

RECIPE RECOMMENDER ASSIGNMENT

By
Himanshu Gorey
Purnima N S
Shilpa Gururaj

ABSTRACT

To use Spark on Elastic Map Reduce service (EMR) - in AWS and create a recommender with EDA on the given dataset.

PROJECT GOALS



GOAL 1

To understand the use of AWS EMR cluster, S3 bucket utilization and its working, with hands on practice on the same.

GOAL 2

Create and develop a project, and perform EDA on it to extract insightful results on performance improvements of the company.

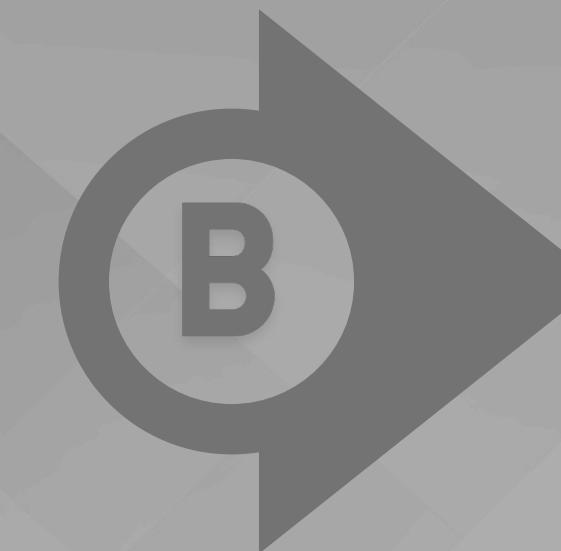


METHODOLOGY

The stepwise actions performed for the project.



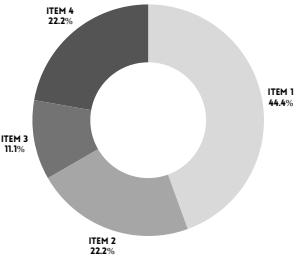
Obtaining dataset,
and creation of S3
buckets and EMR
cluster as per
recommendations.



Creation of a spark
session with python
[pySpark] and
reading the data.



Feature extraction
and assertion check
for the sets and
reupload the
processed data into a
parquet file.



DATA DOWNLOADING

Downloading the below CSV files from the links shared in problem statement.

Raw Recipes Data

https://raw-recipes-clean-upgrad.s3.amazonaws.com/Raw_recipes_cleaned.csv

#Raw Interactions Data

https://raw-interactions-upgrad.s3.amazonaws.com/Raw_interactions_cleaned.csv

CREATING S3 BUCKET AND EMR CLUSTER IN AWS

Create a S3 bucket in public mode within AWS with given specifications.

The screenshot shows the AWS S3 console interface. On the left, there's a sidebar with options like Buckets, Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Block Public Access settings for this account, Storage Lens (Dashboards, Storage Lens groups, AWS Organizations settings), and a Feature spotlight section with a '7' badge. The main area is titled 'Amazon S3' and shows an 'Account snapshot - updated every 24 hours' with a link to 'All AWS Regions'. It also has a 'View Storage Lens dashboard' button. Below this, there are tabs for 'General purpose buckets' (selected) and 'Directory buckets'. A table lists three 'General purpose buckets': 'aws-emr-studio-010031137897-us-east-1' (Creation date: October 2, 2024, 11:07:59 UTC+05:30), 'aws-logs-010031137897-us-east-1' (Creation date: October 2, 2024, 10:07:53 UTC+05:30), and 'mybucks326' (Creation date: October 1, 2024, 07:52:19 UTC+05:30). The row for 'mybucks326' is highlighted with a red box. At the top of the main area, there are standard AWS navigation icons: a magnifying glass for search, a gear for options, a question mark for help, and a user icon for N. Virginia. The top right shows the user's email address: voclabs/user3512982=purni.nsp@gmail.com @ 0100-3113-7897.

Name	AWS Region	IAM Access Analyzer	Creation date
aws-emr-studio-010031137897-us-east-1	US East (N. Virginia) us-east-1	View analyzer for us-east-1	October 2, 2024, 11:07:59 (UTC+05:30)
aws-logs-010031137897-us-east-1	US East (N. Virginia) us-east-1	View analyzer for us-east-1	October 2, 2024, 10:07:53 (UTC+05:30)
mybucks326	US East (N. Virginia) us-east-1	View analyzer for us-east-1	October 1, 2024, 07:52:19 (UTC+05:30)

CREATING S3 BUCKET AND EMR CLUSTER IN AWS

Within the s3 bucket upload the following datafiles which has been gathered and the spark jupyter notebook.

The screenshot shows the AWS S3 console interface. The left sidebar includes links for Buckets, Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Block Public Access settings, Storage Lens (Dashboards, Storage Lens groups, AWS Organizations settings), Feature spotlight (7), and AWS Marketplace for S3. The main area displays the 'mybucks326' bucket. The 'Objects' tab is selected, showing four objects:

Name	Type	Last modified	Size	Storage class
interaction_level_df_proces sed/	Folder	-	-	-
RAW_interactions_cleaned.csv	csv	October 2, 2024, 09:50:58 (UTC+05:30)	330.5 MB	Standard
RAW_recipes_cleaned.csv	csv	October 2, 2024, 09:50:58 (UTC+05:30)	280.1 MB	Standard
Receipe_EDA_pyspark_aws.ipynb	ipynb	October 4, 2024, 11:09:29 (UTC+05:30)	923.1 KB	Standard

CREATING S3 BUCKET AND EMR CLUSTER IN AWS

Now that the dataset is ready, we will create an EMR cluster with m4.xlarge task, node and processing.

The screenshot shows the Amazon EMR console interface. At the top, there is a green success message: "Your cluster 'EMRjupyter' has been successfully created." Below this, the navigation bar shows "Amazon EMR > EMR on EC2: Clusters". The main area displays a table of clusters. The first cluster listed is "EMRjupyter", which has a Cluster ID of "j-1XYUJYFWPSE67". This row is highlighted with a red box. The table includes columns for Cluster ID, Cluster name, Status, Creation time (UTC+05:30), and Elapsed time. The status for the cluster is "Starting" with the sub-status "Preparing cluster". The creation time is "7 October 2024 08:49" and the elapsed time is "2 minutes, 4 seconds".

Clusters (6) Info					
<input type="checkbox"/>	<input type="checkbox"/>	Cluster ID	Cluster name	Status	Creation time (UTC+05:30)
<input type="checkbox"/>	<input type="checkbox"/>	j-1XYUJYFWPSE67	EMRjupyter	Starting Preparing cluster	7 October 2024 08:49
<input type="checkbox"/>	<input type="checkbox"/>				
<input type="checkbox"/>	<input type="checkbox"/>				
<input type="checkbox"/>	<input type="checkbox"/>				

CREATING A SPARK SESSION

Create an EMR studio and a workspace to work with the data and connect it to your cluster.

The screenshot shows the Amazon EMR Studio interface. On the left, there's a sidebar with options like 'EMR Serverless', 'EMR on EC2' (which is expanded), 'Clusters', 'Notebooks and Git repos', 'Events', 'Block public access', 'What's new', 'Video tour', and 'Compact mode'. The main area has two sections: 'Studios' and 'Workspaces (notebooks)'. In the 'Studios' section, there are two entries: 'Studio_2' (created on 4 October 2024 at 11:18, authenticated by IAM, with URL https://es-EKI4Q6JKIOCAOD5NHG1...) and 'Studio_1' (created on 2 October 2024 at 11:20, authenticated by IAM, with URL https://es-BXDEFAWRPTYYG40WB7...). Both entries are highlighted with a red box. In the 'Workspaces (notebooks)' section, there are also two entries: 'Studio_2_Workspace_1' (studio 'Studio_2', status 'Idle', cluster ID 'j-2VEUYJ39KY2XU', created on 4 October 2024 at 11:18, last modified by 'user3512982=purni.nsp@gmail.com' on 4 October) and 'Studio_1_Workspace_1' (studio 'Studio_1', status 'Idle', cluster ID 'j-1XCY2RI5WNRNX', created on 2 October 2024 at 11:20, last modified by 'user3512982=purni.nsp@gmail.com' on 2 October). This section also has a red box around its entries. At the top right, there's a notification for 'Receipe_EDA_pyspark_aws (1.ipynb)' which is '923 KB • Done'. The top navigation bar includes the AWS logo, services menu, search bar, and other standard AWS navigation elements.

Studios (2)

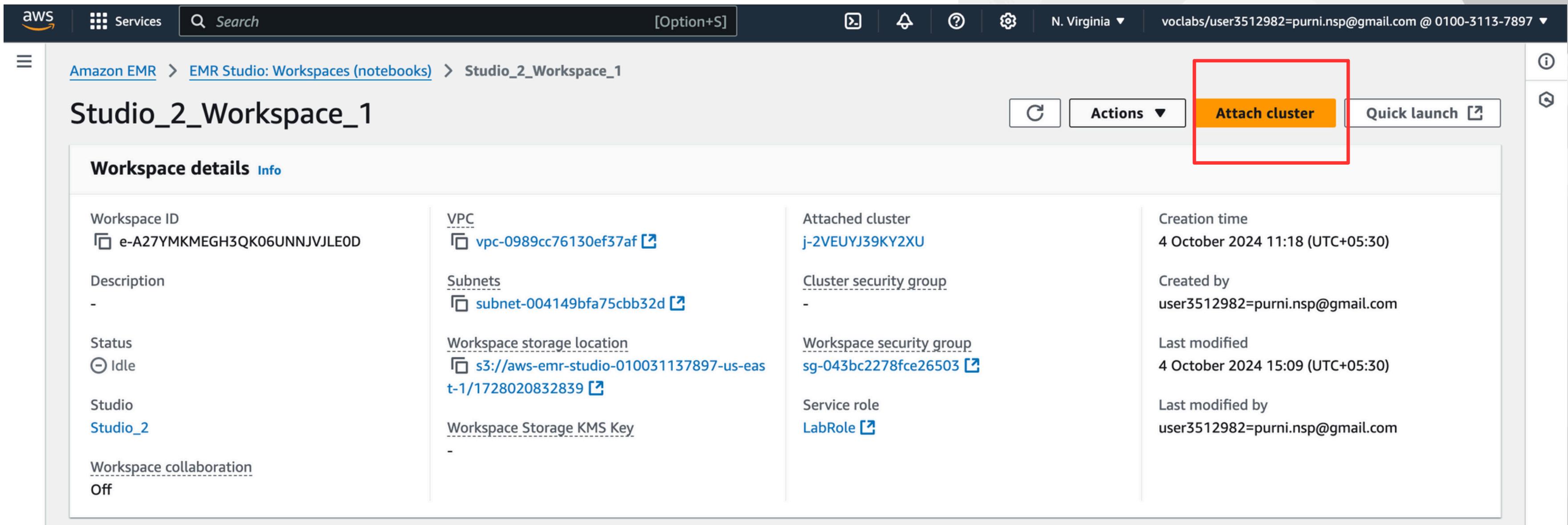
Studio name	Creation time (UTC+05:30)	Authenticated by	Studio Access URL
Studio_2	4 October 2024 11:18	IAM	https://es-EKI4Q6JKIOCAOD5NHG1...
Studio_1	2 October 2024 11:20	IAM	https://es-BXDEFAWRPTYYG40WB7...

Workspaces (notebooks) (2)

Workspace name	Studio name	Status	Cluster ID	Creation time (UTC+05:30)	Last modified by	Last mod
Studio_2_Workspace_1	Studio_2	Idle	j-2VEUYJ39KY2XU	4 October 2024 11:18	user3512982=purni.nsp@gmail.com	4 Octobe
Studio_1_Workspace_1	Studio_1	Idle	j-1XCY2RI5WNRNX	2 October 2024 11:20	user3512982=purni.nsp@gmail.com	2 Octobe

CREATING A SPARK SESSION

Now that a studio and workspace has been created, we have to launch the workspace either with attached or quick launch and then attach a running cluster.



The screenshot shows the AWS EMR Studio workspace details page for 'Studio_2_Workspace_1'. The 'Actions' dropdown menu at the top right is open, with the 'Attach cluster' option highlighted by a red box. The rest of the page displays workspace details such as VPC, Subnets, Attached cluster, and Service role.

Workspace details <small>Info</small>	
Workspace ID	e-A27YMKMEGH3QK06UNNJVLE0D
Description	-
Status	Idle
Studio	Studio_2
Workspace collaboration	Off
VPC	vpc-0989cc76130ef37af
Subnets	subnet-004149bfa75cbb32d
Workspace storage location	s3://aws-emr-studio-010031137897-us-eas-t-1/1728020832839
Workspace Storage KMS Key	-
Attached cluster	j-2VEUYJ39KY2XU
Cluster security group	-
Workspace security group	sg-043bc2278fce26503
Service role	LabRole
Creation time	4 October 2024 11:18 (UTC+05:30)
Created by	user3512982=purni.nsp@gmail.com
Last modified	4 October 2024 15:09 (UTC+05:30)
Last modified by	user3512982=purni.nsp@gmail.com



PYSPARK KERNEt IN JUPYTER NOTEBOOK



A spark session with a name “Basics” has been created.

A screenshot of a Jupyter Notebook interface. The top bar shows the title "Receipe_EDA_pyspark_aws". The toolbar includes icons for file operations, cell types, and a "Markdown" dropdown. On the right, it says "No cluster attached." and "PySpark".

Initial Setup ¶

```
[10]: from pyspark.sql import SparkSession
Last executed at 2024-10-04 12:22:04 in 48ms
VBox()
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(height='25px', width='50%'), ...
```

```
[11]: from pyspark.sql import SparkSession
spark = SparkSession.builder.appName("Basics").getOrCreate()
Last executed at 2024-10-04 12:22:04 in 50ms
VBox()
```

READING THE DATASETS

A spark session with a name “Basics” has been created.

```
[*]: # Task 01 Cell 1 out of 1

raw_recipes_df = (spark.read.csv("s3://mybucks326/RAW_recipes_cleaned.csv", header = True, inferSchema=True,
                                  # argument 1, Add an argument to communicate to the compiler that there is a header in the raw data.
                                  # argument 2, Add an argument to ask the compiler to estimate the data types for all columns.
                                  ))
```

```
# Please forward the exact name of data frames and columns as suggested in the code.
# It will ensure that the assert commands function correctly.
```

```
[*]: raw_recipes_df.show(5)
```

```
[8]: raw_recipes_df.printSchema()
```

```
Last executed at 2024-10-07 09:23:55 in 63ms
```

```
root
|-- name: string (nullable = true)
|-- id: integer (nullable = true)
|-- minutes: integer (nullable = true)
|-- contributor_id: integer (nullable = true)
|-- submitted: date (nullable = true)
|-- tags: string (nullable = true)
|-- nutrition: string (nullable = true)
|-- n_steps: integer (nullable = true)
|-- steps: string (nullable = true)
|-- description: string (nullable = true)
|-- ingredients: string (nullable = true)
|-- n_ingredients: integer (nullable = true)
```

EXTRACTING FEATURES & ASSERT CHECKS

Features has been extracted like :

- Nutrition-per-100 calorie columns
- Create time-based features:
 - days_since_submission_on_review_date
 - months_since_submission_on_review_date
 - years_since_submission_on_review_date

```
16]: # Task 02 Cell 1 out of 2
# 2.1 - string operations to remove square brackets

raw_recipes_df = (raw_recipes_df
    .withColumn('nutrition', F.regexp_replace("nutrition","[\[\]]","",""))
    # add code to remove square brackets
    # pyspark function to replace string characters
)
Last executed at 2024-10-07 09:24:04 in 272ms

17]: # Task 02 Cell 2 out of 3
# STEP 2.2 - split the nutrition string into seven individual values.
# Create an object to split the nutrition column
import pyspark
nutrition_cols_split = pyspark.sql.functions.split(raw_recipes_df['nutrition'], ',')

# Write a loop to extract individual values from the nutrition column

for col_index, col_name in enumerate(nutrition_column_names):
    # col_index holds the index number of each column, e.g., calories will be 0
    # col_name holds the name of each column

    raw_recipes_df = (raw_recipes_df.withColumn(col_name, nutrition_cols_split.getItem(col_index).cast("float")
        # pyspark function to extract individual values from the nutrition_cols_split object
        # You can also cast the extracted value to floats in the same code.
    ))
Last executed at 2024-10-07 09:24:05 in 267ms
```

UPLOADING TO A PARQUET FILE

The Final file has been uploaded in the parquet format in the S3 bucket which can be used for futher EDA process.

The screenshot shows the Amazon S3 console interface. At the top, there is a navigation bar with a search field containing "[Option+S]", several icons (magnifying glass, bell, question mark, gear), and location information "N. Virginia" and "voclabs/user3512982=purni.nsp@gmail.com @ 0100-3113-78". Below the navigation bar, the URL "Amazon S3 > Buckets > mybucks326 > interaction_level_df_processed/" is displayed. On the right side of this URL, there is a button labeled "Copy S3 URI". The main content area shows a folder named "interaction_level_df_processed/". Underneath this folder, there is a sub-section titled "Objects (1) Info" with a link to "Info". Below this, there are several action buttons: "C" (Create), "Copy S3 URI", "Copy URL", "Download", "Open", "Delete", "Actions", "Create folder", and "Upload". A note below these buttons states: "Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)". There is also a search bar with the placeholder "Find objects by prefix" and navigation controls (back, forward, settings). At the bottom, there is a table header with columns: "Name", "Type", "Last modified", "Size", and "Storage class". A single object entry is shown: "interaction_level_df_p/" (Folder), "Folder", "-", "-", and "-".

Name	Type	Last modified	Size	Storage class
interaction_level_df_p/	Folder	-	-	-

CONCLUSION

The dataset has been hence analysed to extract the features and provide analysis on the recommendation of receipes.

S3 BUCKET

Data extraction,
creation of S3 bucket.
Upload the data to the
s3 bucket.
S3 is in public mode
for visibility.

EMR CLUSTER

Creation of emr cluster
with given
specialization, and
creation of studio and
workspace and
attaching it with
cluster.

PARAQUET FILE

Once processing the
file is done, feature
extraction is derived,
we upload the
processed data to a
parquet file into the
same s3 bucket

THANK YOU