databricks

Databricks Exam Guide

Databricks Certified Machine Learning Associate



[Fornecer Feedback sobre o Guia do Exame]

Atualizar Informação

Atualizações	Data
Exame publicado	1 de março de 2025
Guia do exame atualizado	1 de março de 2025

Por favor, verifique o guia do exame em nossas páginas da web duas semanas antes da data do seu exame para garantir que você está se preparando para a versão atual do exame.

Objetivo deste Guia de Exame

O objetivo deste guia de exame é fornecer uma visão geral do exame e do que ele abrange, para ajudá-lo a determinar sua prontidão para o exame. Este documento é atualizado sempre que há alterações no exame (e indica quando essas alterações entram em vigor) para que você possa se preparar com confiança. Este guia abrange a versão do exame que está ativa a partir da data de vigência listada acima.

Descrição do Público-Alvo

O exame de certificação Databricks Certified Machine Learning Associate avalia a capacidade de um indivíduo de usar o Databricks para realizar tarefas básicas de machine learning. Isso inclui a capacidade de compreender e usar o Databricks e seus recursos de machine learning, como AutoML, Unity Catalog e funcionalidades selecionadas do MLflow. O exame também avalia a capacidade de explorar dados e realizar Engenharia de Recursos. Além disso, avalia a construção de modelos por meio de treinamento, ajuste, avaliação e seleção. Por fim, é avaliada a capacidade de implantar modelos de machine learning. Espera-se que os indivíduos aprovados neste exame de certificação sejam capazes de concluir tarefas básicas de machine learning usando o Databricks e suas ferramentas associadas.

Sobre o Exame

Item	Detalhe
Número de questões pontuadas	48 questões pontuadas de múltipla escolha e de múltipla seleção
Possíveis questões não pontuadas	Podem ser incluídas até 10 questões não pontuadas (consulte "Conteúdo não pontuado" abaixo)
Número total de questões que podem aparecer	48 questões pontuadas, além das questões não pontuadas
Limite de tempo	90 minutos (o tempo adicional para as questões não pontuadas já está incluído)
Taxa de inscrição	US$ 200, mais impostos aplicáveis, conforme exigido pela legislação local
Forma de aplicação	On-line com supervisão ou em centro de testes
Idiomas disponíveis	Inglês; japonês; português brasileiro (pt-BR); coreano
Materiais de consulta	Não permitidos
Pré-requisito	Não há pré-requisito obrigatório, mas o treinamento relacionado é altamente recomendado
Experiência recomendada	É altamente recomendável ter seis meses de experiência prática na plataforma Databricks, alinhada à estrutura do exame listada
Validade	2 anos

Recertificação: A recertificação é necessária a cada [dois] anos para manter seu estado de certificado. Para se recertificar, você deve fazer o exame atual ao vivo.

Conteúdo não pontuado: Os exames podem incluir itens não pontuados para coletar informações estatísticas para uso futuro. Esses itens não são identificados no exame e não afetam sua pontuação; além disso, o tempo adicional é considerado no Limite de tempo para contabilizá-los.

Preparação Recomendada

Treinamento

- **Conduzido por instrutor: Machine Learning with Databricks**
- **No seu próprio ritmo (disponível na Databricks Academy):**

- Preparação de Dados para Machine Learning
- Implantação de Modelo do Machine Learning
- Desenvolvimento de Modelo do Machine Learning
- Machine Learning Operations

Guia de Preparação para IA - Qualquer Exame Databricks

O Databricks AI Prep Guide mostra como transformar o chatbot de IA que você já usa em um tutor personalizado para qualquer certificação Databricks: configure-o com o guia oficial do exame, identifique as respostas desatualizadas que a IA tende a fornecer, descubra seus pontos fracos e combine tudo isso com a prática prática que todo exame Databricks exige.

Prática Hands-On

Não sabe o que praticar? O Guia de Preparação de IA (consulte a seção Guia de Preparação de IA acima) inclui um prompt em "o passo 5 — Mínimo prático" que faz um chatbot de IA gerar uma lista de verificação personalizada com 6 a 10 tarefas práticas mapeadas para os objetivos deste exame, cada uma pequena o suficiente para ser concluída no Databricks Free Edition.

Documentação da Databricks

Ao se preparar, estude a documentação oficial da Databricks para as áreas da plataforma abordadas neste exame. Ela é a fonte mais atual e autoritativa. Use-a para julgar com que profundidade estudar cada tópico e para garantir que você está aprendendo recursos atualizados e não obsoletos, em vez de material desatualizado.

Documentação oficial: documentação da Databricks

Estrutura do Exame

Seção 1: Databricks Machine Learning (38%)

- Identificar as melhores práticas de uma estratégia de MLOps
- Identificar as vantagens de usar ML runtimes
- Identificar como o AutoML facilita a seleção de modelo/recurso.
- Identificar as vantagens que o AutoML traz para o processo de desenvolvimento de modelos
- Identificar os benefícios de criar tabelas do Feature Store no nível de conta no Unity Catalog no Databricks em comparação com o nível de workspace
- Criar uma tabela do Feature Store no Unity Catalog
- Gravar dados em uma tabela do Feature Store

- Ensinar um modelo com recursos de uma tabela do feature store.
- Pontuar um modelo usando recursos de uma tabela do Feature Store.
- Descreva as diferenças entre tabelas de recursos online e offline
- Identifique a melhor execução usando a API MLflow Client.
- Registrar manualmente métricas, artefatos e modelos em uma execução do MLflow.
- Identificar informações disponíveis na UI do MLflow
- Fazer o registro de um modelo usando a MLflow Client API no registro do Unity Catalog
- Identificar os benefícios de registrar modelos no registro do Unity Catalog em vez do registro do workspace
- Identificar cenários em que a promoção de código é preferível à promoção de modelos e vice-versa
- Definir ou remover uma tag de um modelo
- Promover um modelo desafiante a modelo campeão usando aliases

Seção 2: Processamento de Dados (19%)

- Calcular estatísticas resumidas em um Spark DataFrame usando .summary() ou resumos de dados do dbutils
- Remover outliers de um Spark DataFrame com base em desvio padrão ou IQR
- Criar visualizações para recursos categóricos ou contínuos
- Compare dois recursos categóricos ou dois recursos contínuos usando o método apropriado
- Compare e contraste a imputação de valores ausentes com o valor de média, mediana ou moda
- Imputar valores ausentes com o valor da moda, média ou mediana
- Use codificação one-hot para recursos categóricos
- Identifique e explique os tipos de modelo ou conjuntos de dados para os quais a codificação one-hot é ou não é apropriada.
- Identificar cenários em que a transformação em escala log é apropriada

Seção 3: Desenvolvimento de Modelos (31%)

- Usar fundamentos de ML para selecionar o algoritmo apropriado para um determinado cenário de modelo
- Identificar métodos para mitigar o desequilíbrio de dados nos dados de treinamento
- Comparar estimadores e transformadores
- Desenvolver um pipeline de treinamento
- Usar a operação fmin do Hyperopt para ajustar os hiperparâmetros de um modelo
- Realizar busca aleatória, busca em grade ou busca Bayesian como método de ajuste de hiperparâmetros.
- Paralelizar modelos de nó único para ajuste de hiperparâmetros
- Descreva os benefícios e as desvantagens de usar validação cruzada em vez de uma divisão ensinar-validação.

- Realizar validação cruzada como parte do ajuste do modelo.
- Identifique o número de modelos sendo treinados em conjunto com um processo de grid-search e validação cruzada.
- Use métricas de classificação comuns: F1, Log Loss, ROC/AUC, etc
- Use métricas comuns de regressão: RMSE, MAE, R-squared, etc.
- Escolha a métrica mais adequada para um determinado objetivo de cenário
- Identificar a necessidade de exponenciar variáveis transformadas por log antes de calcular métricas de avaliação ou interpretar previsões
- Avaliar o impacto da complexidade do modelo e do trade-off entre viés e variância no desempenho do modelo

Seção 4: Implantação de Modelos (12%)

- Identificar as diferenças e vantagens das abordagens de model serving: lotes, tempo real e transmissão
- Implantar um modelo personalizado em um model endpoint
- Usar Pandas para realizar inferência em lotes
- Identificar como a inferência de transmissão é realizada com Delta Live Tables
- Implantar e executar query em um modelo para inferência em tempo real
- Dividir dados entre endpoints para inferência em tempo real

Perguntas de Exemplo

O objetivo das questões de exemplo é mostrar a você os objetivos conforme são apresentados no guia do exame e fornecer uma questão de exemplo alinhada a cada objetivo. O guia do exame lista os objetivos que podem ser abordados em um exame; a melhor forma de se preparar é revisar o esboço completo do exame para a sua data de exame acima.

Questão 1

Objetivo: Criar uma tabela do Feature Store no Unity Catalog.

Um cientista de dados deseja criar uma Feature Table para usar em seus modelos. Ele está trabalhando em um workspace com Unity Catalog habilitado e quer que essa Feature Table seja armazenada e governada por ele. Qual é a maneira correta de criar essa Feature Table?

A. Crie uma Delta table com dados, como de costume, e depois use o método `register_table` do `FeatureStoreClient` em Python para registrá-la como uma Feature table no Unity Catalog.
B. Crie uma Tabela Delta vazia no Unity Catalog com a cláusula AS FEATURE STORE via SQL e, em seguida, grave dados nela.
C. Use o método create_table do FeatureEngineeringClient em Python para criar a tabela e, em seguida, grave dados nela.

D. Crie uma Delta Table com dados no Unity Catalog e, em seguida, use o comando ALTER TABLE em SQL para configurá-la como uma Feature Table com a cláusula SET AS FEATURE STORE.

Questão 2

Objetivo: Imputar valores ausentes com o valor da moda, média ou mediana.

Um cientista de dados precisa imputar os valores ausentes em um recurso contínuo. Ele quer fazer isso com o menor esforço possível, mas com resultados corretos. Qual estratégia fará isso?

A. Use o sklearn SimpleImputer, que seleciona automaticamente a melhor metodologia com base na distribuição do recurso
B. Examine a distribuição dos valores e selecione a imputação adequada após a revisão
C. Use .mean(), que é a imputação mais adequada para colunas contínuas
D. Use .mode(), que é a imputação mais adequada para colunas contínuas

Questão 3

Objetivo: Identificar métodos para mitigar o desequilíbrio de dados nos dados de treinamento.

Um cientista de dados está desenvolvendo um modelo que prevê se um Cliente vai cancelar uma inscrição em um serviço. O dataset é altamente desbalanceado, com apenas 10% das instâncias representando Clientes que cancelam. Ele quer garantir que o modelo identifique efetivamente a classe minoritária sem ser tendencioso em relação à classe majoritária. Qual estratégia mitiga diretamente o viés do modelo em relação aos Clientes que não cancelam devido ao desbalanceamento de classes?

A. Normalize os recursos para garantir que estejam na mesma escala, melhorando o desempenho do modelo.
B. Use o aprendizado sensível ao custo atribuindo um custo de classificação incorreta mais alto à classe minoritária durante o treinamento do modelo.
C. Aumente o tamanho do dataset de treinamento coletando mais dados sobre Clientes que não cancelaram.
D. Use um modelo mais simples para reduzir o overfitting, garantindo que ele generalize melhor para a classe minoritária.

Questão 4

Objetivo: Identificar o número de modelos sendo treinados em conjunto com um processo de grid-search e validação cruzada.

Um cientista de dados está ajustando um modelo Support Vector Machine (SVM) usando validação cruzada de 5 folds e GridSearchCV no scikit-learn. A grade de parâmetros inclui três

hiperparâmetros a Otimizar: C com os valores [0.1, 1, 10], kernel com as opções ['linear', 'rbf'] e gamma com os valores [0.01, 0.1, 1]. Quantos modelos diferentes serão treinados no total?

A. 90
B. 18
C. 1
D. Nenhuma das opções acima.

Questão 5

Objetivo: Identificar como a inferência de transmissão é realizada com Delta Live Tables.

Uma empresa possui uma plataforma de podcasts com milhares de usuários. A empresa implementou um algoritmo de detecção de anomalia para identificar baixo engajamento com podcasts com base em uma janela contínua de 10 minutos de eventos de usuários, como ouvir, pausar e sair do podcast. Um engenheiro de machine learning deseja implantar esse modelo em um pipeline de dados de produção que precisa processar dezenas de milhares de eventos por segundo. Como o volume de eventos varia ao longo do dia, o engenheiro precisa que o compute do pipeline seja redimensionado dinamicamente. Qual abordagem de design de pipeline atende a esses requisitos?

A. Crie um pipeline Delta Live Tables que aplique o algoritmo como um Spark UDF.
B. Crie um Structured Streaming Job que aplique o algoritmo como um Spark UDF.
C. Criar um model serving endpoint, criar um Delta Live Tables pipeline que chama um UDF personalizado que invoca o endpoint.
D. Crie um model serving endpoint, crie um Structured Streaming job que chama um UDF personalizado que invoca o endpoint.

Questão 6

Objetivo: Promover um modelo desafiante a modelo campeão usando aliases.

Uma equipe registra versões sucessivas de um modelo no registro de modelo do Unity Catalog. Após a validação mostrar que uma versão recém-registrada supera a versão que está atualmente servindo o tráfego de produção, eles desejam marcar a nova versão como aquela que a produção lê, seguindo a abordagem recomendada. O que eles devem fazer?

A. Use MlflowClient().transition_model_version_stage() para mover a nova versão para o estágio "Production".
B. Use MlflowClient().set_registered_model_alias() para atribuir o alias "champion" à nova versão do modelo.
C. Exclua a versão atual do modelo campeão para que a nova versão seja a única disponível para servir.
D. Use MlflowClient().set_model_version_tag() para definir um tag "champion" = "true" na nova versão do modelo.

Questão 7

Objetivo: Identificar a melhor execução usando a API do MLflow Client.

Um cientista de dados registrou dezenas de execuções do MLflow em um único experimento, cada uma registrando uma métrica validation_rmse. Ele deseja recuperar programaticamente a execução única com o menor validation_rmse usando a MLflow Client API. Qual abordagem realiza isso?

A. Chame search_runs com o ID do experimento, order_by=["metrics.validation_rmse ASC"], e tome a primeira execução retornada.
B. Abra o experimento na interface do MLflow, ordene a tabela de execuções por validation_rmse e leia a linha superior.
C. Liste as execuções com o cliente e selecione a execução com o horário de início mais recente.
D. Chame get_run com o nome do experimento para retornar diretamente a melhor execução do experimento.

Questão 8

Objetivo: Identificar cenários em que a transformação de escala log é apropriada.

Um cientista de dados está preparando recursos para um modelo de regressão de preços de imóveis. Um recurso numérico e o alvo de preço são ambos fortemente assimétricos à direita, com um pequeno número de valores muito grandes que abrangem várias ordens de magnitude. Em qual situação a aplicação de uma transformação de escala log é mais apropriada?

A. Quando um recurso categórico possui muitos níveis distintos e sua cardinalidade precisa ser reduzida.
B. Quando o recurso já está aproximadamente distribuído normalmente e centralizado próximo de zero.
C. Quando um recurso ou alvo tem assimetria positiva e abrange várias ordens de magnitude, para comprimir valores grandes e tornar a distribuição mais simétrica.
D. Quando um recurso contém valores negativos que precisam ser deslocados para o intervalo positivo.

Questão 9

Objetivo: Avaliar o impacto da complexidade do modelo e do trade-off entre viés e variância no desempenho do modelo.

Durante a avaliação, uma cientista de dados observa que seu modelo alcança um erro muito baixo no conjunto de treinamento, mas um erro substancialmente maior no conjunto de validação. Com base no tradeoff entre viés e variância, o que está acontecendo e qual é uma resposta apropriada?

A. O modelo tem alto viés (underfitting); o cientista deve aumentar a complexidade do modelo.
B. O modelo apresenta alto viés causado por poucos recursos; o cientista deve criar recursos adicionais.
C. Tanto o erro de treinamento quanto o erro de validação são altos, indicando alta variância; o cientista deve ensinar o modelo por mais tempo.
D. O modelo tem alta variância (overfitting); o cientista deve reduzir a complexidade do modelo ou adicionar regularização.

Questão 10

Objetivo: Identificar as diferenças e vantagens das abordagens de model serving: lotes, tempo real e transmissão.

Um varejista pontua toda a sua base de clientes uma vez por noite para produzir segmentos de marketing para o dia seguinte. Não há requisito de previsões com baixa latência por requisição. Qual abordagem de model serving é a mais adequada e econômica para essa carga de trabalho?

A. Inferência em lotes que pontua todos os registros em uma programação noturna.
B. Inferência em tempo real por meio de um REST model serving endpoint consultado uma vez por cliente.
C. Inferência de transmissão usando um Pipeline de Structured Streaming em execução contínua.
D. Um endpoint de model serving com GPU sempre ativo, dimensionado para a menor latência possível.

Gabarito & Explicações

#	Resposta	Explicação
1	C	Em um workspace habilitado para Unity Catalog, use o método **`FeatureEngineeringClient.create_table`** para criar a feature table e, em seguida, grave os dados nela. Não existe a cláusula **`SQL AS FEATURE STORE`** nem **`ALTER TABLE ... SET AS FEATURE STORE`**, e **`register_table`** não é o caminho de criação atual.
2	B	Resultados corretos exigem examinar a distribuição da feature e escolher a imputação apropriada. O **`SimpleImputer`** não seleciona automaticamente uma estratégia com base na distribuição, e aplicar cegamente a média (ou a moda) pode ser inadequado para uma determinada feature contínua.
3	B	O aprendizado sensível a custo atribui um custo de classificação incorreta mais alto à classe minoritária (churn), contrapondo-se diretamente ao viés do modelo em relação à classe majoritária. Normalizar features, coletar mais dados da classe majoritária ou usar um modelo mais simples não resolvem o desbalanceamento de classes.
4	A	A grade tem **`3 x 2 x 3 = 18`** combinações de hiperparâmetros, e a validação cruzada com 5 folds treina um modelo por fold para cada combinação; portanto, são treinados **`18 x 5 = 90`** modelos no total.
5	A	Um pipeline Delta Live Tables que aplica o algoritmo como uma Spark UDF lida com alto volume de eventos e redimensiona dinamicamente o compute do pipeline. Rotear pelo endpoint de model serving adiciona overhead, e um job de Structured Streaming não oferece o mesmo autoscaling dinâmico gerenciado.
6	B	No model registry do Unity Catalog, o fluxo de promoção usa aliases em vez das transições de estágio legadas. Atribuir o alias "**`champion`**" à nova versão com **`set_registered_model_alias`** faz com que ela seja a versão que a produção lê. As transições de estágio pertencem ao registry legado do workspace (depreciado), excluir a versão anterior é destrutivo e desnecessário, e uma tag é metadado descritivo que não altera a versão servida.
7	A	O **`search_runs`** do MLflow Client (como **`mlflow.search_runs`**) oferece suporte a ordenação no servidor; ordenar por **`metrics.validation_rmse`** em ordem crescente e pegar o primeiro resultado retorna a melhor run programaticamente. A UI do MLflow não é a API do Client, a run iniciada mais recentemente não é necessariamente a melhor, e **`get_run`** exige um **`run_id`** específico e não classifica runs.
8	C	Uma transformação logarítmica comprime valores grandes e reduz a cauda longa à direita, tornando mais simétrica uma feature ou alvo assimétrico que abrange várias ordens de grandeza. Ela não é uma técnica de redução de cardinalidade, não traz benefício para uma feature já normal e é indefinida para valores zero ou negativos, por isso não é usada para deslocar valores negativos para a faixa positiva.
9	D	Erro de treinamento baixo combinado com erro de validação bem mais alto é a assinatura de alta variância, ou overfitting. A resposta adequada é reduzir a complexidade do modelo

		ou adicionar regularização (ou coletar mais dados de treinamento). Alto viés (underfitting) mostraria, em vez disso, erro alto tanto no conjunto de treinamento quanto no de validação.
10	A	Quando as previsões são geradas em uma agenda fixa sobre o conjunto completo de dados, sem requisito de baixa latência, a inferência em batch é a abordagem mais simples e econômica. Endpoints em tempo real e GPUs sempre ativas acrescentam infraestrutura e custo para uma latência que a carga de trabalho não exige, e a inferência em streaming é voltada a dados que chegam continuamente, e não a um job noturno de pontuação em lote.