Universidade de Évora

Curso de Engenharia Informática

**Aprendizagem Automática**

Relatório da 1ªParte do Trabalho Prático

2020/2021

Árvores de Decisão com Pruning



Leonardo Catarro – nº43025

Diogo Solipa – nª43071

**Objetivo**

Este trabalho tem como objetivo implementar em Python uma classe para gerar árvores de decisão com “Pruning”, usando as diversas medidas de impureza: Gini, Entropia e Erro e REP (Reduced Error Pruning) como método de poda da árvore.

**Análise de Desempenho**

Fazer:

- Análise sobre conjuntos de dados fornecidos,

- Análise das árvores com e sem Pruning, não fizemos!

**🡪Diversas Funções de Homogeneidade/Pureza:**

No desenvolvimento deste trabalho, foram usadas as funções da Entropia, Gini e Erro, como formas do cálculo da impureza dos dados fornecidos.

De entre as três, a Entropia será a que possui pior desempenho, daí maior complexidade temporal, isto porque, além da sua complexidade resultante de iteração pelas listas que armazenam os dados, ainda envolve o cálculo com funções logarítmicas, que são computacionalmente complexas

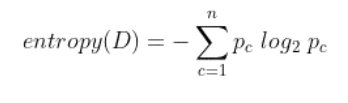
O método que utiliza o cálculo do Gini, poderá ser considerado o intermédio, apenas envolve produtos e cálculo fracionário

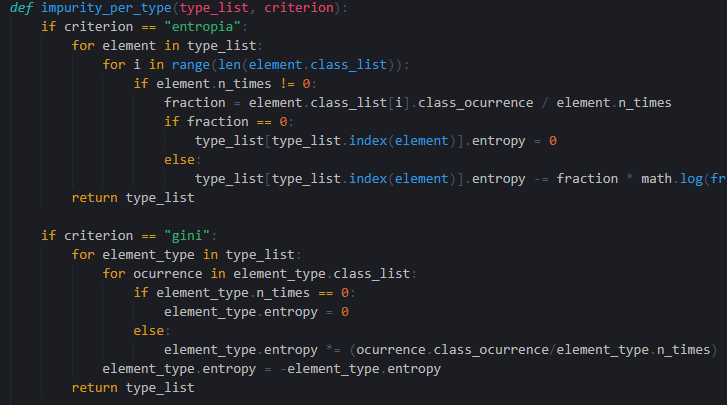
(Nota: Acrescentar mais?)

As fórmulas matemáticas e respetivas implementações são as seguintes:

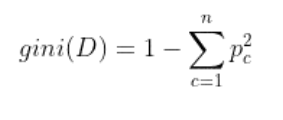
(Nota: Implementação de Gini, Entropia e Erro, todas dentro das funções: attributes\_impurity(criterion) e impurity\_per\_type(type\_list, criterion) )

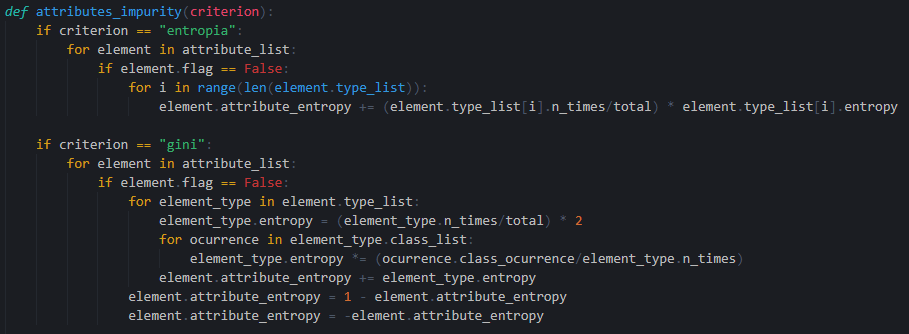
**Entropia: entropia para dados nominais(envolvem mais que 2 classes)**





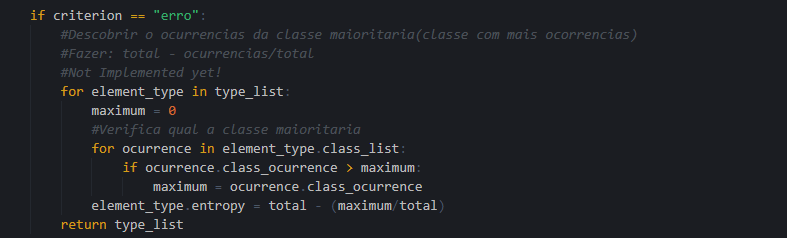
**Gini: (para dados nominais)**





**Classe Minoritária(Erro):**





**Problemas Encontrados**

Ao longo do desenvolvimento deste trabalho fomos nos deparando e enfrentando alguns problemas.

Inicialmente, o armazenamento dos dados, optamos pela programação orientada por objetos, o que para nós nos pareceu a melhor forma para o armazenamento dos dados. Porém, com o desenvolvimento percebemos que talvez não tenha sido a melhor escolha, mas ainda assim conseguimos desenvolver o que era pedido, com a nossa ideia inicial.

De seguida, e talvez o maior problema enfrentado, foi na parte da função recursiva (recursive\_split()), função esta que constrói a árvore, tratando os dados desde a raiz às folhas. Para tal, esta nossa função tende a ser “ajustada” ao exemplo fornecido em “weather.nominal.csv”, podendo responder com dados incorretos a alguns dos restantes ficheiros de input.

Por último, o desenvolvimento do Pruning, onde tivemos bastantes dificuldades, derivado às anteriormente mencionadas, em implementar a poda. Fazendo assim com a nossa árvore apensas seja construída sem Pruning.

(Nota: Manter? Acrescentar mais?)

Descrever os problemas encontrados no desenvolvimento do trabalho, etc…