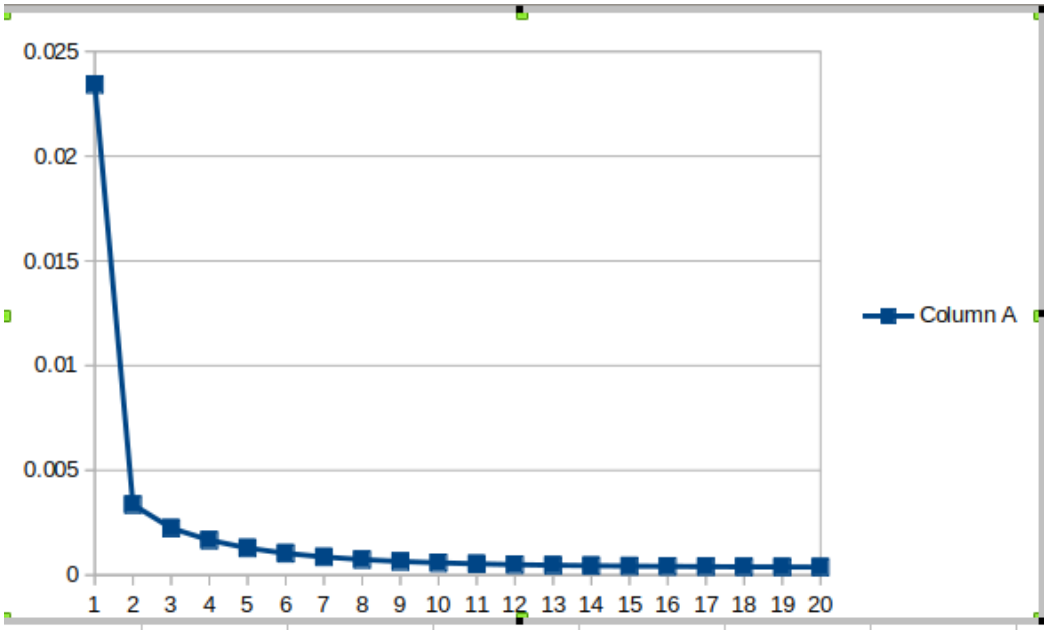
Analysis of neural network with one hidden layer

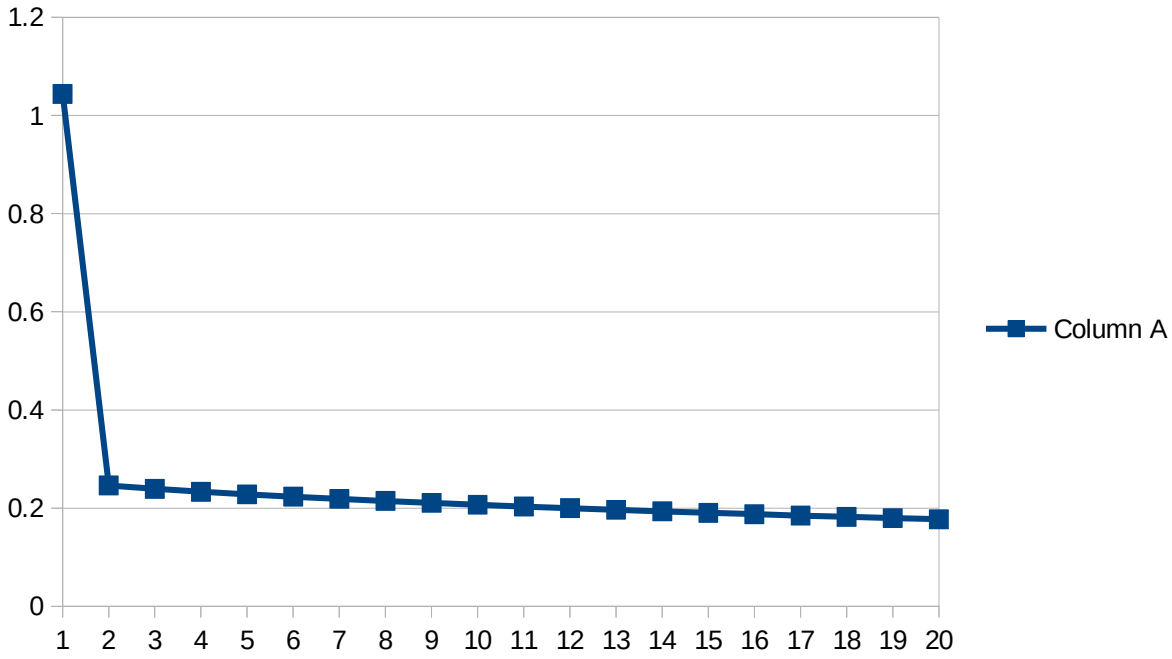
by: Olenburg Egor.

number of training patterns = 60000
learning rate = 0.100000
number of hid layers = 10;
0.028452
0.000666
0.000591
0.000541
0.000508
0.000486
0.000471
0.000460
0.000453
0.000447
0.000443
0.000439
0.000436
0.000433
0.000431
0.000429
0.000427
0.000426
0.000424
0.000423



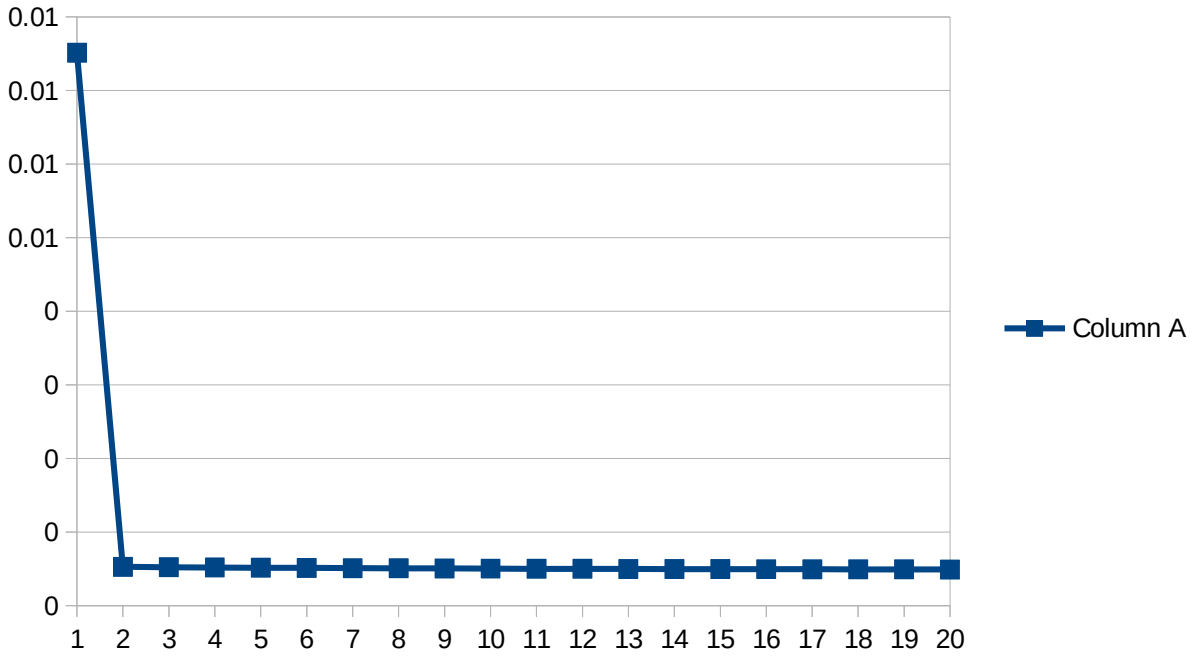
With twenty cycles it seems like on 8th epoch my program reaches assymptote.
I used 60000 patterns and 20 epochs to train my weights, further I used 20 epochs to train 10000 images on these weights in each outer cycle! And these are the numbers that is got to!

number of training patterns = 60000
learning rate = 0.000100
number of hid layers = 10;
1.044137
0.246108
0.239311
0.233393
0.228104
0.223279
0.218806
0.214611
0.210649
0.206888
0.203304
0.199877
0.196594
0.193448
0.190437
0.187558
0.184811
0.182192
0.179699
0.177330



After changing my learning rate from 0.1 to 0.0001 my graph changed much.
My error grew drastically. Although everything was left the same!

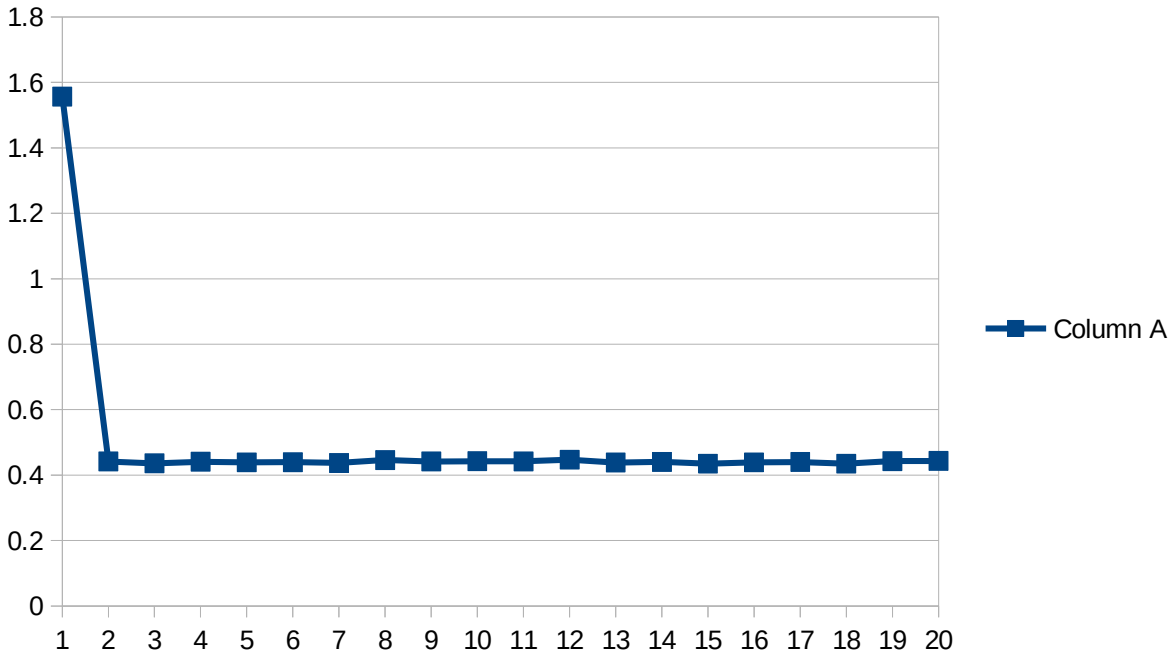
number of training patterns = 60000
learning rate = 0.500000
number of hid layers = 10;
0.007513
0.000526
0.000521
0.000517
0.000514
0.000511
0.000508
0.000506
0.000504
0.000502
0.000500
0.000499
0.000497
0.000496
0.000495
0.000494
0.000493
0.000492
0.000491
0.000490



Using learning rate 0.5 graph seems to be close to 0.1 learning rate, although 0.1 rate's output is a bit smaller.

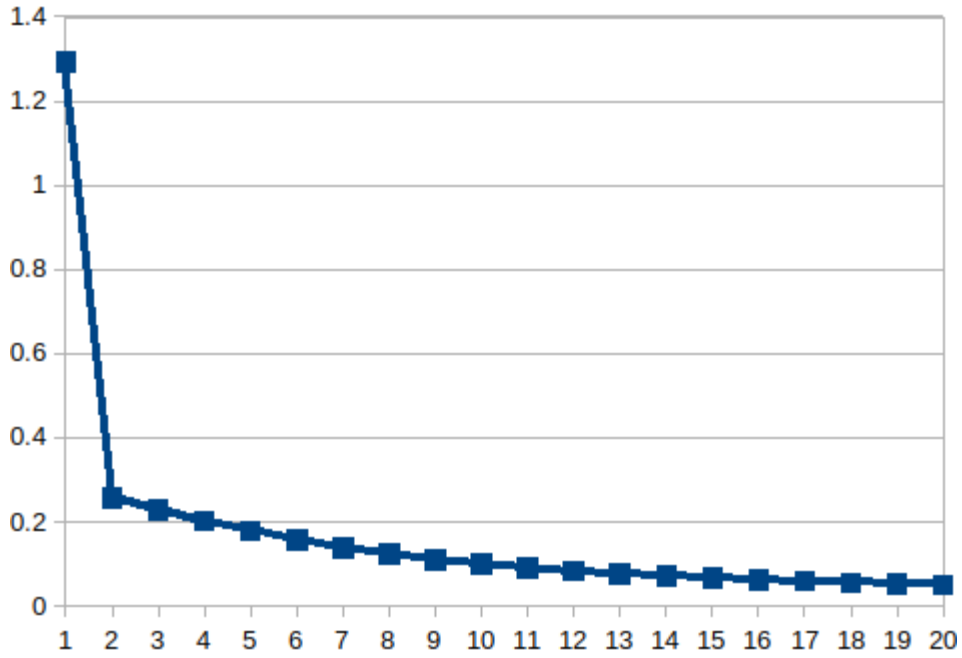
Seems like 0.1 learning rate is the optimal rate to get the lowest error possible using 20 epochs at about 8th I get the minimum error which is 0.000456.

number of training patterns = 60000
learning rate = 0.500000
number of hid layers = 323;
1.556516
0.441515
0.436138
0.441012
0.439083
0.439687
0.437313
0.445960
0.441813
0.442571
0.442059
0.447275
0.438579
0.440585
0.434949
0.439044
0.440283
0.434991
0.442789
0.443151



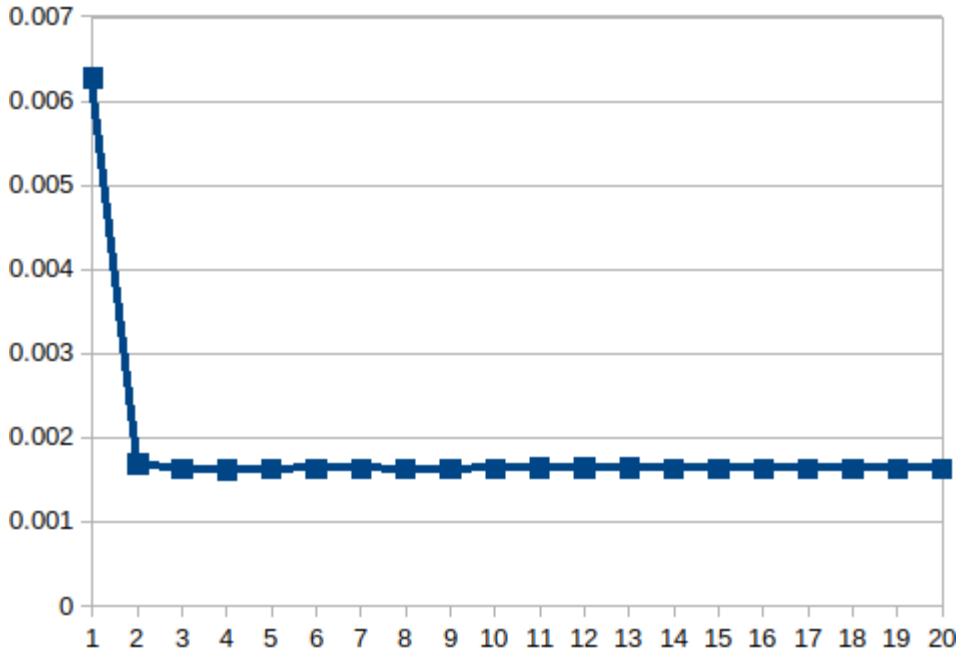
I graph has 323 hidden layers and learning rate is .5
My approximate error in this graph seems to be 0.4.
it is a very high error because my learning rate is very big!

number of training patterns = 60000
learning rate = 0.010000
number of hid layers = 323;
1.293239
0.257495
0.228305
0.202043
0.178416
0.157386
0.139027
0.123337
0.110144
0.099154
0.090012
0.082388
0.075992
0.070580
0.065955
0.061964
0.058487
0.055430
0.052720
0.050301



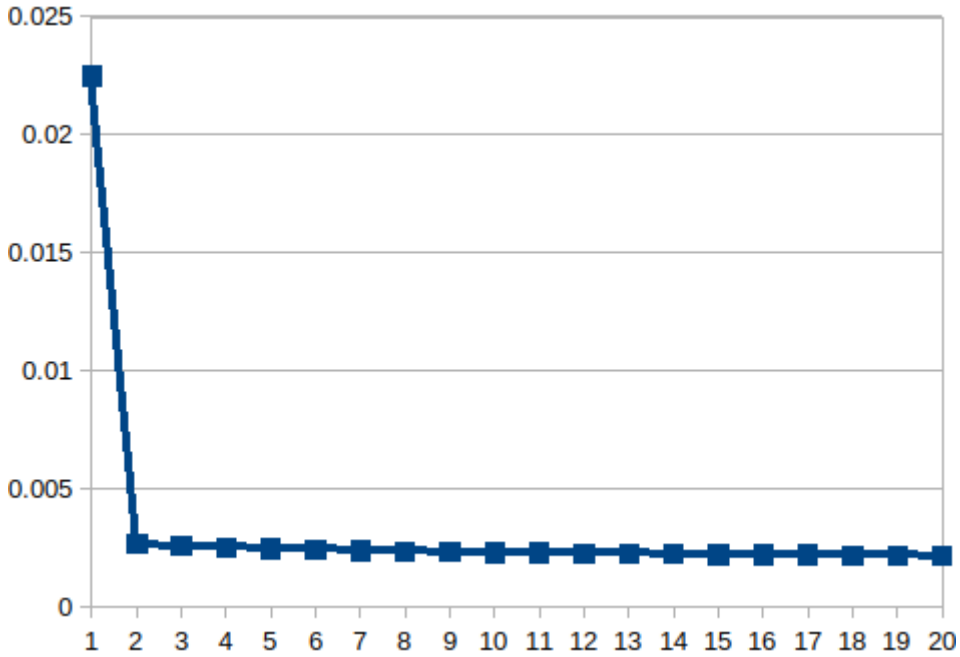
I this graph I use 323 hidden layers although my learning rate is much smaller then the previous graphs. Error got approximately to 0.05. Which in compare to 10 hidden layers error was 0.004
It is clearly seen that when my hidden layer size was 10 the error was much smaller!

number of training patterns = 60000
learning rate = 0.500000
number of hid layers = 785
0.006273
0.001691
0.001627
0.001618
0.001627
0.001632
0.001629
0.001624
0.001625
0.001630
0.001635
0.001639
0.001636
0.001634
0.001634
0.001633
0.001632
0.001632
0.001633
0.001633



In comparing hidden layer number between 785 and 323, error in this layer it is much smaller. Having learning rate of 0.500000.

number of training patterns = 60000
learning rate = 0.010000
number of hid layers = 785
0.022500
0.002661
0.002572
0.002510
0.002462
0.002423
0.002389
0.002360
0.002334
0.002311
0.002290
0.002271
0.002254
0.002238
0.002223
0.002209
0.002195
0.002183
0.002172
0.002161



In comparing hidden layer number between 785 and 323, error in this layer it is much smaller.
Having learning rate of 0.01

My Lowest error was at learning rate 0.1 and hid layer size of 10 = 0.000423

Odd/Even prediction did not influince my network!