Ciência de Dados

WEBINAR 05 – Unidade 03

UNIVERSIDADE CANDIDO MENDES | EAD

Introdução ao Big Data

- Aprendizados de Máquina:
  - Etapas do aprendizado de máquina:
    - Obtenção dos dados: seleção dos dados necessários para realização do treinamento;
    - Limpeza e adequação dos dados: preparação dos dados para que sejam adequados ao modelo de treinamento;
    - Treinamento do modelo: execução de algoritmo adequado a sua natureza (preditiva ou descritiva);
    - Teste: utilização de alguma métrica de avaliação do modelo (Medidas de Acurácia)
      - Exemplo: Regressão Linear -> R^2 / MAE / RMSE / MAPE
    - Ajuste e Refinamento: Adequação do modelo para aprimorar a acurácia do modelo.

Introdução ao Big Data

- Aprendizados de Máquina:
  - Etapas do aprendizado de máquina:
    - Tarefas preditivas: objetivo é estimar o atributo-alvo (variável dependente) de novas instâncias a partir de modelo previamente treinado;
    - Tarefas descritivas: objetivo de explorar um conjunto de dados sem qualquer interferência externa, com intuito de organizar e separar os dados a partir de padrões percebidos;

Introdução ao Big Data

- Aprendizados de Máquina:
  - Tipo de aprendizado:
    - Semi-supervisionado:
      - Conhecidos como modelos híbridos;
      - 1ª etapa: aprendizado supervisionado -> treinamento de algoritmo de classificação utilizando os dados rotulados existentes;
        - Teste: verificação de acurácia;
      - 2ª etapa: aprendizado não supervisionado -> clusterização dos dados incompletos;
      - 3ª etapa: aprendizado supervisionado -> novo treinamento de algoritmo de classificação utilizando os novos dados rotulados;
        - Teste: verificação da acurácia;
      - 4ª etapa: avaliação da evolução da acurácia; Se for suficiente: Fim! Senão: volte a etapa 2ª etapa e refaça a clusterização; Loop até acurácia satisfatória.
      - Ganho: Aumento da base de dados que será utilizada para treinar o algoritmo de classificação;

Introdução ao Big Data

- Aprendizado não supervisionado:
  - Agrupamento (clusterização):
    - K-means (k médias):
      - É um algoritmo de agrupamento que utiliza a distância de centroides para os pontos em seu entorno;
      - Quando um ponto se afasta de um primeiro centroide e aproxima-se de um segundo, este ponto passa a pertencer a este segundo grupo;
      - Os centroides vão sendo reposicionados de acordo com a média dos valores de cada grupo (daí vem o nome k-means);
      - A cada iteração uma nova média é calculada, de acordo com a classificação de cada ponto naquele momento, com isso o centroide é reposicionado;

Introdução ao Big Data

- Algoritmo de Agrupamento - K-means:

1: Definir o número k de clusters a ser assinalado;
2: Randomicamente inicialize k centroides;
3: Repita até que não haja modificações na posição dos centroides:
    4: Calcular a função de distância entre cada ponto e os centroides dos clusters;
    5: Assinale cada ponto ao centroide mais próximo;
    6: Calcule as novas posições dos centroides através da média de cada cluster;

Introdução ao Big Data

- Algoritmo de Agrupamento - K-means:
  - Não é bom para agrupar formas tipo anel; Neste caso utilizar DBSCAN ou algum outro algoritmo que satisfaça a condição de análise

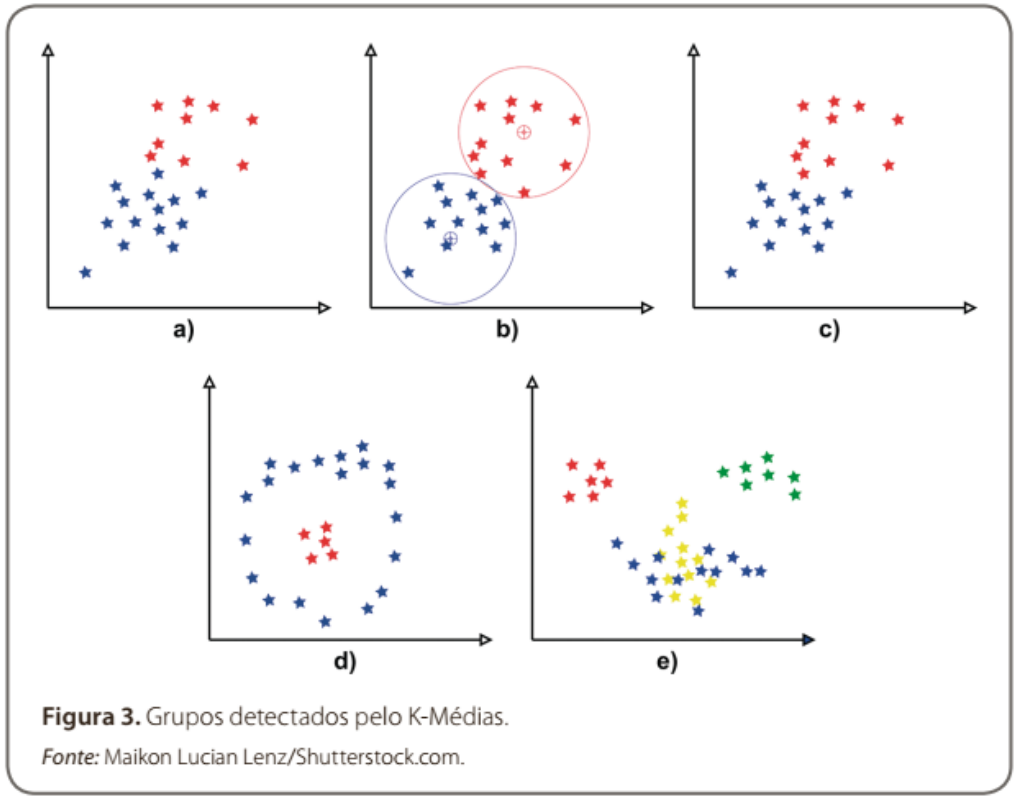
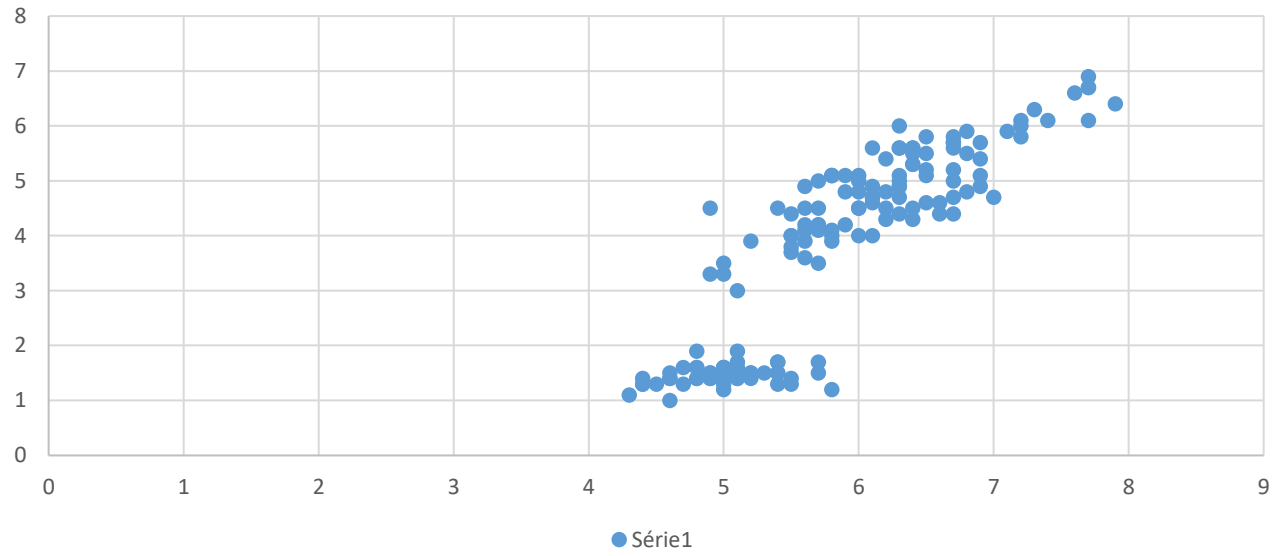


Figura 3. Grupos detectados pelo K-Médias.

Fonte: Maikon Lucian Lenz/Shutterstock.com.

Introdução ao Big Data

- K-means na prática:
  - Queremos achar os k grupos de dados para identificar cada ponto do dataset:
  - 1º Passo: Quantos clusters serão utilizados? k = 3

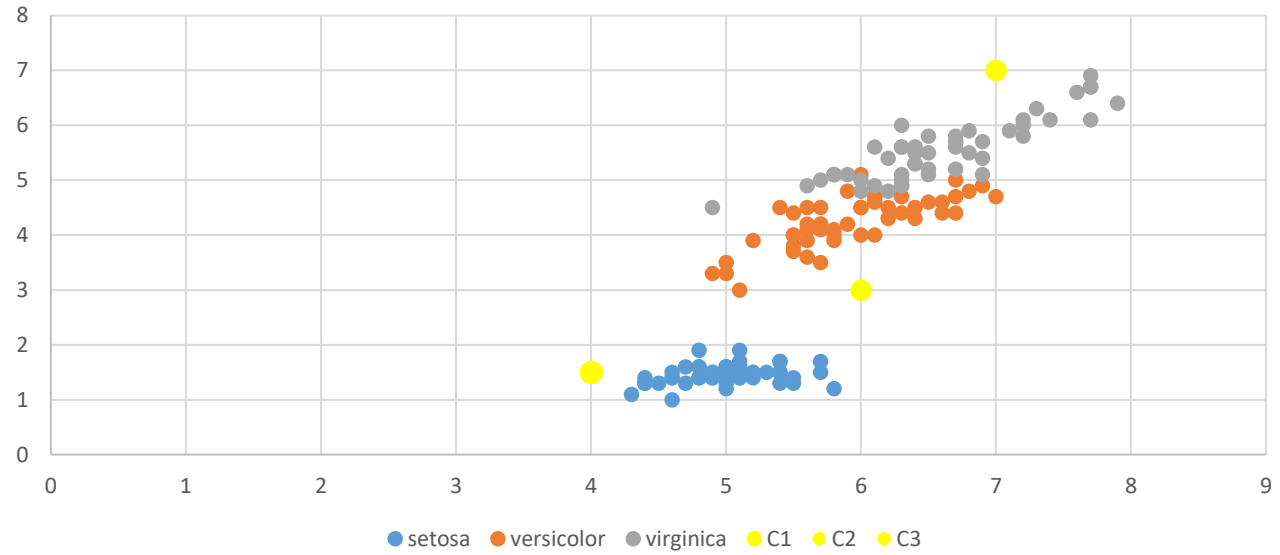


Introdução ao Big Data

- K-means na prática:
  - Queremos achar os k grupos de dados para identificar cada ponto do dataset:
  - 2º Passo: Calcular as distâncias dos pontos para os centroides;
  - 3º Passo: Associar cada ponto ao seu devido centroide;

Início		
	x	y
C1	4,00	1,50
C2	6,00	3,00
C3	7,00	7,00

SQRE: 196,40



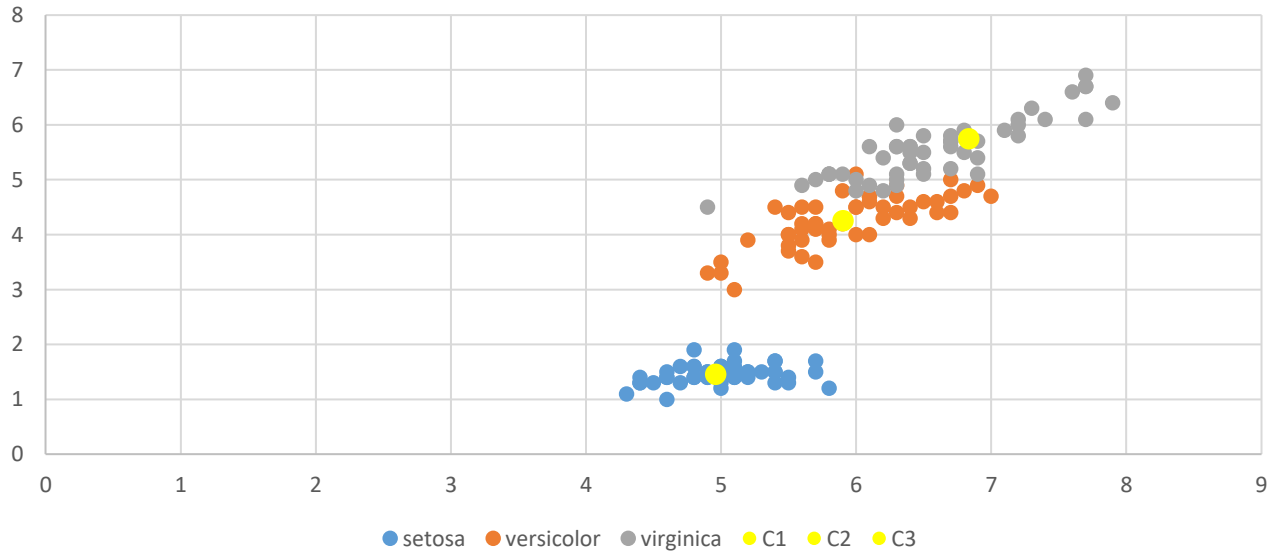
Introdução ao Big Data

- K-means na prática:
  - Queremos achar os k grupos de dados para identificar cada ponto do dataset:
  - 4º Passo: Calcular novas posições dos centroides de acordo com cada ponto associado;
  - 5º Passo: Calcular os critérios de parada

Novos Centroídes - 1ª Iteração		
	x	y
C1	4,96	1,46
C2	5,90	4,26
C3	6,83	5,75

Crit. Parada	
x	y
0,96	0,04
0,10	1,26
0,17	1,25

SQRE: 77,63 (-152,98%)



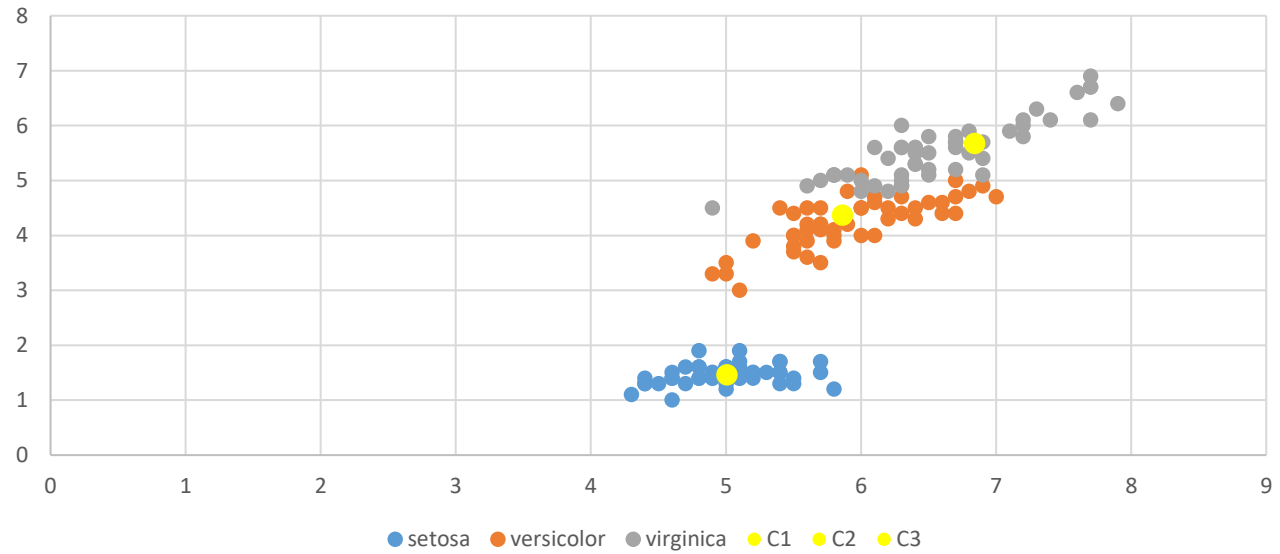
Introdução ao Big Data

- K-means na prática:
  - Queremos achar os k grupos de dados para identificar cada ponto do dataset:
  - Volte ao 2º passo e faça tudo novamente

Novos Centroídes - 2ª Iteração		
	x	y
C1	5,01	1,46
C2	5,86	4,37
C3	6,84	5,68

Crit. Parada	
x	y
0,05	0,00
0,04	0,11
0,00	0,07

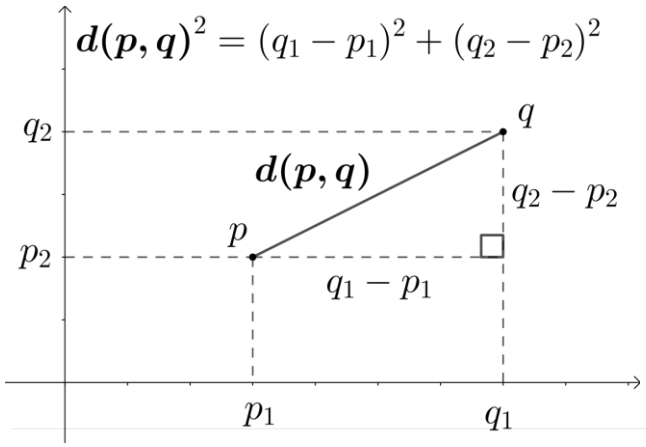
SQRE: 76,41 (-2%)



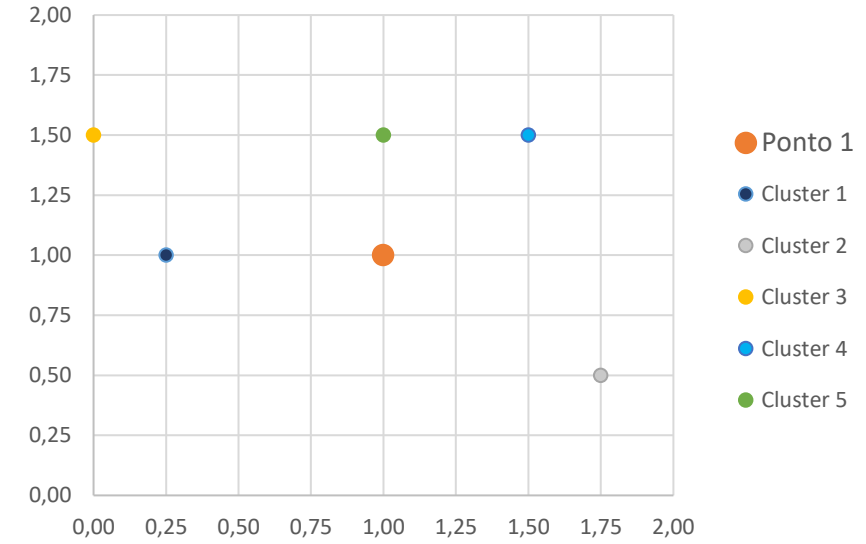
Introdução ao Big Data

- K-means na prática:
  - Como calculamos a distância de cada ponto para os centroides?
  - Existem várias fórmulas de cálculo de distância, uma muito utilizada é a distância euclidiana

$$E(x,y) = \sqrt{\sum_{i=0}^{n}(x_i - y_i)^2}$$



Cluster	X	Y
Cluster 1	0,25	1,00
Cluster 2	1,75	0,50
Cluster 3	0,00	1,50
Cluster 4	1,50	1,50
Cluster 5	1,00	1,50
Ponto 1	1,00	1,00



Ponto - Cluster	Distância	Fórmula
Cluster 1	0,75	RAIZ(((C8-C3)^2)+((D8-D3)^2))
Cluster 2	0,90	RAIZ(((C8-C4)^2)+((D8-D4)^2))
Cluster 3	1,12	RAIZ(((C8-C5)^2)+((D8-D5)^2))
Cluster 4	0,71	RAIZ(((C8-C6)^2)+((D8-D6)^2))
Cluster 5	0,50	RAIZ(((C8-C7)^2)+((D8-D7)^2))

Introdução ao Big Data

- K-means e Python:
  - É preciso importar as bibliotecas da scikit-learn;
  - Instale no seu ambiente de trabalho utilizando conda ou pip;
    - pip install pandas
    - pip install numpy
    - pip install sklearn
  - Fazendo as importações necessárias:
    - import pandas as pd
    - import numpy as np
    - from sklearn.cluster import Kmeans
    - from sklearn.metrics import silhouette_score
    - from sklearn.preprocessing import StandardScaler

Introdução ao Big Data

- K-means e Python:
  - Antes de treinar o algoritmo:
    - Primeiramente:
      - Esteja certo dos valores das suas variáveis independentes;
      - Existem casos onde distâncias possuem escalas diferentes, portanto, é sempre bom transformar seus dados para uma escala padrão, chamado de Padronização;
      - O método fit_transform da biblioteca StandardScaler é responsável por tal processo.
        - `scaler = StandardScaler()`
        - `scaler.fit_transform(dfT_x)`
      - Identifique a quantidade ideal de cluster utilizando os métodos do cotovelo e silhueta;

Introdução ao Big Data

- K-means e Python:
  - Treinando o algoritmo:

```
kmeans_kwargs = {
    'init': 'k-means++',
    'n_init': 50,
    'max_iter': 1000,
}

kmeans = KMeans(n_clusters=3, **kmeans_kwargs)
kmeans.fit(dfT_x_scaler)
```

- n_clusters: int, default=8

Número de clusters e centroides que serão gerados;

- init: {'k-means++', 'random'}

Método de inicialização dos centroides;

- n_init: int, default=10

Número de vezes que o algoritmo será executado com inicializações de centroides diferentes;

- max_iter: int, default=300

Número máximo de iterações para paralização das iterações do algoritmo;

- random_state: int, RandomState instance or None, default=None

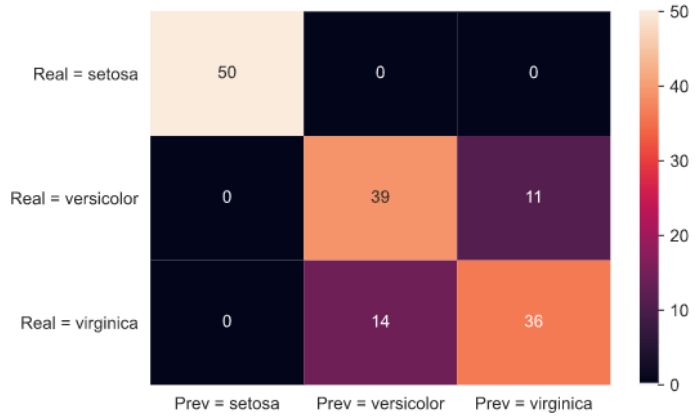
Determina o número randomico de inicialização do algoritmo. Ao fazer o método do cotovelo ou silhueta, procure especificar esse valor para evitar a alteração dos pontos de inicialização.

Introdução ao Big Data

- Matriz de Confusão:
  - É possível identificar os resultados que foram Positivos Verdadeiros, Positivos Falsos, Negativos Verdadeiros e Negativos Falsos;
  - Mede a acurácia do modelo treinado;
    - Acurácia = (VP+VN)/n
    - Precisão = (VP) / (VP+FP)
    - Recall = (VP) / (VP + FN)
    - f1-score: $2 * \frac{precision * recall}{precision + recall}$

		Classificação atual	
		P	**N**
Classificação prevista	**P**	**VP**	FP
	N	FN	**VN**

Onde:
VP = Verdadeiro Positivo;
FP = Falso Positivo;
VN = Verdadeiro Negativo;
FN = Falso Negativo.



$$\text{Acurácia} = \frac{50 + 39 + 36}{150} = 83\%$$

UNIVERSIDADE
CANDIDO
MENDES

EAD