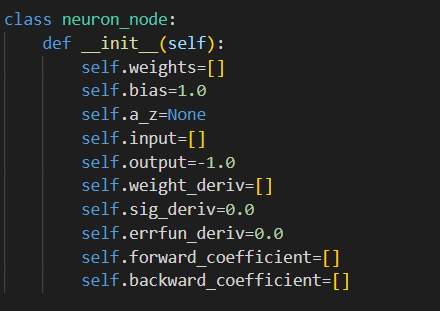
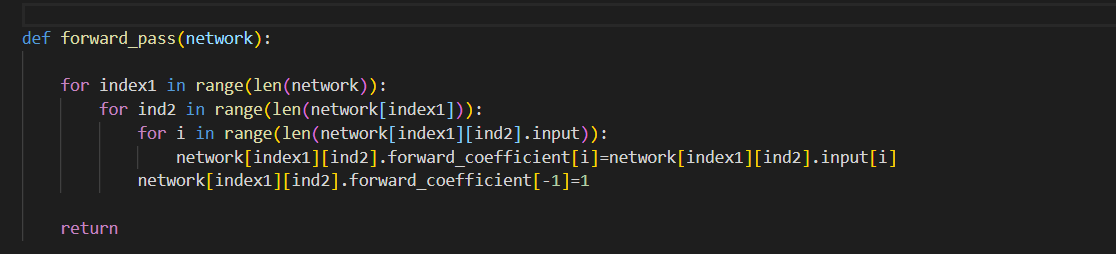
**Report**

**Data structure:**

****

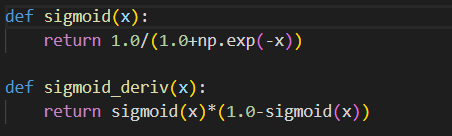
我自定義了一個structure neuron\_node來實作nruron 我的neuron\_network在code叫Network它包含了2個叫layer的list，第一層Layer有2個neurons，第二層有1個neuron，它除了weight和bias還儲存了一些東西，像是現在的input跟現在sigmoid\_deriv的值是多少。

**Forward propagation:**



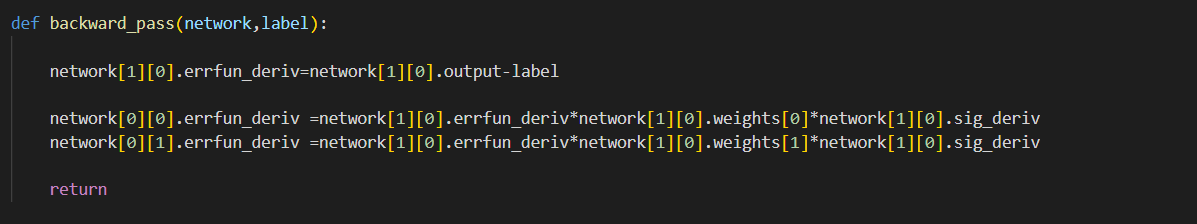
主要動作把weight function對各係數的微分記錄在forwar\_cefficient這個neuron的list裡。

Sigmoid function:



參考講義，基本我沒動到它。

**Back propagation:**



從後面開始算每個neuron對error function的微分，並記錄在errfun\_deriv裡。

**Update weights:**

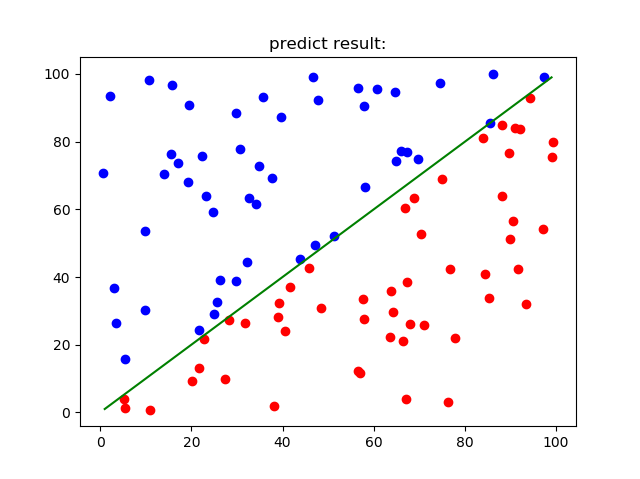


Weight\_deriv\_add會再跑資料迴圈時，讓每個neuron的weight\_derivate加上這次針對這筆資料計算出來的gradient。

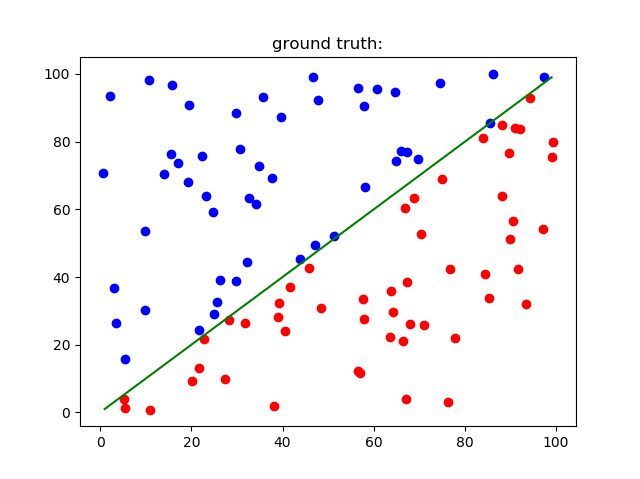
而下面的update\_weight，它是在跑完資料迴圈呼叫，是真正更新每個neuron的weights跟bias，剪掉學習率乘上我們已經算好的gradient。

predictions and ground truth:

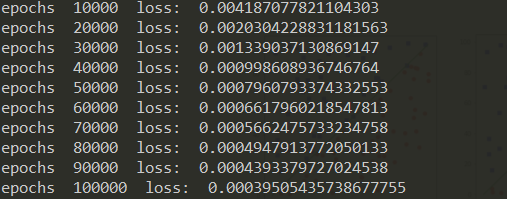
prediction:



Ground truth:



Loss



accuracy.

