U.PORTO

FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

Trabalho Prático nº3-
Decision Trees

Ricardo Araújo Amorim, up202107843
David Rafael Pereira Nogueira, up202108293
Pedro Morim Figueiredo Andrade Leitão, up202107852

Porto
2023

U.PORTO
FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

Índice

Introdução

Árvores de Decisão são algoritmos supervisionados não parametrizados, utilizados para procedimentos de classificação e regressão. É composta por um sistema de nós cujo representam testes em atributos, arestas que representam os possíveis resultados dos mesmos testes e folhas que representam os valores de saída. A estrutura hierárquica da árvore permite que sejam tomadas decisões sequenciais, seguindo os caminhos determinados pelos testes em cada nó.

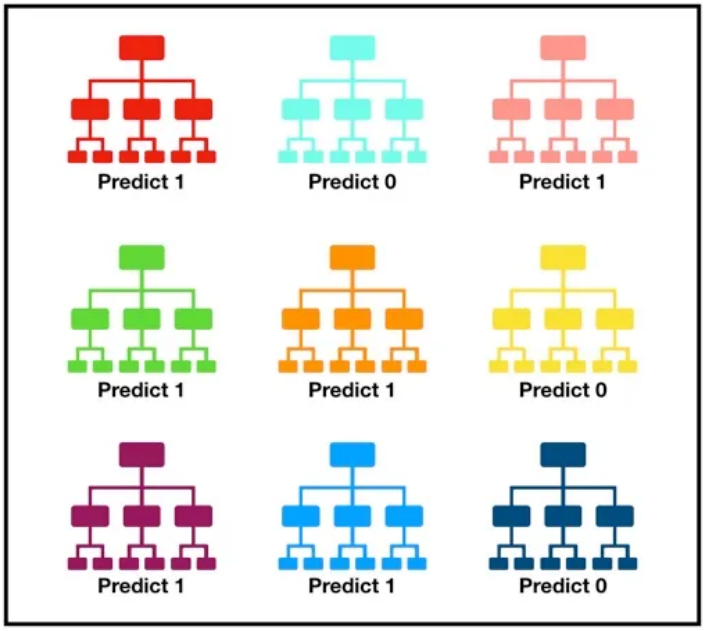
Algoritmos para Indução de Árvores de Decisão

Variações da árvore de decisão

Para além da árvore de decisão, existem outros algoritmos conhecidos que são semelhantes a este algoritmo:

Random Forest

Este tipo de algoritmo consiste em agrupar um grande número de árvores de decisão. Depois em cada uma das árvores envia um resultado usando os dados. É depois selecionado aquele cujo valor foi mais repetido por entre as árvores.

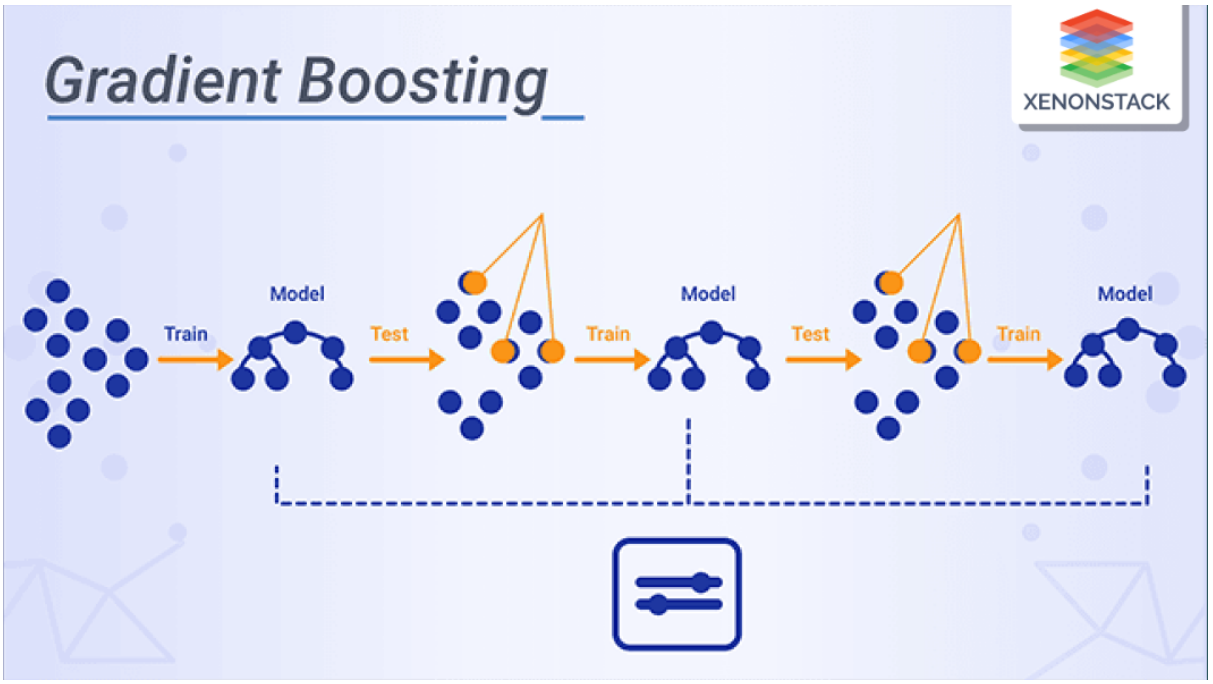


Tally: Six 1s and Three 0s
Prediction: 1

Gradient Boosting

Este algoritmo é determinado pela intuição que o próximo melhor modelo, combinado com os modelos anteriores, minimiza em média o erro de predição. Ou seja, sempre que é testado, é usado todos os modelos anteriores ao atual.

Tal como a Random Forest, este é considerado um algoritmo de aprendizagem em conjunto, no entanto, a diferença é a origem das árvores e como são construídas.



Extreme Gradient Boosting (XGBoost)

Uma versão mais otimizada e mais popular do que o anterior devido à formalização de controle de *over-fitting* , utilizando recursos como regularização, manipulação de valores ausentes e função de perda personalizada, dando melhor performance.

LightGBM

Outra versão de *Gradient Boosting* mas, no entanto, adota uma estratégia de crescimento por folha, em que cada árvore cresce selecionando a melhor divisão em relação ao ganho de informação por folha.

Diferentes métricas utilizadas para selecionar os atributos a colocar na árvore

Ganho de informação (Information Gain): Esta métrica, usada pelo ID3, mede a quantidade de informação ganha ao dividir os dados com base em um atributo em específico. A ideia principal é selecionar o atributo que resulta na maior redução de entropia dos dados.

Índice Gini (Gini Index): Esta métrica é usado no algoritmo CART (Classification and Regression Trees). Ele mede a probabilidade de classificação incorreta de um exemplo escolhido aleatoriamente, se esse exemplo for rotulado aleatoriamente de acordo com a distribuição de classes do nó. Atributos com menor índice Gini são considerados mais importantes.

Ganho de Razão (Gain Ratio): O ganho de razão é uma extensão do ganho de informação utilizado no algoritmo C4.5. Ele ajusta o ganho de informação pelo número de valores possíveis do atributo, para evitar viés em favor de atributos com muitos valores. O ganho de razão leva em conta a quantidade de informação fornecida pelo atributo em relação à sua complexidade.

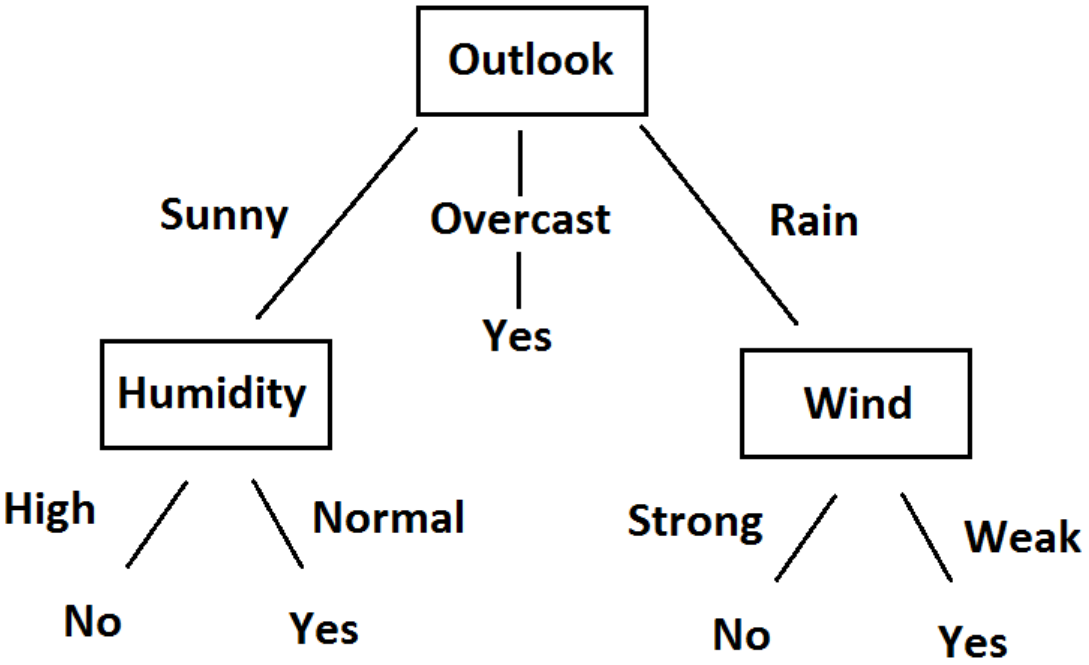
ID3 algorithm

O algoritmo ID3 é um dos algoritmos de árvore de decisão mais amplamente utilizados. Ele segue uma abordagem top-down e greedy para construir árvores de decisão com base no conceito de entropia.

1. **Calcula a entropia da variável alvo:** A entropia é uma medida de impureza ou aleatoriedade na variável alvo.
2. **Calcula o ganho de informação para cada atributo:** O ganho de informação representa a quantidade de informação obtida ao dividir os dados com base em um atributo específico.
3. **Seleciona o atributo com o maior ganho de informação:** O algoritmo escolhe o atributo que maximiza o ganho de informação como o próximo nó da árvore.
4. **Cria um ramo para cada valor do atributo selecionado:** Para cada valor possível do atributo escolhido, o algoritmo cria um ramo na árvore.
5. **Recursivamente**, o algoritmo repete os passos anteriores para cada ramo criado, considerando apenas as instâncias correspondentes.
6. **Repete** até que todas as instâncias sejam classificadas corretamente ou não haja mais atributos disponíveis.

Limitações do ID3

1. **Sensibilidade a atributos com muitos valores distintos:** O ID3 não lida bem com atributos que possuem muitos valores distintos, pois tende a criar árvores profundas e complexas, levando a um possível overfitting.
2. **Ausência de poda (pruning):** O ID3 não realiza poda na árvore gerada, o que pode levar a um desempenho inferior quando aplicado a conjuntos de dados de teste.
3. **Sensibilidade a ruído e dados inconsistentes:** O ID3 é sensível a ruído nos dados de treinamento e pode produzir árvores de decisão menos eficazes quando os dados contêm erros ou inconsistências.



Implementação

Linguagem

Para implementar este problema, usamos a linguagem Java. Escolhemos esta linguagem pois é uma linguagem que todos os membros do grupo estão confortáveis (devido à cadeira de Estruturas de Dados do semestre anterior).

Estruturas de dados

CSV: Lista de Listas da biblioteca do java (List<List<String>>). Decidimos representar assim o csv pois é mais fácil manipular a lista e adicionar e remover elementos que um array normal.

DecisionTree: Nodes, implementado por nós. Escolhemos o node para representar a tree pois achamos que era o mais simples para a representar.

Organização do código

Temos dois ficheiros: ID32.java e DNode.java. Como o nome diz o ID32.java implementa o algoritmo ID3 e o DNode.java implementa o DNode (usado para representar a àrvore).

No ID32.java temos a class ID32 para representar o algoritmo com diversas funções:

- id3 (faz o algoritmo em si)
- bestAtrtribute (escolhe a coluna com maior infogain)
- getMostCommonValue (dá o valor mais comum da classe)
- values (guarda os diferentes valores de uma determinada coluna)
- entropy (calcula a entropia)
- infogain (calcula o infogain)
- removeColumn (remove uma determinada coluna)
- MakeChildren (faz um subCSV que contêm apenas as rows com um determinado valor de uma coluna)
- printTree (dá print à tree)

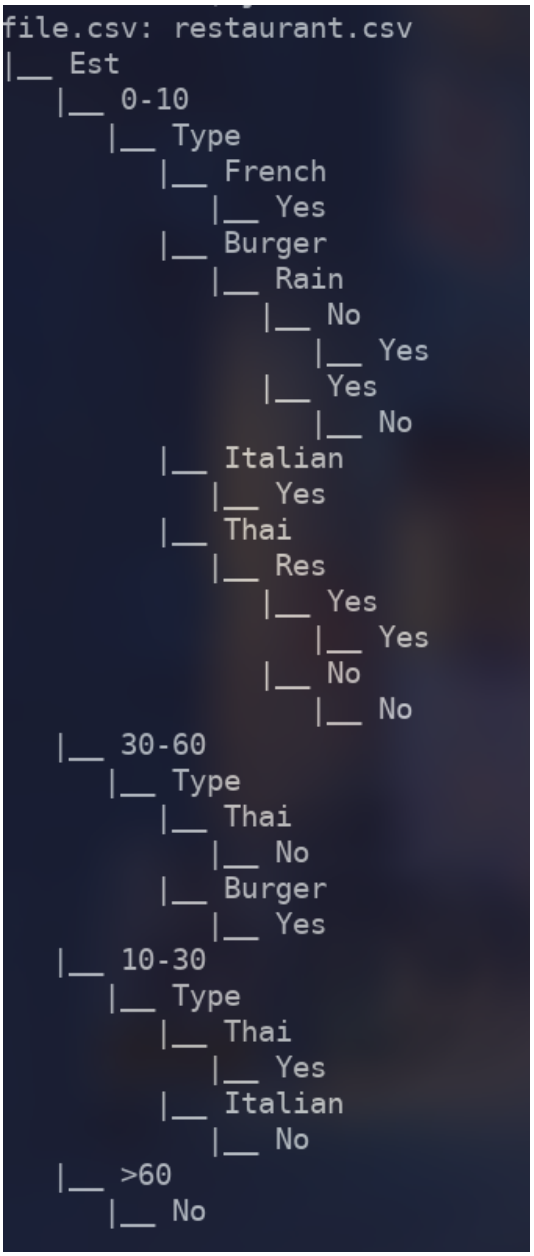
-roundNumber (percorre o csv e arrendoda os double, ex: no caso do iris.csv)

- classifyExample (diz a classe de um exemplo novo dado)
- main

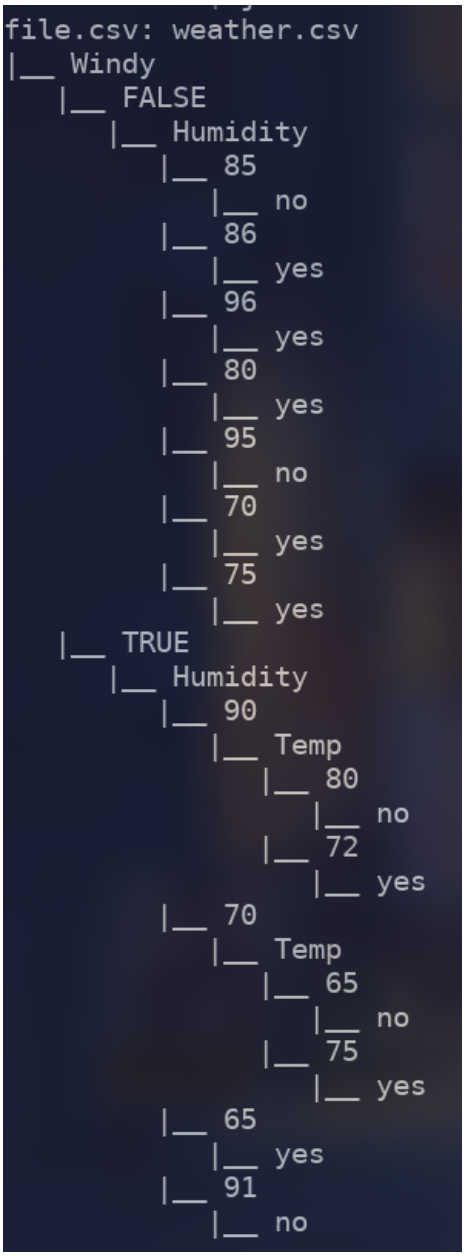
No Dnode.java temos a Class DNode para representar o Node e uma função que é para adicionar filhos.

Resultados

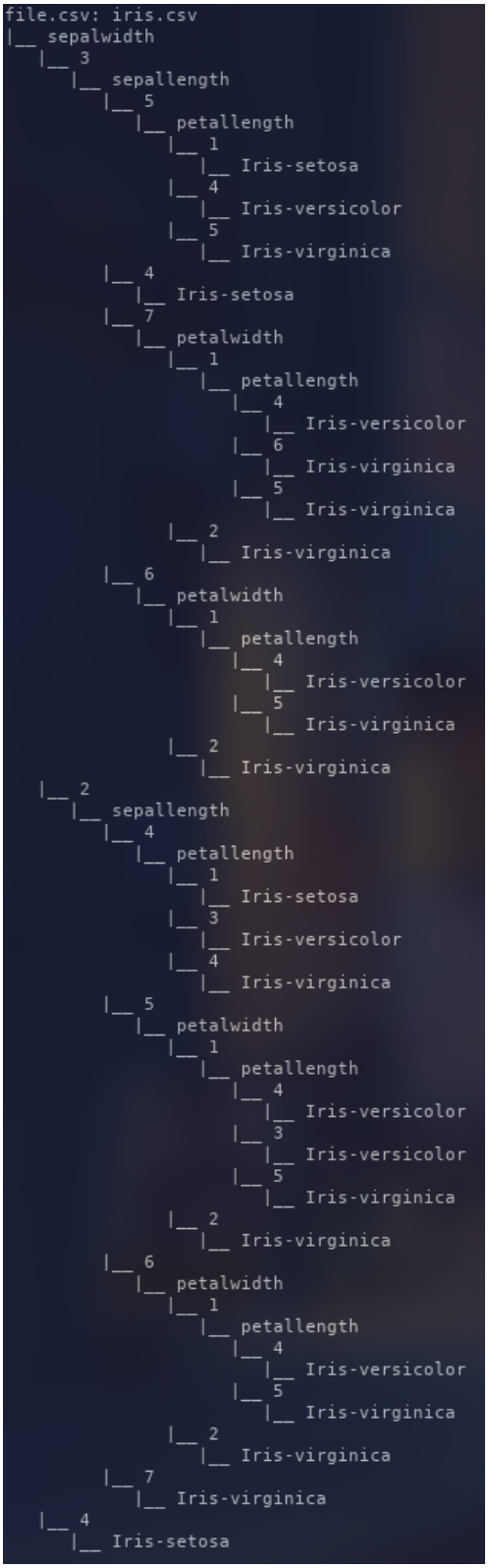
restaurant.csv



weather.csv



iris.csv


file.csv: iris.csv
|__ sepalwidth
|__ 3
|__ sepallength
|__ 5
|__ petallength
|__ 1
|__ Iris-setosa
|__ 4
|__ Iris-versicolor
|__ 5
|__ Iris-virginica
|__ 4
|__ Iris-setosa
|__ 7
|__ petalwidth
|__ 1
|__ petallength
|__ 4
|__ Iris-versicolor
|__ 6
|__ Iris-virginica
|__ 5
|__ Iris-virginica
|__ 2
|__ Iris-virginica
|__ 6
|__ petalwidth
|__ 1
|__ petallength
|__ 4
|__ Iris-versicolor
|__ 5
|__ Iris-virginica
|__ 2
|__ Iris-virginica
|__ 2
|__ sepallength
|__ 4
|__ petallength
|__ 1
|__ Iris-setosa
|__ 3
|__ Iris-versicolor
|__ 4
|__ Iris-virginica
|__ 5
|__ petalwidth
|__ 1
|__ petallength
|__ 4
|__ Iris-versicolor
|__ 3
|__ Iris-versicolor
|__ 5
|__ Iris-virginica
|__ 2
|__ Iris-virginica
|__ 6
|__ petalwidth
|__ 1
|__ petallength
|__ 4
|__ Iris-versicolor
|__ 5
|__ Iris-virginica
|__ 2
|__ Iris-virginica
|__ 7
|__ Iris-virginica
|__ 4
|__ Iris-setosa

Comentários Finais e Conclusão

Neste relatório, exploramos o conceito de árvores de decisão, sua aplicabilidade e diferentes algoritmos utilizados para a construção dessas estruturas.

Discutimos o funcionamento do algoritmo ID3 em detalhes, ressaltando sua abordagem baseada na entropia e no ganho de informação para selecionar os atributos mais relevantes. No entanto, também mencionamos algumas limitações do ID3, como sua tendência a criar árvores muito complexas e propensas a overfitting.

Os resultados demonstram como as decisões são tomadas com base nas características dos dados de entrada e fornecem uma visão clara do processo de classificação ou regressão realizado pela árvore de decisão.

Em conclusão, as árvores de decisão são ferramentas poderosas e versáteis para a tomada de decisões em problemas de classificação e regressão. Com a implementação correta e a escolha adequada dos parâmetros, as árvores de decisão podem ser uma adição valiosa ao conjunto de técnicas de aprendizado de máquina disponíveis para resolver problemas do mundo real.

Referências bibliográficas

- https://towardsdatascience.com/decision-trees-for-classification-id3-algorithm-explained-89df76e72df1
- Slides das aulas teóricas da unidade curricular Inteligência Artifical (2022/2023) (25/05/2023).
- https://www.displayr.com/gradient-boosting-the-coolest-kid-on-the-machine-learning-block/
- https://www.nvidia.com/en-us/glossary/data-science/xgboost/
- https://towardsdatascience.com/understanding-random-forest-58381e0602d2
- https://medium.com/geekculture/step-by-step-decision-tree-id3-algorithm-from-scratch-in-python-no-fancy-library-4822bbfdd88f
- https://towardsdatascience.com/entropy-and-information-gain-in-decision-trees-c7db67a3a293
- Russel,S. & Norvig,P. (2021). Artificial Intelligence: A Modern Approach, Global Edition(4th ed.).Pearson. (22/05/2023)