📖 06a_timeseries_forecasting_vertex_ai_notebook_execution.md

Timeseries Forecasting using sessions in Serverless Spark through Vertex AI

Following are the lab modules:

1. Understanding Data

Data Files

The datasets used for this project are:

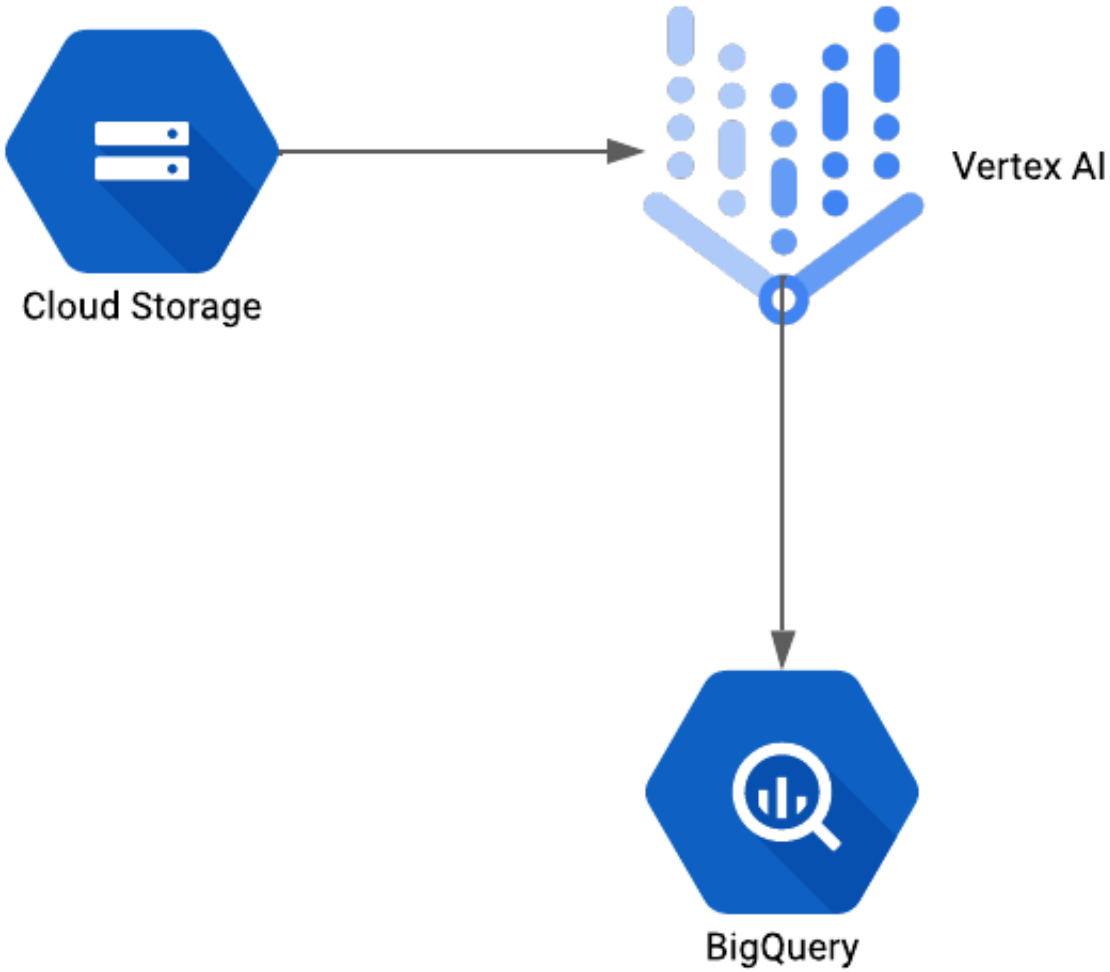
- train.csv: This file contains the date, store, item, sales data.

date - Date of the sale data. There are no holiday effects or store closures.
store - Store ID
item - Item ID
sales - Number of items sold at a particular store on a particular date.

- test.csv:

id- Unique identifier
date - Date of the sale data. There are no holiday effects or store closures.
store - Store ID
item - Item ID

2. Solution Architecture

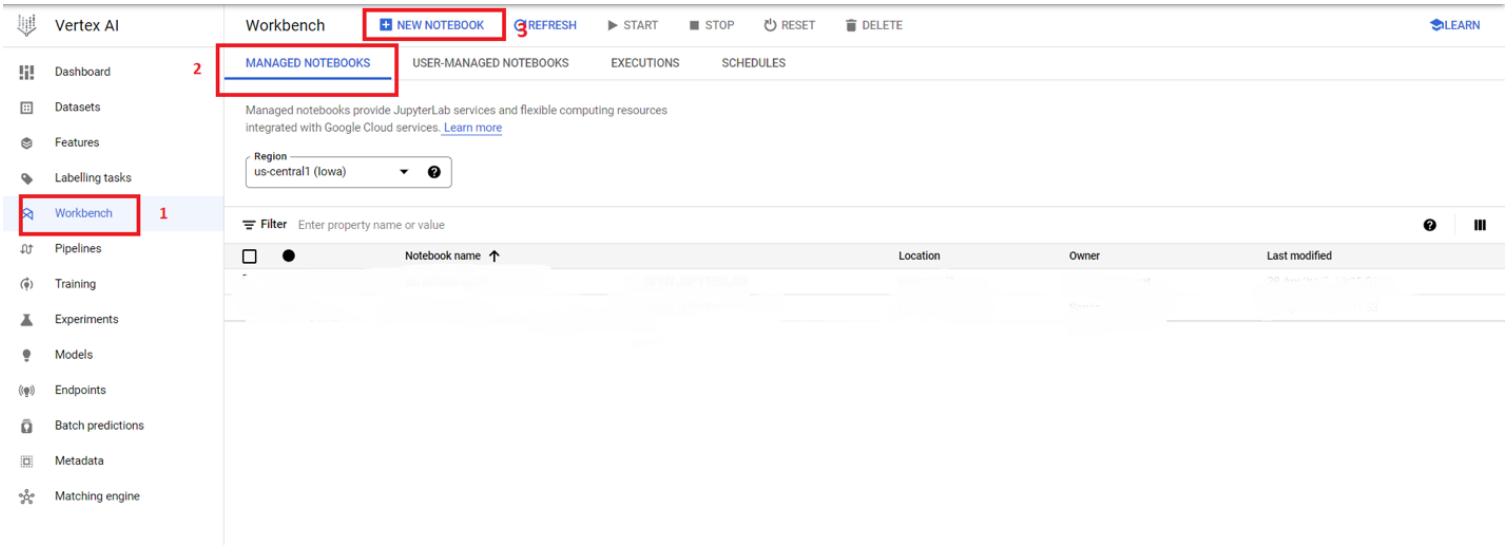


3. Execution

3.1. Run the Batch by creating session.

Creating Notebook in Vertex AI

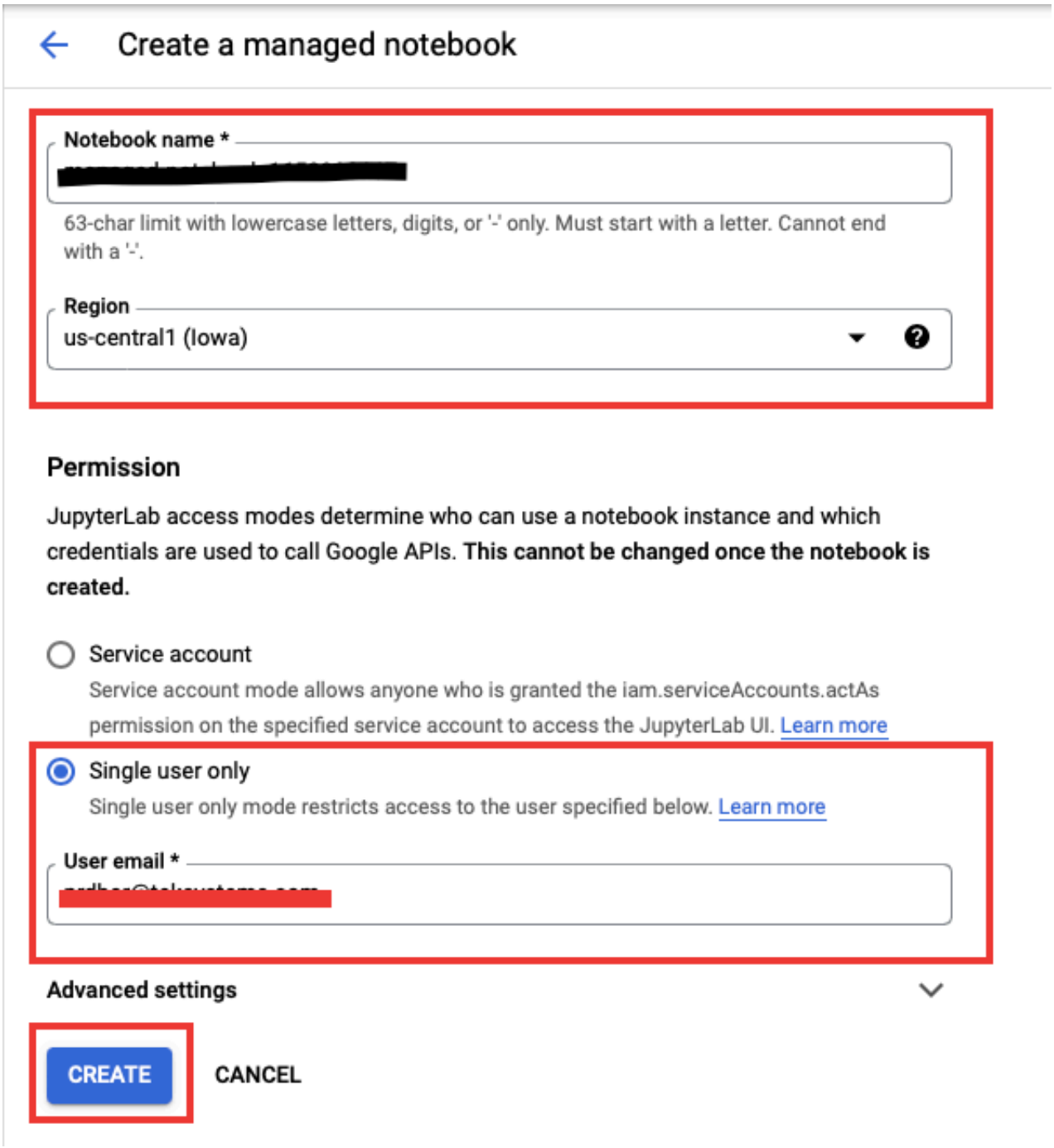
Select Workbench from the left scroll bar of the Vertex AI main page. Select the Managed Notebooks tab. In the Managed Notebooks tab , click the New Notebook icon.



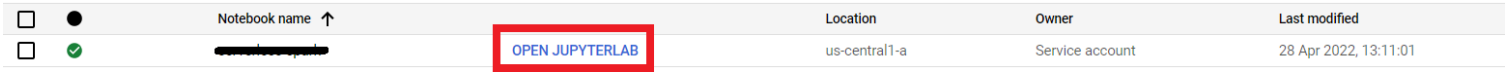
Next, fill in the following values in the Notebook creation window as shown in the images below:

- **Notebook Name** - A unique identifier for your Notebook
- **Region** - The region name provided by the Admin team
- **Permission Type** - Single User Only (Single user only mode restricts access to the specified user)

- Provide a name and region to the notebook and select 'Single User Only' and click 'Create'. We will let the 'Advanced Settings' remain as the default values.



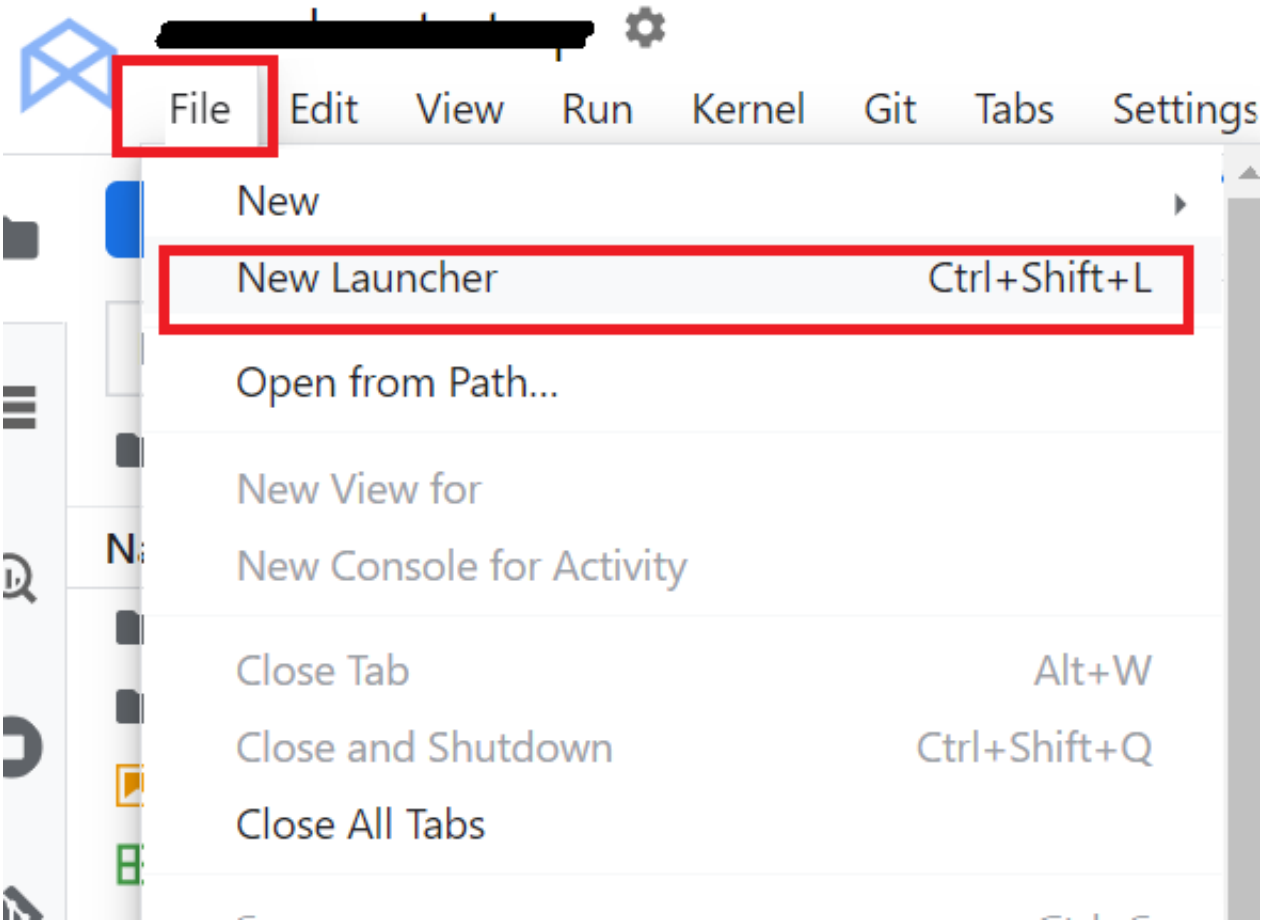
- Once the notebook is running, click the 'OPEN JUPYTERLAB' option next to the Notebook name as shown below

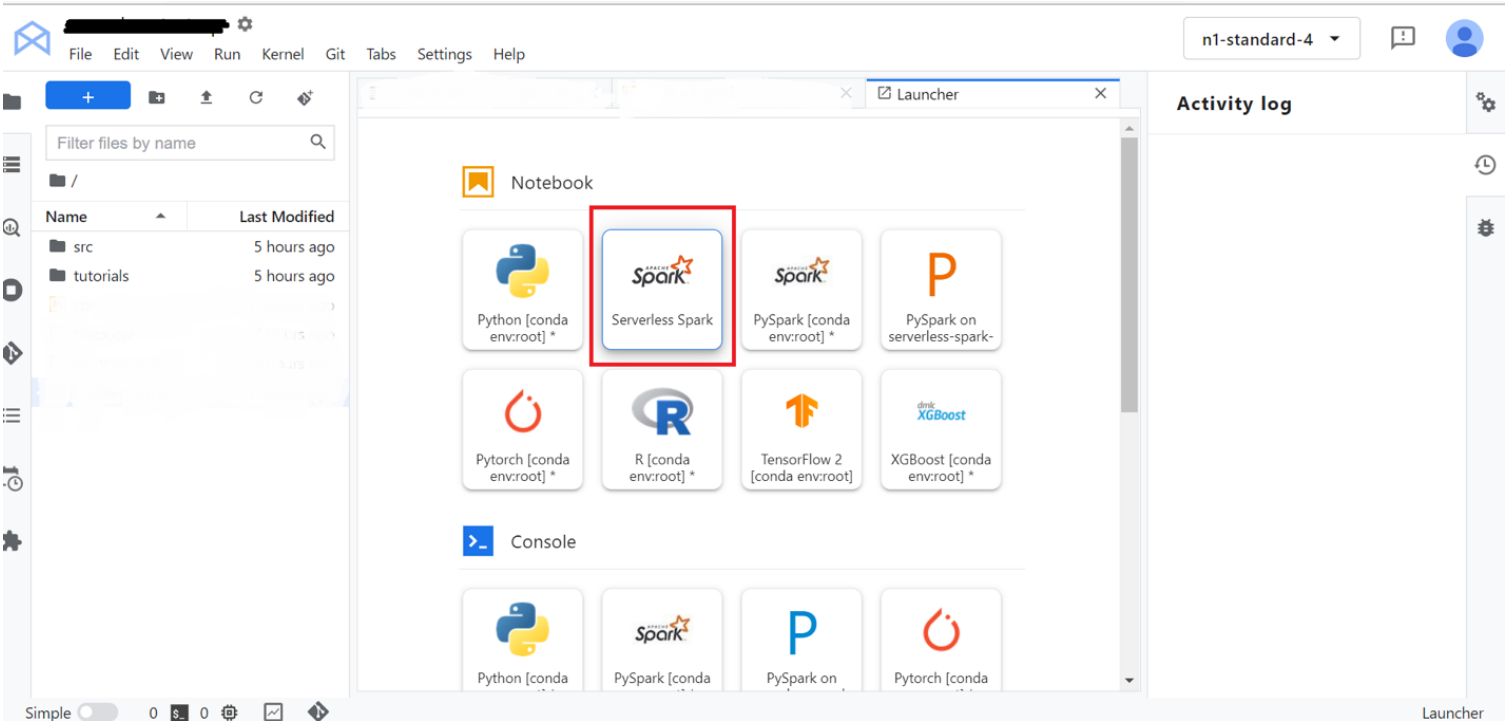


- Follow the on screen instructions to launch the JupyterLab session

Create Serverless Spark Session

- Click on the File and the New launcher and select Serverless Spark





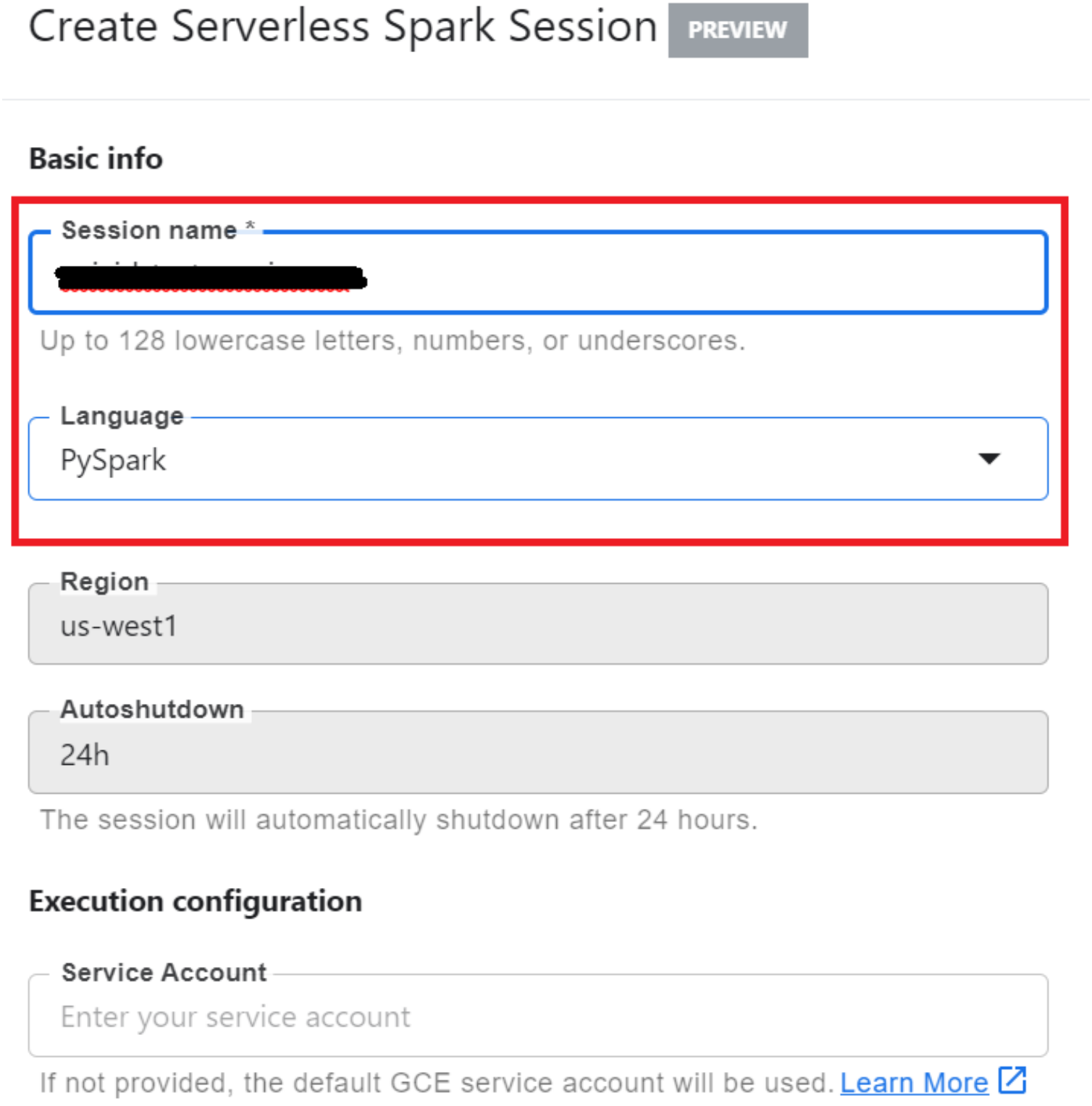
Follow the on screen instructions to create Session

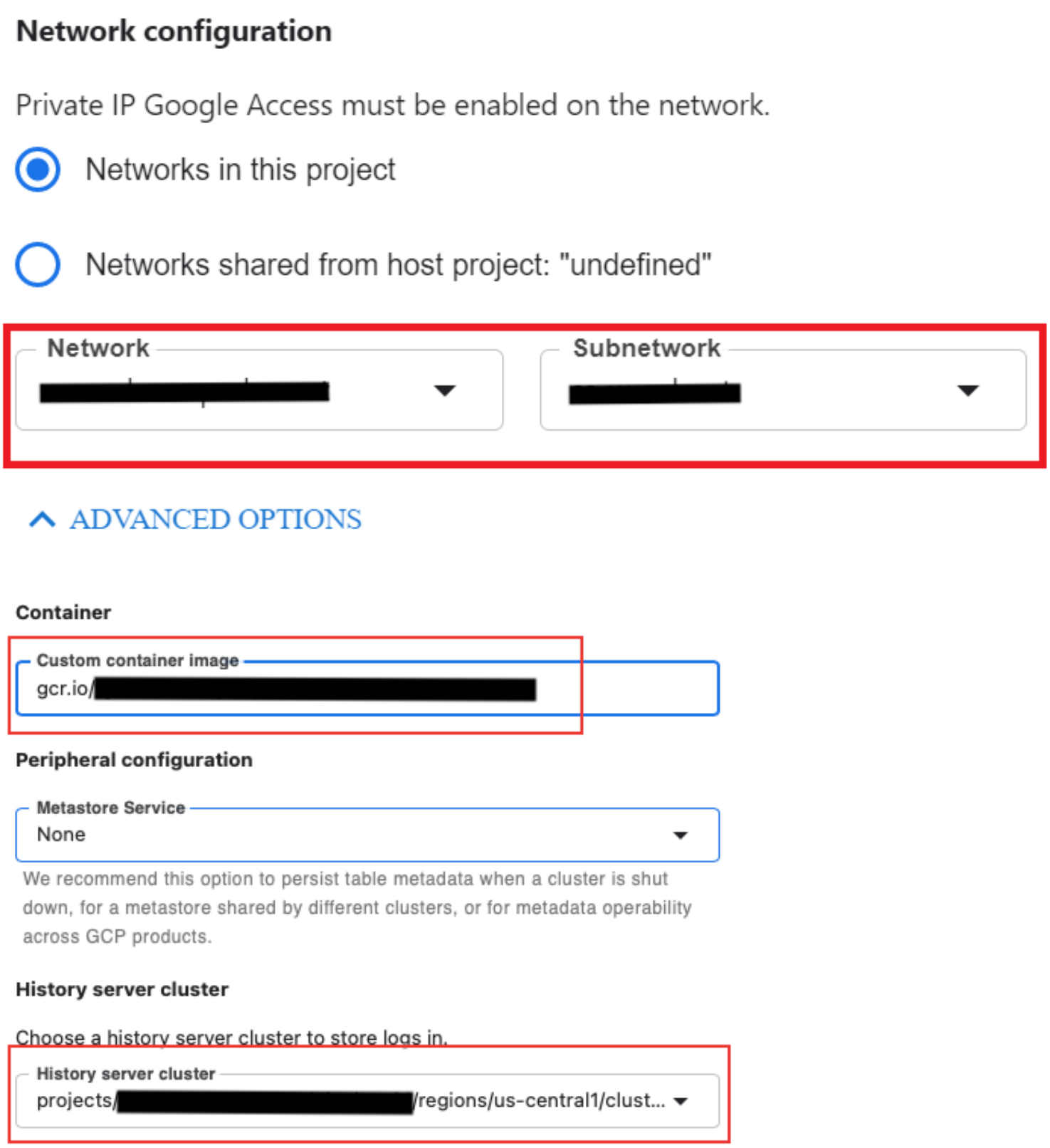
3.2. Provide the details for the Session

Next, fill in the following values in the session creation window as shown in the images below:

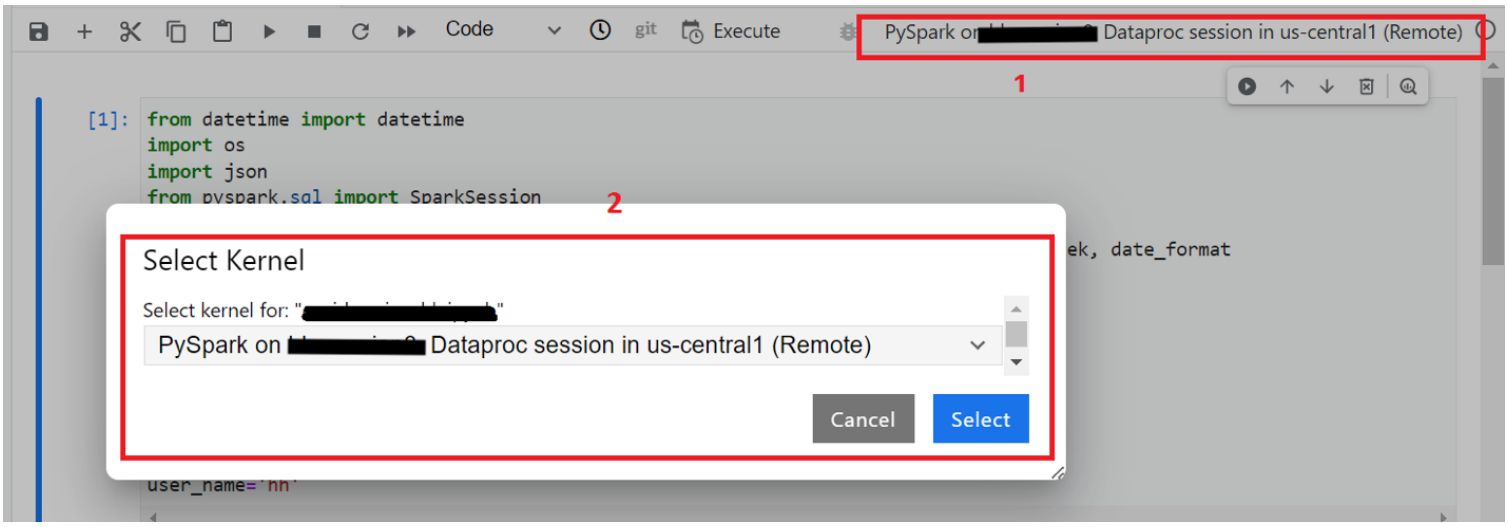
- **Session Name** - A unique identifier for your session
- **Region** - The region name provided by the Admin team
- **Language** - Pyspark
- **Autoshutdown** - 24 hours
- **Service Account** - <UMSA_NAME>@<PROJECT_ID>.iam.gserviceaccount.com
- **Network Configuration** - Select the network and subnetwork provided by the Admin team
- **History Server Cluster** - projects/<PROJECT_ID>/regions/<REGION_NAME>/clusters/<HISTORY_SERVER_NAME>
- **Container** - gcr.io/<PROJECT_ID>/<CONTAINER_IMAGE>:1.0.1

- Click the **SUBMIT** button to create the session.

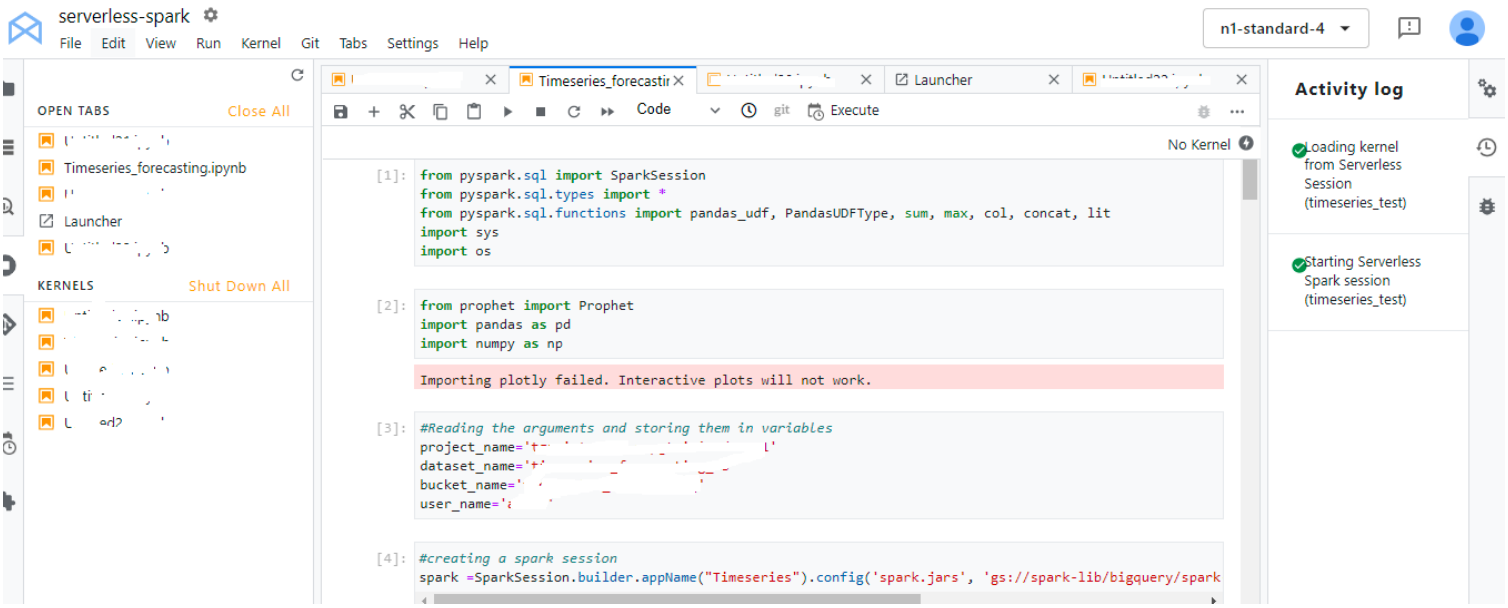

Create Serverless Spark Session
PREVIEW
Basic info
Session name *
[redacted]
Up to 128 lowercase letters, numbers, or underscores.
Language
PySpark
Region
us-west1
Autoshutdown
24h
The session will automatically shutdown after 24 hours.
Execution configuration
Service Account
Enter your service account
If not provided, the default GCE service account will be used. Learn More



- Once the Session is created select 'No Kernel' from the kernel dropdown list and then delete the notebook



- Next, using the browser option from JupyterLab, navigate to the Notebook file located at: <bucket_name> > 'timeseries_forecasting' > 00-scripts > timeseries_forecasting.ipynb
- From the kernel dropdown list, select the kernel for the session created in section 3.2
- Pass the values to the variables project_name, dataset_name, bucket_name as provided by the Admin and replace user_name by your username
- Next, hit the **Execute** button as shown below to run the code in the notebook.



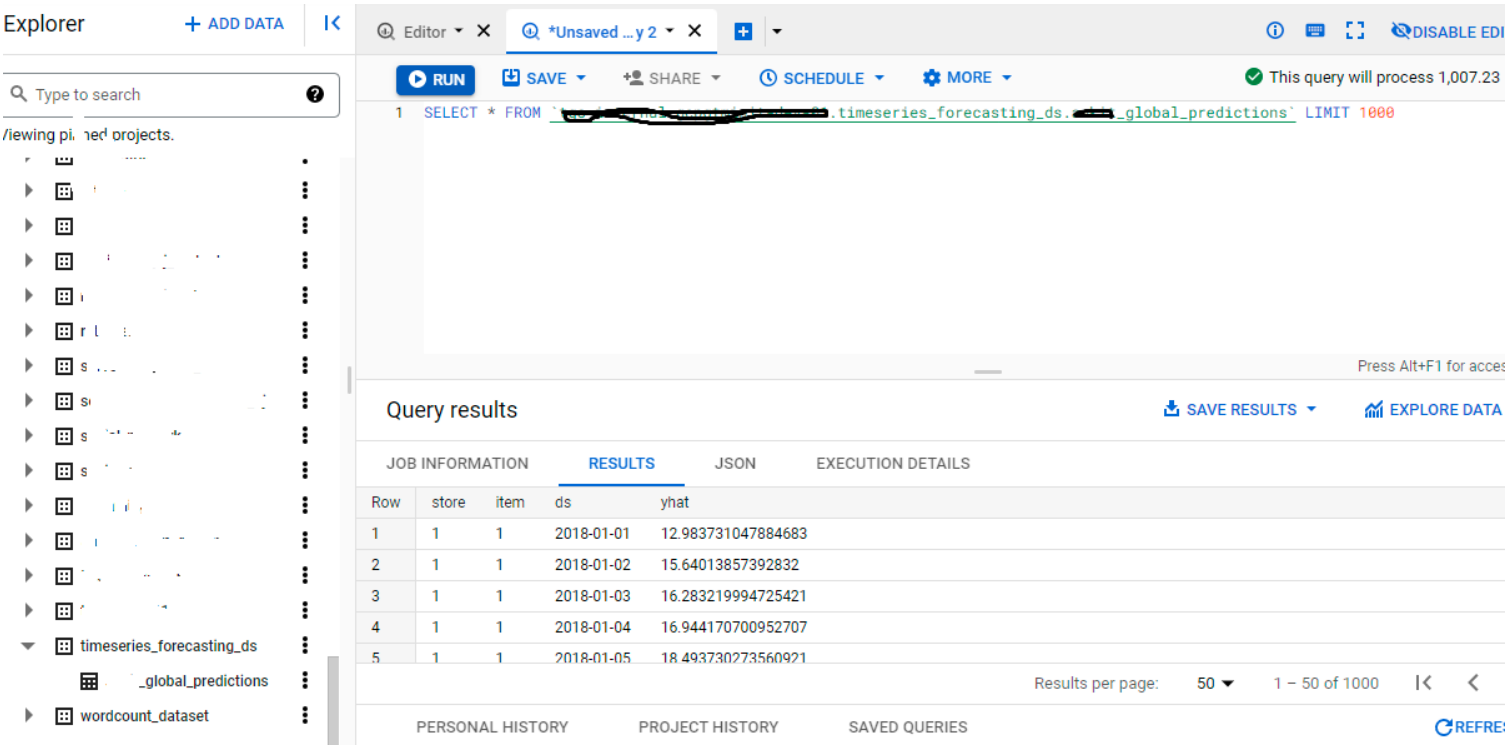
3.3. Check the output table in BigQuery

Navigate to BigQuery Console, and check the **timeseries_forecasting** dataset.
Once the code has successfully executed, new table '<your_name_here>_global_predictions' will be created :

To query the data to find the list of stocks with highest stringency Index, run the following query -

```
select * from `<GCP-PROJECT-NAME>.<BQ-DATASET-NAME>.<user_name>_global_predictions`
```

Note: Edit all occurrences of and to match the values of the variables PROJECT_ID,user_name and BQ_DATASET_NAME respectively

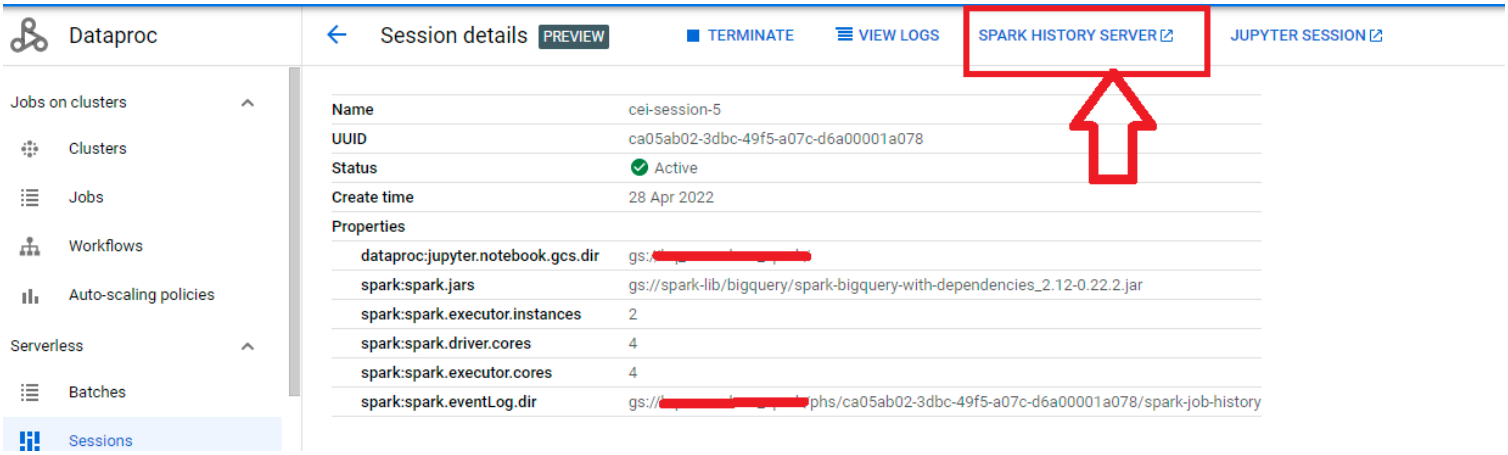


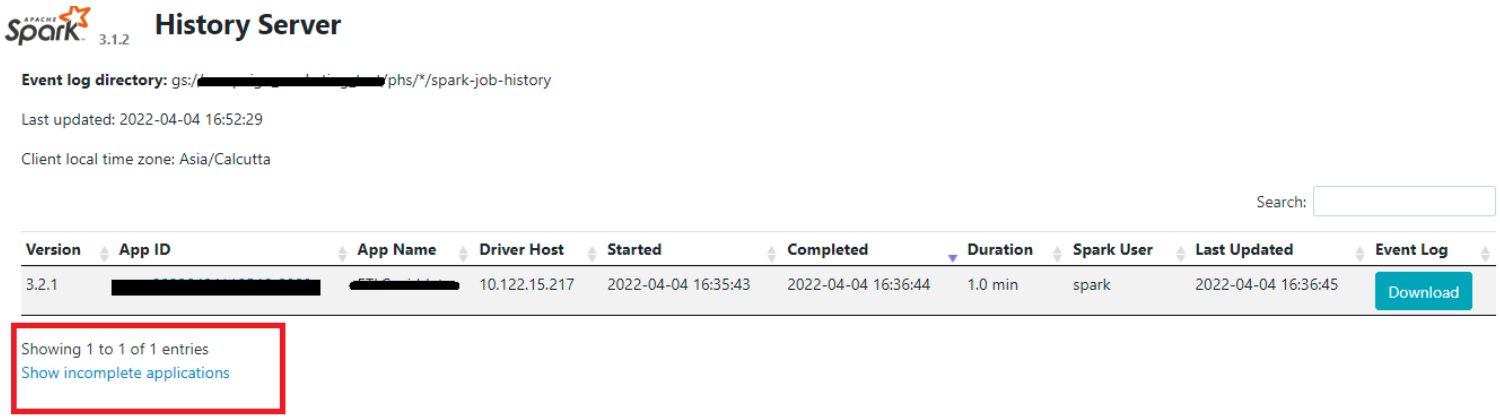
4. Logging

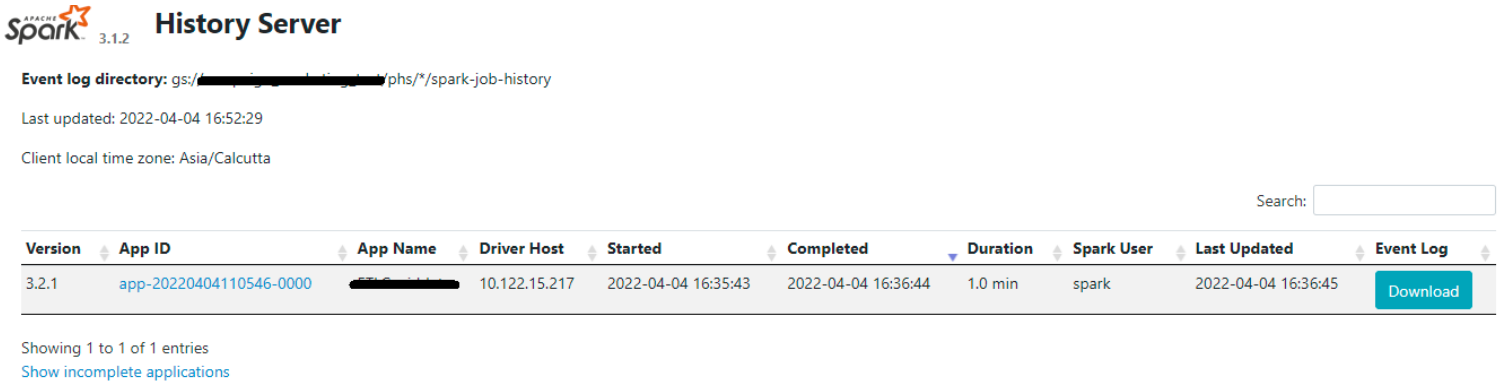
4.1 Persistent History Server logs

To view the Persistent History server logs, click the 'View History Server' button on the Sessions monitoring page and the logs will be shown as below:

As the session is still in active state , we will be able to find the logs in show incomplete applications.


Dataproc
Session details PREVIEW
TERMINATE
VIEW LOGS
SPARK HISTORY SERVER
JUPYTER SESSION
Jobs on clusters
Clusters
Jobs
Workflows
Auto-scaling policies
Serverless
Batches
Sessions
Name cei-session-5
UUID ca05ab02-3dbc-49f5-a07c-d6a00001a078
Status Active
Create time 28 Apr 2022
Properties
dataproc:jupyter.notebook.gcs.dir
spark:spark.jars gs://spark-lib/bigquery/spark-bigquery-with-dependencies_2.12-0.22.2.jar
spark:spark.executor.instances 2
spark:spark.driver.cores 4
spark:spark.executor.cores 4
spark:spark.eventLog.dir


Spark 3.1.2
History Server
Event log directory: gs://[redacted]/phs/*/spark-job-history
Last updated: 2022-04-04 16:52:29
Client local time zone: Asia/Calcutta
Search:
Version App ID App Name Driver Host Started Completed Duration Spark User Last Updated Event Log
3.2.1 [redacted] [redacted] 10.122.15.217 2022-04-04 16:35:43 2022-04-04 16:36:44 1.0 min spark 2022-04-04 16:36:45 Download
Showing 1 to 1 of 1 entries
Show incomplete applications


Spark 3.1.2
History Server
Event log directory: gs://[redacted]/phs/*/spark-job-history
Last updated: 2022-04-04 16:52:29
Client local time zone: Asia/Calcutta
Search:
Version App ID App Name Driver Host Started Completed Duration Spark User Last Updated Event Log
3.2.1 app-20220404110546-0000 [redacted] 10.122.15.217 2022-04-04 16:35:43 2022-04-04 16:36:44 1.0 min spark 2022-04-04 16:36:45 Download
Showing 1 to 1 of 1 entries
Show incomplete applications