Università della Svizzera italiana

Facoltà di scienze informatiche

Machine Learning **2018**

Student: Sébastien Bouquet

Solution for Assignment 1

Due date: 19/10 2018, 09:00pm

1. The Perceptron *(20 Points)*

1. $\bar{y} = \sum_j \bar{x}_j w_j + \bar{b}$ where $\bar{b} = (w_{k+1}, w_{k+1}, ..., w_{k+1})^T, \bar{y}, \bar{x}_j, \bar{b} \in \mathbb{R}^n, \bar{w} \in \mathbb{R}^{k+1}$.
   $\bar{y} = \bar{w}\mathbf{A}$ where $\mathbf{A} = (\bar{x}_1, \bar{x}_2, ..., \bar{x}_k, \mathbb{1}), \mathbb{1} = (1, 1, ..., 1, 1)^T, \mathbb{1} \in \mathbb{R}^n, \mathbf{A} \in \mathbb{R}^{n \times k+1}$.

2. Unvectorized: $\bar{E}_i = \frac{1}{2}(\bar{y}_i - \bar{t}_i)^2$.
   $E = \frac{1}{2}\sum_j (\bar{y}_j - \bar{t}_j)^2$.

3. $$\frac{\partial E}{\partial w_i} = \frac{\partial \frac{1}{2}[(\bar{y}_1 - \bar{t}_1)^2 + (\bar{y}_2 - \bar{t}_2)^2 + ... + (\bar{y}_n - \bar{t}_n)^2]}{\partial w_i} = [(\bar{y}_1 - \bar{t}_1) + (\bar{y}_2 - \bar{t}_2) + ... + (\bar{y}_n - \bar{t}_n)]\frac{\partial \bar{y}}{\partial w_i}$$
   $$= \sum_{j=1}^{n} (\bar{y}_j - \bar{t}_j)\bar{x}_j$$
   .

4. $w_i^{new} = w_i^{old} - \Delta w_i$ where $\Delta w_i = \eta \sum_j (\bar{y}_j - \bar{t}_j)\bar{x}_j$.
   Why can it be used as a learning algorithm ?
   It does update the weights of the inputs compared to the targets by following the gradient which means it should reduce the error at each step.
   The main problem is that the weights cannot modify only a subset of their corresponding input vectors. So the whole input vector will be modified even if only a few subset of its elements are responsible for the error.

5. $\bar{w}^{new} \approx [-0.110, -0.309, 0.190, 2]$ with normalized input vectors or
   $\bar{w}^{new} \approx [-0.1986, -0.3822, 0.0276, 2]$ with non normalized input vectors.
   These results where found using the Jupyter Notebook called *ex1-5.ipynb*.

6. Gradient descent: compute the gradient of the input with regard to the weights at each step of the network. Back propagation: updates the weights and the biases at each layer of the network.

2. Simple Machine Learning Framework *(45 Points)*

I implemented most of the framework but the weight updates seems wrong because the weights only change a little bit in the beginning and then remain constant.
The gradient may also not be correct because I was unable to implement the *check_gradient* function correctly.

3. Handwritten Digit Recognition *(35 Points)*

Unfortunately, I could not even start this exercise since Section 2 was not finished.