# Tutorial Business Analytics

Homework 12

**Exercise 12.3 Backpropagation II**

Continuing with exercise 12.2, consider the following feed-forward neural network that consists of

- An input layer ($l = 0$) representing two-dimensional points

$$a^{[0]} = \left(a_1^{[0]}, a_2^{[0]}\right)^T \in \mathbb{R}^2$$

- A hidden layer $l = 1$ with 2 hidden nodes and sigmoid activation function $g^{[1]}$
- An output layer $l = 2$ with one node and sigmoid activation function $g^{[2]}$.

a) Given the following dataset of $n = 4$ observations (in columns) and initial parameters, perform one forward pass (calculate the *empirical risk* $\mathcal{L}$).

$$X = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix}, \qquad Y = \begin{pmatrix} 0 & 1 & 1 & 1 \end{pmatrix}$$

$$W^{[1]} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad b^{[1]} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad W^{[2]} = \begin{pmatrix} 1 & -1 \end{pmatrix}, \quad b^{[2]} = (0)$$

b) Perform a partial backward pass and update the parameter $W^{[2]}$ using a gradient descent with learning rate $\alpha = 1$. Perform another forward pass on the full data. Did the risk decrease?

### Exercise 12.4 [Programming Exercise]

Perform one backward pass through the network of the previous exercise and update the parameters using gradient descent. Use learning rate $\alpha = 1$.

**Hint:** *For n observations, the full parameter derivatives of the risk function are given by:*

$$dW^{[1]} = \frac{1}{n} dZ^{[1]} X^T, \qquad db^1 = \frac{1}{n} \mathrm{rowSums}(dZ^{[1]}),$$
$$dW^{[2]} = \frac{1}{n} dZ^{[2]} A^{[1]^T}, \qquad \mathrm{d}b^2 = \frac{1}{n} \mathrm{rowSums}(dZ^{[2]})$$

*with*

$$dZ^{[1]} = W^{[2]^T} dZ^{[2]} * A^{[1]} * (1 - A^{[1]}), \qquad dZ^{[2]} = A^{[2]} - Y$$

*where* $*$ *is element-wise multiplication and for a given expression F we use the shorthand notation* $dF = {}^{\partial \mathcal{L}}/_{\partial F}$.

# Tutorial Business Analytics

Homework 12

**Exercise 12.3 Backpropagation II**

Consider the following feed-forward neural network that consists of

- An input layer ($l = 0$) representing two-dimensional points

$$a^{[0]} = \left(a_1^{[0]}, a_2^{[0]}\right)^T \in \mathbb{R}^2$$

- A hidden layer $l = 1$ with 2 hidden nodes and sigmoid activation function $g^{[1]}$
- An output layer $l = 2$ with one node and sigmoid activation function $g^{[2]}$.

a) Given the following dataset of $n = 4$ observations (in columns) and initial parameters, perform one forward pass (calculate the *empirical risk* $\mathcal{L}$).

$$X = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix}, \qquad Y = \begin{pmatrix} 0 & 1 & 1 & 1 \end{pmatrix}$$

$$W^{[1]} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad b^{[1]} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad W^{[2]} = \begin{pmatrix} 1 & -1 \end{pmatrix}, \quad b^{[2]} = (0)$$

To calculate the forward pass, we apply the four formulas for $z^{[1]}, a^{[1]}, z^{[2]}, a^{[2]}$ found in (a). In fact, when applying this formula to *multiple observations at once*, we can write observations in columns as in the data matrix X and still apply the same equations. Instead of vectors $z^{[1]}, a^{[1]}, z^{[2]}, a^{[2]}$ we will then get matrizes $Z^{[1]}, A^{[1]}, Z^{[2]}, A^{[2]}$ that have the individual results for each observation in their columns:

$$Z^{[1]} = W^{[1]}X + b^{[1]} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix}.$$

(Here we've used the following slight abuse of notation: when adding a vector to a matrix (with matching number of rows), we mean adding it to each column of that matrix separately.)

We continue by applying the activation function to each element of $Z^{[1]}$:

$$A^{[1]} = \sigma\left(Z^{[1]}\right) = \sigma\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{1+e^{-0}} & \frac{1}{1+e^{-0}} & \frac{1}{1+e^{-1}} & \frac{1}{1+e^{-1}} \\ \frac{1}{1+e^{-0}} & \frac{1}{1+e^{-1}} & \frac{1}{1+e^{-0}} & \frac{1}{1+e^{-1}} \end{pmatrix}$$

$$\approx \begin{pmatrix} 0.5 & 0.5 & 0.731 & 0.731 \\ 0.5 & 0.731 & 0.5 & 0.731 \end{pmatrix}.$$

Doing the same for the second layer, we get:

$$Z^{[2]} = W^{[2]}A^{[1]} + b^{[2]} = \begin{pmatrix}1 & -1\end{pmatrix}\begin{pmatrix}0.5 & 0.5 & 0.731 & 0.731 \\ 0.5 & 0.731 & 0.5 & 0.731\end{pmatrix} + 0$$

$$= \begin{pmatrix}0 & -0.231 & 0.231 & 0\end{pmatrix},$$

$$A^{[2]} = \sigma\left(\begin{pmatrix}0 & -0.231 & 0.231 & 0\end{pmatrix}\right) \approx \begin{pmatrix}0.5 & 0.442 & 0.558 & 0.5\end{pmatrix}.$$

Using the loss function given in exercise b), we can calculate the loss of the network output $A^{[2]}$ element wise:

$$\begin{aligned}
\ell\left(Y, A^{[2]}\right) \approx\ & -\left[Y * \ln A^{[2]} + (1 - Y) * \ln\left(1 - A^{[2]}\right)\right] \\
=\ & -\Big[\ \begin{pmatrix}0 & 1 & 1 & 1\end{pmatrix} * \ln\left(\begin{pmatrix}0.5 & 0.442 & 0.558 & 0.5\end{pmatrix}\right) \\
& \quad + \begin{pmatrix}1 & 0 & 0 & 0\end{pmatrix} * \ln\left(\begin{pmatrix}0.5 & 0.558 & 0.442 & 0.5\end{pmatrix}\right)\Big] \\
=\ & -\big[\ \begin{pmatrix}0 & 1 & 1 & 1\end{pmatrix} * \begin{pmatrix}-0.693 & -0.815 & -0.584 & -0.693\end{pmatrix} \\
& \quad + \begin{pmatrix}1 & 0 & 0 & 0\end{pmatrix} * \begin{pmatrix}-0.693 & -0.584 & -0.815 & -0.693\end{pmatrix}\big] \\
=\ & -\left[\begin{pmatrix}0 & -0.815 & -0.584 & -0.693\end{pmatrix} + \begin{pmatrix}-0.693 & 0 & 0 & 0\end{pmatrix}\right] \\
=\ & \begin{pmatrix}0.693 & 0.815 & 0.584 & 0.693\end{pmatrix}.
\end{aligned}$$

The *risk* is given by the average loss in each observation:

$$\mathcal{L}\left(Y, A^{[2]}\right) = \frac{1}{n}\sum_{i=1}^{n} \ell\left(y_{(i)}, a_{(i)}^{[2]}\right) = \frac{1}{4} \cdot (0.693 + 0.815 + 0.584 + 0.693) = 0.696$$

b) Perform a partial backward pass and update the parameter $W^2$ using a gradient descent with learning rate $\alpha = 1$. Perform another forward pass on the full data. Did the risk decrease?

We can perform the update step using the gradient found in exercise b).

$$dW^{[2]} = dz^{[2]} \cdot \begin{pmatrix}\frac{\partial z^{[2]}}{\partial w_1^{[2]}} & \frac{\partial z^{[2]}}{\partial w_2^{[2]}}\end{pmatrix} = \left(a^{[2]} - y\right) \cdot a^{[1]^T}$$

In exercise b), however, we only considered the backward pass for a single observation $(x, y)$. When dealing with multiple observations at once, we need some minor adjustments:

1. Replace $dz^{[2]}$ with $dZ^{[2]} = A^{[2]} - Y$
2. Replace $a^{[1]^T}$with $A^{[1]^T}$
3. We are actually interested in the derivative of the *risk*, rather than the univariate *loss* for each observation. The *risk* is $\frac{1}{n}\sum(\text{losses})$, so the calculations above would give us the sum of gradients for individual observations. However, we're interested in the averages, so we have to add a factor of $1/n$.

In total we get:

$$dW^{[2]} = \frac{\partial \mathcal{L}}{\partial W^{[2]}} = \frac{1}{n}\left(A^{[2]} - Y\right)A^{[1]^T}$$

$$= \frac{1}{4}\left(\begin{pmatrix}0.5 & 0.442 & 0.558 & 0.5\end{pmatrix} - \begin{pmatrix}0 & 1 & 1 & 1\end{pmatrix}\right)\begin{pmatrix}0.5 & 0.5\\ 0.5 & 0.731\\ 0.731 & 0.5\\ 0.731 & 0.731\end{pmatrix}$$

$$= \frac{1}{4}\left(\begin{pmatrix}0.5 & -0.558 & -0.442 & -0.5\end{pmatrix}\right)\begin{pmatrix}0.5 & 0.5\\ 0.5 & 0.731\\ 0.731 & 0.5\\ 0.731 & 0.731\end{pmatrix}$$

$$= \frac{1}{4}\begin{pmatrix}-0.718 & -0.744\end{pmatrix} = \begin{pmatrix}-0.179 & -0.186\end{pmatrix}$$

We can thus perform one gradient update step:

$$W_{new}^{[2]} = W^{[2]} - \alpha \cdot dW^{[2]} = \begin{pmatrix}1 & -1\end{pmatrix} - 1 \cdot \begin{pmatrix}-0.179 & -0.186\end{pmatrix} = \begin{pmatrix}1.18 & -0.814\end{pmatrix}$$

To check whether the risk decreased, let’s perform another forward pass. Everything up to $A^{[1]}$ is the same as in exercise c) because we didn’t change anything in the first layer. In layer 2 we get:

$$Z_{new}^{[2]} = W_{new}^{[2]}A^{[1]} + b^{[2]} = \begin{pmatrix}1.18 & -0.814\end{pmatrix}\begin{pmatrix}0.5 & 0.5 & 0.731 & 0.731\\ 0.5 & 0.731 & 0.5 & 0.731\end{pmatrix} + 0$$

$$= \begin{pmatrix}0.183 & -0.005 & 0.455 & 0.267\end{pmatrix}$$

$$A_{new}^{[2]} = \sigma\left(\begin{pmatrix}0.183 & -0.005 & 0.455 & 0.267\end{pmatrix}\right) \approx \begin{pmatrix}0.546 & 0.499 & 0.612 & 0.566\end{pmatrix}$$

Resulting in losses of

$$\ell_{new} = \ell\left(Y, A_{new}^{[2]}\right) = \begin{pmatrix}0.789 & 0.696 & 0.491 & 0.568\end{pmatrix}$$

And a new risk/average loss of

$$\mathcal{L}_{new} = \frac{1}{4}(.789 + .696 + .491 + .568) = 0.635$$

As the previous risk was $0.696$, we see that even updating just a single matrix of parameters improved the neural network.

**Exercise 12.4 [Programming Exercise]**

See R script.