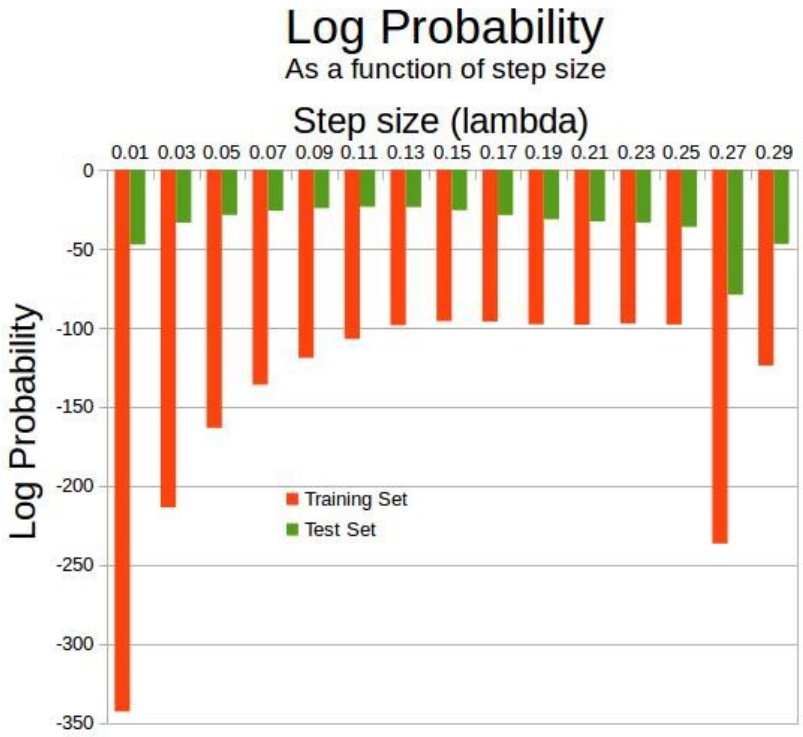
1. The learning rate (lambda) describes the rate at which the bias vector (beta) is changed at each iteration. A higher value for lambda means the bias vector is greatly altered at every update, and a lower value for lambda means the bias vector is not changed very much each update. A value for lambda greater than one will produce a horribly inaccurate model. Therefore, I experimented with lambda values in between zero and 0.3. Here are my results:



As shown above, the value for lambda which maximizes log probability lies between .11 and .17. I chose .15 as my learning rate for all other experiments.

2. The log likelihood of the model for the test set eventually stabilizes and hovers around about -20 (this corresponds to an accuracy of about 93 percent). It takes the algorithm four passes through the training set before the log likelihood stabilizes to this value without going up or down too much.

3. The best three predictor words for an article (starting with the best) are 'runs', 'hit', and 'pitching'. A high beta value for a particular word means that the algorithm will be more likely to classify the article containing that word as an article about baseball. Therefore, I found these three words by computing the largest three values in the beta vector after I had completed all of the passes through the training data. Using similar reasoning, I found the best three predictor words for an article about hockey (starting with the best) are 'hockey', 'playoffs', and 'points'. The only difference was that I used the smallest three values in the beta vector instead of the largest three to find these words.

4. The worst predictor words are 'everywhere', 'blasted', 'intermissions', 'bloody', 'broad', 'deceased', 'hesitate', 'hooked', 'memoriam', 'pitiful', 'racist', 'riel', 'rode', 'silence', 'tone', 'vintage', and 'wrestling'. A beta value which is close to zero for a particular word means that the word will not bias the algorithm very much towards either class. Thus, I found these words by computing the beta values that were closest to zero (in fact, each of these words had a beta value of exactly 0).