A neural simulation-based inference approach for characterizing the Galactic Center γ-ray excess

Siddharth Mishra-Sharma[1,*] and Kyle Cranmer[1,2,†]

[1] *Center for Cosmology and Particle Physics, Department of Physics, New York University, New York, NY 10003, USA*
[2] *Center for Data Science, New York University, 60 Fifth Ave, New York, NY 10011, USA*

(Dated: May 20, 2021)

The nature of the *Fermi* γ-ray Galactic Center Excess (GCE) has remained a persistent mystery for over a decade. Although the excess is broadly compatible with emission expected from annihilating dark matter, an explanation in terms of a population of unresolved point sources remains viable. The effort to uncover the origin of the GCE is hampered in particular by a poor understanding of the diffuse emission of Galactic origin, which can lead to spurious residuals that make it difficult to robustly differentiate smooth emission, as expected due to a dark matter origin, from more "clumpy" emission expected from relatively bright, unresolved point sources. We use neural simulation-based inference methods, in particular conditional density estimation with normalizing flows, in order to extract more information from γ-ray maps of the Galactic Center with the aim of characterizing the contribution of unresolved point sources to the GCE. We do not find a substantial preference for emission of point source origin within our framework.

I. INTRODUCTION

Dark matter (DM) represents one of the major unsolved problems in particle physics and cosmology today. The traditional Weakly-Interacting Dark Matter (WIMP) paradigm envisions production of dark matter in the early Universe through freeze-out of dark matter particles weakly coupled to the Standard Model (SM). In this scenario, one of the most promising methods of detecting a dark matter signal would be through an observation of excess γ-ray photons from DM-rich regions of the sky, produced through the cascade of SM particles resulting from DM annihilation.

The *Fermi* γ-ray Galactic Center Excess (GCE), first identified over a decade ago using data from the *Fermi* Large Area Telescope (LAT), is an excess of photons in the Galactic Center with properties—such as energy spectrum and spatial morphology—broadly compatible with that expected due to annihilating DM. The nature of the GCE remains contentious however, with competing explanations in terms of a population of unresolved astrophysical point sources (PSs) remaining viable. For example, analyses of the spatial morphology of the excess have shown it to be more compatible with that tracing a stellar bulge distribution in the Galactic Center. Furthermore, analyses based on the statistics of photons in the Galactic Center have shown the data to prefer a point source origin of the excess. Recent analyses have however pointed out the potential of unknown systematics—such as the poorly understood morphology of the diffuse foreground emission—to bias these conclusions.

Due to the high-dimensional nature of γ-ray data, traditional analyses techniques have resorted to a hand-crafted reduction of the photon map into summaries such as the probability distribution of photon counts or a wavelet decomposition of the photon map. Novel analysis techniques that can extract more information from the photon maps could do a better job of hedging against unknown systematics in the data compared to traditional analyses based on specific statistics of the map. Machine learning methods in particular have demonstrated promise for general analyses of γ-ray data and in particular focusing on understanding the nature of the GCE. In this paper, we leverage recent developments in the field of simulation-based inference (SBI) in order to weigh in on the nature of the GCE.

This paper is organized as follows. In Sec. II we describe the various components of our analysis method based on simulation-based inference. In Sec. III we describe our simulation pipeline and verify our analysis on simulated data. Section IV presents results on *Fermi* γ-ray data. In Sec. V we explore the susceptibility of the analysis to mismodeling of the background and signal templates, and present systematic variations on our analysis in Sec. VI. We conclude in Sec. VII.

II. METHODOLOGY

We begin by describing our analysis methodology, going over in turn the general principles behind simulation-based inference, posterior estimation using normalizing flows, and learning representative summary statistics from high-dimensional γ-ray maps with neural networks.

A. Simulation-based inference

Of central interest in parameter estimation is often the probability distribution of a set of parameters of interest

* sm8383@nyu.edu; ORCID: 0000-0001-9088-7845
† kyle.cranmer@nyu.edu; ORCID: 0000-0002-5769-7094

θ given some data x—the posterior distribution $p(\theta \mid x)$. Bayes' theorem can be used to obtain the posterior as $p(\theta \mid x) = p(\theta)\, p(x \mid \theta)/\mathcal{Z}$, where $p(x \mid \theta)$ is the likelihood and $\mathcal{Z}$ is the Bayesian evidence. In practice, parameters other than θ—latent variables z—are often involved in the data-generation process, and computing the likelihood involves marginalizing over the latent space, $p(x \mid \theta) = \int \mathrm{d}z\, p(x \mid \theta, z)$. In typical problems of interest, the high dimensionality of the latent space often means that this integral is intractable, necessitating simplifications in statistical treatment as well as theoretical modeling.

Simulation-based inference (SBI) refers to a class of methods for performing inference when the data-generating process does not have a tractable likelihood. In this setting, a model is defined through a simulator as a probabilistic program, often knows as a forward model. Samples x from the simulator then implicitly define a likelihood, $x \sim p(x \mid \theta)$. In the simplest realizations of SBI, samples x' generated from a given prior proposal distribution $p(\theta)$ can be compared to a given dataset of interest x, with the approximate posterior defined by samples that most closely approximate x according to some distance metric. Such methods—usually grouped under the umbrella of Approximate Bayesian Computation (ABC)—are not uncommon in astrophysics and cosmology. Nevertheless, they suffer from several downsides. The curse of dimensionality usually necessitates reduction of data to representative lower-dimensional summary statistics $s(x)$, resulting in loss of information. A notion and measure of distance between summaries from the implicit model and those derived from the dataset of interest is necessary, leading to inexact inference. Additionally, the ABC analysis must be performed anew for each new target dataset.

Recent methods have leveraged advancements in machine learning, in particular the ability of neural networks to extract useful features from high-dimensional data and flexibly approximate functions and distributions, in order to address these issues, enabling new ways of performing inference on complex models defined through simulations. See Ref. [1] for a review of recent developments.

B. Conditional density estimation with normalizing flows

In this paper, we approximate the joint posterior $p(\theta \mid x)$ through a parameterized distribution $\hat{p}_\phi(\theta \mid s)$ conditioned on *learned* summaries $s = s(x)$ from the simulated samples x. This class of simulation-based inference techniques, known as conditional neural density estimation, directly models the posterior distribution given a set of samples drawn from simulator according to some prior proposal distribution $p(\theta)$.

We employ normalizing flows, which provide an efficient way of constructing flexible probability distributions. Normalizing flows model the conditional distribution $\hat{p}_\phi(\theta \mid s)$ as a series of invertible transformations, denoted f and having a tractable inverse and Jacobian, from a base distribution $\pi_z z$, chosen here to be a standard Gaussian, to the target distribution:

$$\hat{p}(\theta \mid x) = \pi_z \left(f^{-1}(\theta)\right) \left|\det \left(\frac{\partial f^{-1}}{\partial \theta}\right)\right| \tag{1}$$

Specifically, we use Masked Autoregressive Flows (MAFs) for posterior estimation. The MAF is built upon blocks of affine transformations where the scaling and shifting factors for each dimension are computed with a Masked Autoencoder for Distribution Estimation (MADE). For a single block, the transformation is expressed as

$$z_i = (\theta_i - \mu_i) \cdot \exp\left(-\alpha_i\right) \tag{2}$$

where $\mu_i = f_{\mu_i}\left(\theta_{1:i-1}; x\right)$ and $\alpha_i = f_{\alpha_i}\left(\theta_{1:i-1}; x\right)$ are scaling and shift factors modeled by a MADE according to the autoregressive condition. This allows for an analytically tractable determinant,

$$\left|\det \left(\frac{\partial f^{-1}}{\partial \theta}\right)\right| = \exp\left(-\sum_i \alpha_i\right) \tag{3}$$

and forward sampling according to Eq. 2. Multiple transformations can be chained together as $f = f_1 \circ f_2 \circ \ldots \circ f_K$ in order to model more expressive posteriors,

$$\hat{p}(\theta \mid x) = \pi_z \left(f^{-1}(\theta)\right) \prod_{i=1}^{K} \left|\det \left(\frac{\partial f_i^{-1}}{\partial z_{i-1}}\right)\right| \tag{4}$$

The log-probability of the posterior can be computed using Eq. 3:

$$\log \hat{p}(\theta \mid x) = \log\left[\pi_z \left(f^{-1}(\boldsymbol{\theta})\right)\right] - \sum_{i=1}^{K} \sum_{j=1}^{N} \alpha_j^i, \tag{5}$$

which acts as the optimization objective.

C. Learning summary statistics with neural networks

The curse of dimensionality makes it computationally prohibitive to condition the density estimation task on the the raw dataset x *i.e.*, the γ-ray pixel counts map in the region of interest (ROI). Instead, we use a neural network to learn lower-dimensional summary features from the map, $s = s_\varphi(x)$. The *DeepSphere* [2, 3] architecture, with a configuration similar to that introduced and employed in Ref. [4], is used here and summarized briefly here. For more details on the summary extractor network, see Ref. [4].

D. Simulations and datasets

We use the datasets and templates from Ref. [5] (packaged with Ref. [6]) to create the simulated maps. The data and templates used correspond to 413 weeks of *Fermi*-LAT Pass 8 data collected between August 4, 2008 and July 7, 2016. The top quarter of photons in the energy range 2–20 GeV by quality of PSF reconstruction (corresponding to PSF3 event type) in the event class `ULTRACLEANVETO` are used. The recommended quality cuts are applied, corresponding to zenith angle less than 90°, `LAT_CONFIG` = 1, and `DATA_QUAL` > 0.1.[1] The maps are spatially binned using HEALPix [7] with `nside` = 128.

The simulated data maps are a combination of smooth (*i.e.*, Poissonian) and PS contributions. Each PS population is completely specified by its spatial and source-count distribution. Photon counts from a generated PS population are put down on a map according to the *Fermi* PSF at 2 GeV, modeled as a King function, using the algorithm implemented in the code package `NPTFit-Sim` [8]. PS populations spatially correlated with t

In addition to the PS emission from subhalos and blazars, we also account for Poissonian astrophysical emission in the simulated maps. These contributions include: *(i)* the Galactic diffuse foreground emission, described by `Model O` from Ref. [9], *(ii)* isotropic emission, *(iii)* resolved PSs from the 3FGL catalog, and *(iv)* emission from the *Fermi* bubbles [10]. The latter three templates are obtained from Ref. [5]. The final maps are obtained by combining a Poisson-fluctuated realization of the combined astrophysical templates with the PS maps. We mask resolved PSs from the 3FGL catalog at a containment radius of 0.8°.

E. Optimization and training

10^6 training samples are generated. Models are trained with batch size 128 using the `AdamW` optimizer with initial learning rate 10^{-3} and weight decay 10^{-5}. Cosine annealing is used to decay the learning rate, training for up to 100 epochs with early stopping if the validation loss has not improved after 10 epochs.

III. TESTS ON SIMULATED DATA

We begin by validating our trained model on simulated data. We create simulated datasets using the best-fit astrophysical model obtained from a non-Poissonian fit in our fiducal ROI, and test the ability of our model to infer the presence of either DM-like or PS-like signals on top of this background.

Figure 1 shows results on maps where the GCE consists of purely DM-like emission. The left column shows the middle-68/95% containment of the point-wise posterior on the source-count distributions of GCE- and disk-correlated point source emission in red and blue, respectively. The middle column shows the posterior on emission of all components in our model. The right- column shows the fraction of DM- and PS-like emission in proportion to the total inferred flux in the ROI. The true underlying quantities from which the data was generated are represented by dotted lines. We see that, in all cases shown, we are successfully able to infer the presence of DM-like emission. Some PS-like emission is inferred in most cases as well, due to a combination of degeneracy with both disk-correlated and DM-like flux. The overall flux of all components corresponds well to their true underlying values.

Figure 1 shows the corresponding results for simulated data containing PS-like emission. We see that PS-like emission is successfully inferred in each case. Furthermore, the model is able to characterize the source-count of PSs through the source count distribution. Some degeneracy with disk-like source is seen.

IV. RESULTS ON *FERMI* DATA

We finally apply our formalism to real *Fermi* data. The results are shown in Fig. 3. The top panel shows results from this work, while the bottom row shows results from an NPTF analysis. No strong preference for either PS-like or DM-like emission is seen, contrary to the NPTF analysis.

A. Signal injection test

Figure 4 shows the results of injecting a DM signal directly onto the *Fermi* data. We see that the injected DM signal is successfully recovered and not misattributed to PS-like emission.

V. SUSCEPTIBILITY TO MISMODELING

The key challenge associated with Galactic Center analyses is that associated with effects of mismodeled signal and background templates. In this section we assess the susceptibility of our simulation-based inference pipeline towards these systematics, discussing foreground and signal modeling in turn.

[1] https://fermi.gsfc.nasa.gov/ssc/data/analysis/documentation/Cicerone/Cicerone_Data_Exploration/Data_preparation.html

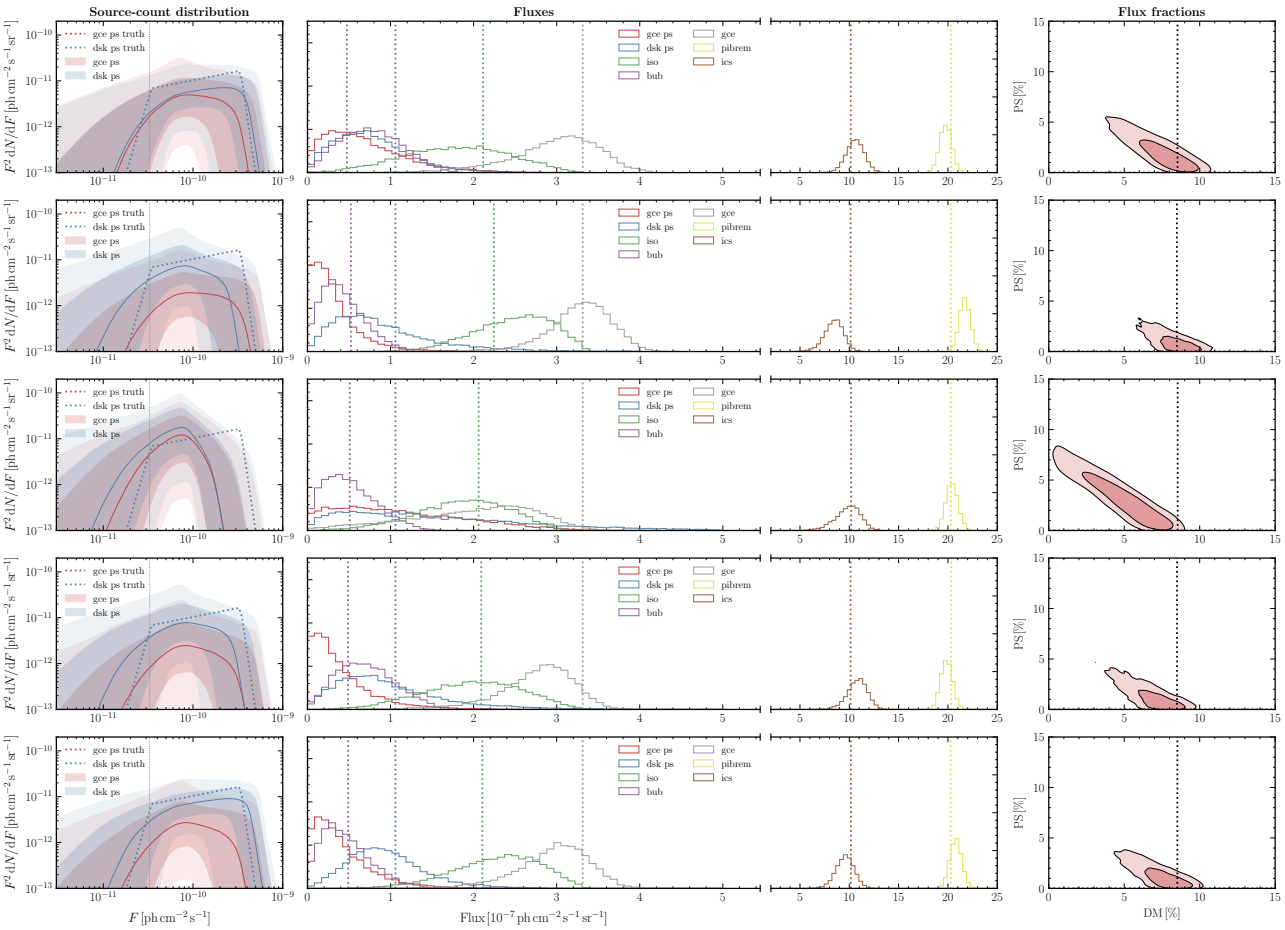

FIG. 1. Results on simulated data where the GCE consists of purely DM-like emission.

A. Foreground mismodeling

B. Signal mismodeling

VI. SYSTEMATIC VARIATIONS ON ANALYSIS

- Different diffuse models (Models A, B, F, p6v11, mismodeling, GP data-driven)
- Different ROIs (15, 20, 25)
- Varying inner slope of NFW profile
- Different disk templates

VII. CONCLUSIONS

The code used to obtain in this paper is available at https://github.com/smsharma/sbi-fermi.

ACKNOWLEDGMENTS

We thank Johann Brehmer, Florian List, Nick Rodd, and Tracy Slatyer for helpful conversations. KC is partially supported by NSF awards ACI-1450310, OAC-1836650, and OAC-1841471, the NSF grant PHY-1505463, and the Moore-Sloan Data Science Environment at NYU. SM is supported by the NSF CAREER grant PHY-1554858, NSF grants PHY-1620727 and PHY-1915409, and the Simons Foundation. This work made use of the NYU IT High Performance Computing resources, services, and staff expertise. This research has made use of NASA's Astrophysics Data System. This research made use of the `astropy` [11, 12], `dynesty` [13], `IPython` [14], Jupyter [15], `matplotlib` [16], `mlflow`, `NPTFit` [6], `NumPy` [17], `PyTorch` [18], `PyTorch Geometric` [19], `PyTorch Lightning` [20], `seaborn` [21], `pandas` [22], `sbi` [23], `scikit-learn` [24], `SciPy` [25], and `tqdm` [26] software packages. We acknowledge the use of the code repository associated with Ref. [4], in particular the as-

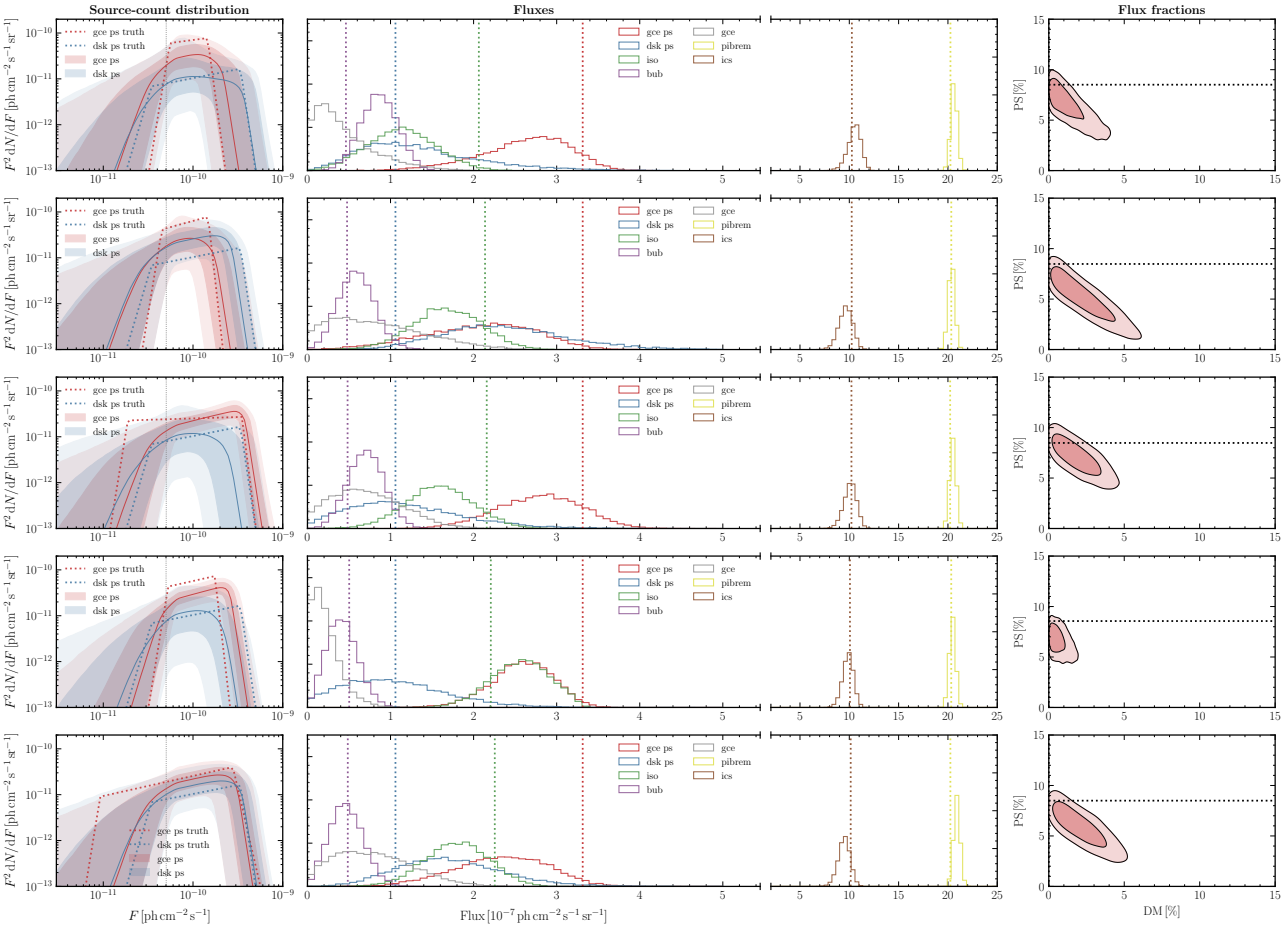

FIG. 2. Results on simulated data where the GCE consists of purely PS-like emission.

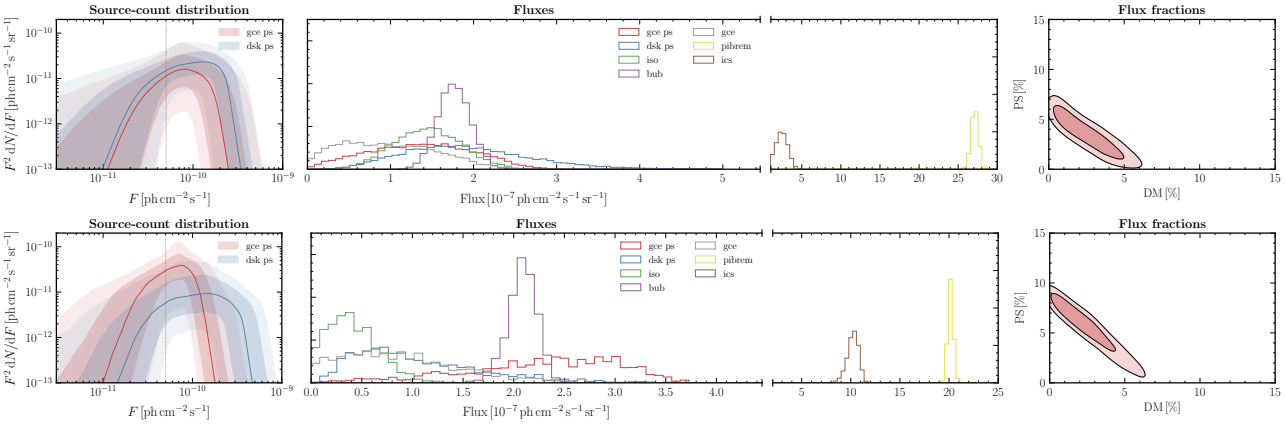

FIG. 3. Fiducial results on data.

sociated data products and templates.[2]

[2] https://github.com/FloList/GCE_NN

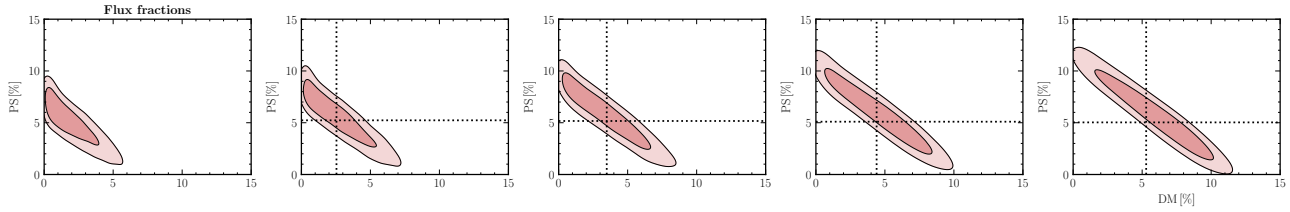

FIG. 4. Signal injection on data.

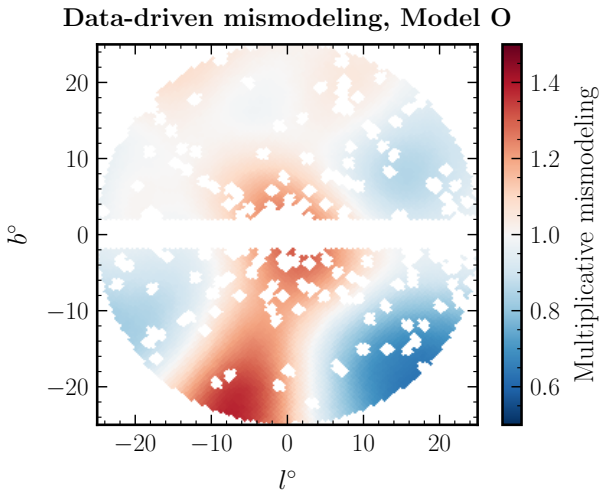

FIG. 5. The median Gaussian process description of multiplicative mismodeling associated with diffuse foreground Model O when applied to the real *Fermi* data.

[1] K. Cranmer, J. Brehmer, and G. Louppe, Proceedings of the National Academy of Sciences **117**, 30055 (2020).

[2] M. Defferrard, M. Milani, F. Gusset, and N. Perraudin, arXiv preprint arXiv:2012.15000 (2020).

[3] N. Perraudin, M. Defferrard, T. Kacprzak, and R. Sgier, Astron. Comput. **27**, 130 (2019), arXiv:1810.12186 [astro-ph.CO].

[4] F. List, N. L. Rodd, G. F. Lewis, and I. Bhat, Phys. Rev. Lett. **125**, 241102 (2020), arXiv:2006.12504 [astro-ph.HE].

[5] S. Mishra-Sharma, N. L. Rodd, and B. R. Safdi, "Supplementary material for NPTFit," (2016).

[6] S. Mishra-Sharma, N. L. Rodd, and B. R. Safdi, Astron. J. **153**, 253 (2017), arXiv:1612.03173 [astro-ph.HE].

[7] K. M. Gorski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelman, Astrophys. J. **622**, 759 (2005), arXiv:astro-ph/0409513.

[8] N. L. Rodd and M. W. Toomey, *NPTFit-Sim* (2017).

[9] M. Buschmann, N. L. Rodd, B. R. Safdi, L. J. Chang, S. Mishra-Sharma, M. Lisanti, and O. Macias, Phys. Rev. D **102**, 023023 (2020), arXiv:2002.12373 [astro-ph.HE].

[10] M. Su, T. R. Slatyer, and D. P. Finkbeiner, Astrophys. J. **724**, 1044 (2010), arXiv:1005.5480 [astro-ph.HE].

[11] A. M. Price-Whelan *et al.*, Astron. J. **156**, 123 (2018), arXiv:1801.02634.

[12] T. P. Robitaille *et al.* (Astropy), Astron. Astrophys. **558**, A33 (2013), arXiv:1307.6212 [astro-ph.IM].

[13] J. S. Speagle, Monthly Notices of the Royal Astronomical Society **493**, 3132 (2020).

[14] F. Perez and B. E. Granger, Computing in Science and Engineering **9**, 21 (2007).

[15] T. Kluyver *et al.*, in *ELPUB* (2016).

[16] J. D. Hunter, Computing In Science & Engineering **9**, 90 (2007).

[17] S. van der Walt, S. C. Colbert, and G. Varoquaux, Computing in Science and Engineering **13**, 22 (2011), arXiv:1102.1523 [cs.MS].

[18] A. Paszke *et al.*, in *Advances in Neural Information Processing Systems 32*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Curran Associates, Inc., 2019) pp. 8024–8035.

[19] M. Fey and J. E. Lenssen, in *ICLR Workshop on Representation Learning on Graphs and Manifolds* (2019).

[20] W. Falcon *et al.*, "Pytorchlightning/pytorch-lightning: 0.7.6 release," (2020).

[21] M. Waskom *et al.*, "mwaskom/seaborn: v0.8.1 (september 2017)," (2017).

[22] W. McKinney, in *Proceedings of the 9th Python in Science Conference*, edited by S. van der Walt and J. Millman (2010) pp. 51 – 56.

[23] A. Tejero-Cantero *et al.*, Journal of Open Source Software **5**, 2505 (2020).

[24] F. Pedregosa *et al.*, Journal of Machine Learning Research **12**, 2825 (2011).

[25] P. Virtanen *et al.*, Nature Methods (2020), https://doi.org/10.1038/s41592-019-0686-2.

[26] C. O. da Costa-Luis, JOSS **4**, 1277 (2019).