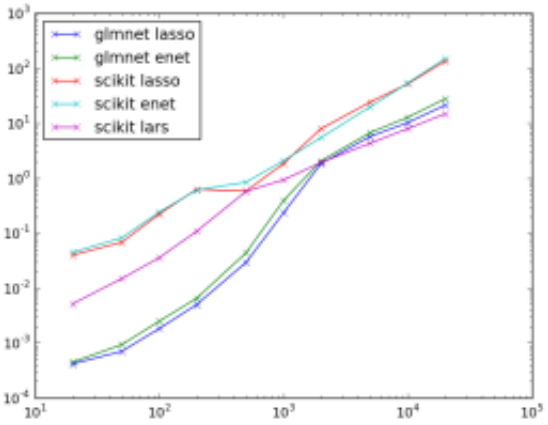
Explainaway's Blog

Just another WordPress.com weblog

Elastic net, LASSO, and LARS in Python

I'm currently looking for implementations of the LASSO (http://www-stat.stanford.edu/~tibs/lasso.html) and Elastic Net (http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.124.4696), otherwise known as L1 and L1/L2 regularised linear regression respectively, in Python. The options seem to be scikit.learn (http://scikit-learn.org/0.11/auto_examples/linear_model/plot_lasso_coordinate_descent_path.html) and glmnet-python (https://github.com/dwf/glmnet-python). The former offers coordinate ascent or the LARS algorithm coded in pure Python (with Numpy obviously), whereas the latter just wraps Jerome Friedman's Fortran code from the R glmnet package.



(https://explainaway.files.wordpress.com/2012/07/python-enet.png)
Runtime comparison between LASSO/Elastic net implementations from scikit.learn and glmnet-python. x-axis: number of features P. y-axis: time in seconds. Synthetic data with N=400, P/10 non-zero coefficients sampled from N(0,9), and 0.01 variance Gaussian noise.

As you might expect, the Fortran code is significantly faster in general, although for large P the LARS scikit.learn implementation is competitive with glmnet, presumably because the Python overhead becomes less noticeable. Unfortunately as far as I can see scikit.learn does not include a LARS implementation for the elastic net.

This entry was posted on July 26, 2012 at 9:26 pm and is filed under Machine learning, Statistics, Uncategorized. You can follow any responses to this entry through the RSS 2.0 feed. You can leave a response, or trackback from your own site.