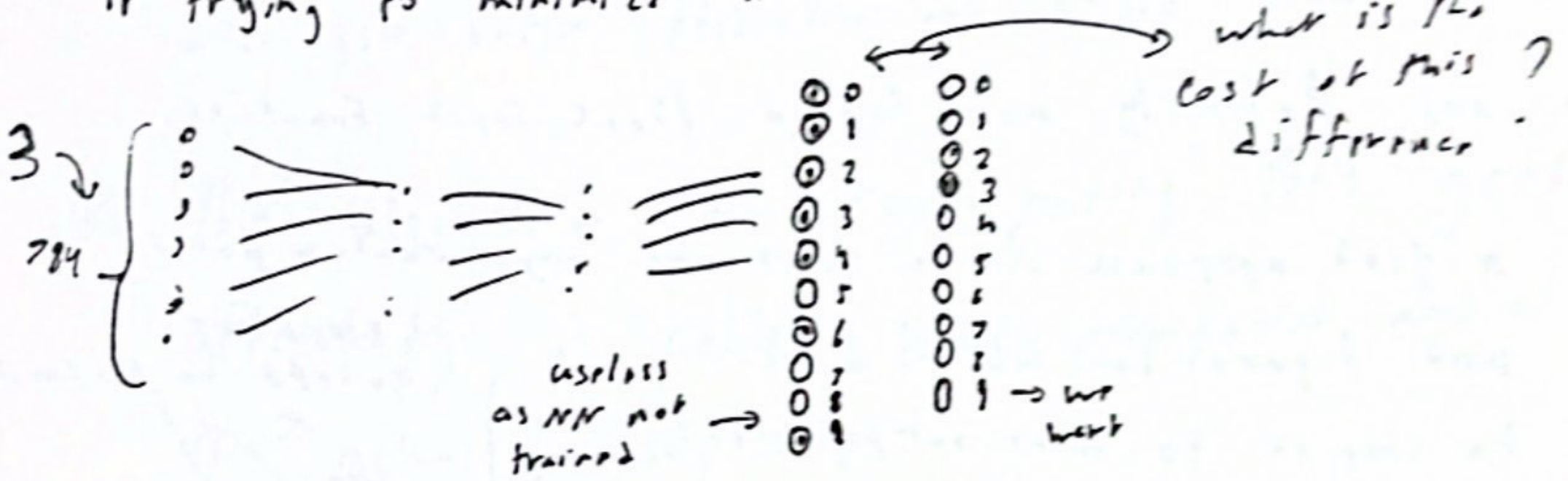
NN cont .. Gradient Decent

- Last parts where NN in action but it was trained how does NN train: we use an algorithm that take alot of images of numbers with there answers and tweaks those 13002 parameters so that given a new unseen image, it a number it can predict it.

- generally in NN we are trying to minimize the error it trying to minimize a cost function



- This NN's parameters are random so to train that we define a cost function a way of telling how wrong the output is:

- what we do is add up the squares of of differences between each of the bad random outputs and the out put we want, we know it since data is labeled

$$3:\begin{cases}(0.43-0.0)^2\\(0.52-0.0)^2\\(0.19-0.0)^2\\(0.20-1)^2\end{cases} \quad \text{Ex: } \sum = 3.37 \quad \text{(Cost of 3 ex)}$$

correct

- This Sum is small when the Network classifies img correctly and large if it doesn't

- So take avg cost of all traing datas cost and tells how good or bad our predictions are

NN input: 784 → alot!
out: 10
params: 13002

cost function: input: 13002 (params)
output: 1 (cost)
parameters: all traing Images