

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: April 22, 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 its origin is hampered in particular by the mischaracterization of the diffuse emission of Galactic origin, which can lead to spurious residuals that make it difficult to robustly differentiate smooth emission, as expected of dark matter origin, from more clumpy emission sources from relatively bright, unresolved point sources. We use neural simulation-based inference methods, in particular conditional density estimation, 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.

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 excess of γ -ray photons from DM-rich regions of the sky, produced through the cascade of SM particles produced as a result of 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—roughly compatible with that expected due to annihilation DM. The nature of the GCE remains contentious—although its properties are similar to that expected from DM annihilation, competing explanations in terms of a population of unresolved astrophysical point sources (PSs) remain viable. Although analyses based on the statistics of photon counts in the Galactic Center have shown the data to prefer a point source origin of the excess, recent analyses have pointed out the potential of unknown systematics in the data—such as the morphology of the diffuse foreground emission—to bias the conclusions of these results. Furthermore, 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.

Novel analysis techniques that can extract more in-

formation from the high-dimensional photon count maps could do a better job of “hedging” against unknown systematics in the data compared to traditional analyses based on summary statistics like the 1-point PDF. 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 θ given some data x —the posterior distribution $p(\theta | x)$. Bayes’ theorem can be used to obtain the posterior as $p(\theta | x) = p(\theta) p(x | \theta) / \mathcal{Z}$, where $p(x | \theta)$ is the likelihood and \mathcal{Z} is the Bayesian evidence. In practice, parameters other than θ —latent variables z —are often in-

* sm8383@nyu.edu; ORCID: 0000-0001-9088-7845

† kyle.cranmer@nyu.edu; ORCID: 0000-0002-5769-7094

volved in the data-generation process, and computing the likelihood involves marginalizing over the latent space, $p(x | \theta) = \int dz p(x | \theta, z)$. 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 known as a forward model. Samples x from the simulator then implicitly define a likelihood, $x \sim p(x | \theta)$. 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 . Such methods—usually grouped under the umbrella term 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.

Recently, methods leveraging advancements in machine learning and probabilistic programming have been developed in order to address these issues, enabling new ways of performing inference on complex models defined through simulations. See Ref. for a review of recent developments.

B. Conditional density estimation with normalizing flows

In this paper, we approximate the joint posterior $p(\theta | x)$ through a parameterized distribution $\hat{p}_\phi(\theta | s)$ conditioned on summaries $s = s(x)$ from the 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 a general way of constructing flexible probability distributions, for posterior estimation. Specifically, we use Masked Au-

to-regressive Flows (MAFs).

C. Learning summary statistics with neural networks

The curse of dimensionality makes it computationally prohibitive to condition the density estimation task on 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, $s = s_\phi(x)$. The *DeepSphere* architecture, with a configuration similar to that introduced and employed in Ref., is used here and summarized briefly here. For more details on the summary extractor network, see Ref.

D.

III. TEST ON SIMULATED DATA

IV. RESULTS ON *FERMI* DATA

A. Signal injection test

V. SUSCEPTIBILITY TO MISMODELING

A. Foreground mismodeling

B. Signal mismodeling

VI. SYSTEMATIC VARIATIONS ON ANALYSIS

VII. CONCLUSIONS

ACKNOWLEDGMENTS

We thank...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* [1, 2], *dynesty*, *IPython* [3], *Jupyter* [4], *matplotlib* [5], *mlflow*, *NPTFit*, *NumPy* [6], *PyTorch*, *PyTorch Geometric*, *PyTorch Lightning*, *seaborn* [7], *pandas* [8], *sbi*, *scikit-learn*, *SciPy* [9], and *tqdm* [10] software packages.

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

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

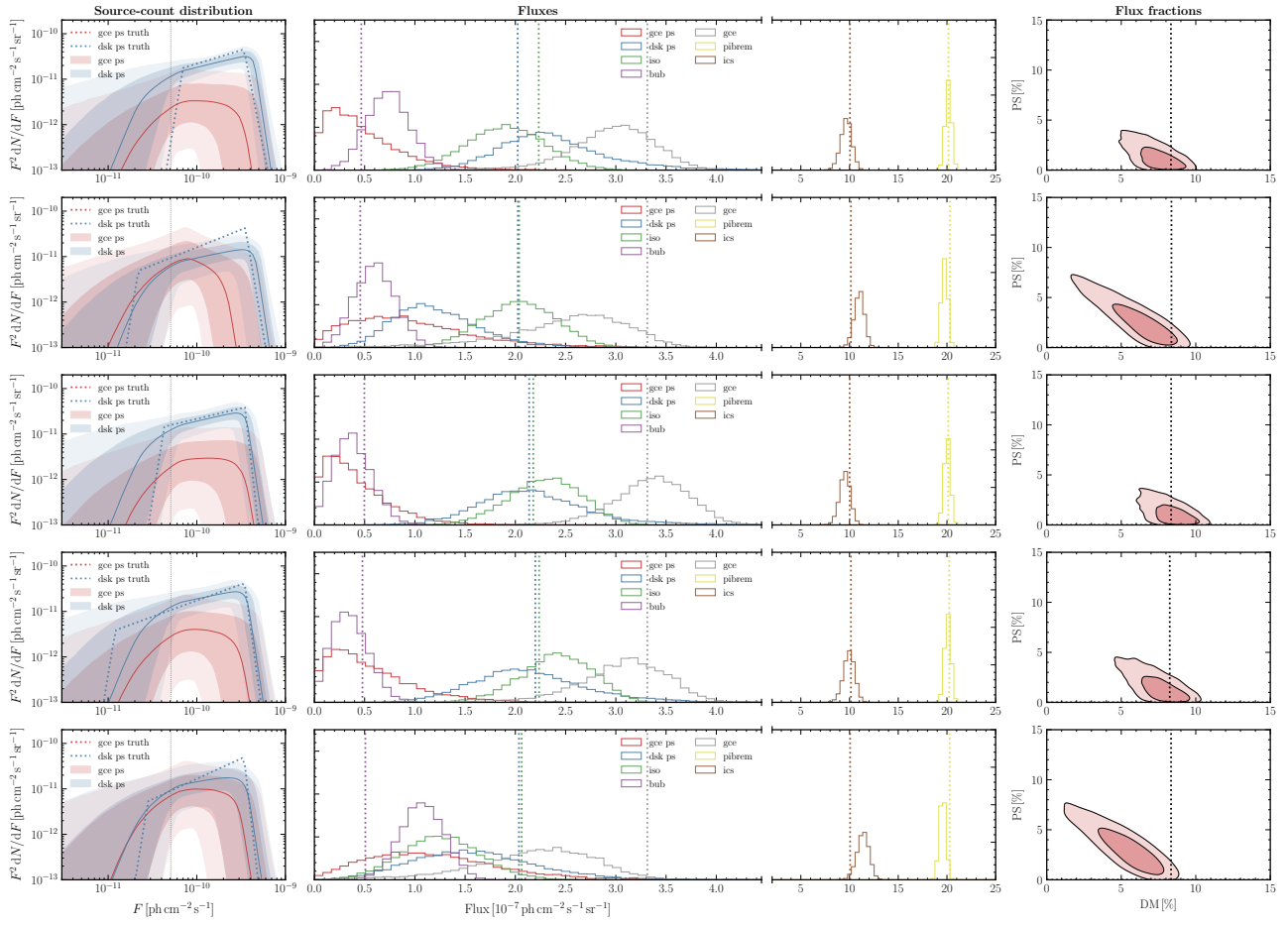


FIG. 1. Results on simulated data where the GCE consists of purely PS-like emission.

- [3] F. Perez and B. E. Granger, *Computing in Science and Engineering* **9**, 21 (2007).
- [4] T. Kluyver *et al.*, in *ELPUB* (2016).
- [5] J. D. Hunter, *Computing In Science & Engineering* **9**, 90 (2007).
- [6] S. van der Walt, S. C. Colbert, and G. Varoquaux, *Computing in Science and Engineering* **13**, 22 (2011), [arXiv:1102.1523 \[cs.MS\]](https://arxiv.org/abs/1102.1523).
- [7] M. Waskom *et al.*, “mwaskom/seaborn: v0.8.1 (september 2017),” (2017).
- [8] W. McKinney, in *Proceedings of the 9th Python in Science Conference*, edited by S. van der Walt and J. Millman (2010) pp. 51 – 56.
- [9] P. Virtanen *et al.*, *Nature Methods* (2020), <https://doi.org/10.1038/s41592-019-0686-2>.
- [10] C. O. da Costa-Luis, *JOSS* **4**, 1277 (2019).

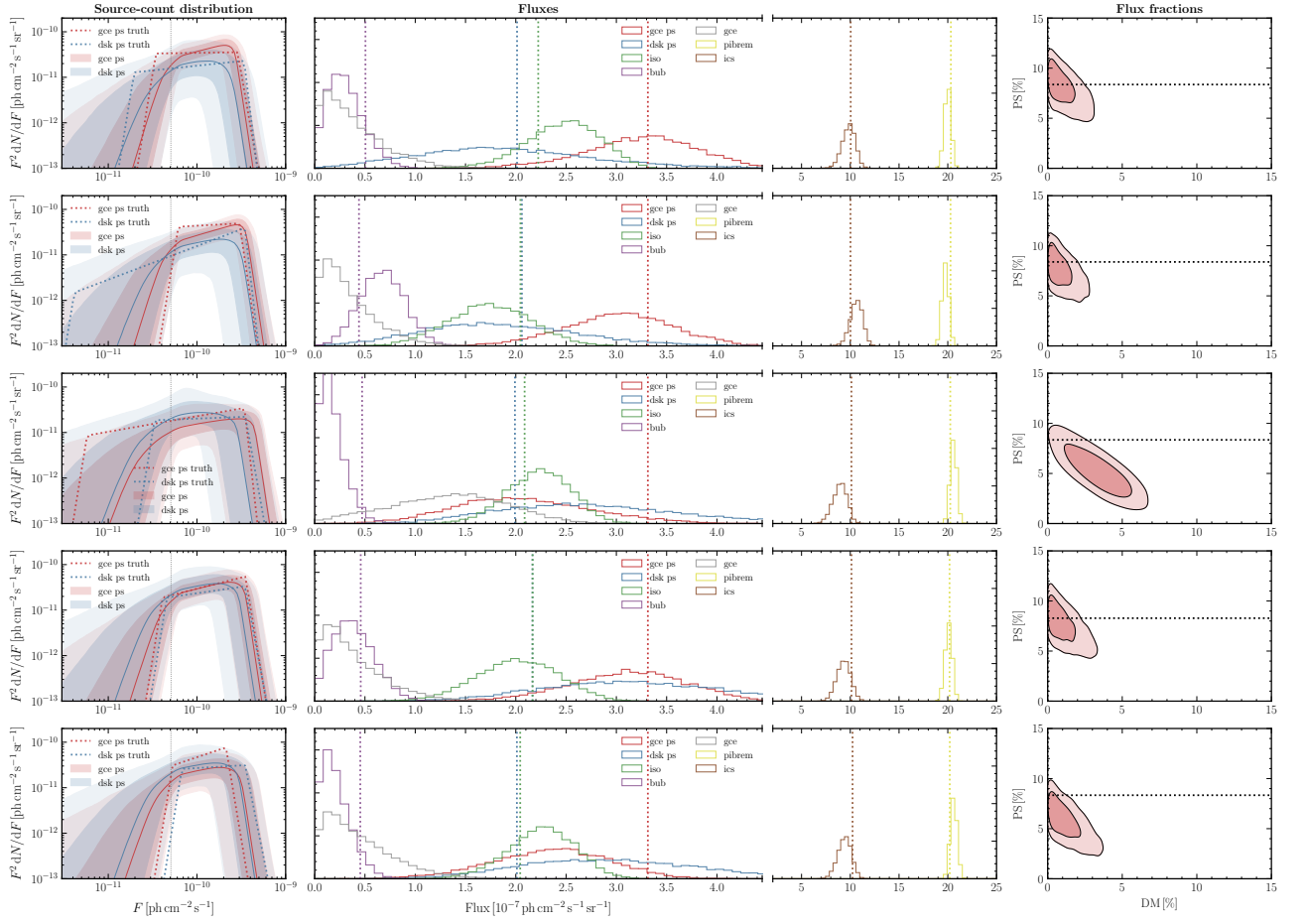


FIG. 2. Results on simulated data where the GCE consists of purely DM-like emission.

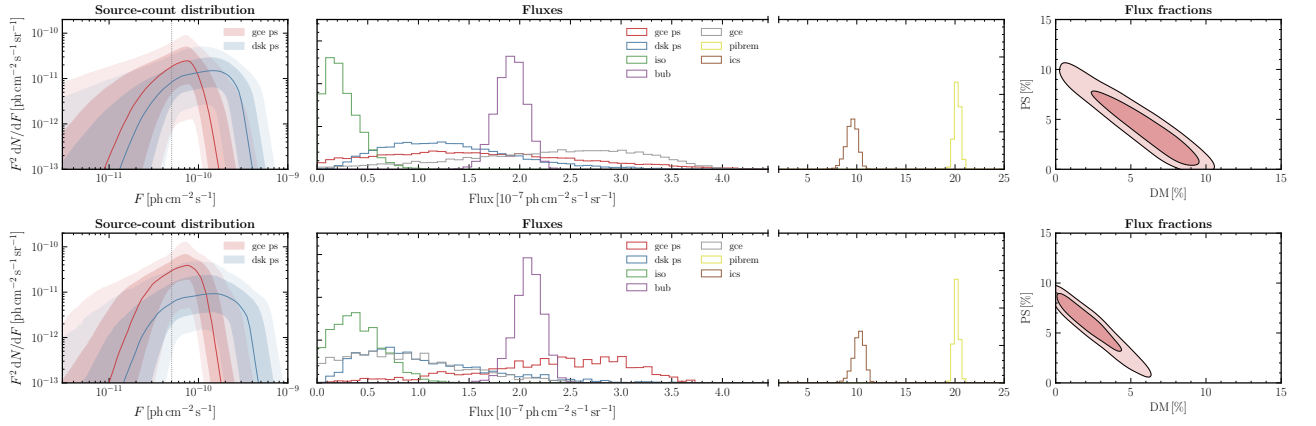


FIG. 3. Fiducial results on data.

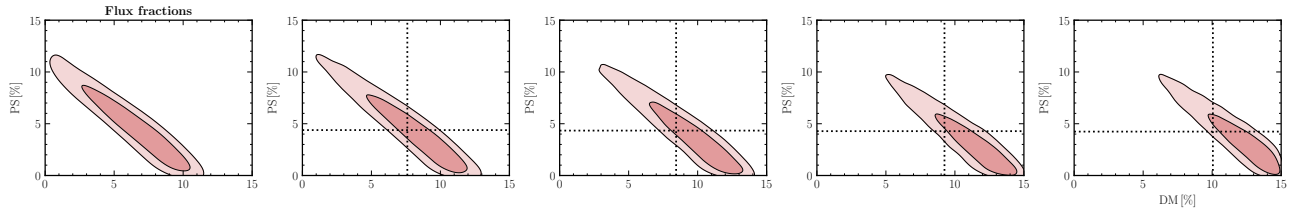


FIG. 4. Signal injection on data.