Atividade Prática
Avaliação de um Algoritmo de Indução de Árvores de Decisão

Este trabalho consiste na execução e validação de um algoritmo de indução de árvores. Os conjuntos de dados podem ser selecionados no repositório UCI (http://archive.ics.uci.edu/ml/) ou no site da ferramenta Weka (http://www.cs.waikato.ac.nz/ml/weka/). Pode também ser usado um dos conjuntos de dados do pacote sklearn.datasets, desde que apropriado para o tipo de tarefa, exceto os conjunto Iris e Linnerrud.

As etapas que devem ser executadas e relatadas são:

1. Selecionar um conjunto de dados adequado para problemas de classificação (conjunto com classes nominais);
2. Se for um conjunto de dados externo a biblioteca scikit-learn, ler o conjunto de dados no formato .csv com a função read_csv(...) (biblioteca Pandas);
3. Se for um conjunto de dados do pacote sklearn.datasets, carregar o conjunto com a função load_<nome_do_arquivo>;
4. Separar o conjunto de dados em matriz de atributos (X) e vetor de classes (y).
5. Transformar os atributos nominais em binários (se existirem), usando a classe OneHotEnconder do módulo sklearn.preprocessing ou usando a função get_dummies(...) da biblioteca Pandas;
6. Implementar uma função que, dado um conjunto de dados, separe esse conjunto em 10 folds (10 subconjuntos).
7. Dados os 10 folds, avaliar um algoritmo de indução de árvores de decisão usando a estratégia 10-fold cross-validation, ou seja, em cada uma das 10 iterações, treinar o algoritmo em 9 folds (folds de treinamento) e testar no fold restante (fold de teste), cada vez com um fold de teste diferente.
8. Quando treinando o algoritmo de indução de árvore de decisão, usar a função fit(...) da classe DecisionTreeClassifier do módulo sklearn.tree, com o atributo criterion='entropy';
9. Escolha um dos folds para gerar a figura da árvore de decisão com a função plot_tree(...) do módulo sklearn.tree;
10. Em cada um dos folds, fazer a classificação dos dados de teste usando a função predict(...) da classe DecisionTreeClassifier do módulo sklearn.tree;
11. Em cada um dos folds, fazer a avaliação do modelo gerado usando os dados de teste e mostrando os resultados das funções confusion_matrix(...) e classification_report(...) do modulo sklearn.metrics;
12. Ao final das 10 execuções do 10-fold cross-validation, apresentar as médias e desvios padrões dos 10 resultados das medidas de avaliação da função classification_report(...) do modulo sklearn.metrics.

Importante: o código deve ser documentado, ou seja, explique claramente todos os procedimentos adotados e descreva todas as etapas definidas. Deve ser entregue um arquivo do tipo IPython Notebook (.ipynb).