Wendy Carvalho
COMP.4220 Machine Learning

Logistic Regression - Assignment 2

Method

This Stochastic Gradient Descent (SGD) model takes in a negative word list, a positive word list (both with counts), and a vocabulary list with weights to predict whether a post is talking about baseball or hockey. These datasets are shuffled for randomness after being initialized. Then, for each training example in the dataset, the model uses the sigmoid function to the probability of that a word belongs to a specific class, which is, essentially, logistic regression. It then uses the value of this probability to calculate the gradient and update the beta vector, which is a vector of the weights associated with each word. Additionally, the model uses a learning rate to control how large each weight update is. It repeats the calculation of each training example until it hits convergence or the iteration limit is reached.

Implementation and Insights

One of the main decisions made during implementation was using the sigmoid function for when y is equal to 0, as this prevented the results of each update from growing larger in its negative values (negative weight divergence) instead of becoming smaller. This ensured that the updates moved the weights closer to the correct values.

It was noted that the learning rate controls the size of the steps the model takes, which determines how quickly the algorithm moves towards a minimum of the loss function. If the learning rate is too high, the model's convergence could be unstable–potentially getting stuck at a local minima, for example. If it is too low, the model will converge very slowly.

This model performs 1064 passes over the data before it stabilizes. Essentially, this is how many iterations it takes for the log probabilities to stabilize at lower values and the accuracies to stabilize at higher values.

Classifying a post as being about baseball or hockey comes down to the features, also known as words, that show up in a post. There are words given in the vocabulary list with an associated weight, which defines how important it is in defining a post to be about baseball or hockey. For example, words, or features, such as "bat" or "pitch" are strong indicators for a post to be about baseball, whereas "goalie" or "puck" are strong indicators for hockey.

Lastly, finding the best and worst predictors involved sorting the beta vector, grabbing their indices, and searching the vocabulary list for them. The large positive coefficients are strong indicators for a post to be about baseball, whereas the large negative coefficients are strong indicators for a post to be about hockey. The coefficients with values close to zero are the worst predictors, in general, as they might be about either sport.

The best ten predictors for baseball are "runs", "baseball", "hit", "pitching", "catcher", "ball", "book", "rickert", "stance", and "bat". The best ten predictors for hockey are "hockey", "playoffs", "golchowy", "goals", "pick", "next", "ice", "biggest", "names", and "playoff". Ten of the worst predictors are "hooked", "hesitate", "riel", "intermissions", "rode", "blasted", "vintage", "hurled", "tone" and "broad.

Limitations

Some limitations of SGD are how noisy it is, which can cause the learning rate to jump around instead of slowly decreasing. Additionally, it may not always be optimal to have such a noisy convergence path, because it may not converge to the exact global minimum but in a suboptimal solution. However, this can be mitigated by a schedule. Also, since it takes in only one training example at a time, it can lose the benefits of vectorization. But, in general, it is much less expensive than a normal gradient descent and it can find the global optima in cases where the normal gradient descent cannot.