Laboratory exercise 10: Logistic regression

Santiago Enrique Cortés
Bárbara & Frick
Av. Calle 82 # 10 18A - 12, Bogotá, Colombia
santiago.cortes@barbara.net.co

Juan Felipe Cerón Uribe
Universidad de los Andes
Cra 1 # 18A - 12, Bogotá, Colombia
jf.ceron10@uniandes.edu.co

1 Introduction

In this paper is exposed the results from a series of experiments involving logistic and soft-max regression. This laboratory was intended for familiarize the authors with basic parameters required for training in a categorical problem. The idea was to experiment with parameters that determines the framework of training: batches, epochs and learning rate.

2 Dataset

The data set chosen for the laboratory was the face emotion recognition challenge or *fer2013*. This data set comes as a data frame with two columns, one a 48×48 matrix with numbers from 0 to 255 and the other a number between 0 and 7. The first of course corresponds to a 48 by 48 gray scale image with a human face exposing an emotion. As it is expected, the second column represents the emotion that is shown by the face in it's corresponding image. The code for the emotions is:angry 0, Disgust 1, Fear 2, happy 3, sad 4, surprise 5, neutral 6. There were a total of 28709 examples of which 3589 are used as the test set. A subset of equal size as the test set is drawn from the training one for validation porpoises.

3 Algorithms

3.1 Logistic regression

The logistic regression arises in the context of supervised binary classification. Let $\{x_1, ..., x_n\}$ a set of n observations in R^m and $(y_1, ..., y_n) \in \{0, 1\}^m$ their corresponding labels. For a fixed vector $\hat{\beta} = (\beta_0, ...\beta_n) \in \mathrm{R}^{m+1}$ the quantity $P(Y = 1|X = x_i)$ is estimated by:

$$\frac{1}{1 + exp(-\hat{\beta} \cdot (1, x_i))} \tag{1}$$

The vector $\hat{\beta}$ is fitted using log-likelihood, in other words, by minimizing the following loss function:

$$-\frac{1}{n} \sum_i (y_i log(\hat{y}_i) + (y_i - 1) log(1 - \hat{y}_i)) \tag{2}$$

3.2 Soft-Max regression

The soft-max function was proposed as a possible solution for the multi-class supervised classification problem. It intends to generalize the approach given by the logistic regression to the binary clasification. Let $\{x_1, ..., x_n\}$ a set of n observations in R^m and $(y_1, ..., y_n) \in \{0, 1\}^m$ their corresponding labels. For a fixed set of vectors $\{\hat{\beta}_0, ..., \hat{\beta}_k\} \subset \mathrm{R}^{m+1}$ the quantity $P(Y = i|X = x_i)$ is estimated by:

$$\frac{exp(\hat{\beta}_i \cdot (1, x_i)}{\sum_{0 \leq i \leq k} exp(-\hat{\beta}_i \cdot (1, x_i))} \tag{3}$$

Just as in the binary case, the parameters are fitted by minimizing the log-likelihood function for the probability given by the soft-max function:

$$-\frac{1}{n} \sum_i (y_i log(\hat{y}_i)) \tag{4}$$

4 Training

For minimizing both loss functions was used algorithm of gradient decent. Let $L(\hat{\theta}, x_1, ..., x_n)$ a smooth loss function of the parameters $\hat{\theta}$. Then the gradient decent algorithm is given by:

$$a_0 = \text{random point} \tag{5}$$

$$a_{n+1} = a_n - \gamma \nabla \mathcal{L}(a_n) \tag{6}$$

Under certain conditions on γ and the function $\mathcal{L}$ convergence to a local minimum can be guaranteed. It is easy to see that the value γ is an important parameter that must be tuned in order that the model achieves a good performance.

Logistic regression Errors

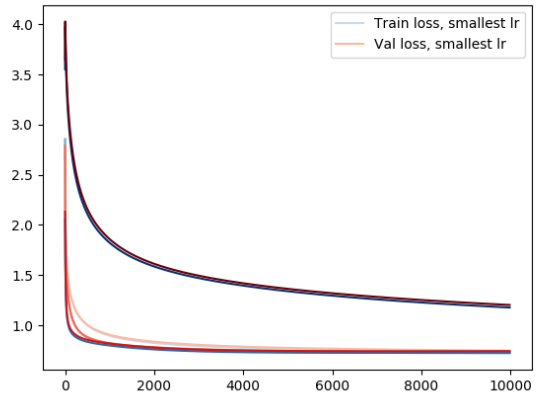

Figure 1. A graphic comparing training an validation error for several learning rates, the intensity shows higher learning rates. Blue colors are for training error and red ones for validation ones.

Soft-max Errors

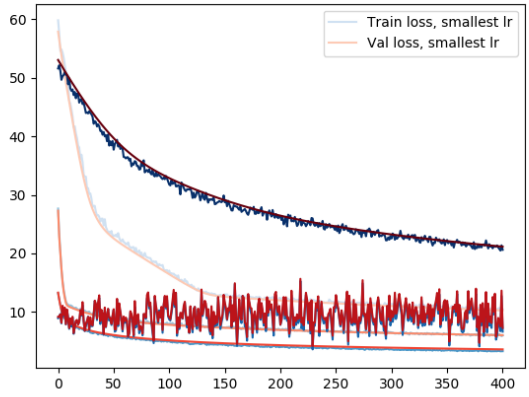

Figure 2. A graphic comparing training an validation error for several learning rates, the intensity shows higher learning rates. Blue colors are for training error and red ones for validation ones.

5 Batches and Epochs

Given that every iteration in the gradient decent depends in the computation of $L(\hat{\theta}, x_1, ..., x_n)$, if n is too large the training can be slow. To overcome this problem every iteration is done over batches,i .e, subsets of equal size of the training set.

With the objective of ensure the convergence to a local minima, several passes trough the training set were done, each pass is called an *epoch*.

For all the experiments batches of 100 images were used, and 10000 epochs. The learning rates used were in the case of the soft-max experiment: 0.1,0.01,0.001,0.0001 and, in the logistic regression: 0.0001, 0.0005,0.002 and 0.001.

6 Results

There are several considerations that must be taking into account after the experiments. The first one, was that the gradient decent for the soft-max increase its computation time with each epoch. Hence, the curves plotted are only for 400 hundred passes versus the one of the logistic regression that did run over all the 10000. The second consideration, that do not appear in the graphics, is that for the logistic regression all learning rates over 0.001 did not converge, further more, made the algorithm show infinite values. In contrast the soft-max was capable of execute all the learning rates that were tried, being 0.1 the biggest of them.

Neither the logistic regression nor the soft-max gave the expected graphic where the validation error starts to grow after a point. That means that more epochcs could be added in order to improve performance.

Precision Recall curve

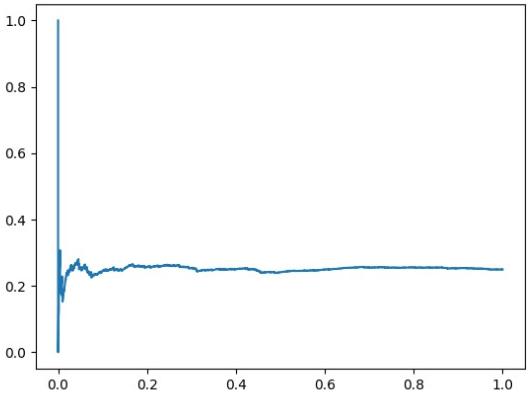

Figure 3. precision recall curve for the logistic regression with 0.0001 as learning rate.

For both the logistic and the soft-max regression, the best performance was achieved with the smallest learning rate of 0.0001.

The logistic regression had the following performance metrics:

1. Best ACA: 0.7506269155753692
2. Max F1-measure: 0.39973142345568485

The soft-Max regression had the following performance metrics:

Figure 4. demonstration of some predictions.

1. Best ACA: 0.126

7 Discussion

The slow learning rate for both the logistic regression and the Soft-max can be explained by the gradient saturation. That is, because of the nature of exponential functions, both gradients can become very small in norm so each iteration did not differ by much from the previous one, thus slowing the training process. It is clear that with all the learning rates the training was slow. The former can be deduced from the fact that no validation error started to differ from the training error during the tests.

The failure of the experiments made with learning rates larger than 0.001 for the logistic regression optimization can be explained by the normalization done in the images. Every vector where between 0 and 1, hence punishing severely high learning rates.

Even tough both models had poor performance the logistic out-perform the soft-max model. It is natural to think that this was caused by the fact that one was modeling a binary classification problem while the other a multi-classification one.

The perfect example of a high learning rate is the highest one used in the soft-max experiments. It is clear how is just going forward and backwards avoiding always the minimum.

The choice of 100 images per badge seems to be a good option. Never the less, it would be nice in a next experiment to use more images per badge and seeing in this way if the training time decrease.

Finally it is important to remark that in the logistic regression all training curves were smooth and in the soft-max none were. It is very interesting how in spite of being very sensible to lower learning rates, the logistic regression produced monotonic training error sequences. The former can be explained because of how the logistic function is, is ”more” convex than the soft max one. The parameters of the logistic are in a lower dimensional space than the one in the soft max, thus are less directions in which the gradient can increase and thus making the graph smoother.All in all, is a good example of how it is always easier to optimize in lower dimensional spaces rather in large dimensional ones.

References