

Analysis Report : Neural Networks

1. In your written response compare the result #1 to result #2 to the actual result explain your findings.

We can see from the screenshot below that result 1 is close to the actual result (0.3) and result 2 is very far off from the actual result. Result 1 is generated by (6,1) hidden layer neural network and we can see the error (0.001...) is very less and hence we can say that the model fits our training data and can give somewhat accurate results. On the other hand Result 2 is generated from (5,3,1) hidden layer neural network which even run for significantly more number of epoch but the error is very high (0.8....) which means the model don't fit the training data and hence give very inaccurate result which we can see in the screenshot.

```
Epoch: 435; Error: 0.0014025177209970913;
Epoch: 450; Error: 0.0014019838502640366;
Epoch: 465; Error: 0.001401506420723023;
Epoch: 480; Error: 0.00140111665177737;
Epoch: 495; Error: 0.0014007760568030547;
The maximum number of train epochs is reached
result 1: [[0.23540323]]
-----EX2-----
Epoch: 100; Error: 0.8282424876293307;
Epoch: 200; Error: 0.828242481502371;
Epoch: 300; Error: 0.8282424753730302;
Epoch: 400; Error: 0.828242469241308;
Epoch: 500; Error: 0.8282424631072021;
Epoch: 600; Error: 0.8282424569707106;
Epoch: 700; Error: 0.8282424508318335;
Epoch: 800; Error: 0.8282424446905685;
Epoch: 900; Error: 0.8282424385469145;
Epoch: 1000; Error: 0.8282424324008699;
The maximum number of train epochs is reached
result 2: [[-0.31855375]]
-----EX3-----
[[-0.09957359  0.26438939]
 [-0.59986275 -0.23720091]
 [-0.42389293 -0.48919369]]
```

2. In your written response compare result #1 to result #3 to the actual result explain your findings.

From the figure below we can see result 3 is very close to the actual result (0.3) compared to result 1. result 1 and result 3 are generated with the same structure of (6,1) neural network only difference being the quantity of training data result 1 consume 10 records and result 3 consumed 100 records which shows result 3 has more number of training data hence it can figure out the pattern in the data clearly even though the error is high compared to error of result 1. while for result 1 there are only 10 input data records and very low error but this didn't help the model to figure out the pattern in data because of very low input data.

```

Epoch: 270; Error: 0.013388099307813899;
Epoch: 285; Error: 0.01349586439843102;
Epoch: 300; Error: 0.013483983302801575;
Epoch: 315; Error: 0.013466598761570709;
Epoch: 330; Error: 0.013438903003688185;
Epoch: 345; Error: 0.0134338543088603;
Epoch: 360; Error: 0.013426790364464408;
Epoch: 375; Error: 0.013422521399035253;
Epoch: 390; Error: 0.013420212333164552;
Epoch: 405; Error: 0.013419535522221623;
Epoch: 420; Error: 0.01341919851030133;
Epoch: 435; Error: 0.01341900178149075;
Epoch: 450; Error: 0.013418594645627037;
Epoch: 465; Error: 0.013417134823143812;
Epoch: 480; Error: 0.013413616548623537;
Epoch: 495; Error: 0.013405862036894192;
The maximum number of train epochs is reached
result 3: [[0.30348696]]

```

```

-----EX4-----
Epoch: 100; Error: 1.630060344349876;
Epoch: 200; Error: 0.9793560109569944;
Epoch: 300; Error: 0.8050625486635882;
Epoch: 400; Error: 0.6932356671232284;
Epoch: 500; Error: 0.610001609099428;
Epoch: 600; Error: 0.5482479210051394;

```

3. In your written response compare the result #3 to result #4 to the actual result, explain your findings.

For result 3 and result 4 number input data records are same 100 but result 3 was generated by (6,1) neural network and result 4 is generated by (5,3,1) neural network. By looking at the error value we can clearly see that result 3 model fits the data more compare to result 4 data as error for result 3 is 0.01... and error for result 4 is 0.6.... This also reflects in the prediction value as result 3 is very close to actual value (0.3) while result 4 is not compared to result 3.

```

Epoch: 450; Error: 0.013418594645627037;
Epoch: 465; Error: 0.013417134823143812;
Epoch: 480; Error: 0.013413616548623537;
Epoch: 495; Error: 0.013405862036894192;
The maximum number of train epochs is reached
result 3: [[0.30348696]]

```

```

-----EX4-----
Epoch: 100; Error: 1.630060344349876;
Epoch: 200; Error: 0.9793560109569944;
Epoch: 300; Error: 0.8050625486635882;
Epoch: 400; Error: 0.6932356671232284;
Epoch: 500; Error: 0.610001609099428;
Epoch: 600; Error: 0.5482479210051394;
Epoch: 700; Error: 0.503382336309991;
Epoch: 800; Error: 0.4700086610417837;
Epoch: 900; Error: 0.497965106690875;
Epoch: 1000; Error: 0.6001253127261018;
The maximum number of train epochs is reached
result 4: [[0.58490558]]

```

```

-----EX5-----
[[-0.09957359  0.26438939 -0.59986275]
 [-0.23720091 -0.42389293 -0.48919369]
 [-0.37648775 -0.18532713 -0.12387903]
 [ 0.04658008 -0.09696658  0.2222634 ]

```

4. In your written response compare the result #5 to result #6 to the actual result, explain your findings.

Result 5 and result 6 consume same input records but result 5 has (6,1) layers and result 6 has (5,3,1) layers in neural network model. We see a major difference in epoch result 6 has way more epoch

compared to result 5 also result 6 is close to actual result (0.5) while result 5 is not compared to result 6. even though result 5 has low error it is giving inaccurate result because of less epoch which means less iteration to figure out the pattern while result 6 has higher error but it has more epoch which means it got more iterations to explore and learn the pattern. And hence we can see the result 6 is more accurate compared to result 5.

```
Epoch: 15; Error: 0.022571200000000000;
Epoch: 60; Error: 0.011873243513861783;
Epoch: 75; Error: 0.01176774066778043;
Epoch: 90; Error: 0.011588113190748045;
Epoch: 105; Error: 0.011335066324955585;
Epoch: 120; Error: 0.011293170859257084;
result 5: [[0.81619441]]
Epoch: 100; Error: 0.2561020458916132;
Epoch: 200; Error: 0.18655098637935405;
Epoch: 300; Error: 0.14283058257552633;
Epoch: 400; Error: 0.11955777954630695;
Epoch: 500; Error: 0.10391242642726549;
Epoch: 600; Error: 0.09115097694906976;
Epoch: 700; Error: 0.08028574763785995;
Epoch: 800; Error: 0.07104310955181968;
Epoch: 900; Error: 0.06319770112524373;
Epoch: 1000; Error: 0.05652413230393474;
The maximum number of train epochs is reached
result 6: [[0.45841217]]
-----All results-----
result 1: [[0.23540323]]    10 data points and structure(6,1)
result 2: [[-0.31855375]]  10 data points and structure(5,3,1)
result 3: [[0.30348696]]   100 data points and structure(6,1)
result 4: [[0.58490558]]   100 data points and structure(5,3,1)
result 5: [[0.81619441]]   10 data points, 3 inputs and structure(6,1)
```

5. Write some final conclusion

Overall from this exercise it I concluded that in order to build a neural network all of the following aspects are very important:

- Number of input record count
- Number of hidden layers in network (Network structure)
- Number of epochs

If any of the above is not appropriate to the given data then it won't give fruitful result and will provide inaccurate predictions.