Covariance Testing

Tassia Ferreira, Tianqing Zhang, Nianyi Chen, Scott Dodelson and others

(Dated: October 17, 2019)

There are a number of codes that compute covariance matrices analytically; the plan is to use these to build TJPCov. In this project, we start along the path of comparing these different codes, building up a suite of tools that can be used to compare covariance matrices. We expect these tools to be useful not only for converging on a single accurate code for computing covariance matrices but also more generally for understanding which parts of the covariance matrix carry the most information (and therefore need the most attention to get right) and which are not relevant (so for example matrices that are not positive definite may still be usable if the negative eigenmodes are not relevant).

I. INTRODUCTION

Almost all analyses of cosmological data use a likelihood function that requires a covariance matrix. If the data set has N data points, the covariance matrix is a symmetric $N \times N$ and captures the errors in the measurements, including those that are correlated from one point to another. As data sets grow, the number of elements in the covariance matrix is becoming too large to verify each individual element. For example, we will focus on cosmic shear measurements from the Dark Energy Survey (DES) [10]; in that case, the data vector has 227 elements, varying with angular separation, and different pairs of tomographic redshift bins. In this case, then, the number of independent elements of the covariance matrix is $227 \times 228/2 = 25,878$. Which of these are most important to get right? What should the tolerance be when comparing different techniques or even different codes using the same technique? How many simulations need to be run in order to obtain accurate enough covariance matrices?

Here we attempt to address these questions by considering several increasingly complex methods of identifying the elements of the covariance matrix that are most relevant. Although we will consider only the one example of DES cosmic shear measurements and projections, we think that our conclusions will be useful more generally. In §II, we describe the data set used and the pair of covariance matrices constructed in order to test different validation schemes. The next 3 sections walk through increasingly complex ways of determining the most important parts of these matrices. Before diving in, we comment that a very simple way to compare two covariance matrices is to see whether they obtain the same final parameter constraints. We will use this metric a bit throughout but want to emphasize here that our goals are a bit more ambitious: simply finding out that two covariance matrices give different results is a black-box approach. One cannot identify the source of the disagreement. The methods described here aim to go a bit deeper to try to understand where the key differences arise and which differences are most important.

Another goal of this work is to reduce the number of relevant elements of the covariance matrices to about 10% their original size. We first do this by "throwing away" 200 modes of the covariance matrix associated with the highest eigenvalues, then those with lowest signal-to-noise ratio, and, finally, we apply a lossless compression scheme capable of reducing the dimension to the number of free parameters, leaving us with a 16×16 covariance matrix.

II. DES COSMIC SHEAR: DATA AND ANALYSIS

In this section, we explain the methodology used for each of the tests employed for the analysis and comparison of the covariance matrices. Our analysis is carried out using cosmic shear statistics $\xi_\pm(\theta)$, focusing on the Year 1 results of the Dark Energy Survey [1, 10] (DESY1). The data is divided into four tomographic redshift bins spanning the interval $0.20 < z < 1.30$, which yields 10 bin-pair combinations, each one containing 20 angular bins between 2.5 and 250 arcmin. We thus begin with 200 data points for each $\xi_+(\theta)$ and $\xi_-(\theta)$, giving 400 in total. We then apply the angular cuts described in [1], which removes the scales most sensitive to baryonic effects; this leaves 167 points for $\xi_+(\theta)$ and 60 for $\xi_-(\theta)$, resulting in 227 data points corresponding to the aforementioned 227×227 covariance matrix.

Table I shows the 16 parameters varied and the priors placed on them. To perform cosmological parameter inference we use the `CosmoSIS` [12] pipeline, while employing the `MultiNest` [4] sampler to explore the parameter space, with 500 `livepoints`, `efficiency` set to 0.3, `tolerance` to 0.1 and `constant efficiency` set to False.

The covariance matrices are obtained using two different codes:

- `Cosmolike` [8] (CL) was used in the initial DESY1 analysis;
- we ran the code used to analyse the KiDS-450 survey [7] (BJ) using DESY1 parameters and tomographic bins but using only the gaussian part.

Thus, throughout, the covariance labels CL and BJ differ for several reasons: first, they are two independent codes

Parameter	Prior
Cosmological	
Ω_m	$\mathcal{U}(0.1, 0.9)$
$\log A_s$	$\mathcal{U}(3.0, 3.1)$
$H_0(\mathrm{kms}^{-1}\mathrm{Mpc}^{-1})$	$\mathcal{U}(55, 91)$
Ω_b	$\mathcal{U}(0.03, 0.07)$
$\Omega_\nu h^2$	$\mathcal{U}(0.0005, 0.01)$
n_s	$\mathcal{U}(0.87, 1.07)$
Astrophysical	
A	$\mathcal{U}(-5, 5)$
η	$\mathcal{U}(-5, 5)$
Systematic	
m^i	$\mathcal{G}(0.012, 0.023)$
Δz^1	$\mathcal{G}(-0.001, 0.016)$
Δz^2	$\mathcal{G}(-0.019, 0.013)$
Δz^3	$\mathcal{G}(0.009, 0.011)$
Δz^4	$\mathcal{G}(-0.018, 0.022)$

TABLE I. List of the priors used in the analysis for parameter constraints ($\mathcal{U}$ denotes flat in the given range and $\mathcal{G}$ is gaussian with mean equal to its first argument and dispersion equal to its second). For the cosmological parameters, we fix $w = -1.0$, $\Omega_k = 0.0$ and $\tau = 0.08$. The astrophysical parameters are associated with the intrinsic alignment, they follow the relation $A(z) = A[(1+z)/1.62]^\eta$. Lastly, for systematics we have m^i corresponding to the shear calibration and Δz^i for the source photo-z shift, with $i = 1, 4$ in both cases.

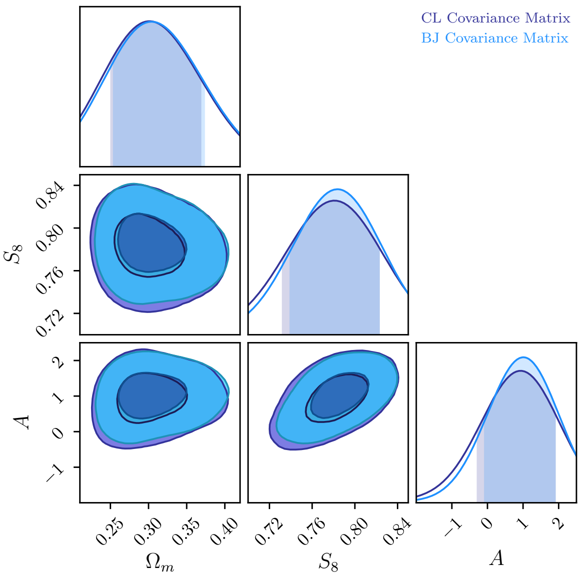

FIG. 1. Constraints on cosmological parameters Ω_m and S_8 and intrinsic alignment parameter A for two covariance matrices produced for cosmic shear for DESY1. The purple curve is for CL while the blue is for BJ. The constraints are about 22% larger for the former, indicating that the two matrices have quantifiable differences between them.

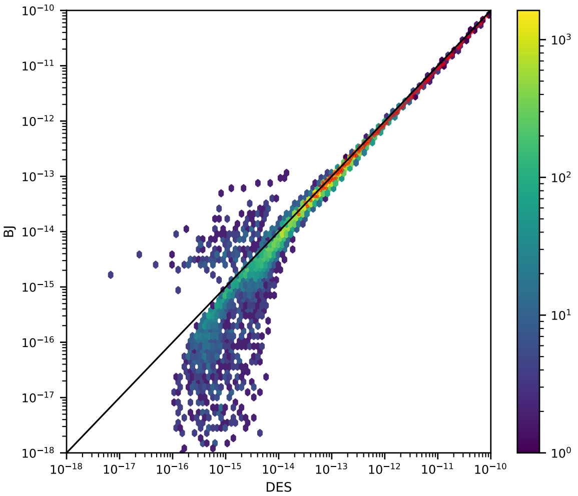

FIG. 2. Scatter plot of the positive elements of the covariance matrices BJ and CL. The reddish tint identifies the diagonal elements and the colour bar varies according to the number of elements in one hexagonal bin, where the darkest blue colour corresponds to only one element, and the brightest yellow shade to 1100.

and, second, BJ was run with very simple settings. To be clear, the BJ code does contain all the functionality in CL, but, in order to accentuate the differences, we ran with the simplest settings. The ensuing larger differences will help us assess different validation techniques.

Figure 1 shows the results for the projected cosmological constraints for DESY1. These projections use the same data vector and cuts, but the two different covariance matrices. The 2σ constraints are as follows: for CL: $\Omega_m = 0.306^{+0.073}_{-0.060}$, $A = 0.852^{+1.005}_{-1.086}$ and $S_8 = 0.784^{+0.200}_{-0.171}$; and for BJ: $\Omega_m = 0.309^{+0.073}_{-0.058}$, $A = 0.948^{+0.916}_{-0.985}$ and $S_8 = 0.787^{+0.196}_{-0.166}$. This shows that the differences we have introduced to the calculation of the two matrices are mensurable in the parameter constraints.

III. ELEMENT-BY-ELEMENT COMPARISON

If there was only a single data point, then the covariance matrix would contain only a single number and comparing two covariance matrices to try to understand why they give different constraints would be as simple as comparing these two numbers. The simplest generalization of this is to do an element-by-element comparison of the two covariance matrices. We make a scatter plot of the elements of the two matrices in Figure 2, where we can see that the elements of CL are, in general, larger than BJ's, differing by up to 4 orders of magnitude. In some ways, this is useful and reassuring, as it aligns with what we see in the parameter constraints: larger elements in the covariance matrix translates to less constraining power.

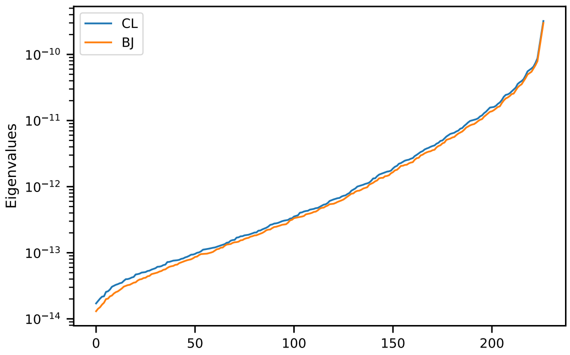

FIG. 3. A log plot showing the 227 eigenvalues of CL (blue) and BJ (orange).

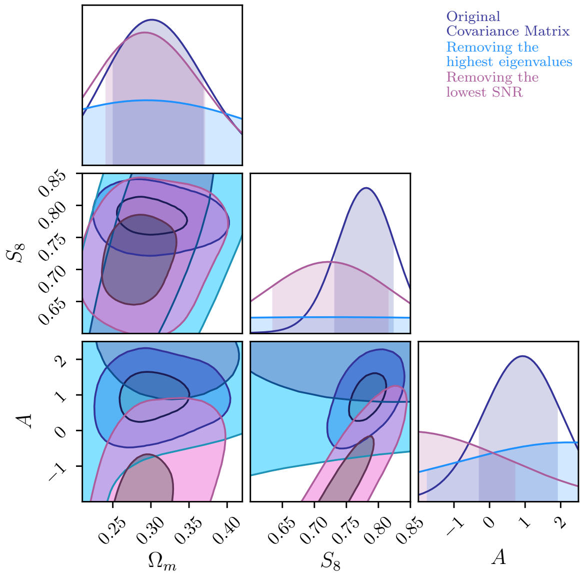

FIG. 4. Constraints on cosmological parameters Ω_m, S_8 and the intrinsic alignment parameter A for the original DESY1 covariance matrix (in purple) and for two new covariance matrices obtained by setting the 200 highest eigenvalues of the original matrix to nine orders of magnitude higher (in blue), and by replacing the 200 lowest values of the SNR to seven order of magnitude lower (in magenta).

The limitation of this brute-force method is that it remains unclear which of the differences are driving the final discrepancies in parameter constraints. This difficulty is an outgrowth of the increasing size of the data sets and hence the growing number of elements of the covariance matrix that any two codes are likely to disagree on. This element-by-element comparison, however, would prove much more useful if we could first determine the important elements. Towards that end, we start by turning to the eigenvalues.

IV. EIGENVALUES

The next simplest thing to try is to explore the eigenvalues of the covariance matrix. Each is associated with a linear combination of the data vector, or a *mode*, and it is possible that identifying the modes that have the most discrepant eigenvalues will give guidance on how to reconcile differences. We plot the eigenvalues for these matrices in Figure 3. At a first glance, both curves show reasonable agreement, with values differing only by an average of $\approx 13\%$.

The lowest eigenvalues correspond to modes with the smallest variance but since they are not normalized, it is unclear how this variance compares to the signal in the mode. Let us nonetheless explore the possibility that the modes with the lowest variance provide the most information and therefore dropping the modes with the largest eigenvalues would not affect the final result.

Our procedure consists of first diagonalizing the covariance matrix in order to calculate its eigenvalues and then replacing the large eigenvalues with numbers much larger (nine orders of magnitude higher), thus removing their effective contribution; we then rotate back to the original basis and perform a cosmological analysis with the new covariance matrix, to constrain the parameters of our model.

Figure 4 shows the results obtained after "removing" the 200 modes with the largest eigenvalues by following the method described above. The constraints are significantly broader for the three parameters shown. This is consistent with the fact that we are throwing away about 90% of the information. However, it is inconsistent with the notion that these modes are irrelevant, in fact, constraints on $S_8 \equiv \sigma_8(\Omega_m/0.3)^{0.5}$ for the original covariance matrix are $0.784^{+0.200}_{-0.171}$, whereas, for this procedure, we obtain $0.679^{+0.533}_{-0.505}$, showing an increase in almost 200%. It is then clear that a different way of ordering the modes, other than simply looking at the eigenvalues, is called for.

V. SIGNAL-TO-NOISE RATIO

Instead of looking only at the "noise" – or the eigenvalues of the covariance matrix – a better way to assess the importance of modes is to consider the signal as well. We can define the expected signal-to-noise ratio (SNR) as

$$\left(\frac{S}{N}\right)^2 = \sum_{ij} T_i C_{ij}^{-1} T_j \ , \tag{1}$$

where T_i and are the predicted theoretical signal for the i^{th} data point and C is the covariance matrix. If C were diagonal, then the eigenvectors would simply be the data points themselves, and we could estimate the SNR squared expected in each mode by simply computing T_i^2/C_{ii}. Then we could throw out the modes with the lowest SNR. Since C is not diagonal, we have to first diagonalize it and then order the values. So, we write the expected SNR squared as

$$\left(\frac{S}{N}\right)^2 = \sum_i \frac{v_i^2}{\lambda_i}, \tag{2}$$

where λ_i are the eigenvalues of the covariance matrix, which is diagonalized with the unitary matrix U, and the eigenvectors are

$$v_i \equiv U_{ij}^T T_j \ . \tag{3}$$

This makes it very clear which modes should be kept and which should be dropped. Modes v_i for which v_i^2/λ_i is very small can be discarded.

After obtaining the SNR for the covariance matrix, we proceed to set the 200 lowest values to seven orders of magnitude lower, which is equivalent to increasing the noise (or decreasing the signal) of these modes. We then obtain a new covariance matrix with the corresponding modified SNR values.

The parameter constraints for this method are shown in Figure 4, where we note that only Ω_m is well constrained (in agreement with those obtained with the original covariance matrix to within a 2σ interval). The constraining power on A and S_8, on the other hand, is very much lost, which suggests that the modes removed do indeed carry relevant information for these parameters. This is surprising since these were identified as those that had the lowest SNR.

Resolving this requires us to tweak our understanding of which modes carry information. The "signal" these modes are ordered by is the amplitude of the data points. The parameters though are sensitive to the shape as well as the amplitude. To address this, we can identify the SNR for each parameter individually. To illustrate this, we take

$$\left(\frac{\partial S/\partial p_\alpha}{N}\right)^2 = \sum_i \frac{(\partial v_i/\partial p_\alpha)^2}{\lambda_i} \tag{4}$$

where $\partial/\partial p_\alpha$ is the derivative with respect to each parameter. This produces the SNR for each parameter of interest. The importance of this procedure is illustrated in Figure 5, which shows the normalised vanilla SNR for a given mode on the x-axis and the SNR for each parameter for the same mode, for brevity, we show only for Ω_m and A. The shaded region is the one excluded in the previous analysis, but clearly there are some low SNR modes there that contain information about the parameters. This is particularly true for the intrinsic alignment parameter A, which seems to explain the poor constraints shown in Figure 4. As a result, simply cutting on raw SNR loses constraining power.

VI. SHRINKAGE

Since the simplest methods of identifying the most relevant modes are flawed, we turn to more sophisticated methods [5, 6, 9] that have been shown to reduce the number of modes significantly but still retain the relevant information. First, there is compression at the map level [2], where linear combinations of the tomographic maps are used to retain as much information as possible. Compression at the map level then significantly reduces the size of the data vector of two-point functions. For example, we will see that most of the information in the four tomographic bins used by DES can be compressed into a single linear combination of those bins. Therefore, instead of $()4\times5)/2$ two-point functions for each angular bin, we need include only one: the compression in the size of the data vector by a factor of 10 means that there are a hundred fewer elements of the covariance matrix to study.

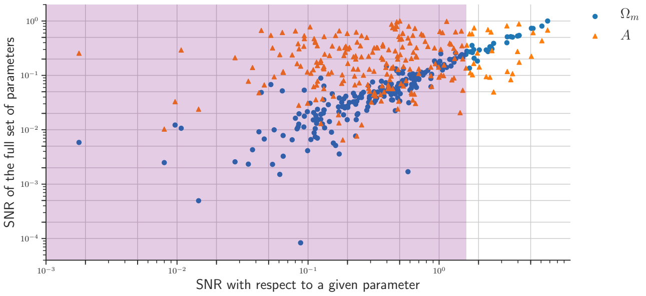

FIG. 5. Scatter plot for the relation between the signal to noise obtained with the covariance matrix for DESY1 for each parameter (x-axis) against that for the full set of parameters (y-axis). The derivatives are shown with respect to Ω_m (blue circle) and for the intrinsic alignment parameters A (orange triangle). The purple rectangle spreads until the two hundredth lowest value of SNR, which corresponds to the values that are modified for parameter constraints.

The second method directly operates on the two-point functions themselves [11], where the modes used are those that maximize the Fisher information for each of the cosmological parameters. Here the size of the data vector is just the number of parameters used to fit the data (i.e. 16), so the compression eliminates even more modes than the tomographic compression. Roughly, though, the numbers are the same: a data vector reduced by a factor of ten and the number of elements in the covariance matrix reduced by a factor of 100.

A. Tomographic Compression

This compression method is based on Karhunen-Loéve decomposition for the shear power spectrum suggested by [2] and later applied to real space two-point function in [3] for CFHTLens survey. This method generally finds the eigenmode with most of the signal-to-noise ratio contribution to the power spectrum, and then transforms the two-point function in real space based on this eigenmode.

With `CosmoSIS`, we can generate the shear power spectrum C_ℓ **[Scott: Is C_l here the same as $C^{ij}(l)$ in**

equations 7 and 8?] of convergence $a_{\ell m}$ for a fiducial cosmology. The cosmology we choose is the best-fit value of the DES Year 1 results for cosmic shear only. With the shear power spectrum $C_\ell = S_\ell + N_\ell$ and its shape noise N_ℓ, we can calculate the Karhunen-Loéve (KL) modes matrix E^p_ℓ via a general eigenvalue problem [**Scott: This isn't clear to me. What is the superscript ^{p}? Where is the tomographic bin index? Is the thing after the arrow correct? If so, please change and edit the next paragraph or two.**]

$$C_\ell E^p_\ell = \lambda^p N_\ell E^p_\ell \ \rightarrow C^{ij}(\ell)E^j_\ell = \lambda^i N^{ij}(\ell)E^j(\ell), \quad (5)$$

and the new observable $b_{\ell m} = E_p \cdot N^{-1} a_{\ell m}$. We should note that C_ℓ is the power spectrum of the convergence of the weak lensing, and E_p is the transformation of basis for the convergence. We should point out that these eigenmodes are uncorrelated, so the power spectrum of the new observable D_ℓ is a diagonal matrix, with 1+SNR of the corresponding eigenmodes on the diagonal elements,

$$D_\ell = < b_{\ell m} b^T_{\ell m} > = E^p_\ell N^{-1} C_\ell N^{-1} E^{pT}_\ell \ , \quad (6)$$

or, if we denote $E^p_\ell N^{-1}$ as R_ℓ, we can write the compression in one simple linear combination of the C_ℓ,

$$D_\ell = R^i_\ell C^{ij}_\ell R^j_\ell = U^{ij}_\ell C^{ij}_\ell \ . \quad (7)$$

The double summation weight U^{ij}_ℓ is what we use to perform tomographic compression. Since the KL-decomposed modes of shear power spectrum are uncorrelated, we can make a compression here by only taking the first one or two modes with the highest SNR. By doing so, we compress 10 tomographic combinations to 1 or 2.

We want, however, to eventually compress the two-point function data vector of DESY1. One possible way is to calculate the two-point function of the KL mode of the shear power spectrum. We can calculate the two-point function from the shear power spectrum by

$$\xi^{ij}_\pm(\theta) = \int \frac{\ell d\ell}{2\pi} J_{0/4}(\ell\theta) C^{ij}(\ell) \ . \quad (8)$$

In order to compress the two-point function based on the compression of the C_ℓ, we need to make sure that the scheme for C_ℓ is ℓ-independent, that is to say, the two-point correlation function of D_ℓ, $\tilde{\xi}_\pm(\theta)$, can be directly calculated from other two-point functions. We then have,

$$\begin{aligned}\tilde{\xi}_\pm(\theta) &= \int \frac{\ell d\ell}{2\pi} J_{0/4}(\ell\theta) D(\ell) \\ &= \int \frac{\ell d\ell}{2\pi} J_{0/4}(\ell\theta) U^{ij} C^{ij}(\ell) \\ &= U^{ij} \xi^{ij}_\pm(\theta) \ , \quad (9)\end{aligned}$$

where U^{ij}, the ℓ-independent compression weight is calculated by

$$U^{ij} = \frac{\int^{\ell_{\max}}_{\ell_{\min}} (2\ell+1) U^{ij}_\ell}{\int^{\ell_{\max}}_{\ell_{\min}} (2\ell+1)} \ . \quad (10)$$

We make a more conservative angular cut than the angular cut discussed in [10], making sure that the cut for both $\xi_\pm$ are uniform in regard to tomographic combinations. For ξ_+, we consider an angular scale from 7.195° to 250.0°. For ξ_-, the angular scale is from 90.579° to 250.0°. Therefore, for the purpose of demonstrating KL-transform, the raw data vector has a length of 190, and by shrinking 10 tomographic combinations for each angle into 1 KL-mode, the data is shrunk to 19, and so the number of elements in the covariance matrices are reduced by 99%.

With `CosmoSIS`, we calculate the shear power spectrum C_ℓ of DES Year 1 with a fiducial cosmology at the best-fit parameters, and ℓ-range $2 - 5000$. The left plot in Figure 6, shows the diagonal elements of the signal part and the noise part of C_ℓ, while the right one shows the KL-transformed eigenmode D_ℓ of C_ℓ. We can see that the first KL mode contains most of the SNR contribution to the power spectrum. If we want to recover most information, we can also include the second mode.

In Figure 7, we plot the normalized KL-eigenmode E^i_ℓ of C_ℓ and its corresponding $W^{ij}_\ell = E^i_\ell E^j_\ell$. Modes with different ℓ are plotted with different depth of the color. We can see that the KL-modes do not depend a lot on scale factor ℓ by a large portion, so we take the weighted average of the eigenmodes E^p_ℓ and its quadratic form W_ℓ over ℓ's and plot them with black lines.

For different ℓ, the KL-modes do vary by a slight amount. We also observe that tomographic bins with higher redshift gains more weight than those with low redshift. This is also shown by the weight on tomographic combination that the combination of bin 3 and bin 4 gains most of the weight to maximize signal to noise ratio. This agrees with the fact that the diagonal elements C_ℓ for low redshift is much less than those with high redshift. Is my N_ℓ actually right?

We use only the first KL-mode to compress the tomographic combination for each angle θ, i.e., 190 data points for 19 angles are compressed into 19 numbers. This compression scheme is also used to compress the CL and BJ covariance matrices, and we plot them in Figure 8.

If we now again make a one-to-one comparison of these two compressed covariance matrices. We can easily notice that the elevated clump at the bottom right for scatter of original matrices [**Scott: not sure what this refers to**], which represent elements with great difference, are gone in the compressed covariance. Instead, the two covariance matrices just have a relative constant difference because of the different model they use. This shows that the divergence between CL and BJ covariance does not affect the overall signal to noise ratio a lot. [**Scott: Shouldn't we make the point that CL is larger and that explains the broader constraints? Is this for Y1 or Y3? Do we know what the constraints look like when running with only this single KL mode?**].

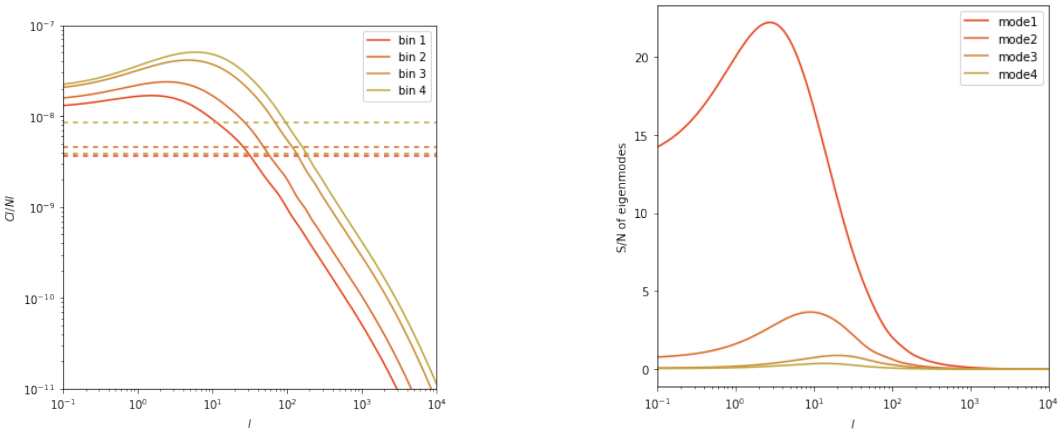

FIG. 6. **Left:** Shear power spectrum of DESY1. Solid lines are diagonal elements of the signal matrix S_ℓ, and dashed lines are the diagonal elements of noise matrix N_ℓ. [**Scott: Change the y-label to be separated by a comma so that people don't think it's the ratio. Also, increase font size and use $C^{ii}(l)$ and $N^{ii}(l)$.**] **Right:** Signal to noise ratio matrix D_ℓ of KL-modes of the power spectrum on the left. Make the plots colour blind friendly.

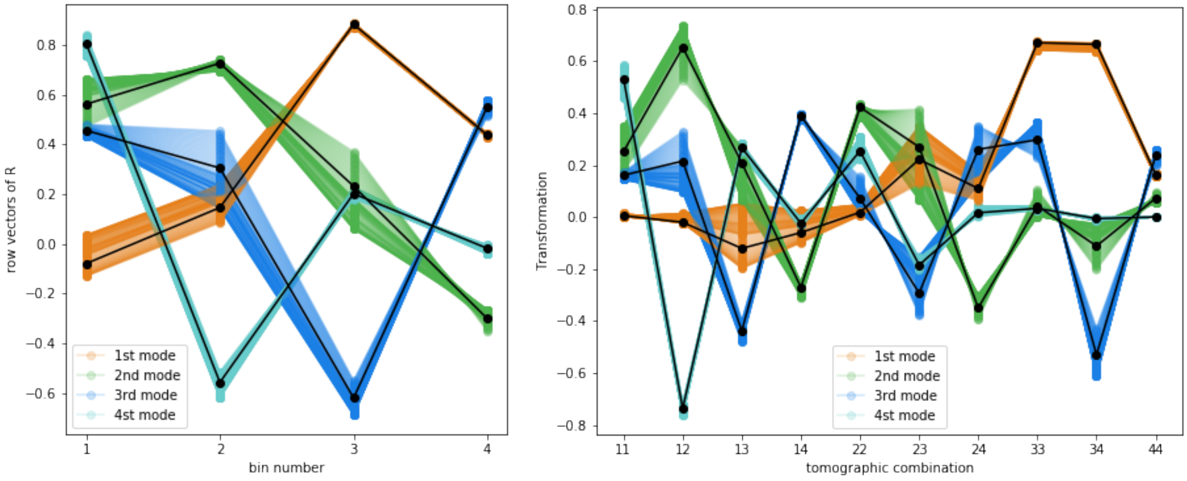

FIG. 7. **Left:** Normalized KL-eigenmodes E_ℓ^p of the shear power spectrum C_ℓ, darkness of the color is representing different ℓ. **Right:** Transformation on tomographic bin combination W_{ij} constructed by the KL-eigenmodes. Black lines are the weighted average of each mode.

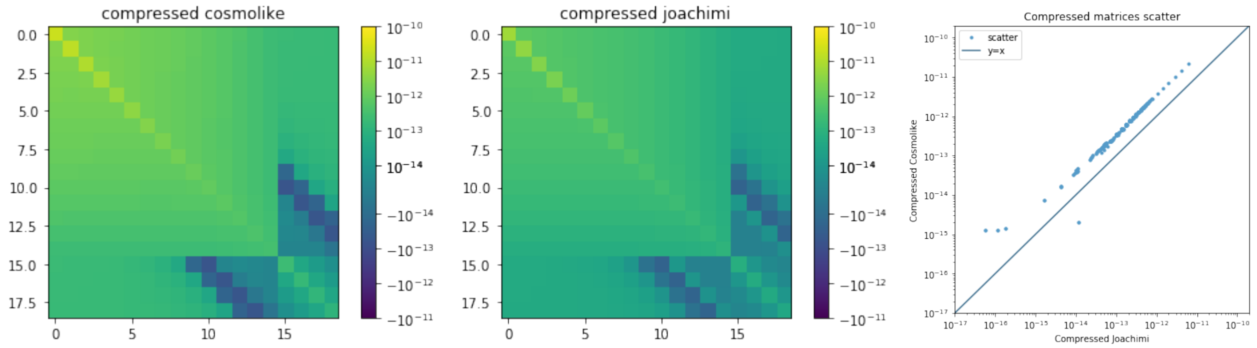

FIG. 8. **Left:** CL covariance matrix compressed by the first KL-mode. **Middle:** BJ covariance matrix compressed by the first KL-mode. **Right:** [**Scott: update plot**] One-to-One scatter of the two compressed matrices

B. Linear combinations of the data vector

The compression here takes place at the two-point level [11], with the compressed data vector containing linear combinations of the many two-point functions. In principle, this might work with only N_p two-point functions where N_p is the number of free parameters, and each mode, or linear combination, contains all the information necessary about the parameter of interest.

For each parameter p_α that is varied, one captures a single linear mode

$$y_\alpha = U_{\alpha i} D_i \ , \tag{11}$$

where D_i are the data points and the coefficients are defined as

$$U_{\alpha i} \equiv \frac{\partial T_j}{\partial p_\alpha} C^{-1}{}_{ji} \ , \tag{12}$$

with T_j being the theoretical prediction for the data point D_j. The now much smaller data set $\{y_\alpha\}$, which contains as few as N_p data points, carries with it its own covariance matrix, with which the χ^2 can be computed for each point in parameter space. Propagating through shows that this covariance matrix is related to the original C_{ij} via

$$C_{\alpha\beta} = U_{\alpha i} C_{ij} U^T_{j\beta}. \tag{13}$$

In our case, our covariance matrix is 227 × 227, while the number of parameters needed to specify the model is only 16, so $C_{\alpha\beta}$ is a 16×16 matrix. We have apparently captured from the initial set of $(227 \times 228)/2 = 25,878$ independent elements of the covariance matrix a small subset (only 136 in this case) of linear combinations of these 25k elements that really matter. If two covariance matrices give the same set of $C_{\alpha\beta}$, it should not matter whether any of the other eighty thousand elements differ from one another.

Ultimately, what matters is how well the likelihood does at extracting parameter constraints. Since most analyses assume a Gaussian likelihood, this boils down to how well the contours in parameter space agree when computing the χ^2 using two different covariance matrices.

Figure 9 compares the constraints obtained for the compressed covariance matrix and data set with results from the full one. The two curves agree extremely well for the parameters shown: Ω_m, S_8 and A. This is also true for all the other cosmological and intrinsic alignment parameters, where their mean values agree at the 2σ confidence level. While the volume of the whole constrained parameter space does increase by about 13%, the constraints for most parameters are less than 4% broader, which shows that the information loss is negligible.

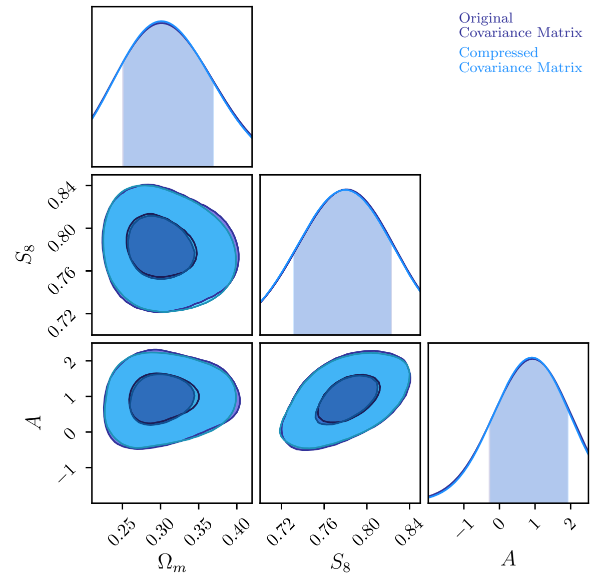

FIG. 9. Constraints on cosmological parameters Ω_m and S_8 and for the intrinsic alignment parameter A for the original DESY1 covariance matrix (in purple) and for the compressed one (in blue).

One relevant point in this analysis is at which point to take the derivative of each parameter. When we wish to compare the results of our compression scheme with those obtained with the full covariance matrix and data set, it is important to derivate each parameter at their respective mean value (obtained by performing the analysis with the full covariance matrix). The shape and variance of the posterior is not dependent on the derivative, but the best-fit value shifts according to the point where the derivative is taken.

We also apply this methodology to comparing the covariance matrices of interest, i. e. CL and BJ. In order to do this, we take two different approaches: first, we assume that $U_{\alpha,i}$ is the same for both covariance matrices and we calculate it with BJ. The second approach is that each compression scheme must use the original covariance matrix that will be compressed, so that $U_{\alpha,i}$ will be different for each covariance matrix. We find that the mean values of the parameter constraints for the two methods agree to 1σ, which shows that they are equivalent to each other. Figure 10 is obtained for the first method, which will be the one adopted from here on, it shows the correlation matrix for BJ and CL, and that of the difference between them; we find this figure important because we can clearly see the difference between the two matrices by simply looking at only $(16 \times 17)/2$ elements, as opposed to having to analyse the larger correlation matrix for the full covariance matrices. It is also crucial that the matrices used for comparison here are those obtained via the same compression scheme, so that we can be sure that their differences are indeed only related to the differences in the original matrices.

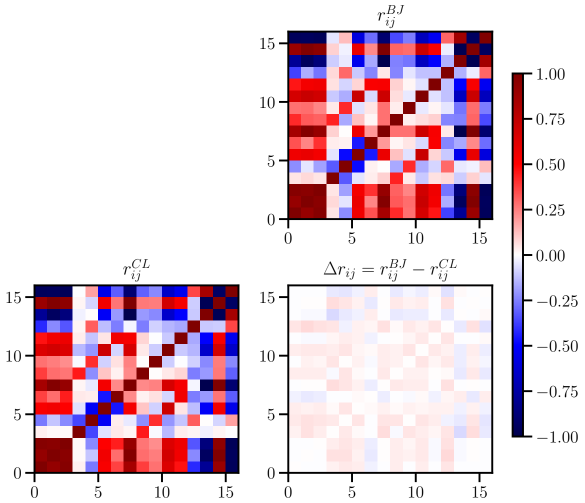

FIG. 10. The upper right and lower left plots display the correlation matrix for BJ and CL respectively, while the lower right is the difference between the two. [**Scott: Since the correlation matrices are similar, can you add a panel that compares the diagonal elements? The differences there might explain the differences in the constraints.**]

VII. TOLERANCE OF THE COMPRESSED MATRICES

Now that we have shown that we are indeed able to compress the covariance matrix into a much simpler and considerably smaller one, our next step is to analyse the amount of error the elements can tolerate while reproducing compatible parameter constraints. In the next sections we test two different ways of perturbing the covariance matrix: first we consider an error to the elements themselves, and then we follow a similar procedure to study the effects of introducing error to the eigenvalues.

One of the issues that arises when arbitrarily modifying the elements of the covariance matrix is that the new one does not necessarily remain positive definite. In this analysis, we take an extra step to ensure that this characteristic is retained.

A. Modifying the elements

To quantify the error tolerance of the elements of the covariance matrix, we introduce error in the following manner: consider that we want to test the impact of an error x%; this can either be an increase of a decrease in the original element, such that what we care about most is not whether the parameter constraints will be larger, but rather how different. For this error to be random, but centered at our desired percentage, we draw δ from a Gaussian distribution, $\mathcal{G}(0, \frac{x}{100})$ and calculate the new value to be

$$C_{\alpha\beta}^{\text{new}} = (1+\delta)C_{\alpha\beta}^{\text{old}} \ . \tag{14}$$

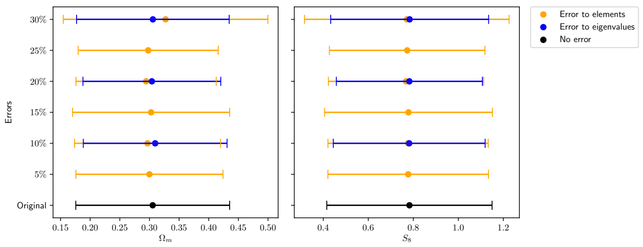

FIG. 11. The upper right and lower left plots display the correlation matrix for BJ and CL respectively, while the lower right is the difference between the two. [**Scott: Since the correlation matrices are similar, can you add a panel that compares the diagonal elements? The differences there might explain the differences in the constraints.**]

We keep the matrix symmetric by making $C_{\alpha\beta} = C_{\beta\alpha}$, and, finally, we check for positive definiteness. We show the constraints on Ω_m and S_8 in Fig. 11, in black for the original, compressed, covariance matrix and compare with those obtained using this procedure in orange. We see that errors of up to 25% translate to $< 10\%$ difference in the constraints. A 30% error, on the other hand, changes results in a difference of 33% and 24%, respectively. It is worthnoting here that we also find that S_8 is less sensitive to these errors.

B. Modifying the eigenvalues

Another way of introducing error to the covariance matrix is to perturb its eigenvalues. For a symmetric matrix, we have

$$C_{\alpha\beta} = Q\Lambda Q^{-1} \ , \tag{15}$$

where $\Lambda = \lambda I$, with λ being the eigenvalues and I the identity matrix; and Q is a square matrix whose columns are composed of the eigenvectors of $C_{\alpha\beta}$. The eigenvalues are then perturbed as described in Eq. 14, with an extra step to garantee $\delta > 1$, then $\lambda^{\text{new}} > 0$, thus keeping the matrix positive definite.

The results for this method are also plotted in Fig. 11, in blue, for Ω_m and S_8. In general, we find that we are not able to reproduce significant changes to the parameter constraints, even with 30% errors, as the resulting constraints are all within 10% of the original one. As such, it is not clear that the results obtained using this procedure is equivalent to modifying the actual elements of the covariance matrix.

VIII. DISCUSSION

By making the transformation described above, we will be able to reorganize the covariance matrices into cosmological informative blocks and uninformative blocks.

Intrinsically, we can tolerate more errors in the uninformative blocks and have a stronger requirement on the informative blocks. This will put a lot of simulation time into more valuable work.

The gold of the invertibility of the transformation is that we can first assign higher tolerance to C3 and assign lower tolerance to C1 and C2, then use the transformation matrix to recover the covariance matrices in the original basis. Then, we can compare the recovered matrix with the original one and check out how much tolerance is allowed on each element. Without invertibility, we cannot accomplish this task, which allows us to precisely quantify the tolerance on each element in the covariance matrices.

IX. CONCLUSION

Acknowledgments

The DESC acknowledges ongoing support from the Institut National de Physique Nucléaire et de Physique des Particules in France; the Science & Technology Facilities Council in the United Kingdom; and the Department of Energy, the National Science Foundation, and the LSST Corporation in the United States. DESC uses resources of the IN2P3 Computing Center (CC-IN2P3–Lyon/Villeurbanne - France) funded by the Centre National de la Recherche Scientifique; the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231; STFC DiRAC HPC Facilities, funded by UK BIS National E-infrastructure capital grants; and the UK particle physics grid, supported by the GridPP Collaboration. This work was performed in part under DOE Contract DE-AC02-76SF00515.

Need to thank CAPES as well.

Appendix A: Invertible Transformation and Tolerance Testing

In the last section, we shrink the data vector and covariance matrices and find the most cosmological-informative modes in both. However, in order to make a tolerance testing of each element in the covariance, we not only need the cosmological-informative modes but also the uninformative modes. Suppose we find the informative set of modes for cosmology, the modes that are orthogonal to them, or the complementary set of the informative one, form a set of modes that are cosmological-uninformative.

To find these modes, we start off with the compression scheme presented in Eq. 12, which will serve as the basis for a rotation matrix W_α. We then use the Gram-Schmidt decomposition to create $227 - N_p$ vectors orthonormal to U_α, thus obtaining a unitary 227×227 matrix. We then have,

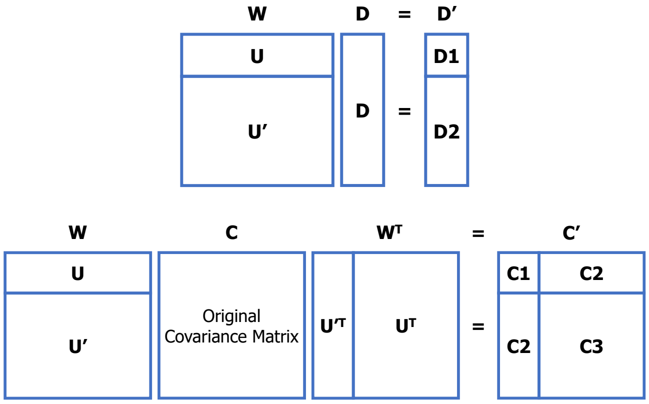

FIG. 12. Illustration of the invertible transformation, W. It's components, U and U' are the compression scheme described in Sections VI and orthogonal components obtained using the Gram-Schmidt decomposition, respectively. **Top:** Transformation of the data vector, where D1 corresponds to compressed set, and D2 are uninformative for parameter constraints. **Bottom:** Transformation of the covariance matrix. The last square on the right, C' is divided into four blocks: C1 is the most informative to cosmology, C2 are also relevant to constraints, but on a lower scale, and, finally, C3 is irrelevant to the χ^2 calculation.

$$C'_{\alpha\beta} = W_{\alpha i} C_{ij} W^T_{j\beta}. \tag{A1}$$

With the invertible transformation W, shown in Figure 12, the data vector and covariance matrix are transformed into sectional blocks. We will describe the meaning of each block here. The data vector is split into two blocks: D1 has the transformed data points that are sensitive to changes in the cosmological parameters, that is, the cosmological-sensitive data; D2 is generated by the modes that are orthogonal to D1, they will, in principle, be unaffected by changes to the cosmological parameters, which makes them uninformative for parameter constraints.

The transformed covariance matrix is split into 4 blocks, with the off-diagonal ones being the transpose of each other. Block C1 is the variance and covariance of the cosmological-sensitive data, D1, it describes how well the cosmological-sensitive data is measured, and is, therefore, the one that contributes most to the χ^2 calculation, and, consequently, to parameter constraints. Block C2 is the cross-correlation between D1 and D2, it is important for parameter constraints because it describe how D1 is affected by the uncertainty of D2. Finally, C3 relates to the uninformative data, D2; it plays a minor role to χ^2, so it affects the parameter constraints least.

The next step is then to test the tolerance of different parts of the transformed covariance matrix. We first compare the results of increasing the error of each block separately by a factor of 100 and compare the results to constraints with the unmodified covariance, then we start by introducing smaller errors to the relevant blocks,

FIG. 13. The upper right and lower left plots display the correlation matrix for BJ and CL respectively, while the lower right is the difference between the two.

C1 and C2 and analyse the corresponding increase in the parameter constraints.

Appendix B: Tolerance Testing

Given that the Block C1 contains the most relevant elements, it was expected that any changes to it would also modify the parameter constraints. It is clear in Figure 13 that this was not the case. Multiplying the elements of Blocs C1 and C3 did not alter the constraints, whereas changes made to Blocks C2 and C1+C2 did. To explain this, we need to look at what happens to χ^2 when using the transformed dataset and covariance matrix. We have,

$$v_i^\alpha \equiv U_{ij}^T \frac{\partial T_j}{\partial p_\alpha}, \tag{B1}$$

[1] Abbott, T. M. C., et al. 2018, Physical Review D, 98, 043526

[2] Alonso, D. 2018, Monthly Notices of the Royal Astronomical Society, 473, 4306

[3] Bellini, E., Alonso, D., Joudaki, S., & Waerbeke, L. V. 2019, arXiv:1903.04957

[4] Feroz, F., Hobson, M. P., & Bridges, M. 2009, Monthly Notices of the Royal Astronomical Society, 398, 1601

[5] Gualdi, D., Manera, M., Joachimi, B., & Lahav, O. 2018, Monthly Notices of the Royal Astronomical Society, 4070, 4045

[6] Joachimi, B. 2017, Monthly Notices of the Royal Astronomical Society: Letters, 466, L83

[7] Köhlinger, F., et al. 2017, Monthly Notices of the Royal Astronomical Society, 471, 4412

[8] Krause, E., & Eifler, T. 2017, Monthly Notices of the Royal Astronomical Society, 470, 2100

[9] Tegmark, M., Taylor, A. N., & Heavens, A. 1997, The Astrophysical Journal, 480, 22

[10] Troxel, M. A., et al. 2018, Phys. Rev., D98, 043528

[11] Zablocki, A., & Dodelson, S. 2016, Physical Review D, 93, 083525

[12] Zuntz, J., Paterno, M., Jennings, E., et al. 2015, Astronomy and Computing, 12, 45