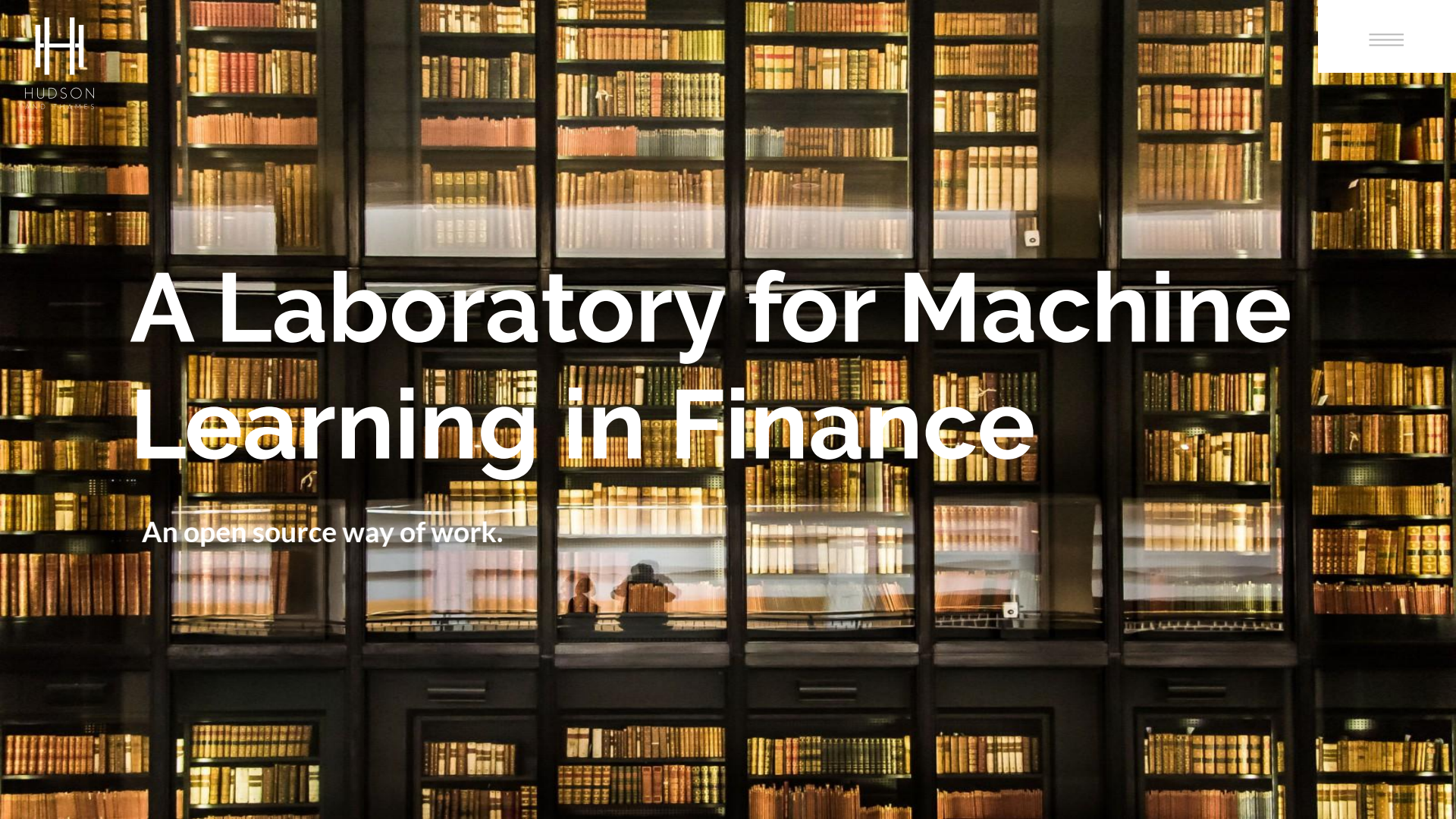


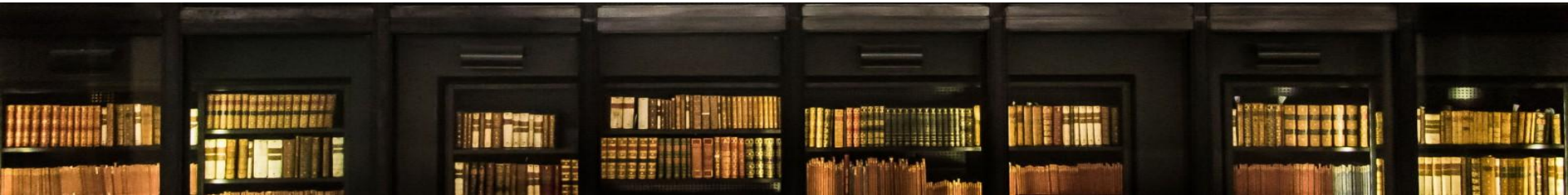
A Laboratory for Machine Learning in Finance

An open source way of work.

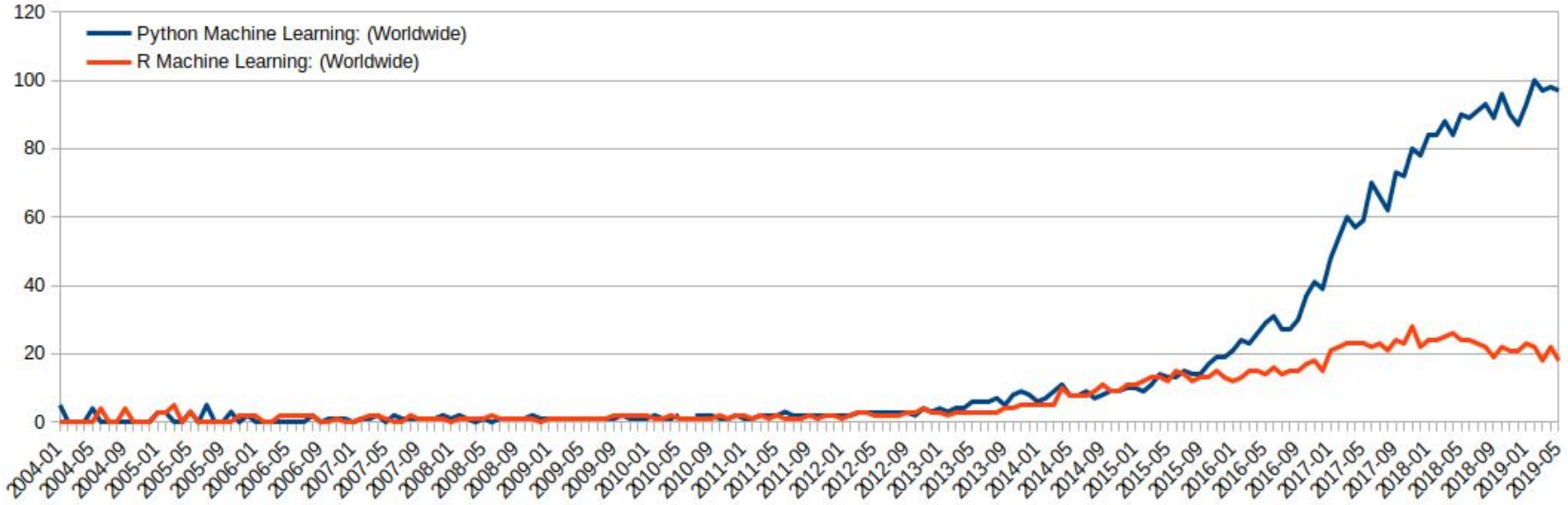
Overview

The textbook Advances in Financial Machine Learning, provides solutions to many of the problems faced by the quant community. One year after of its release there still was no sign of an open-source package, we considered this to be an opportunity.

This presentation highlights the challenges that the mlfinlab package addresses and the design choice we made in building it. We have aspired to build a research platform where we - along with contributors from the community - add tools, techniques, algorithms and research papers to the benefit of all of quantitative finance practitioners.



Python



NumPy

pandas

$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$

scikit learn

TensorFlow™

HUDSON
AND THAMES

Why Most Machine Learning Funds Fail

The 10 Reasons Most Machine Learning Funds Fail

MARCOS LÓPEZ DE PRADO

MARCOS LÓPEZ DE PRADO
is a research fellow at Lawrence Berkeley National Laboratory in Berkeley, CA.
lopezdeprado@lbl.gov

For almost a century, economics and finance have relied almost exclusively on the econometric toolkit to perform empirical analyses. The essential tool of econometrics is multivariate linear regression, an 18th-century technology that was already mastered by Gauss in 1794 (Stigler [1981]). Standard econometric models do not learn. It is hard to believe that something as complex as 21st-century finance could be grasped by something as simple as inverting a covariance matrix.

Every empirical science must build theories based on observation. If the statistical toolbox used to model these observations is linear regression, the researcher will fail to recognize the complexity of the data, and the theories will be awfully simplistic and useless. To this day, no one has been able to prove a theorem stating that risk premiums must be linear. Hence, reducing our analysis to linear regressions is likely a mistake. Econometrics may be a primary reason economics and finance have not experienced meaningful progress over the past 70 years (Calkin and López de Prado [2014a, 2014b]).

For centuries, medieval astronomers made observations and developed theories about celestial mechanics. These theories never considered noncircular orbits because they were deemed unholy and beneath God's plan. The prediction errors were so gross that ever more complex theories had to be devised to account for new observations. It was not until Kepler had the temerity to consider noncircular (elliptical) orbits that, all of a sudden, a much simpler general model was able to predict the position of the planets with astonishing accuracy. What if astronomers had never considered noncircular orbits? Well, what if economists finally started to consider nonlinear functions? Where is our Kepler? Finance does not have a *Principia* because no Kepler means no Newton.

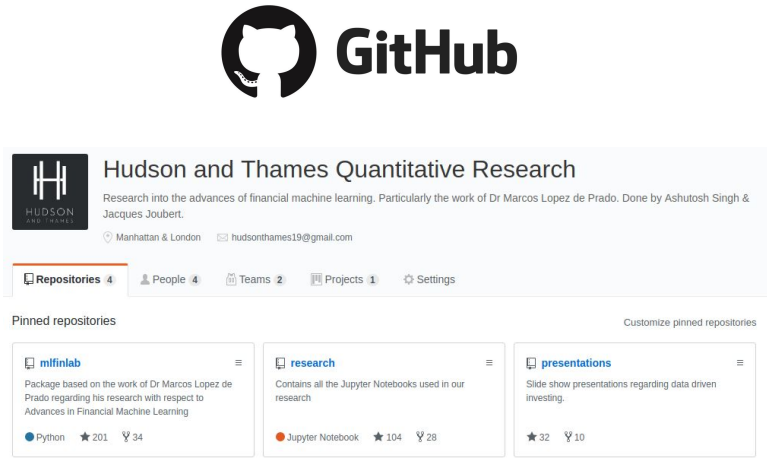
In recent years, quantitative fund managers have experimented and succeeded with the use of machine learning (ML) methods. An ML algorithm learns patterns in a high-dimensional space without being specifically directed. A common misconception is that ML methods are black boxes. This is not necessarily true. When correctly used, ML models do not replace theory; they guide it. Once we understand what features are predictive of a phenomenon, we can build a theoretical explanation that can be tested on an independent dataset. Students of economics and finance would do well to enroll in ML courses rather than econometrics. Econometrics may be good enough to succeed in financial academia (for now), but succeeding in business requires ML.

At the same time, ML is no panacea. The flexibility and power of ML techniques have a dark side. When misused, ML algorithms will confuse statistical flukes

10 Reasons (Journal of Portfolio Management)

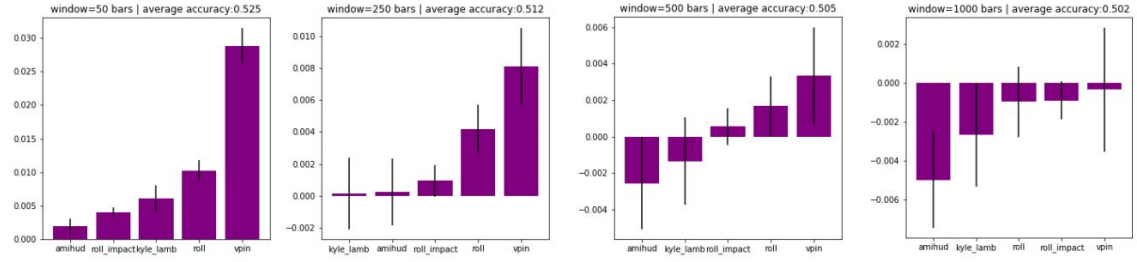
1. **Working in Silos**
2. **Research Through Backtesting**
3. **Chronological Sampling**
4. **Integer Differentiation**
5. **Fixed-Time Horizon Labeling**
6. **Learning Side and Size Simultaneously**
7. Weighting of Non-Independent Identically Distributed Samples
8. Cross-Validation Leakage
9. Walk-Forward (or Historical) Backtesting
10. Backtest Overfitting

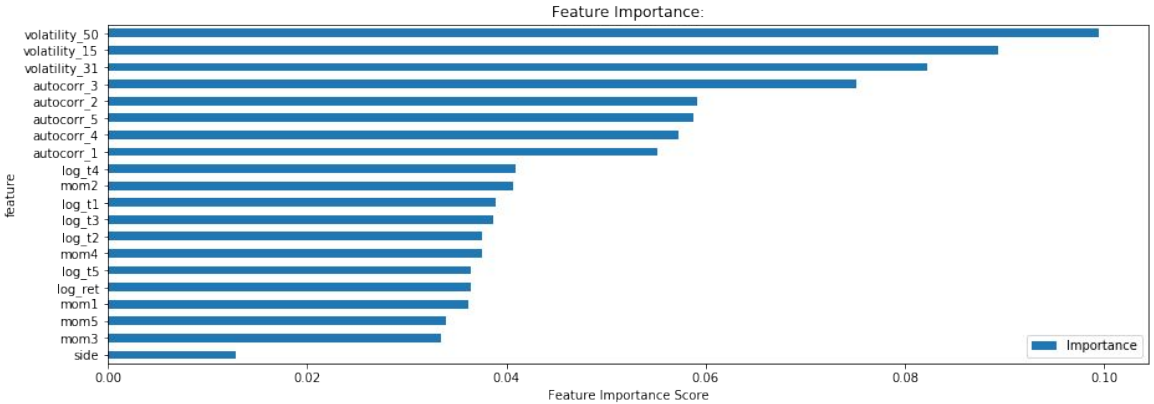
Setting up a Financial Research Lab



HUDSON
AND THAMES

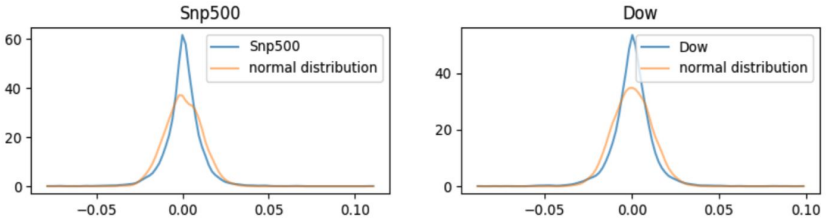
Feature Importance

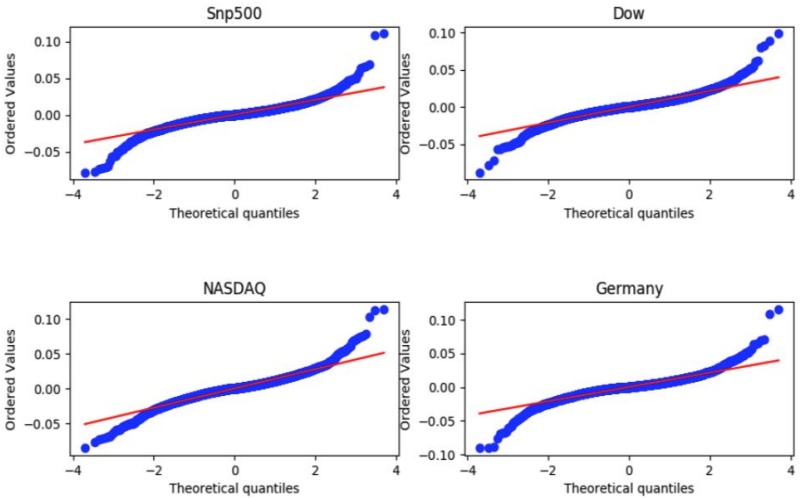




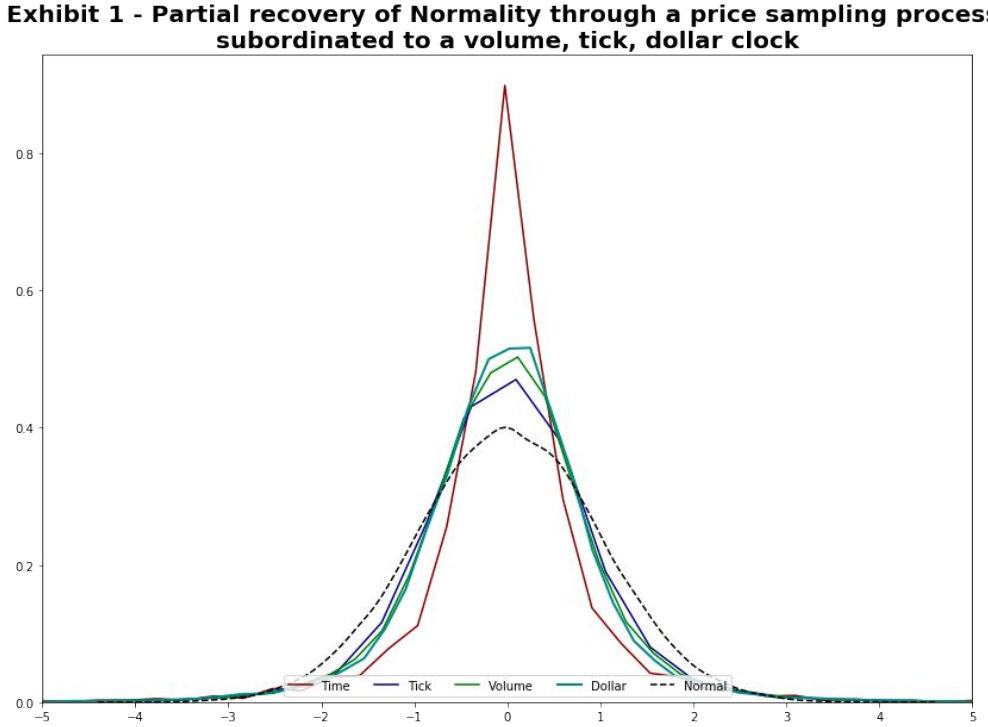
- Research through backtesting - multiple testing - increases the probability of making a false discovery.
- Better to focus on feature importance.
    - Engineer useful features
    - Drop those that contribute to noise
- Make use of Random Forest Algorithm
- Other techniques available
- Work in progress.

Chronological Clock: Distribution Comparison





Better Sampling Techniques



> Chronological Sampling (fixed time interval sampling)
> New Financial Data Structures

Standard Bars:

- Tick Bars
- Volume Bars
- Dollar Bars

Information Driven Bars

- Imbalance Bars
- Run Bars

HUDSON
AND THAMES

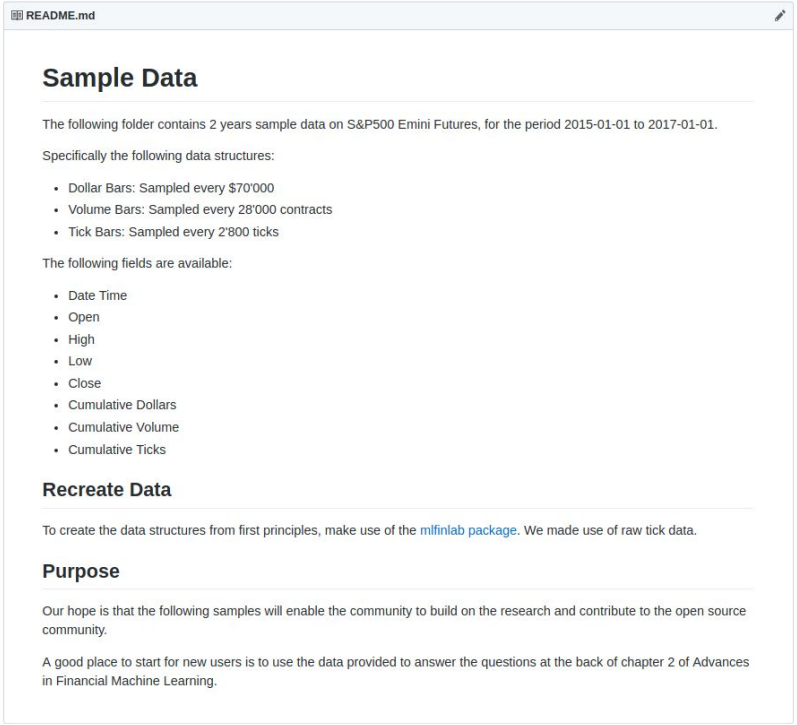
Barriers to Entry

No open-source implementations to create the financial data structures:

1. Must be fast
2. Work on large files
3. Reliable (Unit Tests)

Tick data is expense. No sample data available.

- Provide 2 year sample for various financial data structures. (Not raw tick data)



README.md

Sample Data

The following folder contains 2 years sample data on S&P500 Emini Futures, for the period 2015-01-01 to 2017-01-01.

Specifically the following data structures:

- Dollar Bars: Sampled every $70'000
- Volume Bars: Sampled every 28'000 contracts
- Tick Bars: Sampled every 2'800 ticks

The following fields are available:

- Date Time
- Open
- High
- Low
- Close
- Cumulative Dollars
- Cumulative Volume
- Cumulative Ticks

Recreate Data

To create the data structures from first principles, make use of the mlfinlab package. We made use of raw tick data.

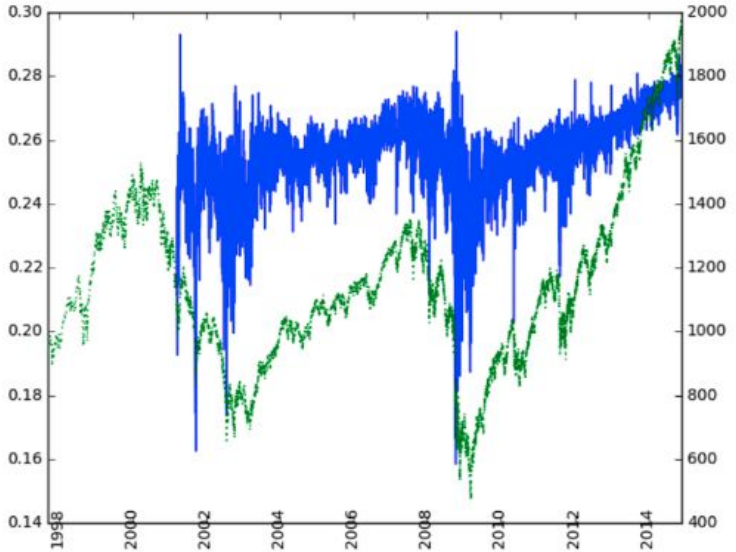
Purpose

Our hope is that the following samples will enable the community to build on the research and contribute to the open source community.

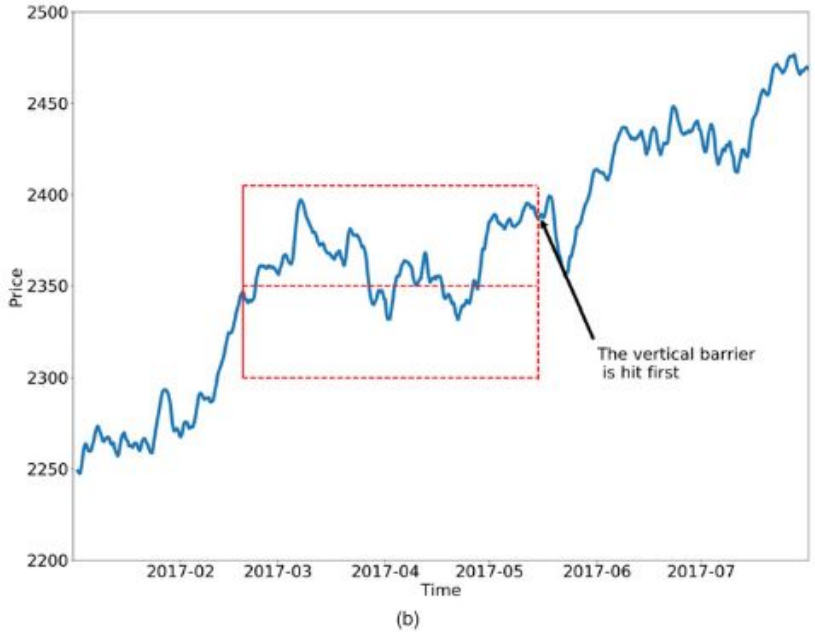
A good place to start for new users is to use the data provided to answer the questions at the back of chapter 2 of Advances in Financial Machine Learning.

Feature Engineering: Maintaining Memory

- In order to perform inferential analyses, researchers need to work with invariant processes, such as:
  - returns on prices (or changes in log-prices)
  - changes in yield
  - changes in volatility
- These operations make the series stationary, at the expense of removing all memory from the original series.
- Memory is the basis for the model's predictive power.
  - For example, equilibrium (stationary) models need some memory to assess how far the price process has drifted away from the long-term expected value in order to generate a forecast.
- The dilemma is:
  - returns are stationary however memory-less; and
  - prices have memory however they are non-stationary.



Financial Labeling Techniques: Triple-Barrier



- The Triple Barrier Method labels an observation according to the first barrier touched out of three barriers.
  - Two horizontal barriers are defined by profit-taking and stop-loss limits, which are a dynamic function of estimated volatility (whether realized or implied).
  - A third, vertical barrier, is defined in terms of number of bars elapsed since the position was taken (an expiration limit).
- The barrier that is touched first by the price path determines the label:
  - Upper horizontal barrier: Label 1.
  - Lower horizontal barrier: Label -1.
  - Vertical barrier: Label 0.

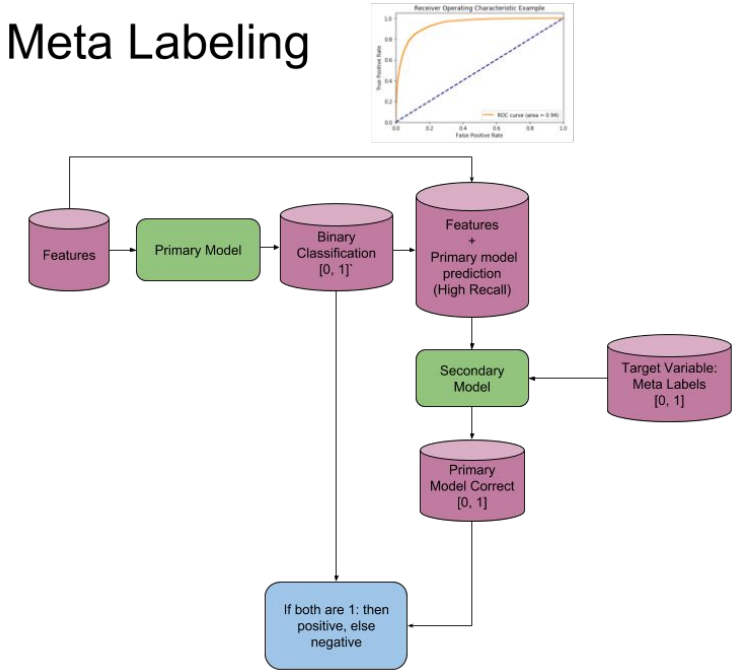
Learning Side and Size

Better to model two separate models.

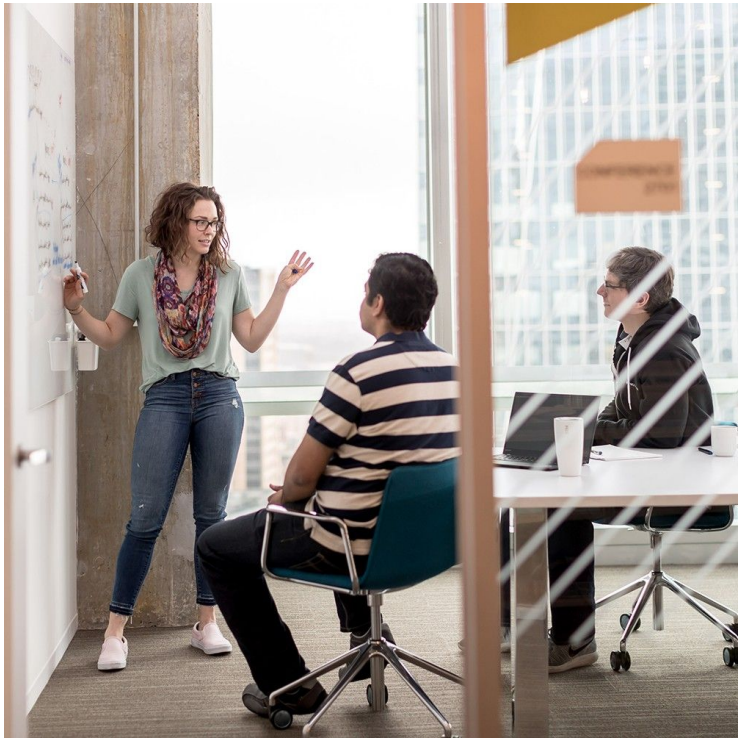
1. Side of the position (alpha model)
2. Size of the position (risk management)

Meta-Labeling

- Takes the side from the primary model (long or short).
- Train a ML model to determine if we should trade on the signal or not.
  - Train Random Forest
  - Use Cross-validation and Grid Search to find the optimal hyperparameters.
- Map confidence level to position size
  - Add bet sizing algorithm



Paradigm Shift

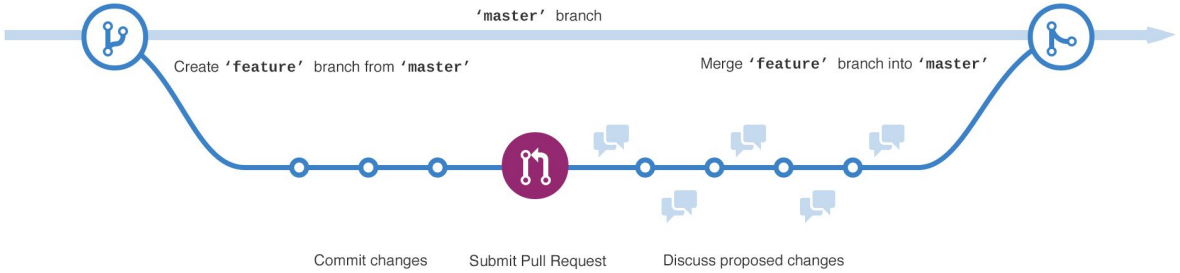


1. Focus on process: statistical properties, sampling, feature engineering, feature importance, ensembling.
2. Avoid research through backtesting.
3. Many-to-one models vs many-to-many
4. Trading approach vs investing
5. Keeping track of the number of trials run (Deflated Sharpe ratio)
6. Models can contain features from:
    a. Price action (microstructure)
    b. Fundamental (Accounting ratios)
    c. Alternative data (Satelites)
    d. Dimensionality reduction


HUDSON
AND THAMES
Package Design

Github Development Platform

GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 31 million developers.



Code review

Project management

Integrations

Team management

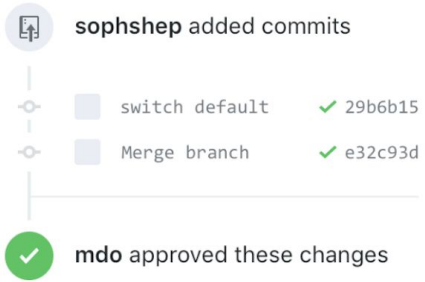
Social coding

Documentation

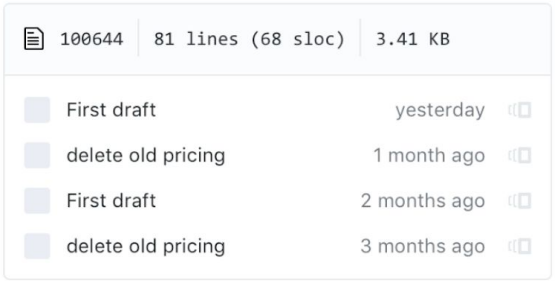
Code hosting

Code Review

Diffs



History



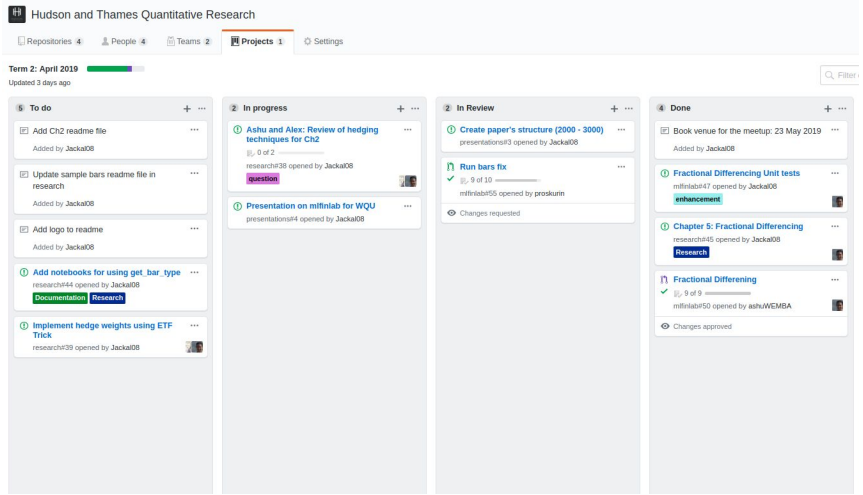
Blame

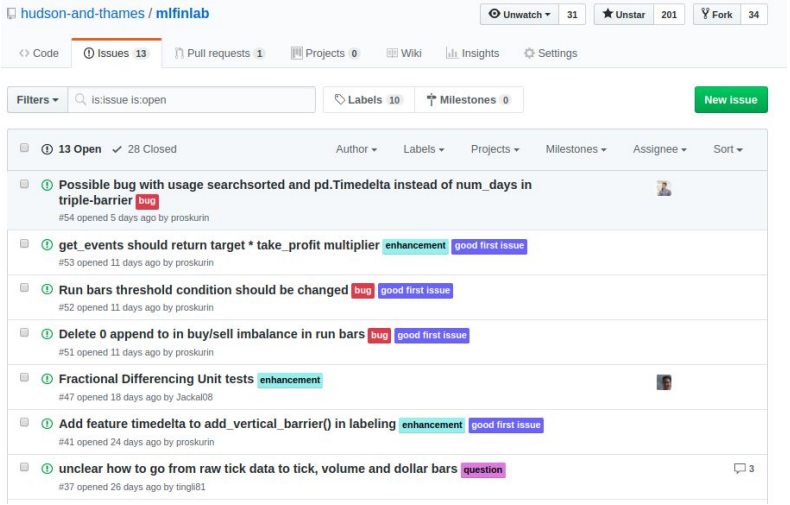
Code review

Project management

HUDSON
AND THAMES

Project Management





Code review

Project management

Hudson and Thames Quantitative Research

Research into the advances of financial machine learning. Particularly the work of Dr Marcos Lopez de Prado. Done by Ashutosh Singh & Jacques Joubert.

Manhattan & London hudsonthames19@gmail.com

Repositories 4 People 4 Teams 2 Projects 1 Settings

Pinned repositories

Customize pinned repositories

mlfinlab

Package based on the work of Dr Marcos Lopez de Prado regarding his research with respect to Advances in Financial Machine Learning

Python ★ 201 34

research

Contains all the Jupyter Notebooks used in our research

Jupyter Notebook ★ 104 28

presentations

Slide show presentations regarding data driven investing.

★ 32 10

Find a repository… Type: All Language: All New

mlfinlab

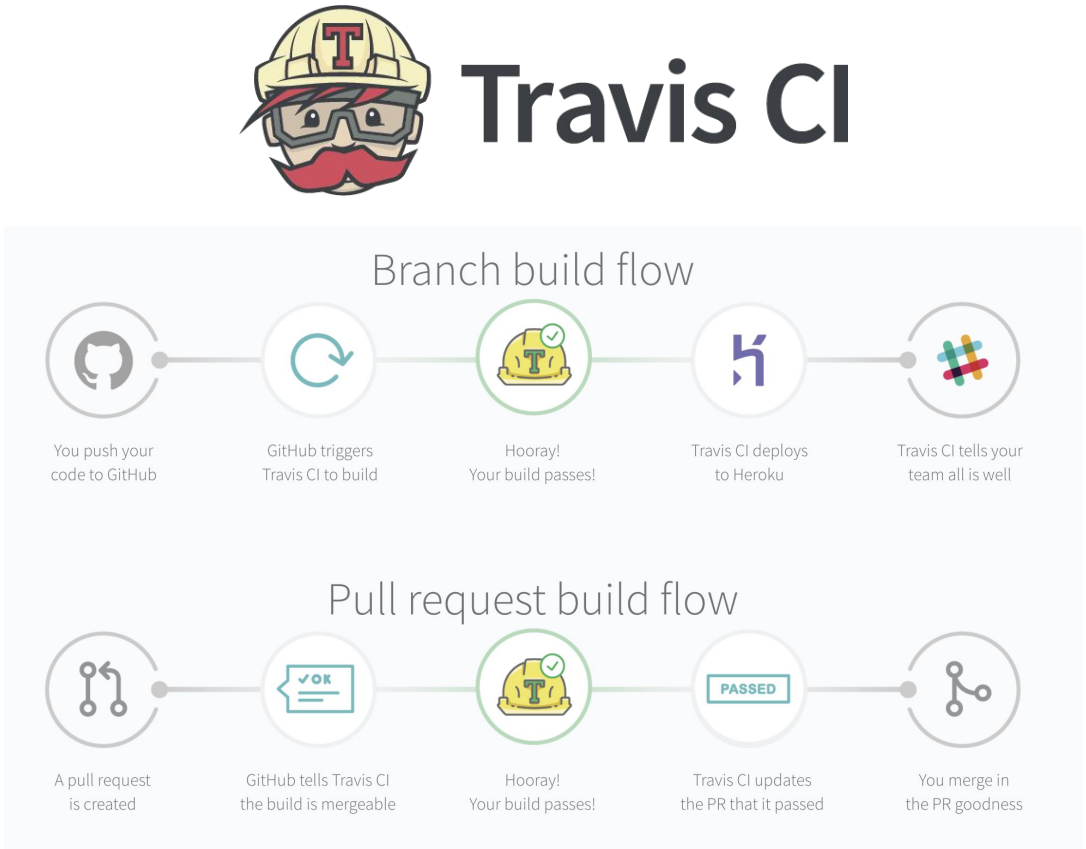
Package based on the work of Dr Marcos Lopez de Prado regarding his research with respect to Advances in Financial Machine Learning

Top languages

Python TeX Jupyter Notebook

Continuous Integration

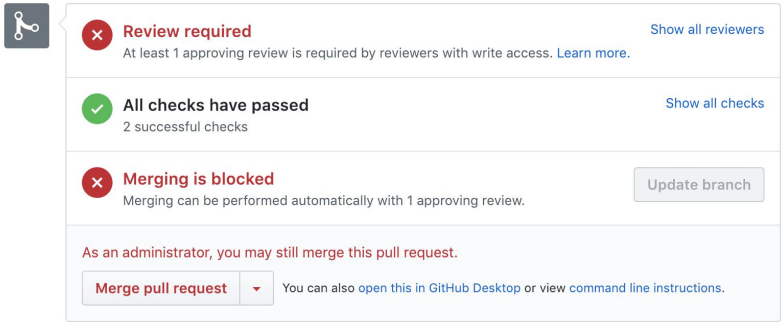
Continuous Integration (CI) is the process of automating the build and testing of code every time a team member commits changes to version control. CI encourages developers to share their code and unit tests by merging their changes into a shared version control repository after every small task completion. Committing code triggers an automated build system to grab the latest code from the shared repository and to build, test, and validate the full master branch (also known as the trunk or main).



HUDSON
AND THAMES

Automated Scripts

1. Code style checks
2. Unit tests pass
3. 100% code coverage
4. Build package and index to PyPi
5. Documentation

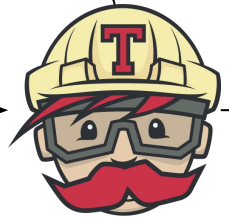



```
Messages by category
--------------------

+-----------+-------+---------+-----------+
|type       |number |previous |difference |
+===========+=======+=========+===========+
|convention |0      |NC       |NC         |
+-----------+-------+---------+-----------+
|refactor   |0      |NC       |NC         |
+-----------+-------+---------+-----------+
|warning    |0      |NC       |NC         |
+-----------+-------+---------+-----------+
|error      |0      |NC       |NC         |
+-----------+-------+---------+-----------+




-----------------------------------
Your code has been rated at 10.00/10

The command "pylint mlfinlab --rcfile=.pylintrc -f text" exited with 0.

$ bash coverage
----Running Code Coverage----
rm: cannot remove '.coverage.*': No such file or directory

Running tests...
----------------------------------------------------------------------
...
----------------------------------------------------------------------
Ran 3 tests in 0.000s

OK

Generating XML reports...
Name                              Stmts   Miss Branch BrPart  Cover   Missing
-----------------------------------------------------------------------------
mlfinlab/data_structures/code.py      4      0      0      0   100%
The command "bash coverage" exited with 0.



Done. Your build exited with 0.
```

HUDSON
AND THAMES

Workflow

GitHub



python™ Package Index


Build Welcoming
Communities

Starting an Organization

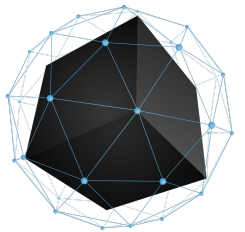
Checklist

- ✓ Description
- ✓ README
- ✓ Code of conduct
- ✓ Contributing
- ✓ License
- ✓ Issue templates — Edit
- ✓ Pull request template

Online Community

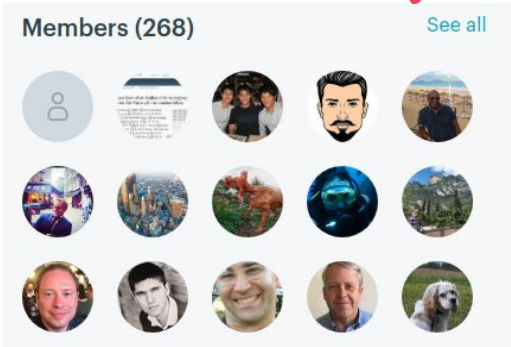






Offline Community

meetup



GridGain

HUDSON
AND THAMES

Co-Founder

Jacques Joubert

Machine learning consultant with buy side hedge fund experience. Skilled in machine learning, quantitative finance, systematic investing, software engineering.

HUDSON
AND THAMES

Co-Founder

Ashutosh Singh, CFA

Experienced Executive with a demonstrated history of working in the financial services industry. Strong business development professional skilled in Equity Research, Hedge Funds, Asset Management, Fixed Income, and Derivatives.

HUDSON
AND THAMES

Thank You