**Programming Assignment 2**

**Cloud Computing**

**Name – Gayatri Aavula**

**UCID – ga348@njit.edu**

**GitHub –** <https://github.com/gayatriaavula/winepred_quality.git>

**DockerHub -** [**https://hub.docker.com/repository/docker/ga348/cs643-programming-assignment-2/**](https://hub.docker.com/repository/docker/ga348/cs643-programming-assignment-2/)

**1.**Log into the AWS Management Console.

**2.** the AWS EMR service from the list of available services. Then, choose EMR on EC2 Clusters.

A screenshot of a computer

Description automatically generated

**3.**You'll be taken to the Clusters page where you'll see that there are currently no active clusters.

**4.**To create a new cluster, click on the "Create cluster" button.

A screenshot of a computer

Description automatically generated

**5.**Give any name you want to the cluster. Additionally, ensure that you select the latest version of EMR available and choose 'Spark Interactive' under the Application bundle options.

A screenshot of a computer

Description automatically generated

**6.**I selected the instance type m5.xlarge for the Primary, Core, and Task instance groups. You have the flexibility to choose any instance type that suits your needs.

A screenshot of a computer

Description automatically generated

**7**. To ensure cluster scalability and provisioning, configure Core's instance size to 1 and Task-1's instance size to 2.

**A screenshot of a computer

Description automatically generated**

**8.** In order to pick the security groups for EC2's Primary node, Core, and task nodes, select the security groups that are displayed below.

A screenshot of a computer

Description automatically generated

**9.** To prevent the cluster from being terminated automatically, make sure that the option to manually end the cluster is selected.

A screenshot of a computer

Description automatically generated

10. create a new key pair

A screenshot of a computer

Description automatically generated

**11.** Provide a custom name for the key pair and select 'ppk' as the file format when creating the key pair

A screenshot of a computer

Description automatically generated

**12**. The key pair named 'programming-2' has been created, and the key has been downloaded and stored on the local system for connecting to the cluster using PuTTY as the SSH server.

A screenshot of a computer

Description automatically generated

**13.** The key pair named 'programming-2' has been created, and you can now browse the key .

**A screenshot of a computer

Description automatically generated**

**14.** Go to the IAM roles and choose them as indicated below.

A screenshot of a computer

Description automatically generated

**15.** Click the button labeled "Create Cluster" to initiate the process of creating the clusters.

A screenshot of a computer

Description automatically generated

16. Cluster created successfully.

A screenshot of a computer

Description automatically generated

17. Navigate to the EC2 instance page. As seen below, there are four EC2 instances launched, one of which is a Master node and the other three are Slave nodes.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

18. To access the "ElasticMapReduce-Master" security group, navigate to the EC2 service and select the appropriate Security ID.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

19.Click on the security group to edit its inbound rules

A screenshot of a computer

Description automatically generated

20. In the Inbound Rules section, select Edit Inbound Rules.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

21. Enter the port numbers 22 and 4040 with the settings indicated below, then click the Add rule button and save the rules.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

22. Create an S3 bucket in AWS services to store the dataset.

23.Choose "Create Bucket".

A screenshot of a computer

Description automatically generated

24.Label your bucket as "dataset-programming-assignment-2". Scroll down and click on the "Create bucket" button.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

25. Access the buckets page to view your newly created bucket.   
26. Click on the bucket name you created.

A screenshot of a computer

Description automatically generated

27. Click the upload button.

A screenshot of a computer

Description automatically generated

28. Select the.csv files from your local system by clicking on "Add Files." Now, click the "Upload button" to place the dataset in the S3 bucket. You now have two.csv files in your S3 bucket: ValidationDataset.csv and TrainingDataset.csv.

A screenshot of a computer

Description automatically generated

29. Connect to the server using PuTTY by specifying the PPK file for authentication.

A screenshot of a computer

Description automatically generated

30. Click on "SSH" under "Auth", select "Credentials", provide the path to the location of the PPK file, and then click "Open".

A screenshot of a computer

Description automatically generated

31.click on accept

A screenshot of a computer

Description automatically generated

32. provide ec2-user

A screen shot of a computer

Description automatically generated

33.To set up credentials, configure them in the Master node EC2 instance.   
To configure credentials for the master node, type the following instructions in your terminal:

# mkdir .aws

# touch .aws/credentials

# vi .aws/credentials

34. Copy and paste the credentials from the AWS Academy page and AWS information.

A screenshot of a computer

Description automatically generated

35.To execute our Spark application, we must first install the required packages.   
Run the commands below:   
# sudo yum update.   
# sudo yum install git.

# pip install pyspark findspark boto3 numpy pandas scikit-learn datetime

36.To clone the GitHub repository, enter the following command:   
#git clone https://github.com/gayatriaavula/winepred\_quality.git

37.To get started with the Spark Application, perform the following commands:

# spark-submit --master yarn CS643\_Programming\_assignment\_2/WineTraining.py # spark-submit --master yarn CS643\_Programming\_assignment\_2/WineTesting.py > output.txt

A screen shot of a computer program

Description automatically generated

A computer screen with many lines

Description automatically generated with medium confidence

A computer screen shot of a black background

Description automatically generated

A screen shot of a computer

Description automatically generated

38. Run the command below to view the results:   
#cat output.txt | grep F1   
The results will be displayed here, along with the Accuracy and F1 scores of the Machine Learning methods used.

A screenshot of a computer

Description automatically generated

# DOCKER IMPLEMENTATION –

1.Update System Packages:

sudo yum update -y

2.Install Docker:

sudo yum install -y docker

3.Start Docker Service:

sudo service docker start

4.Check Docker Service Status:

sudo service docker status

A screen shot of a computer

Description automatically generated

5. Create a Dockerfile and build an image with the docker build command.

sudo docker build -t ga348/cs643-programming-assignment-2 .

6.To check if a Docker image was built, use the following command:   
# sudo docker image ls

A screen shot of a computer

Description automatically generated

You can see here that your docker image has been built.

7. To run the docker image, use the following command:

sudo docker run -it ga348/cs643-programming-assignment-2

Instead of using the image name, you can use the image ID: # sudo docker run -it

8.This results in the same return for Accuracy and F1 scores.

A screenshot of a computer

Description automatically generated

9. Run the following command to submit the produced Docker image to the DockerHub repository:

sudo docker push ga348/cs643-programming-assignment-2

A computer screen shot of a black screen

Description automatically generated

10.Download and execute the Docker image from the DockerHub repository, following the instructions provided on the website.

**Git Bash:**

the steps for moving code from my local system to a GitHub repository using Git Bash.

1.git init

2.git add.

3.git status

4.git commit -m “ updated code”

5.git remote add origin https://github.com/gayatriaavula/winepred\_quality.git

6. git push -u origin main

A computer screen with text on it

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A computer screen with text

Description automatically generated