***Corso di Data Mining***

***Esercitazione Facoltativa propedeutica***

***per l’ammissione all’esame scritto***

**Titolo: Classificazione Multi-Classe con classificatori binari**

Tutti i problemi di classificazione multi-classe (siano C1, C2, …Ci, …CN le N classi, dove N>2,) possono essere approcciati e risolti come N problemi di classificazione binaria. In classificazione binaria si considerano solo due classi (in genere chiamate, rispettivamente, classe positiva e classe negativa). La classe positiva è quella di maggiore interesse. Come noto, in classificazione binaria ha poco senso usare la percentuale di corretta classificazione come metrica per valutare la bontà dei modelli utilizzati, in particolare quando la distribuzione delle classi è sbilanciata.

Lo studente deve risolvere i seguenti punti:

1. Scrivere un programma in Java prenda in ingresso un N file csv (contenuti in una cartella) contenenti un dataset con una sola classe (esempio C1.csv, contiene solo istanze della classe C1, ha tante righe quante le istanze e tante colonne quanti sono gli attributi che descrivono le istanze + la colonna della classe ) e restituisca N dataset binari, contenuti in N file .arff (file1, fil2, …fileN). Ciascun dataset binario, contenuto nel file i-esimo, deve avere come istanze della classe positiva tutte le istanze della classe i-esima Ci (descritta e contenuta nell’i-esimo file CSV). Le istanze della classe negativa devono essere tutte quelle delle classi restanti (contenute e descritte nei file CSV restanti). Ad esempio, si supponga di avere tre file csv (c1.csv, c2.csv, c3.csv). Il primo dataset binario avrà come istanze della classe positiva tutte quelle della classe 1 (in 1.csv) e come istanze della classe negativa quelle della classe 2 e 3 (in c2.csv e c3.csv). Il secondo dataset binario avrà come istanze della classe positiva tutte quelle della classe 2 e come istanze della classe negativa quelle della classe 1 e 3. Infine, il terzo dataset binario avrà come istanze della classe positiva tutte quelle della classe 3 e come istanze della classe negativa quelle della classe 3 e 2.
2. Costruire N modelli di classificazione utilizzando gli N dataset generati al punto 1. Valutare la bontà di ciascuno dei modelli di classificazione in cross validation (5 fold) e riportare per ciascun di essi il TPR, la precisione e la recall. Usare come classificatore il J48
3. Costruire un classificatore J48 utilizzando il dataset che si ottiene mettendo assieme tutte le istanze presenti nei file csv forniti (si crea quindi un dataset multi classe con N classi). Valutare le prestazioni, sempre in cross validation, estraendo per ciascuna classe il TPR, la precisione e la recall. Confrontare i risultati ottenuti con quelli ottenuti nel punto 2).
4. Ripetere il punto 2) ribilanciando la classe positiva usando il filtro SMOTE. Ripetere quindi i confronti con il classificatore del punto 3).
5. Ripetere il punto 2 usando due metaclassificatori che inglobino ciascuno un metodo diverso di attribute selecion (infogain+ ranker (valore infogain >0) e cfsSubsetEval+ best first) e il filtro SMOTE come al punto 4. Per ogni classe (quindi per ogni modello, identificare quali sono le caratteristiche selezionate).

I punti 2-5 devo essere svolti scrivendo dei programmi Java utilizzando le API di Weka.

Lo studente deve consegnare una dettagliata relazione in cui vengono descritti tutte le scelte effettuate, i passi realizzati, i risultati ottenuti (**importante organizzarli in tabelle e discuterli**), discussioni e considerazioni finali. Tutti i codici, gli script e i file prodotti devono essere consegnati e discussi in dettaglio nella relazione.