Databricks

Exam Questions Databricks-Machine-Learning-Associate

Databricks Certified Machine Learning Associate Exam

NEW QUESTION 1
Which of the following hyperparameter optimization methods automatically makes informed selections of hyperparameter values based on previous trials for each iterative model evaluation?

A. Random Search
B. Halving Random Search
C. Tree of Parzen Estimators
D. Grid Search

Answer: C

Explanation:
Tree of Parzen Estimators (TPE) is a sequential model-based optimization algorithm that selects hyperparameter values based on the outcomes of previous trials. It models the probability density of good and bad hyperparameter values and makes informed decisions about which hyperparameters to try next.
This approach contrasts with methods like random search and grid search, which do not use information from previous trials to guide the search process.
References:
? Hyperopt and TPE

NEW QUESTION 2
A data scientist wants to use Spark ML to one-hot encode the categorical features in their PySpark DataFramefeatures_df. A list of the names of the string columns is assigned to theinput_columnsvariable.
They have developed this code block to accomplish this task:

```
ohe = OneHotEncoder(
    inputCols=input_columns,
    outputCols=output_columns
)
ohe_model = ohe.fit(features_df)
ohe_features_df = ohe_model.transform(features_df)
```

The code block is returning an error.
Which of the following adjustments does the data scientist need to make to accomplish this task?

A. They need to specify the method parameter to the OneHotEncoder.
B. They need to remove the line with the fit operation.
C. They need to use StringIndexer prior to one-hot encodinq the features.
D. They need to useVectorAssemblerprior to one-hot encoding the features.

Answer: C

Explanation:
TheOneHotEncoderin Spark ML requires numerical indices as inputs rather than string labels. Therefore, you need to first convert the string columns to numerical indices usingStringIndexer. After that, you can applyOneHotEncoderto these indices. Corrected code:
frompyspark.ml.featureimportStringIndexer, OneHotEncoder# Convert string column to indexindexers = [StringIndexer(inputCol=col, outputCol=col+"_index")forcolininput_columns] indexer_model = Pipeline(stages=indexers).fit(features_df) indexed_features_df = indexer_model.transform(features_df)# One-hot encode the indexed columnsohe = OneHotEncoder(inputCols=[col+"_index"forcolininput_columns], outputCols=output_columns) ohe_model = ohe.fit(indexed_features_df) ohe_features_df = ohe_model.transform(indexed_features_df)
References:
? PySpark ML Documentation

NEW QUESTION 3
The implementation of linear regression in Spark ML first attempts to solve the linear regression problem using matrix decomposition, but this method does not scale well to large datasets with a large number of variables.
Which of the following approaches does Spark ML use to distribute the training of a linear regression model for large data?

A. Logistic regression
B. Spark ML cannot distribute linear regression training
C. Iterative optimization
D. Least-squares method
E. Singular value decomposition

Answer: C

Explanation:
For large datasets with many variables, Spark ML distributes the training of a linear regression model using iterative optimization methods. Specifically, Spark ML employs algorithms such as Gradient Descent or L-BFGS (Limited-memory Broyden–Fletcher–Goldfarb–Shanno) to iteratively minimize the loss function. These iterative methods are suitable for distributed computing environments and can handle
large-scale data efficiently by partitioning the data across nodes in a cluster and performing parallel updates.References:
? Spark MLlib Documentation (Linear Regression with Iterative Optimization).

NEW QUESTION 4
A machine learning engineer would like to develop a linear regression model with Spark ML to predict the price of a hotel room. They are using the Spark DataFrametrain_dfto train the model.
The Spark DataFrametrain_dfhas the following schema:

```
hotel_room_id STRING,
price DOUBLE,
features UDT
```

The machine learning engineer shares the following code block:

```
lr = LinearRegression(featuresCol="features", labelCol="price")
lr_model = lr.fit(train_df)
```

Which of the following changes does the machine learning engineer need to make to complete the task?

A. They need to call the transform method on train df
B. They need to convert the features column to be a vector
C. They do not need to make any changes
D. They need to utilize a Pipeline to fit the model
E. They need to split thefeaturescolumn out into one column for each feature

Answer: B

Explanation:
In Spark ML, the linear regression model expects the feature column to be a vector type. However, if thefeaturescolumn in the DataFrametrain_dfis not already in this format (such as being a column of type UDT or a non-vectorized type), the engineerneeds to convert it to a vector column using a transformer likeVectorAssembler. This is a critical step in preparing the data for modeling as Spark ML models require input features to be combined into a single vector column.
References
? Spark MLlib documentation forLinearRegression:https://spark.apache.org/docs/latest/ml-classification- regression.html#linear-regression

NEW QUESTION 5
A data scientist is using Spark ML to engineer features for an exploratory machine learning project.
They decide they want to standardize their features using the following code block:

```
scaler = StandardScaler(
    withMean=True,
    inputCol="input_features",
    outputCol="output_features"
)
scaler_model = scaler.fit(features_df)
scaled_df = scaler_model.transform(features_df)
train_df, test_df = scaled_df.randomSplit([.8, .2], seed=42)
```

Upon code review, a colleague expressed concern with the features being standardized prior to splitting the data into a training set and a test set.
Which of the following changes can the data scientist make to address the concern?

A. Utilize the MinMaxScaler object to standardize the training data according to global minimum and maximum values
B. Utilize the MinMaxScaler object to standardize the test data according to global minimum and maximum values
C. Utilize a cross-validation process rather than a train-test split process to remove theneed for standardizing data
D. Utilize the Pipeline API to standardize the training data according to the test data's summary statistics
E. Utilize the Pipeline API to standardize the test data according to the training data's summary statistics

Answer: E

Explanation:
To address the concern about standardizing features prior to splitting the data, the correct approach is to use the Pipeline API to ensure that only the training data's summary statistics are used to standardize the test data. This is achieved by fitting the StandardScaler (or any scaler) on the training data and then transforming both the training and test data using the fitted scaler. This approach prevents information leakage from the test data into the model training process and ensures that the model is evaluated fairly. References:
? Best Practices in Preprocessing in Spark ML (Handling Data Splits and Feature Standardization).

NEW QUESTION 6
A data scientist has replaced missing values in their feature set with each respective feature variable??s median value. A colleague suggests that the data scientist is throwing away valuable information by doing this.
Which of the following approaches can they take to include as much information as possible in the feature set?

A. Impute the missing values using each respective feature variable's mean value instead of the median value
B. Refrain from imputing the missing values in favor of letting the machine learning algorithm determine how to handle them
C. Remove all feature variables that originally contained missing values from the feature set
D. Create a binary feature variable for each feature that contained missing values indicating whether each row's value has been imputed
E. Create a constant feature variable for each feature that contained missing values indicating the percentage of rows from the feature that was originally missing

Answer: D

Explanation:
By creating a binary feature variable for each feature with missing values to indicate whether a value has been imputed, the data scientist can preserve information about the original state of the data. This approach maintains the integrity of the dataset by marking which values are original and which are synthetic (imputed). Here are the steps to
implement this approach:
? Identify Missing Values:Determine which features contain missing values.
? Impute Missing Values:Continue with median imputation or choose another method (mean, mode, regression, etc.) to fill missing values.
? Create Indicator Variables:For each feature that had missing values, add a new binary feature. This feature should be '1' if the original value was missing and imputed, and '0' otherwise.
? Data Integration:Integrate these new binary features into the existing dataset. This maintains a record of where data imputation occurred, allowing models to potentially weight these observations differently.
? Model Adjustment:Adjust machine learning models to account for these new features, which might involve considering interactions between these binary indicators and other features.
References
? "Feature Engineering for Machine Learning" by Alice Zheng and Amanda Casari (O'Reilly Media, 2018), especially the sections on handling missing data.
? Scikit-learn documentation on imputing missing values: https://scikit-learn.org/stable/modules/impute.html

NEW QUESTION 7
A data scientist uses 3-fold cross-validation when optimizing model hyperparameters for a regression problem. The following root-mean-squared-error values are calculated on each of the validation folds:
• 10.0
• 12.0
• 17.0
Which of the following values represents the overall cross-validation root-mean-squared error?

A. 13.0
B. 17.0
C. 12.0
D. 39.0
E. 10.0

Answer: A

Explanation:
To calculate the overall cross-validation root-mean-squared error (RMSE), you average the RMSE values obtained from each validation fold. Given the RMSE values of 10.0, 12.0, and 17.0 for the three folds, the overall cross-validation RMSE is calculated as the average of these three values:
Overall CV RMSE=10.0+12.0+17.03=39.03=13.0Overall CV RMSE=310.0+12.0+17.0 =339.0=13.0
Thus, the correct answer is 13.0, which accurately represents the average RMSE across all folds. References:
? Cross-validation in Regression (Understanding Cross-Validation Metrics).

NEW QUESTION 8
A data scientist has created a linear regression model that useslog(price)as a label variable. Using this model, they have performed inference and the predictions and actual label values are in Spark DataFramepreds_df.
They are using the following code block to evaluate the model: regression_evaluator.setMetricName("rmse").evaluate(preds_df)
Which of the following changes should the data scientist make to evaluate the RMSE in a way that is comparable withprice?

A. They should exponentiate the computed RMSE value
B. They should take the log of the predictions before computing the RMSE
C. They should evaluate the MSE of the log predictions to compute the RMSE
D. They should exponentiate the predictions before computing the RMSE

Answer: D

Explanation:
When evaluating the RMSE for a model that predicts log-transformed prices, the predictions need to be transformed back to the original scale to obtain an RMSE that is comparable with the actual price values. This is done by exponentiating the predictions before computing the RMSE. The RMSE should be computed on the same scale as the original data to provide a meaningful measure of error.
References:
? Databricks documentation on regression evaluation: Regression Evaluation

NEW QUESTION 9
A machine learning engineer wants to parallelize the inference of group-specific models using the Pandas Function API. They have developed theapply_modelfunction that will look up and load the correct model for each group, and they want to apply it to each group of DataFramedf.
They have written the following incomplete code block:

```
prediction_df = (df
    .groupby("device_id")
    .______(apply_model, schema=apply_return_schema)
)
```

Which piece of code can be used to fill in the above blank to complete the task?

A. applyInPandas
B. groupedApplyInPandas
C. mapInPandas
D. predict

Answer: A

Explanation:
To parallelize the inference of group-specific models using the Pandas Function API in PySpark, you can use theapplyInPandasfunction. This function allows you to apply a Python function on each group of a DataFrame and return a DataFrame, leveraging the power of pandas UDFs (user-defined functions) for better performance.
prediction_df = (df.groupby("device_id") .applyInPandas(apply_model, schema=apply_return_schema))
In this code:
? groupby("device_id"): Groups the DataFrame by the "device_id" column.
? applyInPandas(apply_model, schema=apply_return_schema): Applies theapply_modelfunction to each group and specifies the schema of the return DataFrame.
References:
? PySpark Pandas UDFs Documentation

NEW QUESTION 10
A machine learning engineer is using the following code block to scale the inference of a single-node model on a Spark DataFrame with one million records:

```
@pandas_udf("double")
def predict(iterator: Iterator[pd.DataFrame]) -> Iterator[pd.Series]:
    model_path = f"runs:/{run.info.run_id}/model"
    model = mlflow.sklearn.load_model(model_path)
    for features in iterator:
        pdf = pd.concat(features, axis=1)
        yield pd.Series(model.predict(pdf))
```

Assuming the default Spark configuration is in place, which of the following is a benefit of using anIterator?

A. The data will be limited to a single executor preventing the model from being loaded multiple times
B. The model will be limited to a single executor preventing the data from being distributed
C. The model only needs to be loaded once per executor rather than once per batch during the inference process
D. The data will be distributed across multiple executors during the inference process

Answer: C

Explanation:
Using an iterator in thepandas_udfensures that the model only needs to be loaded once per executor rather than once per batch. This approach reduces the overhead associated with repeatedly loading the model during the inference process, leading to more efficient and faster predictions. The data will be distributed across multiple executors, but each executor will load the model only once, optimizing the inference process. References:
? Databricks documentation on pandas UDFs: Pandas UDFs

NEW QUESTION 10
Which of the following is a benefit of using vectorized pandas UDFs instead of standard PySpark UDFs?

A. The vectorized pandas UDFs allow for the use of type hints
B. The vectorized pandas UDFs process data in batches rather than one row at a time
C. The vectorized pandas UDFs allow for pandas API use inside of the function
D. The vectorized pandas UDFs work on distributed DataFrames
E. The vectorized pandas UDFs process data in memory rather than spilling to disk

Answer: B

Explanation:
Vectorized pandas UDFs, also known as Pandas UDFs, are a powerful feature in PySpark that allows for more efficient operations than standard UDFs. They operate by processing data in batches, utilizing vectorized operations that leverage pandas to perform operations on whole batches of data at once. This approach is much moreefficient than processing data row by row as is typical with standard PySpark UDFs, which can significantly speed up the computation.
References
? PySpark Documentation on UDFs:https://spark.apache.org/docs/latest/api/python/user_guide/sql/arrow_pandas.html#pandas-udfs-a-k-a-vectorized-udfs

NEW QUESTION 13
A data scientist uses 3-fold cross-validation and the following hyperparameter grid when optimizing model hyperparameters via grid search for a classification problem:
Hyperparameter 1: [2, 5, 10]
Hyperparameter 2: [50, 100]
Which of the following represents the number of machine learning models that can be trained in parallel during this process?

A. 3
B. 5
C. 6
D. 18

Answer: D

Explanation:
To determine the number of machine learning models that can be trained in parallel, we need to calculate the total number of combinations of hyperparameters. The given hyperparameter grid includes:
? Hyperparameter 1: [2, 5, 10] (3 values)
? Hyperparameter 2: [50, 100] (2 values)
The total number of combinations is the product of the number of values for each hyperparameter:3 (values of Hyperparameter 1)??2 (values of Hyperparameter 2)=63 (v s of Hyperparameter 1)??2 (values of Hyperparameter 2)=6
With 3-fold cross-validation, each combination of hyperparameters will be evaluated 3 times. Thus, the total number of models trained will be:6 (combinations)??3 (folds)=186 (combinations)??3 (folds)=18
However, the number of models that can be trained in parallel is equal to the number of hyperparameter combinations, not the total number of models considering cross-validation. Therefore, 6 models can be trained in parallel.
References:
? Databricks documentation on hyperparameter tuning: Hyperparameter Tuning

NEW QUESTION 16
A machine learning engineer has created a Feature Table new_table using Feature Store Client fs. When creating the table, they specified a metadata description with key information about the Feature Table. They now want to retrieve that metadata programmatically.
Which of the following lines of code will return the metadata description?

A. There is no way to return the metadata description programmatically.
B. fs.create_training_set("new_table")
C. fs.get_table("new_table").description
D. fs.get_table("new_table").load_df()
E. fs.get_table("new_table")

Answer: C

Explanation:
To retrieve the metadata description of a feature table created using the Feature Store Client (referred here asfs), the correct method involves callingget_tableon thefsclient with the table name as an argument, followed by accessing thedescription attribute of the returned object. The code snippetfs.get_table("new_table").description correctly achieves this by fetching the table object for "new_table" and then accessing its description attribute, where the metadata is stored. The other options do not correctly focus on retrieving the metadata description.References:
? Databricks Feature Store documentation (Accessing Feature Table Metadata).

NEW QUESTION 21
A data scientist wants to use Spark ML to impute missing values in their PySpark DataFrame features_df. They want to replace missing values in all numeric columns in features_df with each respective numeric column??s median value.
They have developed the following code block to accomplish this task:

```
imputer = Imputer(
    strategy="median",
    inputCols=input_columns,
    outputCols=output_columns
)
imputed_features_df = imputer.transform(features_df)
```

The code block is not accomplishing the task.
Which reasons describes why the code block is not accomplishing the imputation task?

A. It does not impute both the training and test data sets.
B. The inputCols and outputCols need to be exactly the same.
C. The fit method needs to be called instead of transform.
D. It does not fit the imputer on the data to create an ImputerModel.

Answer: D

Explanation:
In the provided code block, theImputerobject is created but not fitted on the data to generate anImputerModel. Thetransformmethod is being called directly on the

Imputerobject, which does not yet contain the fitted median values needed for imputation. The correct approach is to fit the imputer on the dataset first.
Corrected code:
imputer = Imputer(strategy="median", inputCols=input_columns, outputCols=output_columns) imputer_model = imputer.fit(features_df)# Fit the imputer to the dataimputed_features_df = imputer_model.transform(features_df)# Transform the data using the fitted imputer
References:
? PySpark ML Documentation

NEW QUESTION 23
A data scientist has developed a linear regression model using Spark ML and computed the predictions in a Spark DataFrame preds_df with the following schema:
prediction DOUBLE actual DOUBLE
Which of the following code blocks can be used to compute the root mean-squared-error of the model according to the data in preds_df and assign it to the rmse variable?
A)
```
rmse = BinaryClassificationEvaluator(
    predictionCol="prediction",
    labelCol="actual",
    metricName="rmse"
)
```
B)
```
rmse = RegressionEvaluator(
    predictionCol="prediction",
    labelCol="actual",
    metricName="rmse"
)
```
C)
```
rmse = RegressionEvaluator(
    predictionCol="prediction",
    labelCol="actual",
    metricName="rmse"
)
```
D)
```
classification_evaluator = BinaryClassificationEvaluator(
    predictionCol="prediction",
    labelCol="actual",
    metricName="rmse"
)
```
E)
```
rmse = classification_evaluator.evaluate(preds_df)
regression_evaluator = RegressionEvaluator(
    predictionCol="prediction",
    labelCol="actual",
    metricName="rmse"
)
```

A. Option A
B. Option B
C. Option C
D. Option D
E. Option E

Answer: C

Explanation:
The code block to compute the root mean-squared error (RMSE) for a linear regression model in Spark ML should use theRegressionEvaluatorclass withmetricNameset to "rmse". Given the schema ofpreds_dfwith columnspredictionandactual, the correct evaluator setup will specifypredictionCol="prediction"andlabelCol="actual". Thus, the appropriate code block (Option C in your list) that usesRegressionEvaluatorto compute the RMSE is the correct choice. This setup correctly measures the performance of the regression model using the predictions and actual outcomes from the DataFrame.
References:

? Spark ML documentation (Using RegressionEvaluator to Compute RMSE).

NEW QUESTION 25
A data scientist has produced two models for a single machine learning problem. One of the models performs well when one of the features has a value of less than 5, and the other model performs well when the value of that feature is greater than or equal to 5. The data scientist decides to combine the two models into a single machine learning solution.
Which of the following terms is used to describe this combination of models?

A. Bootstrap aggregation
B. Support vector machines
C. Bucketing
D. Ensemble learning
E. Stacking

Answer: D

Explanation:
Ensemble learning is a machine learning technique that involves combining several models to solve a particular problem. The scenario described fits the concept of ensemble learning, where two models, each performing well under different conditions, are combined to create a more robust model. This approach often leads to better performance as it combines the strengths of multiple models.
References
? Introduction to Ensemble Learning: https://machinelearningmastery.com/ensemble-machine-learning-algorithms- python-scikit-learn/

NEW QUESTION 26
A data scientist wants to efficiently tune the hyperparameters of a scikit-learn model in parallel. They elect to use the Hyperopt library to facilitate this process.
Which of the following Hyperopt tools provides the ability to optimize hyperparameters in parallel?

A. fmin
B. SparkTrials
C. quniform
D. search_space
E. objective_function

Answer: B

Explanation:
TheSparkTrialsclass in the Hyperopt library allows for parallel hyperparameter optimization on a Spark cluster. This enables efficient tuning of hyperparameters by distributing the optimization process across multiple nodes in a cluster. fromhyperoptimportfmin, tpe, hp, SparkTrials search_space = {'x': hp.uniform('x',0,1),'y': hp.uniform('y',0,1) }defobjective(params):returnparams['x'] **2+ params['y'] **2spark_trials = SparkTrials(parallelism=4) best = fmin(fn=objective, space=search_space, algo=tpe.suggest, max_evals=100, trials=spark_trials)
References:
? Hyperopt Documentation

NEW QUESTION 27
A machine learning engineer has grown tired of needing to install the MLflow Python library on each of their clusters. They ask a senior machine learning engineer how their notebooks can load the MLflow library without installing it each time. The senior machine learning engineer suggests that they use Databricks Runtime for Machine Learning.
Which of the following approaches describes how the machine learning engineer can begin using Databricks Runtime for Machine Learning?

A. They can add a line enabling Databricks Runtime ML in their init script when creating their clusters.
B. They can check the Databricks Runtime ML box when creating their clusters.
C. They can select a Databricks Runtime ML version from the Databricks Runtime Version dropdown when creating their clusters.
D. They can set the runtime-version variable in their Spark session to ??ml??.

Answer: C

Explanation:
The Databricks Runtime for Machine Learning includes pre-installed packages and libraries essential for machine learning and deep learning, including MLflow. To use it, the machine learning engineer can simply select an appropriate Databricks Runtime ML version from the "Databricks Runtime Version" dropdown menu while creating their cluster. This selection ensures that all necessary machine learning libraries, including MLflow, are pre-installed and ready for use, avoiding the need to manually install them each time.
References
? Databricks documentation on creating clusters: https://docs.databricks.com/clusters/create.html

NEW QUESTION 28
A data scientist wants to tune a set of hyperparameters for a machine learning model. They have wrapped a Spark ML model in the objective functionobjective_functionand they have defined the search spacesearch_space.
As a result, they have the following code block:

```
num_evals = 100
trials = SparkTrials()
best_hyperparam = fmin(
    fn=objective_function,
    space=search_space,
    algo=tpe.suggest,
    max_evals=num_evals,
    trials=trials
)
```

Which of the following changes do they need to make to the above code block in order to accomplish the task?

A. Change SparkTrials() to Trials()
B. Reduce num_evals to be less than 10
C. Change fmin() to fmax()
D. Remove the trials=trials argument
E. Remove the algo=tpe.suggest argument

Answer: A

Explanation:
TheSparkTrials()is used to distribute trials of hyperparameter tuning across a Spark cluster. If the environment does not support Spark or if the user prefers not to usedistributed computing for this purpose, switching toTrials()would be appropriate.Trials() is the standard class for managing search trials in Hyperopt but does not distribute the computation. If the user is encountering issues withSparkTrials()possibly due to an unsupported configuration or an error in the cluster setup, usingTrials()can be a suitable change for running the optimization locally or in a non-distributed manner.
References
? Hyperopt documentation: http://hyperopt.github.io/hyperopt/

NEW QUESTION 30
A data scientist is wanting to explore the Spark DataFrame spark_df. The data scientist wants visual histograms displaying the distribution of numeric features to be included in the exploration.
Which of the following lines of code can the data scientist run to accomplish the task?

A. spark_df.describe()
B. dbutils.data(spark_df).summarize()
C. This task cannot be accomplished in a single line of code.
D. spark_df.summary()
E. dbutils.data.summarize (spark_df)

Answer: E

Explanation:
To display visual histograms and summaries of the numeric features in a Spark DataFrame, the Databricks utility functiondbutils.data.summarizecan be used. This function provides a comprehensive summary, including visual histograms.
Correct code: dbutils.data.summarize(spark_df)
Other options likespark_df.describe()andspark_df.summary()provide textual statistical summaries but do not include visual histograms.
References:
? Databricks Utilities Documentation

NEW QUESTION 34
A data scientist is using MLflow to track their machine learning experiment. As a part of each of their MLflow runs, they are performing hyperparameter tuning. The data scientist would like to have one parent run for the tuning process with a child run for each unique combination of hyperparameter values. All parent and child runs are being manually started with mlflow.start_run.
Which of the following approaches can the data scientist use to accomplish this MLflow run organization?

A. Theycan turn on Databricks Autologging
B. Theycan specify nested=True when startingthe child run for each unique combination of hyperparameter values
C. Theycan start each child run inside the parentrun's indented code block usingmlflow.start runO
D. They can start each child run with the same experiment ID as the parent run
E. They can specify nested=True when starting the parent run for the tuningprocess

Answer: B

Explanation:
To organize MLflow runs with one parent run for the tuning process and a child run for each unique combination of hyperparameter values, the data scientist can specifynested=Truewhen starting the child run. This approach ensures that each child run is properly nested under the parent run, maintaining a clear hierarchical structure for the experiment. This nesting helps in tracking and comparing different hyperparameter combinations within the same tuning process.References:
? MLflow Documentation (Managing Nested Runs).

NEW QUESTION 39
A data scientist has produced three new models for a single machine learning problem. In the past, the solution used just one model. All four models have nearly the same prediction latency, but a machine learning engineer suggests that the new solution will be less time efficient during inference.
In which situation will the machine learning engineer be correct?

A. When the new solution requires if-else logic determining which model to use to compute each prediction
B. When the new solution's models have an average latency that is larger than the size of the original model
C. When the new solution requires the use of fewer feature variables than the original model
D. When the new solution requires that each model computes a prediction for every record
E. When the new solution's models have an average size that is larger than the size of the original model

Answer: D

Explanation:
If the new solution requires that each of the three models computes a prediction for every record, the time efficiency during inference will be reduced. This is because the inference process now involves running multiple models instead of a single model, thereby increasing the overall computation time for each record. In scenarios where inference must be done by multiple models for each record, the latency accumulates, making the process less time efficient compared to using a single model. References:
? Model Ensemble Techniques

NEW QUESTION 42
A data scientist is utilizing MLflow Autologging to automatically track their machine learning experiments. After completing a series of runs for the experiment experiment_id, the data scientist wants to identify the run_id of the run with the best root-mean-square error (RMSE).
Which of the following lines of code can be used to identify the run_id of the run with the best RMSE in experiment_id?
A)

```
mlflow.search_runs(
      experiment_id,
      order_by = ["metrics.rmse DESC"]
)["run_id"][0]
```

B)

```
mlflow.best_run(
      experiment_id,
      order_by = ["metrics.rmse"]
)
```

C)

```
mlflow.search_runs(
      experiment_id,
      order_by = ["metrics.rmse"]
)["run_id"][0]
```

D)

```
mlflow.best_run(
      experiment_id,
      order_by = ["metrics.rmse DESC"]
)
```

A. OptionA
B. Option B
C. Option C
D. Option D

Answer: C

Explanation:
To find the run_id of the run with the best root-mean-square error (RMSE) in an MLflow experiment, the correct line of code to use is:
mlflow.search_runs(experiment_id, order_by=["metrics.rmse"])["run_id"][0]
This line of code searches the runs in the specified experiment, orders them by the RMSE metric in ascending order (the lower the RMSE, the better), and retrieves the run_id of the best-performing run. Option C correctly represents this logic.
References
? MLflow documentation on tracking experiments: https://www.mlflow.org/docs/latest/python_api/mlflow.html#mlflow.search_runs

NEW QUESTION 47
A machine learning engineer is trying to perform batch model inference. They want to get predictions using the linear regression model saved at the pathmodel_urifor the DataFramebatch_df.
batch_dfhas the following schema: customer_id STRING
The machine learning engineer runs the following code block to perform inference onbatch_dfusing the linear regression model atmodel_uri:

```
predictions = fs.score_batch(
    model_uri,
    batch_df
)
```

In which situation will the machine learning engineer??s code block perform the desired inference?

A. When the Feature Store feature set was logged with the model at model_uri
B. When all of the features used by the model at model_uri are in a Spark DataFrame in the PySpark
C. When the model at model_uri only uses customer_id as a feature
D. This code block will not perform the desired inference in any situation.
E. When all of the features used by the model at model_uri are in a single Feature Store table

Answer: A

Explanation:
The code block provided by the machine learning engineer will perform the desired inference when the Feature Store feature set was logged with the model at model_uri. This ensures that all necessary feature transformations and metadata are available for the model to make predictions. The Feature Store in Databricks allows for seamless integration of features and models, ensuring that the required features are correctly used during inference.
References:
? Databricks documentation on Feature Store: Feature Store in Databricks

NEW QUESTION 50
An organization is developing a feature repository and is electing to one-hot encode all categorical feature variables. A data scientist suggests that the categorical feature variables should not be one-hot encoded within the feature repository.
Which of the following explanations justifies this suggestion?

A. One-hot encoding is not supported by most machine learning libraries.
B. One-hot encoding is dependent on the target variable's values which differ for each application.
C. One-hot encoding is computationally intensive and should only be performed on small samples of training sets for individual machine learning problems.
D. One-hot encoding is not a common strategy for representing categorical feature variables numerically.
E. One-hot encoding is a potentially problematic categorical variable strategy for some machine learning algorithms.

Answer: E

Explanation:
One-hot encoding transforms categorical variables into a format that can be provided to machine learning algorithms to better predict the output. However, when done prematurely or universally within a feature repository, it can be problematic:
? Dimensionality Increase:One-hot encoding significantly increases the feature
space, especially with high cardinality features, which can lead to high memory consumption and slower computation.
? Model Specificity:Some models handle categorical variables natively (like decision
trees and boosting algorithms), and premature one-hot encoding can lead to inefficiency and loss of information (e.g., ordinal relationships).
? Sparse Matrix Issue:It often results in a sparse matrix where most values are zero,
which can be inefficient in both storage and computation for some algorithms.
? Generalization vs. Specificity:Encoding should ideally be tailored to specific models and use cases rather than applied generally in a feature repository.
References
? "Feature Engineering and Selection: A Practical Approach for Predictive Models" by Max Kuhn and Kjell Johnson (CRC Press, 2019).

NEW QUESTION 55
A data scientist is attempting to tune a logistic regression model logistic using scikit-learn. They want to specify a search space for two hyperparameters and let the tuning process randomly select values for each evaluation.
They attempt to run the following code block, but it does not accomplish the desired task:

```
distributions = dict(C=uniform(loc=0, scale=4), penalty=['l2', 'l1'])
clf = GridSearchCV(logistic, distributions, random_state=0)
search = clf.fit(feature_data, target_data)
```

Which of the following changes can the data scientist make to accomplish the task?

A. Replace the GridSearchCV operation with RandomizedSearchCV
B. Replace the GridSearchCV operation with cross_validate
C. Replace the GridSearchCV operation with ParameterGrid
D. Replace the random_state=0 argument with random_state=1
E. Replace the penalty= ['12', '11'] argument with penalty=uniform ('12', '11')

Answer: A

Explanation:
The user wants to specify a search space for hyperparameters and let the tuning process randomly select values.GridSearchCVsystematically tries every combination of the provided hyperparameter values, which can be computationally expensive and time-consuming.RandomizedSearchCV, on the other hand, samples hyperparameters from a distribution for a fixed number of iterations. This approach is usually faster and still can find very good parameters, especially when the search space is large or includes distributions.
References
? Scikit-Learn documentation on hyperparameter tuning: https://scikit- learn.org/stable/modules/grid_search.html#randomized-parameter-optimization

NEW QUESTION 58
A machine learning engineer is trying to scale a machine learning pipeline by distributing its feature engineering process.
Which of the following feature engineering tasks will be the least efficient to distribute?

A. One-hot encoding categorical features
B. Target encoding categorical features
C. Imputing missing feature values with the mean
D. Imputing missing feature values with the true median
E. Creating binary indicator features for missing values

Answer: D

Explanation:
Among the options listed, calculating the true median for imputing missing feature values is the least efficient to distribute. This is because the true median requires knowledge of the entire data distribution, which can be computationally expensive in a distributed environment. Unlike mean or mode, finding the median requires sorting the data or maintaining a full distribution, which is more intensive and often requires shuffling the data across partitions.
References
? Challenges in parallel processing and distributed computing for data aggregation like median calculation:https://www.apache.org

NEW QUESTION 60
Which of the following tools can be used to distribute large-scale feature engineering without the use of a UDF or pandas Function API for machine learning pipelines?

A. Keras
B. pandas
C. PvTorch
D. Spark ML
E. Scikit-learn

Answer: D

Explanation:
Spark ML (Machine Learning Library) is designed specifically for handling large-scale data processing and machine learning tasks directly within Apache Spark. It provides tools and APIs for large-scale feature engineering without the need to rely on user-defined functions (UDFs) or pandas Function API, allowing for more scalable and efficient data transformations directly distributed across a Spark cluster. Unlike Keras, pandas, PyTorch, and scikit-learn, Spark ML operates natively in a distributed environment suitable for big data scenarios.References:
? Spark MLlib documentation (Feature Engineering with Spark ML).

NEW QUESTION 62
A data scientist is developing a machine learning pipeline using AutoML on Databricks Machine Learning.
Which of the following steps will the data scientist need to perform outside of their AutoML experiment?

A. Model tuning
B. Model evaluation
C. Model deployment
D. Exploratory data analysis

Answer: D

Explanation:
AutoML platforms, such as the one available in Databricks Machine Learning, streamline various stages of the machine learning pipeline including feature engineering, model selection, hyperparameter tuning, and model evaluation. However, exploratory data analysis (EDA) is typically performed outside the AutoML process. EDA involves understanding the dataset, visualizing distributions, identifying anomalies, and gaining insights into data before feeding it into a machine learning pipeline. This step is crucial for ensuring that the data is clean and suitable for model training but is generally done manually by the data scientist.
References

? Databricks documentation on AutoML: https://docs.databricks.com/applications/machine-learning/automl.html

NEW QUESTION 66

.....

Thank You for Trying Our Product

We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questons and Answers in PDF Format

Databricks-Machine-Learning-Associate Practice Exam Features:

* Databricks-Machine-Learning-Associate Questions and Answers Updated Frequently

* Databricks-Machine-Learning-Associate Practice Questions Verified by Expert Senior Certified Staff

* Databricks-Machine-Learning-Associate Most Realistic Questions that Guarantee you a Pass on Your FirstTry

* Databricks-Machine-Learning-Associate Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year