

MEIC 2018/2019
Aprendizagem - Machine Learning
Homework III
Prof- Manuel Lopes and Prof. Andreas Wichert
Deadline 26/04/2019
Submit on Fenix

Submission instructions:

Submit a single .zip file on Fenix containing:

- 1 pdf with a 2 page report
- 1 .ipynb file (from colab) with the working code

Every file name must be the group number

The report (max 2 page double column) must include a description of the code, a working link to the code on colab, the formulas used and an explanation, the results and a discussion on the results.

1. (10pts) Implement and compare two basic neural network structures with different cost functions, and initializations.
(this code can only rely on functions from numpy and scipy, it is not allowed to use scikit-learn/keras/tensorflow for learning)

(5pts) Consider a single neuron and implement:
-linear unit with quadratic error as loss function
-sigmoid unit with quadratic error as loss function
-sigmoid unit with cross entropy as loss function

Use the famous dataset that contains 150 iris flowers to detect Iris-Viginica or Not Iris-Viginica (two classes) to investigate gradient descent of one unit:
https://scikit-learn.org/stable/auto_examples/datasets/plot_iris_dataset.html

Example code to load and use two classes.

```
from sklearn import datasets
import numpy as np

iris = datasets.load_iris()
print("Iris Data\n", list(iris.keys()))

X = iris.data[:, :2] # we only take the first two features.

y = (iris["target"] == 2).astype(np.int) # 1 if Iris-Virginica, else 0

print(y)

#This is how you would do it in sklearn

from sklearn.linear_model import LogisticRegression

log_reg = LogisticRegression()

log_reg.fit(X,y)

print("Size y:", len(y))

print("Predict:", log_reg.predict(X[:len(y), :]))
```

MEIC 2018/2019
Aprendizagem - Machine Learning
Homework III
Prof- Manuel Lopes and Prof. Andreas Wichert
Deadline 26/04/2019
Submit on Fenix

(5pts) Describe the formulas used for the training. Discuss the convergence (does it always converges, depends on the initialization?) rates of each method, the final training and test error.

2. (10pts) For a more complex dataset implement classifier based on neural networks (it is allowed to use keras/tensorflow).

Use the following dataset: <https://archive.ics.uci.edu/ml/datasets/Breast+Cancer>

(5pts) Consider at least one hidden layer, and test different activation functions (sigmoid, relu and leaky relu), and loss functions. Consider also different methods of regularization.

(5pts) Discuss the training and test error, the convergence rate, the method to select the best model and the regularization.

It is acceptable to base your code on examples from the web. But provide references for everything and you should be able to explain all lines of code.