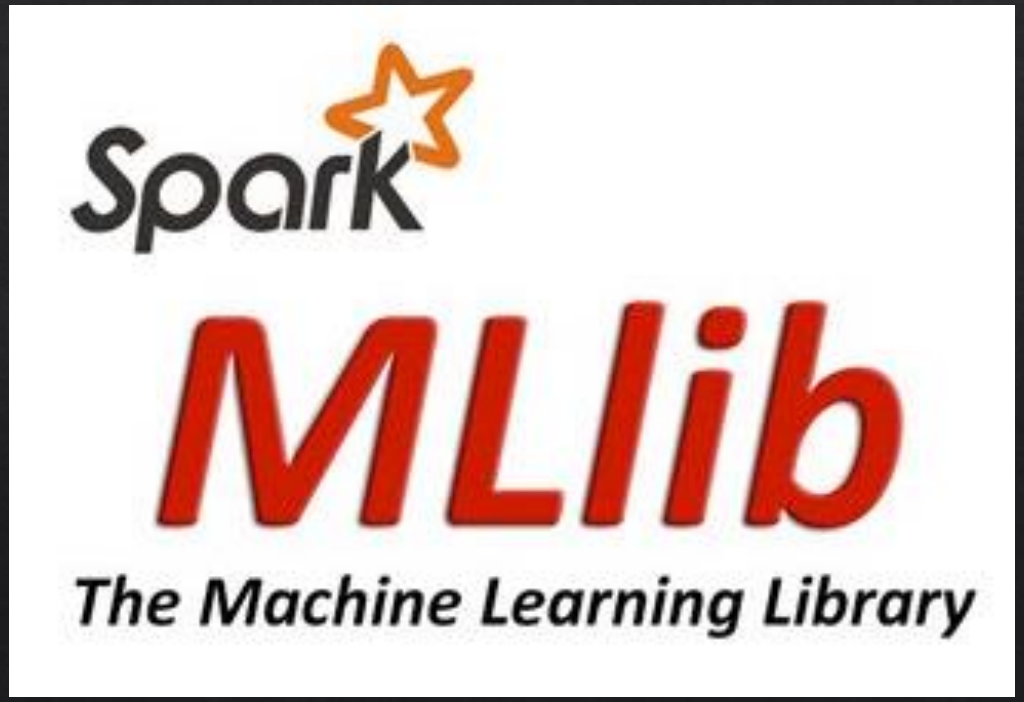
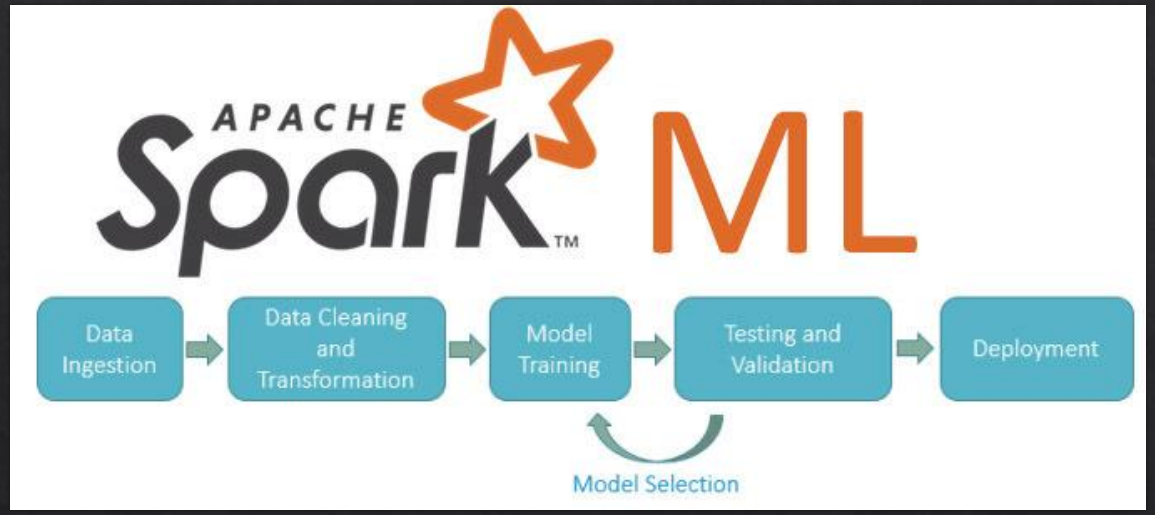
PARTE IV

4. Machine Learning em Streaming de Dados com Spark MLlib



Apache Spark Machine Learning – 2 Bibliotecas

Aplicação de Machine Learning em Dataframes



- Framework Scikit-Learn: rodar em 1 computador
- Framework Spark Machine Learning : rodar em clusters

- Qual das 2 bibliotecas usar vai depender do volume de dados (volume muito grande usar RDD, volume menor usa-se um Dataframe)

Aplicação de Machine Learning em RDD's

Spark MLlib

- O modulo Mllib do Spark contém as funções que implementam ML.
- Execução em paralelo através de Cluster
- Usa o Numpy como base!

Tipos de Analytics

Analytics - É o processo de coletar dados e gerar insights para tomadas de dicisões baseadas em fatos!

Analytics	Descrição
Descritiva	Compreender o que aconteceu (BI)
Exploratória	Descobrir porque alguma coisa aconteceu
Inferencial	Compreender uma população a partir de uma amostra
Preditiva	Prever o que vai acontecer (ML)
Casual	O que ocorre com uma variável quando outra é alterada
Deep	Técnicas avançadas para compreender grandes conjuntos de dados de diversas fontes (IA)

Big Data + Analytics = Big Data Analytics

Análise em tempo real + Analytics = Real-Time Analytics

Análise Exploratória e Preditiva

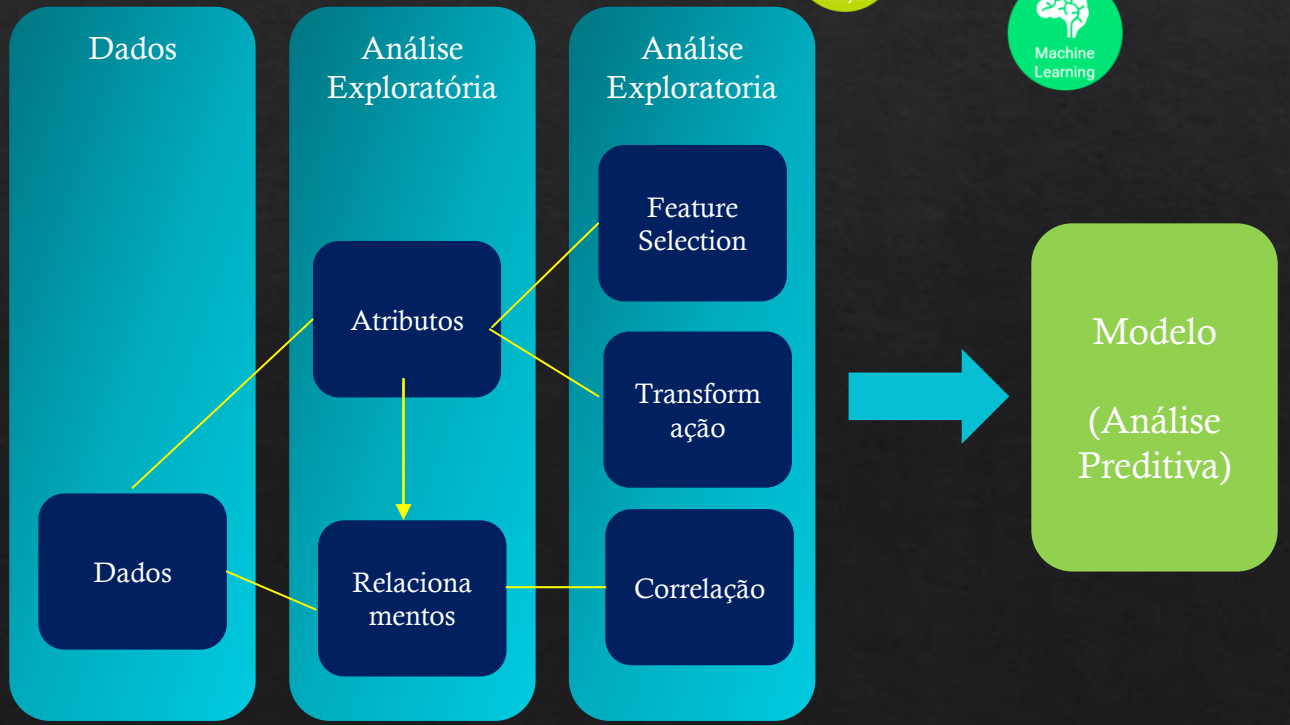
Análise Exploratória

- Compreender variáveis preditoras e target no dataset
- Descobrir padrões e tendências
- Testar Hipóteses
- Validar o processo de coleta dos dados
- Encontrar variáveis chave
- Descobrir a correlação entre as variáveis
- Detectar outliers
- Eliminar variáveis irrelevântes

Ferramentas:

1. Matriz de correlação (Saber quais variáveis usar)
2. Histogramas (Vizualizar a distribuição de 1 variável)
3. Scatterplots (Visualizar a relação entre 2 variáveis)
4. Boxplots (Visualizar outliers, min e max, dispersão dos dados)
5. Principal Component Analysis (Técnica que permite coletar informações de diversas variáveis e traduzir em componentes)

Análise Preditiva



Análise Preditiva - ML

Aprendizado Supervisionado

- Fazer previsões a partir do treinamento com dados de entrada e saída
- Os modelos são constituídos em datasets de **treino**
- Os modelos são usados para prever o futuro

Técnicas:

- Regressão (dados numéricos e contínuos)
- Classificação (classes)

Procedimento:

- Dados históricos com variáveis preditoras e target;
- Split do conjunto de dados em treino(70%) e teste(30%);
- Dados Treino: treinar o modelo
- Dados Teste: testar e validar o modelo
- Métrica para avaliar o modelo: Acurácia (ex)
- Seleção aleatória dos dados em ambos datasets

Aprendizado Não-Supervisionado

- Buscar estrutura ou similaridade oculta nos dados
- Grupos observados baseados em similaridade entre as entidades

Similaridade entre as entidades pode ser:

- Distância entre os valores, presença/ausência de atributos

Técnicas:

- Clustering
- Regras de Associação
- Filtros Colaborativos (Sistemas de Recomendação)

Trande-off entre Viés e Variância

- Reduzir o erro do modelo
- Principais componentes do erro em predições: bias e variance

---

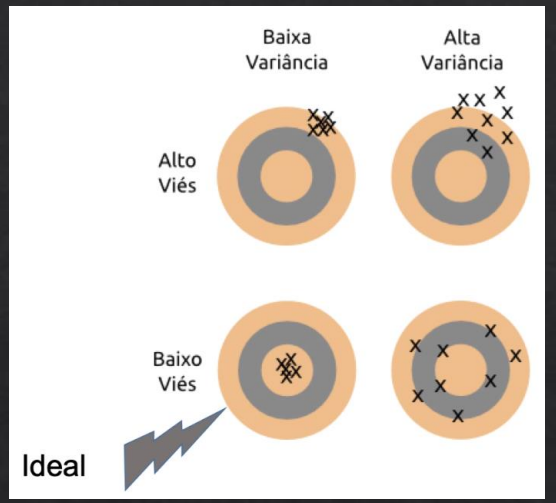
Bias (Viés)

- É a diferença entre o valor esperado da predição do modelo (media das predições) e o valor real que queremos predizer.
- O bias está relacionado à habilidade do modelo em se ajustar aos dados, ou seja, se o problema é um *underfitting*, o modelo tem um alto bias.

Variância

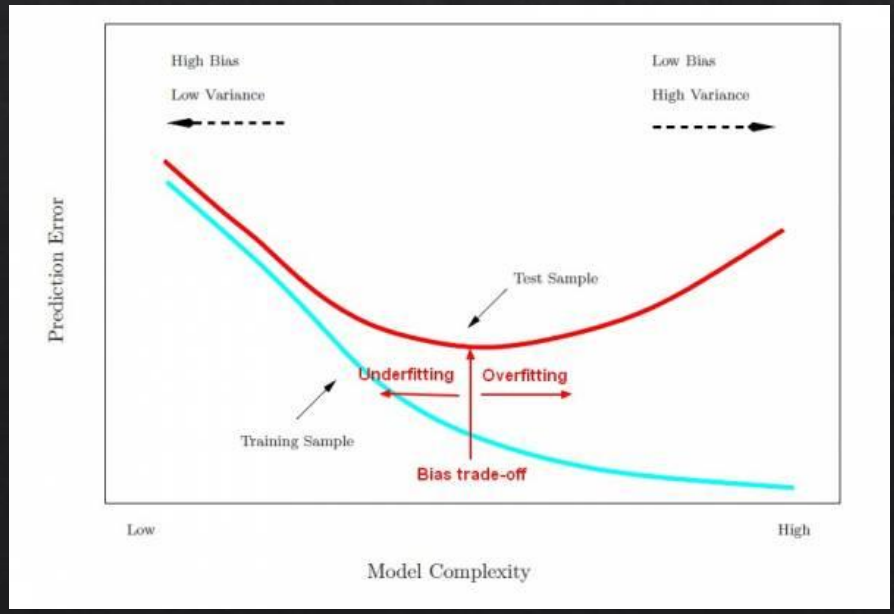
- É a variabilidade das predições.
- A variância está relacionada à habilidade do modelo se ajustar a novos dados, ou seja, se o seu problema é em *overfitting*, o modelo tem uma alta variância.

O objetivo é reduzir o bias ea variância o máximo que pudermos, entretanto, nos deparamos com um trade-off entre underfitting e overfitting.

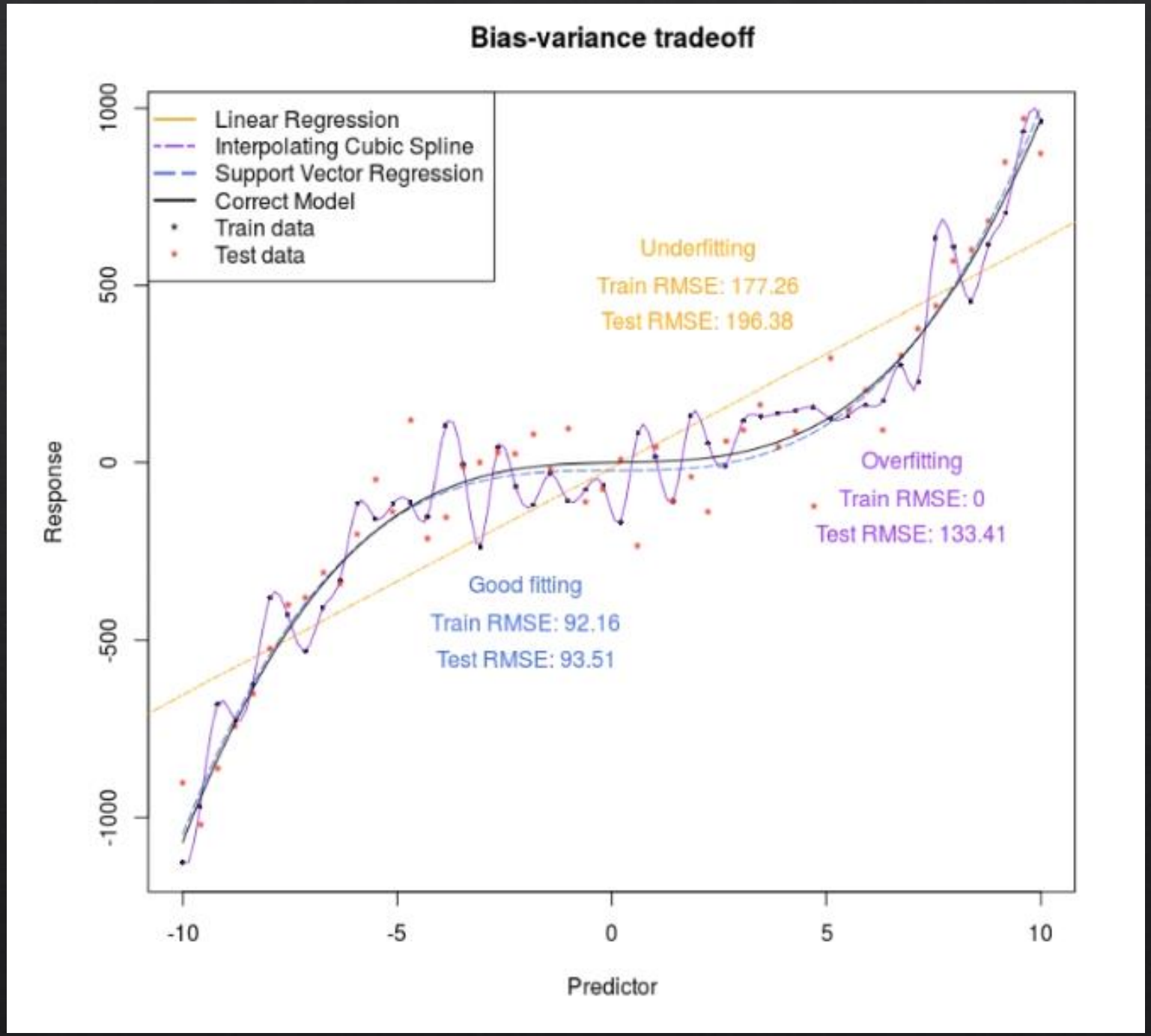


Performance do Modelo com os dados de Teste!

O modelo deve ser generalizável!



- Gráfico bias variance trandeoff

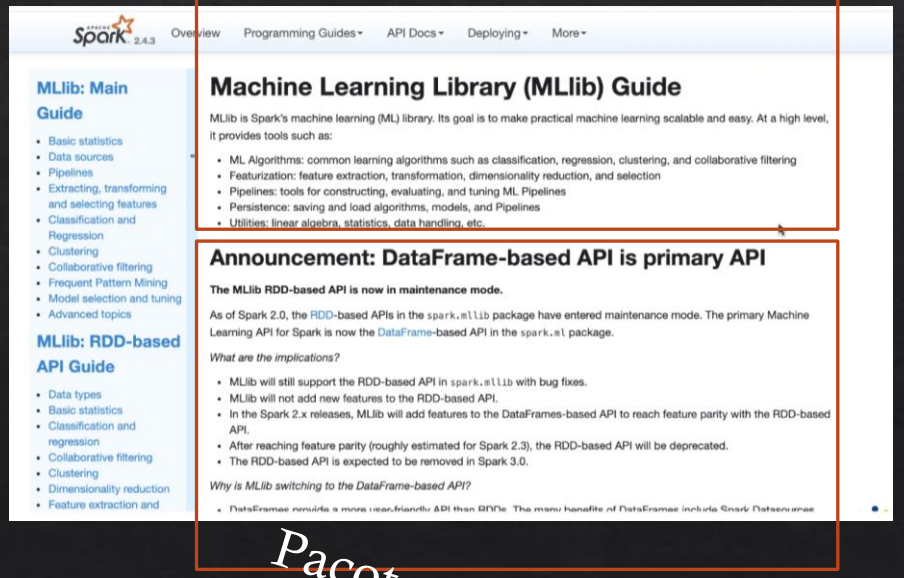


Biblioteca Machine Learning Library -MLlib

- Documentação: https://spark.apache.org
- O Mllib possui 5 módulos de aplicação do Apache Spark (Core), 4 deles são:
  - Spark SQL
  - Spark Streaming
  - Spark GraphX
  - Spark Mllib: (possui 2 APIs dentro do Mllib)

    Mllib: Main Guide (baseado em dataframe, mais recente)

    Mllib: RDD-based API Guide (baseado em RDD, mais antigo)



OBS: Recurso será descontinuado no future: API baseada em RDD!
(a partir da versão 3.0) (Spark Mllib)

Machine Learning com PySpark para executar em cluster

Os conceitos de Machine Learning são os mesmos, porém para executar em cluster usaremos:

- Python – Para construção dos modelos.
- Spark – Para processamento.

Principais Algoritmos de Machine Learning suportados pelo Apache Spark:

- Mllib – Regressão Linear (Aprendizagem supervisionada)
- Mllib – Decision Tree (Aprendizagem supervisionada)
- Mllib – Random Forest (Aprendizagem supervisionada)
- Mllib – Naïve Bayes (Aprendizagem supervisionada)
- Mllib – Clustering-K-Means (Aprendizagem Não-supervisionada)
- Mllib – Sistemas de Recomendação

Consultar a documentação para saber quais os pré requisitos necessários sempre que for utilizar um determinado algoritmo!

Pré-Processamento dos dados no Spark*

- Alguns algoritmos de ML do Spark, especialmente de Regressão, requerem que os dados estejam em um formato específico para treinar o algoritmo, devemos passar os dados em um formato de vetor (denso), pois os dados serão processados de maneira distribuída em um cluster.

Conceitualmente são o mesmo objeto:

- **Vetores esparsos:** são vetores que tem **muitos** valores como **zero**.
- **Vetor denso**: é quando a maioria dos valores no vetor são **diferentes de zero.**
- A maioria dos Algoritmos com Spark espera receber os atributos em um formato de vetor, se for denso ou esparso, Irá depender do dataset original.

De 7 posições passa a ocupar 2 posições:

$$\text{dense} : \underbrace{1.}\ \underbrace{0.}\ \underbrace{0.}\ \underbrace{0.}\ \underbrace{0.}\ \underbrace{0.}\ \underbrace{3.}$$

$$\text{sparse} : \begin{cases} \text{size} : 7 \\ \text{indices} : \underbrace{0}\ \underbrace{6} \\ \text{values} : \underbrace{1.}\ \underbrace{3.} \end{cases}$$

Processos de Machine Learning

1. Definição do problema a ser resolvido
2. Imports, Spark Session e Carga dos dados
3. Limpeza, manipulação e transformação dos dados
4. Análise exploratória para compreender a relação entre os de dados
5. Pré-processamento dos dados
6. Machine Learning

Em cada uma dessas etapas são aplicadas diferentes ferramentas de acordo com o dataset e o problema a ser resolvido.

End