Análise Comparativa entre KNN, DMC e Naive Bayes

Diferentes Conjuntos de Dados

:

RELATÓRIO DOS RESULTADOS ALCANÇADOS

Gleilson Pedro Fernandes - PPGCC*

RESUMO

Este estudo apresenta uma análise comparativa entre os algoritmos de classificação, KNN (K-Nearest Neighbors), DMC (Classificador de Mínima Distância) e Naive Bayes, em três conjuntos de dados distintos: Iris, Coluna Vertebral, Breast Cancer, Dermatology e Dados Artificiais. Os algoritmos foram avaliados em termos de acurácia média e desvio padrão em cada conjunto de dados, bem como por meio da análise das matrizes de confusão, no código foi feito plot de superfície de decisão para cada classificador e um plot de barras com a probabilidade posteriori para cada classe das bases. O objetivo é determinar qual algoritmo é mais adequado para cada conjunto de dados com base em sua capacidade de classificação.

INTRODUÇÃO

A classificação de dados é uma tarefa fundamental em aprendizado de máquina, com aplicações em diversas áreas. Neste estudo, focamos na comparação entre três algoritmos de classificação amplamente utilizados: KNN, DMC e Naive Bayes. Esses algoritmos diferem em suas abordagens para classificação, com o KNN baseando-se na proximidade dos vizinhos, o DMC na distância até os centroides das classes e o Naive Bayes que assume independência condicional entre os atributos. Investigamos a eficácia desses algoritmos em três conjuntos de dados diferentes: Iris, Coluna Vertebral e Dados Artificiais, foi implementada matriz de confusão para uma das realizações que foi escolhida através da que que obteve a acurácia mais próxima da média por ser a mais representativa no desempenho geral do modelo.

CONJUNTO DE DADOS

Os conjuntos de dados utilizados neste estudo são:

1. **Iris**: Este conjunto de dados consiste em medidas de características de flores de íris, com o objetivo de classificar as flores em três espécies: Setosa, Versicolor e Virginica.
2. **Coluna Vertebral**: Este conjunto de dados contém medidas de características vertebrais de pacientes, com o objetivo de classificar as amostras em três classes: DH (espondilolistese), SL (espondilolistese) e NO (normal).
3. **Breast Cancer:** Este conjunto contem características como forma e tamanho das células de tecido mamário e textura, obtida através imagens de biópsias.

4. **Dermatology:** Este conjunto de dados contem imagens de diversas doenças dermatológicas.
5. **Dados Artificiais**: Este conjunto de dados foi gerado artificialmente a partir de duas classes, com distribuições gaussianas multivariadas.

METODOLOGIA

Algoritmos de Classificação

Foram utilizados três algoritmos de classificação:

- **KNN (K-Nearest Neighbors)**: Este algoritmo classifica uma amostra baseando-se nas classes dos k vizinhos mais próximos no espaço de características.
- **DMC (Classificador de Mínima Distância)**: Este algoritmo calcula a distância de uma amostra até os centroides das classes e atribui a classe com a menor distância.
- **Naive Bayes (se baseia no teorema de bayes)**: este classificador assume independência condicional entre os atributos, calcula a probabilidade a posteriori de um ponto pertencer a cada classe e classifica na classe com a maior probabilidade.

AVALIAÇÃO DE DESEMPENHO

O desempenho dos algoritmos foi avaliado utilizando validação cruzada para cada conjunto de dados. Foram calculados as acurácias médias e os desvios padrão para o KNN, DMC e Naive Bayes. Além disso, foram geradas matrizes de confusão para avaliar a precisão das classificações, e analisados plots de superfície de decisão e distribuição das classes e pontos de treinamento e teste mostrando as gaussianas.

CONSIDERAÇÕES FINAIS

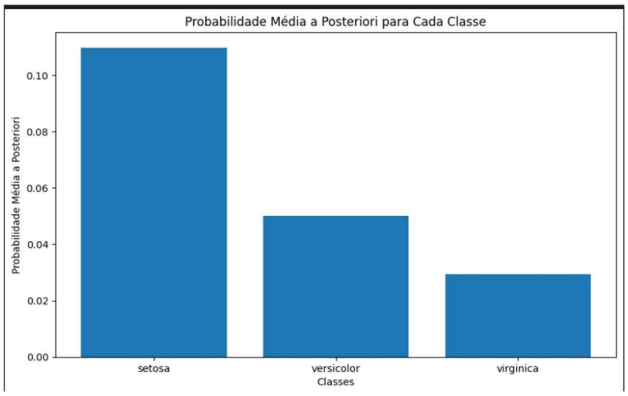
Resultados

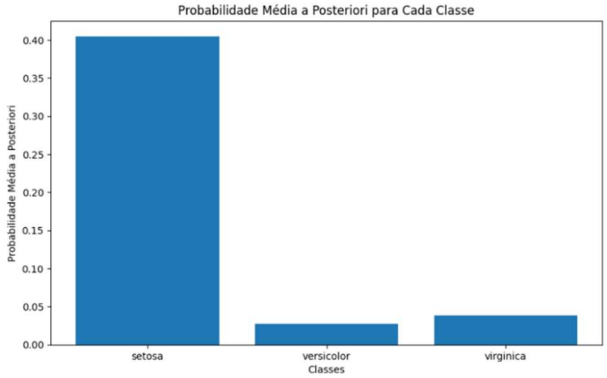
Dataset - Iris			
	KNN	**DMC**	**Naive Bayes**
acurácia	**0.8958**	**0.92955**	**0.94833**
desvio padrão	**0.0345**	**0.02675**	**0.02466**
realização	**0**	**5**	**1**

Dataset – Vertebral Column			
	KNN	**DMC**	**Naive Bayes**
acurácia	**0.7051**	**0.62820**	**0.69230**
desvio padrão	**0.0591**	**0.05053**	**0.04886**
realização	**7**	**19**	**6**

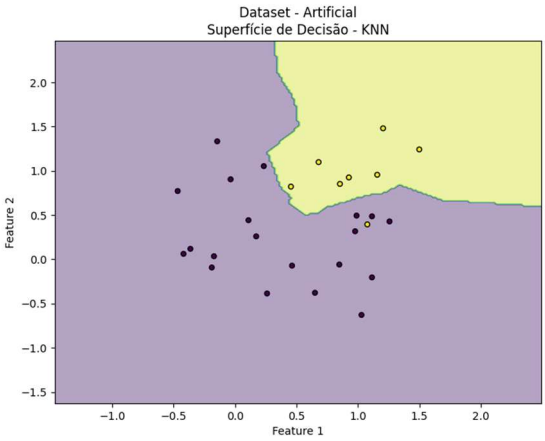
Dataset – Artificial			
	KNN	**DMC**	**Naive Bayes**
acurácia	**0.7051**	**0.62820**	**0.69230**
desvio padrão	**0.0591**	**0.05053**	**0.04886**
realização	**7**	**19**	**6**

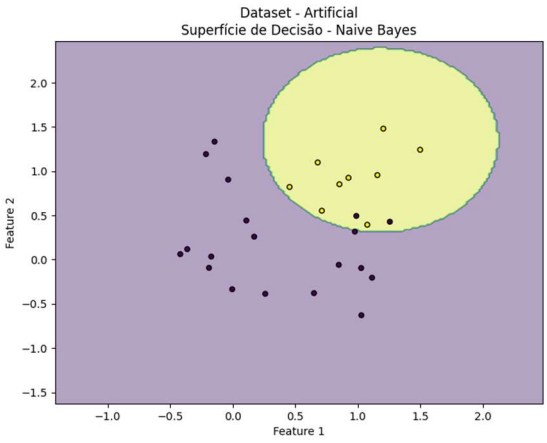
Ao executar as realizações do plot de probabilidade média a posteriori para cada classe, foi interessante ver como muda de acordo com a realização:



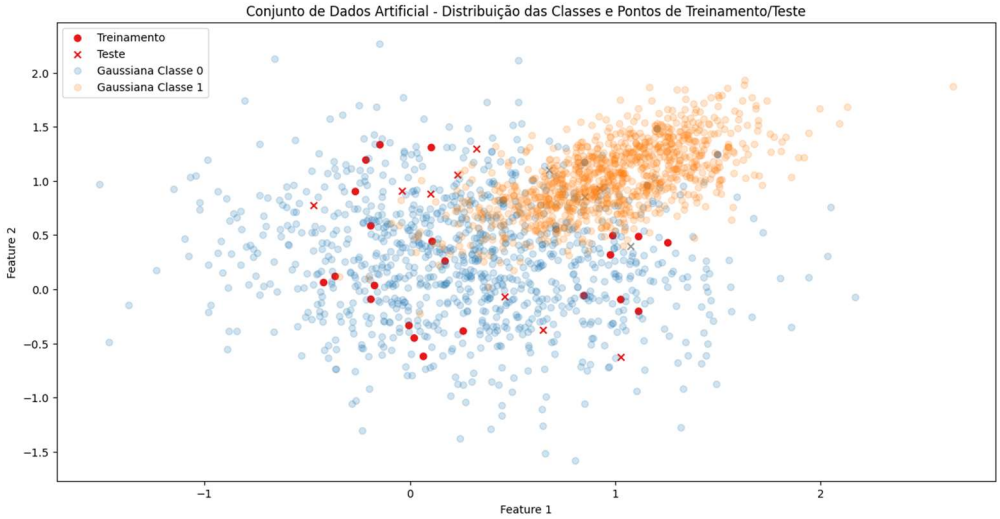


Também ocorre uma mudança significativa nos plots de superfície de decisão:





Os plots onde mostram os conjuntos de dados e treino da uma idéia da distribuição dos dados no conjunto, pode ser utilizado para identificar anomalias e também serve para avaliação da adequação do modelo se os dados estão assumindo corretamente a distribuição dos dados.



CONCLUSÃO

Os resultados variaram de acordo com a base de dados, o KNN é um classificador simples, mas tem a desvantagem de se o k não for bem ajustado ele perde performance, e dependendo do dataset a distância euclidiana pode não ser a melhor escolha. Já o classificador DMC, funciona muito bem com dados de baixa dimensionalidade também é um classificador simples, porém pode não ser tão útil se o dataset for de alta dimensionalidade. O Naive Bayes, é um classificador muito eficiente para treino e teste e se mostrou funcionar bem em ambos os dataset testados, porém seu ponto fraco é que ele pode ser sensível a atributos irrelevantes e a suposição de independência condicional que é como ele trabalha pode não funcionar para todos os dados. Ambos os classificadores são simples e foram aplicados chegando a um resultado satisfatório, ao olhar para um todo o Naive Bayes pareceu mais preciso nos datasets que ele foi aplicado em comparação com o KNN e o DMC, apesar de em alguns momentos o Naive Bayes necessitar de um pouco mais de processamento.

REFERÊNCIAS

Documentação numpy - https://numpy.org/doc/

Documentação matplotlib - https://matplotlib.org/stable/users/index.html

Documentação pandas - https://pandas.pydata.org/docs/

Documentação scikit-learn - https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html

Satckoverflow - https://stackoverflow.com