

## Programming Assignment 2:

In this assignment, EMR cluster with 3 nodes is created.

ec2-3-238-12-110.compute-1.amazonaws.com is the master node and the security group of master node has ssh permissions.

The screenshot shows the AWS EMR console with the cluster 'WineQualityPrediction' selected. The 'Summary' tab is active, displaying the following information:

Cluster info	Applications	Cluster management	Status and time
Cluster ID: j-1915DE7R1JYJ Cluster configuration Instance groups Capacity: 1 Primary, 3 Core, 0 Task	Amazon EMR version: emr-7.1.0 Installed applications: Hadoop 3.3.6, Hive 3.1.3, JupyterEnterpriseGateway 2.6.0, Livy 0.8.0, Spark 3.5.0	Log destination in Amazon S3: aws-logs-471112668218-us-east-1/elasticmapreduce Persistent application UIs: Spark History Server, YARN timeline server, Tez UI	Status: Waiting Creation time: April 25, 2024, 12:28 (UTC-04:00) Elapsed time: 2 hours, 11 minutes

Below the summary, there are tabs for Properties, Bootstrap actions, Instances (Hardware), Steps, Applications, Configurations, Monitoring, Events, and Tags (0). Under Properties, there are sections for Operating system (Info) and Cluster logs (Info). The operating system section shows 'Amazon Linux release'. The cluster logs section shows 'Archive log files to Amazon S3'. At the bottom right of the properties section is a 'Edit' button.

### Connect to EC2 Instance - Master node:

1. Open terminal and go to the folder where the key is saved. Change the permission using chmod command.  
chmod 400 prediction.pem
2. Later on, connect the ec2 instance with the following command:  
ssh -i prediction.pem hadoop@ec2-3-238-12-110.compute-1.amazonaws.com  
ec2-3-238-12-110.compute-1.amazonaws.com is the DNS of ec2 instance master node.

**Uploaded required files to S3:** s3://aws-logs-471112668218-us-east-1/winequality/

```
Last login: Thu Apr 25 10:54:50 on ttys000
complete:13: command not found: compdef
(base) dikhita@Dikhitas-Air ~ % cd Downloads
(base) dikhita@Dikhitas-Air Downloads % ssh -i prediction.pem hadoop@ec2-3-238-12-118.compute-1.amazonaws.com
The authenticity of host 'ec2-3-238-12-118.compute-1.amazonaws.com (3.238.12.118)' can't be established.
ED25519 key fingerprint is SHA256:7B19d5geMVX7Hm8l+/vDXF7TP1y9Ga7oUmrPfQX8.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-238-12-118.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
#_
~~ \####_          Amazon Linux 2023
~~ \###|
~~ \#/
~~ \#/.--> https://aws.amazon.com/linux/amazon-linux-2023
~~
~~ \_/_/
~~ \_/_/`_
~~ \_/_/`_
Last login: Thu Apr 25 16:34:16 2024
EEEEEEEEEEEEEEEEEE MBBBBBBM MBBBBBBM RRRRRRRRRRRRRR
E:::;:::;:::;:::;E M:::::;N M:::::;M R:::::::::::;:R
E:::::;E:::::;E M:::::;M M:::::;M R:::::R R:::::R
E:::;E EEEEEE M:::::;M M:::::;M R:::::R R:::::R
E:::::E EEEEEE M:::::M M:::::M M:::::M R:::::RRRRR:::::R
E:::::::::::E M:::::M M:::::M M:::::M R:::::::::::R
E:::::EEEEE M:::::M M:::::M M:::::M R:::::RRRRR:::::R
E:::::E M:::::M M:::::M M:::::M R:::::R R:::::R
E:::::E EEEEEE M:::::M M:::::M M:::::M R:::::R R:::::R
E:::::EEEEE EEEEEE M:::::M M:::::M R:::::R R:::::R
E:::::EEEEE EEEEEE M:::::M M:::::M R:::::R R:::::R
EEEEEEEEEEEEEEEEE MBBBBBBM MBBBBBBM RRRRRRRR
[.hadoop@ip-172-31-18-247 ~]$ aws s3 cp s3://aws-logs-47111268218-us-east-1/winequality/WineQualityPrediction.py .
download: s3://aws-logs-47111268218-us-east-1/winequality/WineQualityPrediction.py to ./WineQualityPrediction.py
[.Apr 25, 2024 4:48:11 PM org.apache.spark.launcher.Log4jHotPatchOption staticJavaAgentOption
WARNING: spark.log4jHotPatch.enabled is set to true, but /usr/share/log4j-cve-2021-44228-hotpatch/jdk17/Log4jHotPatchFat.jar does not exist at the configured location

Traceback (most recent call last):
  File "/home/hadoop/WineQualityPrediction.py", line 1, in <module>
    import numpy as np # linear algebra
ModuleNotFoundError: No module named 'numpy'
24/04/25 16:48:13 INFO ShutdownHookManager: Shutdown hook called
24/04/25 16:48:13 INFO ShutdownHookManager: Deleting directory /mnt/tmp/spark-2db106ef-5da4-4316-b413-72282c13be2a
[hadoop@ip-172-31-18-247 ~]$ pip install numpy
Defaulting to user installation because normal site-packages is not writeable
Collecting numpy
  Downloading numpy-1.26.4-cp39-cp39-manylinux_2.17_x86_64.manylinux2014_x86_64.whl (18.2 MB)
|██████████| 18.2 MB 4.9 MB/s
Installing collected packages: numpy
  WARNING: The script f2py is installed in '/home/hadoop/.local/bin' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed numpy-1.26.4
[hadoop@ip-172-31-18-247 ~]$ pip install pandas
Defaulting to user installation because normal site-packages is not writeable
Collecting pandas
  Downloading pandas-2.2.2-cp39-cp39-manylinux_2.17_x86_64.manylinux2014_x86_64.whl (13.1 MB)
|██████████| 13.1 MB 6.4 MB/s
Requirement already satisfied: pytz>=2020.9 in /usr/local/lib/python3.9/site-packages (from pandas) (2024.1)
Collecting python-dateutil>=2.8.2
  Downloading python_dateutil-2.9.0.post0-py3-none-any.whl (229 kB)
|██████████| 229 kB 99.1 MB/s
```

> pip install numpy  
 > pip install pandas  
 > spark-submit WineQualityPrediction.py

This will install numpy and pandas and run the file  
 Models generated will be stored in S3 bucket.

```

24/04/25 16:58:16 INFO SparkContext: Starting job: showString at NativeMethodAccessorImpl.java:0
24/04/25 16:58:16 INFO DAGScheduler: Got job 61 (showString at NativeMethodAccessorImpl.java:0) with 1 output partitions
24/04/25 16:58:16 INFO DAGScheduler: Final stage: ResultStage 85 (showString at NativeMethodAccessorImpl.java:0)
24/04/25 16:58:16 INFO DAGScheduler: Missing partitions of final stage: List()
24/04/25 16:58:16 INFO DAGScheduler: Submitting ResultStage 85 (MapPartitionsRDD[278] at showString at NativeMethodAccessorImpl.java:0), which has no missing parents
24/04/25 16:58:16 INFO MemoryStore: Block broadcast_129 stored as bytes in memory (estimated size 898.4 Kib, free 1846.2 Mib)
24/04/25 16:58:16 INFO MemoryStore: Block broadcast_129_piece0 stored as bytes in memory (estimated size 284.0 Kib, free 1045.9 Mib)
24/04/25 16:58:16 INFO BlockManagerInfo: Added broadcast_129_piece0 in memory on ip-172-31-10-247.ec2.internal:42341 (size: 284.0 Kib, free: 1048.3 Mib)
24/04/25 16:58:16 INFO DAGScheduler: Submitting 1 missing tasks from broadcast at DAGScheduler.scale:1656
24/04/25 16:58:16 INFO DAGScheduler: Adding task 0.0 in stage 85.0 (RDD[278] at showString at NativeMethodAccessorImpl.java:0) (first 15 tasks are for partitions Vector(0))
24/04/25 16:58:16 INFO TaskSetManager: Adding task 0.0 in stage 85.0 (RDD[278] at showString at NativeMethodAccessorImpl.java:0) (first 15 tasks are for partitions Vector(0))
24/04/25 16:58:16 INFO TaskSetManager: Starting task 0.0 in stage 85.0 (ID 97) (ip-172-31-14-136.ec2.internal, executor 2, partition 0, RACK_LOCAL, 8404 bytes)
24/04/25 16:58:16 INFO BlockManagerInfo: Added broadcast_129_piece0 in memory on ip-172-31-14-136.ec2.internal:33589 (size: 284.0 Kib, free: 4.8 Gib)
24/04/25 16:58:16 INFO TaskSetManager: Finished task 0.0 in stage 85.0 (ID 97) in 160 ms on ip-172-31-14-136.ec2.internal (executor 2) (1/1)
24/04/25 16:58:16 INFO DAGScheduler: Received TaskKilled event (showString at NativeMethodAccessorImpl.java:0) from pool
24/04/25 16:58:16 INFO DAGScheduler: ResultStage 85 (showString at NativeMethodAccessorImpl.java:0) finished in 0.184 s
24/04/25 16:58:16 INFO DAGScheduler: Job 61 is finished. Cancelling potential speculative or zombie tasks for this job
24/04/25 16:58:16 INFO YarnScheduler: Killing all running tasks in stage 85: Stage finished
24/04/25 16:58:16 INFO DAGScheduler: Job 61 finished: showString at NativeMethodAccessorImpl.java:0, took 0.185515 s
+---+---+---+---+---+
|quality|prediction|
+---+---+---+---+---+
| 5.0 | 5.0 |
| 5.0 | 5.0 |
| 5.0 | 5.0 |
| 6.0 | 6.0 |
| 5.0 | 5.0 |
| 5.0 | 5.0 |
| 5.0 | 5.0 |
| 5.0 | 5.0 |
| 7.0 | 5.0 |
| 7.0 | 5.0 |
| 5.0 | 5.0 |
| 5.0 | 5.0 |
| 5.0 | 5.0 |
| 5.0 | 5.0 |
| 5.0 | 5.0 |
| 5.0 | 5.0 |
| 7.0 | 5.0 |
| 5.0 | 5.0 |
| 4.0 | 5.0 |
| 6.0 | 5.0 |
+---+---+---+---+---+
only showing top 20 rows

24/04/25 16:58:16 INFO SparkContext: Invoking stop() from shutdown hook
24/04/25 16:58:16 INFO SparkContext: SparkContext is stopping with exitCode 0.
24/04/25 16:58:16 INFO MapOutputTrackerMasterEndpoint: Stopped MapOutputTrackerMasterEndpoint
24/04/25 16:58:16 INFO YarnClientSchedulerBackend: Interrupting monitor thread
24/04/25 16:58:16 INFO YarnClientSchedulerBackend: Shutting down all executors
24/04/25 16:58:16 INFO YarnSchedulerBackendYarnDriverEndpoint: Asking each executor to shut down
24/04/25 16:58:16 INFO YarnClientSchedulerBackend: YARN client scheduler backend Stopped
24/04/25 16:58:16 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
24/04/25 16:58:16 INFO OutputCommitCoordinator: Stopped
24/04/25 16:58:16 INFO BlockManager: BlockManager stopped
24/04/25 16:58:16 INFO BlockManagerMaster: BlockManagerMaster stopped
24/04/25 16:58:16 INFO OutputCommitCoordinator: OutputCommitCoordinator stopped!
24/04/25 16:58:16 INFO SparkContext: Successfully stopped SparkContext
24/04/25 16:58:16 INFO ShutdownHookManager: Shutdown hook called
24/04/25 16:58:16 INFO ShutdownHookManager: Deleting directory /mnt/tmp/spark-c4d6bc124-0891-4e6b-8400-b8ba88d28ade
24/04/25 16:58:16 INFO ShutdownHookManager: Deleting directory /mnt/tmp/spark-99d92c64-fd0b-4577-b445-39f25b5894da
24/04/25 16:58:16 INFO ShutdownHookManager: Deleting directory /mnt/tmp/spark-c4d6bc124-0891-4e6b-8400-b8ba88d28ade/pyspark-d20c1d65-b4e2-489c-af76-019abfc8e9ef
[hadoop@ip-172-31-10-247 ~] $

```

The screenshot shows the AWS S3 console interface for the 'winequality' bucket. At the top, there's a search bar and navigation buttons. Below it, the 'Objects' tab is active, showing a list of items:

- Dockerfile**: File type, last modified April 25, 2024, 12:18:04 (UTC-04:00), 982.0 B, Standard storage class.
- LogisticRegression.model/**: Folder.
- RandomForestClassifier.model/**: Folder.
- TrainingDataset.csv**: CSV file type, last modified April 25, 2024, 08:18:51 (UTC-04:00), 67.2 KB, Standard storage class.
- ValidationDataset.csv**: CSV file type, last modified April 25, 2024, 08:18:52 (UTC-04:00), 8.6 KB, Standard storage class.
- WineQualityPrediction.py**: PY file type, last modified April 25, 2024, 12:18:43 (UTC-04:00), 5.8 KB, Standard storage class.

## Docker image:

Installed docker on local computer and created docker image and pushed the docker image on > docker hub with the following command.

> docker build --platform linux/amd64 -t wine-quality-docker:v5.0 .

> docker tag wine-quality-docker:v5.0 dk559/wine-quality-prediction:v5.0

> docker push dk559/wine-quality-prediction:v4.0

```

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/mdjimqo58hbzkmsfv42daf83
(base) dikshita@Dikshitas-Air ProgrammingAssignment % docker build --platform linux/amd64 -t wine-quality-docker:v4.0 .
[+] Building 160.3s (21/21) FINISHED   docker:desktop-linux
--> [internal] load build definition from Dockerfile  0.0s
--> => transferring dockerfile: 1.11kB  0.0s
--> [internal] load metadata for docker.io/library/ce  0.9s
--> [auth] library/centos:pull token for registry-1.d  0.0s
--> [internal] load .dockerrcignore  0.0s
--> => transferring context: 2B  0.0s
--> [ 1/15] FROM docker.io/library/centos:7@sha256:be  4.1s
--> => resolve docker.io/library/centos:7@sha256:be65  0.0s
--> sha256:be65f488b7764ad3638f236 1.20kB / 1.20kB  0.0s
--> sha256:dead07b4d8ed7e29e98de0f4504 529B / 529B  0.0s
--> sha256:eeb6ee3f44bd005103bb561 2.75kB / 2.75kB  0.0s
--> sha256:20473b07ccdf0912cd6f 76.10MB / 76.10MB  2.1s
--> => extracting sha256:2d473b07ccdf0912cd6f1a70335  1.8s
--> [internal] load build context  0.0s
--> => transferring context: 2708  0.0s
--> [ 2/15] RUN yum -y update && yum -y install python 68.0s
--> [ 3/15] RUN python -V  0.1s
--> [ 4/15] RUN python3 -V  0.1s
--> [ 5/15] RUN pip3 install --upgrade pip  2.2s
--> [ 6/15] RUN pip3 install numpy pandas  5.4s
--> [ 7/15] RUN cd /opt && wget https://apache.osus  75.3s
--> [ 8/15] RUN ln -s /opt/spark-3.4.3-bin-hadoop3 /  0.2s
--> [ 9/15] RUN (echo 'export SPARK_HOME=/opt/spark'  0.1s
--> [10/15] RUN mkdir /wineapp  0.2s
--> [11/15] COPY WineQualityPrediction.py /wineapp/  0.0s
--> [12/15] RUN rm /bin/sh && ln -s /bin/bash /bin/sh  0.1s
--> [13/15] RUN /bin/bash -c "source ~/.bashrc"  0.2s
--> [14/15] RUN /bin/bash -c "source ~/.bashrc"  0.2s
--> [15/15] WORKDIR /wineapp  0.0s
--> exporting to image  2.2s
--> => exporting layers  2.2s
--> => writing image sha256:06eec66ceaf1596021910420b  0.0s
--> => naming to docker.io/library/wine-quality-docker  0.0s

View build details: docker-desktop://dashboard/build/desktop-linux/tz2aohofczn360tyurh9tg347

Build multi-platform images faster with Docker Build Cloud: https://docs.docker.com/go/docker-build-cloud
(base) dikshita@Dikshitas-Air ProgrammingAssignment % docker tag wine-quality-docker:v4.0 dk559/wine-quality-prediction:v4.0
(base) dikshita@Dikshitas-Air ProgrammingAssignment % docker push dk559/wine-quality-prediction:v4.0
The push refers to repository [docker.io/dk559/wine-quality-prediction]
5f70bf18a086: Layer already exists
9a9bc32806ed: Pushed
576f14657c96: Pushed
6b678edbcdff: Pushed
64e4dd4fe1ef: Pushed
b2072b221a0e4: Pushed

```

The screenshot shows the AWS CloudWatch Metrics interface. A single metric named 'CPUUtilization' is displayed with a value of 0.00% over a period of 1 hour. The chart has a light blue background with a white grid. The Y-axis is labeled 'CPUUtilization' and ranges from 0.00% to 100.00%. The X-axis is labeled 'Time' and shows the current time. The data point is a small blue dot at the bottom left of the chart area.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
i-09986f5e70c651fe4	Running	m5.xlarge	2/2 checks passed	<a href="#">View alarms</a>	
i-0c2d8b5cf6250fe35	Running	m5.xlarge	2/2 checks passed	<a href="#">View alarms</a>	
i-0f65b303135ccad7a	Running	m5.xlarge	2/2 checks passed	<a href="#">View alarms</a>	
MasterNode	Running	m5.xlarge	2/2 checks passed	<a href="#">View alarms</a>	
<b>DockerEC2</b>	<b>Running</b>	t2.micro	2/2 checks passed	<a href="#">View alarms</a>	

**i-0a0e3a9d2d0bf09e4 (DockerEC2)**

- Details**
- Status and alarms** [New](#)
- Monitoring**
- Security**
- Networking**
- Storage**
- Tags**

**Instance summary** [Info](#)

Instance ID <a href="#">i-0a0e3a9d2d0bf09e4 (DockerEC2)</a>	Public IPv4 address <a href="#">18.234.221.255   open address</a>	Private IPv4 addresses <a href="#">172.31.94.7</a>
IPv6 address -	Instance state <a href="#">Running</a>	Public IPv4 DNS <a href="#">ec2-18-234-221-255.compute-1.amazonaws.com   open address</a>
Hostname type IP name: ip-172-31-94-7.ec2.internal	Private IP DNS name (IPv4 only) <a href="#">ip-172-31-94-7.ec2.internal</a>	

### Run the Docker image on EC2 instance:

```
> sudo yum update -y  
> sudo yum install docker -y  
> sudo service docker start  
> sudo usermod -a -G docker ec2-user  
> sudo docker run dk559/wine-quality-prediction:v5.0
```

I learned about AWS EMR, Dockerfile and Apache Spark by watching YouTube videos

I had used ChatGPT incase I faced any errors by pasting the errors.

Example: I was facing issue with the Docker run command on EC2 instance after creating the docker image. ChatGPT suggested me to pass the platform parameter while creating the image.

The screenshot shows a dark-themed chat interface between a user and ChatGPT. The user, identified by a yellow circular icon with 'DI', asks about a Docker image mismatch. ChatGPT, identified by a blue circular icon with a swirl, responds with an explanation and a solution involving the '--platform' option.

**You**  
The requested image's platform (linux/arm64/v8) does not match the detected host platform (linux/amd64/v3) and no specific platform was requested

**ChatGPT**  
The error message you're encountering indicates a mismatch between the platform of the Docker image you're trying to run and the platform of the host system where Docker is running.  
It seems that the Docker image you're trying to run is built for the `linux/arm64/v8` platform, which is for ARM-based systems, while the host system where Docker is running is `linux/amd64`, which is for x86-based systems.  
To resolve this issue, you have a few options:  
1. **Build Docker Image for the Host Architecture:** If you have access to the Dockerfile used to build the image, you can rebuild the Docker image for the `linux/amd64` platform. You can do this by specifying the `--platform` option when building the image:  
bash  
docker build --platform linux/amd64 -t your-image-name .  
2. **Pull Correct Image for Host Architecture:** If you're pulling the image from a registry, ensure