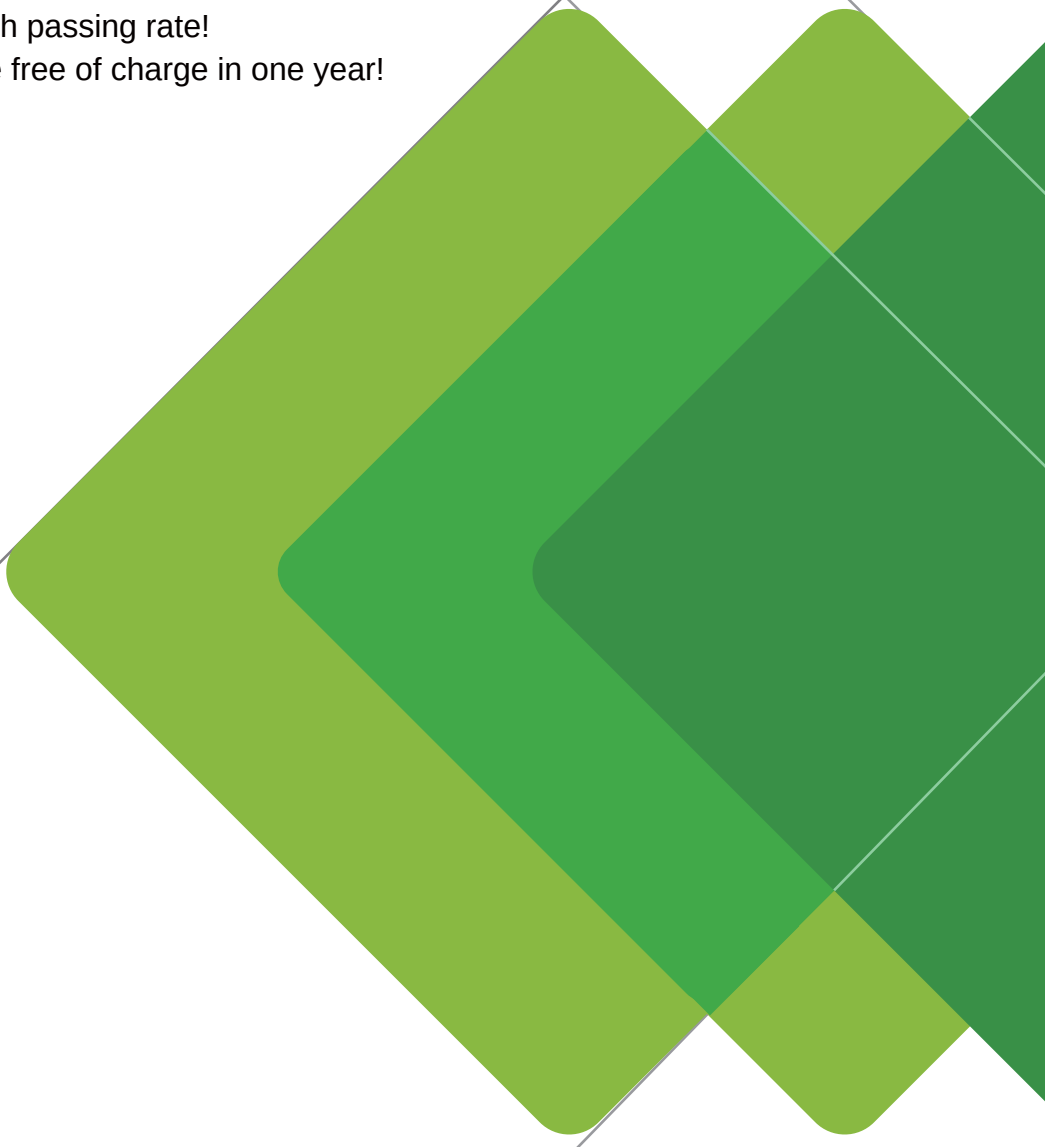
IT-DUMPS Q&A

Accurate study guides, High passing rate!
IT-dumps provides update free of charge in one year!



Exam : **Databricks Machine Learning Associate**

Title : Databricks Certified Machine Learning Associate Exam

Version : DEMO

1.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 as fs), the correct method involves calling get_table on the fs client with the table name as an argument, followed by accessing the description attribute of the returned object. The code snippet fs.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.
Reference: Databricks Feature Store documentation (Accessing Feature Table Metadata).

2.A data scientist has a Spark DataFrame spark_df. They want to create a new Spark DataFrame that contains only the rows from spark_df where the value in column price is greater than 0.
Which of the following code blocks will accomplish this task?
A. spark_df[spark_df["price"] > 0]
B. spark_df.filter(col("price") > 0)
C. SELECT * FROM spark_df WHERE price > 0
D. spark_df.loc[spark_df["price"] > 0,:]
E. spark_df.loc[:,spark_df["price"] > 0]
Answer: B
Explanation:
To filter rows in a Spark DataFrame based on a condition, you use the filter method along with a column condition. The correct syntax in PySpark to accomplish this task is spark_df.filter(col("price")
> 0), which filters the DataFrame to include only those rows where the value in the "price" column is greater than 0. The col function is used to specify column-based operations. The other options provided either do not use correct Spark DataFrame syntax or are intended for different types of data manipulation frameworks like pandas.
Reference: PySpark DataFrame API documentation (Filtering DataFrames).

3.A health organization is developing a classification model to determine whether or not a patient currently has a specific type of infection. The organization's leaders want to maximize the number of positive cases identified by the model.
Which of the following classification metrics should be used to evaluate the model?
A. RMSE
B. Precision
C. Area under the residual operating curve

D. Accuracy
E. Recall
Answer: E
Explanation:
When the goal is to maximize the identification of positive cases in a classification task, the metric of interest is Recall. Recall, also known as sensitivity, measures the proportion of actual positives that are correctly identified by the model (i.e., the true positive rate). It is crucial for scenarios where missing a positive case (false negative) has serious implications, such as in medical diagnostics. The other metrics like Precision, RMSE, and Accuracy serve different aspects of performance measurement and are not specifically focused on maximizing the detection of positive cases alone.
Reference: Classification Metrics in Machine Learning (Understanding Recall).

4.In which of the following situations is it preferable to impute missing feature values with their median value over the mean value?
A. When the features are of the categorical type
B. When the features are of the boolean type
C. When the features contain a lot of extreme outliers
D. When the features contain no outliers
E. When the features contain no missing no values
Answer: C
Explanation:
Imputing missing values with the median is often preferred over the mean in scenarios where the data contains a lot of extreme outliers. The median is a more robust measure of central tendency in such cases, as it is not as heavily influenced by outliers as the mean. Using the median ensures that the imputed values are more representative of the typical data point, thus preserving the integrity of the dataset's distribution. The other options are not specifically relevant to the question of handling outliers in numerical data.
Reference: Data Imputation Techniques (Dealing with Outliers).

5.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.
Reference
"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