Programming Assignment 3

Write a program that implements a single perceptron using the delta rule presented in the lecture. Use the following activation function:

$$\hat{y} = \begin{cases} 1 & \text{if } \vec{w}^T \vec{x} > 0 \\ 0 & \text{else} \end{cases}$$

where $\vec{w}$ is the vector of weights including the bias (w_0). Treat all attributes and weights as double-precision values.
Given are the two data sets[1] named *Example* and *Gauss* as tsv (tabular separated values) files. Your program should be able to read both data sets and treat the *first* value of each line as the class (A or B). In order to get the same results, class A is to be treated as the positive class, hence $y = 1$, and class B as the negative one ($y = 0$). Your task is to correctly implement the perceptron learning rule in batch mode with a constant ($\eta^t = \eta^0$) and an annealing ($\eta^t = \frac{\eta^0}{t}$) learning rate, i.e:

$$\vec{w}^{t+1} \leftarrow \vec{w}^t + \sum_{\vec{x} \in \mathcal{Y}(\vec{x}, \vec{w})} \eta^t (y - \hat{y}) \vec{x}$$

where $\mathcal{Y}(\vec{x}, \vec{w})$ is the set of samples which are misclassified. Please use the number of misclassified points as your error rate (i.e. $|\mathcal{Y}(\vec{x}, \vec{w})|$). The output of your algorithm should be a *single* tsv file, which contains *exactly* two rows after 100 iterations (per variant):

1. The first row contains the tabular separated values for the error of each iteration (starting from iteration 0) with the constant learning rate.

2. The second row follows the same format, but with the annealing learning rate.

The iteration number and any other information should **not** be inside the output file, only the error values. You can check the solution for the *Example* data set in order to compare it to your output file. For each data set, you can acquire one point, if the solution of your program returns correct results. If the program fails, the data format is incorrect or I have to change source code, in order to make it work, you will get zero points. Machine learning libraries are not allowed. You can use libraries for handling the CSV/TSV format and the input parameters.
Your program must *at least* accept the following parameters:

1. **data** - The location of the data file (e.g. /media/data/car.csv).

2. **output** - Where the output tsv should be written to.

[1] http://wwwiti.cs.uni-magdeburg.de/iti_dke/Lehre/Materialien/WS2018_2019/ML/res/perceptron.zip

You can add more, if you want to, e.g. the maximum number of iterations. Please prepare example statements on how to use your program. E.g. for a python program:

```
python3 perceptron.py --data Example.tsv --output Example_Errors.tsv
```

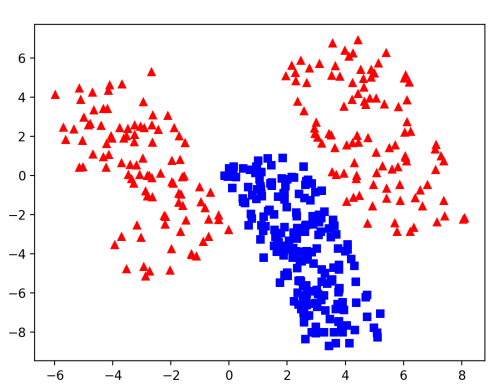
The final program code must be sent via email until Sunday, 9th of December 2018, 23:59 to marcus.thiel@ovgu.de. Please format your e-mail header as follows:

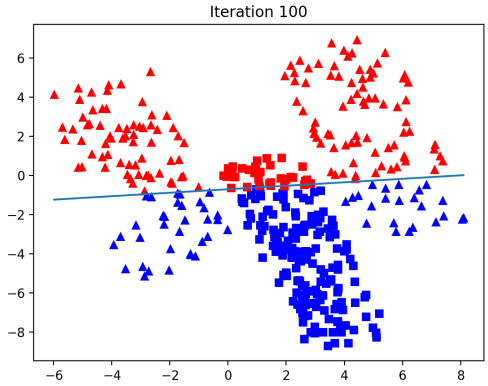
[Exercise Group] ML Programming Assignment 3

Replace *Exercise Group* with the day and time of your exercise group. E.g for Monday from 13:00 to 15:00 it would be:

[Monday 13-15] ML Programming Assignment 3

Please also be prepared to present your solution shortly in front of the class.
The figures below shows the data for the *Example* set and its single perceptron solution after 100 iterations with a constant learning rate.





2 points