Seminar on Communication Engineering(Dec. 3rd)

updated: Oct. 10, 2022

List of Contents

Abstract

- The objective of this class is to allow students to acquire experience of clustering.
- Clustering methods: Hard c-Means (HCM) and Fuzzy c-Means (FCM).
- Clustering target: artificial datasets and real datasets.
- Clustering tool: C++ programming language.
- Environment: repl.it, a Web-based integrated developement environment, Linux-like interface.
- Plan
  1. Short lecture: getting minimal knowledge for clustering.
  2. Environment setting: make an account for repl.it, make a project on replt.it, and upload some files..
  3. Description of assignment (homework) and lecture of clustering theory
  4. Actual experience of clustering:
     i. Artificial dataset as a clustering target (including how to make artificial datasets by yourself)
     ii. Clustering with HCM
       - Check the clustering accuracy using contingency tables and the adjusted rand index (ARI value).
       - Visualizing clustering results and a classification function using the "`gnuplot`" application.
       - How to cluster other datasets by yourself
     iii. Clustering with FCM
       - Checking the clusteirng accuracy using contingency tables and the ARI value.
       - Visualizing clustering results and a classification function using the gnuplot application.
       - How to cluster other datasets by yourself.
     iv. Clustering Real Datasets
     v. End by 15:00 at latest (classes are never rushed and rests will be omitted).

Assignment (Homework)

Upload your report bia Scombz.

- Hard Deadline: 24:00 JST, Dec. 10, 2022.
- The report must include
  - Your name and e-mail address
  - Description of:
    - What the teacher taught, and
    - What you learned,

    on more than single page.
  - As options:
    - You can add figures prepared during the time and you must describe what these figures mean;
    - You can add additional work, such as other artificial datasets that you have prepared, the clustering results for these datasets, the clustering results obtained using FCM with other fuzzifier parameters, and the clustering results for other real datasets.
- If the minimum requirements above are satisfied, you will receive 80 points for this class.

Theory

- Download this file.
- Focus on p.7, 37, 45, 81, and 82.

Environment Setting

- Start any WWW browser (under SIT network or SRAS) to access
  web3.kanz.ice.shibaura-it.ac.jp/
  and then click the link "Seminar on Communication Engineering (Dec. 3rd)". You will then see this page. Next, use your browser to read this page.
- Change Display Language if necessary.
  - Save this file.
  - Open the saved document.
- Access repl.it on your web brouser.
- Sign up, e.g. using your Google account.
- Create a C++ Repl with a title.
- Download Seminar2022Kanzawa.zip, extract it, and upload all files to the Repl.

Artificial Data

Check contents

- Check `2d-Gaussian-2clusters.dat`. This file is an artificial dataset for clustering, which contains 100 two-dimensional (2D) points, where 50 points were generated by Gaussian sampling with a mean of (-1, -1) and a stddev of (0.5,0.5), and the remainder (50) were also generated by Gaussian sampling with the same stddev but with a mean of (1,1).
- View the content of this file:
  1. Click "`2d-Gaussian-2clusters.dat` on "Files" of repl.it We can then see the following.

```
100     2
0.711235        -1.69953
-0.642839       -1.28753
-1.09038        -1.06458
-1.21817        -0.259876
-1.2059 -0.842923
-0.745885       -1.00911
-0.746496       -0.766282
-1.01819        -1.13348
-0.619797       -0.99589
-1.23229        -0.700303
-1.09537        -0.960197
-1.0382 -1.05888
-1.18212        -1.56641
-0.67809        -0.727345
-1.56106        -0.891143
-1.23858        -0.867031
-0.250603       -0.839113
-0.999856       -0.91277
-0.867204       -1.12405
.....
```

  2. The first line, "`100  2`," shows that
     - this file contains 100 objects(datum), and
     - each point is in 2D space.

     The remainder of the file comprises 100 lines (i.e., each line corresponds to each object), where each line has two values (i.e., each column corresponds to each axis).
- Visualize this file using the "`gnuplot`" application.
  1. Start the "`gnuplot`" application by typing

     `gnuplot`

     in the "Console" of repl.it, and select "gnuplot.out"
     We can then see the "`gnuplot`" command prompt "gnuplot>", which shows that we are in the "`gnuplot`" application.
  2. Plot the contents of the file by typing the following.

     `plot "2d-Gaussian-2clusters.dat" every ::1`

     We can then see 100 points in 2D space in a new window, here, the option "`every ::1`" restricts the plotting target to the lines after the first line in the file. Without this option, the first line is also the plotting target. Thus, the point (100, 2), which simply describes the number of objects and the dimensions of object, is also plotted, and this not desireble.
  3. To arrange the grid sizes for both the horizontal and vertical axes, type the following.

     `set size ratio -1`

     `replot`
  4. Therefore, we can find two clusters, each of which contains 50 points (we do not need to count:)). The clusters have centers near (-1,1) and (1,1), respectively.
  5. To save this plot, type

     `set terminal pdf`

     (seting the format to PDF)

     `set output "2d-Gaussian-2clusters.pdf"`

     (setting the filename)

     `replot`

     (plotting the previous plot)

     `set output`

(outputting to a file, and close the file)

```
quit
```

(exiting the "gnuplot" application)

6. Click "2d-Gaussian-2clusters.pdf" on "Files" of repl.it. We then see the contents of the "2d-Gaussian-2clusters.pdf" file, which can be used to prepare your report (you can download this PDF file to the local machine.

Preparing by yourself

- Prepare another artificial dataset.
  1. Click randGaussianMain.cxx on "Files" of repl.it.
  2. We can find the place that indicates the data structure in l.9 as
     ```
     const int dimension=2, eachDataNum=50, clusterNum=2;
     ```
     where:
     - dimension: the dimension of each object (the number of elements);
     - eachDataNum: the number of objects in each cluster; and
     - clusterNum: the number of clusters.

     This data structure is exactly the same as that for the previously checked file "2d-Gaussian-2clusters.dat".
  3. We can also find the place that indicates the cluster information in ll.10-13, where:
     - means: means of the Gaussian distributions corresponding to the clusters; and
     - stddev: stddevs of the Gaussian distributions corresponding to the clusters.

     This cluster-information is exactly the same as that for the previously checked file "2d-Gaussian-2clusters.dat."
  4. We can prepare another dataset by changing these values. For example, change the following:
     - clusterNum: 3 from 2;
     - means: add {-1, 1}; and
     - stddevs: add {0.3, 0.3}.
  5. Compile this file by typing
     ```
     g++ -std=c++11 randGaussianMain.cxx
     ```
     If you obtain an error message, re-edit the file (or ask the teacher).
  6. Produce the dataset by typing
     ```
     ./a.out
     ```
     We then find that the data information is streaming in the Terminal. To save this stream, type
     ```
     ./a.out > 2d-Gaussian-3clusters.dat
     ```
  7. Confirm that the prepared file "2d-Gaussian-3clusters.dat" contains the desired information in a similar manner to the previously confirmed file "2d-Gaussian-2clusters.dat."

Clustering

HCM

Trying HCM

1. We use the following files:
   - hcm_main_2d-Gaussian-2clusters.cxx (main C++-program, where we edit this file to cluster various datasets),
   - hcm.h (HCM C++ class declaration),
   - hcm.cxx (HCM C++ class definition),
   - vector.h (Vector C++ class declaration, which is used for the HCM and VectorArray C++ classes),
   - vector.cxx (Vector C++ class definition),
   - vectorArray.h (VectorArray C++ class declaration, which is used used in the HCM C++ class),
   - vectorArray.cxx (VectorArray C++ class definition),
   - 2d-Gaussian-2clusters.correctCrispMembership (correct clustering information for the "2d-Gaussian-2cluster.dat" file, which is used for producing contingency tables and ARI values, where these are written in the main C++ program file defined above).
2. Compile these file by typing
   ```
   g++ -DCLASSIFICATION_FUNCTION -DCHECK_ANSWER -std=c++11 hcm.cxx vectorArray.cxx vector.cxx hcm_main_2d-Gaussian-2c
   ```
   If you obtain error messages, call the teacher.
3. Execute by typing
   ```
   ./a.out
   ```
   We can then see the contingency table and the ARI value, which shows that HCM produces the exact result.
4. Furthermore, executing the clustering process produces the following three files.
   - HCM-2d-Gaussian-2clusters.result_centers
   - HCM-2d-Gaussian-2clusters.result_membership
   - HCM-2d-Gaussian-2clusters.result_classificationFunction

   We can confirm these files in "Files" of repl.it.
5. To confirm the contents of the "HCM-2d-Gaussian-2clusters.result_centers" file, click HCM-2d-Gaussian-2clusters.result_centers on "Files" of repl.it.
   We can then see that there are two lines with two columns, which is the same format as the dataset file (after the first line in "2d-Gaussian-2clusters.dat"), which we can use to show that the cluster centers are near the means of the Gaussian distributions specified for the "randGaussianMain.cxx" file.
6. Visualize the "HCM-2d-Gaussian-2clusters.result_centers" file as follows.
   i. Start the "gnuplot" application by typing

```
gnuplot
```

ii. In the "gnuplot" command prompt, type

```
plot "2d-Gaussian-2clusters.dat" every ::1, "HCM-2d-Gaussian-2clusters.result_centers"
```

iii. The cluster centers obtained are nearly centralin bothclusters, where the objects are plotted by red plus symbols and the cluster centers by green cross symbols. Note that the plot command can plot more one file by using comma symbols.

iv. To save this plot, type

```
set size ratio -1
```

(setting the axes ratio to 1)

```
set terminal pdf
```

(setting the format to PDF)

```
set output "HCM-2d-Gaussian-2clusters.result_centers.pdf"
```

(setting the filename)

```
replot
```

(plotting the previous plot)

```
set output
```

(outputting to a file and closing the file)

```
quit
```

(exiting the "gnuplot" application).

v. Click "HCM-2d-Gaussian-2clusters.result_centers.pdf" on "Files" of repl.it to see the contents, which can be used to prepare your report (you can copy this PDF file to the local machine.

7. To confirm the contents of the "HCM-2d-Gaussian-2clusters.result_membership" file, click "HCM-2d-Gaussian-2clusters.result_membership" on "Files" of repl.it. Many lines are then visible with four columns.
   - Each line shows the membership information for each object.
   - The columns depict the following:
     - First column: first element value for an object,
     - Secon column: second element value for an object,
     - Third column: membership value to which the object belongs in the first cluster, and
     - Fourth column: membership value to which the object belongs in the seconf cluster.

   For example, the first line shows that an object (-1.18464, -1.2737) belongs to the first cluster at a probability of 100% and to the second cluster at 0%.
8. Visualize the "HCM-2d-Gaussian-2clusters.result_membership" file as follows.

i. Start the "gnuplot" application by typing

```
gnuplot
```

ii. In the "gnuplot" command prompt, type

```
plot "< awk '{if($3>$4){print $1, $2;}}' HCM-2d-Gaussian-2clusters.result_membership", "< awk '{if($4>$3)
```

iii. At the lower left, objects are plotted in red and at the upper right, objects are plotted in green.
   - The red plots represent the previous part of the "gnuplot" command

     ```
     "< awk '{if($3>$4){print $1, $2;}}' HCM-2d-Gaussian-2clusters.result_membership"
     ```

     where
     - The content of the "HCM-2d-Gaussian-2clusters.result_membership" file is given to the awk command, and
     - The awk command executes the procedure: "if the 3rd column is greater than the 4th column, then plot the 1st and the 2nd columns."

     Thus, the red points belongs to the first cluster.
   - The green plots present the latter part of the "gnuplot" command

     ```
     "< awk '{if($3<$4){print $1, $2;}}' HCM-2d-Gaussian-2clusters.result_membership"
     ```

     where
     - the content of the "HCM-2d-Gaussian-2clusters.result_membership" file is given to the awk command, and
     - The awk command executes the procedure: "if the 3rd column is less than the 4th column, then plot the 1st and the 2nd columns."

     Thus, the green points belongs to the second cluster.

   We can see that the clustering procedure is successful.

iv. To save this plot, type

```
set size ratio -1
```

(setting the axes ratio to 1)

```
set terminal pdf
```

(setting the format to PDF)

```
set output "HCM-2d-Gaussian-2clusters.result_membership.pdf"
```

(setting the filename)

```
replot
```

(plotting the previous plot)

```
set output
```

(outputting to a file, and closing the file)

```
quit
```

(exiting the "gnuplot" application).

v. Click "HCM-2d-Gaussian-2clusters.result_membership.pdf" on "Files" of repl.it. to see the contens, which can be used to prepare your report (you can download this PDF file to the local machine.

9. To confirm the content of the "HCM-2d-Gaussian-2clusters.result_classificationFunction" file, click "HCM-2d-Gaussian-2clusters.result_classificationFunction" on "Files" of repl.it. Many lines are then visible with five columns.
   - We are not concerned with the number of lines, which do not correspond to any objects that need to be clustered, but instead they reporesent other objects that need to be classified after clustering.
   - The columns show the following:
     - First column: first element value for a new object,
     - Second column: second element value for a new object,
     - Third column: membership value for which the object belongs to the first cluster,
     - Fourth column: membership value for which the object belongs to the second cluster, and
     - Fifth column: maximal membership value among those in the same line.

   For example, the first line shows that a new object belongs to the 1st cluster at a probability of 100% and to the second cluster at 0%, and the maximal membership value is 1.
10. Visualize the "HCM-2d-Gaussian-2clusters.result_classificationFunction" file as follows.
    i. Start the "gnuplot" application by typing

    ```
    gnuplot
    ```

    ii. In the "gnuplot" command prompt, type

    ```
    splot "HCM-2d-Gaussian-2clusters.result_classificationFunction" using 1:2:3 with lines, "HCM-2d-Gaussian-2clust
    ```

    Two surfaces are then visible, i.e., red and green, and two types of points, i.e., blue and pink. To remove hidden surfaces, type

    ```
    set hidden3d
    ```

    ```
    replot
    ```

    - The red plots present the first part of the "gnuplot" command

      ```
      "HCM-2d-Gaussian-2clusters.result_classificationFunction" using 1:2:3 with lines
      ```

      where
      - The splot command obtains a three-dimensional (3D) plot with three columns in each line.
      - The option of "using 1:2:3" means that the splot command uses the first, second, and third columns (the first element value for a new object, the second element value for new object, and the membership value for which the object belongs to the first cluster).
      - The option "with lines" means that the plotted points are connected by lines, i.e. to make a surface.

      Thus, a red surface represents the classification function for the first cluster, where the bottom plane shows the data space and the vertical axis shows the membership value.
    - The blue plots represent for the second part of the "gnuplot" command

      ```
      "HCM-2d-Gaussian-2clusters.result_classificationFunction" using 1:2:4 with lines
      ```

      where
      - The "splot" command is used to obtain a 3D plot with three columns in each line.
      - The option "using 1:2:4" means that the "splot" command uses the first, second and fourth columns (first element value for a new object, second element value for new object, and the membership value for which the object belongs to second cluster).
      - The option "with lines" means that the plotted points are connected by lines, i.e. to makw a surface.

      Thus, the blue surface represents the classification function for the second cluster, where the bottom plane denotes the data space and the vertical axis shows the membership value.
    - The sky blue plots represent the third part of the "gnuplot" command

      ```
      "< awk '{if($3>$4){print $1, $2, 0;}}' HCM-2d-Gaussian-2clusters.result_membership"
      ```

      where
      - The content of the "HCM-2d-Gaussian-2clusters.result_membership" file is given to the awk command, and
      - The awk command executes the procedure: "if the 3rd column is greater than the 4th column, then plot the 1st column, the 2nd column, and 0."

      Thus, the sky blue points belong to the first cluster. Note that these sky blue points are plotted in the bottom plane because the awkcommand outputs a value of 0 for the vertical axis value.
    - The black plots represent the fourth part of the "gnuplot" command

      ```
      "< awk '{if($4>$3){print $1, $2, 0;}}' HCM-2d-Gaussian-2clusters.result_membership"
      ```

      where
      - The content of the "HCM-2d-Gaussian-2clusters.result_membership" file is given to the awk command, and
      - The awk command executes the procedure: "if the 3rd column is less than the 4th column, then plot the 1st column, the 2nd column, and 0."

      Thus, the black points belongs to second cluster. Note that these black points are plotted in the bottom plane because the awk command outputs a value of 0 for vertical axis values.

    If new object is given near the first cluster, the object is classified into the first cluster, and if a new object is given near the second cluster, the object is classified into the second cluster. We can also see the classification boundary, which is displayed as a polyline rather than a straight line, but only because the sampling width is too large. If the sampling width is set smaller, then we could represent the boundary by a straight line, although the file size would be bigger and the "gnuplot" application will be slower.

    iii. To save this plot, type

    ```
    set ticslevel 0
    ```

    (setting the 0-plane at the bottom)

    ```
    set key outside center bottom
    ```

    (moving the key at the bottom center for the plotting frame)

    ```
    set terminal pdf
    ```

(setting the format to PDF)

```
set output "HCM-2d-Gaussian-2clusters.result_classificationFunction.pdf"
```

(setting the filename)

```
replot
```

(plotting the previous plot)

```
set output
```

(outputting to a file and closing the file)

```
quit
```

(exiting the "gnuplot" application).

iv. Click "HCM-2d-Gaussian-2clusters.result_classificationFunction.pdf" on "Files" of repl.it to see the contents, which can be used to prepare your report (you can doanload this PDF file to the local machine.

Executing HCM in your own case

- Edit hcm_main_2d-Gaussian-2clusters.cxx as follows.
  - We can see that the cluster number is set to 2 from
    ```
    const int centers_number=2;
    ```
    in l.10.
  - In l.13, we can find
    ```
    std::string filenameData("2d-Gaussian-2clusters.dat");
    ```
    which shows that this program intends to cluster the dataset contained in the "2d-Gaussian-2clusters.dat" file (we already confirmed the content of this file).
  - In l.15, we can find
    ```
    std::string filenameCorrectCrispMembership("2d-Gaussian-2clusters.correctCrispMembership");
    ```
    which shows that this program intends to calculate the ARI value of the clustering result obtained using the correct clustering of the "2d-Gaussian-2clusters.correctCrispMembership" file.
  - The remainder processes are the same for other datasets (the full description is omitted).
- View the contents of the "2d-Gaussian-2clusters.correctCrispMembership" file which contain the correct clustering results,
  - We can find many 0/1 values.
  - This file contains two lines, where each line has 100 0/1 values.
  - The number of lines, i.e., two, corresponds to the number of correct clusters.
  - The number of columns, i.e., 100, corresponds to the number of objects.
  - The value "1" indicates "belongs to", whereas the value "0" denotes "not belong to."
  - For example,
    - The first 50 values in the first line are "1,"
    - The last 50 values in the first line are "0,"
    - The first 50 values in the second line are "0",
    - The last 50 values in the secondt line are "1,"

    which means that
    - The first 50 objects belong to the first cluster,
    - The last 50 objects do not belong to the first cluster,
    - The first 50 objects do not belong to the second cluster, and
    - The last 50 objects belong to the second cluster.
- To see how the "2d-Gaussian-2clusters.correctCrispMembership" file, which includes the correct clustering results for the dataset in the "2d-Gaussian-2clusters.dat" file, was prepared from makeCorrectMembership.cxx
  - In l.4, we can find
    ```
    const int eachDataNum[]={50, 50};
    ```
    which shows that the result obtained comprises two clusters and each cluster has 50 objects.
  - Note that the first 50 objects belong to the first cluster and last 50 objects belong to the last cluster. Other cases, such as the objects in order of odd numbers belonging to the first cluster and the objects in order of even numbers belonging to the second cluster, can be ignored without any loss of generality simply because we prepare a dataset for each class.
- If we compile this file by typing

  ```
  g++ makeCorrectMembership.cxx
  ```

  and execute by typing

  ```
  ./a.out
  ```

  we can obtain streaming information in the Console. To save this stream, type

  ```
  ./a.out > 2d-Gaussian-2clusters.correctCrispMembership
  ```
- In summary, to cluster a dataset, we perform the following process.
  - Prepare a dataset file, e.g., using the program file "randGaussianMain.cxx."
  - Prepare a correct clustering results file, e.g., using the program file "makeCorrectMembership.cxx."
  - Edit the clustering program file, "hcm_main_2d-Gaussian-2clusters.cxx."
  - Compile the clustering program file using the g++ command.
  - Execute the clustering program (./a.out).

  As a results, we can obtain
  - The clustering summary (contingency table and ART value) in the Terminal.
  - A file containing cluster centers ("HCM-2d-Gaussian-2clusters.result_centers" for "2d-Gaussian-2clusters.dat" using HCM).
  - A file containing memberships ("HCM-2d-Gaussian-2clusters.result_membership" for "2d-Gaussian-2clusters.dat" using HCM).

- A file containing the classification function ("HCM-2d-Gaussian-2clusters.result_classificationFunction" for "2d-Gaussian-2clusters.dat" using HCM).

FCM

Trying FCM

1. We use the following files:
   - sfcm_main_2d-Gaussian-2clusters.cxx (main C++ program, which we edit to cluster various datasets),
   - sfcm.h (FCM C++ class declaration),
   - sfcm.cxx (FCM C++ class definition),

   We have checked the following files because these files were downloaded for HCM.
   - hcm.h (HCM C++ class declaration),
   - hcm.cxx (HCM C++ class definition),
   - vector.h (Vector C++ class declaration, which is used for HCM and VectorArray C++ classes),
   - vector.cxx (Vector C++ class definition),
   - vectorArray.h (VectorArray C++ class declaration, which is used for HCM C++ class),
   - vectorArray.cxx (VectorArray C++ class definition),
   - 2d-Gaussian-2clusters.correctCrispMembership (correct clustering information for the "2d-Gaussian-2cluster.dat" file, which is used to produce contingency tables and ARI values, where this is written in the main C++ program file defined above).
2. Compile these files by typing

   ```
   g++ -DCLASSIFICATION_FUNCTION -DCHECK_ANSWER -std=c++11 sfcm.cxx hcm.cxx vectorArray.cxx vector.cxx sfcm_main_2d
   ```

   in the Console. If you obtain error messages, call the TA/SA/teacher.
3. Execute by typing

   ```
   ./a.out
   ```

   The contingency table and ARI value then appear, which show that FCM produces the exact result(100%).
4. Furthermore, the execution of clustering produces the following three files.
   - sFCM-Em2.000000-2d-Gaussian-2clusters.result_centers
   - sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership
   - sFCM-Em2.000000-2d-Gaussian-2clusters.result_classificationFunction

   We can confirm these files in "Files" of repl.it.
5. Click "sFCM-Em2.000000-2d-Gaussian-2clusters.result_centers" in "Files" of repl.it to confirm the contents of this file. We can then see two lines with two columns, which is the same format as the dataset file (after the first line in the "2d-Gaussian-2clusters.dat" file), which we can use to find the cluster centers near the means of the Gaussian distributions set in the "randGaussianMain.cxx" file.
6. To visualize the "sFCM-Em2.000000-2d-Gaussian-2clusters.result_centers" file,
   i. Start the "gnuplot" application by typing

   ```
   gnuplot
   ```

   ii. In the "gnuplot" command prompt, type

   ```
   plot "2d-Gaussian-2clusters.dat" every ::1, "sFCM-Em2.000000-2d-Gaussian-2clusters.result_centers"
   ```

   We can then see that the cluster centers obtained are almost central in both clusters, where the objects are plotted by red plus symbols and the cluster centers by green cross symbols. Note that the plot command can plot more than one file using comma symbols.
   iii. To save this plot, type

   ```
   set size ratio -1
   ```

   (setting the axes ratio to 1)

   ```
   set terminal pdf
   ```

   (setting the format to PDF)

   ```
   set output "sFCM-Em2.000000-2d-Gaussian-2clusters.result_centers.pdf"
   ```

   (setting the filename)

   ```
   replot
   ```

   (plotting the previous plot)

   ```
   set output
   ```

   (outputting to a file, and closing the file)

   ```
   quit
   ```

   (exiting the "gnuplot" application).
   iv. Click sFCM-Em2.000000-2d-Gaussian-2clusters.result_centers.pdf on "Files" of repl.it to see the contents, which can be used to prepare your report (you can copy this PDF file to the local machine.
7. To confirm the content of the "sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership" file, click sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership on "Files" of repl.it. Many lines are then visible with four columns.
   - This file contains 100 lines.
   - Each line shows the membership information of each object.
   - These columns show the following:
     - First column: first element value of the object,
     - Second column: second element value of the object,
     - Third column: membership value for which the object belongs to the first cluster, and
     - Fourth column: membership value for which the object belongs to the second cluster.

For example, the first line shows that this object (-1.18464, -1.2737) belongs to the first cluster at a probability of 0.515335% and to the second cluster at 99.4847%.

8. Visualize the "sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership" file as follows:
   i. Start the "gnuplot" application by typing
   ```
   gnuplot
   ```
   on the Console.
   ii. In the "gnuplot" command prompt, type
   ```
   plot "< awk '{if($3>$4){print $1, $2;}}' sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership", "< awk
   ```
   We can then see that the objects at lower left are plotted in red and those at the upper right are plotted in green.
   - The red plots reporesent the first part of the "gnuplot" command
     ```
     "< awk '{if($3>$4){print $1, $2;}}' sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership"
     ```
     where
     - The content of the "sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership" file is given to the awk command, and
     - The awk command executes the procedure: "if the 3rd column is greater than the 4th column, then plot the 1st and the 2nd columns."

     Thus, the red points belong to the first cluster.
   - The green plots reporesent the latter part of the "gnuplot" command
     ```
     "< awk '{if($3<$4){print $1, $2;}}' sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership"
     ```
     where
     - The content of the "sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership" file is given to the awk command, and
     - The awk command executes the procedure: "if the 3rd column is less than the 4th column, then plot the 1st and the 2nd columns."

     Thus, the green points belong to the second cluster.

   We can see that the clustering procedure is successful.
   iii. To save this plot, type
   ```
   set size ratio -1
   ```
   (setting the axes ratio to 1)
   ```
   set terminal pdf
   ```
   (setting the format to PDF)
   ```
   set output "sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership.pdf"
   ```
   (setting the filename)
   ```
   replot
   ```
   (plotting the previous plot)
   ```
   set output
   ```
   (outputting to a file and closing the file)
   ```
   quit
   ```
   (exiting the "gnuplot" application).
   iv. Click sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership.pdf to check the desired plot, which can be used to prepare your report (you can copy this PDF file to the local machine.
9. Click sFCM-Em2.000000-2d-Gaussian-2clusters.result_classificationFunction on "Files" of repl.it. Many lines can then be seen with five columns.
   - We are not concerned with the number of lines, which do not correspond to any objects that need to be clustered, but instead other objects need to be classified after clustering.
   - The columns show the following:
     - First column: first element value for a new object,
     - Second column: second element value for a new object,
     - Third column: membership value for which the object belongs to the first cluster,
     - Fourth column: membership value for which the object belongs to the second cluster, and
     - Fifth column: maximal membership value among those in the same line.

   For example, the first line shows the new object (-4.4414, -4.42013) belongs to the first cluster at a probability of 27.4954% and to the second cluster at 72.5046%, and the maximal membership value is 0.725046.
10. Visualize the "sFCM-Em2.000000-2d-Gaussian-2clusters.result_classificationFunction" file as follows:
    i. Start the "gnuplot" application by typing
    ```
    gnuplot
    ```
    ii. In the "gnuplot" command prompt, type
    ```
    splot "sFCM-Em2.000000-2d-Gaussian-2clusters.result_classificationFunction" using 1:2:3 with lines, "sFCM-Em2.(
    ```
    We can then see two surfaces, i.e., red and green, and two types of points, i.e., blue and pink. To remove hidden surfaces, type
    ```
    set hidden3d
    ```
    ```
    replot
    ```
    - Red plots reporesent the first part of the "gnuplot" command
      ```
      "sFCM-Em2.000000-2d-Gaussian-2clusters.result_classificationFunction" using 1:2:3 with lines
      ```
      where
      - The splot command obtains a 3D plot with three columns in each line

- The option "using 1:2:3" means that the splot command uses the first, second and thid columns (first element value of a new object, second element value of a new object, and the membership value for which the object belongs to the first cluster);
- The option "with lines" means that the plotted points are connected by lines, i.e., to make surface.

Thus, the red surface represents the classification function for the first cluster, where the bottom plane shows the data space and the vertical axis denotes the membership value.

- Blue plots represent the second part of the "gnuplot" command

```
"sFCM-Em2.000000-2d-Gaussian-2clusters.result_classificationFunction" using 1:2:4 with lines
```

where

- The splot command obtains 3D plot with three columns in each line;
- The option "using 1:2:4" means that the splot command uses the first, second, and fourth columns (first element value of a new object, second element value of a new object, and the membership value for which the object belongs to the second cluster);
- The option "with lines" means that the plotted points are connected by lines, i.e., to make surface.

Thus, the blue surface represents the classification function for the second cluster, where the bottom plane shows the data space and the vertical axis denotes the membership value.

- Sky blue plots represent the third part of the "gnuplot" command

```
"< awk '{if($3>$4){print $1, $2, $3;}}' sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership"
```

where

- The content of the "sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership" file is given to the awk command, and
- The awk command executes the procedure: "if the 3rd column is greater than the 4th column, then plot the 1st column, the 2nd column, and the 3rd column".

Thus, the sky blue points belongs to the first cluster. Note that these sky blue points are plotted on a red surface, which shows that the classification function extrapolates from the membership values of previously given objects during clustering.

- Black plots represent the fourth part of the "gnuplot" command

```
"< awk '{if($4>$3){print $1, $2, $4;}}' sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership"
```

where

- The content of the "sFCM-Em2.000000-2d-Gaussian-2clusters.result_membership" file is given to the awk command, and
- The awk command executes the procedure: "if the 3rd column is less than the 4th column, then plot the 1st column, the 2nd column, and the 4th column."

Thus, the black points belong to the second cluster. Note that these black points are plotted on a blue surface, which shows that the classification function extrapolates from the membership values of previously given objects during clustering.

If a new object is given near the first cluster, the object is classified into the first cluster, and if a new object is given near the second cluster, the object is classified into the second cluster. We can also see that in contrast to HCM, the maximal classification function values are obtained at cluster centers. We can also see the classification boundary.

iii. To save this plot, type

```
set ticslevel 0
```

(setting the 0-plane at bottom)

```
set key outside center bottom
```

(mow key at the bottom center for the plot frame)

```
set terminal pdf
```

(setting the format to PDF)

```
set output "sFCM-Em2.000000-2d-Gaussian-2clusters.result_classificationFunction.pdf"
```

(setting the filename)

```
replot
```

(plotting the previous plot)

```
set output
```

(outputting to a file and closing the file)

```
quit
```

(exiting the "gnuplot" application).

iv. Click sFCM-Em2.000000-2d-Gaussian-2clusters.result_classificationFunction.pdf to check the previous plot, which can be used to prepare your report (you can copy this PDF file to the local machine.

Executing FCM in your own case

- To view the contents of the program, click sfcm_main_2d-Gaussian-2clusters.cxx on "Files" of repl.it.
  - We can see that the cluster number is set to 2 from

```
const int centers_number=2;
```

on l.10.

  - In l.13, we can find

```
std::string filenameData("2d-Gaussian-2clusters.dat");
```

which shows that this program intends to cluster the dataset contained within the "2d-Gaussian-2clusters.dat" file (we have already confirmed the contents of this file).

  - In l.15, we can find

```
std::string filenameCorrectCrispMembership("2d-Gaussian-2clusters.correctCrispMembership");
```

which shows that this program intends to calculate the ARI value of the clustering result obtained using the correct clustering results in the "2d-Gaussian-2clusters.correctCrispMembership" file.
    - The descriptions given above are the same as those for the HCM clustering program file "hcm_main_2d-Gaussian-2clusters.cxx", FCM also has a parameter called fuzzifier, which is set in l.35 as the last argument of the "Sfcm" C++ object constructor,
      ```
      Sfcm test(data_dimension, data_number, centers_number, 2.0);
      ```
    - The remaining processes are the same for other datasets (the full description is omitted).
- We already viewed and understood the content of the "2d-Gaussian-2clusters.correctCrispMembership" filein the HCM section above.
- We already understood how the file "2d-Gaussian-2clusters.correctCrispMembership" was made in the HCM section above.
- In summary, we cluster a dataset as follows.
    - Prepare a dataset file, e.g., using the program file "randGaussianMain.cxx."
    - Prepare a correct clustering situation file, e.g., using the program file "makeCorrectMembership.cxx."
    - Edit the clustering program file "sfcm_main_2d-Gaussian-2clusters.cxx."
    - Compile the clustering program file by the g++ command.
    - Execute the clustering program (./a.out).

  We then obtain
    - The clustering summary (contingency table and ARI value) in the Terminal,
    - A file containing cluster centers ("sFCM-Em2.000000-2d-Gaussian-2clusters.result_centers" for "2d-Gaussian-2clusters.dat" using FCM),
    - A file containing memberships ("FCM-Em2.000000-2d-Gaussian-2clusters.result_membership" for "2d-Gaussian-2clusters.dat" using FCM), and
    - A file containing the classification function ("FCM-Em2.000000-2d-Gaussian-2clusters.result_classificationFunction" for "2d-Gaussian-2clusters.dat" using FCM).

Clustering Real Datasets

Iris Dataset

- See the UCI Machine Learning Repository: Iris Data Set.
- This dataset comprised 150 objects with four attributes, which need to be clustered into three clusters.
- We use IrisRaw.dat as the objects-file and Iris.correctCrispMembership as the ground-truth-file, which was arranged for the program file used in the class.
- Copy the "hcm_main_2d-Gaussian-2clusters.cxx" file to the "hcm_main_IrisRaw.cxx" file by selecting "Duplicate" in the dots menu of hcm_main_2d-Gaussian-2clusters.cxx, and rename hcm_main_2d-Gaussian-2clusters(copy).cxx to hcm_main_IrisRaw.cxx.
- Click hcm_main_IrisRaw.cxx to edit.
    - Change the cluster number in l.10 to
      ```
      const int centers_number=3;
      ```
    - Change the objects-filename in l.13 to
      ```
      std::string filenameData("IrisRaw.dat");
      ```
    - Change the ground-truth-filename in l.14 to
      ```
      std::string filenameCorrectCrispMembership("Iris.correctCrispMembership");
      ```
- To compile this file, type
  ```
  g++ -DCHECK_ANSWER -std=c++11 hcm.cxx vectorArray.cxx vector.cxx hcm_main_IrisRaw.cxx
  ```
- To execute clustering, type
  ```
  ./a.out
  ```
  We then obtain the results (the contingency table and the ARI value) in the Terminal, and two files (the cluster centers and the memberships).

Breast Cancer Wisconsin Dataset

- See the UCI Machine Learning Repository: Breast Cancer Wisconsin (Original) Data Set.
- This dataset comprises 683 objects with nine attributes, which need to be clustered into two clusters.
- We use BCWRaw.dat as the objects-file and BCW.correctCrispMembership as the ground-truth-file, which was arranged for the program used in the class.
- Apply the same procedure conducted using the Iris dataset.

Wine Dataset

- See the UCI Machine Learning Repository: Wine Data Set.
- This dataset comprises 178 objects with 13 attributes, which need to be clustered into three clusters.
- We use WineRaw.dat as the objects-file and Wine.correctCrispMembership as the ground-truth-file, which was arranged for the program used in the class.
- Apply the same procedure conducted using the Iris dataset.

Note for FCM

- To use FCM, there is an option to set the fuzzifier parameter (2.0 in the distributed file), which must be greater than 1, e.g., l.35 in the "sfcm_main_2d-Gaussian-2clusters.cxx" file where
  ```
  Sfcm test(data_dimension, data_number, centers_number, 1.35);
  ```

- We can also change the fuzzifier value to

```
test.fuzzifierEm()=1.4;
```

after instantiating the "Sfcm" C++ object "test". Furthermore, we can obtain many clustering results using FCM with various fuzzifier values by running a loop as follows.

```
for(double test.fuzzifierEm()=1.1; test.fuzzifierEm()<=3.0; test.fuzzifierEm()+=0.1){
  //do Clustering
}
```
