



Universidade Tecnológica Federal do Paraná
Campus Campo Mourão
Departamento de Computação - DACOM
Prof. Dr. Diego Bertolini
Disciplina: BCC35-G - Inteligência Artificial



Conteúdo: Redes Neurais Artificiais
Data de Entrega: 22/05/2023

Objetivo: Entender o funcionamento da biblioteca Scikit-Learn e o algoritmo MLP.

1) Em caso de dúvidas, leia o documento disponível no próprio site:

https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html

3) Baixe os conjuntos de treinamento e teste para este e outros experimentos: [download](#)

5) Analise as bases de treinamento e teste e responda (ver os arquivos train and test):

a) Número de classes ; **10 Classes**

b) Número de Instâncias no Treinamento ; **1000 Instâncias**

c) Número de Instâncias no Teste ; **1000 instâncias**

6) Reporte a acurácia através do experimento acima usando o código em python ([aqui](#)). **93%**

7) Altere os parâmetros “hidden_layer_size” e veja se é possível melhorar o desempenho aumentando número de camadas e a quantidade de neurônios. Avalie:

`solver{'lbfgs', 'sgd', 'adam'}`

`learning_rate_initfloat`

`shufflebool, default=True`

`hidden_layer_sizes=(10): 94%`

`hidden_layer_sizes=(10,10): 93%`

`hidden_layer_sizes=(100, 100, 100): 95%`

`hidden_layer_sizes=(500, 500, 500, 500): 96%`

Segunda Parte (Podem Continuar a partir do trabalho do SVM)

Utilizando as características extraídas na atividade de extração de features. Avalie os vetores de características extraídos para os cinco diferentes classificadores (no código acima já temos todos os classificadores comentados).

Demonstre os resultados individuais;

Utilize o mesmo template para os experimentos anteriores. Só descrever as taxas encontradas de forma clara. Não há necessidade de escrever relatório.

Com base nos resultados dos cinco classificadores utilizando as características (3x3) extraídas na atividade de extração de features, podemos fazer algumas análises:

KNN obteve uma acurácia de 76% com apenas 1 vizinho e utilizando a distância euclidiana. Essa acurácia é razoável, mas não é muito alta, o que sugere que o modelo pode não estar capturando completamente a complexidade do conjunto de dados.

SVM obteve a maior acurácia entre os classificadores, com 88%. O uso do kernel RBF pode ter permitido que o SVM aprendesse fronteiras de decisão mais complexas e capturasse melhor as relações entre os dados.

A rede neural obteve uma acurácia de 84% com camadas ocultas de tamanho (500, 500, 500, 500) e utilizando o otimizador Adam. A acurácia é bastante satisfatória, indicando que a rede neural foi capaz de aprender padrões relevantes nos dados. No entanto, é importante observar que a escolha dos hiperparâmetros (como o número de camadas ocultas e seus tamanhos) pode ter um impacto significativo no desempenho do modelo.

Random Forest obteve uma acurácia de 81% com uma profundidade máxima de árvore de 30. A acurácia é considerada boa, indicando que a Random Forest foi capaz de criar um conjunto diversificado de árvores de decisão e obter um bom desempenho geral.

A árvore de decisão obteve a menor acurácia, com 67%. Isso sugere que a árvore de decisão pode estar sofrendo de overfitting, ou seja, memorizando os dados de treinamento em vez de generalizar bem para novos dados.

Em geral, o SVM obteve a maior acurácia, seguido pela rede neural, Random Forest, KNN e árvore de decisão. É importante considerar que esses resultados podem variar dependendo da natureza do conjunto de dados, tamanho do conjunto de treinamento, escolha de hiperparâmetros e outros fatores. Portanto, é essencial realizar uma análise mais aprofundada e considerar diferentes métricas de avaliação antes de selecionar o melhor classificador para um determinado problema.