PROVA 2 - INTELIGÊNCIA ARTIFICIAL

Elabore um único relatório com todos os procedimentos e resultados obtidos em cada questão, argumentando os resultados. Só será aceito o relatório com as soluções das questões. As questões podem ser implementadas utilizando qualquer linguagem de programação ou o *Weka*.

Classificação

1. Utilizando a base disponível no link (https://archive.ics.uci.edu/ml/datasets/Autism+Screening+Adult) elabore uma solução utilizando dois algoritmos de aprendizagem de máquina do seu conhecimento para classificar em doente ou saudável (controle). Os resultados dessa questão deverão ser descritos detalhadamente no relatório através de três pontos principais:
    - Análise da base de dados: identificar instâncias com atributos incompletos, gerar matriz de correlação, identificar a presença de *outliers* e verificar se as classes estão balanceadas. O balanceamento deverá ser ilustrado por meio de gráficos (e.g. histograma);
    - Justifique a escolha dos dois algoritmos de aprendizagem de máquina utilizados e discuta os resultados obtidos em ambos. Explique porque você acredita que os algoritmos escolhidos são mais apropriados para o problema;
    - Analise os resultados considerando matriz de confusão, especificidade, sensibilidade, medida f1 e acurácia. Descreva detalhadamente os resultados obtidos por cada métrica, justificando a diferença entre eles.

    Dicas:
    - A partir da análise da base de dados, para bons resultados possivelmente será necessário pré-processar os dados. Para identificar os *outliers*, recomenda-se ilustrá-los graficamente (e.g. boxplot).
    - Divida o conjunto em treinamento, validação e teste como, por exemplo, 70%, 10%, 20% respectivamente.
    - Para analisar os resultados de cada métrica, identifique os falsos positivos, falsos negativos, verdadeiros positivos e verdadeiros negativos.

    Bônus: Altere dois parâmetros de cada algoritmo de aprendizagem de máquina utilizados na questão e discuta os resultados obtidos. Exemplo: alterar a quantidade de k vizinhos e a função de distância utilizada, alterar a função kernel do SVM, alterar a arquitetura da rede neural (e.x. camadas e função de ativação, alterar o otimizador e a taxa de aprendizado).

2. Utilizando a base disponível no link (https://archive.ics.uci.edu/ml/datasets/Avila), crie os datasets a seguir:

Dataset	% de instâncias
Treino	60%
Validação	20%

Teste	20%

- Elabore uma rede neural de duas camadas para classificação do banco de dados.
- Ao fim do treinamento, avalie o desempenho da rede utilizando a matriz de confusão com o dataset de teste e mostre o valor de acurácia.

Observações:

- Utilize apenas o arquivo **avila-tr.txt**.
- A camada de saída da rede deverá conter um neurônio para cada classe.
- Utilize o dataset de validação para criar algum critério de parada no treinamento.

Bônus: defina uma arquitetura de rede neural ou modelos de *deep learning* que ultrapassem 75% de acurácia.

Regressão

3. Utilizando as bases de treinamento e testes disponível nos links: base_de_treinamento (https://drive.google.com/file/d/1DHigBm7_1kGFiG3vk3zaKnLwHbX-QxfO/view) e base_de_testes (https://drive.google.com/file/d/1NuN9yGSewm7AyfwpWP9Te8AfzvIYRL_y/view) , apresente duas soluções de aprendizagem de máquina que consigam taxas de erro médio abaixo de 2,2m na predição da localização (em termos de coordenadas X e Y) dos pontos da base de testes, utilizando regressão. Os conjuntos de dados (treinamento e testes) contêm o local (em coordenadas X e Y) precedido por 5 leituras de RSSI de 5 APs (Access Points) para cada metro quadrado do vão do andar térreo do LASER e de duas salas existentes no térreo.

   Dica:

   - Teste com vários algoritmos de aprendizagem de máquina para identificar os que apresentam melhores resultados;
   - Para calcular o erro médio da predição da localização nos pontos da base de testes utilize a fórmula da hipotenusa.

Bônus: utilize alguma arquitetura de rede neural recorrente (exemplo: LSTM) para resolver o problema, uma vez que os pontos das bases de treinamento e testes representam pontos de uma trajetória (série temporal).

Clusterização

4. Utilizando a base de clusterização encontrada nesse link (https://drive.google.com/file/d/1_702eOQbimT1HhTuozwMEKV6HHOykLaJ/view), execute os algoritmos de clusterização citados a seguir e compare os resultados.

    - Execute os algoritmos de agrupamento K-means e Hierárquico com os seguintes valores de K: 2, 5, 10 e 100. Compare os agrupamento resultantes dos 2 algoritmos.
    - Escolha um número fixo de K e altere o parâmetro do K-Means referente ao número máximo de iterações: 1, 10 e 100 e o parâmetro de Linkage do Hierárquico, quais diferenças puderam ser observadas?
    - Faça uma comparação entre os 2 algoritmos, qual você acha que teve o melhor desempenho e por quê?

Bônus: normalizar os dados e executar novamente os algoritmos para analisar os novos resultados gerados. Em seguida, fazer os gráficos dos agrupamentos dos dados brutos vs dados normalizados, e um boxplot para mostrar a dispersão destes dados.