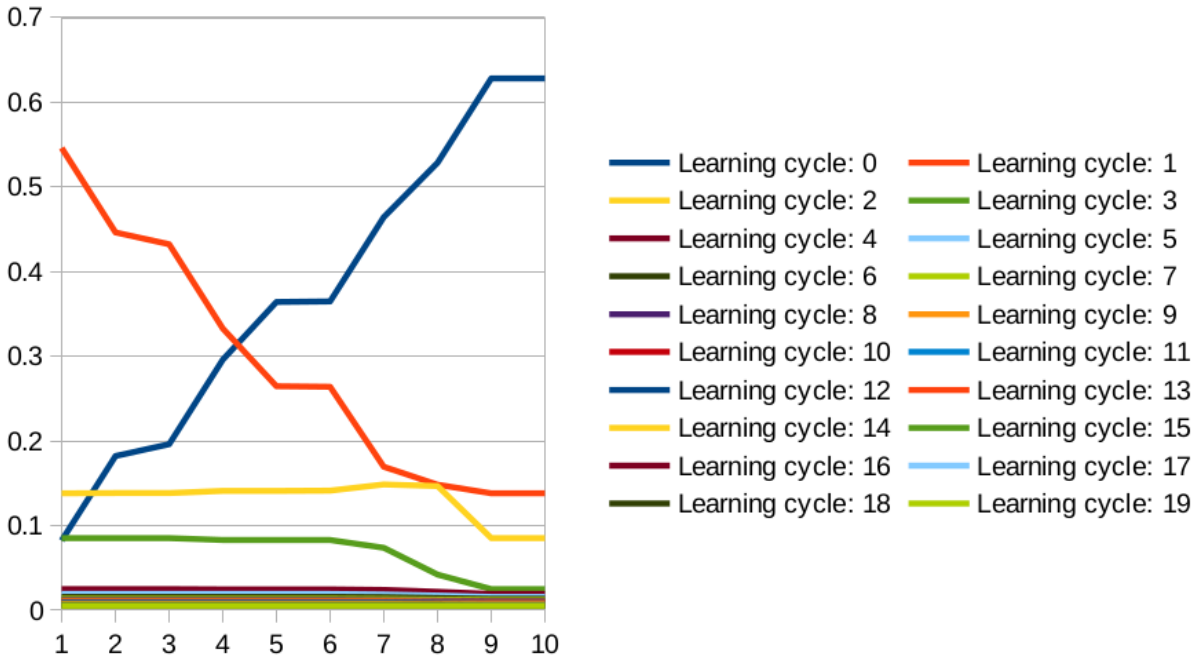
Neural Network Analysis

by: Olenburg Egor,

learning rate = 0.05
Noise = 0

Learning cycle: 0	Learning cycle: 1	Learning cycle: 2	Learning cycle: 3	Learning cycle: 4	Learning cycle: 5	Learning cycle: 6	Learning cycle: 7	Learning cycle: 8	Learning cycle: 9
0.082116	0.546117	0.138216	0.085259	0.024856	0.019165	0.015824	0.013573	0.011934	0.010679
0.182116	0.446134	0.138272	0.085203	0.024853	0.019163	0.015823	0.013572	0.011933	0.010678
0.195978	0.432272	0.138272	0.085203	0.024853	0.019163	0.015823	0.013572	0.011933	0.010678
0.29593	0.332729	0.140985	0.083032	0.024766	0.019107	0.015781	0.013538	0.011905	0.010653
0.364204	0.264471	0.141005	0.083009	0.024764	0.019106	0.01578	0.013538	0.011904	0.010652
0.364761	0.263971	0.14107	0.082969	0.024761	0.019104	0.015778	0.013536	0.011903	0.010651
0.464297	0.169412	0.148436	0.073768	0.024075	0.018682	0.015486	0.01332	0.011735	0.010517
0.528439	0.14813	0.146572	0.042148	0.021603	0.017297	0.014584	0.01268	0.011255	0.010143
0.628223	0.138196	0.085276	0.024857	0.019166	0.015825	0.013573	0.011934	0.010679	0.009682
0.628231	0.138212	0.085261	0.024856	0.019165	0.015824	0.013573	0.011934	0.010679	0.009682

Learning cycle: 10	Learning cycle: 11	Learning cycle: 12	Learning cycle: 13	Learning cycle: 14	Learning cycle: 15	Learning cycle: 16	Learning cycle: 17	Learning cycle: 18	Learning cycle: 19
0.009682	0.008869	0.008192	0.007618	0.007124	0.006695	0.006318	0.005984	0.005686	0.005418
0.009682	0.008869	0.008191	0.007617	0.007124	0.006695	0.006318	0.005984	0.005686	0.005417
0.009682	0.008869	0.008191	0.007617	0.007124	0.006695	0.006318	0.005984	0.005686	0.005417
0.009659	0.008848	0.008173	0.0076	0.007108	0.00668	0.006304	0.005971	0.005673	0.005406
0.009659	0.008848	0.008172	0.0076	0.007108	0.00668	0.006304	0.00597	0.005673	0.005405
0.009658	0.008847	0.008171	0.007599	0.007107	0.006679	0.006303	0.00597	0.005672	0.005405
0.009547	0.008755	0.008093	0.007531	0.007048	0.006627	0.006257	0.005929	0.005636	0.005372
0.009247	0.008507	0.007886	0.007355	0.006897	0.006495	0.006141	0.005826	0.005544	0.00529
0.008869	0.008192	0.007618	0.007124	0.006695	0.006318	0.005984	0.005686	0.005418	0.005175
0.008869	0.008192	0.007618	0.007124	0.006695	0.006318	0.005984	0.005686	0.005418	0.005175



At learning cycle 0, where I randomly assign weights my network is very very bad, it's average error above 0.6.

At the second cycle network gets much better, average error drastically drops down! Seems like when network gets to cycle four error is at a point where function stays practically same, it gets really close to 0.