This is the page for coordination of the GSoC for scikit-learn.

Scikit-learn is a machine learning module in Python. See http://scikit-learn.org for more details.

Scikit-learn is taking part of the GSoC trough the Python Software Foundation: http://wiki.python.org/moin/SummerOfCode

Instructions to student: achieving a good proposal

Difficulty: Scikit-learn is a technical project. Contributing via a GSoC requires a number of expertise in Python coding as well as numerical and machine learning algorithms.

Important: Read: Expectations for prospective students

Application template: https://wiki.python.org/moin/SummerOfCode/ApplicationTemplate2015 Please follow this template.

Also important: A letter from Gaël to former 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 enthusiastic 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 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 Easy 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 enthusiastic community,

Gaël

A list of topics for a Google summer of code (GSOC) 2015

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.

Improve GMM

Possible mentors: Andreas Mueller, Gael Varoquaux, Vlad Niculae

Possible candidate: Wei Xue (xuewei4d)

Application Link: https://github.com/scikit-learn/scikit-learn/wiki/GSoC-2015-Proposal:-Improve-GMM-module

- Reimplement VBGMM and DPGMM based on text-book derivations
- Refurbish current GMM implementation and testing, clean up the API.
- Implement a core-set strategy for GMM (optional)

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

Issue to get started : https://github.com/scikit-learn/scikit-learn/issues/4202

Metric learning

Possible mentor:

Possible candidate: Artem Sobolev (Barmaley-exe)

Proposal: Wiki Page Link

Goal: add some of metric learning algorithms (like NCA, ITML, LMNN) to be used with as transformers to facilitate distance-based methods (like KNN or some of clustering methods). Brian Kulis has a survey and a tutorial on metric learning, that seem to be a good place to start.

Improve the cross-decomposition module

Possible mentor: Michael Eickenberg (backup)

Possible candidate: lucapuggio

Goal: Improve documentation, stability and performance of the cross-decomposition module. Currently, the PLS and CCA modules only work in the `n_samples >= n_features` case, have very rudimentary documentation, and don't clearly explain their usage. They also fail many of the common test for numeric stability and API reasons. The API aspect is highly non trivial to resolve and will require high level reflection upon the scikit-learn API in general and the elaboration of a plan to render it consistent with the transformation of `Y`. The goal of the project is * to improve documentation and provide better usage examples * make the relation to other scikit-learn estimators clear * Improve API compatibility * Fix numeric and runtime issues * Add functionality for the case `n_samples < n_features` (i.e. linear kernel CCA/PLS and general kernel CCA/PLS - in accordance with the development of other kernelized methods such as `KernelRidge` and `GaussianProcessRegressor/Classifier`)

Global optimization based Hyperparameter optimization

Possible mentor: Andreas Mueller

Possible candidate: Christof Angermueller (cangermueller), Hamzeh Alsalhi (hamsal)

Proposal Link: https://github.com/scikit-learn/scikit-learn/wiki/GSoC-2015-Proposal:-Global-optimization-based-Hyper-parameter-optimization

Goal: The goal of this project is to implement a hyper-parameter optimization strategy based on global optimization, as alternatives to `GridSearchCV` and `RandomizedSearchCV` Two interesting approaches are SMAC using Random forests and Spearming, using Gaussian Processes. Both approaches build a model of the expected accuracy for a given parameter setting and trade off expected improvements vs uncertainty in the method, to intelligently search the space of hyperparameters.

Not that use of the GP is dependent on #4270.

References:

- SMAC http://www.cs.ubc.ca/labs/beta/Projects/SMAC/
- spearmint https://github.com/JasperSnoek/spearmint
- Hyperopt http://jaberg.github.io/hyperopt/

Multiple metric support for cross-validation and grid-searches

Mentor: Andreas Mueller, Joel Nothman, Olivier Grisel

Candidate: Raghav R V (raghavrv)

Proposal/Report Link: Wiki Page Link

Goal: Allow the simultaneous evaluation of multiple metric functions (like accuracy, AUC, recall) with a single run on cross-validation or Grid Search.

This project is quite heavy on API design, and will touch many places in scikit-learn. See #1850 and #2759.

Cross-validation and Meta-Estimators for semi-supervised learning

Possible mentor: Andreas Mueller

Possible candidate: Boyuan Deng (bryandeng), Vinayak Mehta (vortex-ape)

Link to proposal: Boyuan's proposal, Vinayak's proposal

Goal: Improve support for semi-supervised learning, implement a meta-estimator for self-taught learning.

The idea is to make semi-supervised learning easier in scikit-learn, by improving the support in cross-validation. Cross-validation objects need to be aware of which samples are labeled, and which are not, to produce meaningful test errors. There are several semi-supervised algorithms that would be good to implement, the easiest being a self-taught learning meta-estimator #1243.

Generalized Additive Models (GAMs)

Possible mentor: Alex Gramfort, Michael Eickenberg (backup)

Possible candidate:

Goal: Add the `additive` (or `additive_model`) directory and implement a few additive models. - [ ] Help finishing up the PR by jcrudy on including pyearth into scikit - [ ] Add Generalized Additive Model (GAM) - [ ] Add SpAM (Sparse Additive Model) - [ ] Add GAMLSS (GAM for Location Scale and Shape) - [ ] Add LISO (LASSO ISOtone for High Dimensional Additive Isotonic Regression)

References:

- GAM (Generalized Additive Model)
  - Tribshirani 86, 1335 Citations - CRAN gam - CRAN mgcv - CRAN bam - CRAN vgam - Wikipedia
- MARS (Multivariate Adaptive Regression Splines) :
  - Friedman 91, 4568 Citations - CRAN R Package
- SpAM
  - NIPS - Han Liu, Pradeep Ravikumar et al - 07, 238 Citations - JMLR Zhao 12 - Yahoo AAAI 15 - CRAN R Package (SAM) Released 2014

- GAMLSS (GAM for Location Scale and Shape)
    - Rigby, Stasinopoulos 05 - 500 odd Citations - Stasinopoulos 07 - Journal Article, 298 citations - CRAN GAMLSS 2014
- LISO (Lasso ISOtone for High Dimensional Additive Isotonic Regression)
    - Paper - CRAN R Package
- High Dimensional Additive Modelling - Lucas et al.
- SpAM implemented by Juemin Yang, in R, as a GSoC 2011 project

Possible mentors

Here are people that have said that they might be available for mentoring:

Gaël Varoquaux, Vlad Niculae, Olivier Grisel, Andreas Mueller, Alexandre Gramfort, Michael Eickenberg (backup for PLS or additive models), Joel Nothman (for the multiple metric support project).