**Lista de Exercícios sobre Extração do Conhecimento com   
Mineração de Dados.**

**Exercícios de Programação em Python: Classificação, Regressão e Clusterização**

**Classificação**

1. **Classificação com Regressão Logística**
   * Implemente um modelo de Regressão Logística para classificar o conjunto de dados *Iris* (ou qualquer outro). Avalie o modelo utilizando a métrica de acurácia e a matriz de confusão.
2. **Classificação com K-Nearest Neighbors (KNN)**
   * Construa um modelo KNN para prever a classe de um conjunto de dados (por exemplo, *Wine*). Utilize validação cruzada para ajustar o número de vizinhos e avalie o desempenho com a métrica *F1-Score*.
3. **Classificação com Support Vector Machines (SVM)**
   * Treine um modelo de SVM no conjunto de dados *Breast Cancer*. Use a métrica de precisão (precision) e *recall* para avaliar o modelo.
4. **Classificação com Árvores de Decisão**
   * Implemente uma árvore de decisão para classificar o conjunto de dados *Titanic*. Use a métrica ROC-AUC para avaliar a performance do modelo.
5. **Classificação com Random Forest**
   * Crie um modelo Random Forest para prever dígitos escritos à mão (utilize o conjunto de dados *Digits* do Scikit-Learn). Avalie o modelo utilizando a matriz de confusão e o *F1-Score*.
6. **Classificação com Gradient Boosting**
   * Treine um modelo de Gradient Boosting no conjunto de dados *Breast Cancer*. Compare o desempenho utilizando a métrica ROC-AUC e a curva ROC.
7. **Classificação Multiclasse com Naive Bayes**
   * Utilize o Naive Bayes para realizar classificação multiclasse no conjunto de dados *Iris*. Avalie o desempenho utilizando a precisão média ponderada (weighted precision) e a matriz de confusão.

**Regressão**

1. **Regressão Linear Simples**
   * Implemente a regressão linear simples para prever o preço de casas usando o conjunto de dados *Boston Housing*. Avalie o modelo utilizando o R² e o erro médio absoluto (MAE).
2. **Regressão Linear Múltipla**
   * Utilize regressão linear múltipla para prever o consumo de energia com base em várias variáveis. Avalie o desempenho utilizando o erro médio quadrático (MSE) e o R².
3. **Regressão com KNN**
   * Aplique KNN para realizar uma tarefa de regressão em um conjunto de dados como *Airfoil Self-Noise*. Compare o desempenho do modelo com a regressão linear usando MSE.
4. **Regressão com Support Vector Regression (SVR)**
   * Treine um modelo de SVR no conjunto de dados *Boston Housing* para prever os preços de casas. Avalie o modelo utilizando MAE e MSE.
5. **Regressão com Árvores de Decisão**
   * Implemente uma árvore de decisão para regressão no conjunto de dados *Bike Sharing*. Utilize a métrica R² e MAE para avaliação do desempenho.
6. **Regressão com Random Forest**
   * Crie um modelo de Random Forest para prever o preço de carros. Compare os resultados com um modelo de regressão linear usando MSE e R².

**Clusterização**

1. **Clusterização com K-Means**
   * Implemente o algoritmo K-Means no conjunto de dados *Iris* (removendo as labels). Avalie a qualidade da clusterização utilizando o coeficiente de Silhueta e a inércia dos clusters.
2. **Clusterização com DBSCAN**
   * Use o algoritmo DBSCAN para agrupar dados de clientes de uma loja. Avalie o desempenho da clusterização utilizando o coeficiente de Silhueta e faça uma análise visual dos clusters.