UNIVERSIDADE ESTADUAL DE CAMPINAS
Curso de Aperfeiçoamento – Fundamentos de Mineração de Dados INF-1020 Fundamentos de Mineração de Dados
Prof. Julio Cesar dos Reis

LISTA - Classificação

1	1,00	
2	1,00	
3	1,00	
4	1,00	
5	1,00	
6	1,00	
7	1,00	
8	1,00	
9	1,00	
10	1,00	
Total	10,0	

Nome Aluno 1: Diego Coimbra

RG Aluno 1: 143.583.89-12

Nome Aluno 2: Nara Guimarães

RG Aluno 2: 44.662.603-x

Instrução: Esta lista de exercícios deve ser resolvida em dupla. A nota será no intervalo [0, 10].

Questão	Valor	Nota

1. Enumere três aplicações do mundo real que podem se beneficiar da classificação supervisionada.

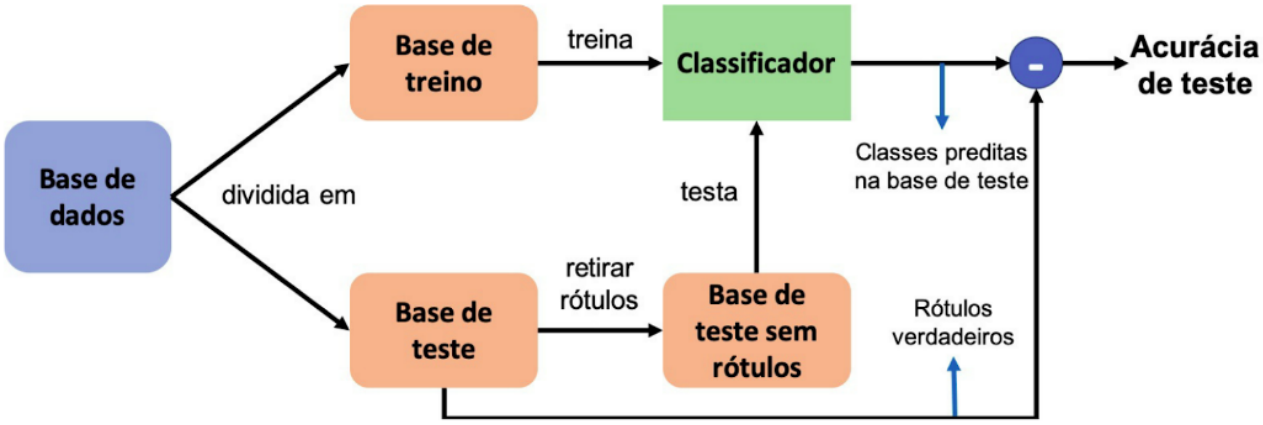
- Classificar e separar os e-mails quanto a spam e mensagens principais
- Identificar transações financeira fraudulentas
- Identificar se um tumor é maligno ou benigno via reconhecimento de imagem
- Analisar imagens de satélite
- Sensoriamento remoto

2. Descreva as atividades nas fases envolvidas na tarefa de classificação.

Para aplicar a classificação, precisamos selecionar um dataset que possua rótulos, pois esse é um tipo de modelo de aprendizagem supervisionada, com esses dados em mãos, faz-se uma separação entre dados de treino e dados de teste (geralmente a separação é de 70x30, mas pode variar um pouco). Os dados de treino são utilizados para ensinar o modelo como os dados se comportam de modo a possuírem aquele rótulo que lhes é associado.

Com o modelo treinado, usa-se os dados de teste, sem os seus rótulos, para que o modelo consiga predizer quais deveriam ser os rótulos associados a eles, a partir do modelo preditivo. Com esses rótulos previstos faz-se então uma comparação com o gabarito que tínhamos desse conjunto de testes. Nesses casos, é interessante usar a matriz de confusão para entender o quão efetiva foi essa predição, e quanto de erro o modelo teve. Assim consegue-se verificar se a aplicação da técnica é interessante e representativa para esse determinado conjunto de dados.

Além dessas etapas, tem-se também o uso desse modelo para dados reais em que não se conhece ainda o rótulo, de modo a predizer a resposta final para os mesmos.

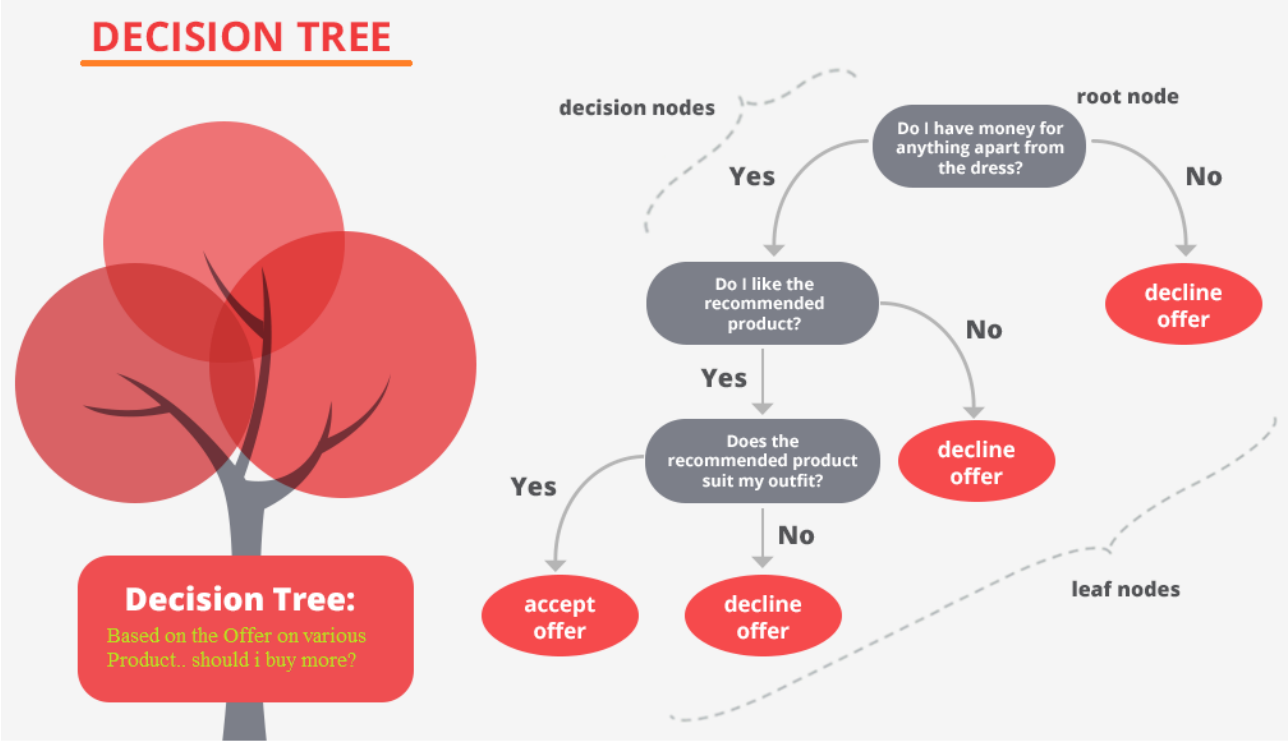


Fonte: http://estatidados.com.br/machine-learning-conceitos-e-modelos-parte-i-aprendizado-supervisionado/, original de ESCOVEDO & KOSHIYAMA, 2020

3. Explique o funcionamento de uma árvore de decisão.

Uma árvore de decisão é um algoritmo de aprendizado de máquina supervisionado que é utilizado para classificação e para regressão. Isto é, pode ser usado para prever categorias discretas (sim ou não, por exemplo) e para prever valores numéricos (o valor do lucro em reais, por exemplo).

Assim como um fluxograma, a árvore de decisão estabelece **nós** (decision nodes) que se relacionam entre si por uma hierarquia. Existe o **nó-raiz** (root node), que é o mais importante, e os **nós-folha** (leaf nodes), que são os resultados finais. No contexto de machine learning, o raiz é um dos atributos da base de dados e o nó-folha é a classe ou o valor que será gerado como resposta.



Fonte: Data Science Foundation

Na ligação entre nós, temos **regras de "se-então"**. Ao chegar em um nó A, o algoritmo se pergunta acerca de uma regra, uma condição, na qual se baseará para optar por um caminho ou

por outro.

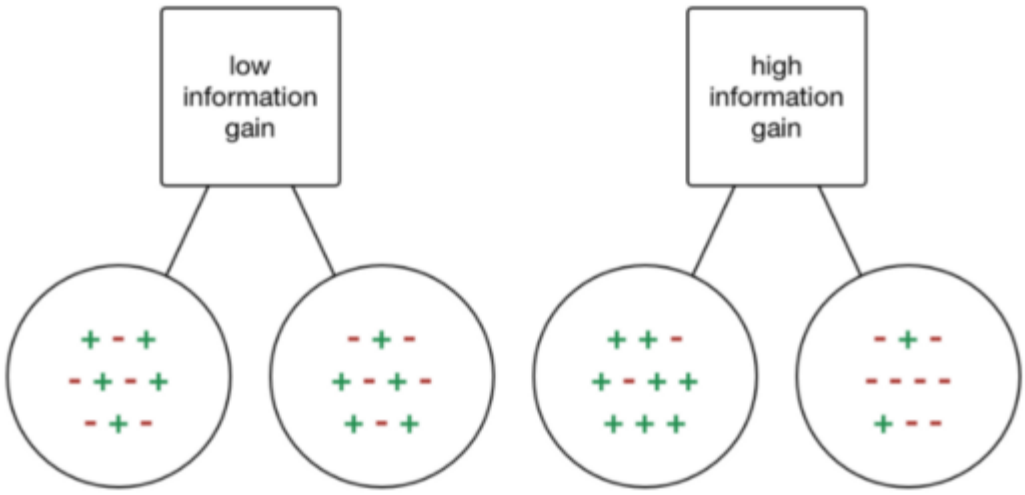
É um algoritmo que segue o que chamamos de “recursivo” em computação. Ou seja, ele repete o mesmo padrão sempre na medida em que vai entrando em novos níveis de profundidade. É como se uma função chamasse a ela mesma como uma segunda função para uma execução paralela, da qual a primeira função depende para gerar sua resposta.

O grande trabalho da árvore é justamente encontrar os nós que vão ser encaixados em cada posição. Para isso, é preciso realizar alguns importantes cálculos. Uma abordagem comum é usar o **ganho de informação e a entropia**. Essas duas variáveis dizem respeito à desorganização e falta de uniformidade nos dados. Quanto mais alta a entropia, mais caóticos e misturados estão os dados. Quanto menor a entropia, mais uniforme e homogênea está a base.

Para definir os posicionamentos, é preciso calcular a entropia das classes de saída e o ganho de informação dos atributos da base de dados. Quem tiver maior ganho de informação entre os atributos é o nó-raiz. Para calcular a esquerda e a direita, deve-se realizar novos cálculos de entropia e ganho com o conjunto de dados que atende à condição que leva à esquerda ou à direita.

Como falamos, para dividir a base de dados em uma árvore, dependemos das condições. A partir delas, dividimos a base em caminhos com análise dos registros que satisfazem determinada condição.

O ganho de informação é calculado a partir dessa lógica. Se quando eu analiso um atributo, **os registros das bases para cada lado são homogêneos ou próximos disso, temos um alto ganho de informação**. Afinal, sabemos que se optarmos por determinada condição, é muito provável que saibamos exatamente a saída esperada ou estejamos mais próximos de descobrir.



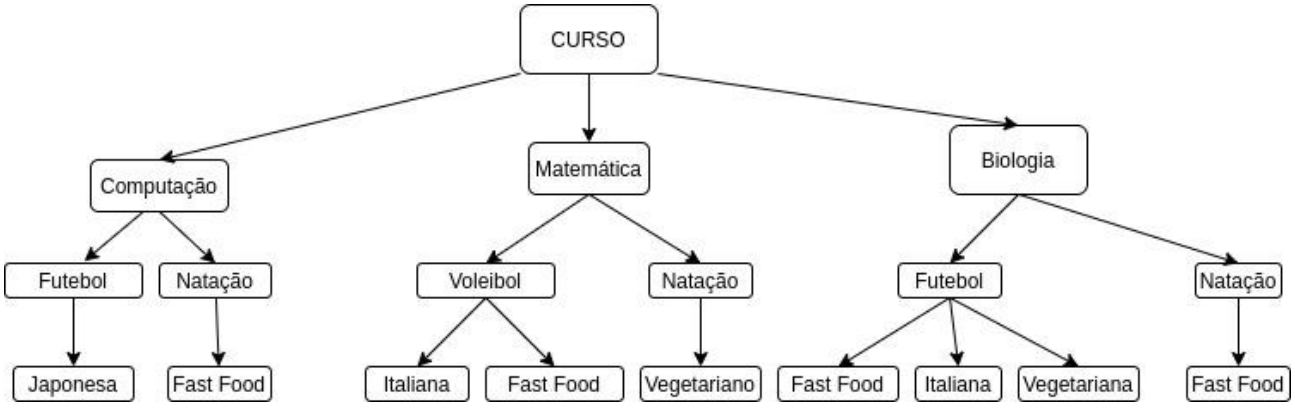
Fonte: TowardsDataScience

Contudo, se o ganho for pequeno, isso quer dizer que os dados estão muito misturados e que, portanto, estamos mais distantes de descobrir as saídas esperadas.

Outro ponto acerca desse algoritmo é que ele foca bastante na tarefa atual, e menos no resultado final. Ou seja, quando está calculando o lado esquerdo de um nó, esse método não considera o outro lado. É o que chamamos de algoritmo ganancioso.

4. Apresente uma árvore de decisão para os seguintes dados:

Curso	Esporte	Tipo de comida
computação	futebol	japonesa
computação	natação	fastfood
computação	natação	fastfood
computação	natação	fastfood
matemática	voleibol	italiana
matemática	natação	vegetariana
matemática	voleibol	fastfood
biologia	futebol	fastfood
biologia	futebol	italiana
biologia	futebol	vegetariana
biologia	futebol	italiana
biologia	natação	fastfood



5. Explique o funcionamento do algoritmo ID3 para indução de árvores de decisão.

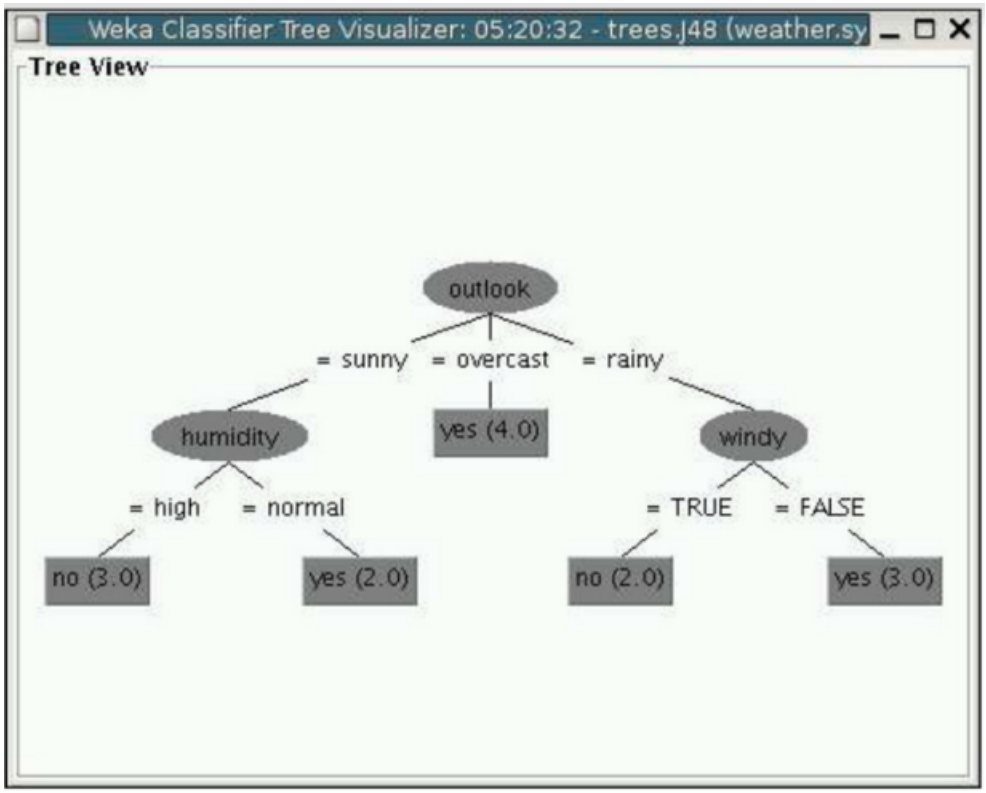
O algoritmo ID3, constrói árvores de decisão a partir da raiz e começa com a pergunta "*que atributo deveria ser testado na raiz da árvore?*". Para responder esta pergunta, cada atributo da instância é avaliado usando um teste estatístico para determinar como este classifica os exemplos de treinamento. O melhor atributo é selecionado e é usado como o teste na posição do nó raiz da árvore. Um descendente do nó raiz é criado então para cada possível valor deste atributo, e os exemplos de treinamento são particionados e associados a cada nó descendente para selecionar o melhor atributo para testar naquele ponto na árvore. Isto forma uma procura para uma árvore de decisão aceitável na qual o algoritmo nunca retrocede para reconsiderar escolhas feitas anteriormente.

6. Considere a árvore de decisão abaixo. Como seriam classificadas as seguintes instâncias?

(a) outlook = sunny, temperature = cool, humidity = high, windy = TRUE (NO - 3.0)

(b) outlook = rainy, temperature = cool, humidity = high, windy = TRUE (NO - 2.0)

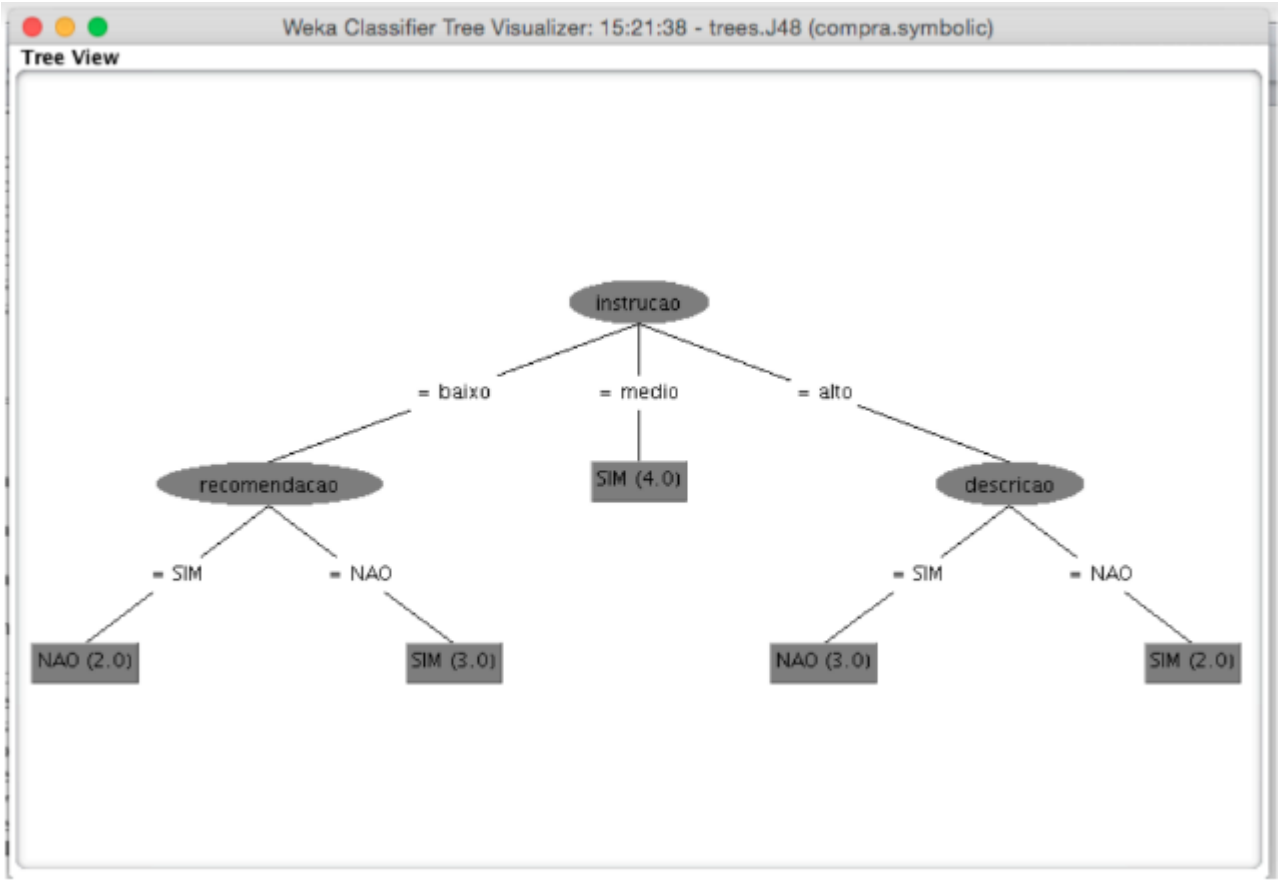
(c) outlook = overcast, temperature = cool, humidity = high, windy = TRUE (YES - 4.0)



7. Considere a árvore de decisão abaixo. Ela refere-se a um modelo que pretende predizer se compras serão efetivadas em um *site* de comércio eletrônico. Quatro variáveis são consideradas no modelo:

- **instrução**: refere-se ao grau de instrução do cliente que pode ser baixo, médio, ou alto;
- **recomendação**: registra se o cliente acessou a ṕagina do produto por um link recomendado (valor SIM) ou não (valor NÃO);
- **descrição**: registra se o cliente buscou informações adicionais sobre o produto, clicando em algum link de descrição (há dois valores possíveis: SIM ou NAO);
- **duração**: refere-se ao tempo dispensado pelo cliente no site, que pode ser baixo, médio, ou alto.

As classes do modelo indicam a efetivação ou não de uma compra (classe *SIM* ou classe *NÃO).*



(a) Indique uma instância de teste que seria classificada como pertencendo a classe “SIM”.

Instrução: baixa, recomendação: não, descrição: sim, duração: baixo (descrição e duração podem assumir valores distintos, uma vez que apenas as variáveis instrução e recomendação que definem se será sim ou não ao final)

(b) Como seriam classificadas as instâncias abaixo?

- duração = baixo, descricao = NAO, recomendação = SIM, instrução = baixo NÃO (2.0)
- duração = baixo, descricao = SIM, recomendação = NAO, instrução = baixo SIM (3.0)

(c) Seria possível classificar uma instância cujo atributo instrução é igual a *médio* na classe “NÃO”? Justifique sua resposta. Não é possível, uma vez que quando a instrução assume valor médio, existe apenas uma opção final para a compra. O valor nesse ramo resultará sempre em um valor positivo, haverá efetivação da compra

(d) Por que o atributo *duração* não foi usado no modelo? Pois este atributo não tem relevância no resultado final, frente os demais. Sendo assim desnecessária sua inclusão na árvore de decisão.

8. Explique o método de classificação supervisionada com base em distâncias.

A classificação supervisionada se baseia na identificação de diferentes classes com comportamentos espectrais diferenciados. Para isso, algoritmos de classificação são adotados para extrair os traços/ características de interesse em um espaço multidimensional.
Os algoritmos de classificação supervisionada são: Distância Mínima, Distância Mahalanobis, Distância de Bhattacharya, Máxima Verossimilhança, Método Paralelepípedo e Método Spectral Angle Mapper.
Métodos de classificação supervisionada podem ser subdivididos em dois grupos:

- Baseados em separabilidade (entropia): árvores de decisão e variantes
- Baseados em particionamento: SVM (support vector machines).

O método da distância mínima calcula a distância espectral entre o vetor de medida para o elemento candidato e a média para cada assinatura de classe. O método se utiliza da medida de distância Euclidiana. Cada elemento será incorporado a um agrupamento através da análise da medida de similaridade de distância Euclidiana, que é dada por:

$$D(x, n) = \sqrt{(x_i - m_i)^2}$$

onde xi é elemento candidato, mi é a média das classes e n é o número de bandas. O classificador compara a distância Euclidiana de cada elemento à média de cada agrupamento. O elemento candidato é designado à classe com média mais próxima, isto é, à classe que apresenta a menor distância Euclidiana.

O método da distância Mahalanobis é, simplesmente, a medida da distância do elemento na posição x do espaço multidimensional ao centro da classe, dividida pelo comprimento do elipsóide na direção de x. Isso tem a propriedade de minimizar a distância do ponto ao centro da média. Para se usar a distância Mahalanobis para classificar um elemento a uma das n classes, inicialmente, calcula-se a matriz de covariância com base nas amostras de treinamento das n classes, e o elemento será destinado à classe na qual a distância de Mahalanobis seja a menor de todas. Esse método supõe que a covariância das amostras são iguais, portanto, é um algoritmo mais rápido que os algoritmos de distância mínima e o máxima verossimilhança. A classificação ocorre por:

$$D_m(x, y) = |x - y|A = \sqrt{(x - y)^T A^{-1} (x - y)}$$

onde Dm(x, y) é a distância entre dados dois vetores e A^{-1} é a matriz de covariância inversa computada a partir de uma distribuição multivariada de entrada. Como pode ser observado, a equação acima também se reduz à norma vetorial, caso A seja uma matriz identidade.

O método máxima verossimilhança (MaxVer) considera a ponderação das distâncias entre as médias dos valores dos elementos das classes, utilizando parâmetros estatísticos. Assume-se que todas as bandas têm distribuição normal e calcula a probabilidade de um dado elemento pertencer

a uma classe específica. Para que a classificação por MaxVer seja precisa, é necessário um número razoavelmente elevado de elementos para cada conjunto de treinamento, esse número permite uma base segura para tratamento estatístico. Na classificação MaxVer cada elemento é destinado à classe que tem a mais alta probabilidade, isto é, a máxima verossimilhança. A equação relacionada a esse método é dada por:

$$x \in w_i \text{ se } p\,(x/w_i)p\,(w_i) > \; p\,(x/w_j)p\,(w_j)$$

onde, a probabilidade p(x/ωi) dá a possibilidade de x pertencer à classe ωi e p(ωi) é a probabilidade de a classe ocorrer. Tantas quanto forem as classes de treinamento selecionadas, tantas serão p(x/ωi).

O MaxVer tem uma base estatística mais complexa e por isso utiliza um tempo bem maior de processamento computacional do que os demais métodos citados.

9. Explique a diferença de funcionamento entre do método de classificação que usa mínima distância euclidiana com o algoritmo dos vizinhos mais próximos.

- Classificação com base na mínima distância euclidiana:

A Distância Euclidiana é definida como a soma da raiz quadrada da diferença entre dois pontos em suas respectivas dimensões (entre todos os seus atributos). É um método simples de ser implementado e interpretado.

Os seus pontos negativos são: a distribuição de classes nem sempre é hiper-esférica, e esta tem seu uso focado em atributos numéricos.

Dentre seus principais usos, cita-se: a modelagem aproximada com mais de um protótipo e o uso de medidas de similaridade para outros tipos de atributos

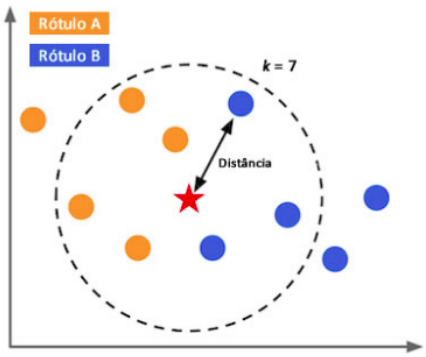
$$D_E(p,q) = \sqrt{(p_1 - q_1)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

A classificação baseada na distância mínima, compara todos os valores fornecidos pela equação acima e classifica o elemento como pertencendo à classe do elemento mais próxima do mesmo.

- Algoritmo dos vizinhos mais próximos (KNN):

A ideia principal do KNN é determinar o rótulo de classificação de uma amostra baseado nas amostras vizinhas advindas de um conjunto de treinamento. A variável k representa a quantidade de vizinhos mais próximos que serão utilizados para averiguar de qual classe a nova amostra pertence.

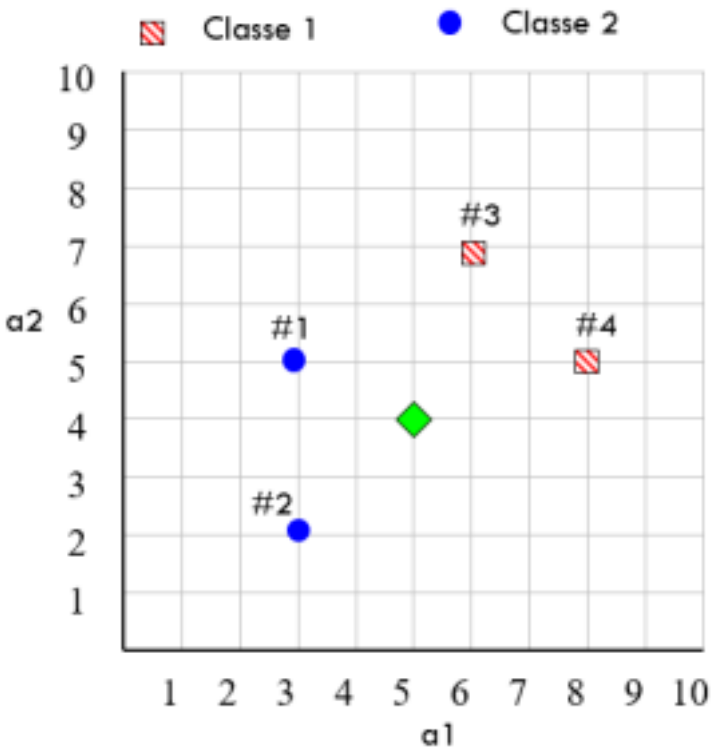
Calcular a distância é fundamental para o KNN. Existem diversas métricas de distância, e a escolha de qual usar varia de acordo com o problema. A mais utilizada é a distância Euclidiana, porém pode-se utilizar a distância de Minkowsky ou a distância de Chebyshev.



10. Aplique o algoritmo dos vizinhos mais próximos sendo $k = 1$ considerando a figura abaixo.

(a) Apresente os cálculos e responda a classe predita.

(b) Para $k = 2$ há empate?



a) Usando a distância entre pontos para

Ponto x = (5,4)

Ponto 1 = (3,5)

Ponto 2 = (3,2)

Ponto 3 = (6,7)

Ponto 4 = (8,5)

x-1 = sqrt((5-3)² + (4-5)²) = sqrt(4+1) = 2,24

x-2 = sqrt((5-3)² + (4-2)²) = sqrt(4+4) = 2,83

x-3 = sqrt((5-6)² + (4-7)²) = sqrt(1+9) = 3,16

x-4 = sqrt((5-8)² + (4-5)²) = sqrt(9+1) = 3,16

ranqueamento em ordem crescente da distância:

1,2,3/4

Usando k-1, o ponto em questão encontra-se mais próximo da classe 2 (bolinha azul)

b) Como pelo ranqueamento apresentou nas duas primeiras posições elementos da classe 2, usando um k=2, faz com que o ponto em questão também seja classificado como pertencente a esta classe. Não havendo assim empate.