FGV EMAp
ESCOLA DE MATEMÁTICA APLICADA

Técnicas e Algoritmos em Ciência de Dados
Tarefa 1

Este trabalho deve ser entregue até 23 de março de 2023, às 8:00 da manhã.
As submissões tardias serão penalizadas em 10% **por hora de atraso**.

Tópicos de aprendizagem avaliados

Esta tarefa testará alguns conceitos básicos de Ciência de Dados e programação Python para análise de dados (Pandas, Seaborn). Especificamente, carregamento de dados, visualizações simples e avaliação de modelos de classificação e regressão pré-treinados.

Instruções

Identificação
Por favor, escolha um número aleatório de 6 dígitos e escreva-o na primeira célula do notebook. Certifique-se de manter uma cópia desse número, pois ele será usado para fornecer o feedback (por favor, evite números triviais, como 000000 ou 123456 – obrigado).

Submissão
Envie seus arquivos através do ECLASS. Os arquivos que você envia não podem ser lidos por nenhum outro aluno. Você pode substituir seu envio quantas vezes quiser, reenviando-o, embora apenas a última versão enviada seja mantida.

Se você tiver problemas, no último minuto, envie sua tarefa por e-mail como um anexo em alberto.paccanaro@fgv.br com o assunto "URGENTE – SUBMISSÃO TAREFA 1". No corpo da mensagem, explique o motivo de não enviar através do ECLASS.

IMPORTANTE

- Seu envio consistirá em um único Python notebook implementando suas soluções.
- O nome do arquivo será o número aleatório que o identifica (por exemplo, 568423.ipynb)
- Esta tarefa contém 3 partes. Certifique-se de que o seu código para todas as 3 partes é colocado nas células de código correspondentes no notebook.
- NÃO ENVIE NENHUM CONJUNTO DE DADOS, apenas o código.
- Qualquer função auxiliar que você utilizar deve ser incluída no notebook – não envie scripts adicionais.

CONSELHOS SOBRE EXERCÍCIOS DE BÔNUS

Observe que esses exercícios são mais difíceis e demorados do que os exercícios padrão. Eu aconselho você a ter uma para toda a tarefa antes de tentar responder a esses exercícios opcionais.

Todo o trabalho que você enviar deve ser exclusivamente seu próprio trabalho. As submissões serão verificadas para isso.

Critérios de avaliação

Esta atividade é obrigatória e vale 10% da sua nota final total para este curso. Para obter notas máximas para cada pergunta, você deve respondê-la corretamente, mas também completamente. Notas serão dadas por escrever códigos bem estruturados.

EXERCÍCIOS

Parte 1 – Carregamento e Visualização de Dados – valor desta seção: 20%

Baixe os dados do ECLASS

- A1_template.ipynb (em inglês)
- Iris.csv

Aqui estão as etapas que você precisará implementar:

a) Carregar os dados em um DataFrame pandas
b) Para cada feature no conjunto de dados, crie uma figura com duas subplotagens, uma acima da outra. No primeiro subgráfico, plote um histograma dos valores da feature para todas as classes combinadas. No segundo subgráfico, plote um histograma dos valores da feature para cada classe separadamente. Use a coluna "Species" como o identificador de classe, e uma cor diferente para cada classe.
Para cada feature, a figura deveria ser semelhante à Figura 1.

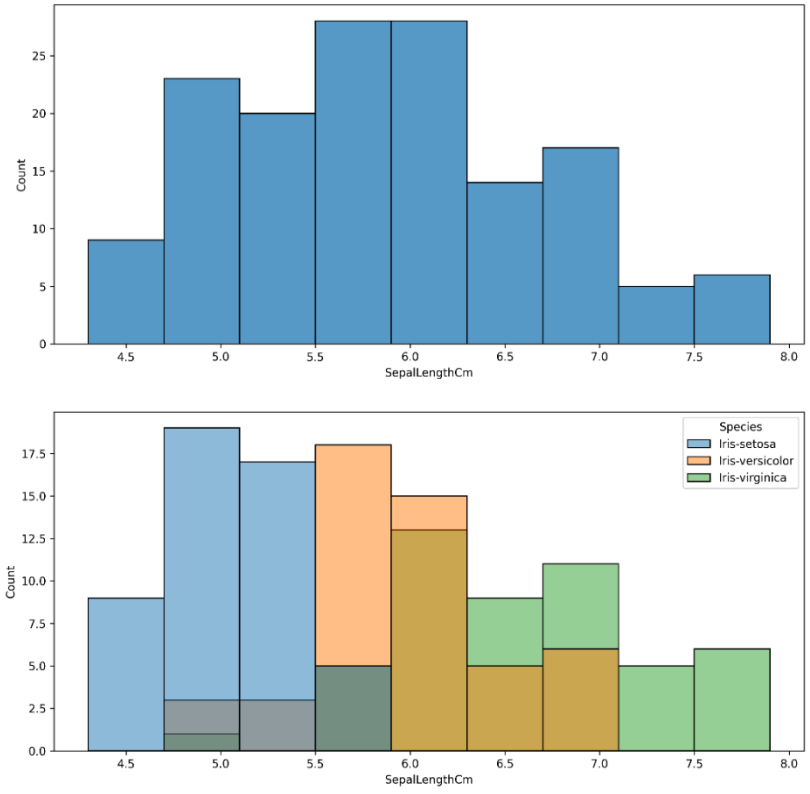

Figura 1

Parte 2 – Classificação Binária – Avaliação de Desempenho – valor desta seção: 50%

Baixe os dados do ECLASS.

- classifiers_dict.p
- mushroom_test_data.p

O objetivo deste exercício é avaliar o desempenho de vários modelos de classificação pré-treinados na predição se um cogumelo é comestível ou venenoso. Para isso, geraremos curvas ROC para cada algoritmo e determinaremos a área sob a curva. Usando essas métricas, decidiremos qual classificador é mais adequado para a tarefa. Observe que você precisará implementar o código para calcular a curva ROC para cada classificador.

1) Siga as instruções no notebook de template fornecido para carregar os classificadores e os dados de teste.
2) Para cada classificador:
   - calcular as saidas (seguindo as instruções do notebook de template).
   - calcular a matriz de confusão, a Taxa de Falsos Positivos (False Positive Rate - FPR) e a Taxa de Verdadeiros Positivos (True Positive Rate – TPR) em diferentes limiares (para isso, você terá que usar os rótulos verdadeiros encontrados em y_test).
   - produzir uma figura contendo uma curva ROC para cada classificador. O gráfico resultante deveria ser semelhante ao exibido na Figura 2. O gráfico precisará ter uma legenda para identificar os diferentes classificadores. A figura também deve conter a linha em 45 graus representando o desempenho de um classificador aleatório.
   - calcular a área sob a curva ROC (usando a função fornecida no notebook). Isso deve ser impresso na célula de código relevante no notebook fornecido.

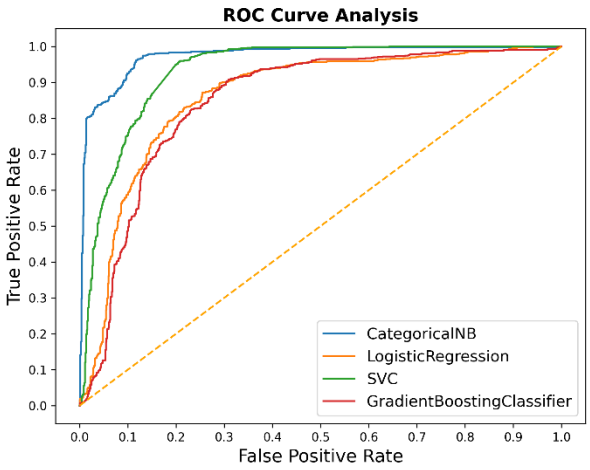

Figura 2

Importante: você não pode usar funções pré-existentes como sklearn.metrics.roc_curve, ou similares. Você precisa escrever seu próprio código para calcular a curva.

BÔNUS (10% de **notas extras):** Você ganha pontos de bônus se fizer o gráfico usando um DataFrame do Pandas e uma única chamada para uma função de plotagem.

Parte 3 – **Regressão - RMSE – valor desta seção: 30%**

Baixe os dados do ECLASS.

- linear_regression_model.p
- boston_testing_data.p

O objetivo deste exercício é avaliar o desempenho do nosso modelo de regressão na predição de preços de casas calculando a Raiz do Erro Quadrático Médio (Root Mean Squared Error - RMSE) de nossas predições no conjunto de testes.

1) Siga as instruções no notebook de template fornecido para carregar os modelos e os dados de teste.
2) Para este exercício, você terá que calcular:
   - Os valores preditos para o conjunto de testes (seguindo as instruções no notebook)
   - O RMSE para o conjunto de teste.

Importante: você não pode usar funções pré-existentes como sklearn.metrics.mean_squared_error, ou similares. Você precisa escrever seu próprio código para calcular o erro.

BÔNUS (5% de notas extras): *Você ganha pontos de bônus se calcular o RMSE vetorizando seu código, ou seja, sem loops.*