Genetic Algorithm & Fuzzy Logic

Semester-5

Practical - 10

Name: Shivam Tawari

Roll no: A-58

Aim: Implementation of different defuzzification techniques

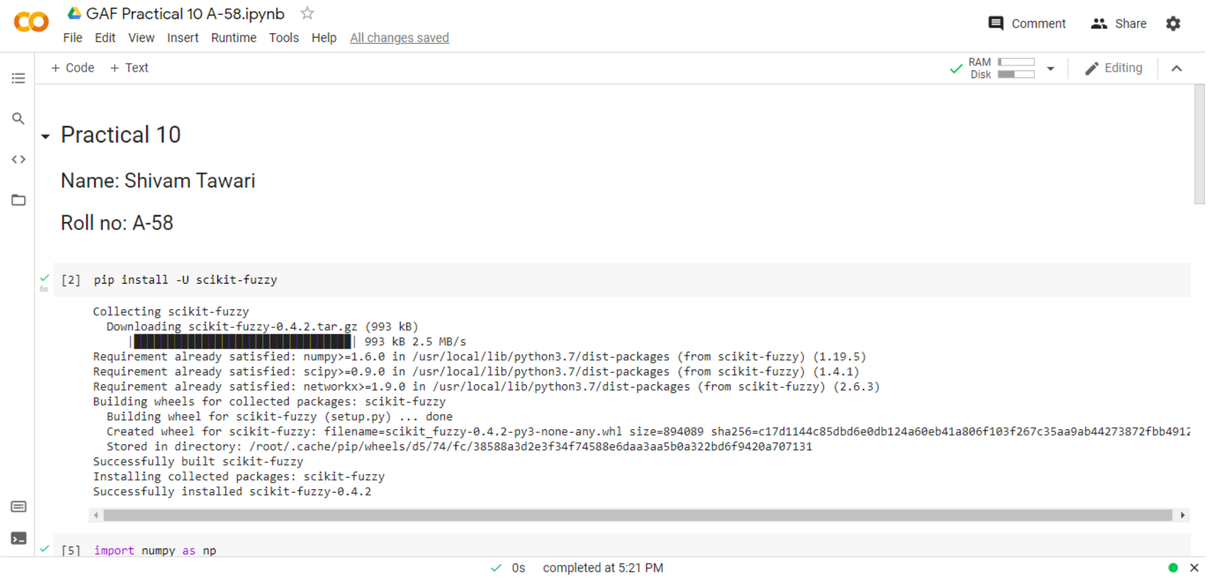
Theory:

What is Defuzzification?

It may be defined as the process of reducing a fuzzy set into a crisp set or to convert a fuzzy member into a crisp member.

We have already studied that the fuzzification process involves conversion from crisp quantities to fuzzy quantities. In a number of engineering applications, it is necessary to defuzzify the result or rather “fuzzy result” so that it must be converted to crisp result. Mathematically, the process of Defuzzification is also called “rounding it off”.

Code and Output:


GAF Practical 10 A-58.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
+ Code + Text
RAM Disk
Editing

Practical 10

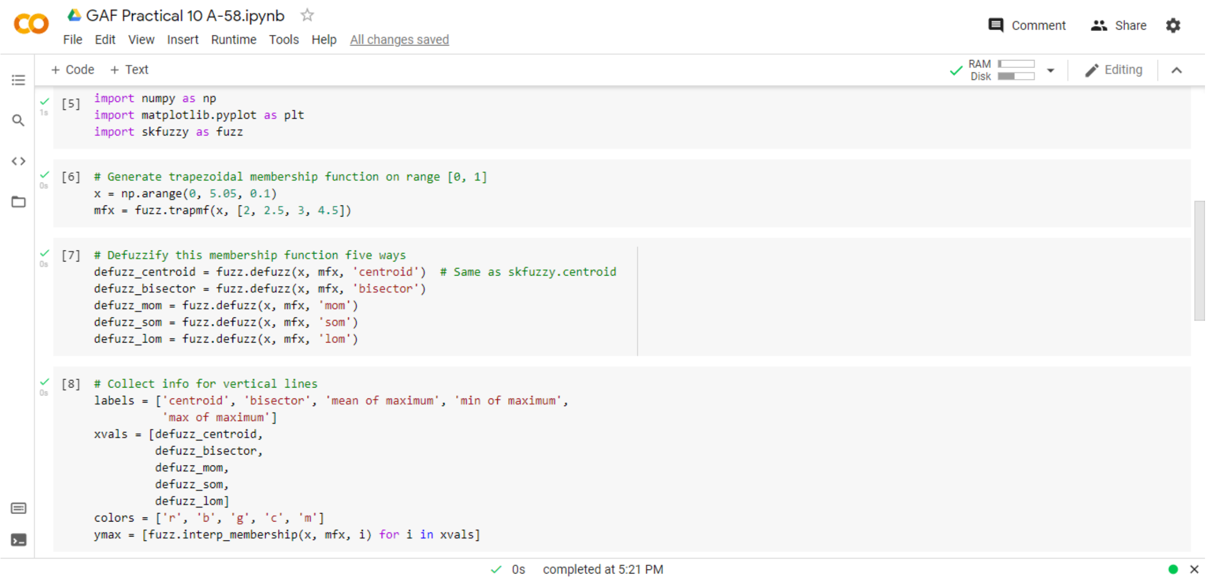
Name: Shivam Tawari

Roll no: A-58

```
[2] pip install -U scikit-fuzzy
```

```
Collecting scikit-fuzzy
  Downloading scikit-fuzzy-0.4.2.tar.gz (993 kB)
     |████████████████████████████████| 993 kB 2.5 MB/s
Requirement already satisfied: numpy>=1.6.0 in /usr/local/lib/python3.7/dist-packages (from scikit-fuzzy) (1.19.5)
Requirement already satisfied: scipy>=0.9.0 in /usr/local/lib/python3.7/dist-packages (from scikit-fuzzy) (1.4.1)
Requirement already satisfied: networkx>=1.9.0 in /usr/local/lib/python3.7/dist-packages (from scikit-fuzzy) (2.6.3)
Building wheels for collected packages: scikit-fuzzy
  Building wheel for scikit-fuzzy (setup.py) ... done
  Created wheel for scikit-fuzzy: filename=scikit_fuzzy-0.4.2-py3-none-any.whl size=894089 sha256=c17d1144c85dbd6e0db124a60eb41a806f103f267c35aa9ab44273872fbb491
  Stored in directory: /root/.cache/pip/wheels/d5/74/fc/38588a3d2e3f34f74588e6daa3aa5b0a322bd6f9420a707131
Successfully built scikit-fuzzy
Installing collected packages: scikit-fuzzy
Successfully installed scikit-fuzzy-0.4.2
```

```
[5] import numpy as np
```

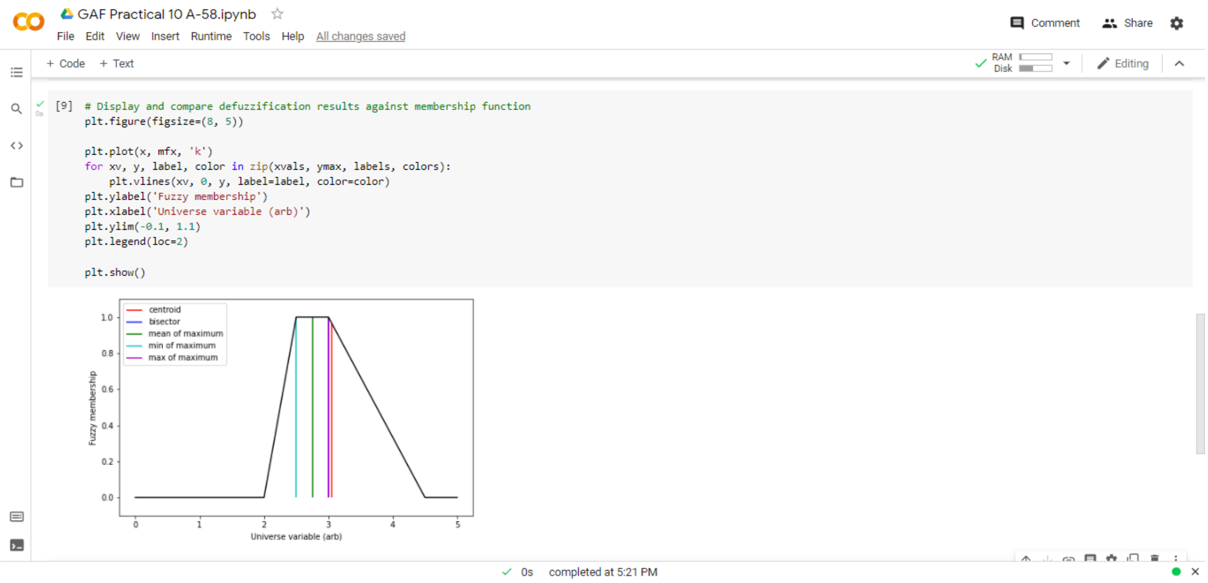

GAF Practical 10 A-58.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
+ Code + Text
RAM Disk
Editing

```
[5] import numpy as np
    import matplotlib.pyplot as plt
    import skfuzzy as fuzz
```

```
[6] # Generate trapezoidal membership function on range [0, 1]
    x = np.arange(0, 5.05, 0.1)
    mfx = fuzz.trapmf(x, [2, 2.5, 3, 4.5])
```

```
[7] # Defuzzify this membership function five ways
    defuzz_centroid = fuzz.defuzz(x, mfx, 'centroid')  # Same as skfuzzy.centroid
    defuzz_bisector = fuzz.defuzz(x, mfx, 'bisector')
    defuzz_mom = fuzz.defuzz(x, mfx, 'mom')
    defuzz_som = fuzz.defuzz(x, mfx, 'som')
    defuzz_lom = fuzz.defuzz(x, mfx, 'lom')
```

```
[8] # Collect info for vertical lines
    labels = ['centroid', 'bisector', 'mean of maximum', 'min of maximum',
              'max of maximum']
    xvals = [defuzz_centroid,
             defuzz_bisector,
             defuzz_mom,
             defuzz_som,
             defuzz_lom]
    colors = ['r', 'b', 'g', 'c', 'm']
    ymax = [fuzz.interp_membership(x, mfx, i) for i in xvals]
```


GAF Practical 10 A-58.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
+ Code + Text
RAM Disk
Editing

```
[9] # Display and compare defuzzification results against membership function
    plt.figure(figsize=(8, 5))

    plt.plot(x, mfx, 'k')
    for xv, y, label, color in zip(xvals, ymax, labels, colors):
        plt.vlines(xv, 0, y, label=label, color=color)
    plt.ylabel('Fuzzy membership')
    plt.xlabel('Universe variable (arb)')
    plt.ylim(-0.1, 1.1)
    plt.legend(loc=2)

    plt.show()
```

Conclusion: Hence, Implementation of different defuzzification techniques has been successfully.