

Teste de Habilidades para Cientista de Dados

Introdução

Bem-vindo(a) ao teste de habilidades para a posição de Cientista de Dados. Este teste foi desenvolvido para avaliar suas competências em Python, Deep Learning, Machine Learning e MLOps. Certifique-se de ler todas as instruções com atenção e seguir as orientações fornecidas em cada seção.

Prazo de Realização

O prazo para a conclusão deste teste é de **7 dias corridos** a partir do momento em que ele foi recebido. É fundamental que todos os materiais sejam enviados dentro deste período.

Orientações para a Entrega

Ao final do teste, você deverá enviar:

1. **Todo o código desenvolvido:** Deve ser organizado e acompanhado de comentários explicativos sempre que necessário.
2. **Materiais utilizados:** Inclua datasets manipulados, arquivos auxiliares e qualquer outro recurso usado.
3. **Documentação e justificativas:** Forneça explicações claras para cada decisão tomada no desenvolvimento das soluções, bem como respostas detalhadas às questões conceituais.

O envio pode ser realizado por meio de um repositório no GitHub ou através do upload dos arquivos em um link compartilhado. Certifique-se de que todos os arquivos necessários estão incluídos.

Informações Importantes

- Links para os datasets:
 1. [Conjunto de vendas \(CSV Exemplo\)](#)
 2. [Conjunto Iris \(Dataset Iris\)](#)
 3. [Conjunto Titanic \(Dataset Titanic\)](#)
- Certifique-se de que os links estão funcionais antes de iniciar as atividades.

CrITÉrios de Avaliação

Suas respostas serão avaliadas com base nos seguintes critérios:

- Correção técnica e aderência ao que foi solicitado.
- Clareza e organização do código.
- Qualidade das justificativas e explicações.
- Criatividade e inovação nas soluções apresentadas.

Contato

Em caso de dúvidas durante o desenvolvimento do teste, entre em contato com o responsável pelo processo. Estamos à disposição para auxiliá-lo(a).

Desejamos um excelente desempenho no teste!

Este teste avalia suas habilidades em **Python, Deep Learning, Machine Learning e MLOps**. Por favor, leia cada pergunta com atenção e forneça suas soluções nas seções apropriadas. Você pode enviar seu código via um repositório do GitHub ou fazer o upload dos arquivos necessários.

Seção 1: Python

1.1 Manipulação de Dados

Carregue o conjunto de dados fornecido em formato CSV e realize as seguintes tarefas:

1. Calcule as vendas totais por região e categoria.
2. Identifique o produto de melhor desempenho em cada categoria.

Conjunto de dados: [Baixar Aqui](#)

(Este é um dataset de vendas de amostra com regiões e categorias que pode ser manipulado para essa tarefa.)

1.2 Implementação de Algoritmo

Escreva uma função para implementar o algoritmo de agrupamento **K-means** do zero em Python, sem usar bibliotecas externas como sklearn.

1.3 Tratamento de Erros

Escreva uma função que receba dois inputs e realize a divisão, tratando exceções como:

- Divisão por zero.
 - Tipos de entrada inválidos.
-

Seção 2: Deep Learning (25 pontos, 45 minutos)

2.1 Construir uma Rede Neural (10 pontos)

Usando **PyTorch**, treine uma rede neural simples no conjunto de dados **Iris**:

- Use **2 camadas ocultas** com ativações ReLU.
- A camada de saída deve usar softmax.
- Avalie a performance.

Conjunto de dados Iris: [Baixar Aqui](#)

2.2 Explicação de Conceitos

Explique a diferença entre:

1. **Dropout** e **Batch Normalization**.
 2. **Transfer Learning** e **Fine-Tuning**.
-

2.3 Ajuste de Hiperparâmetros

Dado um modelo pré-treinado, quais etapas você seguiria para:

- Evitar **overfitting**.
 - Reduzir o tempo de treinamento mantendo o desempenho.
-

Seção 3: Machine Learning

3.1 Construir um Modelo (15 pontos)

Usando o conjunto de dados **Titanic** fornecido:

1. Pré-processe os dados (tratar valores ausentes, codificar variáveis categóricas).
2. Treine um **Random Forest Classifier** para prever sobrevivência.
3. Avalie o modelo usando **precisão**, **recall** e **F1-score**.

Conjunto de dados Titanic: [Baixar Aqui](#)

3.2 Pergunta Conceitual

O que são **viés** e **variância** em machine learning? Forneça exemplos de:

- Um modelo com **alto viés**.
 - Um modelo com **alta variância**.
-

Seção 4: MLOps

4.1 CI/CD para Modelos de ML

Escreva um plano ou pseudocódigo para automatizar:

- O treinamento de um modelo de machine learning.
- A implantação em produção (usando ferramentas como **Docker**, **Kubernetes** ou plataformas em nuvem).

4.2 Implantação de Modelo (10 pontos)

Dado um modelo pré-treinado (model.pkl), escreva uma **API Flask** para servir previsões do modelo. Inclua endpoints para:

- /predict (aceita entrada JSON e retorna previsões).
 - /health (retorna "Serviço em execução").
-

4.3 Monitoramento e Retreinamento

Descreva como você:

- Monitoraria o desempenho de um modelo implantado.
 - Automatizaria o retreinamento quando o desempenho do modelo deteriorar.
-

Links:

1. **Conjunto de vendas:** [CSV Exemplo](#)
2. **Conjunto Iris:** [Dataset Iris](#)
3. **Conjunto Titanic:** [Dataset Titanic](#)

Se precisar de mais ajuda ou tiver dúvidas, fico à disposição!