

# UCSD COGS 118A: Winter 2018

Search this site

Instructor: Zhuowen Tu

Introduction to machine learning I:

Syllabus

Calendar and Class Notes

Homework Assignments

Reading List

Class Announcements

Contact Us

Useful Links

Study Section

[Homework Assignments](#) >

## Final Project, due 11:59pm, March 23, 2018

posted Feb 6, 2018, 9:22 AM by Zhuowen Tu [ updated Mar 20, 2018, 1:42 PM ]

Report format: Write a report with >1,000 words (excluding references) including main sections: a) abstract, b) introduction, c) method, d) experiment, e) conclusion, and f) references. You can follow the paper format as e.g. leading machine learning journals such as Journal of Machine Learning Research (<http://www.jmlr.org/>) or IEEE Trans. on Pattern Analysis and Machine Intelligence (<http://www.computer.org/web/tpami>), or leading conferences like NIPS (<https://papers.nips.cc/>) and ICML ([http://icml.cc/2016/?page\\_id=151](http://icml.cc/2016/?page_id=151)).

Bonus points: If you feel that your work deserves bonus points due to reasons such as: a) novel ideas and applications, b) large efforts in your own data collection/preparation, c) state-of-the-art classification results, or d) new algorithms, please create a "Bonus Points" section to specifically describe why you deserve bonus points.

Note that requirement for the word count (>1,000) only applies to a single-student project. For team-based projects, each team only needs to write one final report but the role of each team member needs to be clearly defined and specified.

Word count:

One-person team: >1,000 (excluding references)

Two-persons team: > 1,800 (excluding references)

### Option 1 (no team work allowed):

In this project you will choose any three classifiers out of those tested in [Caruana and Niculescu-Mizil](#) on three datasets from the UCI repository <http://archive.ics.uci.edu/ml/>. **Note that the same classifier type with different kernels (e.g. SVM using linear or RBF; boosting using decision stump or decision tree), are NOT considered as different classifiers.** Please read the paper by Caruana and Niculescu-Mizil carefully. In your experiments, for each classifier, you will train and test it on at least three datasets. Therefore, there are minimum a total of  $3 \times 3 = 9$  individual training and testing. Each time, you will need to do cross-validation to find your proper hyper-parameters corresponding to the type of classifier being used.

Train your classifiers using the setting (not all metrics are needed) described in the empirical study by [Caruana and Niculescu-Mizil](#). You are supposed to reproduce consistent results as in the paper. However, do expect some small variations. When evaluating the algorithms, you don't need to use all the metrics that were reported in the paper. Using one metric, e.g. the classification accuracy, is sufficient. Please report the cross-validated classification results with the corresponding learned hyper-parameters.

You can alternatively or additionally adopt the datasets and classifiers reported in a follow-up paper, [Caruana et al. ICML 2008](#).

You are encouraged to use Python, but using other programming languages and platforms is ok. The candidate classifiers include:

1. Boosting family classifiers

<http://www.mathworks.com/matlabcentral/fileexchange/21317-adaboost>

or

<https://github.com/dmlc/xgboost>

2. Support vector machines

<http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

3. Random Forests

<http://www.stat.berkeley.edu/~breiman/RandomForests/>

4. Decision Tree

<http://www.rulequest.com/Personal/> (please see also see a sample matlab code in the attachment)

5. K-nearest neighbors

<http://www.mathworks.com/matlabcentral/fileexchange/19345-efficient-k-nearest-neighbor-searchusing-jit>

6. Neural Nets

<http://www.cs.colostate.edu/~anderson/code/>

<http://www.mathworks.com/products/neural-network/code-examples.html>

7. Logistic regression classifier

8. Bagging family

The links above are for your reference. You can implement your own classifier or download other versions you like online (But you need to make sure the code online is reliable). You are supposed to write a formal report describing about the experiments you run and the corresponding results (plus code).

### Grading

**Note that if you do well by satisfying the minimum requirement e.g. 3 classifiers on 3 datasets with cross-validation, you will receive a decent score but not the full 100 points. We are looking for something a bit more and please see the guideline below.**

When reporting the experimental results, there are two main sets of comparisons we are looking for:

- For each dataset on each partition, show the comparison for different algorithms, and hopefully be consistent with the findings in the paper with Random Forests being the best etc.
- For each classifier on each partition, show the comparison on different partitions and you are supposed to show the increase of test accuracy (decrease of test error) with more training data and less test data.


The merit and grading of your project can be judged from aspects described below that are common when reviewing a paper:

1. How challenging and large are the datasets you are studying? (10 points)
2. Any aspects that are new in terms of algorithm development, uniqueness of the data, or new applications? (10 points)
3. Is your experimental design comprehensive? Have you done thoroughly experiments in tuning hyper-parameters and performing cross validation (you should also try different data partitions, e.g 20% training and 80% testing, 50% training and 50% testing, and 80% training and 20% testing for multiple rounds, **e.g. 3 times each for the above three partitions and compute average scores to remove potentials of having accidental results**); try to report both the training and testing errors after cross-validation; it is encouraged to also report the training and validation errors during cross-validation using classification error/accuracy curves w.r.t. the hyper-parameters. (50 points)
4. Is your report written in a professional way with sections including abstract, introduction, data and problem description, method description, experiments, conclusion, and references? (30 points)
5. Bonus points will be assigned to projects in which new ideas have been developed and implemented, or thorough experiments where extensive empirical studies have been carried out (e.g. evaluated on  $\geq 5$  classifiers and  $\geq 4$  datasets).

#### Option 2:

You are also welcome to work on a project in which supervised learning techniques are utilized. However, the level of difficulty in carrying out the project should at least on par with that of option 1. Please discuss your project idea with the instructor and get a consent first. If you are teaming up to work on challenging projects, please discuss it with the instructor too.



 UCI-how-to-load-data (with example dataset).zip ... Weijian Xu, Mar 21, 2018, 1:54 PM

v.1



#### Comments

You do not have permission to add comments.