Deep Learning Assignment 3

Anonymous Author(s)
Affiliation
Address
`email`

1 General Questions

(a)

(b)

2 Softmax regression gradient calculation

(a)

(b)

3 Chain rule

(a) Let

$$f = \frac{a}{b} \tag{1}$$

$$a = x^2 + \sigma(y) \tag{2}$$

$$b = 3x + y - \sigma(x) \tag{3}$$

$$\frac{\partial f}{\partial x} = \frac{\frac{\partial a}{\partial x} \cdot b - \frac{\partial b}{\partial x} \cdot a}{b^2} = \frac{2x \cdot b - (3 - \frac{\partial \sigma(x)}{\partial x}) \cdot a}{b^2} \tag{4}$$

$$\frac{\partial f}{\partial y} = \frac{\frac{\partial a}{\partial y} \cdot b - \frac{\partial b}{\partial y} \cdot a}{b^2} = \frac{\frac{\partial \sigma(y)}{\partial y} \cdot b - a}{b^2} \tag{5}$$

Where

$$\frac{\partial \sigma(x)}{\partial x} = \sigma(x) \cdot (1 - \sigma(x)) \tag{6}$$

(b) For $x = 1, y = 0$,

$$\sigma(x) = \sigma(1) = 0.269 \tag{7}$$

$$\sigma(y) = \sigma(0) = 0.5 \tag{8}$$

Submitted to 29th Conference on Neural Information Processing Systems (NIPS 2016). Do not distribute.

$$a = 1 + \sigma(0) = 1.5 \tag{9}$$

$$b = 3 * 1 + 0 - \sigma(1) = 2.731 \tag{10}$$

So we can calculate derivitive of f by:

$$\frac{\partial f}{\partial x} = \frac{2 * 1 * 2.731 - (3 - 0.269 * (1 - 0.269)) * 1.5}{2.731^2} = 1.633 \tag{11}$$

$$\frac{\partial f}{\partial y} = \frac{0.5 * (1 - 0.5) * 2.731 - 1.5}{2.731^2} = -0.1096 \tag{12}$$

4 Variants of pooling

(a) SpatialMaxPooling SpatialAveragePooling SpatialAdaptivePooling

(b)

(c)

5 Convolution

(a) Assume we use zero padding and step size = 1, then: $(5-3+1)*(5-3+1) = 9$ values will be generated.

(b) Let X be a 3x3 matrix on Image Matrix and W is the 3x3 convolution filter.
According to the definition of convolution, each element in the output is the point product of these two matrix.

$$F = \sum W \cdot *X \tag{13}$$

For example: $F_1 1 = 4*4+3*5+3*2+5*3+5*3+5*2+2*4+4*3+3*4 = 109$

So we have the output F = $\begin{pmatrix} 109 & 92 & 72 \\ 108 & 85 & 74 \\ 110 & 74 & 79 \end{pmatrix}$

(c) Let $\frac{\partial E}{\partial X^(i-1)} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$

The definition of F = $\begin{pmatrix} \sum_{i=1}^{3}\sum_{j=1}^{3} X_ijW_ij & \sum_{i=2}^{4}\sum_{j=1}^{3} X_ijW_ij & \sum_{i=3}^{5}\sum_{j=1}^{3} X_ijW_ij \\ \sum_{i=1}^{3}\sum_{j=2}^{4} X_ijW_ij & \sum_{i=2}^{4}\sum_{j=2}^{4} X_ijW_ij & \sum_{i=3}^{5}\sum_{j=2}^{4} X_ijW_ij \\ \sum_{i=1}^{3}\sum_{j=3}^{5} X_ijW_ij & \sum_{i=2}^{4}\sum_{j=3}^{5} X_ijW_ij & \sum_{i=3}^{5}\sum_{j=3}^{5} X_ijW_ij \end{pmatrix}$

$$\frac{\partial E}{\partial X_ij} = \frac{\partial}{\partial X_ij} ... = \sum W_ij \frac{\partial E}{\partial X^(i-1)} \tag{14}$$

result = $\begin{pmatrix} 4 & 7 & 10 & 6 & 3 \\ 9 & 17 & 25 & 16 & 8 \\ 11 & 23 & 34 & 23 & 11 \\ 7 & 16 & 24 & 17 & 8 \\ 2 & 6 & 9 & 7 & 3 \end{pmatrix}$

6 Optimization

(a)

(b)

(c)

(d)

7 Top-k error

Top-k error: the fraction of test images for which the correct label is not among the k labels considered most probable by the model.

$$E = \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}_{R_i > k} \tag{15}$$

Where function R counts the number of probability for prediction $\hat{y}_c$ that's larger than the probability of true label $\hat{y}_L$ for each test image.

$$R = \sum\nolimits_{c \in C} \mathbb{1}_{\hat{y}_c > \hat{y}_L} \tag{16}$$

The top-1 errors represents the error rate that the prediction is not the same with true label. This is useful to compare the performance of different models. However, it cannot show how how good the model is in general. Sometimes the categories in a image may be ambiguous and it may be described as multiple labels. The order of several possible labels is ambiguous. Using top-5 errors provides a more general understanding of learning ability of the model.

8 t-SNE

(a)

(b)

9 Proximal gradient decent

(a)

(b)

(c)

(d)