Università di Pisa

SCUOLA DI INGEGNERIA

Corso di Laurea in Artificial Intelligence and Data Engineering

Kmeans

Candidati

Alice Nannini
Fabio Malloggi
Marco Parola
Stefano Poleggi

Relatore

Dr. Nicola Tonellotto

Anno Accademico 2019–2020

Contents

1 Introduzione

Le due implementazioni dell'algoritmo kmeans sviluppate devono essere eseguite con i seguenti input:

- File contenente il dataset
- Numero di centroidi/cluster
- Directory di output
- Numerosità di campioni nel dataset (l'algoritmo può essere eseguito facendo l'ipotesi di conoscere questo valore)

La terminazione dell'algoritmo può avvenire a causa di due eventi:

- Si è superato una threshold relativa al numero di iterazioni che possono essere eseguite
- I centri calcolati al passo i-esimo e al passo i+1-esimo non discostano oltre una certa threshold (norma euclidea)

2 Dataset

I dataset per i test finali sono stati generati con un script python mostrato in seguito ed hanno il seguente formato *'dataset_numPoints_kClusters_dimPoints'*.

```
import random

# inputs: n (records), k (clusters), d (dimensions)
numPoints = [1000,10000,100000]
kClusters = [7,13]
dimPoints = [3,7]


for n in numPoints:
    for k in kClusters:
        for d in dimPoints:
            # open a new file
            f = open("data/dataset_"+str(n)+"_"+str(k)+"_"+str(d)+".txt", "a")

            # compute the interval for creating the clusters
            interval = round(n/(2*k))
            count = 0
            print("dataset_"+str(n)+"_"+str(k)+"_"+str(d)+"; int: "+str(interval))

            # compute each point
            for i in range(n):
                if( (i%interval)==0 and i!=0):
                    count = count + 2

                x = ""
                for j in range(d):
```

```
            x = x + str( interval*count + random.random()*interval )
            x = x + " "
        x = x + "\n"
        # write the new point coordinates in the file
        f.write(x)

f.close()
```

Lista dei file per il dataset genereati dal precedente codice:

- dataset_100000_13_3.txt
- dataset_100000_13_7.txt
- dataset_100000_7_3.txt
- dataset_100000_7_7.txt
- dataset_10000_13_3.txt
- dataset_10000_13_7.txt
- dataset_10000_7_3.txt
- dataset_10000_7_7.txt
- dataset_1000_13_3.txt
- dataset_1000_13_7.txt
- dataset_1000_7_3.txt
- dataset_1000_7_7.txt

3 Implementazione Hadoop

4 Implementazione Spark

5 Test e risultati