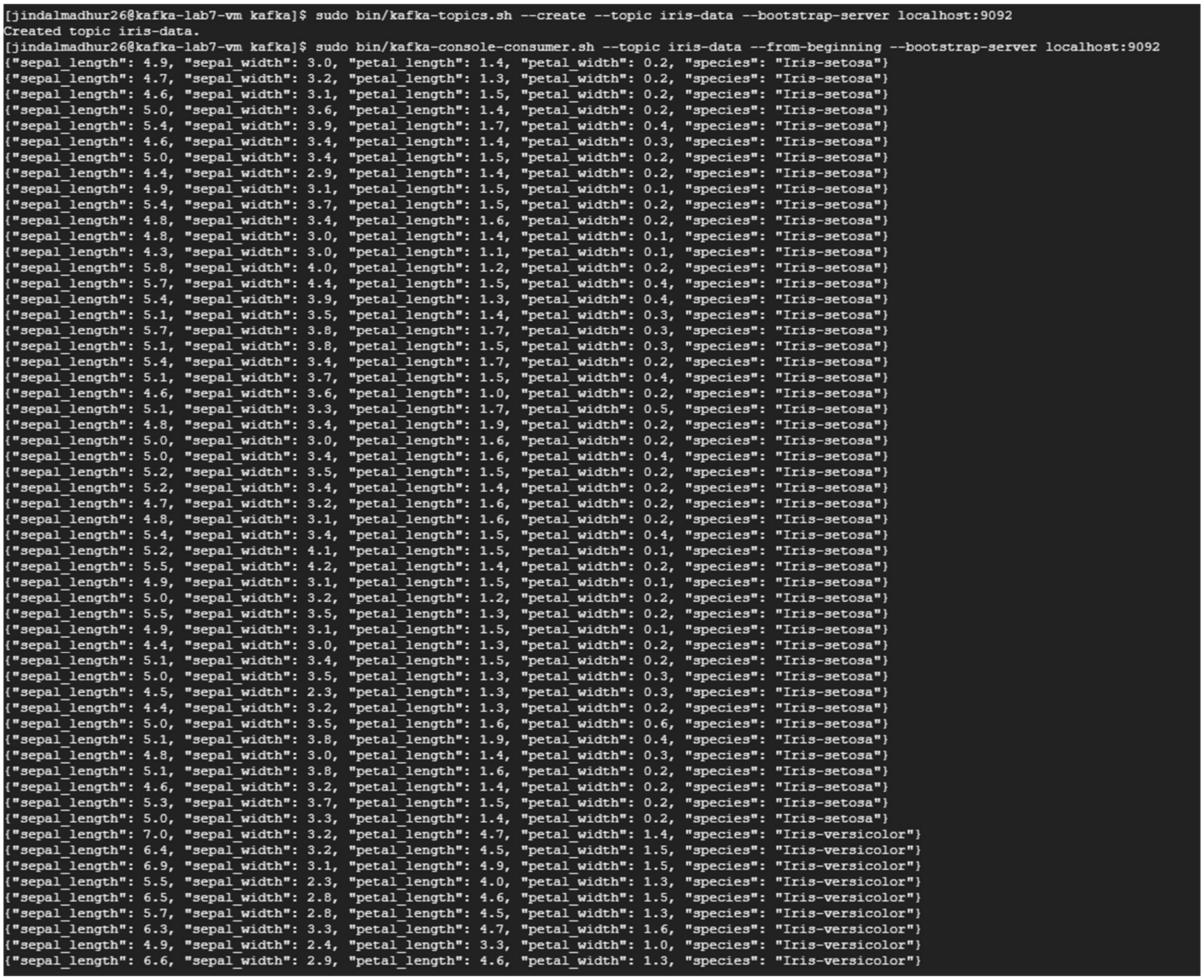
Lab 7 | ME18B059 | Madhur Jindal

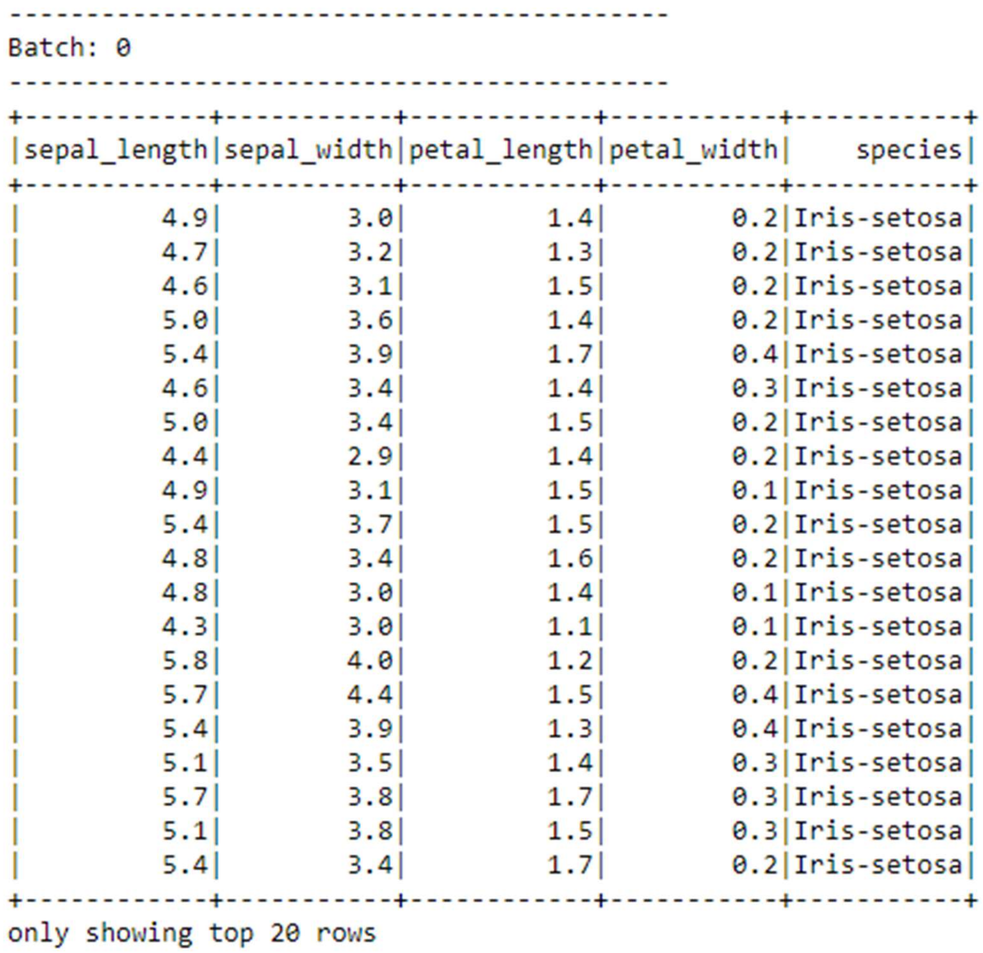
1. The producer_json.py file contains the code for reading in the csv file and publishing messages as json format to the topic: iris-data



```
[jindalmadhur26@kafka-lab7-vm kafka]$ sudo bin/kafka-topics.sh --create --topic iris-data --bootstrap-server localhost:9092
Created topic iris-data.
[jindalmadhur26@kafka-lab7-vm kafka]$ sudo bin/kafka-console-consumer.sh --topic iris-data --from-beginning --bootstrap-server localhost:9092
{"sepal_length": 4.9, "sepal_width": 3.0, "petal_length": 1.4, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 4.7, "sepal_width": 3.2, "petal_length": 1.3, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 4.6, "sepal_width": 3.1, "petal_length": 1.5, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.0, "sepal_width": 3.6, "petal_length": 1.4, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.4, "sepal_width": 3.9, "petal_length": 1.7, "petal_width": 0.4, "species": "Iris-setosa"}
{"sepal_length": 4.6, "sepal_width": 3.4, "petal_length": 1.4, "petal_width": 0.3, "species": "Iris-setosa"}
{"sepal_length": 5.0, "sepal_width": 3.4, "petal_length": 1.5, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 4.4, "sepal_width": 2.9, "petal_length": 1.4, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 4.9, "sepal_width": 3.1, "petal_length": 1.5, "petal_width": 0.1, "species": "Iris-setosa"}
{"sepal_length": 5.4, "sepal_width": 3.7, "petal_length": 1.5, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 4.8, "sepal_width": 3.4, "petal_length": 1.6, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 4.8, "sepal_width": 3.0, "petal_length": 1.4, "petal_width": 0.1, "species": "Iris-setosa"}
{"sepal_length": 4.3, "sepal_width": 3.0, "petal_length": 1.1, "petal_width": 0.1, "species": "Iris-setosa"}
{"sepal_length": 5.8, "sepal_width": 4.0, "petal_length": 1.2, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.7, "sepal_width": 4.4, "petal_length": 1.5, "petal_width": 0.4, "species": "Iris-setosa"}
{"sepal_length": 5.4, "sepal_width": 3.9, "petal_length": 1.3, "petal_width": 0.4, "species": "Iris-setosa"}
{"sepal_length": 5.1, "sepal_width": 3.5, "petal_length": 1.4, "petal_width": 0.3, "species": "Iris-setosa"}
{"sepal_length": 5.7, "sepal_width": 3.8, "petal_length": 1.7, "petal_width": 0.3, "species": "Iris-setosa"}
{"sepal_length": 5.1, "sepal_width": 3.8, "petal_length": 1.5, "petal_width": 0.3, "species": "Iris-setosa"}
{"sepal_length": 5.4, "sepal_width": 3.4, "petal_length": 1.7, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.1, "sepal_width": 3.7, "petal_length": 1.5, "petal_width": 0.4, "species": "Iris-setosa"}
{"sepal_length": 4.6, "sepal_width": 3.6, "petal_length": 1.0, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.1, "sepal_width": 3.3, "petal_length": 1.7, "petal_width": 0.5, "species": "Iris-setosa"}
{"sepal_length": 4.8, "sepal_width": 3.4, "petal_length": 1.9, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.0, "sepal_width": 3.0, "petal_length": 1.6, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.0, "sepal_width": 3.4, "petal_length": 1.6, "petal_width": 0.4, "species": "Iris-setosa"}
{"sepal_length": 5.2, "sepal_width": 3.5, "petal_length": 1.5, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.2, "sepal_width": 3.4, "petal_length": 1.4, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 4.7, "sepal_width": 3.2, "petal_length": 1.6, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 4.8, "sepal_width": 3.1, "petal_length": 1.6, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.4, "sepal_width": 3.4, "petal_length": 1.5, "petal_width": 0.4, "species": "Iris-setosa"}
{"sepal_length": 5.2, "sepal_width": 4.1, "petal_length": 1.5, "petal_width": 0.1, "species": "Iris-setosa"}
{"sepal_length": 5.5, "sepal_width": 4.2, "petal_length": 1.4, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 4.9, "sepal_width": 3.1, "petal_length": 1.5, "petal_width": 0.1, "species": "Iris-setosa"}
{"sepal_length": 5.0, "sepal_width": 3.2, "petal_length": 1.2, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.5, "sepal_width": 3.5, "petal_length": 1.3, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 4.9, "sepal_width": 3.1, "petal_length": 1.5, "petal_width": 0.1, "species": "Iris-setosa"}
{"sepal_length": 4.4, "sepal_width": 3.0, "petal_length": 1.3, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.1, "sepal_width": 3.4, "petal_length": 1.5, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.0, "sepal_width": 3.5, "petal_length": 1.3, "petal_width": 0.3, "species": "Iris-setosa"}
{"sepal_length": 4.5, "sepal_width": 2.3, "petal_length": 1.3, "petal_width": 0.3, "species": "Iris-setosa"}
{"sepal_length": 4.4, "sepal_width": 3.2, "petal_length": 1.3, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.0, "sepal_width": 3.5, "petal_length": 1.6, "petal_width": 0.6, "species": "Iris-setosa"}
{"sepal_length": 5.1, "sepal_width": 3.8, "petal_length": 1.9, "petal_width": 0.4, "species": "Iris-setosa"}
{"sepal_length": 4.8, "sepal_width": 3.0, "petal_length": 1.4, "petal_width": 0.3, "species": "Iris-setosa"}
{"sepal_length": 5.1, "sepal_width": 3.8, "petal_length": 1.6, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 4.6, "sepal_width": 3.2, "petal_length": 1.4, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.3, "sepal_width": 3.7, "petal_length": 1.5, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 5.0, "sepal_width": 3.3, "petal_length": 1.4, "petal_width": 0.2, "species": "Iris-setosa"}
{"sepal_length": 7.0, "sepal_width": 3.2, "petal_length": 4.7, "petal_width": 1.4, "species": "Iris-versicolor"}
{"sepal_length": 6.4, "sepal_width": 3.2, "petal_length": 4.5, "petal_width": 1.5, "species": "Iris-versicolor"}
{"sepal_length": 6.9, "sepal_width": 3.1, "petal_length": 4.9, "petal_width": 1.5, "species": "Iris-versicolor"}
{"sepal_length": 5.5, "sepal_width": 2.3, "petal_length": 4.0, "petal_width": 1.3, "species": "Iris-versicolor"}
{"sepal_length": 6.5, "sepal_width": 2.8, "petal_length": 4.6, "petal_width": 1.5, "species": "Iris-versicolor"}
{"sepal_length": 5.7, "sepal_width": 2.8, "petal_length": 4.5, "petal_width": 1.3, "species": "Iris-versicolor"}
{"sepal_length": 6.3, "sepal_width": 3.3, "petal_length": 4.7, "petal_width": 1.6, "species": "Iris-versicolor"}
{"sepal_length": 4.9, "sepal_width": 2.4, "petal_length": 3.3, "petal_width": 1.0, "species": "Iris-versicolor"}
{"sepal_length": 6.6, "sepal_width": 2.9, "petal_length": 4.6, "petal_width": 1.3, "species": "Iris-versicolor"}
```

2. The subscriber_json.ipynb contains the code for subscribing to the topic iris-data using Spark Structured streaming and connecting to Kafka service. The output for the dataframe created using the raw input reading from the topic is presented below.

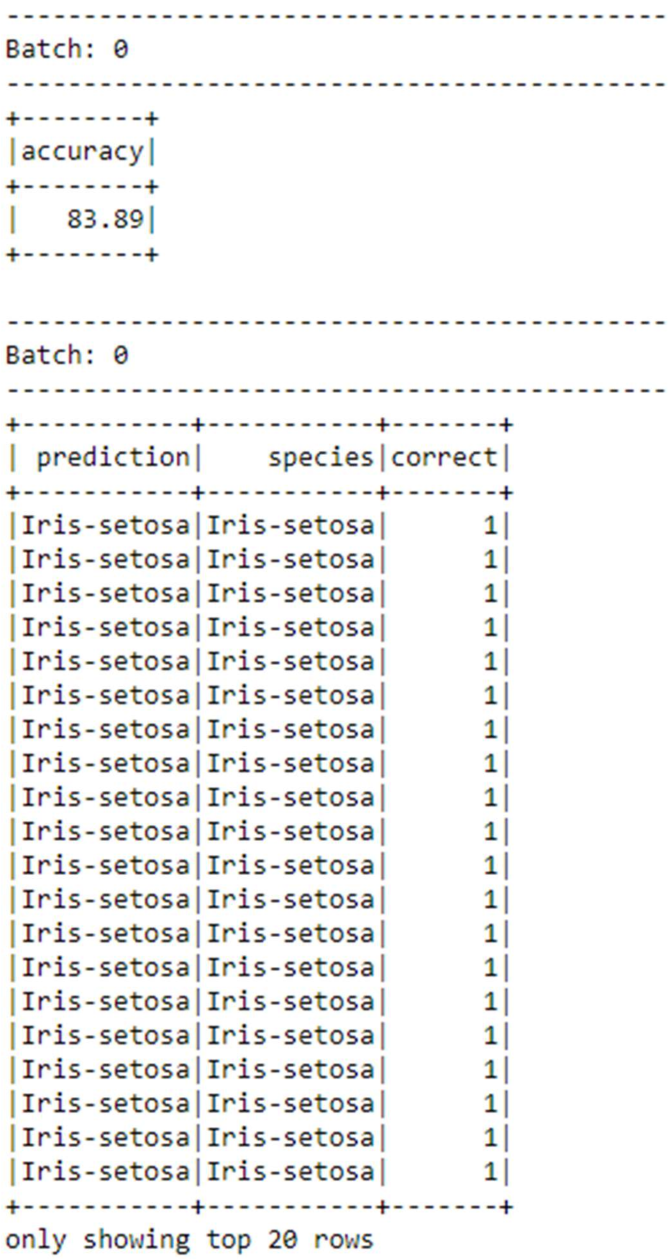
Lab 7 | ME18B059 | Madhur Jindal



```
-------------------------------------------
Batch: 0
-------------------------------------------
+------------+-----------+------------+-----------+-----------+
|sepal_length|sepal_width|petal_length|petal_width|    species|
+------------+-----------+------------+-----------+-----------+
|         4.9|        3.0|         1.4|        0.2|Iris-setosa|
|         4.7|        3.2|         1.3|        0.2|Iris-setosa|
|         4.6|        3.1|         1.5|        0.2|Iris-setosa|
|         5.0|        3.6|         1.4|        0.2|Iris-setosa|
|         5.4|        3.9|         1.7|        0.4|Iris-setosa|
|         4.6|        3.4|         1.4|        0.3|Iris-setosa|
|         5.0|        3.4|         1.5|        0.2|Iris-setosa|
|         4.4|        2.9|         1.4|        0.2|Iris-setosa|
|         4.9|        3.1|         1.5|        0.1|Iris-setosa|
|         5.4|        3.7|         1.5|        0.2|Iris-setosa|
|         4.8|        3.4|         1.6|        0.2|Iris-setosa|
|         4.8|        3.0|         1.4|        0.1|Iris-setosa|
|         4.3|        3.0|         1.1|        0.1|Iris-setosa|
|         5.8|        4.0|         1.2|        0.2|Iris-setosa|
|         5.7|        4.4|         1.5|        0.4|Iris-setosa|
|         5.4|        3.9|         1.3|        0.4|Iris-setosa|
|         5.1|        3.5|         1.4|        0.3|Iris-setosa|
|         5.7|        3.8|         1.7|        0.3|Iris-setosa|
|         5.1|        3.8|         1.5|        0.3|Iris-setosa|
|         5.4|        3.4|         1.7|        0.2|Iris-setosa|
+------------+-----------+------------+-----------+-----------+
only showing top 20 rows
```

The dataframe is then transformed using the pipeline model, which gives us the prediction column which is used for calculating if the answer is correct (correct column) and then finally averaged to get the accuracy value. The screenshot depicts the above. A pdf file subscriber.pdf is attached which contains the pdf conversion of the ipynb file with the output.

Lab 7 | ME18B059 | Madhur Jindal

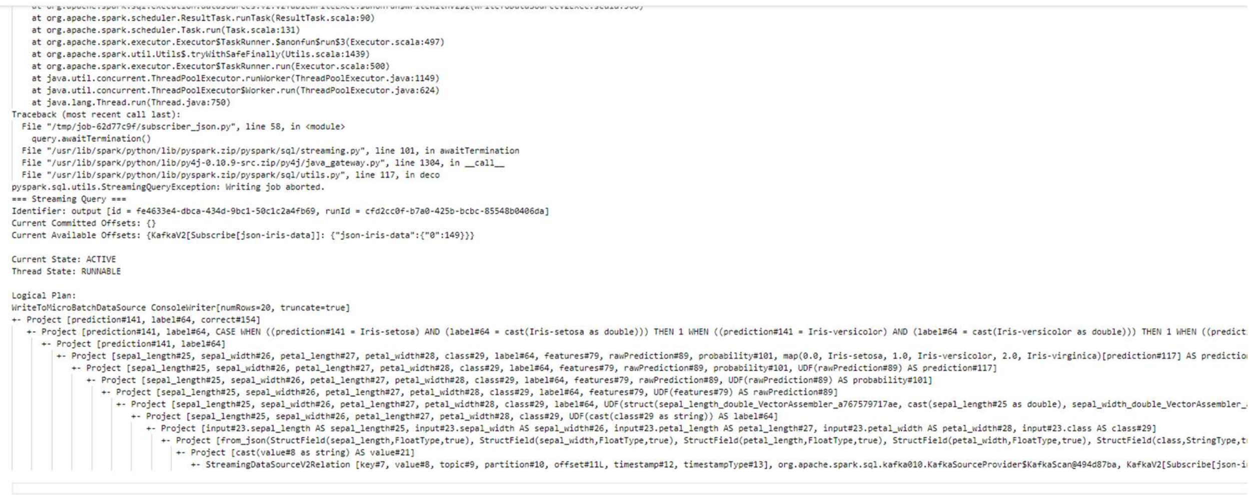


```
-------------------------------------------
Batch: 0
-------------------------------------------
+--------+
|accuracy|
+--------+
|   83.89|
+--------+

-------------------------------------------
Batch: 0
-------------------------------------------
+-----------+-----------+-------+
| prediction|    species|correct|
+-----------+-----------+-------+
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
|Iris-setosa|Iris-setosa|      1|
+-----------+-----------+-------+
only showing top 20 rows
```

P.s. this is performed in the jupyter notebook of the cluster because there was an error while submitting a dataproc pyspark job which is given below. This could be due to faulty jar file, not satisfying the dependencies which has been taken care of in the jupyter notebook. Below is the screenshot for the same.

Lab 7 | ME18B059 | Madhur Jindal

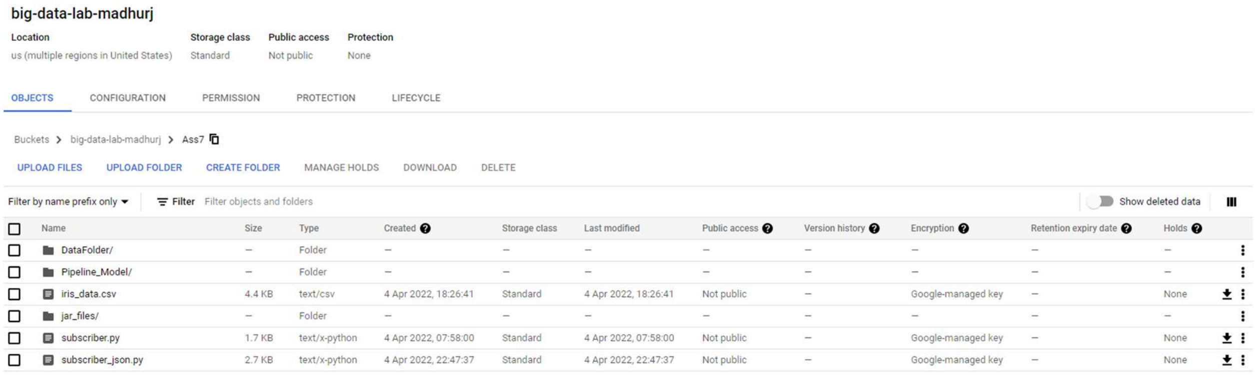


```
    at org.apache.spark.scheduler.ResultTask.runTask(ResultTask.scala:90)
    at org.apache.spark.scheduler.Task.run(Task.scala:131)
    at org.apache.spark.executor.Executor$TaskRunner.$anonfun$run$3(Executor.scala:497)
    at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1439)
    at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:500)
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
    at java.lang.Thread.run(Thread.java:750)
Traceback (most recent call last):
  File "/tmp/job-62d77c9f/subscriber_json.py", line 58, in <module>
    query.awaitTermination()
  File "/usr/lib/spark/python/lib/pyspark.zip/pyspark/sql/streaming.py", line 101, in awaitTermination
  File "/usr/lib/spark/python/lib/py4j-0.10.9-src.zip/py4j/java_gateway.py", line 1304, in __call__
  File "/usr/lib/spark/python/lib/pyspark.zip/pyspark/sql/utils.py", line 117, in deco
pyspark.sql.utils.StreamingQueryException: Writing job aborted.
=== Streaming Query ===
Identifier: output [id = fe4633e4-dbca-434d-9bc1-50c1c2a4fb69, runId = cfd2cc0f-b7a0-425b-bcbc-85548b0406da]
Current Committed Offsets: {}
Current Available Offsets: {KafkaV2[Subscribe[json-iris-data]]: {"json-iris-data":{"0":149}}}

Current State: ACTIVE
Thread State: RUNNABLE

Logical Plan:
WriteToMicroBatchDataSource ConsoleWriter[numRows=20, truncate=true]
+- Project [prediction#141, label#64, correct#154]
   +- Project [prediction#141, label#64, CASE WHEN ((prediction#141 = Iris-setosa) AND (label#64 = cast(Iris-setosa as double))) THEN 1 WHEN ((prediction#141 = Iris-versicolor) AND (label#64 = cast(Iris-versicolor as double))) THEN 1 WHEN ((predict
      +- Project [prediction#141, label#64]
         +- Project [sepal_length#25, sepal_width#26, petal_length#27, petal_width#28, class#29, label#64, features#79, rawPrediction#89, probability#101, map(0.0, Iris-setosa, 1.0, Iris-versicolor, 2.0, Iris-virginica)[prediction#117] AS predictio
            +- Project [sepal_length#25, sepal_width#26, petal_length#27, petal_width#28, class#29, label#64, features#79, rawPrediction#89, probability#101, UDF(rawPrediction#89) AS prediction#117]
               +- Project [sepal_length#25, sepal_width#26, petal_length#27, petal_width#28, class#29, label#64, features#79, rawPrediction#89, UDF(rawPrediction#89) AS probability#101]
                  +- Project [sepal_length#25, sepal_width#26, petal_length#27, petal_width#28, class#29, label#64, features#79, UDF(features#79) AS rawPrediction#89]
                     +- Project [sepal_length#25, sepal_width#26, petal_length#27, petal_width#28, class#29, label#64, UDF(struct(sepal_length_double_VectorAssembler_a767579717ae, cast(sepal_length#25 as double), sepal_width_double_VectorAssembler_
                        +- Project [sepal_length#25, sepal_width#26, petal_length#27, petal_width#28, class#29, UDF(cast(class#29 as string)) AS label#64]
                           +- Project [input#23.sepal_length AS sepal_length#25, input#23.sepal_width AS sepal_width#26, input#23.petal_length AS petal_length#27, input#23.petal_width AS petal_width#28, input#23.class AS class#29]
                              +- Project [from_json(StructField(sepal_length,FloatType,true), StructField(sepal_width,FloatType,true), StructField(petal_length,FloatType,true), StructField(petal_width,FloatType,true), StructField(class,StringType,t
                                 +- Project [cast(value#8 as string) AS value#21]
                                    +- StreamingDataSourceV2Relation [key#7, value#8, topic#9, partition#10, offset#11L, timestamp#12, timestampType#13], org.apache.spark.sql.kafka010.KafkaSourceProvider$KafkaScan@494d87ba, KafkaV2[Subscribe[json-i
```

3. The code for creating the Pipeline model is present in the pipeline_model.py



☐	Name	Size	Type	Created	Storage class	Last modified	Public access	Version history	Encryption	Retention expiry date	Holds
☐	DataFolder/	–	Folder	–	–	–	–	–	–	–	–
☐	Pipeline_Model/	–	Folder	–	–	–	–	–	–	–	–
☐	iris_data.csv	4.4 KB	text/csv	4 Apr 2022, 18:26:41	Standard	4 Apr 2022, 18:26:41	Not public	–	Google-managed key	–	None
☐	jar_files/	–	Folder	–	–	–	–	–	–	–	–
☐	subscriber.py	1.7 KB	text/x-python	4 Apr 2022, 07:58:00	Standard	4 Apr 2022, 07:58:00	Not public	–	Google-managed key	–	None
☐	subscriber_json.py	2.7 KB	text/x-python	4 Apr 2022, 22:47:37	Standard	4 Apr 2022, 22:47:37	Not public	–	Google-managed key	–	None

The screenshot for the bucket.