Mathematical exercise: Probability & Linear Regression

May 25, 2020

1 Differentiate the Sigmoid function

Show that the derivation of the sigmoid function (S)

$$S(x) = \frac{1}{1+e^{-x}}$$

results in

$$= S(x)(1 - S(x))$$

(Hint: You need to apply the quotient rule). Can you think of why the properties of the sigmoid's derivative is specifically convenient for neural networks? (You will receive points for the derivation, but not for this thinking exercise)

2 Differentiate the softmax function and insert the result into the derivative of the cross-entropy loss w.r.t one input

2.1 Derivative of Softmax

The softmax function takes a C-dimensional vector z as input and outputs a C-dimensional vector y:

$$y_c = \frac{e^{z_c}}{\sum_{d=1}^{C} e^{z_d}} \quad \text{for } c = 1 \cdots C$$

Show that the derivative of this results in:

$$\frac{\partial y_i}{\partial z_j} = \begin{cases} y_i(1 - y_j) & if \ i = j \\ -y_j . y_i & if \ i \neq j \end{cases}$$

$$\frac{\partial y_i}{\partial z_j} = \begin{cases} y_i(1 - y_j) & if \ i = j \\ -y_j . y_i & if \ i \neq j \end{cases}$$

2.2 Cross-Entropy Loss

The cross-entropy loss is expressed as:

$$L = -\sum_i t_i log(y_i)$$

Here t is the target vector

Show that it's partial derivative w.r.t. to the input z_i results in $y_i - t_i$ (Hint: You need to apply the chainrule)