

# Problem Set 1

**Name:** Manuel Rodriguez

**Late Submission:** No

## Contents

<b>1</b>	<b>Generate the Dataset</b>	<b>2</b>
1.1	A . . . . .	2
1.2	B . . . . .	2
1.3	Controlling Randomness . . . . .	2
1.3.1	C . . . . .	2
1.4	D . . . . .	2

# 1 Generate the Dataset

## 1.1 A

The data is going to be composed of points taken from two different, randomly-generated, Gaussian distributions. Without looking at the plots, I would expect that the two classes are relatively different with the two having clearly different centers, but I would not expect the data to be linearly separable since it is unlikely that the centers are far apart and the uncertainties generated by both Gaussians are so small that there are no data points from the two distributions that are close together. Looking at the data, we can tell that there are two clear centers, but as expected, there is some overlap in the middle of the graph.

## 1.2 B

For  $\text{dims} = 3$ , I think the data would be more likely to be linearly separable since there are more degrees of freedom for the points to be distributed across and thus there is less of a chance that they will disperse along the same axis or have nearby centers.

## 1.3 Controlling Randomness

### 1.3.1 C

An element of randomness that needs to be addressed for a perceptron is the set of points which are evaluated when determining error and updating weights. If these points are not chosen randomly, there would be a bias toward fitting for the first points in the model which could be fatal if the points are ordered, for example, by simple ascending order. Another element of randomness to consider is the initial values of all the weights, obviously they cannot be zero because then you face vanishing gradients, but also they must not be biased toward any particular target function, so a popular choice if initializing the weights to small, normally distributed values.

### 1.3.2 D

Done :D