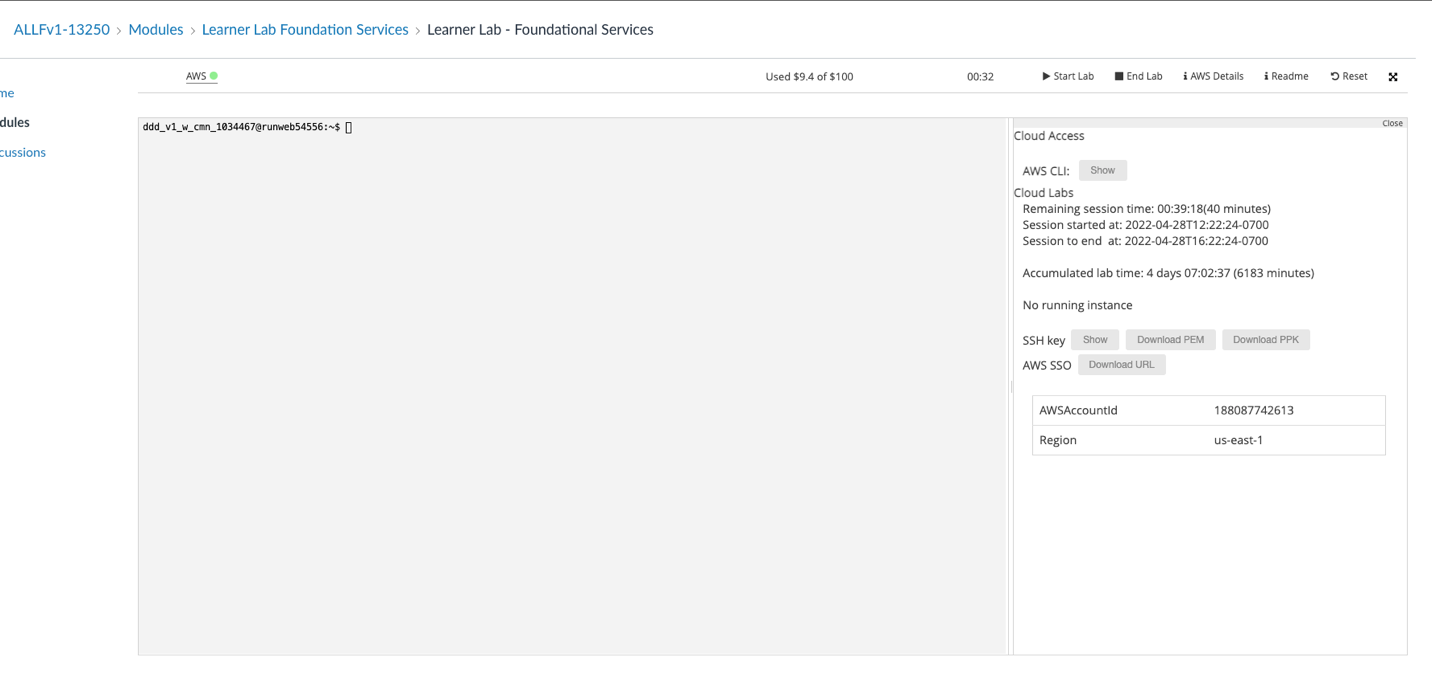
Link to GitHub repository: <https://github.com/rimpal1/Wine-Quality-Prediction.git>

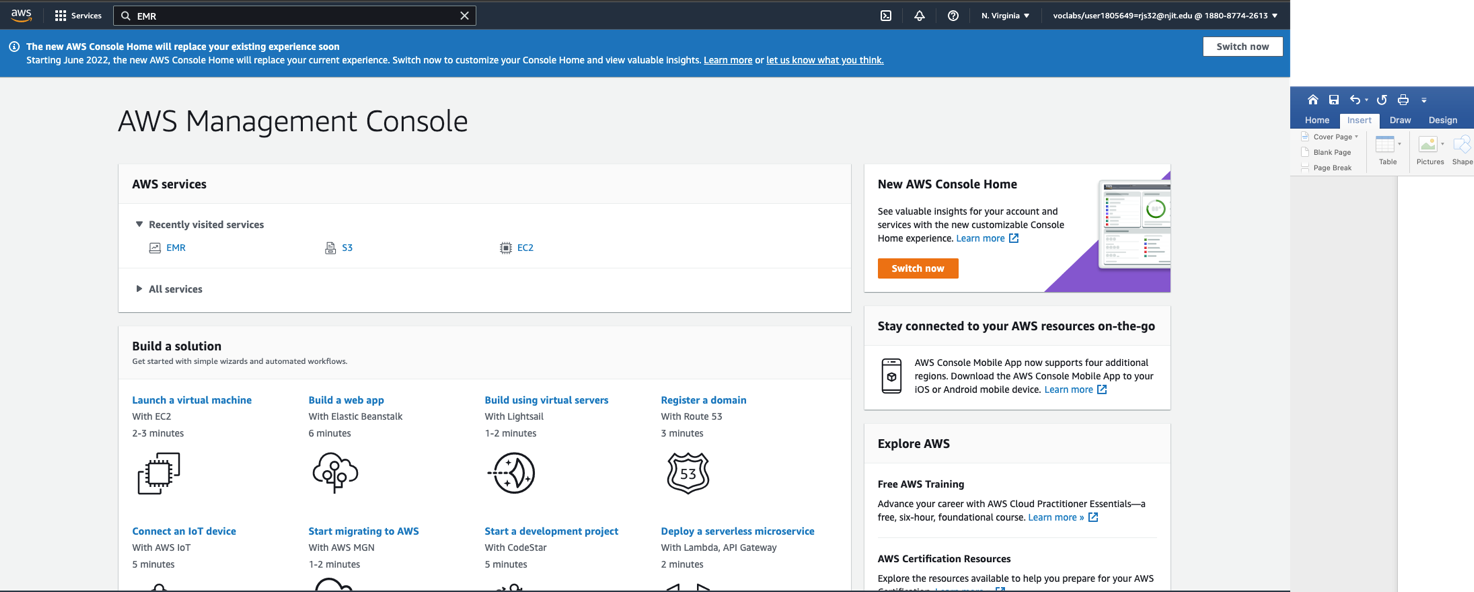
Link to docker container: rimpal1/wine-quality-pred

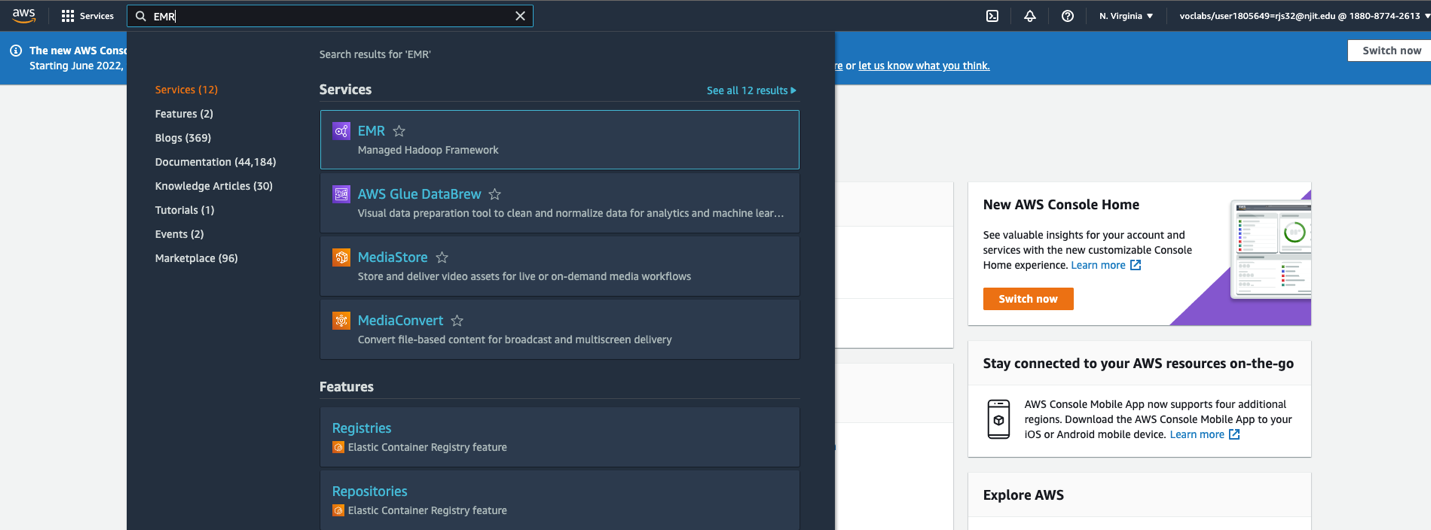
**Step-by-step instructions on how to set up cloud environment:**

1. **Parallel Training implementation**

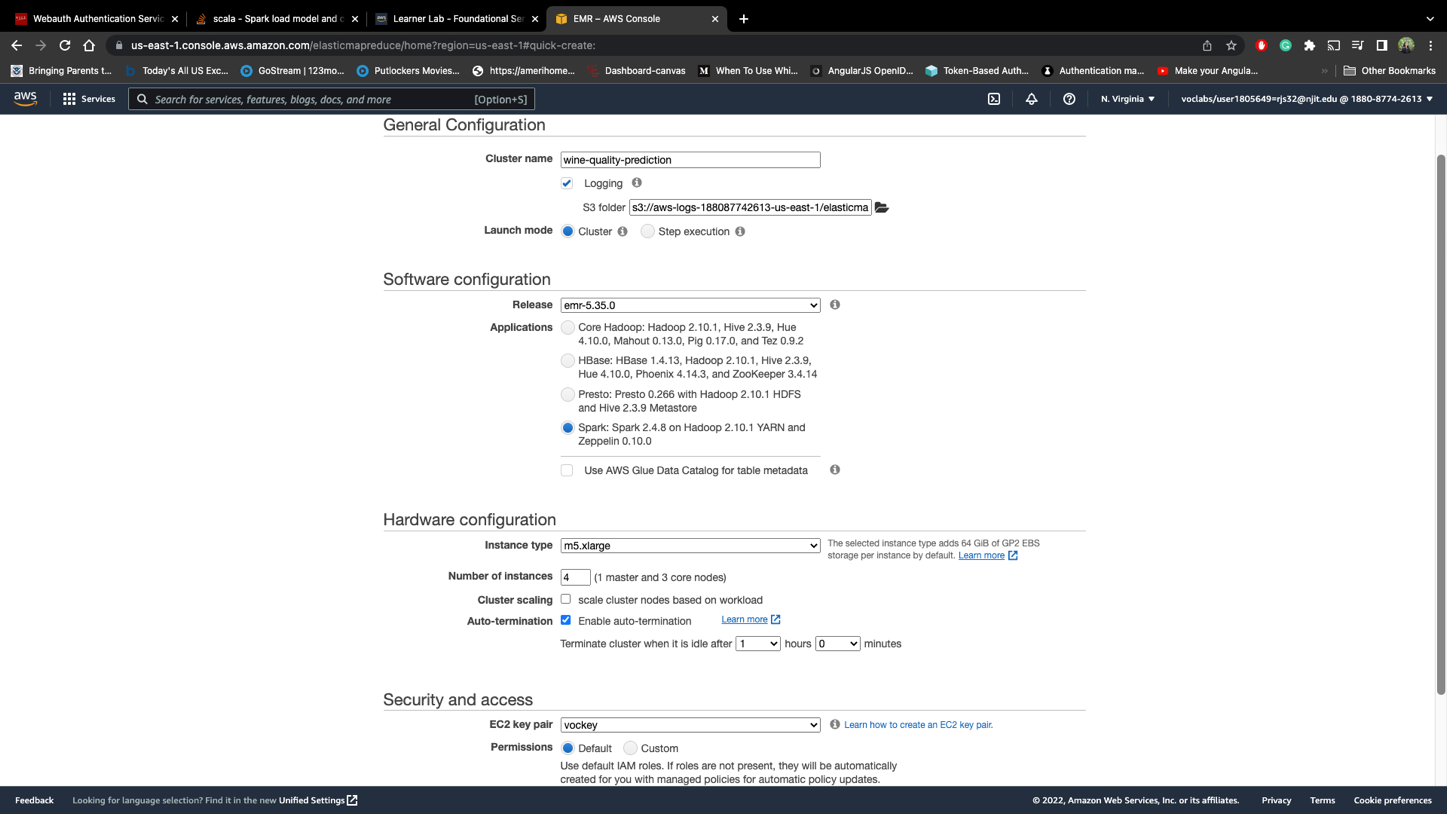
* *How to create an EMR cluster?*

1. Go to AWS → Learner Lab – Foundational Services.
2. Start the lab.
3. Go to “AWS Details” → we need to download the single sign on URL, so click on “Download URL”****
4. For creating the cluster, go to the recently downloaded URL.
5. Now, we need to create EC2 instances’ cluster.
6. Look for “EMR” in the AWS Management Console.

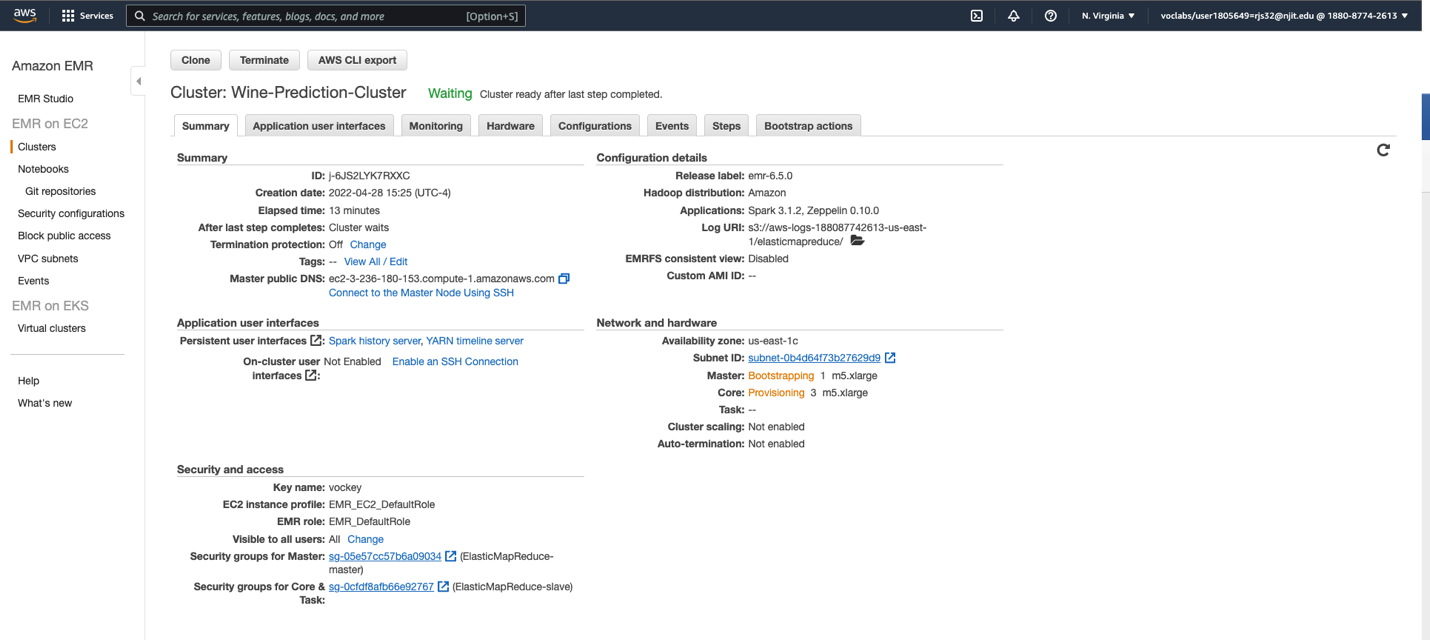
****

****

1. Now, click on create cluster → Hardware Configuration → Number of instances as 4 – 1 master and 3 cores (slaves)



1. Now, Start the cluster – this will take several minutes to complete.



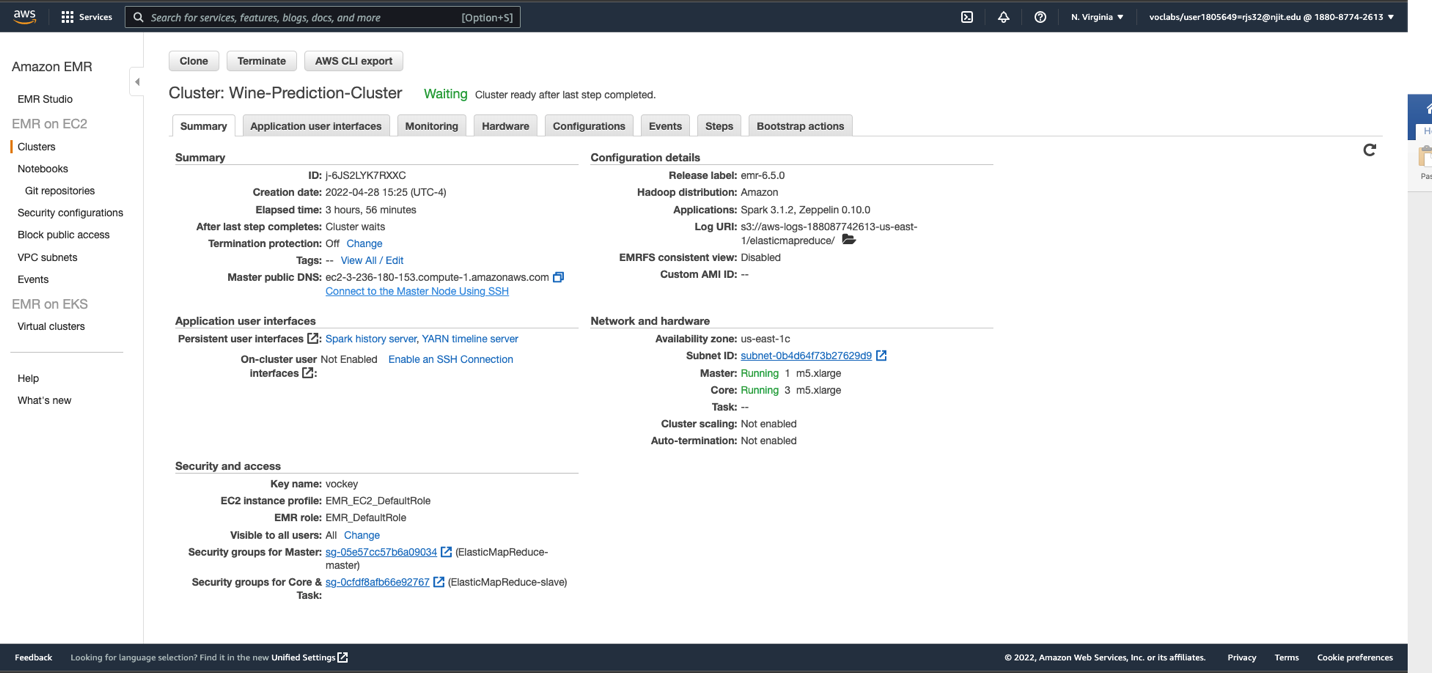
1. After creating the cluster, following will be shown on dashboard.

****

Now, in order to access the cluster, we need to update the inbound rules – in order to allow ourselves access the cluster through SSH.

Follow the steps below for connecting the cluster.

1. Click on the “Connect to the Master Node using SSH”



1. In order to copy the file to HDFS for master and slave nodes, we need to first copy files to EMR local file system and then put it on HDFS.

rimpalsuthar@Rimpals-MacBook-Pro Documents % chmod 400 labsuser.cer

rimpalsuthar@Rimpals-MacBook-Pro Documents % scp -i labsuser.cer /Users/rimpalsuthar/Desktop/Spring-2022/Cloud-computing/assignment2/Wine-Quality-Prediction/data/\* hadoop@ec2-3-236-180-153.compute-1.amazonaws.com:/home/hadoop

TrainingDataset.csv 100% 67KB 1.0MB/s 00:00

ValidationDataset.csv 100% 8664 510.9KB/s 00:00

Wine-Quality-Prediction-1.0.jar 100% 6491 324.0KB/s 00:00

[hadoop@ip-172-31-27-52 ~]$ hadoop fs -put \* .

Found 3 items

-rw-r--r-- 1 hadoop hdfsadmingroup 68706 2022-04-28 04:51 TrainingDataset.csv

-rw-r--r-- 1 hadoop hdfsadmingroup 8664 2022-04-28 04:51 ValidationDataset.csv

-rw-r--r-- 1 hadoop hdfsadmingroup 6491 2022-04-28 04:51 Wine-Quality-Prediction-1.0.jar

1. Now we can submit the spark job to run in the cluster mode (In parallel) using below command. It will read training and validation files from HDFS. Once completed, it will write the final model to HDFS. The Jar will take three args {1} = training set, {2} = validation set, {3} = processed modal path".

[hadoop@ip-172-31-8-200 ~]$ **spark-submit --deploy-mode cluster --class com.njit.winequalitypred.AppInitModelTrainer Wine-Quality-Prediction-1.0.jar hdfs://ip-172-31-8-200.ec2.internal:8020/user/hadoop/TrainingDataset.csv hdfs://ip-172-31-8-200.ec2.internal:8020/user/hadoop/ValidationDataset.csv hdfs://ip-172-31-8-200.ec2.internal:8020/user/hadoop/model/**

22/04/28 21:05:03 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

22/04/28 21:05:03 INFO RMProxy: Connecting to ResourceManager at ip-172-31-8-200.ec2.internal/172.31.8.200:8032

22/04/28 21:05:03 INFO Client: Requesting a new application from cluster with 3 NodeManagers

…………….

………

…….

22/04/28 21:05:35 INFO Client: Application report for application\_1651174283219\_0005 (state: RUNNING)

22/04/28 21:05:36 INFO Client: Application report for application\_1651174283219\_0005 (state: FINISHED)

22/04/28 21:05:36 INFO Client:

client token: N/A

diagnostics: N/A

ApplicationMaster host: ip-172-31-4-237.ec2.internal

ApplicationMaster RPC port: 40715

queue: default

start time: 1651179907745

final status: **SUCCEEDED**

tracking URL: http://ip-172-31-8-200.ec2.internal:20888/proxy/application\_1651174283219\_0005/

user: hadoop

22/04/28 21:05:36 INFO ShutdownHookManager: Shutdown hook called

22/04/28 21:05:36 INFO ShutdownHookManager: Deleting directory /mnt/tmp/spark-d74ff508-eeba-4f92-b768-ae66f0b1fe27

22/04/28 21:05:36 INFO ShutdownHookManager: Deleting directory /mnt/tmp/spark-6c4d3819-996a-4d40-a57a-ded575bcc8be

[hadoop@ip-172-31-8-200 ~]$

[hadoop@ip-172-31-8-200 ~]$

[hadoop@ip-172-31-8-200 ~]$ hadoop fs -ls

SLF4J: Class path contains multiple SLF4J bindings.

SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: Found binding in [jar:file:/usr/share/aws/emr/emrfs/lib/slf4j-log4j12-1.7.12.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: See http://www.slf4j.org/codes.html#multiple\_bindings for an explanation.

SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]

Found 5 items

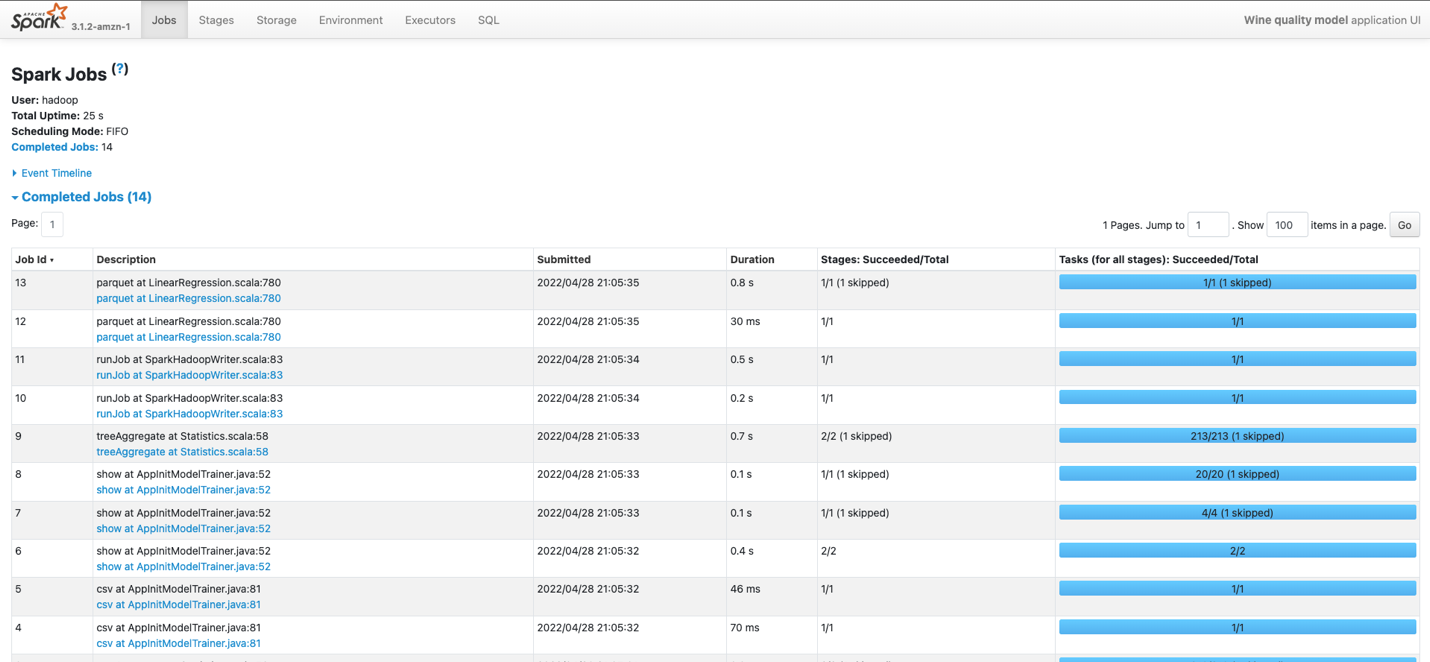
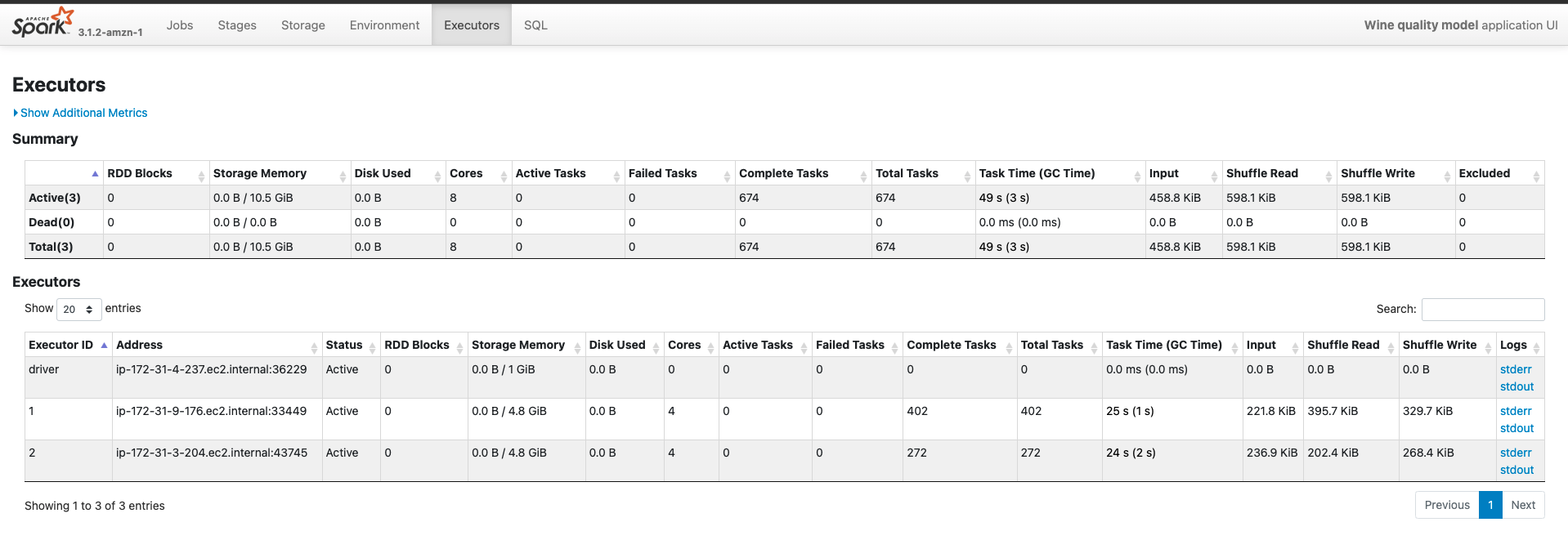
drwxr-xr-x - hadoop hdfsadmingroup 0 2022-04-28 21:05 .sparkStaging

-rw-r--r-- 1 hadoop hdfsadmingroup 68706 2022-04-28 20:08 TrainingDataset.csv

-rw-r--r-- 1 hadoop hdfsadmingroup 8664 2022-04-28 20:08 ValidationDataset.csv

-rw-r--r-- 1 hadoop hdfsadmingroup 6491 2022-04-28 20:08 Wine-Quality-Prediction-1.0.jar

drwxr-xr-x - hadoop hdfsadmingroup 0 2022-04-28 21:05 model

****

1. Now that we have built the model, we will perform the prediction application test and model performance metrics against the validation file. As we can see, now we are using Prediction class which expects two args; {1} = Test file, {2} = built model path

[hadoop@ip-172-31-8-200 ~]$ **spark-submit --class com.njit.winequalitypred.Prediction Wine-Quality-Prediction-1.0.jar hdfs://ip-172-31-8-200.ec2.internal:8020/user/hadoop/ValidationDataset.csv hdfs://ip-172-31-8-200.ec2.internal:8020/user/hadoop/model/**

+--------------------+-------+------------------+

| features|quality| prediction|

+--------------------+-------+------------------+

|[4.7,0.6,0.17,2.3...| 6| 5.775033061579631|

|[8.3,0.42,0.38,2....| 6| 5.826299201753907|

|[10.0,0.48,0.24,2...| 6| 5.48258072234084|

|[10.9,0.53,0.49,4...| 6| 6.032238864423105|

|[5.0,1.02,0.04,1....| 4| 4.677494723971572|

|[9.3,0.715,0.24,2...| 5| 5.409135180742741|

|[7.8,0.46,0.26,1....| 6| 5.318677113694324|

|[5.6,0.615,0.0,1....| 5| 5.042239679245807|

|[9.3,0.39,0.44,2....| 5| 6.029804263394677|

|[7.6,0.9,0.06,2.5...| 5|4.9943292794806595|

|[10.9,0.21,0.49,2...| 6| 6.460044027268772|

|[8.1,0.545,0.18,1...| 6| 5.158607763016146|

|[7.4,0.66,0.0,1.8...| 5| 5.015281677181566|

|[7.0,0.62,0.08,1....| 5| 4.907881419038748|

|[8.5,0.28,0.56,1....| 7| 5.829505199425492|

|[11.6,0.32,0.55,2...| 7| 6.238443699113239|

|[10.5,0.51,0.64,2...| 7| 6.117838445884546|

|[13.7,0.415,0.68,...| 6| 5.930627832320679|

|[11.9,0.38,0.49,2...| 5| 5.762192588031679|

|[8.0,0.33,0.53,2....| 6| 5.542901695484243|

+--------------------+-------+------------------+

only showing top 20 rows

Absolute mean error:0.5294072418854212

1. In order to run using docker, clone the git repo, run command “docker-compose up -d” that will start the spark cluster. Once the cluster is up and running, run the below command. Note: it needs to have exact file name with “Test-File.csv” with absolute path.

docker run --rm -it --network wine-quality-prediction\_default -v ${pwd}/data/Test-File.csv:/usr/src/app/data/Test-File.csv --name wine-quality-pred-test --link spark-master:spark-master rimpal1/wine-quality-pred