

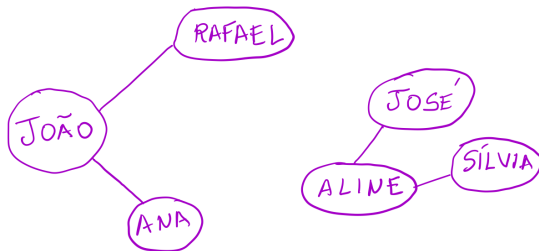
Floresta de Caminhos Ótimos

Advanced Institute for Artificial Intelligence – AI2

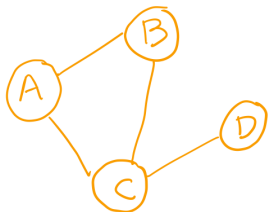
<https://advancedinstitute.ai>

Introdução

Existe um conjunto de abordagens que tratam o problema de classificação de padrões como sendo uma tarefa de particionamento em **grafos**. No entanto, o que seriam esses chamados grafos? Grafos podem ser entendidos como estruturas de dados que são compostas por **vértices** e **arestas** e que, dependendo de suas propriedades, podem modelar diferentes problemas.

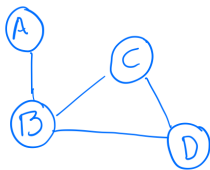


A Teoria dos Grafos é uma área que estuda o comportamento dos grafos e propõe análises teóricas e desenvolvimento de algoritmos para eles. Matematicamente falando, um grafo é definido como $G = (\mathcal{V}, \mathcal{E})$, em que \mathcal{V} denota o conjunto de vértices (nós) e \mathcal{E} corresponde ao conjunto de arestas (pares de nós). Vejamos o exemplo abaixo.

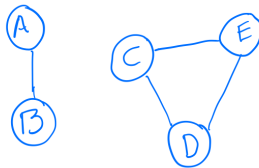


Neste caso, temos que $G = (\mathcal{V}, \mathcal{E})$, em que $\mathcal{V} = \{A, B, C, D\}$ e $\mathcal{E} = \{(A, B), (A, C), (B, C), (C, D)\}$. Note que a **relação de adjacência é simétrica**, ou seja, as arestas (A, B) e (B, A) são iguais neste caso.

Assim sendo, problemas que podem ser modelados como sendo grafos são beneficiados por inúmeros algoritmos já desenvolvidos para diversas aplicações. Grafos podem ser classificados de acordo com sua topologia e relação de adjacência, principalmente. Com relação à topologia, podemos dividir os grafos em:

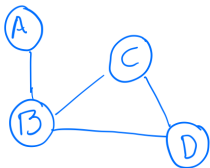


Grafo conexo

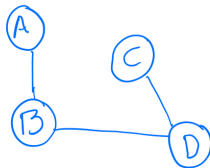


Grafo desconexo

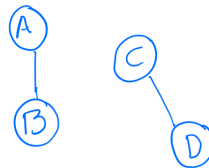
Ou ainda em:



Grafo cíclico

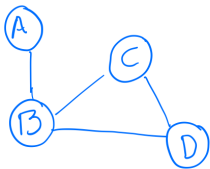


Grafo acíclico (árvore)

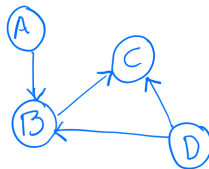


Floresta

De acordo com a sua relação de adjacência, podemos classificar os grafos em:

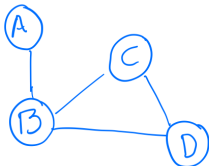


Grafo não direcionado

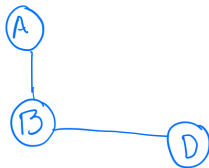


Grafo direcionado

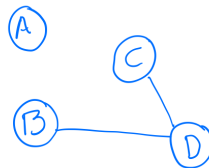
Grafos são, de maneira geral, definidos como sendo conjuntos de vértices e arestas. Desta forma, conseguimos classificá-los, ainda, de acordo com seus subconjuntos: **subgrafo gerador** e **subgrafo gerado**.



Grafo original



Subgrafo gerado



Subgrafo gerador

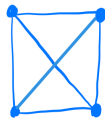
Um grafo é dito ser completo quando todos os pares de nós estão conectados. Uma família bastante conhecida de grafos que obedece à esta propriedade é conhecida por K_z , em que z denota o número de nós.



K_1 (grafo trivial)

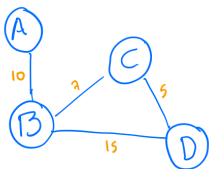


K_2

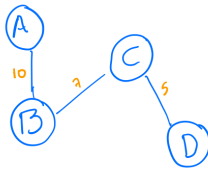


K_4

Grafos podem ser ainda **ponderados** em suas arestas (distâncias entre cidades, por exemplo). Desta forma, temos que nosso grafo pode ser definido como segue: $G = (\mathcal{V}, \mathcal{E}, w)$, em que $w : \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}$ é a função que associa pesos às arestas. De acordo com essa definição, temos um importante tipo de grafo chamado de **árvore geradora mínima**, do inglês *minimum spanning tree* - MST, que é um subgrafo gerador, do tipo árvore, cujo somatório dos pesos das arestas é **mínimo**.



Grafo original

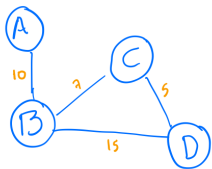


MST

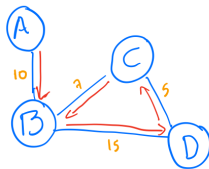
A técnica Floresta de Caminhos Ótimos, do inglês *Optimum-Path Forest* - OPF, visa modelar o problema de classificação das amostras como sendo uma tarefa de particionamento de um grafo em grupos de amostras com o mesmo rótulo. Neste sentido, as amostras do conjunto de dados correspondem aos nós do grafo e as arestas são definidas por alguma relação de adjacência escolhida previamente.

Definição do problema: dados um conjunto de treinamento rotulado e com m amostras, ou seja, $\mathcal{X}^1 = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_m, y_m)\}$ e um grafo $G = (\mathcal{V}, \mathcal{E}, w)$, nosso objetivo é então particionar G em grupos de amostras com o mesmo rótulo. Temos que $\mathcal{V} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\}$, \mathcal{E} deve ser escolhido de alguma forma que satisfaça o seu problema e w é geralmente escolhido como uma função distância entre duas amostras, ou seja, $w(\mathbf{x}_i, \mathbf{x}_j)$ calcula a distância entre os nós (amostras) \mathbf{x}_i e \mathbf{x}_j . De maneira geral, o classificador OPF usa a distância euclidiana para este fim.

Como particionar o conjunto de dados de treinamento em grupos? O classificador OPF usa a ideia de **competição** entre alguns nós denominados de **protótipos**, os quais tentam conquistar as demais amostras do conjunto de treinamento atribuindo **melhores** custos à elas por meio de **caminhos de custo ótimo**. O que são esses caminhos?

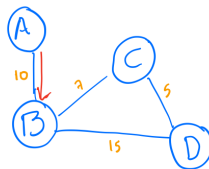


Grafo original



Passeio

$$\pi_B = \langle A, B, D, C, B \rangle$$



Caminho

$$\pi_B = \langle A, B \rangle$$

O classificador OPF opera em duas etapas, como de costume com outros classificadores, isto é, treinamento e teste. A etapa de treinamento é responsável por particionar o grafo em grupos de amostras com o mesmo rótulo (árvores de caminhos ótimos, do inglês *optimum-path trees* - OPTs), enquanto que a etapa de teste associa, à cada amostra do conjunto de teste, a sua amostra de treinamento mais **fortemente conexa**.

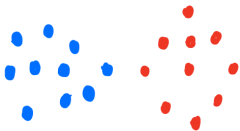
A etapa de treinamento é composta por três passos:

- Criação do grafo por meio da escolha da relação de adjacência.
- Escolha das amostras protótipos e da função de custo.
- Processo de conquista.

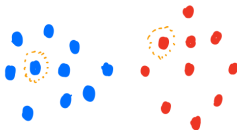
Dependendo da relação de adjacência escolhida, metodologia para estimar protótipos e função de custo, diferentes classificadores OPF podem ser obtidos. Iremos abordar a primeira versão proposta, que possui as seguintes características:

- Relação de adjacência: grafo completo.
- Metodologia para estimar protótipos: MST.
- Função de custo: valor máximo de aresta ao longo do caminho.

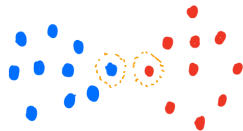
Qual a principal motivação para uso da MST para escolha dos nós protótipos? De maneira análoga ao classificador SVM, queremos as amostras mais próximas de classes diferentes.



Conjunto de treinamento

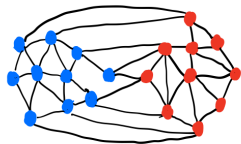


Escolha de protótipos
aleatória

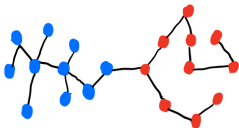


Escolha de protótipos
pela MST

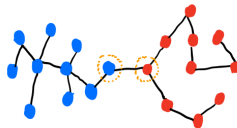
Como funciona a escolha de protótipos pela MST? Basicamente, dado o grafo de treinamento e arestas ponderadas pelas distâncias entre os respectivos nós, calculamos a MST e depois selecionamos os nós mais próximos de classes diferentes. Note que temos, ao menos, **um protótipo para cada classe**, mas nada impede que tenhamos mais.



Grafo de treinamento

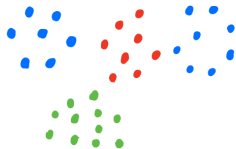


MST

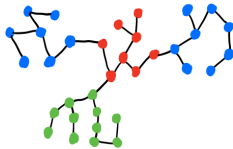


Protótipos selecionados

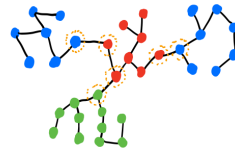
Um outro exemplo de situação bastante comum. Note que o classificador OPF possui suporte à classificação por múltiplas de classes de maneira **nativa**.



Grafo de treinamento



MST



Protótipos selecionados

Como funciona a função de custo de caminho? Como dito anteriormente, OPF pode trabalhar com diferentes opções para relação de adjacência e metodologia para estimar protótipos. O mesmo ocorre com a função de custo. A versão que veremos faz uso da função de custo definida como segue:

$$C(\mathbf{x}_i) = \begin{cases} 0 & \text{caso } \mathbf{x}_i \in \mathcal{S} \\ \infty & \text{caso contrário,} \end{cases} \quad (1)$$

e

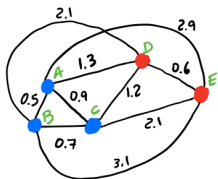
$$C_{\mathbf{x}_j}(\mathbf{x}_i) = \max\{C(\mathbf{x}_j), w(\mathbf{x}_j, \mathbf{x}_i)\}. \quad (2)$$

Neste caso, temos que \mathcal{S} corresponde ao conjunto de protótipos escolhidos pela MST, $C_{\mathbf{x}_j}(\mathbf{x}_i)$ corresponde ao custo oferecido pela amostra \mathbf{x}_j para a amostra \mathbf{x}_i e $w(\mathbf{x}_j, \mathbf{x}_i)$ denota a distância (peso da aresta) entre os nós \mathbf{x}_j e \mathbf{x}_i , tal que $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}^1$.

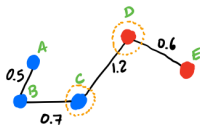
O algoritmo do OPF pode ser entendido como um problema de otimização, em que a ideia é atribuir o **menor custo possível** à cada amostra de treinamento. Desta forma, a ideia seria resolver o seguinte problema de otimização:

$$C(\mathbf{x}_i) = \min \left\{ \max_{\forall \mathbf{x}_j \in \mathcal{X}^1 \setminus \{\mathbf{x}_i\}} \{C(\mathbf{x}_j), w(\mathbf{x}_j, \mathbf{x}_i)\} \right\}. \quad (3)$$

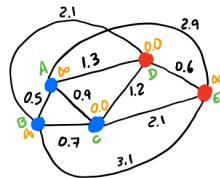
Vejamos um exemplo de funcionamento da etapa de treinamento do algoritmo do OPF.



Grafo de treinamento

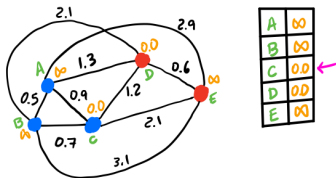


MST

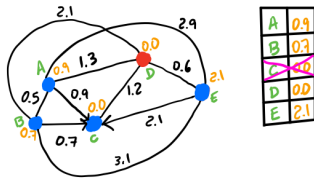


A	∞
B	∞
C	0.0
D	0.0
E	0.0

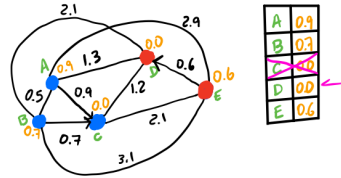
Inicialização dos custos



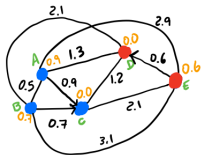
Retirada do protótipo
da fila de prioridades



Custos atualizados

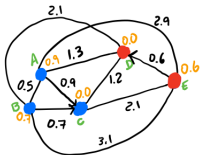


Próximo protótipo
sai da fila



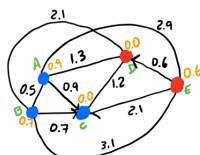
A	0.9
B	0.7
C	0.0
D	0.0
E	0.6

Custos atualizados



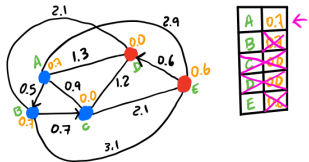
A	0.9
B	0.7
C	0.0
D	0.0
E	0.6

Próxima amostra sai
da fila de prioridades

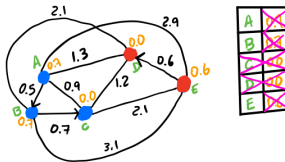


A	0.9
B	0.7
C	0.0
D	0.0
E	0.6

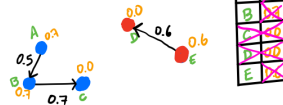
Sem atualização de custo e próxima
amostra sai da fila de prioridades



Custos atualizados



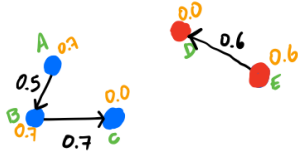
Próxima amostra sai
da fila de prioridades



Sem atualização de custo e
fim do treinamento

No final da etapa de treinamento, temos a geração de uma **floresta de caminhos ótimos**, que é composta por **árvores de caminhos ótimos** cujas raízes são os nós protótipos.

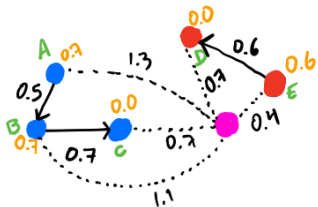
passos direcionais \neq passos



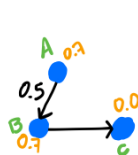
Já a etapa de classificação funciona da seguinte maneira: dada uma amostra $\mathbf{x} \in \mathcal{X}^2$, conectamos ela à todas as amostras do conjunto de treinamento. Em seguida, verificamos quem oferece o melhor custo, de maneira similar à Equação 3, ou seja:

$$C(\mathbf{x}) = \min\left\{\max_{\forall \mathbf{x}_i \in \mathcal{X}^1} \{C(\mathbf{x}), w(\mathbf{x}_i, \mathbf{x})\}\right\}. \quad (4)$$

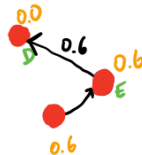
A amostra $\mathbf{x}_i \in \mathcal{X}^1$ que oferecer o menor custo conquista a amostra de teste e oferece o seu rótulo (classe) à ela.



Amostra de teste é conectada à
todas as amostras de treinamento



Amostra recebe o seu
rótulo final



O classificador OPF possui outras variantes supervisionadas (OPF com relação de adjacência por k -vizinhos mais próximos), semi-supervisionadas e não supervisionadas. Já foi aplicado em diferentes situações, tais como:

- Recuperação de imagens (nativo).
- Classificação por múltiplos rótulos.
- Classificação de imagens (medicina, engenharia e sensoriamento remoto, dentre outras).
- Classificação de gêneros musicais.
- Classificação de sinais.
- Em conjunto com aprendizado em profundidade.