Shear measurement bias

Arnau Pujol, Jerome Bobin, Florent Sureau, Axel Guinot, Martin Kilbinger

To cite this version:

Arnau Pujol, Jerome Bobin, Florent Sureau, Axel Guinot, Martin Kilbinger. Shear measurement bias: II. A fast machine-learning calibration method. Astronomy and Astrophysics - A&A, 2020, 643, pp.A158. 10.1051/0004-6361/202038658 . cea-03015651

HAL Id: cea-03015651

https://cea.hal.science/cea-03015651

Submitted on 20 Nov 2020

A&A 643, A158 (2020)
https://doi.org/10.1051/0004-6361/202038658
© A. Pujol et al. 2020

Astronomy & Astrophysics

Shear measurement bias

II. A fast machine-learning calibration method

Arnau Pujol[1,2,3,4], Jerome Bobin[1,2,5], Florent Sureau[1,2], Axel Guinot[1,2], and Martin Kilbinger[1,2,6]

[1] DEDIP/DAP, IRFU, CEA, Université Paris-Saclay, 91191 Gif-sur-Yvette, France
e-mail: arnaupv@gmail.com
[2] AIM, CEA, CNRS, Université Paris-Saclay, Université Paris Diderot, Sorbonne Paris Cité, 91191 Gif-sur-Yvette, France
[3] Institut d'Estudis Espacials de Catalunya (IEEC), 08034 Barcelona, Spain
[4] Institute of Space Sciences (ICE, CSIC), 08193 Barcelona, Spain
[5] Institute of Particle and Cosmos Physics (IPARCOS), Universidad Complutense de Madrid, 28040 Madrid, Spain
[6] Institut d'Astrophysique de Paris, UMR7095 CNRS, Université Pierre & Marie Curie, 98bis boulevard Arago, 75014 Paris, France

Received 15 June 2020 / Accepted 24 September 2020

ABSTRACT

We present a new shear calibration method based on machine learning. The method estimates the individual shear responses of the objects from the combination of several measured properties on the images using supervised learning. The supervised learning uses the true individual shear responses obtained from copies of the image simulations with different shear values. On simulated GREAT3 data, we obtain a residual bias after the calibration compatible with 0 and beyond *Euclid* requirements for a signal-to-noise ratio >20 within ~15 CPU hours of training using only ~10^5 objects. This efficient machine-learning approach can use a smaller data set because the method avoids the contribution from shape noise. The low dimensionality of the input data also leads to simple neural network architectures. We compare it to the recently described method Metacalibration, which shows similar performances. The different methods and systematics suggest that the two methods are very good complementary methods. Our method can therefore be applied without much effort to any survey such as *Euclid* or the *Vera C. Rubin* Observatory, with fewer than a million images to simulate to learn the calibration function.

Key words. gravitational lensing: weak – methods: numerical – methods: data analysis – methods: observational – methods: statistical – cosmology: observations

1. Introduction

Weak gravitational lensing by the large-scale structure has become an important tool for cosmology in recent years. Light deflection by tidal fields of the inhomogeneous matter on very large scales causes small deformations of images of high-redshift galaxies. This cosmic shear contains valuable information about the growth of structures in the Universe and can help to shed light on the nature of dark matter and dark energy. The amount of shear that is induced by weak lensing is very small, at the percent level, and should be estimated based on high-accuracy galaxy images for a reliable cosmological inference: measurement biases need to be reduced to the sub-percent level to pass the requirement of upcoming large cosmic-shear experiments, such as the ESA space mission *Euclid* (Laureijs et al. 2011), the NASA space satellite Roman Space Telescope (Akeson et al. 2019), or the ground-based *Vera C. Rubin* Observatory, previously referred to as the Large Synoptic Survey Telescope (LSST Science Collaboration 2009).

Shear is estimated by measuring galaxy shapes and averaging out their intrinsic ellipticity. This estimate is in general biased by noise, inappropriate assumptions about the galaxy light distribution, uncorrected point spread function (PSF) residuals, and detector effects such as the brighter-fatter effect or the charge transfer inefficiency (Bridle et al. 2009, 2010; Kitching et al. 2011, 2012, 2013; Refregier et al. 2012; Kacprzak et al. 2012; Melchior & Viola 2012; Taylor & Kitching 2016; Massey et al. 2007, 2013; Voigt & Bridle 2010; Bernstein 2010; Zhang & Komatsu 2011; Kacprzak et al. 2012, 2014; Mandelbaum et al. 2015; Clampitt et al. 2017). The resulting shear biases are complex functions of many parameters that describe galaxy and instrument properties. These include the galaxy size, flux, morphology, signal-to-noise ratio (S/N), intrinsic ellipticity, PSF size, anisotropy and its alignment with respect to the galaxy orientation, and many more (Zuntz et al. 2013; Fenech Conti et al. 2017; Hoekstra et al. 2015, 2017; Pujol et al. 2020)

To achieve the sub-percent shear bias that is expected in future cosmic-shear surveys, the shear estimates typically need to be calibrated using a very high number of simulated images, for instance to overcome the statistical variability induced by galaxy intrinsic shapes (Massey et al. 2013). Furthermore, these simulations need to adequately span the high-dimensional space of parameters that determines the shear bias. Otherwise, regions of parameter space that are underrepresented in the simulations compared to the observations can lead to incorrect bias correction. The selection of objects needs to closely match the real selection function to avoid selection biases.

Existing calibration methods requiring extensive image simulations select a few parameters of interest a priori, often galaxy size and S/N, for which the shear bias variation is estimated (Zuntz et al. 2018). The shear bias is computed using various methods such as fitting to the parameters (Jarvis et al. 2016; Mandelbaum et al. 2018a) or *k*-nearest neighbours (Hildebrandt et al. 2017).

Machine-learning techniques have also been employed for shear estimation and calibration. Gruen et al. (2010) trained an artificial neural network (NN) to minimise the shear bias from parameters measured in the moment-based method KSB (Kaiser et al. 1995). More recently, Tewes et al. (2019) presented an artificial neural network for supervised learning to obtain shear estimates from a few fitted parameters from images using adaptive weighted moments via regression and using image simulations with varying galaxy features as a training set.

An alternative shear calibration method that does not require image simulations and is based on the data themselves is the so-called meta-calibration (Huff & Mandelbaum 2017). This approach computes the shear response matrix by adding low shear values to deconvolved observed galaxy images. A hybrid method is a self-calibration (Fenech Conti et al. 2017), for which noise-free simulated images are created and re-measured, according to the best-fit parameters measured on the data, to reduce noise bias.

This paper extends previous work of machine learning for shear calibration. In the companion paper, Pujol et al. (2020), hereafter Paper I, we have explored the dependence of shear bias on various combinations of input and measured parameters. We demonstrated the complexity of this shear bias function and showed that it is important to account for correlations between parameters. A multi-dimensional parameter space of galaxy and PSF properties is therefore set up to learn the shear bias function using a deep-learning architecture to regress the shear bias from these parameters.

This paper is organized as follows. Section 2 presents the definition of shear bias and a review of our method for measuring shear bias that was introduced in Pujol et al. (2019; hereafter PKSB19). In Sect. 3 we introduce our new shear calibration method. Section 4 presents the simulated images and the input data used for the training, testing, and validation of our method to produce the results of this paper, which are discussed and compared to an existing method in Sect. 5. After a discussion of several points regarding the new method in Sect. 6, we conclude with a summary of the study in Sect. 7.

2. Shear bias

2.1. Shear bias definition

In the weak-lensing regime, the observed ellipticity of a galaxy $e_{\rm i}^{\rm obs}$ is an estimator of the reduced shear g_i for component $i = 1, 2$. In general, however, this estimator is biased by pixel noise, PSF residuals, inaccurate galaxy models, and other effects (see Mandelbaum 2018 for a recent review). The bias of the estimated shear, $g^{\rm obs}$, is usually expressed by the following equation:

$$\langle e_{\rm i}^{\rm obs} \rangle = g_i^{\rm obs} = c_i + (1 + m_i) g_i, \tag{1}$$

where c_i and m_i are the additive and multiplicative shear biases, respectively. For a constant shear, if the mean intrinsic ellipticity of a galaxy sample is zero, we can measure the shear from the average observed ellipticities using the above relation.

We can also define the response of the ellipticity measurement of an image to linear changes in the shear (Huff & Mandelbaum 2017),

$$R_{ij} = \frac{\partial e_{\rm i}^{\rm obs}}{\partial g_j}. \tag{2}$$

The multiplicative bias of a population can be obtained from the average shear responses, as described in Pujol et al. (2019),

$$1 + m_i = \langle R_{ii} \rangle. \tag{3}$$

In a similar way, the additive population shear bias can be obtained from the average of the individual additive biases,

$$c_i = \langle e_{\rm i}^{\rm obs} - e_{\rm i}^{\rm I} - R_{ii} g_i \rangle = \langle e_{\rm i}^{\rm obs} \rangle, \tag{4}$$

where $e_{\rm i}^{\rm I}$ is the intrinsic ellipticity, and the second equality holds if $\langle e_{\rm i}^{\rm I} \rangle = 0$ and $\langle R_{ii} g_i \rangle = 0$ (which is true if R_{ii} is not correlated with g_i and $\langle g_i \rangle = 0$).

2.2. Shear bias measurements

We used two methods to estimate shear bias, both of which we briefly describe in this section. For more details on the different methods for estimating shear bias in simulations we refer to PKSB19.

First, to test the residual shear bias after calibration, we used the common approach to measure shear bias from a linear fit to Eq. (1). We simulated each galaxy with its orthogonal pair to guarantee that the average intrinsic ellipticity is zero. This improves the precision of the shear bias estimation by a factor of ~3, as shown in PKSB19.

Second, to study the shear bias dependences, we used the method introduced in PKSB19: We measured the individual shear responses and additive biases from each galaxy image by using different sheared versions of the same image. The multiplicative and additive bias of a population was then obtained from the average of these quantities. This method has been proven to be more precise by a factor of ~12 than the linear fit with orthogonal-pair noise cancellation (PKSB19).

These individual shear responses and additive biases are also used as quantities for the supervised machine-learning algorithm of our calibration method (Sect. 3.2). We denote the biases measured from simulations as described here "true" biases $m_i^{\rm t}, c_i^{\rm t}$, indicated with the superscript "t". Our goal in this paper is to regress these biases in a high-dimensional parameter space where this function is living.

2.3. Dependences

Shear bias depends in a complex way on various properties of the observed galaxy and image properties. In Paper I we explore some of these dependences using the same simulated GREAT3 images as in this paper.

In Fig. 1 we show an example of shear bias dependences with respect to three galaxy properties. These properties are true parameters from the image simulations, therefore they do not correspond to noisy measurements. Each panel shows that the multiplicative bias m_1 depends on the Sérsic index n and the half-light radius $R_{\rm b}$. In addition, these dependences change with the galaxy flux F (the three panels show increasing ranges of flux from top to bottom). We therefore need to know all three quantities to constrain m_1, and its dependence on n, $R_{\rm b}$, and F cannot be separated. This is just one example, but it illustrates the general very complex functional form of shear bias with respect to many galaxy properties. For shear calibration of upcoming high-precision surveys, this sets very high demands on image simulations, which need to densely sample a high-dimensional parameter space of galaxy and image properties.

3. Deep-learning shear calibration

3.1. Why choose machine learning for shear calibration?

Section 2.3 and Paper I gave a glimpse of shear bias as a very complex, non-linear function acting in a high-dimensional

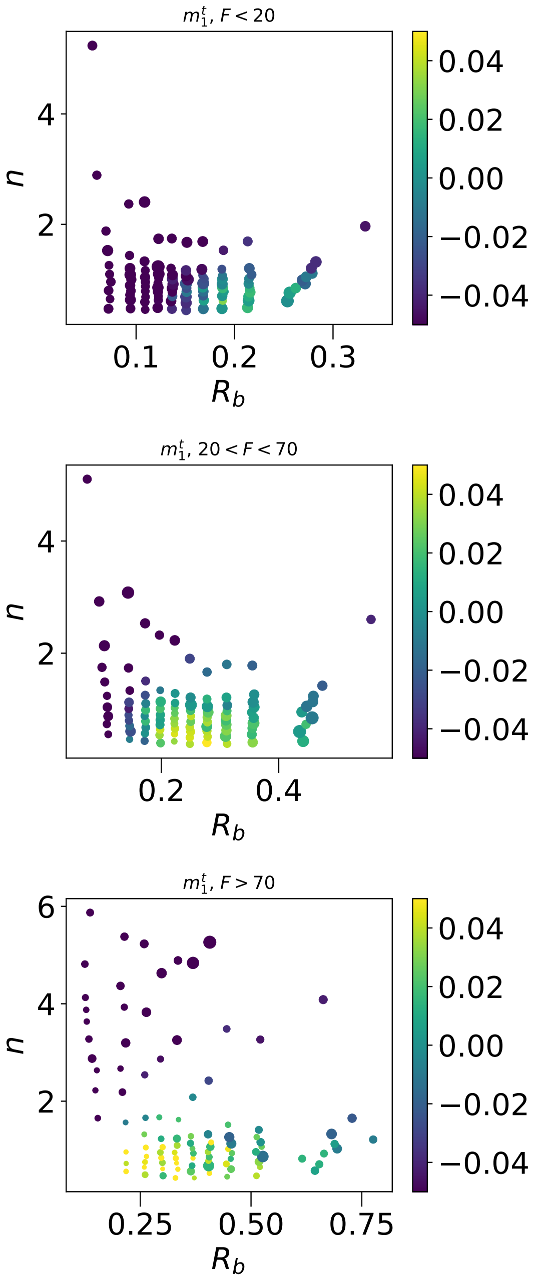

Fig. 1. Colour-coded true multiplicative shear bias m_1^t as a function of input galaxy properties. The y-axis shows the Sérsic index n, and the x-axis is the half-light radius R_b. *Top, middle*, and *bottom panels*: galaxies with different fluxes, corresponding to $F < 20$, $20 < F < 70$, and $F > 70$, respectively. Each point is to the mean over an equal number of galaxies, and the point size is inversely proportional to the error bar, such that large points are more significant.

parameter space of galaxy and image properties. For a successful shear calibration to sub-percent residual biases as is required for large upcoming surveys (e.g. Massey et al. 2013), this function needs to be modelled very accurately. This is true whether the calibration is performed galaxy by galaxy or globally by forming the mean over an entire galaxy population: shear bias measured on simulations is always marginalised over some unaccounted-for parameters, explicitly or implicitly. If shear bias depends on some of the unaccounted parameters, then the mean bias is sensitive to the distribution of these parameters over the population used, and a mismatch of this distribution between simulations and observations produces an incorrect shear bias estimate and calibration. For this reason, it is crucial to model shear bias as a function of a wide range of properties to minimise the effect of the remaining unaccounted-for parameters. With this we would still not have control of the shear bias dependence on the distribution over the remaining unaccounted-for parameters, but we would have already captured the most significant dependences. Generating a sufficiently large number of image simulations in this high-dimensional, non-separable space obviously sets enormous requirements for computation time and storage for shear calibration.

The exact form of the shear bias function is not interesting perse. In addition, it is very difficult to determine this function from first principles, based on physical considerations (Refregier et al. 2012; Taylor & Kitching 2016; Hall & Taylor 2017; Tessore & Bridle 2019). In general, for most simulation-based calibration methods, empirical functions are fitted. However, we can make a few very basic and general assumptions about this function. For example, galaxies with similar properties are expected to have a similar shear bias under a given shape measurement method. Furthermore, this function is expected to be smooth (with a possible stochasticity coming from noise). These basic properties make the shear bias function ideal to be obtained with machine-learning (ML) techniques.

The problem of estimating the shear bias from galaxy image parameters requires finely capturing the dependences described above. On the one hand, and from a mathematical point of view, the smoothness of the relationship between shear bias and parameters tells us that shear bias should belong to a smooth low-dimensional manifold. Estimating such a manifold structure then is reduced to some regression problem. On the other hand, the relationship between the measured parameters and the sought-after shear is very intricate, which impedes the use of standard regression methods, but for which ML methods are very well suited.

Machine learning can account for many parameters and model a very complex, high-dimensional function of shear bias. The dependence on unaccounted-for parameters of the marginalised shear bias is expected to be weak, and the calibration becomes less sensitive to the particular population that is simulated. If properly designed, we do not need to know the exact important properties that affect shear bias beforehand because the algorithm can learn the important combination of parameters to constrain shear bias. We still need to know the property distribution of the observed galaxies, which are noisy and might be biased. However, as we show below, the ML training set can be different from the test set to some extent, with only relatively small calibration bias; see also the discussion in Sect. 6.2.

3.2. Neural network shear correction

In this section we describe our new method based on ML that we call neural network shear correction (NNSC). We describe the concepts of the ML approach, the learning, and the calibration steps. We have made the code publicly available[1].

3.2.1. Concept

The objective of the ML here is to infer an estimate of the shear bias from a large number of galaxy image measurement parameters. To this end, a deep neural network (DNN), and more

[1] https://github.com/arnaupujol/nnsc

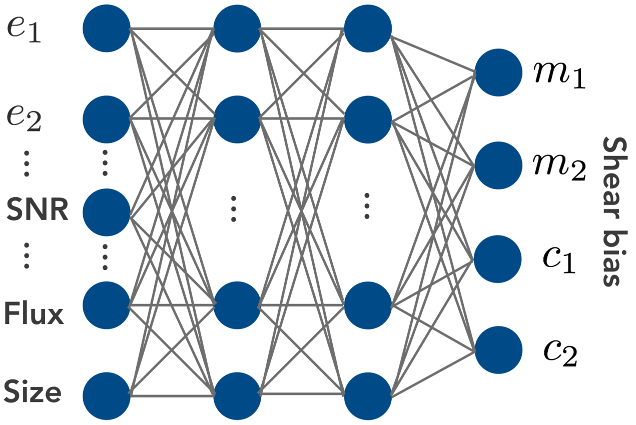

Fig. 2. Visual schema of the ML approach of the method. A set of measured image properties is used as input features for the neural network. The system is then trained to produce the shear bias parameters as output.

precisely, a multi-layer feed-forward neural network, is particularly well adapted to solve the underlying regression problem.

More precisely, a feed-forward neural network is composed of L layers, taking measured image properties as inputs and providing the shear bias as output. The resulting network aims at mapping the relationship between the measured properties of the galaxy images and the shear bias. The parameters of the network are then learnt in a supervised manner by minimising the residual between the true and the estimated shear bias. The shear bias is estimated individually for each galaxy given their measured properties.

If x_i denotes a vector containing m measured properties $x_i[j]$ for $j = 1, \ldots, m$ for a single galaxy i, then the output of the first layer $\ell = 1$ of the neural network is defined by some vector $h_j^{(1)}$ of size m_1,

$$h_i^{(1)} = \mathcal{A}\left(\boldsymbol{W}^{(1)} x_i + b^{(1)}\right), \tag{5}$$

where $\boldsymbol{W}^{(1)}$ ($b^{(1)}$) stands for the weight matrix (the bias vector) at layer $\ell = 1$, x_i is the vector of the galaxy with index i, and j is the index referring to a measured property. The term $\mathcal{A}$ is the so-called activation function, which applies entry-wise on its argument. For a layer $\ell = n$, the output vector of size m_n is defined as

$$h_i^{(n)} = \mathcal{A}\left(\boldsymbol{W}^{(n)} h_i^{(n-1)} + b^{(n)}\right). \tag{6}$$

For $\ell = 5$, the output vector $h_j^{(L)}$ stands for the estimated shear bias components. Only in this last layer is the activation function not used, so that we gain a linear function to obtain the shear bias components. A visual schema of the method is shown in Fig. 2.

3.2.2. Learning

The learning stage amounts to estimating the parameters $\left\{\boldsymbol{W}^{(\ell)}, b^{(\ell)}\right\}_{\ell=1,\ldots,L}$ by minimising the following cost function defined as some distance (the χ^2) between the shear bias components and their estimates:

$$C = \frac{1}{b_\mathrm{s}} \sum_{i=0}^{b_\mathrm{s}} \sum_{\alpha=1}^{2} (m_{i,\alpha}^\mathrm{t} - m_{i,\alpha}^\mathrm{e})^2 + (c_{i,\alpha}^\mathrm{t} - c_{i,\alpha}^\mathrm{e})^2, \tag{7}$$

where m_α^t, c_α^t and m_α^e, c_α^e are the true and estimated αth component of the shear multiplicative and additive bias, respectively, and b_s is the number of objects used in each training step (also referred to as the batch size). We use as true shear bias the values from Eqs. (2) and (4) obtained as described in Sect. 2 and presented in Pujol et al. (2019). The NNSC learns to estimate these shear biases.

We used $m = 27$ measured properties used as input for the model as described in Sect. 4.2, and details of the network architecture and the implementation of the learning stage are given in Appendix A.

3.2.3. Calibration

The NNSC method estimates the shear responses and biases of individual galaxies from the measurements of 27 properties applied to the images. The shear bias and the corresponding calibration were made for a previously chosen shape measurement. Any shape measurement algorithm can be chosen for this purpose. We used the estimation from the KSB method using the software SHAPELENS. When these estimations were completed, we applied the shear calibration over the statistics of interest, in our case, the estimated shear from Eq. (1). The bias calibration was applied as $\langle \boldsymbol{R} \rangle^{-1} \langle e^\mathrm{obs} - \langle c \rangle \rangle$, where $\boldsymbol{R}$ and c are the estimated average shear response and additive bias, respectively (see Sheldon & Huff 2017). This calibration is similar to other common approaches, and we expect similar behaviours for the post-calibration bias as discussed in Gillis & Taylor (2019).

Our method gives estimates for the individual shear bias of objects, in common with the new method METACALIBRATION. However, the two methods are very different. While NNSC relies on image simulations for a supervised ML approach, METACALIBRATION uses the data images themselves to obtain the individual shear responses. To do this, the original data images are deconvolved with an estimated PSF, and after some shear is applied, they are re-convolved with a slightly higher PSF. Because this method is very complementary with respect to NNSC and has recently been used in surveys such as the Dark Energy Survey (DES; Zuntz et al. 2018), we used it in this study for a calibration comparison of both models. For more details of METACALIBRATION, we refer to a description of the implementation in Appendix B and the original papers (Huff & Mandelbaum 2017; Sheldon & Huff 2017).

4. Data

4.1. Image simulations

We considered two sets of GALSIM simulations (Rowe et al. 2015). They correspond essentially to the control-space-constant (CSC) and real-space-constant (RSC) branch simulated in GREAT3 (Mandelbaum et al. 2014), with some modifications to ensure precise measurement of the shear response as prescribed in PKSB19.

The CSC branch contains galaxies with parametric profiles (either a single Sérsic or a de Vaucouleurs bulge profile to which an exponential disc was added) obtained from fits to *Hubble* Space Telescope (HST) data from the COSMOS survey with realistic selection criteria (Mandelbaum et al. 2014). This data set is intended to provide a realistic distribution of galaxy properties (in particular in terms of morphology, size, and S/N), which we therefore used for training and testing our calibration network.

Table 1. Measured properties for the training process of NNSC.

GFIT	SEXTRACTOR	KSB
Galaxy ellipticity $e_{1,\mathrm{GFIT}}$, $e_{2,\mathrm{GFIT}}$	Galaxy flux F_{out}	Ellipticity $e_{1,\mathrm{KSB}}$, $e_{2,\mathrm{KSB}}$
Axis ratio	Galaxy size	Ellipticity with respect to the PSF $e_{+,\mathrm{KSB}}$, $e_{\times,\mathrm{KSB}}$
Orientation angle	S/N_{obs}	Axis ratio q_{KSB}
Galaxy flux	Galaxy magnitude	Orientation angle β_{KSB}
Disc radius	PSF flux	Window function size
Bulge radius	PSF size	S/N
Disc fraction	PSF S/N	
Number of χ^2 evaluations	PSF magnitude	
Noise level	PSF FWHM	

As for GREAT3, the two million galaxy images were divided into 200 images of 10 000 galaxies, to each of which a different pre-defined shear and PSF was applied. Each galaxy was randomly oriented, and its orthogonal version was also included in the data set to allow for nulling the average intrinsic ellipticity. Out of these 200 images, we selected a first set for training and a second set for testing and comparing calibration approaches. For the training set, we followed the approach of PKSB19 described above to obtain an estimate of the true shear response that needed to be learnt. For each galaxy in the training set, five sheared versions were simulated keeping PSF and noise realisations the same. The shear g for each galaxy was chosen as $g_i = \{(g_1, g_2)_i\} = \{(0, 0), (\pm 0.02, 0), (0, \pm 0.02)\}$.

To further investigate the effect of model bias on our procedure, the network predictions were also tested for more realistic galaxies simulated as in the RSC branch of GREAT3. These galaxies are based on actual observations from the HST COSMOS survey, fully deconvolved with the HST PSF (see the procedure in Mandelbaum et al. 2013), before we applied random rotation, translation, and the prescribed shear followed by convolution with the target PSF and resampling in the target grid. In this scenario, the same procedure as for CSC was followed to obtain estimates of the shear response for these realistic galaxies, which were then compared to the network predictions based on the CSC training set.

4.2. Learning input data

The image properties that are used to estimate the shear bias can be chosen depending on the interest. The NNSC learns to estimate shear bias as a function of these properties, which means that the more properties we use, the more capable the NNSC is to learn complex dependences (if we use the appropriate training).

We used 27 measured properties as input for the NNSC. These properties correspond to the output from the GFIT (Gentile et al. 2012; Mandelbaum et al. 2015) software (properties such as ellipticities, fluxes, sizes, fitted disc fraction, and other fitting statistics), the SHAPELENS (Viola et al. 2011) KSB implementation (ellipticities, S/N, and the size of the window function) and SEXTRACTOR (Bertin & Arnouts 1996) software (properties such as flux, size, S/N, and magnitude from both the galaxy and the PSF). We refer to Paper I for the details on the algorithms and implementations and to Table 1 for the list of measured properties we used for the training. In the following we report the results obtained with the selected network as described in Appendix A associated with the superscript “fid”, referring to the fiducial implementation of the method.

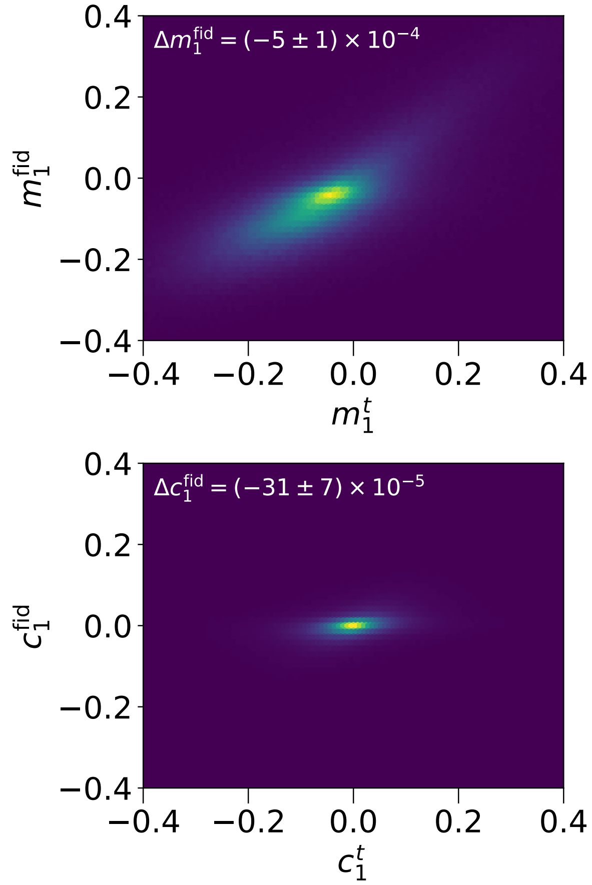

Fig. 3. Comparison between true and estimated shear bias. The multiplicative shear bias m_1 is shown in the *top panel*, and in the *bottom panel*, we show the additive bias c_1.

5. Results

5.1. Bias predictions

In Fig. 3 we show the distribution of estimated and true shear biases in the validation set of the CSC branch. We show m_1 (top panel) and c_1 (bottom panel), but similar results are found for m_2 and c_2. The estimated and true biases are correlated, although the relation is scattered. The value distribution is also narrower for the estimated than for the true biases because the estimated bias is a function of the measured parameters with no noise stochasticity. This has been learned from the stochastic true values that are affected by noise (which is the main cause of the scatter of the true-bias values), but the estimated function is not stochastic.

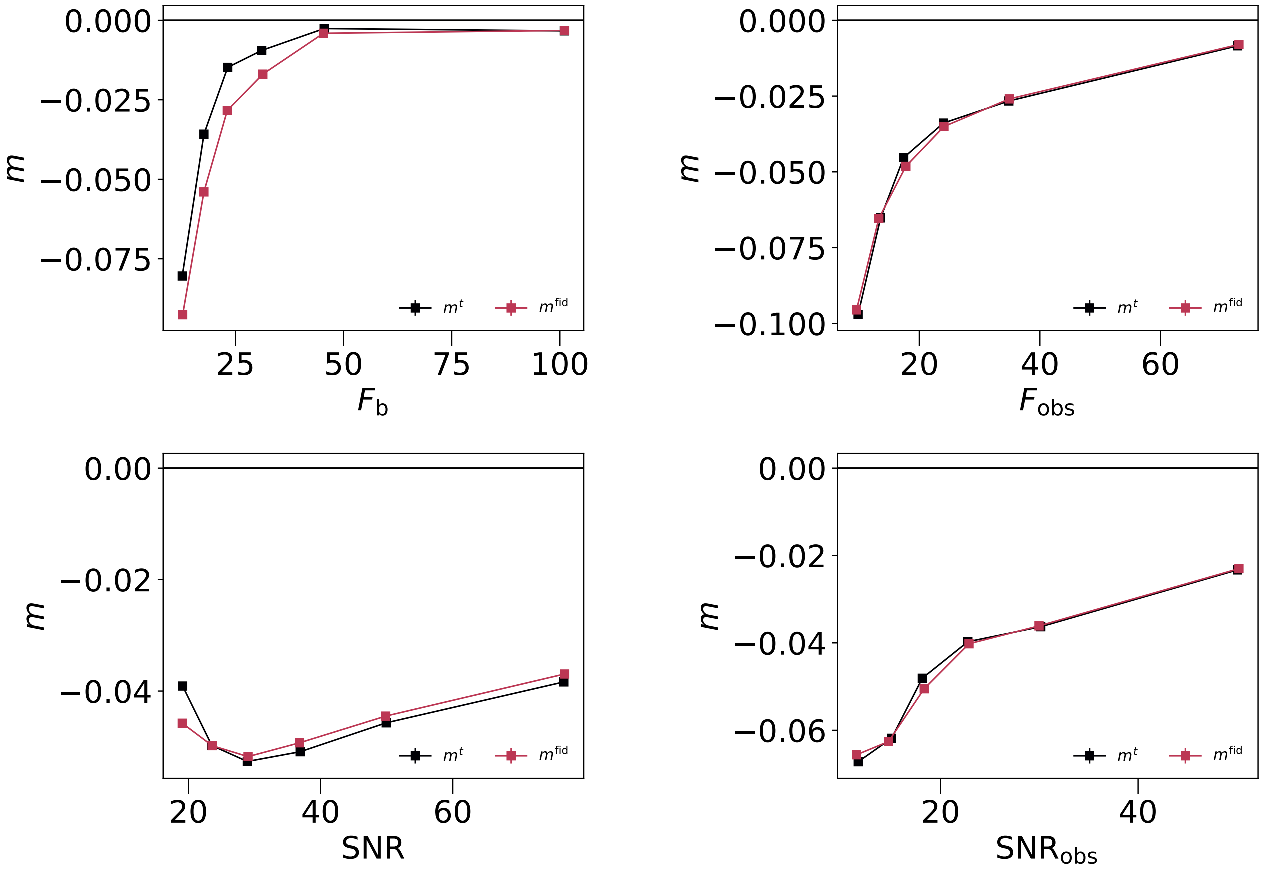

Fig. 4. Comparison of estimated vs. true multiplicative shear bias m_1 as a function of several properties. *Top panels*: m as a function of galaxy flux; *bottom panels*: m as a function of S/N. *Left panels*: input simulation properties from the galsim parameters; *right panels*: the measured parameters from SExtractor that were used as inputs in training the NN.

The errors on the estimated average biases, defined as $\Delta m_{1,2} = \langle m^{\rm fid}_{1,2} - m^{\rm t}_{1,2}\rangle$ (and analogous for $c_{1,2}$), are $\Delta m_1 = (4.9 \pm 1.1) \times 10^{-4}$ and $\Delta c_1 = (-3.1 \pm 0.7) \times 10^{-4}$ (similar values are found for the second components, with $\Delta m_2 = (0.0 \pm 1.1) \times 10^{-4}$ and $\Delta c_2 = (1.6 \pm 0.7) \times 10^{-4}$). These values are well below the *Euclid* requirements ($\Delta m < 2 \times 10^{-3}$ and $\Delta c < 5 \times 10^{-4}$), although a proper test of *Euclid* simulations should be made to quantify the performance for this mission and to quantify the effct on post-calibration bias, for which the requirements are set (Massey et al. 2013). However, this performance was obtained using only 128 000 objects with ∼15 CPU training hours.

To obtain this precision, we used a validation set of about 1 800 000 objects. According to the results from PKSB19, we expect an error on the mean bias of $\sim 3 \times 10^{-4}$. However, here we show the error on $m^{\rm t}_1 - m^{\rm est}_1$. If these two quantities are correlated (as they are), the error on their difference can be smaller, as we show.

5.2. 1D dependences

In Fig. 4 we show some examples of shear bias dependences for different cases. In black we show the true multiplicative bias obtained as described in Sect. 2.2 that was used for the supervised training. The dark red line corresponds to the performance of the NNSC estimation, referred to as $m^{\rm fid}$ because it represents the fiducial training parametrisation used for this paper (for other parametrisations, see Appendix A). In the left panels we show dependences on input simulation parameters. These are parameters that were used to generate the image simulations with GalSim, but they were not used for NNSC. The training therefore does not have access to these properties. In the right panels we show dependences on measured parameters that were used for the training. The top panels show the m dependence on galaxy flux, and the bottom panels the dependence on S/N. The excellent performance in the right panels shows that the training correctly reproduces the dependences on the measured parameters that were used as the training input. The left panels show that although the performance is not perfect, the measured parameters used in NNSC capture enough information to reproduce the dependences with good precision[2].

5.3. 2D dependencies

Figure 5 shows examples of the multiplicative shear bias 2D dependences. Here the multiplicative bias m_1 is represented in colour, the left panels show the true bias, and the right panels the estimates from NNSC. In this case, the top four panels show the dependences as a function of input simulation parameters (not accessible for NNSC), and the bottom panels show the dependences on measured parameters used for the training. NNSC clearly predicts the shear bias as a function of combinations of two input properties well. As before, the method was trained to

[2] Only single Sérsic galaxies were used in the top left panel because the true bulge flux only contains a fraction of the flux information for disc galaxies. For this reason, the average difference between estimated and true bias is different than in the rest of the panels, where the whole population was used.

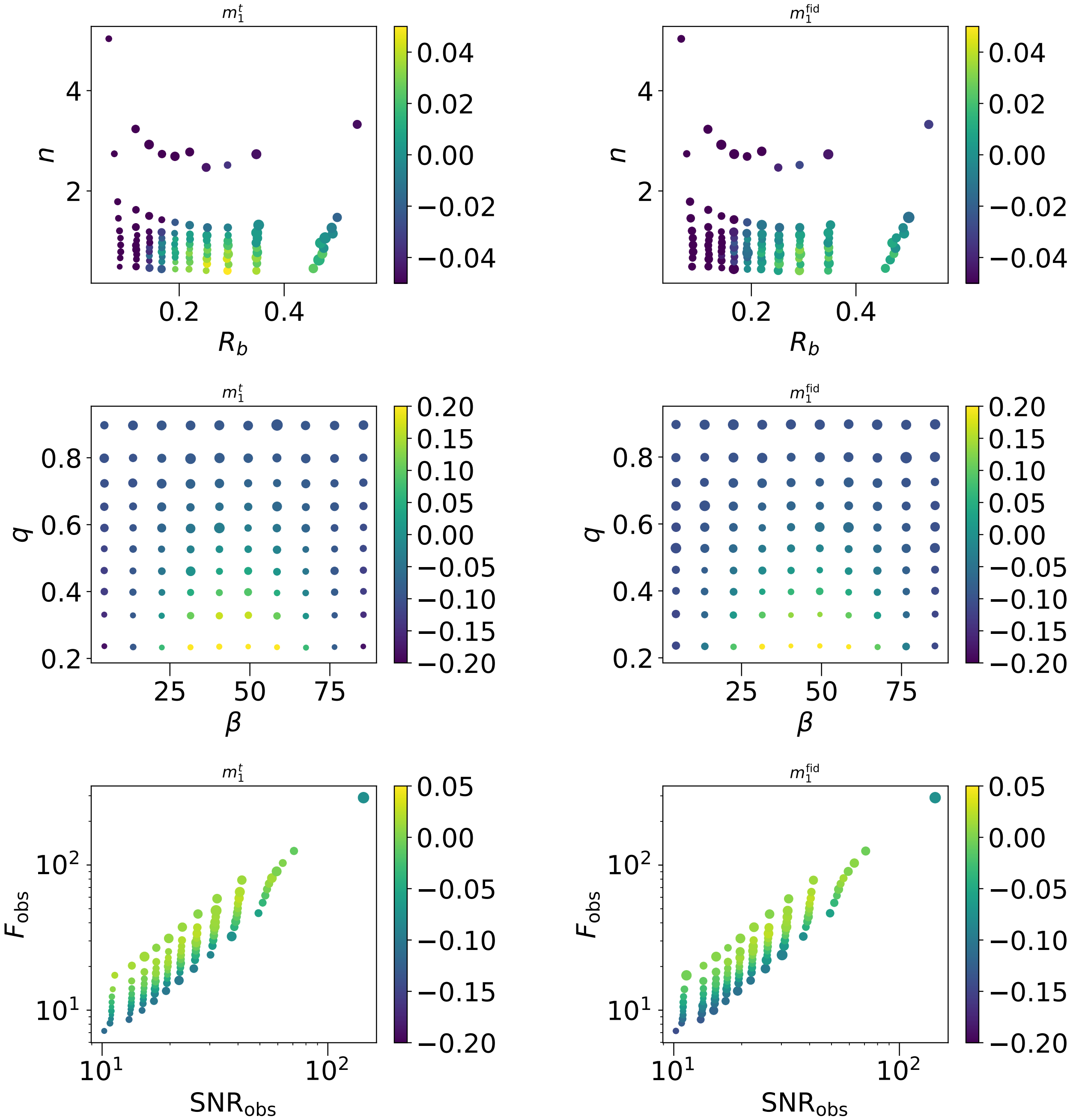

Fig. 5. Simultaneous comparison of true (*left panels*) and estimated (*right panels*) multiplicative shear bias m_1 as a function of two properties. The multiplicative shear bias is represented in colours. Each point is to the mean over an equal number of galaxies, and the point size is inversely proportional to the error bar, so that large points are more significant. *Top panels*: dependences on Sérsic index n and bulge half-light radius in the simulation. *Middle panels*: dependences on intrinsic ellipticity modulus q and orientation angle β in the simulation. *Bottom panels*: dependences on the measured flux and S/N from SExtractor.

describe shear bias as a function of the measured properties, in consistency with the good performance in the bottom panels, but the predictions on the input simulation parameters depend on how strongly these properties are constrained by the measured parameters. The method describes shear bias as a function of shape parameters very well (middle panels), but it underestimates the values for some galaxies with a very low Sérsic index n and intermediate radius because n was not estimated and no properties referring to this parameter were used for the training. The performance of the model would improve when more measured properties on the training that are correlated with n were used (e.g. a fitting parameter estimation of the galaxy profile).

5.4. Residual bias

In order to test the performance of the shear calibrations, we analysed the residual bias estimated from a linear fit of Eq. (1) after the galaxy samples were corrected for their bias. Here we include MetaCalibration as a reference for an advanced shear calibration method so that we can compare our performance with currently used approaches. With this we do not aim to show a competitive comparison of the methods, but to confirm the consistency of NNSC with respect to what can be expected for a reliable method. The two methods are intrinsically different and affected by different systematics, therefore a combination

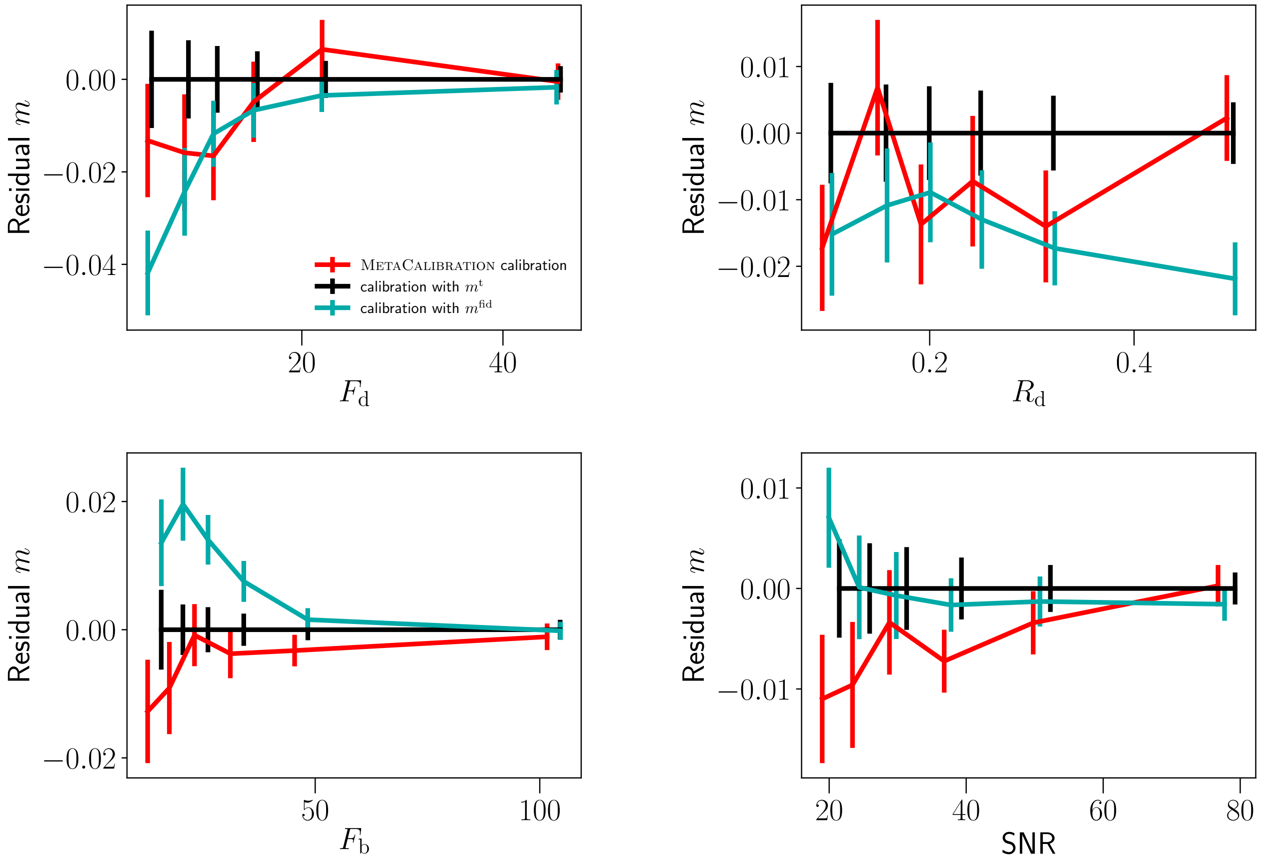

Fig. 6. Residual multiplicative shear bias as a function of input galaxy properties: disc flux (*top left*), disc half-light radius (*top right*), bulge flux (*bottom left*), and S/N (*bottom right*). Red lines show the calibration from METACALIBRATION. The black lines show the calibration using the true bias, and the cyan lines show the calibration using the bias estimate from NNSC.

of the two methods can be a very complementary and robust approach for scientific analyses. Moreover, the two methods can be differently optimised for the performance in different types of data, and here we do not pretend to show the best-case scenario for any of them. For details of the implementation done for METACALIBRATION, see Appendix B.

In Fig. 6 we show the residual multiplicative bias m as a function of several input properties found for three different approaches. In black, the calibration was made using the true shear bias obtained from the image simulations. This represents the best-case scenario where the shear bias has been perfectly estimated and gives an estimate of the statistical uncertainty of the measurement. The red and cyan lines show the residual biases from METACALIBRATION and NNSC, respectively.

Both methods show a residual bias of less than 1% for most of the cases, and the performance depends on the galaxy populations. In general, very good performances are found for both methods for bright or large galaxies. In the case of METACALIBRATION, the residual multiplicative bias increases to about 2% for small and dim galaxies, showing that the sensitivity of the method depends on the signal of the image, as expected. For NNSC, the performance depends on the explored property; it extends from negligible residual bias for any S/N to a residual bias of up to 4% for galaxies with a very dim disc. We recall that these input properties from the simulations that were not used for the training, therefore the performance of the calibration depends on the correlation between the measured properties and these input properties. Because we use an estimate of the S/N in the training, the performance of the calibration is excellent even for galaxies with a very low S/N. On the other hand, galaxies with a low disc flux are not well characterised from the measured properties we used in the training. They can be a combination of dim galaxies and galaxies with a very small disc fraction, and no measured properties aim to describe the morphology of the disc regardless of the contribution from the bulge. Because of this, the NNSC performance on those galaxies is worse. However, in real data applications we will never have access to this input information, and the galaxies will always be selected from measured properties that can be included in the training set, so that this problem will not appear on real data applications as it does here. Instead, this will produce selection effects that are discussed in Sect. 6.1.1.

5.5. Robustness with realistic images

The NNSC model has been trained with a specific set of image simulations based on the CSC branch from GREAT3. To evaluate the potential effect of applying this model to real data with no further training, we used the NNSC model that was trained with the GREAT3-CSC images but applied to calibrate GREAT3-RSC images instead. In Fig. 7 we show the estimated bias compared to the real bias obtained from sheared versions of the images as in Pujol et al. (2019) using Eqs. (3) and (4). The top panel shows the dependence of m on S/N, the bottom panels shows the error on the estimation of m. An error of up to 6% for the lowest S/N and of about 1% for galaxies with $S/N > 20$ is evident. This indicates that the model trained on analytic, simpler galaxy images does not yield perfect estimate

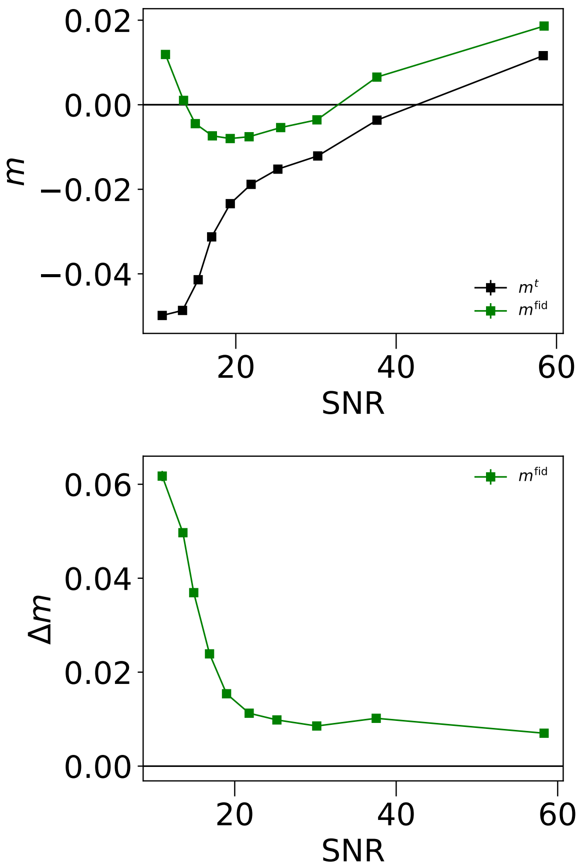

Fig. 7. Multiplicative bias predictions for GREAT3-RSC images as a function of S/N. *Top panel*: bias estimate from NNSC (green line) compared to the true bias (black line). *Bottom panel*: difference between the estimated and true bias as a function of S/N.

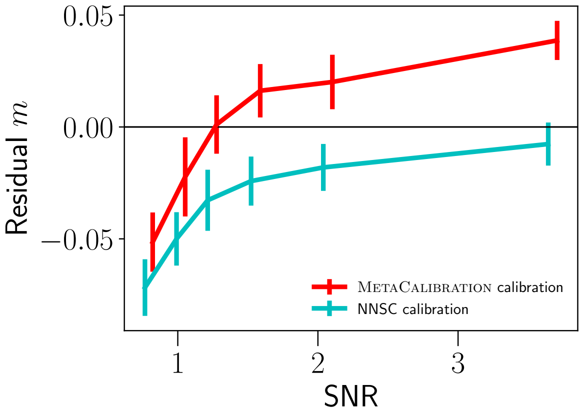

Fig. 8. Residual multiplicative bias after calibration for realistic images using MetaCalibration (red line) and the calibration from the bias estimate of NNSC (cyan line).

for realistic galaxy images, but it still obtains errors of about 1% for the majority of the cases.

Figure 8 shows the residual bias after the calibration was applied to the GREAT3-RSC images for NNSC and MetaCalibration. Again, the training for the NNSC calibration was that of the CSC branch. The difference in S/N values with respect to previous figures is that now we use the S/N estimate from KSB (we used the input parameter for GalSim or the S/N from GFIT for the remainder). For NNSC a residual bias is visible that decreases with S/N, consistent with the bias estimates from Fig. 7. It is beyond the scope of the paper to improve this calibration by applying a refinement to these images because we wish to show, on the one hand, the potential performance of the method (Sects. 5.1–5.3), and on the other hand, the effect of applying a crude calibration to a realistic data set for which the model was not trained (this section). Otherwise, an easy improvement could be achieved with a refinement of the model on realistic galaxy images, or by identifying poorly estimated objects (we found that the main contribution of this error comes from objects whose shear responses were estimated outside the range of values represented by the training, indicating a misrepresentation of these objects).

MetaCalibration also shows a significant residual bias dependence on S/N for GREAT3-RSC images. Although the residual bias over the entire population is very weak and consistent with previous analyses (Huff & Mandelbaum 2017), the method shows a negative residual bias for galaxies with a low S/N and a positive one for galaxies with a high S/N. We found this to be specific for the RSC images and this particular implementation. Different from CSC images, the estimated shear responses of MetaCalibration show a weak dependence on S/N. This is caused by some images that our KSB implementation interprets to be small, and a small window function was applied to them for the shape measurement. This produces a very similar calibration factor (~5% positive) for all S/N values, producing a shift of about 5% in the residual bias. We ignored the origin of this low sensitivity; it can come from a combination of factors. First, RSC images are created from pixelated and noisy real images that have been deconvolved with their PSF to which then a shear was applied, making these images imperfect. In additiont, MetaCalibration is ran, which again modified the images with a deconvolution, shearing, re-convolution, and a noise addition. Finally, KSB estimates the optimal size of the window function to estimate the galaxy shape.

6. Discussion

6.1. Potential limitations and solutions for NNSC

6.1.1. Selection bias

The aim of this paper is to show the performance of NNSC in calibrating shear measurement bias. To this end, we applied the following catalogue selection in order to remove any selection bias from the data. Any galaxy whose detection or shape measurement failed in any of the processes was removed from the catalogue, together with all its shear versions. This included not only the shear versions that were simulated with GalSim, but also the images derived from the MetaCalibration processing so that MetaCalibration has no selection bias either. Moreover, when the data were split into bins of a measured variable, all the shear versions were removed as well if a galaxy fell in different bins for different shear versions. With this, the selected data of our results were identical for all shear versions, and selection bias was forced to be zero.

This procedure can be applied here for the purpose of presenting a method in simulations, but in real data selection, effects cannot be avoided and need to be calibrated. In particular, selection bias comes from the fact that the selection function depends on the shear and is usually of the same order of magnitude as shear measurement bias (Fenech Conti et al. 2017; Mandelbaum et al. 2018b). MetaCalibration takes into account the shear dependence of selection effects and also calibrates selection bias in a similar procedure as it does for measurement bias, as described in Sheldon & Huff (2017). Sheldon et al. (2020) also explored a calibration of selection and measurement bias simultaneously in the presence of blended objects.

Analogously, NNSC can potentially be used to also calibrate selection bias. This would involve applying the same selection process to all galaxies independently of the shear and measuring the shear responses to this selection, as described in Sheldon & Huff (2017) and in Sect. 7.2 of PKSB19:

$$\left\langle R_{\alpha\beta}\right\rangle \approx \frac{\left\langle e_{\alpha}^{\mathrm{obs},+}\right\rangle - \left\langle e_{\alpha}^{\mathrm{obs},-}\right\rangle}{2\Delta g_{\beta}}, \tag{8}$$

where the ellipticities are measured for the case with no shear, and the + and − superscripts refer to the applied selection, corresponding to the catalogues obtained from the positive and negative shear versions, respectively. A supervised training could be applied to learn the shear response on selection. This would involve adapting the method so that the selection is specified in the input data (e.g. with weights specifying the selection for each shear version). Then the cost function involves the average selection shear response over a subset of the catalogue, as

$$C = \frac{1}{b_{\mathrm{s}}}\sum_{i=0}^{b_{\mathrm{s}}}\sum_{\alpha=1}^{2}\sum_{\beta=1}^{2}\left(\frac{w_i^{+}e_{i,\alpha}^{\mathrm{obs}} - w_i^{-}e_{i,\alpha}^{\mathrm{obs}}}{2\Delta g_{\beta}} - R_{i,\alpha\beta}^{\mathrm{e}}\right)^2, \tag{9}$$

where $w_i^{+,-}$ specifies the selection of galaxy i for the cases with positive or negative shear (it can be a weight from 0 for undetected cases to 1 for detected cases with the full signal), $e_{i,\alpha}^{\mathrm{obs}}$ is the observed ellipticity for the case with no shear, and $R_{i,\alpha\beta}^{\mathrm{e}}$ is the output estimated shear response of the training.

6.1.2. Model bias

For the results of our method in this paper, we used the same type of population for the training process as for the test and calibration. However, it is known that shear bias depends on the galaxy profile models (known as model bias). The images used in this paper consist of a mix of single Sérsic galaxies and galaxies with the sum of a bulge (de Vaucouleurs) and a disc (exponential). For the original training, testing, and calibration, we used a population in which 61% of the galaxies have single Sérsic profiles (this corresponds to the fraction in the whole set of images).

Here we quantify the effect of the model bias on our method that comes from these two different models by testing the performance of the method when different single Sérsic fractions are used for the training and the testing steps. In Fig. 9 we show the performance of the estimated multiplicative bias using different Sérsic population fractions for the training data as specified in the legends (m^{fid} corresponds to the original fraction of 61%, and m^{t} shows the true bias). The top panels show the results applied to the original population with a Sérsic fraction of 61%, in the middle panels we show the results on galaxies with a bulge and a disc (Sérsic fraction of 0%), and in the bottom panels we show the results applied to Sérsic galaxies (Sérsic fraction of 100%). The left panels show the bias dependence, and the differences with respect to the true are shown in the right panels.

In all the cases, the best performance is obtained when the training population coincides with the test population. On the other hand, all the cases give different shear bias predictions for the different test populations. For example, the model trained with only Sérsic galaxies predicts $m \sim -0.06$ for the galaxies with a disc and a bulge and $m \sim -0.02$ for the single Sérsic galaxies. Although the true m changes a 5−6% between the two populations, training with the single Sérsic population alone gives ∼1% error on the other population. In the opposite case, the model trained with disc and bulge galaxies predicts $m \sim 0.08$ for the same population and $m \sim 0.02$−0.08 for the single Sérsic population. This means that all the predictions are sensitive to the differences between the different populations, although the model bias still remains non-zero. This means that we have captured only a part of the dependency of the bias on S/N and type of galaxy. The red lines in the middle panel and the light green lines in the bottom panels show the extreme cases when the models were trained with a completely different population, and they give a model bias of ∼1% for bright galaxies. Possible ways to reduce this model bias could be (a) training with more complex models, (b) using more input measured complementary properties that can help to increase the sensibility to more complex images, and (c) using convolutional neural networks (CNNs) to exploit the information at the pixel level.

Finally, in Fig. 10 we show the same bias predictions, but now applied to RSC images. This shows the effect of model bias when different analytical models are trained and applied to realistic images. The fiducial model gives better predictions on RSC images than the extreme cases where the training is only made with one galaxy model, at least for $S/N > 20$. A good performance is encouraging for the practical effect of model bias in real observations with our training, and using more sophisticated or realistic models for image simulations would potentially improve the performance, as discussed in Sect. 5.5.

6.2. Advantages of the NNSC method

We presented NNSC as a new method for shear calibration. The method is different from others such as METACALIBRATION, self-calibration methods, the commonly applied calibration from shear bias measurements in binned properties from simulations or other ML approaches. We do not claim one method or approach to be better than the other, but we highlight several strengths of our method.

First, NNSC has the advantage that it can obtain a good performance in a matter of hours with a few thousand images, which is a very efficient ML approach for shear calibration. This is because, on the one hand, the input data are a reduced set of measured properties that significantly reduce the architecture dimensionality (compared to other ML approaches such as convolutional neural networks), and on the other hand, by focusing on minimising the error on the estimated individual shear response, we avoided the shape noise contribution of the intrinsic ellipticity (see PKSB19 for a detailed discussion of this contribution). It could also be possible to build an estimate of the shear bias straight from the galaxy and PSF images (e.g. from aCNN), but this would require a much more complex network architecture (e.g. accounting for the effect of the PSF, galaxy morphology, etc.). Learning from image properties already reduces the complexity of the images, without losing too much information of the shear bias.

As an example of an ML approach, Tewes et al. (2019) minimised the residual bias over the average measured ellipticities so that the output wass an unbiased shear estimator. Because of the intrinsic ellipticity contribution, the method requires a very large catalogue (10^6−10^7 objects) and uses a batch size of 500 000 objects (distributed so that the intrinsic ellipticity cancels out) to ensure that in each minimisation step, the shape noise from the intrinsic ellipticity is low. One suggestion to

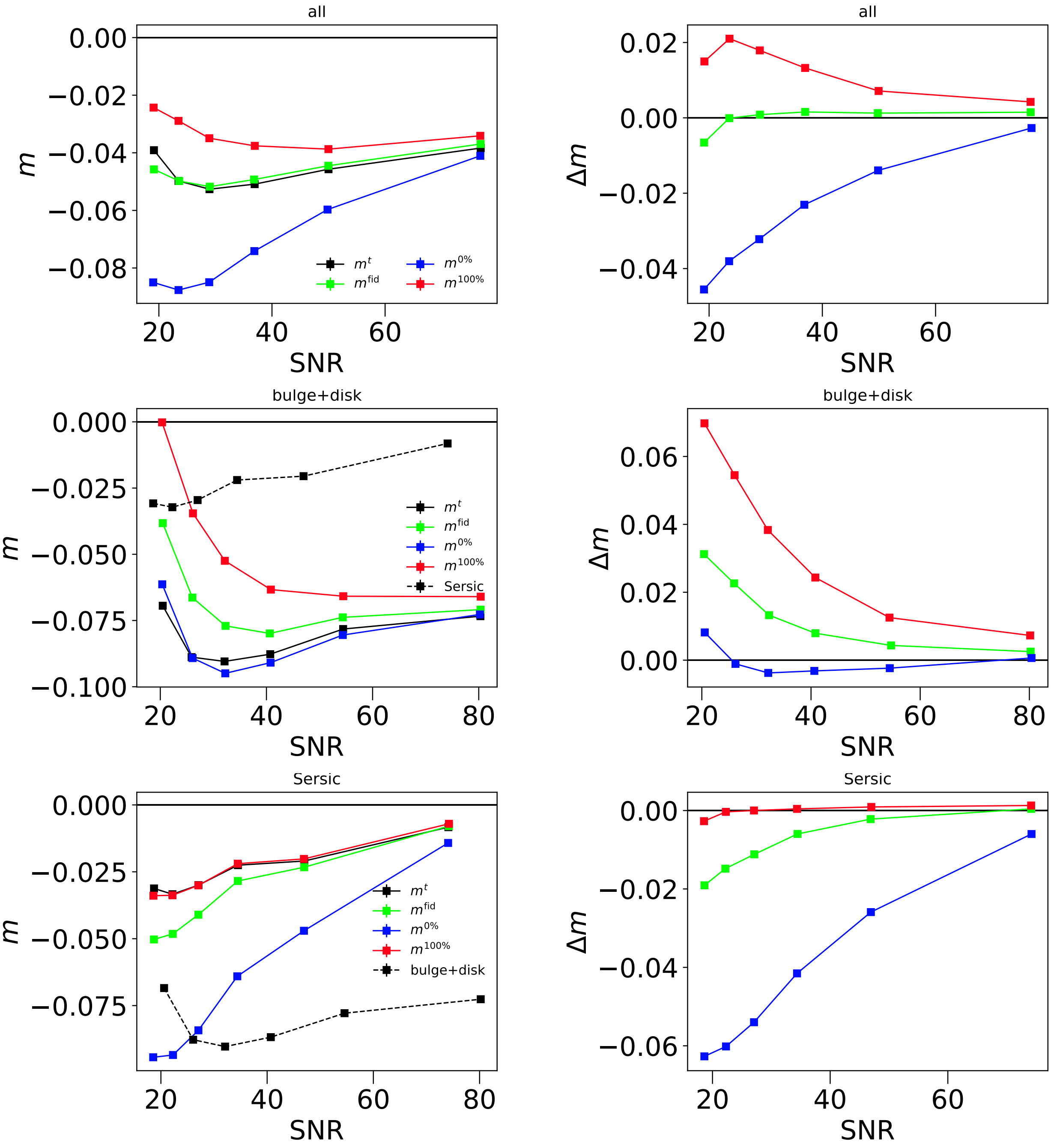

Fig. 9. Shear bias predictions using only single Sérsic (in red), only bulge+disc (in blue), or the real population (green) tested on the whole population (*top*), on only disc+bulge galaxies (*middle*) and on single Sérsic galaxies (*bottom*). Solid lines show the true bias of the populations, and the dashed black lines show the shear bias of the excluded population (the true bias for single Sérsic galaxies in the middle panels and for bulge+disc in the *bottom panel*).

improve the computational performance of Tewes et al. (2019) would be to minimise the individual responses (using sheared versions of the same images as in this paper and in PKSB19) instead of the residual bias over average ellipticities.

In another approach, Gruen et al. (2010) minimised shear bias for a KSB estimator by training the ellipticity measurement errors using the original measurements from KSB, the flux measured from SExtractor, and some tensors involved in the shape measurement process. The approach is similar to ours in the sense that they considered a set of properties to estimate errors on the shape measurements, but our method directly estimates the shear response, which avoids shape noise. As in Tewes et al. (2019), Gruen et al. (2010) used ∼10^7 objects for the training sets.

Another potential of the NNSC method is that it can be easily implemented for any survey for which we have image simulations (this is required for a good validation of the survey exploitation). To apply NNSC, we only need to produce copies of the same image simulations with different shear values applied, so that we can obtain individual shear responses. The set of measured properties to be used for the training is arbitrary and can be chosen according to the interests and the pipeline output

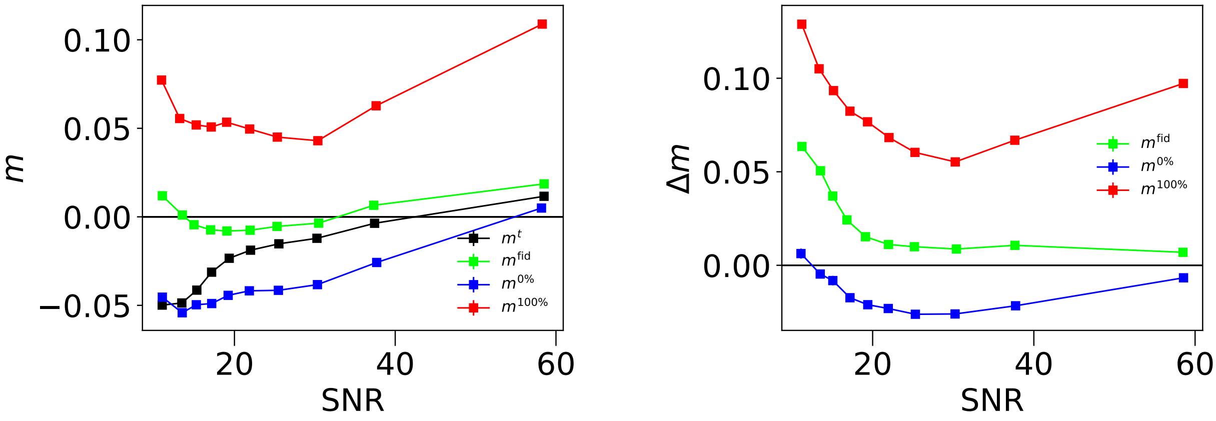

Fig. 10. Same as in Fig. 9, but applied to the realistic RSC images.

of the surveys. Even a simple application using a few properties of interest will already be an improvement with respect to common calibrations obtained from shear bias measurements in simulations as a function of two properties only. Moreover, as described in Sect. 6.1.1, the method can be extended to also calibrate selection bias.

Finally, the method allows us to use a large set of measured properties as input. When properly trained, this allows the calibration to be more reliable than the particular population distribution of the training with respect to the real data. If the bias dependences on many properties are correctly learnt, the particular distribution of the population over these properties should not affect the performance of the calibration (provided that no other unaccounted-for properties affect shear bias and that the simulated population is representative of the real data).

6.3. Importance of using complementary methods

We have used two different methods (METACALIBRATION and NNSC) and analysed their performances in image simulations. Both methods perform well, and they are complementary because they do not share the same systematics. Different implementations of the METACALIBRATION pipeline, as well as using other shape estimators, might improve or better adapt to the particular data. As for our model, the scope of this paper is not to show the best implementation of METACALIBRATION for these particular data, but to include it to compare NNSC with an advanced modern code.

Because the two models are complementary, using both for the same scientific analyses is a more suited approach to ensure the reliability of the results. For this reason, we encourage researchers to include at least two independent shear measurements and calibration methods for scientific analyses on galaxy surveys, as was done in Dark Energy Survey (Jarvis et al. 2016; Zuntz et al. 2018). This allows us to better identify systematics from the discrepancies between the methods that otherwise might be missed. At the same time, consistent results from two different and independent methods always give reliability to the scientific results, a crucial aspect for future precision cosmology. The combination of NNSC and METACALIBRATION brings a good complementarity because it uses an ML approach based on measurements from image simulations and a method that is independent of image simulations, but limited by other numerical processes.

7. Conclusions

Machine learning is a promising and emerging tool for astronomical analyses because it can characterise complex systems from large data sets. It is then especially well suited for shear calibration, where many systematics complicate the behaviour of shear bias and its calibration.

We presented a new shear calibration method based on ML that we call neural network shear correction (NNSC). We have also made the code publicly available. The method is based on galaxy image simulations that are produced several times with different shear versions, but the remaining conditions are preserved. With this, we obtained the individual shear response of the objects that served us for a supervised ML algorithm for estimating the shear responses of objects from an arbitrarily large set of measured properties (S/N, size, flux, ellipticity, PSF properties, etc.) through a regression approach.

This ML approach allowed us to characterise the complexity of shear bias dependences as a function of many properties, a complexity that we explored in Pujol et al. (2020). The advantage of ML is that it is an especially well suited approach to reproducing very complex systems so that we can include a large set of properties on which shear bias can depend. With these, the algorithm identifies the contribution of each of the properties and their correlations to estimate shear bias. With the individual shear bias estimates of galaxies, we can then apply a shear calibration based on the average statistics of shape measurements as in many other shear calibration approaches.

We used image simulations based on the GREAT3 (Mandelbaum et al. 2014) control-space-constant branch to explore the method, apply the training algorithm of NNSC, and evaluate its performance through tests and validations. We obtained a performance beyond *Euclid* requirements ($\Delta m_1 = (4.9 \pm 1.1) \times 10^{-4}$, $\Delta m_2 = (0.0 \pm 1.1) \times 10^{-4}$, $\Delta c_1 = (-3.1 \pm 0.7) \times 10^{-4}$ and $\Delta c_2 = (1.6 \pm 0.7) \times 10^{-4}$) for the estimates of the average shear biases, and we showed shear bias dependences on one and two properties below the 1% error. This performance was achieved with only ~15 CPU training hours using 128 000 objects, which is a very cheap and fast training compared to common ML approaches (the method from Tewes et al. 2019 takes about two CPU days with 10^6–10^7 objects). This means that the NNSC approach has great potential, but is also very easy to apply to galaxy surveys because it only requires different shear versions of the same image simulations and low computational and storage capacities.

We compared the residual bias after calibration as a function of several input properties with the advanced method MetaCalibration. This resulted in similar performances and showed different dependences because of the differences between the approaches and their systematics.

We quantified the effect of model bias by applying the calibration to different galaxy morphologies than were used for the training. We found errors of a few percent in some extreme cases that could be improved by refining the training with a more proper or sophisticated data set. Although it is not developed in this paper, the method can be easily adapted to also learn selection bias and calibrate for it just by adding information about the data selection as input data for the training and applying small changes in the cost function.

Our method is an implementation of ML based on a simple DNN architecture leading to fast calibration that can be easily applied to weak-lensing analyses of current and future galaxy surveys and can be a good complementary method to combine with other approaches and gain sensitivity to systematics and robustness to the science.

Acknowledgements. AP, FS and JB acknowledge support from a European Research Council Starting Grant (LENA-678282). This work has been supported by MINECO grants AYA2015-71825, PGC2018-102021-B-100 and Juan de la Cierva fellowship, LACEGAL and EWC Marie Sklodowska-Curie grant Nos 734374 and 776247 with ERDF funds from the EU Horizon 2020 Programme. IEEC is partially funded by the CERCA program of the Generalitat de Catalunya.

References

Akeson, R., Armus, L., Bachelet, E., et al. 2019, ArXiv e-prints [arXiv:1902.05569]
Bernstein, G. M. 2010, MNRAS, 406, 2793
Bertin, E., & Arnouts, S. 1996, A&AS, 117, 393
Bridle, S., Shawe-Taylor, J., Amara, A., et al. 2009, Ann. Appl. Stat., 3, 6
Bridle, S., Balan, S. T., Bethge, M., et al. 2010, MNRAS, 405, 2044
Clampitt, J., Sánchez, C., Kwan, J., et al. 2017, MNRAS, 465, 4204
Fenech Conti, I., Herbonnet, R., Hoekstra, H., et al. 2017, MNRAS, 467, 1627
Gentile, M., Courbin, F., & Meylan, G. 2012, ArXiv e-prints [arXiv:1211.4847]
Gillis, B. R., & Taylor, A. N. 2019, MNRAS, 482, 402
Gruen, D., Seitz, S., Koppenhoefer, J., & Riffeser, A. 2010, ApJ, 720, 639
Hall, A., & Taylor, A. 2017, MNRAS, 468, 346
Hildebrandt, H., Viola, M., Heymans, C., et al. 2017, MNRAS, 465, 1454
Hoekstra, H., Herbonnet, R., Muzzin, A., et al. 2015, MNRAS, 449, 685
Hoekstra, H., Viola, M., & Herbonnet, R. 2017, MNRAS, 468, 3295
Huff, E., & Mandelbaum, R. 2017, ArXiv e-prints [arXiv:1702.02600]
Jarvis, M., Sheldon, E., Zuntz, J., et al. 2016, MNRAS, 460, 2245
Kacprzak, T., Zuntz, J., Rowe, B., et al. 2012, MNRAS, 427, 2711
Kacprzak, T., Bridle, S., Rowe, B., et al. 2014, MNRAS, 441, 2528
Kaiser, N., Squires, G., & Broadhurst, T. 1995, ApJ, 449, 460
Kannawadi, A., Hoekstra, H., Miller, L., et al. 2019, A&A, 624, A92
Kitching, T., Balan, S., Bernstein, G., et al. 2011, Ann. Appl. Stat., 5, 2231
Kitching, T. D., Balan, S. T., Bridle, S., et al. 2012, MNRAS, 423, 3163
Kitching, T. D., Rowe, B., Gill, M., et al. 2013, ApJS, 205, 12
Laureijs, R., Amiaux, J., Arduini, S., et al. 2011, ArXiv e-prints [arXiv:1110.3193]
LSST Science Collaboration (Abell, P. A., et al.) 2009, ArXiv e-prints [arXiv:0912.0201]
Mandelbaum, R. 2018, ARA&A, 56, 393
Mandelbaum, R., Slosar, A., Baldauf, T., et al. 2013, MNRAS, 432, 1544
Mandelbaum, R., Rowe, B., Bosch, J., et al. 2014, ApJS, 212, 5
Mandelbaum, R., Rowe, B., Armstrong, R., et al. 2015, MNRAS, 450, 2963
Mandelbaum, R., Miyatake, H., Hamana, T., et al. 2018a, PASJ, 70, S25
Mandelbaum, R., Lanusse, F., Leauthaud, A., et al. 2018b, MNRAS, 481, 3170
Massey, R., Heymans, C., Bergé, J., et al. 2007, MNRAS, 376, 13
Massey, R., Hoekstra, H., Kitching, T., et al. 2013, MNRAS, 429, 661
Melchior, P., & Viola, M. 2012, MNRAS, 424, 2757
Pujol, A., Kilbinger, M., Sureau, F., & Bobin, J. 2019, A&A, 621, A2
Pujol, A., Sureau, F., Bobin, J., et al. 2020, A&A, 641, A164
Refregier, A., Kacprzak, T., Amara, A., Bridle, S., & Rowe, B. 2012, MNRAS, 425, 1951
Rowe, B. T. P., Jarvis, M., Mandelbaum, R., et al. 2015, Astron. Comput., 10, 121
Sheldon, E. S., & Huff, E. M. 2017, ApJ, 841, 24
Sheldon, E. S., Becker, M. R., MacCrann, N., & Jarvis, M. 2020, ApJ, 902, 138
Taylor, A. N., & Kitching, T. D. 2016, ArXiv e-prints [arXiv:1605.09130]
Tessore, N., & Bridle, S. 2019, New Astron., 69, 58
Tewes, M., Kuntzer, T., Nakajima, R., et al. 2019, A&A, 621, A36
Viola, M., Melchior, P., & Bartelmann, M. 2011, MNRAS, 410, 2156
Voigt, L. M., & Bridle, S. L. 2010, MNRAS, 404, 458
Zhang, J., & Komatsu, E. 2011, MNRAS, 414, 1047
Zuntz, J., Kacprzak, T., Voigt, L., et al. 2013, MNRAS, 434, 1604
Zuntz, J., Sheldon, E., Samuroff, S., et al. 2018, MNRAS, 481, 1149

Appendix A: Details of the training

In this section we describe the training procedure and the architecture, cost function, and hyper-parameters we implemented for the results. This configuration was chosen according to the performances found during the optimisation process. Even if these optimisation parameters could be further studied and optimised, we already obtained competitive results with the procedure and the optimisation tests described below.

Our configuration consists of a DNN with four hidden layers of 30 units per layer. The input consists of the 27 measured properties as described before, and the output consists on the four shear bias components.

We first applied a whitening with principal component analysis (PCA) to the input data in order to decorrelate the 27 properties, and we normalised them to be between 0 and 1. This procedure aims at avoiding some properties or information to dominate their contribution due to their value ranges or correlations.

Here we recall the cost function C, which minimises the χ^2 on the estimated shear biases,

$$C = \frac{1}{b_\mathrm{s}} \sum_{i=0}^{b_\mathrm{s}} \sum_{\alpha=1}^{2} (m^\mathrm{t}_{i,\alpha} - m^\mathrm{e}_{i,\alpha})^2 + (c^\mathrm{t}_{i,\alpha} - c^\mathrm{e}_{i,\alpha})^2, \tag{A.1}$$

where m^t_α, c^t_α and m^e_α, c^e_α are the true and estimated αth component shear multiplicative and additive bias, respectively, and b_s is the batch size. Given that usually $m_{i,\alpha} \gg c_{i,\alpha}$, the contribution of $m_{i,\alpha}$ in the current approach dominates the minimisation process, but additional weights can be applied to some biases when the performance of the estimation of these biases is to be differently. Here we include the four bias components in the cost function to estimate them simultaneously, but separate trainings for each of the components can also be made. Separate trainings for each of the components might allow using simpler architecture and faster learning, but it would miss the correlation between the components in the learned model.

We constrained the hyper-parameters of the algorithm (contamination level, number of training objects, number of epochs, batch size, learning rate, and learning decay rate) by analyzing the performance and convergence in a wide hyper-parameter space, choosing the final hyperparameter set from the best cases that we found by avoiding overfitting according to the cost function values of the training and test sets (see Fig. A.1 for the evolution of the cost function during the training of the fiducial example). For this, we evaluated the cost function in the training and the test sets to ensure that, on one hand, the costs converge during the training, and on the other hand, that the cost in the training set is not significantly lower than the cost in the test set, which would be a symptom of overfitting. In some cases, the cost in the test set was found to be lower than in the training set. These cases were also discarded for this accidental overfitting.

In Fig. A.2 we show the performance of the shear bias estimates for different hyper-parameters of the ML optimisation. The left panels show the shear bias dependence as a function of S/N for the true multiplicative bias and the estimated biases. The right panels show the difference between the estimated and true bias as a function of S/N.

In the top panel we show the performance for a different number of objects used in the training set, going from 16 000 to 256 000 objects (the fiducial case was computed with 128 000 objects). More objects improve the performance, which reaches some plateau for more than 50 000 objects. This is a very small number of objects compared with most of the shear calibration approaches (Zuntz et al. 2018; Tewes et al. 2019; Kannawadi et al. 2019).

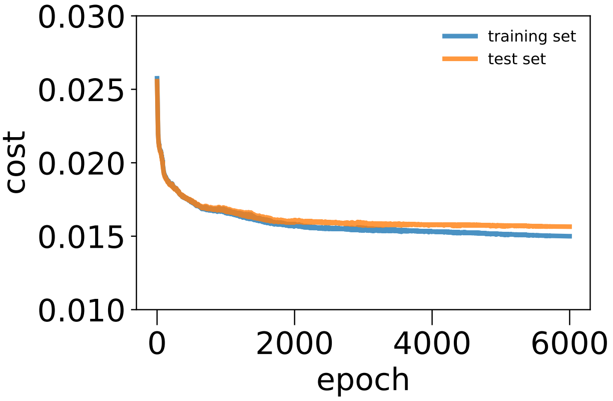

Fig. A.1. Evolution of the cost function for the training and the test sets during the training on the fiducial implementation of NNSC.

The middle panels show the performance as a function of the number of epochs used in the training (with 6000 epochs for the fiducial case). In this case, the training converges for more than 6000 epochs. For our final case we used 6000 epochs, which took about 15 CPU hours of training, but we note that similar results are obtained with longer trainings. This shows that our method has a very fast training compared to other ML approaches.

In the bottom panels we compare the performance using different architectures. Our final case uses four hidden layers with 30 nodes in each layer, but here we compare it with a case of a narrower architecture (with four hidden layers with 30, 20, 10, and 10 nodes), a shallow one (with two hidden layers of 90 and 10 units), and a larger one (with four hidden layers of 50 nodes each). The performances as a function of S/N are very similar, with no obvious conclusion about which architecture is giving better predictions. However, Fig. A.3 shows the interest of using a wider and deeper architecture. In the top left panel, we show the true m_1 as a function of two properties (the two ellipticity components measured by KSB). The other panels show the difference between the estimated and the true m_1 for the large architecture (top right), the shallow (bottom left), and the narrow (bottom right) architectures. The narrow and shallow architectures are not able to capture the entire complexity of the 2D dependence as efficiently as our fiducial architecture. This is proof that a wide and deep neural network allows us to better capture the complexity of the system. Although not shown here, the performances of the larger architecture are very similar to those of the fiducial one, and the performance is poorer for the same training time. This indicates that the largest architecture does not help improve the performance and loses efficiency of the training, and for this reason, we kept the four hidden layers of 30 nodes as the fiducial model for the purpose of this paper.

For the remaining hyper-parameters we did not find a strong dependence on the batch size, showing good performance between 16 and 256 (32 were finally chosen), and we found that adding no noise contamination to the input data was optimal for the performance. About the activation function of the layers, we used a leaky ReLU function for the nodes. We found similar performances using tanh functions or combinations of both, but with a significantly slower convergence. The results shown in the paper for the chosen model were obtained with about 15 CPU hours.

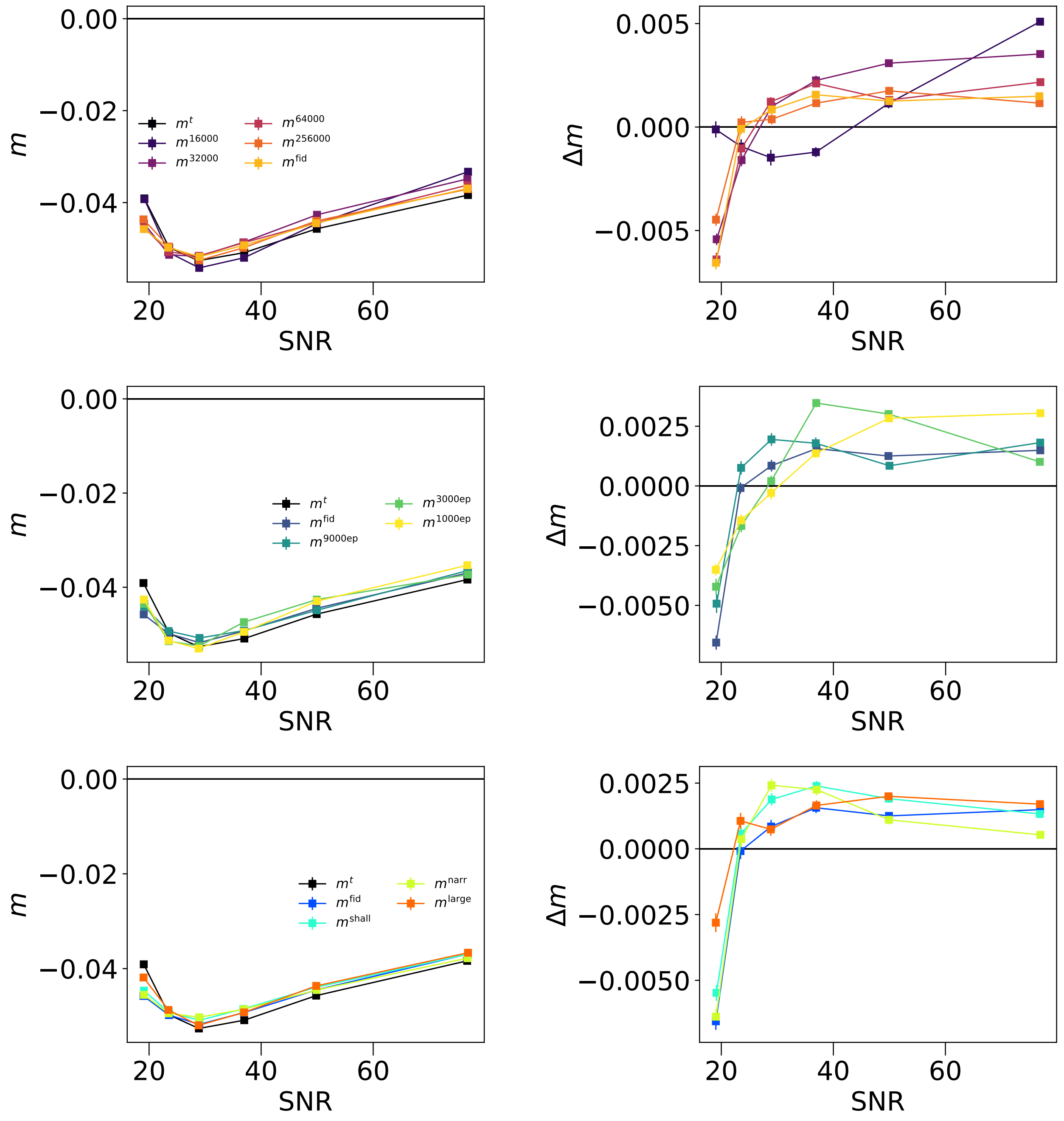

Fig. A.2. Comparison of estimated multiplicative shear bias m_1 (*left*) and its error Δm_1 (*right*) as a function of S/N for different ML hyperparameters. *Top panels*: performance for different numbers of objects used in the training, from 10 000 to 400 000. *Middle panels*: performances for the chosen architecture for different number of epochs, from 1000 to 9000. *Bottom panels*: performance for different architectures. The chosen one, shown in blue, corresponds to four hidden layers of 30 nodes each. The shallow architecture, shown in green, has only two hidden layers of 30 nodes. The narrow architecture, shown in orange, corresponds to four hidden layers of 30, 20, 10, and 10 nodes.

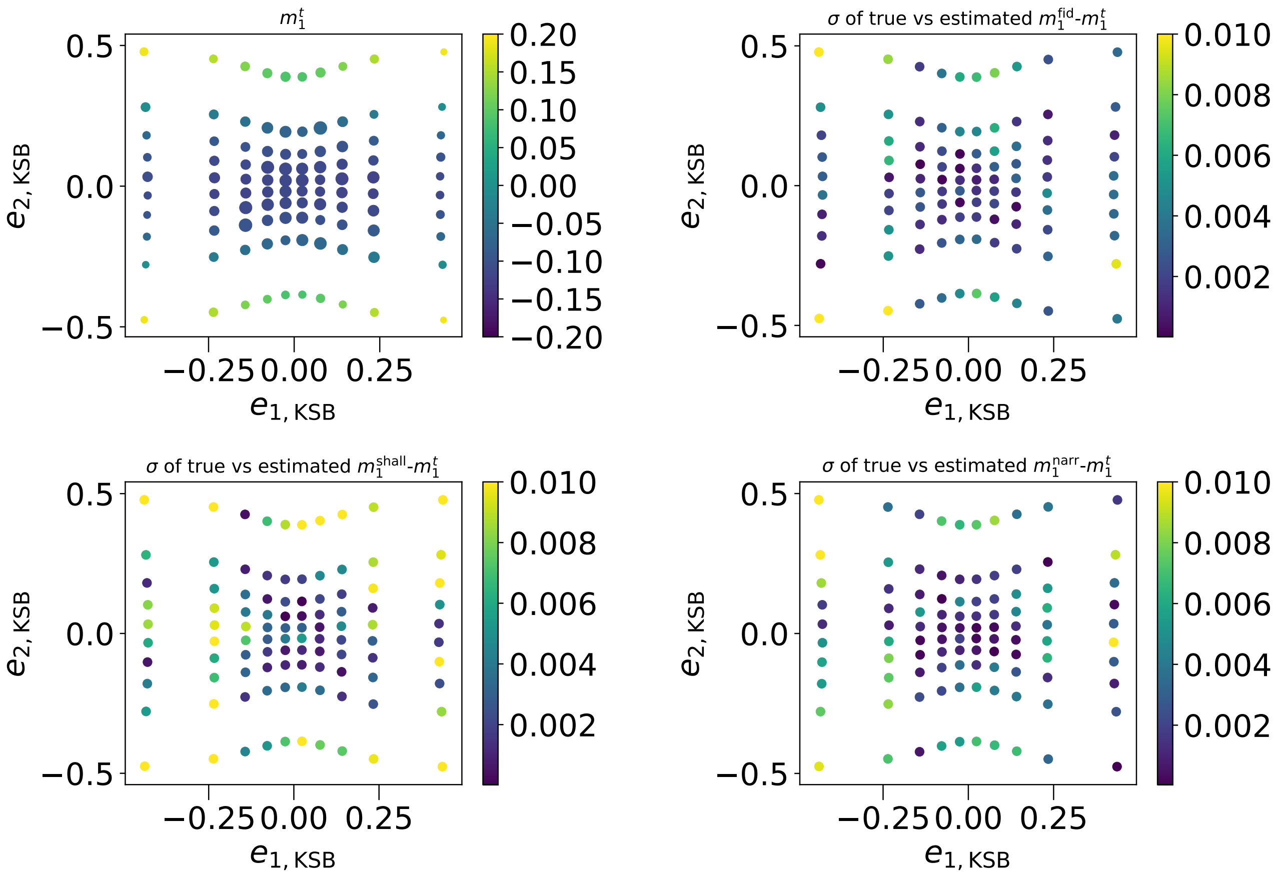

Fig. A.3. Comparison of estimated multiplicative shear bias m_1 as a function of galaxy measured ellipticities. *Top left panel*: true values of m_1, and the remaining panels show the difference between the estimated and the true for the fiducial architecture (*top right*), the shallow architecture (*bottom left*), and the narrow one (*bottom right*).

Appendix B: Metacalibration

The calibration method METACALIBRATION has been presented in Huff & Mandelbaum (2017) and Sheldon & Huff (2017), with a publicly available implementation[3]. It has been tested on simulations and applied to the Dark Energy Survey (DES) Y1 data (Zuntz et al. 2018), showing a very good performance. We used this as a reference recent calibration method for comparison with our new approach. We briefly describe the idea of this method and refer to Huff & Mandelbaum (2017) and Sheldon & Huff (2017) for more details.

METACALIBRATION is based on measuring the shear response of the individual galaxy images without any need of image simulations. To do so, METACALIBRATION uses the original real image to generate sheared versions of it. With these sheared versions, the shear response is obtained using Eq. (2) as in our method. The shear calibration is then applied to the data using the mean of these individual shear responses and their propagation through the statistics of interest.

To generate the sheared versions of the original image, METACALIBRATION first deconvolves the image with the PSF (which is assumed to be perfectly known). Then the shear distortion is applied, and the image is reconvolved with a slightly higher PSF. As this process induces correlated sheared noise, a noise image following the same procedure but with opposite sign shear is also added to reduce the effect of this correlated shear on the shear response. The final noise realisations can be significantly different than the original one, which can have an effect on the shear response. Because of this, a non-sheared new image is also generated with the same process. On this new image, shear is measured and calibrated for science analyses. In addition, the additive bias can be measured using Eq. (18) from Huff & Mandelbaum (2017), and the shear response coming from selection biases can be estimated (Sheldon & Huff 2017).

METACALIBRATION has the advantage that no image simulations for the calibration are required (although its performance can only be tested in simulations). On the other hand, the method depends on the numerical processes involving deconvolution, reconvolution, and the treatment of noise.

METACALIBRATION allows for different implementations regarding the characterisation of the PSF, including a Gaussian parameter fitting of the PSF (so that a combination of Gaussian profiles is used as the PSF), a symmetrisation (where three different rotations of the PSF are stacked to avoid a contribution of its ellipticity), and using the true PSF directly. We used the true PSF, but we found very similar results using the symmetrisation. Moreover, METACALIBRATION can be applied for any shape measurement algorithm for which the method calibrates its shear bias. We used METACALIBRATION to calibrate the KSB mesurements from the software SHAPELENS.

[3] https://github.com/esheldon/ngmix/wiki/Metacalibration