**Classificador K-NN em Python**

**Introdução**

O classificador K-NN é um Classificador estatístico não paramétrico baseado na NN-Rule que classifica o padrão de acordo com a distância entre o padrão e os seus K vizinhos mais próximos, segundo a medida de distância (métrica). O fator de ponderação do K-NN é a métrica utilizada.

Considere o Exemplo:

 **Tabela1**

|  |  |  |
| --- | --- | --- |
| **(Espessura da gordura)** | **(Peso da carne)** | **(Classificação)** |
| 7 | 7 | Ruim |
| 7 | 4 | Ruim |
| 3 | 4 | Bom |
| 1 | 4 | Bom |

Para o conjunto de dados da **tabela 1** é possível criar um modelo estatístico que permite estimar a classe de um novo padrão de dados utilizando um classificador K-NN. Por exemplo, vamos utilizar o **padrão de teste** **** e testa-lo para saber se é classificado como BOM ou RUIM considerando-se o conjunto de dados da **tabela 1**. Existem técnicas para selecionar o valor de **K** (Número de vizinhos próximos), mas consideraremos o número 3 (3-NN) apenas como exemplo.

**1º passo:** Calcular a distância entre todos os vetores e o vetor de teste. A métrica neste caso será a distância euclidiana.

|  |  |  |
| --- | --- | --- |
| **(Espessura da gordura)** | **(Peso da carne)** | **(D.E.)2** |
| 7 | 7 | (7-3)2+(7-7)2=16 |
| 7 | 4 | (7-3)2+(4-7)2=25 |
| 3 | 4 | (3-3)2+(4-7)2=9 |
| 1 | 4 | (1-3)2+(4-7)2=13 |

**2º passo:** Selecionar os 3 vizinhos mais próximos (RANK):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **(Espessura da gordura)** | **(Peso da carne)** | **[DE(3,7)]2** | **RANK** | **Selecionado** |
| 7 | 7 | 16 | 3 | SIM |
| 7 | 4 | 25 | 4 | NÃO |
| 3 | 4 | 9 | 1 | SIM |
| 1 | 4 | 13 | 2 | SIM |

**3º passo:** Tomando os 3-NN veja a categoria (classificação) predominante do padrão (3,7).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **(Espessura da gordura)** | **(Peso da carne)** | **[DE(3,7)]2** | **RANK** | **3-NN** |
| 7 | 7 | 16 | 3 | Ruim |
| 3 | 4 | 9 | 1 | Bom |
| 1 | 4 | 13 | 2 | Bom |

Entre os 3-NN o padrão foi classificado uma vez como **RUIM** e duas como **BOM**, sendo assim, o padrão (3,7) é considerado **BOM**, ou seja, pertence à classe **BOM**.

**Implementação do classificador utilizando a biblioteca sklearn em python**

Para esta prática iremos utilizar o ambiente Colab Research. Este ambiente possibilita o desenvolvimento de software na linguajem Python utilizando um navegador web.

1. Acesse o ambiente Colab através do link: <https://colab.research.google.com>
2. Faça login utilizando seu e-mail e senha da USP
3. Na tela inicial clique em criar **Novo notebook**, este procedimento irá criar um notebook Jupyter. (Para mais detalhes sobre o Notebook Jupyter, visite o site [www.jupyter.org](http://www.jupyter.org))



1. Renomeie o notebook para ProjetoKnn, clicando sobre o nome atual.



Célula

1. Clique na célula do notebook e digite o seguinte código que serve para importar as bibliotecas que serão utilizadas, depois clique no ícone ‘Executar Célula’



1. Crie uma nova célula de código clicando em **“+ Código”.** Digite e execute o seguinte código que cria a base de dados do nosso classificador baseado na tabela 1:



1. Em uma nova célula, digite e execute o código a seguir que cria o modelo de classificador.



1. O seguinte código pode ser utilizado para exibir uma figura mostrando cada elemento da base de dados com sua respectiva classe:



**Os próximos passos servem para classificar um ponto qualquer que não esteja na base de dados**

1. Crie uma lista contendo as entradas de um novo ponto a ser classificado:

****

1. Utilize o modelo classificador criado para predizer a classe do novo ponto

****

Resultado da operação

1. O seguinte código pode ser utilizado para plotar o novo ponto junto com a base de dados



**Exemplo 2**

1. Crie um novo notebook
2. Baixe no computador o arquivo ‘Carcacas.txt’ disponível no e-disciplinas
3. Clique no ícone ‘Arquivos’ do notebook no Colab e depois no ícone ‘Fazer upload para armazenamento da sessão’



Fazer Upload

Arquivos

1. Faça upload do arquivo ‘Carcacas.txt’
2. Escreva o código a seguir para criar o modelo classificador baseado nos dados do arquivo ‘Carcacas.txt’.





1. Teste o classificador com os seguintes valores:

0 0

0 1

1 0

1 1

0.465 0.0256

0.856 0.917

0.205 0.374

0.799 0.965