DP-100^{Q&As}

Designing and Implementing a Data Science Solution on Azure

Pass Microsoft DP-100 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.passapply.com/dp-100.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Microsoft
Official Exam Center

 Instant Download After Purchase

 100% Money Back Guarantee

 365 Days Free Update

 800,000+ Satisfied Customers

QUESTION 1

HOTSPOT

You create an Azure Machine Learning workspace and a dataset. The dataset includes age values for a large group of diabetes patients. You use the dp.mean function from the SmartNoise library to calculate the mean of the age value. You

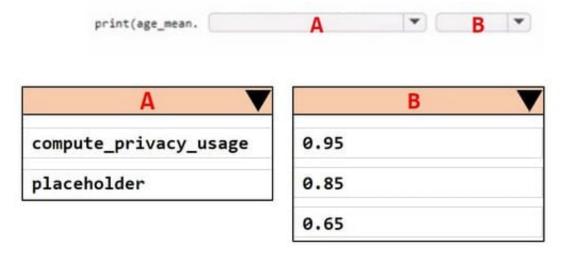store the value in a variable named age.mean.

You must output the value of the interval range of released mean values that will be returned 95 percent of the time.

You need to complete the code.

Which code values should you use? To answer, select the appropriate options in the answer area

NOTE: Each correct selection is worth one point.

Hot Area:

print(age_mean. [A ▼] [B ▼]

A ▼
compute_privacy_usage
placeholder

B ▼
0.95
0.85
0.65

Correct Answer:

print(age_mean. [A ▼] [B ▼]

A ▼
compute_privacy_usage
placeholder

B ▼
0.95
0.85
0.65

QUESTION 2

HOTSPOT

You are running a training experiment on remote compute in Azure Machine Learning.

The experiment is configured to use a conda environment that includes the mlflow and azureml-contrib-run packages.

You must use MLflow as the logging package for tracking metrics generated in the experiment.

You need to complete the script for the experiment.

How should you complete the code? To answer, select the appropriate options in the answer area.

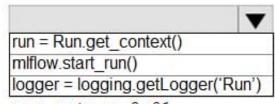NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

```
import numpy as np
# Import library to log metrics
```

▼
from azureml.core import Run
import mlflow
import logging

```
# Start logging for this run
```

▼
run = Run.get_context()
mlflow.start_run()
logger = logging.getLogger('Run')

```
reg_rate = 0.01
# Log the reg_rate metric
```

▼
run.log('reg_rate', np.float(reg_rate))
mlflow.log_metric('reg_rate', np.float(reg_rate))
logger.info(np.float(reg_rate))

```
# Stop logging for this run
```

▼
run.complete()
mlflow.end_run()
logger.setLevel(logging.INFO)

Correct Answer:

Answer Area

```
import numpy as np
# Import library to log metrics
```

▼
from azureml.core import Run
import mlflow
import logging

```
# Start logging for this run
```

▼
run = Run.get_context()
mlflow.start_run()
logger = logging.getLogger('Run')

```
reg_rate = 0.01
# Log the reg_rate metric
```

▼
run.log('reg_rate', np.float(reg_rate))
mlflow.log_metric('reg_rate', np.float(reg_rate))
logger.info(np.float(reg_rate))

```
# Stop logging for this run
```

▼
run.complete()
mlflow.end_run()
logger.setLevel(logging.INFO)

Box 1: import mlflow

Import the mlflow and Workspace classes to access MLflow\'s tracking URI and configure your workspace.

Box 2: mlflow.start_run()

Set the MLflow experiment name with set_experiment() and start your training run with start_run().

Box 3: mlflow.log_metric(\\' ..\\')

Use log_metric() to activate the MLflow logging API and begin logging your training run metrics.

Box 4: mlflow.end_run()

Close the run:

run.endRun()

Reference:

https://docs.microsoft.com/en-us/azure/machine-learning/how-to-use-mlflow

QUESTION 3

HOTSPOT

You are implementing hyperparameter tuning for a model training from a notebook. The notebook is in an Azure Machine Learning workspace. You add code that imports all relevant Python libraries.

You must configure Bayesian sampling over the search space for the num_hidden_layers and batch_size hyperparameters.

You need to complete the following Python code to configure Bayesian sampling.

Which code segments should you use? To answer, select the appropriate options in the answer area

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

```
param_sampling = BayesianParameterSampling( {
        "learning_rate": uniform(0.05, 0.1),
        "batch_size":        A        ▼ (        B        ▼ (16, 128, 16))
    }
)
```

A ▼
loguniform
Placeholder

B ▼
range
Placeholder

Correct Answer:

Answer Area

```
param_sampling = BayesianParameterSampling( {
        "learning_rate": uniform(0.05, 0.1),
        "batch_size":        A        ▼ (        B        ▼ (16, 128, 16))
    }
)
```

A   ▼
loguniform
Placeholder

B   ▼
range
Placeholder

QUESTION 4

You use Azure Machine Learning designer to create a training pipeline for a regression model.

You need to prepare the pipeline for deployment as an endpoint that generates predictions asynchronously for a dataset of input data values.

What should you do?

A. Clone the training pipeline.

B. Create a batch inference pipeline from the training pipeline.

C. Create a real-time inference pipeline from the training pipeline.

D. Replace the dataset in the training pipeline with an Enter Data Manually module.

Correct Answer: C

You must first convert the training pipeline into a real-time inference pipeline. This process removes training modules and adds web service inputs and outputs to handle requests. Incorrect Answers:

A: Use the Enter Data Manually module to create a small dataset by typing values.

Reference: https://docs.microsoft.com/en-us/azure/machine-learning/tutorial-designer-automobile-price-deploy
https://docs.microsoft.com/en-us/azure/machine-learning/algorithm-module-reference/enter-data-manually

QUESTION 5

You are evaluating a completed binary classification machine learning model.

You need to use the precision as the evaluation metric.

Which visualization should you use?

A. Violin plot

B. Gradient descent

C. Box plot

D. Binary classification confusion matrix

Correct Answer: D

Incorrect Answers:

A: A violin plot is a visual that traditionally combines a box plot and a kernel density plot.

B: Gradient descent is a first-order iterative optimization algorithm for finding the minimum of a function. To find a local minimum of a function using gradient descent, one takes steps proportional to the negative of the gradient (or approximate gradient) of the function at the current point.

C: A box plot lets you see basic distribution information about your data, such as median, mean, range and quartiles but doesn\\'t show you how your data looks throughout its range.

References: https://machinelearningknowledge.ai/confusion-matrix-and-performance-metrics-machine-learning/

Latest DP-100 Dumps   DP-100 PDF Dumps   DP-100 VCE Dumps