Artificial Intelligence II

Nefeli Tavoulari

Fall Semester 2021

1

The Mean Squared Error loss function is defined as:

$\mathcal{MSE} = \frac{1}{m} \sum_{i=1}^{m} (h_w(x_i) - y_i)^2$

Calculating the gradient of MSE, we can find out the parameters which minimize the loss.

$\nabla_{\mathbf{w}} \mathcal{MSE} = \nabla_{\mathbf{w}} \frac{1}{m} \sum_{i=1}^{m} (h_w(x_i) - y_i)^2$

The features x_i and the predicted values y_i can be considered as constants, since the gradient is with respect to w, which is the model's parameter vector.
(scalar multiplication rule)

$\nabla_{\mathbf{w}} \mathcal{MSE} = \frac{1}{m} \nabla_{\mathbf{w}} \sum_{i=1}^{m} (h_w(x_i) - y_i)^2$

Then, assuming we only have one instance (x,y) and using the power rule and the chain rule for derivatives:

$\nabla_{\mathbf{w}} \mathcal{MSE} = \nabla_{\mathbf{w}} (h_w(x) - y)^2$

$\nabla_{\mathbf{w}}\mathcal{MSE} = 2(h_w(x) - y)\nabla_{\mathbf{w}}(h_w(x) - y)$

where $h_w(x) = w_0x_0 + ... + w_nx_n = w^Tx$: **(*)**

$\nabla_{\mathbf{w}}\mathcal{MSE} = 2(h_w(x) - y)\nabla(w_0x_0 + ... + w_nx_n - y)$

$\nabla_{\mathbf{w}}\mathcal{MSE} = 2(h_w(x)-y)(\frac{\partial(w_0x_0+...+w_nx_n-y)}{\partial w_0}, ..., \frac{\partial(w_0x_0+...+w_nx_n-y)}{\partial w_n})$

$\nabla_{\mathbf{w}}\mathcal{MSE} = 2(h_w(x) - y)(x_0, ..., x_n)$

$\nabla_{\mathbf{w}}\mathcal{MSE} = 2(h_w(x) - y)(x)$

so, from **(*)**:

$\nabla_{\mathbf{w}}\mathcal{MSE} = 2(w^Tx - y)(x)$

which equals to the following vector:

$$\begin{bmatrix} 2(w^Tx - y)(x_0) \\ ... \\ 2(w^Tx - y)(x_n) \end{bmatrix}$$

Now, for m training instances:

$\nabla_{\mathbf{w}}\mathcal{MSE} = \frac{2}{m}\sum_{i=1}^{m}(h_w(x_i) - y_i)\nabla_{\mathbf{w}}(h_w(x_i) - y_i)$

$$\nabla_{\mathbf{w}}\mathcal{MSE} = \begin{bmatrix} \frac{2}{m}(w^Tx_1 - y_1)(x_{1,0}) + ... + \frac{2}{m}(w^Tx_m - y_m)(x_{m,0}) \\ ... \\ \frac{2}{m}(w^Tx_1 - y_1)(x_{1,n}) + ... + \frac{2}{m}(w^Tx_m - y_m)(x_{m,n}) \end{bmatrix}$$

So: $\nabla_{\mathbf{w}}\mathcal{MSE} = \frac{2}{m}(\mathrm{X}^T(\mathrm{Xw - y}))$.

2

In this notebook I used a dataset provided by Dr. Saptarshi Ghosh and Soham Poddar from the Department of Computer Science and Engineering, IIT Kharagpur, India and trained a multinomial Logistic Regression classifier, which classifies a tweet as Neutral, Pro-vax or Anti-vax.

First of all, I performed some pre-processing and feature engineering on the data, I removed all empty or duplicate rows, I lowercased everything, performed stemming, stopwords removal, special characters removal, removal of languages other than english, url removal etc.
Then, I vectorized and calculated TF-IDF on the textual data, so as to perform Logistic Regression. I did the same process for the validation data too, of course.

Finally, using different metrics, precision, recall, f1, accuracy, confusion matrix, I were able to see the quality of my model.