TP4 Data Mining: Softmax Classifier - Gradient Descent - L2 Regularizer
OBLIGATORY

Tuesday 9th April, 2019
To be sent to Frantzeska.Lavda@etu.unige.ch
deadline: Monday 29th April, 2019, 23:59

1 Objective

The goals of this TP is to optimize the softmax classifier with an $L2$ norm regularizer.

2 Detailed Instructions

This TP is obligatory. You are going to fill a few missing functions in the python script to implement the exercises that we ask. So first of all read and understand the given python script. To run your code you have to run the TP4_softmax_gd_regularizer.ipynb notebook. You are going to use the iris data set.

You have to send a **formal** report and your code.

For this TP the following steps will need to be done:

1. The cost function of the softmax classifier
   (a) Write down the cost function of the softmax classifier on your report
   (b) Derive the gradient of the softmax classifier cost with respect to the weights (learning parameters) (Write down the derivations on your report)
   (c) Write down the cost function of the softmax classifier when a L_2 regularizer is added
   (d) Derive the the gradient of the softmax classifier cost when a L_2 regularizer is added (Write down on your report the final gradient)

2. Based on your derivations, implement function for the softmax classifier and its derivative with an L_2 regularizer. If you don't include in your report your derivations your code will not be taken into account. Same if your code it is not based on your derivations.

   (a) Fill the softmax() function inside the Softmax_classifier.py

   (b) Fill the softmax_loss() function inside the Softmax_classifier.py

   (c) Fill the softmax_loss_gradient() function inside the Softmax_classifier.py

3. Implement Stochastic Gradient Descent, SGD

   (a) Fill the missing part of the train() function inside the Softmax_classifier.py

   (b) The size of the batch is given as argument in the train() function and in your experiments you will use online SGD, 2. SGD with mini batch = 20, 3.mini_batch = size of the training data set (Gradient Descent, GD)

4. Fill the missing part in TP4_softmax_gd_regularizer.ipynb notebook and train your classifier for different learning rates, regularizes, and size of mini bach. Set the number of the iteration equal to 500 iteration and add a stopping criterion.

   (a) Comment **in details** how the different learning rates, regularizes, and size of mini bach influence the performance of the classifier. Which is the effect of very large/ small regularizes? How the learning rate change the prediction? Which (and why) is the optimal why to update the weights? etc..

3 Reminders

- The Softmax classifier is the generalization of the binary Logistic Regression classifier to multiple classes. Tthe Softmax classifier gives as output normalized class probabilities

- When we minimize the cost function using Gradient Descent (GD) the weights are updated after seeing all the training instances

- When we minimize the cost function using Stachastic Gradient Descent (SGD) the weights are updated after seeing a mini batch the training instances. When the size of the mini batch is equal to 1 is called single sample update/ online SGD and the weights are updated each time we see a new instance.

- The L2 regularizer is also called Ridge Regression. It adds "squared magnitude" of weights as penalty term to the cost function. If the regularizer parameter is zero then you can imagine we get back to the cost function.