Cosmic shear covariance matrix comparison

Tassia Ferreira

September 16, 2021

PPGCosmo
Federal University of Espírito Santo

Motivation

Covariance matrices are among the most difficult pieces of end-to-end cosmological analyses.

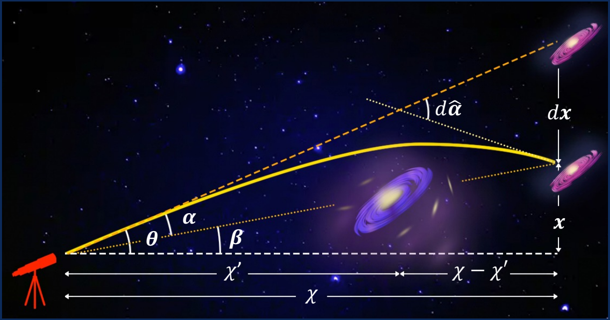
As data vectors increase, the number of elements in the covariance matrix grows quadratically.

Compression schemes provide a simpler method of analysing the covariance matrices.

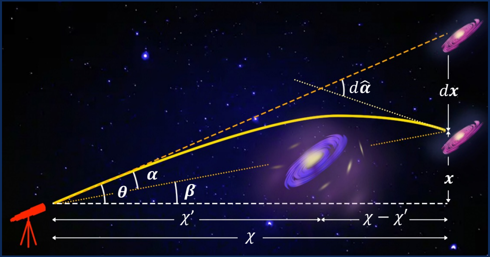
Cosmic Shear

Cosmic Shear: Magnification and distortion

- Characterised by the distortion of images by large scale structures;
- used to probe the dark matter distribution of the universe.



Cosmic Shear: Magnification and distortion



$$\boldsymbol{\alpha} = \boldsymbol{\nabla}_\theta \psi \qquad d\hat{\boldsymbol{\alpha}} = \frac{2}{c^2} \boldsymbol{\nabla}_\perp \Phi(\boldsymbol{x}, \chi) d\chi' \ .$$

Magnification matrix:

$$A_{ij} = \frac{\partial \beta_i}{\partial \theta_j} \ .$$

Cosmic Shear: Magnification and distortion

Magnification matrix:

$$\mathbf{A} = \begin{pmatrix} -\frac{1}{2}\left(\psi_{11} - \psi_{22}\right) & -\psi_{12} \\ -\psi_{12} & \frac{1}{2}\left(\psi_{11} - \psi_{22}\right) \end{pmatrix} + \left[1 - \frac{1}{2}\left(\psi_{11} + \psi_{22}\right)\right]\delta_{ij} \ ,$$

where

$$\psi_{ij} = \frac{\partial^2 \psi}{\partial\theta_i \partial\theta_j} \ .$$

Cosmic Shear: Magnification and distortion

Magnification matrix:

$$\mathbf{A} = \begin{pmatrix} -\frac{1}{2}\left(\psi_{11} - \psi_{22}\right) & -\psi_{12} \\ -\psi_{12} & \frac{1}{2}\left(\psi_{11} - \psi_{22}\right) \end{pmatrix} + \left[1 - \frac{1}{2}\left(\psi_{11} + \psi_{22}\right)\right] \delta_{ij}$$

$$= \gamma \begin{pmatrix} \cos 2\varphi & \sin 2\varphi \\ \sin 2\varphi & -\cos 2\varphi \end{pmatrix} + \kappa\, \delta_{ij} \; .$$

Cosmic Shear: Magnification and distortion

$$\mathbf{A} = \begin{pmatrix} 1-\kappa-\gamma_t & -\gamma_\times \\ -\gamma_\times & 1-\kappa+\gamma_t \end{pmatrix} ,$$

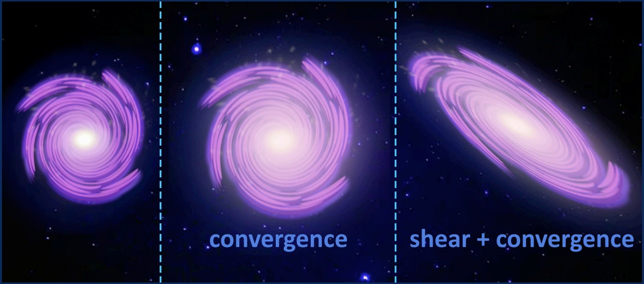
where,

$$\gamma = \gamma_t + i\gamma_\times$$
$$= |\gamma| \, e^{2i\varphi} .$$

κ : convergence,
γ : shear.

Cosmic Shear: Magnification and distortion

$$\mathbf{A} = \begin{pmatrix} 1-\kappa-\gamma_t & -\gamma_\times \\ -\gamma_\times & 1-\kappa+\gamma_t \end{pmatrix} .$$

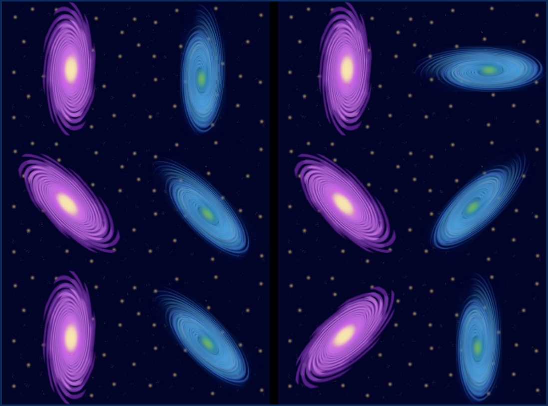


Cosmic Shear: Shear correlation

$\langle \gamma_t \gamma_t \rangle$

$\langle \gamma_\times \gamma_\times \rangle$

$\langle \gamma_t \gamma_\times \rangle \;, \langle \gamma_\times \gamma_t \rangle$



Cosmic Shear: Two-point correlation function

$$\xi_{\pm} = \langle \gamma_t \gamma_t' \rangle \pm \langle \gamma_\times \gamma_\times' \rangle \ ,$$

$$= \int \frac{\ell \, d\ell}{2\pi} P_\kappa(\ell) J_{0/4}(\ell\phi) \ ,$$

where we can obtain information about the matter distribution via

$$P_\kappa(\ell) = \int d\chi \frac{q^2(\chi)}{f_k^2(\chi)} P_\delta \left(\frac{\ell}{f_k(\chi)} \right) \ ,$$

which contains terms relating to $\Omega_m, H_0, a(t)$ and δ.

Data and covariance matrices

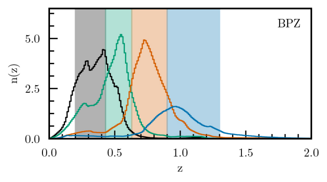
Data set: Dark Energy Survey Year 1

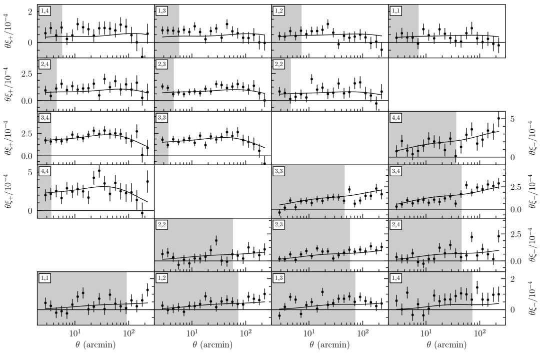
→ 227 data points of cosmic shear taken from DESY1;

→ 4 tomographic redshift bins with $0.20 < z < 1.30$;

→ 20 angular bins with $2.5 < \theta < 250$ arcmin;

→ 16 parameters.





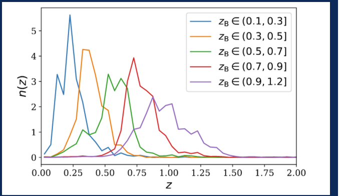
Data set: Kilo-Degree Survey 1000

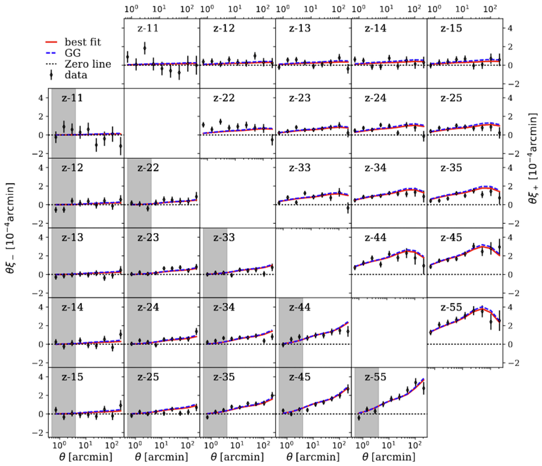
→ 235 data points of cosmic shear taken from KiDS-1000;

→ 5 tomographic redshift bins with 0.10 < z < 1.20;

→ 9 angular bins with 0.5 < θ < 500 arcmin;

→ 13 parameters.





Covariance Matrices

DES Covariance Matrix (DCM)

→ it is the same used in the DESY1 analysis;

→ it was generated by `CosmoLike`;

→ 227×227.

Gaussian Covariance Matrix (GCM)

→ contains only the Gaussian part;

→ it was generated using the same code for the KiDS-1000 survey;

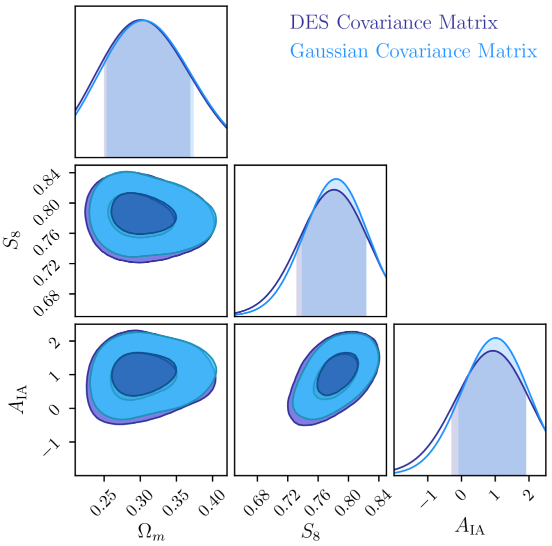
→ 227×227.

KiDS Covariance Matrix (KCM)

→ it is the same used in the KiDS-1000 analysis;

→ 235×235.

Covariance Matrices: Original constraints

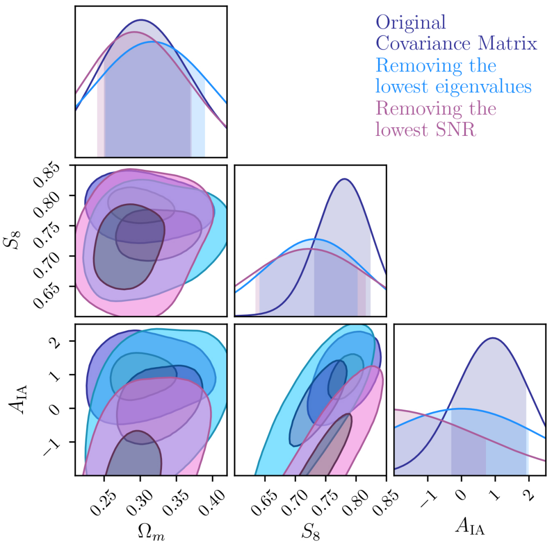


Compression and Transformation Methods

Compression Scheme: Eigenvalues and Signal-to-noise ratio

(i) Identify the 200 lowest eigenvalues;
(ii) set them to ~ 0;
(iii) run a full cosmological analysis;
(iv) repeat for the 200 modes with lowest signal-to-noise ratio.

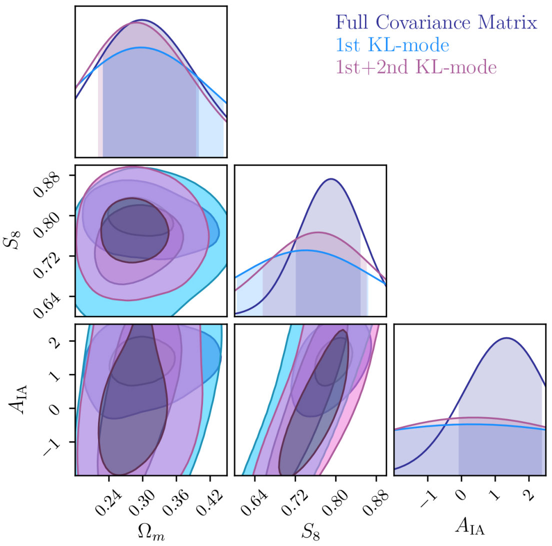
Compression Scheme: Eigenvalues and Signal-to-noise ratio



Compression Scheme: Karhunen-Loéve decompositon

- Based on the Karhuen-Loéve (KL) decomposition for the shear power spectrum;
- finds the eigenmode with highest signal-to-noise contribution;
- transforms the two-point function of this eigenmode into real space.

Compression Scheme: Karhunen-Loéve decompositon



Compression Scheme: MOPED

Massively Optimised Parameter Estimation and Data compression

- compresses the data vector down to the same dimensionality as the number of free parameters;
- the weighing vector is chosen in such a way as to maximise the Fisher matrix.

Compression Scheme: MOPED

Massively Optimised Parameter Estimation and Data compression

- compresses the data vector down to the same dimensionality as the number of free parameters;
- the weighing vector is chosen in such a way as to maximise the Fisher matrix.

Weighing vector

$$\mathbf{b}_\alpha = \boldsymbol{\mu}_{,\alpha}\mathbf{C}^{-1}$$

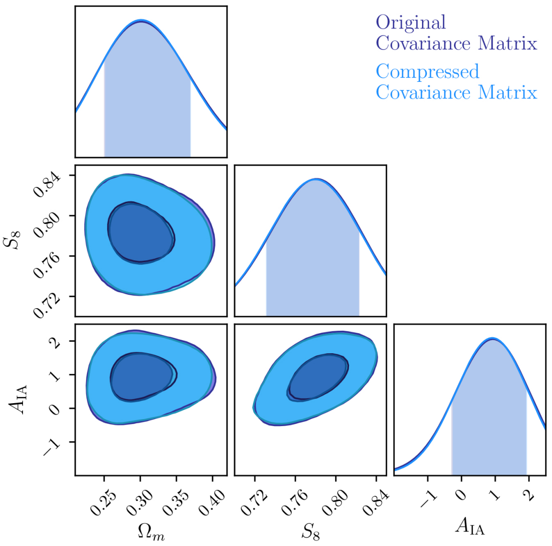
Compressed data set

$$\mathbf{y}_\alpha = \mathbf{b}_\alpha \mathbf{x}$$

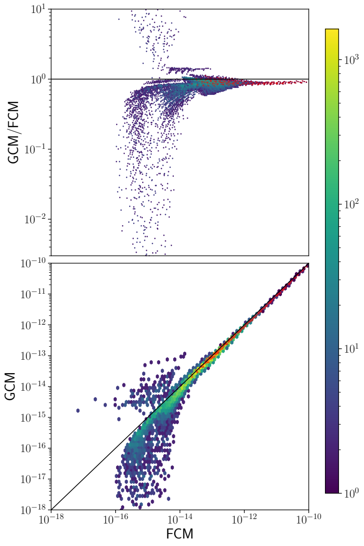
Compressed covariance matrix

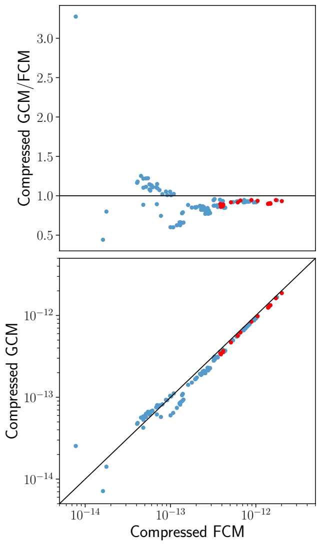
$$\mathbf{C}_{\alpha\beta} = \mathbf{b}_\alpha\ \mathbf{C}\ \mathbf{b}_\alpha^t$$

Compression Scheme: Eigenvalues and Signal-to-noise ratio



Compression Scheme: Covariance comparison





Invertible Transformation

MOPED's compression vector **b** takes an $N \times N$ covariance matrix and shrinks it down to $n \times n$, where n is the number of free parameters.

We want to ensure that the number of relevant elements remains the same as in the MOPED compressed covariance matrix.

Invertible Transformation

The new transformation is then,

$$\mathbf{B} = (\mathbf{b} \quad U) \ ,$$

where U has dimension $(N - n) \times N$. We want to find U such that

$$\mathbf{C}^{\text{trans}} = \mathbf{B}^t \mathbf{C} \mathbf{B} = \begin{pmatrix} \mathbf{b}^t \mathbf{C} \mathbf{b} & 0 \\ 0 & U \mathbf{C} U^t \end{pmatrix} ,$$

which implies

$$\mathbf{b}^t \mathbf{C} U = 0 \ .$$

Invertible Transformation

The new transformation is then,

$$\mathbf{B} = (\mathbf{b} \quad U) \ ,$$

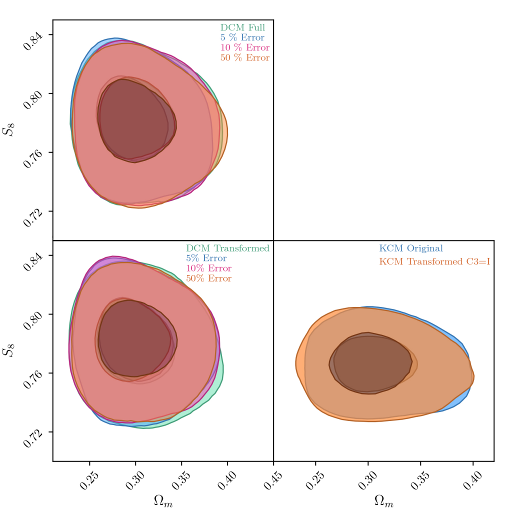
where U has dimension $(N-n) \times N$. We want to find U such that

$$\mathbf{C}^{\text{trans}} = \mathbf{B}^t \mathbf{C} \mathbf{B} = \boxed{\begin{pmatrix} C1 & 0 \\ 0 & C3 \end{pmatrix}} \ ,$$

which implies

$$\mathbf{b}^t \mathbf{C} U = 0 \ .$$

Invertible Transformation: Example



Covariance Matrix Comparison

Covariance Matrix Comparison

The comparison done with $C1$, which is the MOPED compressed covariance matrix.

Consider two compressed covariance matrices, C_{base} and C_{test}. The analysis is then divided into two parts:
$\rightarrow$ the diagonal elements, n–dimensional vector $\mathcal{D}$;
$\rightarrow$ the independent elements of the correlation matrix, the $n(n-1)/2$–dimensional vector $\mathcal{C}$.

Covariance Matrix Comparison: The algorithm

(i) create a mock sample $\{\mathcal{D}_{\delta,i}\}$, of size m, by perturbing, with a given error percentage δ, the vector $\mathcal{D}_{\text{base}}$ (or $\mathcal{C}_{\text{base}}$);

For the diagonal part, the mocks were generated by drawing $\mathcal{E}_{\mathcal{D}}$ from a multivariate Gaussian distribution $\mathcal{G}[0_n, \delta^2 I_n]$, such that,

$$\mathcal{D}_{\delta,i} = (1 + \mathcal{E}_{\mathcal{D}})\mathcal{D}_{\text{base}} \ .$$

Covariance Matrix Comparison: The algorithm

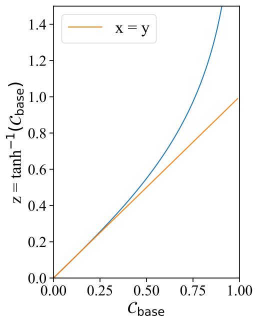
For the elements of the correlation matrix, on the other hand, we used the hyperbolic tangent function and corrected it for the Jacobian,

$$z_i = \tanh^{-1}\left(\mathcal{C}_{\text{base}}\right) \ ,$$

$$\delta z_i = \mathcal{E}_{\mathcal{C}}\left[\cosh\left(z_i + \frac{1}{2}\mathcal{E}_{\mathcal{C}}\right)\right]^2 \ ,$$

where $\mathcal{E}_{\mathcal{C}}$ is drawn similarly to $\mathcal{E}_{\mathcal{D}}$. Our perturbed vector then becomes,

$$\mathcal{C}_{\delta,i} = \tanh\left(z_i + \delta z_i\right) \ .$$



Covariance Matrix Comparison: The algorithm

(i) create a mock sample $\{\mathcal{D}_{\delta,i}\}$, of size m, by perturbing, with a given error percentage δ, the vector $\mathcal{D}_{\text{base}}$ (or $\mathcal{C}_{\text{base}}$);

(ii) produce the sample covariance matrix S_δ from the generated mocks:

$$S_\delta = \frac{1}{m-1}\sum_{i=1}^{m}\left(\mathcal{D}_{\delta,i} - \overline{\mathcal{D}_\delta}\right)\left(\mathcal{D}_{\delta,i} - \overline{\mathcal{D}_\delta}\right)^t \;;$$

(iii) obtain the fiducial χ^2–distribution using:

$$\chi^2_{\delta,i} = \left(\mathcal{D}_{\delta,i} - \mathcal{D}_{\text{base}}\right) S_\delta^{-1} \left(\mathcal{D}_{\delta,i} - \mathcal{D}_{\text{base}}\right)^t ;$$

Covariance Matrix Comparison: The algorithm

(v) calculate $\chi^2_{\text{test}} = (\mathcal{D}_{\text{test}} - \mathcal{D}_{\text{base}})\ S_{\delta}^{-1}\, (\mathcal{D}_{\text{test}} - \mathcal{D}_{\text{base}})^t$;

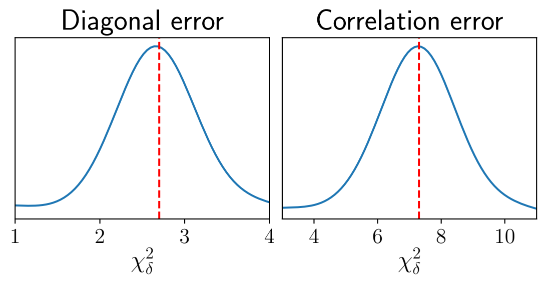
(vi) find $\delta^{\mathcal{D}}_{\text{test}}$ such that χ^2_{test} is the maximum of the fiducial χ^2–distribution above;

(vii) find $\sigma_\delta = (\delta_+ - \delta_-)/2$, where δ_+ is the value that makes χ^2_{test} fall at the right-hand border of the 68% probability interval of the χ^2–distribution, and similarly for δ_-.

For steps (v-vi), we use `lmfit`'s minimize function and define the default to use Powell's method. For the mocks, we take the sample size m to be 1000. This returns a value of δ_{test} in a few minutes, but, for robustness, we recommend setting this to > 5000.

Covariance Matrix Comparison: Results

We test our method by comparing the DCM and GCM covariance matrices.



The diagonal elements of the $C1$ block of GCM differ by 2.6 ± 0.2%, while the correlations differ by 7.8 ± 0.1%.

Covariance Matrix Comparison: Results

The diagonal elements of the $C1$ block of GCM differ by 2.6 $\pm$ 0.2%, while the correlations differ by 7.8 $\pm$ 0.1%.

For the better constrained parameters, Ω_m and S_8, we find a difference in the mean values of 1.06% and 0.74%, respectively. Their 68% contour levels are 1.38% and 2.72% larger.

On average, for the 16–dimensional parameter space, their mean values differ by 8.01% with constraints about 2.59% larger.

Conclusion

Conclusion

We see that the values found with our algorithm can be reliably related to the differences in the parameter constraints.

While our method does not replace a full cosmological analysis, it is considerably faster (taking only about 0.5 CPUh on a laptop).