CIS 419/519: Homework 2

{Ransford Antwi}

Although the solutions are my own, I consulted with the following people while working on this homework: {Celine Lee}

4.1.1 Winnow hyperparameter sweep:

α	Sparse	Dense
1.1	0.92	0.90
1.01	0.9294	0.9338
1.005	0.9337	0.9316
1.0005	0.9146	0.9304
1.0001	0.5761	0.8346

Averaged Winnow hyperparameter sweep:

α	Sparse	Dense
1.1	0.9196	0.9027
1.01	0.93	0.934
1.005	0.9337	0.93
1.0005	0.915	0.9304
1.0001	0.58	0.8346

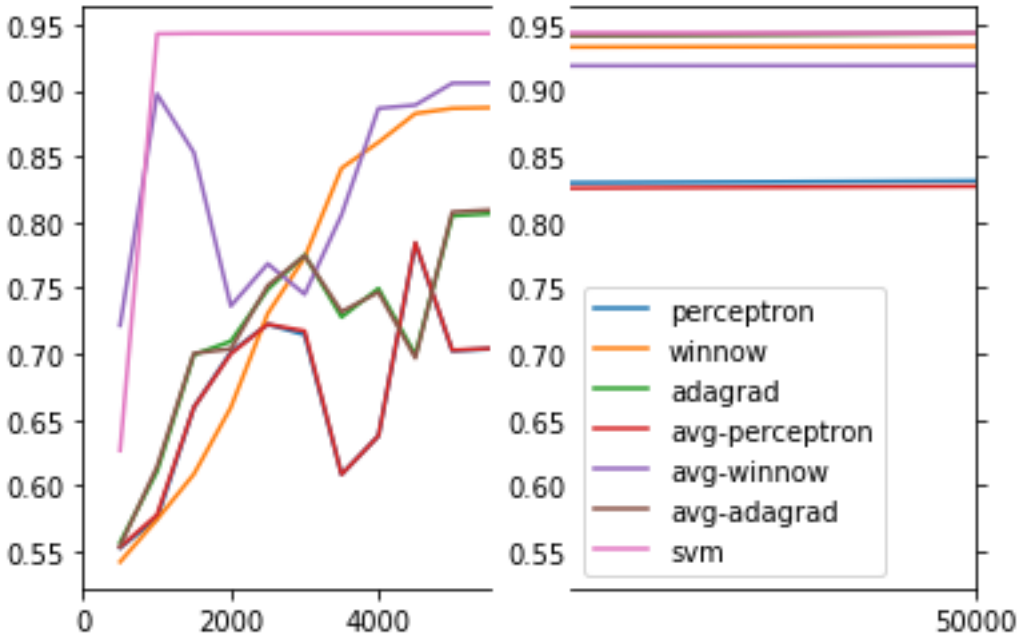
Perceptron with AdaGrad hyperparameter sweep:

η	Sparse	Dense
1.5	0.943	0.9372
0.25	0.9004	0.9372
0.03	0.6208	0.6217
0.005	0.5034	0.4917
0.001	0.4971	0.4971

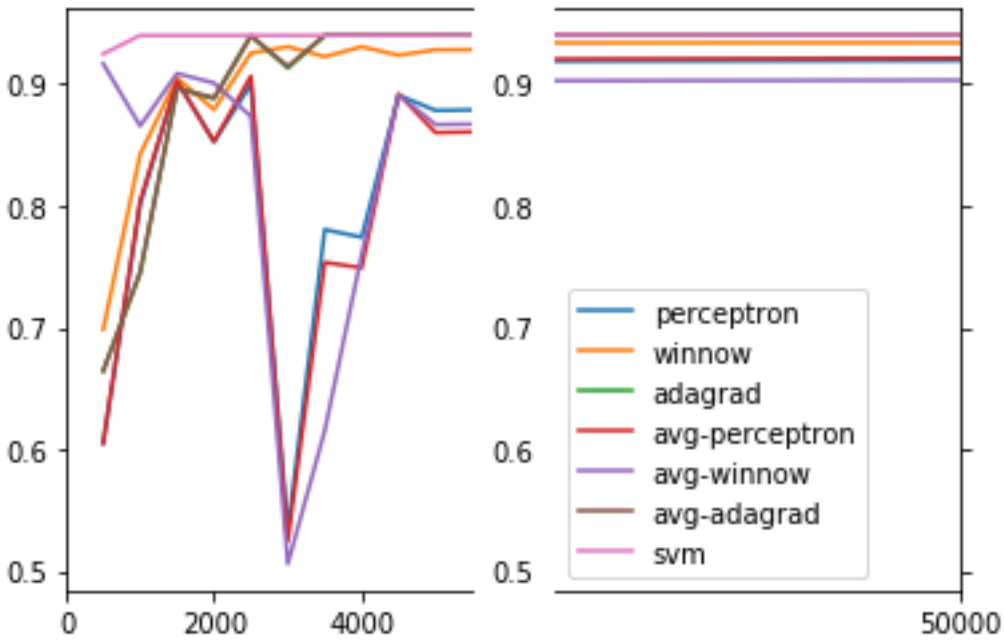
Averaged Perceptron with AdaGrad hyperparameter sweep:

η	Sparse	Dense
1.5	0.9433	0.9372
0.25	0.9171	0.9372
0.03	0.7332	0.6217
0.005	0.5875	0.4971
0.001	0.4988	0.4971

4.1.2 Sparse learning curves:



Dense learning curves:



(a) The general trend when comparing the standard version to the averaged version of an algorithm is that the accuracy is slightly higher with the averaged version. For example with the averaged perceptron, my implementation had an accuracy of 0.91 but the averaged version had an accuracy of 0.93. This increase in accuracy can be attributed to the fact that when dealing with the averaged versions, it ensures that we don't make the same mistake again i.e. we don't repeat a mistake on the same example.

(b) Not all 50,000 examples were necessary to achieve the best performance. Approximately 42000 examples were needed to achieve the best performance.

4.1.3 Final test accuracies

Model	Sparse	Dense
Perceptron	0.8277	0.9193
Winnow	0.9337	0.9317
Perceptron with AdaGrad	0.9433	0.9372
Averaged Perceptron	0.8248	0.9212
Averaged Winnow	0.9337	0.9316
Averaged Perceptron with AdaGrad	0.9433	0.9372
SVM	0.9433	0.9372

4.2 CoNLL and Enron F_1 scores:

Model	CoNLL Test F_1	Enron Test F_1
Averaged Perceptron	0.79	0.174
SVM	0.827	0.24

The F_1 scores on CoNLL and Enron are very dissimilar. The scores were much higher for CoNLL than for the Enron datatset. A major reason for the higher F_1 score for CoNLL is that both models were trained on the CoNLL data, including the one used to predict labels for Enron. The Enron datatset only had 368 sentences compared to the 14 987 training sentences for CoNLL.