Linear Models and SGD on Brain Image Data

Yiran Cao

$$L_{log}(w) = \sum_l \ln(1 + \exp(-y_l w \cdot x_l)) + \lambda \sum_i w_i^2$$

$$Loss(X_l) = \ln(1 + \exp(-y_l w \cdot X_l)) + \frac{\lambda}{N} \sum_i w_i^2$$

$$\frac{\partial Loss(X_l)}{\partial w_i} = \frac{-y_l X_l^i \exp(-y_l w \cdot X_l)}{1 + \exp(-y_l w \cdot X_l)} + \frac{2\lambda}{N} w_i$$

Parameter Tuning

For this part I tried combinations from $\eta \in \{0.1, 0.01, 0.001, 0.001\}$ and $\lambda \in \{0.1, 0.1, 1\}$ on the dev set. Accuracy of each combination of parameters are shown in Table 1.

(a) Logistic Loss

λ	η	Accuracy
0.3	0.0001	0.6
0.3	0.001	0.85
0.3	0.01	0.9
0.3	0.1	0.9
0.1	0.0001	0.6
0.1	0.001	0.65
0.1	0.01	0.9
0.1	0.1	0.9
1	0.0001	0.6
1	0.001	0.9
1	0.01	0.9
1	0.1	0.9

(b) Hinge Loss

λ	η	Accuracy
0.3	0.0001	0.6
0.3	0.001	0.7
0.3	0.01	0.75
0.3	0.1	0.8
0.1	0.0001	0.6
0.1	0.001	0.75
0.1	0.01	0.75
0.1	0.1	0.75
1	0.0001	0.6
1	0.001	0.75
1	0.01	0.75
1	0.1	0.8

Table 1: Results on Dev Set

Results on Test Data

For each loss function, I pick several combinations of parameters that give the highest accuracy on the dev set and test them on test dataset. The results are shown in the Table 2.

Inspecting the Models Parameters

The weight vector we learned is a 258391-dimension vector. With a little rearrangement, we will see that $258391 = 55 * 4698 + 1$. My interpretation of this equation would be that, each trial has 258391 attributes, including information of 55 brain images overtime, each images having 4698 voxels, adding 1 bias term.

(a) Logistic Loss

λ	η	Accuracy
0.3	0.01	0.8235
0.3	0.1	0.8235
0.1	0.01	0.8235
0.1	0.1	0.8235
1	0.01	0.8235
1	0.1	0.8529

(b) Hinge Loss

λ	η	Accuracy
0.3	0.1	0.8235
1	0.1	0.8235

Table 2: Results on Test Set

With this interpretation, $W[0, 1, \cdots, 4697]$ are weights of voxels in the 1st image, and $W[4698, 4699, \cdots, 9395]$ are weights of voxels in the 2nd image, and so on.

The basic idea of my approach is making the 55 brain images vote. That is, for each image, calculating the weight of each ROI and pick one that is of the most importance. So we would have 55 "most important ROI" votes, and among which the majority wins.

Since the absolute value of weights can represent the attribute's importance, for each ROI , I take the average of the absolute value of weights of voxels in that ROI and take the result as the weight of that ROI.

With this approach (linear_brain.py, line 202-211), the most important ROI for predicting whether the subject is viewing a picture or a sentence is RIPL