A list of topics for a Google summer of code (GSOC) 2013

Disclaimer: This list of topics is currently being updated from last year's, and some information (like the names of possible mentors) is not definitive. Please e-mail the list with any questions.

Important: Expectations for prospective students

Also important: A letter from Gaël to last year's applicants. His suggestions are just as relevant this year.

Hi folks,

The deadline for applications is nearing. I'd like to stress that the scikit-learn will only be accepting high-quality application: it is a challenging, though rewarding, project to work with. To maximize the quality of your application, here are a few advice:

1. First discuss on the mailing list a pre-proposal. Make sure that both the scikit-learn team and yourself are entousiastic about the idea. Try to have one or two possible mentors that hold a dialog with you.

2. Satisfy the PSF requirements (http://wiki.python.org/moin/SummerOfCode/Expectations) briefly:

    - Demonstrate to your prospective mentor(s) that you are able to complete the project you've proposed
    - Blog for your GSoC project.
    - Contribute at least one patch to the project

I'd add the the patch should be somewhat substantial, not just fixing typos.

To contribute patch, please have a look at the [contribution guide] (http://scikit-learn.org/dev/developers/index.html#contributing-code) and the EasyFix issues in the tracker.

3. In parallel with 2, start a online document (google doc, for instance) to elaborate your final proposal, and if you manage to convince mentors, you can get feedback on it.

As a final note, I want to stress that GSOC projects are ambitious: we are talking about a few months of full time work. Thus the ideas proposed are idea challenging, and the students are supposed to draw a battle plan, with difficult variants and less difficult variants. The GSOC is a full major set of contributions, not a single pull request.

Good luck, I am looking forward to seeing the proposals. You'll see, the scikit is a big friendly and enthousiastic community,

Gaël

Add scipy.sparse matrix input support to the Decision Tree Implementation

Possible Mentor: ~~Andreas Mueller~~

Possible candidate:

Goal: make it possible to fit decision trees and randomized ensembles of trees on a scipy.sparse CSC data matrix.

Online Low Rank Matrix Completion

Possible mentor: Olivier Grisel, Vlad Niculae, Peter Prettenhofer (backup)

Possible candidate:

Goal: Online or Minibatch SGD or similar on a squared l2 reconstruction loss + low rank penalty (nuclear norm) on scipy.sparse matrix: the implicit components of the sparse input representation would be interpreted by the algorithms as missing values rather than zero values.

Application: Build a scalable recommender system example, e.g. on the movielens dataset.

TODO: find references in the literature. Matrix Factorization Jungle

Online Non Negative Matrix Factorization

Possible mentor: Olivier Grisel, Vlad Niculae

Possible candidate:

Goal: Online or Minibatch NMF using SGD + positive projections (or any other out-of-core algorithms) accepting both dense and sparse matrix as input (decomposition components can be dense array only).

Application: Build a scalable topic model e.g. on million of Wikipedia abstracts for instance using this script.

References:

- http://research.microsoft.com/apps/pubs/default.aspx?id=143211

Note: it is possible that we will combine the two Online Non Negative Matrix Factorization + Matrix Completion ideas in a single project. Please prospective students feel free to write proposals on one or the other or both ideas at the same time.

Robust PCA

Algorithms for decomposing a design matrix into a low rank + sparse components.

Possible mentor: ?

Possible candidate: Kerui Min (Minibio: "I'm a graduate student at UIUC who is currently pursuing the research work related to low-rank matrices recovery & Robust PCA.")

Applications: ?

References:

- http://perception.csl.uiuc.edu/matrix-rank/home.html
- http://www.icml-2011.org/papers/41_icmlpaper.pdf (randomized algorithm supposedly scalable to larg-ish datasets)

Generalized Additive Models

Possible mentor: Paolo Losi, Alex Gramfort, (others?)

Goal: Implement one of the state of art methods for Generalized Additive Models Sparse Version of it is SpAM

References:

- arxiv.org/pdf/0711.4555
- http://code.google.com/p/google-summer-of-code-2011-r/downloads/detail?name=Juemin_Yang.tar.gz
- http://en.wikipedia.org/wiki/Generalized_additive_model
- http://arxiv.org/abs/0806.4115
- http://www.stats.ox.ac.uk/~meinshau/liso.pdf

Coordinated descent in linear models beyond squared loss (eg Logistic)

Possible mentors: Alex Gramfort, Gael Varoquaux

Possible candidate:

Goal: Implement state of art methods for optimizing sparse linear models using coordinate descent.

One objective to avoid the dependency on LibLinear for the LogisticRegression model in order to allow warm restart and Elastic-Net regularization (L1 + L2)

A second objective is to improve the Lasso coordinate descent using strong rules to automatically discard features.

References:

- http://www.jmlr.org/papers/volume11/yuan10c/yuan10c.pdf
- http://www-stat.stanford.edu/~jbien/jrssb2011strong.pdf

Improve GMM

Possible mentors: Gael Varoquaux, Vlad Niculae

Possible candidate: Jim Holmström (Minibio: "Machine learning postgraduate student at KTH, Sweden with a bachelor in Engineering physics.")

- Refurbish the current GMM code to put it to the scikit's standards
- Implement a core-set strategy for GMM

http://las.ethz.ch/files/feldman11scalable-long.pdf http://videolectures.net/nips2011_faulkner_coresets/