Time series analysis when “good enough” is good enough

Boris Gorelik, Ph.D
Data Scientist
AUTOMATTIC
boris@gorelik.net

PyCon Israel
Bar Ilan University. June 2017

Time series analysis when “good enough” is good enough

In deep trouble
The simplicity ladder
The hole is deeper than I thought
Light at the end of the tunnel
The ladder isn’t tall enough
Towards the bright future

THIS PRESENTATION IS BASED ON A TRUE STORY. FOR DRAMATIC AND NARRATIVE PURPOSES, AND TO MAKE ME, THE PRESENTER, LOOK BETTER, THIS PRESENTATION CONTAINS FUNCTIONALIZED SCENES AND DIALOGUES, AS WELL AS MODIFIED TIME FLOW. NO ANIMALS WERE HARMED DURING THE PREPARATION OF THIS PRESENTATION


AUTOMATTIC

WordPress.com

POSTS WRITTEN

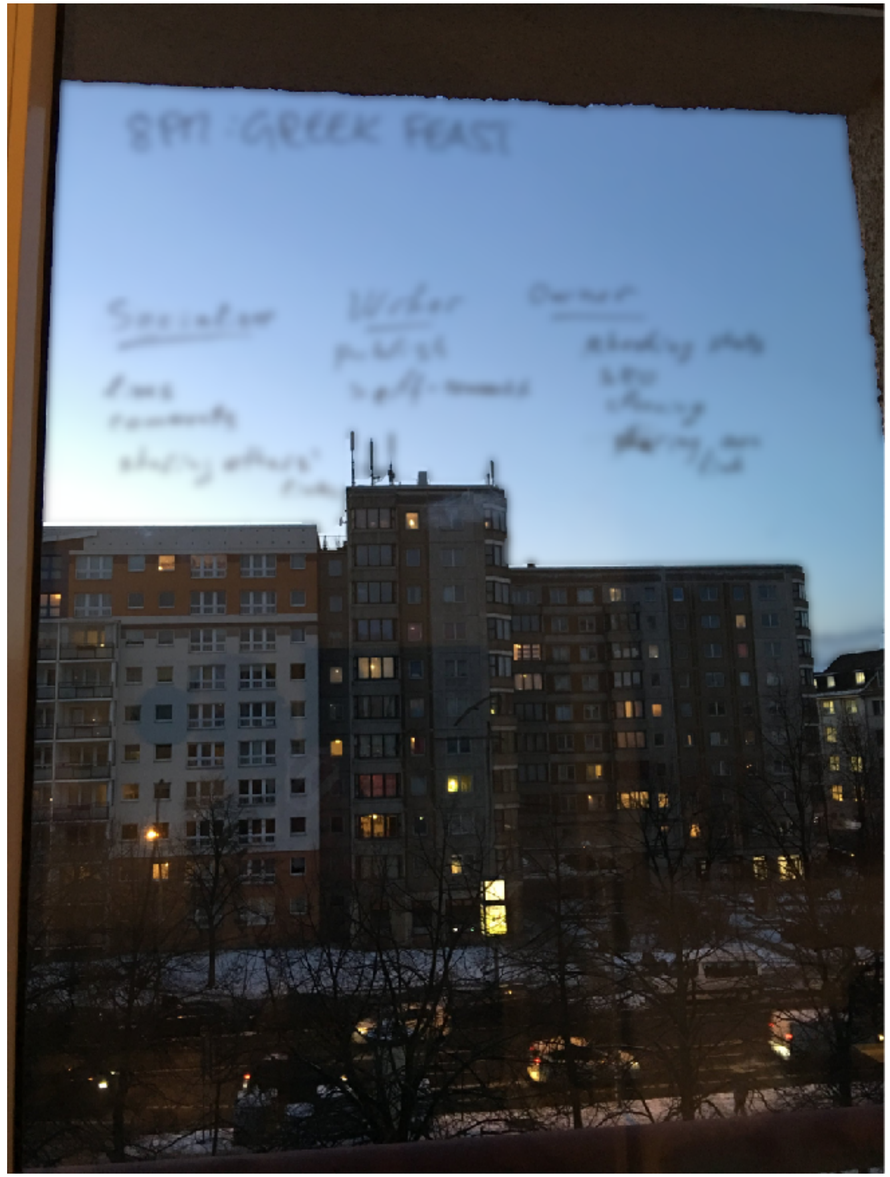
595,795,035

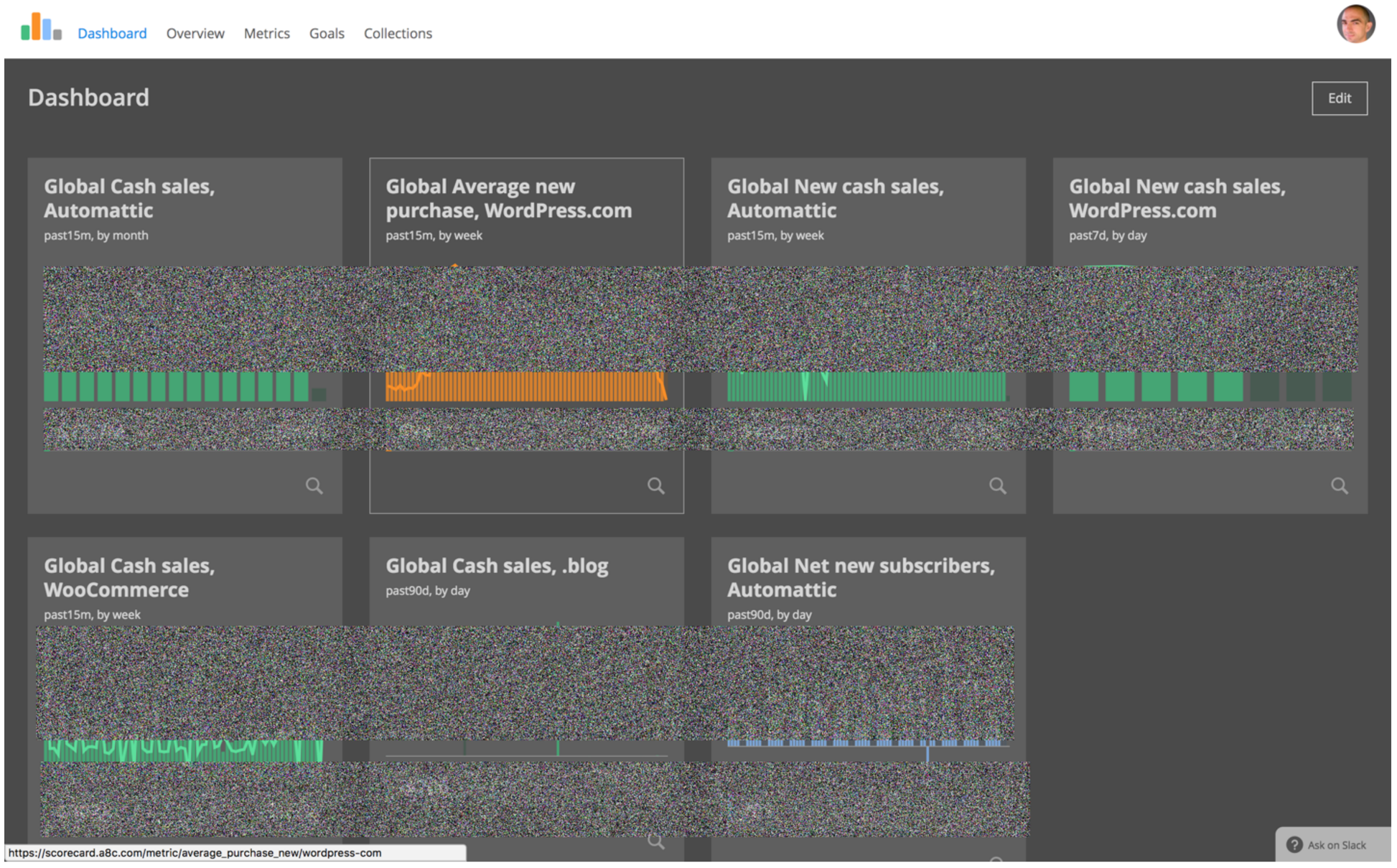
Comments Written

457,596,906 is a lot of comments (including this one, meta).


Greenland
Finland
Sweden
Norway
Russia
Kazakhstan
Mongolia
China
Afghanistan
Iran
Pakistan
Saudi Arabia
Egypt
Libya
Algeria
Mali
Niger
Chad
Sudan
Nigeria
Ethiopia
Kenya
DR Congo
Tanzania
Angola
Namibia
Botswana
Madagascar
Colombia
Brazil
Chile
Papua New Guinea
Australia
New Zealand


ÖFFNUNGSZEITEN


8PM: GREEK FEAST


Dashboard
Overview
Metrics
Goals
Collections
Dashboard
Edit
Global Cash sales, Automattic
past15m, by month
Global Average new purchase, WordPress.com
past15m, by week
Global New cash sales, Automattic
past15m, by week
Global New cash sales, WordPress.com
past7d, by day
Global Cash sales, WooCommerce
past15m, by week
Global Cash sales, .blog
past90d, by day
Global Net new subscribers, Automattic
past90d, by day
https://scorecard.a8c.com/metric/average_purchase_new/wordpress-com
Ask on Slack

Carly,
data science lead

Stu,
CFO

Carly,
data science lead

Stu,
CFO

Carly,
data science lead

Dashboard

Global Cash sales, Automattic
past15m, by month

Global Average new purchase, WordPress.com
past15m, by week

Global New cash sales, Automattic
past15m, by week

Global New cash sales, WordPress.com
past7d, by day

Anomaly detection on time series
</problem>

It’s not that simple

It is!

Time series is a tough problem. You're clearly mistaken.

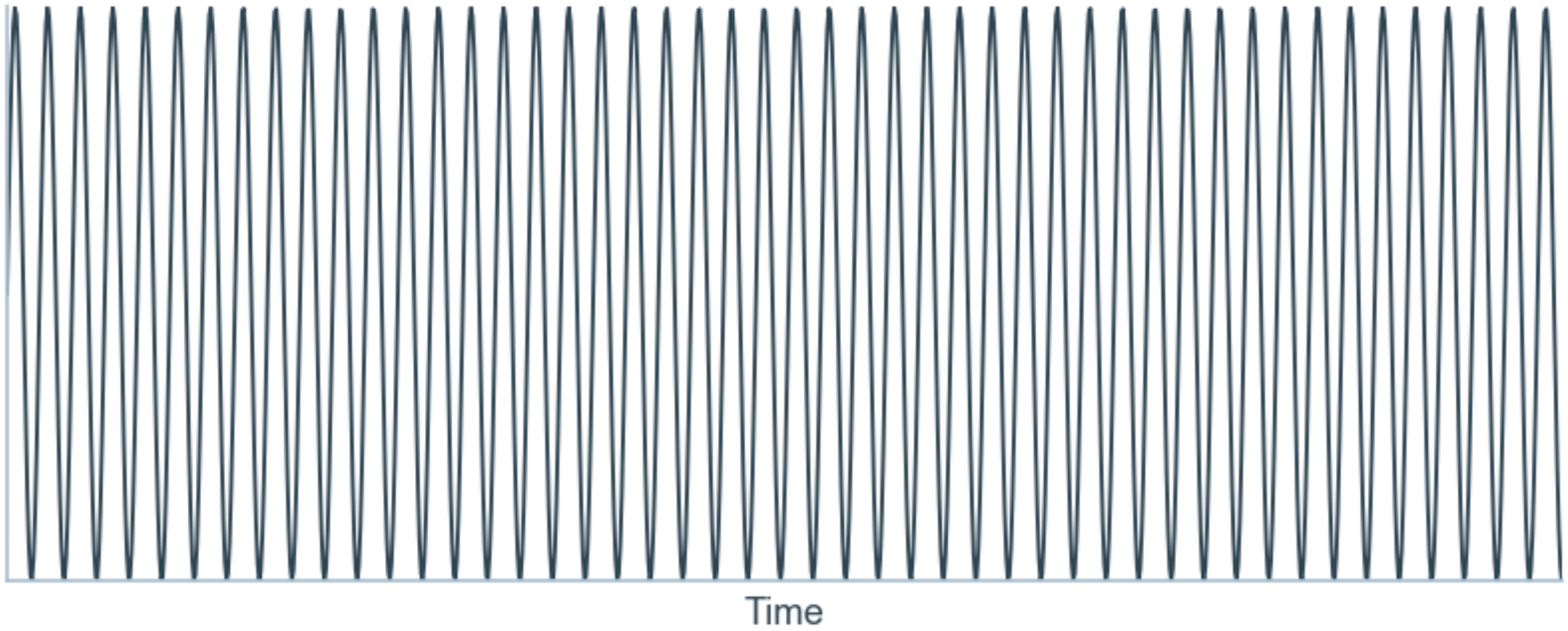
It’s solved problem.
An intern can do that!

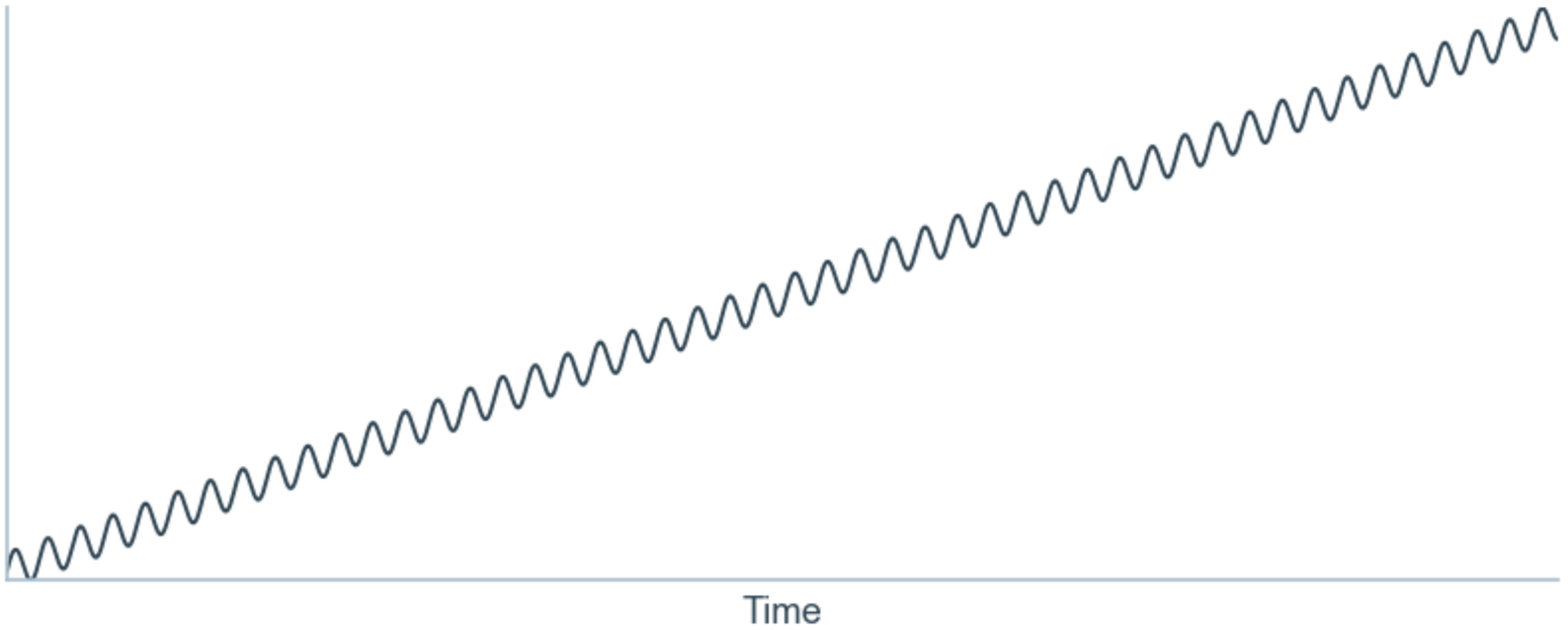
Solved problem?
Take it and solve it!

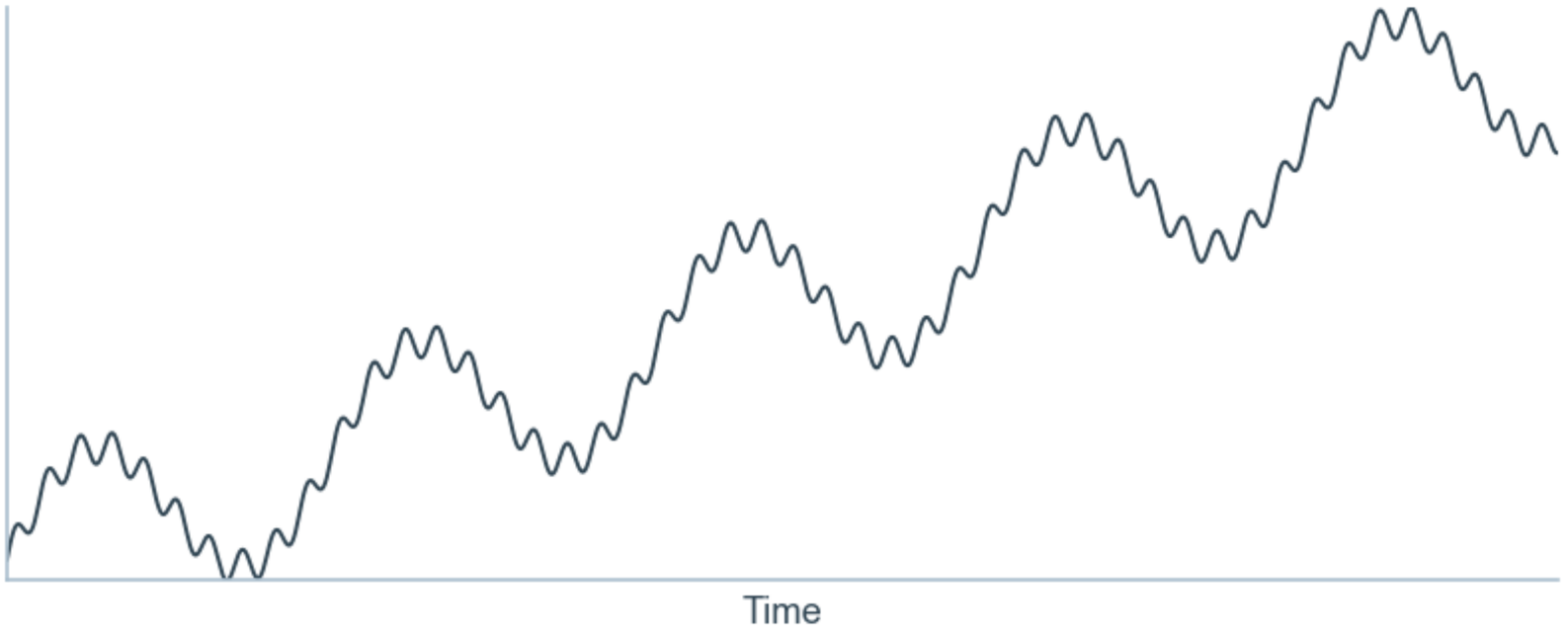
Time series analysis when “good enough” is good enough

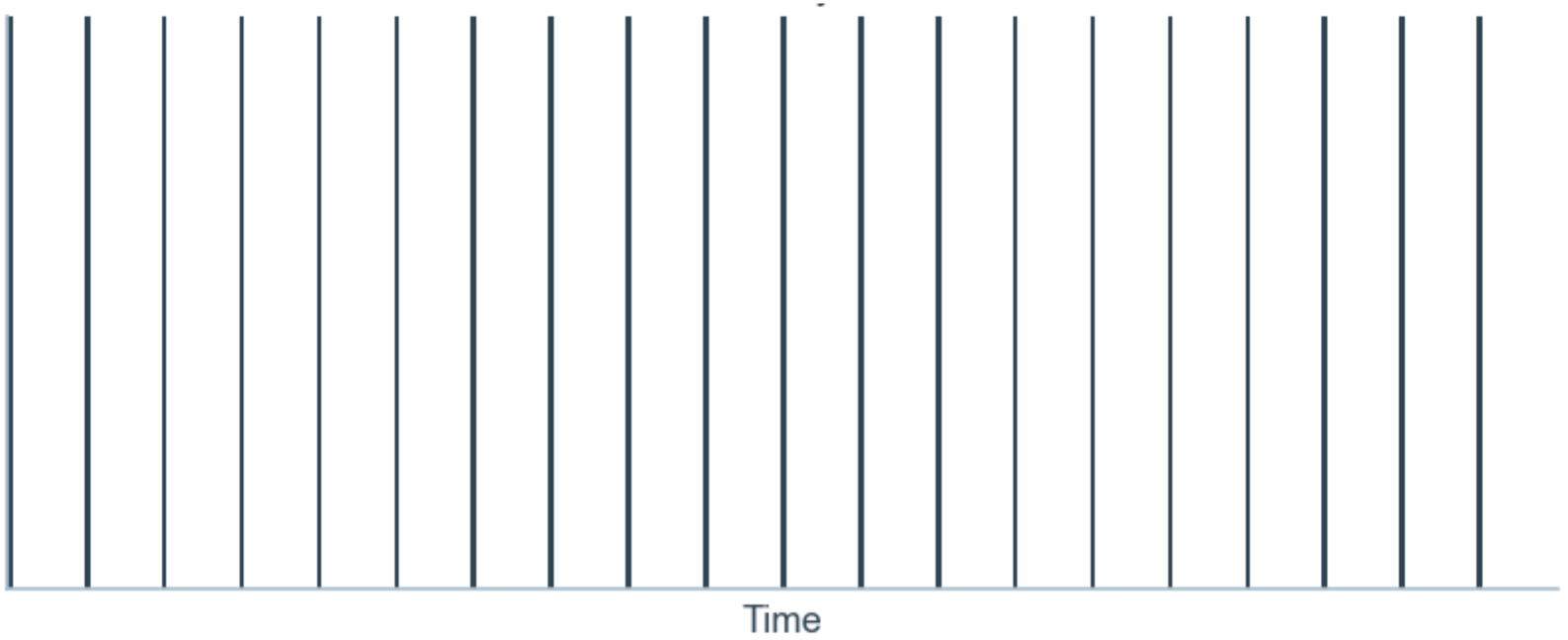
In deep trouble
The simplicity ladder
The hole is deeper than I thought
Light at the end of the tunnel
The ladder isn’t tall enough
Towards the bright future

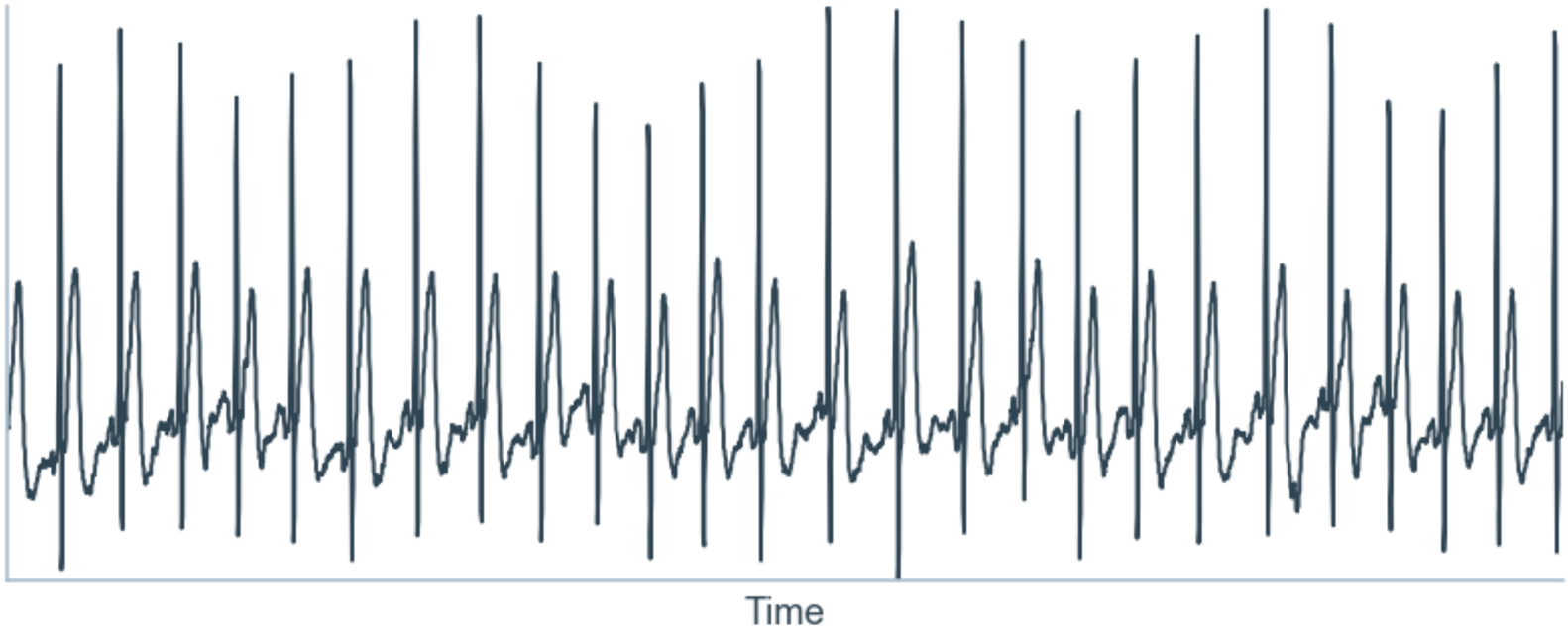
Time series is a tough problem


Time

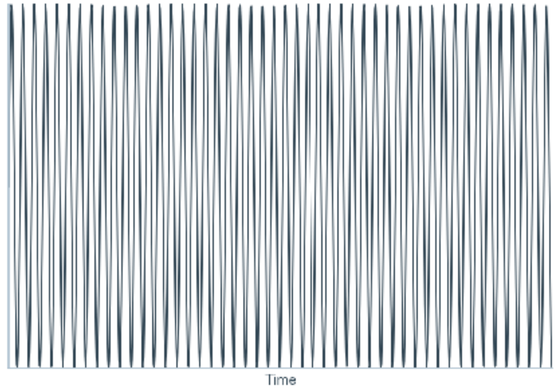

Time

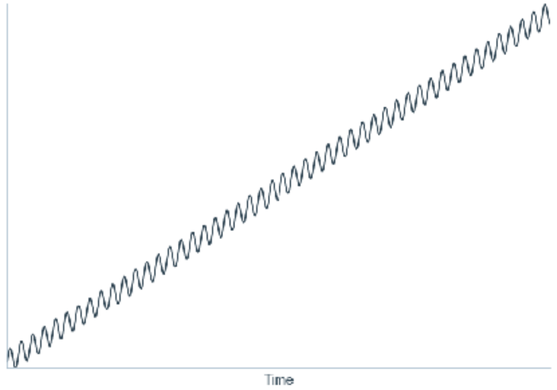

Time

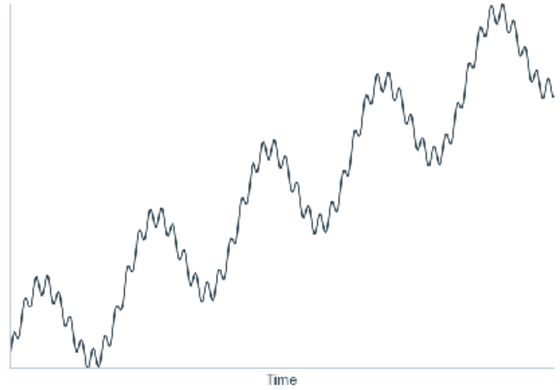

Time

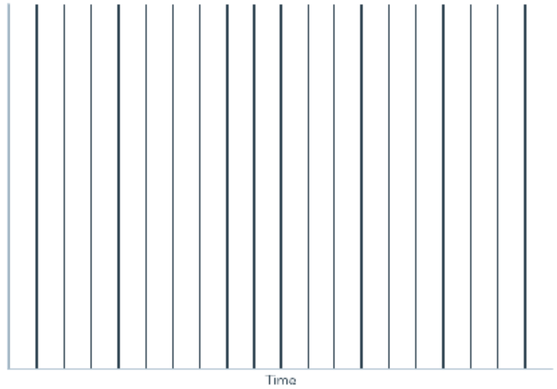

Time

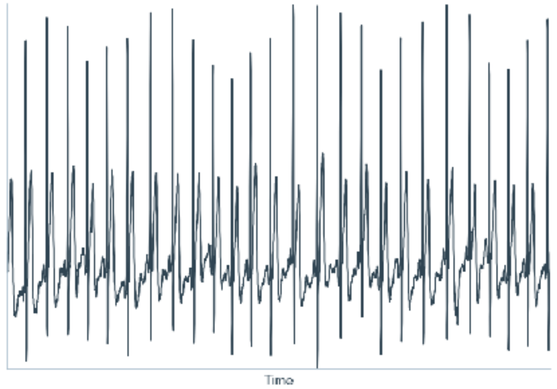
Time series is a tough problem











Time series is a tough problem

- malicious users
- multiple seasonality
- spikes
- drops
- over-saturation
- under-saturation
- concept drift
- error costs

Time series is a tough problem

- malicious users
- multiple seasonality
- spikes
- drops
- over-saturation
- under-saturation
- concept drift
- **error costs←**

Time series is a tough problem

- malicious users
- multiple seasonality
- spikes
- drops
- over-saturation
- under-saturation
- concept drift
- ~~error costs~~

Time series is a tough problem

- malicious users
- multiple seasonality
- **spikes←**
- **drops←**
- over-saturation
- under-saturation
- concept drift
- ~~error costs~~

Time series is a tough problem

- malicious users
- multiple seasonality
- ~~spikes~~
- ~~drops~~
- over-saturation
- under-saturation
- concept drift
- ~~error costs~~

Time series is a tough problem

- malicious users
- multiple seasonality
- ~~spikes~~
- ~~drops~~
- **over-saturation←**
- **under-saturation←**
- concept drift
- ~~error costs~~

Time series is a tough problem

- malicious users
- multiple seasonality
- ~~spikes~~
- ~~drops~~
- ~~over-saturation~~
- ~~under-saturation~~
- concept drift
- ~~error costs~~

Time series is a tough problem

- **malicious users←**
- multiple seasonality
- ~~spikes~~
- ~~drops~~
- ~~over-saturation~~
- ~~under-saturation~~
- concept drift
- ~~error costs~~

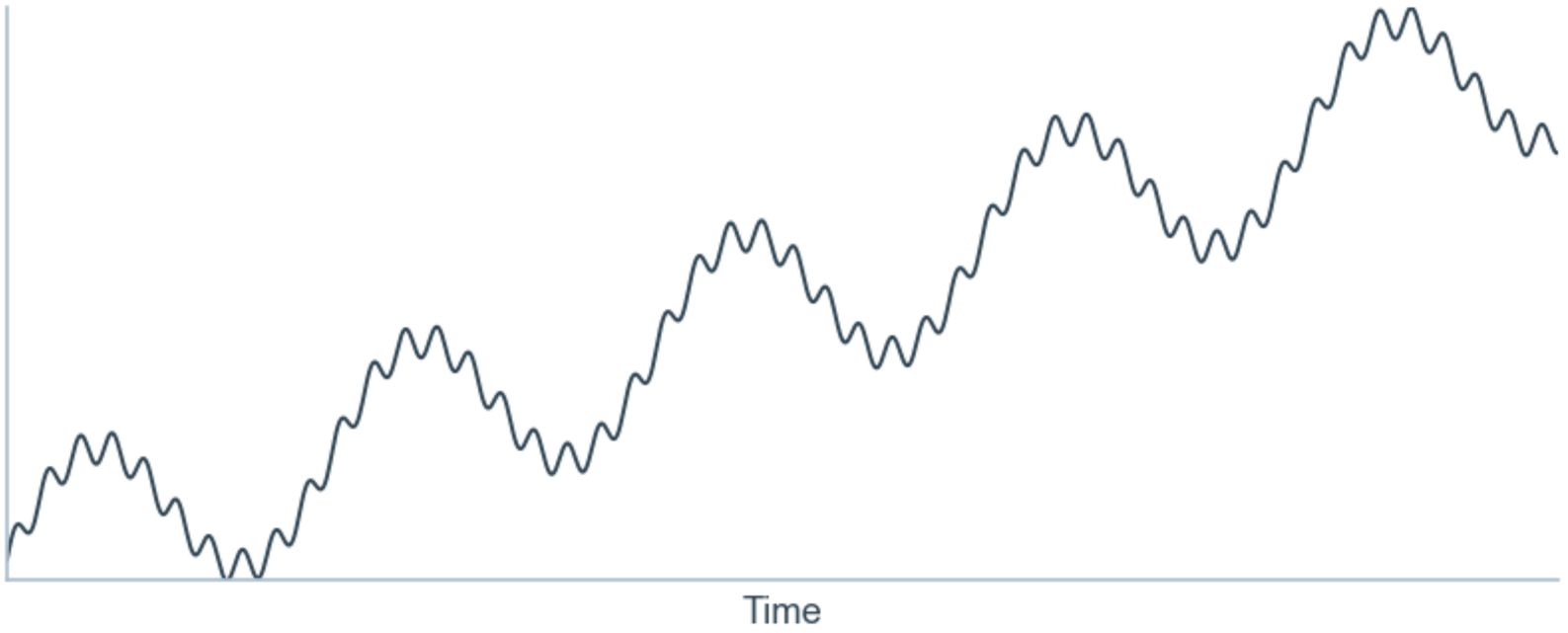
Time series is a tough problem

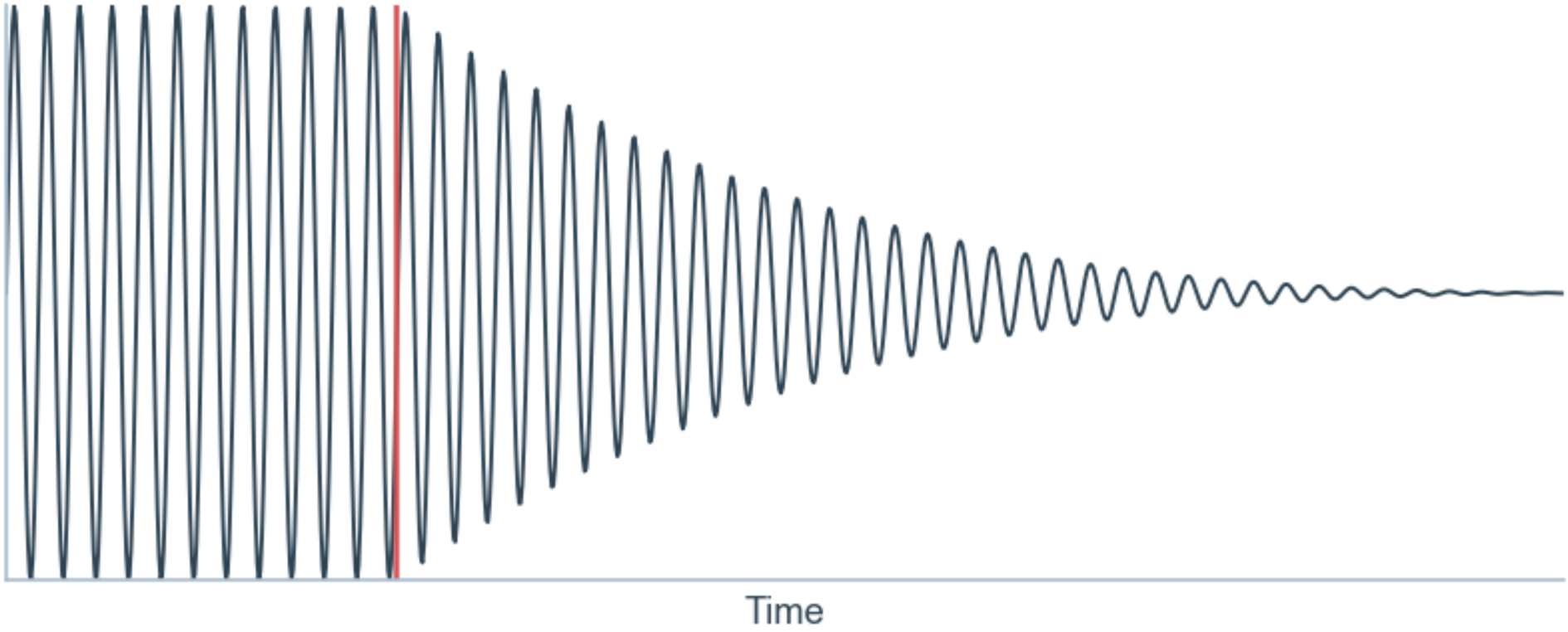
- ~~malicious users~~ *
- multiple seasonality
- ~~spikes~~
- ~~drops~~
- ~~over-saturation~~
- ~~under-saturation~~
- concept drift
- ~~error costs~~

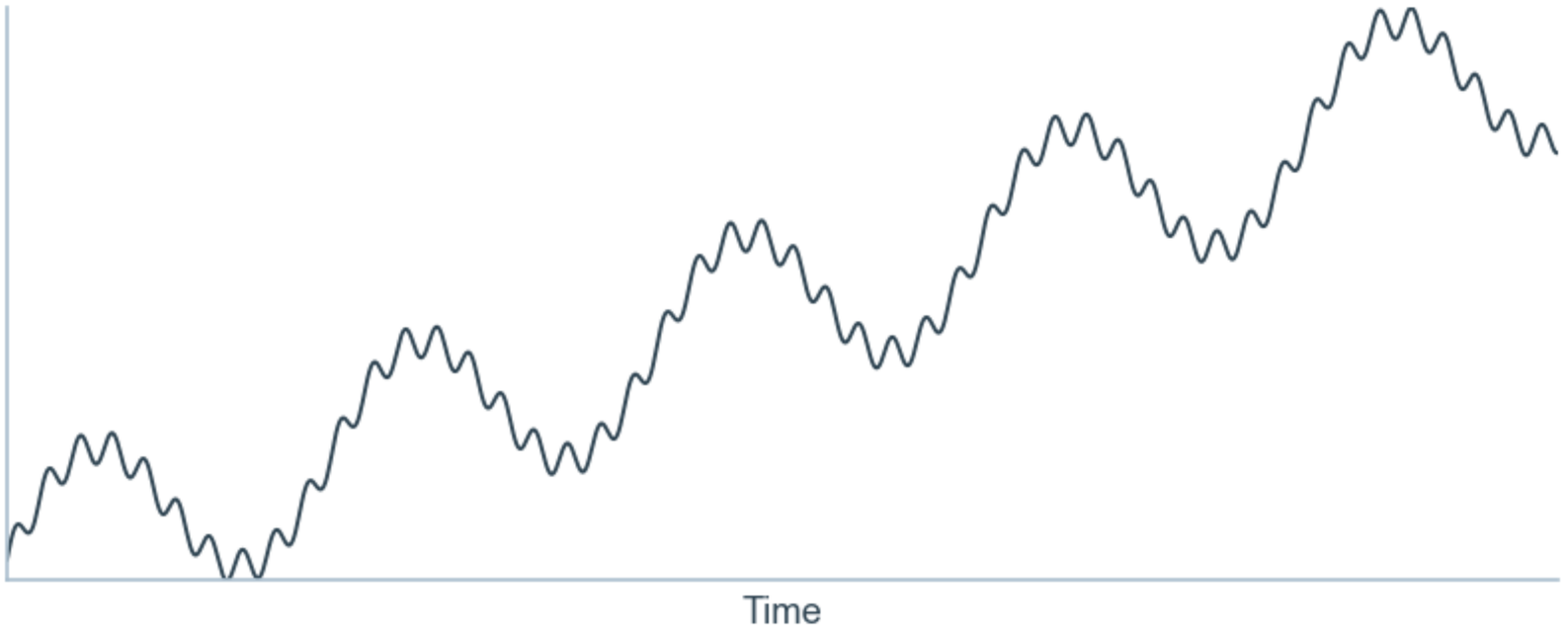
* touch wood

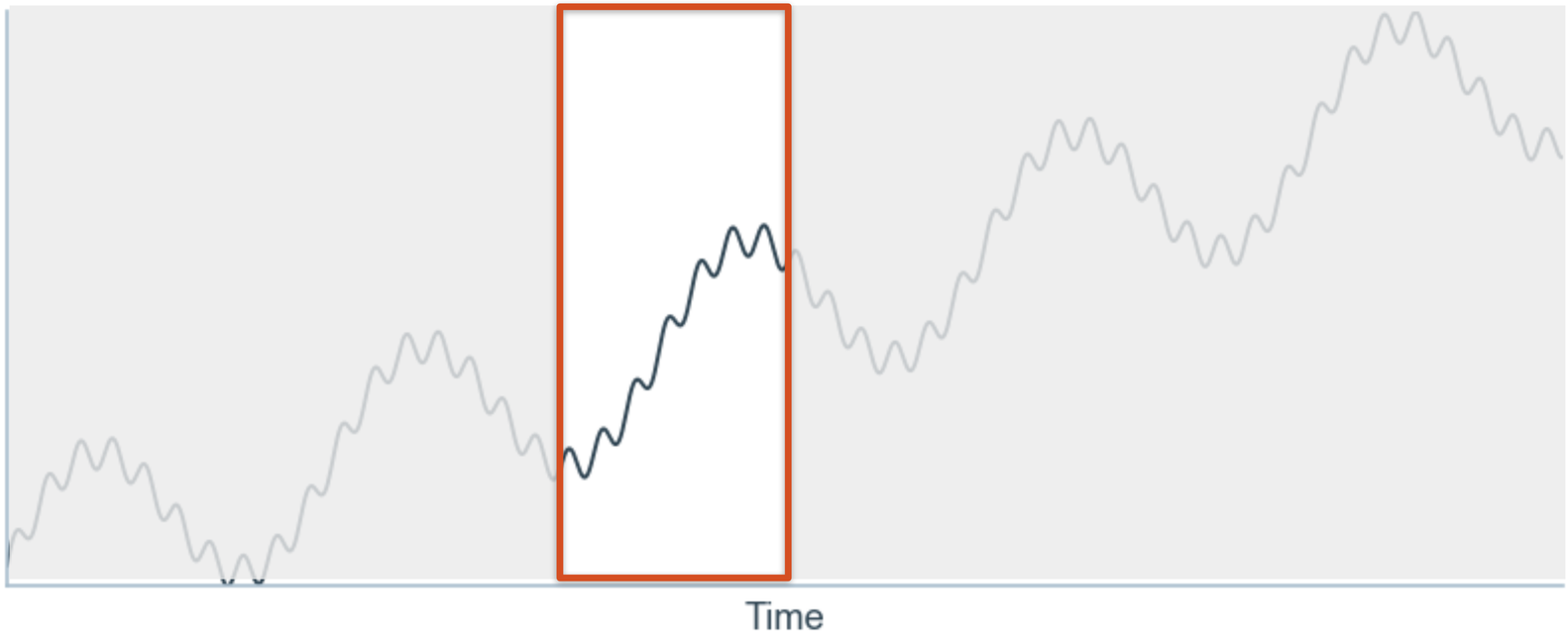
Time series is a tough problem

- ~~malicious users~~
- **multiple seasonality**
- ~~spikes~~
- ~~drops~~
- ~~over-saturation~~
- ~~under-saturation~~
- **concept drift**
- ~~error costs~~


Time


Time


Time

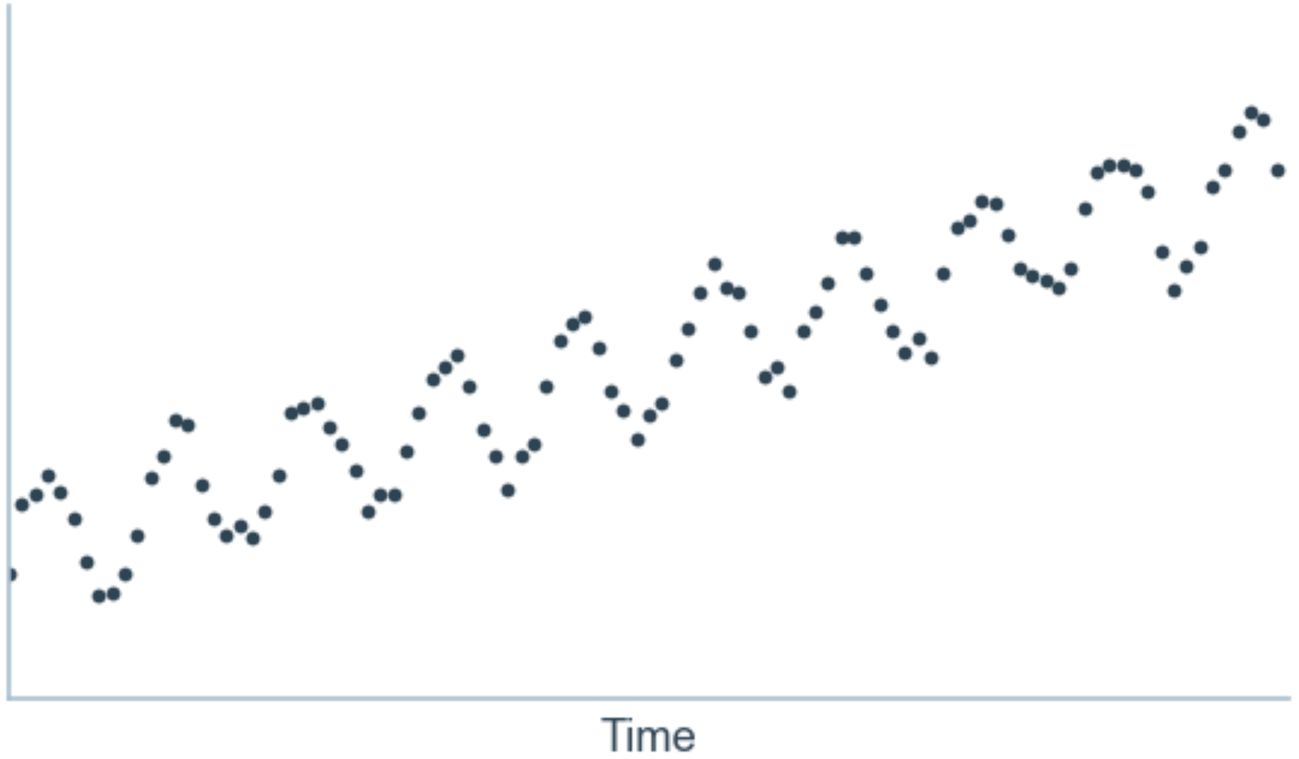

Time

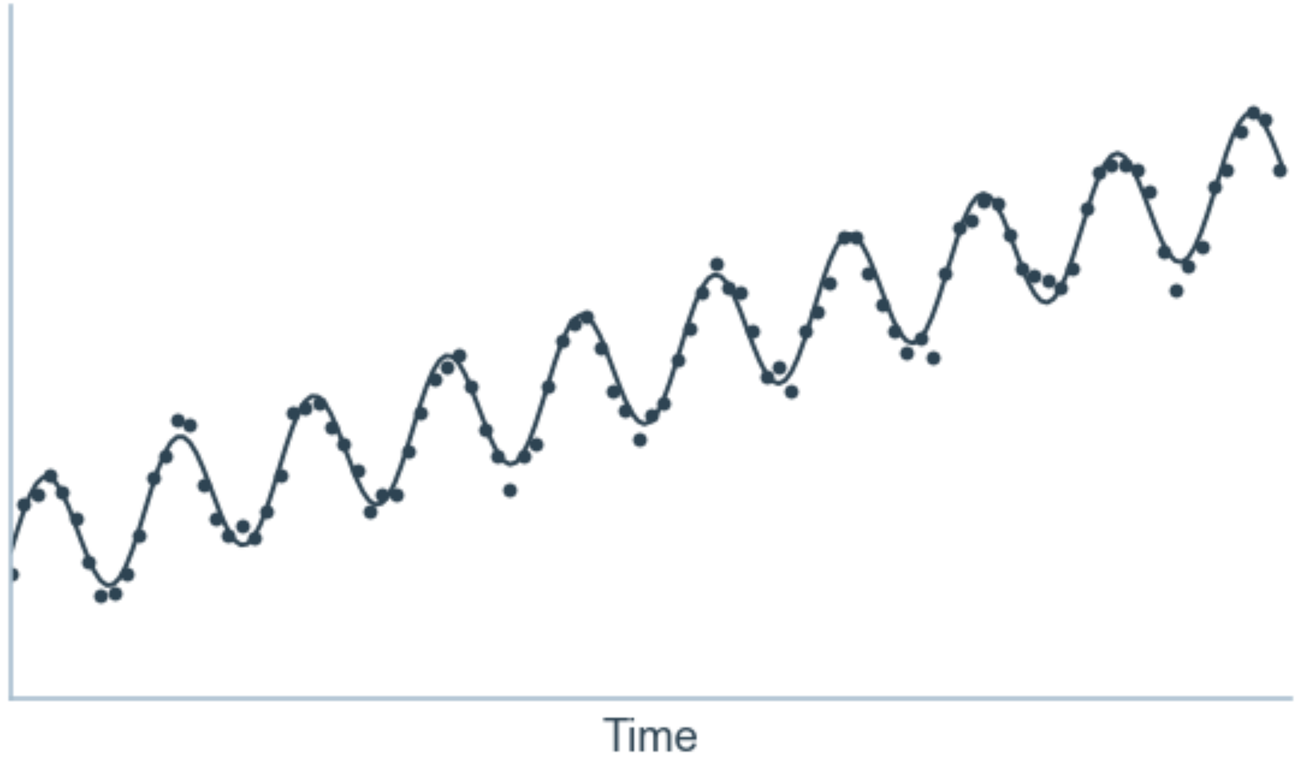
Time series is a tough problem

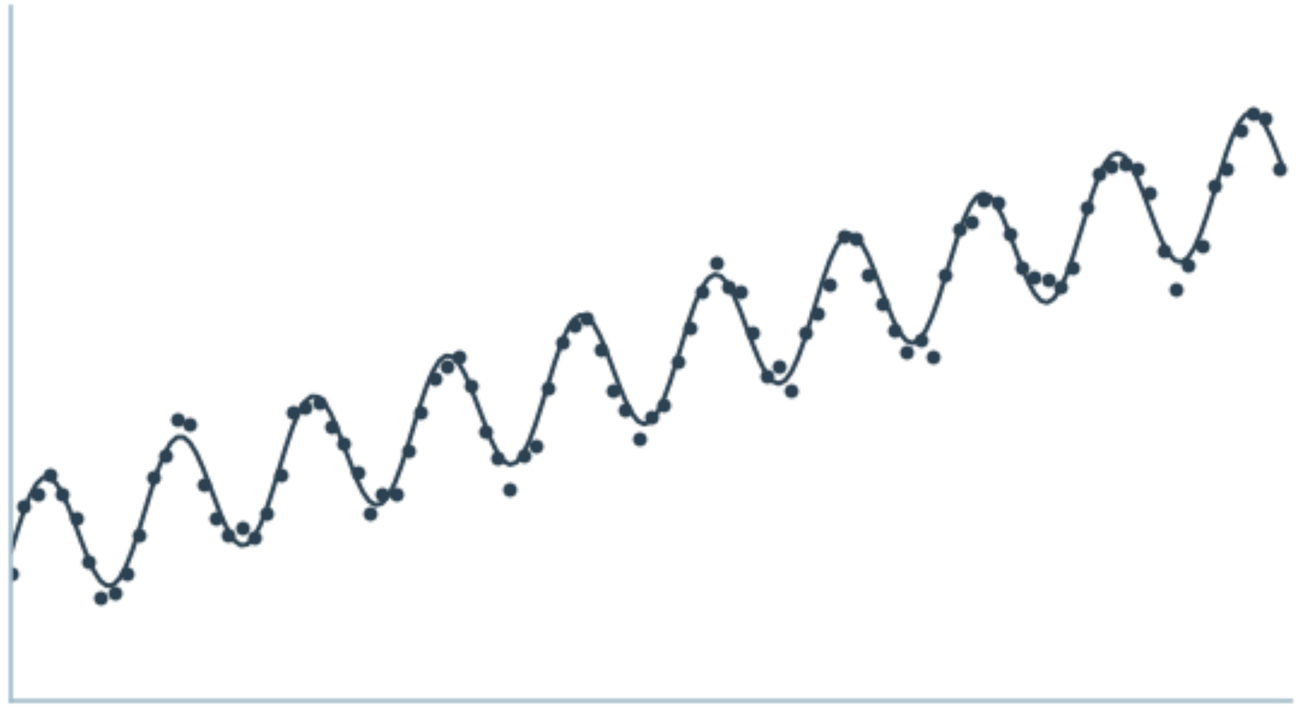
- ~~malicious users~~
- ~~multiple seasonality~~ → limit the history & the resolution
- ~~spikes~~
- ~~drops~~
- ~~over-saturation~~
- ~~under-saturation~~
- ~~concept drift~~ → limit the history & the resolution
- ~~high-frequency fluctuations~~
- ~~error costs~~

Time series is NOT a tough problem

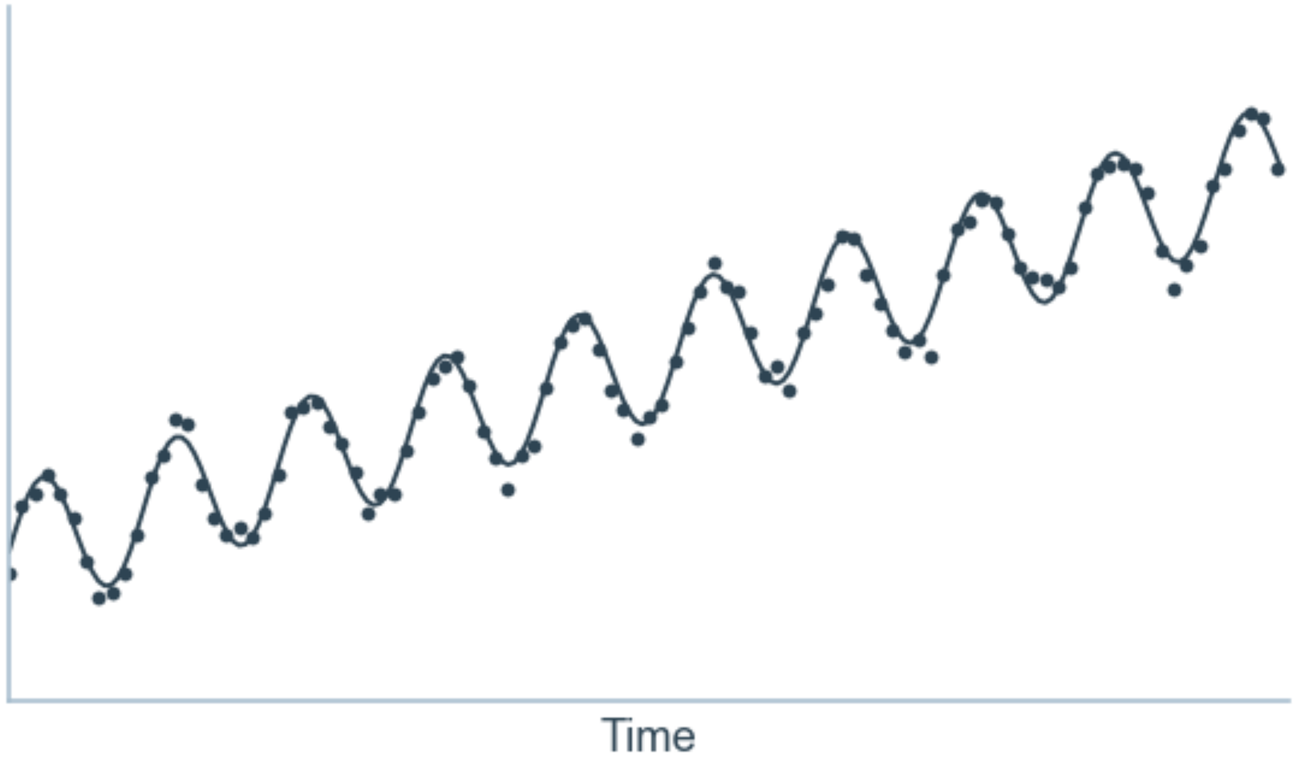
- ~~malicious users~~
- ~~multiple seasonality~~
- ~~spikes~~
- ~~drops~~
- ~~over-saturation~~
- ~~under-saturation~~
- ~~concept drift~~
- ~~high-frequency fluctuations~~
- ~~error costs~~

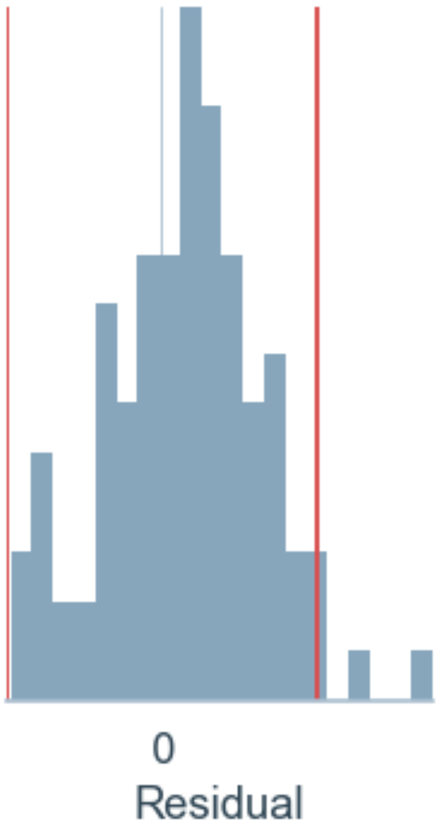

Time

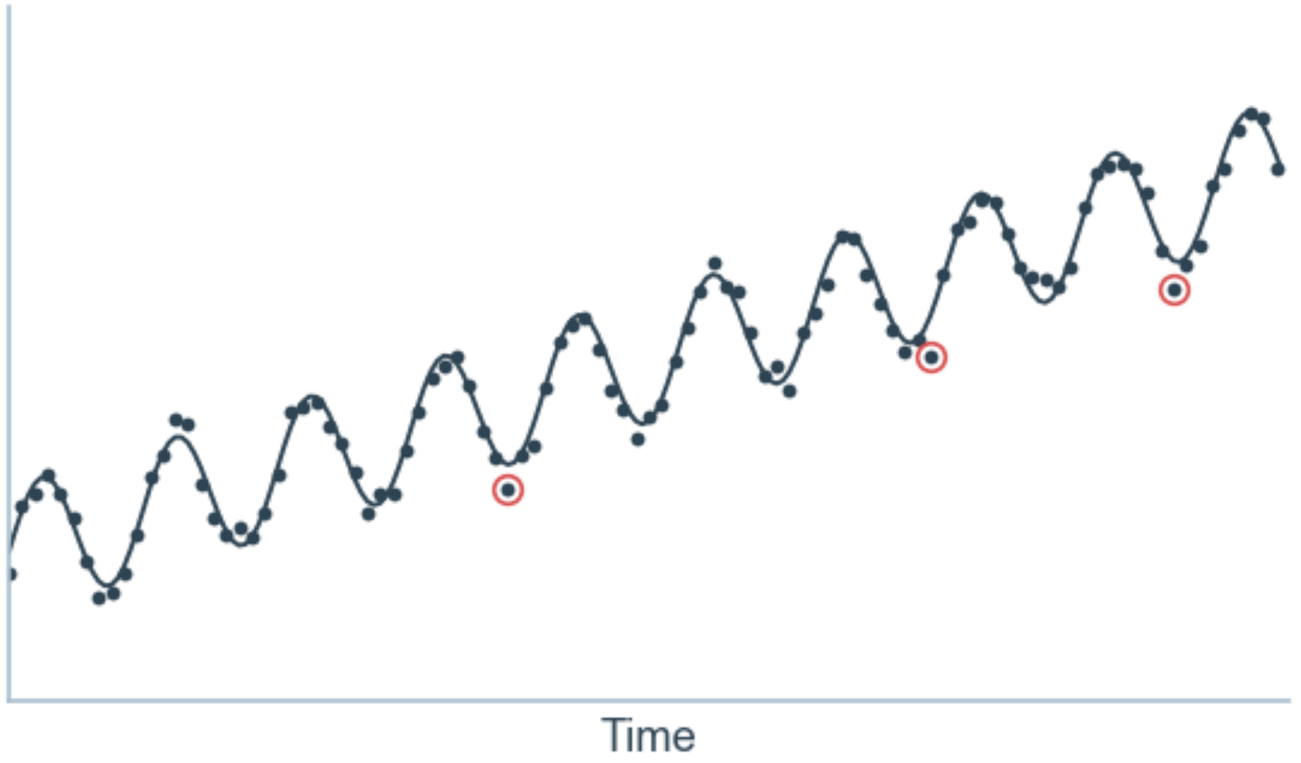

Time

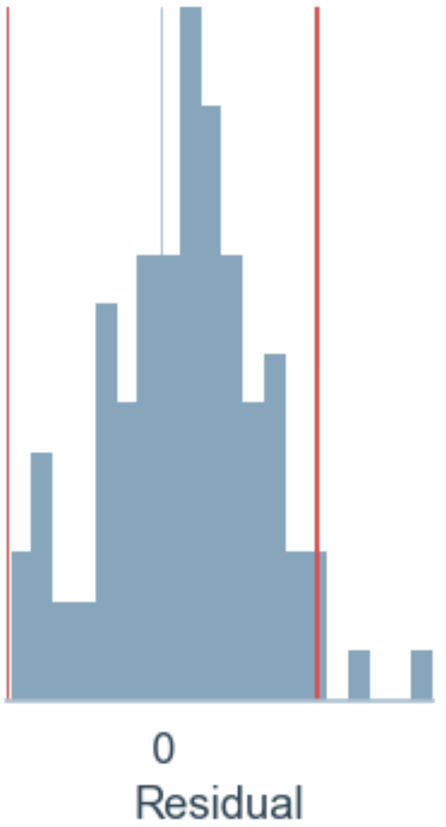

Time


0
Residual


Time


0
Residual


Time

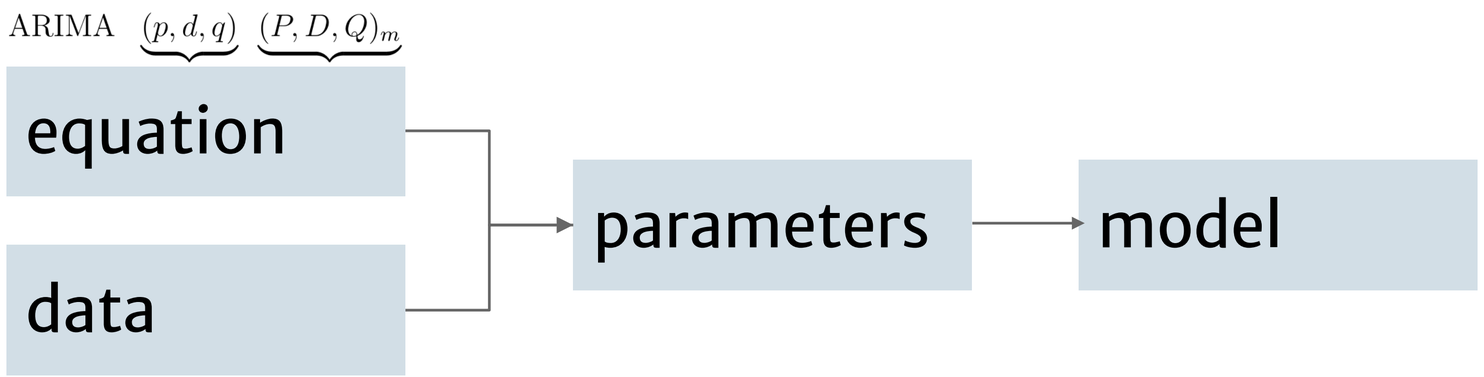

0
Residual

How hard can it be?

- autoregressive integrated moving average (ARIMA)
- fast Fourier transform
- hidden Markov model
- recurrent neural networks
- support vector machines
- ...

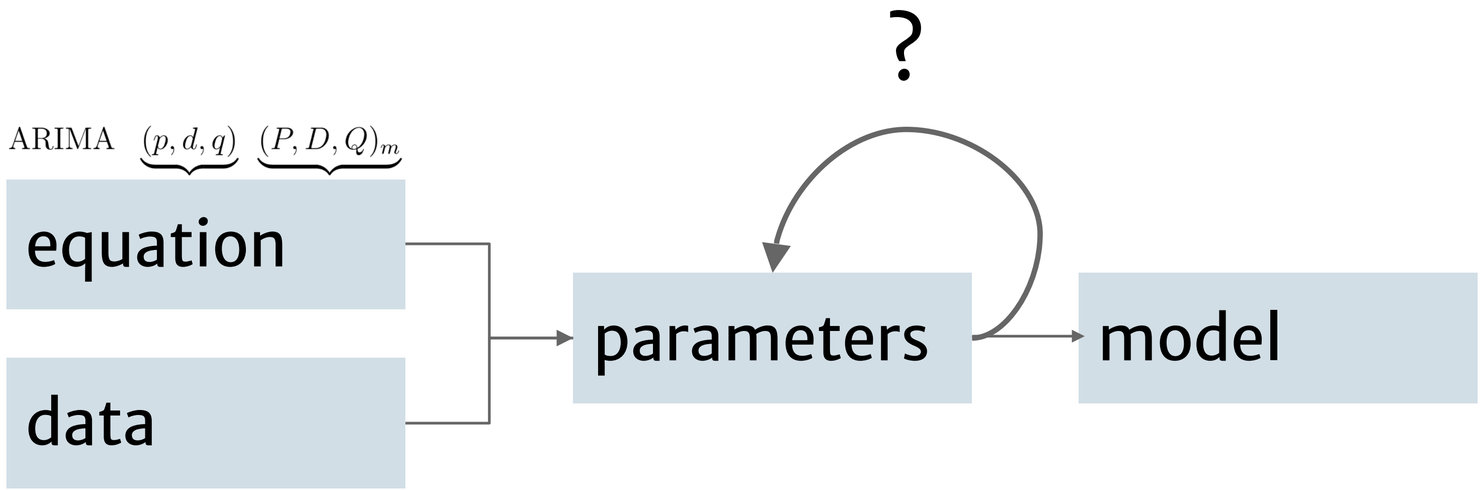
autoregressive integrated moving average

$$\text{ARIMA}\quad \underbrace{(p,d,q)}_{\substack{\uparrow\\ \text{Non-seasonal part}\\ \text{of the model}}}\quad \underbrace{(P,D,Q)_m}_{\substack{\uparrow\\ \text{Seasonal part}\\ \text{of the model}}}$$


ARIMA (p, d, q) $(P, D, Q)_m$
equation
data
parameters
model

Time series analysis when “good enough” is good enough

In deep trouble
The simplicity ladder
The hole is deeper than I thought
Light at the end of the tunnel
The ladder isn’t tall enough
Towards the bright future


?
ARIMA $\underbrace{(p,d,q)}$ $\underbrace{(P,D,Q)_m}$
equation
data
parameters
model

Dashboard

Global Cash sales, Automattic

past15m, by month

Global Average new purchase, WordPress.com

past15m, by week

Global New cash sales, Automattic

past15m, by week

Global New cash sales, WordPress.com

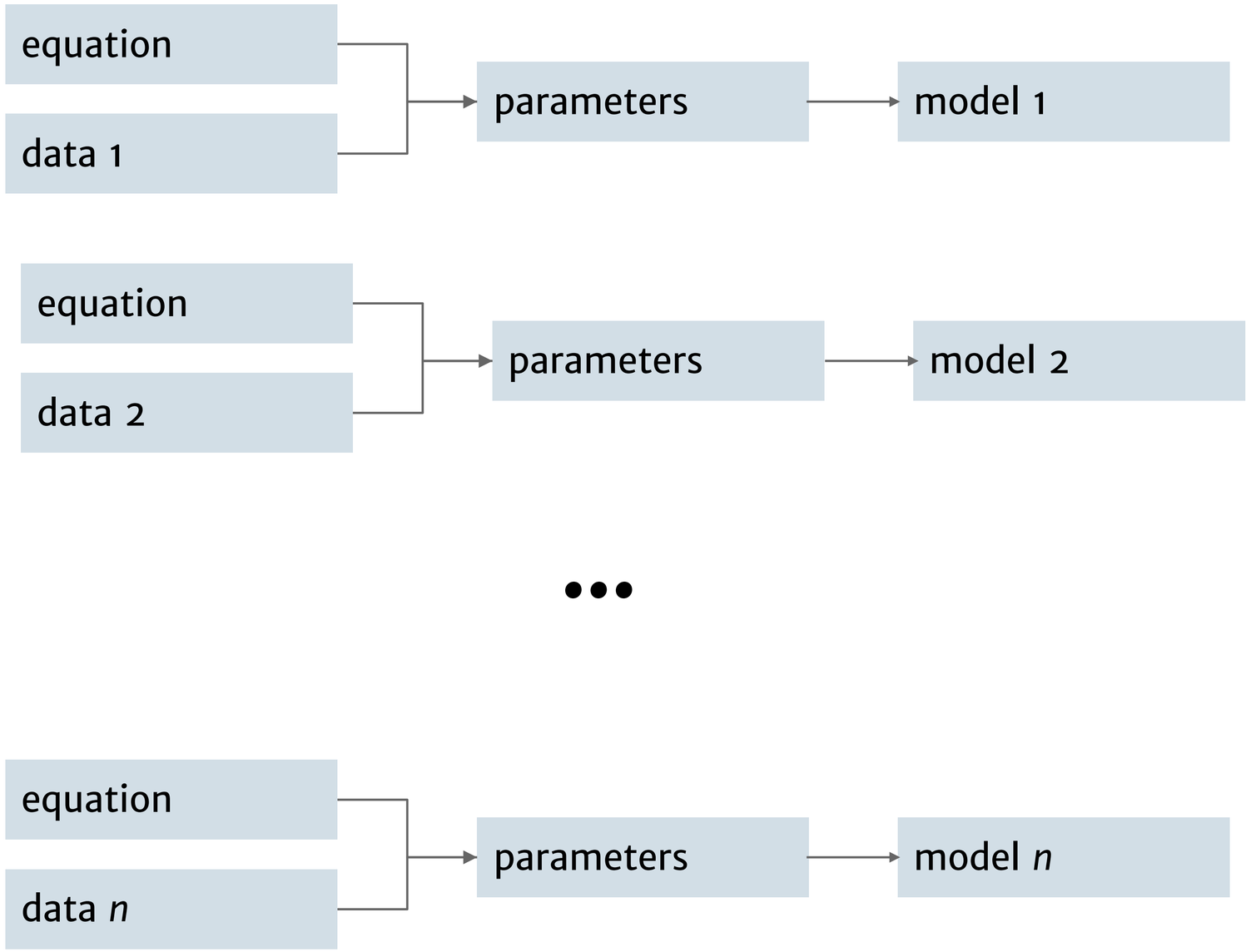
past7d, by day

Global Cash sales, WooCommerce

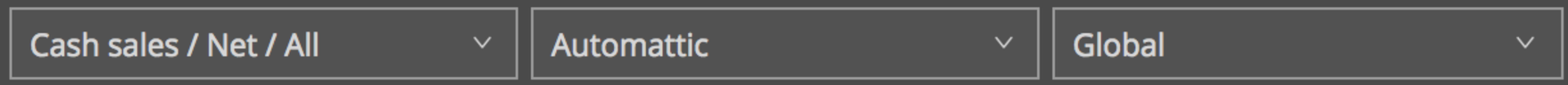
past15m, by week

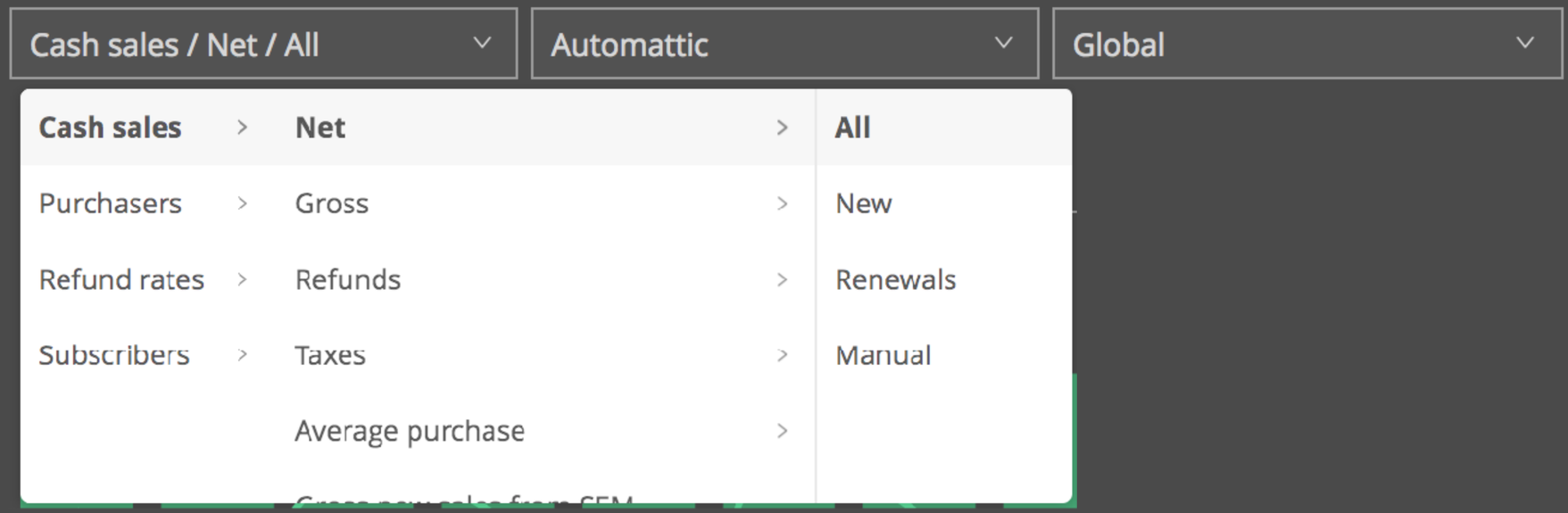
Global Cash sales, .blog

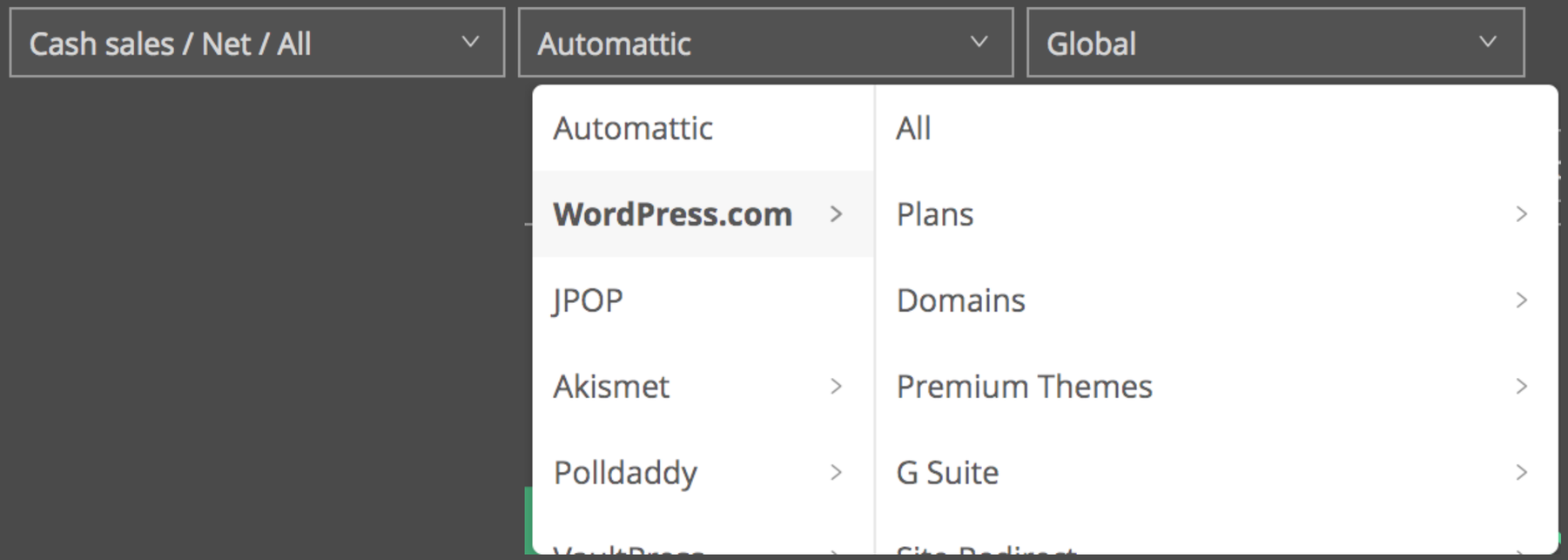
past90d, by day

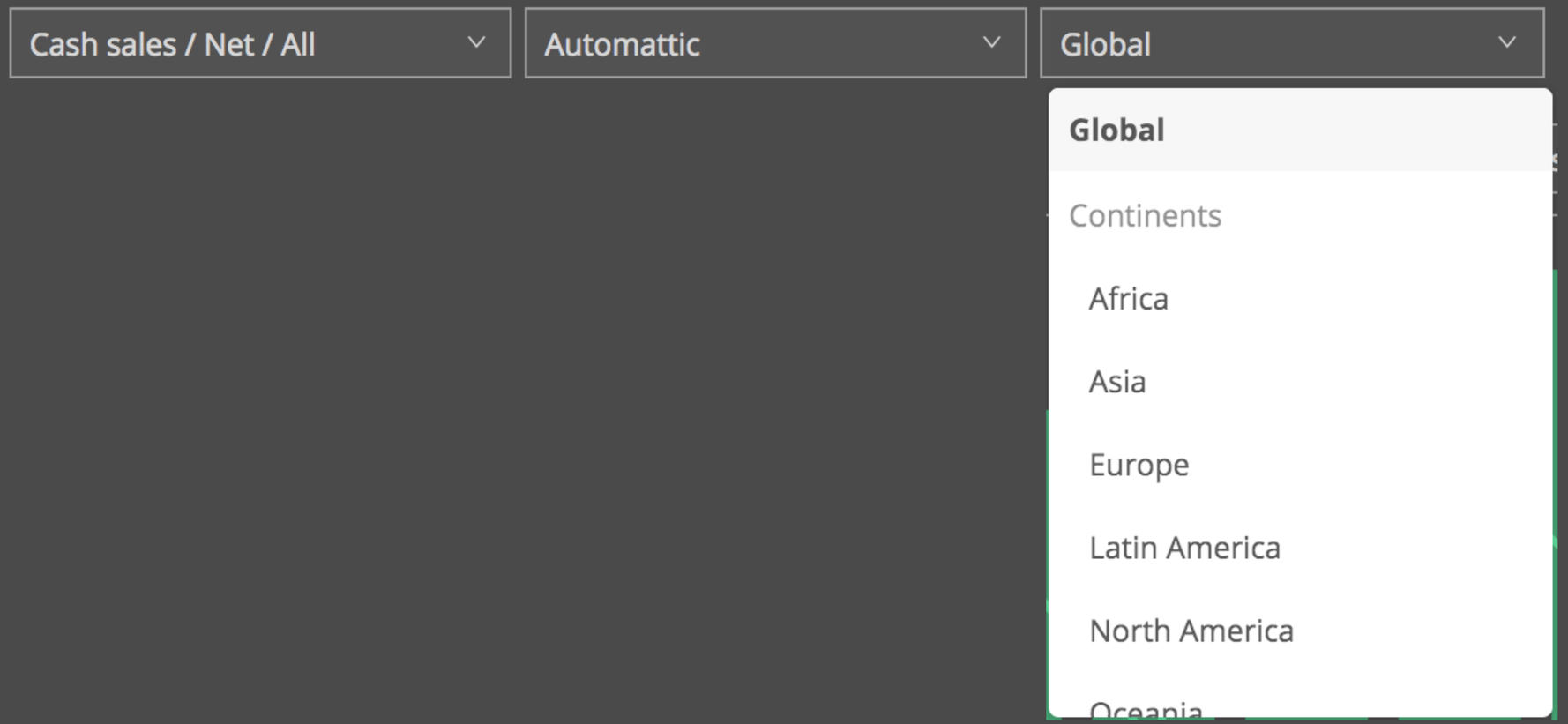

equation
data 1
parameters
model 1
equation
data 2
parameters
model 2
equation
data *n*
parameters
model *n*

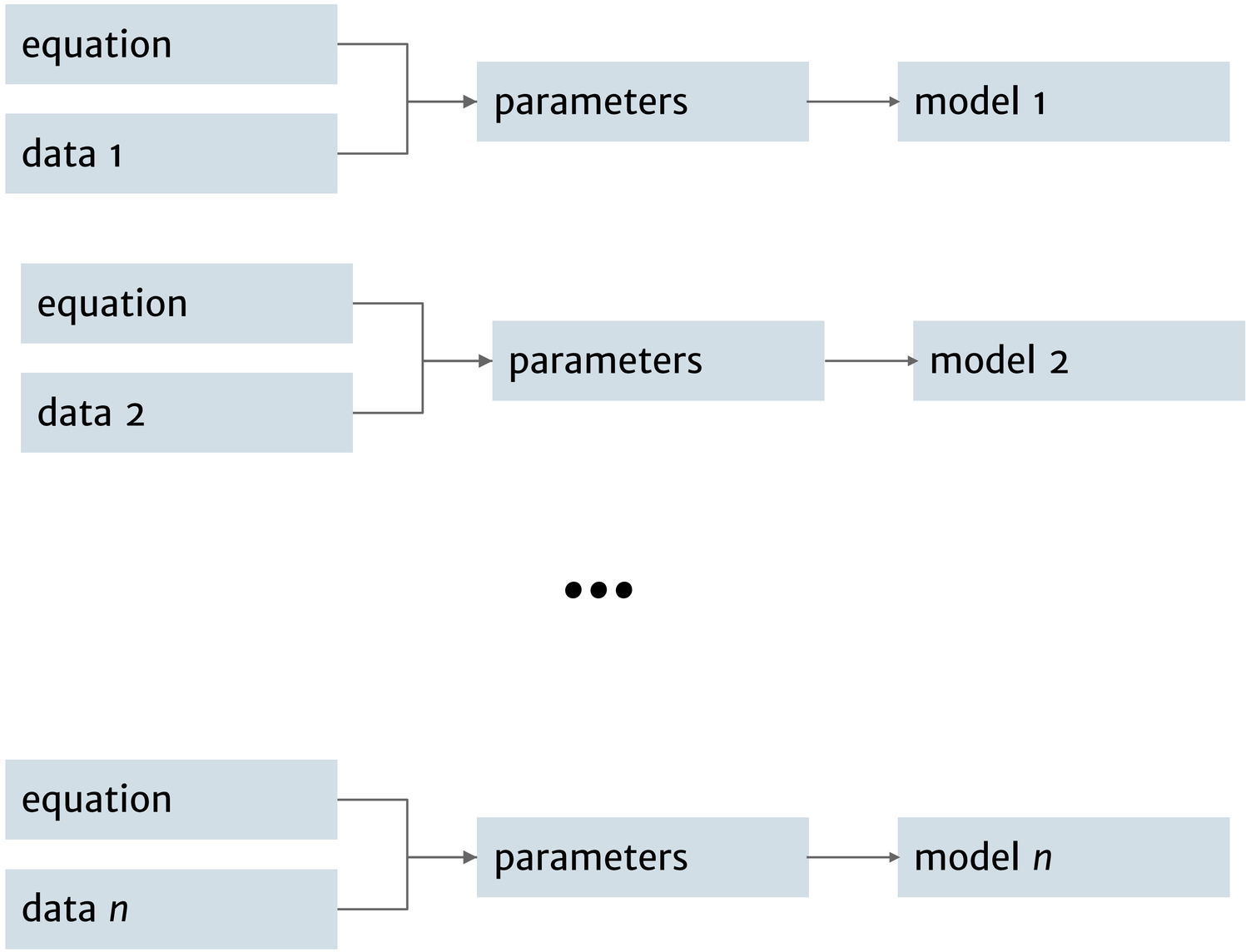
N≅10


Cash sales / Net / All
Automattic
Global

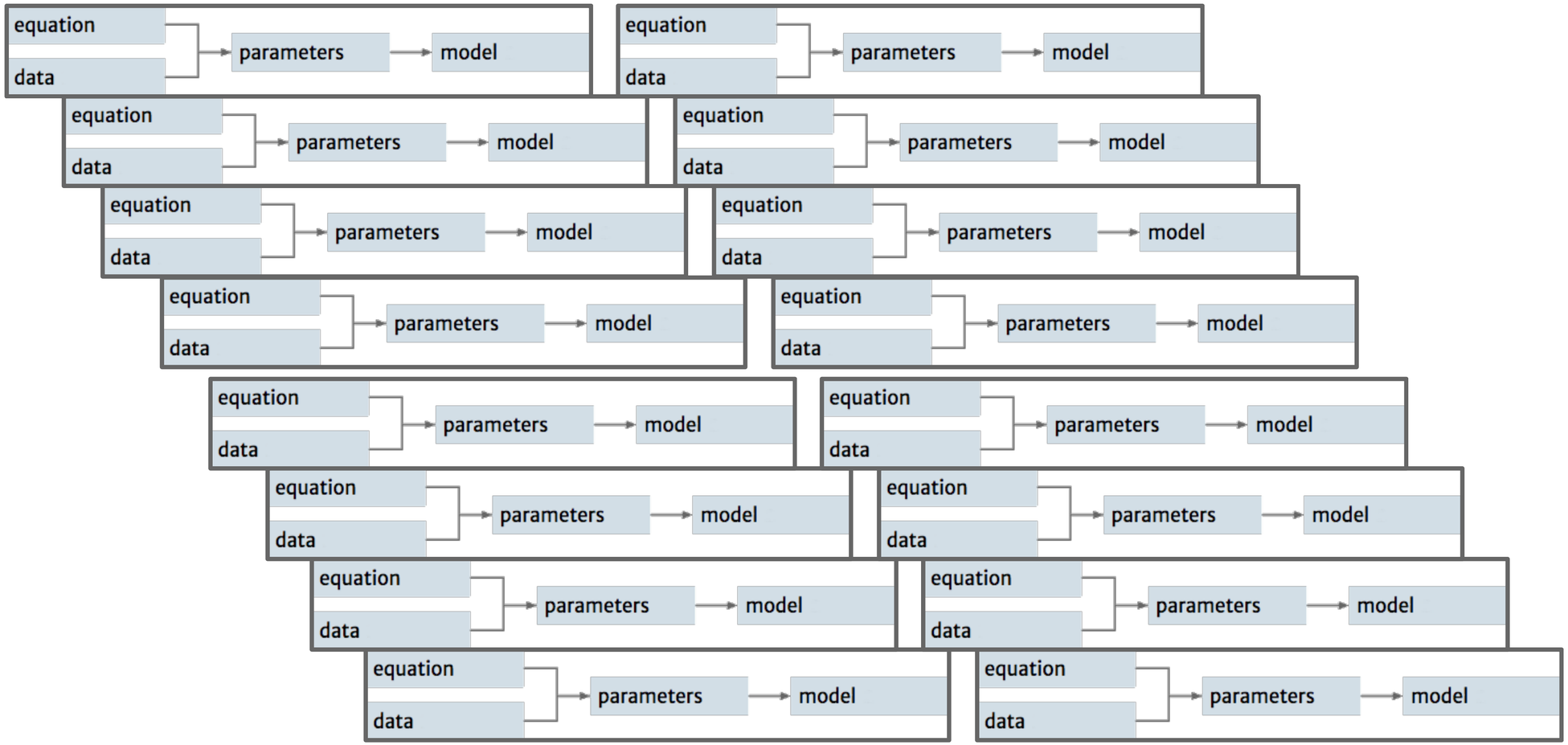

Cash sales / Net / All
Automattic
Global
Cash sales
Purchasers
Refund rates
Subscribers
Net
Gross
Refunds
Taxes
Average purchase
All
New
Renewals
Manual


Cash sales / Net / All
Automattic
Global
Automattic
WordPress.com
JPOP
Akismet
Polldaddy
All
Plans
Domains
Premium Themes
G Suite


Cash sales / Net / All
Automattic
Global
Global
Continents
Africa
Asia
Europe
Latin America
North America


equation
data 1
parameters
model 1
equation
data 2
parameters
model 2
equation
data *n*
parameters
model *n*

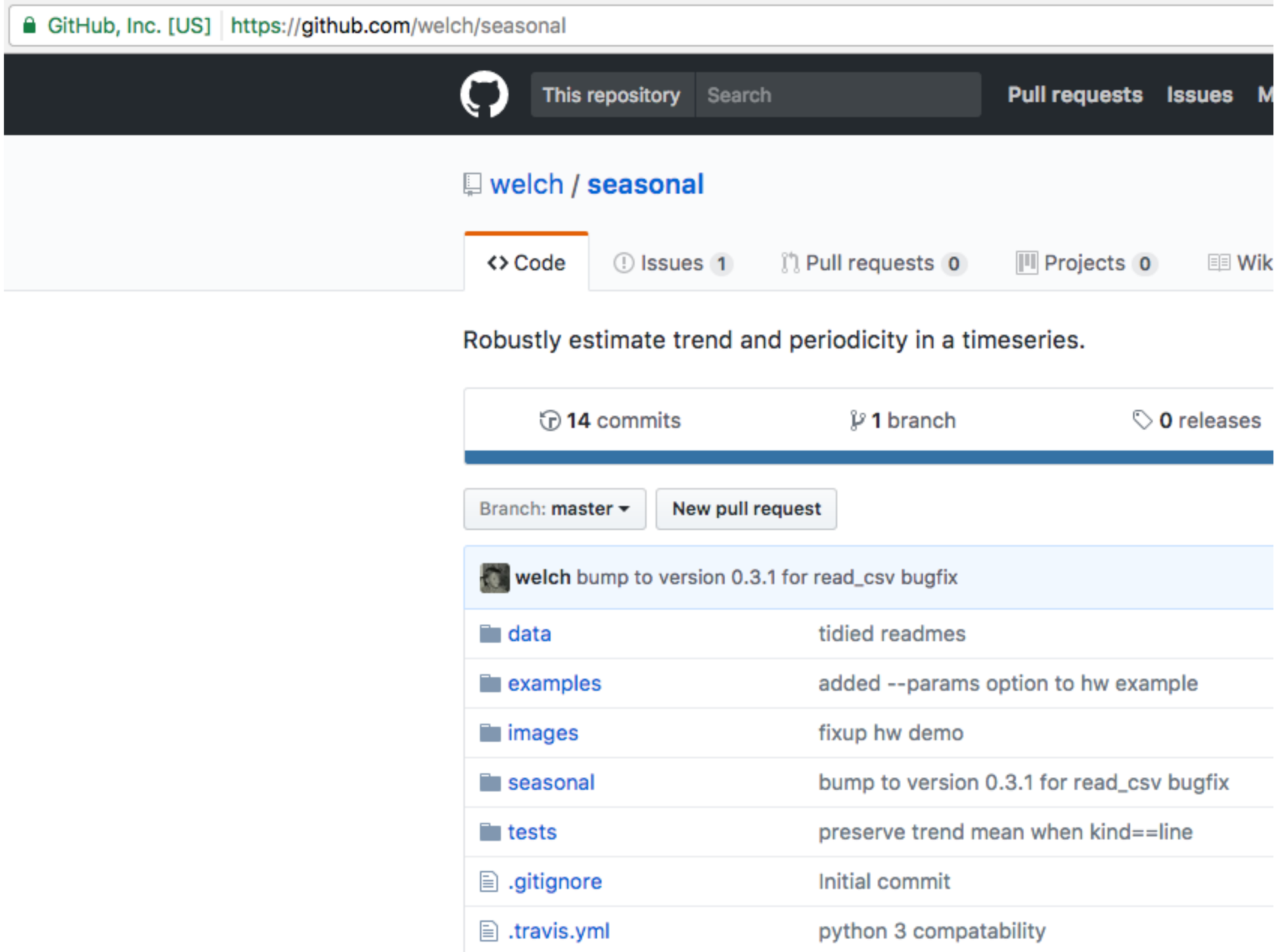
N≅10



N≅1,000

Time series analysis when “good enough” is good enough

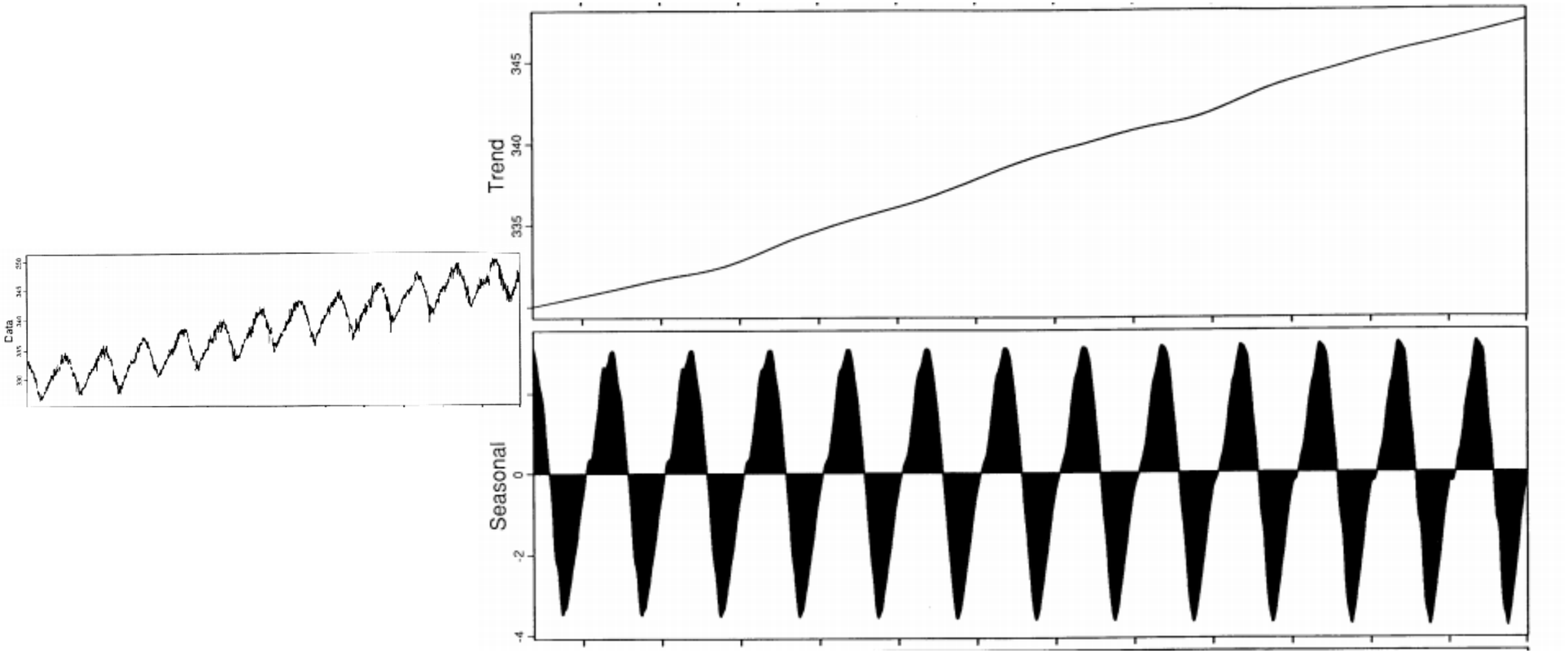
In deep trouble
The simplicity ladder
The hole is deeper than I thought
Light at the end of the tunnel
The ladder isn’t tall enough
Towards the bright future

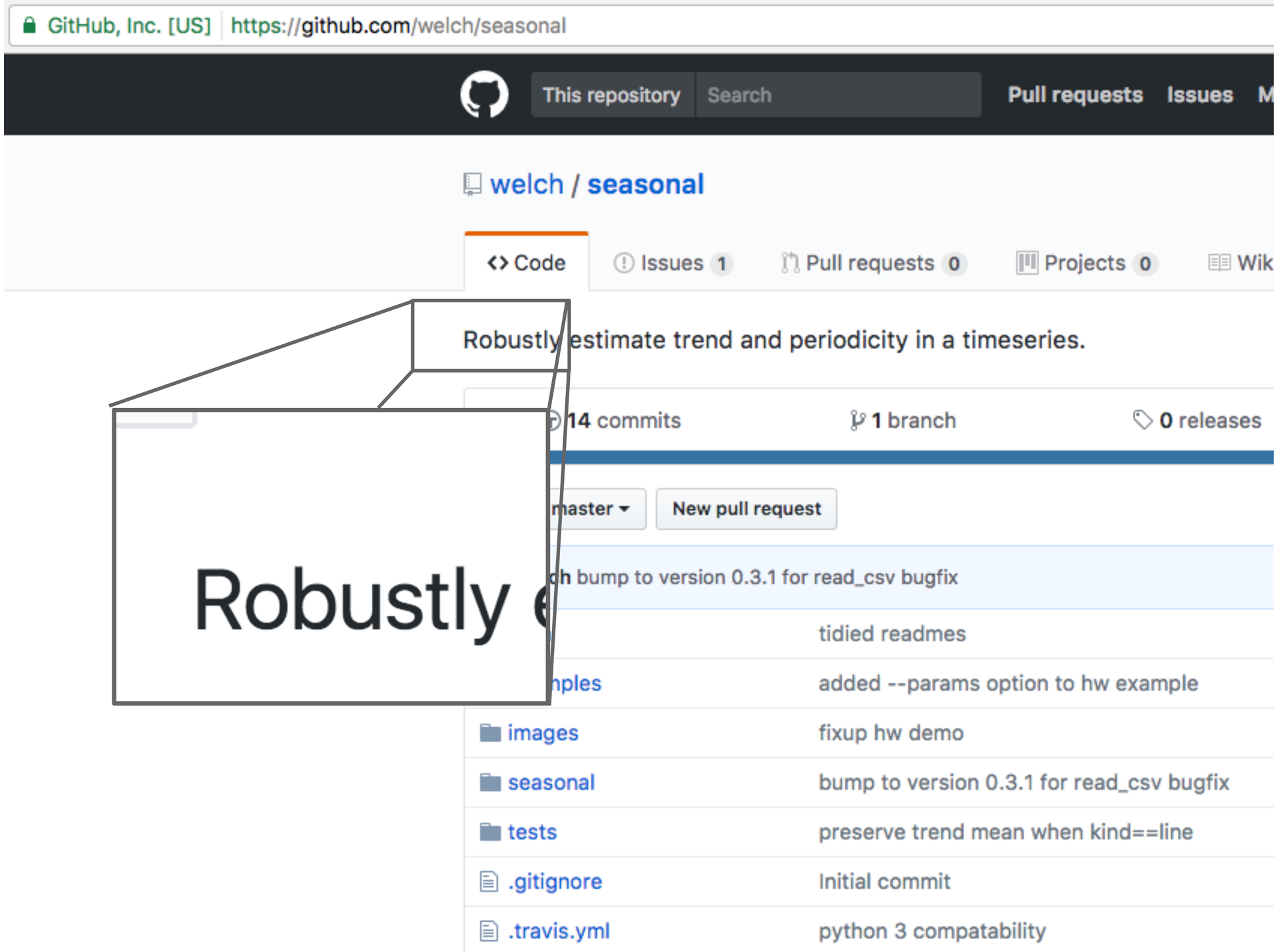

GitHub, Inc. [US]
https://github.com/welch/seasonal
This repository
Search
Pull requests
Issues
welch / seasonal
Code
Issues 1
Pull requests 0
Projects 0
Wik
Robustly estimate trend and periodicity in a timeseries.
14 commits
1 branch
0 releases
Branch: master
New pull request
welch bump to version 0.3.1 for read_csv bugfix
data
tidied readmes
examples
added --params option to hw example
images
fixup hw demo
seasonal
bump to version 0.3.1 for read_csv bugfix
tests
preserve trend mean when kind==line
.gitignore
Initial commit
.travis.yml
python 3 compatability

Journal of Official Statistics
Vol. 6. No. 1, 1990, pp. 3–73

STL: A Seasonal-Trend Decomposition Procedure Based on Loess

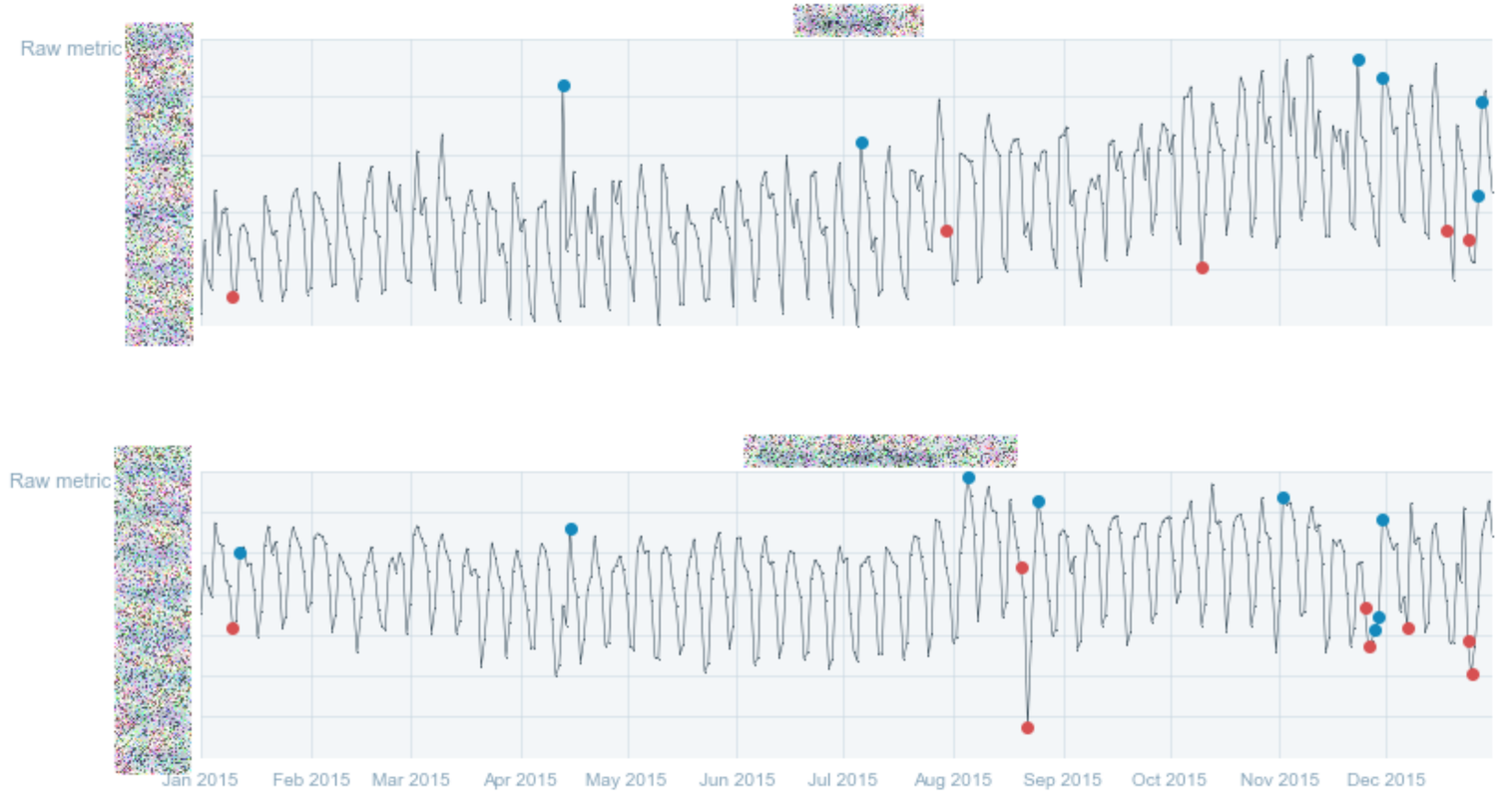
Robert B. Cleveland,[1] William S. Cleveland,[2] Jean E. McRae,[2] and Irma Terpenning[2]




GitHub, Inc. [US]
https://github.com/welch/seasonal
This repository
Search
Pull requests
Issues
welch / seasonal
Code
Issues 1
Pull requests 0
Projects 0
Robustly estimate trend and periodicity in a timeseries.
14 commits
1 branch
0 releases
master
New pull request
bump to version 0.3.1 for read_csv bugfix
tidied readmes
added --params option to hw example
images
fixup hw demo
seasonal
bump to version 0.3.1 for read_csv bugfix
tests
preserve trend mean when kind==line
.gitignore
Initial commit
.travis.yml
python 3 compatability
Robustly

```python
class SeasonalModel:
    def __init__(self, seasons, trend):
        self.seasons = seasons
        self.trend = trend

    def to_dict(self):
        if self.seasons is None:
            seasons = None
        else:
            seasons = list(self.seasons)
        if self.trend is None:
            trend = None
        else:
            trend = list(self.trend)
        return {'seasons': seasons, 'trend': trend}

    def copy(self):
        seasons = self.seasons[:] if self.seasons is not None else None
        trend = self.trend[:] if self.trend is not None else None
        return SeasonalModel(seasons=seasons, trend=trend)


class TimeseriesEstimator_seasonal(TimeseriesEstimator):
    """
    Use the `seasonal`[1] module to model the data. Assess anomaly using the residuals

    [1]: https://github.com/welch/seasonal

    """


    def __init__(self, points_per_day, period_days, history_days, trend_forecast_method="average", model=None, history=No

        super(TimeseriesEstimator_seasonal, self).__init__(
            points_per_day=points_per_day, history_days=history_days,
            model=model, history=history, residuals=residuals, scores=scores
        )
        # call to `super` above asserts that proper values of `history`, `residuals`,
        # and `scores`. In this case, we need to add `model` to the similar constraint.
        # Unlike the previous case, here we will test the arguments to the
        # constructor, and not the actual object memebrs.
        optional_arguments = ('history', 'residuals', 'model', 'scores')
        optional_defined = [locals()[a] is not None for a in optional_arguments]
        if np.any(optional_defined):
            assert np.all(optional_defined)
```

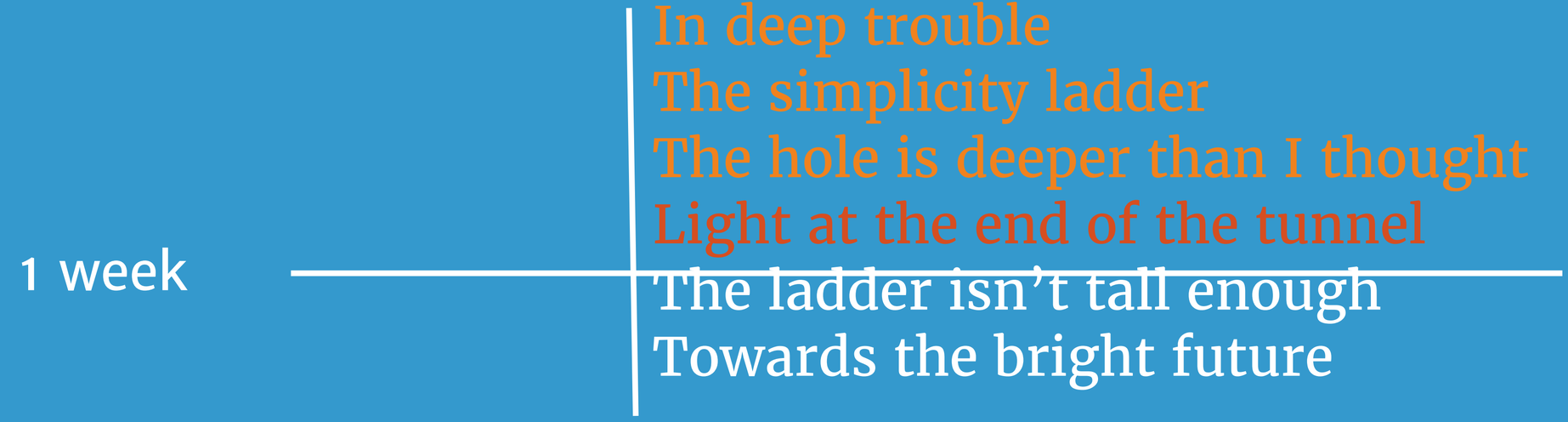
In [18]:
```
fig, axes = plt.subplots(2, 1, figsize=(15, 7), sharex=True)
for metric, ax in zip(metric_names, axes):
    visualize_model(
        timestamps=np.array(tbl_2015.index),
        obs=tbl_2015[metric], scores=scores_2015[metric],
        threshold=0.999, str_title=metric, ax=ax
    )
figdisp(fig, title="Labeled data 2015")
```



a2f2

anomaly analysis & future forecast

Time series analysis when “good enough” is good enough



¯_(ツ)_/¯

Time series analysis when “good enough” is good enough

In deep trouble
The simplicity ladder
The hole is deeper than I thought
Light at the end of the tunnel
The ladder isn’t tall enough
Towards the bright future

Welcome to Bokeh ¶

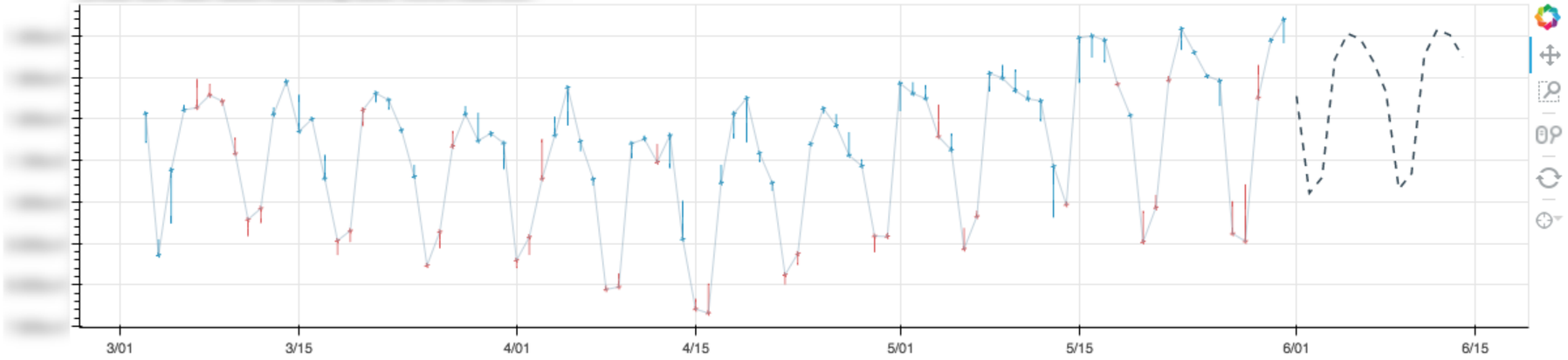
Bokeh is a Python interactive visualization library that targets modern web browsers for presentation. Its goal is to provide elegant, concise construction of novel graphics in the style of D3.js, and to extend this capability with high-performance interactivity over very large or streaming datasets. Bokeh can help anyone who would like to quickly and easily create interactive plots, dashboards, and d applications.

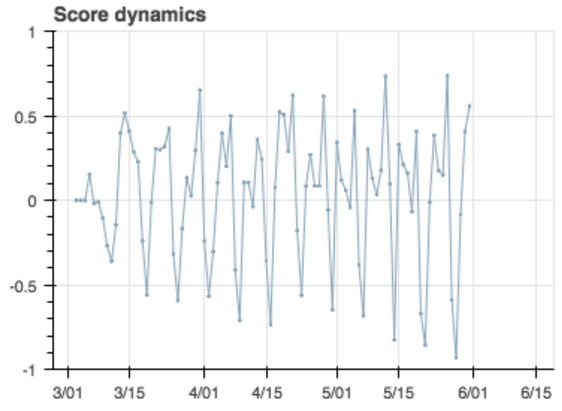
To get started using Bokeh to make your visualizations, see the User Guide.

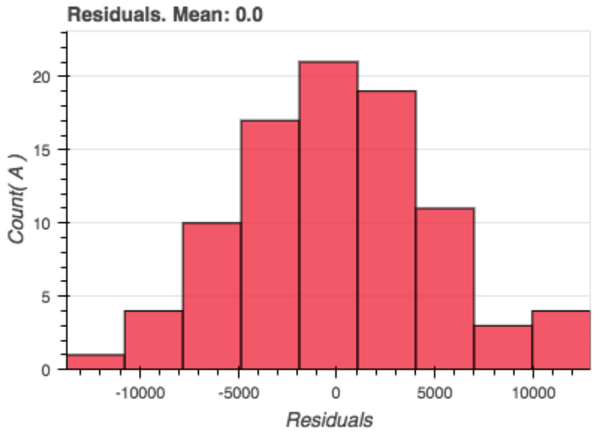
To see examples of how you might use Bokeh with your own data, check out the Gallery.

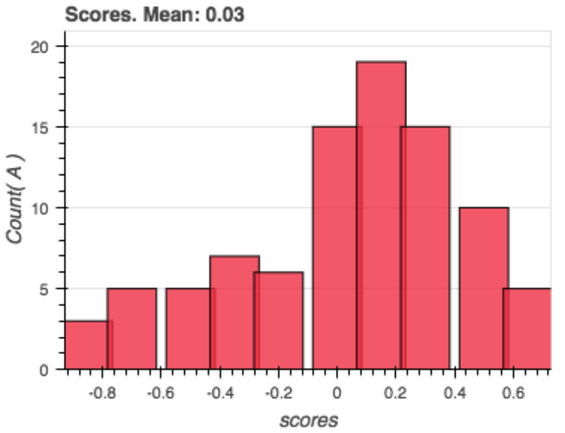
A complete API reference of Bokeh is at Reference Guide.

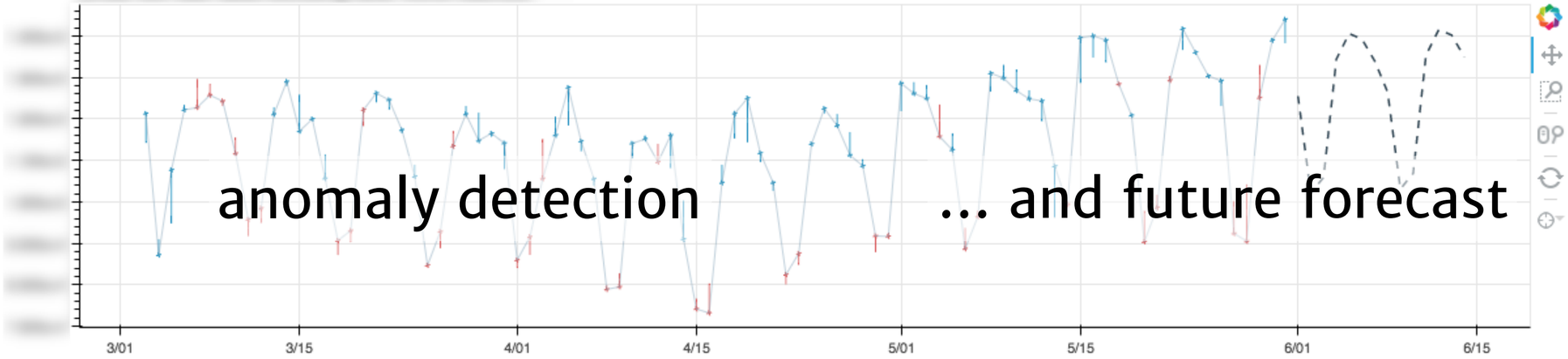
If you are interested in contributing to Bokeh, or extending the library, see the Developer Guide.

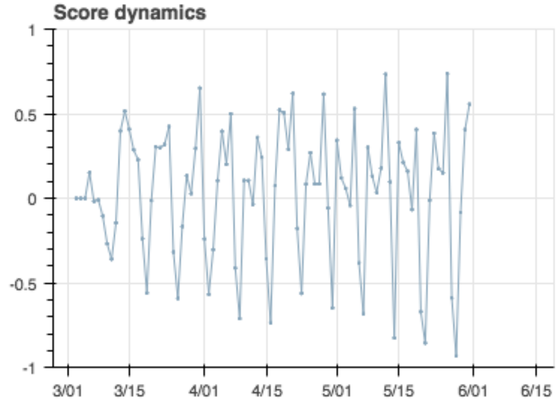

3/01
3/15
4/01
4/15
5/01
5/15
6/01
6/15

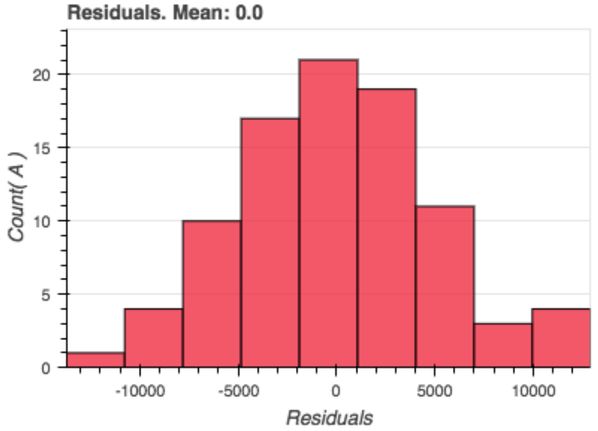

Score dynamics
1
0.5
0
-0.5
-1
3/01
3/15
4/01
4/15
5/01
5/15
6/01
6/15

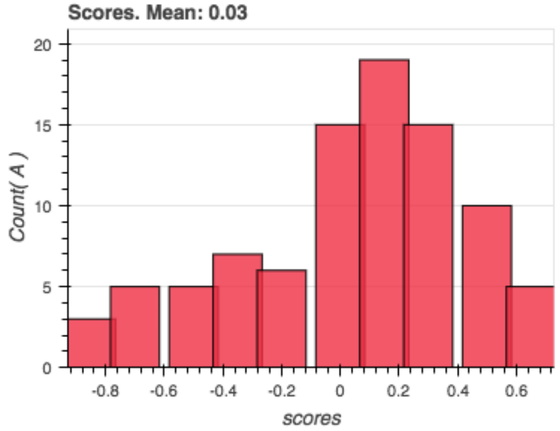

Residuals. Mean: 0.0
Count(A)
20
15
10
5
0
-10000
-5000
0
5000
10000
Residuals

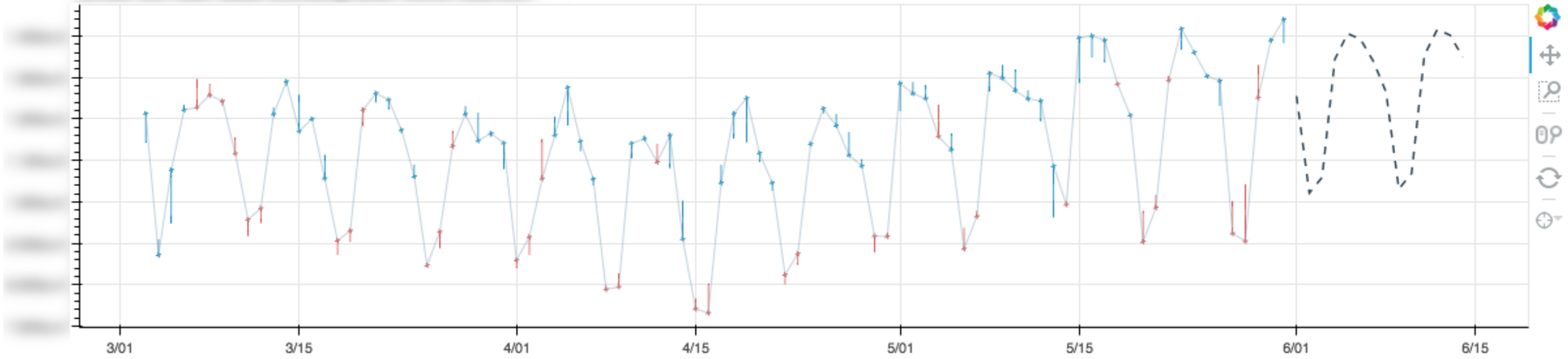

Scores. Mean: 0.03
Count(A)
20
15
10
5
0
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
scores

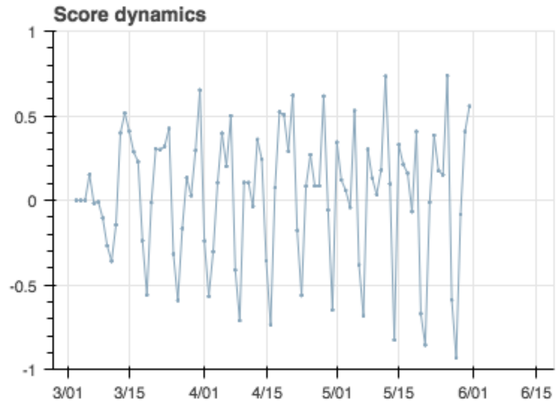

anomaly detection
... and future forecast
3/01
3/15
4/01
4/15
5/01
5/15
6/01
6/15

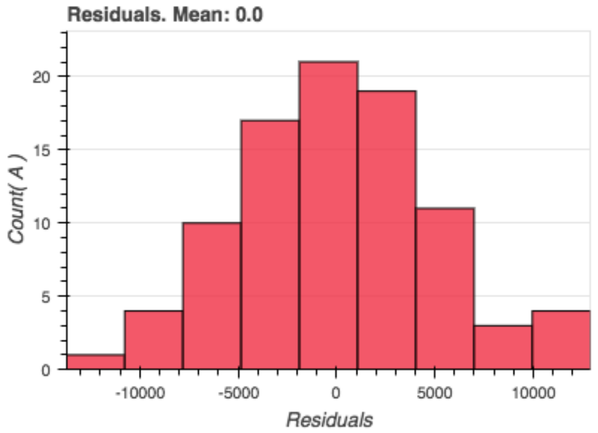

Score dynamics
1
0.5
0
-0.5
-1
3/01
3/15
4/01
4/15
5/01
5/15
6/01
6/15

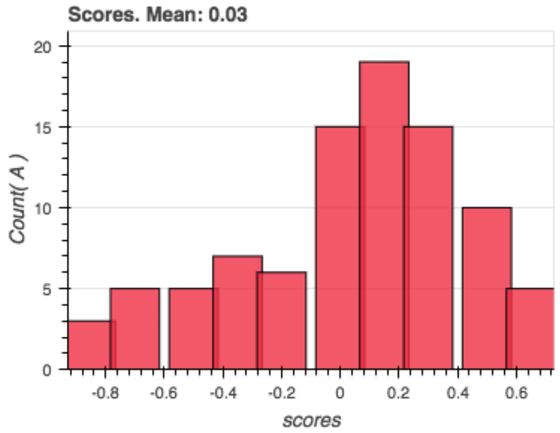

Residuals. Mean: 0.0
Count(A)
20
15
10
5
0
-10000
-5000
0
5000
10000
Residuals


Scores. Mean: 0.03
Count(A)
20
15
10
5
0
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
scores

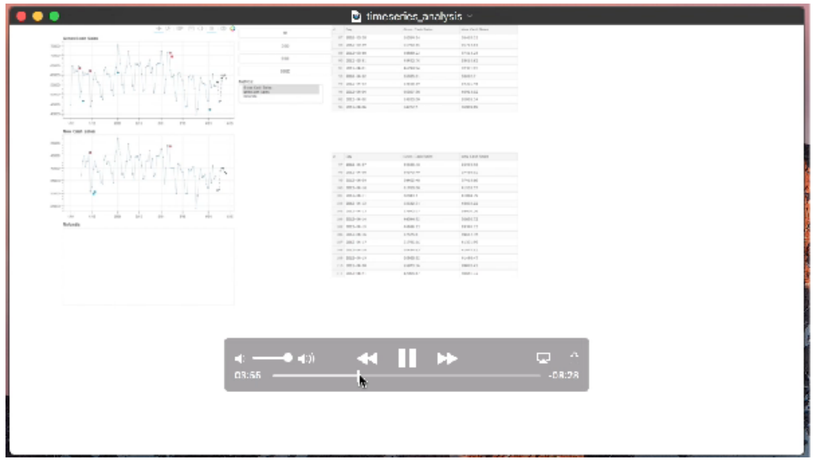

3/01
3/15
4/01
4/15
5/01
5/15
6/01
6/15

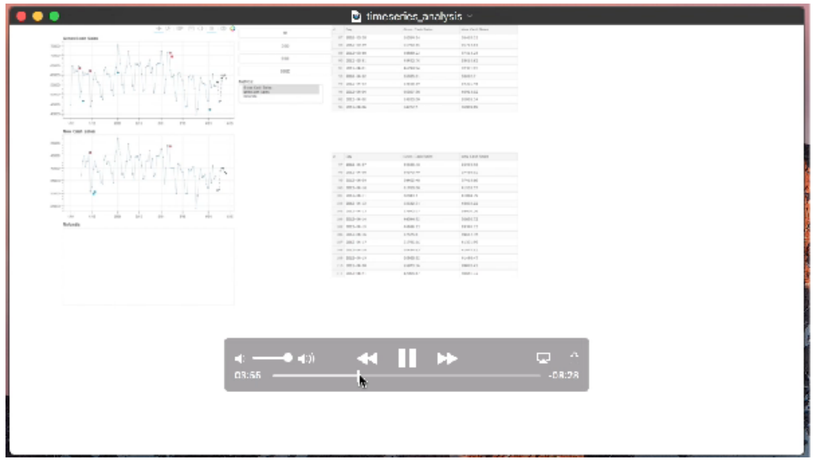

Score dynamics
1
0.5
0
-0.5
-1
3/01
3/15
4/01
4/15
5/01
5/15
6/01
6/15

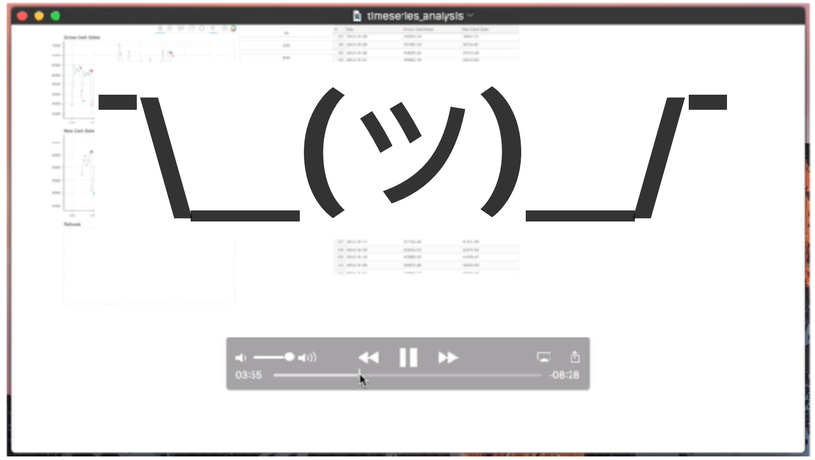

Residuals. Mean: 0.0
Count(A)
20
15
10
5
0
-10000
-5000
0
5000
10000
Residuals

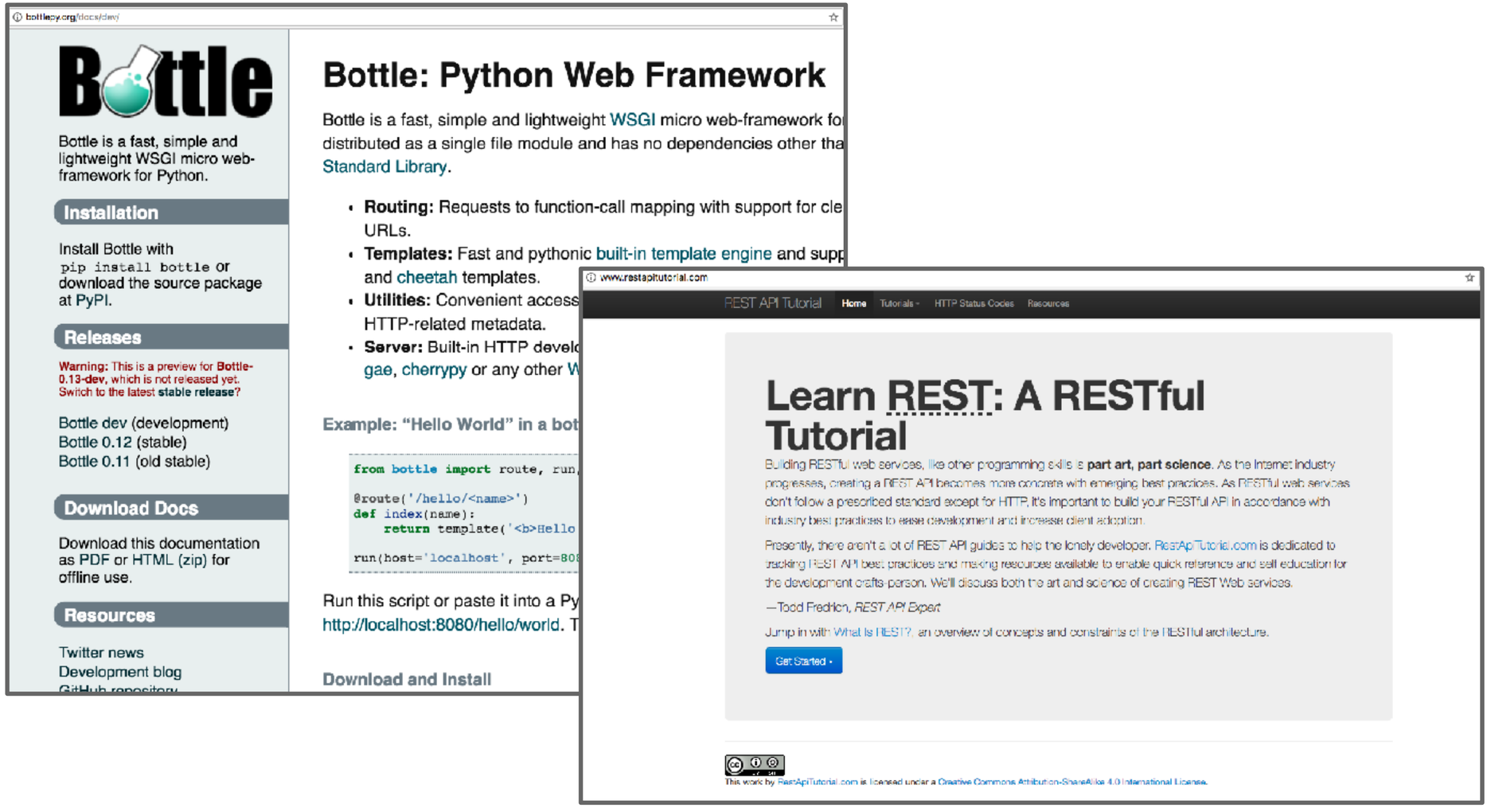

Scores. Mean: 0.03
Count(A)
20
15
10
5
0
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
scores

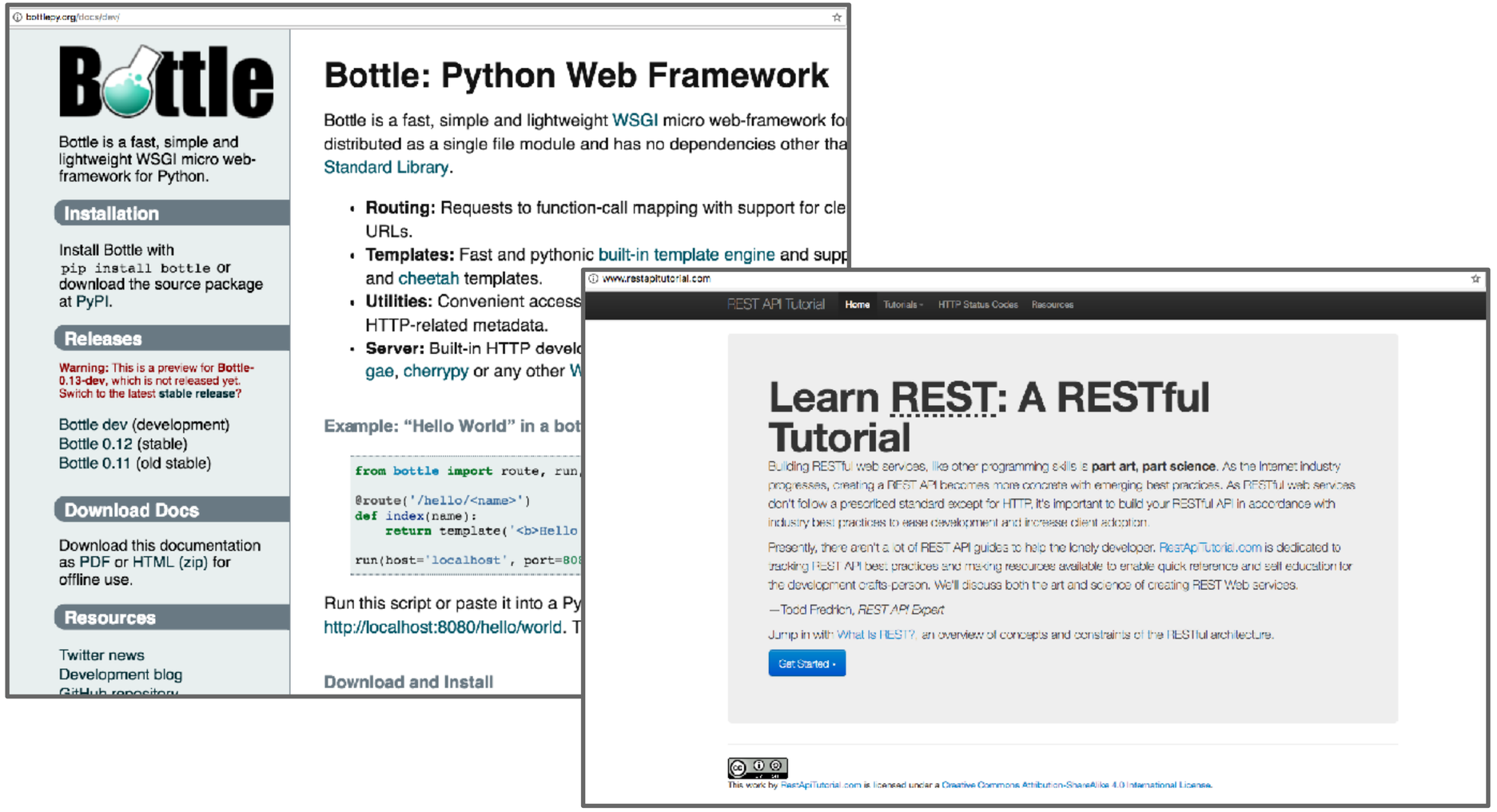
¯_(ツ)_/¯


timeseries_analysis

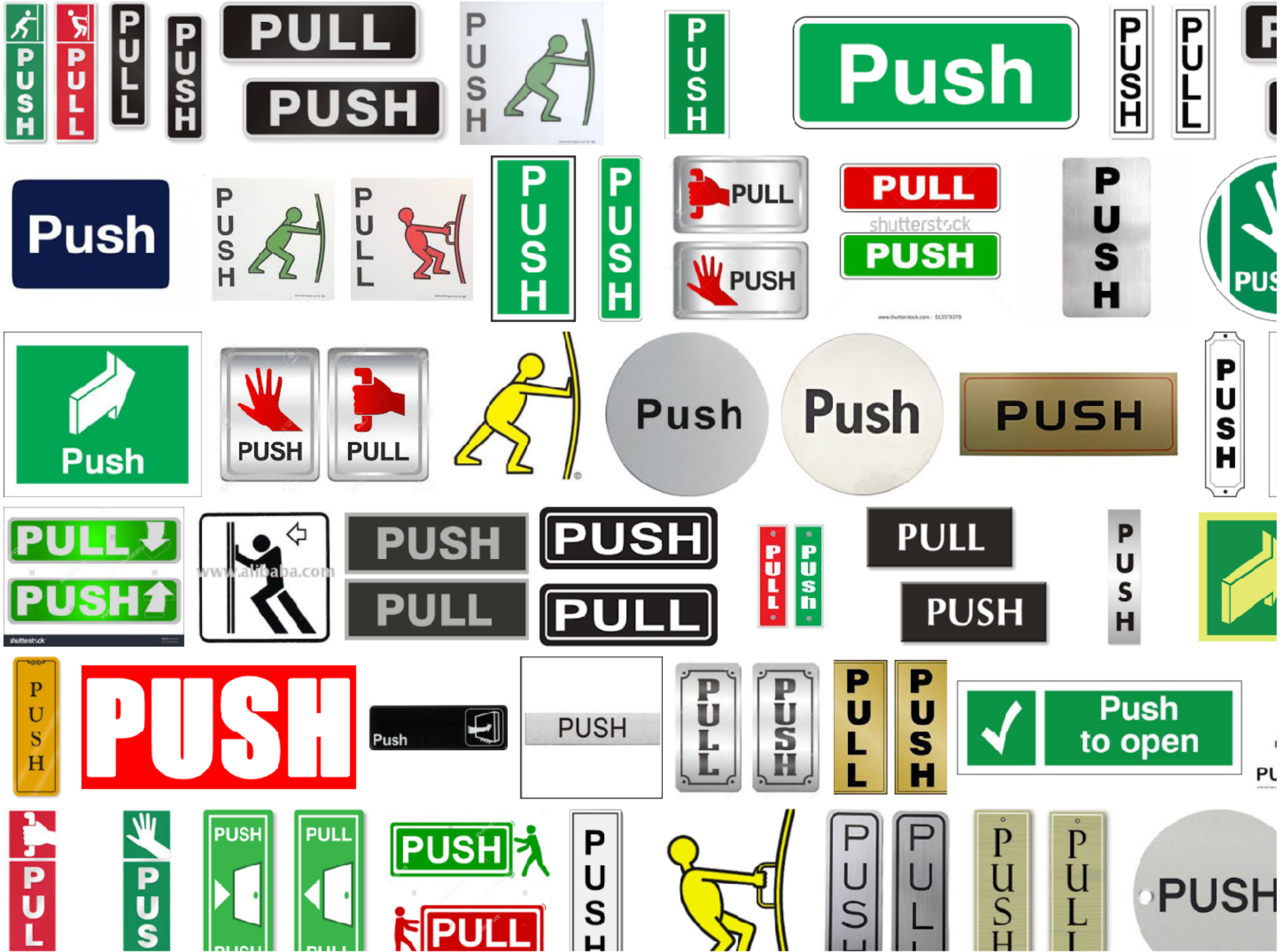

timeseries_analysis

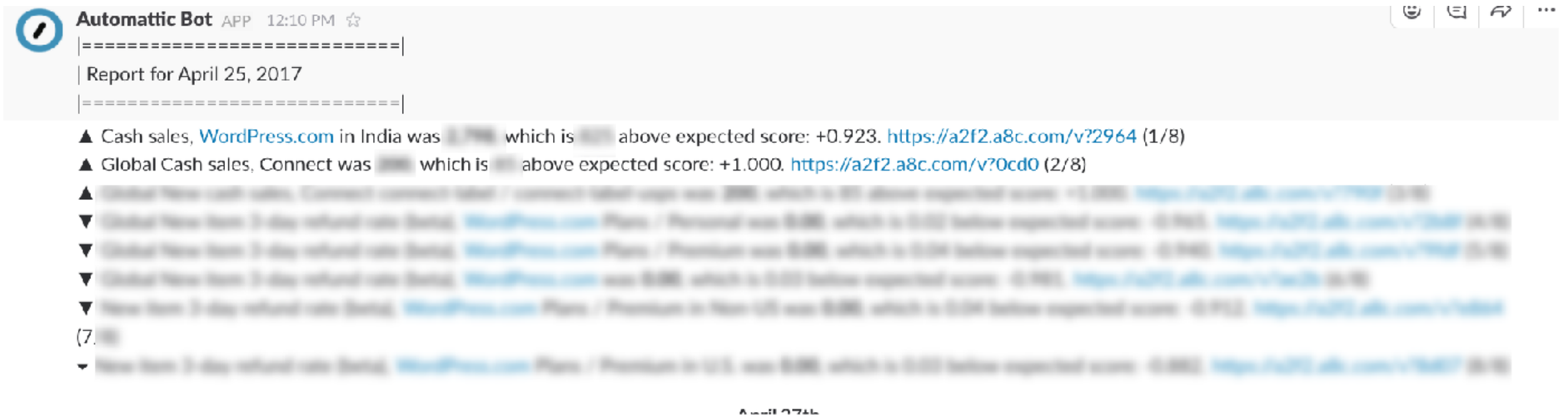

timeseries_analysis
¯_(ツ)_/¯
03:55
-08:18


bottlepy.org/docs/dev/
Bottle
Bottle is a fast, simple and lightweight WSGI micro web-framework for Python.
Installation
Install Bottle with pip install bottle or download the source package at PyPI.
Releases
Warning: This is a preview for Bottle-0.13-dev, which is not released yet. Switch to the latest stable release?
Bottle dev (development)
Bottle 0.12 (stable)
Bottle 0.11 (old stable)
Download Docs
Download this documentation as PDF or HTML (zip) for offline use.
Resources
Twitter news
Development blog
Bottle: Python Web Framework
Bottle is a fast, simple and lightweight WSGI micro web-framework fo
distributed as a single file module and has no dependencies other tha
Standard Library.
Routing: Requests to function-call mapping with support for cle
URLs.
Templates: Fast and pythonic built-in template engine and supp
and cheetah templates.
Utilities: Convenient access
HTTP-related metadata.
Server: Built-in HTTP develo
gae, cherrypy or any other W
Example: "Hello World" in a bot
from bottle import route, run,
@route('/hello/<name>')
def index(name):
return template('<b>Hello
run(host='localhost', port=80
Run this script or paste it into a Py
http://localhost:8080/hello/world. T
Download and Install
www.restapitutorial.com
REST API Tutorial
Home
Tutorials
HTTP Status Codes
Resources
Learn REST: A RESTful Tutorial
Building RESTful web services, like other programming skills is part art, part science. As the Internet industry progresses, creating a REST API becomes more concrete with emerging best practices. As RESTful web services don't follow a prescribed standard except for HTTP, it's important to build your RESTful API in accordance with industry best practices to ease development and increase client adoption.
Presently, there aren't a lot of REST API guides to help the lonely developer. RestApiTutorial.com is dedicated to tracking REST API best practices and making resources available to enable quick reference and self education for the development crafts-person. We'll discuss both the art and science of creating REST Web services.
—Todd Fredrich, REST API Expert
Jump in with What is REST?, an overview of concepts and constraints of the RESTful architecture.
Get Started
This work by RestApiTutorial.com is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.


bottlepy.org/docs/dev/
Bottle
Bottle is a fast, simple and lightweight WSGI micro web-framework for Python.
Installation
Install Bottle with pip install bottle or download the source package at PyPI.
Releases
Warning: This is a preview for Bottle-0.13-dev, which is not released yet. Switch to the latest stable release?
Bottle dev (development)
Bottle 0.12 (stable)
Bottle 0.11 (old stable)
Download Docs
Download this documentation as PDF or HTML (zip) for offline use.
Resources
Twitter news
Development blog
Bottle: Python Web Framework
Bottle is a fast, simple and lightweight WSGI micro web-framework fo
distributed as a single file module and has no dependencies other tha
Standard Library.
Routing: Requests to function-call mapping with support for cle
URLs.
Templates: Fast and pythonic built-in template engine and supp
and cheetah templates.
Utilities: Convenient access
HTTP-related metadata.
Server: Built-in HTTP develo
gae, cherrypy or any other W
Example: "Hello World" in a bot
from bottle import route, run,
@route('/hello/<name>')
def index(name):
return template('<b>Hello
run(host='localhost', port=80
Run this script or paste it into a Py
http://localhost:8080/hello/world. T
Download and Install
www.restapitutorial.com
REST API Tutorial
Home
Tutorials
HTTP Status Codes
Resources
Learn REST: A RESTful Tutorial
Building RESTful web services, like other programming skills is part art, part science. As the Internet industry progresses, creating a REST API becomes more concrete with emerging best practices. As RESTful web services don't follow a prescribed standard except for HTTP, it's important to build your RESTful API in accordance with industry best practices to ease development and increase client adoption.
Presently, there aren't a lot of REST API guides to help the lonely developer. RestApiTutorial.com is dedicated to tracking REST API best practices and making resources available to enable quick reference and self education for the development crafts-person. We'll discuss both the art and science of creating REST Web services.
—Todd Fredrich, REST API Expert
Jump in with What is REST?, an overview of concepts and constraints of the RESTful architecture.
Get Started
This work by RestApiTutorial.com is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

¯_(ツ)_/¯


PUSH
PULL
PULL
PUSH
PULL
PUSH
PUSH
PUSH
Push
PUSH
PULL
Push
PUSH
PULL
PUSH
PUSH
PULL
PUSH
PULL
shutterstock
PUSH
PUSH
Push
PUSH
PULL
Push
Push
PUSH
PUSH
PULL
PUSH
shutterstock
www.alibaba.com
PUSH
PULL
PUSH
PULL
PULL
PUSH
PULL
PUSH
PUSH
PUSH
PUSH
Push
PUSH
PULL
PUSH
PULL
PUSH
Push
to open
PUSH
PULL
PUSH
PULL


Automattic Bot APP 12:10 PM
|==========|
| Report for April 25, 2017
|==========|
▲ Cash sales, WordPress.com in India was which is above expected score: +0.923. https://a2f2.a8c.com/v?2964 (1/8)
▲ Global Cash sales, Connect was which is above expected score: +1.000. https://a2f2.a8c.com/v?0cd0 (2/8)

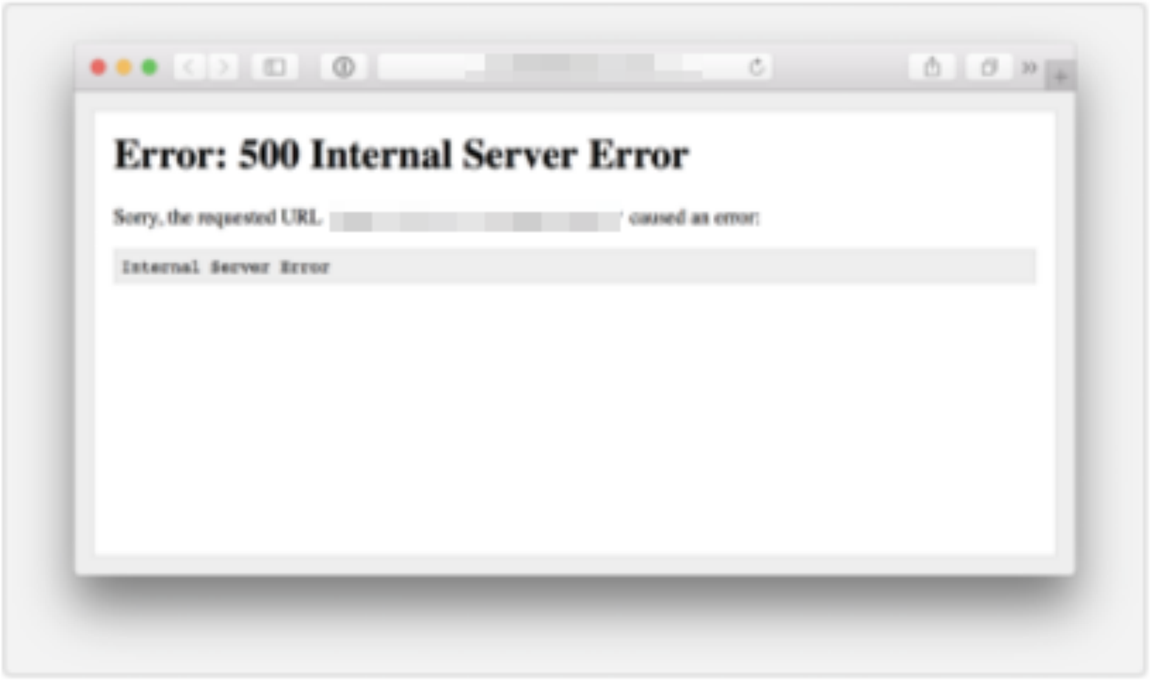
¯_(ツ)_/¯

Stu West 12:13 AM

@boris is a2f2 offline? i haven't gotten any alerts in a few days, #scorecard-a2f2 doesn't seem to have gotten any since 2:05am UTC on 9 May, #scorecard-a2f2-debug only has process messages no actual alerts, and older alert links give me (edited)

Stu West 12:13 AM

uploaded this image: **Screen Shot 2017-05-11 at 2.12.52 PM.png**

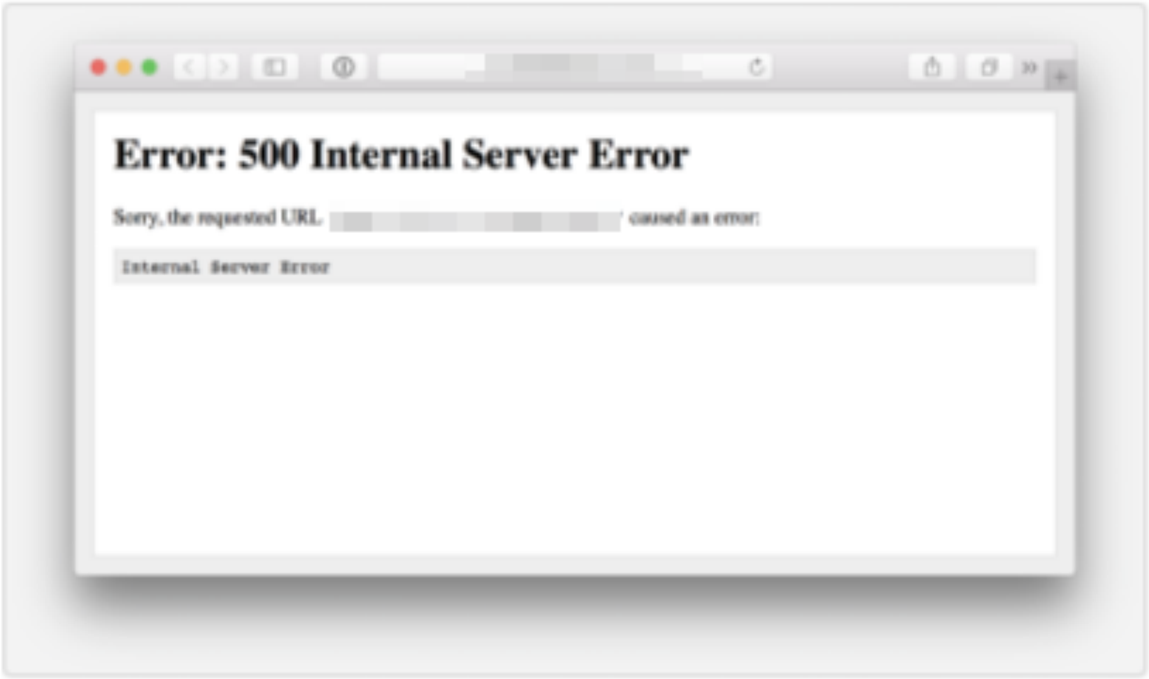


Stu West 12:13 AM

@boris is a2f2 offline? i haven't gotten any alerts in a few days, #scorecard-a2f2 doesn't seem to have gotten any since 2:05am UTC on 9 May, #scorecard-a2f2-debug only has process messages no actual alerts, and older alert links give me (edited)

Stu West 12:13 AM

uploaded this image: **Screen Shot 2017-05-11 at 2.12.52 PM.png**



\(^ヮ^)/

Stu West 12:15 AM

i can't live without my alerts!!

(*≧ω≦*)


Stu West 5:58 AM
scorecard alerts just sent me a slack message flagging today's woo renewals as an anomaly: https://scorecard.a8c.com/metric/c
i think this is the impact of the

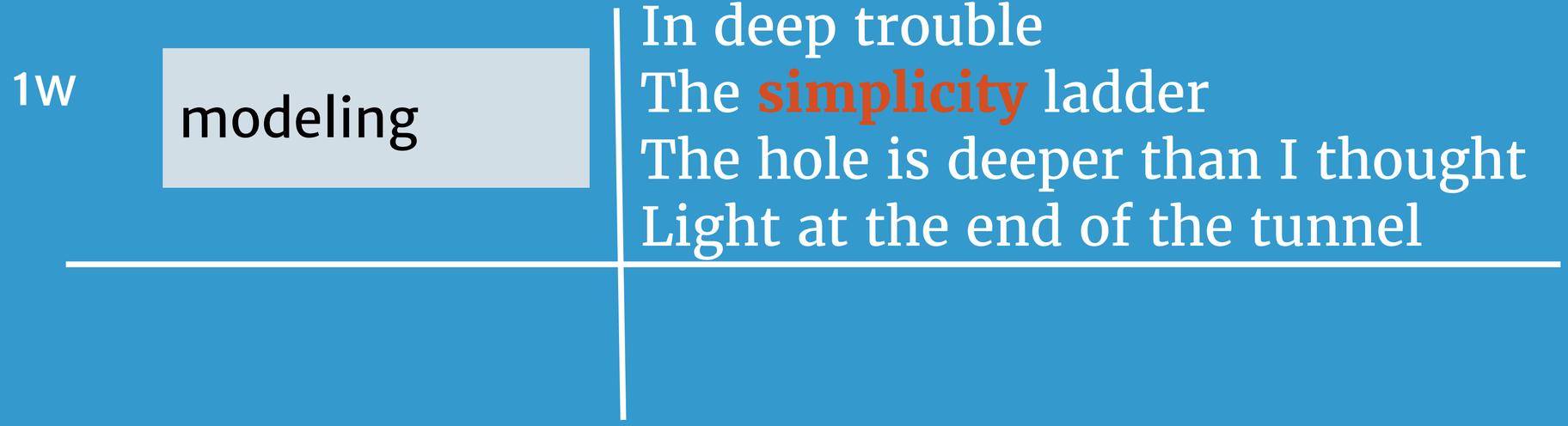
a2f2 reached the critical mass

10 users (2% of the company)

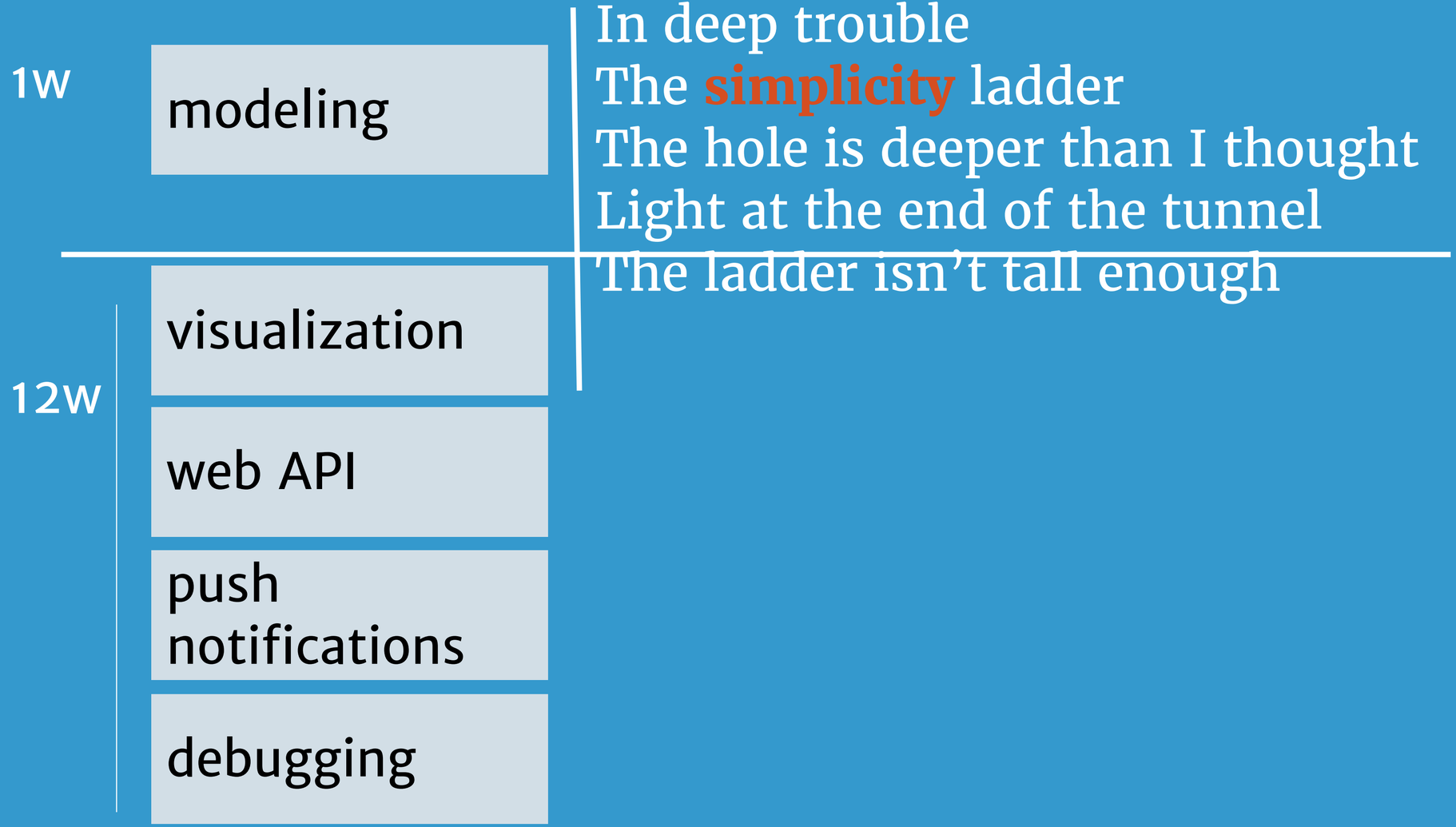
Time series analysis when “good enough” is good enough

In deep trouble
The simplicity ladder
The hole is deeper than I thought
Light at the end of the tunnel
The ladder isn’t tall enough
Towards the bright future

Time series analysis when “good enough” is good enough



Time series analysis when “good enough” is good enough



Time series analysis when “good enough” is good enough

In deep trouble
The **simplicity** ladder
The hole is deeper than I thought
Light at the end of the tunnel
The ladder isn’t tall enough
Towards the bright future

- autoregressive integrated moving average (ARIMA)
- fast Fourier transform
- hidden Markov model
- recurrent neural networks
- support vector machines
- …

http://data.blog
boris@gorelik.net

Boris Gorelik, Ph.D
Data Scientist
AUTOMATTIC

Deliver first, improve later

Boris Gorelik, Ph.D
Data Scientist
AUTOMATTIC
boris@gorelik.net