Neural Networks

Project 3

Stanisław Kaźmierczak

Faculty of Mathematics and Information Science,
Warsaw University of Technology

2021

Overview and timetable

- 30 points for the project
- groups of 2 people
- starting each new week of delay results in minus 5 points (max. 2 weeks delay)
- timetable

1.	18.05	Project Presentation
2.	25.05	Tutorial hours
3.	01.06	Initial presentation of the first part
4.	08.06	Project I (first part) deadline
5.	15.06	Project I (second part) deadline

Rules

- tutorial hours are optional
- other classes are obligatory
- absence from mandatory classes results in -2 points
- let me know a day before if you plan to attend tutorial hours later than 15 minutes from the beginning
- if I don't receive such information and there is no one present, I consider such classes as completed
- source code and report should be sent before the beginning of the class
- you can utilize code from external sources (books, articles, blogs) provided that:
  - reference is cited in the report
  - some modifications to the original solution are applied
- violation of the above or any other kind of plagiarism results in a failing grade

Reports

The report should include:

- description of the research problem, understandable to the person who didn't see the content of the task
- instruction of the application (containing information on how to reproduce results)
- theoretical introduction
- description of the conducted experiments
- statistically processed results (presented clearly)
- conclusions, presumed reasons for successes/failures and further research proposals

Reports

Some additional remarks:

- if the experiment isn't described in the report it is regarded as not conducted
- the report is an official document, so please take care of its formal form (table of contents, bibliography, captions under figures, tables, etc.)
- results should be commented
- to obtain statistically significant results, each experiment ought to be repeated multiple times
- in addition to the mean, standard deviation should also be calculated (in some scenarios worse mean with low variance may be a more desirable result than a better mean with high variance)

Project III - multilayer perceptron

Topic: **Multilayer perceptron (MLP) employing backpropagation algorithm**

Dataset for the first part of the project will be provided during the first class.

Dataset for the second part: MNIST
`https://www.kaggle.com/c/digit-recognizer`

General guidelines

- take care of reproducibility by initializing a random number generator with a constant seed
- training models on other datasets than mentioned in the task description is not allowed

Project III - multilayer perceptron

- low-level implementation is required; you can use only some basic packages like NumPy, pandas, etc.
- MLP parameters are:
  - number of hidden layers and number of neurons in hidden layers
  - activation function
  - bias presence
  - batch size
  - number of iterations
  - learning rate
  - momentum
  - problem type: classification or regression
- during the project presentation, you will be asked to train and test your network on new (unseen before) datasets
- take care of reproducibility by initializing a random number generator with a constant seed

Project III - multilayer perceptron

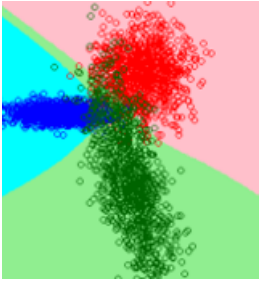
- elements to analyze:
  - how does activation function affect the model's accuracy? Experiment with sigmoid and two other activation functions. The activation function in an output layer should be chosen accordingly to the problem;
  - how does the number of hidden layers and number of neurons in hidden layers impact the model's accuracy? Analyze different architectures;
  - how does the loss function affect the model's accuracy? Consider two different loss functions for both classification and regression.
- the application should plot training and test error
- user should be able to track learning process iteration by iteration (visualization of edges' weights) as well as a propagated error (visualization of an error on each edge)
- visualization of a training set and classification/regression result (as a background)

Assessment: source code, report

Project III - multilayer perceptron

	A	B	C	D
1	x,y,cls			
2	-0.00292545510455966,0.722530109807849,1			
3	0.504570618271828,-0.789261351339519,2			
4	0.894131738692522,-0.720432524569333,1			
5	0.220206981524825,0.12407802278176,2			
6	0.267815329600126,0.925336269196123,1			
7	-0.27340721199289,0.479693677276373,1			
8	-0.287346473895013,-0.972735119983554,2			
9	0.467718373052776,0.81061793724075,1			
10	0.883102677632123,-0.181434742640704,1			

	A	B
1	x,y	
2	-5,-1253	
3	-4.99,-1247.368296	
4	-4.98,-1241.753168	
5	-4.97,-1236.154592	
6	-4.96,-1230.572544	
7	-4.95,-1225.007	
8	-4.94,-1219.457936	
9	-4.93,-1213.925328	
10	-4.92,-1208.409152	
11	-4.91,-1202.909384	
12	-4.9,-1197.426	
13	-4.89,-1191.958976	
14	-4.88,-1186.508288	
15	-4.87,-1181.073912	



Project III - multilayer perceptron

- The second part of the project consists in fitting the implemented model to the deep learning *Hello world* dataset - MNIST
- The achieved score should be registered in Kaggle platform

Assessment: source code, extended report (or supplement to the report)