Analysis of Clustering techniques on HPCM cluster

Instructor: Dr. LONGZHUANG LI

BY

Vinay Datta Pinnaka
Naveen Chada

To evaluate clustering mechanisms, we have selected two clustering algorithms of our choice:

1. K – Means Clustering
2. Gaussian Mixture Model

1. Data Preprocessing

- The provided dataset Heterogeneity Activity Recognition is combination of both Numerical and Nominal attributes. But for clustering techniques categorical data is binaries to perform any distance calculation athematic.
- In Phones_accelerometer.csv the following attributes are nominal; Index, User, Mobile, Device and gt.
- In Map-reduce framework reader obtain the data and store in vector format when map function occurs. In map function we have tweaked the code to convert the nominal data to binary.
- The following pseudo code explains the process.

data. Map (

- ⇨ **Parse the single instance of the data file**
- ⇨ **If decimal**
  - o **Push decimal value to a vector**
- ⇨ **Else not decimal**
  - o **If string == "nexus4"**
    - ▪ **Push 1.0 to vector**
  - o **Else**
    - ▪ **Push 0.0 to vector)**

2. K - Means Clustering

- **Quality measure**

  Quality of the cluster is measured using Sum of squared error and the running time. From the following table we can conclude that if we increase the K value the quality of the cluster increased at the cost of increased running time.

K value	Sum of squared error	Running time
2	1.039472877889469E8	12 min
3	6.653591045777614E7	13 min
6	4.293524216600166E7	> 13 min around 14 min

Table 1: Squared error and Running time for three k values 2, 3 and 6

- **Parameter chosen**

  Two parameters chosen for the K-means clustering are K value and no. of iterations.

  K = 2, 3, 6 no. of iterations = 20

- **Results Summary**

```
[vpinnaka@hpcm K-means]$ /cm/shared/apps/hadoop/spark-2.0.0-bin-hadoop2.6/bin/spark-submit --class my.spark.JavaKMeans
Example --master local[8]   target/first-example-1.0-SNAPSHOT.jar
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
16/11/17 23:50:32 INFO SparkContext: Running Spark version 2.0.0
16/11/17 23:50:33 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java
classes where applicable
16/11/17 23:50:33 INFO SecurityManager: Changing view acls to: vpinnaka
16/11/17 23:50:33 INFO SecurityManager: Changing modify acls to: vpinnaka
16/11/17 23:50:33 INFO SecurityManager: Changing view acls groups to:
16/11/17 23:50:33 INFO SecurityManager: Changing modify acls groups to:
```

Fig 1: Running K – Means cluster in HPCM using local node

```
16/11/18 00:09:07 INFO BlockManager: Removing RDD 3
Cluster centers:
 [0.0,0.0,0.13670416167168986,0.6924350461207206,9.88654745971075,0.1799614239851888,0.07797975470957003,0.19540606460
147977,0.15977618616129952,0.12689638345412804,0.10985690131221616,0.15012328577611767]
 [0.0,0.0,2.4667378201172365,0.5745369725407995,6.822808291342862,0.055848377047730415,0.31710119006672627,0.023186646
88135693,0.18565311176406998,0.15746592438998022,0.15429839129708428,0.10644611287453688]
16/11/18 00:09:07 INFO MemoryStore: Block broadcast_36 stored as values in memory (estimated size 376.0 B, free 91.6 M
B)
16/11/18 00:09:07 INFO MemoryStore: Block broadcast_36_piece0 stored as bytes in memory (estimated size 547.0 B, free
91.6 MB)
```

Fig 2: Cluster centers obtained

- **Clusters center**

No of clusters: 3
Cluster centers:
[0.0,0.0,0.13670416167168986,0.6924350461207206,9.88654745971075,0.1799614239851888,0.07797975470957003,0.19540606460147977,0.15977618616129952,0.12689638345412804,0.10985690131221616,0.15012328577611767]

[0.0,0.0,2.4667378201172365,0.5745369725407995,6.822808291342862,0.055848377047730415,0.31710119006672627,0.02318664688135693,0.18565311176406998,0.15746592438998022,0.15429839129708428,0.10644611287453688]

Cost: 1.0394728778894691E8
Within Set Sum of Squared Errors = 1.039472877889469E8

No of clusters: 3
Cluster centers:
[0.0,0.0,0.06219152683776111,0.709319344781593,6.938888096256519,0.10558773851278157,5.121309182214717E-6,0.06779613482602424,0.24898415174292782,0.2255553815936246,0.20363203247202666,0.14843919567156702]

[0.0,0.0,0.17037911122392668,0.6676992598736323,10.335784962092696,0.185263009776722,0.09721752874072 63,0.2119186761538 3933,0.1518866779975004,0.10868542754323296,0.09894803845097014,0.14608064133700882]

[0.0,0.0,5.618310022072511,0.47968198704298437,7.8037517573379604,0.04301752580752963,0.7406907425930753,0.024892825110562684,0.04292335660384509,0.04155101272088133,0.03815805037593035,0.06876648678817555]

Cost: 6.6535910457776144E7
Within Set Sum of Squared Errors = 6.653591045777614E7

No of clusters: 6
Cluster centers:
[0.0,0.0,0.2078482181231742,0.20884877582183745,9.925967052580516,0.2120351847044271,0.15212410636566523,0.24013080821412755,0.1015024619257779,0.09051138681577116,0.06546453714268262,0.1382315148315484]

[0.0,0.0,0.08047498236973342,0.625706187120293,5.445337517468857,0.02489580039070961,6.920113517542142E-7,1.0310969141137792E-4,0.3204462366000652,0.25925028874173656,0.27049062912828026,0.12481255142509358]

[0.0,0.0,0.05532521322164563,0.39623038543772937,7.887993259100342,0.15036084955304113,3.0175059503745388E-5,0.11671602619489556,0.2009651603178832,0.20192413842845344,0.16343916192820102,0.16656448851802186]

[0.0,0.0,0.06640278061484987,2.3318814895778974,9.204107741702416,0.1890957171103803,0.0033166430611566086,0.248754589400861,0.14916753213137088,0.14303985123122823,0.1117750050401527,0.15485066202485034]

[0.0,0.0,0.22535020028560557,1.1115591977097832,13.700044948358867,0.05890063537042656,1.963747261799912E-6,2.3564967141598944E-5,0.40453880904619816,0.14609003192071174,0.23756727061713703,0.15287772433112315]

[0.0,0.0,5.631783821033357,0.4799804168841276,7.792288266172938,0.041577285713209296,0.7472997340161932,0.024515835860923332,0.04138195657201415,0.04061165262943744,0.03657117950400309,0.06804235570421951]

Cost: 4.293524216600167E7
Within Set Sum of Squared Errors = 4.293524216600166E7

3. Gaussian Mixture Model

- **Quality measure**
  In Gaussian Mixture model the quality of the cluster is measured based on the weights. If the weight of the Gaussian cluster is high which indicates quality as well.

K value	Weight	Running time
1	0.435494	35 min
2	0.041777	
3	0.276022	
4	0.225889	
5	0.020758	
6	0.000059	

Table 2: Gaussian mixture model

- **Parameter selection**
  Gaussian mixture model provides option to select k value

  K = 6

- **Results Summary**

```
0.0  0.0  0.39867627659312        -0.0205392938283711     ...
0.0  0.0  -0.39867354326513676    0.020517682231444546    ...
0.0  0.0  7.770091688290468E-10   3.155615726229785E-9    ...
weight=0.000059
mu=[0.0,0.0,2.5137208377058964,4.497900014729491,18.41355405096553,0.4493303288799037,1.6881640512840783E-23,9.42408902E-23,5.916519714540902E-4,0.006429414976237 47,0.01084361260489283,0.5325904972576047]
sigma=
0.0  0.0  0.0                   0.0                     ... (12 total)
0.0  0.0  0.0                   0.0                     ...
0.0  0.0  13.604402875677335    -0.6284908621981093     ...
0.0  0.0  -0.6284908621981093   15.592099479567597      ...
0.0  0.0  -5.725260573834225    -8.52306151819237       ...
0.0  0.0  -0.631168778419854    -1.2360194631554573     ...
0.0  0.0  1.1336176001341313E-22  -5.248323690948957E-23  ...
0.0  0.0  -6.161006654975788E-23  -2.3743128411855412E-22  ...
0.0  0.0  0.0011257802243621164  0.003628564067481139    ...
0.0  0.0  -0.008914773353775067  0.0607238367574983      ...
0.0  0.0  0.010993223814924694   0.08355677434619532     ...
0.0  0.0  0.6285037265507403     1.0890750619440335      ...
16/11/17 04:23:39 INFO SparkUI: Stopped Spark web UI at http://10.20.1.189:4040
16/11/17 04:23:39 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
16/11/17 04:23:39 INFO MemoryStore: MemoryStore cleared
16/11/17 04:23:39 INFO BlockManager: BlockManager stopped
16/11/17 04:23:39 INFO BlockManagerMaster: BlockManagerMaster stopped
16/11/17 04:23:39 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
16/11/17 04:23:39 INFO SparkContext: Successfully stopped SparkContext
16/11/17 04:23:39 INFO ShutdownHookManager: Shutdown hook called
16/11/17 04:23:39 INFO ShutdownHookManager: Deleting directory /tmp/spark-6b4c121a-557c-4e26-9d11-bb9ee97c24d6
[vpinnaka@hpcm Gaussian]$ ls
```

Fig 3: Gaussian mixture model output

- **Cluster weights**

  weight=0.435494
  mu=[0.0,0.0,1.5363085949158728,0.5148393829650462,8.995378164569221,0.324367507699225,0.3501577841739908,0.3254722386481933,8.081489277359751E-8,8.782062884445425E-7,1.4811501219137795E-6,9.093275042993156E-12]
  weight=0.041777
  mu=[0.0,0.0,2.646532735374572 5,0.5209536504498096,9.50479990250662,1.233513111981035E-7,2.4037163060825356E-26,1.3452366428206602E-25,0.48453732361685586,9.154613620162105E-6,1.5439831458725545E-5,0.515437653175987]
  weight=0.276022
  mu=[0.0,0.0,9.366633599399334E-5,0.8101843117530264,8.768720330141125,1.8669738695535586E-8,3.638125521072608E-27,2.0360721228804747E-26,0.5347277914233577,0.46526980678405405,2.3368832972887593E-6,1.4346906671530781E-11]
  weight=0.225889
  mu=[0.0,0.0,1.2443142229139623E-4,0.7764898199827016,8.926679921389685,2.281326885288173E-8,4.445564931908055E-27,2.4879545183049171E-26,1.558040081584443E-7,1.6931045137055826E-6,0.49112890436353884,0.5088691674303406]
  weight=0.020758
  mu=[0.0,0.0,3.995519042050 4603,0.5022954293977818,8.644899090131615,2.482588438719335E-7,4.837758312806161E-26,2.7074450237862813E-25,1.6954923551493908E-6,0.3851891814631407,0.6148082599202364,1.907764496063604E-10]
  weight=0.000059
  mu=[0.0,0.0,2.5137208377058964,4.497900014729491,18.41355405096553,0.4493303288799037,1.6881640512840783E-23,9.44778772408902E-23,5.916519714540902E-4,0.00642941497623747,0.01084361260489283,0.5325904972576047]

4. Comparison of K-means and Gaussian mixture model

When comparing two clustering techniques K-means and Gaussian we can conclude that K-means is best in running time but when comparing the quality Gaussian is best because weights are calculated on convergance.