Simulation based inference with domain adaptation for strong gravitational lensing

Marcos E. Tamargo-Arizmendi[a)]

(Dated: 5 December 2023)

Simulation based inference leverages machine learning to carry out Bayesian inference in systems with intractable likelihoods. However, transitioning a network trained on simulated data to real data runs the risk of encountering domain shift, leading to performance losses. We attempt to implement domain adaptation into the `sbi` neural posterior estimation framework using the Maximum Mean Discrepancy as an additional network loss, using masked autoregressive flow (MAF) as our density estimator. We test the network on a set of 400,000 simulated strong gravitational lensing images generated using `deeplenstronomy`. The source domain is defined as low noise whereas the target domain has a noise profile sampled from experimentally derived DES survey conditions. We find that SBI appears robust against small changes in the data with similar performance on source and target. Moreover, while DA does lead to performance improvements, they are marginal at 6% less inference error.

I. INTRODUCTION

Gravitational lensing is a phenomenon whereby the trajectory of light is deflected as it passes through a gravitational potential. For systems with sufficiently high surface mass densities, the effect is significant enough such that it can be directly observed. In the strong lensing regime, the image of the lensed object is split into multiple images, magnified, and displaced relative to the true on-sky position[1]. Lenses of this kind are powerful tools for studying both the lens as well as the lensed object. However, these studies are all contingent on being able to obtain a reasonable model of the lens. Modeling gravitational lenses often follows a Bayesian inference procedure using posterior density samplers and forward modeling programs in order to iteratively approximate the data using the model. Such methods traditionally require closed-form likelihood functions.[2] However, in reality, the likelihood functions for these models are intractable and often need to be approximated, leading to reduced quality of inference.[3]

The problem of performing parameter inference using mechanistic models with intractable likelihoods has been a longstanding challenge in many fields of research. While methods do exist for performing inference in such scenarios, these often suffer from expensive computational requirements and the curse of dimensionality, meaning that sampling efficiency goes down with the dimensionality of the problem[3]. However, recent improvements in the performance and reliability of machine learning models has allowed for a new class of methods for tackling this problem. Simulation Based Inference (SBI) leverages the capacity of neural networks for regression tasks in order to learn the likelihood of an observation using simulated data generated from a mechanistic model[3]. SBI offers several advantages over alternative methods. First, SBI requires only one upfront computationally expensive simulation step, such that inference is amortized. Moreover, Machine Learning models are able to take advantage of the full data space, reducing the need for summary statistics. Finally, since the likelihood is not being approximated, SBI offers improved quality of inference over other methods[3].

While a powerful technique, the practical implementation of SBI for inference on real data is hindered by the problem of domain shift which arises when two data sets have different properties or are drawn from different distributions[4]. In such cases, networks trained on some source domain will experience performance losses when applied to a target domain composed of similar but slightly different data[5] (for example simulated and observational data). For the purpose of this study we consider only the problem of covariate shift, where the conditional probability relating labels θ to data x remains the same $p_s(\theta|x) = p_t(\theta|x)$, but the marginal probabilities of the data differ[5] $p_s(x) \neq p_t(x)$. For instance, the presence of noise in target data represents a covariate shift when source data are noiseless. The class of techniques aimed at learning models for use across domains is known as domain adaptation.[6]

In this study, we attempt to introduce a feature-based domain adaptation scheme into the existing simulation based inference package `sbi` in order to parameterize galaxy-galaxy strong gravitational lenses by directly inferring the posterior. Given the low number of existing DES strong lensing exemplars with fully developed models, we approximate the difference between real and simulated data by producing a source domain with stable noise conditions and a simulated target domain with variable DES survey conditions. Ultimately, the goal of implementing this algorithm is to develop a easily scalable inference method which allows for the rapid parametrization of galaxy-galaxy strong lensing observations. This would allow researchers to take greater advantage of the rapidly increasing lensing datasets obtained from new astronomical surveys. In §II we discuss density estimation procedure which we attempt to apply domain adaptation to. In §III we discuss the Maximum Mean Discrepancy,

[a)]Fermi National Accelerator Laboratory

the feature-based distance measure we employ as the additional loss function in network training. In §IV we go over the simulated lensing sample and in §V we give an overview of the network architecture we employ. Finally in §VI and VII we go over the experiments carried out on the network, their results and the implications of our findings.

II. AUTOMATIC POSTERIOR TRANSFORMATION

We carry out posterior estimation using the publicly available simulation-based inference package `sbi`[7], utilizing its implementation of Automatic Posterior Transofrmation (APT or SNPE-C) to perform posterior inference[8]. APT is a Bayesian neural posterior estimation method which leverages elements of synthetic likelihood methods and recasts inference as a density ratio estimation task, allowing it to incorporate flow-based density estimators with arbitrary proposal distributions[8]. The inference procedure follows by maximizing the probability of the simulation parameters under a proposal posterior which can be transformed into the true posterior. We do so by minimizing the negative log probability

$$\mathcal{L} = -\sum_{i=1}^{N} \log \Pr(\theta_i) \tag{1}$$

of the proposal posterior[8].

As a density estimator we implement Masked Autoregressive Flow (MAF), which leverages the benefits of both autoregressive models and normalizing flows[9]. Autoregressive models estimate a target density by decomposing it into a product of 1D conditionals[10] $p(x) = \prod_i p(x_i|x_{1:i-1})$. Normalizing flows estimate a target density as an invertible, differentiable transformation f with a tractable jacobian of a simpler base density[11] $\pi_u(\mathbf{u})$ given by

$$p(x) = \pi_u(f^{-1}(x)) \left| \det \left(\frac{\partial f^{-1}}{\partial x} \right) \right|. \tag{2}$$

When used to generate data, autoregressive models correspond to a differentiable transformation of an external source of randomness[12] and for specific parametrizations these turn out to be equivalent to normalizing flows[9]. MAF takes advantage of this property by stacking multiple autoregressive models with different orderings of the input variables and then modelling the density in a single forward pass through the flow[9]. In this way, MAF is made more flexible than a standard normalizing flow.

III. DIVERGENCE ALIGNMENT WITH MMD

To perform domain adaptation, we follow a statistic divergence alignment procedure which relies on minimizing the distance between the feature representations of the source and target data[13]. In principle, this is achieved by calculating a metric which measures the divergence between the first or higher-order statistics of the source and target domains, and introducing this metric as an additional loss during training. We use the Maximum Mean Discrepancy (MMD), a metric which measures the nonparametric distance between the mean embeddings of two distributions in a Reproducing Kernel Hilbert Space (RKHS)[14]. This is achieved by defining a mapping of the domains into the RKHS, $\mathcal{H}$, and minimizing the distance between the mean embeddings $\mu_s, \mu_t \in \mathcal{H}$ of these mappings

$$\mathrm{MMD}^2[\mathcal{F}, s, t] = \|\mu_s - \mu_t\|^2_{\mathcal{H}}. \tag{3}$$

Using the fact that $\mu := \frac{1}{n}\sum_{i=1}^{n} \varphi(x_i)$ where $\varphi(x)$ is a feature mapping in $\mathcal{H}$, and that $\langle \varphi(x), \varphi(y) \rangle_{\mathcal{H}} = k(x, y)$, it is possible to obtain the distance in terms of a kernel function[15] k

$$\begin{aligned}\mathrm{MMD}^2[\mathcal{F}, X, Y] = \frac{1}{m^2}\sum_{i,j=1}^{M} k(x_i, x_j) + \frac{1}{n^2}\sum_{i,j=1}^{M} k(y_i, y_j) \\ - \frac{2}{mn}\sum_{i,j=1}^{M} k(x_i, y_j),\end{aligned} \tag{4}$$

where $x, y \in X, Y$ are random variables drawn from distributions p, q respectively. In order to capture a range of mean embeddings We choose k to be a linear combination of Gaussian radial basis functions given by

$$k(x, x') = e^{-\frac{\|x - x'\|^2}{2\sigma^2}}, \tag{5}$$

where x' is an independent copy of x with the same distribution.[16]

IV. STRONG LENSING SIMULATIONS

We simulate the strong lensing datasets using `deeplenstronomy`[17], a wrapper package for `lenstronomy`[18] designed for generating large lensing datasets for machine learning. We produce a total of 400,000 $g-$band images of galaxy-galaxy lensing systems with 200,000 source and target samples each. For each simulation, we define the image and survey conditions, as well as the mass and light profiles.

Both source and target are produced with identical image parameters but varying survey conditions. Source survey conditions were fixed at a seeing of 0.9", a magnitude zero-point of 30.0, and a sky-brightness of 23.5 mag arcsec^{-2} at 10 exposures. Target survey parameters were drawn from experimentally measured DES survey conditions as specified in Abbot et. al (2018)[19]. In Figure 1 we display a sample of 10 randomly selected lenses from each domain in order to illustrate the differences.

We parametrize the lens galaxy mass profile using a singular isothermal ellipsoid (SIE) mass distribution[20].

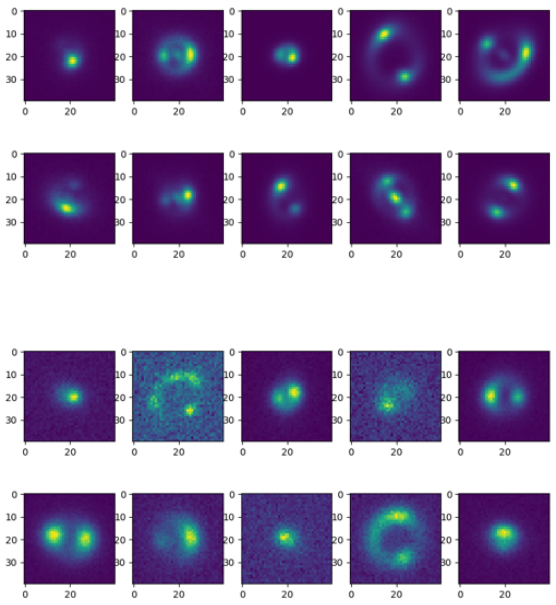

FIG. 1. Random sample of 10 lenses from source and target domains illustrating differences in samples

Parameter	Train Set Priors	Test Set Priors
θ_E	$\mathcal{U}(0.3, 4.0)$	$\mathcal{U}(0.5, 3.0)$
e_1	$\mathcal{U}(-0.8, 0.8)$	$\mathcal{U}(-0.2, 0.2)$
e_2	$\mathcal{U}(-0.8, 0.8)$	$\mathcal{U}(-0.2, 0.2)$
x	$\mathcal{U}(-2, 2)$	$\mathcal{U}(-1, 1)$
y	$\mathcal{U}(-2, 2)$	$\mathcal{U}(-1, 1)$
γ_1	$\mathcal{U}(-0.05, 0.05)$	$\mathcal{U}(-0.05, 0.05)$
γ_2	$\mathcal{U}(-0.05, 0.05)$	$\mathcal{U}(-0.05, 0.05)$

TABLE I. SIE Parameter priors used to generate training and test sets where $\mathcal{U}(a, b)$ is a uniform distribution with upper and lower limits a, b.

The SIE model is specified using the Einstein radius θ_E, the ellipticity (e_{m1}, e_{m2}), the position relative to the image origin (x, y), and a shear profile defined by two shear terms (γ_1, γ_2). We generate a test set with truncated priors in order to avoid edge effects caused by sample bias at the upper and lower prior limits. We summarize the parameter priors for both the training and test sets in Table I.

We model the light distribution using a Sérsic luminosity profile parameterized by the Sérsic index n, the half-light or effective radius R_e, the ellipticity (e_{l1}, e_{l2}), and position[21] (x, y). These parameters are held fixed for the lens galaxy at $(x, y) = (0, 0)$, $R_e = 1''$, $n = 1$, $(e_{l1}, e_{l2}) = (0, 0.5)$ with only the magnitude allowed to vary uniformly from 22.0-25.0. The parameters for the lensed galaxy are specified in Table II. Since we are not fitting for these parameters, both training and testingg sets are generated with identical light profile priors.

Parameters	Priors
m_e	$\mathcal{U}(19, 24)$
$R_e('')$	$\mathcal{U}(0.5, 1)$
n	$\mathcal{U}(2, 4)$
x	0.0
y	0.0
e_{l1}	$\mathcal{U}(-0.2, 0.2)$
e_{l2}	$\mathcal{U}(-0.2, 0.2)$

TABLE II. Sérsic profile parameters for lensed galaxy where $\mathcal{U}(a, b)$ is a uniform distribution with upper and lower limits a, b.

V. NETWORK ARCHITECTURE

Given the high dimensionality of the simulation outputs, we use a Convolutional Neural Network to extract and summarize important features from the data. These summaries are then used as the inputs for the density estimator. Both the embedding network and density estimator are trained simultaneously during the inference step. The embedding net consists of three convolutional layers and a single fully connected layer with a maxpooling and batch normalization operation between each convolutional layer. The network along with parameters is summarized in Table III. The MAF density estimator is constructed with 400 hidden units with 20 transformations each. The latent feature representation used for the MMD calculation is obtained between the convolutional and fully connected layers of the embedding network. The MMD distance is then simply introduced as an additional loss added to the standard negative log-probability loss

$$\mathcal{L}_{tot} = -\sum_{i=1}^{N} \log \Pr(\theta_i) + \lambda \mathcal{L}_{\mathrm{MMD}^2}, \tag{6}$$

where λ is a parameter weight which controls the influence of $\mathcal{L}_{\mathrm{MMD}^2}$. The network is allowed to train until convergence, defined as 20 epochs without improvement in $\mathcal{L}_{tot}$.

In order to determine the optimal value for λ we employ the hyperparameter tuning software `optuna`[22]. During the tuning process, network training occurs identically to normal operation with the exception of being limited to a duration of 20 epochs. Through this process, an optimal weight value of $\lambda \approx 9.95$ was obtained. In addition to the optimal value, we also test $\lambda = 0.02$ in order to determine the influence of MMD on the optimization process. We train the network on an NVIDIA Ampere A100 20GB GPU.

Layer	Output Shape	Parameters
Conv2d	[-1,8,42,42]	k=3, s=1, p=2
BatchNorm2d	[-1,8,42,42]	k=3, s=1
MaxPool2d	[-1,8,21,21]	k=2, s=2
Conv2d	[-1,16,21,21]	k=3, s=1, p=1
BatchNorm2d	[-1,16,21,21]	k=3, s=1
MaxPool2d	[-1,16,10,10]	k=2, s=2
Conv2d	[-1,32,10,10]	k=3, p=same
BatchNorm2d	[-1,32,10,10]	k=3, s=1
MaxPool2d	[-1,32,5,5]	k=2, s=2
Linear	[-1,20]	in_ft=$(32 \times 5 \times 5, 5)$

TABLE III. Embedding Net summary where k is kernel size, s is stride, and p is padding

VI. RESULTS

A. Posterior validation

To evaluate network performance, we generate 10,000 test samples half of which are source and half are target domain. We check the performance of the network by drawing posterior sample $p(\theta|x_o)$ given an observation x_o and comparing the posterior mean with the ground truth parameters θ_o. We first check that the ground truth parameters fall within the support of the posteriors. In Figures 8 and 9 we demonstrate this posterior check for a randomly selected lens in the target domain with both no DA implementation and MMD at $\lambda = 0.02$, respectively. As can be seen, without DA the posterior was generally not well recovered with the parameter distributions lacking tight convergence and exhibiting correlations. This was only marginally improved for some parameters after DA was implemented, nevertheless, network performance appears comparable and for some parameters appears worse. Moreover, while DA is able to achieve better fits on some dimensions, such as θ_E and e_{m2}, these are not significant enough to constitute a definitive improvement. As an additional qualitative check on performance, we plot in Figures 2 and 4 the true vs predicted parameter values for all five dimensions as a density map. We also plot the true versus predicted parameter values with corresponding uncertainty in Figures 6 and 7. While slight improvements can be observed in the parameter fitting, especially in the case of θ_E, the performance is generally comparable between the no DA and MMD networks. Moreover, if we compare Figure 2 to the source domain in Figure 5, we see that performance differences between the two domains are minimal.

As quantitative checks on model performance we calculate the χ^2 test statistic

$$\chi^2 = \frac{1}{N}\sum_{i=1}^{N}\frac{(\theta - \hat{\theta})^2}{\sigma^2}, \tag{7}$$

metric	No DA	MMD; $\lambda = 0.02$	MMD; $\lambda = 9.95$
χ^2	1.164	1.084	1.193
R^2	-0.1106	0.04701	-0.2020
MSE	0.04504	0.04160	0.05088

TABLE IV. Performance scores for network trained both without Domain Adaptation and MMD at $\lambda = 0.02$ and $\lambda = 9.95$

the coefficient of determination

$$R^2(\theta, \hat{\theta}) = 1 - \frac{\sum_{i=1}^{N}(\theta_i - \tilde{\theta}_i)^2}{\sum_{i=1}^{N}(\theta_i - \bar{\theta})^2}, \tag{8}$$

and the mean squared error

$$MSE(\theta, \hat{\theta}) = \frac{1}{N}\sum_{i=0}^{N-1}(\theta_i - \hat{\theta})^2, \tag{9}$$

where θ is the true parameter value, $\hat{\theta}$ is the posterior mean, σ is the standard deviation of the posterior, $\bar{\theta}$ is the mean over the ground truth values, and $\tilde{\theta}$ is the mean predicted values. The test statistics were calculated for each dimension of the parameter space and the mean across all five scores was taken to be the overall model score. We summarize the network scores in Table IV. From the metric values we can see that there was a slight improvement in performance with MMD implemented at $\lambda = 0.02$. The χ^2 and MSE scores improved by 6.8% and 7.6% respectively while the coefficient of determination saw an improvement of 142%. However, in the case of the R^2 score, while apparent improvements were large, visual inspection of the predicted parameter plots does not reveal the discrepancy suggested by the score. Moreover, the R^2 for the MMD trial is nevertheless much less than 1, and therefore does not indicate good model performance overall. Comparing these results with a DA at $\lambda = 9.95$, we see that an excessive weighting of $\mathcal{L}_{MMD^2}$ leads to worse model performance, with an $\sim 82\%$ lower R^2 value, and $\sim 13\%$ higher MSE.

Of the three networks, the best performing was the network trained with and MMD weight of $\lambda = 0.02$. For all other values of λ, network performance was worse than with no domain adaptation. That said, improvements were marginal.

B. Uncertainty Calibration

In addition to evaluating the predictive power of the network, we perform a simulation-based calibration procedure using the rank statistics of the ground truth parameters in order to constrain the validity of the network uncertainty. For a well-calibrated inference algorithm, the ranks of the ground truth parameters under the inferred posterior must follow a uniform distribution[23]. We therefore perform a series of qualitative and quantitative

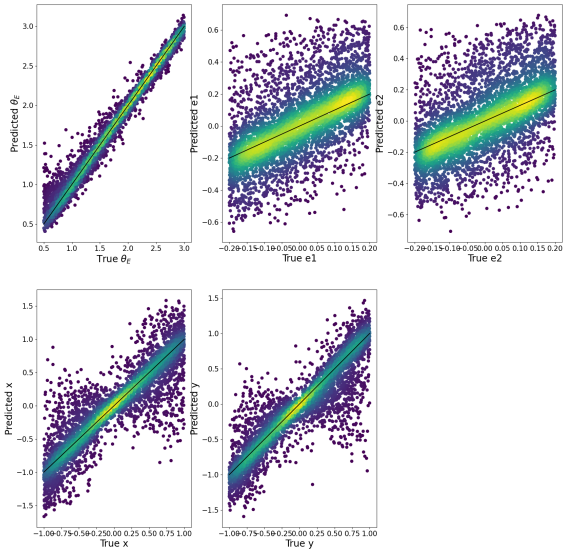

FIG. 2. True versus predicted parameter values for target domain with no domain adaptation implemented. The points are colored according to number density with yellower points indicating higher density.

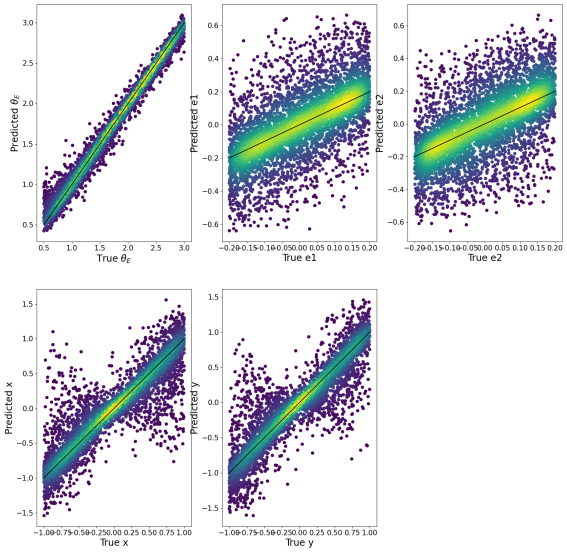

FIG. 3. Caption

FIG. 4. True versus predicted parameter values for target domain with MMD at $\lambda = 0.02$ implemented. The points are colored according to number density with yellower points indicating higher density.

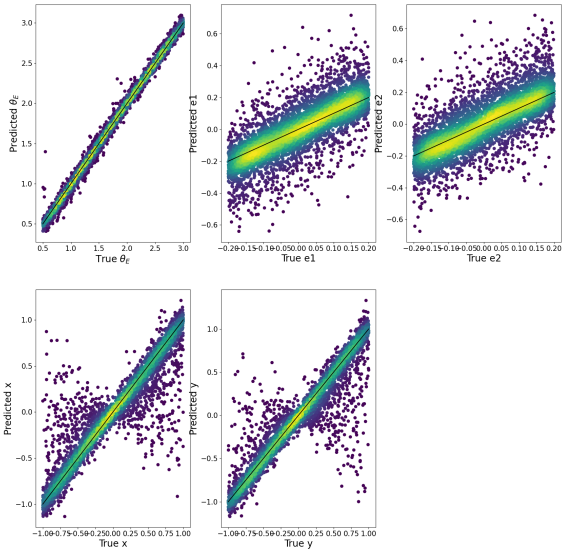

FIG. 5. True versus predicted parameter values for source domain with no DA. The points are colored according to number density with yellower points indicating higher density

checks of uniformity. The simplest such check, shown in Figure 10 is plotting the ranks in a histogram, allowing for a diagnosis of auto-correlation effects in the posterior inference. From Figure 10, we can see that the uncertainties are slightly better calibrated following domain adaptation, in particular looking at e_{m1}, e_{m2} and y. Nevertheless, most parameters exhibit some deviation from uniformity. Histograms exhibiting a $\cup$-shaped distribution indicate an underestimation of posterior variance, while a peaked distribution indicates an overestimation. The variance in the ellipticity terms is therefore underestimated. Moreover, the right skewness of the θ_E posterior ranks indicates an overestimation of the posterior mean.

Similarly, we can check the empirical cumulative distribution function of the ranks under uniformity. As can be seen in Figure 11, the ranks distribution following domain adaptation can be said to be better described by a uniform distribution as the CDF of all parameters, except for θ_E, approach the line of uniformity.

As a quantitative check of uniformity, we perform a Kolmogorov-Smirnov test against a uniform distribution for each dimension of the parameter space. We also perform a classifier two-sample test (C2ST) comparing the rank ensemble with a target distribution. The C2ST relies on training a binary classifier to distinguish between two samples and returning accuracy scores from cross-validation[24]. Assuming both ensembles are drawn from the same distribution, the scores should not be better than chance, that is, 0.5. We perform the C2ST check-

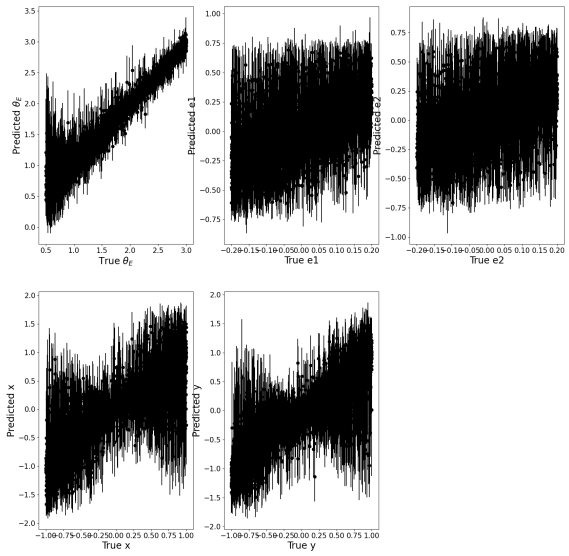

FIG. 6. True versus predicted parameter values for target domain with no domain adaptation.

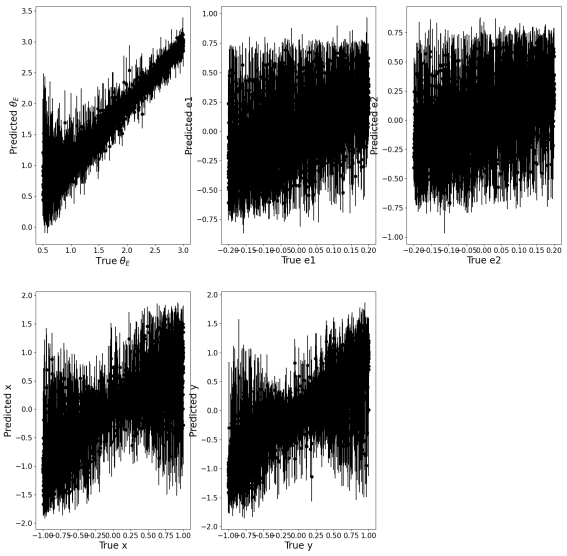

FIG. 7. True versus predicted parameter values for target domain with MMD at $\lambda = 0.02$ implemented.

metric	No DA	MMD; $\lambda = 0.02$	MMD; $\lambda = 9.95$
KS p-val	1.099×10^{-4}	7.840×10^{-3}	2.568×10^{-11}
C2ST - Ranks	0.5779	0.5786	0.5811
C2ST - DAP	0.5668	0.5621	0.5731

TABLE V. Caption

ing uniformity of the ranks as well as the similarity of the data-averaged posterior (DAP) to the prior. For a well-calibrated Bayesian analysis, we expect the DAP, or the average of the posterior expectation with respect to our generated data, to be distributed according to the prior.[23] . The results of these tests are summarized in Table V. Mirroring the qualitative checks, from the KS p-values, we can see that the ranks distributions are not uniform for any of the network runs. Moreover, the C2ST ranks test shows that the ranks are, on average, distributed less uniformly when domain adaptation is applied. That said, the DAP seems to better resemble the prior post-DA at $\lambda = 0.02$.

VII. DISCUSSION

In this paper, we attempt to implement domain adaptation into the `sbi` package using the Maximum Mean Discrepancy between the source and target domain distributions as an additional loss in the training procedure. We introduce this DA scheme into an automatic posterior transformation NPE procedure within the package. The MMD is introduced with a weighting parameter λ,

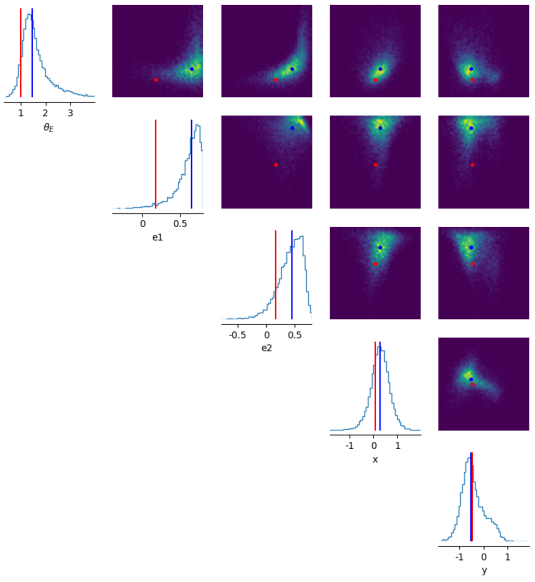

FIG. 8. Parameter posteriors for a randomly selected simulated galaxy-galaxy lens in the target domain with no domain adaptation. The red line and dot represent the true parameter value. In blue is the mean posterior value inferred by the network.

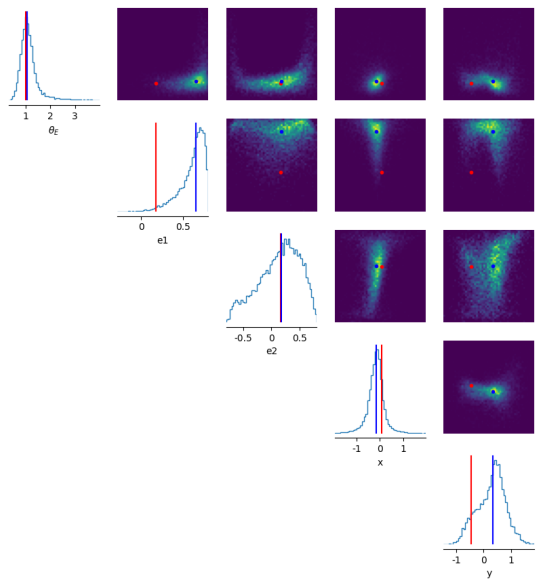

FIG. 9. Parameter posteriors for randomly selected simulated galaxy=galaxy lens in the target domain with MMD at $\lambda = 0.02$. The red line and dot represent the true parameter value. In blue is the mean posterior value inferred by the network.

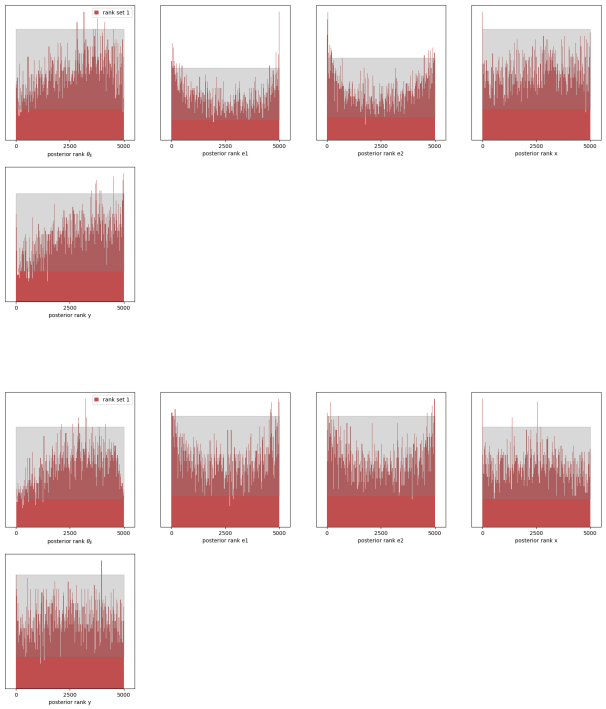

FIG. 10. Histograms of rank distributions of inferred parameters under true parameters for target domain with no domain adaptation (top) and MMD at $\lambda = 0.02$ (bottom). Gray bar represents the 99% confidence interval for a uniform distribution.

optimized using `optuna`, which found an optimal weight of 9.95. In addition, we test DA with $\lambda = 0.02$. We test the network on a sample of 400,000 simulated galaxy-galaxy lenses generated using the `deeplenstronmy` package. The lenses were divided into 200,000 source and 200,000 target domain lenses with target domain lenses generated with noise profiles derived from DES survey conditions. For each lens we attempt to fit 5 SIE parameters $\theta_E, e_{m1}, e_{m2}, x, y$.

Network performance exhibited marginal improvement following an MMD implementation with weighting of $\lambda = 0.02$. Given the small nature of the improvement, it is not possible to completely rule out random variation in network parameter weighting as the cause. That said, increasing the MMD weight to 9.95 caused a decrease in performance such that some variation can be attributed to the implementation of DA. Similarly, the network uncertainty was better calibrated post-DA, at least based on the KS test and the C2ST-DAP. The ranks distribution, however, deviated more from uniformity post-DA according to the C2ST-Ranks test. This is in tension with the qualitative checks on uniformity, which appear to show the ranks distribution across most parameters converging to a uniform distribution. Moreover, there were still significant deviations from uniformity in the ranks statistics, therefore limiting network viability. Nevertheless, baseline performance was adequate in recovering parameter values and from the χ^2 we can see that the true value often fell within the network variance. In particular, θ_E was consistently well inferred by the network with a target MSE of 1.108×10^{-2} without DA and 8.762×10^{-3} post-DA which was the lowest of all the parameters.

While present results show limited improvement post-DA, these do not necessarily indicate that this is a non-viable research avenue. It is important to note that performance on source and target domains is similar indicating that SBI is more robust against changes in the data. Despite this, applying domain adaptation still led to marginal network improvement. It therefore might prove fruitful to test the method on source and target domains which exhibit a greater degree of divergence. Moreover, there were several limitations which prevent the study from conclusively ruling out the viability of DA as applied to SBI. First, a limited number of network architectures were tested with little variation, mainly consisting of the number of convolutional layers introduced. Second, only the MMD weight underwent a hyperparameter tuning procedure, with other network parameters optimized only through trial and error. The hyperparameter tuning was moreover apparently unsuccessful, as the stated optimal weight did not yield the best network performance. Improvements in the hyperparameter tuning procedure could therefore yield better results. Beyond errors arising from procedure, the MMD is not

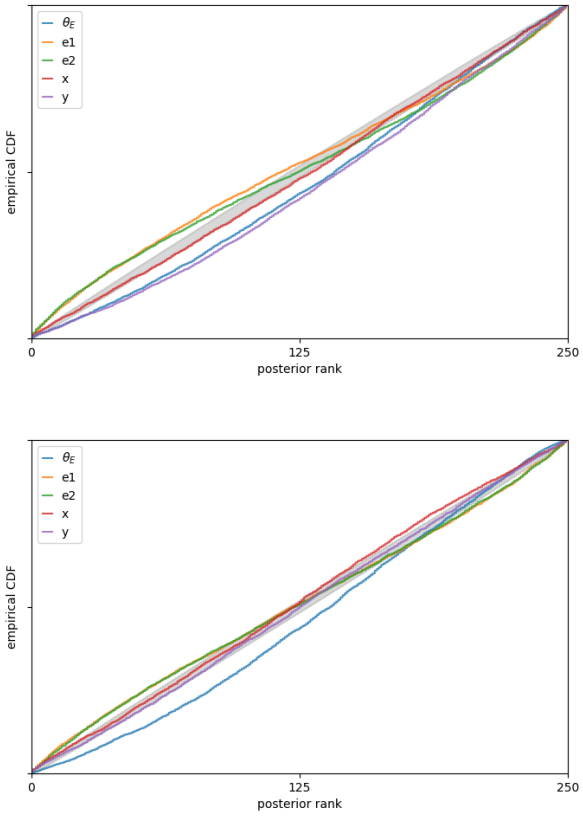

FIG. 11. Empirical cumulative density functions of rank distributions of inferred parameters under true parameters for target domain with no domain adaptation (top) and MMD at $\lambda = 0.02$ (bottom). Gray region represents the 99% confidence interval for a uniform distribution.

the only such divergence measure that can be used for this application, with other options available such as the correlation-alignment loss[25] and the KL-divergence[26].

VIII. ACKNOWLEDGMENTS

I would like to thank my advisors Dr. Aleksandra Ćiprijanović and Dr. Brian Nord for all their guidence and patience throughout the project.

This manuscript has been authored by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics.

This work was supported in part by the U.S. Department of Energy, Office of Science, Office of Workforce Development for Teachers and Scientists (WDTS) under the Science Undergraduate Laboratory Internships Program (SULI).

[1] P. Schneider, J. Ehlers, and E. E. Falco, *Gravitational Lenses* (1992).

[2] R. Legin, Y. Hezaveh, L. Perreault-Levasseur, and B. Wandelt, "A framework for obtaining accurate posteriors of strong gravitational lensing parameters with flexible priors and implicit likelihoods using density estimation," The Astrophysical Journal **943**, 4 (2023).

[3] K. Cranmer, J. Brehmer, and G. Louppe, "The frontier of simulation-based inference," Proceedings of the National Academy of Science **117**, 30055–30062 (2020), arXiv:1911.01429 [stat.ML].

[4] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-Adversarial Training of Neural Networks," arXiv e-prints , arXiv:1505.07818 (2015), arXiv:1505.07818 [stat.ML].

[5] A. Farahani, S. Voghoei, K. Rasheed, and H. R. Arabnia, "A Brief Review of Domain Adaptation," arXiv e-prints , arXiv:2010.03978 (2020), arXiv:2010.03978 [cs.LG].

[6] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," Neurocomputing **312**, 135–153 (2018).

[7] A. Tejero-Cantero, J. Boelts, M. Deistler, J.-M. Lueckmann, C. Durkan, P. J. Gonçalves, D. S. Greenberg, and J. H. Macke, "sbi: A toolkit for simulation-based inference," Journal of Open Source Software **5**, 2505 (2020).

[8] D. S. Greenberg, M. Nonnenmacher, and J. H. Macke, "Automatic Posterior Transformation for Likelihood-Free Inference," arXiv e-prints , arXiv:1905.07488 (2019), arXiv:1905.07488 [cs.LG].

[9] G. Papamakarios, T. Pavlakou, and I. Murray, "Masked Autoregressive Flow for Density Estimation," arXiv e-prints , arXiv:1705.07057 (2017), arXiv:1705.07057 [stat.ML].

[10] B. Uria, M. Côté, K. Gregor, I. Murray, and H. Larochelle, "Neural autoregressive distribution estimation," CoRR **abs/1605.02226** (2016), 1605.02226.

[11] D. Jimenez Rezende and S. Mohamed, "Variational Inference with Normalizing Flows," arXiv e-prints , arXiv:1505.05770 (2015), arXiv:1505.05770 [stat.ML].

[12] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," in *Advances in Neural Information Processing Systems*, Vol. 29, edited by D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (Curran Associates, Inc., 2016).

[13] X. Liu, C. Yoo, F. Xing, H. Oh, G. El Fakhri, J.-W. Kang, and J. Woo, "Deep Unsupervised Domain Adaptation: A Review of Recent Advances and Perspectives," arXiv e-prints , arXiv:2208.07422 (2022), arXiv:2208.07422 [cs.CV].

[14] Y. Zhang, Y. Zhang, Y. Wei, K. Bai, Y. Song, and Q. Yang, "Fisher Deep Domain Adaptation," arXiv e-prints , arXiv:2003.05636 (2020), arXiv:2003.05636 [cs.LG].

[15] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," Journal of Machine Learning Research **13**, 723–773 (2012).

[16] A. Ćiprijanović, D. Kafkes, K. Downey, S. Jenkins, G. N. Perdue, S. Madireddy, T. Johnston, G. F. Snyder, and B. Nord, "DeepMerge - II. Building robust deep learning algorithms for merging galaxy identification across domains," **506**, 677–691 (2021), arXiv:2103.01373 [astro-ph.IM].

[17] R. Morgan, B. Nord, S. Birrer, J. Y.-Y. Lin, and J. Poh, "deeplenstronomy: A dataset simulation package for strong gravitational lensing," Journal of Open Source Software **6**, 2854 (2021).

[18] S. Birrer and A. Amara, "lenstronomy: Multi-purpose gravitational lens modelling software package," Physics of the Dark Universe **22**, 189 – 201 (2018).

[19] T. M. C. Abbott, F. B. Abdalla, S. Allam, A. Amara, J. Annis, J. Asorey, S. Avila, O. Ballester, M. Banerji, W. Barkhouse, L. Baruah, M. Baumer, K. Bechtol, M. R. Becker, A. Benoit-Lévy, G. M. Bernstein, E. Bertin, J. Blazek, S. Bocquet, D. Brooks, D. Brout, E. Buckley-Geer, D. L. Burke, V. Busti, R. Campisano, L. Cardiel-Sas, A. C. Rosell, M. C. Kind, J. Carretero, F. J. Castander, R. Cawthon, C. Chang, X. Chen, C. Conselice, G. Costa, M. Crocce, C. E. Cunha, C. B. D'Andrea,

L. N. da Costa, R. Das, G. Daues, T. M. Davis, C. Davis, J. D. Vicente, D. L. DePoy, J. DeRose, S. Desai, H. T. Diehl, J. P. Dietrich, S. Dodelson, P. Doel, A. Drlica-Wagner, T. F. Eifler, A. E. Elliott, A. E. Evrard, A. Farahi, A. F. Neto, E. Fernandez, D. A. Finley, B. Flaugher, R. J. Foley, P. Fosalba, D. N. Friedel, J. Frieman, J. García-Bellido, E. Gaztanaga, D. W. Gerdes, T. Giannantonio, M. S. S. Gill, K. Glazebrook, D. A. Goldstein, M. Gower, D. Gruen, R. A. Gruendl, J. Gschwend, R. R. Gupta, G. Gutierrez, S. Hamilton, W. G. Hartley, S. R. Hinton, J. M. Hislop, D. Hollowood, K. Honscheid, B. Hoyle, D. Huterer, B. Jain, D. J. James, T. Jeltema, M. W. G. Johnson, M. D. Johnson, T. Kacprzak, S. Kent, G. Khullar, M. Klein, A. Kovacs, A. M. G. Koziol, E. Krause, A. Kremin, R. Kron, K. Kuehn, S. Kuhlmann, N. Kuropatkin, O. Lahav, J. Lasker, T. S. Li, R. T. Li, A. R. Liddle, M. Lima, H. Lin, P. López-Reyes, N. MacCrann, M. A. G. Maia, J. D. Maloney, M. Manera, M. March, J. Marriner, J. L. Marshall, P. Martini, T. McClintock, T. McKay, R. G. McMahon, P. Melchior, F. Menanteau, C. J. Miller, R. Miquel, J. J. Mohr, E. Morganson, J. Mould, E. Neilsen, R. C. Nichol, F. Nogueira, B. Nord, P. Nugent, L. Nunes, R. L. C. Ogando, L. Old, A. B. Pace, A. Palmese, F. Paz-Chinchón, H. V. Peiris, W. J. Percival, D. Petravick, A. A. Plazas, J. Poh, C. Pond, A. Porredon, A. Pujol, A. Refregier, K. Reil, P. M. Ricker, R. P. Rollins, A. K. Romer, A. Roodman, P. Rooney, A. J. Ross, E. S. Rykoff, M. Sako, M. L. Sanchez, E. Sanchez, B. Santiago, A. Saro, V. Scarpine, D. Scolnic, S. Serrano, I. Sevilla-Noarbe, E. Sheldon, N. Shipp, M. L. Silveira, M. Smith, R. C. Smith, J. A. Smith, M. Soares-Santos, F. Sobreira, J. Song, A. Stebbins, E. Suchyta, M. Sullivan, M. E. C. Swanson, G. Tarle, J. Thaler, D. Thomas, R. C. Thomas, M. A. Troxel, D. L. Tucker, V. Vikram, A. K. Vivas, A. R. Walker, R. H. Wechsler, J. Weller, W. Wester, R. C. Wolf, H. Wu, B. Yanny, A. Zenteno, Y. Zhang, J. Zuntz, D. Collaboration), S. Juneau, M. Fitzpatrick, R. Nikutta, D. Nidever, K. Olsen, A. Scott, and N. D. Lab), "The dark energy survey: Data release 1," The Astrophysical Journal Supplement Series **239**, 18 (2018).

[20] R. Kormann, P. Schneider, and M. Bartelmann, "Isothermal elliptical gravitational lens models." **284**, 285–299 (1994).

[21] J. L. Sérsic, "Influence of the atmospheric and instrumental dispersion on the brightness distribution in a galaxy," Boletin de la Asociacion Argentina de Astronomia La Plata Argentina **6**, 41–43 (1963).

[22] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," arXiv e-prints , arXiv:1907.10902 (2019), arXiv:1907.10902 [cs.LG].

[23] S. Talts, M. Betancourt, D. Simpson, A. Vehtari, and A. Gelman, "Validating Bayesian Inference Algorithms with Simulation-Based Calibration," arXiv e-prints , arXiv:1804.06788 (2018), arXiv:1804.06788 [stat.ME].

[24] D. Lopez-Paz and M. Oquab, "Revisiting Classifier Two-Sample Tests," arXiv e-prints , arXiv:1610.06545 (2016), arXiv:1610.06545 [stat.ML].

[25] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Computer Vision – ECCV 2016 Workshops*, edited by G. Hua and H. Jégou (Springer International Publishing, Cham, 2016) pp. 443–450.

[26] M. Sugiyama, S. Nakajima, H. Kashima, P. von Bünau, and M. Kawanabe, "Direct importance estimation with model selection and its application to covariate shift adaptation," in *Neural Information Processing Systems* (2007).