Alberi di decisione con dati mancanti

Descrizione elaborato:

Questo elaborato testa l'algoritmo degli alberi di decisione con dati mancanti per tre dataset diversi. Ogni dataset è diviso in due parti: una parte di allenamento e una di test (circa il 70% di istanze nella prima e il 30% nella seconda). Per ogni dataset vengono costruiti quattro alberi di decisione: uno a partire dal dataset di allenamento completo, uno con il 10% di dati mancanti, un altro con il 20% di dati mancanti e infine uno con il 50% di dati mancanti. Successivamente, per ogni albero, viene eseguita la classificazione su un dataset di test e vengono stampate le percentuali delle classificazioni corrette (con classificazione corretta si intende una foglia che classifica correttamente l'etichetta dell'istanza con probabilità > 74%).

Descrizione algoritmo:

Questo algoritmo costruisce un albero per individuare la classe di appartenenza (etichetta) di una istanza a partire da un dataset di allenamento. L'albero è formato da due diversi tipi di nodi: i nodi di decisione e le foglie. I nodi di decisione sono composti da una domanda e due puntatori ai figli: quello che risponde True alla domanda e quello che risponde False. Le foglie invece classificano il dato inserito.
Il criterio secondo cui viene fatta una certa domanda, e dunque fatta la divisione nei due rami True e False, dipende dalla funzione info_gain. Per qualsiasi domanda possibile viene calcolata l'impurità dei rami True e False grazie alla funzione gini, successivamente viene calcolato il guadagno di quella suddivisione tramite proprio la funzione info_gain. Il nodo sarà determinato dalla domanda con il guadagno maggiore.
Per gestire i dataset con dati mancanti è stata implementata la funzione data_fill la quale calcola la probabilità dei valori che può assumere ogni attributo e successivamente assegna casualmente un valore agli attributi non determinati a seconda della distribuzione di probabilità calcolata precedentemente.
Inoltre per evitare il problema dell'overfitting, molto usale per questo tipo di

algoritmo, ho deciso che se per un nodo non esiste alcuna domanda che ha un guadagno maggiore di 0.005 allora tale nodo diventa una foglia.

Risultati:

Tabella classificazioni

Tabella classificazioni	0%	10%	20%	50%
Chess	96%	94%	89%	78%
Balance	74%	70%	60%	44%
Nursery	83%	82%	79%	75%

Note sulla tabella delle classificazioni: Nella prima colonna sono inseriti i nomi dei dataset utilizzati. Nella prima riga sono inserite le percentuali di dati mancanti dei dataset durante la costruzione dell'albero di decisione. Gli elementi all'interno della tabella indicano le percentuali di classificazione corrette effettuate dagli alberi di decisione.

Analisi:

Analizzando la tabella dei risultati notiamo che le prestazioni dell'algoritmo sono diverse tra i vari dataset. Questo perché i dataset hanno un numero differente di istanze, di attributi e di grandezze di dominio. Il dataset con la percentuale migliore di classificazioni è Chess che ha il maggior numero di attributi e quello con il dominio della label minore. Il dataset di Nursery ha delle buone percentuali grazie soprattutto al maggior numero di istanze (più di 10.000). Il dataset Balance a causa delle poche istanze (600 circa) ha una precisione peggiore.
Per quanto riguarda l'andamento delle prestazioni della classificazione all'aumentare dei dati mancanti si nota che tutti i dataset peggiorano ma quelli con più istanze lo fanno meno.