

PERCEPTRON:

Net Id: kxk152430

20 different combinations of Iterations and Learning Rate:

With out Stopwords:

S.NO	Iterations	Learning Rate	Perceptron
1	500	0.01	84.10042
2	500	0.05	82.63598
3	400	0.2	72.59414
4	400	0.9	72.38493
5	700	0.01	91.00419
6	700	0.5	90.58578
7	300	0.005	69.66527
8	300	0.015	65.271965
9	800	0.3	66.7364
10	650	0.6	88.07532
11	390	0.7	77.40586
12	290	0.6	60.46025
13	610	0.1	73.01256
14	630	0.4	87.656906
15	530	0.8	71.12971
16	820	0.9	81.799164
17	820	0.01	72.803345
18	320	0.01	69.66527
19	670	0.05	73.64017
20	720	0.04	91.42259

With Stopwords:

S.NO	Iterations	Learning Rate	Perceptron
1	500	0.01	81.02929
2	500	0.05	80.98326
3	400	0.2	71.0795
4	400	0.9	71.4217
5	700	0.01	76.98744
6	700	0.5	84.728035
7	300	0.005	65.732216
8	300	0.015	71.54812
9	800	0.3	59.20502
10	650	0.6	83.80753
11	390	0.09	71.757324
12	290	0.1	60.251045
13	610	0.001	60.46025
14	630	0.4	62.761505
15	530	0.8	63.933056
16	820	0.9	79.958157
17	820	0.01	62.97071
18	320	0.01	37.238495
19	670	0.05	65.271965
20	720	0.04	73.221756

Comparison with Naive Bayes, Logistic Regression and Weka Perceptron:

With out Stopwords:

S. N O	Iterations	Learning Rate	Perceptron	Weka Multilayer Perceptron	Logistic Regression	Naive Bayes
1	500	0.01	84.10042	92.6778	78.4195	85.35565
2	650	0.6	88.07532	79.2216	69.3443	
3	700	0.01	91.00419	90.1402	91.25994	
4	500	0.05	82.63598	94.9791	69.02917	
5	400	0.2	72.59414	94.9791	92.27011	

With Stopwords:

S. N O	Iterations	Learning Rate	Perceptron	Weka Multilayer Perceptron	Logistic Regression	Naive Bayes
1	500	0.01	81.02929	91.5667	86.7882	77.61507
2	650	0.6	83.80753	72.8033	65.0779	
3	700	0.01	76.98744	91.40362	87.12556	
4	500	0.05	80.98326	93.9791	67.029177	
5	400	0.2	71.0795	90.5891	88.5412	

WEKA NEURAL NETWORKS:

Hidden Layers	Units	Learning Rate	Iterations	Momentum	Accuracy
1	1	0.2	500	0.3	96.0251 %
1	1	0.2	200	0.3	95.8159 %
1	1	0.3	500	0.9	72.8033 %
1	3	0.2	800	0.3	95.1883 %
1	3	0.2	150	0.3	95.1883 %
1	3	0.2	150	0.6	82.636 %
1	2	0.9	500	0.2	73.6402 %
1	2	0.3	500	0.7	72.8033 %
2	1,2	0.2	500	0.3	95.1883 %
2	2,1	0.4	400	0.1	73.6402 %
2	3,1	0.6	600	0.4	72.8033 %
2	3,4	0.6	600	0.5	72.8033 %
2	2,3	0.1	550	0.1	95.3975 %
3	2,3,1	0.3	500	0.3	72.8033 %
3	1,2,3	0.4	550	0.2	82.9016 %

Report/Conclusion:

If you use a low learning rate with a high iteration number the learning process will be more conservative and more likely to high a good minimum. High momentum pushes the model from getting stuck at the local minima. So momentum can be moderate. For our dataset different hidden layers and units are required.