Data-driven modeling
APAM E4990
Spring 2012

Final project

The main objective of the final project is for you to apply techniques we've discussed in class to real-world data. Simple techniques that work reasonably well and scale to practical problems are preferred to advanced techniques that show marginal improvement over baselines at the expense of interpretability, complexity, and/or computational cost. If you plan to use techniques beyond what we've covered in class, please check with me before doing so.

You may either use a pre-compiled data set or build your own data set—e.g. from APIs or public web sites[1] In either case, make sure to explicitly note details of how the data were collected and provide either a reference for obtaining the data or code to acquire it. Sources for possible data sets are listed below.

Your project should roughly follow the general paradigm we've discussed in class. After acquiring data, some sanity checks and visualization are useful to gain a basic understanding of the data. It is often necessary to clean or filter the data to deal with problematic observations—e.g., missing data, extreme outliers, etc. Next, specify the goal of the modeling problem—e.g., regression, classification, clustering, dimensionality reduction, recommendation system, etc.—along with the model(s) you're considering. In most cases this should include a loss function that quantifies model fit, along with an algorithm for optimizing this loss function. Clearly define measures which quantify performance—e.g. accuracy, confusion matrix, ROC, etc.—and evaluate these measures on both training and test data to assess fit and generalization. Be sure to address the issue of complexity control, as discussed in class.

As with past homeworks, your project should include both executable, well-commented code and a full report (as PostScript or PDF) that enables the reader to understand (read: reproduce) your results. There should be one main executable file, clearly indicated, that produces results, including figures. Your code should depend only on standard libraries and not assume the presence of special packages. If you modify the format of your data from the original in which it was collected or compiled, be sure to include all scripts necessary to do so (which should at some point be called by the main executable file). In addition to the issues mentioned above, discuss the practical aspects of your project, including the scalability and computational complexity of the storage and runtime for the methods used. While your report will not explicitly be graded on length, it should be several pages including figures and relevant citations (i.e., more than 2, probably less than 10).

[1] If you scrape data from a public website, make sure this is permitted and doesn't violate any terms of service.

Possible data sources

Sources for possible data sets include, but of course are not limited to:

- http://programmableweb.com
- http://developer.yahoo.com/yql/
- http://infochimps.org
- http://theinfo.org
- http://delicious.com/jhofman/data
- http://delicious.com/pskomoroch/dataset
- http://flowingdata.com/2009/10/01/30-resources-to-find-the-data-you-need/
- http://aws.amazon.com/publicdatasets/
- http://webscope.sandbox.yahoo.com/
- http://archive.ics.uci.edu/ml/
- http://networkdata.ics.uci.edu/
- http://netwiki.amath.unc.edu/SharedData/SharedData

Most of these pages contain links to other data sources and/or APIs, and are intended as pointers to cover a large set of references.