ASSIGNMENT #1

Q.1 Given the row vector $p = [1 \ -2 \ 5]$ and a Column Vector $q = \begin{bmatrix} 2 \\ 4 \\ 8 \end{bmatrix}$. Solve the following ÷

i) $p * q$ ii) $q * p$ iii) $p^T * q^T$ (T → Transpose)

iv. Sum and Mean of all the Components of Vectors p and q.

v) Number of elements in vectors p and q.

vi) Maximum and Minimum Values in p and q.

vii) Product of Elements of vector p and q.

viii) Arrange the elements of vectors p and q in Ascending Order.

Solution ÷

$p = [1 \ -2 \ 5]$

$q = [2; 4; 8]$

i) >> p*q ↵

ans = 34

ii) >> q*p ↵

$$ans = \begin{bmatrix} 2 & -4 & 10 \\ 4 & -8 & 20 \\ 8 & -16 & 40 \end{bmatrix}$$

iii) >> p' * q' ↵

$$ans = \begin{bmatrix} 2 & 4 & 8 \\ -4 & -8 & -16 \\ 10 & 20 & 40 \end{bmatrix}$$

iv) >> sum(p) ↵

ans = 4

>> mean(p) ↵ ans = 1.333

>> sum (a) ↵ ans = 14

>> mean (a) ↵ ans = 4.6667

v. >> length (b) ↵ ans = 3

>> length (a) ↵ ans = 3

vi) >> max (b) ↵ ans = 5

>> min (b) ↵ ans = -2

>> max (a) ↵ ans = 8

>> min (a) ↵ ans = 2

vii) >> prod (b) ↵ ans = -10

>> prod (a) ↵ ans = 64

viii) >> sort (b) ↵

ans = -2 1 5

>> sort (a) ↵

ans =
2
4
8

Ques-2 Given $A = \begin{bmatrix} 5 & 10 \\ 15 & 20 \end{bmatrix}$; $B = \begin{bmatrix} 2 & 4 \\ 6 & 8 \end{bmatrix}$.

Solve the following :-

i) A+B   ii) A−B   iii) A*B   iv). A^2

v). A/B or AB^{-1} vi) A\B or A^{-1}

Solution :-

A= [5 10; 15 20] ↵

B= [2 4; 6 8] ↵

i) >> A+B ↵   ans = $\begin{bmatrix} 7 & 14 \\ 21 & 28 \end{bmatrix}$

ii) >> A−B ↵   ans = $\begin{bmatrix} 3 & 6 \\ 9 & 12 \end{bmatrix}$

iii) >> A * B ↵   ans= $\begin{bmatrix} 70 & 100 \\ 150 & 220 \end{bmatrix}$

iv. >> A^2 or A*A ↵   ans = $\begin{bmatrix} 175 & 250 \\ 375 & 550 \end{bmatrix}$

v) >> A/B ↵   ans = $\begin{bmatrix} 2.5000 & 0.0000 \\ 0 & 2.5000 \end{bmatrix}$

or

>> A*B^-1 ↵   ans = $\begin{bmatrix} 2.5000 & 0 \\ 0.0000 & 2.5000 \end{bmatrix}$

vi) >> A\B ↵   ans = $\begin{bmatrix} 0.4000 & 0.0000 \\ 0 & 0.4000 \end{bmatrix}$

or

>> A^-1*B ↵   ans = $\begin{bmatrix} 0.4000 & 0 \\ 0 & 0.4000 \end{bmatrix}$

25/01/19

Scanned by CamScanner

ASSIGNMENT #2

Q1 Consider the following fuzzy sets.

$$\underset{\sim}{A} = \left\{ \frac{1}{2} + \frac{0.4}{3} + \frac{0.6}{4} + \frac{0.3}{5} \right\}$$

$$\underset{\sim}{B} = \left\{ \frac{0.3}{2} + \frac{0.2}{3} + \frac{0.6}{4} + \frac{0.5}{5} \right\}$$

Calculate $A \cup B$, $A \cap B$, $\bar{A}$, $\bar{B}$, $\overline{A-B} = (A \cap \bar{B})$ by a Matlab program.

Solution:- Script :-

```
function assignment2
A = [1  0.4  0.6  0.3];
B = [0.3  0.2  0.6  0.5];
disp('A Union B is:-');
mx = max(A,B);
disp(mx);
mi = min(A,B);
disp('A Intersection B is :-');
disp(mi);
ac = 1-A;
disp('A Complement is');
disp(ac);
bc = 1-B;
disp('B Complement is :-');
disp(bc);
a = min(A, 1-B);
disp(' A-B is:');
disp(a);
```

Save file :- file name :- $\boxed{\text{assignment2.m}}$

To run this program :- >> assignment2

25/01/19

ASSIGNMENT # 3

Ques. 1 A = [0 1 0.5 0.4 0.6]
B = [0 0.5 0.7 0.8 0.4]
C = [0.3 0.9 0.2 0 1]
PHI = [0 0 0 0 0]

Write a MATLAB program to implement fuzzy set properties.
Commutative law, Associative law, Distributive law, Involution law, Idempotency law, Identity law, Demorgan's law.

Solution:-

```
function Assignment3
  a = [0 1 0.5 0.4 0.6]
  b = [0 0.5 0.7 0.8 0.4]
  c = [0.3 0.9 0.2 0 1]
  Phi = [0 0 0 0 0]
disp('1. Commutative law :-');
if (max(a,b) == max(b,a) & min(a,b) == min(b,a)
    disp(' a & b follows Commutative law');
else
    disp(' a & b doesnot follows Commutative law');
end
disp('2. Associative law');
if ( max (a, max (b,c)) == max (max(a,b), c) &
        min(a, min(b,c)) == min (min(a,b), c))
    disp(' a,b,c follows Associative law');
else
    disp(' a,b,c doesnot follows Associative law');
end.
disp('3. Distributive law');
if (max(a, min (b,c)) == min (max(a,b), max(a,c)) &
        min(a, max(b,c)) == max(min(a,b), min(a,c)))
```

```
    disp ('a,b,c follows distributive law');
else
    disp ('a,b,c doesnot follows distributive law');
end
disp('4. Idempotency law');
if( max(a,a) == a & min(a,a)==a)
    disp ('a, follows Idempotency law');
else disp ('a doesnot follows Idempotency law');
end
disp('5. Involution law:');
if((1-(1-a))==a)
    disp (' a follows Involution law');
else
    disp (' a doesnot follows Involution law');
end
disp ('6. Identity law');
if( max(a,phi) ==a & min(a,phi) ==phi)
    disp (' a follows Identity law');
else
    disp (' a doesnot Identity law');
end
disp ('7. Demorgans law');
if((1- max(a,b)) == min (1-a, 1-b) & (1-min(a,b)) = max(
    disp (' a,b follows Demorgans law');
else
    disp (' a,b doesnot follows Demorgans law');
end.
```

O/p?

1/02/19

ASSIGNMENT # 4

Ques.1 Find the fuzzy relation between these Vectors R and S using max-min method by a MATLAB Program.

$$R = \begin{bmatrix} 0.2 & 0.3 & 0.4 \\ 0.3 & 0.5 & 0.7 \\ 1 & 0.8 & 0.6 \end{bmatrix}$$

$$S = \begin{bmatrix} 0.1 & 1 \\ 0.4 & 0.2 \\ 0.3 & 0.7 \end{bmatrix}$$

Solution:-

```
function assignment4
r = [0.2 0.3 0.4; 0.3 0.5 0.7; 1 0.8 0.6];
s = [0.1 1; 0.4 0.2; 0.3 0.7];
st = s' ;
ans = [];
for i = 1:3
    for j = 1:2
        a = 0
        for k = 1:3
            a = max(min(r(i,k), st(j,k)),a);
        end
        ans(i,j) = a;
    end
end
disp('Answer is :-');
disp(ans);
```

Output :-

Answer is :-

$$\begin{bmatrix} 0.3 & 0.4 \\ 0.4 & 0.7 \\ 0.4 & 1 \end{bmatrix}$$

Akanksha
8/2/19

Scanned by CamScanner

ASSIGNMENT #5

Q.1 Fuzzy Control of a water heater:- (using Mamdani method).

Settings:-

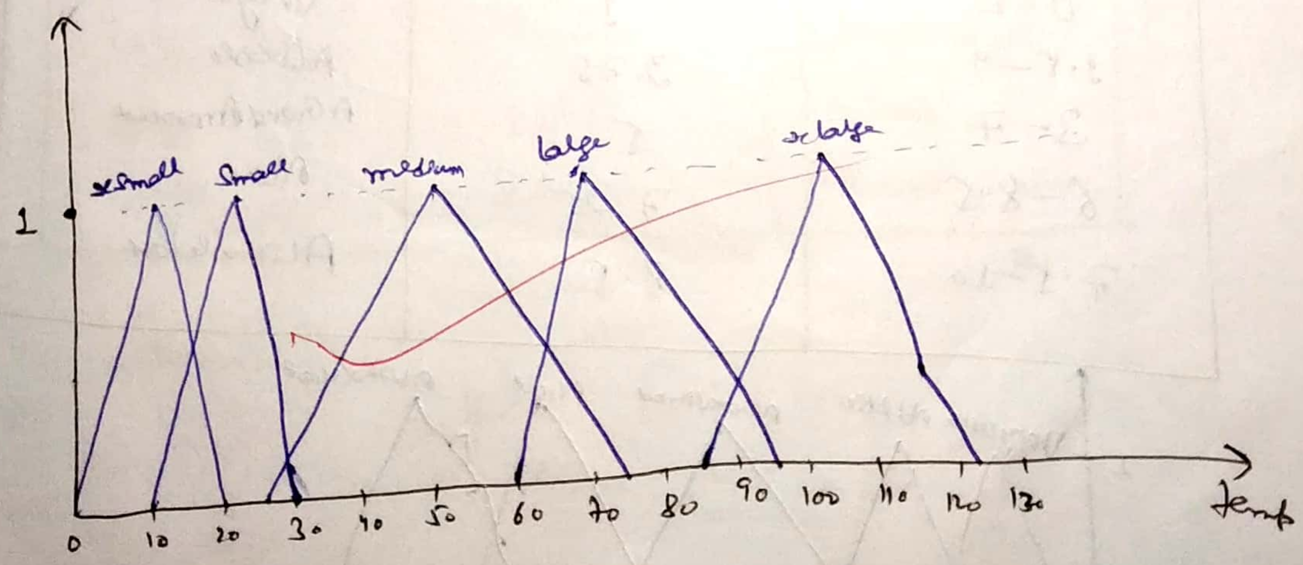
And method → min.
Or method → max.
Implication → min.
Aggregation → max.
Defuzzification → Centroid.

Input :- TempSense, LabelSense
Output :- HeatKnob

*** TempSense :-** Range: [0 125]

Fuzzy Variable Ranges for temp-sense.

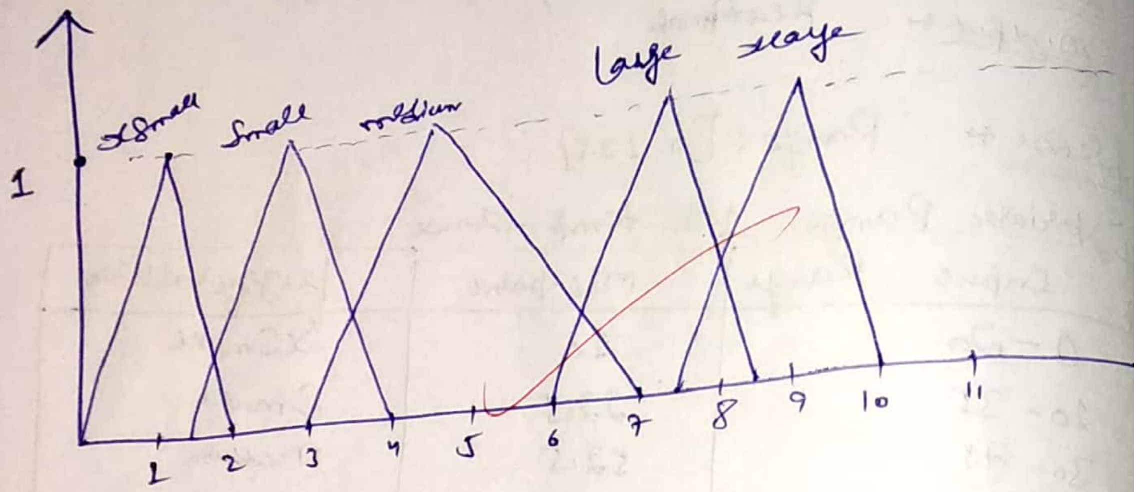
Crisp Input Range	Mid-point	fuzzy Variable
0 – 20	10	XSmall
10 – 35	22.5	Small
30 – 75	52.5	Medium
60 – 95	77.5	Large
95 – 125	105	Xlarge



LabelSense Range: [1 10]

2) Fuzzy Variable Ranges for LabelSense

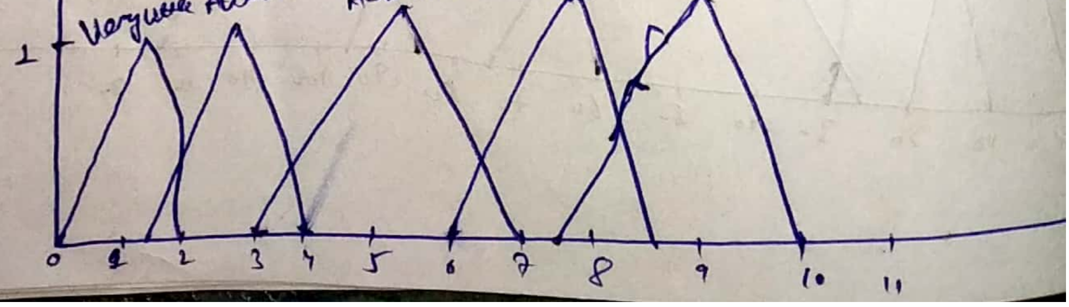
Crisp Input Range	Mid Point	Fuzzy Variable
0-2	1	XSmall
1.5-4	2.75	Small
3-7	5	Medium
6-8.5	7.25	Large
7.5-10	8	XLarge



HeatKnob Label: [1 10]

3). Fuzzy Variable Ranges for HeatKnob

Crisp Input Range	Mid-point	Fuzzy Variable
0-2	1	VeryLittle
1.5-4	2.75	ALittle
3-7	5	A Good Amount
6-8.5	7.25	ALot
7.5-10	8.5	AWholeLot



Scanned by CamScanner

Nineteen (19) Rules :-

S.No.	Temp Sense	Level Sense	Heat Knob
1.	X Small	X Small	A Good Amount
2.	X Small	Small	A Lot
3.	X Small	Medium	A Whole Lot
4.	X Small	Large	A Whole Lot
5.	X Small	X Large	A Whole Lot
6.	Small	X Small	A Little
7.	Small	Small	A Good Amount
8.	Small	Medium	A Lot
9.	Small	Large	A Lot
10.	Small	X Large	A Lot
11.	Medium	X Small	Very Little
12.	Medium	Small	Very Little
13.	Medium	Medium	A Good Amount
14.	Medium	Large	A Lot
15.	Medium	X Large	A Lot
16.	Large	Small	Very Little
17.	Large	Medium	Very Little
18.	Large	Large	A Little
19.	Large	X Large	A Good Amount

View Rules

Inputs: [0.5 ; 6.5]

Heat 4.52

☆ Observations :-

Sr. No.	Input	TempSense	LevelSense	Reaction
1.	[65 ; 6.5]	65	6.5	4.52
2.	[65 ; 6.6]	65	6.6	4.69
3.	[65 ; 6.8]	65	6.8	4.96
4.	[65 ; 7]	65	7.0	5.38
5.	[66 ; 6.5]	66	6.5	4.45
6.	[67 ; 6.5]	67	6.5	4.31
7.	[68 ; 6.5]	68	6.5	4.22
8.	[66 ; 7]	66	7	5.24
9.	[66 ; 8]	66	8	5.09
10.	[66 ; 9]	66	9	5.88
11.	[67 ; 9]	67	9	5.76
12.	[68 ; 9]	68	9	5.65
13.	[69 ; 9]	69	9	5.55
14.	[70 ; 9]	70	9	5.45
15.	[71 ; 9]	71	9	5.36

27/7/19

ASSIGNMENT # 6

Q1. Demonstration of processing on dataset "Student.arff".

Soln → Dataset. Student. arff

@ relation student

@ attribute age {<30, 30-40, >40}

@ attribute income {low, medium, high}

@ attribute student {Yes, No}

@ attribute credit-rating {fair, excellent}

@ attribute buyspc {Yes, No}

@ data

%

<30, high, no, fair, no

<30, high, no, excellent, no

30-40, high, no, fair, yes

>40, medium, no, fair, Yes

>40, low, Yes, excellent, Yes

>40, low, Yes, fair, Yes

30-40, low, Yes, excellent, no

<30, medium, no, fair, no

<30, low, Yes, fair, no

>40, medium, yes, fair, Yes

<30, medium, Yes, excellent, Yes

30-40, medium, no, excellent, Yes

30-40, high, Yes, fair, yes

>40, medium, no, excellent, no

%

Ques 2 Demonstration of preprocessing on dataset labor.arff.

Soln :- Dataset labor.arff

@relation 'labor-neg-data'

@attribute 'duration' real

@attribute 'wage-increase-first-year' rea

@attribute 'wage-increase-second-year' real

@attribute 'wage-increase-third-year' real

@attribute 'cost-of-living-adjustment' {'none', 't

@attribute 'working-hours' real

@attribute 'Pension' {'none', 'ret_allw', 'empl.

@attribute 'standby-pay' real

@attribute 'shift-differential' real

@attribute 'education-allowance' {'yes', 'no'}

@attribute 'statutory-holidays' real

@attribute 'Vacation' {'below_average', 'average',

@attribute 'longterm-disability-assistance' {'yes'

@attribute 'contribution-to-dental-plan' {'none', 'half

@attribute 'bereavement-assistance' {'yes', 'no'}

@attribute 'contribution-to-health-plan' {'none', 'half',

@attribute 'class' {'bad', 'good'}

Scanned by CamScanner

15

<u>Ues3</u> Demonstration of Association rule process on dataset Contactlenses.arff using apriori algorithm.

<u>Soln:-</u>

<u>Dataset Contactlenses.arff</u>

@ relation Contact-lenses
@ attribute age { young, pre-presbyopic, presbyopic}
@ attribute spectacle-prescrip { myope, hypermetrope}
@ attribute astigmatism { no, yes}
@ attribute tear-prod-rate { reduced, normal}
@ attribute Contact-lenses { soft, hard, none}

@ data

%
% 24 instances
%

Young, myope, no, reduced, none
Young, myope, no, normal, soft
Young, myope, yes, normal, hard
young, hypermetrope, no, reduced, none
young, hypermetrope, no, normal, soft
presbyopic, myope, no, reduced, none
presbyopic, myope, yes, normal, hard
pre-presbyopic, myope, no, normal, soft

%

Ques 4 Demonstration of Association rule process dataset test.arff apriori algorithm.

Soln :- Dataset test.arff

```
@relation test
@attribute Admissionyear {2005, 2006, 2007, 2008, 20
@attribute Course {cse, mech, it, ece}

@data
%
2005, cse
2005, it
2005, cse
2006, mech
2006, it
2006, ece
2007, it
2007, cse
2008, it
2008, cse
2009, it
2009, ece
%
```

Ques 5 Demonstration of Classification rule process on dataset Student.arff using j-48 algorithm. 17

Soln

Data Student.arff

@relation Student
@attribute age {<30, 30-40, >40}
@attribute income {low, medium, high}
@attribute Student {yes, no}
@attribute Credit-rating {fair, excellent}
@attribute buyspc {yes, no}

@data

%

<30, high, no, fair, no
<30, high, no, excellent, no
30-40, medium, no, fair, yes
>40, medium, no, fair, yes
>40, low, yes, fair, yes
>40, low, yes, excellent, no
30-40, low, yes, excellent, yes
<30, medium, no, fair, no
<30, low, yes, fair, no
>40, medium, yes, fair, yes
<30, medium, yes, excellent, yes
30-40, medium, no, excellent, yes
30-40, medium, no, excellent, yes
30-40, high, yes, fair, yes
>40, medium, no, excellent, no

%

Ques. Demonstration of classification rule process dataset employee.arff using J-48 algorithm.

Soln:- Dataset employee.arff

@relation employee

@attribute age {25, 27, 28, 29, 30, 35, 48}

@attribute salary {10K, 15K, 17K, 20K, 25K, 30K,

@attribute performance {good, avg, poor}

@data

%

25, 10K, poor

27, 15K, poor

27, 17K, poor

28, 17K, poor

29, 20K, avg

30, 25K, avg

29, 25K, avg

30, 20K, avg

35, 32K, good

48, 34K, good

48, 32K, good

%

Ques7 Demonstration of Classification rule process on dataset employee.arff using id3 algorithm.

Aim:-

Data set employee.arff

@relation employee
@attribute age {25, 27, 28, 29, 30, 35, 48}
@attribute salary {10k, 15k, 17k, 20k, 25k, 30k, 35k, 32k}
@attribute performance {good, avg, poor}

@data
%
25, 10k, poor
27, 15k, poor
27, 17k, poor
28, 17k, poor
29, 20k, avg
30, 25k, avg
29, 25k, avg
30, 20k, avg
35, 32k, good
48, 34k, good
48, 32k, good
%

Ques 8 Demonstration of classification rule process dataset employee.arff using naive bayes algo

Soln:- Data set employee.arff

@relation employee

@attribute age {25, 27, 28, 29, 30, 35, 48}

@attribute salary {10k, 15k, 17k, 20k, 25k, 30k, 3[illegible]

@attribute performance {good, avg, poor}

@data

%

25, 10k, poor

27, 15k, poor

27, 17k, poor

28, 17k, poor

29, 20k, avg

30, 25k, avg

29, 25k, avg

30, 20k, avg

35, 32k, good

48, 34k, good

48, 32k, good

%

Demonstration of Clustering rule process on dataset iris.arff using simple K-means.

Dataset iris.arff

@relation iris
@attribute sepallength Real
@attribute sepalwidth Real
@attribute petallength Real
@attribute petalwidth Real
@attribute class {Iris-setosa, Iris-versicolor, Iris-Virginica}

@Data
%
5.1, 3.5, 1.4, 0.2, Iris-setosa
4.9, 3.0, 1.4, 0.2, Iris-setosa
5.0, 3.3, 1.4, 0.2, Iris-setosa
7.0, 3.2, 4.7, 1.4, Iris-versicolor
6.4, 3.2, 4.5, 1.5, Iris-versicolor
5.1, 2.5, 3.0, 1.1, Iris-versicolor
6.5, 3.0, 5.2, 2.0, Iris-virginica
6.2, 3.4, 5.4, 2.3, Iris-virginica
6.3, 2.5, 5.0, 1.9, Iris-virginica.
%

Ques 10 Demonstration Of Clustering rule process on dataset Student.arff using simple K-means.

Soln→ Dataset Student.arff

```
@relation Student
@attribute age {<30, 30-40, >40}
@attribute income {low, medium, high}
@attribute student {yes, no}
@attribute credit rating {fair, excellent}
@attribute buyspc {yes, no}
@data
%
<30, high, no, fair, no
<30, high, no, excellent, no
30-40, high, no, fair, yes
>40, medium, no, fair, yes
>40, low, yes, excellent, yes
>40, low, yes, fair, yes
30-40, low, yes, excellent, no
<30, medium, no, fair, no
<30, low, yes, fair, no
>40, medium, yes, fair, yes
<30, medium, yes, excellent, yes
30-40, medium, no, excellent, yes
30-40, high, yes, fair, yes
>40, medium, no, excellent, no
%
```

29/3/19

ASSIGNMENT # 7

- Create a New Script :-

$$\text{function } Z = \text{func}(x)$$
$$Z = -(15*x - x*x);$$

Save it as $\boxed{\text{func.m}}$ in the same folder where matlab is installed.

- Open Command Window :-

>> Optimtool ↵

Problem Setups and Results

- Solver :- ga - Genetic Algorithm.

Problem

- fitness function : @func
- No. of Variables : 1

Constraints

- Bounds : lower : 0 upper : 15

Options

population

- population type : Double Vector
- Creation function : Uniform
- Initial Range - specify : [0;15]

Fitness Scaling

- Scaling function : Rank

Selection

- Selection function : Stochastic Uniform

Mutation

- Mutation function : Adaptive feasible

Crossover

Crossover function : Scattered

Hybrid function

Hybrid function : None

Plot functions

☑ Best fitness ☑ Best Individual

Click on Start Button

• **Output :-**

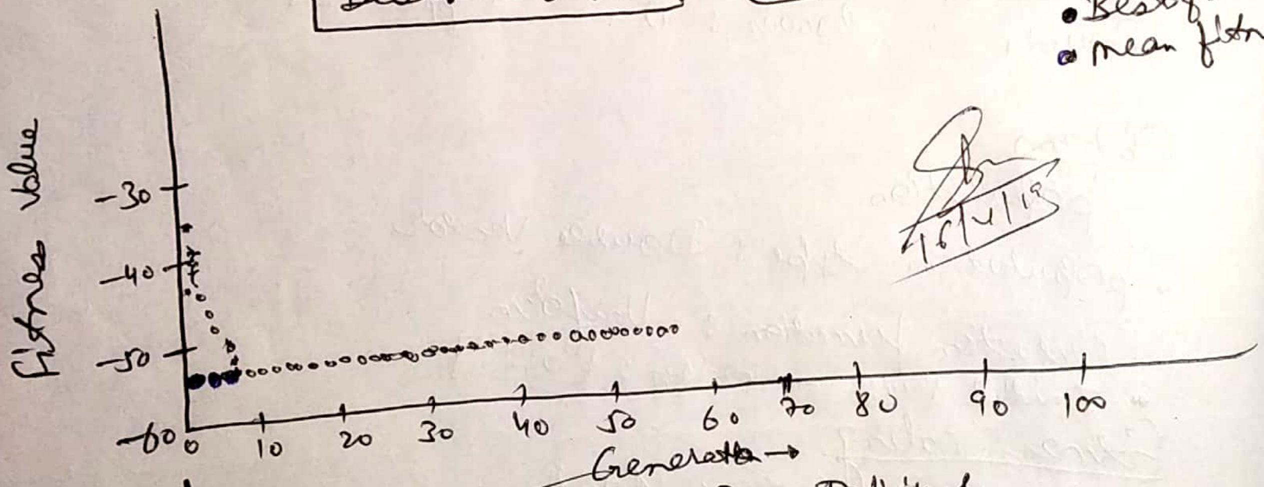
Current Iteration : 57

Optimization running

Objective function Value : -56.2499999903

Optimization terminated : average change in the fitness value less than options function Tolerance.

Best: -56.25 Mean : -56.25



16/4/19

Current Best Individual