LINEAR REGRESSION - A BRIEF REMINDER

Linear regression is the simplest regression algorithm where the output of the algorithm (y) is linearly dependent on the inputs (x).

y = ax+b

For a model with one variable (one feature), there is an exact formula to calculate the values of the model parameters.

In this task we will use the iterative Batch Gradient Descent method to find the optimal values of the model parameters.

1. Predict the y variable with the current model for each object x_i : $\hat{y}_i = ax_i + b$
2. Calculate the error *E* (loss) of our model using a function **mean squared error (MSE)** as the mean of the squared differences between the real value y_i and predicted value $\hat{y}_i$.
3. Calculate the gradient for each model parameter (*a* and *b*) using backpropagation algorithm
   - $\frac{\delta E}{\delta a}$ - derivative of error *E* with respect to parameter *a*
   - $\frac{\delta E}{\delta b}$ - derivative of error *E* with respect to parameter *b*
4. Update the parameter values with **delta rule**: $\Delta w \ = \ -lr * \frac{\delta E}{\delta w}$
   - *w* - parameter: *a* or *b*
   - Δw - value by which we should change the parameter *w*
   - *lr* - learning rate
   - $\frac{\delta E}{\delta w}$ - derivative of error *E* with respect to parameter *w* (*a* or *b*)
5. Repeat 1-4

1) **Linear layer:** Given is the following data set. Calculate first step of optimization. Assume that model is **y = ax+b**, **learning rate is equal to 0.2**, and ***a* is 1,** and ***b* is 1**. As a loss function use mean squared error.

$$MSE(Y, \hat{Y}) \ = \ \frac{1}{n}\sum_{i}^{n}(y_i - \hat{y}_i)^2 \ = \ E \qquad \hat{y}_i = ax_i + b \qquad \Delta w \ =- lr * \frac{\delta E}{\delta w}$$

	Input	True output	Predicted output	1 iteration			2 iteration	
ID	x_i	y_i	$\hat{y}_i$	MSE	$\frac{\delta E}{\delta b}$	$\frac{\delta E}{\delta a}$	$\hat{y}_i$	MSE
0	-1.0	-3.0	0.0	9.0	0.4(0+3)=1.2	-1*1.2=-1.2	-1.0	4
1	-0.5	-2.0	0.5	6.25	0.4(0.5+2)=1.0	-0.5*1.0=-0.5	-0.4	2,56
2	0.0	-1.0	1.0	4.0	0.4(1+1)=0.8	0*0.8=0	0.2	1,44
3	0.5	0.0	1.5	2.25	0.4(1.5)=0.6	0.5*0.6=0.3	0.8	0.64
4	1.0	1.0	2.0	.1.0	0.4(2-1)=0.4	1*0.4=0.4	1.4	0.16
	Sum			22.5	4	-1		8,8
	Loss			4.5				1.76

Gradient equations:

$$\frac{\delta E}{\delta \hat{y}_i} = -\frac{1}{n}\sum_i^n 2(y_i - \hat{y}_i) = \frac{2}{n}\sum_i^n (\hat{y}_i - y_i)$$

$$\frac{\delta \hat{y}_i}{\delta a} = x_i$$

$$\frac{\delta \hat{y}_i}{\delta b} = 1$$

$$\frac{\delta E}{\delta a} = \sum_i^n \frac{\delta E}{\delta \hat{y}_i}\frac{\delta \hat{y}_i}{\delta a} = \frac{2}{n}\sum_i^n (\hat{y}_i - y_i)x_i$$

$$\frac{\delta E}{\delta b} = \sum_i^n \frac{\delta E}{\delta \hat{y}_i}\frac{\delta \hat{y}_i}{\delta b} = \frac{2}{n}\sum_i^n (\hat{y}_i - y_i)$$

Update parameters:

a' = 1-0.2(-1) = 1.2

b' = 1-0.2(4) = 0.2

2) **CONVOLUTION:** Given is the following data set. Perform convolution operations using the mask below.

CONVOLUTION - A BRIEF REMINDER

Convolution is an operation performed on elements lying next to each other.

For each element of the matrix, the weighted sum of the elements from the neighborhood is calculated, where the weights are the values from the mask.

$$y_{i,j} = x_{i,j} * w_{0,0} + x_{i,j+1} * w_{0,1} + x_{i+1,j} * w_{1,0} + x_{i+1,j+1} * w_{1,1}$$

Step 1:

Input:

0	0	0	0	0	0
0	1	0	1	1	1
0	1	1	1	0	1
0	1	1	0	1	0
0	1	1	1	0	0
0	0	0	0	0	0

Mask:

0	1
-1	0

Output:

0				

$$y_{0,0} = x_{0,0} * w_{0,0} + x_{0,1} * w_{0,1} + x_{1,0} * w_{1,0} + x_{1,1} * w_{1,1}$$

0*0 + 0*1 + 0*-1 + 1*0 = 0

Step 2:

Input:

0	0	0	0	0	0
0	1	0	1	1	1
0	1	1	1	0	1
0	1	1	0	1	0
0	1	1	1	0	0
0	0	0	0	0	0

Mask:

0	1
-1	0

Output:

0	-1			

$$y_{0,1} = x_{0,1} * w_{0,0} + x_{0,2} * w_{0,1} + x_{2,0} * w_{1,0} + x_{1,2} * w_{1,1}$$

0*0 + 0*1 + 1*-1 + 0*0 = -1

Step 7:

Input:

0	0	0	0	0	0
0	1	0	1	1	1
0	1	1	1	0	1
0	1	1	0	1	0
0	1	1	1	0	0
0	0	0	0	0	0

Mask:

0	1
-1	0

Output:

0	-1	0	-1	-1
1	-1			

1*0 + 0*1 + 1*-1 + 1*0 = -1

Step 17:

Input:

0	0	0	0	0	0
0	1	0	1	1	1
0	1	1	1	0	1
0	1	1	0	1	0
0	1	1	1	0	0
0	0	0	0	0	0

Mask:

0	1
-1	0

Output:

0	-1	0	-1	-1
1	-1	0	0	1
1	0	0	0	0
1	0			

1*0 + 1*1 + 1*-1 + 1*0 = 0

Step 25:

Input:

0	0	0	0	0	0
0	1	0	1	1	1
0	1	1	1	0	1
0	1	1	0	1	0
0	1	1	1	0	0
0	0	0	0	0	0

Mask:

0	1
-1	0

Output:

0	-1	0	-1	-1
1	-1	0	0	1
1	0	0	0	0
1	0	-1	0	0
1	1	1	0	0

0*0 + 0*1 + 0*-1 + 0*0 = 0