

Open science software development



Tim Head

28 September 2017



Tim Head

✉ tim@wildtreotech.com

🐦 [@betatim](#)

Personal

Pragmatic

Problematic

Code for your latest paper

IMPROVING GAIA PARALLAX PRECISION WITH A DATA-DRIVEN MODEL OF STARS

LAUREN ANDERSON,¹ DAVID W. HOGG,^{1,2,3,4} BORIS LEISTEDT,^{2,5}
ADRIAN M. PRICE-WHELAN,⁶ AND JO BOVY^{1,7,8}

¹*Center for Computational Astrophysics, Flatiron Institute, 162 Fifth Ave, New York, NY 10010, USA*

²*Center for Cosmology and Particle Physics, Department of Physics, New York University, 726 Broadway, New York, NY 10003, USA*

³*Center for Data Science, New York University, 60 Fifth Ave, New York, NY 10011, USA*

⁴*Max-Planck-Institut für Astronomie, Königstuhl 17, D-69117 Heidelberg*

⁵*NASA Einstein Fellow*

⁶*Department of Astrophysical Sciences, Princeton University, 4 Ivy Lane, Princeton, NJ 08544, USA*

⁷*Department of Astronomy and Astrophysics, University of Toronto, 50 St. George Street, Toronto, ON M5S 3H4, Canada*

⁸*Alfred P. Sloan Fellow*

ABSTRACT

Converting a noisy parallax measurement into a posterior belief over distance requires inference with a prior. Usually this prior represents beliefs about the stellar density distribution of the Milky Way. However, multi-band photometry exists for a large fraction of the *Gaia* *TGAS* Catalog and is incredibly informative about stellar distances. Here we use *2MASS* colors for 1.4 million *TGAS* stars to build a noise-deconvolved

[Code](#) [Pull requests 0](#) [Projects 0](#) [Wiki](#) [Insights](#)

data-driven photometric parallaxes, built with Gaia and 2MASS

143 commits

1 branch

0 releases

4 contributors

Branch: master

[New pull request](#)[Create new file](#)[Upload files](#)[Find file](#)[Clone or download](#)

davidwhogg tweaks

Latest commit 4a8d72e 28 days ago

paper	arxiver	3 months ago
presentation	tweaks	28 days ago
toy	paper figure updates	4 months ago
README.md	readme	4 months ago
comparePrior.py	plot tweaks;	4 months ago
demo_plots.py	tweaks	4 months ago
drawEllipse.py	splitting directories	4 months ago
dustDiff.py	splitting directories	4 months ago
plots.py	last tweaks	4 months ago
stellarTwins.py	added stellar twins	4 months ago
testXD.py	paper figure updates	4 months ago
tightPosterior.py	tweaks	4 months ago

README.md

photoParallax

Capitalizing on the simplicity of stars for *good*.

authors

- Lauren Anderson (Flatiron)

A General-Purpose Counting Filter: Counting Quotient Filter

7 commits

1 branch

0 releases

1 contributor

BSD-3-Clause

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

prashantpandey Merge branch 'master' of github.com:splatlab/cqf

Latest commit 0f937c7 on Apr 12

LICENSE	Initial commit	8 months ago
Makefile	Added an alternate implementation of select on machine words to work ...	6 months ago
README.md	Update README.	6 months ago
gqf.c	Fixing a small assertion bug in the CQF.	6 months ago
gqf.h	Adding code for the CQF.	8 months ago
main.c	Adding code for the CQF.	8 months ago

README.md

cqf

A General-Purpose Counting Filter: Counting Quotient Filter (CQF)

Overview

The CQF supports approximate membership testing and counting the occurrences of items in a data set. This general-purpose AMQ is small and fast, has good locality of reference, scales out of RAM to SSD, and supports deletions, counting (even on skewed data sets), resizing, merging, and highly concurrent access.

API

- 'of insert(item, count)': insert an item to the filter

Why do it?

- Making source code available helps your ideas spread faster
- More people will understand what you did
- Sharing your code will raise your profile
- Basically free, no promise of support or further development

Recent uploads

August 30, 2017 (v2) Working paper Open Access

View

Introducing Parsl: A Python Parallel Scripting Library

Babuji, Yadu; Brizius, Alison; Chard, Kyle; Foster, Ian; Katz, Daniel S.; Wilde, Michael; Wozniak, Justin

Researchers frequently rely on large-scale and domain-specific workflows to conduct their science. These workflows may integrate a variety of independent software functions and external applications. However, developing and executing such workflows can be difficult, requiring complex...

Uploaded on September 15, 2017

1 more version(s) exist for this record

September 6, 2017 (v4) Software Open Access

View

geodynamics/pylith: PyLith v2.2.1

Brad Aagaard; Charles Williams; Matthew Knepley; Eric Heien

Added new examples. examples/3d/subduction: New suite of examples for a 3-D subduction zone. This intermediate level suite of examples illustrates a wide range of PyLith features for quasi-static simulations. examples/2d/subduction: Added quasi-static spontaneous rupture earthquake cycle...

Uploaded on September 6, 2017

3 more version(s) exist for this record

August 31, 2017 (v1) Figure Open Access

View

Fig. 1 in *Vanmanenia orcicampus* | a new species of loach from the Plain of Jars | Laos (Teleostei: Gastromyzontidae)

Maurice Kottelat

Zenodo now supports DOI versioning!



Read more about it, in our newest blog post.

Using GitHub?



Check out our GitHub integration. Software Preservation Made Simple!

Zenodo in a nutshell

- **Research. Shared.** — all research outputs from across all fields of research are welcome! Sciences and Humanities, really!
- **Citeable. Discoverable.** — uploads gets a Digital Object Identifier (DOI) to make them easily and uniquely citeable.
- **Communities** — create and curate your own community for a workshop, project, department, journal, into which you can accept or reject uploads. Your own complete digital repository!

generation	move and update conversion script	5 months ago
models	default to better path and only one file	5 months ago
.gitignore	add 3d viz + move notebooks to analysis/	5 months ago
LICENSE	Create LICENSE	3 months ago
README.md	Update README.md	a month ago

README.md

CaloGAN

Simulating 3D High Energy Particle Showers in Multi-Layer Electromagnetic Calorimeters with Generative Adversarial Networks.

This repository contains what you'll need to reproduce M. Paganini (@mickypaganini), L. de Oliveira (@lukedeo), B. Nachman (@bnachman), *CaloGAN: Simulating 3D High Energy Particle Showers in Multi-Layer Electromagnetic Calorimeters with Generative Adversarial Networks* [[arXiv:1705.02355](https://arxiv.org/abs/1705.02355)].

You are more than welcome to use the open data and open-source software provided here for any of your projects, but we kindly ask you that you please cite them using the DOIs provided below:

Asset	Location
Training Data (GEANT4 showers, \perp to center)	DOI 10.17632/pvn3xc3wy5.1
Source Code (this repo!)	DOI 10.5281/zenodo.584155

For any use of paper ideas and results, please cite

```
@article{paganini_calogan,
  author      = "Paganini, Michela and de Oliveira, Luke and Nachman, Benjamin",
  title       = "{CaloGAN: Simulating 3D High Energy Particle Showers in Multi-Layer Electromagnetic Calorimeters with Generative Adversarial Networks}",
  year        = "2017",
  eprint      = "1705.02355"
```

Living in an Ivory Basement

Stochastic thoughts on science, testing, and programming.

[misc](#)

[personal](#)

[python](#)

[science](#)

[teaching](#)

[testing](#)

[training](#)

Please destroy this software after publication. kthxbye.

tl;dr? A while back [I wrote that there are three uses of research software](#): replication, reproduction, and reuse. The world of computational science would be better off if people clearly delineated whether or not they wanted anyone else to reuse their software, and I think it's a massive mistake to expect that everyone's software should be reusable.

Fri 17 April 2015

By [C. Titus Brown](#)

In [science](#).

tags: [software sustainability](#)

A few months back, I reviewed a pretty exciting paper - one I will probably highlight on my blog, when it comes out. The paper outlined a fairly simple concept for comparing sequences and then used that to develop some new ultra-scalable functionality. The theory seemed novel, the computational results were pretty good, and I recommended acceptance (or minor revisions). This was in spite of the fact that the authors stated quite clearly that they had produced largely unusable software.

Other reviewers were not quite so forgiving, however -- one reviewer declined to review the paper until they could run the software on their own data.

***What does successful software
look like?***

Plan A - join someone else

How to find existing projects?

- Ask a friend
- Ask google
- Specialist websites like libraries.io

Ask a friend

How to find other projects?

The image shows two side-by-side screenshots of Google search results. The left screenshot shows the search for "bayesian optimization github" with approximately 160,000 results. The right screenshot shows the search for "optimize hyperparameters bayesian github" with approximately 61,700 results. Both screenshots display a list of search results, including links to GitHub repositories and scholarly articles.

Google bayesian optimization github Sign in

All Videos News Images Shopping More Settings Tools

About 160,000 results (0.53 seconds)

Scholarly articles for bayesian optimization github

Practical **bayesian optimization** of machine learning ... - Snoek - Cited by 939
... foundation for assessing **bayesian optimization** of ... - Eggenberger - Cited by 72
Scalable **bayesian optimization** using deep neural ... - Snoek - Cited by 92

GitHub - fmfN/BayesianOptimization: A Python implementation of ...
<https://github.com/fmfN/BayesianOptimization>
README.md. **Bayesian Optimization**. Pure Python implementation of bayesian global optimization with gaussian processes. pip install **bayesian-optimization**.

GitHub - SheffieldML/GPyOpt: Gaussian Process Optimization using GPy
<https://github.com/SheffieldML/GPyOpt>
Gaussian Process Optimization using GPy. ... @Misc(gpyopt2016, author = {The GPyOpt authors}, title = {{GPyOpt}: A **Bayesian Optimization** framework in ...

GitHub - rmcantin/bayesoPt: BayesOpt: A toolbox for bayesian ...
<https://github.com/rmcantin/bayesoPt>
bayesoPt - BayesOpt: A toolbox for **bayesian optimization**, experimental design and stochastic bandit

GitHub - RuiShu/Neural-Net-Bayesian-Optimization: We use a ...
<https://github.com/RuiShu/Neural-Net-Bayesian-Optimization>
Neural-Net-Bayesian-Optimization - We use a modified neural network instead of Gaussian process for **Bayesian optimization**.

Google optimize hyperparameters bayesian github Sign in

All Videos News Images Shopping More Settings Tools

About 61,700 results (0.28 seconds)

GitHub - JasperSnoek/spearmint: Spearmint is a package to perform ...
<https://github.com/JasperSnoek/spearmint>
Spearmint is a package to perform **Bayesian optimization** according to the algorithms ... to GP **hyperparameter** samples and candidates it is **optimizing** over.

GitHub - RuiShu/Neural-Net-Bayesian-Optimization: We use a ...
<https://github.com/RuiShu/Neural-Net-Bayesian-Optimization>
A popular technique for **hyperparameter** tuning is **Bayesian optimization**, which canonically uses a Gaussian process to interpolate the **hyperparameter** space.

GitHub - fmfN/BayesianOptimization: A Python implementation of ...
<https://github.com/fmfN/BayesianOptimization>
BayesianOptimization - A Python implementation of global optimization with gaussian processes.

GitHub - HIPS/Spearmint: Spearmint Bayesian optimization codebase
<https://github.com/HIPS/Spearmint>
Spearmint **Bayesian optimization** codebase. Contribute to Spearmint development by creating an account on GitHub.

GitHub - Yelp/MOE: A global, black box optimization engine for real ...
<https://github.com/Yelp/MOE>
Inside, MOE uses **Bayesian** global optimization, which performs optimization using ... **Optimizing** the **hyperparameters** of the Gaussian Process (model selection) ...

Bayesian Optimization of Hyperparameters - GitHub
<https://github.com/yanyachen/rBayesianOptimization>

How to choose?

How to choose between all the options?

- Do I know someone else who is using it?
- Works well with what I already have?
- Can I modify it?
- Does it look “good”?
- Documentation and examples?
- How can I get help?
- Is the project still alive?

How to choose between the options?

RuiShu / Neural-Net-Bayesian-Optimization

Watch 8

Star 43

Fork 18

Code

Issues 0

Pull requests 1

Projects 0

Wiki

Insights

We use a modified neural network instead of Gaussian process for Bayesian optimization.

114 commits

2 branches

0 releases

1 contributor

MIT

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

RuiShu committed on GitHub Create LICENSE.md

Latest commit e98dc32 on Jun 29, 2016

data	Add trainer toggle	2 years ago
learning_objective	Cleaned up print statements	2 years ago
mpi	Cleaned up print statements	2 years ago
sequential	Cleaned up print statements	2 years ago
utilities	Cleaned up print statements	2 years ago
.gitignore	Cleaned up print statements	2 years ago
LICENSE.md	Create LICENSE.md	a year ago
README.md	Cleaned up print statements	2 years ago

How to choose between the options?

fmfn / BayesianOptimization

Watch 59

Star 820

Fork 232

Code

Issues 7

Pull requests 2

Projects 0

Wiki

Insights

A Python implementation of global optimization with gaussian processes.

optimization

gaussian-processes

bayesian-optimization

python

simple

127 commits

1 branch

1 release

6 contributors

MIT

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

fmfn committed on GitHub Merge pull request #47 from amanbh/master

Latest commit f243cfc on Mar 13

bayes_opt	fixes initialize function	7 months ago
examples	fixes alpha value in sklearn example and runs notebook till the end	7 months ago
tests	Fixes several bugs with example scripts and notebooks.	9 months ago
.gitignore	Small fixes and style changes	2 years ago
LICENSE	adds license and readme	3 years ago
README.md	updates installation guide and adds author email to setup	9 months ago

How to choose between the options?

README.md

Bayesian Optimization

Pure Python implementation of bayesian global optimization with gaussian processes.

```
pip install bayesian-optimization
```

This is a constrained global optimization package built upon bayesian inference and gaussian process, that attempts to find the maximum value of an unknown function in as few iterations as possible. This technique is particularly suited for optimization of high cost functions, situations where the balance between exploration and exploitation is important.

Quick Start

In the [examples](#) folder you can get a grip of how the method and this package work by:

- Checking out this [notebook](#) with a step by step visualization of how this method works.

How to choose between the options?

####Setting up Spearmint

STEP 1: Installation

1. Install [python](#), [numpy](#), [scipy](#), [pymongo](#). For academic users, the [anaconda](#) distribution is great. Use numpy 1.8 or higher. We use python 2.7.
2. Download/clone the spearmint code
3. Install the spearmint package using pip: `pip install -e \<path/to/spearmint/root\>` (the `-e` means changes will be reflected automatically)
4. Download and install MongoDB: <https://www.mongodb.org/>
5. Install the pymongo package using e.g., `pip pip install pymongo` or `anaconda conda install pymongo`

STEP 2: Setting up your experiment

1. Create a callable objective function. See `./examples/simple/branin.py` as an example
2. Create a config file. There are 3 example config files in the `./examples` directory. Note 1: There are more parameters that can be set in the config files than what is shown in the examples, but these parameters all have default values. Note 2: By default Spearmint assumes your function is noisy (non-deterministic). If it is noise-free, you should set this explicitly as in the `./examples/simple/config.json` file.

STEP 3: Running spearmint

1. Start up a MongoDB daemon instance:
`mongod --fork --logpath <path/to/logfile\> --dbpath <path/to/dbfolder\>`
2. Run spearmint: `python main.py \<path/to/experiment/directory\>`

STEP 4: Looking at your results

Spearmint will output results to standard out / standard err. You can also load the results from the database and manipulate them directly.

How to get unstuck?

How to get help?

- In general people love to help.
- Show that you tried to help yourself.
- People are volunteering their time, respect it.
- Grammar and spelling, srsly.
- Make a simple example <https://stackoverflow.com/help/mcve>

People like helping,

...

How to get help?

 **jupyter**
nbviewer

JUPYTER FAQ

GPYOpt / manual

GPYOpt: armed bandits optimization

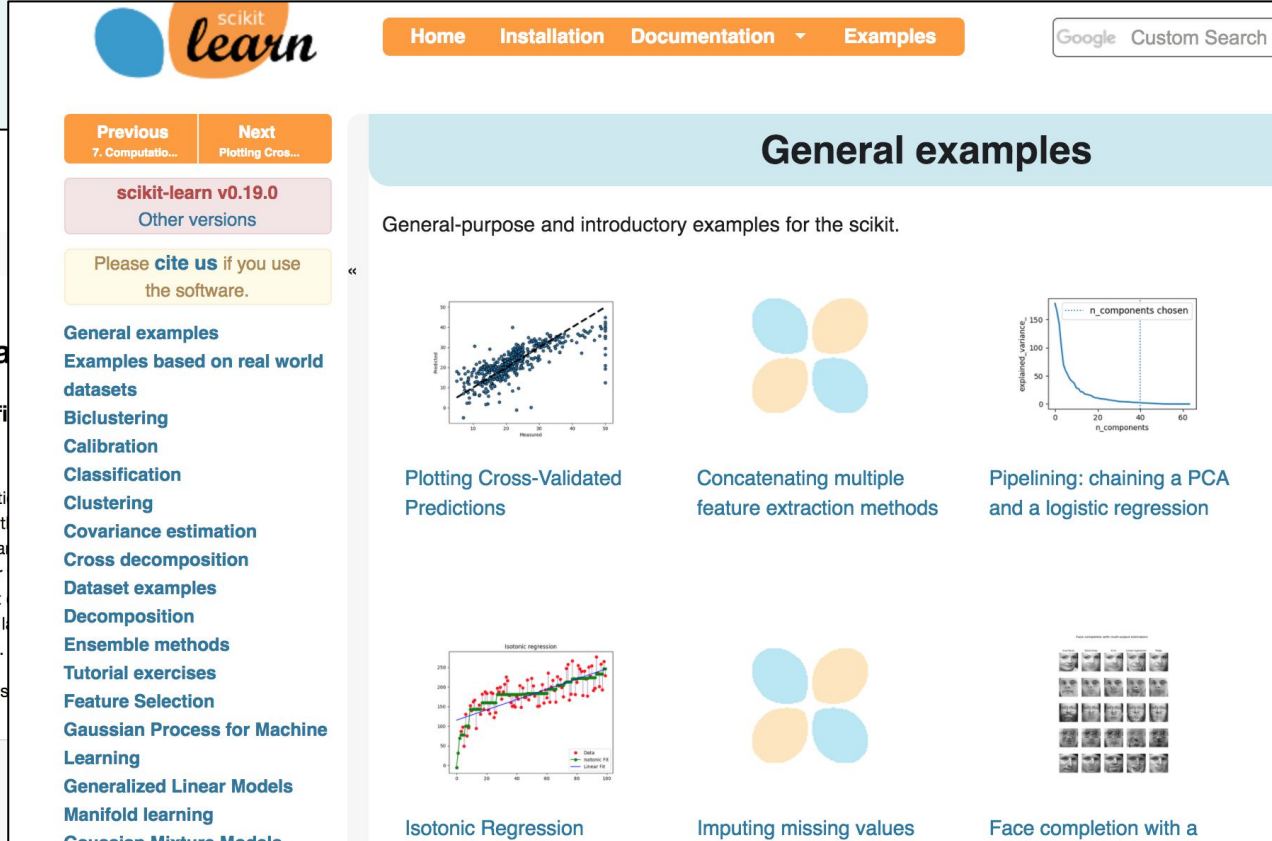
Written by **Javier Gonzalez**, University of Sheffield

Last updated Monday, 14 March 2016.

In this notebook we will see how to do armed bandits optimization using data of weather forecasts at weather stations across more than 3000 United States. The project [OpenWeatherMap project](#) provides a lot of information and at that [dataset](#) it is possible to find the weather data for each station in the United States for the April 22, 2014. The data for each station is available as well as the forecasts for the next 7 days.

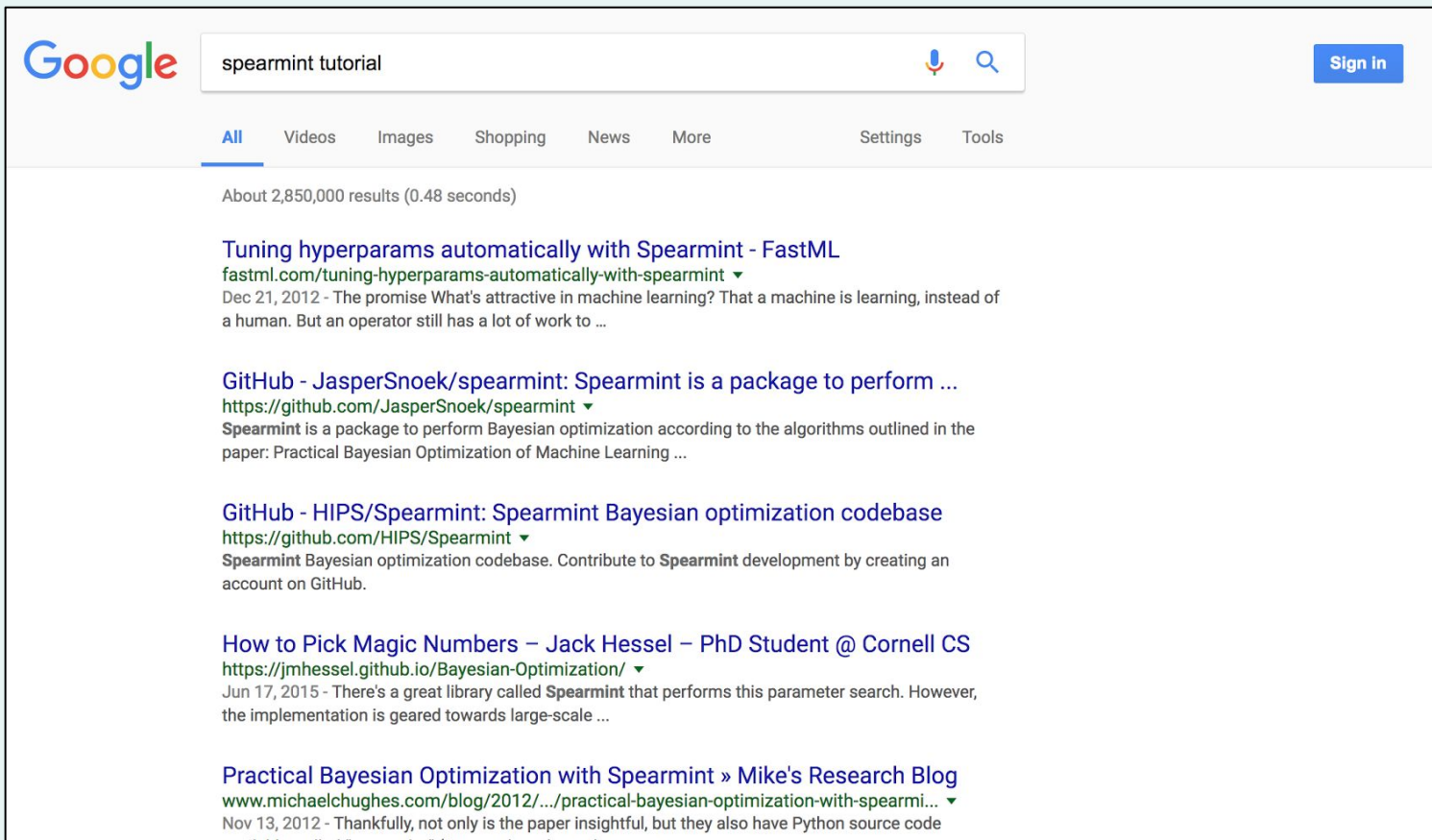
We start by loading the packages that we will need in our analysis:

```
In [1]: %matplotlib inline
import matplotlib as mpl
import matplotlib.pyplot as plt
import numpy as np
import sys
```



The screenshot shows the scikit-learn website's 'General examples' page. At the top, there is a navigation bar with 'Home', 'Installation', 'Documentation', and 'Examples'. A search bar is on the right. Below the navigation, there are buttons for 'Previous' (7. Computatio...) and 'Next' (Plotting Cros...). A central banner reads 'scikit-learn v0.19.0' with a link to 'Other versions'. A yellow box asks users to 'cite us' if they use the software. The main content area is titled 'General examples' and lists various topics: 'General examples', 'Examples based on real world datasets', 'Biclustering', 'Calibration', 'Classification', 'Clustering', 'Covariance estimation', 'Cross decomposition', 'Dataset examples', 'Decomposition', 'Ensemble methods', 'Tutorial exercises', 'Feature Selection', 'Gaussian Process for Machine Learning', 'Generalized Linear Models', 'Manifold learning', and 'Causation Mixture Models'. The 'General examples' section is expanded, showing a grid of example plots and their titles: 'Plotting Cross-Validated Predictions' (with a scatter plot), 'Concatenating multiple feature extraction methods' (with a 2x2 grid of circles), 'Pipelining: chaining a PCA and a logistic regression' (with a plot of explained variance vs n_components), 'Isotonic Regression' (with a plot of data points and a fitted line), 'Imputing missing values' (with a 2x2 grid of circles), and 'Face completion with a' (with a grid of face images).

How to get help?



The image shows a Google search interface with the query "spearmint tutorial". The search results are displayed below the navigation bar, which includes tabs for "All", "Videos", "Images", "Shopping", "News", "More", "Settings", and "Tools". The search results are sorted by relevance, showing about 2,850,000 results in 0.48 seconds. The top results are:

- Tuning hyperparams automatically with Spearmint - FastML**
fastml.com/tuning-hyperparams-automatically-with-spearmint
Dec 21, 2012 - The promise What's attractive in machine learning? That a machine is learning, instead of a human. But an operator still has a lot of work to ...
- GitHub - JasperSnoek/spearmint: Spearmint is a package to perform ...**
<https://github.com/JasperSnoek/spearmint>
Spearmint is a package to perform Bayesian optimization according to the algorithms outlined in the paper: Practical Bayesian Optimization of Machine Learning ...
- GitHub - HIPS/Spearmint: Spearmint Bayesian optimization codebase**
<https://github.com/HIPS/Spearmint>
Spearmint Bayesian optimization codebase. Contribute to Spearmint development by creating an account on GitHub.
- How to Pick Magic Numbers – Jack Hessel – PhD Student @ Cornell CS**
<https://jmhessel.github.io/Bayesian-Optimization/>
Jun 17, 2015 - There's a great library called Spearmint that performs this parameter search. However, the implementation is geared towards large-scale ...
- Practical Bayesian Optimization with Spearmint » Mike's Research Blog**
www.michaelchughes.com/blog/2012/.../practical-bayesian-optimization-with-spearmi...
Nov 13, 2012 - Thankfully, not only is the paper insightful, but they also have Python source code

***Don't ask for help when
you are angry.***

Grammur and Speeling. Srsly.

Contribute back?

How to contribute?

- Learn the mechanics of working in a team
- Make your changes official so you don't have to keep modifying code
- Get reputation and credit for your work
- Mentorship from (world) experts on the topic
- Meet interesting people

I am too busy ...



This repository

Search

Pull requests

Issues

Marketplace

Explore



scikit-learn / scikit-learn

Watch

1,836

Star

21,389

Fork

11,465

Code

Issues 901

Pull requests 578

Projects 5

Wiki

Insights

[MRG+1] Document oob estimates of RandomForest* and Bagging* #4489

Edit

Merged

glouppe merged 2 commits into scikit-learn:master from betatim:rf-oob-docs on Apr 2, 2015

Conversation 4

Commits 2

Files changed 1

+6 -1



betatim commented on Apr 2, 2015

Contributor



Reviewers

No reviews

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Notifications

Added a sentence each to the RandomForest* and Bagging* sections of the narrative documentation mentioning how to enable out-of-bag estimates of the generalisation error. This is meant to fix #4290

betatim added some commits on Apr 2, 2015

Mention oob_score in narrative documentation of RandomForest*

fd3c630

Mention oob_score in the narrative docs for Bagging*

88811be

glouppe changed the title from Document oob estimates of RandomForest* and Bagging* to [MRG+1] Document oob estimates of RandomForest* and Bagging* on Apr 2, 2015



glouppe commented on Apr 2, 2015

Owner





[MRG+1] Document oob estimates of RandomForest* and Bagging* #4489

Edit

Merged gloupe merged 2 commits into scikit-learn:master from betatim:rf-oob-docs on Apr 2, 2015

Conversation 4 Commits 2 Files changed 1

Changes from all commits Jump to... +6 -1

Unified Split Review changes

7 doc/modules/ensemble.rst

No coverage <> View

@@ -65,7 +65,9 @@ taking as input a user-specified base estimator along with parameters

65 specifying the strategy to draw random subsets. In particular, ``max_samples``
66 and ``max_features`` control the size of the subsets (in terms of samples and
67 features), while ``bootstrap`` and ``bootstrap_features`` control whether

68 -samples and features are drawn with or without replacement. As an example, the

68 +samples and features are drawn with or without replacement. When using a subset

69 +of the available samples the generalization error can be estimated with the

70 +out-of-bag samples by setting ``oob_score=True``. As an example, the

69 71 snippet below illustrates how to instantiate a bagging ensemble of

70 72 :class:`KNeighborsClassifier` base estimators, each built on random subsets of

[MRG + 2] Rename scorers like `mse` to `neg_mse` #7261

Edit

Merged GaelVaroquaux merged 15 commits into scikit-learn:master from betatim:negative-scorers on Sep 8, 2016

Conversation 99

Commits 15 Files changed 9

+112 -45



betatim commented on Aug 27, 2016 • edited

Contributor + 😊 ✎

Reference Issue

Fixes [#2439](#)

What does this implement/fix? Explain your changes.

Renaming scorers for which smaller is better (like MSE) to `neg_mse` so that they fit the idea of "bigger is better".

Reviewers
No reviews

Assignees
No one assigned

Labels
Blocker

***Creating software is a craft,
be an apprentice.***

nilearn - examples are reproducible science

http://nilearn.github.io/auto_examples/02_decoding/plot_miyawaki_reconstruction.html

Plan B - write your own

Choices and preliminary things

- Make it extremely focussed, one small part of your PhD thesis, that lots of people want
- What is the mission of your project?
- Persuade a friend to join you

scikit-optimize:

- ***easy to install***
- ***good documentation***
- ***expensive, black-box functions***

Manoj, Gilles and Tim

Getting started, from zero to one

***Keep it simple.
Very simple.***

***Simple
code base***

Quality is free

```
1 • import sys, random
2 • import os
3 • print('one'); print('two')
4
5 x = 3
6 y = 3.141
7 • if x == 1: print('one')
8
9     if abs(x) > 3 and y % 2 == 0:
10         print('woah!')
11 • def myfunc():
12     • rando = random.random()
13     • return random.randint(0,100)
14
15 • def multiply (x,y) :
16     • return x * y
17
18 print(multiply(myfunc(), myfunc()))
19
```

E231 - missing whitespace after ','

***Discuss code changes
with someone***

From one to ten

Get users!

How do others choose between all the options?

- Do I know someone else who is using it?
- Works well with what I already have?
- Can I modify it?
- Does it look “good”?
- Documentation and examples?
- How can I get help?
- Is the project still alive?

***How do others profit
when they contribute?***

How do others get help?

[Getting Started with Sphinx-Gallery](#)[Configuration](#)[Frequently Asked Questions](#)[Sphinx-Gallery Syntax](#)[Sphinx-Gallery Utilities](#)[Sphinx-Gallery API Reference](#)[Gallery of Examples](#)[Secondary gallery](#)[Change Log](#)[Fork sphinx-gallery on Github](#)

Seamless end-to-end tracing for Python

[Read the Docs](#)

v: latest ▾

[Read the Docs](#)

v: latest ▾

[Frequently Asked Questions](#)[Sphinx-Gallery Syntax](#)[Sphinx-Gallery Utilities](#)[Sphinx-Gallery API Reference](#)[Gallery of Examples](#)[Secondary gallery](#)[Change Log](#)[Fork sphinx-gallery on Github](#)

Welcome to Sphinx-Gallery's documentation!

A [Sphinx](#) extension that builds an HTML gallery of examples from any set of Python scripts.

It is extracted from the scikit-learn project and aims to be an independent general purpose extension.

The code of the project is on Github: [Sphinx-Gallery](#)

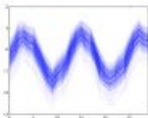
Why Sphinx-Gallery?

- Simple examples that run out of the box are the best way to learn a library
- Pleasing, organized, visual layouts
- Links, searching, backlinks throughout examples and documentation

What does it look like?

Here is an example gallery generated from a few Python scripts.

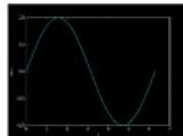
Examples using `numpy.sin`



Seaborn example



Plotting simple sin function



Plotting simple sin function on a black background

Here we put only the examples of our gallery that use a specific function. This display granularity

Automate things



About pytest

pytest is a mature full-featured Python testing tool that helps you write better programs.

Table Of Contents

- [Home](#)
- [Contents](#)
- [Install](#)
- [Examples](#)
- [Customize](#)
- [Contact](#)
- [Talks/Posts](#)
- [Changelog](#)
- [Backwards Compatibility](#)
- [License](#)

pytest: helps you write better programs

- [Features](#)
- [Documentation](#)
- [Bugs/Requests](#)

pytest: helps you write better programs

The `pytest` framework makes it easy to write small tests, yet scales to support complex functional testing for applications and libraries.

An example of a simple test:

```
# content of test_sample.py
def inc(x):
    return x + 1

def test_answer():
    assert inc(3) == 5
```

To execute it:


```
$ pytest
===== test session starts =====
platform linux -- Python 3.x.y, pytest-3.x.y, py-1.x.y, pluggy-0.x.y
rootdir: $REGENDOC_TMPDIR, inifile:
collected 1 item

test_sample.py F

===== FAILURES =====
_____ test_answer _____

    def test_answer():
>     assert inc(3) == 5
E       assert 4 == 5
E         + where 4 = inc(3)
```

```
test_sample.py:5: AssertionError
----- 1 failed in 0.12 seconds -----
```

 v: latest ▾

From ten to infinity

What if you left now?

Summary

Don't maintain code

Join an existing project

This is a long term investment

??!?

Tim Head

✉ tim@wildtreotech.com

🐦 @betatim

More reading

- <https://opensource.guide/>
- <https://mozilla.github.io/leadership-training/>
- Can I modify it?
- Does it look “good”?
- Documentation and examples?
- How can I get help?
- Is the project still alive?



Tim Head

✉ tim@wildtreotech.com

🐦 [@betatim](#)