✿ Custom View Settings

Question #33   *Topic 1*

You have a demand forecasting pipeline in production that uses Dataflow to preprocess raw data prior to model training and prediction. During preprocessing, you employ Z-score normalization on data stored in BigQuery and write it back to BigQuery. New training data is added every week. You want to make the process more efficient by minimizing computation time and manual intervention. What should you do?

A. Normalize the data using Google Kubernetes Engine.

B. Translate the normalization algorithm into SQL for use with BigQuery.

C. Use the normalizer_fn argument in TensorFlow's Feature Column API.

D. Normalize the data with Apache Spark using the Dataproc connector for BigQuery.

Question #34   *Topic 1*

You need to design a customized deep neural network in Keras that will predict customer purchases based on their purchase history. You want to explore model performance using multiple model architectures, store training data, and be able to compare the evaluation metrics in the same dashboard. What should you do?

A. Create multiple models using AutoML Tables.

B. Automate multiple training runs using Cloud Composer.

C. Run multiple training jobs on AI Platform with similar job names.

D. Create an experiment in Kubeflow Pipelines to organize multiple runs.

Question #35   *Topic 1*

You are developing a Kubeflow pipeline on Google Kubernetes Engine. The first step in the pipeline is to issue a query against BigQuery. You plan to use the results of that query as the input to the next step in your pipeline. You want to achieve this in the easiest way possible. What should you do?

A. Use the BigQuery console to execute your query, and then save the query results into a new BigQuery table.

B. Write a Python script that uses the BigQuery API to execute queries against BigQuery. Execute this script as the first step in your Kubeflow pipeline.

C. Use the Kubeflow Pipelines domain-specific language to create a custom component that uses the Python BigQuery client library to execute queries.

D. Locate the Kubeflow Pipelines repository on GitHub. Find the BigQuery Query Component, copy that component's URL, and use it to load the component into your pipeline. Use the component to execute queries against BigQuery.

You are building a model to predict daily temperatures. You split the data randomly and then transformed the training and test datasets. Temperature data for model training is uploaded hourly. During testing, your model performed with 97% accuracy; however, after deploying to production, the model's accuracy dropped to 66%. How can you make your production model more accurate?

A. Normalize the data for the training, and test datasets as two separate steps.

B. Split the training and test data based on time rather than a random split to avoid leakage.

C. Add more data to your test set to ensure that you have a fair distribution and sample for testing.

D. Apply data transformations before splitting, and cross-validate to make sure that the transformations are applied to both the training and test sets.

← Previous Questions
Next Questions →