INTRODUZIONE ALL'USO DI LIBRERIE DI ML

Università degli Studi di Roma Tor Vergata

C.D. Hromei 19/01/2022

OUTLINE

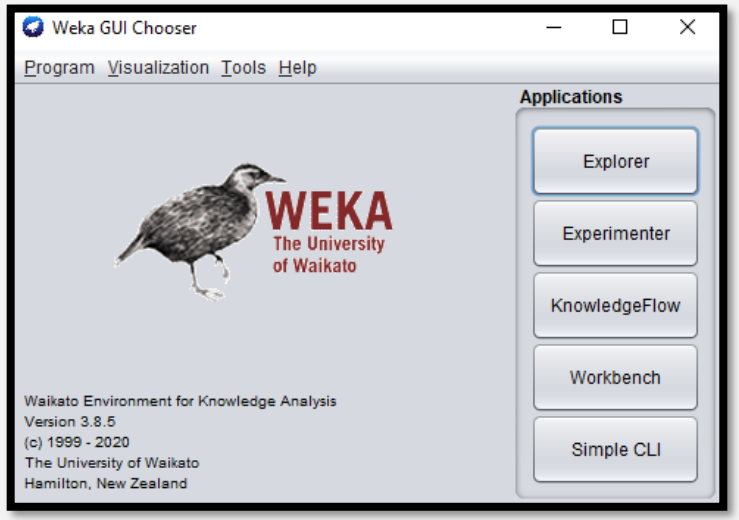
- Introduzione a WEKA
- Formalismo ARFF
- Costruzione esempi per l'addestramento
- Addestramento e visualizzazione risultati
- Introduzione alle metriche di valutazione
- Valutazione del modello

INTRODUZIONE A WEKA

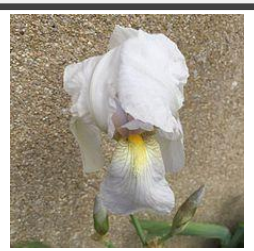
- E' un tool di Machine Learning (ML) con Interfaccia Grafica che permette di:
  - Caricare datasets
  - Trasformare l'input in una o più rappresentazioni
  - Avviare algoritmi (p.e. di addestramento)
  - Visualizzare statistiche dei datasets e dei risultati
- Ottimo inizio per coloro che si approcciano per le prime volte al ML perché pone l'attenzione più sulle procedure del ML che sulla matematica o la programmazione

GETTING STARTED

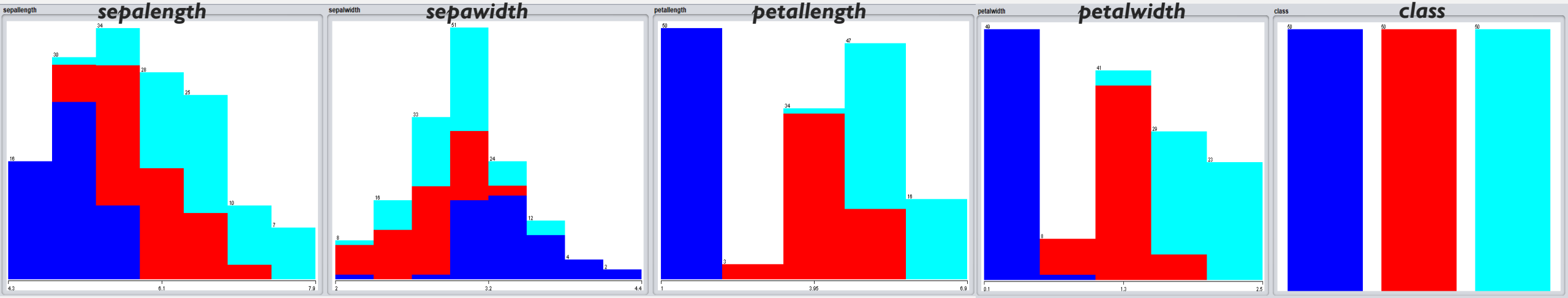
- Visitare la pagina di WEKA e scaricare la versione in base al sistema operativo
  - Per Windows è necessario scaricare anche l'ultima versione di JAVA
  - Per Mac, WEKA funziona standalone
- Avviare il tool e selezionare la voce Explorer, che utilizzeremo per addestrare un modello di ML basato sui Decision Trees



IL DATASET IRIS



- "*Open file*" permette di caricare in memoria un dataset.
  Localizzare la cartella di installazione di WEKA, entrare nella cartella "*data*" e selezionare "*iris.arff*"
- E' un dataset piuttosto famoso, spesso utilizzato nell'ambito di ML. Contiene 150 istanze con 4 attributi ciascuna (descrizione della lunghezza/larghezza, ecc) e la classe per la specie del fiore iris (una tra *setosa*, *versicolor* e *virginica*)



IL FORMALISMO ARFF

Richiede di dichiarare alcuni **campi**:

1. **Relation**: associa un nome al dataset
   a. *@RELATION <relation-name>*
2. **Attribute**: specifica il nome e il tipo del valore dell'attributo
   a. *@ATTRIBUTE <attribute-name> <datatype>*
   b. I **tipi** (*datatype*) possono essere *numeri*, *stringhe*, *date* oppure *insieme di valori*
      i. *@ATTRIBUTE sepallength NUMERIC*
      ii. *@ATTRIBUTE class {Setosa,Versicolor,Virginica}*
3. **Data**: una singola riga denota l'inizio del segmento dei dati (cioè un esempio)
   a. *@DATA*

1.4, 0.2, Setosa
1.4, 0.1, Versicolor

COSTRUZIONE ESEMPI DI ADDESTRAMENTO

- Cercate il file "*iris.arff*", apritelo con un editor (p.e. Notedpad++) e aggiungete, alla fine, nuovi esempi seguendo la sintassi *ARFF*:

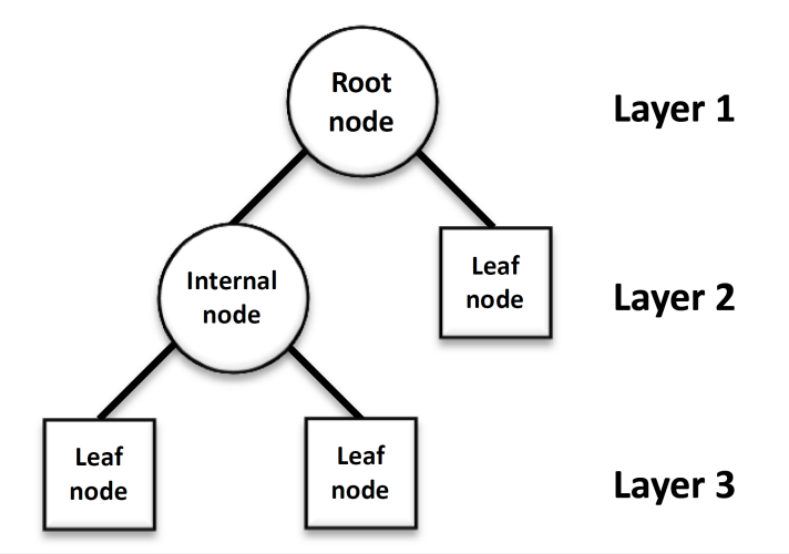
@DATA

5.1,	3.5,	1.4,	0.2,	Iris-setosa
4.9,	3.0,	1.4,	0.2,	Iris-setosa
7.0,	3.2,	4.7,	1.4,	Iris-versicolor
6.4,	3.2,	4.5,	1.5,	Iris-versicolor
6.3,	3.3,	6.0,	2.5,	Iris-virginica
5.8,	2.7,	5.1,	1.9,	Iris-virginica
	…			…
5.0,	3.1,	3.3,	0.9,	Iris-setosa

- Oppure fatelo direttamente da WEKA, premendo su "*edit*" dopo aver caricato il dataset

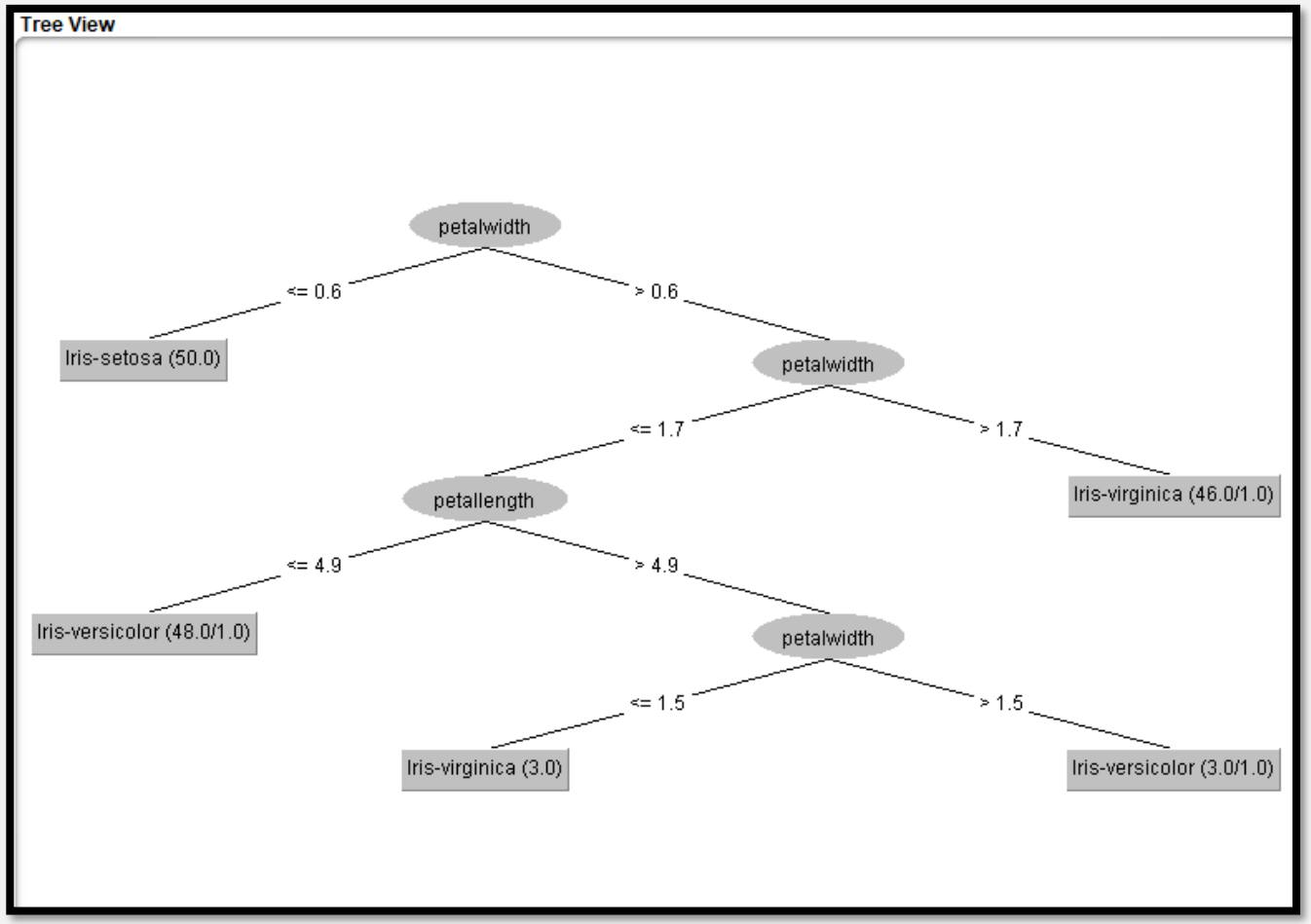
ADDESTRARE UN MODELLO

- Passare, in alto, alla scheda "*Classify*" e selezionare un algoritmo di addestramento:
  - Utilizzeremo un *Decision Tree* (*J48*, sotto la voce "*trees*") prima con cross-validation 10 (10-fold)
  - E' uno strumento di supporto alle decisioni che utilizza un modello ad albero per rappresentare una funzione ipotesi ***h*** in cui ogni nodo interno è un attributo, ogni foglia è il risultato
  - Sceglie come radice dell'albero l'attributo più discriminante, cioè che meglio divide il training set (secondo l'Information Gain)
- In ouput, WEKA produce la matrice di confusione e varie metriche di valutazione

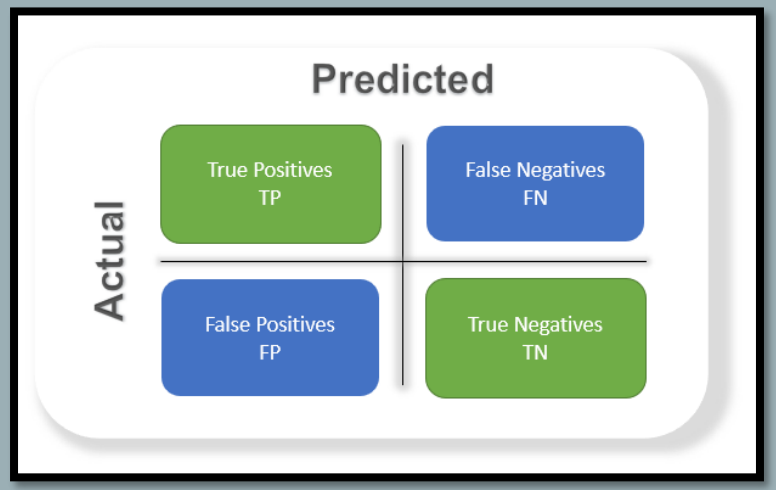


VISUALIZZARE I RISULTATI

- Click destro sul risultato nella lista di sinistra e poi selezionare "*visualize tree*"
- L'algoritmo ha trovato la feature (attributo) più discriminante: la <u>larghezza del petalo</u>
  - Minore o uguale a 0.6 → Iris-setosa
  - Maggiore di 0.6 → ho bisogno di ragionare sulle altre features
  - Questo permette di classificare, cioè assegnare una classe, a nuovi esempi nel minor tempo possibile mantenendo una certa accuratezza



INTRODUZIONE METRICHE DI VALUTAZIONE (1)



- **Matrice di confusione**: matrice N x N, con N = numero totale delle classi
  Confronta i risultati, per classe, delle predizioni con un oracolo, che fornisce la vera classe di appartenenza di un esempio. Classe *X*:
  - **TP**: esempio positivo, predizione positiva
  - **FP**: esempio negativo, predizione positiva
  - **FN**: esempio positivo, predizione negativa
  - **TN**: esempio negativo, predizione negativa
- **Accuracy**: il rapporto tra il numero di predizioni corrette e il numero totale di predizioni. Funziona bene se è presente un numero simile di campioni appartenenti a ogni classe, cioè gli esempi sono ben distribuiti. Valori vicini allo 0 corrispondono a modelli poco accurati, valori vicino a 1 invece a modelli molto accurati.

$$\frac{TP + TN}{TP + FP + TN + FN}$$

INTRODUZIONE METRICHE DI VALUTAZIONE (2)

- **Precision**: misura la precisione del modello quando classifica un esempio come positivo. Valori vicino a 1 indicano che è preciso nel distinguere tra esempi positivi e negativi, mentre valori vicini a 0 che non lo è.

$$\frac{TP}{TP + FP}$$

- **Recall**: misura quant'è accurato il modello a classificare gli esempi positivi nella classe positiva. Similmente, valori vicini a 1 indicano che è preciso nel riconoscere gli esempi positivi e valori vicino a 0 mostrano una certa confusione.

$$\frac{TP}{TP + FN}$$

- **F-measure**: media armonica di Precision e Recall. E' utile se si vuole trovare un equilibrio tra le due, ma non una semplice media aritmetica. La F-measure, infatti, penalizza il risultato se i due valori si discostano molto tra di loro (Precision = 1, Recall = 0 → F-measure = 0)

$$\frac{2}{\frac{1}{precision} + \frac{1}{recall}} = \frac{2 * precision * recall}{precision + recall}$$

VALUTARE IL MODELLO

E' necessario l'utilizzo di un Test Set che il modello non ha mai usato durante l'addestramento:

- N-fold cross-validation: è una procedura d'addestramento che divide il Dataset in *N* fold (segmenti) e addestra N modelli per poi selezionare il migliore:
  0. Dividere il dataset in N segmenti
  1. Tenere il segmento ***i*** da parte come Test Set e designare il resto come Training Set
  2. Addestrare un modello sul Training Set
  3. Calcolare una misura di valutazione (F-measure?)
  4. Se l'attuale modello è migliore del precedente, salvare il modello.
  5. Ripetere passi da 1 a 4, aumentando ***i***, fino al termine dei segmenti. Il modello salvato, sarà il migliore.

SUMMERAIZING

- WEKA è un tool di Machine Learning (ML) con Interfaccia Grafica che permette di: *caricare datasets*, *trasformare l'input*, *avviare algoritmi* (p.e. di addestramento), *visualizzare statistiche*
- Il dataset deve essere descritto utilizzando il formalismo ARFF, in cui si definiscono gli attributi e i tipi di valori (*@ATTRIBUTES*), e, infine, gli esempi del dataset (*@DATA*).
- Una volta caricato il dataset in memoria, è possibile trasformare l'input in rappresentazioni o filtrare esempi e addestrare il modello desiderato.
- Alla fine, WEKA mostra le metriche calcolate e, nel caso dei *Decision Trees*, è possibile visualizzare l'albero di decisione.
- Le metriche di valutazione più comuni sono Precision, Recall e F-Measure (media armonica tra le due)
- La procedura di selezione del miglior modello è chiamata "*N-fold cross-validation*" che addestra *N* modelli e li valuta su *N* Test Set, salvando il modello che massimizza una misura di valutazione scelta.