6.867 Final Project
Fall 2016

Important Dates:

- Proposal due: November 15, 11:59PM (via CMT)
- Proposal discussions: Week of November 21 (there will be a web sign-up for times)
- Project report due: December 13, 11:59PM (via CMT)
- Once you hand in the project report, you're done.

Process

The final project in 6.867 will consist of the following components: proposal and proposal discussion, research project, and project report.

The proposal (one per team) will be a written document, 1–2 pages long, outlining the work to be done. It should include a plan with at least 4 steps, and indicate your internal deadlines for each of those steps. If there are multiple participants, the division of responsibility should be made clear. In addition, please include your assessment of what the "risks" to the project are: that is, what things do you think might turn out to be more difficult than planned, and what thoughts do you have about how to mitigate the risks?

If you are going to do an empirical study, be sure that you think about what method to use as a "baseline". It might be running a simple off-the-shelf algorithm or comparing to what happens if you predict the most common class.

Remember that almost anything will turn out to be harder and more time-consuming than you expect. Try to arrange your project so that there are intermediate milestones that can serve as alternative finishing points, in case you don't get to the end. It will be much better to turn in a polished version of a small-scale project than to find yourself at the end of the term with a three-quarters implemented system of great depth and scope.

A proposal interview (one per team) of about 10 minutes will be scheduled for all teams. This is your chance to get feedback on your proposal, with some ideas about how to structure your experiments, etc.

The project report (one per team) will be a written document of about $4n$ pages in double-column conference format, where n is the number of people in your group, including whatever graphs and tables are necessary to make your point. The report is the means by which you communicate the process and results of your project, so it should be clear, coherent, and well written. Do not dump out large quantities of data or code or uninterpreted charts. Emulate the expositional style of a technical conference paper. But, you do not need an abstract or a detailed related work section. But, be sure to cite and very quickly explain any technical work you referenced in formulating and carrying out your project.

The main goals are to make clear what your findings are, why you think they came out the way they did, and why that might be important and to be precise enough to allow someone to replicate your experiments (or verify your proofs).

Projects

You have 4 weeks to do this project, and we expect it to take about 6 person-hours per week; so that's 25 person-hours for a single-person project and 50 person-hours for a two-person project, etc. You'll have to make a plan and stick to it, to avoid getting behind and doing a bad rush job at the end.

Here are some ideas for types of projects. Replicating an experiment is probably the best option if you don't have a concrete idea of something different to do.

Apply a technique Take one or more of the methods that we have talked about in class, or that we are about to cover, and apply them to a problem. Compare their performance and elucidate why they perform differently, if they do. Do they do a good job on the problem?

This is most interesting if you can apply it to some other research question or problem you know about. A big issue here is being sure that you can get the data you need.

You don't necessarily have to implement all (or even any) of the algorithms you use. There are several toolkits available with many learning algorithms already implemented in them. However, if you don't do any implementation yourself, we would expect something much deeper in the way of problem formulation or modeling.

If you decide to implement a numerical algorithm, keep in mind that there may be numerical problems such as you've experienced in the homework: for instance, problems may be ill-conditioned or products of probabilities may go to zero (necessitating the use of logs for intermediate values). We can help you with these sorts of problems during office hours.

There are repositories of data available; links are on the Project Resources page (the link is on the Piazza Course Page, under Resources). **Running existing implementations of algorithms on standard data sets is a bare minimum project which cannot earn more than a C.**

Replicate an experiment Often, the best way to understand something is to replicate an experiment reported in the literature.

There are huge numbers of papers in these journals and conferences, some of which are good, some bad, some hard, some easy. If you're going to pick your own paper, please post a private question on Piazza before you hand in your project proposal.

The project resources page (see the end of this handout) has a list of papers you might to try to replicate. It's a kind of arbitrary selection, and in no way exhaustive. Many of them are journal articles, and maybe too big to replicate all of; if so, pick a subpart. If one of these topics interests you, check the references in the paper; maybe you'll find a precursor paper that is a better starting point.

Note that replicating someone else's results is notoriously difficult. There are often a lot of things left unsaid in technical papers, which have a real effect on the outcome. Part of the value of trying to do this is learning how to be clearer and more complete in future papers you may write.

Something else If you are more theoretically inclined, and have an idea for a theoretical direction to pursue, or want to do something else different, we're certainly open to the idea. The proposal / discussion process will be very important in this event.

Collaboration

Make completely clear in your paper which software you wrote and which software you used but did not write.

You may do your project in groups of up to 3 (if you have a very strong proposal and a good reason, we would consider groups of larger size). If you work in a group, you must:

- Make clear before you start what the division of labor will be.
- Make clear in the written report what the division of labor actually was (it's fine if it deviates from the proposal, but it must be specific and accurate).
- Be sure that all participants understand all of the work.

Projects done by n people will be expected to have n times as much technical depth and content as those done by a single person. For joint projects, the written work may be done jointly.

Be sure to cite all papers and web sites consulted during the course of your project, as well as to acknowledge other students or TAs who helped you substantially. Citations do not count against your page limit.

Relationship to other classes You may use a single project for 6.867 and another class that you are taking concurrently, or a graduate or undergraduate research project. If it is one project for two classes you must: (1) produce a project that is twice as large in depth and content as would have been required for either class individually; (2) obtain permission from the instructor of the other class; and (3) make clear to us what other class this project is being used for. If it is for your graduate or undergraduate research, make the context clear, and delineate the part of the overall project that is to be considered your project for this class.

Grading

The grading will be broken down as follows:

- proposal + meeting(10%)
- research project : technical content (75%)
- project report : presentation, writing, clarity (15%)

Resources

Some Data repositories

- `http://www.ics.uci.edu/~mlearn/MLRepository.html` The UC Irvine repository
- `http://yann.lecun.com/exdb/mnist/` Character recognition

- http://host.robots.ox.ac.uk/pascal/VOC/ The PASCAL Visual Object Classification Challenge
- http://www.cs.cmu.edu/~enron/ Enron: A Dataset for Email Classification
- https://archive.org/download/stackexchange Stackexchange data (heres a small project someone built on this data: http://p.migdal.pl/tagoverflow/?site=stackoverflow&size=16)

Machine learning algorithm libraries

- http://www.shogun-toolbox.org/ Shogun large-scale machine learning matlab toolbox
- http://www.mathworks.com/products/statistics/ Matlab statistics toolbox
- http://scikit-learn.sourceforge.net/stable/ scikit.learn: Python machine learning modules
- http://mc-stan.org/ For groups interested in doing a Bayesian modeling project.

Journals and conferences Journal papers are usually easier to read, because they are longer and have better exposition and motivation.

- http://www.jmlr.org Journal of Machine Learning]
- http://www.springerlink.com.libproxy.mit.edu/content/100309/ Machine Learning Journal (URL provided is for MIT access only)
- http://books.nips.cc NIPS Conferences
- http://www.machinelearning.org/icml.html International Conference on Machine Learning

Possible papers to read and replicate Of the papers below, the ones that someone on the staff has looked at recently are marked with '*'. The others are from previous versions of the course.

Supervised Classification

- (*) Ali Rahimi and Ben Recht, Random Features for Large-Scale Kernel Machines NIPS 2007
- (*) Fastfood: Approximate Kernel Expansions in Loglinear Time, Le, Sarlos, Smola (2014) `https://arxiv.org/abs/1408.3060` (matlab code = `https://www.mathworks.com/matlabcentral/fileexchange/49142-fastfood-kernel-expansions`)
- (*) Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
- (*) Yoshua Bengio, Pascal Lamblin, Dan Popovici, Hugo Larochelle. Greedy layer-wise training of deep networks. in NIPS, 2007.
- (*) Ha, Dai, Le. "HyperNetworks", arXiv preprint arXiv:1609.09106 (2016).
- (*) Graves, Alex. "Generating sequences with recurrent neural networks." arXiv preprint arXiv:1308.0850 (2013). (A really clear, introductory paper on training a character RNN to generate text).
- (*) Sashank Reddi, Ahmed Hefny, Suvrit Sra, Barnabas Poczos, Alexander J. Smola. Stochastic variance reduction for nonconvex optimization. International Conference on Machine Learning (ICML 2016)
- Fei Sha and Lawrence K. Saul, "Large Margin Hidden Markov Models". Advances of Neural Information and Processing System, 2006.
- Ryan Rifkin and Aldebaro Klautau, "In Defense of One-vs-All Classification". Journal of Machine Learning Research, Volume 5 (Jan): 101-141, 2004.
- Ben Taskar, and Carlos Guestrin and Daphne Koller, "Max-Margin Markov Networks". Advances of Neural Information and Processing System, 2003.
- Andrew Y. Ng and Michael I. Jordan, "On Discriminative vs. Generative Classifiers: A Comparison of Logistic Regression and Naive Bayes". Advances of Neural Information and Processing System, 2001.
- Yoav Freund and Robert E. Schapire, "Large Margin Classification Using the Perceptron Algorithm". Machine Learning, Volume 37, Issue 3, 1999.
- Robert E. Schapire, Yoav Freund, Peter Bartlett, and Wee Sun Lee, "Boosting the Margin: A New Explanation for the Effectiveness of Voting Methods". The Annals of Statistics, Volume 26, Issue 5, 1998.
- Thorsten Joachims, "Text Categorization with Suport Vector Machines: Learning with Many Relevant Features". Proceedings of the European Conference on Machine Learning, 1998.

- Rich Caruana, ”Multitask Learning”. Machine Learning, Volume 28, 41-75, 1997.
- Nick Littlestone, ”Learning Quickly when Irrelevant Attributes Abound: A New Linear-Threshold Algorithm”. Machine Learning, Volume 2, Issue 4, 1988. (And see later work on multiplicative update algorithms.)
- Yann LeCun, Leon Bottou, Yoshua Bengio, Patrick Haffner, Gradient-Based Learning Applied to Document Recognition, Proceedings of the IEEE, November 1998.

Semi-supervised Classification

- Andreas Argyriou, Mark Herbster, and Massimiliano Pontil, ”Combining Graph Laplacians for SemiSupervised Learning”. Advances of Neural Information and Processing Systems, 2005.
- Thorsten Joachims, ”Transductive Inference for Text Classification Using Support Vector Machines”. Proceedings of International Conference on Machine Learning, 1999.
- Avrim Blum and Tom Mitchell, ”Combining Labeled and Unlabeled data with Co-Training”. Proceedings of the 11th Annual Conference on Computational Learning Theory, 1998.

Unsupervised Learning

- (*) Zou, Hui, Trevor Hastie, and Robert Tibshirani. ”Sparse principal component analysis.” Journal of computational and graphical statistics 15.2 (2006): 265-286.
- (*) Ding, Chris HQ, Tao Li, and Michael I. Jordan. ”Convex and semi-nonnegative matrix factorizations.” IEEE transactions on pattern analysis and machine intelligence 32.1 (2010): 45-55.
- Lawrence K. Saul and Sam T. Roweis, ”Think Globally, Fit Locally: Unsupervised Learning of Low-Dimensional Manifolds”, Journal of Machine Learning Research, Volume 4, 119-155, 2003.
- Michael E. Tipping and Christopher M. Bishop, ”Probabilistic Principal Component Analysis”, Journal of Royal Statistics Society (B), Volume 61, Part 3, 611-622, 1999.
- L. Zelnik-Manor and P. Perona, “Self-tuning Spectral Clustering,” NIPS 2004

Reinforcement Learning

- (*) Volodymyr, et al. ”Playing Atari with deep reinforcement learning.” arXiv preprint arXiv:1312.5602 (2013). (Probably quite hard to repliate)

Graphical Models and Inference

- (*) O. Capp, A. Guillin, J.-M. Marin, and C. P. Robert, Population Monte Carlo, Journal of Computational and Graphical Statistics, vol. 13, no. 4, pp. 907929, 2004.

- (*) F. Lindsten, M. I. Jordan, and T. B. Schn, Particle Gibbs with Ancestor Sampling, Journal of Machine Learning Research, vol. 15, pp. 21452184, 2014.

- (*) T. Broderick, B. Kulis, and M. I. Jordan, MAD-Bayes: MAP-based Asymptotic Derivations from Bayes, presented at the International Conference on Machine Learning, 2013.

- (*) A. Korattikara, Y. Chen, and M. Welling, Austerity in MCMC Land: Cutting the Metropolis-Hastings Budget, presented at the International Conference on Machine Learning, 2014.

- Chong Wang, Bo Thiesson, Christopher Meek, and David Blei, ”Markov Topic Models”. Proceedings of the 12th International Conference on Artificial Intelligence and Statistics (AISTATS) 2009.

- David Blei and Jon McAuliffe, ”Supervised Topic Models”. Advances of Neural Information and Processing Systems, 2007.

- David M. Blei, Andrew Y. Ng, and Michael I. Jordan, ”Latent Dirichlet Allocation”. Journal of Machine Learning Research, Volume 3, 993-1022, 2003.

- Zhe Chen, ”Bayesian Filtering: From Kalman Filters to Particle Filters, and Beyond”. Statistics, 2003.

- Jonathan S. Yedidia and William T. Freeman, and Yair Weiss, ”Understanding Belief Propagation and Its Generalization”. Exploring Artificial Intelligence in the New Millennium, Chap. 8, pp. 239-236, 2003.

- Matthew J. Beal and Zoubin Ghahramani, ”The Variational Bayesian EM Algorithm for Incomplete Data: with Application to Scoring Graphical Model Structures”. Bayesian Statistics 7, 2003.

- John Lafferty, Andrew McCallum, and Fernando Pereira, ”Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data”. Proceedings of 18th International Conference on Machine Learning, 2001.

- C.K.I. Williams, ”Prediction with Gaussian Processes: From Linear Regression to Linear Prediction”. Learning in Graphical Models, pages 599-621. The MIT Press, 1999.

Model Selection

- Michael Kearns, Yishay Mansour, Andrew Y. Ng, and Dana Ron, ”An Experimental and Theoretical Comparison of Model Selection Methods”. Proceeding of 18th Annual Conference on Computational Learning Theory, 1995.

Others

- (*) Guy Shani, Davic Heckerman, and Ronen I. Brafman, ”An MDP-Based Recommender System”. Journal of Machine Learning Research, Volume 6, 1265-1295, 2005.

- (*) Thomas Gaertner, John Lloyd, and Peter Flach, ”Kernels and Distances for Structured Data”. Machine Learning, Volume 57 Issue 3, 2004. (You’d need to know something about logic to pursue this one)

- (*) Amir Beck and Marc Teboulle, ”A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems.” SIAM Imaging Science 2009

- (*) Duchi, John, Elad Hazan, and Yoram Singer, ”Adaptive Subgradient Methods for Online Learning and Stochastic Optimization.” JMLR 2011

- (*) Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge. ”A neural algorithm of artistic style.” arXiv preprint arXiv:1508.06576 (2015)

- (*) Goodfellow, Ian, et al. ”Generative adversarial nets.” Advances in Neural Information Processing Systems.

- (*) Kenji Kawaguchi. Deep Learning without Poor Local Minima. In Advances in Neural Information Processing (NIPS), 2016 (A theoretical analysis)

- Viola, Paul, and Michael Jones. ”Rapid object detection using a boosted cascade of simple features.” Computer Vision and Pattern Recognition, 2001.