Project
Data Mining: Learning from Large Data Sets
FS 2015

LAS Group, Machine Learning Institute
Dept. of Computer Science, ETH Zürich
Prof. Dr. Andreas Krause
Web: las.ethz.ch/courses/datamining-f15/
Email questions to:
lucic@inf.ethz.ch

Project 2, Oct 18, 2015
Large Scale Image Classification

1 Introduction

In this task our goal is to classify images based on their visual content. We provide a labeled dataset containing two images from two classes: **Nature** and **People**. While feature selection is a very important part of image classification pipeline, it is out of the scope of this course and won't be part of this task. Instead, we provide a set of features that has been extracted from each image. Your task is to train a model that perfroms well given the feature representation. Furthermore, we will make use of Support Vector Machines and implement the solution using Parallel Stochastic Gradient Descent.

2 Input and Output Specification

A set of 400 features has been extracted from each picture. We provide $100k$ images, $50k$ from each category.

The training set is given in the file "training". Each line in the file corresponds to an image and is formatted as follows:

- 401 space-separated doubles.
- The first element in the line is the class $y \in \{+1, -1\}$ which correspond to Nature and People class, respectively.
- The following 400 elements are real numbers which represent the feature values $x_0 ... x_{399}$

3 Evaluation and Grading

Your task is to provide a Map function and a Reduce function written in Python.

The output of the Reduce function should be the set of space-separated coefficients of your final model. The model produced by the Reducer will be used for evaluation on a separate dataset. The prediction of your model on a test instance $\mathbf{x}$ will be calculated as $\hat{y} = \mathbf{w}^T\mathbf{x}$. If you decide to apply any transformation to the given features your predictions will be given by $\hat{y} = \mathbf{w}^T\phi(\mathbf{x})$. By default $\phi(\mathbf{x}) = \mathbf{x}$.

If you apply transformations to the original features you have to change the "transform" function in the provided mapper template. Otherwise, your submission will not be evaluated correctly as we will not use the same feature transform for the evaluation. The evaluation code is provided in the supporting material.

Each submission will be scored according to the classification accuracy which is the percentage of the test instances that were classified correctly.

We will compare the score of the submission with two baseline predictions: a weak one (called "baseline easy") and a strong one (called "baseline hard"). These will have a score of PBE and PBH respectively, calculated as described above. Both baselines will appear in the rankings together with the score of your submitted predictions.

Performing better than the weak baseline will give you 50% of the grade, and matching or exceeding the strong baseline on the **test set** will give you 100% of the grade. This allows you to check if you are getting at least 50% of the grade by looking at the ranking. If your prediction performance on the **test set** (P_{test}) is in between the baselines, the grade is computed as:

$$\text{Grade} = \left(\frac{PBH_{test} - P_{test}}{PBH_{test} - PBE_{test}} \right) \times 50\% + 50\%$$

The number of submissions per team is limited to 20. Time limit per submission is 5 minutes. The number of mappers that will be used is 10. There will be only one reducer. The submission with the highest accuracy on the evaluation set will be used for grading.

3.1 Visual Test

We provide the actual images for a small subset of the training dataset. Furthermore, there is a script that evaluates the predictions and produces a visualization of the classification accuracy of your model. You can find it under the folder "visual_test".

3.2 Additional Python Libraries

You are free to use NumPy (1.9.2), SciPy (0.15.1) as well as scikit-learn (0.16.1).

3.3 Report

You have to submit one report per group describe your work. You should indicate in the report which projects members contributed to which parts of your solution. Please keep the reports brief (under 2 pages). You are requested to upload a ZIP archive containing the team report *and* the code. We included a template for LaTeX in the file `report.tex`. If you do not want to use LaTeX, please use the same sections as shown in `report.pdf`. The reports are uploaded on the same page as the test set submissions. Keep in mind that you only have to submit one report per group. Please indicate the contribution of each group member to the project.

3.4 Deadline

The submission system will be open from **Monday, 18.10.2015, 08:00** until **Sunday, 01.11.2015, 23:59:59**.