Depth OR Width?

Yuxuan luo,Zhongwu Xie

Shenzhen University

2018.9.19

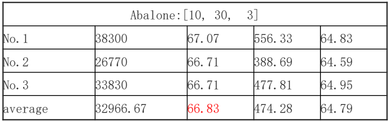
Outlines

1. Previous Work
   - Using the UCI datasets
   - Create the more complicated datasets
   - Investigate the approximation by superposotions of a sigmoid function

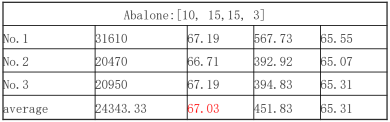
2. Future work

Using the UCI datasets

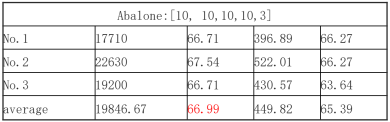
- Abalone(4177)



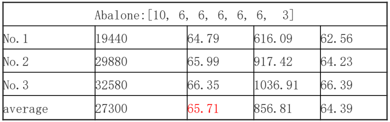
Abalone:[10, 30, 3]				
No. 1	38300	67.07	556.33	64.83
No. 2	26770	66.71	388.69	64.59
No. 3	33830	66.71	477.81	64.95
average	32966.67	66.83	474.28	64.79



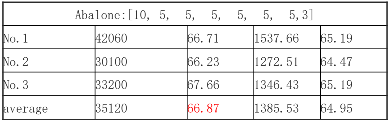
Abalone:[10, 15,15, 3]				
No. 1	31610	67.19	567.73	65.55
No. 2	20470	66.71	392.92	65.07
No. 3	20950	67.19	394.83	65.31
average	24343.33	67.03	451.83	65.31



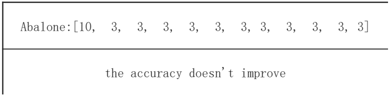
Abalone:[10, 10,10,10,3]				
No. 1	17710	66.71	396.89	66.27
No. 2	22630	67.54	522.01	66.27
No. 3	19200	66.71	430.57	63.64
average	19846.67	66.99	449.82	65.39



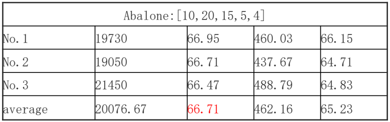
Abalone:[10, 6, 6, 6, 6, 6, 3]				
No. 1	19440	64.79	616.09	62.56
No. 2	29880	65.99	917.42	64.23
No. 3	32580	66.35	1036.91	66.39
average	27300	65.71	856.81	64.39



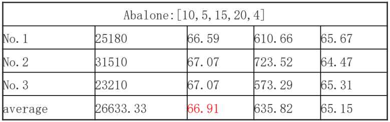
Abalone:[10, 5, 5, 5, 5, 5, 5,3]				
No. 1	42060	66.71	1537.66	65.19
No. 2	30100	66.23	1272.51	64.47
No. 3	33200	67.66	1346.43	65.19
average	35120	66.87	1385.53	64.95



Abalone:[10, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3]
the accuracy doesn't improve



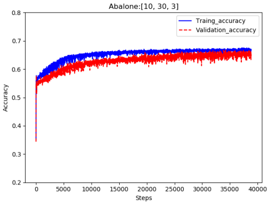
Abalone:[10,20,15,5,4]				
No. 1	19730	66.95	460.03	66.15
No. 2	19050	66.71	437.67	64.71
No. 3	21450	66.47	488.79	64.83
average	20076.67	66.71	462.16	65.23

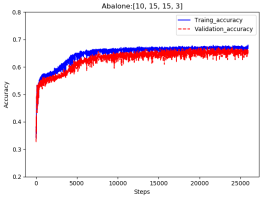


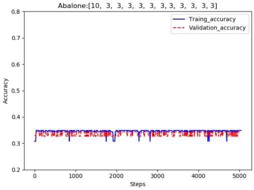
Abalone:[10,5,15,20,4]				
No. 1	25180	66.59	610.66	65.67
No. 2	31510	67.07	723.52	64.47
No. 3	23210	67.07	573.29	65.31
average	26633.33	66.91	635.82	65.15

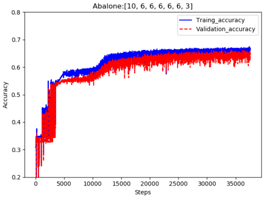
Using the UCI datasets

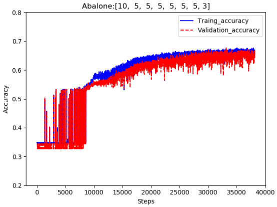
- Abalone(4177)





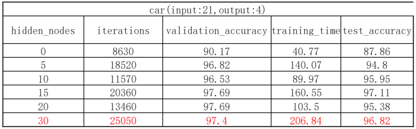




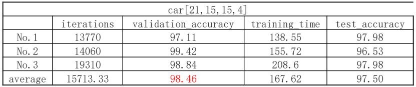


Using the UCI datasets

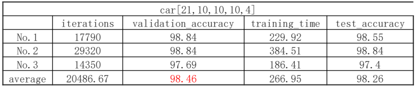
- Car(1728)



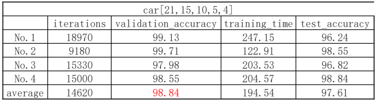
car(input:21,output:4)				
hidden_nodes	iterations	validation_accuracy	training_time	test_accuracy
0	8630	90.17	40.77	87.86
5	18520	96.82	140.07	94.8
10	11570	96.53	89.97	95.95
15	20360	97.69	160.55	97.11
20	13460	97.69	103.5	95.38
30	25050	97.4	206.84	96.82



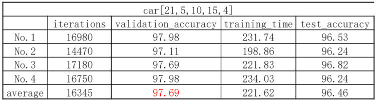
car[21,15,15,4]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	13770	97.11	138.55	97.98
No.2	14060	99.42	155.72	96.53
No.3	19310	98.84	208.6	97.98
average	15713.33	98.46	167.62	97.50



car[21,10,10,10,4]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	17790	98.84	229.92	98.55
No.2	29320	98.84	384.51	98.84
No.3	14350	97.69	186.41	97.4
average	20486.67	98.46	266.95	98.26



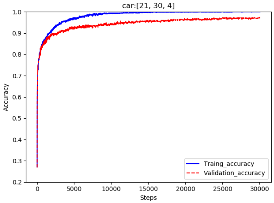
car[21,15,10,5,4]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	18970	99.13	247.15	96.24
No.2	9180	99.71	122.91	98.55
No.3	15330	97.98	203.53	96.82
No.4	15000	98.55	204.57	98.84
average	14620	98.84	194.54	97.61

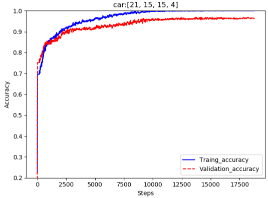


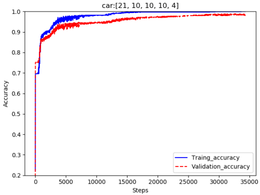
car[21,5,10,15,4]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	16980	97.98	231.74	96.53
No.2	14470	97.11	198.86	96.24
No.3	17180	97.69	221.83	96.82
No.4	16750	97.98	234.03	96.24
average	16345	97.69	221.62	96.46

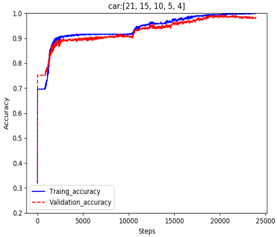
Using the UCI datasets

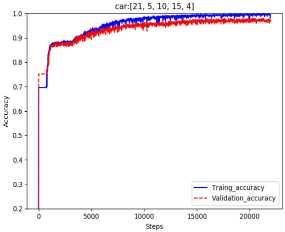
- Car(1728)



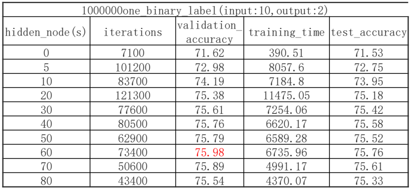






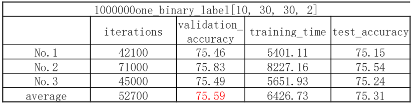


- 1000000one_binary_label(1000000)



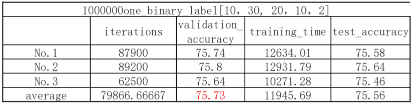
1000000one_binary_label(input:10,output:2)

hidden_node(s)	iterations	validation_accuracy	training_time	test_accuracy
0	7100	71.62	390.51	71.53
5	101200	72.98	8057.6	72.75
10	83700	74.19	7184.8	73.95
20	121300	75.38	11475.05	75.18
30	77600	75.61	7254.06	75.42
40	80500	75.76	6620.17	75.58
50	62900	75.79	6589.28	75.52
60	73400	75.98	6735.96	75.76
70	50600	75.89	4991.17	75.61
80	43400	75.54	4370.07	75.33



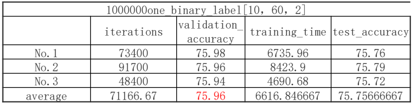
1000000one_binary_label[10，30，30，2]

	iterations	validation_accuracy	training_time	test_accuracy
No.1	42100	75.46	5401.11	75.15
No.2	71000	75.83	8227.16	75.54
No.3	45000	75.49	5651.93	75.24
average	52700	75.59	6426.73	75.31



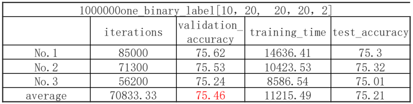
1000000one_binary_label[10，30，20，10，2]

	iterations	validation_accuracy	training_time	test_accuracy
No.1	87900	75.74	12634.01	75.58
No.2	89200	75.8	12931.79	75.64
No.3	62500	75.64	10271.28	75.46
average	79866.66667	75.73	11945.69	75.56



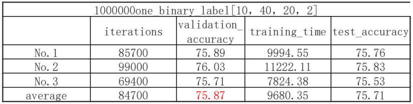
1000000one_binary_label[10，60，2]

	iterations	validation_accuracy	training_time	test_accuracy
No.1	73400	75.98	6735.96	75.76
No.2	91700	75.96	8423.9	75.79
No.3	48400	75.94	4690.68	75.72
average	71166.67	75.96	6616.846667	75.75666667



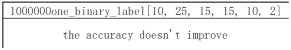
1000000one_binary_label[10，20，20，20，2]

	iterations	validation_accuracy	training_time	test_accuracy
No.1	85000	75.62	14636.41	75.3
No.2	71300	75.53	10423.53	75.32
No.3	56200	75.24	8586.54	75.01
average	70833.33	75.46	11215.49	75.21



1000000one_binary_label[10，40，20，2]

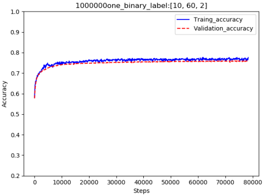
	iterations	validation_accuracy	training_time	test_accuracy
No.1	85700	75.89	9994.55	75.76
No.2	99000	76.03	11222.11	75.83
No.3	69400	75.71	7824.38	75.53
average	84700	75.87	9680.35	75.71

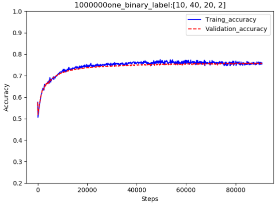


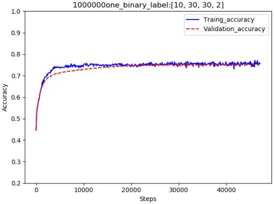
1000000one_binary_label[10，25，15，15，10，2]

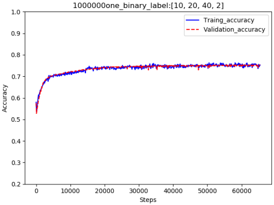
the accuracy doesn't improve

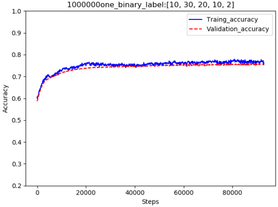
- 1000000one_binary_label(1000000)

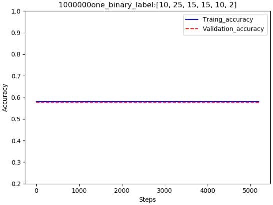

1000000one_binary_label:[10, 60, 2]


1000000one_binary_label:[10, 40, 20, 2]


1000000one_binary_label:[10, 30, 30, 2]


1000000one_binary_label:[10, 20, 40, 2]


1000000one_binary_label:[10, 30, 20, 10, 2]

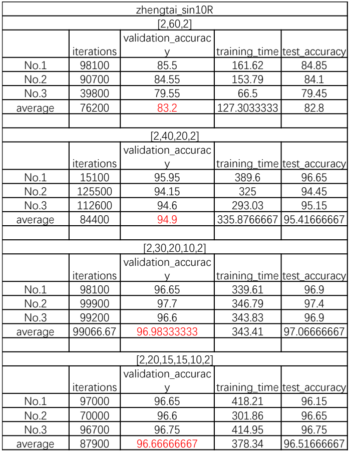

1000000one_binary_label:[10, 25, 15, 15, 10, 2]

Outlines

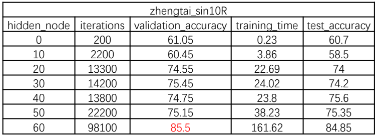
1. Previous Work
   - Using the UCI datasets
   - Create the more complicated datasets
   - Investigate the approximation by superposotions of a sigmoid function
2. Future work

More complicate datasets

- Zhengtai_sin(10R)



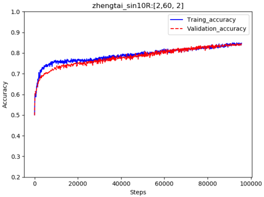
zhengtai_sin10R				
[2,60,2]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	98100	85.5	161.62	84.85
No.2	90700	84.55	153.79	84.1
No.3	39800	79.55	66.5	79.45
average	76200	83.2	127.3033333	82.8
[2,40,20,2]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	15100	95.95	389.6	96.65
No.2	125500	94.15	325	94.45
No.3	112600	94.6	293.03	95.15
average	84400	94.9	335.8766667	95.41666667
[2,30,20,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	98100	96.65	339.61	96.9
No.2	99900	97.7	346.79	97.4
No.3	99200	96.6	343.83	96.9
average	99066.67	96.98333333	343.41	97.06666667
[2,20,15,15,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	97000	96.65	418.21	96.15
No.2	70000	96.6	301.86	96.65
No.3	96700	96.75	414.95	96.75
average	87900	96.66666667	378.34	96.51666667

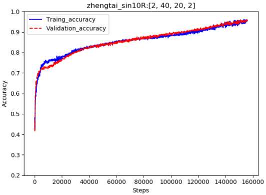


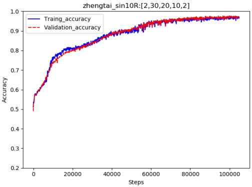
zhengtai_sin10R				
hidden_node	iterations	validation_accuracy	training_time	test_accuracy
0	200	61.05	0.23	60.7
10	2200	60.45	3.86	58.5
20	13300	74.55	22.69	74
30	14200	75.45	24.02	74.2
40	13800	74.75	23.8	75.6
50	22200	75.15	38.23	75.35
60	98100	85.5	161.62	84.85

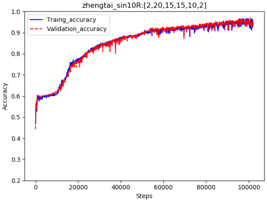
More complicate datasets

- Zhengtai_sin(10R)



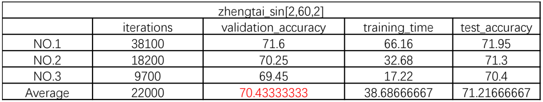




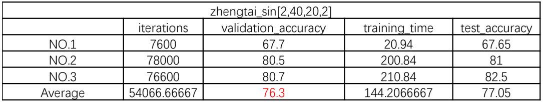


More complicate datasets

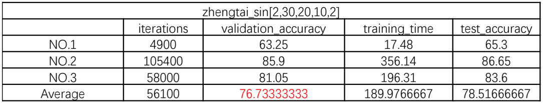
- Zhengtai_sin(10R)_R3



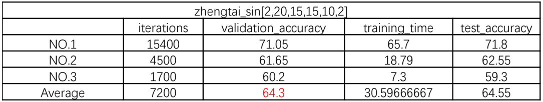
zhengtai_sin[2,60,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	38100	71.6	66.16	71.95
NO.2	18200	70.25	32.68	71.3
NO.3	9700	69.45	17.22	70.4
Average	22000	70.43333333	38.68666667	71.21666667



zhengtai_sin[2,40,20,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	7600	67.7	20.94	67.65
NO.2	78000	80.5	200.84	81
NO.3	76600	80.7	210.84	82.5
Average	54066.66667	76.3	144.2066667	77.05



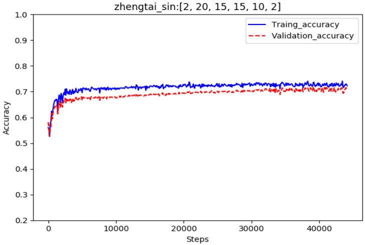
zhengtai_sin[2,30,20,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	4900	63.25	17.48	65.3
NO.2	105400	85.9	356.14	86.65
NO.3	58000	81.05	196.31	83.6
Average	56100	76.73333333	189.9766667	78.51666667

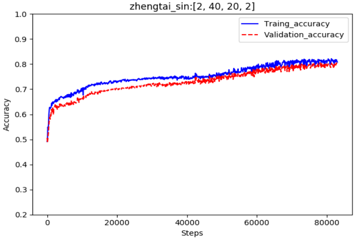


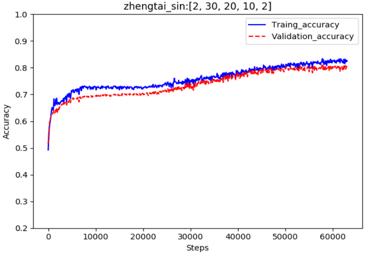
zhengtai_sin[2,20,15,15,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	15400	71.05	65.7	71.8
NO.2	4500	61.65	18.79	62.55
NO.3	1700	60.2	7.3	59.3
Average	7200	64.3	30.59666667	64.55

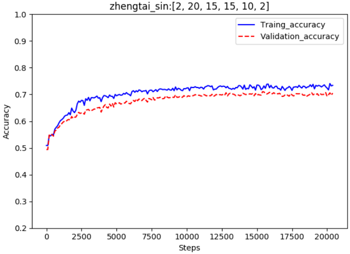
More complicate datasets

- Zhengtai_sin(10R)_R3



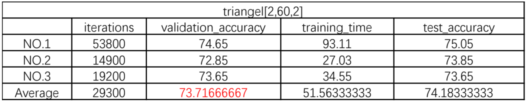




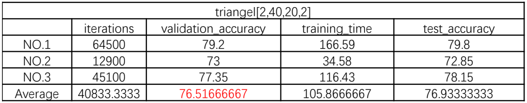


More complicate datasets

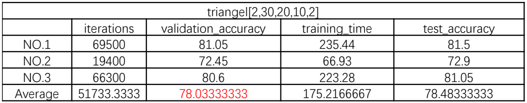
- Triangel



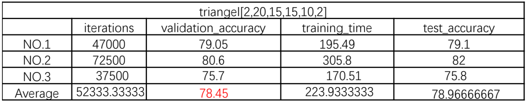
triangel[2,60,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	53800	74.65	93.11	75.05
NO.2	14900	72.85	27.03	73.85
NO.3	19200	73.65	34.55	73.65
Average	29300	73.71666667	51.56333333	74.18333333



triangel[2,40,20,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	64500	79.2	166.59	79.8
NO.2	12900	73	34.58	72.85
NO.3	45100	77.35	116.43	78.15
Average	40833.3333	76.51666667	105.8666667	76.93333333



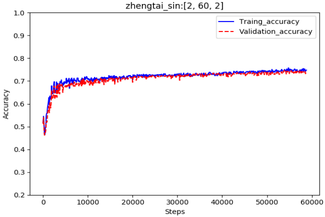
triangel[2,30,20,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	69500	81.05	235.44	81.5
NO.2	19400	72.45	66.93	72.9
NO.3	66300	80.6	223.28	81.05
Average	51733.3333	78.03333333	175.2166667	78.48333333

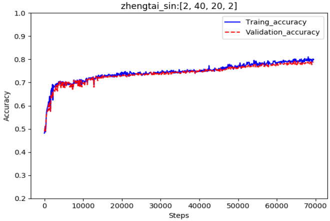


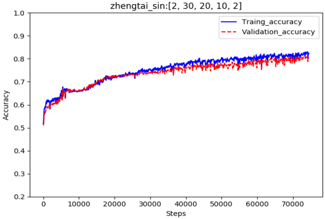
triangel[2,20,15,15,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	47000	79.05	195.49	79.1
NO.2	72500	80.6	305.8	82
NO.3	37500	75.7	170.51	75.8
Average	52333.33333	78.45	223.9333333	78.96666667

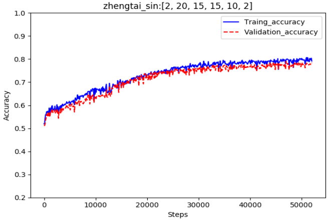
More complicate datasets

- Triangel


zhengtai_sin:[2, 60, 2]


zhengtai_sin:[2, 40, 20, 2]


zhengtai_sin:[2, 30, 20, 10, 2]


zhengtai_sin:[2, 20, 15, 15, 10, 2]

More complicate datasets

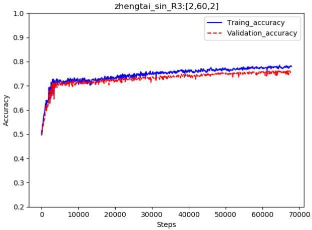
- Triangel_sin10x

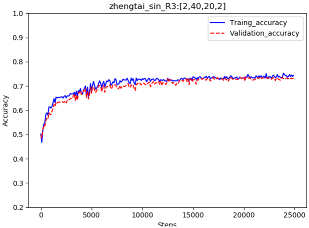
triangel_sin10x_				
[2,60,2]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	62800	76.4	107.03	77.45
No.2	19500	72.4	34.32	72.35
No.3	9000	72.35	16.19	72.9
average	30433.33	73.71666667	52.51333333	74.23333333
[2,40,20,2]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	43500	77.1	114.51	78.15
No.2	9900	73.6	27.35	73.5
No.3	18600	73.5	49.73	73.05
average	24000	74.73333333	63.86333333	74.9
[2,30,20,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	62900	79.6	217.61	81.2
No.2	67300	81.9	234.63	81.6
No.3	62800	81.45	219.11	82
average	64333.33	80.98333333	223.7833333	81.6
[2,20,15,15,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	38500	76.9	166.64	77.2
No.2	77000	81.25	332.72	80.5
No.3	23800	71.8	102.69	71.65
average	46433.33	76.65	200.6833333	76.45

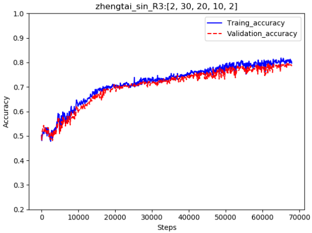
triangel_sin10x_				
hidden_node	iterations	validation_accuracy	training_time	test_accuracy
0	10800	56.75	8.35	56.2
10	19600	65.7	32.38	65.8
20	31500	70.75	51.51	71.35
30	9500	70.75	16.03	71.45
40	5300	71.05	9.49	71.5
50	17500	72.5	30.37	73
60	15300	72.25	26.62	73.7
70	32800	74.9	60.62	75.8
80	7200	72.85	13.8	72.6
100	54200	77.8	101.13	78.8

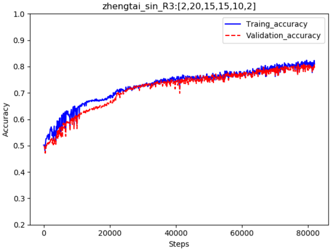
More complicate datasets

- Triangel_sin10x


zhengtai_sin_R3:[2,60,2]

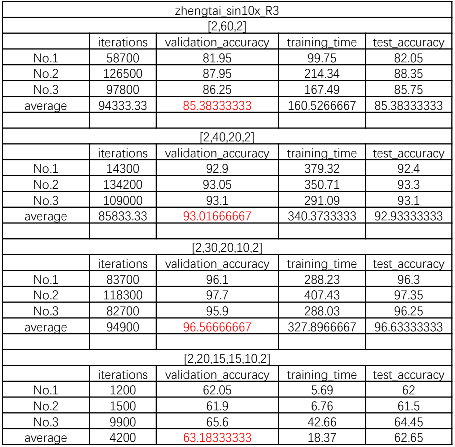

zhengtai_sin_R3:[2,40,20,2]


zhengtai_sin_R3:[2, 30, 20, 10, 2]

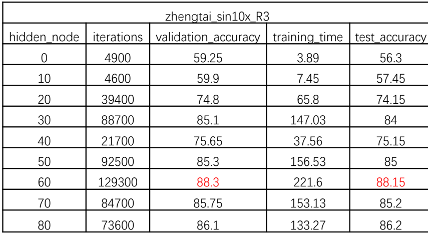

zhengtai_sin_R3:[2,20,15,15,10,2]

More complicate datasets

- Zhengtai_sin10x_R3



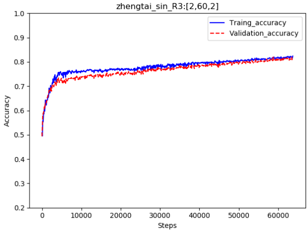
zhengtai_sin10x_R3				
[2,60,2]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	58700	81.95	99.75	82.05
No.2	126500	87.95	214.34	88.35
No.3	97800	86.25	167.49	85.75
average	94333.33	85.38333333	160.5266667	85.38333333
[2,40,20,2]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	14300	92.9	379.32	92.4
No.2	134200	93.05	350.71	93.3
No.3	109000	93.1	291.09	93.1
average	85833.33	93.01666667	340.3733333	92.93333333
[2,30,20,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	83700	96.1	288.23	96.3
No.2	118300	97.7	407.43	97.35
No.3	82700	95.9	288.03	96.25
average	94900	96.56666667	327.8966667	96.63333333
[2,20,15,15,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
No.1	1200	62.05	5.69	62
No.2	1500	61.9	6.76	61.5
No.3	9900	65.6	42.66	64.45
average	4200	63.18333333	18.37	62.65

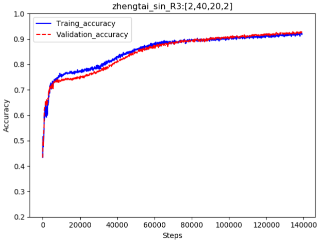


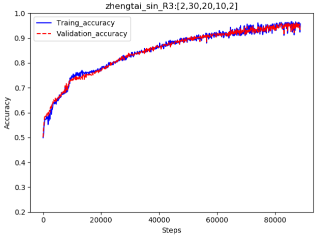
zhengtai_sin10x_R3				
hidden_node	iterations	validation_accuracy	training_time	test_accuracy
0	4900	59.25	3.89	56.3
10	4600	59.9	7.45	57.45
20	39400	74.8	65.8	74.15
30	88700	85.1	147.03	84
40	21700	75.65	37.56	75.15
50	92500	85.3	156.53	85
60	129300	88.3	221.6	88.15
70	84700	85.75	153.13	85.2
80	73600	86.1	133.27	86.2

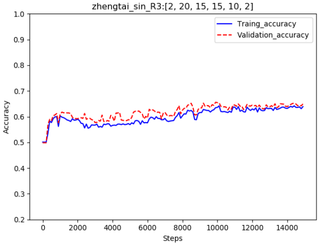
More complicate datasets

- Zhengtai_sin10x_R3



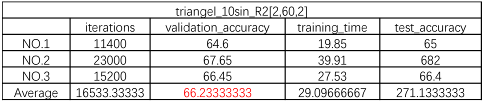




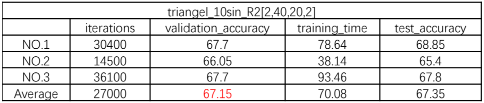


More complicate datasets

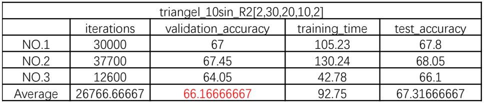
- Triangel_10sin_R2



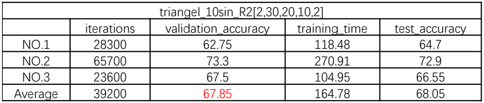
triangel_10sin_R2[2,60,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	11400	64.6	19.85	65
NO.2	23000	67.65	39.91	682
NO.3	15200	66.45	27.53	66.4
Average	16533.33333	66.23333333	29.09666667	271.1333333



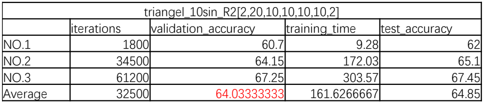
triangel_10sin_R2[2,40,20,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	30400	67.7	78.64	68.85
NO.2	14500	66.05	38.14	65.4
NO.3	36100	67.7	93.46	67.8
Average	27000	67.15	70.08	67.35



triangel_10sin_R2[2,30,20,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	30000	67	105.23	67.8
NO.2	37700	67.45	130.24	68.05
NO.3	12600	64.05	42.78	66.1
Average	26766.66667	66.16666667	92.75	67.31666667



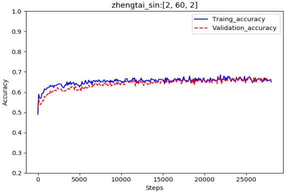
triangel_10sin_R2[2,30,20,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	28300	62.75	118.48	64.7
NO.2	65700	73.3	270.91	72.9
NO.3	23600	67.5	104.95	66.55
Average	39200	67.85	164.78	68.05

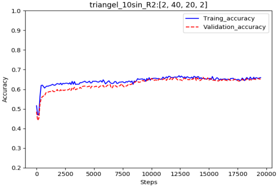


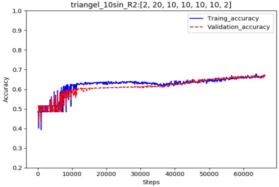
triangel_10sin_R2[2,20,10,10,10,10,2]				
	iterations	validation_accuracy	training_time	test_accuracy
NO.1	1800	60.7	9.28	62
NO.2	34500	64.15	172.03	65.1
NO.3	61200	67.25	303.57	67.45
Average	32500	64.03333333	161.6266667	64.85

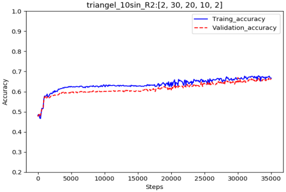
More complicate datasets

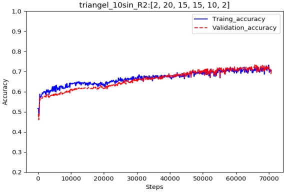
- Triangel_10sin_R2


zhengtai_sin:[2, 60, 2]


triangel_10sin_R2:[2, 40, 20, 2]


triangel_10sin_R2:[2, 20, 10, 10, 10, 10, 2]


triangel_10sin_R2:[2, 30, 20, 10, 2]


triangel_10sin_R2:[2, 20, 15, 15, 10, 2]

Outlines

1. Previous Work
   - Using the UCI datasets
   - Create the more complicated datasets
   - Investigate the approximation by superposotions of a sigmoid function
2. Future work

http://neuralnetworksanddeeplearning.com/chap4.html

Future work

- Try to analyze the NN with more hidden layers.
- Learn to calculate the complexity of the dataset.
- Implement the CNN.

Future work

- Try to analyze the NN with more hidden layers.
- Learn to calculate the complexity of the dataset.
- Implement the CNN.

Future work

- Try to analyze the NN with more hidden layers.
- Learn to calculate the complexity of the dataset.
- Implement the CNN.