Business Intelligence per Big Data

Esercitazione n.6

L'obiettivo dell'esercitazione è il seguente:
Applicare algoritmi di data mining per la classificazione al fine di analizzare dati reali mediante l'utilizzo dell'applicazione RapidMiner.

Dati strutturati

Il dataset denominato Users (Users.xls) raccoglie dati anagrafici e lavorativi relativi a circa 1000 persone contattate da un'azienda per proporgli l'iscrizione ad un loro servizio. Per tali utenti è noto se, dopo essere stati contattati, si sono iscritti al servizio proposto oppure no (valore del campo Response). La campagna di promozione del servizio continua e il personale della compagnia deve decidere chi, tra un elenco di circa 30000 persone non ancora contattate (Prospects.xls), potrebbe essere interessato al servizio. Idealmente, per massimizzare gli incassi e minimizzare le spese, vorremmo contattare tutte e solo le persone interessate al servizio sponsorizzato.

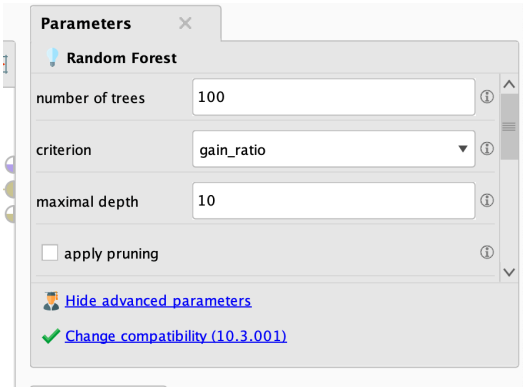
La lista completa degli attributi dei dataset a disposizione (Users.xls e Prospescts.xls) è riportata di seguito.

1. Age
2. Workclass
3. Education
4. Marital status
5. Occupation
6. Relationship
7. Race
8. Sex
9. Native country
10. Response

Obiettivo 1 – Random Forest

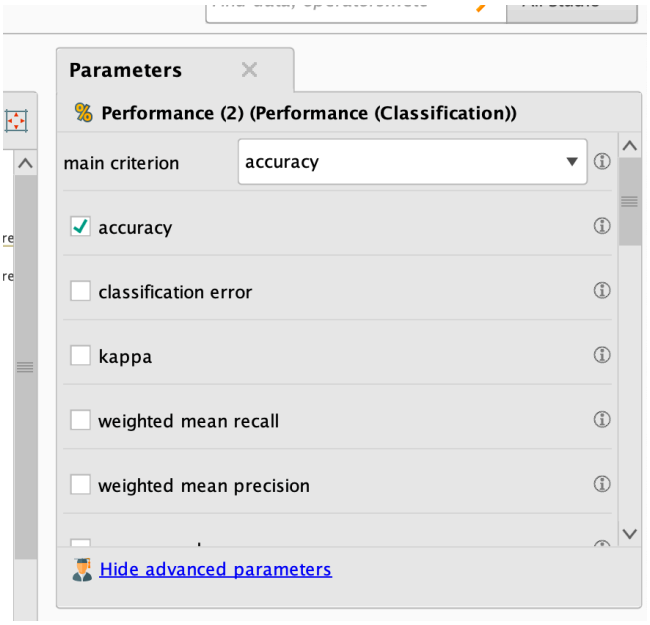
Replicare quanto fatto nel Laboratorio 5, ma con il classificatore Random Forest al posto del Decision Tree.

- Applicare i parametri "number of trees"=100 e "maximal_depth" = 10.



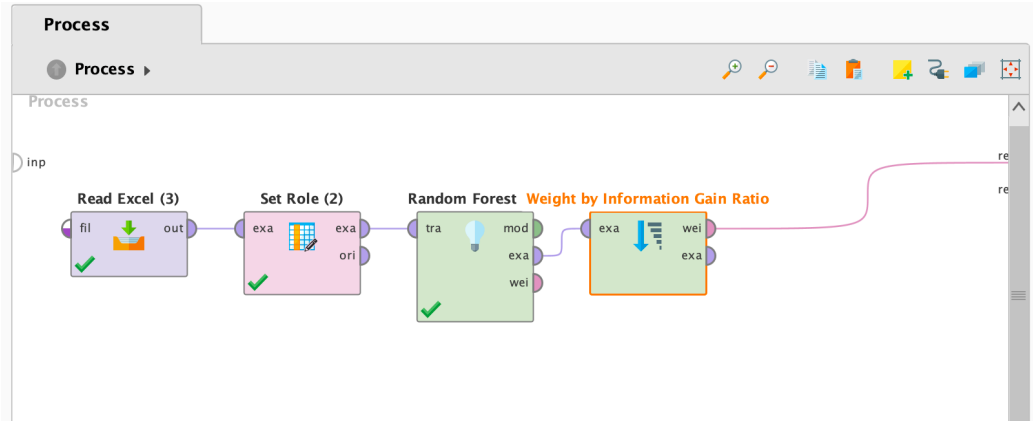
- Come variano le performance (accuracy, precision e recall) rispetto al Decision Tree?

- Per analizzare le performance, utilizzare il blocco *Apply Model* e poi *Performance* impostando come “main criterion” l’accuratezza.



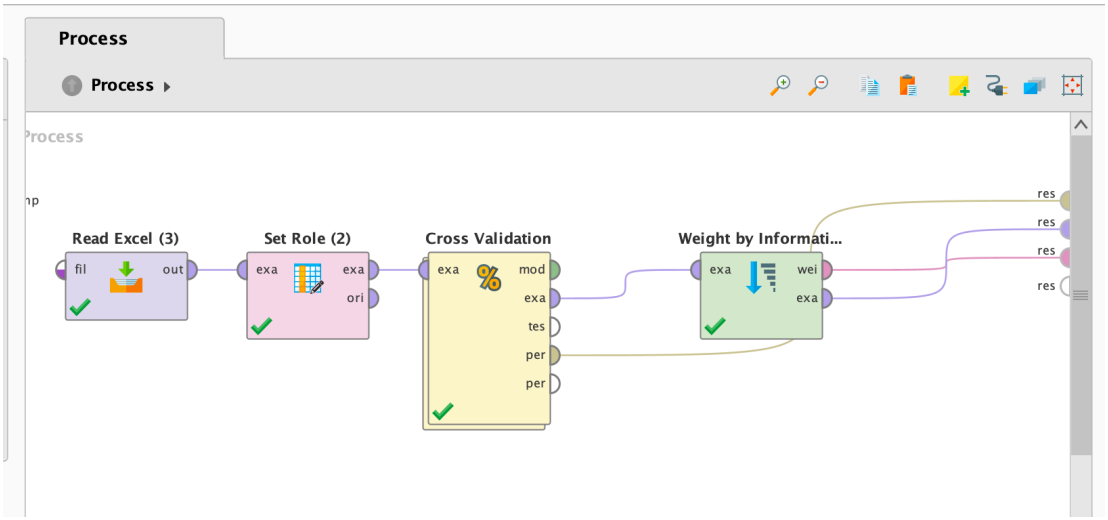
Obiettivo 2 – Analisi del Random Forest

- Analizzare l’importanza delle variabili nel modello Random Forest. Per farlo, è necessario inserire il blocco *Weight by Information Gain Ratio* con il parametro *sort direction=descending* e collegarlo come mostrato in figura. Quali sono le variabili più importanti per la classificazione?



- Ottimizzare i parametri del modello Random Forest utilizzando la Cross-validation. Come variano le performance dopo l’ottimizzazione?

- Analizzare la matrice di confusione del modello Random Forest. Quali classi vengono predette correttamente e quali no?

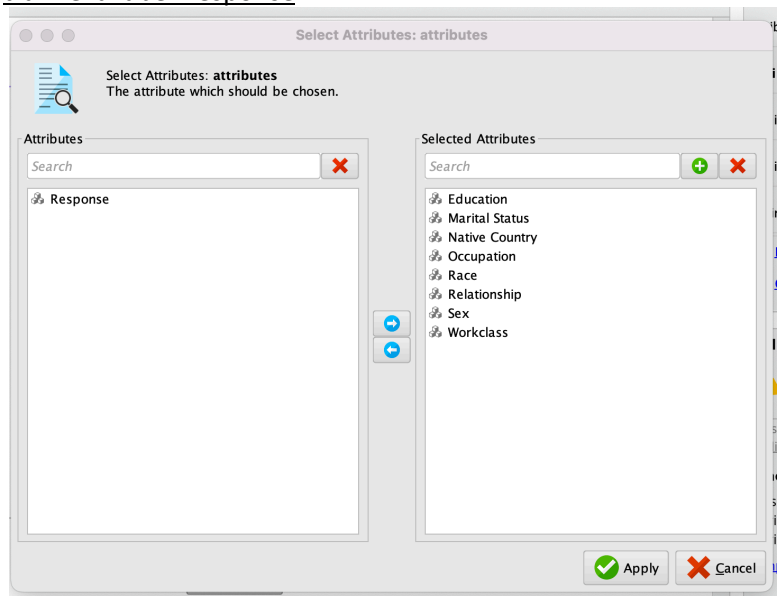


- Testare il modello Random Forest con diversi numeri di alberi [20, 50, 100, 150, 200]. C'è un numero ottimale di alberi, tra quelli proposti, che massimizza le performance del modello?

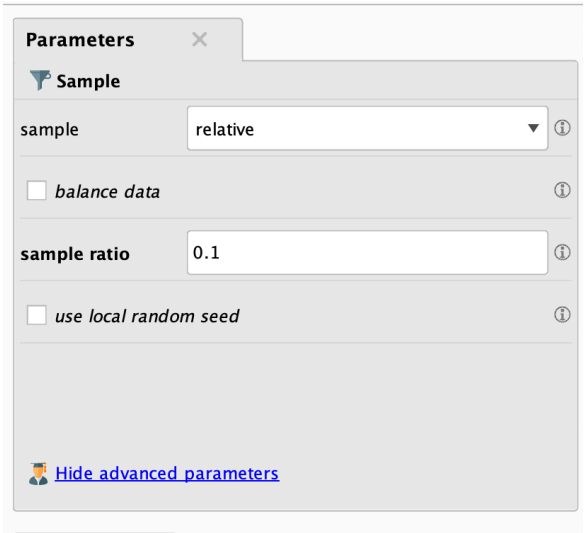
Obiettivo 3 – Support Vector Machine (SVM)

Replicare quanto fatto nell'Obiettivo 1 e Obiettivo 2, ma con il classificatore SVM al posto del Random Forest.

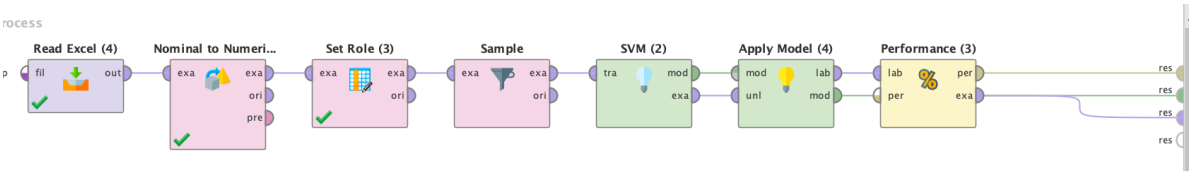
- Applicare il blocco *Nominal to Numerical*. Selezionare "attribute filter type = subset" e selezionare tutti gli attributi tranne la label Response.



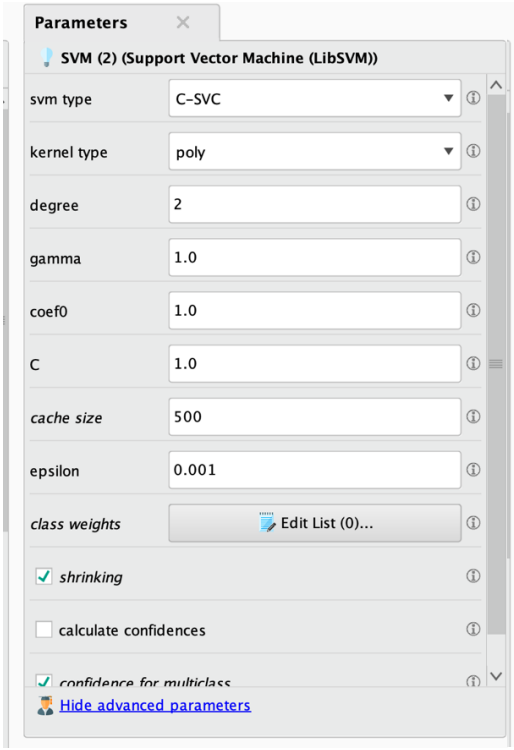
- Applicare l'operatore *Set Role* impostando Response come label.
- Effettuare un sample dei dati. Utilizzare l'operatore Sample e impostare i parametri come segue:

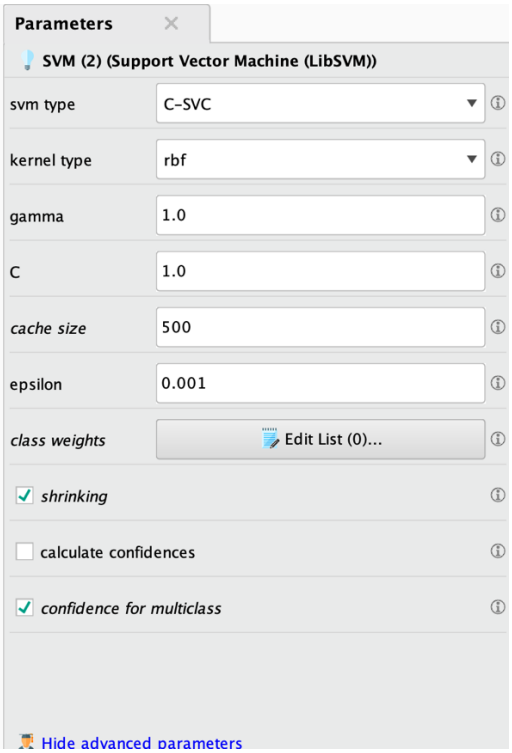


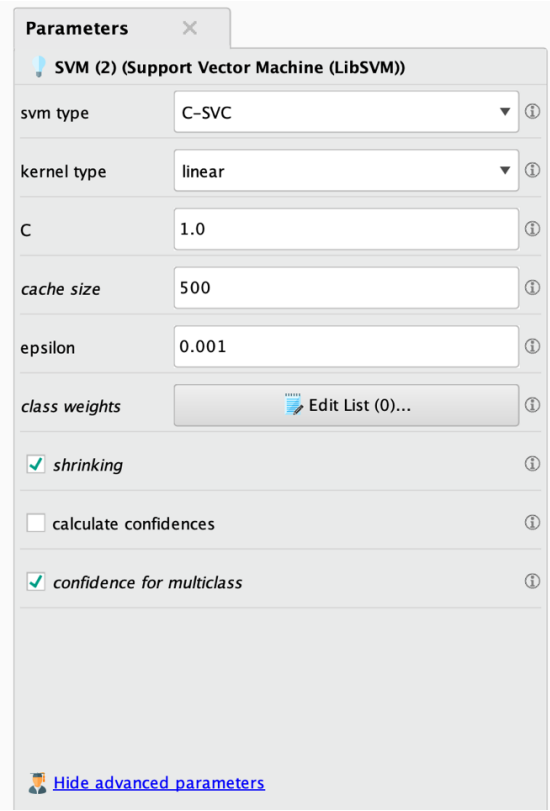
- Scegliere l'operatore *Support Vector Machine (LibSVM)*



- provare le seguenti configurazioni:



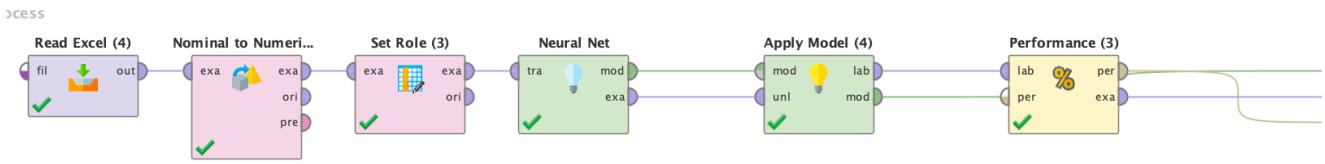




- Analizzare la matrice di confusione ottenuta nei tre casi. Quale configurazione è la migliore?

Obiettivo 5 – Reti Neurali
Applicare una rete neurale al dataset scegliendo l'operatore *Neural Net*.

- Scegliere il numero di hidden layers inserendo un hidden layer *h1* di dimensione 64 e un hidden layer *h2* di dimensione 128.
- Impostare numero di *training cycles=20, learning rate=0.001.*



- Runnare il processo e visualizzare le *Performance*.

Dalla matrice di confusione ottenuta, si può dedurre che la rete non stia funzionando come dovrebbe. È probabile che le dimensioni eccessive della rete siano la causa dei problemi, specialmente considerando la limitata quantità di dati in input.

- Modificare il numero di layer tenendo solo hidden layer h2=128.
- Runnare nuovamente il processo confrontando le performance con la precedente versione. I risultati sono migliorati?
- Provare altre configurazioni cambiando i parametri dell'operatore *Neural Net* e monitorare le performances.