Learning Activation in Neural Network

Problem -Abstract—The choice of Activation Functions (AF) has proven to be an important factor that affects the performance of an Artificial Neural Network (ANN). Use a 1-hidden layer neural network model that adapts to the most suitable activation function according to the data-set. The ANN model can learn for itself the best AF to use by exploiting a flexible functional form, k0 + k1 ∗ x with parameters k0, k1 being learned from multiple runs. You can use this code-base for implementation guidelines and help. <https://github.com/sahamath/MultiLayerPerceptron>

**A picture containing chart

Description automatically generatedFirst Model** - This Model Is Build Using Sigmoid Activation Function , A Sigmoid Function Is Constrained By A Pair Of [Horizontal Asymptotes](https://en.wikipedia.org/wiki/Horizontal_asymptotes) As X Tends To +- Infinity .

**Accuracy** - 0.9967213

**F1 Score** – 0.9967426

**Second model** – y = k0 +k1\*x, in this it is clear that k1 is a variable and k0 is a intercept

*On finding the best value for K1 and K0 ,used lots of iteration of data but best values is below.*

**Best value of k1** ,I use k1 = -300 and made loop to run 200 times with an increment of 5 ,assuming that k0 =0 ,by this best value of k1 is 5

And for **Best value of k0** , I use k0 = -300 and loop to run for 200 times with an increment of 5 , assuming that k1 = 5 , by this we get k0 = 5

![Graphical user interface, chart

Description automatically generated]()**Accuracy** = .9934426

**F1 score** = .9935064