1) The classification of the feature vector X depends only on the value of Xk. Therefore, the value of Y as a function of Xk is as shown in Fig 1. From the figure, it is clear that the data is linearly separable, which suggests there must exist a perceptron that is capable of perform the classification. This perceptron is, however, not unique since there exist multiple lines that can separate the data. The ideal perceptron would have the largest margin which would be the case when the parameters are w=[0,0, ……., -2/Ɛ] and b=0. These parameters would give a minimum margin of $\sqrt{Ɛ^2 + 1}$ and is geometrically represented by the red line which is actually perpendicular to the line connecting (Ɛ,1) and (-Ɛ, -1) in Fig 1.

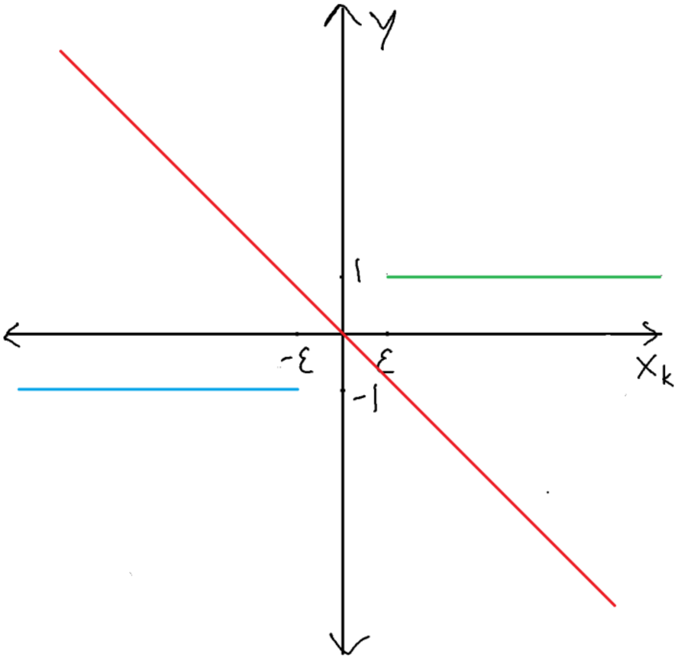


Fig 1. Linear separability of the data

2) The generated perceptron has weights that are not equal to the weights mentioned in the previous answer. However, it can be observed that the weights w1, …., wk-1 are quite low hovering between -2 and +2 and the weight wk+1 is much higher having a value of around 16. The bias, while not 0, is stillquite close as -1.

3) The change in average number of steps over 20 trials for different values of Ɛ is presented in Fig 2. The average number of steps decreases with an increase in Ɛ because the minimum possible margin of the perceptron increases with Ɛ. As this margin increases, the space of all possible linear separators becomes larger making it easier for the algorithm to terminate quicker.

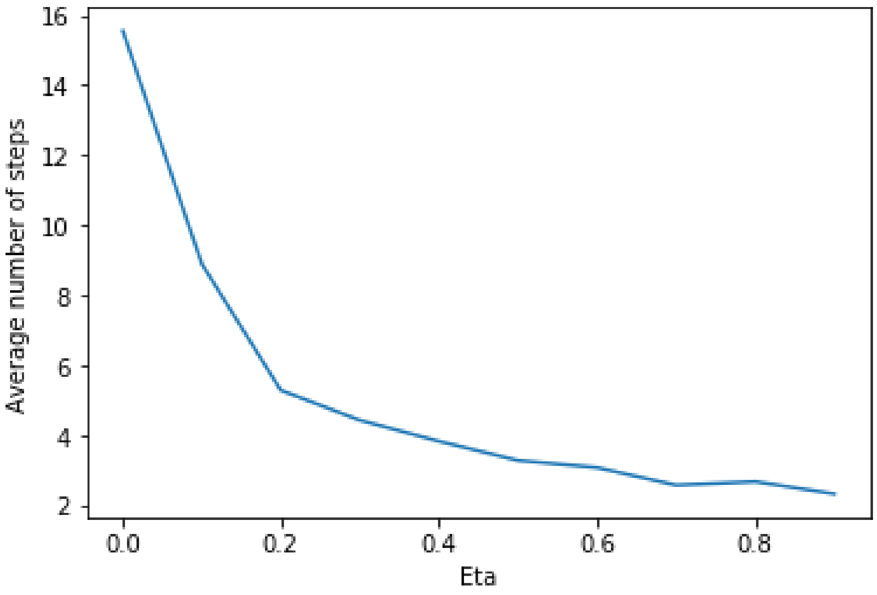
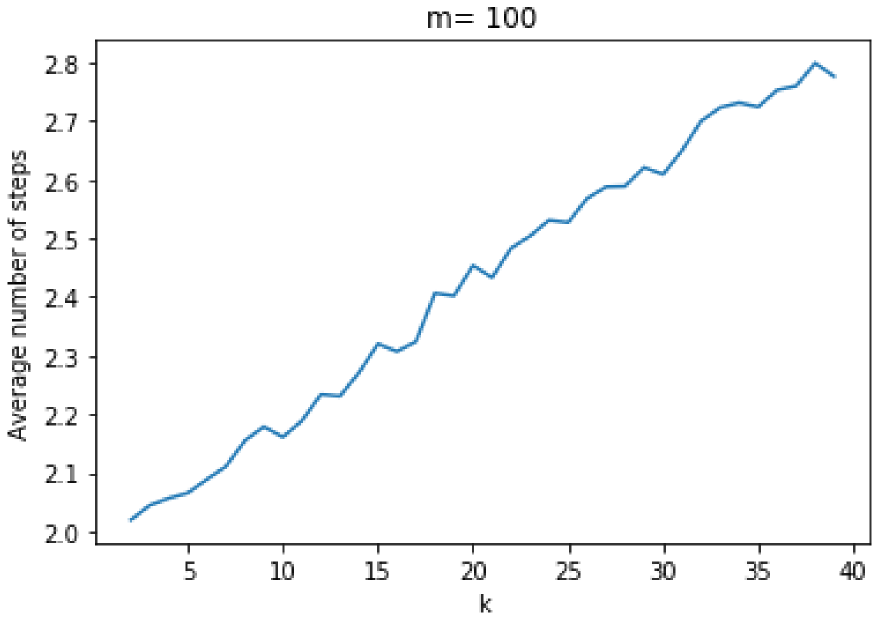
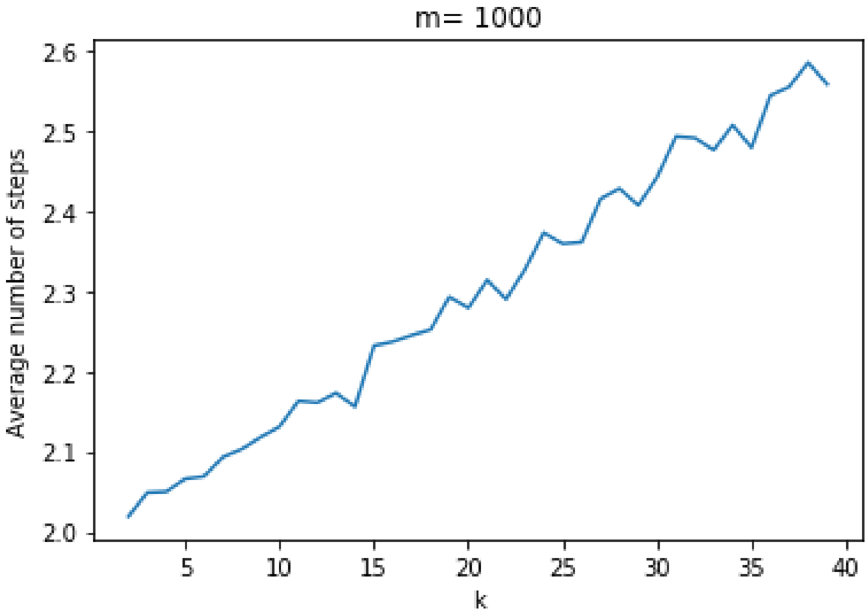


Fig 2. Number of steps v/s Ɛ

4) The number of steps as a function of the number of features k is shown in Fig 3 a) m=100 and b) m=1000. As can be seen, the number of steps increases with the increase in k. Even without the value of the output label having any dependence on X1, …, Xk-1, as an increase in k creates an increase in the number of constraints that the linear separator must satisfy, therefore, increasing the time taken to reach an answer. However, the number of steps does not vary much with respect to m.



(a)



(b)

Fig 3. Number of steps vs number of features k

5) The change in weights and bias is given by equation (1) which is derived from the weight update equation of a perceptron.

$$||w_T|| \le \sqrt{T} \quad (1)$$

Where $w_T$ is the magnitude of the weight vector after T steps.

One possible heuristic is the Euclidean distance between the closest pair of data points, where both the points belong to the different binary classes. Given this distance between the closest points as d, the margin Y can only be as large as d. We also know that the learning algorithm for a perceptron on linearly separable data satisfies the condition:

$$T \le \frac{1}{\max_{w} \gamma(w)^2}$$

Therefore, given d, the number of steps T must be less than $\frac{1}{d^2}$ exceeding which the algorithm can be terminated and the data can be concluded to be linearly inseparable. However, the closest pair of points problem is extremely computationally intensive. In a practical setting such a procedure may not be viable, and instead the number of steps can be capped at a threshold T.