1.

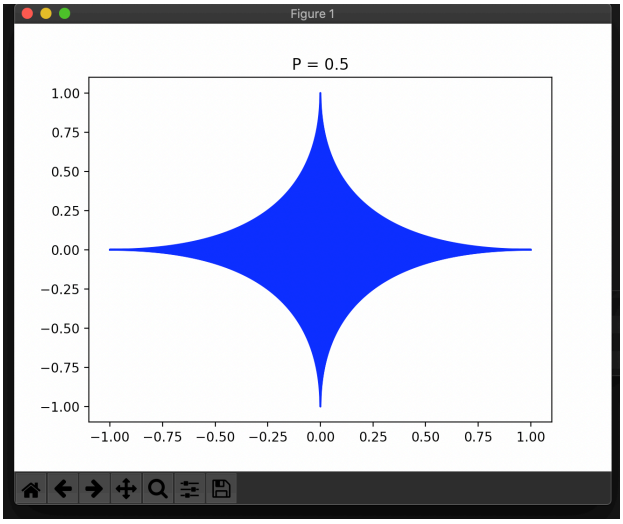
The following are the properties a norm of a vector must satisfy -

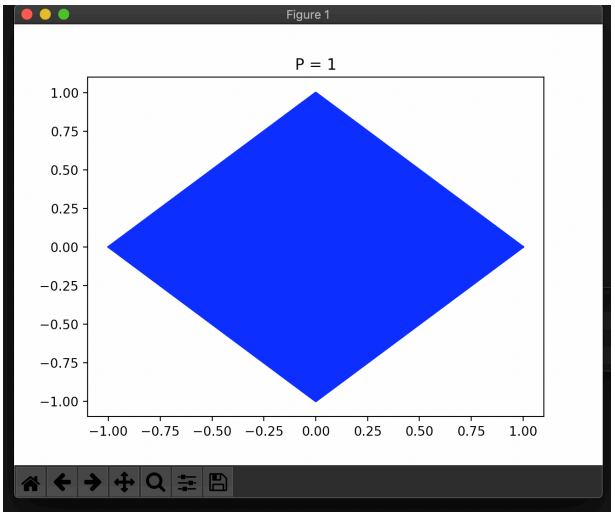
1. $p(\mathbf{u} + \mathbf{v}) \leq p(\mathbf{u}) + p(\mathbf{v})$ (being *subadditive* or satisfying the *triangle inequality*).
2. $p(a\mathbf{v}) = |a|\, p(\mathbf{v})$ (being *absolutely homogeneous* or *absolutely scalable*).
3. If $p(\mathbf{v}) = 0$ then $\mathbf{v} = \mathbf{0}$ is the zero vector (being *positive definite* or being *point-separating*).

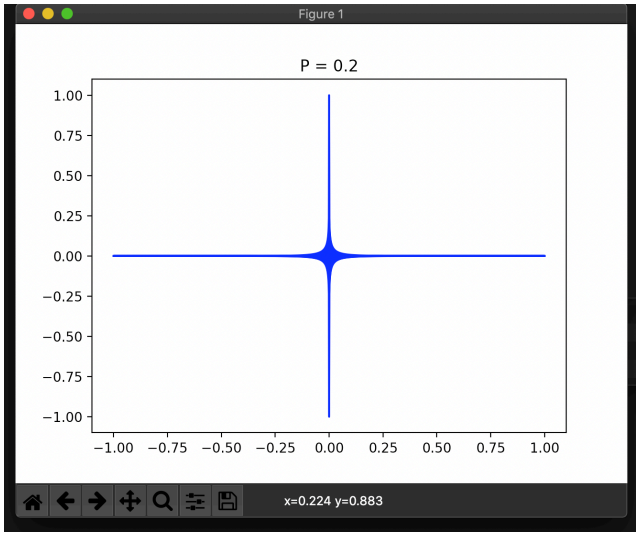
In the case of 0<p<1, the first property of triangle inequality is violated hence, it won't be a norm.
If we take a look at the graphs of 0<p<1, we will notice that the graphs are not convex anymore

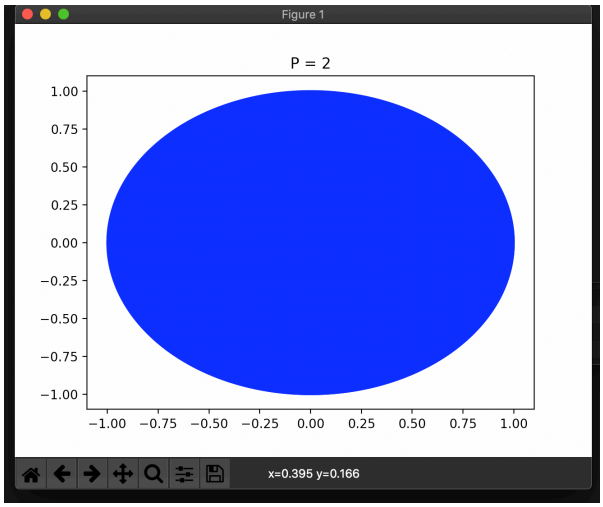
For example - consider the points (1,0) and (0,1)
Consider p = 0.5
p(1,0) = 1, p(0,1) = 1
p((1,0) + (0,1)) = 4

p((1,0) + (0,1)) > p(1,0) + p (0,1) which clearly violates the law









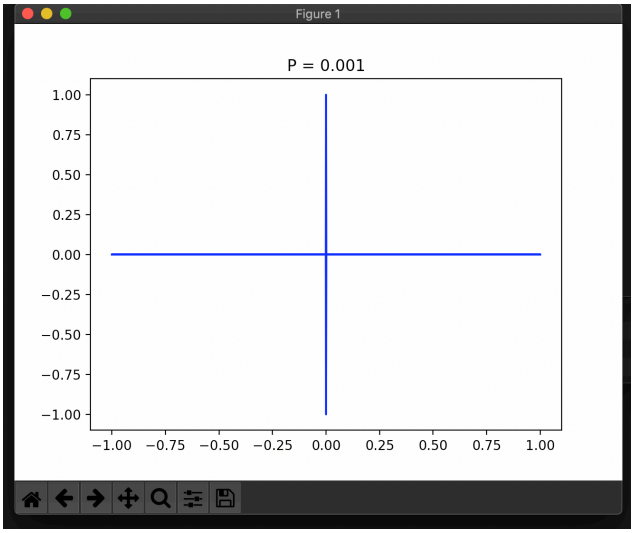


Figure 1
P = 0.001
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00 −0.75 −0.50 −0.25 0.00 0.25 0.50 0.75 1.00

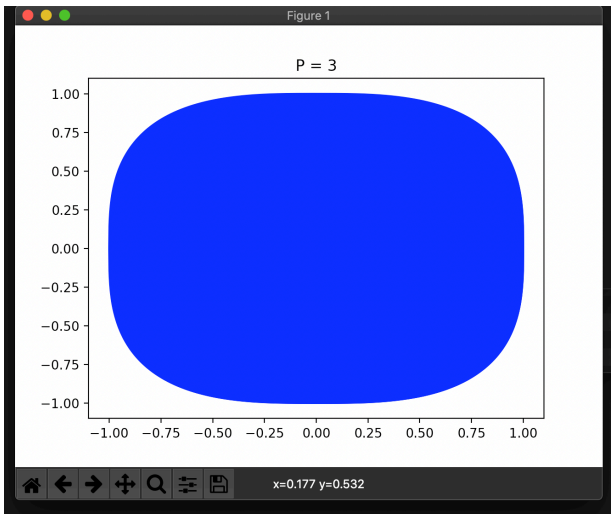


Figure 1
P = 3
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00 −0.75 −0.50 −0.25 0.00 0.25 0.50 0.75 1.00
x=0.177 y=0.532

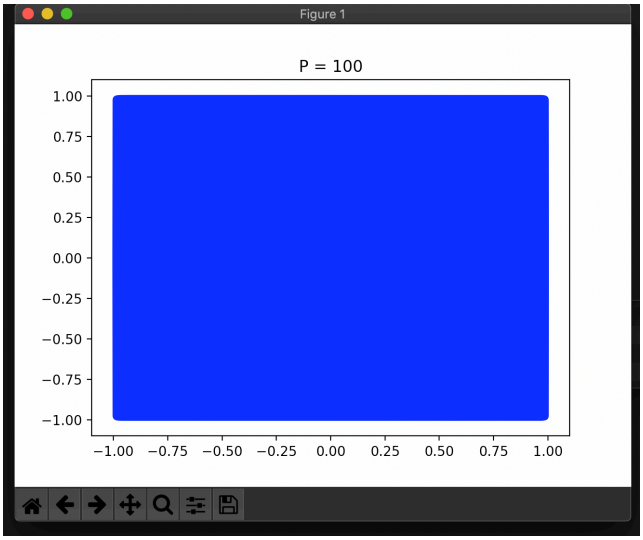


Figure 1
P = 100
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00 −0.75 −0.50 −0.25 0.00 0.25 0.50 0.75 1.00

2.

What I did -

Making the distribution

- I followed a multivariate normal distribution to create a normally distributed data. Below is the formula -

$$p(\mathbf{x};\mu,\Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}}\exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)\right)$$

In our case,
the dimension n is 2,
covariance matrix is 2X2 matrix with only same and non-zero diagonal elements
mean is a 1X2 matrix (mean along x and y axis)

Covariance matrix = sigma^2 * [I]
Where I - is a 2X2 identity matrix
Reason - here the covariance of X,Y is 0 as both are IID's, By assuming both diagonal elements are sigma^2 I'm considering X and Y to have the same variance

- **Choosing the random points**

Random values for X and Y are chosen between 2 SD of the mean. We are specifically choosing only points between 2 SD is cause 98% of the points following a Gaussian distribution fall within this range

- **Finding the probabilities of each point**

The probability of each point is calculated using the above mentioned formula.
A sample size of 400 is considered and the occurrence of every point is calculated using their probabilities.

- **Plotting the Points**

The points are plotted against their occurrence. We can visually see that the data follows a normal distribution. The below 2 pictures are PDFs in the forms of histograms and 3D mesh

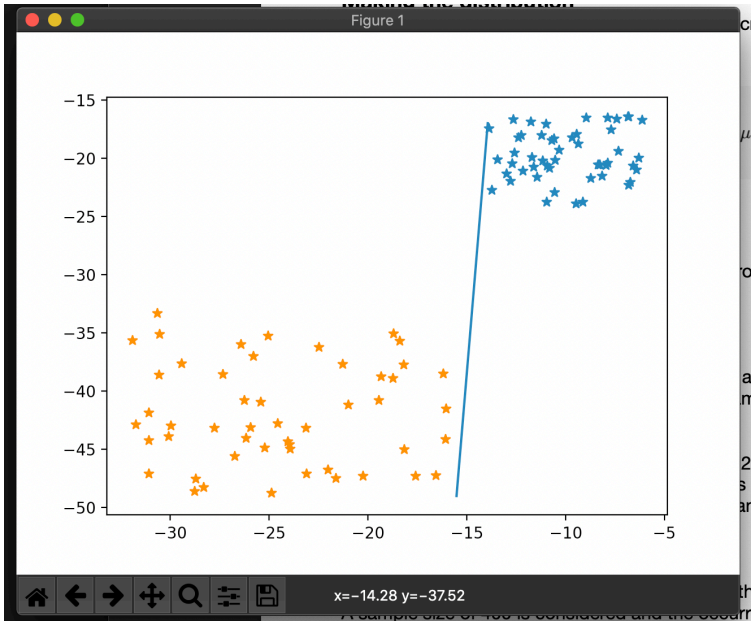
- **Creating and plotting the second dataset**

The first dataset is created using the mean and sigma preset at the beginning of the code. The second dataset is created by choosing a random but appropriate mean that doesn't overlap with the first dataset.

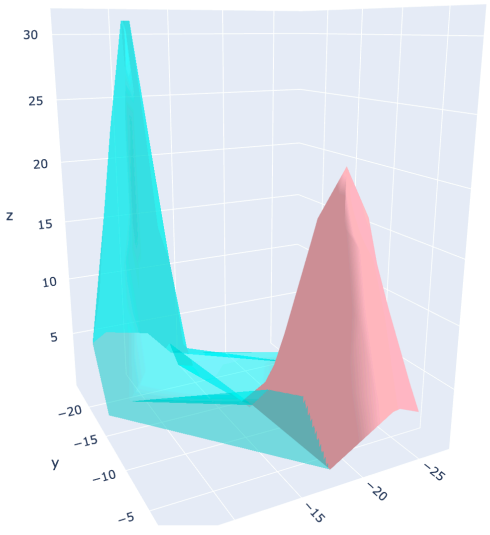
Once the mean is chosen, sigma is chosen at random and the data points are calculated again, between 2SD of the mean.
Occurrences are calculated and the graph is plotted

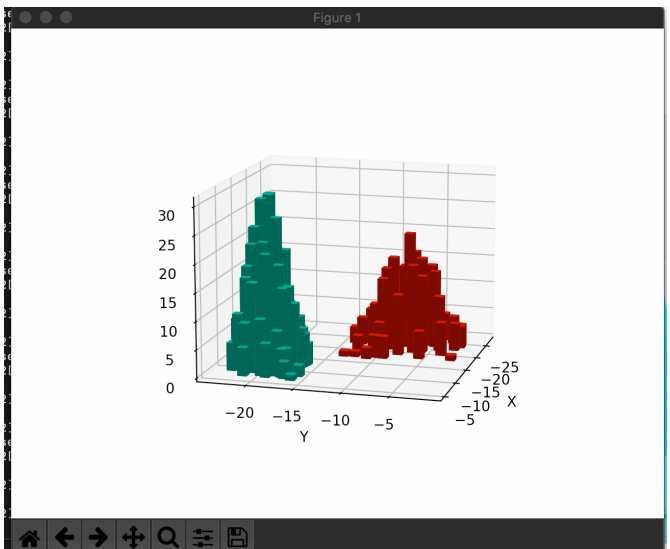
- **Classifying the data points**

The data points are classified according to PLA



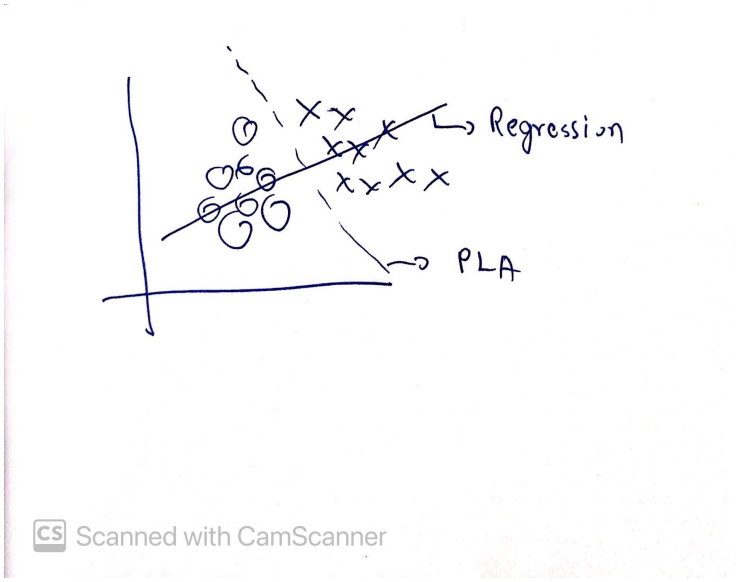
B . From the histogram graph we can see that both the distributions are clearly separated and have different peaks. Therefore, we can say that they are linearly separable
In the 3D mesh, again it can be seen that the peaks are distinct and the distributions are clearly distinct. (The overlap can be ignored, its not related to the data created)





C

No regression may not converge as in regression we try to find a line that fits the data based on the least squares method i..e distance of the point from the line. Consider the example below -



A regression line best fits the data in the above scenario if it passes through both the data points and hence won't be able to classify the data.

3

The Monty Hall problem

It is observed that a player has higher chances of winning if they choose to switch the door after monty opens the door that doesn't have a prize in it. The theoretical probability distributions are given below -

Chosen Door	Prize Door	Win/Lose upon Switch	Win/Lose upon No Switch
1	1	Lose	Win
1	2	Win	Lose
1	3	Win	Lose
2	1	Win	Lose
2	2	Lose	Win
2	3	Win	Lose
3	1	Win	Lose
3	2	Win	Lose
3	3	Lose	Win
		6 wins	3 wins
		6/9 = 66.6%	3/9 = 33.3%

Monty always opens the door which doesn't have a prize in it, allowing you to make a choice of switching between doors.
By weighing the number of times the player wins or loses for all possible combinations of choices we observe that upon switching the door the player wins 66% of the time.

Simulation
- The number of trials considered is 10000.
- During each trial the prize door and the chosen door are randomly assigned
- If the prize door is chosen door, then the win is registered under Win upon not switching
- If prize is not in the chosen door, then the win is registered under switching.
- Prob winning upon switching = No of times we win upon switching / total trials

-Prob

```
(midsem) sisir-macbook-pro-4:midsem sisirreddy$ python3 3.py
 Prob of winning if you switch the door  0.6676
 Prob of winning if dot't switch the door  0.3324
(midsem) sisir-macbook-pro-4:midsem sisirreddy$ python3 3.py
 Prob of winning if you switch the door  0.6671
 Prob of winning if dot't switch the door  0.3329
```

winning upon **Not** switching = No of times we win upon **Not** switching / total trials

4.

A.

Observations -
I get different means and variance every time I run the program because the program randomly selects data points using the sample size specified by the user. This value can sometimes be equal to the sample mean depending on the random data selected from the csv file.
As I keep increasing the sample size the mean and variance of selected sample come close to the population mean and variance. Sometimes, a sample size of 100 also produces mean and variance very close to population mean and variance. This can be a rough representation of the population data.

Random Number Generator
I have used an LCG to generate my random numbers.

$$X_{n+1} = (aX_n + c) \mod m$$

I have selected the values of **a**,**c** and **m** according to the Hull-Dobell Theorm. The requirements for having a period of length m for all seed values are

-

1. m and c are relatively prime,
2. $a - 1$ is divisible by all prime factors of m,
3. $a - 1$ is divisible by 4 if m is divisible by 4.

```
a = 8989542241
c = 95231
m = 2 ** 64
```

B.

Theory - The expectation of sample variance is equal to population variance
Proof -
My sample variance = variance of 100 random samples
Expectation of sample variance =

CASE A

Case 1 - calculating variance of 100 sets each having 100 random samples from given data set
E(sample variance) = 1/n * (sum of variance of all sets)

```
(midsem) sisir-macbook-pro-4:midsem sisirreddy$ python3 test.py
Population mean  249.455676294528  Population Variance  101793.40152100599
 sets of 100 samples variance  106167.47088563854
Mean of the samples is  257.27353427786505
No of samples  100
No of trials  100
```

Case 2 - calculating variance of 300 sets each having 100 random samples from given data set
E(sample variance) = 1/n * (sum of variance of all sets)

```
(midsem) sisir-macbook-pro-4:midsem sisirreddy$ python3 test.py
Population mean  249.455676294528  Population Variance  101793.40152100599
 sets of 100 samples variance  99723.25829526549
Mean of the samples is  252.09050240003464
No of samples  100
No of trials  300
```

Case 3 - calculating variance of 500 sets each having 100 random samples from given data set

```
(midsem) sisir-macbook-pro-4:midsem sisirreddy$ python3 test.py
Population mean  249.455676294528  Population Variance  101793.40152100599
 sets of 100 samples variance  102278.87422033568
Mean of the samples is  250.4046571666574
No of samples  100
No of trials  500
```

Case 4 - 1000 sets with 100 samples in each set

```
(midsem) sisir-macbook-pro-4:midsem sisirreddy$ python3 test.py
Population mean  249.455676294528  Population Variance  101793.40152100599
 sets of 100 samples variance  102805.80975662035
Mean of the samples is  249.75898821309278
No of samples  100
No of trials  1000
```

Case 5 - 1500 sets with 100 samples in each set

```
(midsem) sisir-macbook-pro-4:midsem sisirreddy$ python3 test.py
Population mean  249.455676294528  Population Variance  101793.40152100599
 sets of 100 samples variance  101720.8058976765
Mean of the samples is  247.91563246881694
No of samples  100
No of trials  1500
```

Case 6 - 4500 sets with 100 samples in each set

```
(midsem) sisir-macbook-pro-4:midsem sisirreddy$ python3 test.py
Population mean  249.455676294528  Population Variance  101793.40152100599
 sets of 100 samples variance  100919.35052900159
Mean of the samples is  250.10222336978543
No of samples  100
No of trials  4500
```

CASE B

Taking samples of size 1000

Sample variance = variance of 1000 random samples selected from dataset
E(Sample Variance) =

Case 1 - 100 sets of size 1000

```
(midsem) sisir-macbook-pro-4:midsem sisirreddy$ python3 test2.py
Population mean  249.455676294528  Population Variance  101793.40152100599
 sets of 1000 samples variance  102180.146589702
Mean of the samples is  248.7590646801316
No of samples  1000
No of trials  100
```

Case 2 - 300 sets of size 1000

```
(midsem) sisir-macbook-pro-4:midsem sisirreddy$ python3 test2.py
Population mean  249.455676294528  Population Variance  101793.40152100599
 sets of 1000 samples variance  96200.45952313324
Mean of the samples is  243.22232455890355
No of samples  1000
No of trials  300
```

Case 3 - 500 sets of size 1000

```
(midsem) sisir-macbook-pro-4:midsem sisirreddy$ python3 test2.py
Population mean  249.455676294528  Population Variance  101793.40152100599
 sets of 1000 samples variance  103257.5118528758
Mean of the samples is  250.02003524039526
No of samples  1000
No of trials  500
```

Case 4 - 1000 sets of size 1000

```
(midsem) sisir-macbook-pro-4:midsem sisirreddy$ python3 test2.py
Population mean  249.455676294528  Population Variance  101793.40152100599
 sets of 1000 samples variance  101738.31751809458
Mean of the samples is  250.06216172443052
No of samples  1000
No of trials  1000
```

It can clearly be seen that as the no of trials increase, the E(sample variance) converges to the population variance