ENFORCING PROBABILISTIC CONSTRAINTS WITH DATA CONSISTENT INVERSION AND NORMALIZING FLOWS

Anonymous authors
Paper under double-blind review

ABSTRACT

Flow-based models such as Normalizing Flows (NFs) have become a powerful framework for learning and sampling from complex probability distributions. While highly expressive flow architectures can accurately approximate target distributions, they often introduce unnecessary computational cost due to overparameterization, which is particularly acute when only localized distributional changes must be incorporated into an existing stochastic model. In this work, we introduce a Data-Consistent Inversion (DCI)-based framework to rebase input distributions that allows for (1) systematic refinement of NFs to achieve desired distributional properties while avoiding excessive overparameterization, and (2) adapting pre-trained stochastic models. Using the iterative variant of DCI (iDCI), we construct a reweighted sample set from a NF base that is subsequently used to learn an updated base distribution whose push-forward constraints are satisfied by design. This enables the sequential refinement of coarse NF models, allowing them to approach a target distribution to a prescribed tolerance without retraining from scratch. Furthermore, the proposed approach provides a mechanism for adapting pre-trained stochastic models to distribution drift, offering a new perspective on domain adaptation through distribution-consistent updates. Numerical experiments demonstrate that the proposed methodology efficiently improves model fidelity while reducing the burden associated with full retraining, eliminating the need to retain or reuse the original target samples employed during pre-training, i.e., model improvements can be achieved using only the drifted observations and the pre-trained model itself.

1 INTRODUCTION

Stochastic inverse problems (SIPs) arise in many scientific and engineering applications where uncertainty in model inputs must be inferred from uncertainty observed in model outputs. Data Consistent Inversion (DCI) provides a measure-theoretic framework for solving such problems by constructing an updated probability measure whose push-forward through a model exactly matches an observed probability measure (Butler et al., 2018; 2020). Unlike Bayesian inference, which updates uncertainty through conditional probabilities (Kaipio & Somersalo, 2005; Tarantola, 2005; Stuart, 2010; Dashti & Stuart, 2017), DCI directly enforces consistency between predicted and observed distributions. The practical application of density-based DCI, however, relies on accurate estimation and representation of probability densities. While the updated density admits a simple multiplicative form, the required predicted and observed densities are rarely available in closed form and must instead be estimated from finite samples introducing challenges in high-dimensional settings where classical density estimators may become computationally expensive or statistically unreliable. To mitigate these difficulties, Iterative Data Consistent Inversion (iDCI) was recently introduced as a framework for solving generalized stochastic inverse problems using multiple lower-dimensional observational constraints (Jiang et al., 2026). By iteratively imposing push-forward marginal constraints, iDCI avoids the need to estimate a full observed joint distribution. However, the resulting solution is generally an information projection (I-projection) onto the constraint set and therefore need not coincide with the true underlying joint distribution. In parallel, Normalizing Flows (NFs) have emerged as powerful generative models capable of learning complex high-dimensional probability distributions while retaining exact likelihood evaluation and efficient sampling (Dinh et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2016; Papamakarios et al., 2021). In this work, we combine iDCI and NFs to develop a framework for distribution refinement and adaptation. We show that the discrepancy ratios computed by iDCI naturally induce sample weights that can be incorporated directly into NF training via a modified loss. This observation enables three capabilities: (i) learning NF representations of iDCI-updated distributions from weighted samples, (ii) progressively refining existing NF models through repeated iDCI-based updates, and (iii) performing distribution-level domain adaptation by learning an adapted input measure while keeping an existing forward model fixed. Related work has shown that generative models can benefit from learned or reweighted latent/base distributions Stimper et al. (2022); Issenhuth et al. (2022), but this work differs by using explicit push-forward distributional constraints rather than relying only on a generative training loss.

2 Data Consistent Inversion (DCI)

DCI is a measure-theoretic framework developed for solving SIPs by quantifying uncertainty over model parameters $\boldsymbol{\theta} \in \Theta$ (Butler et al., 2018; 2020). Given a measurable quantity of interest (QoI) map $Q : \Theta \to \mathcal{D}$, and an observed probability measure P_{obs} defined on the data space $\mathcal{D}$, the objective is to determine a probability measure $P_{\boldsymbol{\theta}}$ on the parameter space whose push-forward through Q agrees with the observed measure. Formally, the DCI solution satisfies the *data-consistency property*

$$P_{\boldsymbol{\theta}}(Q^{-1}(E)) = P_{\text{obs}}(E), \qquad \forall E \in \mathcal{B}_{\mathcal{D}},$$

where $\mathcal{B}_{\mathcal{D}}$ denotes the Borel σ-algebra on $\mathcal{D}$. In other words, the push-forward measure induced by the inferred parameter distribution matches the prescribed observed measure on the QoI space.

In general, the QoI map is nonlinear and many-to-one resulting in a SIP that is not well-posed. Density-based DCI has become the most widely used framework and resolves the issue of non-uniqueness by introducing an initial probability density, denoted π_{init}, which serves as a initial reference distribution for the parameters. Its push-forward through the QoI map results in the predicted density, denoted $\pi_{\text{pred}} = Q_\sharp \pi_{\text{init}}$, which characterizes the model-predicted variability in the observable quantities. Combining π_{pred} with a prescribed observed density π_{obs} yields the DCI updated density

$$\pi_{\text{update}}(\boldsymbol{\theta}) = \pi_{\text{init}}(\boldsymbol{\theta}) \frac{\pi_{\text{obs}}(Q(\boldsymbol{\theta}))}{\pi_{\text{pred}}(Q(\boldsymbol{\theta}))} = \pi_{\text{init}}(\boldsymbol{\theta}) r(Q(\boldsymbol{\theta})), \qquad r(Q(\boldsymbol{\theta})) = \frac{\pi_{\text{obs}}(Q(\boldsymbol{\theta}))}{\pi_{\text{pred}}(Q(\boldsymbol{\theta}))},$$

where the quantity r is commonly referred to as the *update ratio* that quantifies the discrepancy between the observed and predicted distributions and selectively reweights the initial density to achieve data consistency. A notable feature of DCI is that the conditional probability structure inherited from the initial density is preserved along generalized contours of the QoI map while only the relative likelihood across contours is modified. Consequently, DCI may be viewed as the least informative modification of the initial distribution required to satisfy the observational constraints.

The density-based DCI solution is derived using the Disintegration Theorem, which establishes the existence, uniqueness (for a fixed initial measure), and stability of the updated solution under appropriate assumptions (Butler et al., 2020; Pilosov et al., 2023). This distinguishes DCI from Bayesian inference, which derives updates through Bayes' rule, conditional probabilities and a likelihood function. Although both frameworks quantify uncertainty, DCI explicitly seeks a probability measure whose push-forward reproduces the observed distribution exactly, thus being more straigthforward to use for solving SIPs.

A fundamental assumption underlying DCI is the predictability condition, requiring the observed measure to be absolutely continuous with respect to the predicted measure. In the density-based setting, this is commonly expressed as the existence of a constant $C > 0$ such that

$$\pi_{\text{obs}}(Q(\boldsymbol{\theta})) \leq C\, \pi_{\text{pred}}(Q(\boldsymbol{\theta}))$$

for all relevant $\boldsymbol{\theta}$. Intuitively, this condition ensures that the observational information is supported within the variability predicted by the initial model. Under this assumption, the updated density is well-defined and satisfies the desired consistency condition. Moreover, it allows, for straightforward sampling and density estimation techniques, such as kernel density estimation (KDE) (Zambom & Dias, 2013) to be employed.

3 ITERATIVE DATA CONSISTENT INVERSION (IDCI)

Iterative Data Consistent Inversion (iDCI) is designed to handle scenarios where either estimating a full observed joint density is impractical due to limited data availability or high-dimensional output spaces or only dependence information between marginals is unavailable (Jiang et al., 2026). Rather than enforcing consistency with a single multivariate observed distribution, iDCI imposes a collection of lower-dimensional observational constraints sequentially through the Generalized Stochastic Inverse Problem (GSIP). Let $Q_i : \Theta \to \mathcal{D}_i$, $i = 1, \ldots, k$, denote a collection of measurable QoI maps together with corresponding observed probability measures $P_{\text{obs},i}$ on output spaces $\mathcal{D}_i$. The GSIP seeks a probability measure $P_{\boldsymbol{\theta}}$ on the parameter space satisfying the multiple push-forward constraints given by $(P_{\boldsymbol{\theta}} \circ Q_i^{-1})(E) = P_{\text{obs},i}(E), \forall E \in \mathcal{B}_{\mathcal{D}_i}, i = 1, \ldots, k$. The GSIP formulation is substantially more flexible because it permits the incorporation of multiple observational constraints without requiring estimation of a full joint observed distribution over $\mathcal{D} = \mathcal{D}_1 \times \cdots \times \mathcal{D}_k$. Instead, only the individual observed marginal densities associated with each QoI map need to be estimated.

From an optimization perspective, the GSIP solution may be characterized as the probability measure closest to a prescribed initial measure in the Kullback–Leibler (KL) divergence sense while simultaneously satisfying all push-forward constraints. Given an initial density π_{init}, the objective is therefore to determine a density that matches each observed marginal while remaining as close as possible to the reference distribution. The iDCI algorithm constructs this solution through a sequence of iterative updates. Starting from $\pi^{(0)} = \pi_{\text{init}}$, the algorithm cycles through the observational constraints and applies a DCI-type update at each step. If constraint i is selected at iteration n, the density update is given by

$$\pi^{(n)}(\boldsymbol{\theta}) = \pi^{(n-1)}(\boldsymbol{\theta}) \frac{\pi_{\text{obs},i}(Q_i(\boldsymbol{\theta}))}{\pi^{(n-1)}_{\text{pred},i}(Q_i(\boldsymbol{\theta}))}, \qquad \text{where} \qquad \pi^{(n-1)}_{\text{pred},i} = (Q_i)_\sharp \pi^{(n-1)}$$

denotes the predicted marginal density induced by the current iterate. Under generalized predictability assumptions, the resulting sequence converges to a limiting density π_{iDCI}, which corresponds to the information projection (I-projection) of the initial distribution onto the set of all measures satisfying the prescribed push-forward constraints (Jiang et al., 2026). As with standard DCI, the limiting solution depends on the choice of the initial density and should therefore be interpreted as the data-consistent distribution that remains closest to the specified reference measure.

An important distinction between DCI and iDCI is that the latter generally does not recover the true joint distribution underlying the observations. The iterative procedure enforces consistency only with respect to the specified push-forward marginals and therefore does not explicitly constrain the dependence structure among model parameters. Consequently, multiple joint distributions may satisfy the same collection of marginal constraints. This limitation can be understood through Sklar's theorem (Nelsen, 2006). Any absolutely continuous multivariate density may be decomposed into marginal densities and a copula density encoding the dependence structure:

$$\pi(\theta_1, \ldots, \theta_d) = c(F_1(\theta_1), \ldots, F_d(\theta_d)) \prod_{i=1}^{d} \pi_i(\theta_i),$$

where F_i and π_i denote the marginal cumulative distribution functions and densities, respectively, and c is the copula density. Matching the push-forward marginals constrains only the marginal components of this decomposition. Distinct copula functions may therefore produce different joint distributions while preserving exactly the same marginal behavior.

As a result, the iDCI solution should be interpreted as the information projection satisfying the available observational constraints rather than as a guaranteed reconstruction of the true data-generating distribution. Recent work has sought to address this limitation through copula-based extensions that incorporate dependence information into the iterative update procedure (Butler et al., 2026). Such approaches augment the marginal-consistency framework of iDCI with explicit dependence modeling and, under appropriate assumptions, can recover the true DCI solution.

4 NORMALIZING FLOWS

Normalizing Flows (NFs) are likelihood-based generative models that represent a target density through an invertible transformation of a simple reference distribution (Dinh et al., 2016; Papa-

makarios et al., 2021). Let $z \sim \pi_z$ denote a latent variable with known density and let $x = g(z)$ be an invertible mapping. The resulting data space density is given by the change-of-variables formula

$$\pi_x(x) = \pi_z(g^{-1}(x)) \left|\det J_{g^{-1}}(x)\right|.$$

Because likelihoods are available exactly, NFs have become a popular tool for density estimation, uncertainty quantification, simulation-based inference, and scientific machine learning. Throughout this work, NFs are employed primarily as flexible density estimators and generative samplers. The specific flow architecture is not central to the proposed methodology, and the framework is compatible with a wide range of NF architectures including RealNVP (Dinh et al., 2016), Neural Spline Flows (Durkan et al., 2019), and Masked Autoregressive Flows (Papamakarios et al., 2017).

5 Flow-based Representations of DCI and iDCI Solutions

Suppose that samples from the initial distribution $\boldsymbol{\theta}^{(i)} \sim \pi_{\text{init}}$ are available. DCI and iDCI naturally associate a weight to each sample through the corresponding update ratio. In the DCI setting, these weights are given by $r(Q(\boldsymbol{\theta}))$ while in iDCI analogous weights are generated iteratively from the ratios of observed and predicted marginal densities associated with the current iterative constraint. Consequently, both methods produce a weighted sample representation of the target pullback distribution. The primary objective of this work is to convert such weighted samples into an explicit probabilistic representation of the updated distribution. To this end, let $\hat{\pi}(\boldsymbol{\theta}; \boldsymbol{\phi})$ denote a NF approximation of the target density, where the target may correspond either to π_{update} or π_{iDCI}. We compute the optimal parameters $\boldsymbol{\phi}$ that minimize the KL divergence

$$\begin{aligned}\boldsymbol{\phi} = \arg\min_{\boldsymbol{\phi}} D_{\text{KL}}\big(\pi_{\text{update}} \,\|\, \hat{\pi}(\cdot;\boldsymbol{\phi})\big) = \arg\min_{\boldsymbol{\phi}} \mathbb{E}_{\boldsymbol{\theta}\sim\pi_{\text{update}}}\Big[\log \pi_{\text{update}}(\boldsymbol{\theta}) - \log \hat{\pi}(\boldsymbol{\theta};\boldsymbol{\phi})\Big] \\ = \arg\min_{\boldsymbol{\phi}} \mathbb{E}_{\boldsymbol{\theta}\sim\pi_{\text{update}}}\Big[-\log \hat{\pi}(\boldsymbol{\theta};\boldsymbol{\phi})\Big] = \arg\min_{\boldsymbol{\phi}} \mathbb{E}_{\boldsymbol{\theta}\sim\pi_{\text{init}}}\Big[-r(Q(\boldsymbol{\theta}))\log \hat{\pi}(\boldsymbol{\theta};\boldsymbol{\phi})\Big], \qquad (1)\end{aligned}$$

where the final equality follows from the DCI update formula and a change of measure. Since independent samples from π_{init} are readily available, the objective in equation 1 may be approximated using a Monte Carlo estimator. The resulting optimization problem is therefore equivalent to weighted maximum-likelihood training, where the sample weights are provided directly by the DCI or iDCI update ratios. In practice, this allows the information contained in the discrepancy weights to be distilled into the parameters of a NF, producing a compact surrogate representation of the updated pullback density.

An explicit NF representation provides two important advantages. First, once training is completed, likelihood evaluations become inexpensive, enabling direct evaluation of the learned approximation $\hat{\pi}(\boldsymbol{\theta}; \boldsymbol{\phi})$ without repeatedly performing DCI or iDCI updates. Second, because a NF is defined as an invertible transformation of a simple base distribution, generating surrogate samples from the learned density is computationally inexpensive. Consequently, after training, the flow provides a fast generative model for the DCI or iDCI solution.

6 Refinement of Base Distributions in Normalizing Flows

As previously discussed, a NF requires a base distribution $\pi_{\mathbf{z}}$, and a standard Gaussian base distribution is often sufficient in practice. However, significant mismatches between the base and target distributions may lead to density artifacts or require increased model complexity and training effort. To address this issue, we propose a two-stage refinement strategy based on iDCI that progressively adapts the base distribution to better reflect the structure of the target density. In Stage 1, an initial NF is trained to represent the target density, producing a predictive distribution π_{pred}. Samples generated from its base distribution are propagated through the learned flow and used to compute iDCI discrepancy weights with respect to observed marginal densities. In Stage 2, these weights are used to learn an updated base distribution that approximates the pullback iDCI solution. This new base model is then composed with the original NF and the resulting augmented model is fine-tuned on the target samples. A schematic of this procedure is shown in Figure 1 with the caption summarizing the steps within each stage of the procedure.

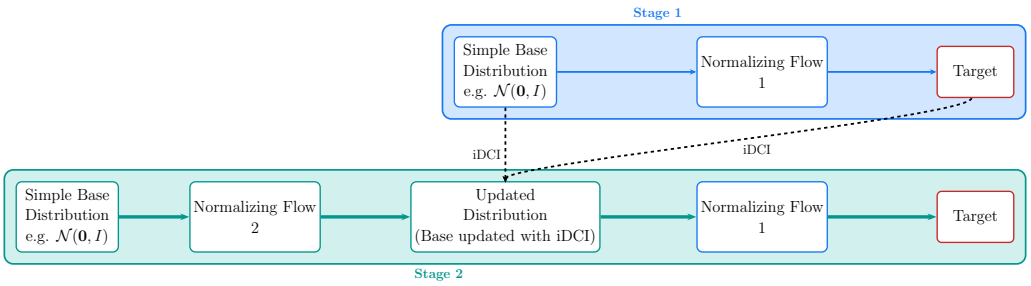


Figure 1: Proposed two stage (S1 and S2) workflow for NF base distribution refinement using iDCI. **S1.a** Train an initial NF using target samples. The resulting model represents the predictive density π_{pred}. **S1.b** Generate m samples from the current base distribution, $\{\mathbf{z}_i\}_{i=1}^m$, and compute their push-forward samples $\{g(\mathbf{z}_i; \boldsymbol{\phi})\}_{i=1}^m$. **S1.c** Estimate the observed marginal densities $\pi_{\text{obs},i}$. **S2.a** Compute iDCI weights that quantify discrepancies between the predicted and observed marginals. **S2.b** Train a second NF on the latent samples $\{\mathbf{z}_i\}_{i=1}^m$ using the computed weights. This distribution becomes the refined base distribution and approximates π_{iDCI}. **S2.c** Augment the original NF with the newly learned base NF. **S2.d** Jointly retrain the augmented model using the target samples.

Two-Dimensional Density Estimation Example. To illustrate the proposed rebasing strategy, we compare three approaches for a two-dimensional multimodal example: **(1) Proposed rebasing method** using sequential NF refinement guided by iDCI weights; **(2) Sequential NF training** using an identical sequential architecture without iDCI correction; and **(3) Full NF training** where a single NF is trained using the complete architecture from the outset, which serves as the baseline comparison. To ensure a fair comparison, all methods were allocated the same training budget. The sequential approaches employed k iterations during the first two stages followed by p joint training iterations, while the full NF model was trained for an equivalent total number of optimization steps.

We now summarize the problem setup. Each NF consists of two coupling layers with hidden dimensions $[4, 4]$. The augmented model therefore contains four coupling layers. An AdamW optimizer with learning rate 6×10^{-4} was utilized. For the training budget, we set $k = 1200$ and $p = 2000$ for the iterations. A total of 2500 target samples were generated from an equally weighted mixture of two Gaussian distributions, and Gaussian KDEs were utilized for every marginal density estimate required by iDCI. Figures 2–4 summarize the results.

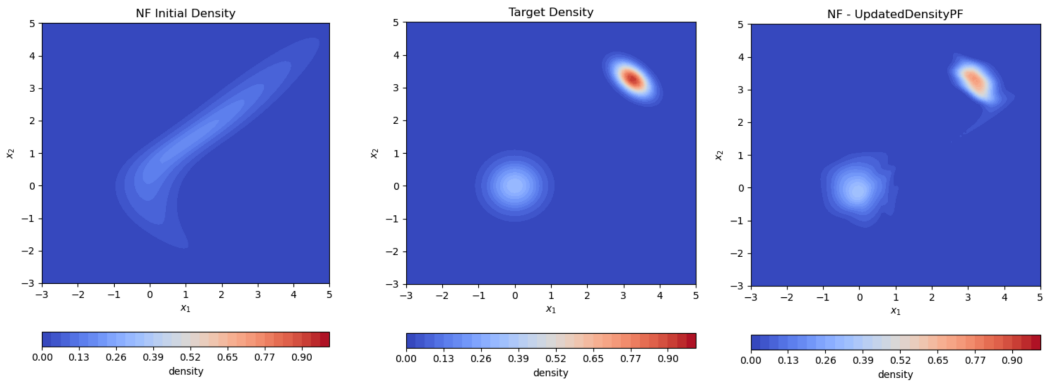


Figure 2: Left-to-right: Initial NF, Target, and Push-forward of Updated/Re-based NF densities. We notice that the initial NF density produces a distribution that is not bimodal like the true target distribution, only being able to cover its support with a KL divergence of 1.1676 from the target, while the re-basing produces a qualitative match whose KL divergence is within 0.0713 of the target.

The initial NF successfully captures the support of the target distribution but remains largely unimodal, yielding $D_{\text{KL}}(\pi_{\text{target}} \| \pi_{\text{init}}) = 1.1676$. Applying iDCI to $m = 6000$ generated samples produces a corrected base distribution that differs substantially from the original Gaussian base and better reflects the pullback structure of the target density (shown in Figure 3). After rebasing and

joint retraining, the final model achieves $D_{\mathrm{KL}}(\pi_{\mathrm{target}}\|\pi_{\mathrm{Final}}) = 0.0713$, accurately recovering both target modes while significantly reducing the density artifact connecting them.

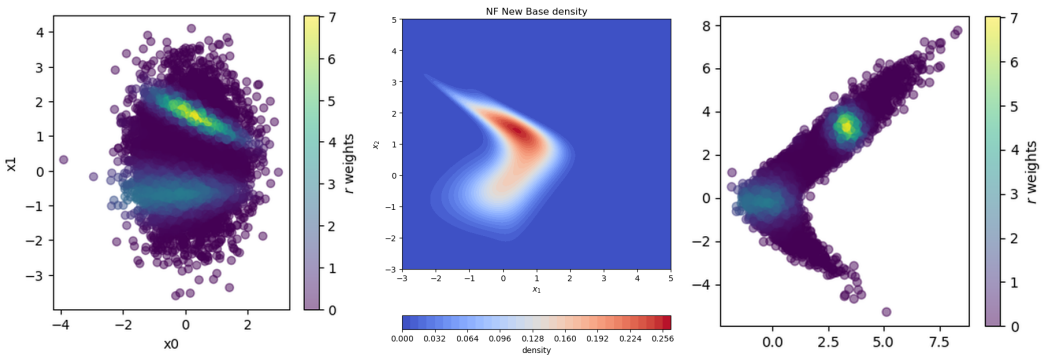


Figure 3: Left: Generated $m = 6000$ initial samples for iDCI-based re-basing. Center: The NF learned updated distribution that re-bases the original NF. Right: The associated $m = 6000$ predicted samples (push-forward of the initial). In the left and right plots, the r-values illustrate the iDCI learned weights, which demonstrates how this newly learned base pushes-forward through the prior NF to reconstruct the right plot of Fig. 2.

As baseline comparisons, the Sequential NF Training (left plot in Figure 4) approach produced a similarly accurate approximation of the target density, achieving $D_{\mathrm{KL}}\left(\pi_{\mathrm{target}}\|\pi_{\mathrm{FinalSeq}}\right) = 0.0905$. This suggests that the observed improvement is not solely due to the sequential architecture but also to the data-consistent correction introduced by iDCI. The Full NF baseline achieved $D_{\mathrm{KL}}\left(\pi_{\mathrm{target}}\|\pi_{\mathrm{FullNF}}\right) = 0.1475$, demonstrating that a sufficiently expressive NF can approximate the target distribution directly (right plot in Figure 4 associated density plot). Nevertheless, its performance remained inferior to both sequential approaches. This result suggests that progressively increasing model expressiveness can be more effective than training a larger architecture from the outset. In particular, the proposed rebasing framework provides a constructive mechanism for refining model complexity only when required by the target distribution.

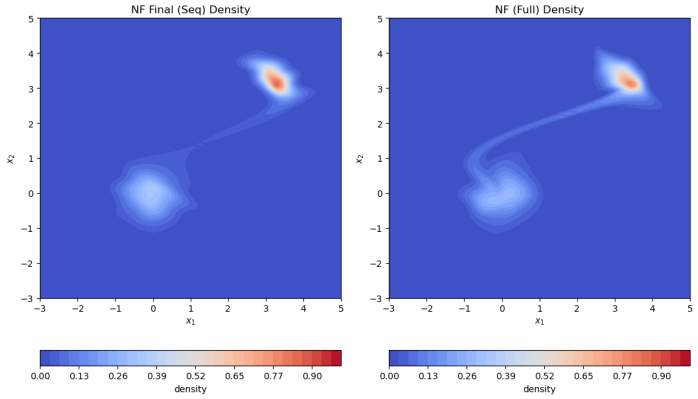


Figure 4: Densities obtained using Sequential NF Training (left) and the Full NF approach (right). A significant artifact remains between the modes compared to the staged re-basing approach.

Table 1 summarizes the computational costs of all methods. The proposed approach incurs additional overhead due to the iDCI correction stage and weighted optimization procedure. However, the current implementation has not been fully optimized, and the reported runtimes should not be interpreted as intrinsic limitations of the methodology. As expected, Full NF Training is the least expensive approach because it avoids the intermediate refinement stages, although it also produced the least accurate density approximation. The additional computational cost of rebasing is therefore exchanged for improved approximation accuracy and a principled framework for adaptive model refinement. Additionally, the utilization of separate samples for re-basing with iDCI is not fully

	Training of NF 1	iDCI	Training of NF 2	Joint Training	Total
Our approach	2.893	2.893	4.702	8.796	19.284
Sequential NF Training	2.791	-	3.625	8.536	14.952
Full NF Training	-	-	-	13.509	13.509

Table 1: Comparison of computational times (in seconds) and density approximation performance for the proposed methodology and the baseline approaches. Despite the overhead of re-basing, it achieves improved density estimation performance through iterative refinement and dependency-structure correction.

explored in terms of further savings achieved by optimizing this number to the necessary amount for a distribution that, in principle, is simpler to approximate than the original target since it is defined as a perturbation of the original base.

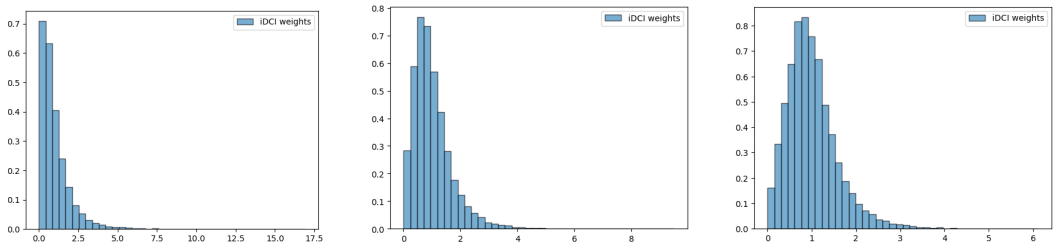
7 Distribution Adaptation of Stochastic Models

In many real-world applications, stochastic models operate in environments that evolve over time. Consequently, the probability distribution that originally governed the system may differ from the distribution associated with newly observed data, a phenomenon commonly referred to as *distributional drift*. When such a shift occurs, a pre-trained stochastic model may no longer produce outputs that accurately reflect the current environment, despite the underlying physical or computational mapping remaining valid. Let $g : \mathcal{X} \to \mathcal{Y}$ denote an existing model that pushes forward an initial probability distribution π_{init} into an output distribution, $\pi_{\text{init}} \xrightarrow{g} \pi_g$. Suppose that new observations indicate a desired output distribution π_{obs} that differs from π_g only marginally along a small subset of dimensions. Retraining the model g may be computationally prohibitive, infeasible, or impossible when g is only available as a black-box simulator. Therefore, rather than modifying the stochastic operator itself, we seek to adapt the probability measure supplied to the model.

The proposed methodology is motivated DCI perspective, where observed output densities are used to infer a probability measure on the model inputs that is consistent with the available data. In particular, we again utilize iDCI framework to compute importance weights, referred to here as r-weights, that quantify how individual samples from the original input distribution contribute to the discrepancy between the current model predictions and the observed target behavior. These weights induce a reweighted measure on the input space, effectively identifying how probability mass should be redistributed within the original base distribution to match the specific distributional drifts observed/imposed by the new target. This idea is closely related to importance-weighting and covariate-shift methodologies, where distributional differences are represented through density-ratio corrections rather than modifications to the predictive mechanism itself (Shimodaira, 2000; Sugiyama et al., 2008; Huang et al., 2007).

As before, to represent this adapted measure, we train a NF using the weighted samples generated by iDCI. The resulting NF provides a flexible and tractable approximation of a new base distribution, denoted by $\tilde{\pi}$, while preserving a valid probabilistic representation. This adaptation framework may be summarized as $\tilde{\pi} \xrightarrow{g} \tilde{\pi}_g$, where $\tilde{\pi}$ denotes the adapted base distribution learned with the NF from the iDCI-weighted samples, and $\tilde{\pi}_g$ represents its push-forward through the fixed operator g. The objective is to construct $\tilde{\pi}$ such that $\tilde{\pi}_g \approx \pi_{\text{obs}}$, thereby ensuring consistency between the model predictions and the newly observed data while leaving the original model unchanged. In this sense, the NF acts as a transport mechanism between the original and adapted input measures, while the physical or computational model g remains untouched (Marzouk et al., 2016).

The methodology may also be viewed as a form of domain adaptation. Classical domain adaptation methods typically seek to learn a new predictive model or align latent feature representations between source and target domains (Pan & Yang, 2010; Kouw & Loog, 2019). In contrast, our approach assumes that the forward operator g remains valid and instead adapts the distribution supplied to it. Thus, rather than transferring predictive knowledge through feature alignment, knowledge transfer is achieved by modifying the underlying probability measure associated with the model inputs. Consequently, the proposed framework may be regarded as a distribution-level domain adaptation strategy, where adaptation is performed entirely through the construction of a



(a) Weights after the first rebasing stage ($n = 1$).

(b) Weights after the second rebasing stage ($n = 2$).

(c) Weights after the third rebasing stage ($n = 3$).

Figure 5: Evolution of r-weight distributions across successive rebasement stages. The progressive narrowing and concentration of the weight distribution towards a peak centered at $r = 1$ (as seen from stage $n = 3$) indicates that the predictive model's output (π_{pred}) is sequentially aligning with the observed marginal constraints (π_{obs}), thus achieving improved distributional consistency.

new input measure represented by a NF and inferred using the data-consistent weighting principles of iDCI.

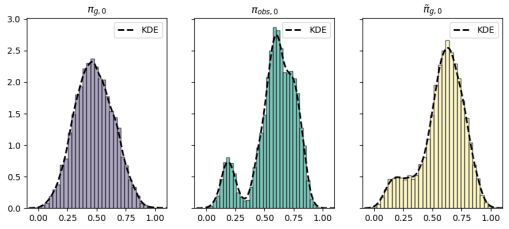
To verify the efficacy of our approach, we designed a synthetic 10 dimensional example. Our baseline stochastic model g was established as a Real NVP architecture (18 coupling layers, [36, 36] hidden units) which maps a standard Gaussian prior (π_{init}) to a more complex density π_{g}, which is defined by a mixture distribution composed of non-standard Gaussians. We then increased the complexity of this system by perturbing π_{target}: the joint target manifold remains generally unchanged, except distinct multimodal structures on three specific marginal dimensions (dimensions 0, 3, and 9). This introduces a mismatch between the current model and the new given target distribution, which we will denote as the observed distribution π_{obs}.

To adapt to the new target, we employ our proposed rebasing mechanism. After generating initial (and their corresponding predicted samples) from the current model, we compute iDCI weights so that the 1D marginal distribution of π_{obs} are enforced. This leads to training a corrected Normalizing Flow, which functions as the updated base distribution ($\tilde{\pi}$).

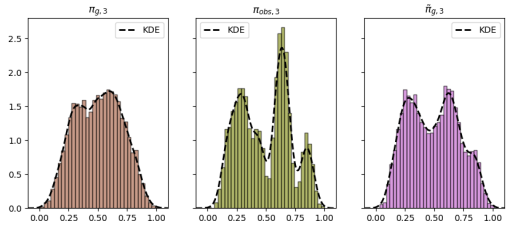
A useful aspect of our framework is the possibility of performing sequential refinement. We can consider the resulting updated distribution $\pi^{(n)}_{\text{update}}$ from any stage n as the initial base distribution $\pi^{(n+1)}_{\text{init}}$ for a subsequent rebasing procedure. This allows us to systematically augment and refine the Normalizing Flow's learned representation over multiple stages. The process can be continued until convergence is observed, which is indicated by the stabilization of the discrepancy weights r. Specifically, as the iterative process converges, the expected value of r-values distribution peaks around unity. When this expectation is close to 1, it means the predicted π_{pred} and observed marginals π_{obs} are in agreement, suggesting that the new base has successfully adjusted for the distributional mismatch quantified by iDCI.

We demonstrate this principled approach by performing three consecutive rebasings: here each one being a RealNVP with 4 coupling layers each with [36, 36] hidden units. As shown in Figure 5, the evolution of the r-weight histogram provides strong evidence for this iterative refinement: the initial weights display a wide distribution, significantly deviating from unity; however, after each subsequent iteration, the weight distribution progressively shifts toward a peak centered near $r = 1$. This highlights that applying iDCI sequentially is an effective strategy for driving the model's density estimate toward greater fidelity with respect to the imposed push-forward constraints.

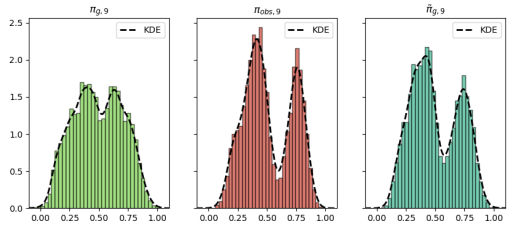
The obtained results of this process are outlined in Figure 6. We observe a significant improvement in the marginal distributions derived from the push-forward of the new base $\tilde{\pi}_g$, when compared to those of the original model π_g, relative to the drifted observed marginals π_{obs}. This improvement is also reflected in the KL divergences, which decreased from 0.3576, 0.0547, and 0.0972 for π_g to 0.0279, 0.0242, and 0.0117 for $\tilde{\pi}_g$ in dimensions 0, 3, and 9, respectively.. Furthermore, we note that this approach successfully operates using only the marginalized KDE representations of the observed densities. This is particularly valuable in scenarios where joint samples from π_{obs} are unavailable or costly to generate. Still, it is possible to incorporate such full joint samples, which



(a) Marginal distribution for Dimension 0.



(b) Marginal distribution for Dimension 3.



(c) Marginal distribution for Dimension 9.

Figure 6: Marginal distributions comparison for dimensions 0, 3, and 9. The figure compares three densities: π_g (the current model's predicted marginal), π_{obs} (the target observed marginal), and $\hat{\pi}_g$ (the push-forward of the rebased NF model). As shown, the refined estimate $\hat{\pi}_g$ achieves an improved approximation to π_{obs} compared to π_g. Quantitatively, this improvement is quantified by the 1D marginal KL divergence reduction: 92%, 55%, and 88% respectively for dimensions 0, 3, and 9.

can be beneficial for capturing dependency (copula) structures that are not fully characterized using only iDCI and rebasing procedures.

8 Final Discussion

This work introduced a framework that combines Data-Consistent Inversion (DCI), Iterative Data-Consistent Inversion (iDCI), and Normalizing Flows (NFs) for learning and refining probability distributions in stochastic inverse problems. Using the discrepancy weights generated by DCI and iDCI, we showed how flow-based models can be trained to represent updated pullback distributions while retaining the efficient sampling and likelihood-evaluation capabilities of normalizing flows. Three main contributions were presented. First, we developed a weighted NF training strategy that learns explicit representations of DCI and iDCI solutions directly from samples and associated update weights. Second, we introduced a rebasing methodology in which iDCI is used to iteratively refine the base distribution of an existing NF, providing a constructive mechanism for progressively improving model fidelity. Third, we demonstrated how the same framework can be applied to distribution-level domain adaptation by learning a new input measure while keeping an existing stochastic operator fixed. The numerical examples showed that incorporating iDCI-informed corrections into the latent space can improve density approximation quality relative to standard NF training strategies while simultaneously providing interpretable intermediate distributions. More broadly, the proposed framework establishes a connection between data-consistent inversion and modern generative modeling, enabling updated distributions to be represented through compact and computationally efficient surrogate models. Future work will focus on improving the computational scalability of the iDCI procedure through parallel and batched implementations, as well as investigating training strategies that improve the numerical stability of the learned flow representations.

AI use statement

Generative AI tools were used as writing and software-development assistants during the preparation of this work. Microsoft Copilot (GPT-based) assisted with improving the readability and

presentation of several manuscript sections, refining portions of the Python package developed for this research, and identifying potentially relevant references for further review. A locally deployed Gemma 4 model was also used to suggest alternative wording and revisions for selected portions of the text. All generated suggestions were reviewed by the authors, who take full responsibility for the final text, software, scientific claims, methodology, results, and conclusions presented in this paper.

Reproducibility statement

All code used in this work will be made publicly available through a GitHub repository upon publication as it contains identifying information for the authors. The repository will contain the Python packages developed for this work, along with Jupyter notebooks reproducing the examples, numerical experiments, and figures presented throughout the paper. Documentation and instructions necessary to reproduce the reported results will also be provided.

Acknowledgments

This work was supported by an NSF grant. Details will be provided upon acceptance as the award information contains identifying information for the authors.

References

T Butler, J Jakeman, and Tim Wildey. Combining push-forward measures and bayes' rule to construct consistent solutions to stochastic inverse problems. *SIAM Journal on Scientific Computing*, 40(2):A984–A1011, 2018.

Troy Butler, Timothy Wildey, and Tian Yu Yen. Data-consistent inversion for stochastic input-to-output maps. *Inverse Problems*, 36(8):085015, 2020.

Troy Butler, Tianyi Jiang, João Silva, Harri Hakula, and Timothy Wildey. Copula transformations for data-consistent inversion, 2026. URL https://arxiv.org/abs/2609.02832.

Masoumeh Dashti and Andrew M. Stuart. The bayesian approach to inverse problems. In Roger Ghanem, David Higdon, and Houman Owhadi (eds.), *Handbook of Uncertainty Quantification*, pp. 311–428. Springer, Cham, 2017. doi: 10.1007/978-3-319-12385-1_7. URL https://doi.org/10.1007/978-3-319-12385-1_7.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. In *Advances in Neural Information Processing Systems*, volume 32, pp. 7511–7522, 2019. URL https://papers.nips.cc/paper/8969-neural-spline-flows.

Jiayuan Huang, Arthur Gretton, Karsten Borgwardt, Bernhard Schölkopf, and Alex Smola. Correcting sample selection bias by unlabeled data. 2007.

Thibaut Issenhuth, Ugo Tanielian, David Picard, and Jérémie Mary. Latent reweighting, an almost free improvement for GANs. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1668–1677, 2022. URL https://openaccess.thecvf.com/content/WACV2022/html/Issenhuth_Latent_Reweighting_an_Almost_Free_Improvement_for_GANs_WACV_2022_paper.html.

Tianyi Jiang, Troy Butler, Timothy Wildey, Tim Kutta, and Haonan Wang. Iterative data-consistent inversion with multiple push-forward constraints, 2026. URL https://arxiv.org/abs/2603.00033.

Jari P. Kaipio and Erkki Somersalo. *Statistical and Computational Inverse Problems*, volume 160 of *Applied Mathematical Sciences*. Springer, New York, 2005. doi: 10.1007/b138659. URL https://doi.org/10.1007/b138659.

Wouter M. Kouw and Marco Loog. A review of domain adaptation without target labels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(3):766–785, 2019.

Youssef Marzouk, Tarek Moselhy, Matthew Parno, and Alessio Spantini. Sampling via measure transport: An introduction. *Handbook of Uncertainty Quantification*, 2016.

Roger B. Nelsen. *An Introduction to Copulas*. Springer Series in Statistics. Springer, New York, 2nd edition, 2006. ISBN 978-0-387-28659-4. doi: 10.1007/0-387-28678-0.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems*, volume 30, pp. 2338–2347, 2017.

George Papamakarios, Eric Nalisnick, Danilo Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021.

Michael Pilosov, Carlos del Castillo-Negrete, Tian Yu Yen, Troy Butler, and Clint Dawson. Parameter estimation with maximal updated densities. *Computer Methods in Applied Mechanics and Engineering*, 407:115906, 2023.

Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

Vincent Stimper, Bernhard Schölkopf, and José Miguel Hernández-Lobato. Resampling base distributions of normalizing flows. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 4915–4936. PMLR, 2022. URL https://proceedings.mlr.press/v151/stimper22a.html.

Andrew M. Stuart. Inverse problems: A bayesian perspective. *Acta Numerica*, 19:451–559, 2010. doi: 10.1017/S0962492910000061. URL https://doi.org/10.1017/S0962492910000061.

Masashi Sugiyama, Taiji Suzuki, Shinichi Nakajima, Hisashi Kashima, Paul von Bünau, and Motoaki Kawanabe. Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, 60(4):699–746, 2008.

Albert Tarantola. *Inverse Problem Theory and Methods for Model Parameter Estimation*. Society for Industrial and Applied Mathematics, Philadelphia, 2005. doi: 10.1137/1.9780898717921. URL https://doi.org/10.1137/1.9780898717921.

Adriano Z Zambom and Ronaldo Dias. A review of kernel density estimation with applications to econometrics. *International Econometric Review*, 5(1):20–42, 2013.