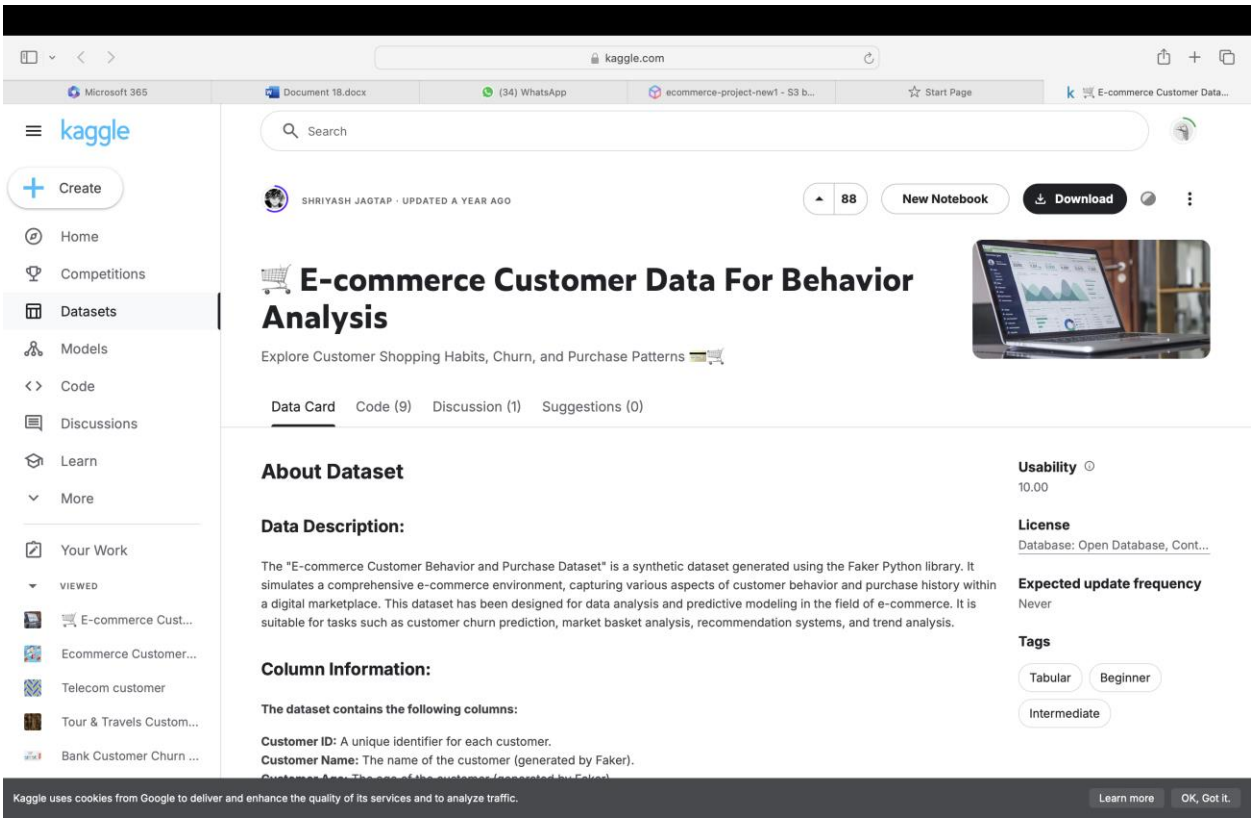
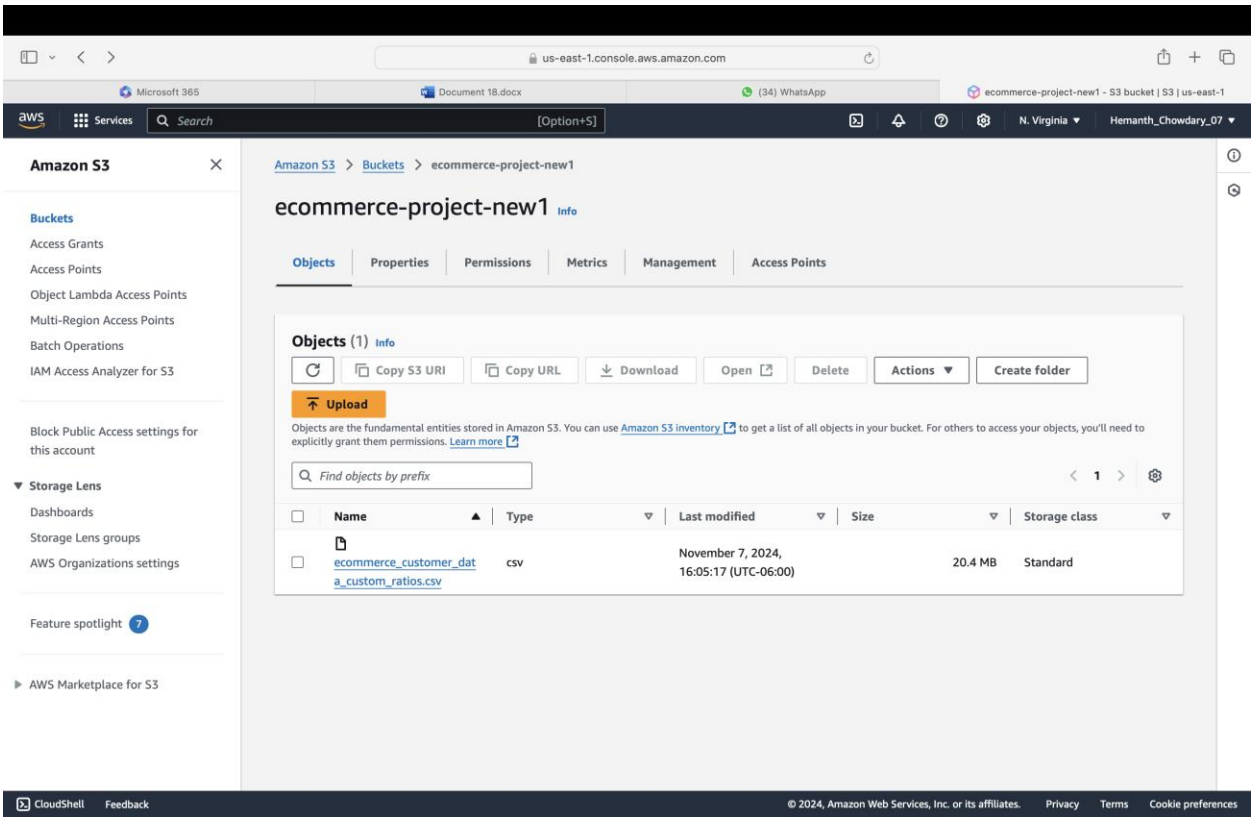
STEPS:

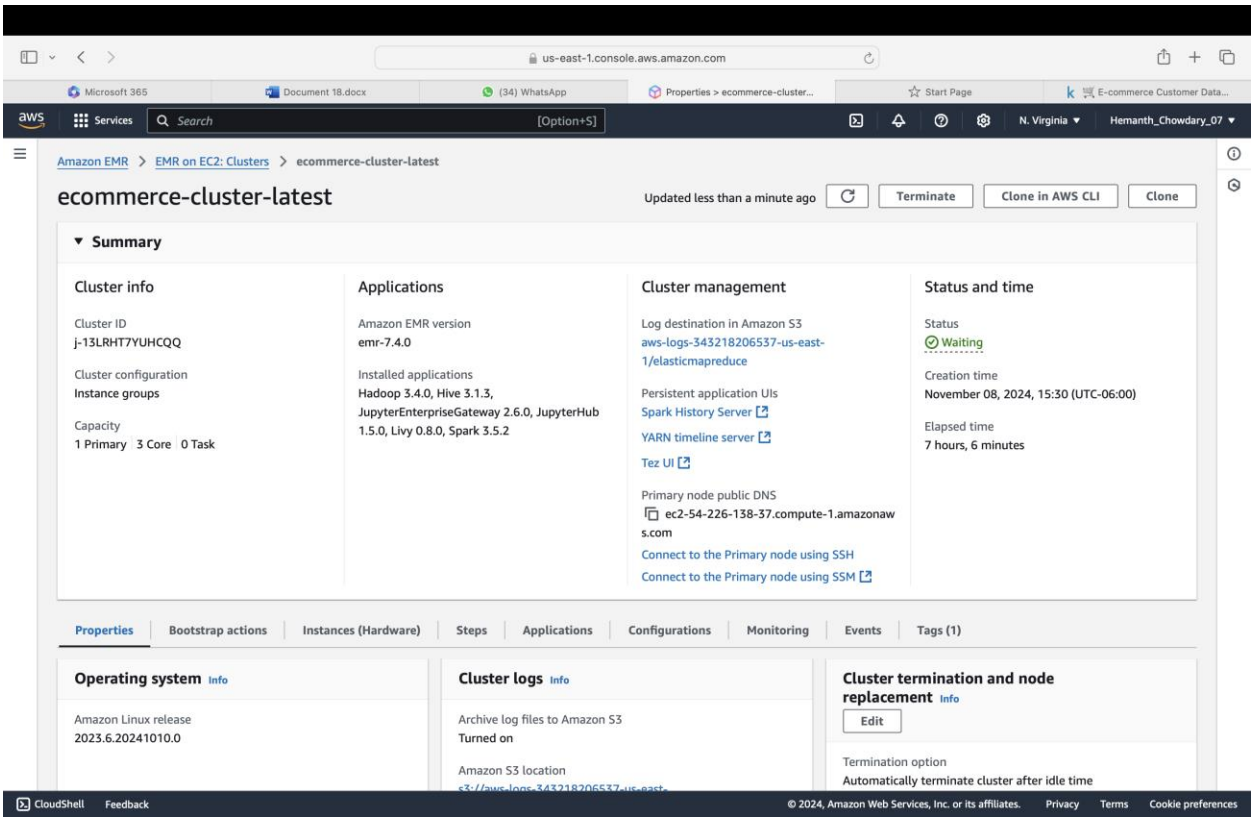
1. First we will download the data set from Kaggle and then we will store it on our system.



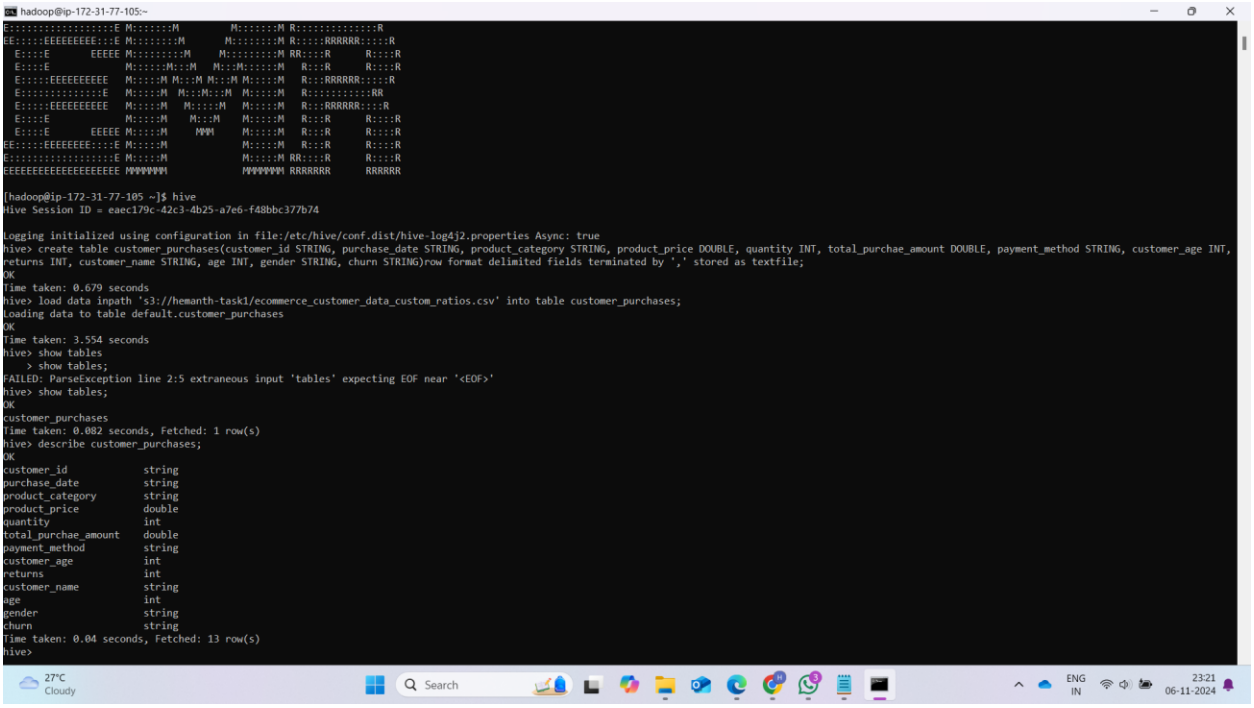
2. Then we will login into our AWS account and then create S3 bucket with project name and upload our data set init.



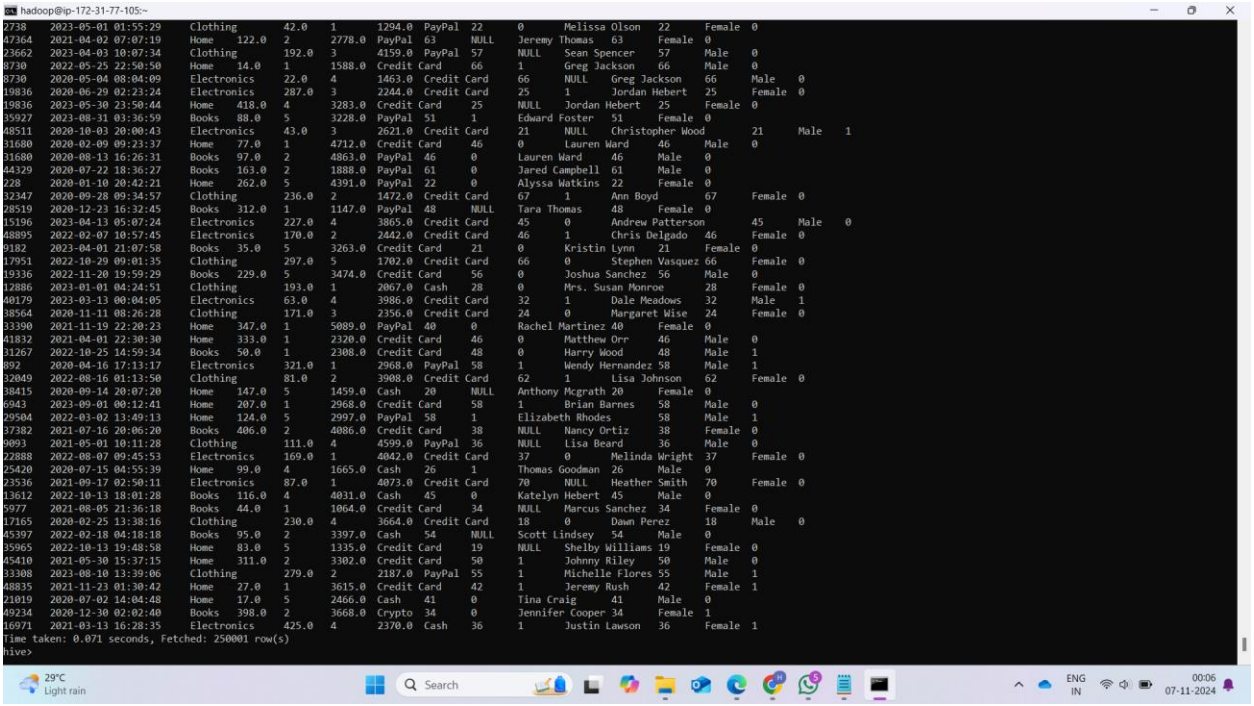
3. Then we will create a Cluster in EMR with all the permissions, key pairs and policies and run the cluster.



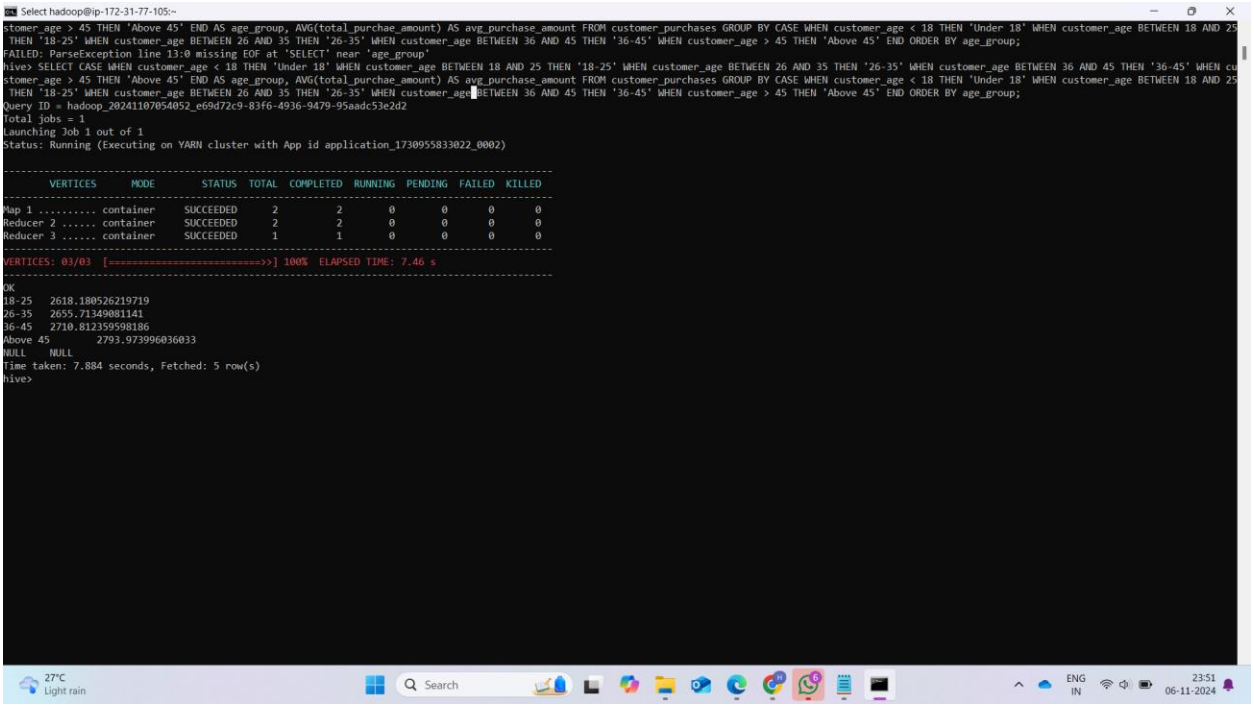
4. Then we will Copy the DNS and we will run EMR in our terminal using the DNS id and Keypair we used in EMR cluster.
5. Then we will start Hive in EMR.
6. Then we will Load the data set which is in S3 bucket by giving its path and create a table with that data.
7. Then we ill analyze the data in HIVE by running some queriues to know the properties and structure of our data.

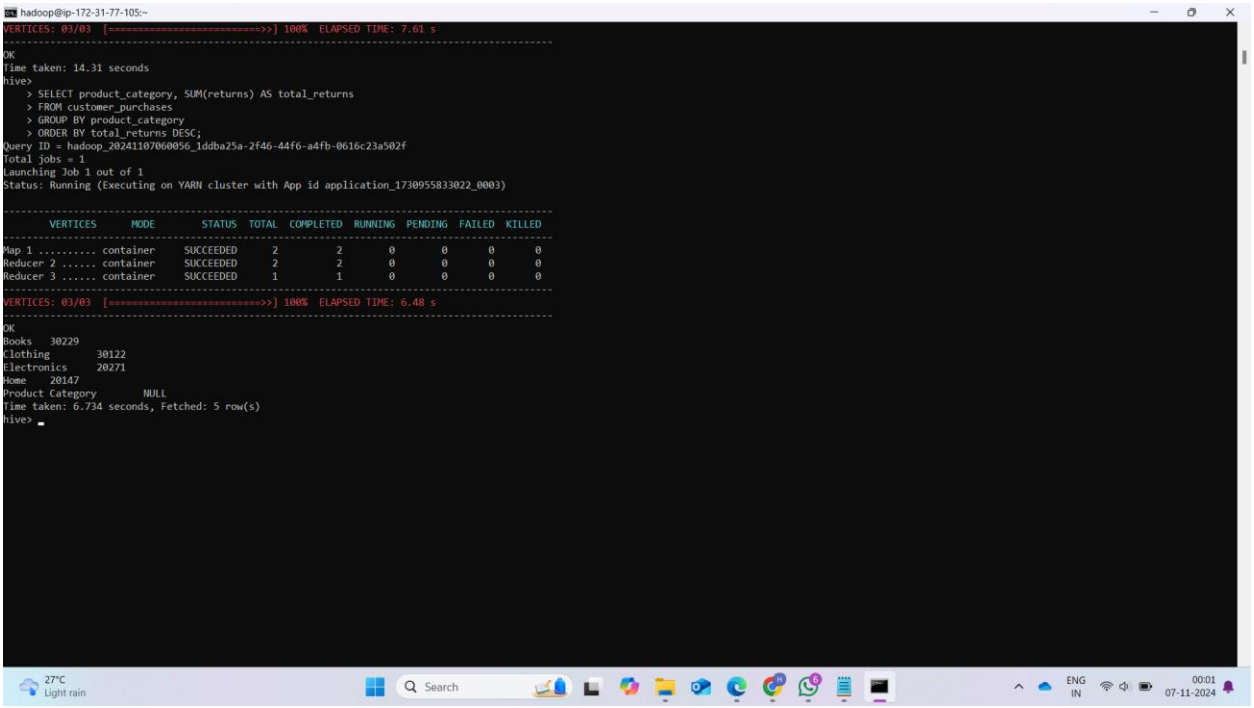


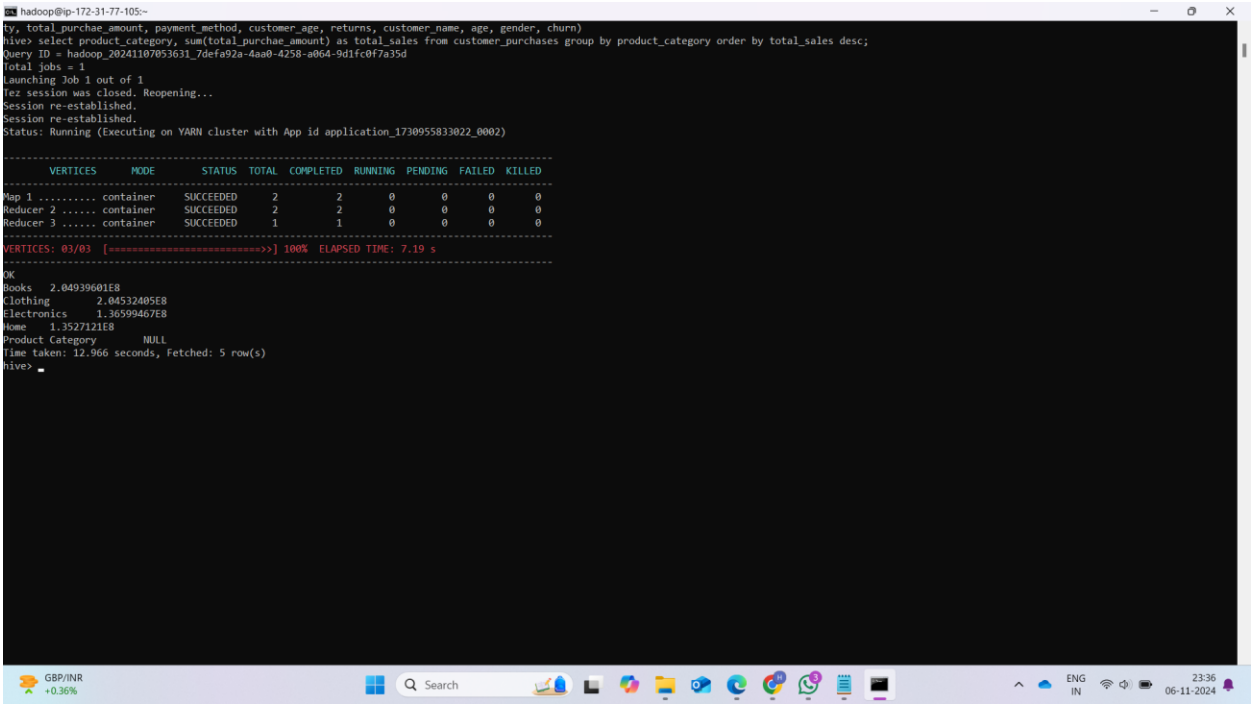
```
hadoop@ip-172-31-77-105:~
E::::::::::::::::::E M::::::M         M::::::M R::::::::::::::R
EE::::::EEEEEEEEE:::E M:::::::M       M:::::::M R:::::RRRRRR:::::R
  E:::::E       EEEEE M::::::::M     M::::::::M RR:::::R     R:::::R
  E:::::E             M:::::::M::M   M::M:::::::M   R::::R     R:::::R
  E::::::EEEEEEEEEE   M:::::M M:::M M:::M M:::::M   R::::RRRRRR:::::R
  E:::::::::::::::E   M:::::M  M:::M:::M  M:::::M   R:::::::::::RR
  E::::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R::::RRRRRR:::::R
  E:::::E             M:::::M    M:::M    M:::::M   R::::R     R:::::R
  E:::::E       EEEEE M:::::M     MMM     M:::::M   R::::R     R:::::R
EE::::::EEEEEEEE:::E M:::::M             M:::::M   R::::R     R:::::R
E::::::::::::::::::E M:::::M             M:::::M RR::::R     R:::::R
EEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR     RRRRRR

[hadoop@ip-172-31-77-105 ~]$ hive
Hive Session ID = eaec179c-42c3-4b25-a7e6-f48bbc377b74

Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j2.properties Async: true
hive> create table customer_purchases(customer_id STRING, purchase_date STRING, product_category STRING, product_price DOUBLE, quantity INT, total_purchae_amount DOUBLE, payment_method STRING, customer_age INT,
returns INT, customer_name STRING, age INT, gender STRING, churn STRING)row format delimited fields terminated by ',' stored as textfile;
OK
Time taken: 0.679 seconds
hive> load data inpath 's3://hemanth-task1/ecommerce_customer_data_custom_ratios.csv' into table customer_purchases;
Loading data to table default.customer_purchases
OK
Time taken: 3.554 seconds
hive> show tables
    > show tables;
FAILED: ParseException line 2:5 extraneous input 'tables' expecting EOF near '<EOF>'
hive> show tables;
OK
customer_purchases
Time taken: 0.082 seconds, Fetched: 1 row(s)
hive> describe customer_purchases;
OK
customer_id             string
purchase_date           string
product_category        string
product_price           double
quantity                int
total_purchae_amount    double
payment_method          string
customer_age            int
returns                 int
customer_name           string
age                     int
gender                  string
churn                   string
Time taken: 0.04 seconds, Fetched: 13 row(s)
hive>
```



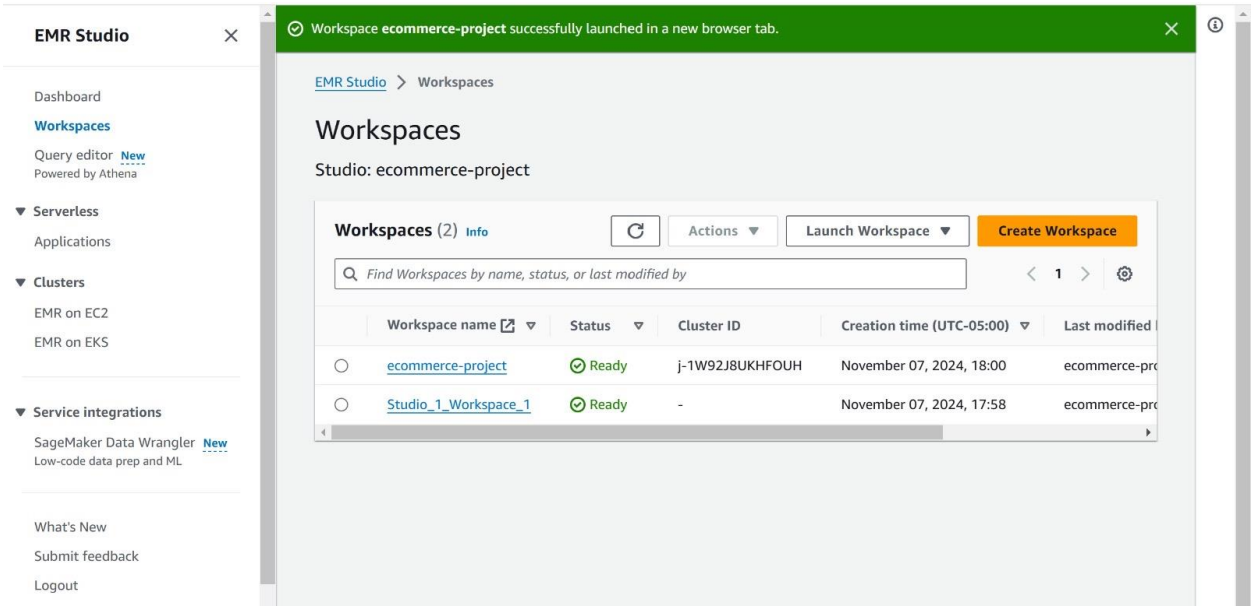
```
hadoop@ip-172-31-77-105:~
2738    2023-05-01 01:55:29     Clothing        42.0    1       1294.0  PayPal  22      0       Melissa Olson   22      Female  0
47364   2021-04-02 07:07:19     Home    122.0   2       2778.0  PayPal  63      NULL    Jeremy Thomas   63      Female  0
23662   2023-04-03 10:07:34     Clothing        192.0   3       4159.0  PayPal  57      NULL    Sean Spencer    57      Male    0
8730    2022-05-25 22:50:50     Home    14.0    1       1588.0  Credit Card     66      1       Greg Jackson    66      Male    0
8730    2020-05-04 08:04:09     Electronics     22.0    4       1463.0  Credit Card     66      NULL    Greg Jackson    66      Male    0
19836   2020-06-29 02:23:24     Electronics     287.0   3       2244.0  Credit Card     25      1       Jordan Hebert   25      Female  0
19836   2023-05-30 23:50:44     Home    418.0   4       3283.0  Credit Card     25      NULL    Jordan Hebert   25      Female  0
35927   2023-08-31 03:36:59     Books   88.0    5       3228.0  PayPal  51      1       Edward Foster   51      Female  0
48511   2020-10-03 20:00:43     Electronics     43.0    3       2621.0  Credit Card     21      NULL    Christopher Wood        21      Male    1
31680   2020-02-09 09:23:37     Home    77.0    1       4712.0  Credit Card     46      0       Lauren Ward     46      Male    0
31680   2020-08-13 16:26:31     Books   97.0    2       4863.0  PayPal  46      0       Lauren Ward     46      Male    0
44329   2020-07-22 18:36:27     Books   163.0   2       1888.0  PayPal  61      0       Jared Campbell  61      Male    0
228     2020-01-10 20:42:21     Home    262.0   5       4391.0  PayPal  22      0       Alyssa Watkins  22      Female  0
32347   2020-09-28 09:34:57     Clothing        236.0   2       1472.0  Credit Card     67      1       Ann Boyd        67      Female  0
28519   2020-12-23 16:32:45     Books   312.0   1       1147.0  PayPal  48      NULL    Tara Thomas     48      Female  0
15196   2023-04-13 05:07:24     Electronics     227.0   4       3865.0  Credit Card     45      0       Andrew Patterson        45      Male    0
48895   2022-02-07 10:57:45     Electronics     170.0   2       2442.0  Credit Card     46      1       Chris Delgado   46      Female  0
9182    2023-04-01 21:07:58     Books   35.0    5       3263.0  Credit Card     21      0       Kristin Lynn    21      Female  0
17951   2022-10-29 09:01:35     Clothing        297.0   5       1702.0  Credit Card     66      0       Stephen Vasquez 66      Female  0
19336   2022-11-20 19:59:29     Books   229.0   5       3474.0  Credit Card     56      0       Joshua Sanchez  56      Male    0
12886   2023-01-01 04:24:51     Clothing        193.0   1       2067.0  Cash    28      0       Mrs. Susan Monroe       28      Female  0
40179   2023-03-13 00:04:05     Electronics     63.0    4       3986.0  Credit Card     32      1       Dale Meadows    32      Male    1
38564   2020-11-11 08:26:28     Clothing        171.0   3       2356.0  Credit Card     24      0       Margaret Wise   24      Female  0
33390   2021-11-19 22:20:23     Home    347.0   1       5089.0  PayPal  40      0       Rachel Martinez 40      Female  0
41832   2021-04-01 22:30:30     Home    133.0   1       2320.0  Credit Card     46      0       Matthew Orr     46      Male    0
31267   2022-10-25 14:59:34     Books   50.0    1       2308.0  Credit Card     48      0       Harry Wood      48      Male    1
892     2020-04-16 17:13:17     Electronics     321.0   1       2968.0  PayPal  58      1       Wendy Hernandez 58      Male    1
32049   2022-08-16 01:13:50     Clothing        81.0    2       3908.0  Credit Card     62      1       Lisa Johnson    62      Female  0
38415   2020-09-14 20:07:20     Home    147.0   5       1459.0  Cash    20      NULL    Anthony Mcgrath 20      Female  0
6943    2023-09-01 00:12:41     Home    207.0   1       2968.0  Credit Card     58      1       Brian Barnes    58      Male    0
29504   2022-03-02 13:49:13     Home    124.0   5       2997.0  PayPal  58      1       Elizabeth Rhodes        58      Male    1
37382   2021-07-16 20:06:20     Books   406.0   2       4086.0  Credit Card     38      NULL    Nancy Ortiz     38      Female  0
9093    2021-05-01 10:11:28     Clothing        111.0   4       4599.0  PayPal  36      NULL    Lisa Beard      36      Male    0
22888   2022-08-07 09:45:53     Electronics     169.0   1       4042.0  Credit Card     37      0       Melinda Wright  37      Female  0
25420   2020-07-15 04:55:39     Home    99.0    4       1665.0  Cash    26      1       Thomas Goodman  26      Male    0
23536   2021-09-17 02:50:11     Electronics     87.0    1       4073.0  Credit Card     70      NULL    Heather Smith   70      Female  0
13612   2022-10-13 18:01:28     Books   116.0   4       4031.0  Cash    45      0       Katelyn Hebert  45      Male    0
5977    2021-08-05 21:36:18     Books   44.0    1       1064.0  Credit Card     34      NULL    Marcus Sanchez  34      Female  0
17165   2020-02-25 13:38:16     Clothing        230.0   4       3664.0  Credit Card     18      0       Dawn Perez      18      Male    0
45397   2022-02-18 04:18:18     Books   95.0    2       3397.0  Cash    54      NULL    Scott Lindsey   54      Male    0
35965   2022-10-13 19:48:58     Home    83.0    5       1335.0  Credit Card     19      NULL    Shelby Williams 19      Female  0
45410   2021-05-30 15:37:15     Home    311.0   2       3302.0  Credit Card     50      1       Johnny Riley    50      Male    0
33308   2023-08-10 13:39:06     Clothing        279.0   2       2187.0  PayPal  55      1       Michelle Flores 55      Male    1
48835   2021-11-23 01:30:42     Home    27.0    1       3615.0  Credit Card     42      1       Jeremy Rush     42      Female  1
21019   2020-07-02 14:04:48     Home    17.0    5       2466.0  Cash    41      0       Tina Craig      41      Male    0
49234   2020-12-30 02:02:40     Books   398.0   2       3668.0  Crypto  34      0       Jennifer Cooper 34      Female  1
16971   2021-03-13 16:28:35     Electronics     425.0   4       2370.0  Cash    36      1       Justin Lawson   36      Female  1
Time taken: 0.071 seconds, Fetched: 250001 row(s)
hive>
```

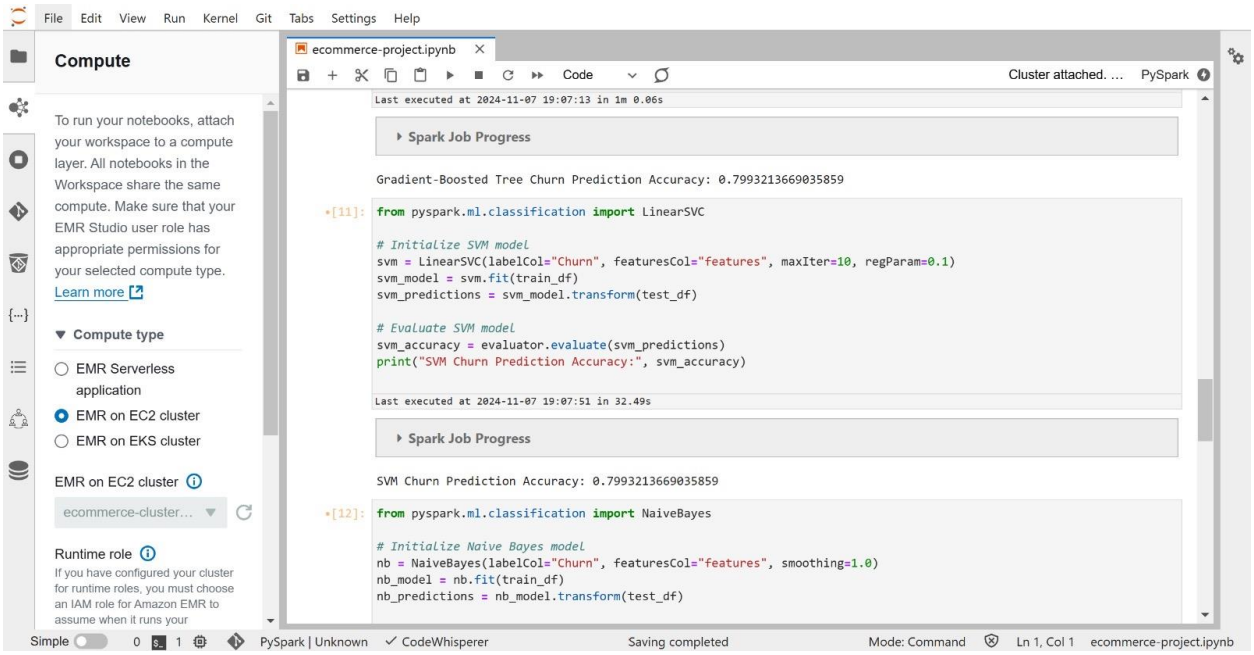

Select hadoop@ip-172-31-77-105:~
stomer_age > 45 THEN 'Above 45' END AS age_group, AVG(total_purchae_amount) AS avg_purchase_amount FROM customer_purchases GROUP BY CASE WHEN customer_age < 18 THEN 'Under 18' WHEN customer_age BETWEEN 18 AND 25
THEN '18-25' WHEN customer_age BETWEEN 26 AND 35 THEN '26-35' WHEN customer_age BETWEEN 36 AND 45 THEN '36-45' WHEN customer_age > 45 THEN 'Above 45' END ORDER BY age_group;
FAILED: ParseException line 13:0 missing EOF at 'SELECT' near 'age_group'
hive> SELECT CASE WHEN customer_age < 18 THEN 'Under 18' WHEN customer_age BETWEEN 18 AND 25 THEN '18-25' WHEN customer_age BETWEEN 26 AND 35 THEN '26-35' WHEN customer_age BETWEEN 36 AND 45 THEN '36-45' WHEN cu
stomer_age > 45 THEN 'Above 45' END AS age_group, AVG(total_purchae_amount) AS avg_purchase_amount FROM customer_purchases GROUP BY CASE WHEN customer_age < 18 THEN 'Under 18' WHEN customer_age BETWEEN 18 AND 25
THEN '18-25' WHEN customer_age BETWEEN 26 AND 35 THEN '26-35' WHEN customer_age BETWEEN 36 AND 45 THEN '36-45' WHEN customer_age > 45 THEN 'Above 45' END ORDER BY age_group;
Query ID = hadoop_20241107054052_e69d72c9-83f6-4936-9479-95aadc53e2d2
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1730955833022_0002)
VERTICES MODE STATUS TOTAL COMPLETED RUNNING PENDING FAILED KILLED
Map 1 ..... container SUCCEEDED 2 2 0 0 0 0
Reducer 2 ..... container SUCCEEDED 2 2 0 0 0 0
Reducer 3 ..... container SUCCEEDED 1 1 0 0 0 0
VERTICES: 03/03 [==========>>] 100% ELAPSED TIME: 7.46 s
OK
18-25 2618.180526219719
26-35 2655.71349081141
36-45 2710.81235598186
Above 45 2793.97399603603 3
NULL NULL
Time taken: 7.884 seconds, Fetched: 5 row(s)
hive>
27°C Light rain
Search
ENG IN
23:51 06-11-2024

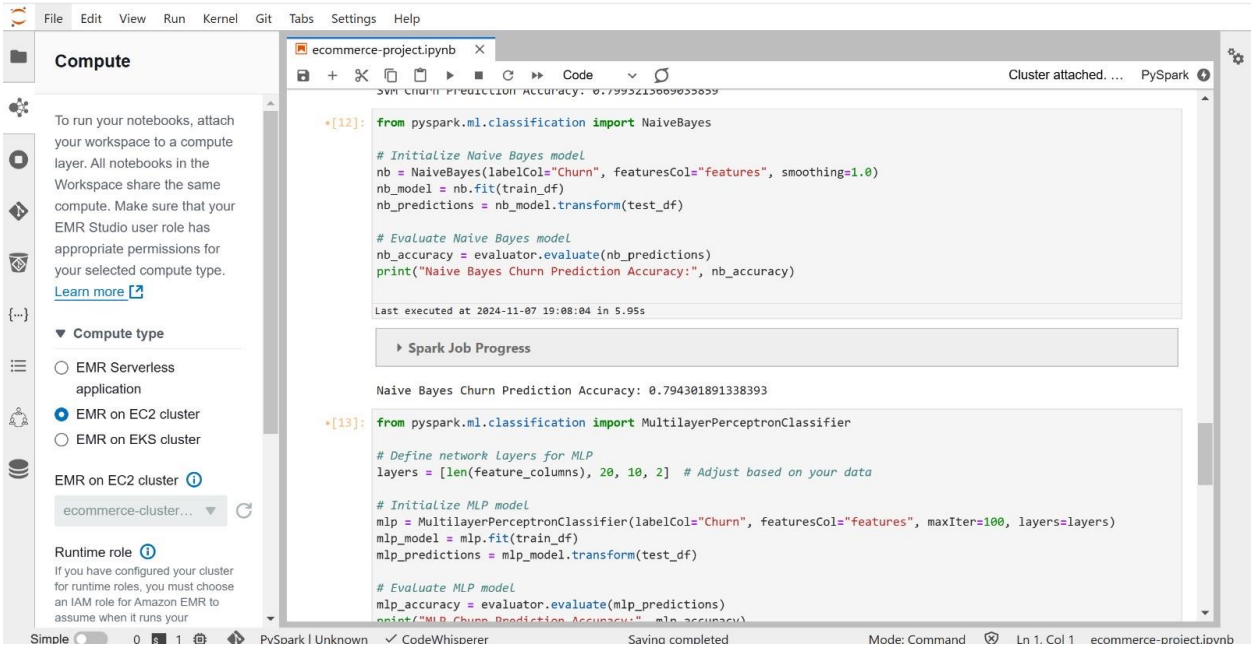

hadoop@ip-172-31-77-105:~
VERTICES: 03/03 [==========>>] 100% ELAPSED TIME: 7.61 s
OK
Time taken: 14.31 seconds
hive>
> SELECT product_category, SUM(returns) AS total_returns
> FROM customer_purchases
> GROUP BY product_category
> ORDER BY total_returns DESC;
Query ID = hadoop_20241107060056_1ddba25a-2f46-44f6-a4fb-0616c23a502f
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1730955833022_0003)
VERTICES MODE STATUS TOTAL COMPLETED RUNNING PENDING FAILED KILLED
Map 1 ..... container SUCCEEDED 2 2 0 0 0 0
Reducer 2 ..... container SUCCEEDED 2 2 0 0 0 0
Reducer 3 ..... container SUCCEEDED 1 1 0 0 0 0
VERTICES: 03/03 [==========>>] 100% ELAPSED TIME: 6.48 s
OK
Books 30229
Clothing 30122
Electronics 20271
Home 20147
Product Category NULL
Time taken: 6.734 seconds, Fetched: 5 row(s)
hive>
27°C Light rain
Search
ENG IN
00:01 07-11-2024

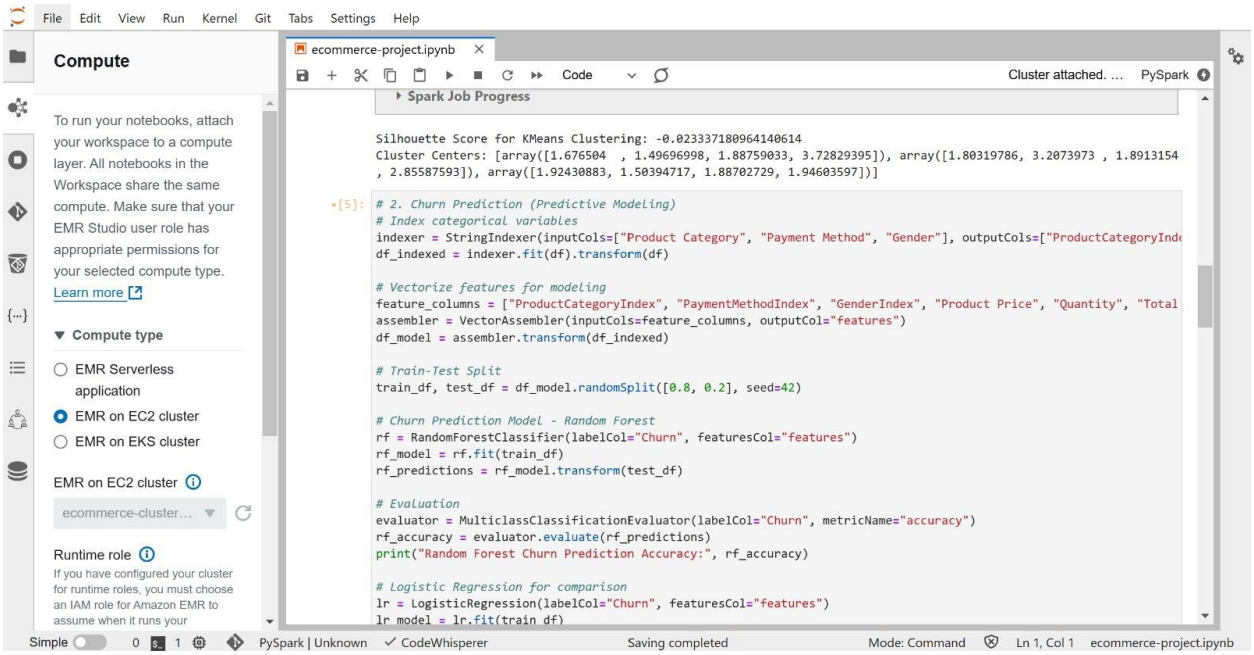


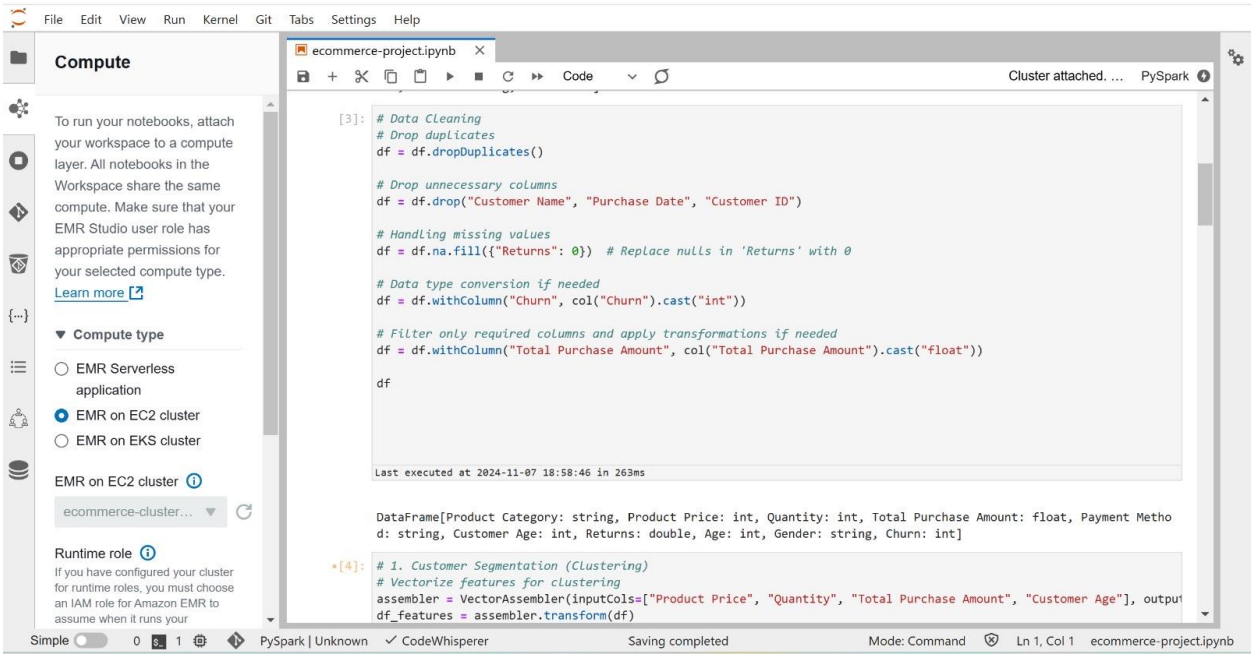
8. Now Data analysis is completed in HIVE. Then we will Preprocess the data and analyse, train and the test the data in Pyspark In aws using machine learning models.
9. So we will use the same bucket with our data set and run the cluster with required policies including Jupiter Notebook to run the Pyspark Code.
10. So we run the cluster and then we will go to studios and create a new workspace to run our pyspark quieries in jupiter notebook.
11. Then we will quick launch our workspace and Jupiter notebook will be opened and now we can create a notebook with ecommerce name and we will run our queries by selecting Pyspark as kernel.


EMR Studio
Workspace ecommerce-project successfully launched in a new browser tab.
Dashboard
Workspaces
Query editor New
Powered by Athena
Serverless
Applications
Clusters
EMR on EC2
EMR on EKS
Service integrations
SageMaker Data Wrangler New
Low-code data prep and ML
What's New
Submit feedback
Logout
EMR Studio > Workspaces
Workspaces
Studio: ecommerce-project
Workspaces (2) Info
Actions
Launch Workspace
Create Workspace
Find Workspaces by name, status, or last modified by
Workspace name
Status
Cluster ID
Creation time (UTC-05:00)
Last modified
ecommerce-project
Ready
j-1W92J8UKHFOUH
November 07, 2024, 18:00
Studio_1_Workspace_1
Ready
-
November 07, 2024, 17:58

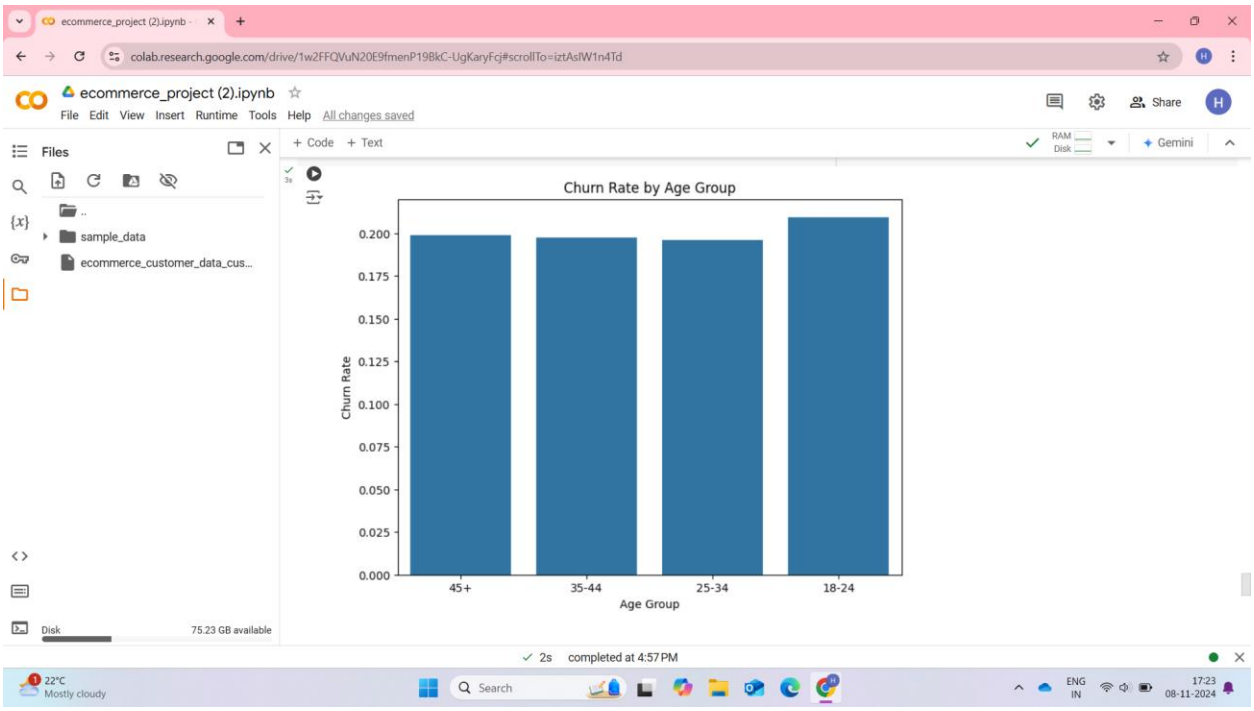

File Edit View Run Kernel Git Tabs Settings Help
Compute
To run your notebooks, attach your workspace to a compute layer. All notebooks in the Workspace share the same compute. Make sure that your EMR Studio user role has appropriate permissions for your selected compute type.
Learn more
Compute type
EMR Serverless application
EMR on EC2 cluster
EMR on EKS cluster
EMR on EC2 cluster
ecommerce-cluster...
Runtime role
If you have configured your cluster for runtime roles, you must choose an IAM role for Amazon EMR to assume when it runs your
ecommerce-project.ipynb
Code
Cluster attached. ...
PySpark
Last executed at 2024-11-07 19:07:13 in 1m 0.06s
Spark Job Progress
Gradient-Boosted Tree Churn Prediction Accuracy: 0.7993213669035859
[11]:
from pyspark.ml.classification import LinearSVC
# Initialize SVM model
svm = LinearSVC(labelCol="Churn", featuresCol="features", maxIter=10, regParam=0.1)
svm_model = svm.fit(train_df)
svm_predictions = svm_model.transform(test_df)
# Evaluate SVM model
svm_accuracy = evaluator.evaluate(svm_predictions)
print("SVM Churn Prediction Accuracy:", svm_accuracy)
Last executed at 2024-11-07 19:07:51 in 32.49s
Spark Job Progress
SVM Churn Prediction Accuracy: 0.7993213669035859
[12]:
from pyspark.ml.classification import NaiveBayes
# Initialize Naive Bayes model
nb = NaiveBayes(labelCol="Churn", featuresCol="features", smoothing=1.0)
nb_model = nb.fit(train_df)
nb_predictions = nb_model.transform(test_df)
Simple
PySpark | Unknown
CodeWhisperer
Saving completed
Mode: Command
Ln 1, Col 1
ecommerce-project.ipynb

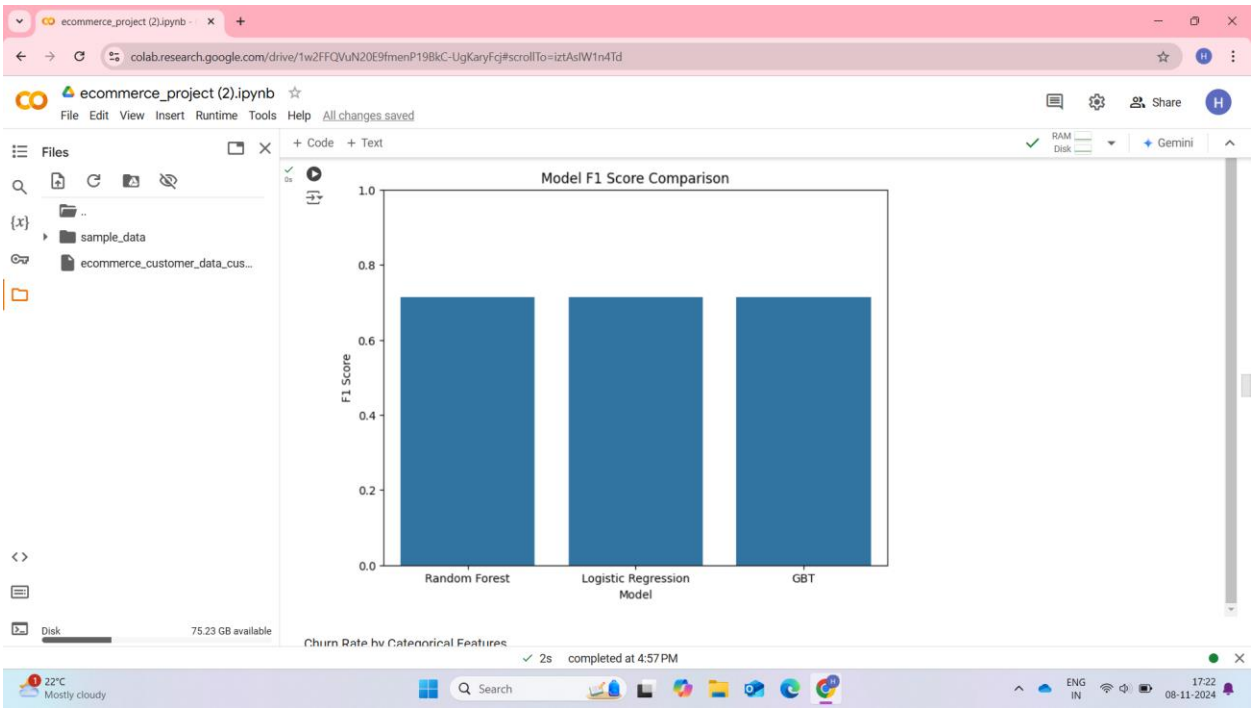

File Edit View Run Kernel Git Tabs Settings Help
Compute
To run your notebooks, attach your workspace to a compute layer. All notebooks in the Workspace share the same compute. Make sure that your EMR Studio user role has appropriate permissions for your selected compute type.
Learn more
Compute type
EMR Serverless application
EMR on EC2 cluster
EMR on EKS cluster
EMR on EC2 cluster
ecommerce-cluster...
Runtime role
If you have configured your cluster for runtime roles, you must choose an IAM role for Amazon EMR to assume when it runs your
ecommerce-project.ipynb
Code
Cluster attached. ...
PySpark
*[12]:
from pyspark.ml.classification import NaiveBayes
# Initialize Naive Bayes model
nb = NaiveBayes(labelCol="Churn", featuresCol="features", smoothing=1.0)
nb_model = nb.fit(train_df)
nb_predictions = nb_model.transform(test_df)
# Evaluate Naive Bayes model
nb_accuracy = evaluator.evaluate(nb_predictions)
print("Naive Bayes Churn Prediction Accuracy:", nb_accuracy)
Last executed at 2024-11-07 19:08:04 in 5.95s
Spark Job Progress
Naive Bayes Churn Prediction Accuracy: 0.794301891338393
*[13]:
from pyspark.ml.classification import MultilayerPerceptronClassifier
# Define network layers for MLP
layers = [len(feature_columns), 20, 10, 2] # Adjust based on your data
# Initialize MLP model
mlp = MultilayerPerceptronClassifier(labelCol="Churn", featuresCol="features", maxIter=100, layers=layers)
mlp_model = mlp.fit(train_df)
mlp_predictions = mlp_model.transform(test_df)
# Evaluate MLP model
mlp_accuracy = evaluator.evaluate(mlp_predictions)
Simple 0 1 PySpark | Unknown CodeWhisperer Saving completed Mode: Command Ln 1, Col 1 ecommerce-project.ipynb

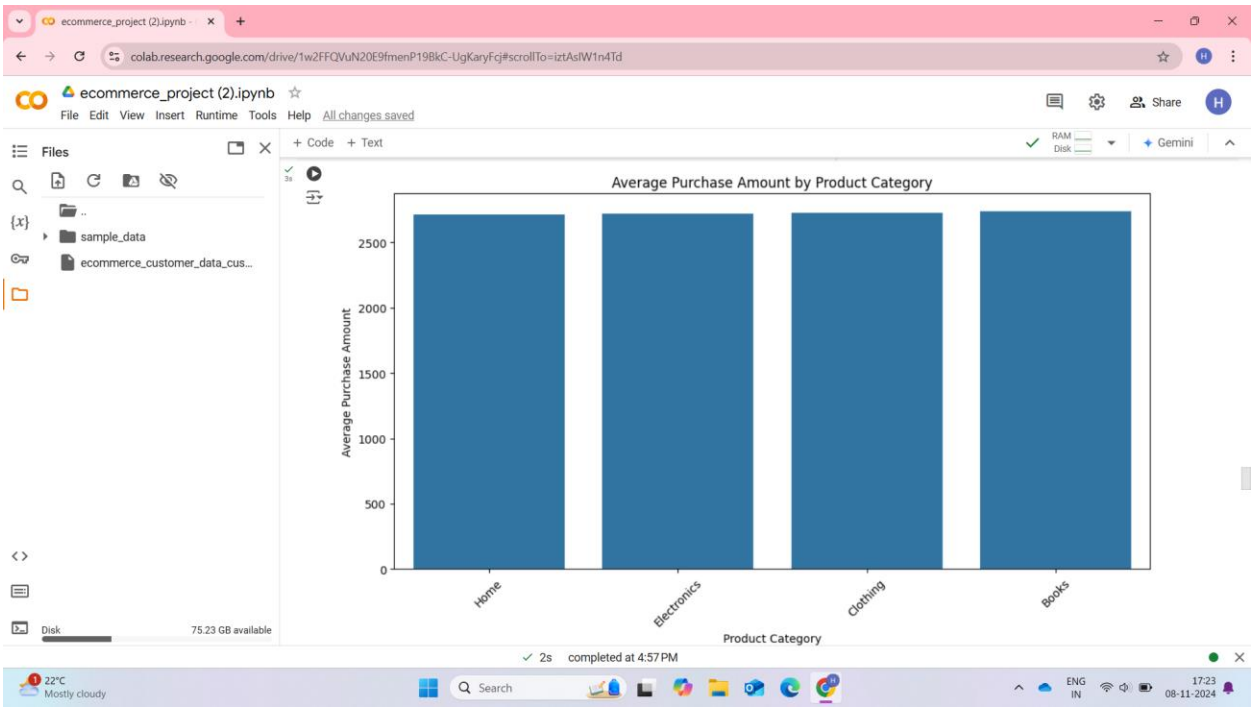

File Edit View Run Kernel Git Tabs Settings Help
Compute
To run your notebooks, attach your workspace to a compute layer. All notebooks in the Workspace share the same compute. Make sure that your EMR Studio user role has appropriate permissions for your selected compute type.
Learn more
Compute type
EMR Serverless application
EMR on EC2 cluster
EMR on EKS cluster
EMR on EC2 cluster
ecommerce-cluster...
Runtime role
If you have configured your cluster for runtime roles, you must choose an IAM role for Amazon EMR to assume when it runs your
ecommerce-project.ipynb
Code
Cluster attached. ...
PySpark
Spark Job Progress
Silhouette Score for KMeans Clustering: -0.023337180964140614
Cluster Centers: [array([1.676504 , 1.49696998, 1.88759033, 3.72829395]), array([1.80319786, 3.2073973 , 1.8913154
, 2.85587593]), array([1.92430883, 1.50394717, 1.88702729, 1.94603597])]
*[5]:
# 2. Churn Prediction (Predictive Modeling)
# Index categorical variables
indexer = StringIndexer(inputCols=["Product Category", "Payment Method", "Gender"], outputCols=["ProductCategoryInd
df_indexed = indexer.fit(df).transform(df)
# Vectorize features for modeling
feature_columns = ["ProductCategoryIndex", "PaymentMethodIndex", "GenderIndex", "Product Price", "Quantity", "Total
assembler = VectorAssembler(inputCols=feature_columns, outputCol="features")
df_model = assembler.transform(df_indexed)
# Train-Test Split
train_df, test_df = df_model.randomSplit([0.8, 0.2], seed=42)
# Churn Prediction Model - Random Forest
rf = RandomForestClassifier(labelCol="Churn", featuresCol="features")
rf_model = rf.fit(train_df)
rf_predictions = rf_model.transform(test_df)
# Evaluation
evaluator = MulticlassClassificationEvaluator(labelCol="Churn", metricName="accuracy")
rf_accuracy = evaluator.evaluate(rf_predictions)
print("Random Forest Churn Prediction Accuracy:", rf_accuracy)
# Logistic Regression for comparison
lr = LogisticRegression(labelCol="Churn", featuresCol="features")
lr_model = lr.fit(train_df)
Simple 0 1 PySpark | Unknown CodeWhisperer Saving completed Mode: Command Ln 1, Col 1 ecommerce-project.ipynb

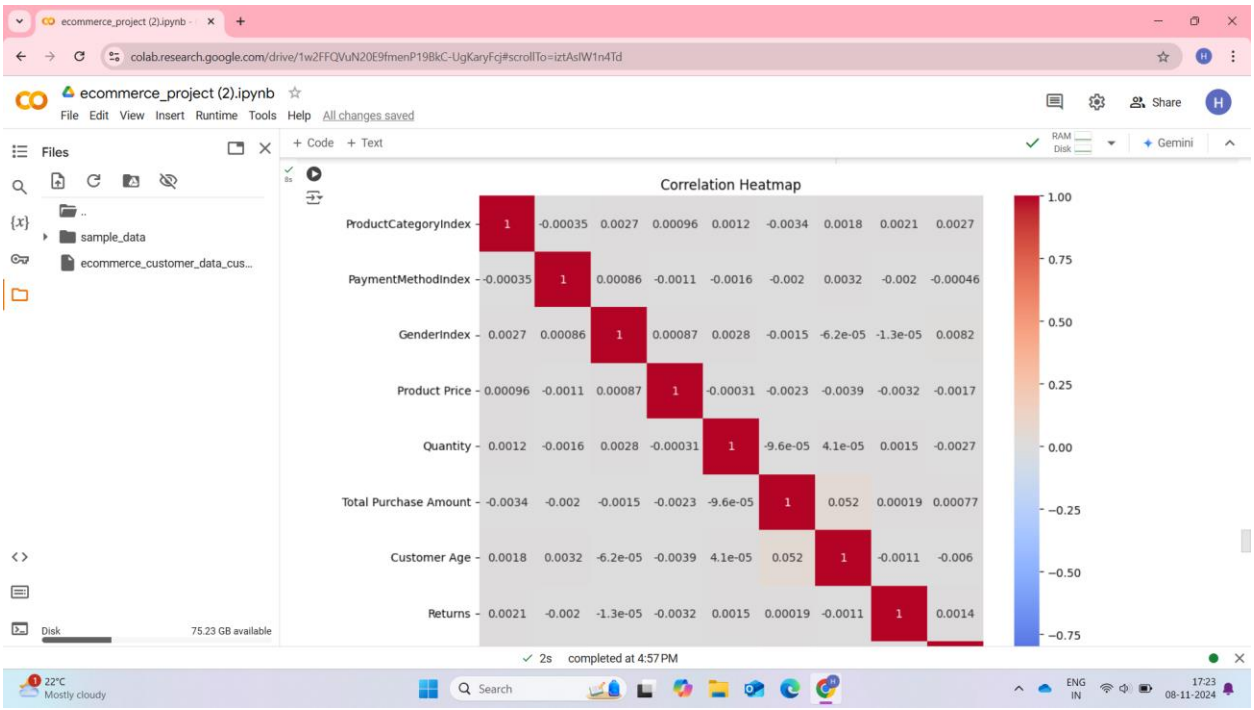


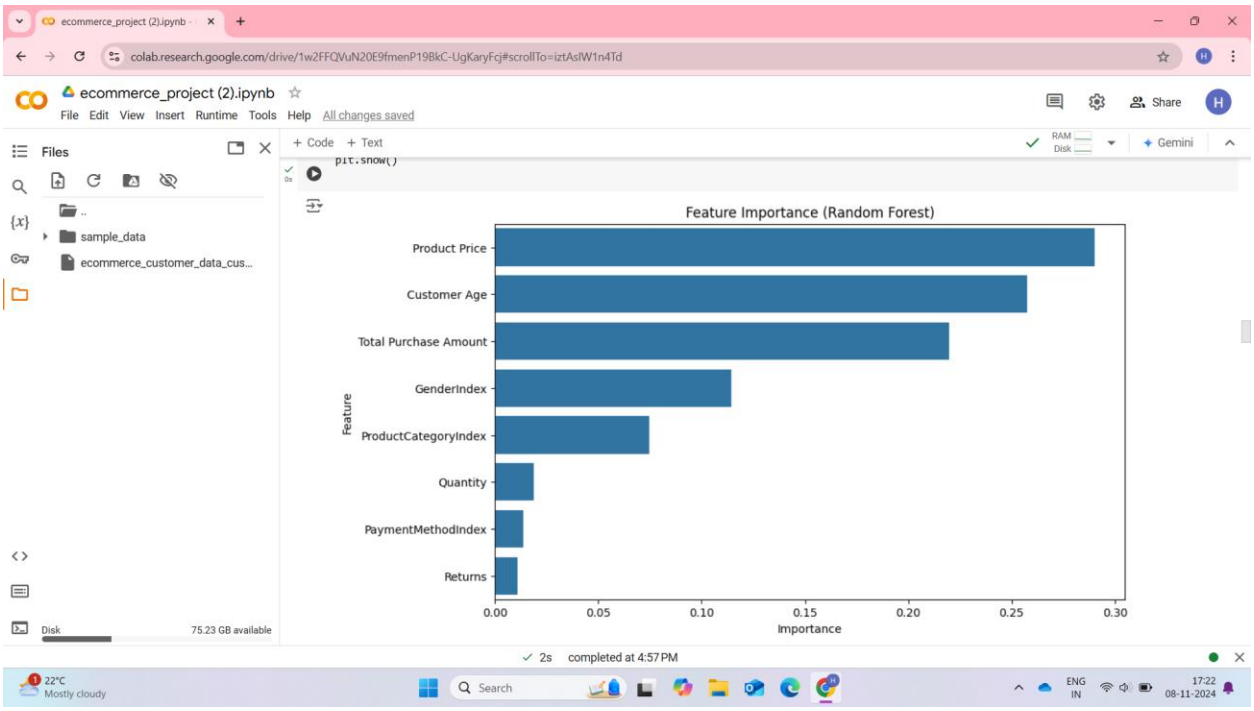
12. Then we will Visualize the data with our machine learning Models with accuray and f1 score suing pandas, matplot and seaborn libraries.


ecommerce_project (2).ipynb
colab.research.google.com/drive/1w2FFQVuN20E9fmenP19BkC-UgKaryFcj#scrollTo=iztAsIW1n4Td
File Edit View Insert Runtime Tools Help All changes saved
Share
+ Code + Text
Gemini
Files
sample_data
ecommerce_customer_data_cus...
Disk 75.23 GB available
Churn Rate by Age Group
Churn Rate
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
45+
35-44
25-34
18-24
Age Group
2s completed at 4:57 PM
22°C Mostly cloudy
Search
ENG IN
17:23 08-11-2024


ecommerce_project (2).ipynb
colab.research.google.com/drive/1w2FFQVuN20E9fmenP19BkC-UgKaryFcj#scrollTo=iztAsIW1n4Td
File Edit View Insert Runtime Tools Help All changes saved
Share
+ Code + Text
Gemini
Files
sample_data
ecommerce_customer_data_cus...
Disk 75.23 GB available
Model F1 Score Comparison
F1 Score
1.0
0.8
0.6
0.4
0.2
0.0
Random Forest
Logistic Regression
GBT
Model
Churn Rate by Categorical Features
2s completed at 4:57 PM
22°C Mostly cloudy
Search
ENG IN
17:22 08-11-2024


ecommerce_project (2).ipynb
colab.research.google.com/drive/1w2FFQVuN20E9fmenP19BkC-UgKaryFcj#scrollTo=iztAsIW1n4Td
File Edit View Insert Runtime Tools Help All changes saved
Share
+ Code + Text
Gemini
Files
sample_data
ecommerce_customer_data_cus...
Disk 75.23 GB available
Average Purchase Amount by Product Category
Average Purchase Amount
Product Category
Home
Electronics
Clothing
Books
2s completed at 4:57 PM
22°C Mostly cloudy
Search
ENG IN
17:23 08-11-2024


ecommerce_project (2).ipynb
colab.research.google.com/drive/1w2FFQVuN20E9fmenP19BkC-UgKaryFcj#scrollTo=iztAsIW1n4Td
File Edit View Insert Runtime Tools Help All changes saved
Share
+ Code + Text
Gemini
Files
sample_data
ecommerce_customer_data_cus...
Disk 75.23 GB available
Correlation Heatmap
ProductCategoryIndex
PaymentMethodIndex
GenderIndex
Product Price
Quantity
Total Purchase Amount
Customer Age
Returns
2s completed at 4:57 PM
22°C Mostly cloudy
Search
ENG IN
17:23 08-11-2024


ecommerce_project (2).ipynb
colab.research.google.com/drive/1w2FFQVuN20E9fmenP19BkC-UgKaryFcj#scrollTo=iztAsIW1n4Td
File Edit View Insert Runtime Tools Help All changes saved
Share
Files
sample_data
ecommerce_customer_data_cus...
Disk 75.23 GB available
+ Code + Text
Gemini
plt.show()
Feature Importance (Random Forest)
Product Price
Customer Age
Total Purchase Amount
GenderIndex
ProductCategoryIndex
Quantity
PaymentMethodIndex
Returns
Feature
0.00
0.05
0.10
0.15
0.20
0.25
0.30
Importance
2s completed at 4:57 PM
22°C
Mostly cloudy
Search
ENG
IN
17:22
08-11-2024