

Relatório Final do Trabalho Prático relacionado ao algoritmo KNN

GCC128 - Inteligência Artificial:

Sistemas de informação – 14A

Gustavo de Jesus Teodoro - 202311146

Thiago Lima Pereira - 202310057



Lavras - MG

1. Resultados da Comparação

O gráfico de acurácia (não presente na saída de texto, mas gerado pelo código) mostra o desempenho geral dos modelos. A implementação do **Sklearn** apresentou acurácia ligeiramente superior para $k=1$, alcançando 97.37%, contra 94.74% da versão **Hardcode**. Para $k=3$, a acurácia se igualou em 94.74% para ambas. Já para $k=5$ e $k=7$, a versão **Sklearn** manteve a acurácia de 94.74%, enquanto a **Hardcode** teve uma leve queda, para 92.11%.

Precisão por Classe

A análise da precisão por classe revelou insights mais detalhados:

- **Iris-setosa**: Ambas as implementações alcançaram 100% de precisão para todos os valores de k , mostrando que essa classe é facilmente separável.
- **Iris-versicolor**: A precisão da versão **Hardcode** para $k=1$ e $k=3$ foi de 90.91%, caindo para 90% para $k=5$ e $k=7$. Já a versão **Sklearn** manteve uma precisão de 93% para $k=1$ e 92% para os demais valores de k .
- **Iris-virginica**: A precisão da versão **Hardcode** para $k=1$ e $k=3$ foi de 90.91%, caindo para 83.33% para $k=5$ e $k=7$. A versão **Sklearn** demonstrou maior estabilidade, com precisão de 100% para $k=1$ e 92% para $k=3$, $k=5$ e $k=7$.

Revocação por Classe

A análise da revocação também indicou diferenças de desempenho:

- **Iris-setosa**: Ambas as implementações alcançaram 100% de revocação para todos os valores de k .
- **Iris-versicolor**: A **Hardcode** teve revocação de 90.91% para $k=1$ e $k=3$, caindo para 81.82% para $k=5$ e $k=7$. A **Sklearn** manteve a revocação em 100% para $k=1$ e 92% para os demais valores.
- **Iris-virginica**: A revocação da versão **Hardcode** foi de 90.91% para todos os valores de k . A versão **Sklearn** apresentou revocação de 92% para $k=1$ e 92% para os demais, indicando uma performance mais consistente.

Análise e Conclusão

A comparação entre as duas abordagens demonstra a principal vantagem de se utilizar uma biblioteca otimizada como o **scikit-learn**: **robustez e consistência**. A versão **Sklearn** não apenas obteve resultados de acurácia e precisão ligeiramente melhores para a maioria dos casos, mas também manteve um desempenho mais estável e confiável à medida que o valor de k mudou.

A implementação **Hardcode**, embora didaticamente valiosa para a compreensão do funcionamento interno do algoritmo, apresentou variações de desempenho mais acentuadas e resultados inferiores. Isso pode ser atribuído a diferenças na implementação da função `train_test_split` e no tratamento dos dados, como a não utilização da estratégia de `stratify=y` do **scikit-learn**, que garante que a proporção de cada classe seja mantida nos conjuntos de treino e teste. A implementação da matriz de confusão e do cálculo de precisão e revocação também pode ter gerado pequenas discrepâncias nos resultados.

Em suma, para fins de aprendizado, a implementação **Hardcode** é um exercício fundamental. Para aplicações práticas e produção, no entanto, o uso de bibliotecas como **scikit-learn** se demonstrou uma abordagem mais recomendada, devido à sua eficiência, otimização e estabilidade.