**Homework #2: Sentiment**

# Problem 1

## Problem 1a

We have a weight vector of dimension 6 for the 6 different unique words appearing in the review. For the first pass we have

From the above, we see that with our initialized weights being zero, the margin for the first pass is 1, but since the gradient , we see that the weights are unable to update via gradient descent, which in turn reduces the weight vector to all zeros, the same as the value to which they were initialized:

## Problem 1b

Suppose that we have the following reviews as our dataset:

* (-1) not good
* (-1) bad
* (+1) good
* (+1) not bad

The fundamental property of the linear classifier is that its learning score is a linear combination of weights and features:

This means that in order for the classifier to have zero error, the features must be linearly separable, i.e. for two categories of points there must be some hyperplane with component coefficients such that every point in satisfies and each point in satisfies (assuming the hyperplane passes through the origin).

However, since the feature-vector is a mapping of word frequencies, **linear separation becomes geometrically impossible, because there is no linear / constant separator which can divide the word frequencies on their own.** Plotting the sample feature vectors from our dataset, which exist in , illustrates this point:

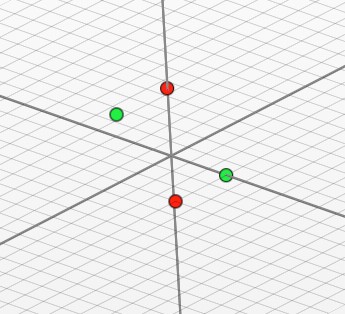


Figure 1: Plot of the sample feature points in . Red points denote bad reviews (y=-1) while green denote good reviews (y=1). Note that no 3D plane can be drawn which perfectly separates bad from good reviews, indicating linear inseparability.

In order to fix this problem we have to augment the feature vector somehow. Since part of the problem is that the existing features are used in both classification outcomes, one such augmentation that would fix it would be **to include bigrams of the words as features (e.g. counting the number of instances of “not good” in succession as opposed to just the words individually).** This will augment the feature vector into a higher dimension in which linear separability is possible.

# Problem 2

## Problem 2a

Since we are using the squared loss, its general expression is given by

In this case, we use a nonlinear predictor in the form of a sigmoid which is of the form

By substitution we then have

## Problem 2b

Computing the gradient of the loss, we apply the chain rule, letting such that:

Applying the derivative identity for the sigmoid function, , we have

## Problem 2c

Substituting y=1 into the above for the SGD run on the data point provided in the problem statement, we have

Finding the points which make the gradient small involve finding the roots of the expression on the left-hand side, which has a non-trivial root at p=1. Therefore, it is **theoretically** possible to obtain a zero-gradient result given the above analytical solution. However, recall that

Therefore the underlying weights which obtain this zero gradient are given by

Which simplifies to

Therefore, for some given value of , it follows that

Based on the above, **it is not actually possible to get the magnitude of the gradient to zero, since it only approaches zero as the magnitude of the weights approaches an infinitely large magnitude.** This is easily verified by visual inspection of the sigmoid curve, i.e. the curve “flattens” for very large values of the independent variables, which mathematically translates to a smaller gradient magnitude.

## Problem 2d

The largest magnitude the gr