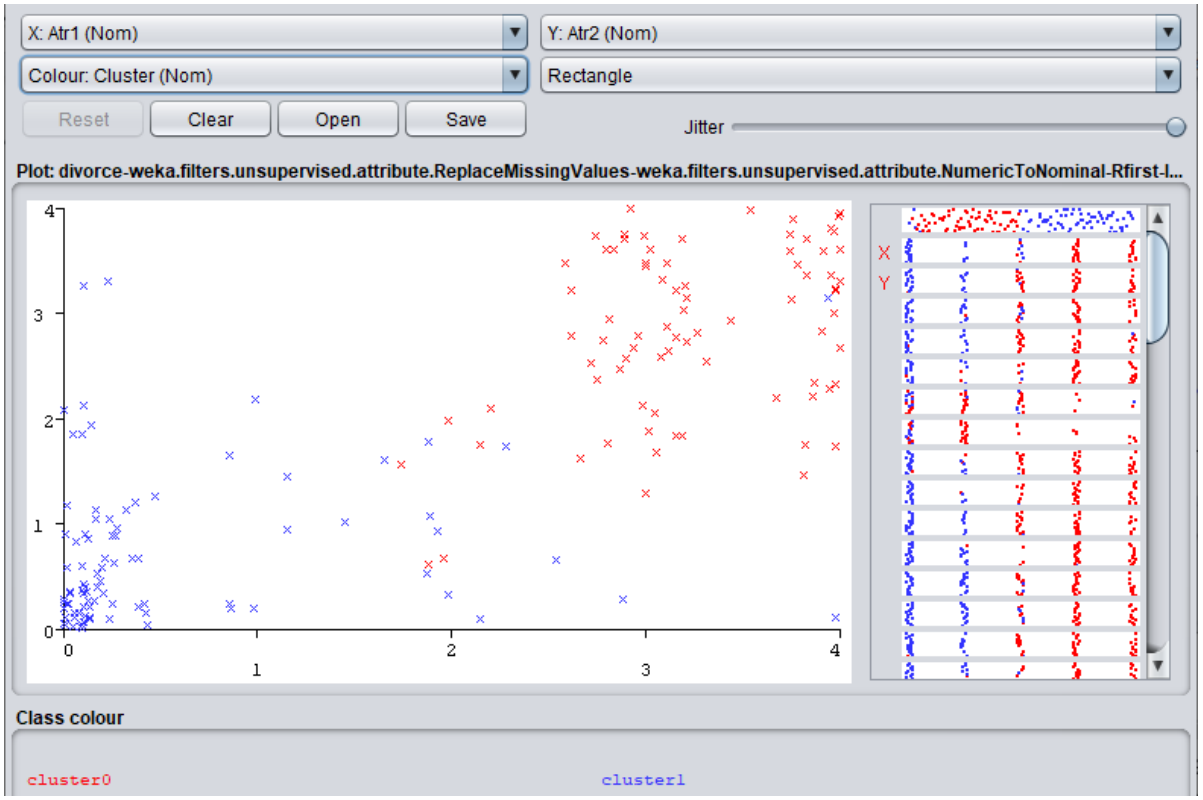
- 0 ← Married
- 1 ← Divorced

Terminata la fase di *preprocessing*, è possibile iniziare ad applicare gli algoritmi visti a lezione sui dati raccolti.
Si procede dunque ad applicare la prima tecnica di clustering studiata, l'algoritmo **K-Means**. L'algoritmo K-Means è un approccio di **clustering partizionale**, basato sulla scelta di K centroidi, rappresentanti K cluster distinti, ai quali verranno poi assegnati tutti i punti del data set.
Un punto viene assegnato ad un centroide piuttosto che ad un altro basandosi sulla distanza tra il punto ed il centroide stesso, scegliendo quello che genera la distanza minima.
Durante la prima applicazione dell'algoritmo verrà ignorato l'attributo di classe.
Si assegna a K valore due, dato che l'attributo classe contiene due valori distinti.

Una volta applicato l'algoritmo si possono analizzare i risultati.

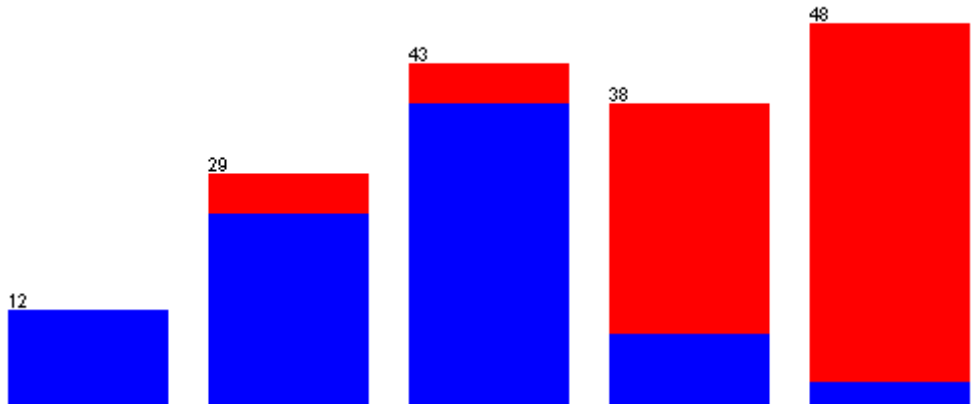


Come si può ben osservare, l'algoritmo K-means è riuscito ad individuare due cluster, approssimativamente, delle medesime dimensioni, come ci si aspetterebbe osservando i dati. Infatti, il campione di partenza è formato da due gruppi composti da 86 ed 84 persone. Tramite la funzionalità di Weka *"Visualize cluster assignment"* è possibile anche visualizzare graficamente i dati ed il comportamento del clustering su di essi. Dato il gran numero di attributi, è impraticabile inserire tutti i grafici in questa sede, tuttavia si possono inserire alcuni di essi ed osservare il loro comportamento. Ad esempio, il primo grafico che viene presentato mette in relazione, *sull'asse delle X*, l'attributo Atr1 e, *sull'asse delle Y*, l'attributo Atr2.

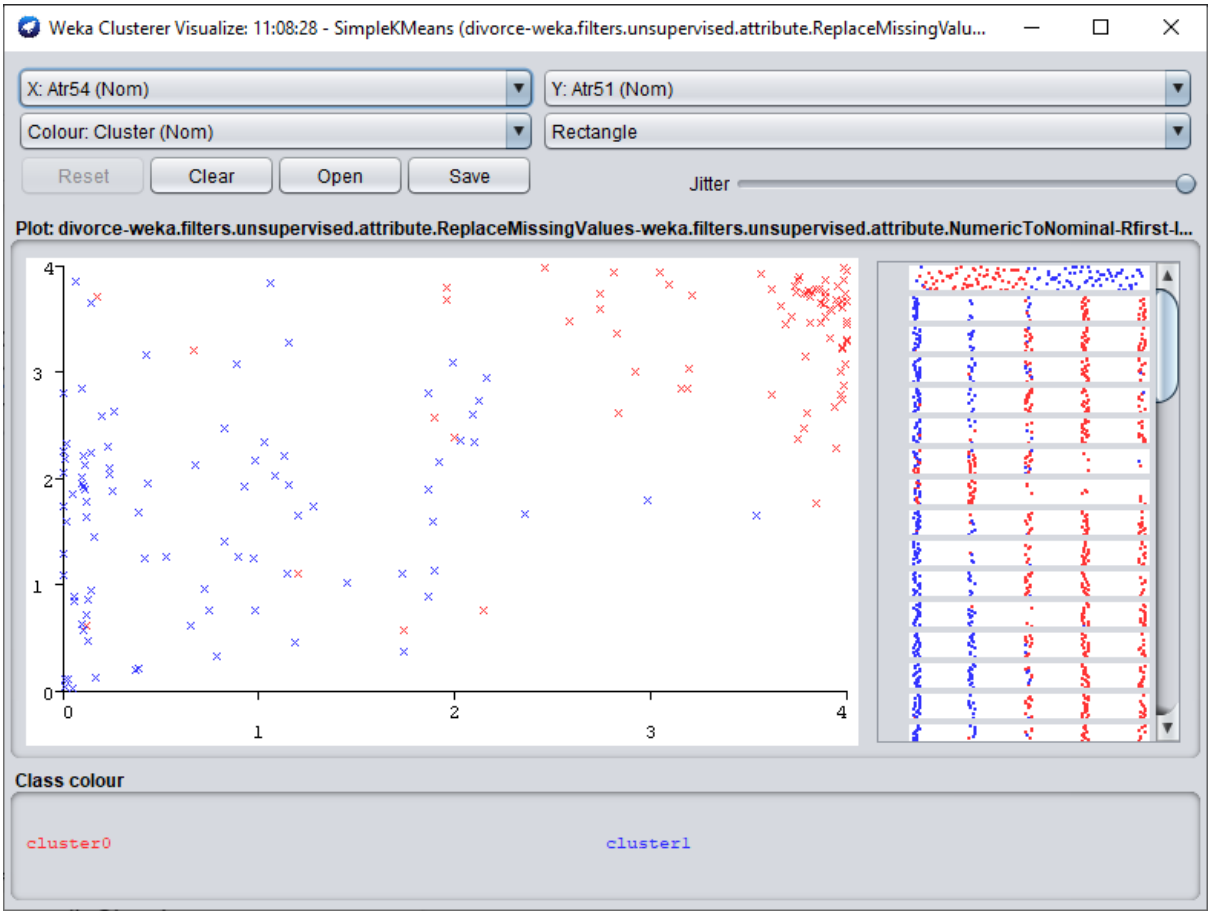


Si può facilmente osservare come individui che hanno fornito risposte simili alle prime due domande appartengono, tendenzialmente, alla medesima classe.
Difatti, è possibile osservare due cluster ben definiti, eccetto per alcuni punti di rumore riconducibili principalmente al “cluster1”.
Il comportamento dei dati osservato durante la procedura di clustering è ben visibile anche osservando i grafici relativi ai primi due attributi. Infatti, in entrambi i casi, valori bassi delle risposte corrispondono, solitamente, all’appartenenza alla classe “Married” e, viceversa, valori alti corrispondono all’appartenenza alla classe “Divorced”.

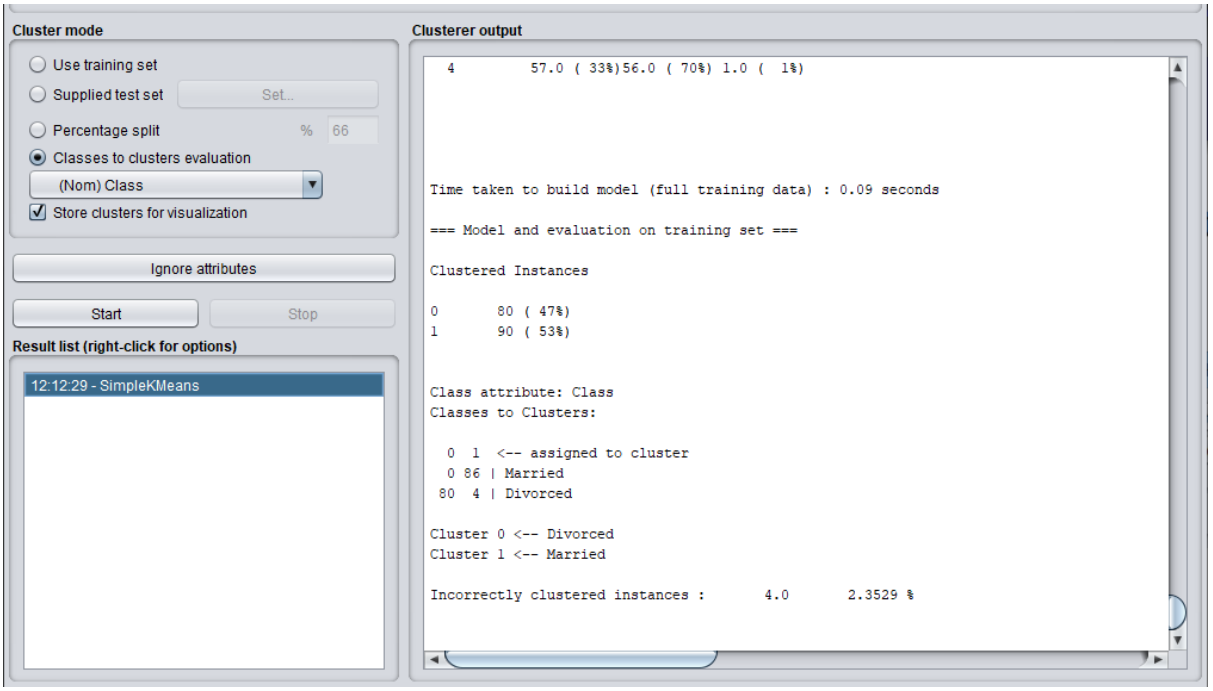
Si può invece osservare una diversa relazione effettuando un grafico tra l’attributo numero 54, i cui valori non si discostano molto da quelli visti nei primi due attributi, ed il numero 51, il quale si comporta in modo più particolare.



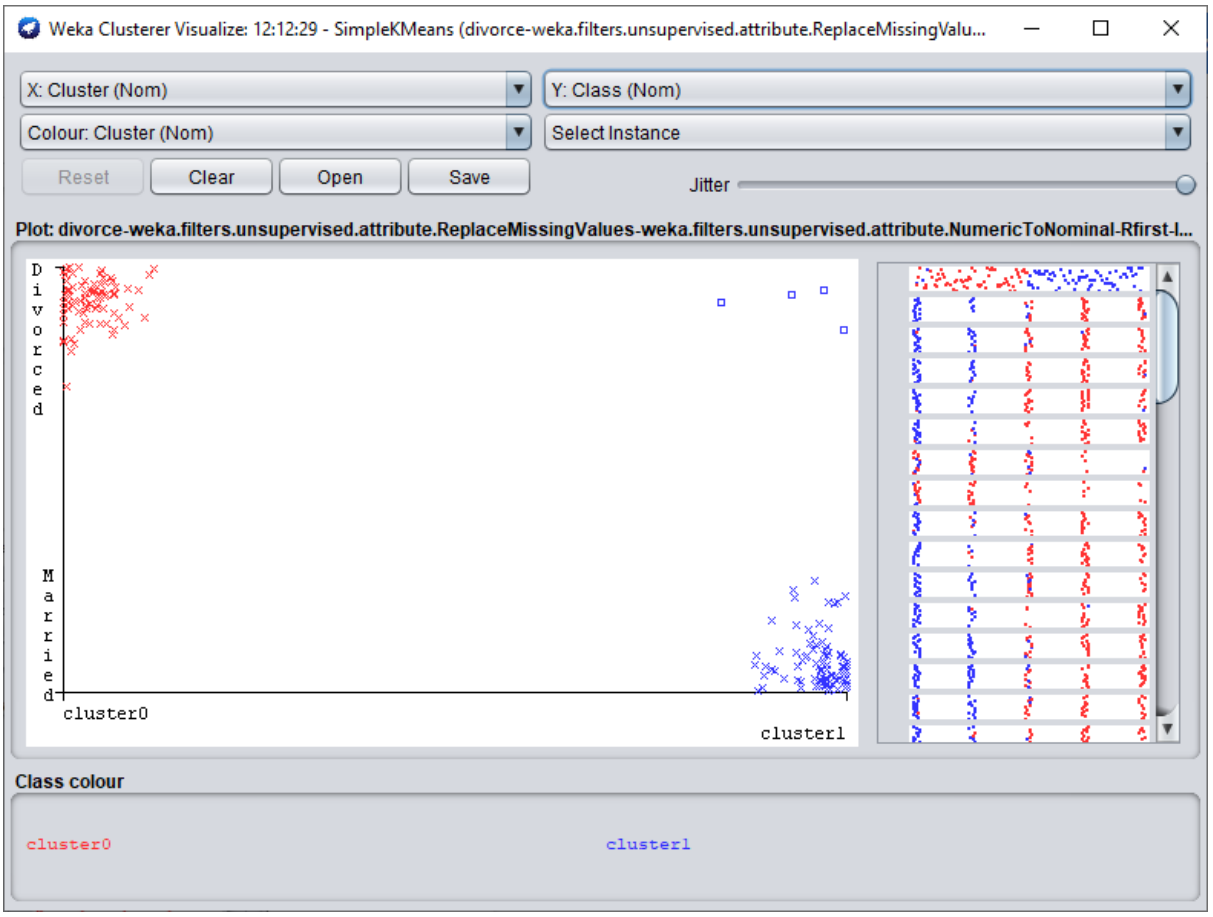
Infatti, in questo caso gli individui sposati sono distribuiti più uniformemente tra le risposte e nella maggior parte dei casi hanno restituito valori intermedi. Si può osservare questo comportamento anche nel grafico ottenuto tramite il K-Means.



Si può osservare come, nel grafico, i punti del cluster1 *(che contiene principalmente individui di classe "Married")* siano distribuiti verso valori più piccoli sull'asse delle X, *corrispondente all'attributo 54*, e come, d'altra parte, siano distribuiti più uniformemente sull'asse delle Y, *corrispondente all'attributo 51*.
Sull'asse delle Y vi è, tuttavia, una maggiore densità di punti intorno al valore 2, come osservato dal grafico dell'attributo 51 sui dati.
Date le dimensioni dei cluster, è indubbio che alcuni record sono stati mal classificati, ma a prima impressione, ed anche studiando il comportamento dei dati sui grafici, sembrerebbero essere molto pochi.
Indubbiamente, per verificare questa tesi, è opportuno effettuare un test per la valutazione del cluster, usando la classe esplicita fornita.
E' possibile effettuare questo test in Weka tramite l'apposita modalità *"Classes to clusters evaluation"*.
Una volta selezionata sarà sufficiente selezionare il nome della classe ed eseguire l'algoritmo di clustering sul dataset.
Si possono osservare i risultati di questa classificazione nella finestra Clusterer output, in Weka. Viene fornita anche una matrice che consente di osservare quanto accurata è stata la classificazione e, dunque, quanti oggetti della medesima classe sono stati inseriti nello stesso cluster.

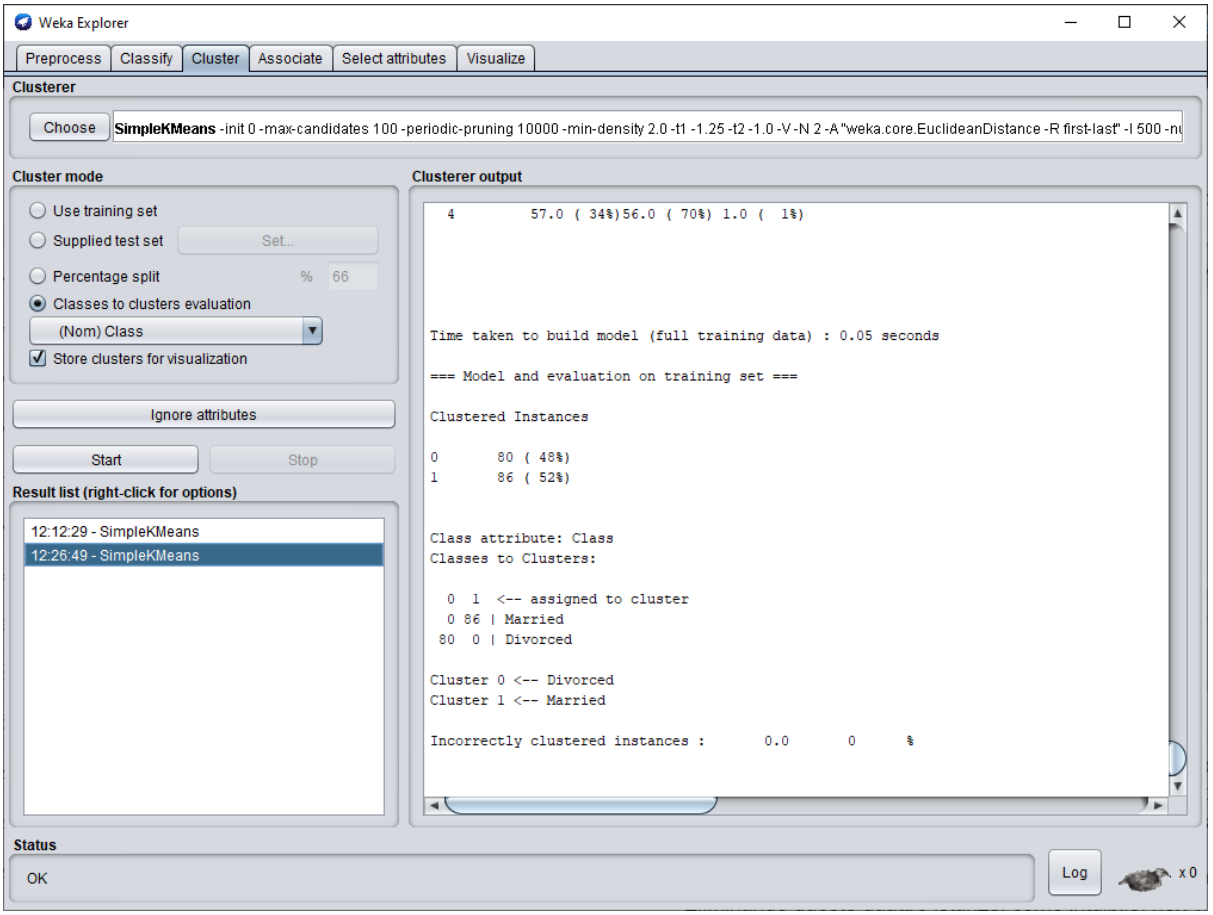


Come previsto, l'errore di clustering sul dataset è piuttosto basso. Infatti, solo quattro attributi sono stati inseriti nel cluster sbagliato.



Dal grafico in alto si ha un'idea del comportamento dei due cluster ed è possibile visualizzare anche i quattro elementi che sono stati mal classificati (sono i quattro

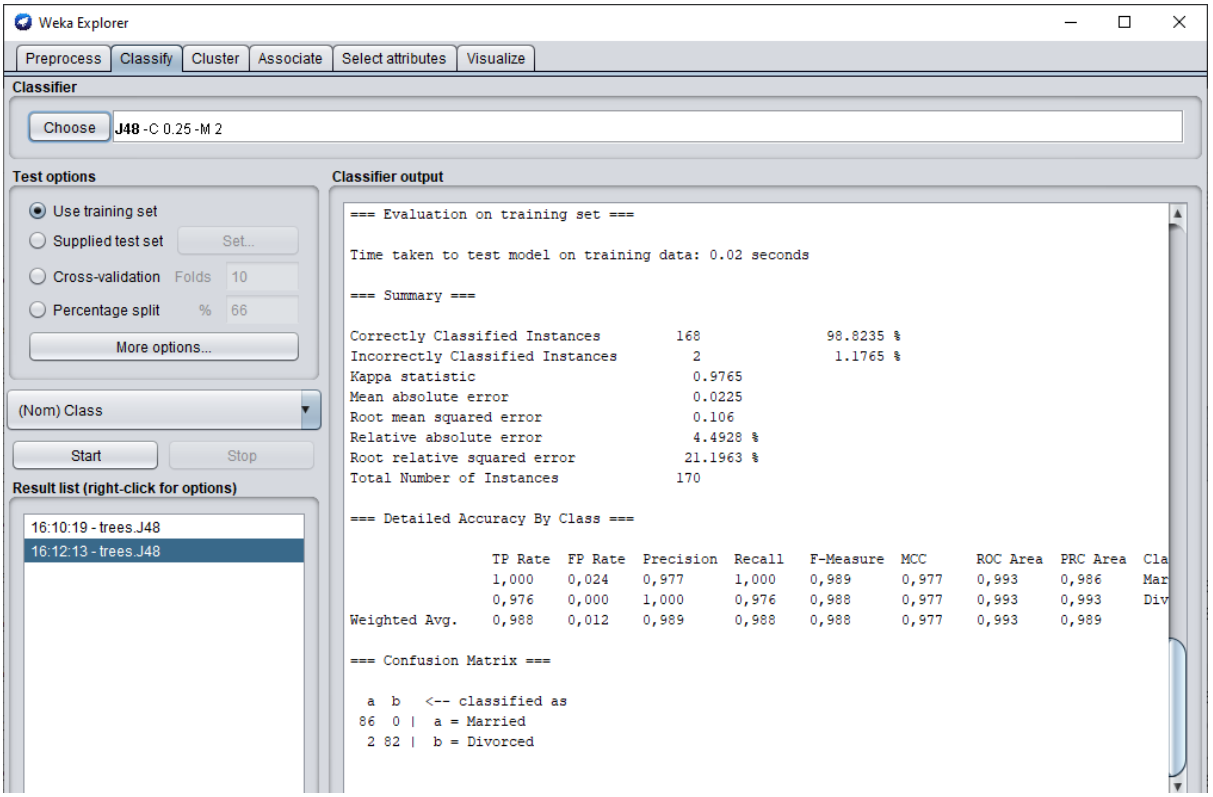
quadratini blu in alto a sinistra). Tali record, con grande probabilità, sono degli outliers. Cliccando su ogni punto è possibile identificare a quale istanza corrisponde, in modo tale da permetterne la rimozione nel data set.
In questo caso, i quattro outliers corrispondono alle istanze di valore: [1, 5, 6, 10].
Eliminando queste quattro istanze, come intuibile, non vi saranno più errori nel clustering, come osservabile nell'immagine sottostante.



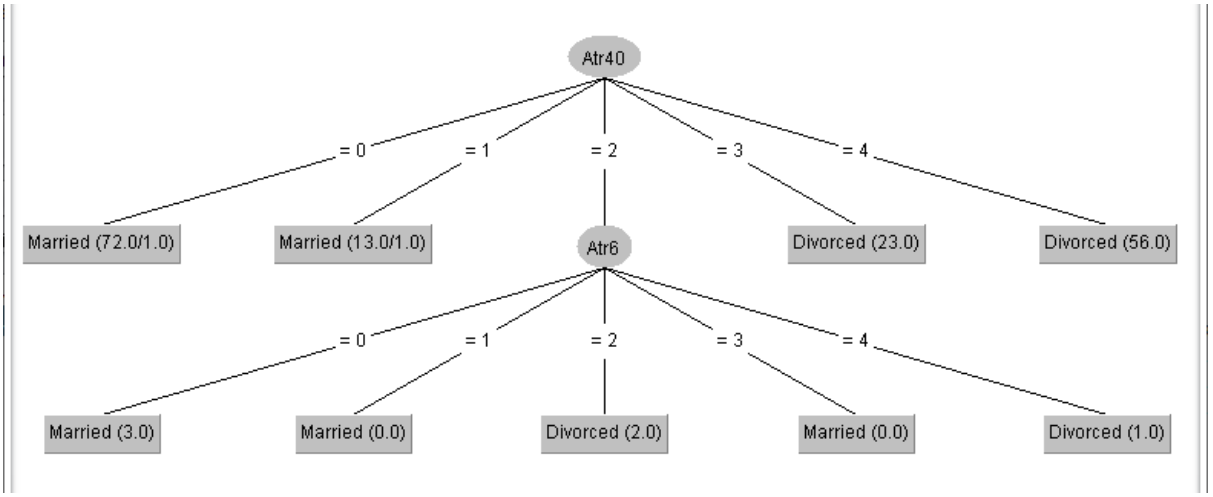
Tuttavia, prima di optare per l'eliminazione di tale istanze, potrebbe convenire eseguire alcuni test aggiuntivi.

Applicate delle procedure di clustering sul dataset, è opportuno passare alla fase di classificazione, motivo principale per la raccolta di tali dati.
L'obiettivo da raggiungere è, dunque, quello di cercare un modello che sia in grado di predire, con un certo margine di errore, le possibilità di divorzio di una coppia.
Le finalità della classificazione sono proprio quelle di ricercare un modello di predizione della classe. Vi sono molte tecniche di classificazione, disponibili anche in Weka.
I fautori dello studio, ripreso in questa sede, hanno utilizzato una classificazione basata sulle reti neurali, ottenendo un livello di accuratezza massimo di oltre il *98%*.
D'altra parte, non essendo le reti neurali argomento del corso, nel progetto verranno utilizzate le tecniche di classificazione studiate.

Come primo passo, si applicherà l'algoritmo di classificazione *C4.5*, basato sugli alberi di decisione. L'implementazione dell'algoritmo in Weka corrisponde al classificatore *J48*.
Come prima opzione, costruiamo l'albero di decisione sul training set, il quale verrà totalmente utilizzato anche per la procedura di validazione *(opzione "Use training set")*.



In tal modo si ottiene una accuratezza del modello che raggiunge quasi il *99%*. Infatti, osservando la matrice di confusione, si può facilmente osservare come le uniche due istanze che vengono mal classificate siano due istanze di classe Divorced *(classificate come Married)*. Naturalmente, è possibile, ed è utile, osservare l'albero di decisione creato dall'algoritmo.



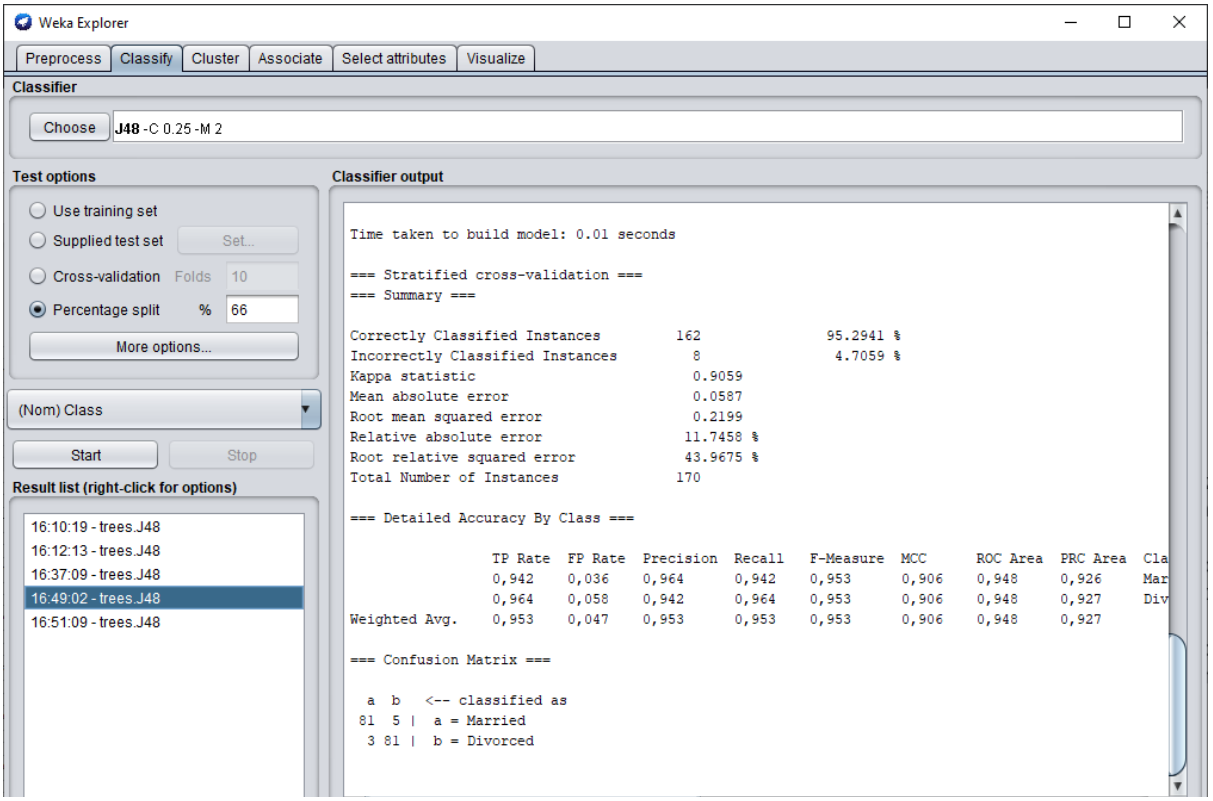
L’algoritmo utilizza solo due attributi dei 54 disponibili. Nello specifico vi è un attributo particolarmente divisivo, il numero 40, che permette facilmente di distinguere record di classi distinte.
L’attributo corrisponde alla seguente situazione:

40. We're just starting a discussion before I know what's going on.

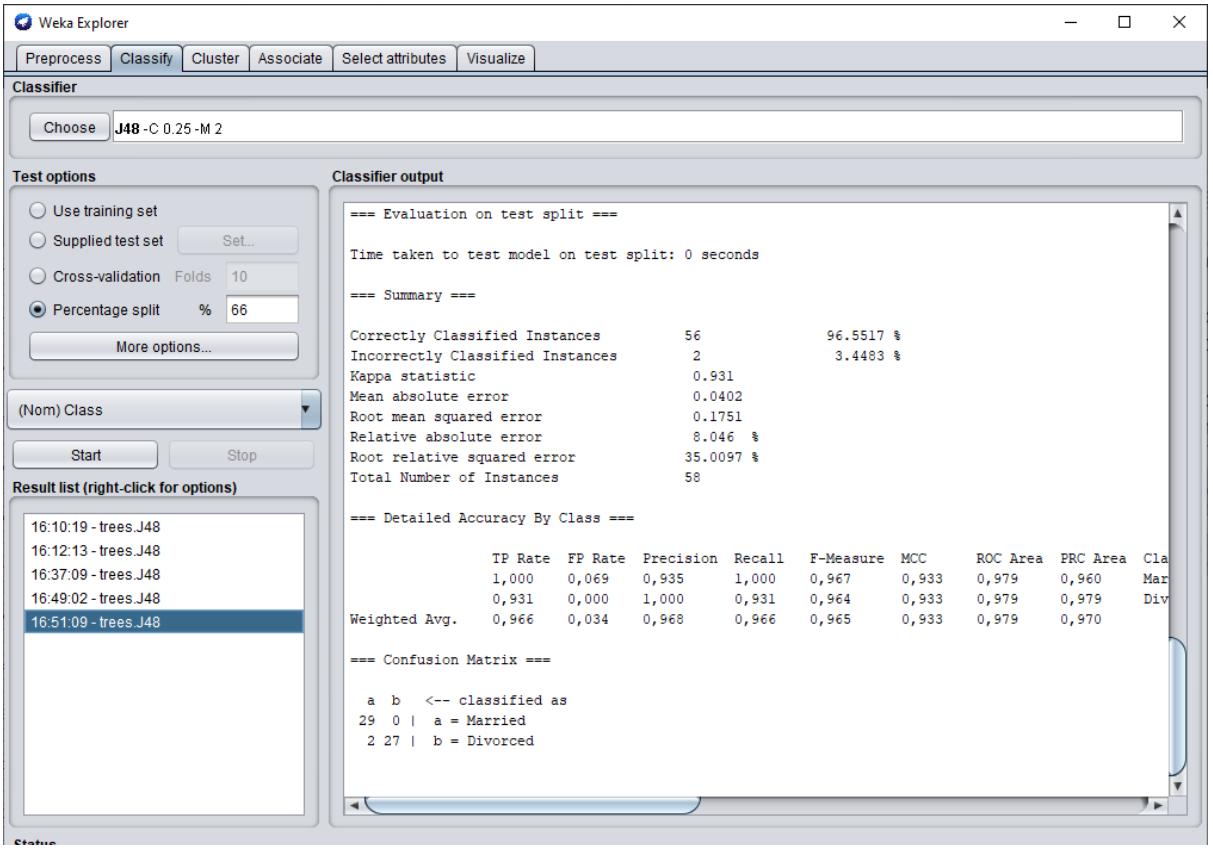
Trattandosi di questioni relative alla sfera sentimentale, dunque estremamente soggettive, è difficile dimostrare formalmente che questo sia vero.
Tuttavia, può apparire ragionevole che nel momento in cui le discussioni inizino prima che uno dei due coniugi si renda conto di ciò che stia accadendo, vi possa essere un problema di comunicazione nella coppia che potrebbe culminare in un divorzio.
Nel caso di una risposta più incerta alla domanda numero 40 (assegnando il valore due), l’algoritmo ha individuato un secondo split che può migliorare il guadagno di informazione. Lo split è stato effettuato sull’attributo numero 6, il quale corrisponde alla situazione:

6. We don't have time at home as partners.

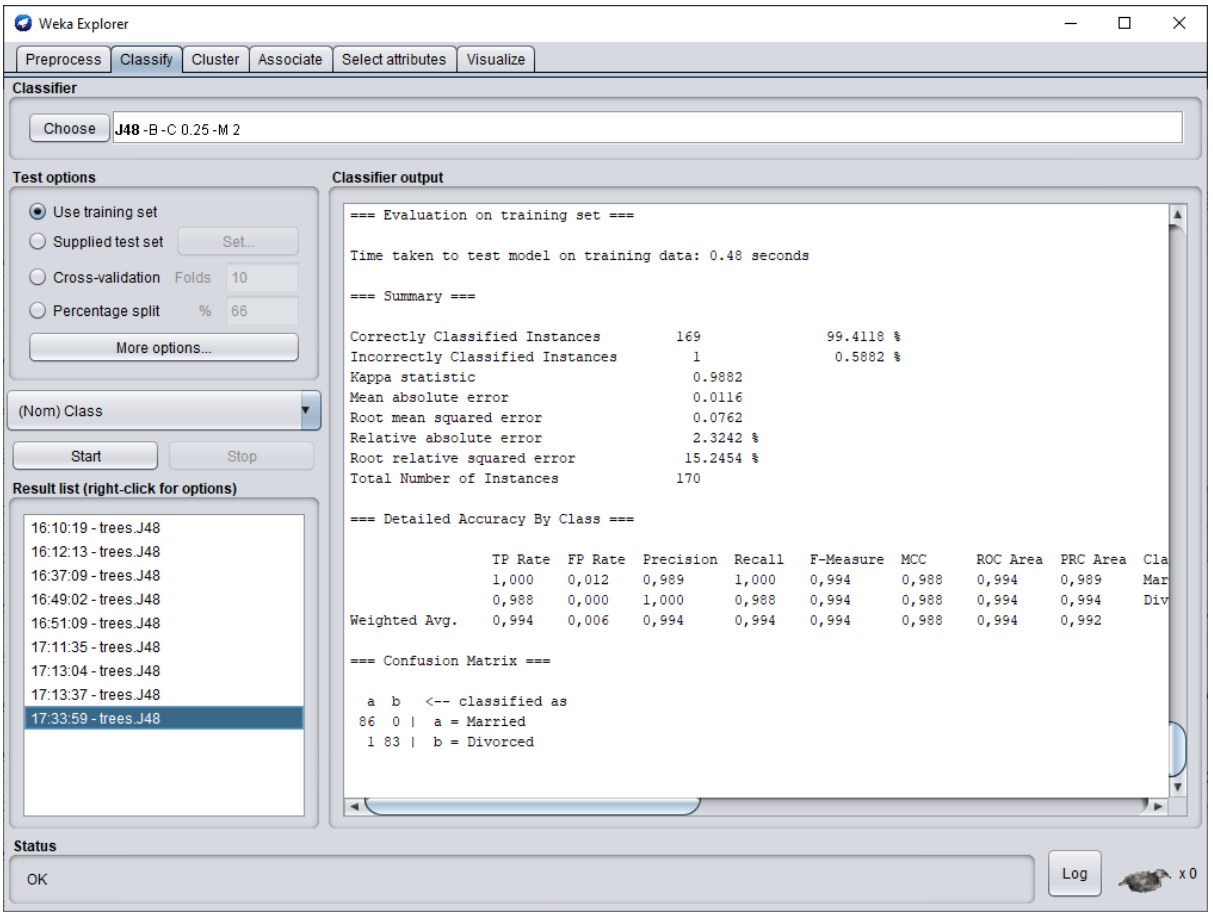
Anche in questo caso, valori degli attributi più grandi tendono verso la classe divorzio e viceversa. Come per l’attributo precedente, anche in questo caso è ragionevole il risultato trovato. Infatti, ritenendo l’affermazione come falsa, dunque avendo tempo a casa da spendere come partner, vi è una maggiore probabilità che il matrimonio rimanga intatto.
Gli errori di classificazione si sono verificati su due istanze: [5, 70].
Le due istanze, appartenenti alla classe Divorced, sono state classificate come appartenenti alla classe Married.
L’istanza 70 sembrerebbe avere un comportamento molto affine al comportamento di un oggetto di classe Divorced, con valori delle risposte tendenzialmente alti. Infatti, uno dei pochi attributi che assume valore zero in tale istanza è l’attributo 40. Questo ci fa ipotizzare che non si tratti di un outliers, ma un semplice caso particolare da non tenere troppo in considerazione.
Diverso è il discorso dell’istanza numero 5, che ha un comportamento anomalo rispetto agli altri elementi della sua classe *(tale istanza era già stata individuata come possibile outliers durante la procedura di clustering)*.
Conviene tener d’occhio il comportamento di questa istanza.
Provando ad effettuare altri tipi di validazione del modello, come la Cross Validation o la validazione su una percentuale del training set *(Percentage Split)*, si osserveranno risultati diversi.
Il modello usato per la validazione è sempre il medesimo, ma si potranno verificare differenze sull’accuratezza e, di conseguenza, nella matrice di confusione.



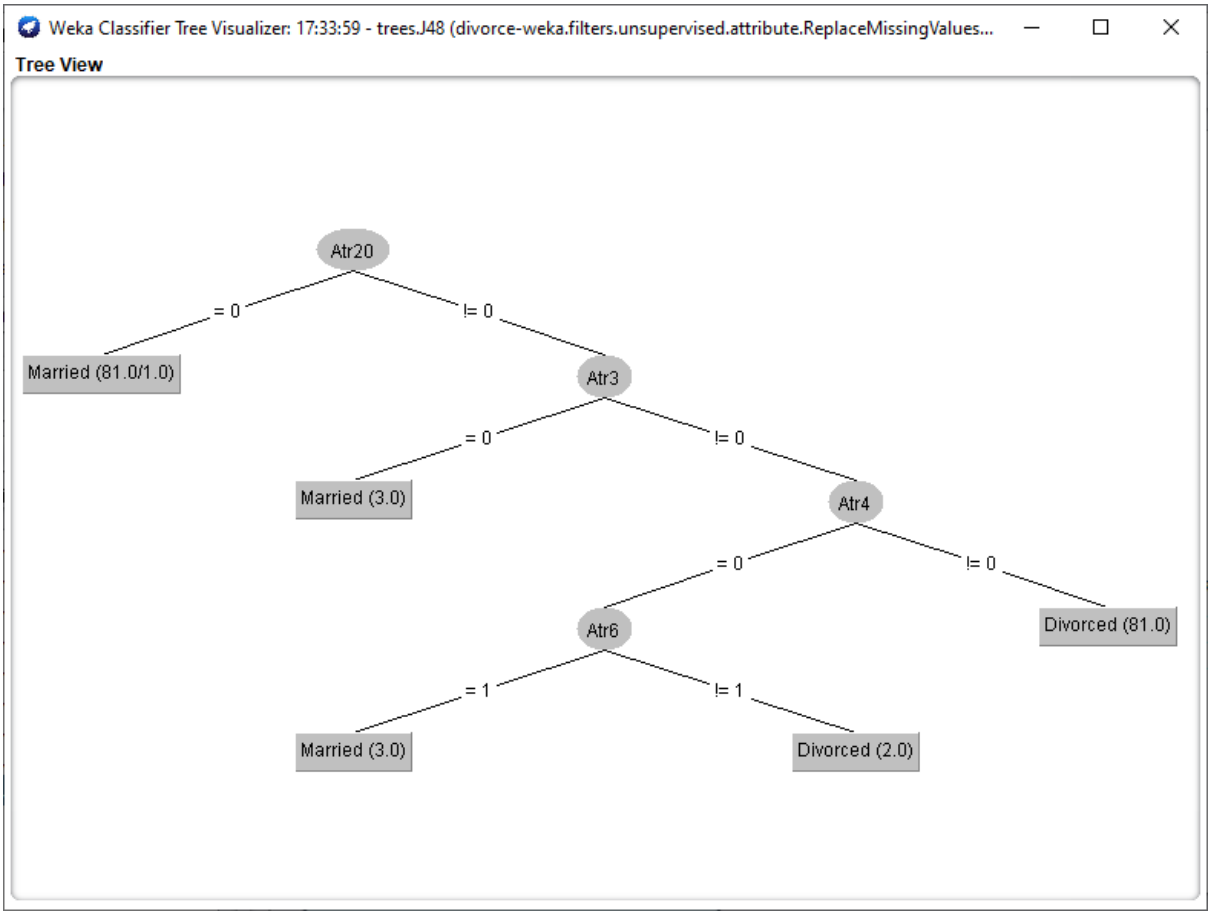
Usando la Cross Validation *(nell'immagine posta sopra)* è possibile osservare una riduzione dell'accuratezza e si verificano degli errori nella classificazione di oggetti di classe Married *(non presenti in precedenza)*.



Usando il *33%* del dataset per validare il modello, si ha una lieve riduzione dell'accuratezza, nonostante gli errori nella matrice di confusione siano dello stesso tipo e numero della prima applicazione dell'algoritmo.
In generale, si può osservare come per tutte le prove di validazione effettuate, i livelli di accuratezza del modello siano particolarmente alti *(sempre sopra il 95%)*.
Questo consente di ipotizzare che il modello costruito sui dati sia un buon modello.
Volendo provare qualche altra combinazione sui dati, si potrebbe impostare il vincolo di avere solo split binari nell'albero binario. In questo modo dovrebbe cambiare sensibilmente il modello creato.



Si ottiene in questo modo un livello di accuratezza che supera il 99%, avendo solo un errore di classificazione su 170 istanze.
Visualizzando l'albero di decisione si potrà osservare quando diverso sia dall'albero generato precedentemente.

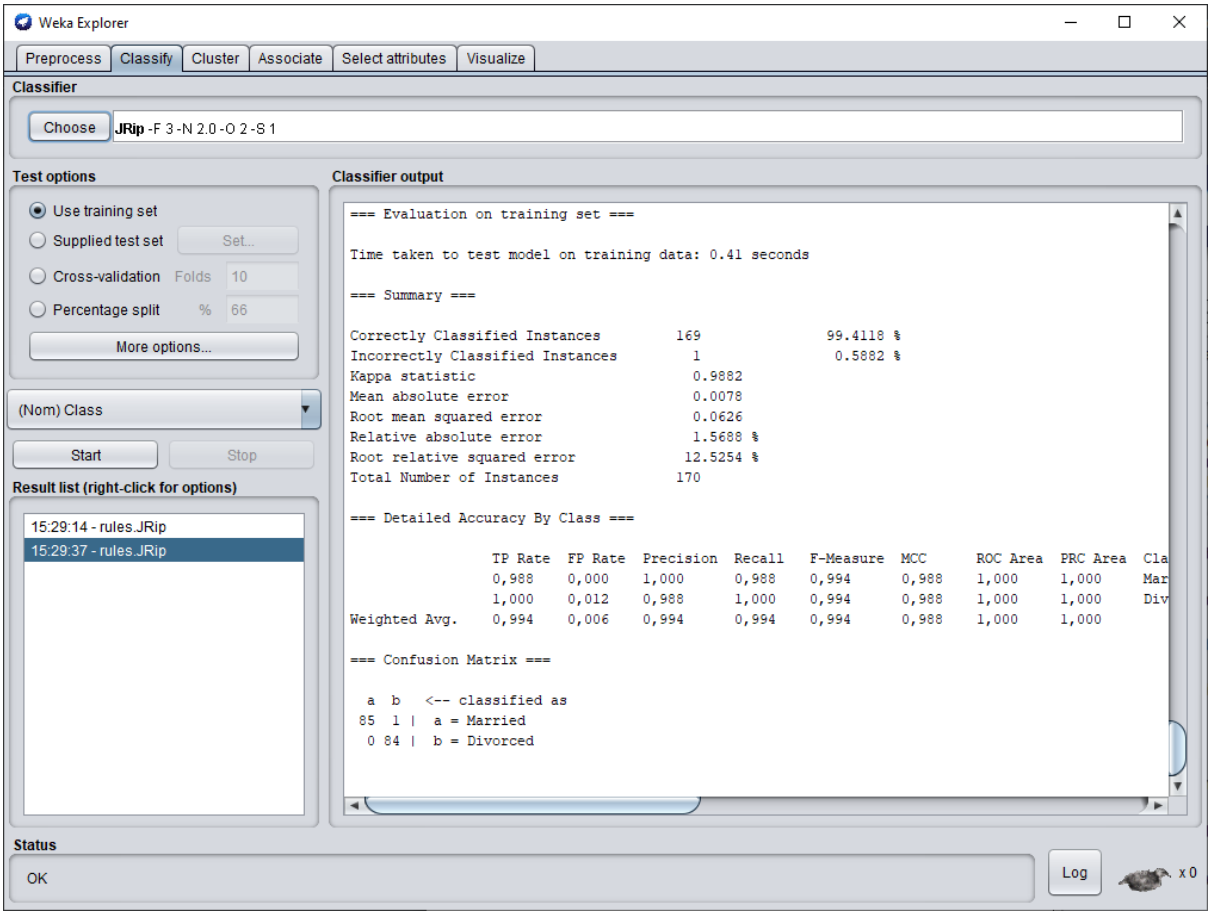


L'unico attributo che viene riutilizzato nel nuovo albero di decisione è il numero 6. Molto diversa, come prevedibile, è anche l'altezza dell'albero, data la necessità di utilizzare esclusivamente split binari.
Come osservato in precedenza, e anche durante le attività di clustering, tendenzialmente valori delle risposte tendenti allo zero implicano l'appartenenza alla classe Married, valori superiori (specie tre e quattro) implicano, tendenzialmente, una maggiore predisposizione al divorzio.

A partire dagli alberi di decisione trovati si potrebbero definire delle regole per la classificazione, come ad esempio:

- Alt20 = 0 → Married
- {(Alt20 > 0) Λ (Alt3 > 0) Λ (Alt4 > 0)} → Divorced

Tuttavia, non è l'unico approccio per la ricerca di regole sul dataset. Si può anche usare un approccio diretto, tramite l'applicazione di un algoritmo di classificazione basato su regole.
Tale approccio di classificazione è basato sulla costruzione di regole, mutuamente esclusive ed esaustive, che possano costituire un modello valido per i dati.
In Weka, è possibile selezionare diversi algoritmi di classificazione basati sulla costruzione di regole. In questo caso, verrà utilizzato l'algoritmo JRip.



Anche applicando l'algoritmo JRip sul training set si ottiene un'accuratezza estremamente alta, con un unico errore di classificazione. Infatti, un oggetto di classe Married, nello specifico l'istanza numero 143, è stato classificato come Divorced, come si può osservare dalla matrice di confidenza.

Valutato l'algoritmo, di particolare interesse è esaminare le regole che sono state generate come modello.

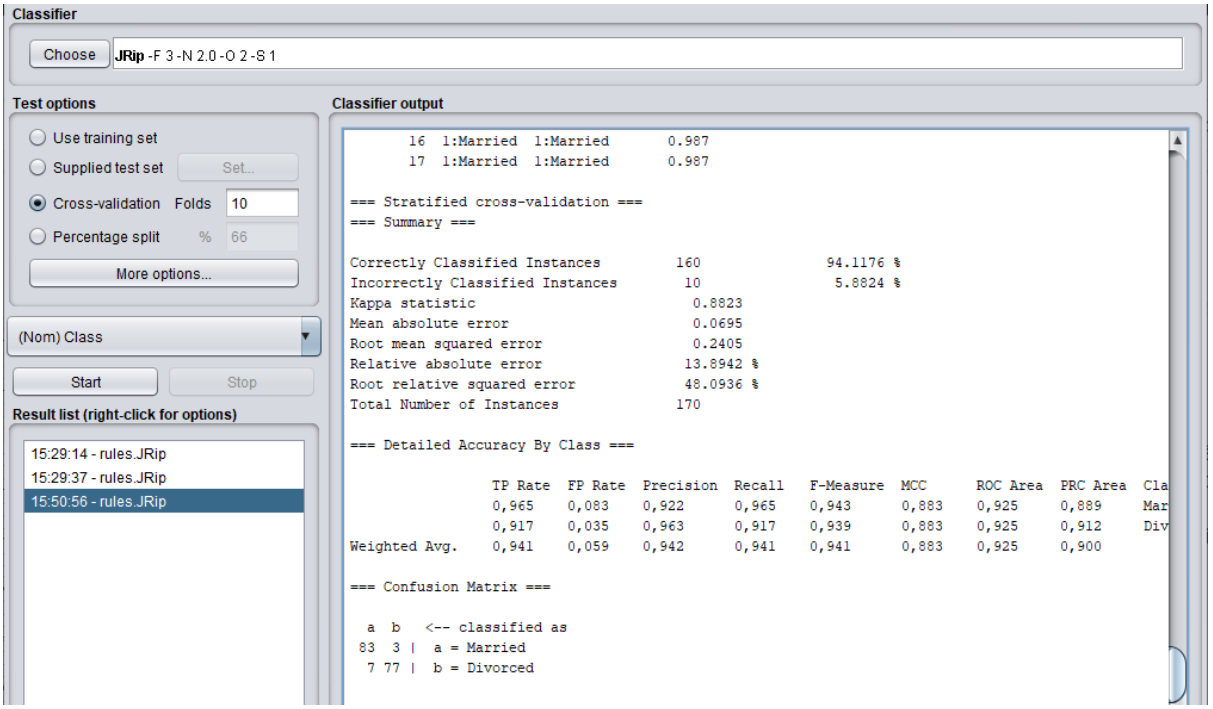
Anche le regole generate confermano la tesi secondo cui **valori più alti delle risposte corrispondano a maggiori probabilità di divorzio**.

```
JRIP rules:
===========

(Atr41 = 4) => Class=Divorced (60.0/0.0)
(Atr19 = 3) => Class=Divorced (17.0/0.0)
(Atr26 = 2) => Class=Divorced (3.0/0.0)
(Atr3 = 4) => Class=Divorced (3.0/1.0)
(Atr9 = 2) => Class=Divorced (2.0/0.0)
 => Class=Married (85.0/0.0)

Number of Rules : 6
```

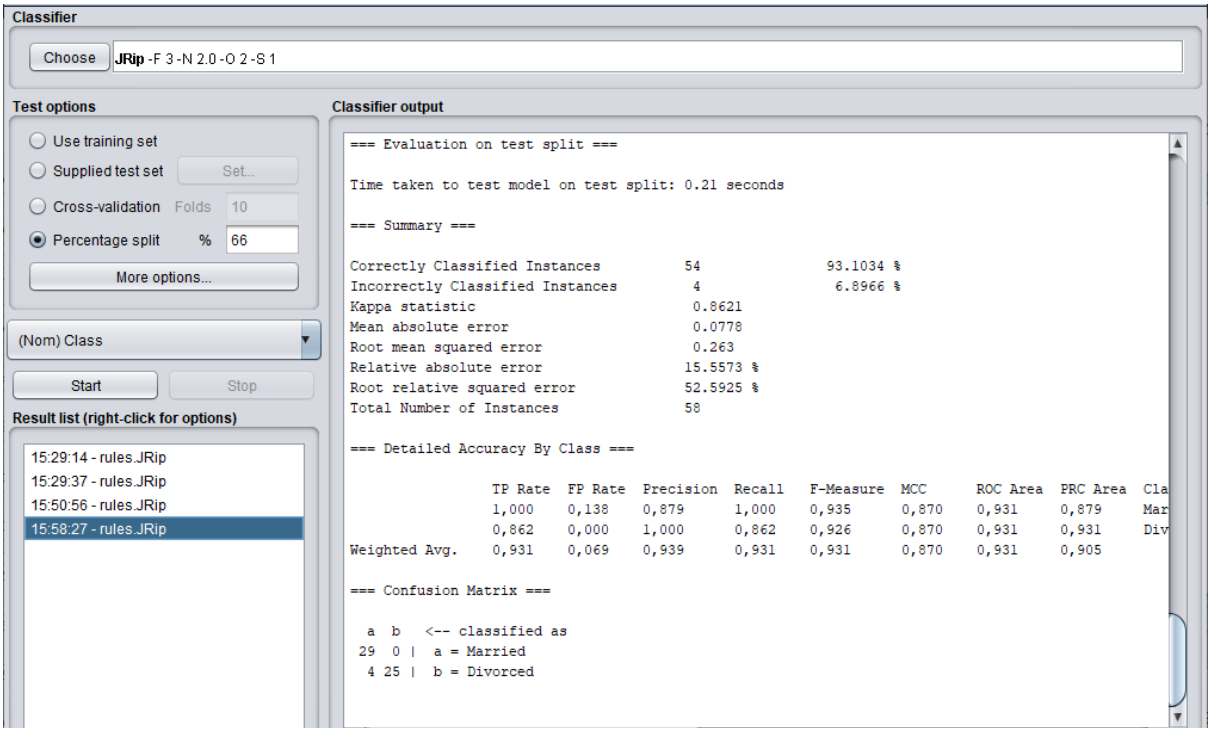
In generale si osserva come le regole generate si concentrino nel ricercare gli elementi di classe Divorced e, successivamente, qualora nessuna di queste regole sia soddisfatta, si utilizza una regola di default che assegna ai record la classe Married.

Sul training set questo modello è un ottimo modello, come osservabile anche dal valore dell'accuratezza, ma la bassa cardinalità di alcune regole potrebbe indurre ad un problema di overfitting. Sarebbe interessante validare questo modello su un test set completamente diverso, ma sfortunatamente non vi è la disponibilità. Tuttavia, si

possono utilizzare altri metodi di validazione *(visti in precedenza per la classificazione basata su alberi)*: la cross validation ed il percentage split.
In entrambi i casi, il modello sarà composto dalle medesime regole, cambierà solo il metodo di validazione.

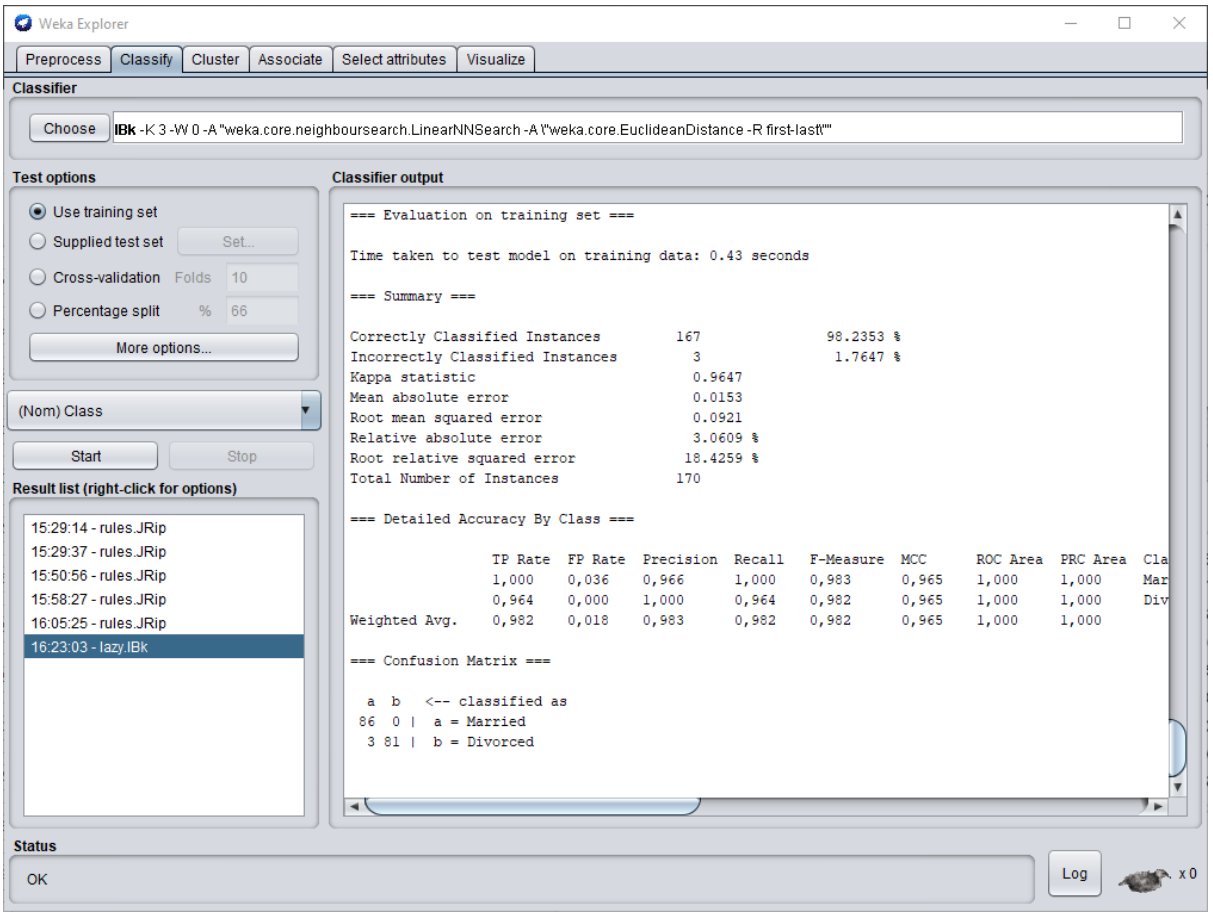


Si può già notare come, in questo caso, l'accuratezza sia scesa in maniera significativa, con diversi errori di classificazione. Nonostante il valore ottenuto sia inferiore rispetto a tutti i valori ottenuti in precedenza, si tratta comunque di un risultato ottimo.



Usando una validazione di tipo Percentage Split, con il *33%* del dataset usato come test set, l'accuratezza del modello diminuisce maggiormente.
Tuttavia, tale valore si aggira sempre intorno al *95%*, dunque le regole generate risultano essere comunque un buon modello per la classificazione.
Si sono dunque osservati diversi metodi di classificazione, tutti più o meno validi ed efficienti.

Volendo usufruire di questi dati per un ultimo test, vi si può eseguire un algoritmo di classificazione di tipo lazy. Più nello specifico, l'algoritmo studiato a lezione è il K Nearest Neighbor, applicabile in Weka sotto il nome IBK.
Di default Weka imposta il valore k, ovvero il numero di vicini da considerare per la classificazione, pari a uno. Questo valore potrebbe risultare estremamente basso, e potrebbe comportare degli errori dovuti ad un'eccessiva sensibilità al rumore.
Si assegna dunque il valore tre al parametro k.



Anche utilizzando un algoritmo di tipo lazy si ottiene un modello estremamente accurato, con un numero di errori minimo. Tali istanze, sono state già ritrovate come fonti di errori in precedenti analisi, si tratta delle istanze numero uno, cinque e sei.
Inoltre, osservando le istanze che hanno causato tali errori si può, effettivamente, notare come queste istanze si comportino in maniera anomala rispetto al pattern ritrovato in precedenza (la classe Divorced corrisponde a risposte tendenti al valore quattro).

Per esempio, si osservi l'istanza 6, che è stata classificata come Married nonostante la sua classe di appartenenza sia Divorced.

```
Instance: 6
            Atr1 : 0
            Atr2 : 0
            Atr3 : 1
            Atr4 : 0
            Atr5 : 0
            Atr6 : 2
            Atr7 : 0
            Atr8 : 0
            Atr9 : 0
           Atr10 : 1
           Atr11 : 0
           Atr12 : 2
           Atr13 : 1
           Atr14 : 0
           Atr15 : 2
           Atr16 : 0
           Atr17 : 2
           Atr18 : 1
           Atr19 : 0
           Atr20 : 1
           Atr21 : 0
           Atr22 : 0
           Atr23 : 0
           Atr24 : 0
           Atr25 : 2
           Atr26 : 2
           Atr27 : 0
           Atr28 : 0
           Atr29 : 0
```

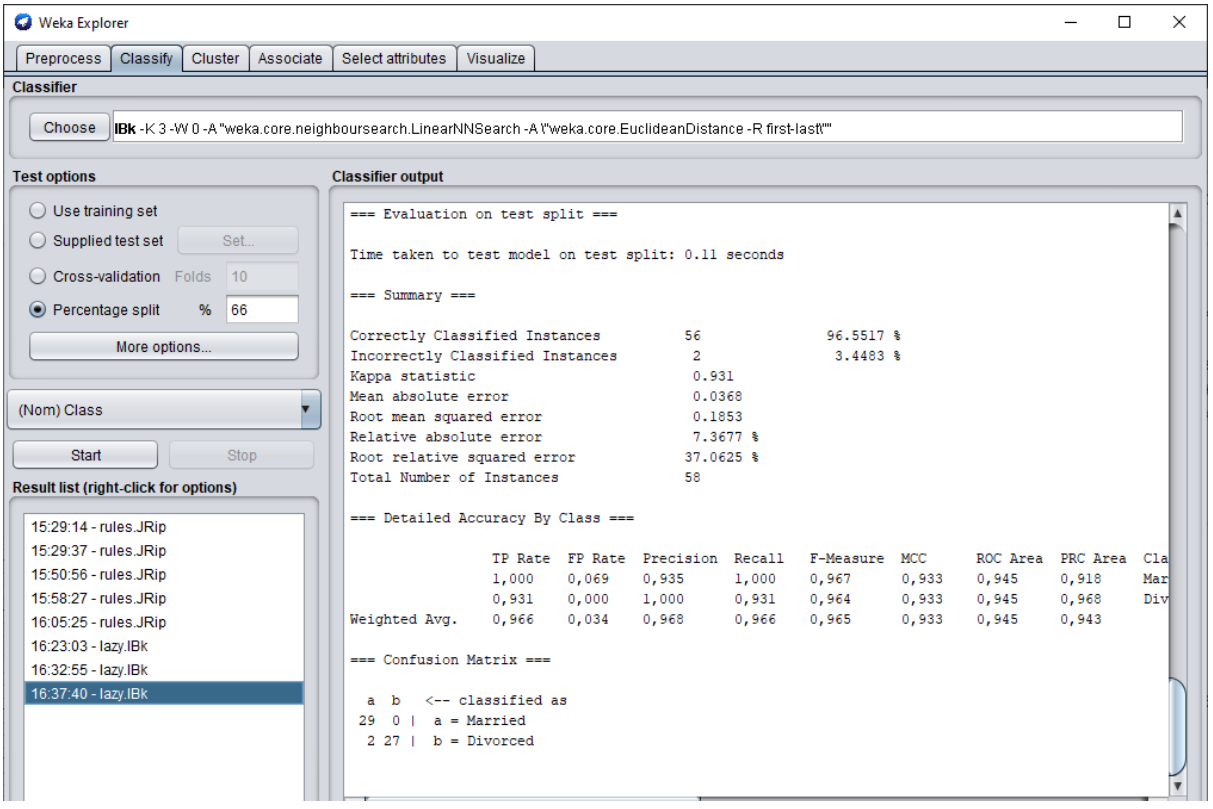
Come si osserva facilmente il suo comportamento è anomalo e potrebbe facilmente essere considerata un outliers.

Anche nel caso dell'algoritmo KNN, come fatto in precedenza, si possono utilizzare metodi di validazione differenti, per avere maggiore sicurezza riguardo la bontà di questo modello.

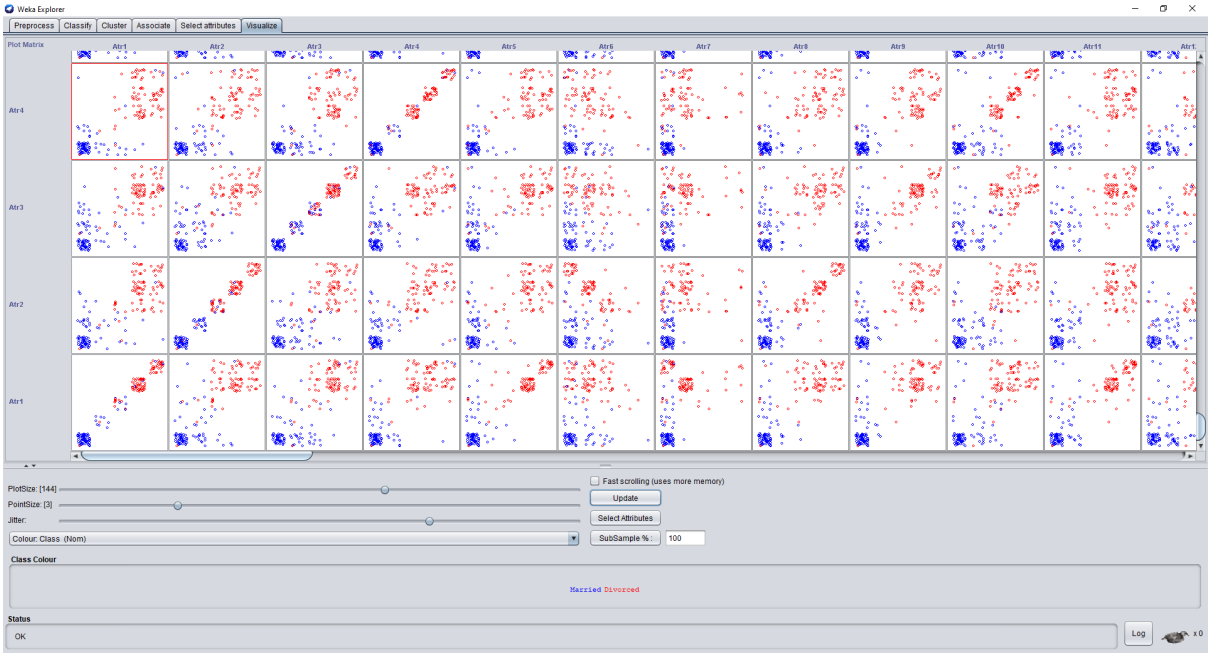
Partendo dalla Cross Validation, i risultati ottenuti non sono molti dissimili da quelli appena incontrati. Vi sono quattro errori di classificazione, tutti relativi alla classe Divorced. In basso è riportata la matrice di confusione.

```
=== Confusion Matrix ===

  a  b   <-- classified as
 86  0 |  a = Married
  4 80 |  b = Divorced
```

Anche utilizzando la validazione di tipo Percentage Split (sul 33% del training set), non si ottiene un risultato troppo dissimile da quelli precedenti. Ci si ritrova comunque con un'accuratezza estremamente alta ed un numero di errori minimo.

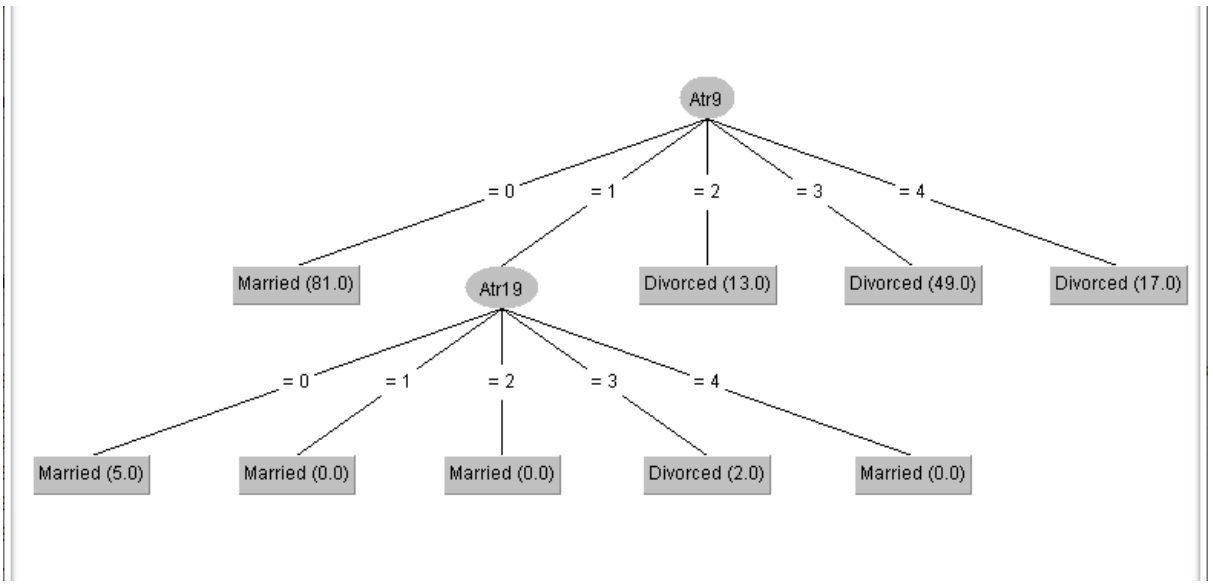


Una volta applicati i vari algoritmi, conviene visualizzare i dati nel dataset posti in grafici, tramite la sezione visualize. Dato il grosso quantitativo di attributi presenti, è impossibile riportare tutti i grafici della sezione in questa sede, tuttavia se ne proporrà un esempio.



Si può osservare come, tendenzialmente, è possibile riconoscere due gruppi distinti, uno per ogni classe. Specialmente nel caso della classe Married, identificata dal colore blu. Si nota in quasi tutti i grafici un insieme di gruppi ben visibile, anche se per gli ultimi attributi la situazione cambia leggermente.
Più frammentato è il gruppo di elementi appartenenti alla classe Divorced.
Avendo osservato il comportamento nel database nella sua interezza, sono stati identificate alcune istanze con comportamenti piuttosto anomali. Durante questo percorso si sono incontrate più di una volta, le istanze uno, cinque e sei.
Queste istanze hanno dei comportamenti estremamente anomali e di conseguenza possono essere considerate degli outlier e, di conseguenza, essere rimosse dal dataset.
Facendo ciò, si osserverà che, oltre ad eliminare alcuni errori di classificazione, come nel caso dell'algoritmo KNN, vi saranno importanti modifiche ai modelli di classificazione trovati. I maggiori cambiamenti si riscontrano nel modello di classificazione ad albero e nelle regole generate dall'algoritmo JRip.

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

JRIP rules:
===========

(Atr39 = 4) => Class=Divorced (65.0/1.0)
(Atr17 = 3) => Class=Divorced (13.0/0.0)
(Atr5 = 4) => Class=Divorced (3.0/0.0)
(Atr9 = 2) => Class=Divorced (1.0/0.0)
 => Class=Married (85.0/0.0)

Number of Rules : 5
```



Nonostante il comportamento degli attributi sia tendenzialmente il medesimo, gli attributi utilizzati sono completamente diversi rispetto a quelli visti in precedenza, sia per le regole che per l'albero.
Volendo fare un test, è possibile provare ad eliminare altre istanze, prese casualmente, del database. Rieseguendo l'algoritmo, l'albero risultante o le regole risultanti non varieranno, o avranno minime variazioni.

In generale, tutti i modelli analizzati, come anticipato, mostrano la presenza di un determinato pattern nei dati. La tendenza a rispondere alle domande con valori numerici più alti (3 - 4) è indice che la possibilità di un divorzio nella coppia possa essere imminente. Al contrario, delle risposte tendenti al valore zero indicano una maggiore stabilità nella coppia.
Assumendo che i dati forniti non contengano errori, questi modelli potrebbero essere utili per affrontare delle situazioni sentimentali difficili, ad esempio nel corso di terapie di coppia. Magari, prestando maggiore attenzione a determinati aspetti della vita coniugale, come la mancanza di comunicazione evidenziata dal primo modello di classificazione, potrebbe cambiare in meglio le vite delle persone ed evitare separazioni spiacevoli.
Questi dati potrebbero essere addirittura applicate a coppie indecise sul proprio futuro, magari evitando la formazione di coppie destinate ad una separazione.
Sarà pur vero che le questioni sentimentali sono complesse e non sempre dettate da leggi fisiche incontrovertibili, ma un piccolo aiuto da parte dell'esperienza collettiva può rendere la vita di ogni individuo migliore.