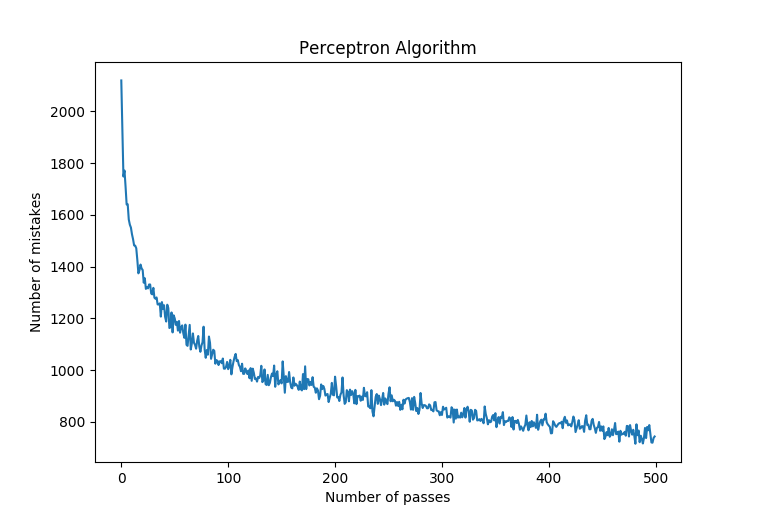
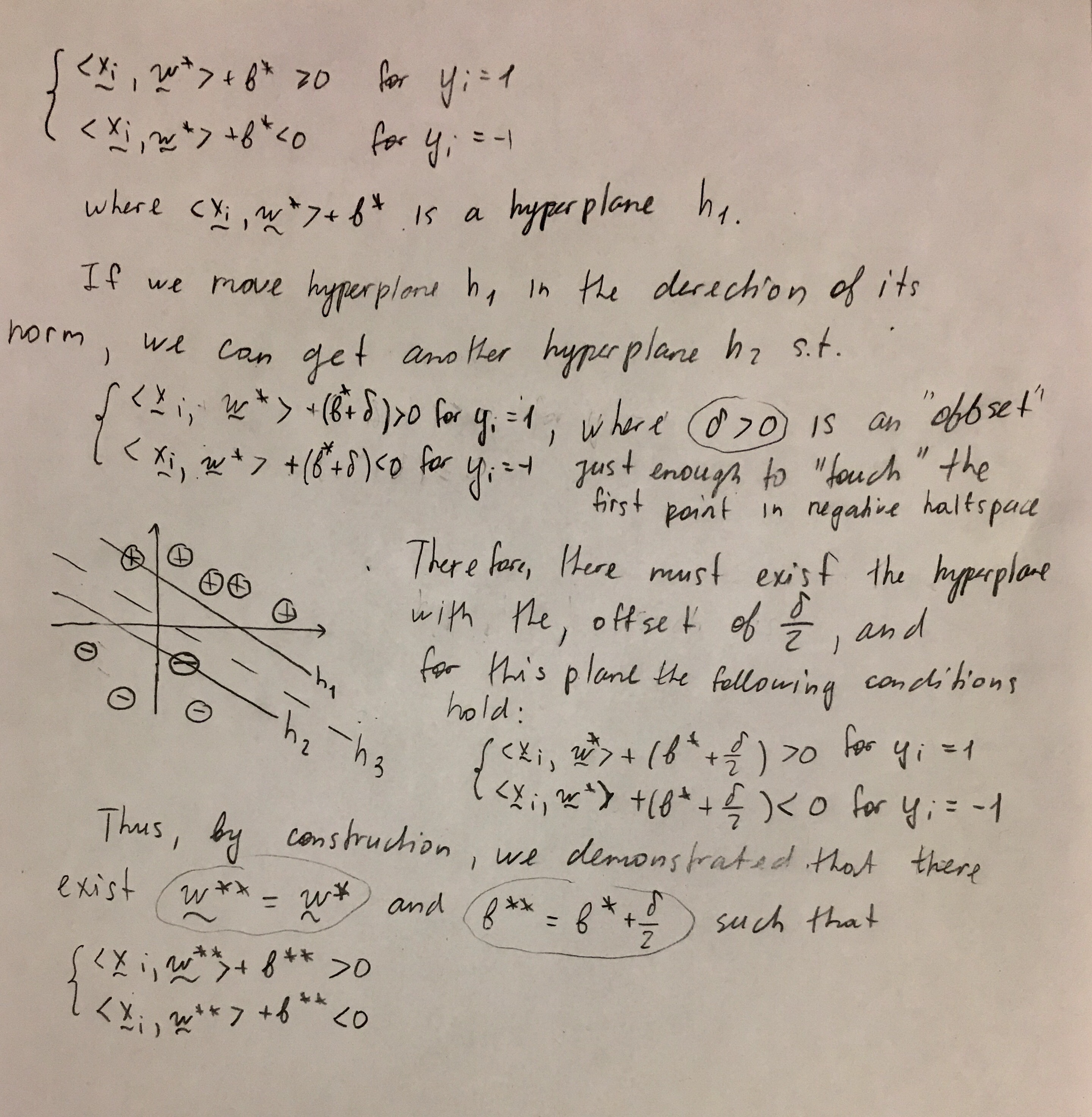
**Perceptron Leaning**



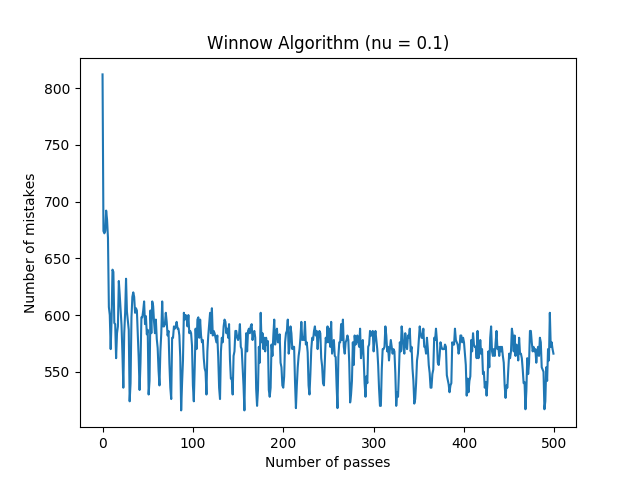


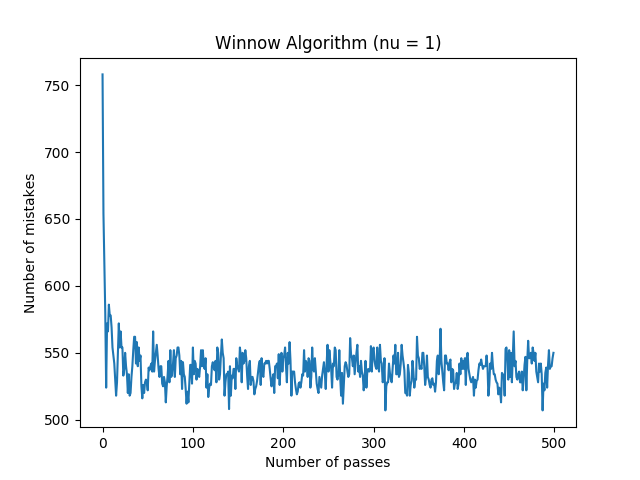
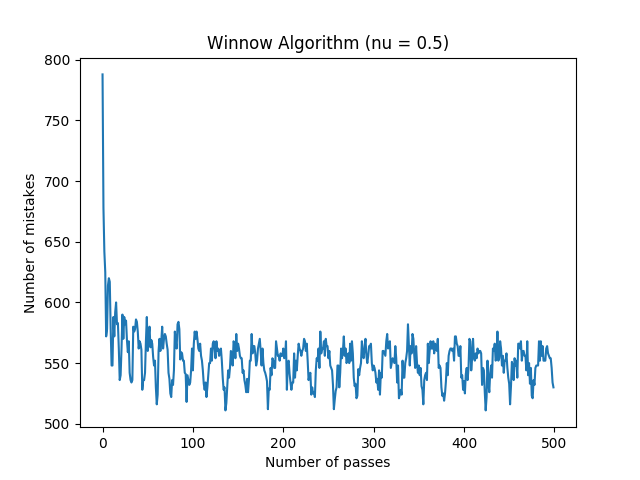
**Winnow Learning**

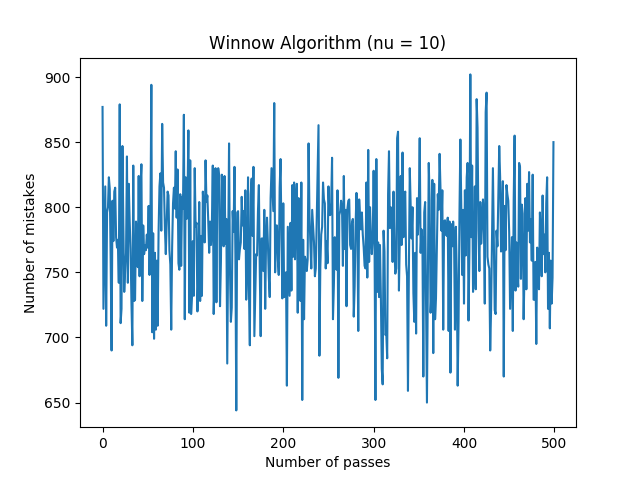
We can define two weight vectors as and . According to this definition both weight vectors are greater or equal to zero. It can be seen that Therefore, . Thus the non-negativity assumption is not violated, and can take positive and negative values.

**X normalized:**

Below are number of mistakes vs number of passes for winnow algorithm for different values of .





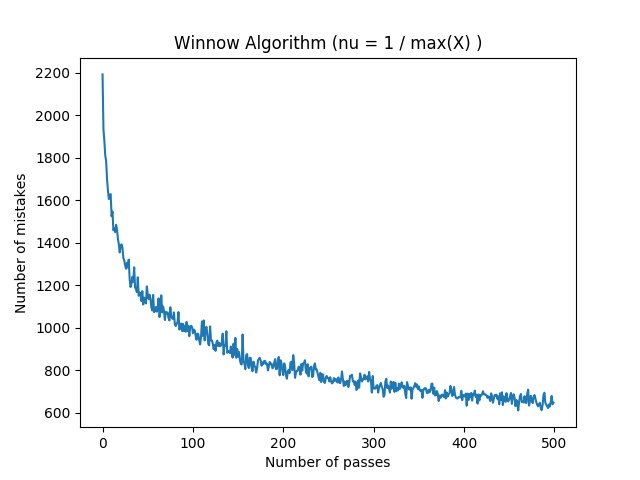


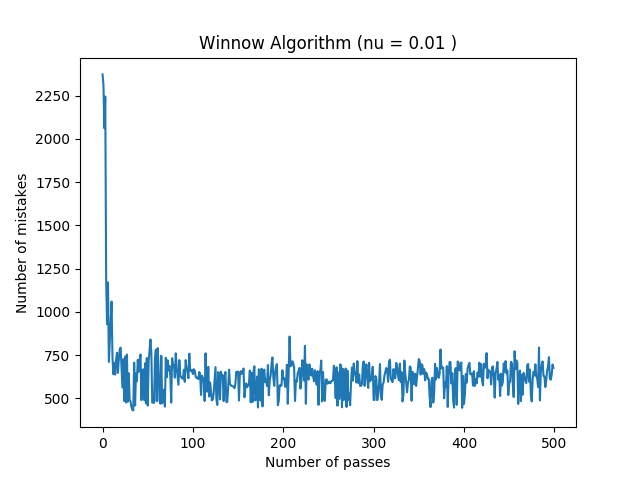
As we can see, the algorithm demonstrates the best performance when the step size = 1, which is exactly one over the maximum absolute value of X. When is too high, the algorithm doesn’t converge, and when is too low the fluctuations around are higher.

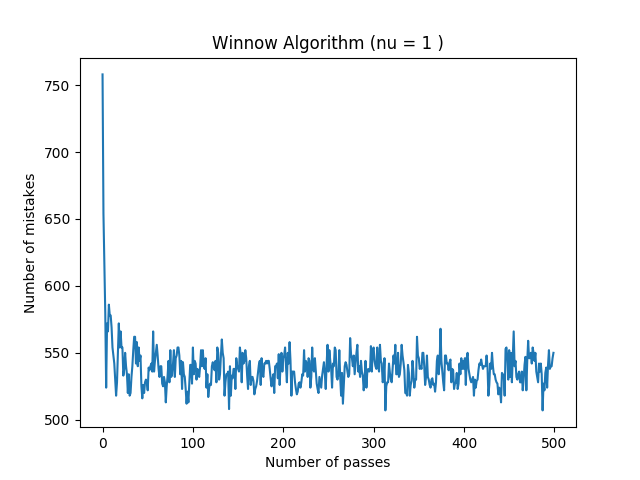
**X not normalized:**

Interestingly, much smoother results were obtained without normalization.

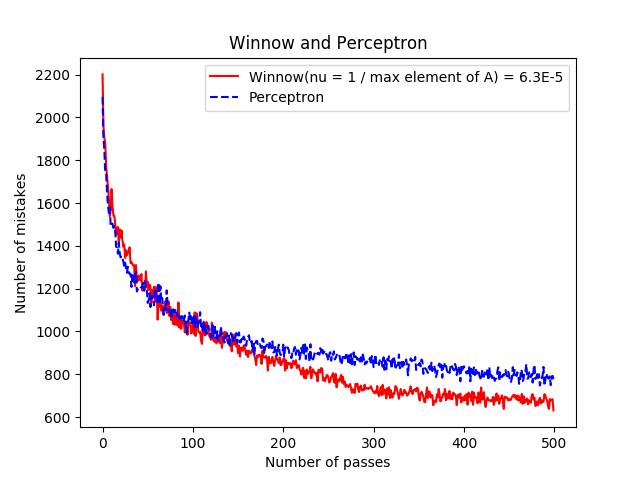
Thus, when equals one over the maximum element of X, which is 6.312e-05, algorithm has the following performance:







**Perceptron vs. Winnow**



Based on the experiments, it was observed that Winnow converges faster than perceptron. Perceptron doesn’t have a step size that needs to be optimized, whereas for Winnow the step size needs to be adjusted. However, the value of one over the value of the maximum element of X seems to work well. Balanced version of Winnow algorithm was able to classify spam, whereas the original version didn’t work. Overall, both algorithms are good for binary classification, and each of them has its advantages and disadvantages.