INFO3406 – Introduction to Data Analytics
Semester 2, 2016

Assignment 1. Image Similarity Matching and Classification

Submission deadline: Wednesday, week 9, 5pm (21 September).

Late policy submission: A penalty of -1 mark will apply for each day late and the assignment will not be accepted if it is submitted more than 7 days after the due date. The cut-off time is 5pm.

This assignment can be completed individually or in pairs. Working in pairs is encouraged. Both students will receive the same mark.

Submission instructions:

> You need to submit an Electronic version (report + code + plagiarism cover sheet) via eLearning. All files should be zipped together in a single file. The zip file should be named 0123456.zip, where 0123456 is your SID. In case of a pair submission, put both SIDs separated by an underscore: 0123456_0789123.zip. Only one of the two students needs to submit.

Programming language: You are encouraged to write the program in Python. Alternatively, you can also use Matlab, Java, or C++ but we need to be able to test your code on the University machines. You need to include instructions on how to compile (if necessary) and run your code.

You should write your own code to calculate the similarity scores and classification. However, if you are running an optimization algorithm, you can use off-the-shelf libraries such as nlopt or scipy.optimize. You are NOT allowed to use sophisticated libraries such as scikit-learn.

Weight: This assignment is worth 20 marks = 20% of your final mark.

The goal of this assignment is to; 1) implement a large-scale image similarity matching system; 2) determine classification accuracy.

Other resources: 80 million tiny images: a large dataset for non-parametric object and scene recognition (http://people.csail.mit.edu/torralba/publications/80millionImages.pdf) will be useful.

Instructions:

Part I

Download the CIFAR-10 dataset available at http://www.cs.toronto.edu/~kriz/cifar.html. It consists of 60000 32x32 colour images in 10 physical object classes. The dataset is divided into five training batches and one test batch, each with 10000 images. The test batch contains exactly 1000 randomly-selected images from each class. You can use the test batch ONLY for testing, NOT for training. Your algorithm should be able to classify a query image of the same size.

Part II

Download the CIFAR-100 dataset available at http://www.cs.toronto.edu/~kriz/cifar.html. This dataset is similar to the CIFAR-10, except it has 100 classes containing 600 images each. There are 500 training images and 100 testing images per class. You can use the testing images ONLY for testing, NOT for training. For a query image of the same size, your classifier should be able to,

1) Infer the superclass (e.g. aquatic mammals)
2) Infer the class (e.g. beaver)

Report:

- Similarity metrics you used
- Your attempts to make the classifier robust (**invariant to translation, mirror, rotation,** etc.)
- What images will be/not be properly classified? For instance, a bird perched on a tree will be accurately classified while a flying bird will not. Identify the reasons for misclassification and plausible corrective measures.
- *Accuracy score* and *confusion matrix or precision/recall* to evaluate the accuracy of your classification. Use test images for this purpose.
  Hint: See the description in http://scikit-learn.org/stable/modules/model_evaluation.html.
- Speed-accuracy trade-off
- References in the IEEE style.

Evaluation:

Query images (height×width×colour_channels=32×32×3 pixels, unit8 RGB, png, image files will be used) for evaluation. For instance, http://www.cs.toronto.edu/~kriz/cifar-10-sample/automobile1.png. However, these images may NOT be from the test set or training set.

The evaluators will name files as “img00”, “img01”, “img02”, etc. and save in a folder named ”INFO3406_assignment1_query”. Your program should be able to query all images and output a single csv file that only contains the output labels. For example, the output may be “0, 2, 1, 3, 6, 6, etc”, where each number corresponds to a class.

INFO3406 Assignment 1 – Marking Sheet
Marked out of 20

Student(s):

	Your mark	Comments
1. [10 marks] Report [0.5 marks] Introduction – What is the aim of the study – Why is this study (the problem) important [5 marks] Methods – Similarity metrics are well described – Attempts to make the classifier robust [3 marks] Analysis of results and discussion – Accuracy score – Confusion matrix – Reasons for misclassifications [0.5 mark] Conclusions and future work – Meaningful conclusions based on the results – Meaningful future work suggested [0.5 mark] Discussion (meaningful and relevant personal reflection) [0.5 mark] English and presentation – academic style, grammatical sentences, no spelling mistakes – good structure and layout; consistent formatting – appropriate citation and referencing		
2. [5 marks] Classifier accuracy Classifier accuracy (%) × 5		
3. [4 marks] Code Code runs and computes the correct posterior probability using likelihood weighting.		
4. [1 mark] At the discretion of the marker - for impressing the marker, excelling expectation. Examples include fast code, very well documented code, extensive analysis of the results		
Penalties: – 2 mark maximum for badly written code or code that is not well documented and difficult to read. – 2 marks for not including instructions on how to run your code – Penalty for late submission: -1 mark for each day late		
Total (out of 20):		