CS 643 - 851 - Cloud computing

Zachary Patronick

12/7/2024

Module 7 Assignment 1: Programming Assignment 2

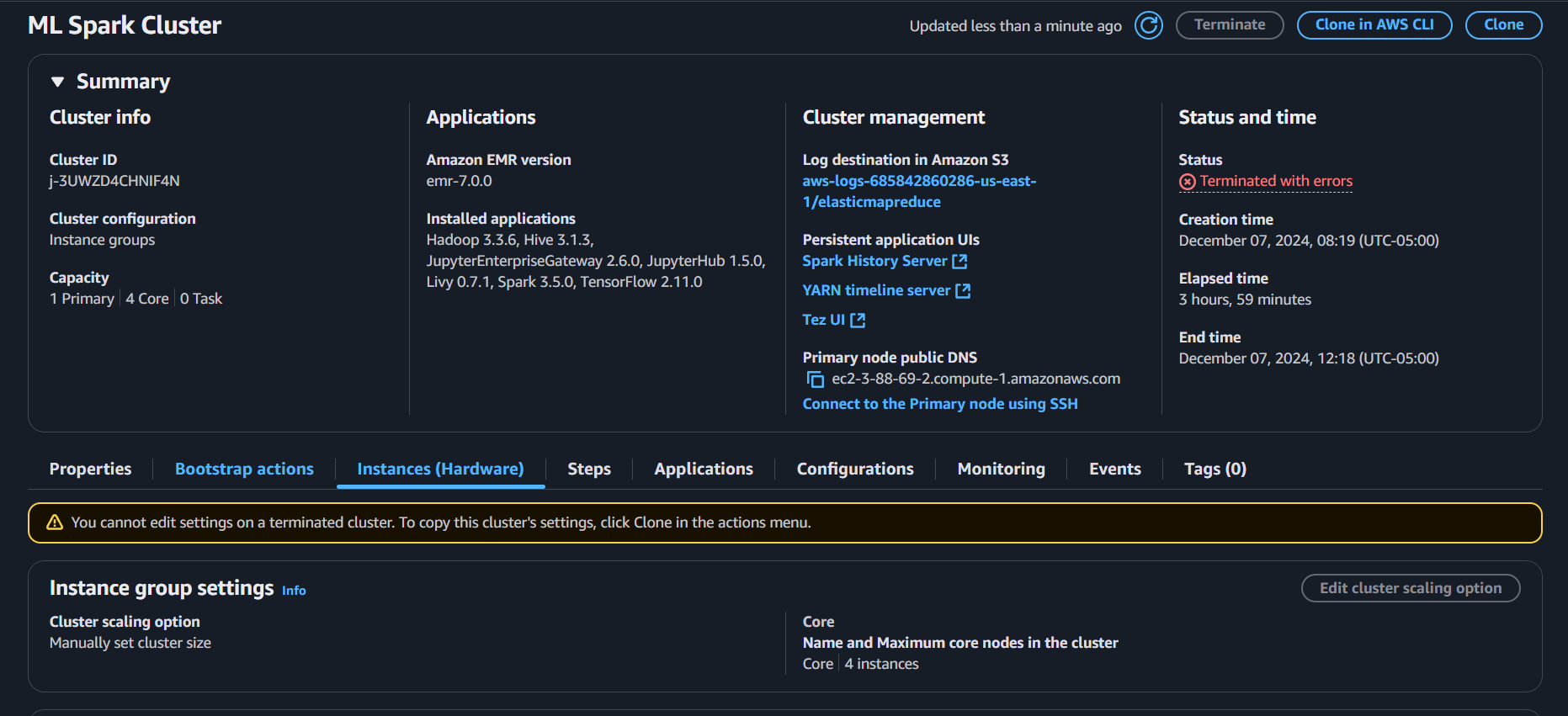
Link to Github:

Overview:

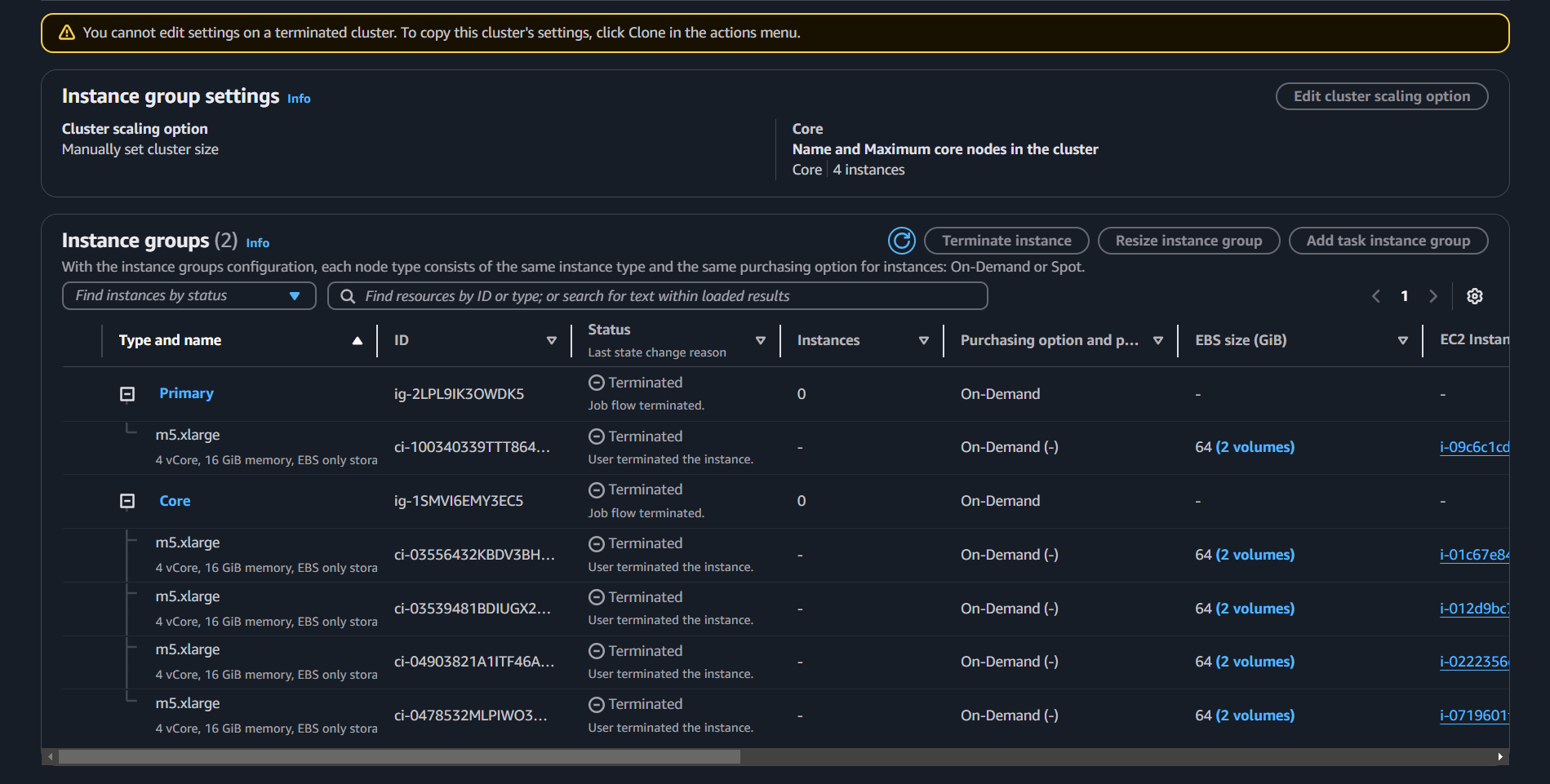
* There are 3 applications here:
  + Pyspark\_training\_v2.py (for training) and Pyspark\_prediction.py (for docker classifier). These are built with pyspark ml models in mind.
  + ClassifierComparison\_updated\_v3.py which is a standalone application that i built meant to be the core of the above but cannot be deployed on top of a spark session so it does all the training and predictions in one as a replacement for either to show work done or at the very least a comparison of methods used.

Creating an Emr cluster:

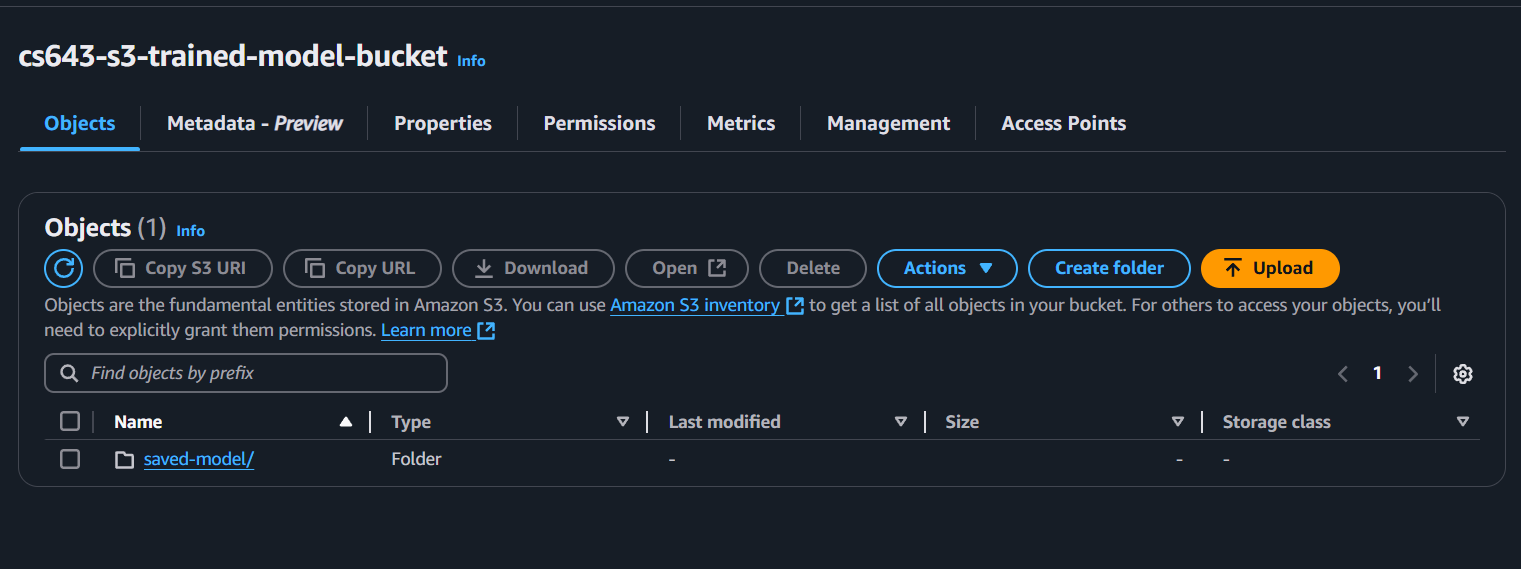
* 1 master needed; specs used are m5.xlarge | default memory
* 4 core instances are needed, for simplicity also use m5.xlarge | default memory
  + Required field changes for emr setup
    - Select amazon linux vm | enable hadoop
    - Key pair = Vockey
    - Iam instance profile = emr\_ec2\_defaultRole
  + Network Settings
    - Allow any inbound connection from port 80, 22, and 443
    - Allow any outbound connections
* IAM Profile
  + Setup by space owner -> in this case the instructor controls this.
* S3 Bucket for storing and pulling data model.



\*note, ignore the termination error, it ran for 4 hours and closed due to student account limits\*

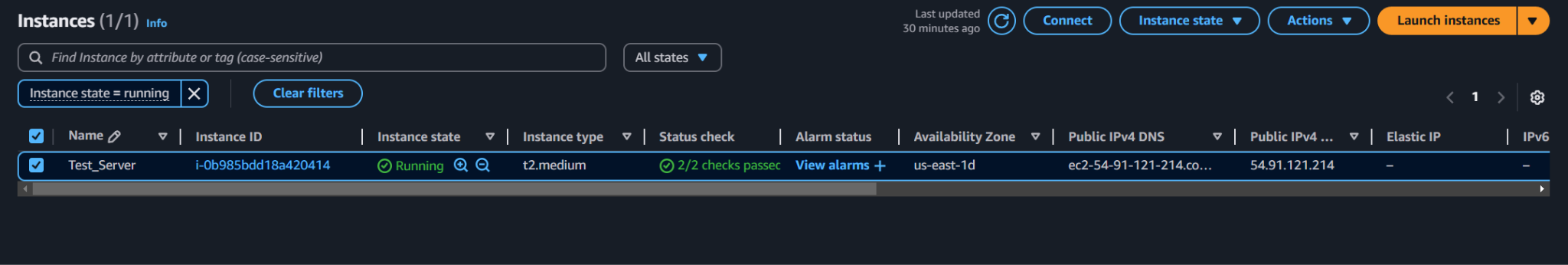


S3 bucket:



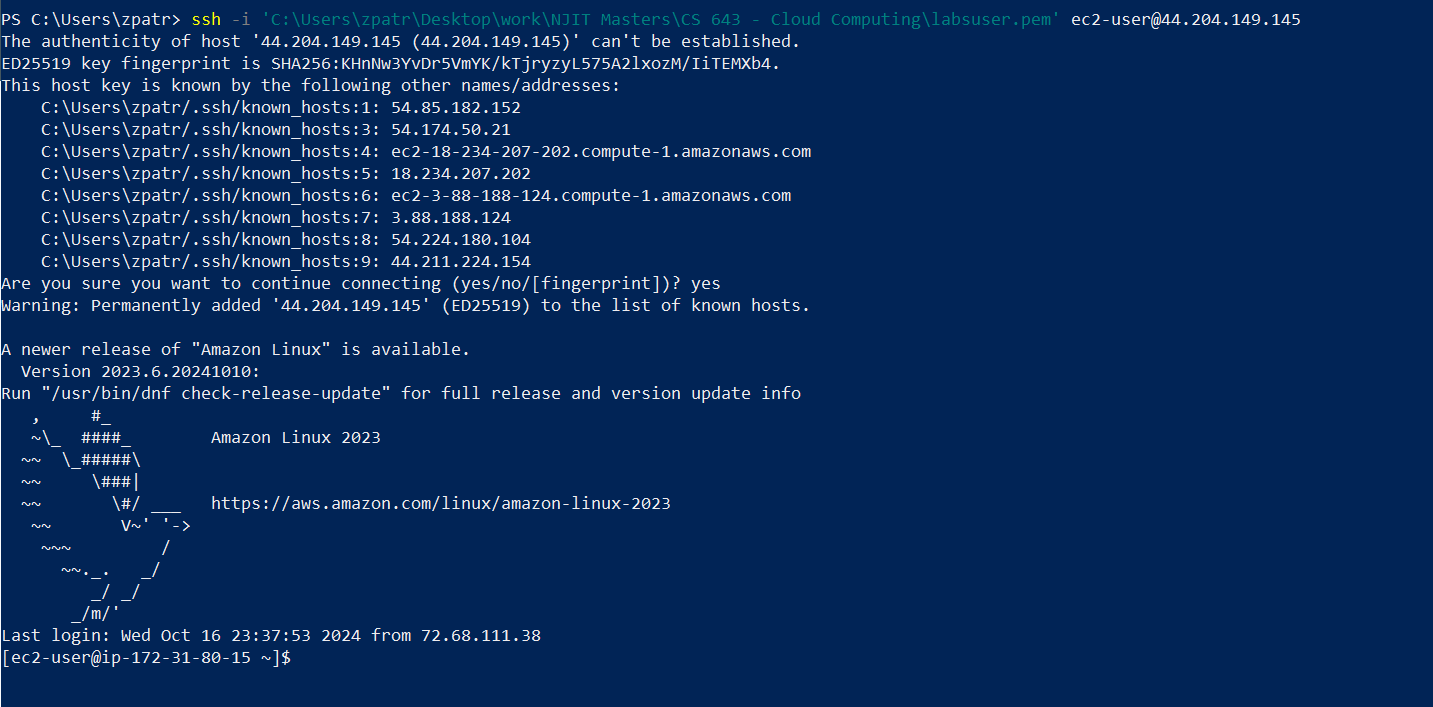
EC2 Instance:

* 1 needed for docker image building



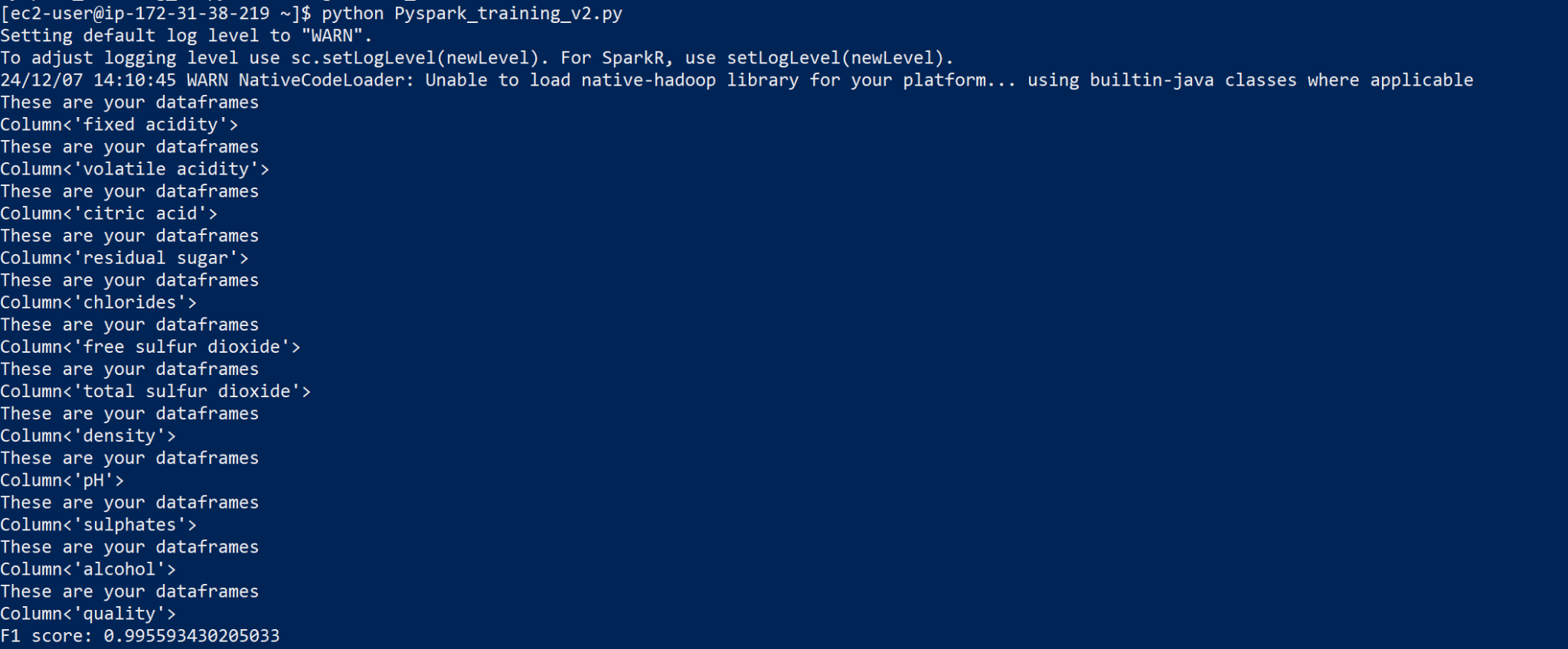
SSH into Emr Master node from powershell to run application:

* Ensure the pem file is the same on both instances
* In my case the command looked like this:



Running the training program:

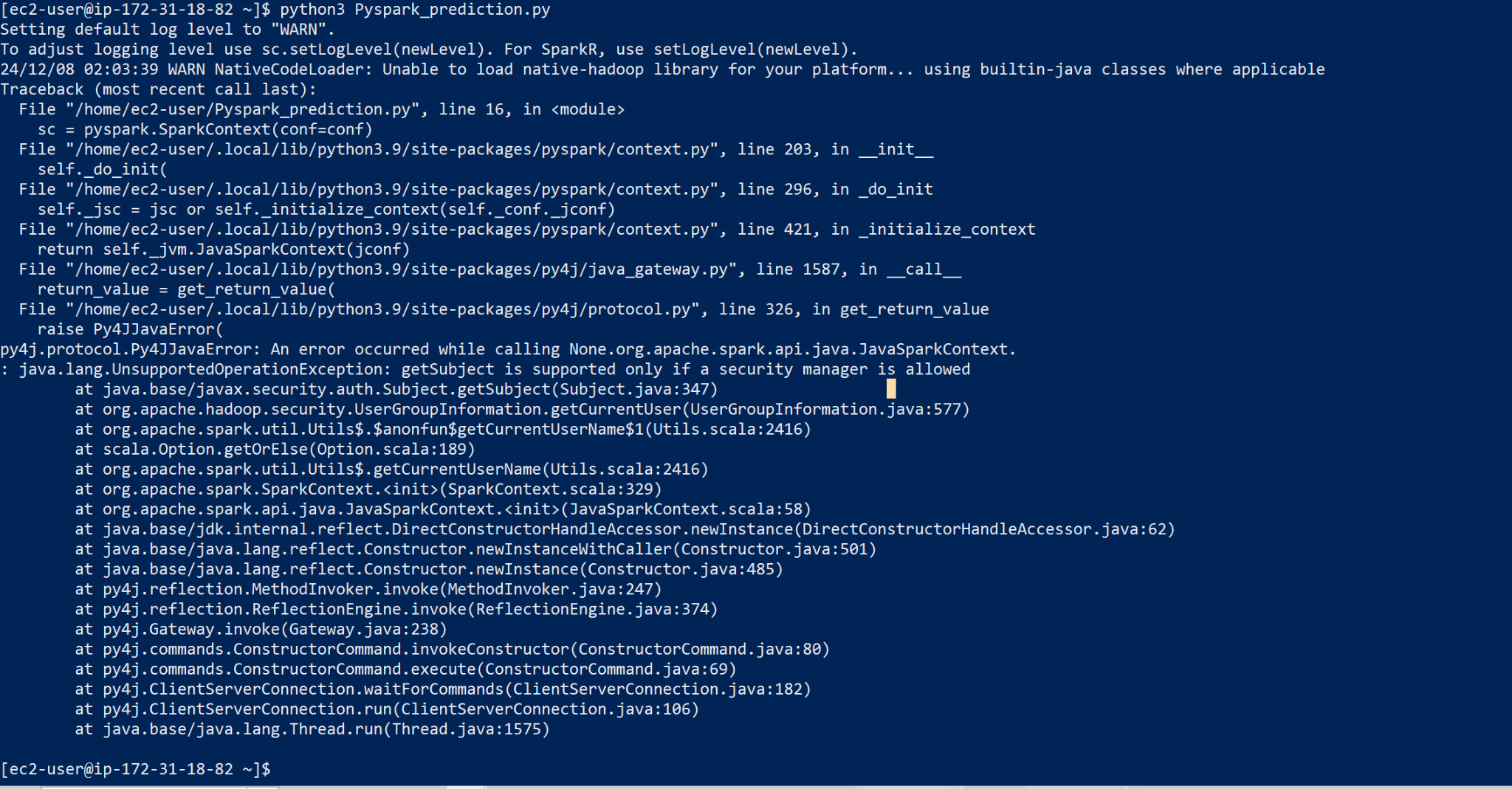
* First and foremost, you will need to ensure pyspark is installed using pip install pyspark
* This application requires both files to be inside the default folder location as well as data being in a CSV document within a specific format. This specific format is that all headers have their own cell with data in their respective column. Please see the example below:
* The first run of this application will create a folder with the saved model in a folder structure with .parquet file format.
  + We will use this paraque file later when we load it for the testing program



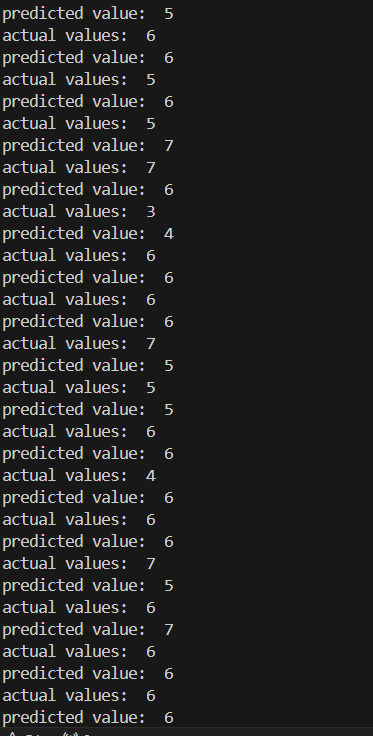
* This will also return a prediction and f1 accuracy score of the predictions for the model for comparison in later evaluation of predictions.
* We will need to have an S3 Bucket to save to and load from.
  + Due to restrictions on students not being able to set policies for S3 buckets, we are going to save it locally in this version of the app, but we have setup an S3 Bucket and commented out tabs for how it should be used if properly permissioned buckets were available.

Running the predictions program:

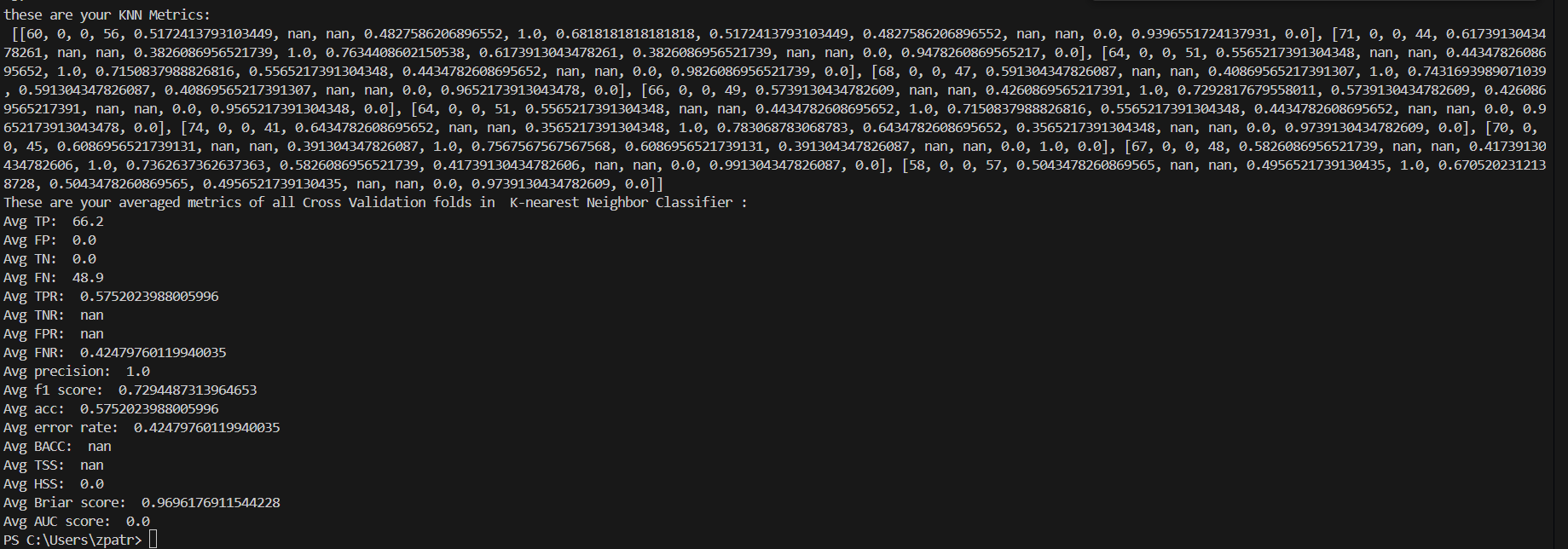
* This needs to be installed in the root directory.
* It will reach out to the s3 bucket, load the model and perform a prediction on it to also show an f1 score
  + Currently this is not working on a docker image or ec2 instance as I could not get the environment properly set up to run it. Seems to be some error related to security manager on it:



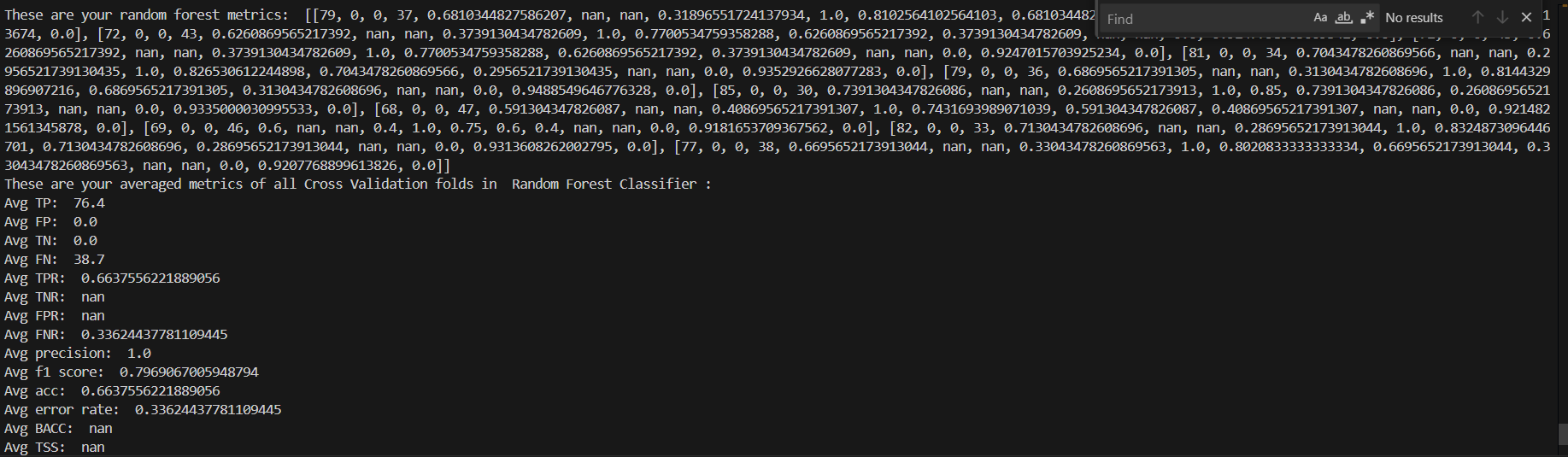
This is output of a local KNN model running this solo on the same dataset that is not within the cloud:



Knn model:



Random forest model:



This is ClassifierComparison\_updated\_v3.py | a standalone application that can be deployed to single machines for using random forest or knn models also using k-fold validation method. Originally I had built this to be the main method of deployment, but it seems we cannot impart a lazy learning model like knn over a pyspark session and is not supported so this was rebuilt into Pyspark\_predictions using the built in pyspark native libraries instead.