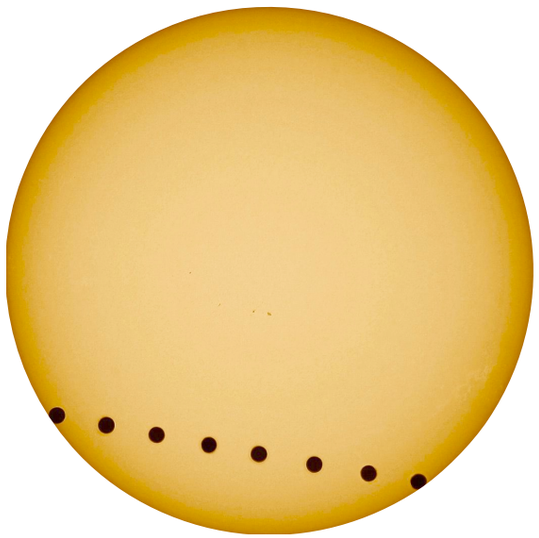


EVEREST

EPIC VARIABILITY EXTRACTION AND REMOVAL FOR EXOPLANET SCIENCE TARGETS

RODRIGO LUGER

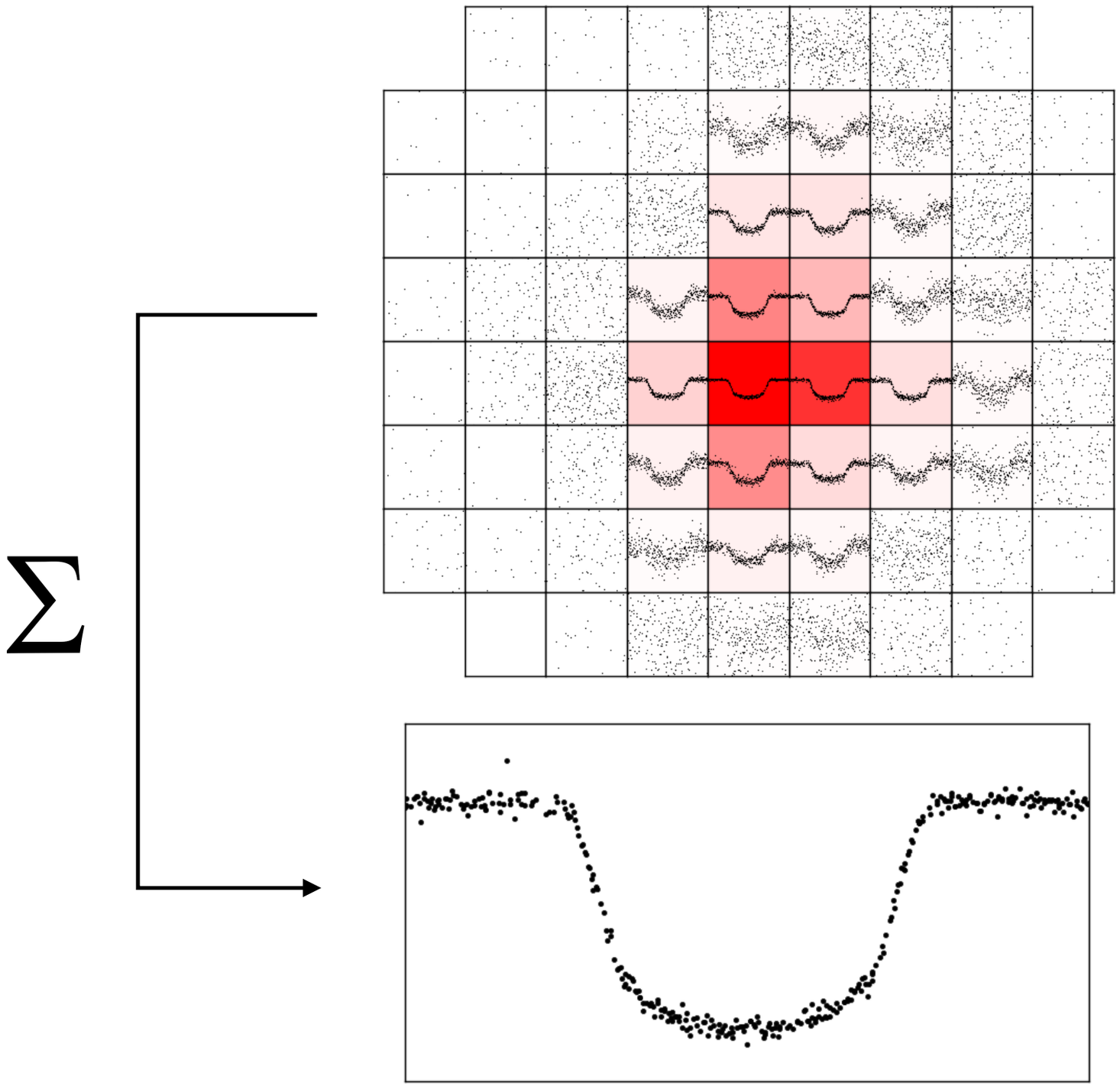
with ERIC AGOL, ETHAN KRUSE, DANIEL FOREMAN-MACKEY,
NICHOLAS SAUNDERS, RORY BARNES, ANDREW BECKER, DRAKE DEMING

1

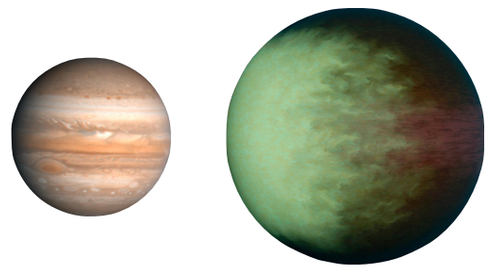
K2

A broken telescope

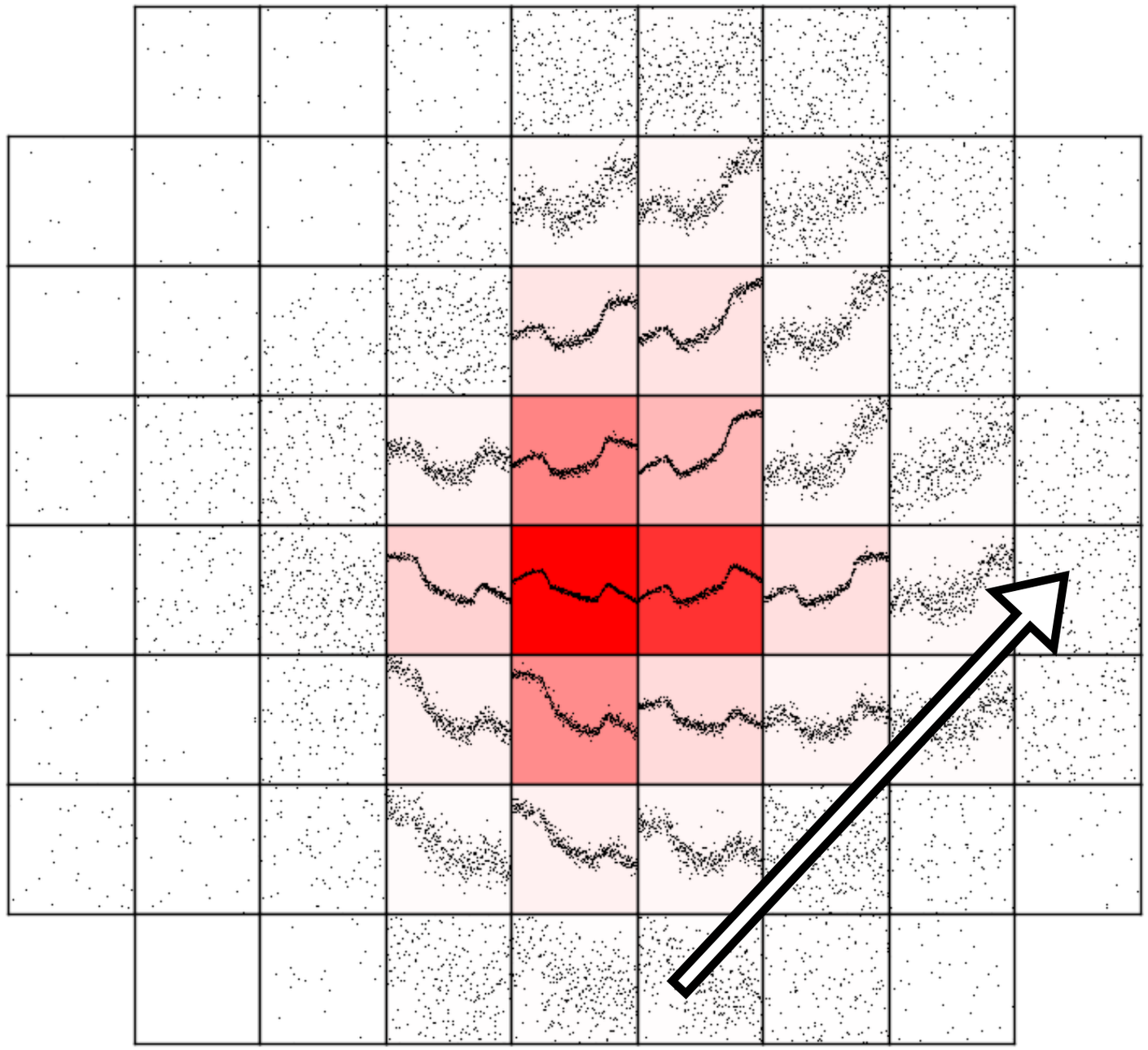
A typical postage stamp, in an almost perfect world



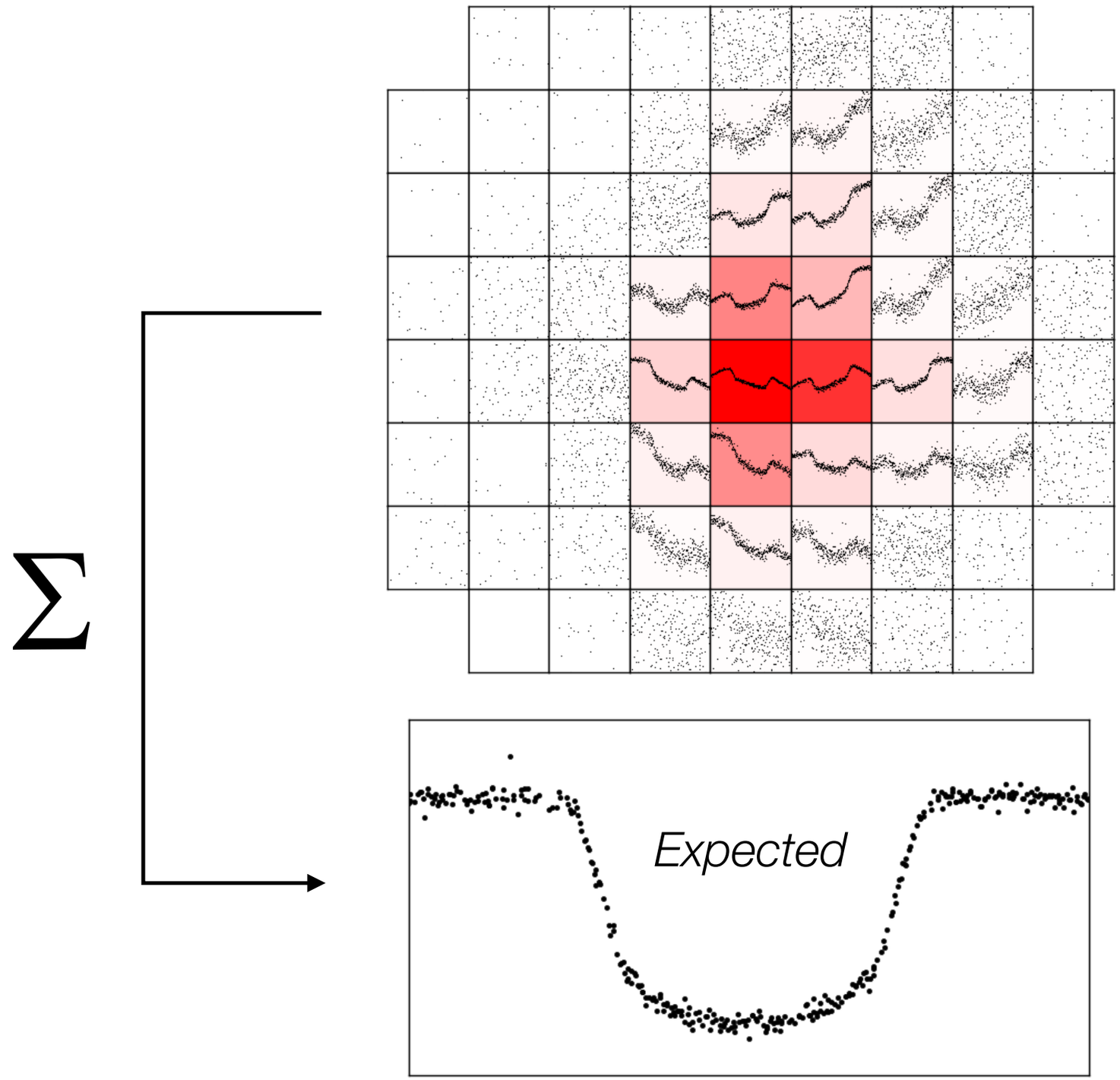
Kepler-6b



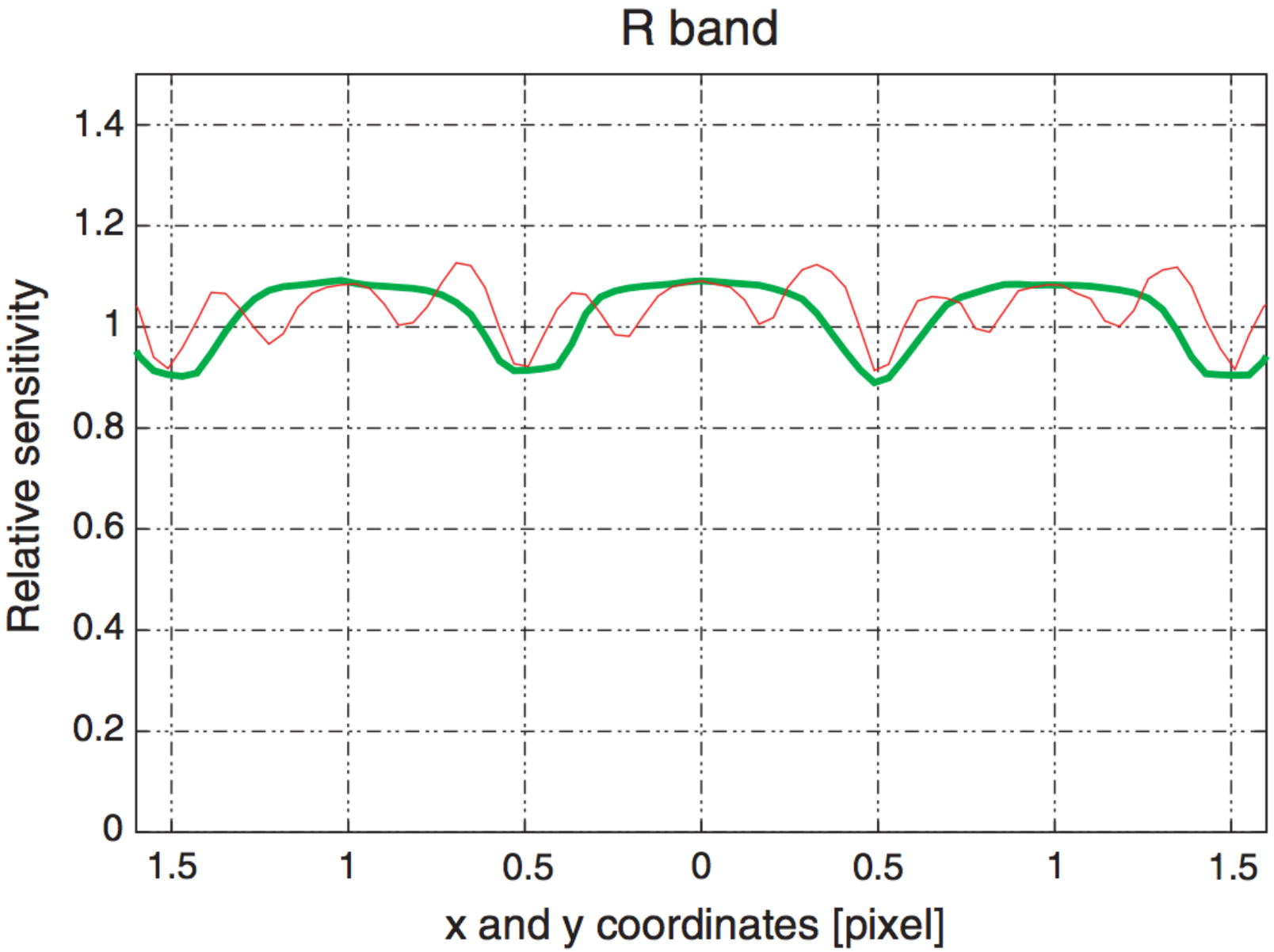
A typical postage stamp in the real world



A typical postage stamp in the real world

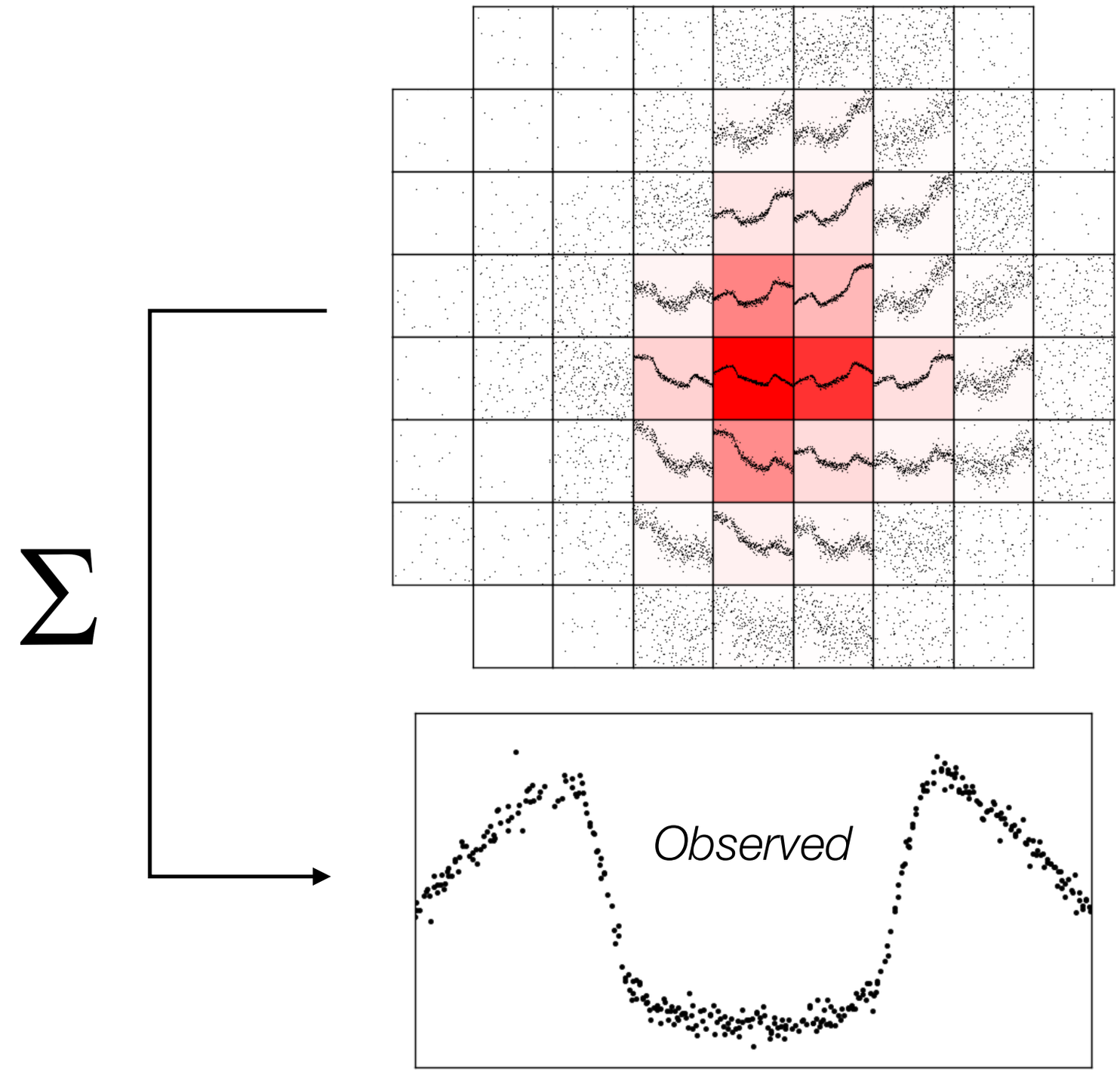


Intrapixel sensitivity



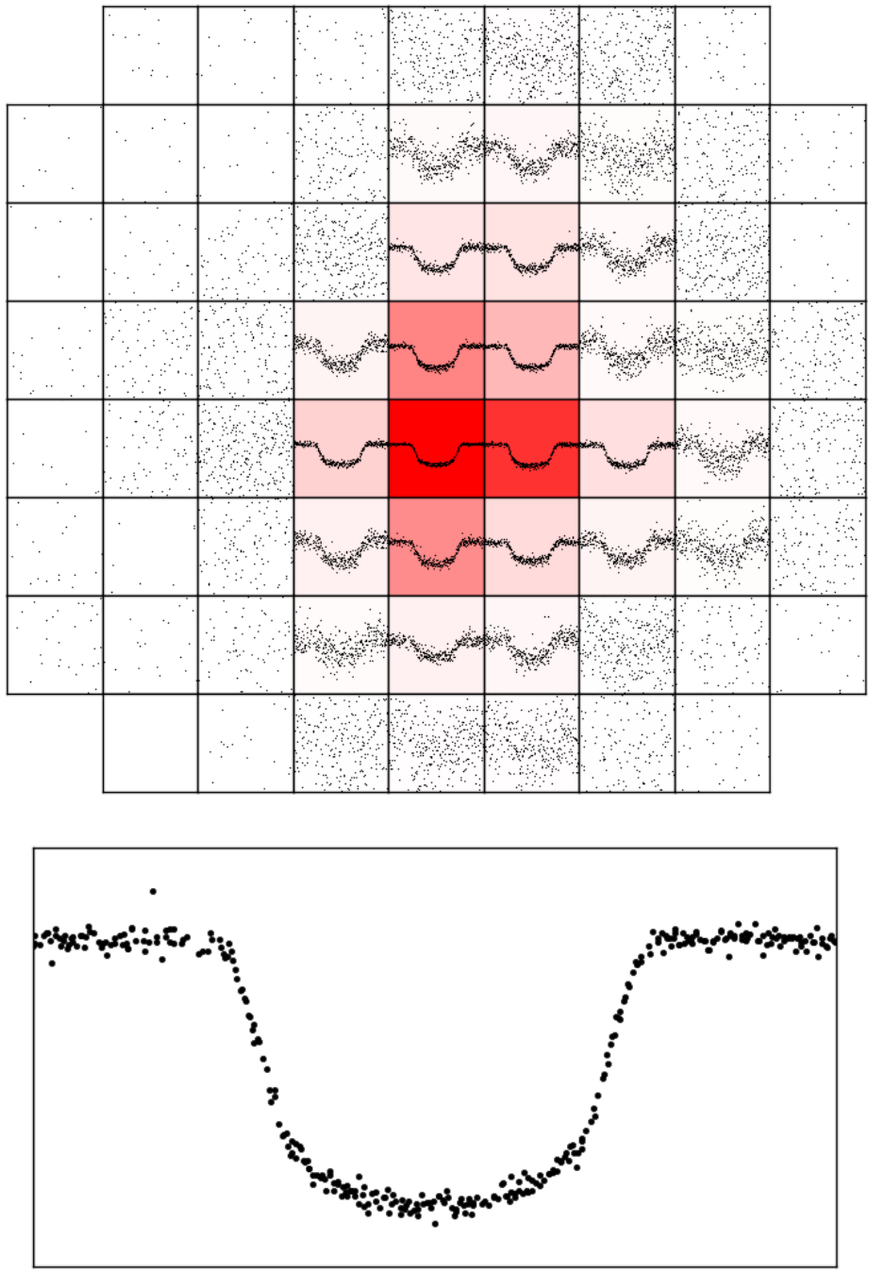
Toyozumi (2005)

A typical postage stamp in the real world

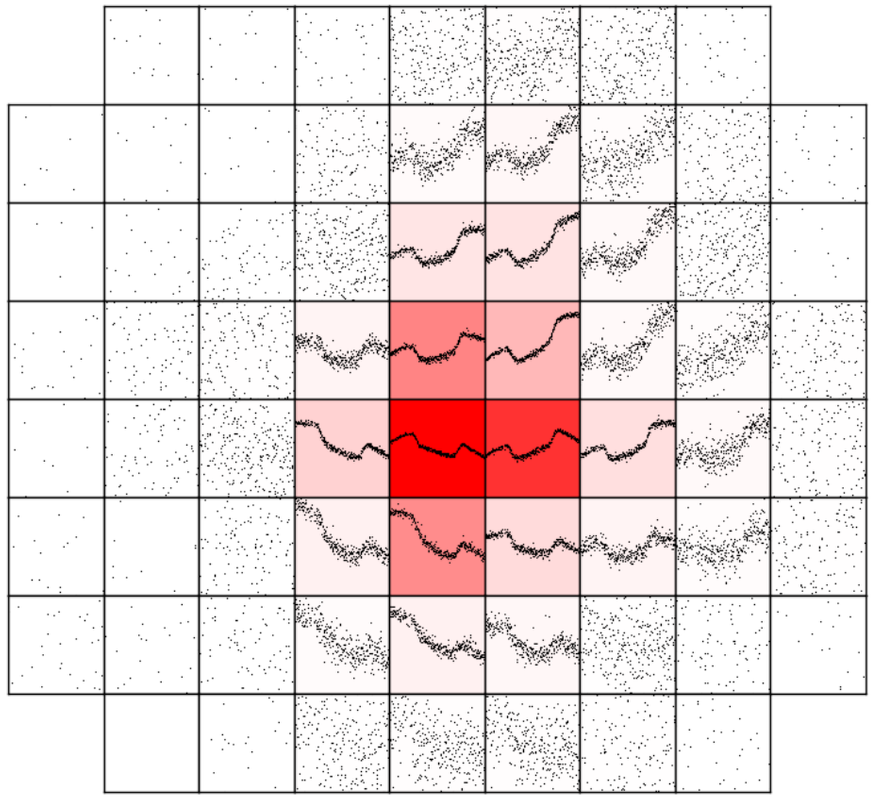


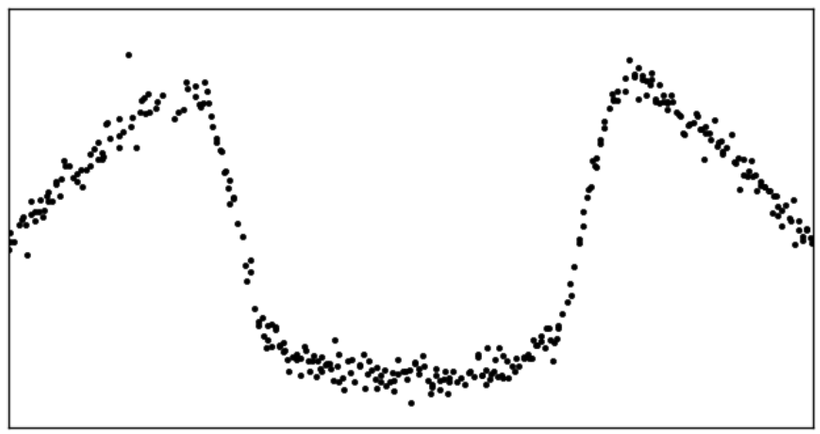
A typical postage stamp in the real world

Ideal

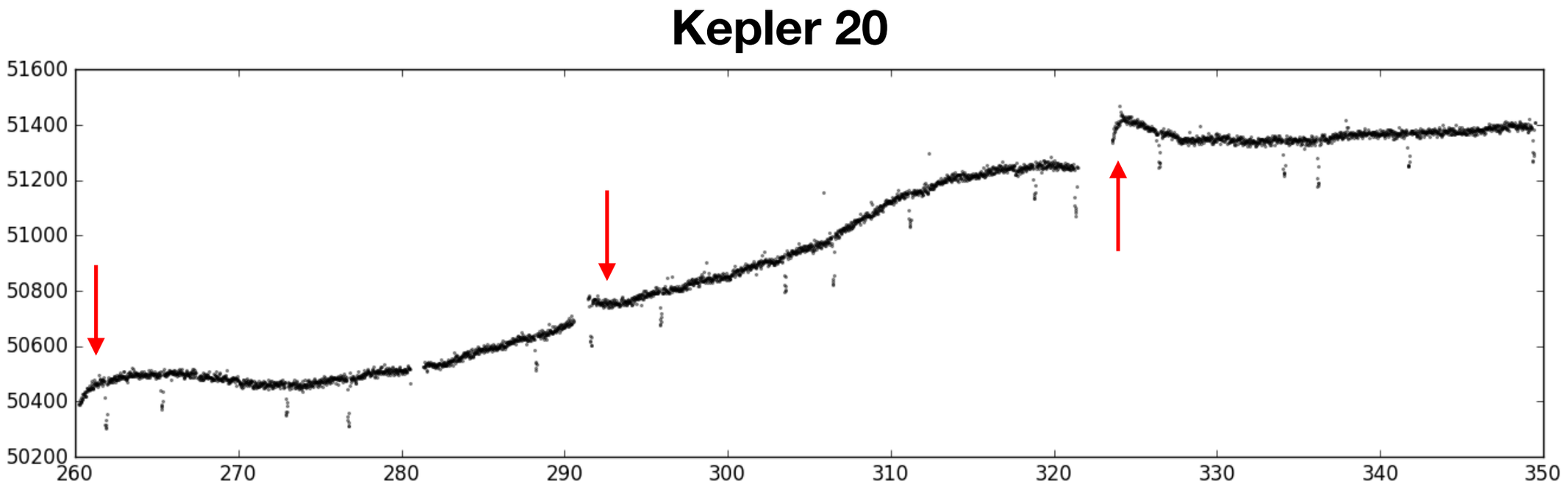


Observed

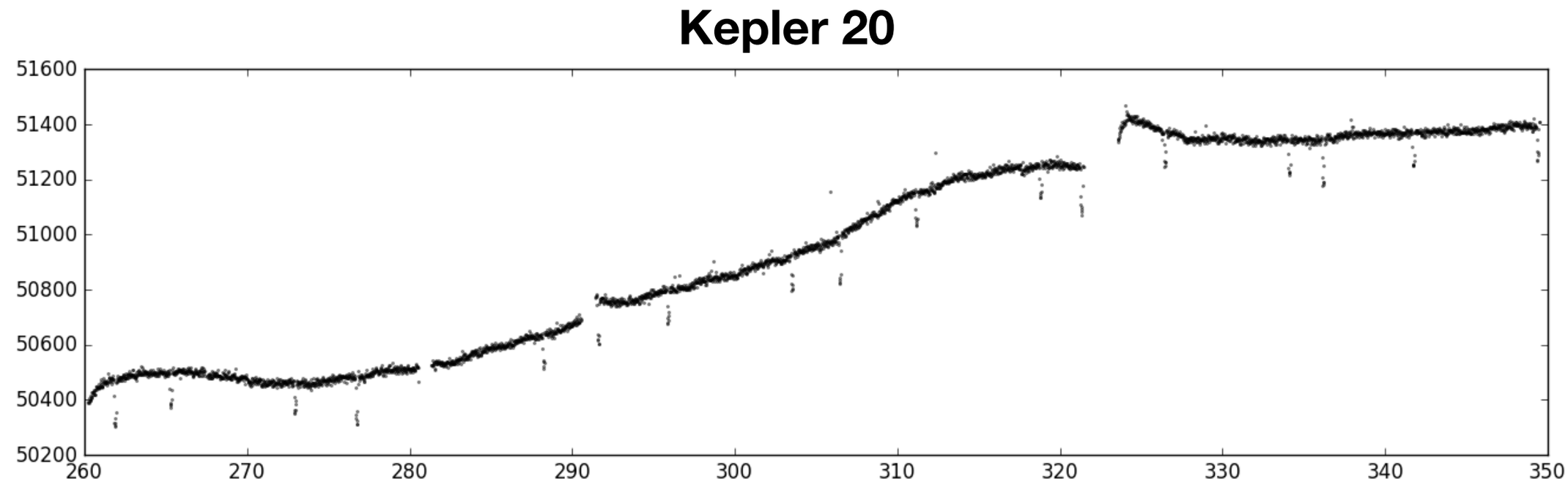


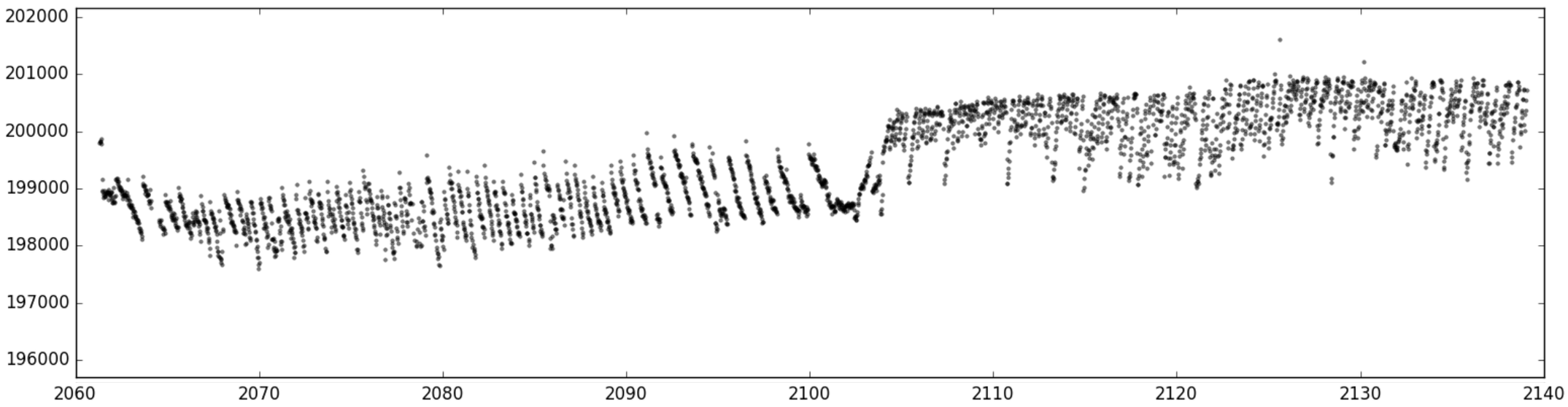


Real Light Curves



Real Light Curves



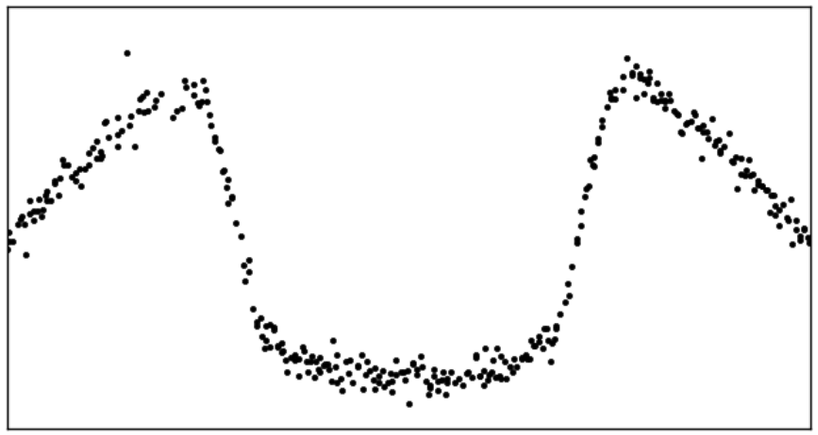


2

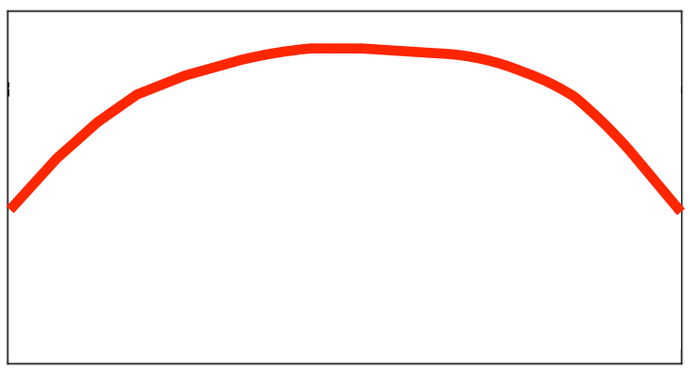
EVEREST

Science with a broken telescope

Back to the simple example…

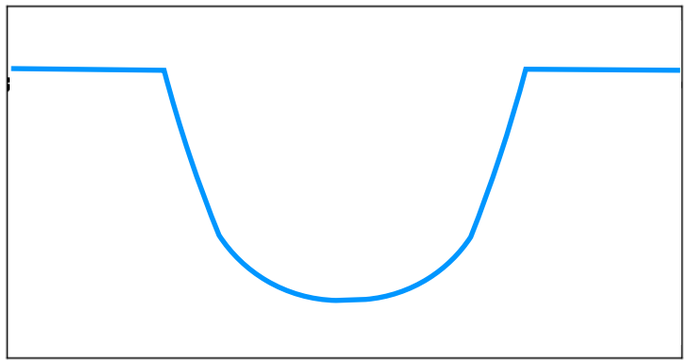


=



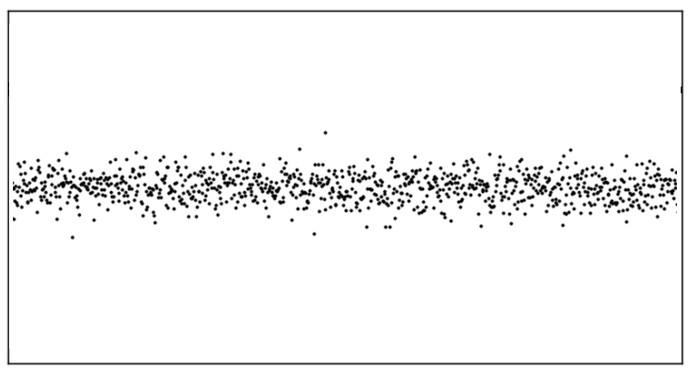
instrumental

+



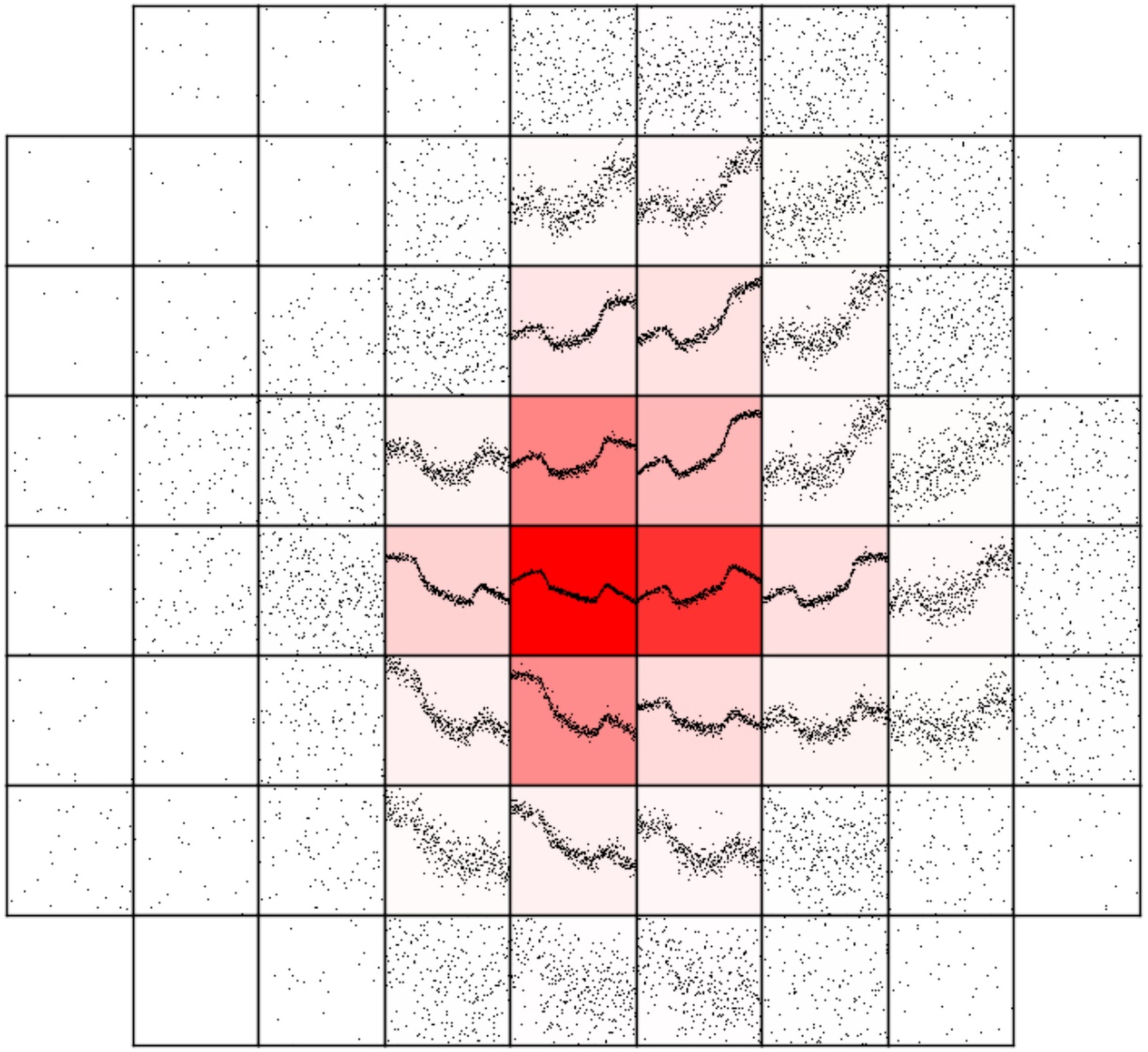
astrophysical

+



noise

Pixel Level Decorrelation

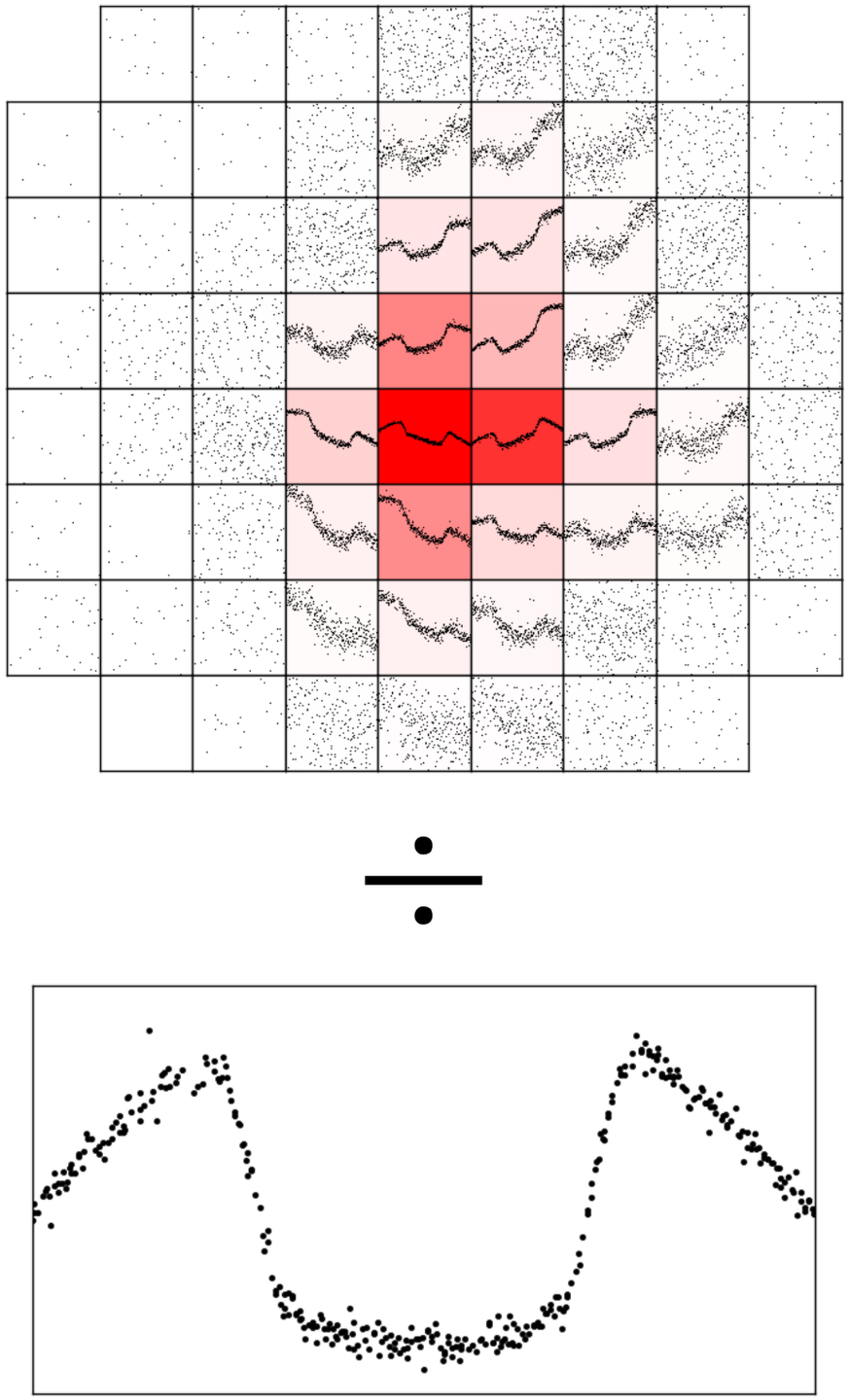


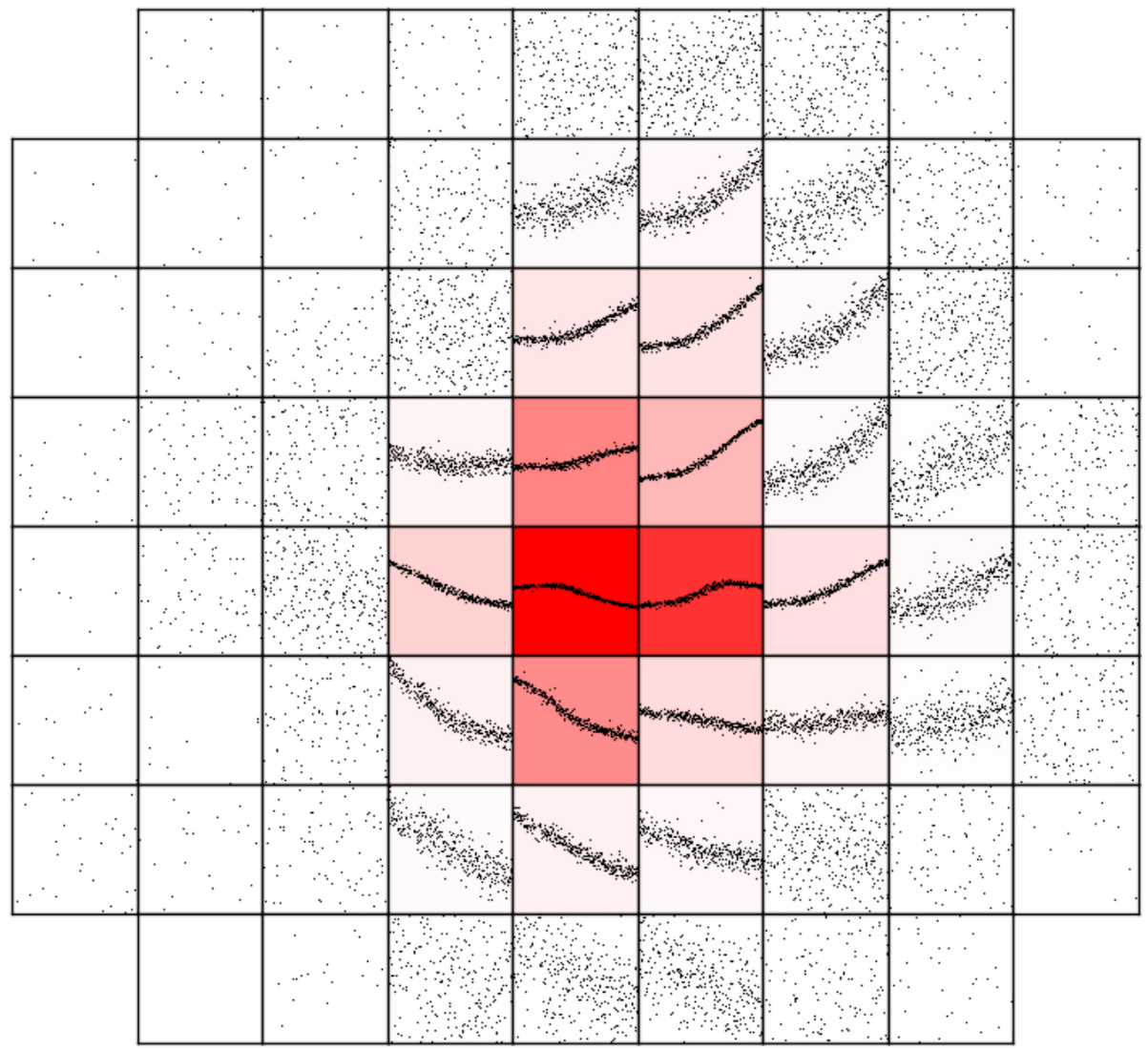
Astrophysical signals are the same in all pixels

Instrumental signals are different

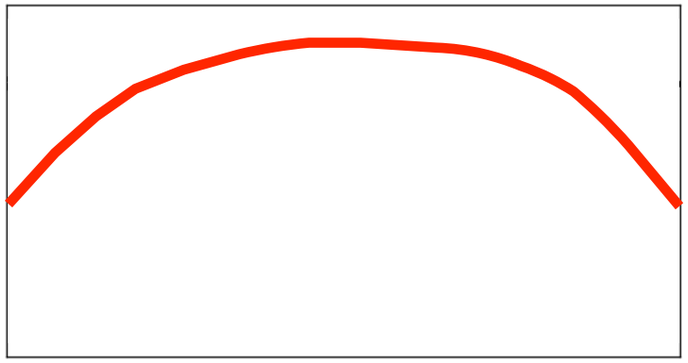
Deming et al. (2015)

Pixel Level Decorrelation





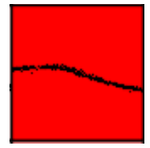
Pixel Level Decorrelation

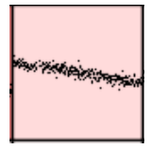


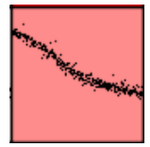
instrumental

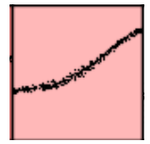
=

$\sum$

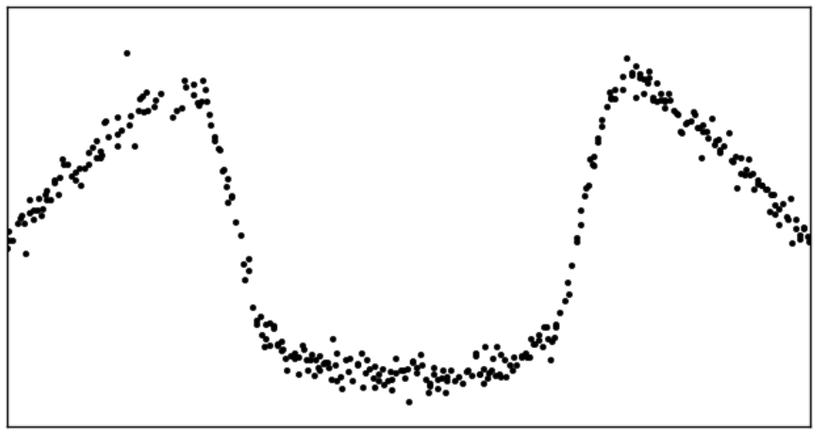
c_0 x 

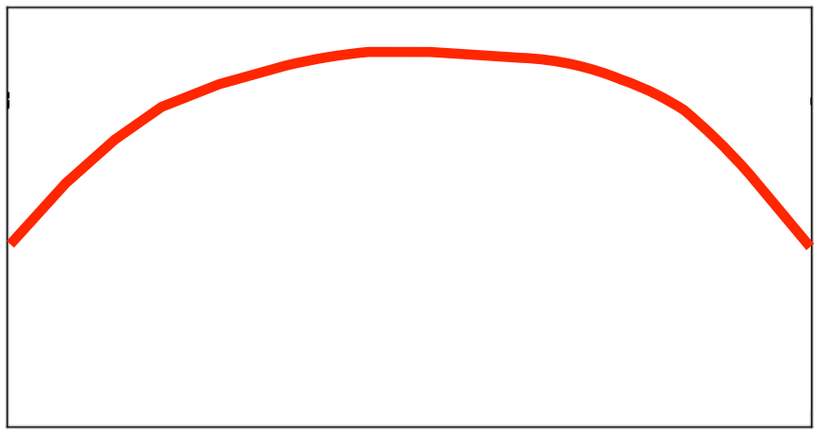
c_1 x 

c_2 x 

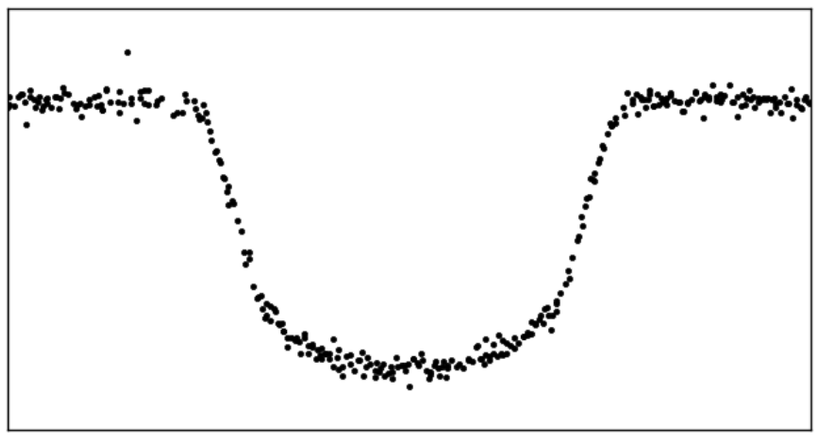
c_3 x 

Pixel Level Decorrelation

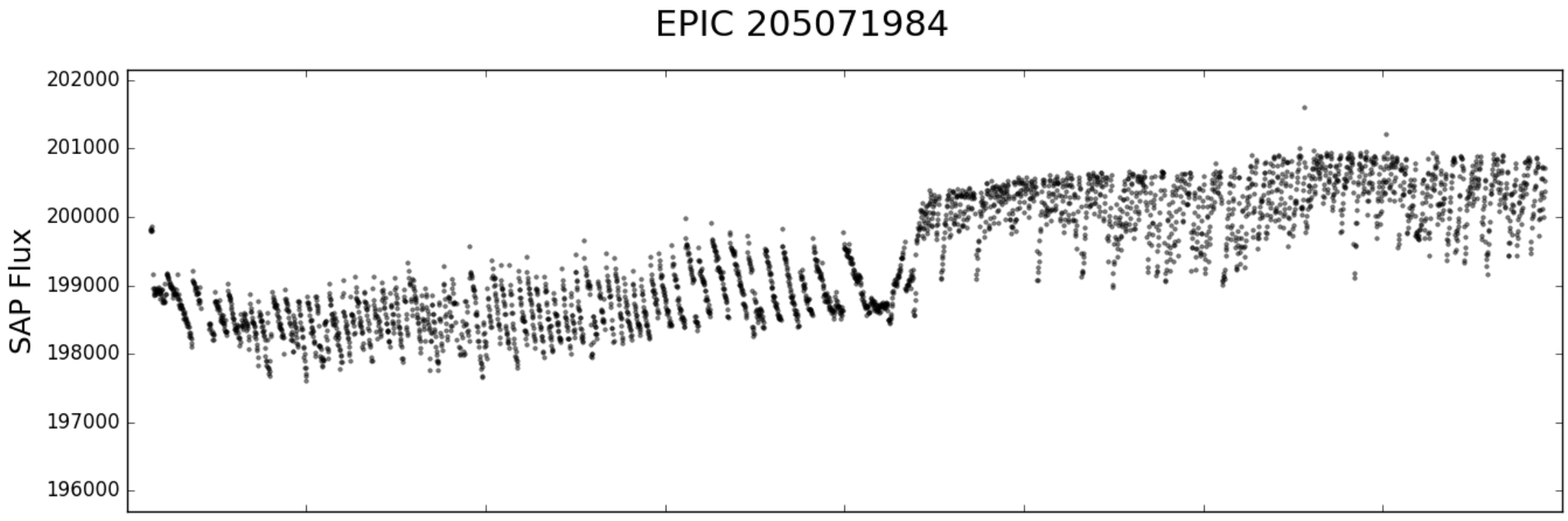




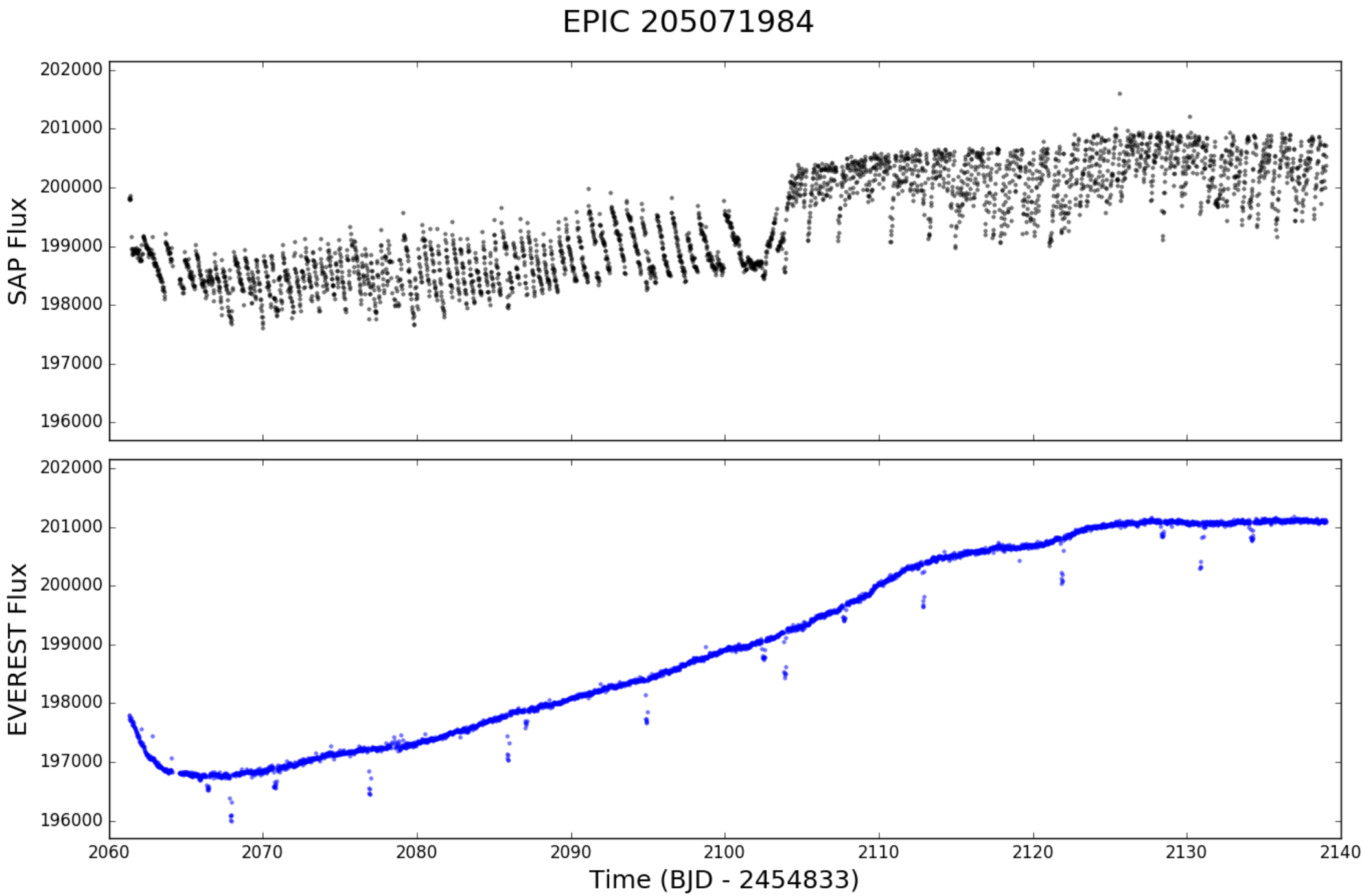
instrumental



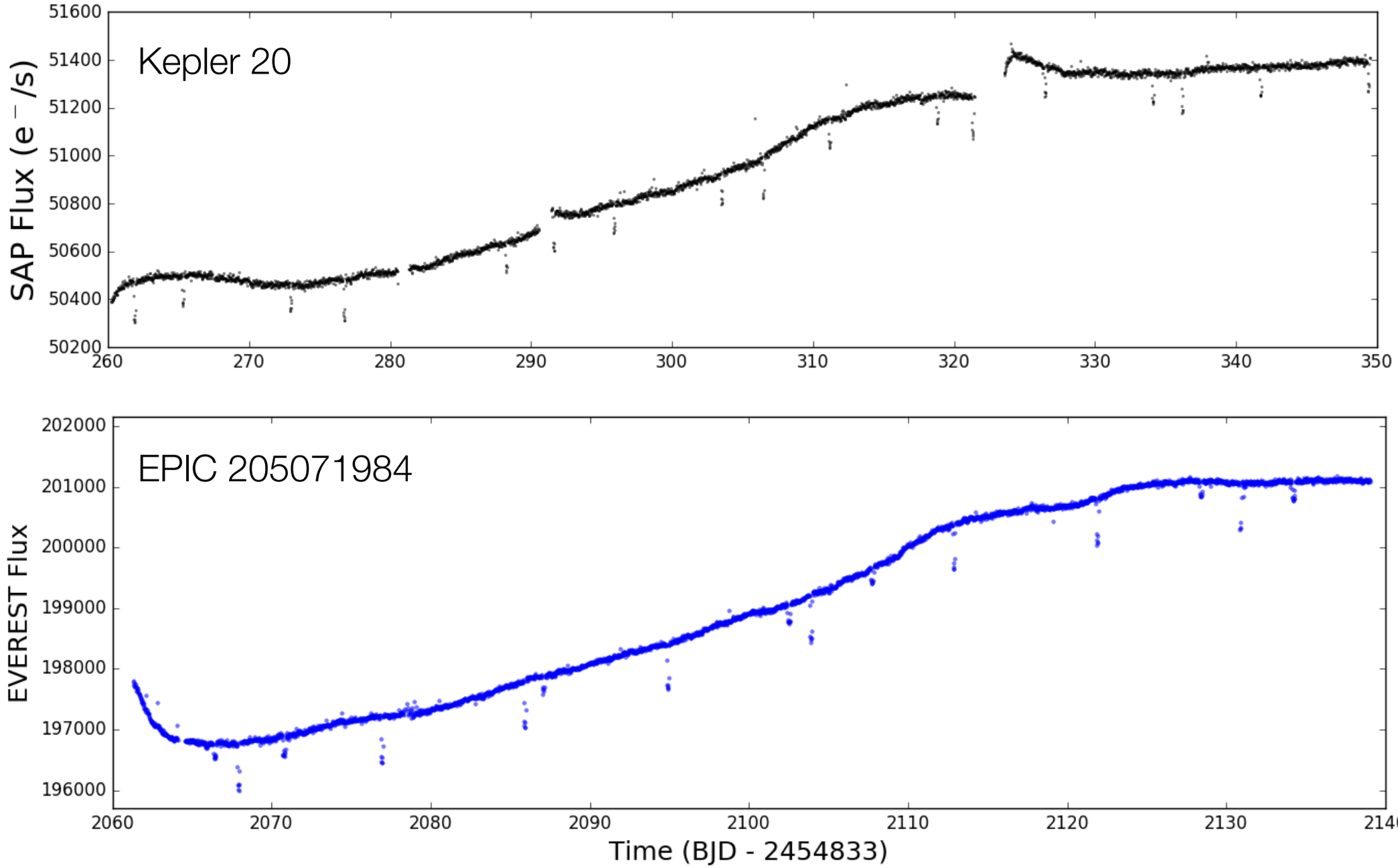
A typical K2 light curve



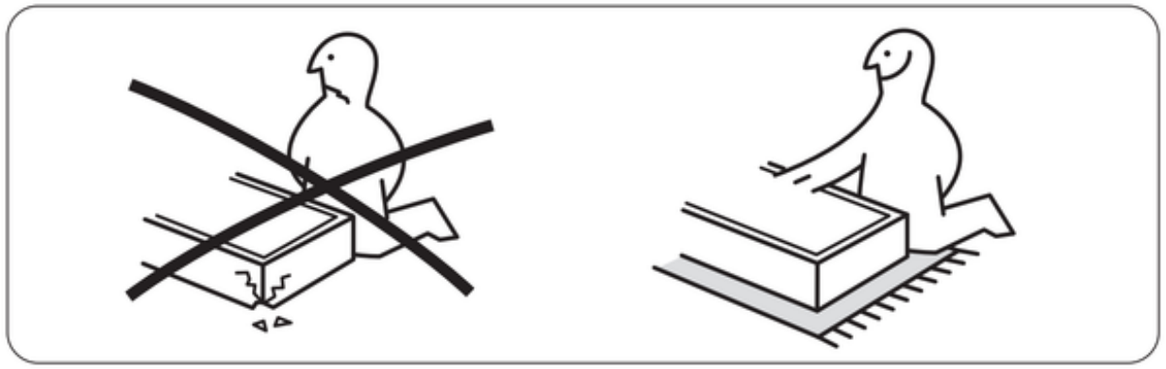
A typical K2 light curve

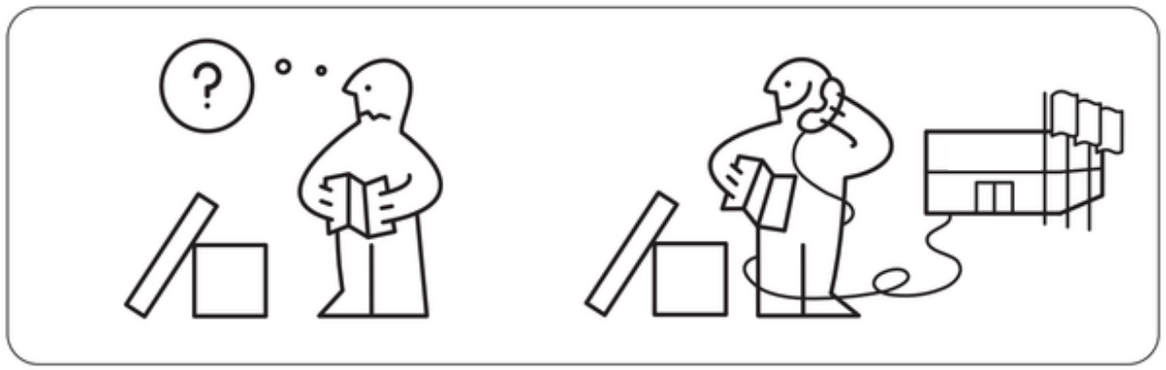


EVEREST regains Kepler precision for bright stars



3





User guide

and quick start tips

Installation

Using `everest` is easy once you get it set up.

```
git clone --recursive https://github.com/rodluger/everest
cd everest
python setup.py install --user
```

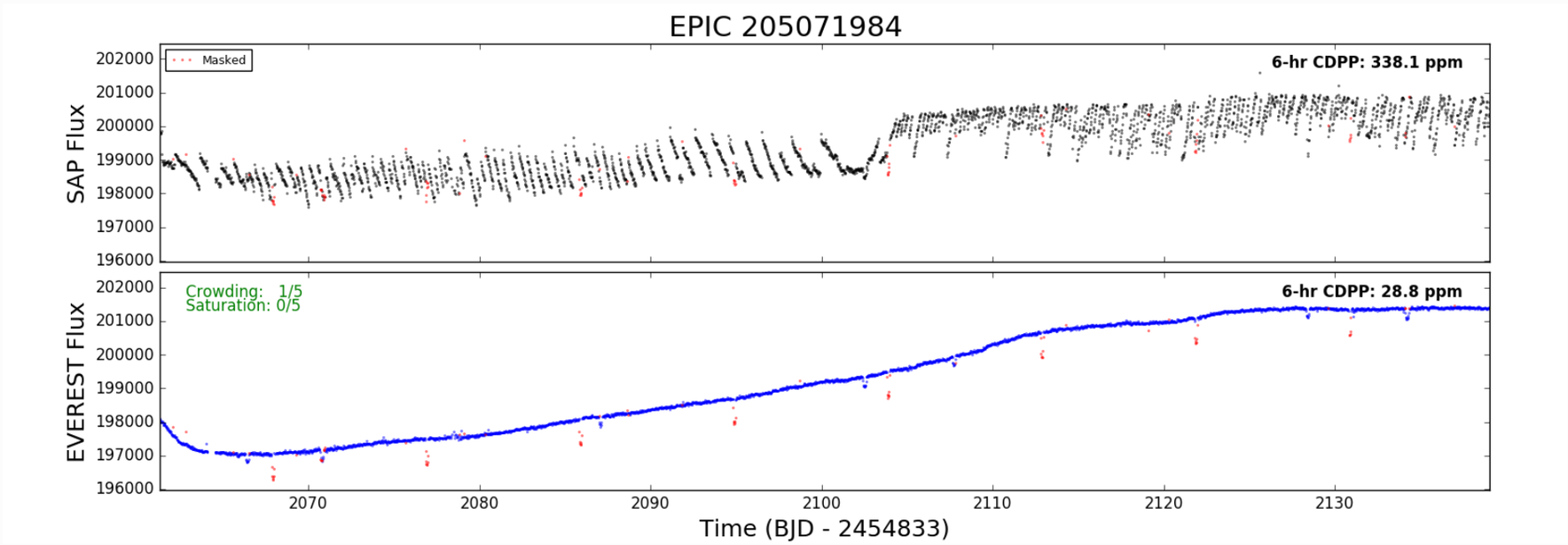
Quick Access

For quick access to the light curves in the catalog, you can use the `everest` command line tool.

Open up a terminal and call

```
everest 205071984
```

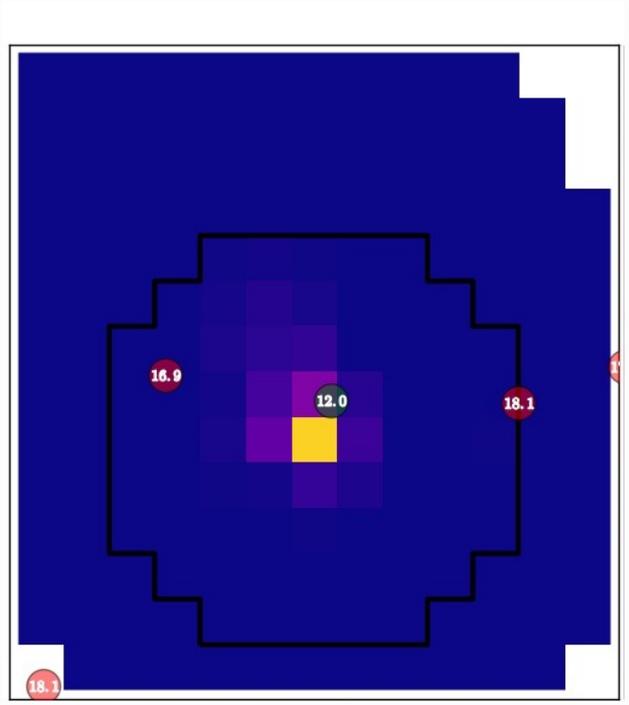
to bring up a window with the raw and de-trended light curves for EPIC 205071984:

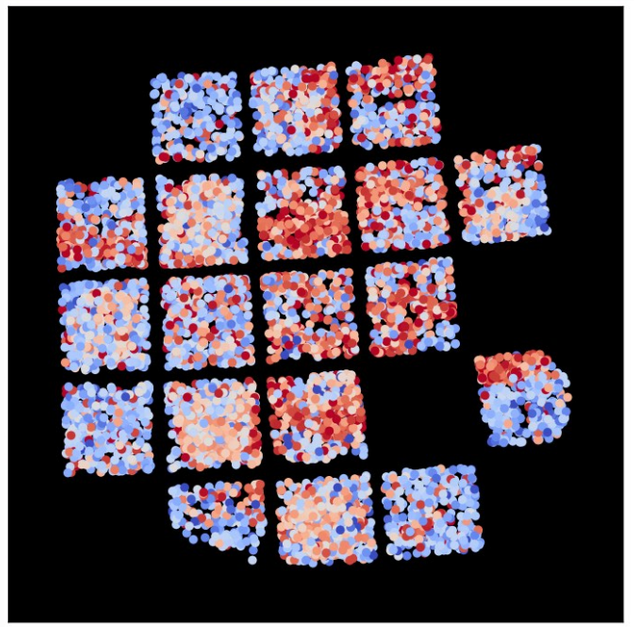


http://staff.washington.edu/rodluger/everest_docs

Quick Access

`id`	The **EPIC** target number *or* the *K2* campaign number (*required*)
`-a`	Plot the aperture used for the photometry
`-c`	Plot the location of the target on the CCD
`-i`	Plot in interactive mode
`-k`	Plot all stars on the *K2* field of view (*interactive*)
`-n`	Plot the contamination analysis
`-p`	Plot the **EVEREST** de-trended light curve (*default*)
`-r`	Plot the autocorrelation/GP fitting result
`-s`	Plot the postage stamp for the target
`-x`	Plot the cross-validation analysis
`--k2sff`	Plot the **K2SFF** light curve for the target
`--k2sc`	Plot the **K2SC** light curve for the target
`--k2varcat`	Plot the **K2VARCAT** light curve for the target



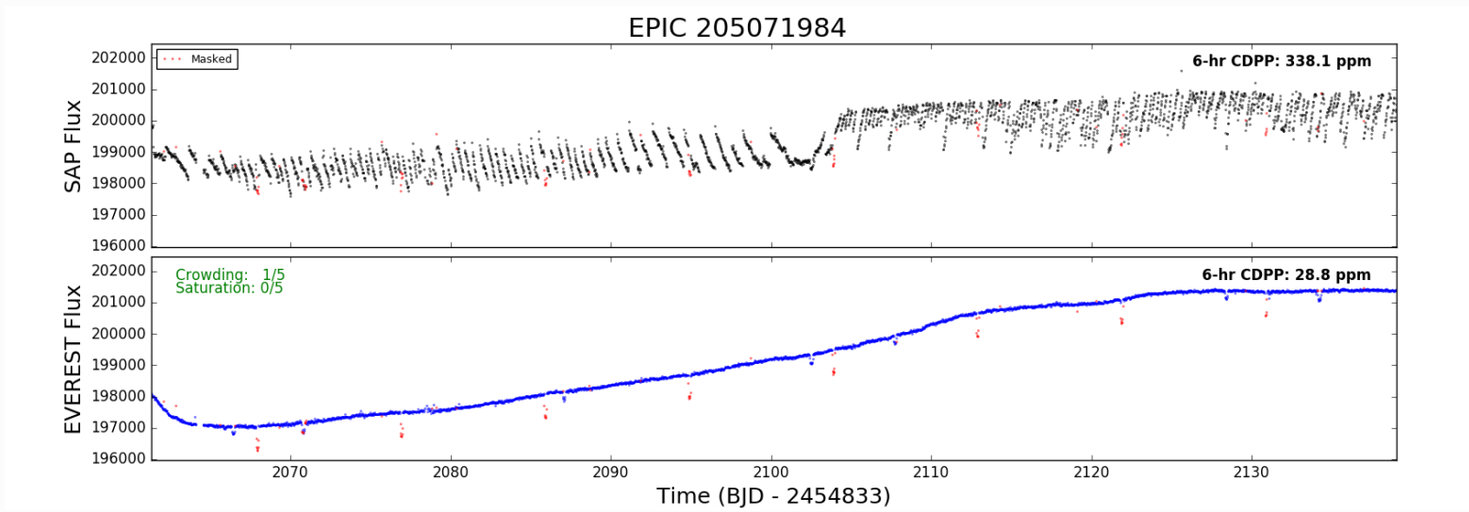


User Tools

```
from everest import Everest
star = Everest(205071984)
```

```
time = star.time
flux = star.flux
```

```
import matplotlib.pyplot as pl
star.plot()
pl.show()
```



Custom Masks

If you're using `everest` for exoplanet/eclipsing binary science, you will likely want to apply a mask to any transits in the light curve to prevent the transits from getting washed out by the least-squares fitting step. The de-trended light curves provided in the catalog automatically mask out large outliers, but it is still strongly recommended that all transits be masked during the de-trending step to minimize de-trending bias. This can be done **easily** and **quickly** as follows:

```
star.set_mask(transits = [( 8.992, 2067.93, 0.25),
                          (20.661, 2066.42, 0.25),
                          (31.716, 2070.79, 0.25)])
```

In the snippet above, we tell `everest` to mask three transiting planets. The first element in each sublist is the period, in days, of the planet. The second element is the time of first transit (BJD - 2454833), and the third is the total transit duration, also in days.

https://archive.stsci.edu/prepds/everest/

Barbara A. MIKULSKI ARCHIVE FOR SPACE TELESCOPES

MAST | STScI | Tools | Mission Search | Search Website | Follow Us | Register | Forum

About MAST | Getting Started

EPIC Variability Extraction and Removal for Exoplanet Science Targets ("EVEREST")

Luger et al. 2016, arXiv:1607.00524

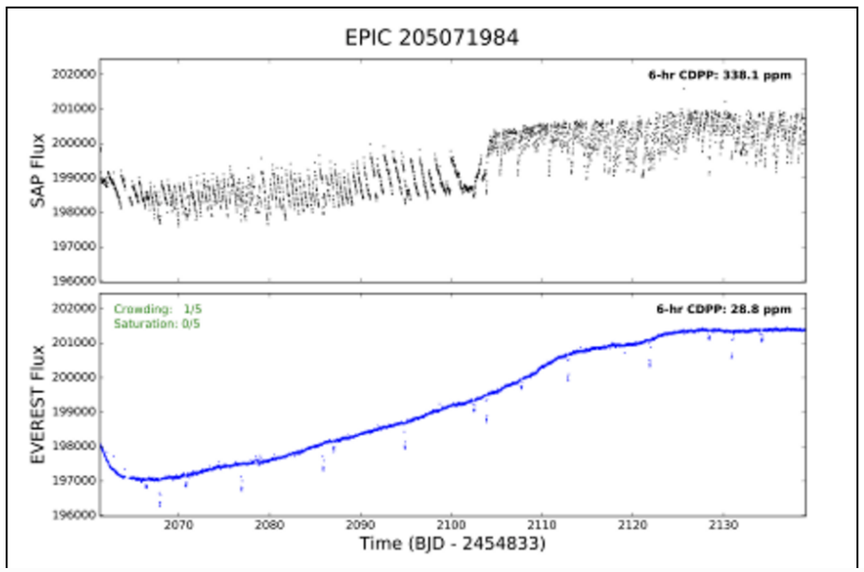
Source code available on GitHub. Documentation is available at MAST (applicable to data archived here) and at Washington (updated by the team as their software evolves).

Introduction | **Description of Data Products** | **Data Access** | **Download README**

Introduction

EVEREST is an open-source pipeline for removing instrumental systematics from K2 light curves, using a combination of pixel-level decorrelations to remove spacecraft pointing error and Gaussian processes to capture astrophysical variability. Corrected light curves have precision comparable to the original Kepler mission for targets brighter than Kp = 13, and within a factor of 2 for fainter targets. Transit injection and recovery tests have been performed to validate the pipeline, and comparisons with other K2 detrended HLSPs is performed (consult the Luger et al. 2016 paper for further details).

NOTE: Since EVEREST performs least-squares fits to reduce the noise in K2 light curves, astrophysical features such as transits and eclipses can sometimes be slightly shallower in the de-trended dataset. In order to prevent this, EVEREST automatically masks outliers prior to computing the fits. However, low signal-to-noise



Everest 2.0: Late January 2017

- ~10-20% CDPP improvement
- De-trend saturated stars
- Less over-fitting of variable stars
- Short cadence light curves

Paper ..... The Astronomical Journal, 152:100 (14pp), 2016 October

User-friendly code ..... https://github.com/rodluger/everest

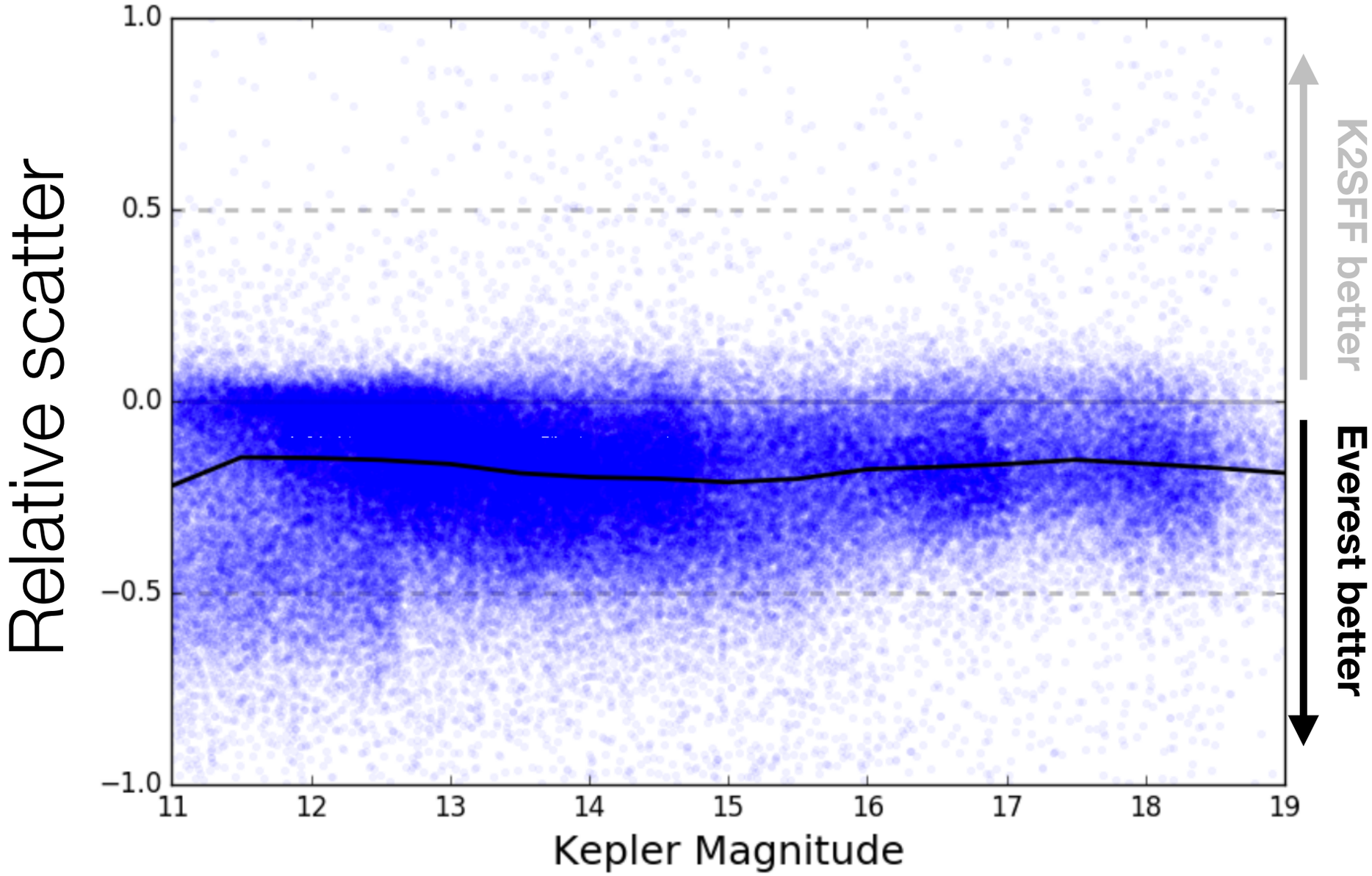
Documentation ..... http://staff.washington.edu/rodluger/everest_docs

Light curves ..... https://archive.stsci.edu/prepds/everest

4

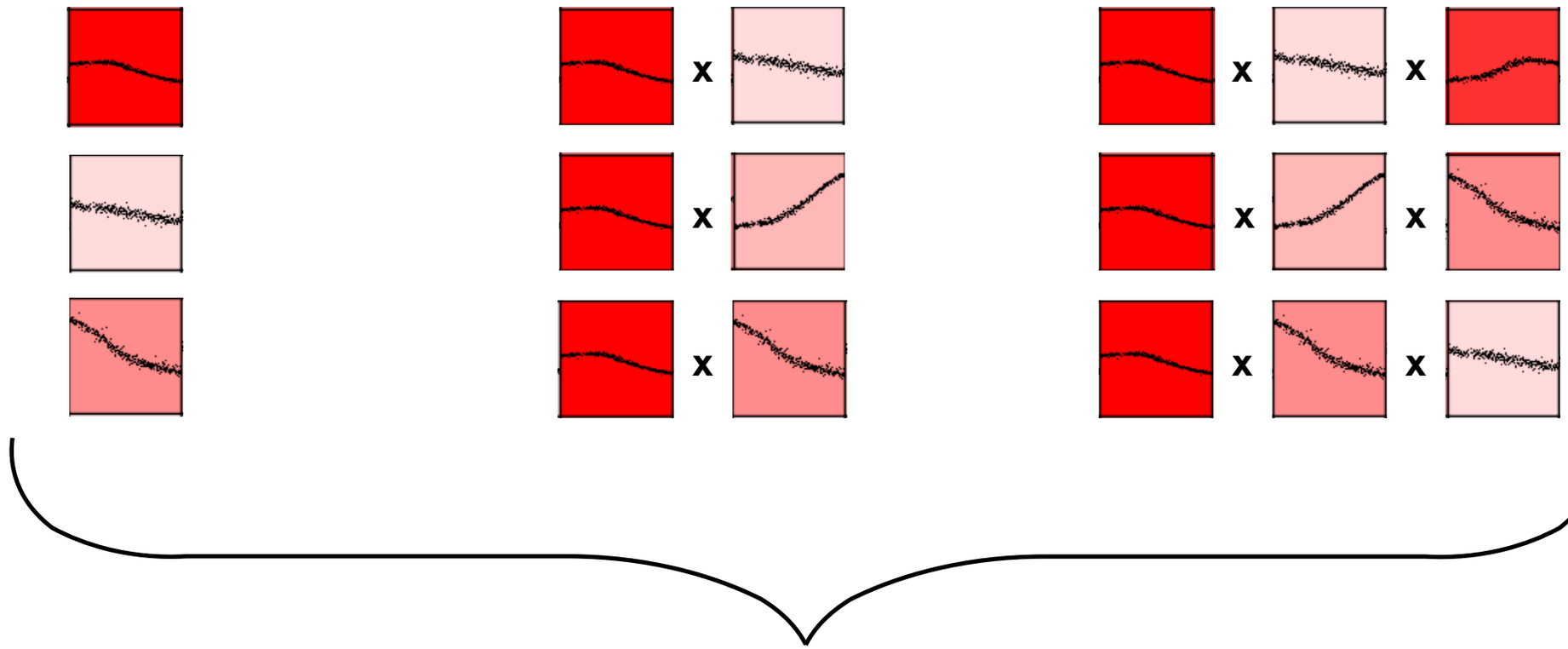
Backup slides

EVEREST vs. K2SFF



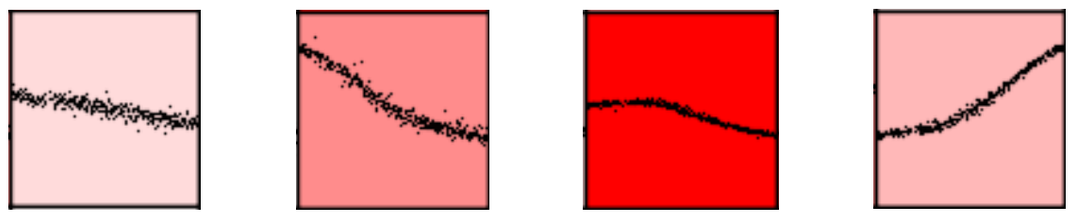
EVEREST: *EPIC Variability Extraction and Removal for Exoplanet Science Targets*

1st order *2nd order* *3rd order*



design matrix

$$\mathbf{X} = \left(\quad \right)$$



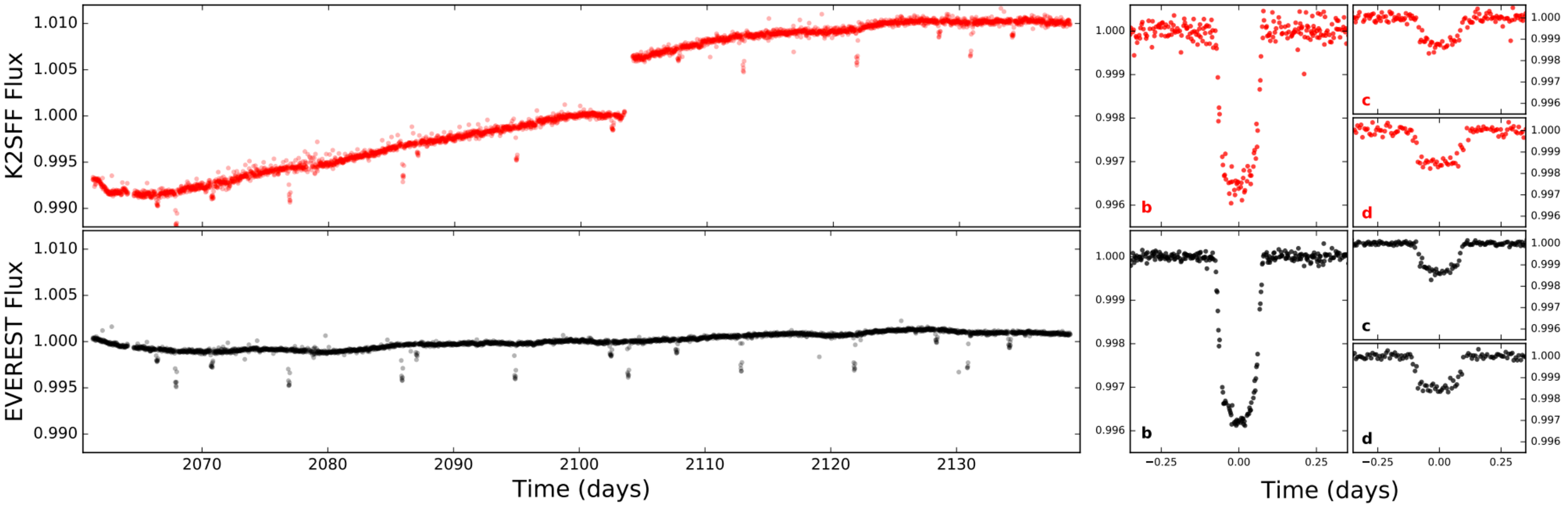
model

$$\mathbf{m} = \mathbf{X} \cdot \mathbf{c}$$

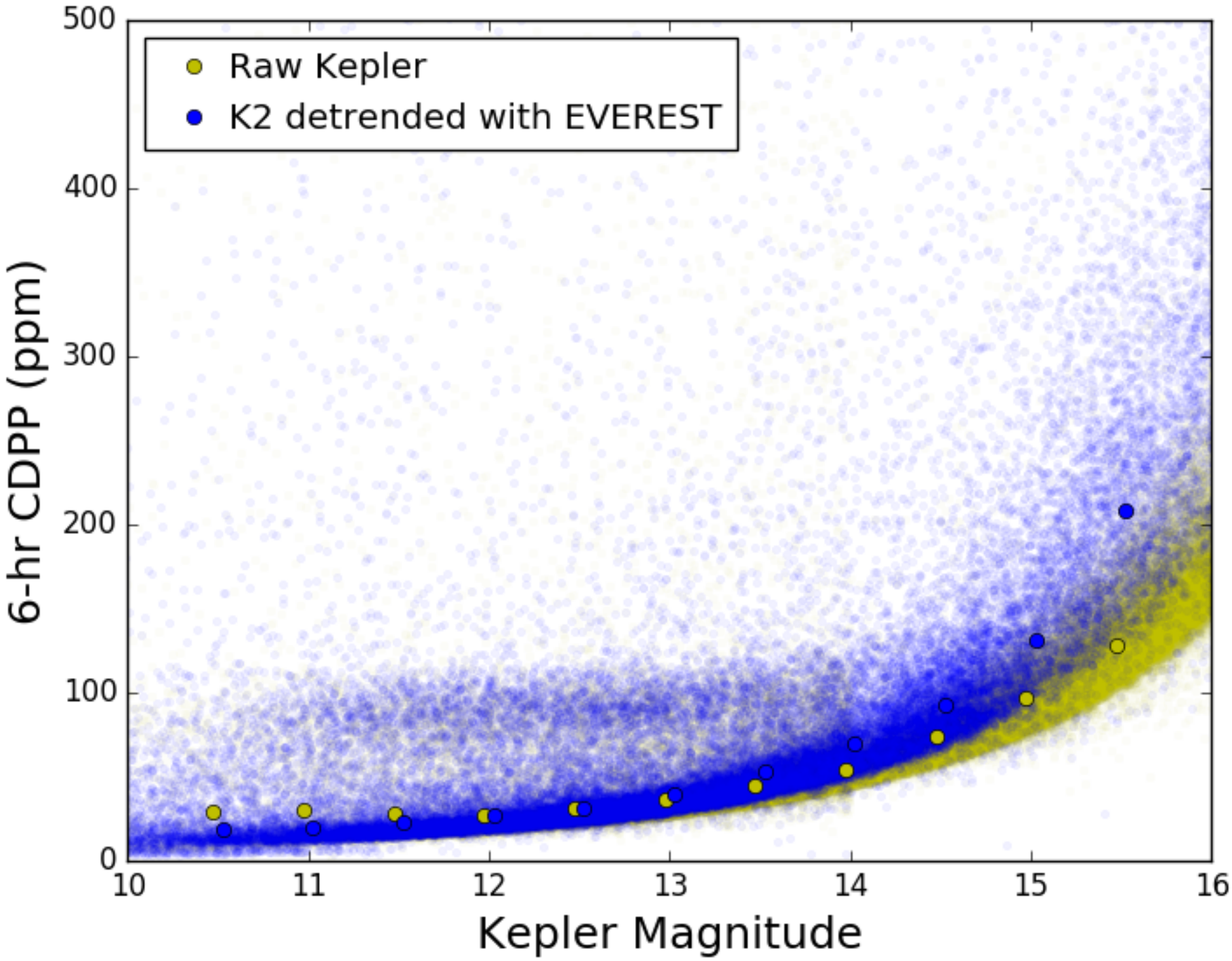
$$\mathbf{c} = \left(\mathbf{X}^\top \cdot \mathbf{K}^{-1} \cdot \mathbf{X}\right)^{-1} \cdot \left(\mathbf{X}^\top \cdot \mathbf{K}^{-1} \cdot \mathbf{y}\right)$$

covariance matrix (Gaussian Process) *flux*

EVEREST vs. K2SFF



EVEREST


Raw Kepler
K2 detrended with EVEREST
6-hr CDPP (ppm)
500
400
300
200
100
0
10
11
12
13
14
15
16
Kepler Magnitude

Cross-validation