Inteligência Artificial

2º Projeto

Grupo 12

Guilherme Saraiva - 93717

Sara Ferreira - 93756

Introdução:

Neste relatório pretendemos analisar a solução proposta para o segundo projeto da unidade curricular de Inteligência Artificial do primeiro semestre do ano letivo de 2020/2021. O problema apresentado tem por objetivo estudar o comportamento de árvores de decisão a partir de conjuntos de dados.

Análise do algoritmo base:

Para a criação da árvore de decisão utilizámos o algoritmo lecionado nas teóricas – *Decision Tree Learning* (DTL). Este algoritmo utiliza uma abordagem *"top-down greedy"* para a construção da árvore. Esta abordagem significa que a árvore vai ser construída pelo topo e a cada chamada recursiva é escolhido o melhor atributo para criar os seus nós. Para a escolha do melhor atributo medimos a quantidade, em bits, da informação oferecida pelo atributo. Para este efeito, calculou-se a quantidade de incerteza numa distribuição de probabilidade – entropia. O melhor atributo é aquele com maior quantidade de informação.

Contudo, apesar deste algoritmo ser eficiente para a construção de árvores de decisão, o DTL não sabe lidar com os problemas de ruído e de *overfitting*.

Resolução do problema do ruído:

O problema do ruído acontece quando exemplos com o mesmo atributo têm diferentes atributos.

Para resolver esse problema simplesmente criámos uma condição no DTL em que verificávamos se a lista de atributos estava vazia e, caso estivesse, retornava–se a classificação mais frequente dos respetivos exemplos. Desta forma, cada nó vai possuir uma classificação por maioria para o seu conjunto de exemplos.

Resolução do problema das árvores curtas (*overfitting*):

Este problema surge quando a árvore é construída de modo a que se ajuste a todas as amostras do conjunto de dados. Assim, a árvore perde capacidade de generalização criando ramificações desnecessárias.

Para solucionar este problema recorremos ao *pruning* da árvore. Este *pruning* foi executado em duas fases. Na primeira fase, verificou-se recursivamente se cada nó tinha dois filhos iguais, e caso isso se verificasse, faz se o *prune* do nó ficando o filho no lugar desse mesmo nó. Na segunda fase, averiguámos se os filhos de cada nó possuíam o mesmo atributo e, em caso afirmativo, ajustámos a árvore com as respetivas rotações caso possuíssem ramos iguais.

Análise critica dos resultados:

Após terminar o DTL, observou-se que para conjuntos de dados que não possuíssem ruído o erro era de 0 enquanto que os conjuntos que apresentavam ruído tinham um erro bastante elevado. Este erro diminuiu consideravelmente com a resolução do problema do ruído descrito anteriormente.

Foi também de notar que após resolver o *overfitting*, a árvore resultante ficou muito mais compacta e menos redundante.

Referências:

https://fenix.tecnico.ulisboa.pt/downloadFile/1689468335655514/cap18-ArvoresdeDecisao.pdf

https://scikit-learn.org/stable/modules/tree.html#tree

https://en.wikipedia.org/wiki/Decision_tree_pruning

http://athena.ecs.csus.edu/~mei/177/ID3_Algorithm.pdf

https://cs.adelaide.edu.au/~dsuter/Harbin_course/DecisionTrees.pdf