

psst! go to:

aas233.dfm.io

how can new
Data Analysis Methods
get more out of the
Kepler/K2 Data?

dan foreman-mackey
cca@flatiron // dfm.io // github.com/dfm

spoiler alert

by using
gradients

we can fit more
ambitious
models

everyone should learn about

autodiff

and

hmc

* automatic differentiation & Hamiltonian Monte Carlo

exoplanet.dfm.io

The screenshot shows a web browser window with the URL `https://exoplanet.dfm.io/en/latest/`. The page title is "exoplanet 0.1.3 documentation". In the top right corner, there are links for "next", "modules", and "index".

Table Of Contents

- exoplanet
 - User guide
 - Tutorials
 - License & attribution
 - Changelog
 - 0.1.3 (2019-01-09)
 - 0.1.2 (2018-12-13)
 - 0.1.1 (IPO; 2018-12-06)

Related Topics

Documentation index

- Next: Installation

Quick search

exoplanet

GitHub `dfm/exoplanet` license MIT build passing docs passing DOI 10.5281/zenodo.2536576

powered by `starry` powered by `celerite` powered by `PyMC3` powered by `AstroPy`

exoplanet is a toolkit for probabilistic modeling of transit and/or radial velocity observations of **exoplanets** and other astronomical time series using **PyMC3**. **PyMC3** is a flexible and high-performance model building language and inference engine that scales well to problems with a large number of parameters. *exoplanet* extends **PyMC3**'s language to support many of the custom functions and distributions required when fitting exoplanet datasets. These features include:

- A fast and robust solver for Kepler's equation.
- Scalable Gaussian Processes using `celerite`.
- Fast and accurate limb darkened light curves using `starry`.
- Common reparameterizations for **limb darkening parameters**, and **planet radius and impact parameter**.
- And many others!

All of these functions and distributions include methods for efficiently calculating their *gradients* so that they can be used with gradient-based inference methods like **Hamiltonian Monte Carlo**, **No U-Turns Sampling**, and **variational inference**. These methods tend to be more robust than the methods more commonly used in astronomy (like **ensemble samplers** and **nested sampling**) especially when the model has more than a few parameters. For many exoplanet applications, *exoplanet* (the code) can improve the typical performance by orders of magnitude.

exoplanet is being actively developed in a **public repository on GitHub** so if you have any trouble, **open an issue** there.

v: latest

aas233.dfm.io

aas233.dfm.io

aas233.ipynb - Colaboratory

https://colab.research.google.com/gist/dfm/da5f73a0b791424b1d92b8df278bd55b/aas233.ipynb

aas233.ipynb

File Edit View Insert Runtime Tools Help

SHARE Sign in

CONNECT EDITING

Demo for Dan Foreman-Mackey's talk at AAS233

If you start running this notebook at the beginning of my talk, it should be finished by the end. This notebook uses the [exoplanet](#) code to fit the light curve of the transiting exoplanet K2-18b.

First, we need to install the dependencies:

```
[ ] !pip uninstall -y astropy
    !pip install exoplanet==0.1.3 astropy>=3.1.1 corner
```

Then we download the light curve de-trended using [everest](#):

```
[ ] import matplotlib as mpl
    mpl.rcParams.update(mpl.rcParamsDefault)
    %matplotlib inline
    mpl.rcParams["savefig.dpi"] = 100
    mpl.rcParams["figure.dpi"] = 100
    mpl.rcParams["font.size"] = 16

    import numpy as np
    import matplotlib.pyplot as plt

    from astropy.io import fits
    from scipy.signal import savgol_filter

    # Download the data
    lc_url = "https://archive.stsci.edu/hlsp/everest/v2/c01/201900000/12552/hlsp_everest_k2_11c_201912552-c01_kepler_v2.0_lc.fits"
    with fits.open(lc_url) as hdus:
        lc = hdus[1].data
        lc_hdr = hdus[1].header

    # Work out the exposure time
    texp = lc_hdr["FRAMETIM"] * lc_hdr["NUM_FRM"]
```


*let's start
with some* **audience
participation**

who here has...

heard of mcmc

?

who here has...

used mcmc

?

who here has...

heard of hmc
Hamiltonian Monte Carlo

?

who here has...

used hmc

?

ADS Search

https://ui.adsabs.harvard.edu/#search/q="hamiltonian%20monte%20carlo"%20database%3Aastronomy&sort=date%20desc%2C%20bibcode%20desc&p_...

ads Feedback ORCID About Sign Up Log In

QUICK FIELD: Author First Author Abstract Year Fulltext All Search Terms

Start New Search "hamiltonian monte carlo" database:astronomy

Your search returned 36 results

Date Export Explore

AUTHORS

- Carlo, L 5
- Carlo, B 4
- Francesco, A 3
- Hobson, M 3
- Kitaura, F 3

more

COLLECTIONS

- astronomy 36
- physics 8
- general 2

REFEREED

- non-refereed 19
- refereed 17

KEYWORDS

PUBLICATIONS

BIB GROUPS

SIMBAD OBJECTS

Hide highlights Show abstracts Hide Sidebars

1 2019MNRAS.482.1096R 2019/01 cited: 1

Radio galaxy shape measurement with Hamiltonian Monte Carlo in the visibility domain

Rivi, M.; Lochner, M.; Balan, S. T. *and 2 more*

*Radio galaxy shape measurement with **Hamiltonian Monte Carlo** in the visibility domain from visibility data of radio continuum surveys, instead of from image data. We apply a **Hamiltonian Monte Carlo***

2 2018arXiv181007443B 2018/10

Bayesian inference for binary neutron star inspirals using a Hamiltonian Monte Carlo Algorithm

Bouffanais, Yann; Porter, Edward K.

*Bayesian inference for binary neutron star inspirals using a **Hamiltonian Monte Carlo** Algorithm we conduct a feasibility study using a **Hamiltonian Monte Carlo** algorithm (HMC). The HMC is a sampling*

3 2018arXiv181006559L 2018/10

STARRY: Analytic Occultation Light Curves

Luger, Rodrigo; Agol, Eric; Foreman-Mackey, Daniel *and 3 more*

0 selected

Years Citations Reads

refereed non refereed

Year Range	Refereed	Non-refereed	Total
1982-1984	2	0	2
1985-1987	0	0	0
1988-1990	0	0	0
1991-1993	0	0	0
1994-1996	0	0	0
1997-1999	2	3	5
2000-2002	1	1	2
2003-2005	0	1	1
2006-2008	2	1	3
2009-2011	2	2	4
2012-2014	1	0	1
2015-2017	6	4	10
2018-2019	1	8	9

Limit results to papers from 1982 to 2019 Apply

ADS Search

https://ui.adsabs.harvard.edu/#search/q="hamiltonian%20monte%20carlo"%20database%3Aastronomy&sort=date%20desc%2C%20bibcode%20desc&p_...

ads Feedback ORCID About Sign Up Log In

QUICK FIELD: Author First Author Abstract Year Fulltext All Search Terms

Start New Search "hamiltonian monte carlo" database:astronomy

Your search returned **36** results

Date Export Explore

AUTHORS

- Carlo, L 5
- Carlo, B 4
- Francesco, A 3
- Hobson, M 3
- Kitaura, F 3

COLLECTIONS

- astronomy 36
- physics 8
- general 2

REFEREED

- non-refereed 19
- refereed 17

KEYWORDS

PUBLICATIONS

BIB GROUPS

SIMBAD OBJECTS

Hide highlights Show abstracts Hide Sidebars

1 2019MNRAS.482.1096R 2019/01 cited: 1

Radio galaxy shape measurement with Hamiltonian Monte Carlo in the visibility domain

Rivi, M.; Lochner, M.; Balan, S. T. *and 2 more*

Radio galaxy shape measurement with Hamiltonian Monte Carlo in the visibility domain from visibility data of radio continuum surveys, instead of from image data. We apply a Hamiltonian Monte Carlo

2 2018arXiv181007443B 2018/10

Bayesian inference for binary neutron star inspirals using a Hamiltonian Monte Carlo Algorithm

Bouffanais, Yann; Porter, Edward K.

Bayesian inference for binary neutron star inspirals using a Hamiltonian Monte Carlo Algorithm we conduct a feasibility study using a Hamiltonian Monte Carlo algorithm (HMC). The HMC is a sampling

3 2018arXiv181006559L 2018/10

STARRY: Analytic Occultation Light Curves

Luger, Rodrigo; Agol, Eric; Foreman-Mackey, Daniel *and 3 more*

0 selected

Years Citations Reads

refereed non refereed

Limit results to papers from 1982 to 2019 Apply

how can new
Data Analysis Methods
get more out of the
Kepler/K2 Data?

dan foreman-mackey
cca@flatiron // dfm.io // github.com/dfm

what is
data analysis
anyways?

the probability of
parameters
given some
data

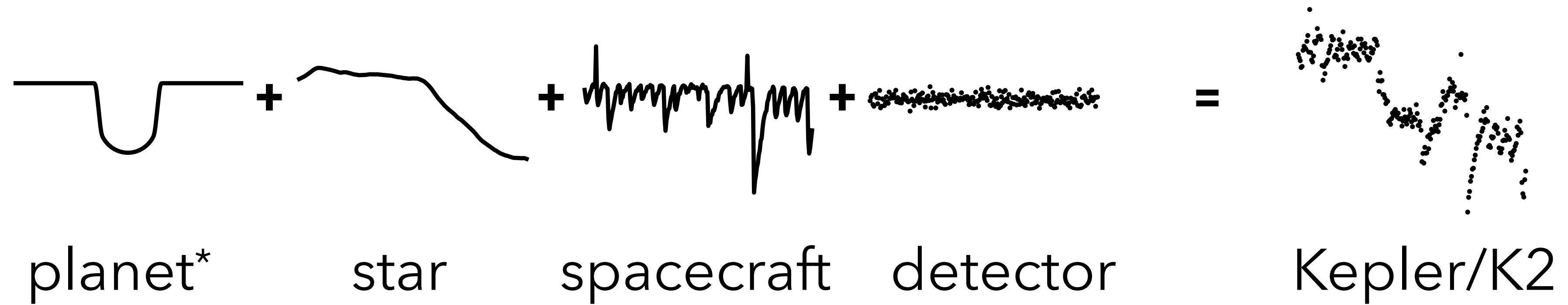
$p(\text{physics} \mid \text{data})$

we need a
model*

and a way to do
inference

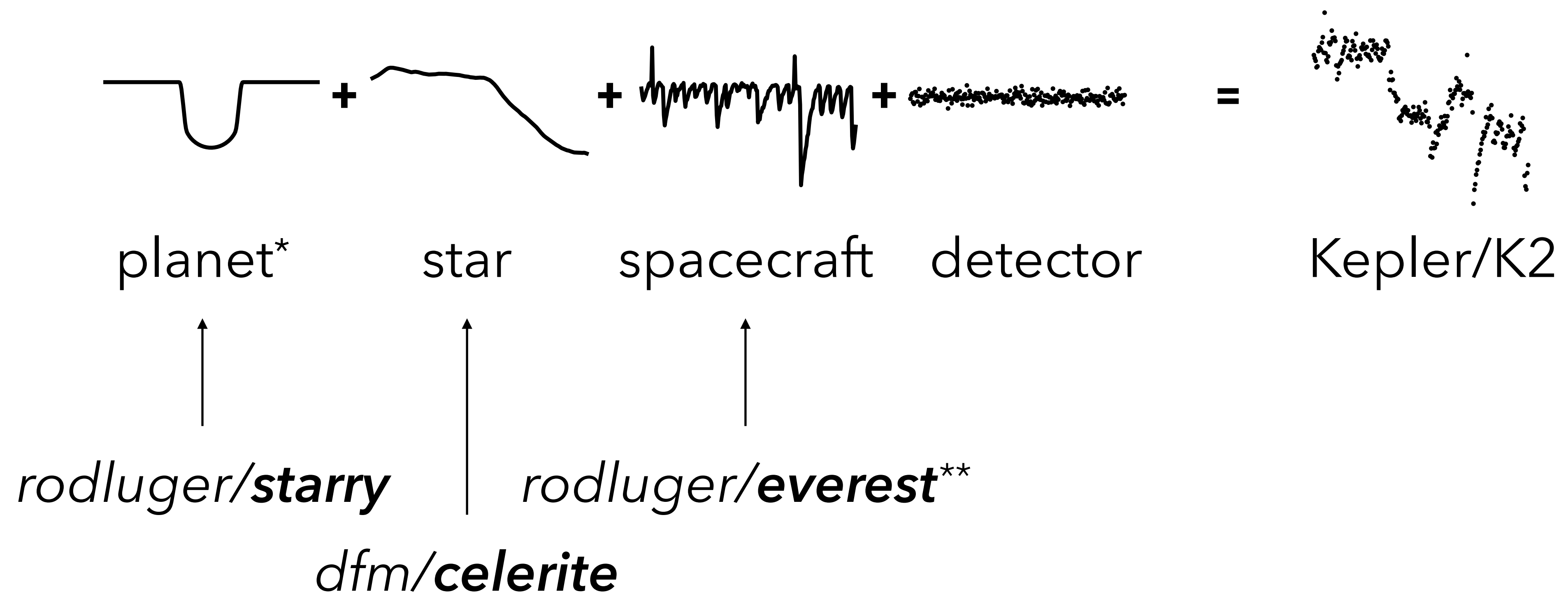
* & probably a way to evaluate it (efficiently?)

our **model**



* optional, of course

our **model**



* optional, of course

** these are all GitHub repositories

we need a
model*

and a way to do
inference

* & probably a way to evaluate it (efficiently?)

"inference" is a fancy word for

fitting

in astronomy, we often use

mcmc

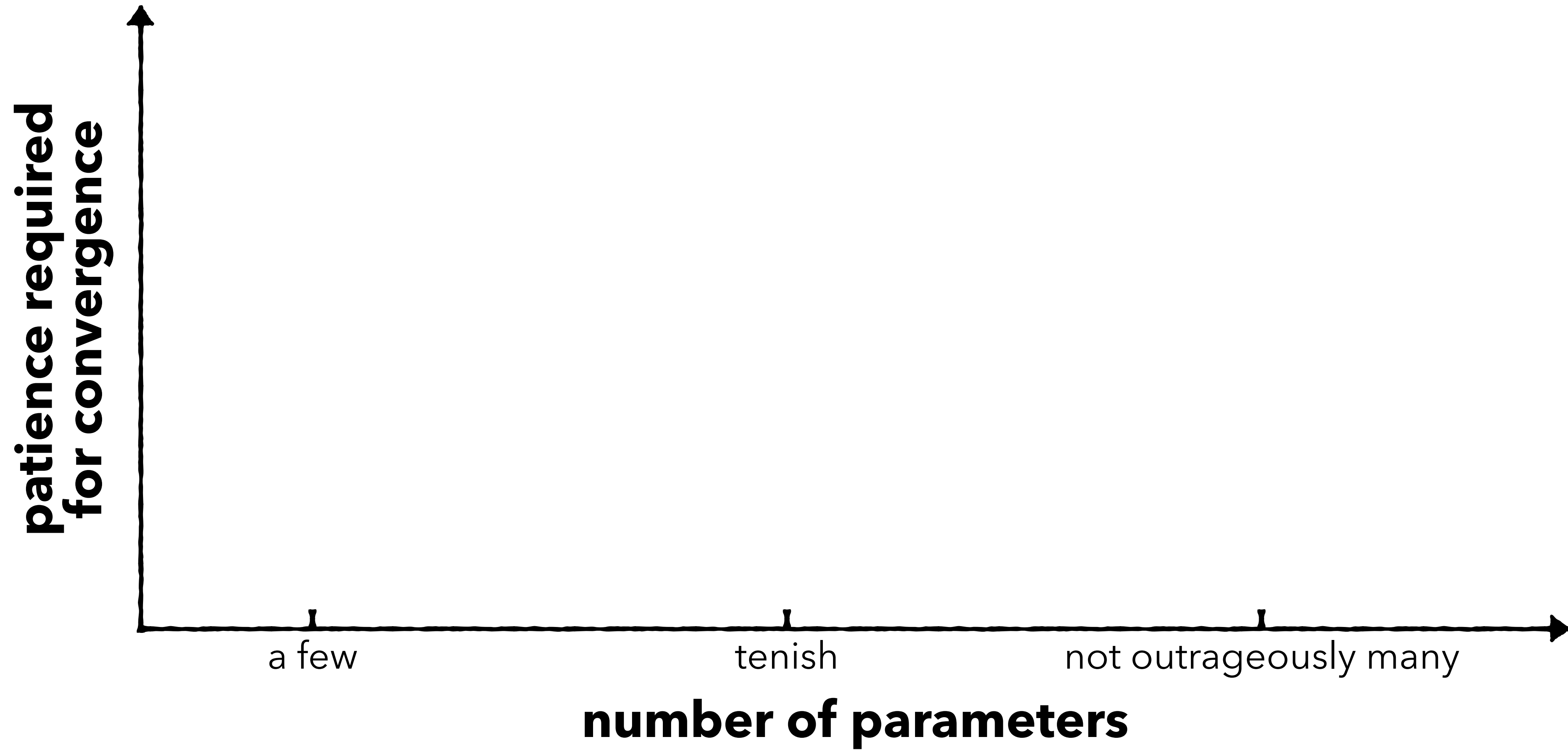
but

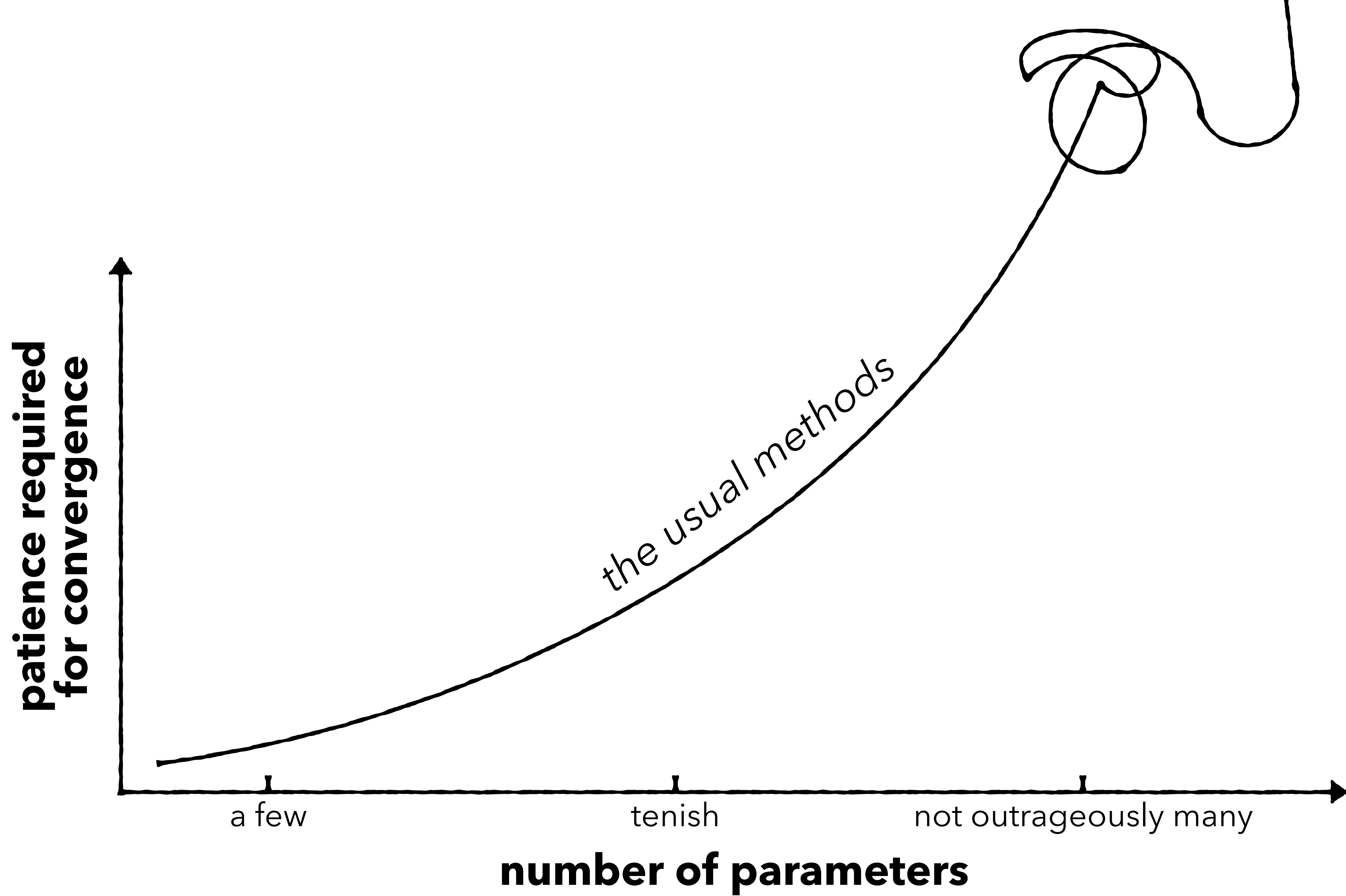
there are some problems

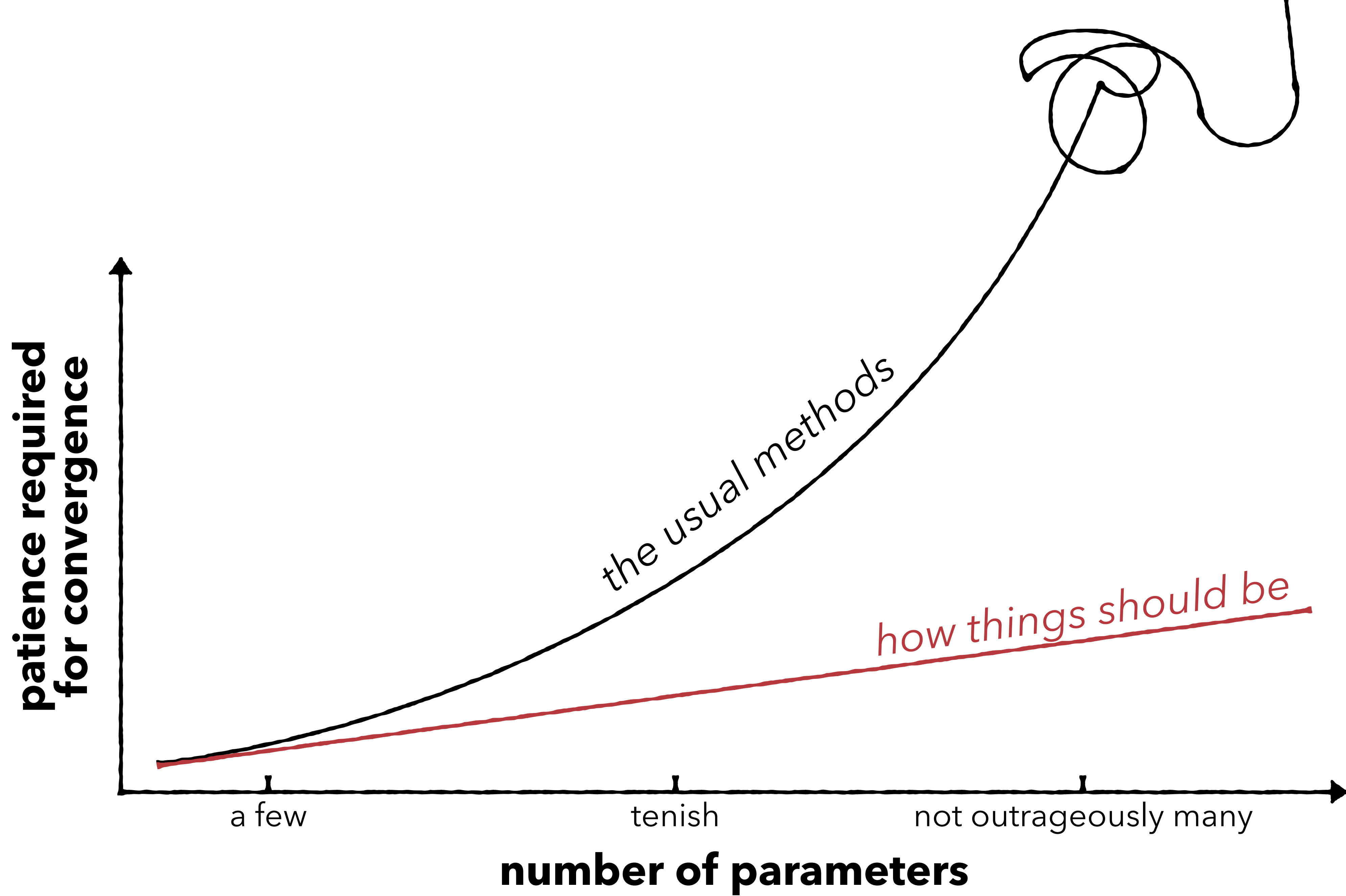
an incomplete list of reasons why the Inference Button™ might not Just Work

- 1 you have a lot of **data**
- 2 you have a lot of **parameters**
- 3 you have a lot of **datasets**
- 4 turns out **inference is Hard***

* especially when you try to automate it







the ultimate goal is to

reduce

*the number of
model evaluations needed*

Hamiltonian

Monte

Carlo

but hmc needs

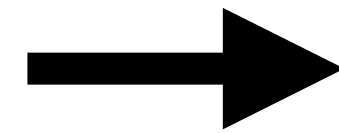
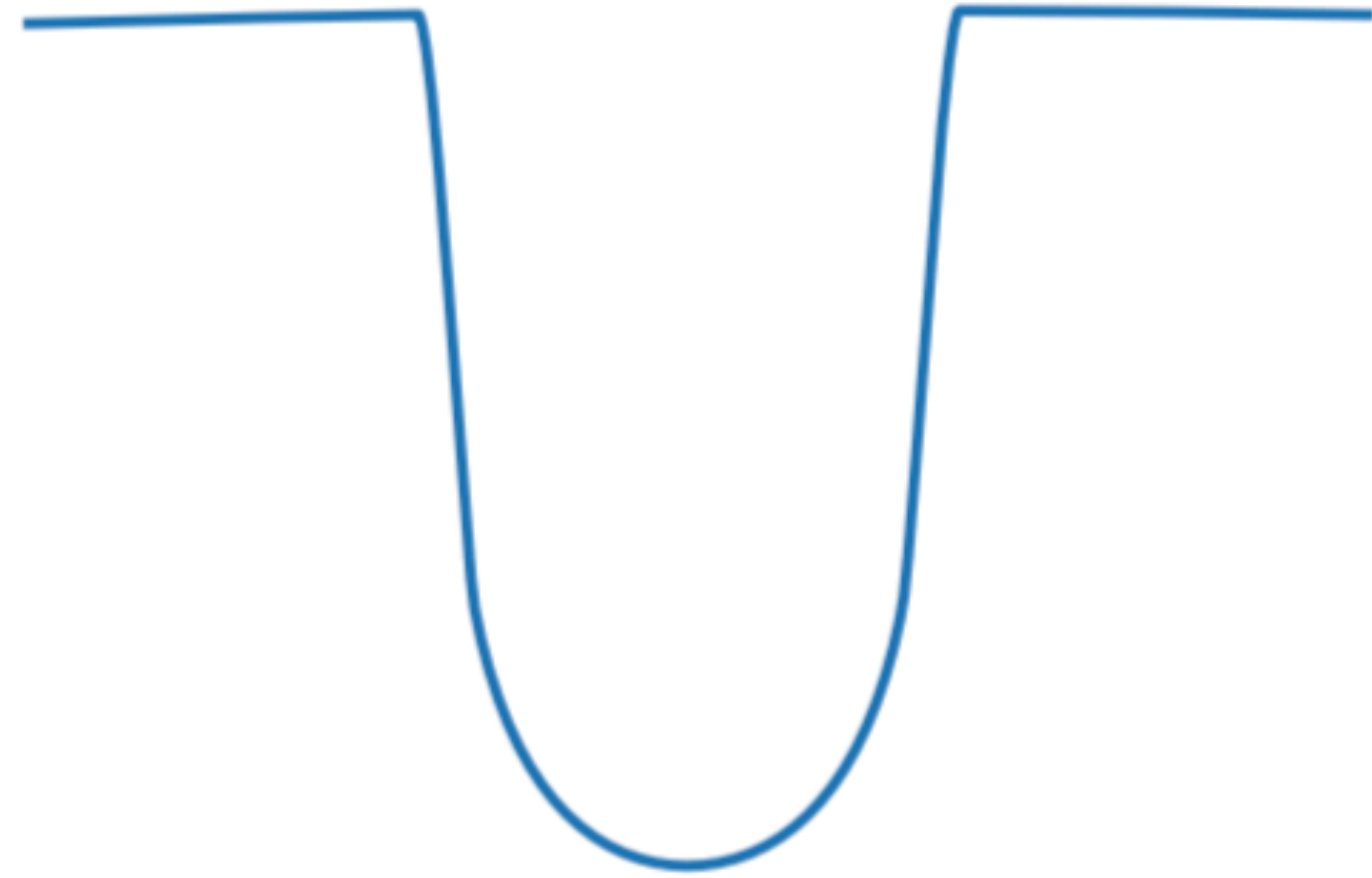
derivatives*

* *not those derivatives*

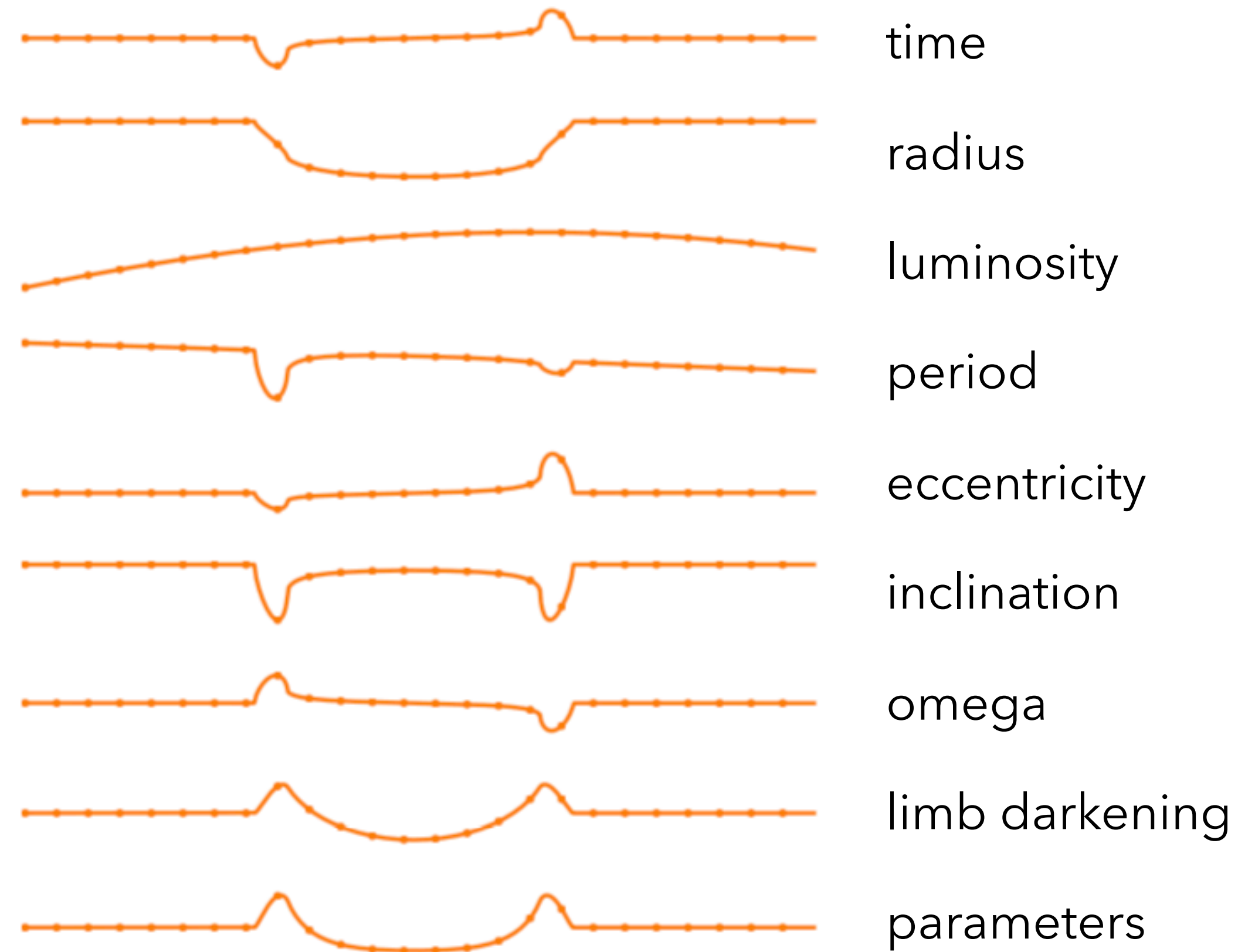
$d p(\text{physics} \mid \text{data})$

$d \text{physics}$

transit light curve



derivative w.r.t.



credit: Rodrigo Luger

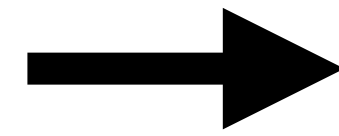
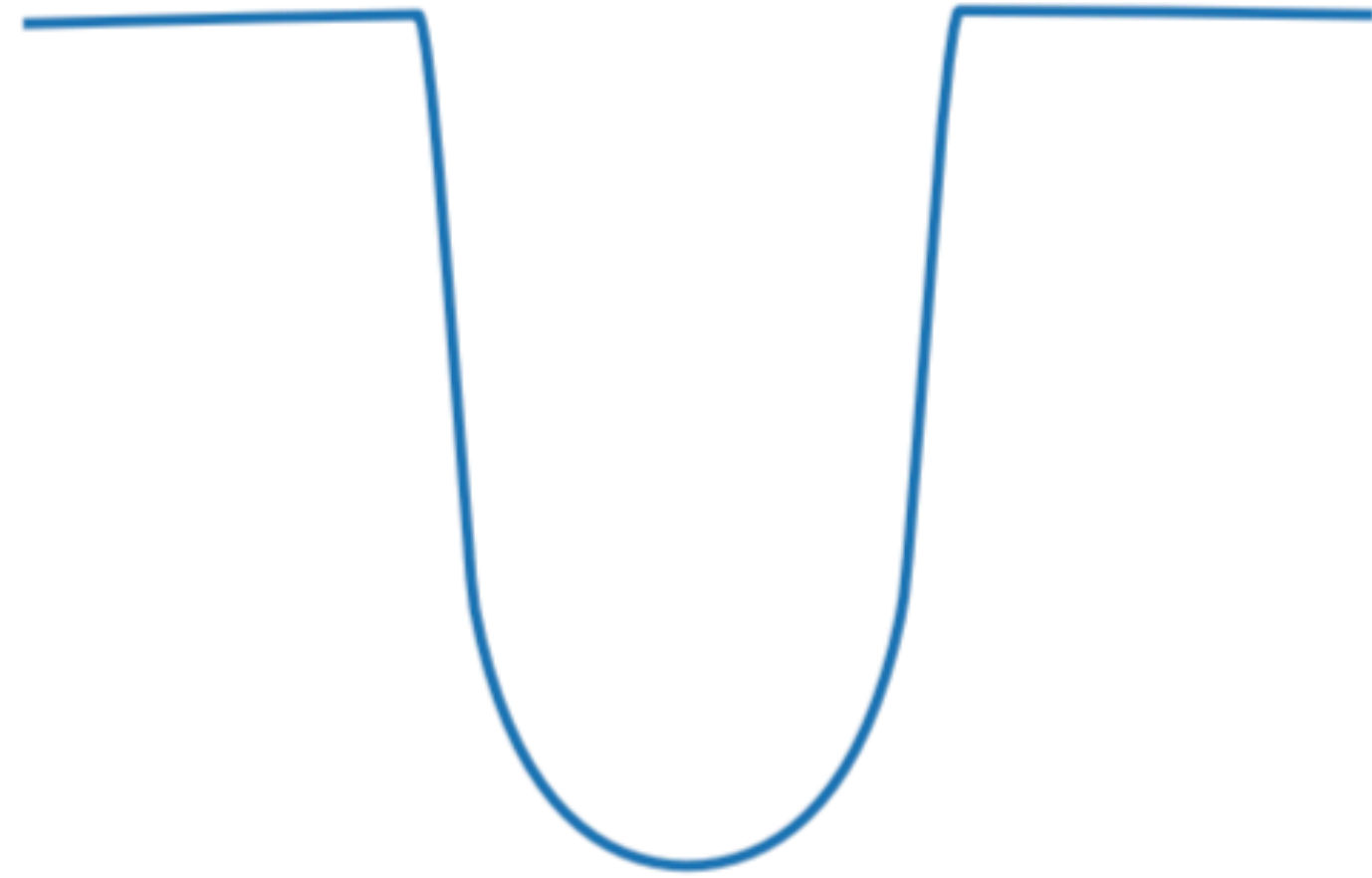
computing all these

derivatives

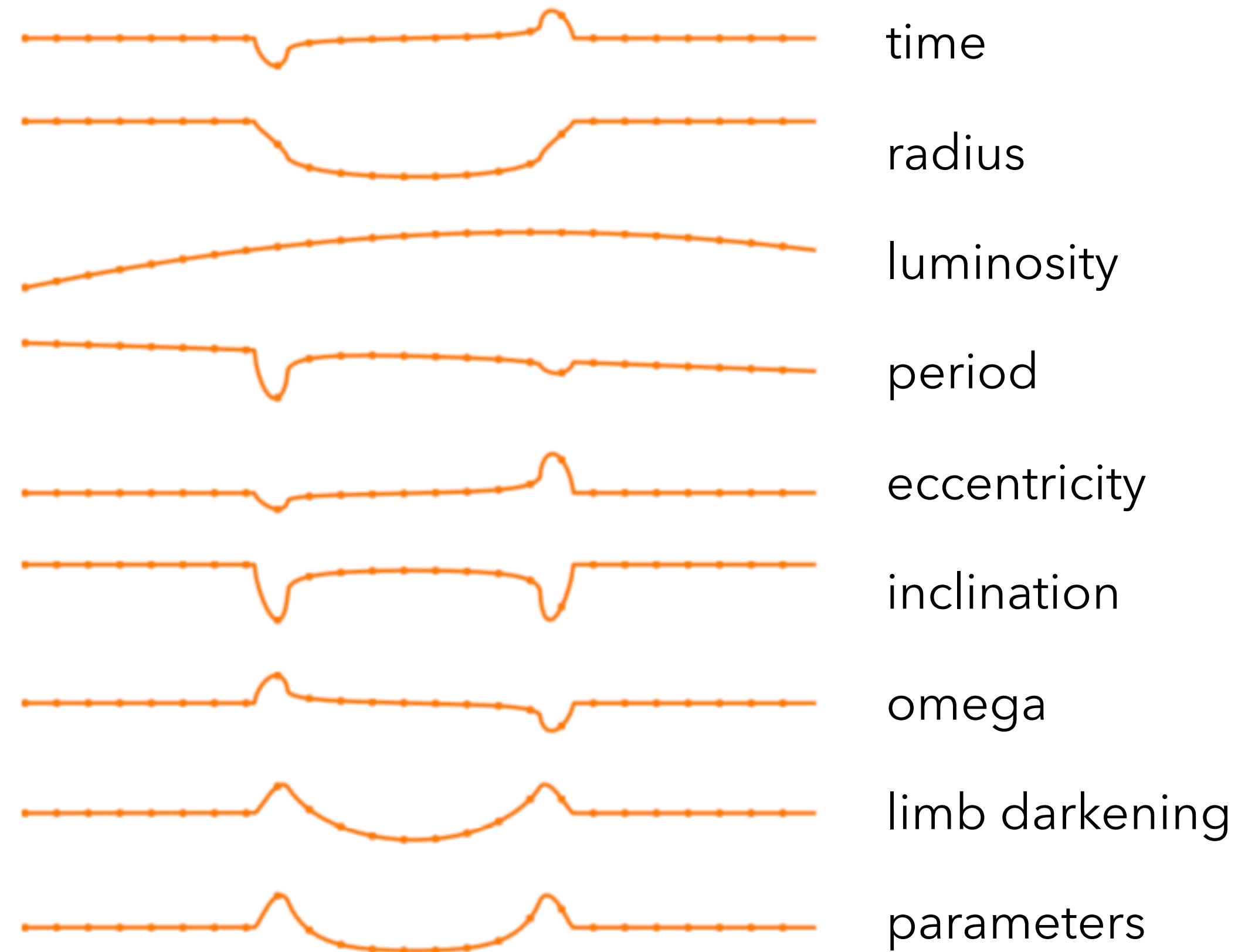
doesn't sound

super fun

transit light curve



derivative w.r.t.



credit: Rodrigo Luger

automatic differentiation

* AKA "backpropagation"; AKA "the chain rule"

exoplanet.dfm.io

The screenshot shows a web browser window with the URL `https://exoplanet.dfm.io/en/latest/`. The page title is "exoplanet 0.1.3 documentation". In the top right corner, there are links for "next", "modules", and "index".

Table Of Contents

- exoplanet
 - User guide
 - Tutorials
 - License & attribution
 - Changelog
 - 0.1.3 (2019-01-09)
 - 0.1.2 (2018-12-13)
 - 0.1.1 (IPO; 2018-12-06)

Related Topics

Documentation index

- Next: Installation

Quick search

exoplanet

GitHub `dfm/exoplanet` license MIT build passing docs passing DOI 10.5281/zenodo.2536576

powered by `starry` powered by `celerite` powered by `PyMC3` powered by `AstroPy`

exoplanet is a toolkit for probabilistic modeling of transit and/or radial velocity observations of **exoplanets** and other astronomical time series using **PyMC3**. **PyMC3** is a flexible and high-performance model building language and inference engine that scales well to problems with a large number of parameters. *exoplanet* extends **PyMC3**'s language to support many of the custom functions and distributions required when fitting exoplanet datasets. These features include:

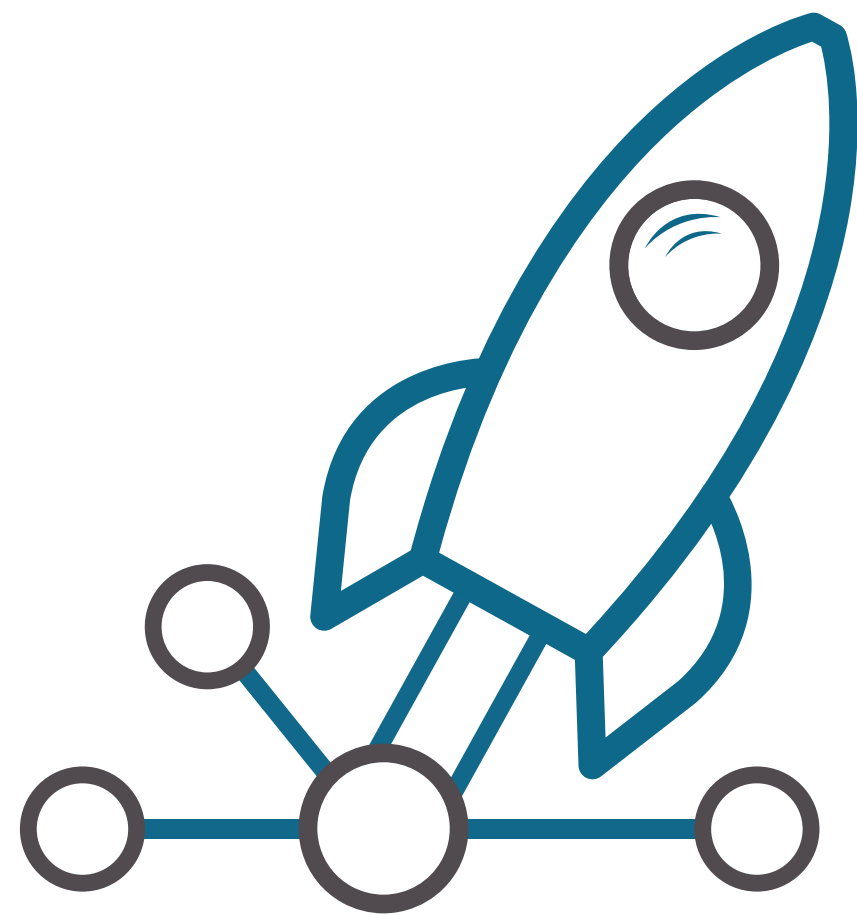
- A fast and robust solver for Kepler's equation.
- Scalable Gaussian Processes using `celerite`.
- Fast and accurate limb darkened light curves using `starry`.
- Common reparameterizations for **limb darkening parameters**, and **planet radius and impact parameter**.
- And many others!

All of these functions and distributions include methods for efficiently calculating their *gradients* so that they can be used with gradient-based inference methods like **Hamiltonian Monte Carlo**, **No U-Turns Sampling**, and **variational inference**. These methods tend to be more robust than the methods more commonly used in astronomy (like **ensemble samplers** and **nested sampling**) especially when the model has more than a few parameters. For many exoplanet applications, *exoplanet* (the code) can improve the typical performance by orders of magnitude.

exoplanet is being actively developed in a **public repository on GitHub** so if you have any trouble, **open an issue** there.

v: latest

built on top of



PyMCS3

exoplanet.dfm.io

exoplanet — exoplanet 0.1.3 do x

← → ↻ <https://exoplanet.dfm.io/en/latest/> ☆ ☰ ⋮

Tutorials

- [A quick intro to PyMC3 for exoplaneteers](#)
 - [Hello world \(AKA fitting a line to data\)](#)
 - [A more realistic example: radial velocity exoplanets](#)
- [PyMC3 extras](#)
 - [Custom tuning schedule](#)
 - [Evaluating model components for specific samples](#)
- [Radial velocity fitting](#)
 - [The radial velocity model in PyMC3](#)
 - [Sampling](#)
 - [Phase plots](#)
- [Transit fitting](#)
 - [The transit model in PyMC3](#)
 - [Sampling](#)
 - [Phase plots](#)
 - [Citations](#)
- [Scalable Gaussian processes in PyMC3](#)
- [Gaussian process models for stellar variability](#)
 - [A Gaussian process model for stellar variability](#)
- [Case study: K2-24, putting it all together](#)
 - [Datasets and initializations](#)
 - [A joint transit and radial velocity model in PyMC3](#)
 - [Sigma clipping](#)
 - [Sampling](#)

v: latest ▾

exoplanet.dfm.io

Case study: K2-24, putting it all together

exoplanet 0.1.3 documentation » [previous](#) | [next](#) | [modules](#) | [index](#)

Table Of Contents

Case study: K2-24, putting it all together

- Datasets and initializations
- A joint transit and radial velocity model in PyMC3
- Sigma clipping
- Sampling
- Phase plots
- Citations

Related Topics

Documentation index

- Previous: Gaussian process models for stellar variability
- Next: Fitting TESS data

Quick search

Note

This tutorial was generated from an IPython notebook that can be downloaded [here](#).

Case study: K2-24, putting it all together

In this tutorial, we will combine many of the previous tutorials to perform a fit of the K2-24 system using the K2 transit data and the RVs from [Petigura et al. \(2016\)](#). This is the same system that we fit in the [Radial velocity fitting](#) tutorial and we'll combine that model with the transit model from the [Transit fitting](#) tutorial and the Gaussian Process noise model from the [Gaussian process models for stellar variability](#) tutorial.

Datasets and initializations

To get started, let's download the relevant datasets. First, the transit light curve from [Everest](#):

```
import numpy as np
import matplotlib.pyplot as plt

from astropy.io import fits
from scipy.signal import savgol_filter

# Download the data
lc_url = "https://archive.stsci.edu/hlsp/everest/v2/c02/203700000/71098/hlsp_everest_k2_
with fits.open(lc_url) as hdus:
    lc = hdus[1].data
```

v: latest ▾

exoplanet

is the plumbing for your

pipeline

it will help
you

get more out of the
Kepler/K2
data

exoplanet
has support for

- 1 **multiplanet** systems
- 2 **RVs & transits**
- 3 **scalable** gaussian processes
- 4 and much more

* and all the derivatives

exoplanet
is

1 robust

2 efficient

3 well tested

4 & well documented

exoplanet.dfm.io

I can't wait to see what you build!

* PS how's your code doing?