Trabalho Prático: Construção e Análise de Árvores de Decisão

As árvores de decisão são algoritmos de Machine Learning amplamente utilizados para resolver problemas de classificação e regressão. Elas funcionam como um fluxograma, onde cada decisão é baseada em uma variável do seu conjunto de dados, tornando o modelo fácil de entender e interpretar.

Algumas das aplicações das árvores de decisão incluem:

- Entender padrões nos dados de forma clara e organizada.
- Tomar decisões com base em regras explícitas e fáceis de interpretar.
- Aprender a partir de exemplos, sem precisar de cálculos complexos.

O objetivo deste trabalho é que vocês, em duplas, explorem o uso de árvores de decisão para resolver um problema real, desde a escolha do conjunto de dados até a análise dos resultados obtidos.

Passo 1: Escolha do Dataset

A dupla deverá escolher um dataset que contenha no mínimo 500 amostras e possua pelo menos cinco variáveis preditoras, além da variável alvo (rótulo do problema). O dataset poderá ser utilizado para uma tarefa de **classificação** ou **regressão**, dependendo do problema que a dupla deseja resolver.

É fundamental que a dupla defina claramente qual problema será resolvido.

Exemplos de problemas:

- **Classificação**:
  - Prever se um paciente tem uma determinada doença com base em exames clínicos.
  - Determinar se um cliente pagará um empréstimo com base em seu histórico financeiro.
  - Identificar se um e-mail é spam com base em seu conteúdo e metadados.
- **Regressão**:
  - Estimar o preço de um imóvel com base em suas características, como tamanho, localização e número de quartos.
  - Prever a demanda por um produto em determinado período com base em dados históricos de vendas.
  - Estimar o tempo de entrega de uma encomenda com base em fatores como distância, tráfego e método de envio.
- O conjunto de dados pode ser obtido em plataformas como Kaggle, UCI Machine Learning Repository, entre outras.

Passo 2: Divisão dos Dados

A dupla deve dividir a base de dados em conjuntos de treino e teste para avaliar o desempenho do modelo. Geralmente, essa divisão é feita em proporções como 70% para treino e 30% para teste, mas isso pode variar dependendo do tamanho da base e do problema. Essa divisão é crucial porque:

- O conjunto de teste representa dados nunca vistos pelo modelo, permitindo avaliar como ele se comportará na prática.
- Separar os dados antes do pré-processamento garante que o modelo não tenha acesso a informações do conjunto de teste durante o treinamento, o que poderia levar a uma avaliação otimista e irrealista.
- Comparar o desempenho do modelo no treino e no teste ajuda a identificar se ele está memorizando os dados (overfitting) ou generalizando bem.

OBS: Sem essa divisão, o modelo pode parecer performar bem durante o desenvolvimento, mas falhar ao ser aplicado a novos dados.

Passo 3: Pré-Processamento

Após selecionar e dividir o dataset, a dupla deverá realizar uma etapa de pré-processamento no conjunto de treino, caso seja necessário. A dupla deve analisar as características do dataset e aplicar as técnicas adequadas, como:

- Tratamento de valores ausentes (por exemplo, preenchimento com médias, medianas ou remoção de registros incompletos).
- Codificação de variáveis categóricas (por exemplo, one-hot encoding ou label encoding).
- Normalização ou padronização dos dados para garantir que todas as variáveis estejam na mesma escala.

Importante:

- Todas as estatísticas (como médias, desvios padrão ou categorias) devem ser calculadas apenas no conjunto de treino.
- As mesmas transformações devem ser aplicadas ao conjunto de teste, sem recalcular estatísticas. Isso evita vazamento de dados e garante uma avaliação justa do modelo.

Um bom pré-processamento pode fazer uma grande diferença na qualidade das previsões. Ignorar essa etapa ou realizá-la de forma inadequada pode comprometer o desempenho do modelo, mesmo que ele seja teoricamente robusto (métricas com valor alto). Portanto, é essencial dedicar atenção ao pré-processamento.

Passo 4: Criação da árvore de decisão

Em seguida, a dupla deverá construir um modelo de árvore de decisão utilizando uma biblioteca como o Scikit-Learn. A escolha dos hiperparâmetros, como a profundidade máxima da árvore e o critério de divisão, deve ser feita com justificativas baseadas nos dados e na necessidade de equilibrar desempenho e interpretabilidade. Modelos muito profundos podem memorizar os dados de treino (overfitting), enquanto modelos rasos podem não capturar padrões importantes.

Aqui estão os principais hiperparâmetros que a dupla deve considerar:

- Profundidade máxima da árvore (max_depth): Define o número máximo de níveis (ou "profundidade") que a árvore pode ter. Controlar esse parâmetro é importante para evitar overfitting (quando o modelo se ajusta demais aos dados de treino → decora os dados) ou underfitting (quando o modelo é muito simples para capturar padrões importantes).
- Critério de divisão: Define a métrica usada para decidir como dividir os dados em cada nó da árvore.
- Para classificação, os critérios mais comuns são o Índice de Gini (mede a impureza dos dados) e a Entropia (mede a desordem).
- Para regressão, geralmente são utilizados o MSE (erro quadrático médio) ou o MAE (erro médio absoluto), que avaliam a diferença entre os valores previstos e os reais.
- Número mínimo de amostras por folha (min_samples_leaf): Define o número mínimo de amostras necessárias para criar uma folha (nó final) da árvore. Esse parâmetro ajuda a evitar folhas com poucas amostras, o que pode indicar overfitting.
- Número mínimo de amostras para divisão (min_samples_split): Define o número mínimo de amostras necessárias para dividir um nó interno. Esse parâmetro controla a complexidade da árvore, evitando divisões que não agregam valor ao modelo.

Passo 5: Avaliação do Modelo

Após a criação da árvore de decisão, a dupla deve avaliar o desempenho do modelo utilizando as métricas.

Métricas para classificação:

- Acurácia: Porcentagem de previsões corretas.
- Precisão: Proporção de previsões positivas que foram corretas.
- Recall: Proporção de casos positivos reais que foram identificados corretamente.
- F1-score: Média harmônica entre precisão e recall, útil para problemas com classes desbalanceadas.

Métricas para regressão:

- MSE (Erro Quadrático Médio): Mede a média dos erros ao quadrado.
- MAE (Erro Médio Absoluto): Mede a média dos erros absolutos.

Passo 6: Interpretação do Modelo

Após a avaliação, a dupla deve interpretar e explicar os resultados obtidos. Para isso, pode-se utilizar:

- Visualização da árvore:
  - Usar ferramentas como plot_tree do Scikit-Learn para visualizar a estrutura da árvore.
  - Explicar como a árvore toma decisões com base nas variáveis.
- Importância das variáveis:
  - Analisar quais variáveis tiveram maior impacto nas decisões da árvore.
  - Discutir por que essas variáveis são relevantes para o problema (comparando as métricas delas).
- Regras geradas:
  - Examinar as regras criadas pela árvore e verifique se fazem sentido no contexto do problema (verificar se fazem sentido no “mundo real”).
  - Explicar como essas regras podem ser úteis para a tomada de decisão.

Observações de Entrega:

Data: 26/03
Enviar para: luisalvaro@inf.ufsm.br

O que deve ser entregue:

1. Código Python (.py): Comentado, explicando as partes principais (principalmente como cada passo foi feito) e as escolhas feitas (por exemplo, hiperparâmetros da árvore de decisão).
2. Dataset (.csv): O dataset escolhido.
3. Relatório (.pdf):
   - Contendo os principais resultados obtidos e a interpretação da árvore, feita no passo 6.
   - O relatório pode conter prints dos resultados e das árvores, com explicações logo abaixo.
   - A maneira como o relatório será feito fica a critério da dupla, desde que esteja claro e organizado.
4. Vídeo (.mp4):
   - Com duração entre 5 e 10 minutos, explicando rapidamente o código e, principalmente os resultados finais.

Formato de Entrega:

- Todos os arquivos (código, dataset, relatório e vídeo) devem ser colocados em uma pasta zipada.
- O nome da pasta deve seguir o formato: NomeDaDupla.zip (substitua "NomeDaDupla" pelo nome dos integrantes) e **ambos os integrantes da dupla devem enviar a pasta**.

OBS: Qualquer dúvida, enviar para ecradaelli@inf.ufsm.br