Anisotropic Gaussian Processes for PSF modeling
&
Implementation within Piff

Pierre-François Léget - LPNHE/IN2P3

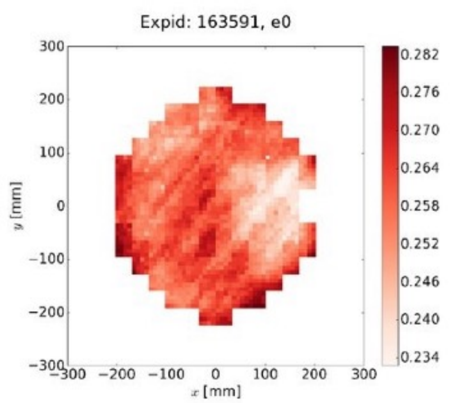
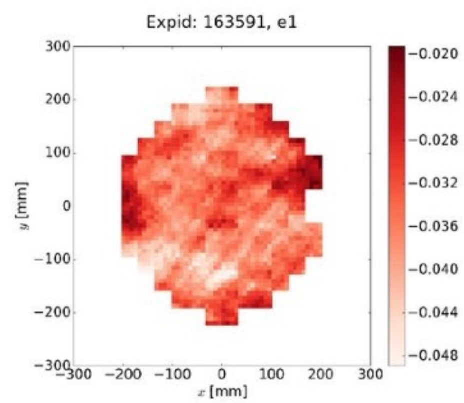
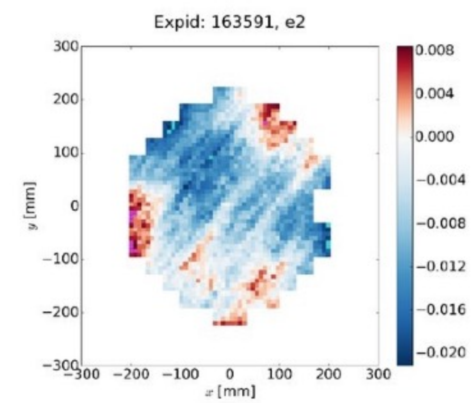
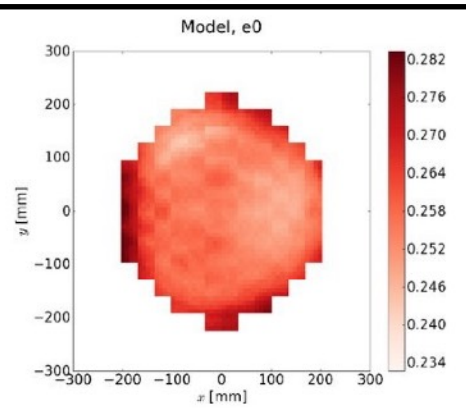
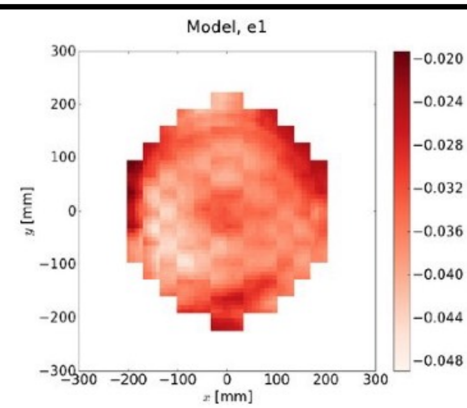
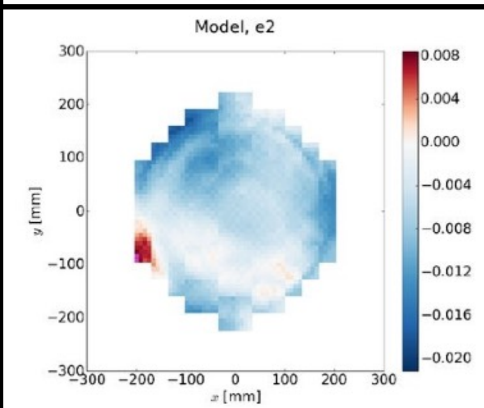
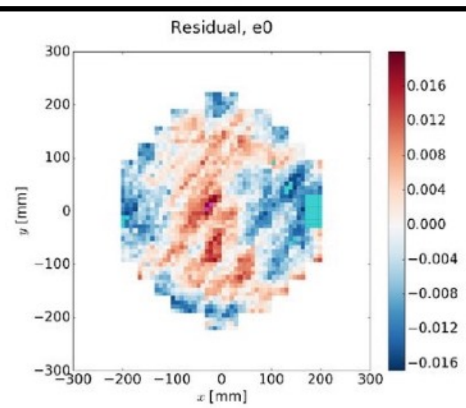
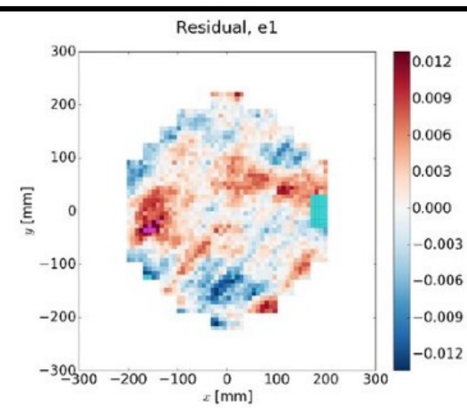
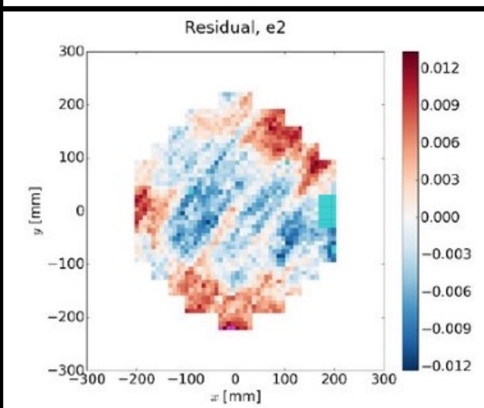
PSFs In the Full FoV (Piff) package

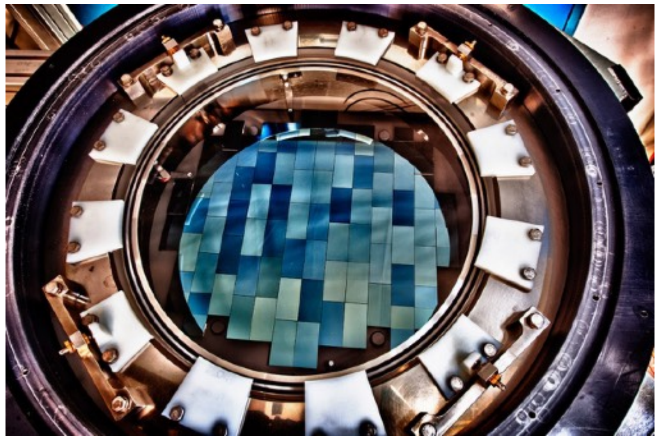
- Piff is a new python software for PSF estimation developed initially to replace PSFex in DES and now also developed for LSST
- Modular package where it is easy to implement new PSF modeling and interpolation scheme over the FoV
- Package with unit testing and code review
- Will be used for the Weak-Lensing analysis of DES Y3
- Contributors:

Mike Jarvis, Chris Davis, Josh Meyers, Pierre-François Léget, Erin Sheldon, Gary Bernstein, Aaron Roodman, Pat Burchat, Daniel Gruen, Ares Hernandez, Andres Navarro, Flavia Sobreira, Reese Wilkinson, Joe Zuntz, Sarah Burnett

PSF decomposition

For a given exposure

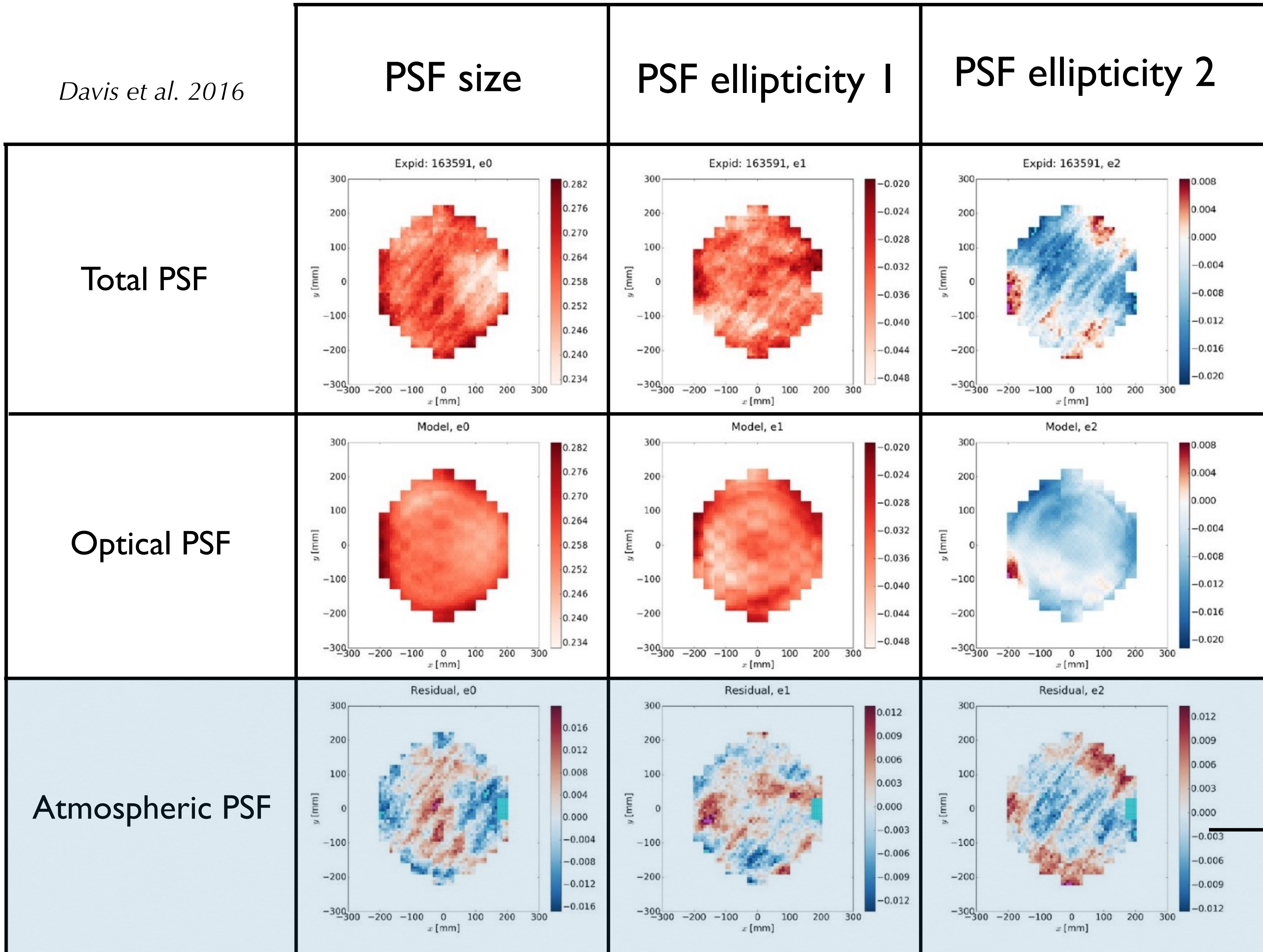
Davis et al. 2016	PSF size	PSF ellipticity 1	PSF ellipticity 2
Total PSF			
Optical PSF			
Atmospheric PSF			

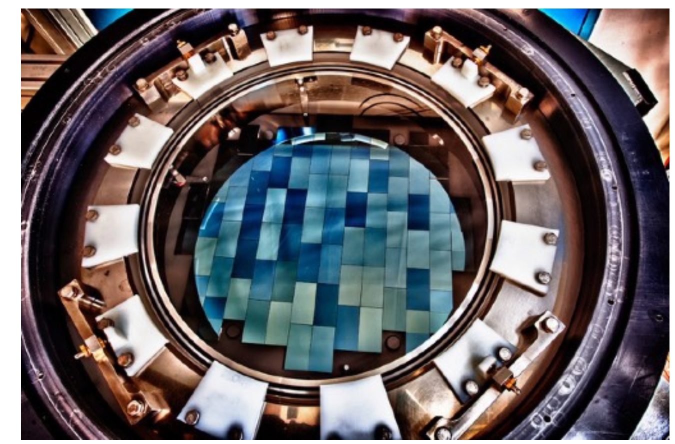


Dark Energy Camera

PSF decomposition

For a given exposure





Dark Energy Camera

Model that as a gaussian processes

(

- Quick introduction to Gaussian process: How it works

  - Gaussian processes are a probabilistic way to interpolate data

  - Interpolation driven by the correlation function (aka kernel) described by “hyperparameters”; e.g.,

    - amplitude of the fluctuations

    - correlation lengths

    - …

  - No problem taking into account measurement uncertainties

  - Natural output is the mean and covariance at desired locations

- Quick introduction to Gaussian process: How it works

1) Choose a correlation function (kernel)

2) Fit hyperparameters

3) Compute interpolation

- Involves inverting large matrices: N*N matrix, where N = number of stars for PSF ($N \sim 10^4$ for LSST full field of view)
    - Many matrix inversions for optimization of hyperparameters.
    - Only one matrix inversion for interpolation!

- Quick introduction to Gaussian Process: example in 1D

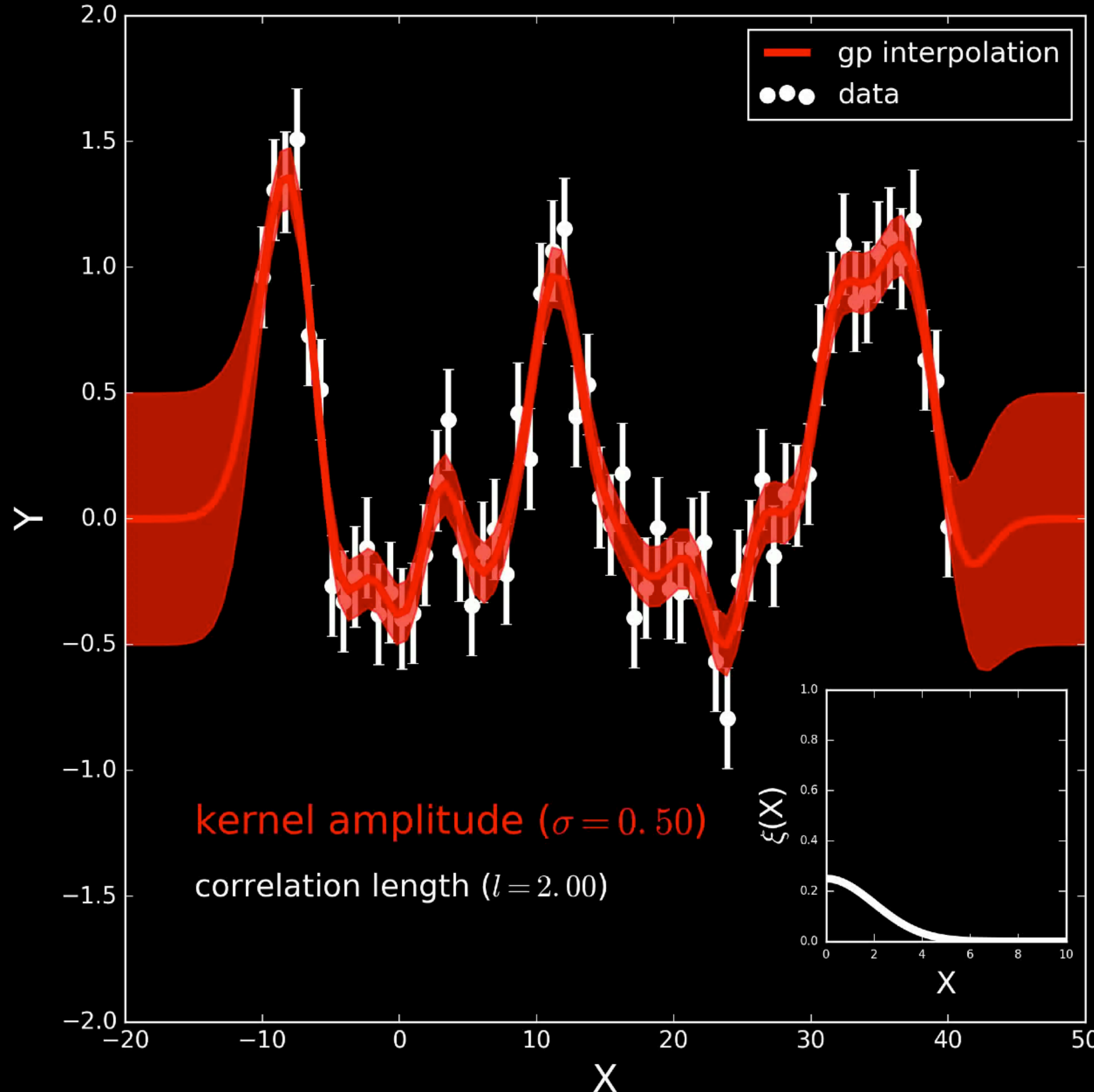
- Data generated with multivariate Gaussian and a squared exponential kernel

$$\xi(x) = \sigma^2 \exp\left(-\frac{1}{2}\left(\frac{x}{l}\right)^2\right)$$

- σ = 0.5, l = 2.0

- Vary hyperparameters used in the gp interpolation

- Animation shows impact of assumed hyperparameters on the interpolation



https://youtu.be/PXA3GhE-PwY

Hyperparameter fitting: the problem

- Classical method is to do a maximum likelihood fit (ML)

$$\mathcal{L} \propto \frac{1}{|\boldsymbol{\xi}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\mathbf{y}^T \boldsymbol{\xi}^{-1}\mathbf{y}\right)$$

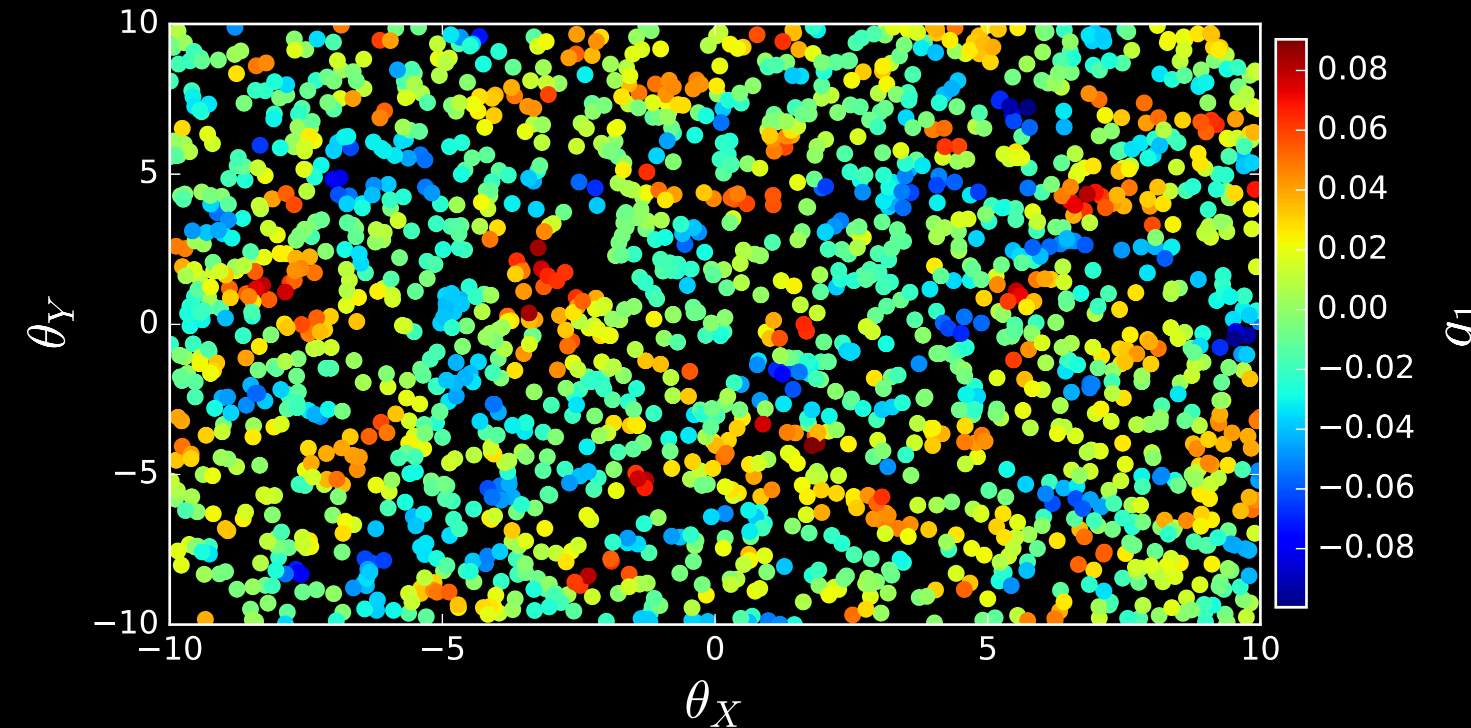
- But you must then invert at each step of the maximization an N*N matrix (N~10^4 for LSST field of view)
- Possibility of local maxima
  - Difficult to determine if hyperparameters are optimal without inverting many N*N matrices.
- Difficult to determine whether kernel was best choice for correlations in data

Hyperparameter fitting: Solution :)

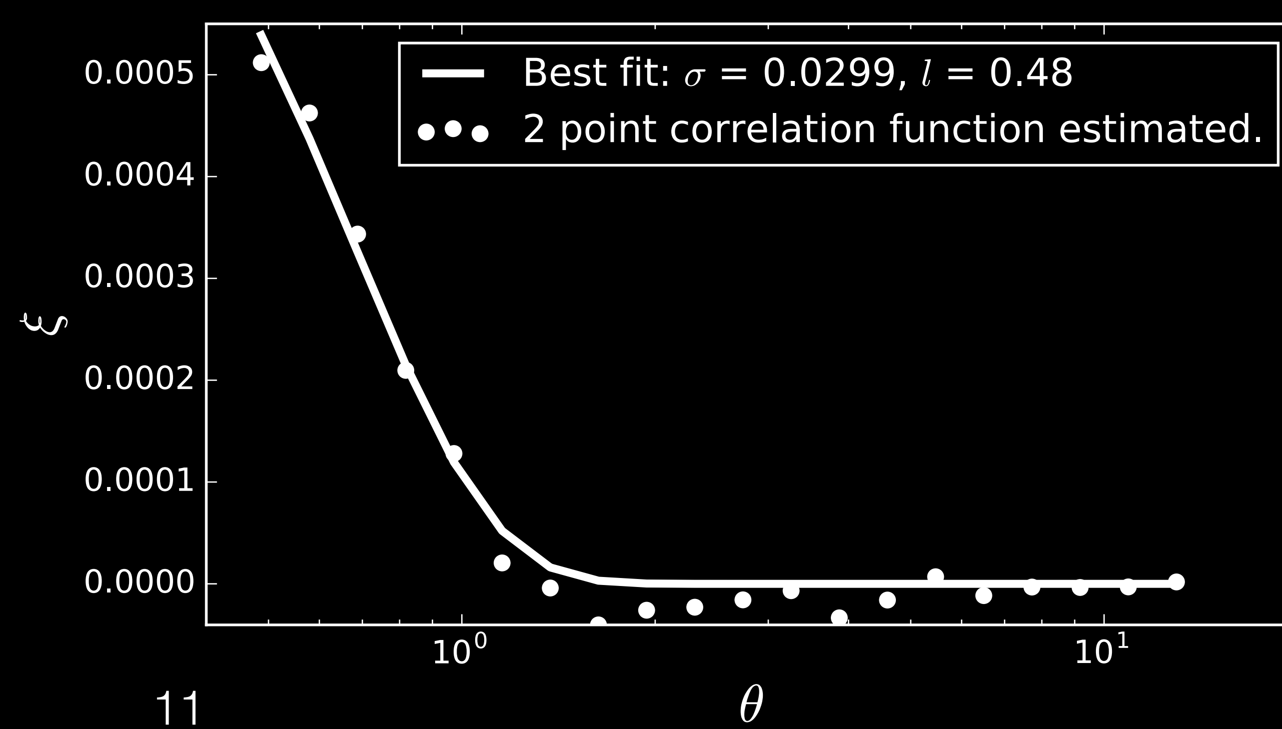
- Basic idea:
  - Instead of using maximum likelihood to find hyperparameters, use two-point correlation function
  - Indeed, Kernel ~ two-point correlation function
  - How it works:
    - Measure two-point correlation function on the data (using treecorr)
    - Fit the (1D) two-point correlation function with different kernels to find the best kernel and optimal hyperparameters
    - Use these hyperparameters to interpolate with gp

Hyperparameter fitting: Solution :)

Gaussian random field
generated with squared exponential kernel
σ =0.03, l=0.5 (2000 points)



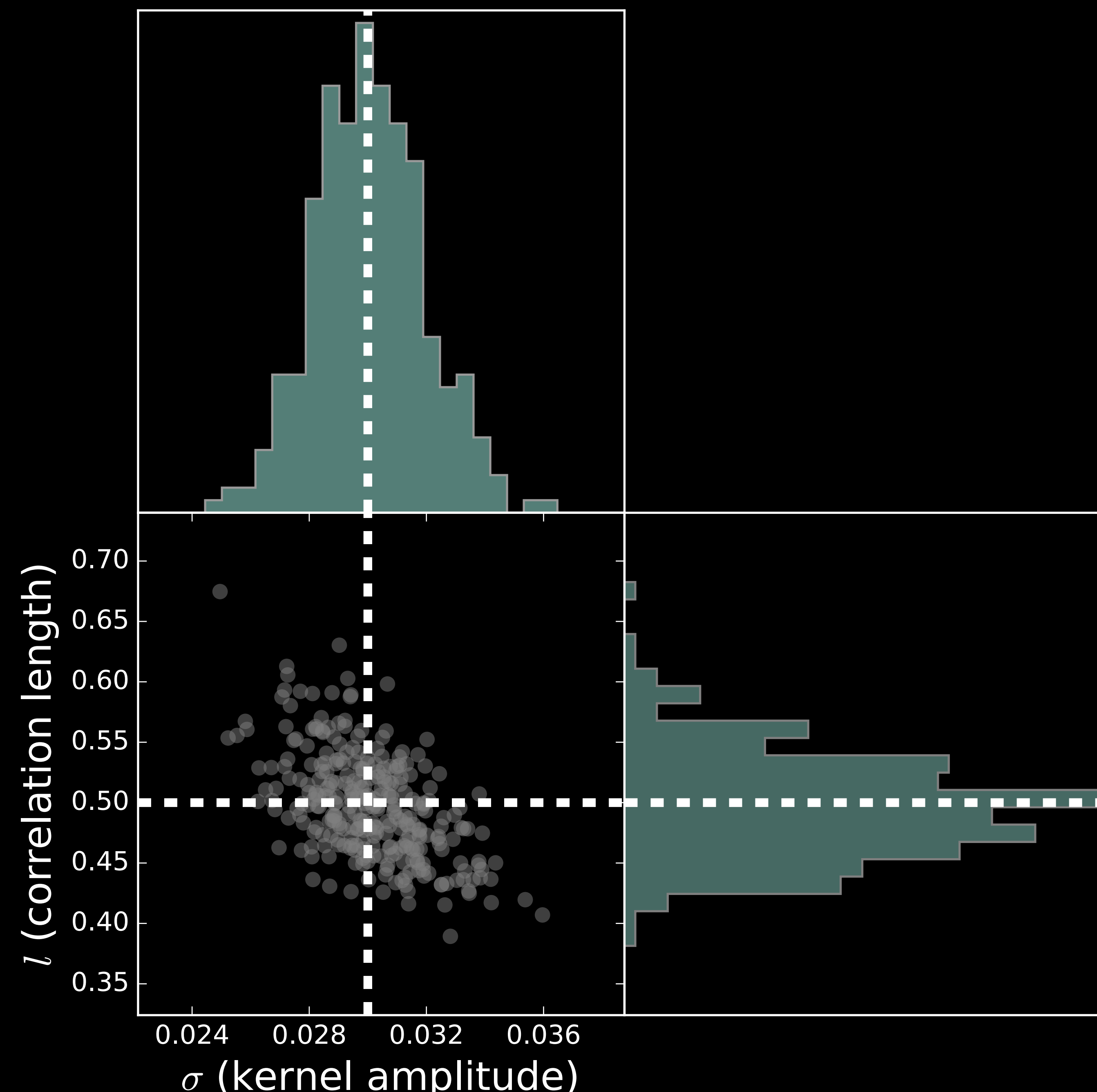
Fit 2-point correlation function
with squared exponential kernel
σ =0.0299, l=0.48



Hyperparameter fitting: Solution :)

300 realizations of
Gaussian random field
generated with squared exponential kernel
σ =0.03, l=0.5

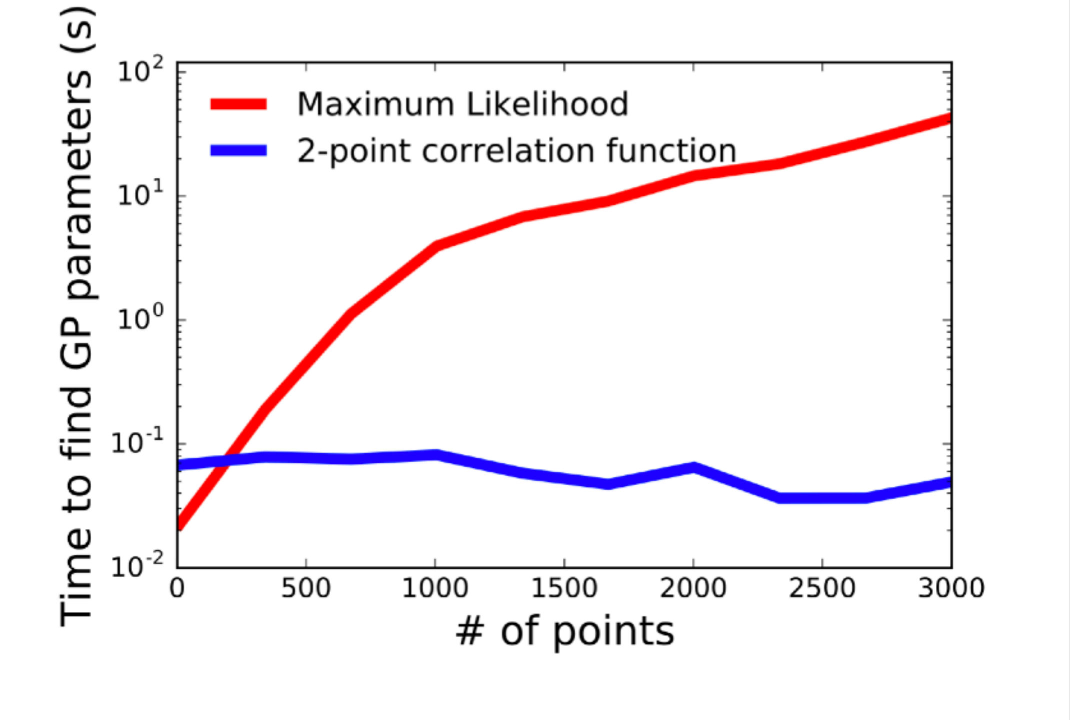
Hyperparameter fit using 2-point
correlation function
with squared exponential kernel for
300 realizations

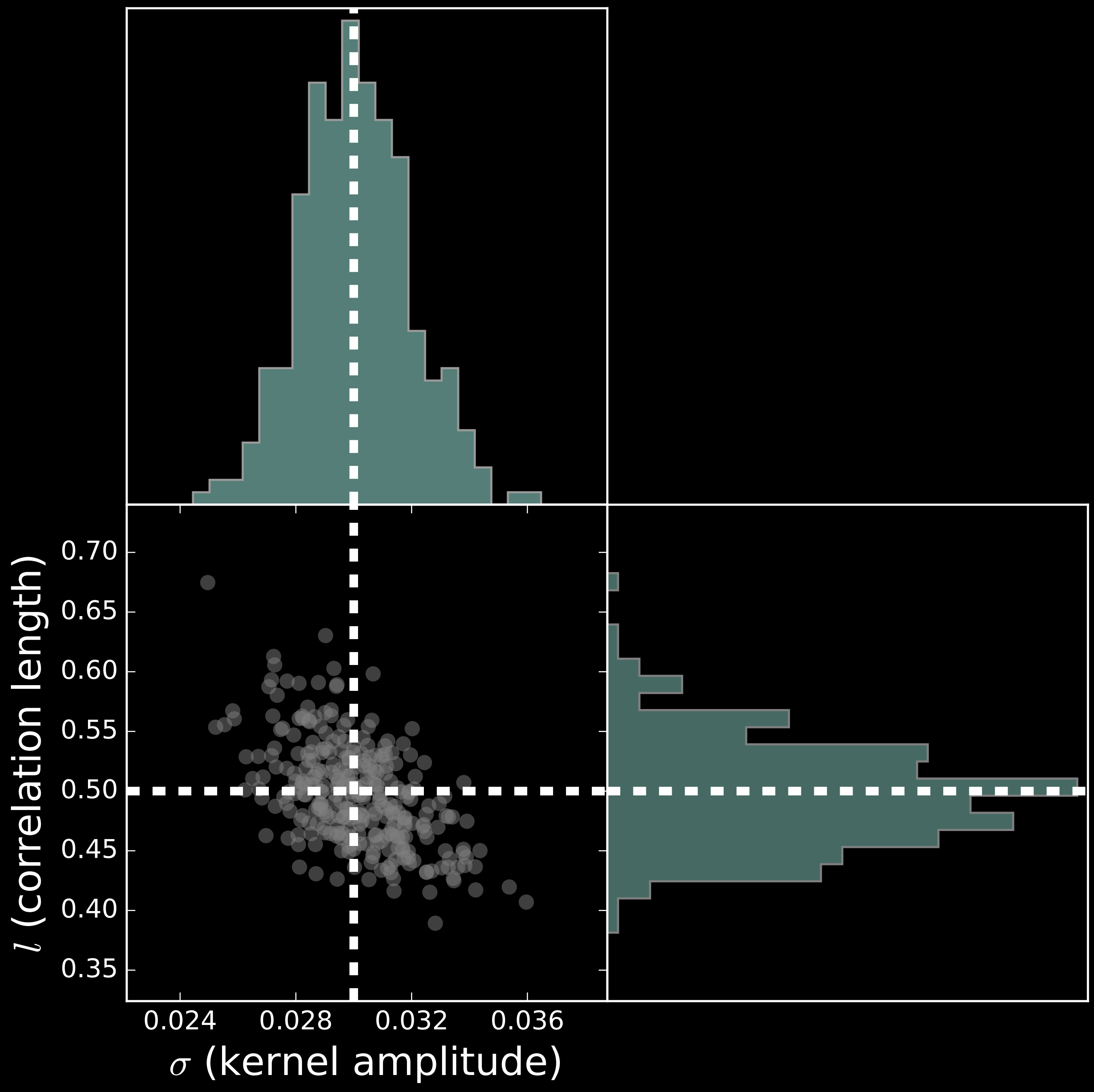


Hyperparameter fitting: Solution :)

300 realizations of
Gaussian random field
generated with squared exponential kernel
σ =0.03, l=0.5

Hyperparameter fit using 2-point
correlation function
with squared exponential kernel for
300 realizations

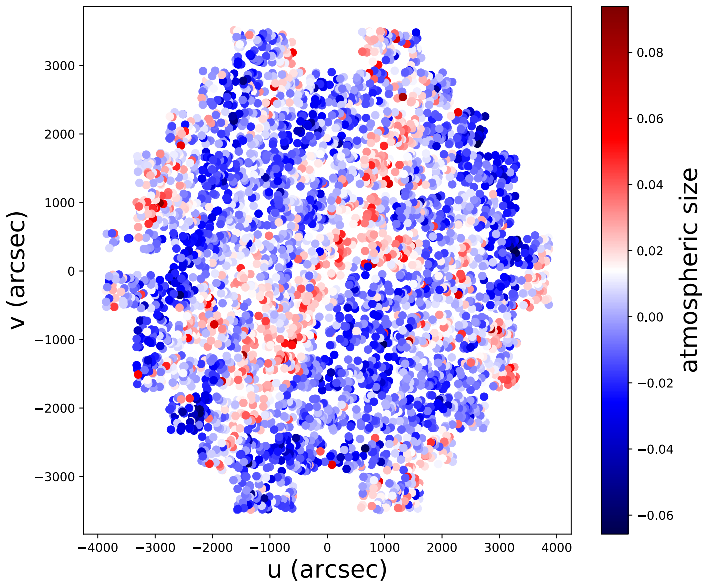




)

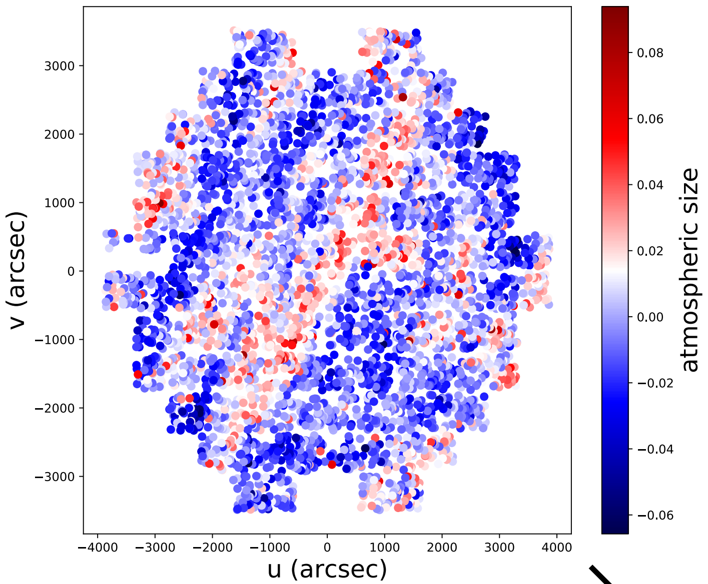
- The current scheme implemented within Piff

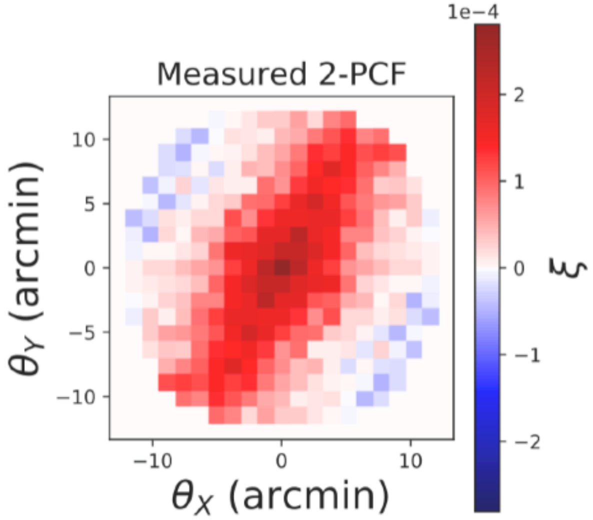
Atmospheric PSF
parameter



- The current scheme implemented within Piff

Atmospheric PSF parameter

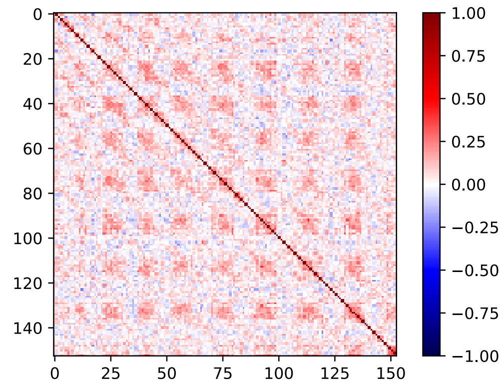




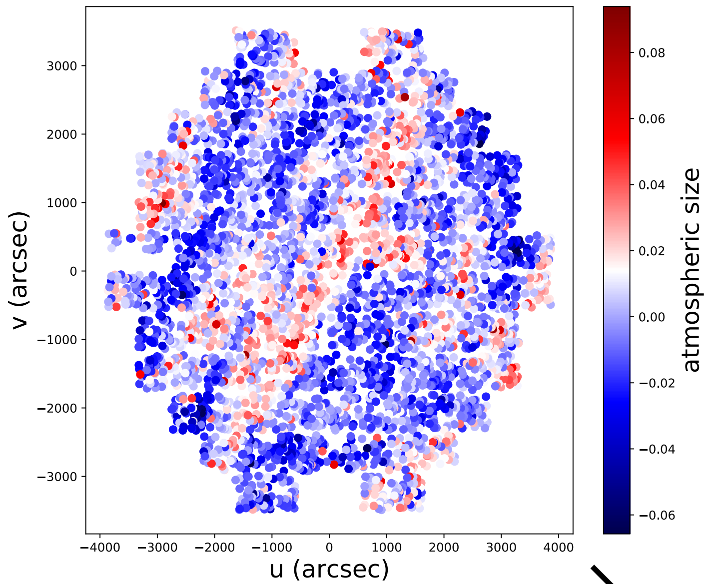
Compute 2D
2-point correlation function
with treecorr

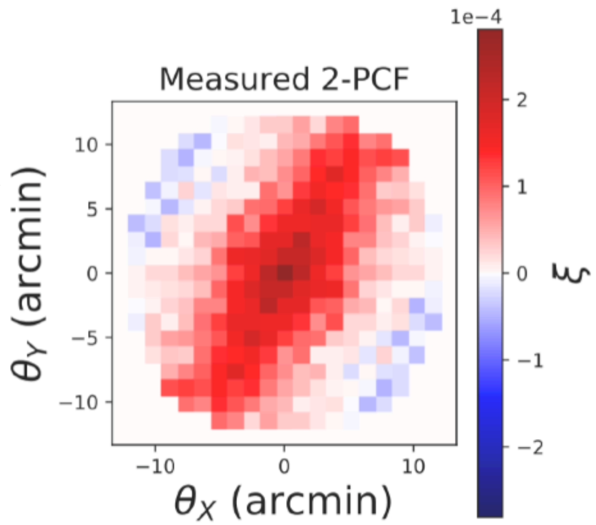
- The current scheme implemented within Piff

Compute covariance matrix using bootstrap resampling



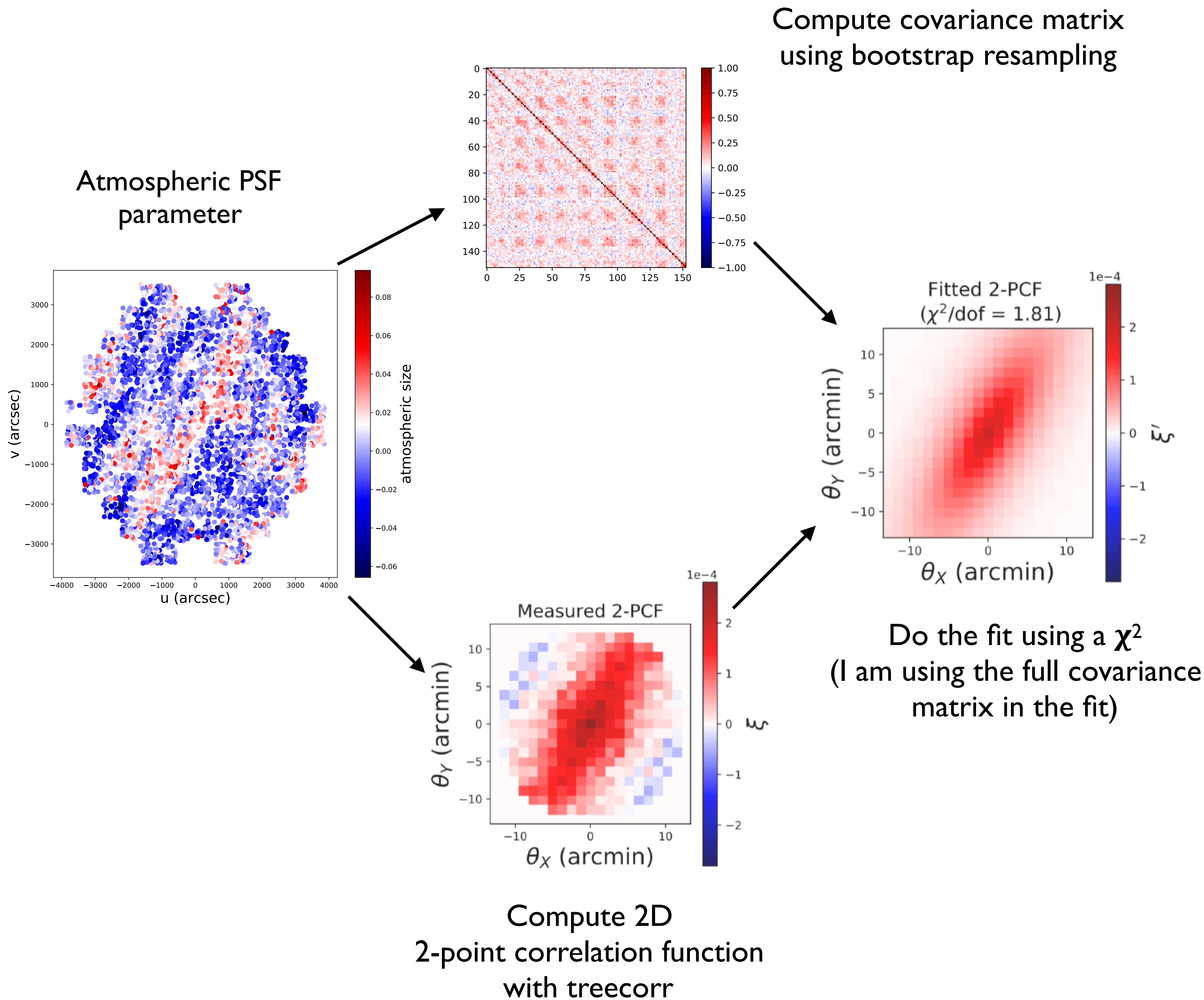
Atmospheric PSF parameter





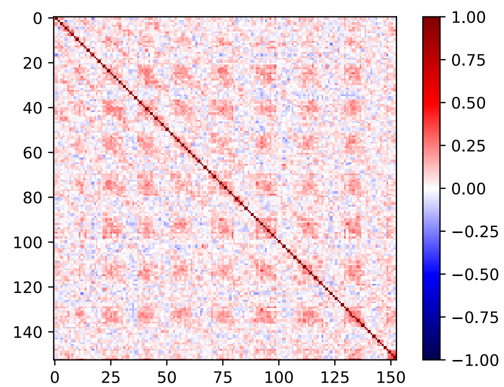
Compute 2D
2-point correlation function
with treecorr

- The current scheme implemented within Piff

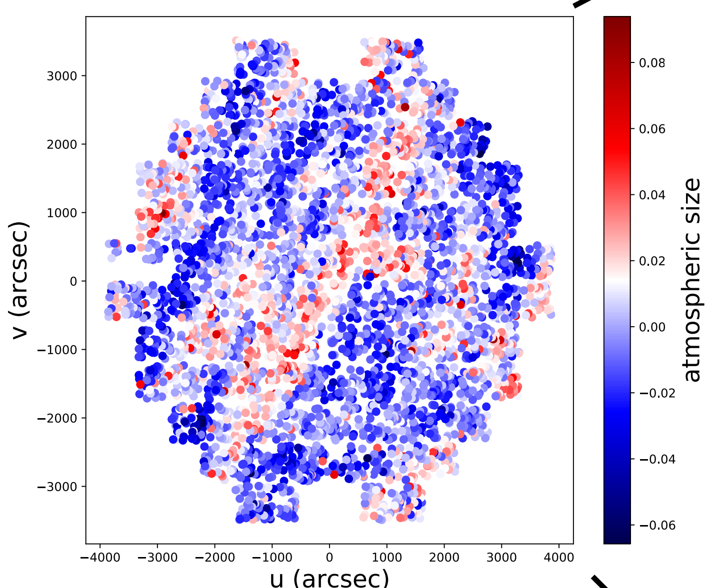


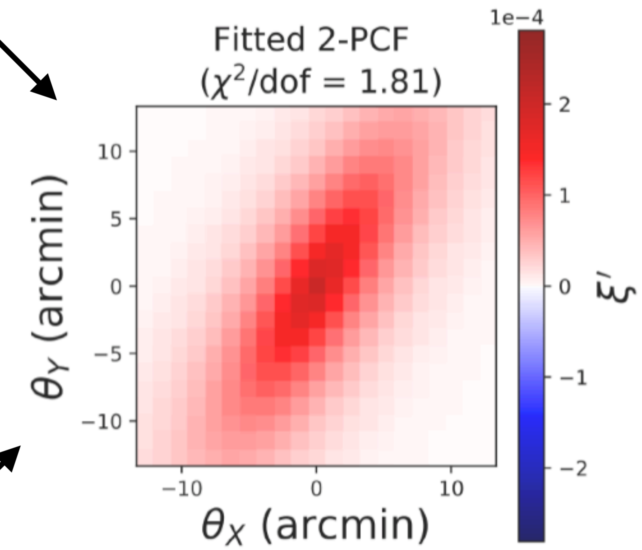
- The current scheme implemented within Piff

Compute covariance matrix using bootstrap resampling



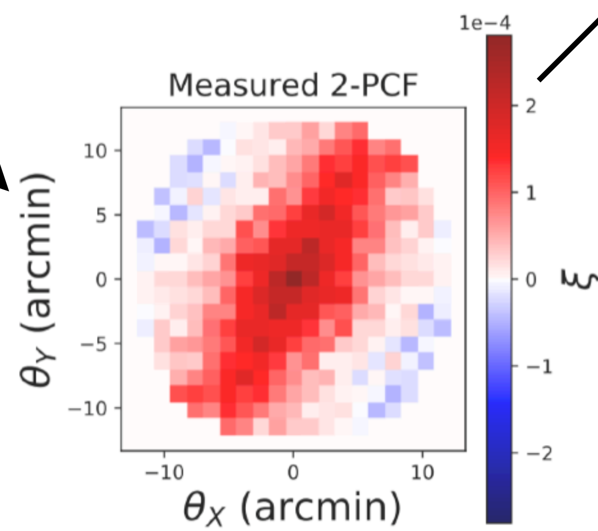
Atmospheric PSF parameter



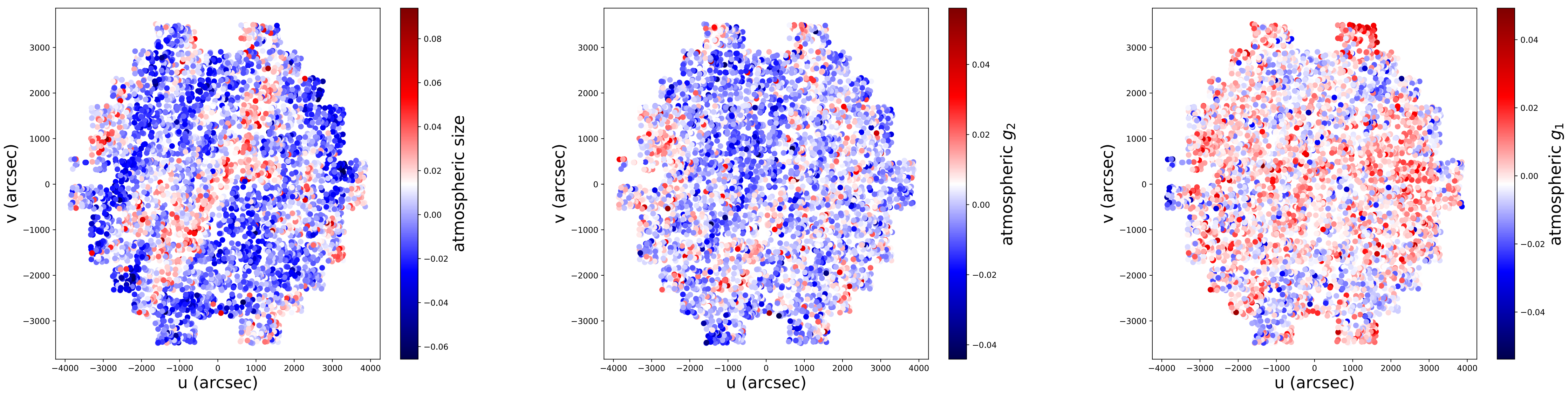


Do GP interpolation with found hyperparameters

Do the fit using a χ^2
(I am using the full covariance matrix in the fit)

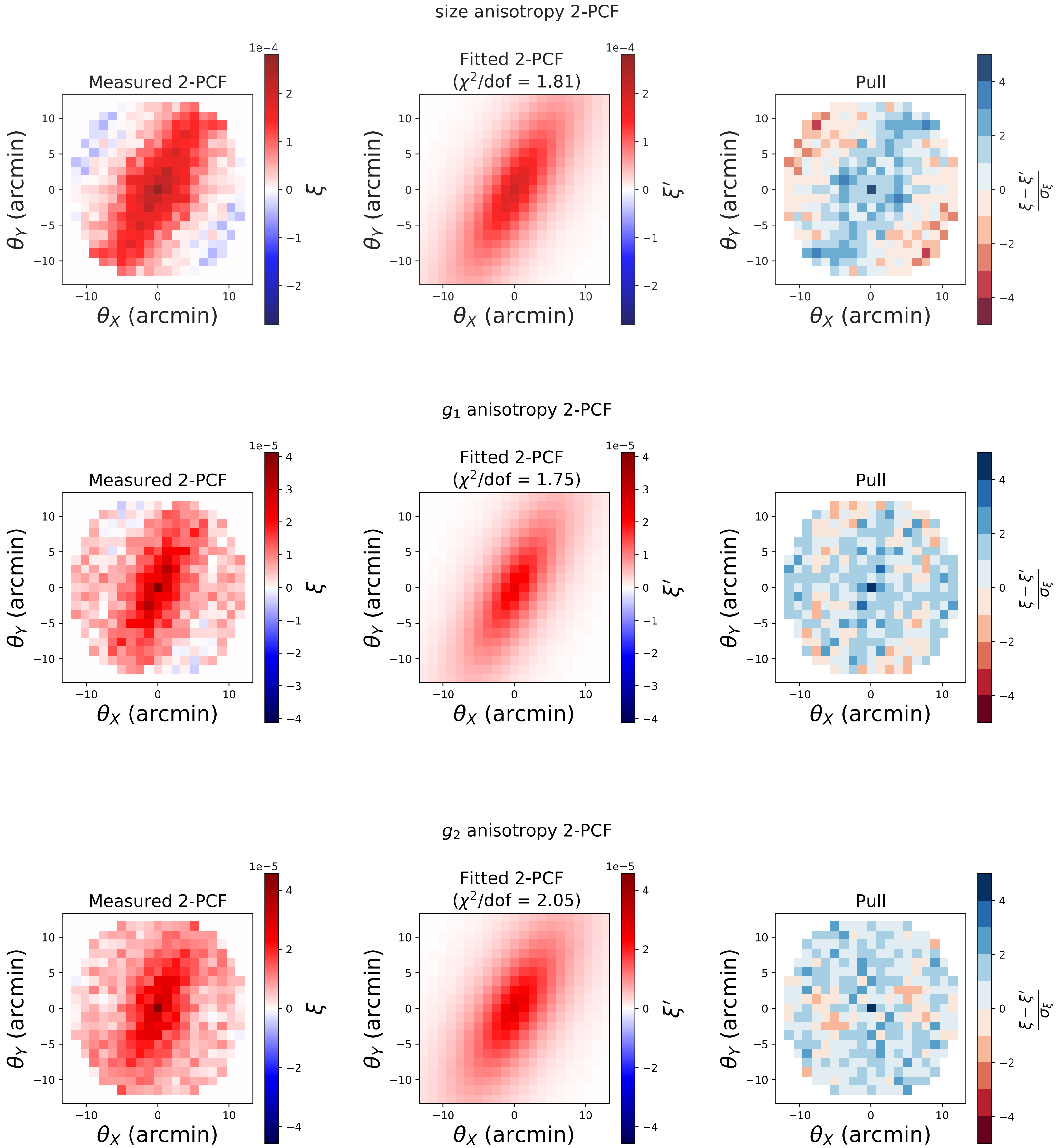


Compute 2D
2-point correlation function
with treecorr

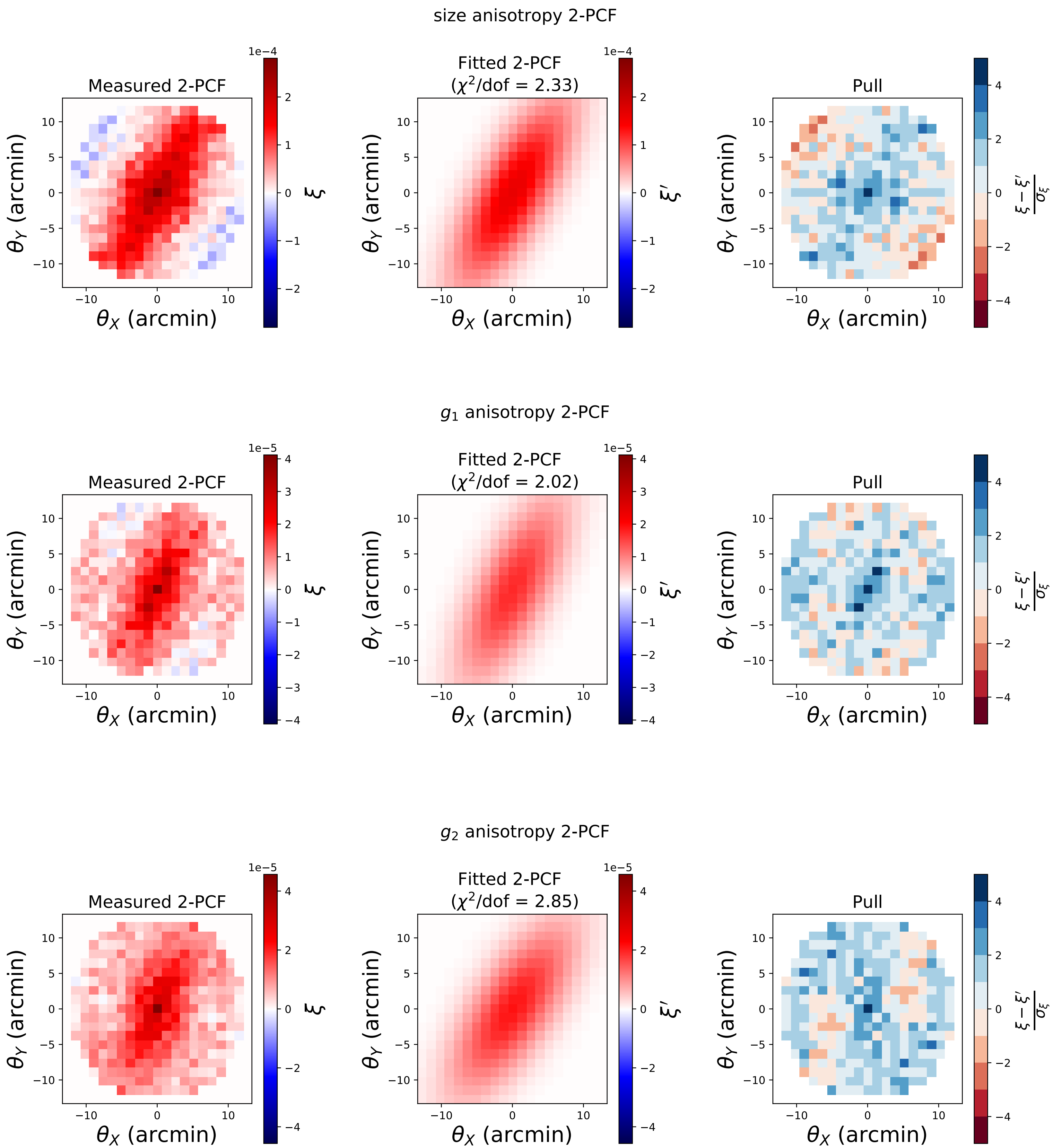


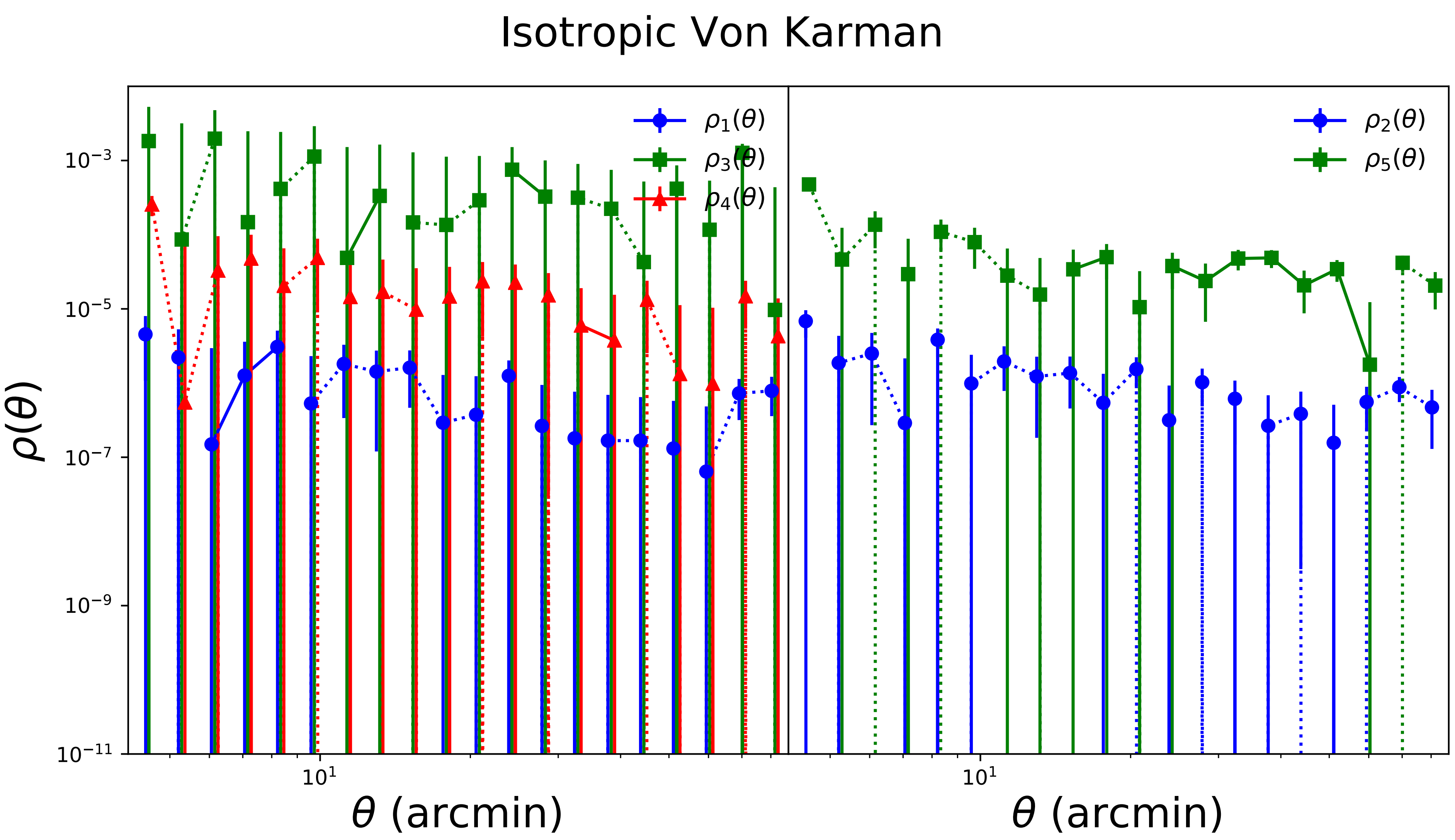
- Exemple on real data (exp id: 510463)
  - Optical model removed and work only on the atmospheric parameters
  - size, g1, g2
- 80% used for training (hyperparameters fitting and interpolation)
- 20% kept for validation to compute Rowe Stat

- Von-Karman correlation function

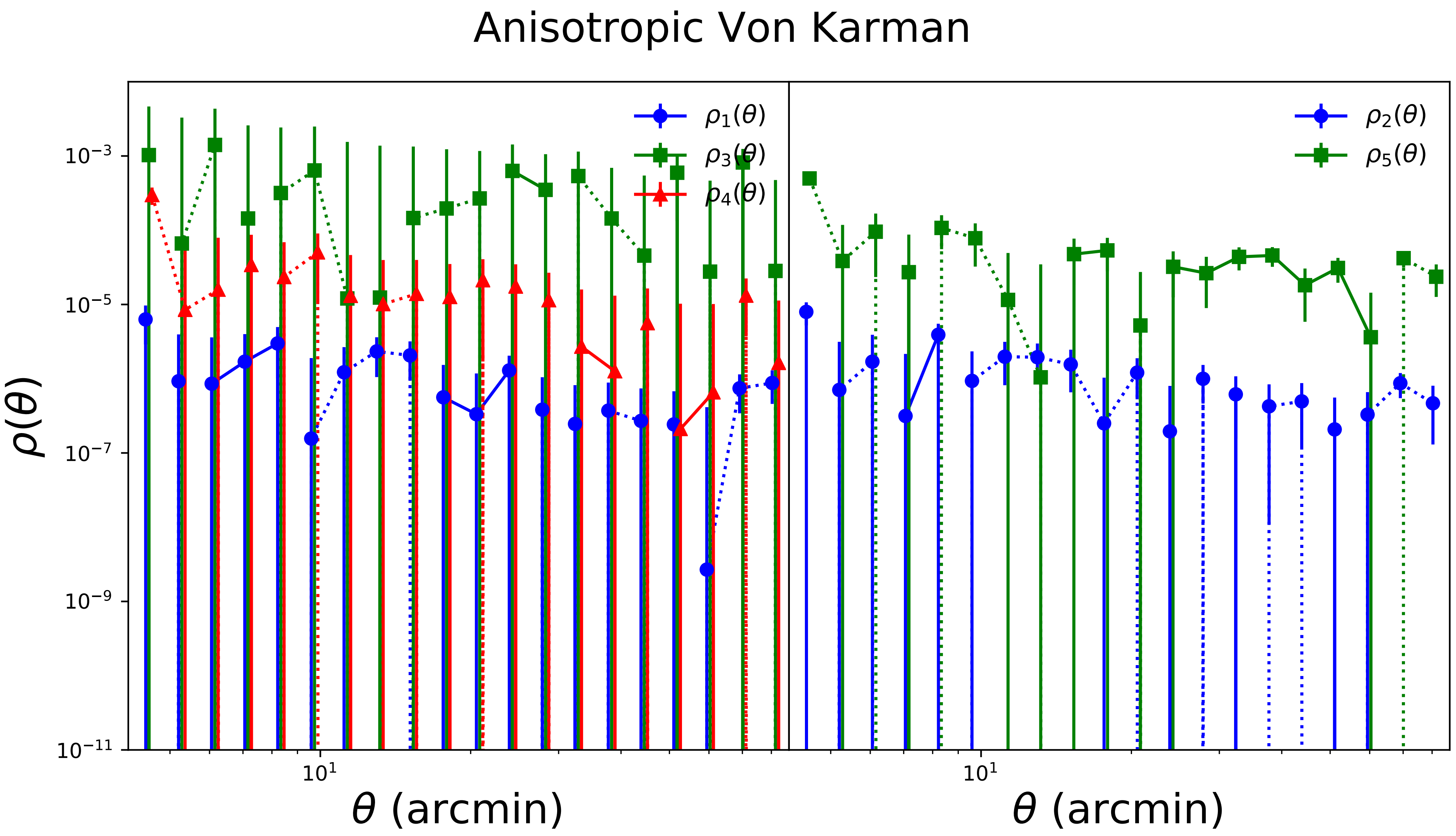


- Gaussian correlation function





Isotropic Von Karman
$\rho(\theta)$
10^{-3}
10^{-5}
10^{-7}
10^{-9}
10^{-11}
10^{1}
θ (arcmin)
$\rho_1(\theta)$
$\rho_3(\theta)$
$\rho_4(\theta)$
$\rho_2(\theta)$
$\rho_5(\theta)$



Anisotropic Von Karman
$\rho(\theta)$
$\rho_1(\theta)$
$\rho_3(\theta)$
$\rho_4(\theta)$
$\rho_2(\theta)$
$\rho_5(\theta)$
10^{-3}
10^{-5}
10^{-7}
10^{-9}
10^{-11}
10^{1}
θ (arcmin)
θ (arcmin)

- What is next:
  - Start to run on ~6000 images of DES Y3 in order to look the overall improvement.

The Optical and Atmospheric PSF model

PSF profile $\sim$ Optical part of the PSF $\otimes$ Atmospheric part of the PSF

The Optical and Atmospheric PSF model

PSF profile $\sim$ Optical part of the PSF as a Fraunhofer Diffraction $\otimes$ Atmospheric part of the PSF

$$\overbrace{I(u,v)}^{\text{PSF profile}} \sim \overbrace{\left| F\left\{ \underbrace{P(\rho,\theta)}_{\text{Pupil function}} e^{2\pi i \underbrace{W(\rho,\theta)}_{\text{Wavefront}}/\lambda} \right\} \right|}^{\text{Optical part of the PSF}}$$

Pupil function

Wavefront

The Optical and Atmospheric PSF model

PSF profile ~ Optical part of the PSF as a Fraunhofer Diffraction $\otimes$ Atmospheric part of the PSF

$$\overbrace{I(u,v)}^{\text{PSF profile}} \sim \overbrace{\left| F\left\{ P(\rho,\theta) e^{2\pi i W(\rho,\theta)/\lambda} \right\} \right|}^{\text{Optical part of the PSF as a Fraunhofer Diffraction}}$$

Pupil function: $P(\rho,\theta)$

Wavefront: $W(\rho,\theta)$

Wavefront decomposed as a double Zernike polynomial that depends on the focal plane coordinate

$$W(\rho,\theta) = \sum_i \left[a_{i,reference}(u,v) + a_{i,corr}(u,v) \right] Z_i(\rho,\theta)$$

$$a_{i,corr}(u,v) = \sum_j b_{i,j}(u,v) Z_j(\rho,\theta)$$

The Optical and Atmospheric PSF model

PSF profile ~ Optical part of the PSF as a Fraunhofer Diffraction $\otimes$ Atmospheric part of the PSF as a Kolmogorov profile

$$\underbrace{I(u,v)}_{\text{PSF profile}} \sim \underbrace{\left| F\left\{ P(\rho,\theta) e^{2\pi i W(\rho,\theta)/\lambda} \right\} \right|}_{\text{Optical part of the PSF}} \otimes \underbrace{K(\alpha(u,v), g_1(u,v), g_2(u,v))}_{\text{Atmospheric part of the PSF}}$$

Pupil function: $P(\rho,\theta)$

Wavefront: $W(\rho,\theta)$

Wavefront decomposed as a double Zernike polynomial that depends on the focal plane coordinate

$$W(\rho,\theta) = \sum_i \left[a_{i,reference}(u,v) + a_{i,corr}(u,v) \right] Z_i(\rho,\theta)$$

$$a_{i,corr}(u,v) = \sum_j b_{i,j}(u,v) Z_j(\rho,\theta)$$

The Optical and Atmospheric PSF model

PSF profile $\sim$ Optical part of the PSF as a Fraunhofer Diffraction $\otimes$ Atmospheric part of the PSF as a Kolmogorov profile

$$\overbrace{I(u,v)}^{\text{PSF profile}} \sim \overbrace{\left| F\left\{ \underbrace{P(\rho,\theta)}_{\text{Pupil function}} e^{2\pi i \overbrace{W(\rho,\theta)}^{\text{Wavefront}}/\lambda} \right\} \right|}^{\text{Optical part}} \otimes \overbrace{K(\alpha(u,v), g_1(u,v), g_2(u,v))}^{\text{Atmospheric part}}$$

Pupil function

Wavefront

Second moment of
the Kolmogorov profile
(size, ellipticity)

Wavefront decomposed as a double Zernike polynomial
that depends on the focal plane coordinate

$$W(\rho,\theta) = \sum_i \left[a_{i,reference}(u,v) + a_{i,corr}(u,v) \right] Z_i(\rho,\theta)$$

$$a_{i,corr}(u,v) = \sum_j b_{i,j}(u,v) Z_j(\rho,\theta)$$

The Optical and Atmospheric PSF model

PSF profile ~ Optical part of the PSF as a Fraunhofer Diffraction $\otimes$ Atmospheric part of the PSF as a Kolmogorov profile

$$I(u,v) \sim \left| F\left\{ P(\rho,\theta) e^{2\pi i W(\rho,\theta)/\lambda} \right\} \right| \otimes K(\alpha(u,v), g_1(u,v), g_2(u,v))$$

Pupil function

Wavefront

Second moment of the Kolmogorov profile (size, ellipticity)

Wavefront decomposed as a double Zernike polynomial that depends on the focal plane coordinate

$$W(\rho,\theta) = \sum_i \left[a_{i,reference}(u,v) + a_{i,corr}(u,v) \right] Z_i(\rho,\theta)$$

$$a_{i,corr}(u,v) = \sum_j b_{i,j}(u,v) Z_j(\rho,\theta)$$

Kolmogorov parameters modeled as a Gaussian Process drive by a Von-Karman correlation function

$$\alpha(u,v) \sim N(\alpha_0(u,v), \xi)$$

$$g_1(u,v) \sim N(g_{1,0}(u,v), \xi)$$

$$g_2(u,v) \sim N(g_{2,0}(u,v), \xi)$$

The Optical and Atmospheric PSF model

PSF profile $\sim$ Optical part of the PSF as a Fraunhofer Diffraction $\otimes$ Atmospheric part of the PSF as a Kolmogorov profile

$$\overbrace{I(u,v)}^{\text{PSF profile}} \sim \overbrace{\left| F\left\{ \underbrace{P(\rho,\theta)}_{\text{Pupil function}} e^{2\pi i \overbrace{W(\rho,\theta)}^{\text{Wavefront}}/\lambda} \right\} \right|}^{\text{Optical part}} \otimes \overbrace{K(\alpha(u,v), g_1(u,v), g_2(u,v))}^{\text{Atmospheric part}}$$

Pupil function

Wavefront

Second moment of
the Kolmogorov profile
(size, ellipticity)

Wavefront decomposed as a double Zernike polynomial that depends on the focal plane coordinate

$$W(\rho,\theta) = \sum_i \left[a_{i,reference}(u,v) + a_{i,corr}(u,v) \right] Z_i(\rho,\theta)$$

$$a_{i,corr}(u,v) = \sum_j b_{i,j}(u,v) Z_j(\rho,\theta)$$

Kolmogorov parameters modeled as a Gaussian Process drive by a Von-Karman correlation function

$$\alpha(u,v) \sim N(\alpha_0(u,v), \xi)$$

$$g_1(u,v) \sim N(g_{1,0}(u,v), \xi)$$

$$g_2(u,v) \sim N(g_{2,0}(u,v), \xi)$$