The Final Project of "introduction to Statistical Learning and Machine Learning"

Yanwei Fu

December 2, 2016

Abstract

(1) This is the second project of our course. The project is released on Dec 1st, 2016. The deadline is 5:00pm, Jan 19, 2017. Please send the report to cliao15@fudan.edu.cn. The late submission is also acceptable; however, you will be penalized 10% of total scores for EVERY TWO DAYS' delay (by 5:00pm of that day). In other words, you won't get any scores for this project if you submit it later than Feb. 9. 2017.

(2) You will get 3-4 papers soon after for you to review. That is, you will review anonymously (double blind) 3 papers submitted by your colleagues. The reviews are due back on Feb. 10th at 5pm (tentative). Each review should be about 1 page. It must assign marks between 1 and 10 to each paper, as suggested by NIPS instructions. For submission of your reviews, email me a PDF file of 3-4 pages only. The title of each page should be the paper title of the paper under review with your score in brackets. In general, we will try to assign each paper with at least 5 reviewers. If you are failed to submit the reviews on time for one paper-review you assigned, you will get 1% penalty of your total scores for EACH DAY's delay on this project.

(4) Note that if you are not satisfied with the initial report, the updated report will also be acceptable given the necessary score penalty of late submission.

(5) OK! That's all. Please let me know if you have any additional doubts of this project. Enjoy!

1 Introduction

1.1 Collaboration Policy

You are allowed to work in a group with at most two collaborators. you will be graded on the creativity of your solutions, and the clarity with which you are able to explain them. If your solution does not live up to your expectations, then you should explain why and provide some ideas on how to improve it. You are free to use any third-party ideas or code that you wish as long as it is publicly available. You must provide references to any work that is not your own in the write-up.

1.2 Writing Policy

Final project (20%) is finished by one team. Each team should have up to 3 students; and will solve a real-world Big-Data problem. The final report should be written in English. The main components of the report will cover

1. Introduction to background and potential applications (2%);
2. Review of the state-of-the-art (3%);
3. Algorithms and critical codes in a nutshell (10%);

4. Experimental analysis and discussion of proposed methodology (5%).

Please refer to our latex example: `http://yanweifu.github.io/courses/Chap9_Computation_learning/IEEE_TAC_2016.zip`

1.3 Submitting Policy

The paper must be in NIPS format (downloadable from [1]) and it must be double-blind. That is, you are not allowed to write your name on it etc. For more info, please read: NIPS reviewieng and double blind policy.

Package your code and a copy of the write-up pdf document into a zip or tar.gz file called finalProject-*your-student-id1_student-id2_student-id3.[zip|tar.gz]. Also include functions and scripts that you had used. To submit the report, email the pdf file to cliao15@fudan.edu.cn. In the submission email, you should well explain the authours and co-workers of this project.

1.4 Evaluation of Final Projects

You will get 3-4 papers soon after for you to review. That is, you will review anonymously (double blind) 3 papers submitted by your colleagues. The reviews are due back on Feb. 10th at 5pm. Each review should be about 1 page. It must assign marks between 1 and 10 to each paper, as suggested by NIPS instructions. For submission of your reviews, email me a PDF file of 3-4 pages only. The title of each page should be the paper title of the paper under review with your score in brackets. e.g. "Least Squares for Energy Prediction (7)". The rest of each page (one page per review) should be a text evaluation of the work you are reviewing. It should be based on the following NIPS criteria:

Overview: you should briefly summarize the main content of this paper, as well as the Pros and Cons (advantages and disadvantage) in general. This part aims at showing that you had read and at least understand this paper.

Quality: Is the paper technically sound? Are claims well-supported by theoretical analysis or experimental results? Is this a complete piece of work, or merely a position paper? Are the authors careful (and honest) about evaluating both the strengths and weaknesses of the work?

Clarity: Is the paper clearly written? Is it well-organized? (If not, feel free to make suggestions to improve the manuscript.) Does it adequately inform the reader? (A superbly written paper provides enough information for the expert reader to reproduce its results.)

Originality: Are the problems or approaches new? Is this a novel combination of familiar techniques? Is it clear how this work differs from previous contributions? Is related work adequately referenced?

Significance: Are the results important? Are other people (practitioners or researchers) likely to use these ideas or build on them? Does the paper address a difficult problem in a better way than previous research? Does it advance the state of the art in a demonstrable way? Does it provide unique data, unique conclusions on existing data, or a unique theoretical or pragmatic approach?

You project mark will mostly be based on the scores you get from your colleagues. I will simply play the role of chair and calibrate the scores to make sure there is no bias. I will also control 20-40% of the mark and this will be based on the quality of the reviews.

[1] https://nips.cc/Conferences/2016/PaperInformation/StyleFiles

1.4.1 Minimum Requirements

For all the projects listed below, in general you should devise your own machine learning algorithms which target at each specific problem of each project. You should compare with the machine learning algorithms taught in this course/mini-projects, which, include but not limited to, *linear regression/classification, K-NN/NN, logistic regression, linear/RBF kernel SVM, Neural network* as well as *tree-based methods*. Thus, the minimum requirements, as you can image, just apply and compare with these methods; and explain the advantage and disadvantage of using these methods for the project problem. Note that, your algorithms can be derived from one of these machine learning algorithms; and feel free to use any machine learning package you like.

2 Potential Projects

2.1 Detect the location of keypoints on face images

2.1.1 Introduction to this project

This project comes from Kaggle: `https://www.kaggle.com/c/facial-keypoints-detection`.

The objective of this task is to predict keypoint positions on face images. This can be used as a building block in several applications, such as: tracking faces in images and video analysing facial expressions detecting dysmorphic facial signs for medical diagnosis biometrics / face recognition Detecing facial keypoints is a very challenging problem. Facial features vary greatly from one individual to another, and even for a single individual, there is a large amount of variation due to 3D pose, size, position, viewing angle, and illumination conditions. Computer vision research has come a long way in addressing these difficulties, but there remain many opportunities for improvement.

This getting-started competition provides a benchmark data set and an R tutorial to get you going on analysing face images. For more details, please refer to that website.

2.1.2 Data File

Each predicted keypoint is specified by an (x,y) real-valued pair in the space of pixel indices. There are 15 keypoints, which represent the following elements of the face:

left_eye_center, right_eye_center, left_eye_inner_corner, left_eye_outer_corner, right_eye_inner_corner, right_eye_outer_corner, left_eyebrow_inner_end,
left_eyebrow_outer_end, right_eyebrow_inner_end, right_eyebrow_outer_end, nose_tip,
mouth_left_corner, mouth_right_corner, mouth_center_top_lip, mouth_center_bottom_lip

Left and right here refers to the point of view of the subject.

In some examples, some of the target keypoint positions are misssing (encoded as missing entries in the csv, i.e., with nothing between two commas).

The input image is given in the last field of the data files, and consists of a list of pixels (ordered by row), as integers in (0,255). The images are 96x96 pixels.

Data files can be downloaded from the project websites or Kaggle websites:

training.csv: list of training 7049 images. Each row contains the (x,y) coordinates for 15 keypoints, and image data as row-ordered list of pixels.

test.csv: list of 1783 test images. Each row contains ImageId and image data as row-ordered list of pixels

submission FileFormat.csv: list of 27124 keypoints to predict. Each row contains a RowId, ImageId, FeatureName, Location. FeatureName are "left_eye_center_x," "right_eyebrow_outer_end_y," etc. Location is what you need to predict.

2.1.3 Submission and Evaluation

For the paper writing purposes, you may split the training set into 90% and 10% respectively. Since we donot have the ground-truth for test.csv, the 10% held-out data can be served as the testing data to evaluate your own algorithms. You may also want to make a submission to Kaggle website for this dataset.

2.2 Understanding Unstructured Unstructured Social Activity

We have a dataset for understanding event/activity happened in the video. The dataset is well described and can be downloaded in `http://yanweifu.github.io/USAA/download/`. The low-level features for each video have been extracted.

You may want to read our papers

> [1] Attribute Learning for Understanding Unstructured Social Activity Y. Fu, T. Hospedales, T. Xiang and S. Gong, ECCV 2012;
>
> [2] Learning to detect unseen object classes by between-class attribute transfer, C. H. Lampert, H. Nickisch, and S. Harmeling, CVPR 2009.

2.2.1 Minimum requirements

For this project, you need to compare both in supervised learning and zero-shot learning scenarios.

For supervised learning, we have pre-defined split and the testing data also come with the ground-truth. You can just try it.

As for zero-shot learning (ZSL), we use three splits for ZSL, and the zero-shot (testing classes) are [1,2,4,7], [1,6,7,8],[2,4,5,6]. For the details of the idea of ZSL, please also refer to [2].

2.3 Large-scale video classification

Our group has one big dataset for video understanding. The dataset can be downloaded from `http://bigvid.fudan.edu.cn/data/fcvid/`. Note that low-level features have been extracted and provided in the link.

The whole dataset is well described in `http://bigvid.fudan.edu.cn/FCVID/`. For this dataset, please read

> [3] Exploiting Feature and Class Relationships in Video Categorization with Regularized Deep Neural Networks, Yu-Gang Jiang, Zuxuan Wu, Jun Wang, Xiangyang Xue, Shih-Fu Chang. 2015

2.3.1 Minimum requirements

For this task, one can only finish the task of supervised learning for all 239 classes. The minimum requirements include:

1. randomly sampling the training instances of each video classes from few training instances to large number of training instances;
2. comparing different types of features; and discuss the differences and complementarity of each type of features.
3. exploring the relationships of all classes as Sec.4.2.2 in [3].

2.4 Other projects.

You can also try other projects listed in `https://www.kaggle.com/datasets`. However, please let TA and me know first in order to get the approval. Please note that, once you decide which one to choose, you need to download the dataset as soon as possible, in case of the competition run out of time.

The preferred lists include,

1. Alzheimer's MRI deep learning: `https://www.kaggle.com/jesseab/alzheimers-mri-deep-learning`.
2. Visual Question Answering:`http://www.visualqa.org`.