Free Questions for Databricks-Machine-Learning-Associate by dumpshq

Shared by Valentine on 09-08-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

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"]
)
```

Options:

A- Option A

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.

Reference

MLflow documentation on tracking experiments: https://www.mlflow.org/docs/latest/python_api/mlflow.html#mlflow.search_runs

Question 2

Question Type: **MultipleChoice**

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?

Options:

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.

Reference

Databricks documentation on creating clusters: https://docs.databricks.com/clusters/create.html

Question 3

Question Type: MultipleChoice

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?

Options:

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.

Reference

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

Question 4

Question Type: **MultipleChoice**

Which of the following approaches can be used to view the notebook that was run to create an MLflow run?

Options:

A- Open the MLmodel artifact in the MLflow run paqe

B- Click the 'Models' link in the row corresponding to the run in the MLflow experiment paqe

C- Click the 'Source' link in the row corresponding to the run in the MLflow experiment page

D- Click the 'Start Time' link in the row corresponding to the run in the MLflow experiment page

Answer:

C

Explanation:

To view the notebook that was run to create an MLflow run, you can click the 'Source' link in the row corresponding to the run in the MLflow experiment page. The 'Source' link provides a direct reference to the source notebook or script that initiated the run, allowing you to review the code and methodology used in the experiment. This feature is particularly useful for reproducibility and for understanding the context of the experiment. Reference:

MLflow Documentation (Viewing Run Sources and Notebooks).

Question 5

Question Type: MultipleChoice

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?

Options:

A- They can turn on Databricks Autologging

B- They can specify nested=True when starting the child run for each unique combination of hyperparameter values

C- They can start each child run inside the parent run's indented code block using mlflow.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 tuning process

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 specify nested=True when 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. Reference:

MLflow Documentation (Managing Nested Runs).

Question 6

Question Type: MultipleChoice

A machine learning engineer is converting a decision tree from sklearn to Spark ML. They notice that they are receiving different results despite all of their data and manually specified hyperparameter values being identical.

Which of the following describes a reason that the single-node sklearn decision tree and the Spark ML decision tree can differ?

Options:

A- Spark ML decision trees test every feature variable in the splitting algorithm

B- Spark ML decision trees automatically prune overfit trees

C- Spark ML decision trees test more split candidates in the splitting algorithm

D- Spark ML decision trees test a random sample of feature variables in the splitting algorithm

E- Spark ML decision trees test binned features values as representative split candidates

Answer:

E

Explanation:

One reason that results can differ between sklearn and Spark ML decision trees, despite identical data and hyperparameters, is that Spark ML decision trees test binned feature values as representative split candidates. Spark ML uses a method called 'quantile binning' to reduce the number of potential split points by grouping continuous features into bins. This binning process can lead to different splits compared to sklearn, which tests all possible split points directly. This difference in the splitting algorithm can cause variations in the resulting trees. Reference:

Spark MLlib Documentation (Decision Trees and Quantile Binning).

Question 7

Question Type: MultipleChoice

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?

Options:

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. Reference:

Spark MLlib Documentation (Linear Regression with Iterative Optimization).

Question 8

Question Type: MultipleChoice

Which of the following machine learning algorithms typically uses bagging?

Options:

A- Gradient boosted trees

B- K-means

C- Random forest

D- Linear regression

E- Decision tree

Answer:

C

Explanation:

Random Forest is a machine learning algorithm that typically uses bagging (Bootstrap Aggregating). Bagging involves training multiple models independently on different random subsets of the data and then combining their predictions. Random Forests consist of many

decision trees trained on random subsets of the training data and features, and their predictions are averaged to improve accuracy and control overfitting. This method enhances model robustness and predictive performance. Reference:

Ensemble Methods in Machine Learning (Understanding Bagging and Random Forests).

Question 9

Question Type: MultipleChoice

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?

Options:

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.

Reference

Introduction to Ensemble Learning: https://machinelearningmastery.com/ensemble-machine-learning-algorithms-python-scikit-learn/

Question 10

Question Type: MultipleChoice

A data scientist has been given an incomplete notebook from the data engineering team. The notebook uses a Spark DataFrame spark_df on which the data scientist needs to perform further feature engineering. Unfortunately, the data scientist has not yet learned the PySpark DataFrame API.

Which of the following blocks of code can the data scientist run to be able to use the pandas API on Spark?

Options:

A- import pyspark.pandas as ps
df = ps.DataFrame(spark_df)

B- import pyspark.pandas as ps
df = ps.to_pandas(spark_df)

C- spark_df.to_sql()

D- import pandas as pd
df = pd.DataFrame(spark_df)

E- spark_df.to_pandas()

Answer:

A

Explanation:

To use the pandas API on Spark, which is designed to bridge the gap between the simplicity of pandas and the scalability of Spark, the correct approach involves importing the pyspark.pandas (recently renamed to pandas_api_on_spark) module and converting a Spark DataFrame to a pandas-on-Spark DataFrame using this API. The provided syntax correctly initializes a pandas-on-Spark DataFrame, allowing the data scientist to work with the familiar pandas-like API on large datasets managed by Spark.

Reference

Pandas API on Spark Documentation: https://spark.apache.org/docs/latest/api/python/user_guide/pandas_on_spark/index.html