K-Means clustering algorithm in MapReduce

Assignment 2

Panourgia Evangelia (t8190130)
Papadatos Iwannis (t8190314)
Professor: Damianos Chatziantwniou

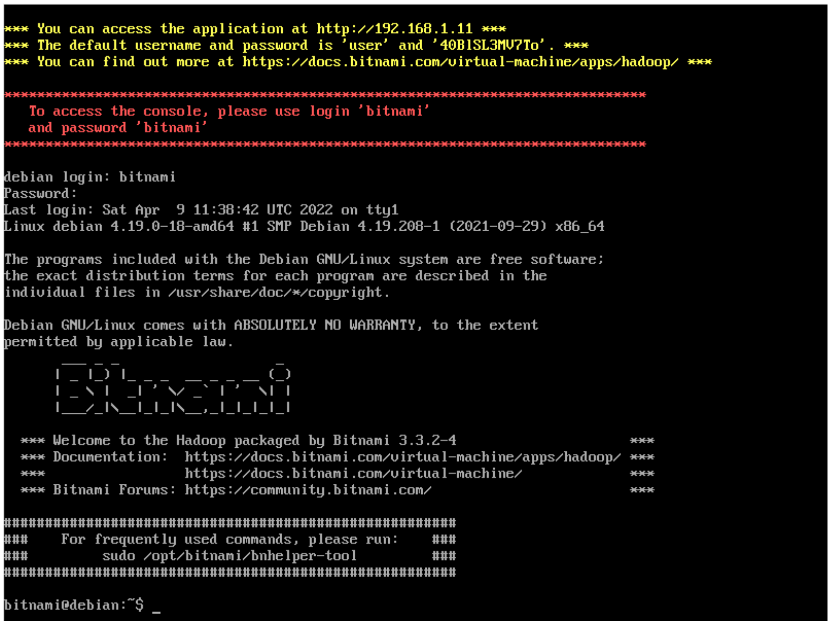
Latest Update: April 9, 2022

School of Management Science and Technology,
Athens University of Economics and Business

Contents

1 Hadoop Installation

As far as the installation of Hadoop, we decided to download Hadoop packaged by Bitnami [1] and Oracle VirtualBox[2]. When the package for Hadoop was downloaded, we import it to Oracle VirtualBox and run it. After, these steps you can see in your pc the image shows in fig. 1. We note that as default username, password is the word bitname(look red letters). Now, you should set the scene by installing git[2] (command : sudo apt-get install git), pip3(sudo apt install python3-pip)[3] and mrjob(pip3 install mrjob)[4]. So, now you have the necessary tools - software so as to run the code.



```
*** You can access the application at http://192.168.1.11 ***
*** The default username and password is 'user' and '40B1SL3MV7To'. ***
*** You can find out more at https://docs.bitnami.com/virtual-machine/apps/hadoop/ ***

*****************************************************************************
   To access the console, please use login 'bitnami'
   and password 'bitnami'
*****************************************************************************

debian login: bitnami
Password:
Last login: Sat Apr  9 11:38:42 UTC 2022 on tty1
Linux debian 4.19.0-18-amd64 #1 SMP Debian 4.19.208-1 (2021-09-29) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
       ___ _ _                   _
      | _ |_) |_ _ _  __ _ _ __ (_)
      | _ \ |  _| ' \/ _` | '  \| |
      |___/_|\__|_||_\__,_|_|_|_|_|

  *** Welcome to the Hadoop packaged by Bitnami 3.3.2-4                  ***
  *** Documentation:  https://docs.bitnami.com/virtual-machine/apps/hadoop/ ***
  ***                 https://docs.bitnami.com/virtual-machine/           ***
  *** Bitnami Forums: https://community.bitnami.com/                      ***

#######################################################
###    For frequently used commands, please run:    ###
###         sudo /opt/bitnami/bnhelper-tool         ###
#######################################################

bitnami@debian:~$ _
```

Figure 1: Install Bitnami

2 Dataset Creation

The file dataGenerate.py was written for generating data - points in the form (x, y). Specifically, it reads data from file centroids.csv which contains three centroids in the form (x, y) and then generate 8000 data points using python library skewnorm[5]. Last but not least, user can see the the generated data graphically by entering python3 generateData.py -d, ehere -d parameters call suitable function to draw[6] data points(see fig 2.).

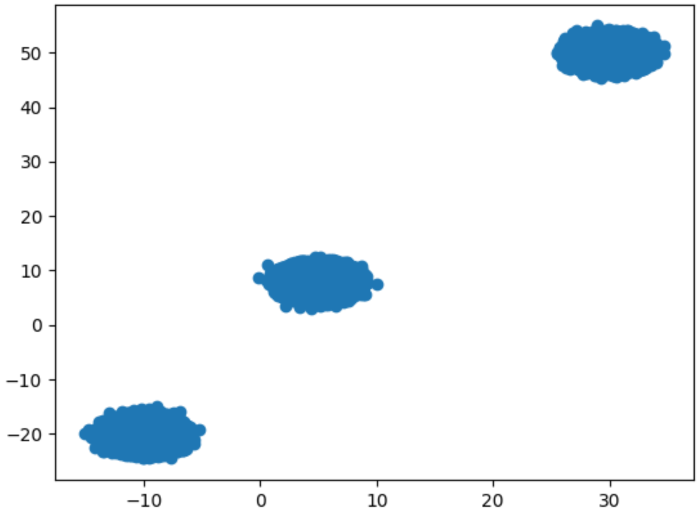


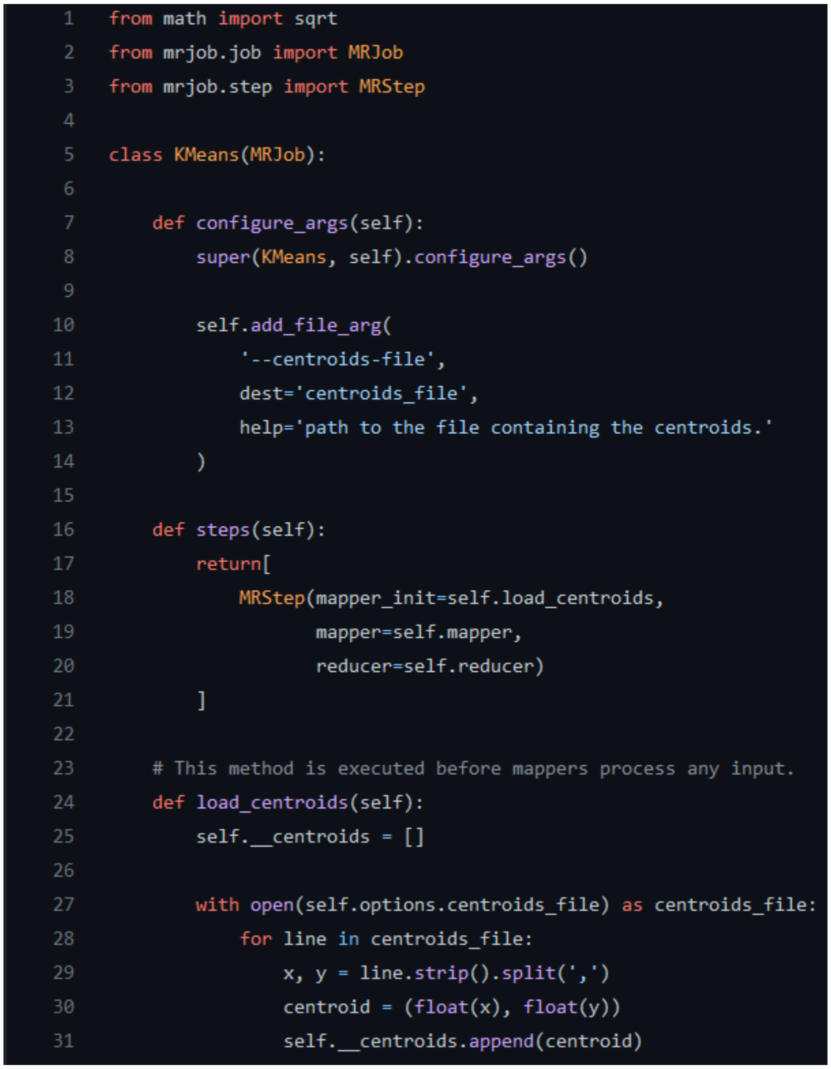
Figure 2: Dataset Creation

Now, we can see the code for data generation.

3 K-Means Clustering Algorithm

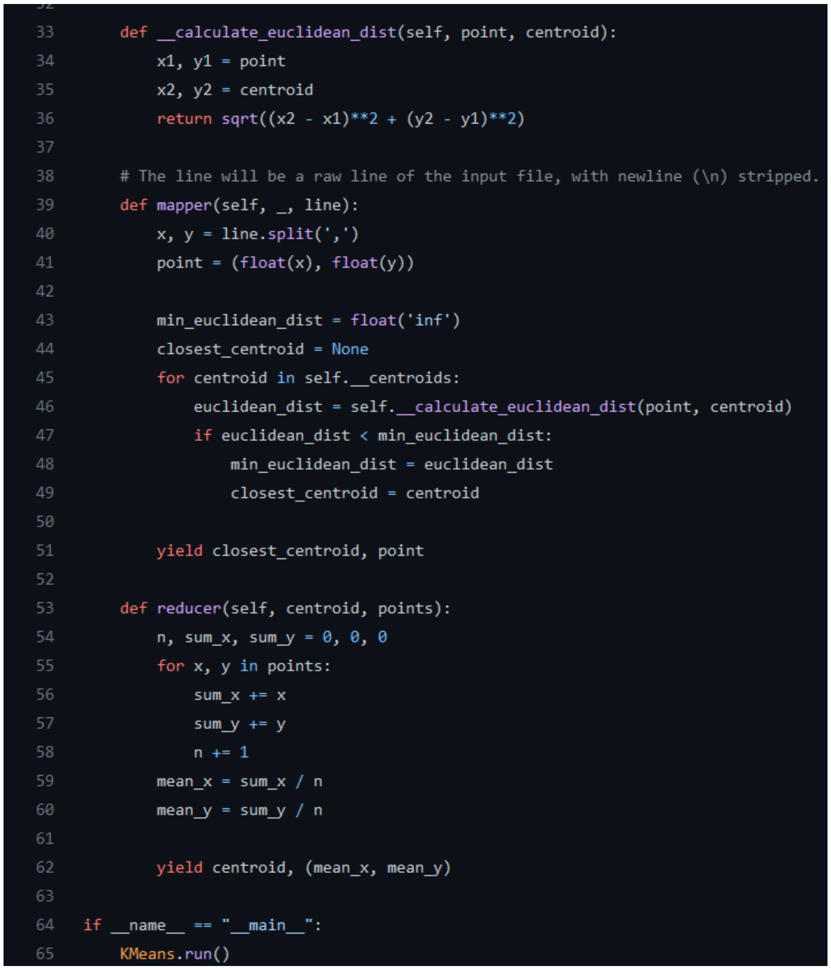
The file kmeans.py was written to implement the algorithm kmeans[6] using MapReduce. For this aim, we used the python library mrjob. Firstly, we defined configuration and steps. Then, we write the function for mapper and reducer. For mapper, we calculate euclidean distance and for each point (x, y) we calculate all distances from all centroids and we hold the minimum distance, so we assign point to suitable centroid. Last but not least, the only work of reducer is to revise cluster centers as mean of assigned observations.

Now, we can see the code for kmeans implementation.



```python
from math import sqrt
from mrjob.job import MRJob
from mrjob.step import MRStep

class KMeans(MRJob):

    def configure_args(self):
        super(KMeans, self).configure_args()

        self.add_file_arg(
            '--centroids-file',
            dest='centroids_file',
            help='path to the file containing the centroids.'
        )

    def steps(self):
        return[
            MRStep(mapper_init=self.load_centroids,
                   mapper=self.mapper,
                   reducer=self.reducer)
        ]

    # This method is executed before mappers process any input.
    def load_centroids(self):
        self.__centroids = []

        with open(self.options.centroids_file) as centroids_file:
            for line in centroids_file:
                x, y = line.strip().split(',')
                centroid = (float(x), float(y))
                self.__centroids.append(centroid)
```

Figure 3: kmeans implementation



```
    def __calculate_euclidean_dist(self, point, centroid):
        x1, y1 = point
        x2, y2 = centroid
        return sqrt((x2 - x1)**2 + (y2 - y1)**2)

    # The line will be a raw line of the input file, with newline (\n) stripped.
    def mapper(self, _, line):
        x, y = line.split(',')
        point = (float(x), float(y))

        min_euclidean_dist = float('inf')
        closest_centroid = None
        for centroid in self.__centroids:
            euclidean_dist = self.__calculate_euclidean_dist(point, centroid)
            if euclidean_dist < min_euclidean_dist:
                min_euclidean_dist = euclidean_dist
                closest_centroid = centroid

        yield closest_centroid, point

    def reducer(self, centroid, points):
        n, sum_x, sum_y = 0, 0, 0
        for x, y in points:
            sum_x += x
            sum_y += y
            n += 1
        mean_x = sum_x / n
        mean_y = sum_y / n

        yield centroid, (mean_x, mean_y)

if __name__ == "__main__":
    KMeans.run()
```

Figure 4: kmeans implementation

3.1 HDFS

3.2 Run

So, git clone + commands

References

[1] Hadoop packaged by Bitnami, ***https://bitnami.com/stack/hadoop/virtual-machine***

[2] Oracle VirtualBox, ***https://www.virtualbox.org/***

[3] Install pip3, ***https://linuxize.com/post/how-to-install-pip-on-ubuntu-18.04/***

[4] Install mrjob, ***https://pypi.org/project/mrjob/***

[5] skewnorm, ***https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.skewnorm.html***

[6] matplotlib, ***https://matplotlib.org/***