Novel Uncertainty Quantification through Perturbation-Assisted Sample Synthesis

Yifei Liu, Rex Shen, Xiaotong Shen

***Abstract*—This paper introduces a novel Perturbation-Assisted Inference (PAI) framework utilizing synthetic data generated by the Perturbation-Assisted Sample Synthesis (PASS) method. The framework focuses on uncertainty quantification in complex data scenarios, particularly involving unstructured data while utilizing deep learning models. On one hand, PASS employs a generative model to create synthetic data that closely mirrors raw data while preserving its rank properties through data perturbation, thereby enhancing data diversity and bolstering privacy. By incorporating knowledge transfer from large pre-trained generative models, PASS enhances estimation accuracy, yielding refined distributional estimates of various statistics via Monte Carlo experiments. On the other hand, PAI boasts its statistically guaranteed validity. In pivotal inference, it enables precise conclusions even without prior knowledge of the pivotal's distribution. In non-pivotal situations, we enhance the reliability of synthetic data generation by training it with an independent holdout sample. We demonstrate the effectiveness of PAI in advancing uncertainty quantification in complex, data-driven tasks by applying it to diverse areas such as image synthesis, sentiment word analysis, multimodal inference, and the construction of prediction intervals.**

***Index Terms*—Uncertainty Quantification, Diffusion, Normalizing Flows, Large pre-trained Models, Multimodality, High-dimensionality.**

I. Introduction

Uncertainty quantification is pivotal in scientific exploration and drawing reliable conclusions from data, especially working with complex modeling techniques such as deep neural networks. Despite recent advancements showcasing the potential of Artificial Intelligence in facilitating data-driven discoveries, a reproducibility crisis has emerged in various fields, including biomedicine and social sciences, occasionally leading to false discoveries [1]. A key issue contributing to this crisis is the lack of methods for quantifying uncertainty in over-parametrized models, like neural networks, prioritizing predictive accuracy using many non-learnable parameters such as hyperparameters. As a result, these studies may become exaggerated and irreproducible. To address these challenges, we develop a generative inference framework designed to provide uncertainty quantification for data of any type.

Diverse methodologies for uncertainty quantification are prevalent in the literature. Approaches such as those in [2], [3], [4], [5] evaluate the predictive model's outcome uncertainty, with broad applications spanning adversarial attacks to anomaly detection [6], [7]. Furthermore, studies [8], [9] investigate uncertainty within large language models for question-answering tasks. Nevertheless, prevailing metrics like negative log-likelihood often forgo solid statistical foundations, such as confidence or probability assertions, within the framework of statistical inference.

In statistical inference, the quantification of uncertainty is imperative. Classical techniques like Bootstrap [10] solve conventional statistical problems. Yet, uncertainty in complex models, particularly those involving deep networks and unstructured data, as indicated by [3], remains less explored. The conformal inference method [11], [12] offers a practical tool for valid uncertainty quantification. However, its effectiveness is significantly influenced by the underlying prediction model and the selection of the conformal score, which can lead to overly cautious inferential outcomes. With recent progress, such as [13]'s introduction of hypothesis testing for feature significance using asymptotic methods, a comprehensive examination of statistical uncertainty quantification becomes imperative. Our focus herein is statistical inference, specifically hypothesis testing, which quantifies the uncertainty of a hypothesis test's outcome or conveys a confidence declaration concerning prediction uncertainty, as detailed in Section V.

This paper introduces the novel Perturbation-Assisted Inference (PAI) framework that employs Perturbation-Assisted Sample Synthesis (PASS) as its core generator, ensuring validity as if we had conducted Monte Carlo (MC) simulations with a known data-generating distribution. To clarify the core concept of our approach, envision statistics computed via a machine learning or statistical technique on a training data set. These statistics may embody a predicted outcome in supervised learning or a test statistic in hypothesis testing. By generating multiple iterations of these statistics on synthetic data that emulate the original data's distribution, we gauge the variability of these statistics across data sets with an analogous distribution to the original by applying the same analytical method. PASS generates these synthetic data sets, while PAI procures reliable inferences from them, employing Monte Carlo techniques.

PASS synthesizes data that mirrors the original data closely, encompassing both tabular and unstructured data such as gene expressions and text. Its distinct edge is in harnessing pre-trained generative models to heighten generation precision. With an emphasis on inference, PASS augments synthetic data diversity and privacy through data perturbation, retaining the original sample's ranks, which supports personalization and

This work was supported in part by NSF grant DMS-1952539, and NIH grants R01AG069895, R01AG065636, R01AG074858, U01AG073079. (Corresponding author: Xiaotong Shen.)

Yifei Liu is with the School of Statistics, University of Minnesota, MN, 55455 USA (email: liu00980@umn.edu).

Rex Shen is with the Department of Statistics, Stanford University, CA, 94305 USA (email:rshen0@stanford.edu).

Xiaotong Shen is with the School of Statistics, University of Minnesota, MN, 55455 USA (email: xshen@umn.edu).

data amalgamation [14]. Through neural networks, PASS maps a base distribution into a target one, drawing from the round-trip transformation strategy used in normalizing flows [15], [16] or diffusion models [17], [18], and broadens the conventional univariate data generation approach by transposing the cumulative distribution function from a uniform base to preserve the original data's univariate ranks.

The PAI framework is a significant advancement in statistical inference, particularly for unstructured, multimodal, and tabular data. It exceeds traditional methods in reliability and breadth of application, chiefly through synthetic data created by PASS to emulate any statistic's distribution and properties via Monte Carlo testing. This framework, in contrast to classical methods requiring bias corrections, deduces the distribution of a test statistic via an approximated data generation distribution, thereby facilitating finite-sample inference. Additionally, it trumps resampling methodologies by producing independent synthetic samples for inference. This function promotes broader applications, including data integration, sensitivity analysis, and personalization, thereby widening the gamut of statistical inference into new domains. Specifically,

(1) **Inference for Unstructured and Multimodal Data.** The PAI framework broadens the scope of statistical inference from numerical to unstructured and multimodal data through synthetic data generation. Section V demonstrates the validity of PAI when PASS estimates the data-generating distribution via pre-trained generative models such as normalizing flow or diffusion models.
(2) **Pivotal Inference.** PAI offers exact inference for any pivotal while controlling the Type-I error, which surpasses classical methods that necessitate knowledge of a test statistic's distribution, as supported by Theorem 2.
(3) **General Inference.** The PAI framework enables approximate inference for non-pivotal statistics while maintaining control over Type-I errors. It achieves this by using an estimated distribution well approximating the data-generating distribution, as illustrated in Theorem 1.
(4) **Accounting for Modeling Uncertainty.** PAI distinguishes itself from conventional methods by incorporating modeling uncertainty into the Monte Carlo experiments for uncertainty assessment, leading to more credible conclusions.

To demonstrate PAI's capabilities, we address statistical inference challenges in three previously untapped areas: (1) image synthesis, (2) sentiment analysis using DistilBERT [19], and (3) multimodal inference from text to image generation based on text prompts. Moreover, we also contrast PAI with the conformal inference approach [11] for prediction uncertainty in regression problems. In these scenarios, PAI quantifies uncertainty for generative models that involve hyperparameter optimization, considering the statistical uncertainty of such tuning in the inference process and leveraging pre-trained models to refine the accuracy of learning the data-generating distribution. Contemporary research underscores the significance of sample partitioning in inference to avert data dredging [20], [21]. Demonstrated through these applications, PAI conducts innovative hypothesis testing for image synthesis, word inference in sentiment analysis, and generated images from varying text prompts via stable diffusion techniques, thus providing uncertainty quantification for numerical and unstructured data where tests are not analytically tractable.

This paper comprises the following sections: Section II establishes the foundation of PASS, enabling the estimation of any statistic's distribution through Monte Carlo simulations. Section III introduces the PAI framework and the PASS generator. Section IV offers a statistical validation of the PAI framework. Section V develops tests for comparing synthetic images generated by diffusion models [17], [18] and other deep generative models such as GLOW [22] and DCGAN [23], also addressing the evaluation of word significance in sentiment analysis using DistilBERT, multimodal inference from texts to images. Section VI presents numerical experiments. This section additionally contrasts the PAI methodology with the conformal inference approach in quantifying prediction uncertainty in regression problems. Supplementary materials include implementation details for the numerical examples, technical specifics, multivariate ranks, and learning theory for normalizing flows.

II. Perturbation-Assisted Sample Synthesis

Given a d-dimensional random sample $\boldsymbol{Z} = (\boldsymbol{Z}_i)_{i=1}^n$ from a cumulative distribution function (CDF) $F_{\boldsymbol{Z}}(\cdot) = F_{\boldsymbol{Z}}(\cdot\,;\boldsymbol{\theta})$, or data-generating distribution, $\boldsymbol{Z}_i \sim F_{\boldsymbol{Z}}$; $i = 1, \ldots, n$, we estimate a statistic $H(\boldsymbol{Z})$'s distribution, where $\boldsymbol{\theta}$ is a vector of unknown parameters and H is a vector of known functions that may be analytically intractable. Here, $\boldsymbol{Z}$ could be an independently and identically distributed sample or its continuous latent vector representation obtained through, for example, a latent normalizing flow ([15], [16]) and VAE [24] for images and a numerical embedding such as BERT-style transformer for texts. Subsequently, we assume that $F_{\boldsymbol{Z}}$ is absolutely continuous and use the continuous latent vectors of unstructured data or a continuous surrogate of non-continuous data [25] for a downstream task.

A. Sample Synthesis

Generation via Transport. To generate a random sample $\boldsymbol{Z}' = (\boldsymbol{Z}'_i)_{i=1}^n$ from a cumulative distribution $F_{\boldsymbol{Z}}$, we construct a transport G mapping a base distribution of $\boldsymbol{U}$ to that of $\boldsymbol{Z}$, preferably simple, like the Uniform or Gaussian, where $\boldsymbol{U} = (\boldsymbol{U}_i)_{i=1}^n$ is a sample from the base distribution $F_{\boldsymbol{U}}$. In the univariate case, we generate $\boldsymbol{Z}'_i = G(\boldsymbol{U}_i)$ by choosing $G = F_{\boldsymbol{Z}}^{-1}$ with $\boldsymbol{U}_i$ sampled from the Uniform distribution $U[0,1]$; $i = 1, \ldots, n$. However, this generative method is no longer valid in the multivariate case as the multivariate analogy of $F_{\boldsymbol{Z}}^{-1}$ does not exist. In such a situation, the reconstruction of G mapping $\mathcal{R}^d$ to $\mathcal{R}^d$ is challenging.

Linkage between Generated and Original Data. Sample $\boldsymbol{Z}'$ generated from the base distribution of $\boldsymbol{U}$ may not accurately represent $\boldsymbol{Z}$ if they are unrelated to $\boldsymbol{Z}$. When $d = 1$, $\boldsymbol{Z}'$ retains the ranks of $\boldsymbol{Z}$ if $\boldsymbol{U}$ retains the ranks of $\boldsymbol{Z}$, by

the non-decreasing property of $G = F_{\boldsymbol{Z}}$. As argued in [25], $\boldsymbol{Z}'$ connects to the original sample $\boldsymbol{Z}$ by rank preservation. This aspect is crucial for personalized inference, outlier detection, and data integration. To generalize this concept of rank preservation to the multivariate situation, we consider a transport T mapping from $F_{\boldsymbol{Z}}$ to $F_{\boldsymbol{U}}$, which is not necessarily invertible. However, the invertibility ensures a round-trip transformation between $F_{\boldsymbol{Z}}$ and $F_{\boldsymbol{U}}$ is uniquely determined. We then align the multivariate ranks of $(\boldsymbol{U}_i)_{i=1}^n$ with those of $(T(\boldsymbol{Z}_i))_{i=1}^n$, which preserves the ranks of $(\boldsymbol{Z}_i)_{i=1}^n$ using its representation $(T(\boldsymbol{Z}_i))_{i=1}^n$ in the space of the base variables $\boldsymbol{U}$. The reader is directed to the supplementary materials for detailed information on multivariate ranks. In other words, this alignment preserves the ranks of $(\boldsymbol{Z}_i)_{i=1}^n$ by $(\boldsymbol{U}_i)_{i=1}^n$ when T is invertible and recovers the univariate case. In practice, we may reconstruct G with $T = G^{-1}$ as in the case of normalizing flow or treat a non-invertible T separately as in a diffusion model; see subsequent paragraphs for examples.

Perturbation for Diversity and Protection. Recent research in denoising diffusion models ([26], [17], [27]) has demonstrated that adding Gaussian noise in the forward diffusion process and subsequent denoising to estimate the initial distribution $F_{\boldsymbol{Z}}$ in the reverse process can effectively improve the diversity of generated samples. Moreover, adding noise in a certain form of data perturbation [14] can allow $\boldsymbol{Z}'$ to satisfy the differential privacy standard [25] for privacy protection.

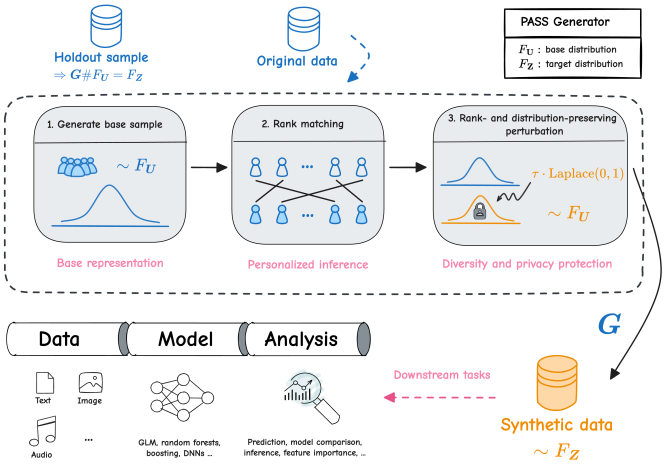


Fig. 1. Flowchart illustrating the PASS approach with rank matching and distribution-preserving perturbation. PASS generates a synthetic sample that closely retains the multivariate ranks of the original sample, ensuring privacy protection. The transport G is applied to align the base distribution with the target distribution (for example, the original distribution).

This discussion leads to the generation scheme of PASS, which comprises three components, transport estimation, rank preservation, and data perturbation:

(1) Sample $\boldsymbol{U} = (\boldsymbol{U}_i)_{i=1}^n$ from the base distribution $F_{\boldsymbol{U}}$;

(2) Compute the permutation map $r(\cdot)$ on $\{1, \ldots, n\}$ to align the multivariate ranks of $(\boldsymbol{U}_{r(i)})_{i=1}^n$ with those of $(T(\boldsymbol{Z}_i))_{i=1}^n$, where T is a transport map from $F_{\boldsymbol{Z}}$ to $F_{\boldsymbol{U}}$. Please see Section III in the supplementary materials for additional details regarding $r(\cdot)$.

(3) Generate $\boldsymbol{Z}' = (\boldsymbol{Z}_i')_{i=1}^n$ by adding noise $(\boldsymbol{e}_i)_{i=1}^n \sim F_{\boldsymbol{e}}$:

$$\boldsymbol{Z}_i' = G(\boldsymbol{V}_i), \quad \boldsymbol{V}_i = W(\boldsymbol{U}_{r(i)}; \boldsymbol{e}_i); i = 1, \ldots, n, \quad (1)$$

where W is a known perturbation function that injects noise to $\boldsymbol{U}$ while preserving the base distribution, that is, $(\boldsymbol{V}_i)_{i=1}^n$ will still be random sample from $F_{\boldsymbol{V}} = F_{\boldsymbol{U}}$, and G is a transport map that pushes $F_{\boldsymbol{V}}$ to $F_{\boldsymbol{Z}}$. An illustration is provided in Fig 1.

Notable is that the equation (1) can be applied to embeddings of original data for dimension reduction, as demonstrated in studies such as [26], [17], [27]. In (1), G and T represent generation and rank preservation, respectively. For simplicity, we estimate G by assuming its inevitability. However, in certain cases, it is advantageous not to impose the invertibility on G while estimating T separately, as in diffusion models. As for perturbation, we can select W to preserve the multivariate ranks of $(\boldsymbol{U}_{r(i)})_{i=1}^n$ by $\boldsymbol{V}_i$, even after adding noise $(\boldsymbol{e}_i)_{i=1}^n$ (see Theorem 1). For example, Section IV in the supplementary materials presents an additive form of W. Regarding the noise distribution $F_{\boldsymbol{e}}$, we typically parametrize it as $\boldsymbol{e} = \tau\boldsymbol{\epsilon}$, with $\tau > 0$ denoting the perturbation size and $\boldsymbol{\epsilon} \sim F_{\boldsymbol{\epsilon}}$ representing a standardized noise distribution. When privacy is not a concern, we can conveniently set $\tau = 0$ and select W as the identity map. Additionally, when personalization and data integration are not the primary focus, as in Section V, we can choose $r(i) = i$; $i = 1, \ldots, n$.

Separation of (G, T) from a Downstream Task. Ideally, we can repurpose the original sample $\boldsymbol{Z}$ to estimate the transports G and T while executing a downstream task. However, this approach is debatable regarding the validity of the downstream analysis [28]. Whereas it offers valid inference for a pivotal statistic $H(\boldsymbol{Z})$, as demonstrated in Theorem 2, it may yield overly optimistic conclusions in post-selection inference [28]. To circumvent this problem, we recommend using an independent holdout sample, usually available from other studies on the same population. For example, training examples for similar images could serve as holdout data to learn the data-generating distribution for inference, as illustrated in Section VI. By separating downstream analysis from estimating G and T, we guarantee the validity of an inference even with finite sample size; see Theorem 2. If holdout data is unavailable, a possible alternative is sample splitting, with one subsample acting as a holdout sample. This method can yield valid conclusions but may compromise statistical power [29].

B. Data-Generating Distribution

Given a holdout sample $\mathbb{S}_h = (\boldsymbol{Z}_i)_{i=1}^{n_h}$, our objective is to construct $\tilde{F}_{\boldsymbol{Z}}$, or equivalently $\tilde{G}$, in order to estimate the data-generating distribution $F_{\boldsymbol{Z}} = F_{\boldsymbol{V}} \circ G^{-1}$. Building on this foundation, PASS generates $\boldsymbol{Z}' = (\boldsymbol{Z}_i')_{i=1}^n$ following $\tilde{F}_{\boldsymbol{Z}}$, as detailed in Lemma 1. Subsequently, we propose employing generative models to reconstruct $\tilde{F}_{\boldsymbol{Z}}$, either *explicitly* by approximating G with an invertible $\tilde{G}$, as in $\tilde{F}_{\boldsymbol{Z}} = F_{\boldsymbol{V}} \circ \tilde{G}^{-1}$ as in normalizing flows [22], [30], [31], or *implicitly* through sampling as in diffusion modeling [17], [18]. Consequently,

large pre-trained generative models can enhance the estimation accuracy of the data-generating distribution.

Explicit Estimation. We suggest estimating G by maximizing a likelihood function $L(G;\mathbb{S}_h)$, which is parameterized through the distribution of $\boldsymbol{V}$. Specifically, we obtain an estimated transport $\tilde{G}$ by

$$\tilde{G} = \underset{G\in\mathcal{F}}{\arg\min}\ (L(G;\mathbb{S}_h) + \lambda P(G)), \tag{2}$$

where $\mathcal{F}$ is a predefined function class, such as normalizing flows, $P(G)$ is a nonnegative penalty function, and $\lambda \geq 0$ is a regularization parameter. In (2), its constrained version can serve the same purpose, as described by [32]. Furthermore, due to the nature of $\tilde{G}$, we can explicitly obtain the analytical form of $\tilde{F}_{\boldsymbol{Z}}$ and the corresponding density, for example, in normalizing flows [22], [30], [31].

Distribution Estimation of a Statistic $H(\boldsymbol{Z})$ by PASS. Given an estimate $\tilde{G}$, we can obtain an estimated distribution $\tilde{F}_{\boldsymbol{Z}} = F_{\boldsymbol{V}} \circ \tilde{G}^{-1}$ when $\tilde{G}$ is invertible. Notably, PASS can generate synthetic samples using $\boldsymbol{Z}' = \tilde{G}(\boldsymbol{V}) \sim \tilde{F}_{\boldsymbol{Z}}$ derived from (1). Based on this, we propose a Monte Carlo method for estimating the CDF $F_{H(\boldsymbol{Z})}$ of any statistic $H(\boldsymbol{Z})$. Specifically, we generate D independent synthetic samples $(\boldsymbol{Z}'^{(d)})_{d=1}^{D}$ using (1), and construct the PASS estimate as an empirical CDF: $\tilde{F}_{H(\boldsymbol{Z}')}(\boldsymbol{x}) = D^{-1}\sum_{d=1}^{D} I(H(\boldsymbol{Z}'^{(d)}) \leq \boldsymbol{x})$ for estimating F_H, where each $\boldsymbol{Z}'^{(d)}$ is from $\tilde{F}_{\boldsymbol{Z}}$ by PASS. Refer to Section IV for statistical guarantee and justification of this approach.

C. Sampling Properties of PASS

Lemma 1 presents the sampling properties of $\boldsymbol{Z}'$ generated by PASS.

Lemma 1. *(Sampling properties of PASS) Given $\boldsymbol{Z}' = (\boldsymbol{Z}'_i)_{i=1}^{n}$ generated from (1) using $\tilde{G}$, assume that $\tilde{F}_{\boldsymbol{Z}}$ is independent of $\boldsymbol{Z} = (\boldsymbol{Z}_i)_{i=1}^{n}$. Then,*

1) *(Within-sample) $\boldsymbol{Z}' = (\boldsymbol{Z}'_i)_{i=1}^{n}$ is an independent and identically distributed (iid) sample of size n according to $\tilde{F}_{\boldsymbol{Z}}$ when $\boldsymbol{Z}$ is independent and identically distributed.*
2) *(Independence) $H(\boldsymbol{Z}')$ is independent of $\boldsymbol{Z}$ for any permutation-invariant H in that $H(\boldsymbol{Z}) = H(\boldsymbol{Z}_\pi)$ with $\boldsymbol{Z}_\pi = (\boldsymbol{Z}_{\pi(i)})_{i=1}^{n}$, where π represents any permutation map on $\{1,\ldots,n\}$.*

Lemma 1 highlights the two advantages of a generated PASS sample $\boldsymbol{Z}'$. First, its iid property is unique and not shared by any resampling approach. Second, the conditional distribution of the PASS statistic $H(\boldsymbol{Z}')$ given $\boldsymbol{Z}$ is the same as its unconditional distribution, a property not shared by existing resampling methods. This aspect is somewhat surprising because the permutation invariance of a test statistic H allows for rank preservation of $\boldsymbol{Z}'$ without imposing dependence between $\boldsymbol{Z}'$ and $\boldsymbol{Z}$. Note that a common test statistic H is invariant concerning the permutation of the sample order for an iid sample [33]. These two aspects ensure that the PASS sample $H(\boldsymbol{Z}')$ accurately represents $H(\boldsymbol{Z})$, leading to a reliable estimate of the distribution of $H(\boldsymbol{Z})$.

III. Perturbation-Assisted Inference

For inference, data scientists often use a statistic $H(\boldsymbol{Z})$ for hypothesis testing or a confidence interval concerning $\boldsymbol{\theta}$ or its functions. Based on the PASS framework described in Section II, we estimate the distribution of $H(\boldsymbol{Z})$, which permits a valid inference through Monte Carlo simulation. We introduce a generative inference framework called Perturbation-Assisted Inference (PAI). PAI involves two independent samples: an inference sample $\mathbb{S} = (\boldsymbol{Z}_i)_{i=1}^{n}$ via $H(\boldsymbol{Z})$ and a holdout sample $\mathbb{S}_h = (\boldsymbol{Z}_i)_{i=1}^{n_h}$ for estimating the generating distribution via PASS. However, if $H(\boldsymbol{Z})$ is pivotal, then holdout and inference samples can be the same, as suggested by Theorem 2.

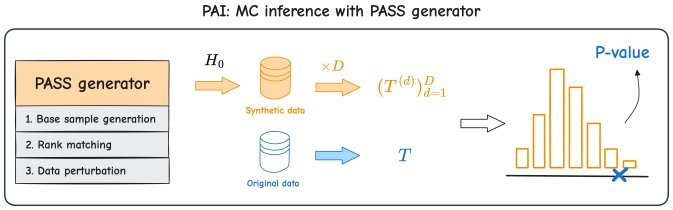


Fig. 2. Estimating the distribution of the test statistic under the null hypothesis (H_0) through Perturbation-Assisted Inference (PAI) using the PASS generator: A Monte Carlo (MC) approach.

To perform a hypothesis test, we proceed as follows:

(1) **Estimation of Null Distribution of $H(\boldsymbol{Z})$.** Under the null hypothesis H_0, we use the holdout sample $\mathbb{S}_h$ for the data-generating distribution of PASS in (1) to estimate the null distribution of a test statistic $H(\boldsymbol{Z})$, which avoids sample reuse. Specifically, we generate D independent copies of synthetic samples $\boldsymbol{Z}'^{(d)}; d = 1,\ldots,D$ via (1), where $\tilde{F}_{\boldsymbol{Z}}(\cdot) = F_{\boldsymbol{Z}}(\cdot\ ;\tilde{\boldsymbol{\theta}}^0)$ with $\tilde{\boldsymbol{\theta}}^0$ being an estimate of $\boldsymbol{\theta}$ under H_0. Then, we compute the empirical distribution $\tilde{F}_{H(\boldsymbol{Z}')}(\boldsymbol{x}) = D^{-1}\sum_{d=1}^{D} I(H(\boldsymbol{Z}'^{(d)}) \leq \boldsymbol{x})$ for any real $\boldsymbol{x}$ as the PASS estimate of F_H given D independent copies of synthetic samples $\{\boldsymbol{Z}'^{(d)}\}_{d=1}^{D}$ via (1), where each sample $\boldsymbol{Z}'^{(d)}$ is from $\tilde{F}_{\boldsymbol{Z}}$ by PASS.

(2) **Inference.** We use the empirical null distribution $F_{H(\boldsymbol{Z}')}$ to compute the rejection probability based on a trained machine learner evaluated on an inference sample $\mathbb{S}_i$ to draw an inference. Moreover, we can convert a test into a confidence set.

Connection with Other Generative Models. PASS is compatible with various generative models for estimating the transport G in (1), which can utilize large pre-trained models to enhance the accuracy of distribution estimation. Unlike other generators, PASS maintains the ranks of an inference sample and incorporates noise to diversify or safeguard the original data.

Connection with Resampling. The resampling approach tailors for low-dimensional numerical data [10], where $F_{\boldsymbol{Z}}$ can be accurately estimated based on $\boldsymbol{Z}$. However, these methods struggle with high-dimensional data due to the curse of dimensionality. Additionally, the resampled data is only conditionally independent, even when $\boldsymbol{Z}$ is independent. For example, in the parametric bootstrap, conditioning on $\boldsymbol{Z} \sim N(\boldsymbol{\mu}, \boldsymbol{I})$, a sample $\boldsymbol{Z}^B \sim N(\hat{\boldsymbol{\mu}}, \boldsymbol{I})$ assuming known

identity covariance matrix $\boldsymbol{I}$ and $\hat{\boldsymbol{\mu}}$ is the estimated mean vector from $\boldsymbol{Z}$. However, for its unconditional distribution, $\mathrm{E}\,\boldsymbol{Z}^B = \mathrm{E}\,\hat{\boldsymbol{\mu}}$ and $\mathrm{Var}\,\boldsymbol{Z}^B = \mathrm{Var}\,\hat{\boldsymbol{\mu}} + \boldsymbol{I}$. This approach can lead to overly optimistic conclusions in post-selection inference as $\hat{\boldsymbol{\mu}}$ depends on a selected model [34], [11], [29].

In contrast, PASS produces an independent sample when a holdout sample is independent of $\boldsymbol{Z}$, as discussed in Lemma 1, which enables valid inference. Moreover, a PASS sample preserves the ranks of an inference sample, facilitating personalization and data integration. Crucially, PASS can generate numerical, unstructured, and multimodal data, such as image-text pairs, allowing PAI to transcend the traditional inference framework and tackle complex problems involving unstructured and multimodal data inference.

IV. Statistical guarantee and justification

Given PASS samples $\boldsymbol{Z}'^{(d)}$ from $\tilde{F}_{\boldsymbol{Z}}$ estimated on an independent holdout sample, we provide a guarantee of validity of PAI by investigating PASS's estimation error of $\tilde{F}_{H(\boldsymbol{Z}')}$, as measured by the Kolmogorov-Smirnov Distance: $\mathrm{KS}(\tilde{F}_{H(\boldsymbol{Z}')}, F_H) = \sup_{\boldsymbol{x}} |\tilde{F}_{H(\boldsymbol{Z}')}(\boldsymbol{x}) - F_H(\boldsymbol{x})|$. Next, we perform the error analysis for non-pivotal inference and pivotal inference.

A. General Inference with Holdout

Theorem 1. *(Validity of PAI) Assume that the estimated data-generating distribution by PASS on a holdout sample $\mathbb{S}_h$ of size n_h is independent of an inference sample $\mathbb{S}$. Moreover, H is a permutation-invariant statistic calculated on $\mathbb{S}$. Then, the reconstruction error of $\tilde{F}_{H(\boldsymbol{Z}')}$ with the MC size D by PASS satisfies: for any small $\delta > 0$, with probability at least $(1-\delta)$,*

$$KS(\tilde{F}_{H(\boldsymbol{Z}')}, F_H) \leq \sqrt{\frac{\log(4/\delta)}{2D}} + |\mathbb{S}| \cdot TV(\tilde{F}_{\boldsymbol{Z}}, F_{\boldsymbol{Z}}), \qquad (3)$$

where $TV(\tilde{F}_{\boldsymbol{Z}}, F_{\boldsymbol{Z}})$ is the total variation distance between the distributions of $\tilde{F}_{\boldsymbol{Z}}$ and $F_{\boldsymbol{Z}}$. Hence, PAI yields a valid test on $\mathbb{S}$ provided that $|\mathbb{S}| \cdot TV(\tilde{F}_{\boldsymbol{Z}}, F_{\boldsymbol{Z}}) \to 0$ as $n_h \to \infty$ and $D \to \infty$.

Remark 1. *Note that $|\mathbb{S}| \cdot TV(\tilde{F}_{\boldsymbol{Z}}, F_{\boldsymbol{Z}}) \to 0$ requires that the holdout size $n_h = |\mathbb{S}_h|$ should be larger than the inference size $n = |\mathbb{S}|$ as $TV(\tilde{F}_{\boldsymbol{Z}}, F_{\boldsymbol{Z}}) \to 0$ at a rate slower than n_h^{-1}.*

Remark 2. *For a diffusion model defined by a d-dimensional Brownian motion, Theorem 5.1 of [35] establishes the error bound between $\tilde{F}_{\boldsymbol{Z}}$ and $F_{\boldsymbol{Z}}$: under regularity conditions:*

$$\mathrm{E}[TV(\tilde{F}_{\boldsymbol{Z}}, F_{\boldsymbol{Z}})] = O(n_h^{-r/(2r+d)} (\log n_h)^{\frac{5d+8r}{2d}}),$$

where the data-generating distribution $F_{\boldsymbol{Z}}$ belongs to the Besov ball $B^r_{p,q}([-1,1]^d, C)$ with radius $C > 0$ and the L_p-modulus of smoothness $r > d(1/p - 1/2)_+$, as measured by the L_q-norm $(p, q > 0)$.

Remark 3. *For normalizing flows, Proposition 1 in the Supplement Material provides an error bound for $TV(\tilde{F}_{\boldsymbol{Z}}, F_{\boldsymbol{Z}})$ expressed in terms of the estimation and approximation errors of a flow, which implies that $TV(\tilde{F}_{\boldsymbol{Z}}, F_{\boldsymbol{Z}}) \to 0$ as $n_h \to +\infty$ when the approximation error tends to zero, which we expect as a flow serves as a universal approximator for complex distributions [36].*

Theorem 1 suggests that the estimation error of the PASS estimate, $\tilde{F}_{H(\boldsymbol{Z}')}$, is governed by two factors: the Monte Carlo (MC) error, $\sqrt{\frac{\log(4/\delta)}{2D}}$, and the estimation error of the data-generating distribution, $\mathrm{TV}(\tilde{\boldsymbol{Z}}, \boldsymbol{Z})$. The MC error diminishes to 0 as the MC size, D, increases, while the latter depends on the estimation method of G in (1) applied to a holdout sample, $\mathbb{S}_h$, which in general goes to 0 as $n_h \to +\infty$. Moreover, PASS can utilize large pre-trained models to boost learning accuracy via knowledge transfer, which we may regard as an increase in n_h.

B. Pivotal Inference without Holdout

This subsection generalizes the previous result to a pivotal $H(\boldsymbol{Z}) = T(\tilde{\boldsymbol{\theta}}, \boldsymbol{\theta})$ for parameter $\boldsymbol{\theta}$, where T is a transformation and $\tilde{\boldsymbol{\theta}}$ is an estimate of $\boldsymbol{\theta}$ based on $\boldsymbol{Z}$. In this situation, PAI does not require a holdout sample, $F_{\boldsymbol{Z}}(\cdot) = F_{\boldsymbol{Z}}(\cdot\,; \boldsymbol{\theta})$ is parametrized by $\boldsymbol{\theta}$, and $\tilde{F}_{\boldsymbol{Z}}(\cdot) = F_{\boldsymbol{Z}}(\cdot\,; \tilde{\boldsymbol{\theta}}) = F_{\boldsymbol{V}} \circ G^{-1}(\cdot\,; \tilde{\boldsymbol{\theta}})$, where $\tilde{\boldsymbol{\theta}}$ can be any estimator of $\boldsymbol{\theta}$ due to the pivotal property and (2) is no longer required. Moreover, given PASS samples $\{\boldsymbol{Z}'^{(d)}\}_{d=1}^D$ from $\tilde{F}_{\boldsymbol{Z}}$ using PASS, the PAI pivotal is $H(\boldsymbol{Z}'^{(d)}) = T(\tilde{\boldsymbol{\theta}}^{(d)}, \tilde{\boldsymbol{\theta}})$, where $\tilde{\boldsymbol{\theta}}^{(d)}$ is an estimate of $\boldsymbol{\theta}$ on $\boldsymbol{Z}'^{(d)}$; $d = 1, \ldots, D$.

Theorem 2. *(Validity of PAI for Pivotal Inference) The conclusion of Theorem 1 holds with $TV(\tilde{\boldsymbol{Z}}, \boldsymbol{Z}) = 0$ provided that $H(\boldsymbol{Z})$ is pivotal for $\boldsymbol{\theta}$. Hence, PAI yields a valid test on $\mathbb{S}$ provided that $D \to \infty$.*

Theorem 2 establishes that the PASS estimate $\tilde{F}_{H(\boldsymbol{Z}')}$ can exactly recover F_H without any estimation error of the data-generating distribution, provided that $H(\boldsymbol{Z})$ is pivotal, even though the estimation error occurs when estimating $F_{\boldsymbol{Z}}$. This result improves the previous findings in [14] and justifies using an inference sample $\mathbb{S}$ alone to estimate $F_{\boldsymbol{Z}}$ for making pivotal inferences.

V. Applications

A. Image Synthesis

In image synthesis, deep generative models have been popular due to the quality of generated synthetic images from original images. Recently, researchers have demonstrated that cascaded diffusion models [37] can generate high resolution with high-fidelity images surpassing BigGan-deep [38] and VQ-VAE2 [39] on the Fréchet inception distance (FID). However, such a comparison lacks uncertainty quantification. Subsequently, we fill the gap to draw a formal inference with the uncertainty quantification for comparing two distributions.

Given two multivariate Gaussian distributions $P_0 = N(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ and $P = N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, the FID score is defined as $\mathrm{FID}(P_0, P) = \|\boldsymbol{\mu}_0 - \boldsymbol{\mu}\|_2^2 + \mathrm{tr}\left(\boldsymbol{\Sigma}_0 + \boldsymbol{\Sigma} - 2\left(\boldsymbol{\Sigma}\boldsymbol{\Sigma}_0\right)^{\frac{1}{2}}\right)$, where $\|\cdot\|_2$ is the L_2-norm, and tr denotes the trace of a matrix. For measuring the quality of generated images, we usually calculate FID on the feature maps extracted via Inception-V3 model [40], a pre-trained vision model that has a great capacity

for extracting visual signals. In our case, P_0 and P would be the original and generated distributions of those feature maps. Here, we test

$$H_0 : \text{FID}(P_0, P) = 0, \quad H_a : \text{FID}(P_0, P) > 0. \tag{4}$$

Then, we construct a test statistic as follows: $T = \text{FID}(\hat{P}_0, \hat{P})$, the empirical FID score between the empirical distribution of test images $\hat{P}_0$ and that of synthesized test images $\hat{P}$ using a trained model, on feature maps from the Inception-V3 model.

To train PASS for PAI inference, we create two independent sets of images denoted by $\mathbb{S}_h = (\boldsymbol{Z}_i)_{i=1}^{n_h}$ and $\mathbb{S} = (\boldsymbol{Z}_i)_{i=n_h+1}^{n_h+n}$ for holdout and inference, where $\boldsymbol{Z}_i$ represents the i-th image. For image generation, we further split the inference sample $\mathbb{S}$ into training and test sets for training and evaluating a generator, which is a common practice. Then, we proceed in three steps. First, we train a PASS generator on a holdout sample $\mathbb{S}_h$ to generate the null distribution under the null that there is no difference between the PASS and the candidate generators under H_0. Second, we train a candidate generator on the training set, with which we evaluate its performance using the test statistic $T = \text{FID}(\hat{P}_0, \hat{P})$ on the test set, where $\hat{P}_0$ and $\hat{P}$ are the estimated distributions from the baseline and the candidate generator. Third, we generate D independent copies of synthetic images $(\boldsymbol{Z}_i'^{(d)})_{i=1}^{n}$ from the null distribution using PASS; $d = 1, \ldots, D$. Then, we compute the corresponding test statistics $(T^{(d)})_{d=1}^{D}$ to obtain the empirical null distribution of T on $\mathbb{S}$, where $T^{(d)} = \text{FID}(\hat{P}_0, \hat{P}^{(d)})$ evaluated on $\mathbb{S}$, and $\hat{P}^{(d)}$ is obtained on $(\boldsymbol{Z}_i'^{(d)})_{i=1}^{n}$. Finally, we compute a two-sided[1] P-value using $(T^{(d)})_{d=1}^{D}$ and T based on $\mathbb{S}$. For detailed steps of this computation, refer to Algorithm 1 in the supplementary materials. An illustrative representation of this procedure can be found in Figure 3.

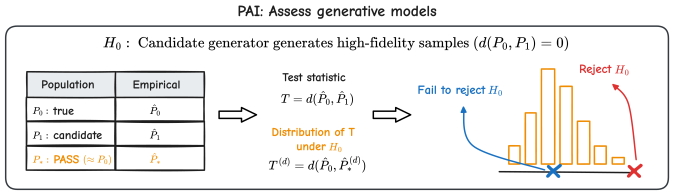


Fig. 3. Illustration of assessing generative models using PAI. $d(\cdot, \cdot)$ represents distributional distance. A test statistic in the tails (red) suggests statistical evidence against the candidate model generating high-fidelity samples. Conversely, a test statistic near the mode (blue) indicates the opposite. For further details, see Algorithm 1 in the supplementary materials.

B. Sentiment Word Inference

Given the unstructured nature of data and the complexity of modeling techniques such as transformer-based models, inferring important words for a learning task can be challenging. In this section, we perform a significance test for the feature relevance of a collection of positive, negative, and neutral words for sentiment analysis of text reviews labeled as positive or negative.

[1]Given that the knowledge is unknown concerning the performance of a candidate generator over the PASS generator, we perform a two-sided test to avoid Type-III error.

Let $\mathcal{W}_M$ be the words of interest. Consider the null hypothesis H_0 and its alternative hypothesis H_a:

$$H_0 : R(f^0) - R(f^0_{\mathcal{W}_M}) = 0, \quad H_a : R(f^0) - R(f^0_{\mathcal{W}_M}) < 0, \tag{5}$$

where R represents the risk under the data distribution, and f^0 and $f^0_{\mathcal{W}_M}$ are two population risk minimizers of decision functions on all words $\mathcal{W}$ and those with masked words of $\mathcal{W}$, respectively. The masked words of $\mathcal{W}$ are those highly attended words of $\mathcal{W}$ by transformer-based models such as BERT [41] on training samples. It is important to note that masking highly attended words of $\mathcal{W}$ is crucial since state-of-the-art embedding models such as BERT can infer words that other embedding models such as Word2Vec [42] are incapable of. For more details, refer to Section VI-B.

PAI constructs a test statistic T using the empirical risk R_n evaluated on an inference sample that is independent of the training sample:

$$T = \frac{R_n(\hat{f}) - R_n(\hat{f}_{\mathcal{W}_M})}{SE(R_n(\hat{f}) - R_n(\hat{f}_{\mathcal{W}_M}))}, \tag{6}$$

where $\hat{f}$ and $\hat{f}_{\mathcal{W}_M}$ are the corresponding trained decision functions, R_n is the empirical risk evaluated on an independent inference sample, and $SE(\cdot)$ denotes the standard error. Refer to Figure 4 for a visual representation of the test statistic.

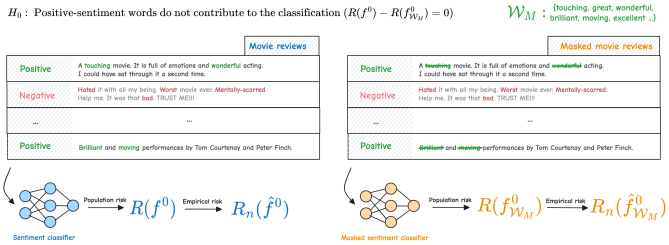


Fig. 4. Illustration of the black-box test statistic [13] employed for assessing feature significance within sentiment classification. If the tested words hold importance for the classification, the risk associated with the masked classifier is expected to be elevated.

For sentiment analysis, we further split the inference sample $\mathbb{S}$ into training and test sets for training and evaluating a classifier, as in Section V-A. Then, we proceed in three steps. First, we train normalizing flows on $\mathbb{S}_h$ to generate the joint null distribution of masked embeddings and sentiment labels under H_0. Second, we train sentiment classifiers $\hat{f}$ and $\hat{f}_{\mathcal{W}_M}$ respectively on the training set and its masked version to get test statistic (6). Third, we generate D datasets on the embedding space $\mathbb{E}^{(d)} = (\boldsymbol{X}_i'^{(d)}, Y_i'^{(d)})_{i=1}^{n}$ from the null distribution estimated by PASS to compute corresponding test statistics $(T^{(d)})_{d=1}^{D}$ on the test set to obtain the empirical null distribution of T, where $\boldsymbol{X}_i'^{(d)}$ and $Y_i'^{(d)}$ represent embedding and corresponding sentiment label, $T^{(d)} = \bar{R}^{(d)}/SE(\bar{R}^{(d)})$ and $\bar{R}^{(d)} = R_n(\hat{f}^{(d)}) - R_n(\hat{f}^{(d)}_{\mathcal{W}_M})$ are calculated on $\mathbb{S}^{(d)}$. Finally, we obtain the P-value of T evaluated on the test set by comparing its value with the empirical distribution of $(T^{(d)})_{d=1}^{D}$, c.f., Algorithm 2 in the supplementary materials for details and Figure 5 for an illustration of this procedure.

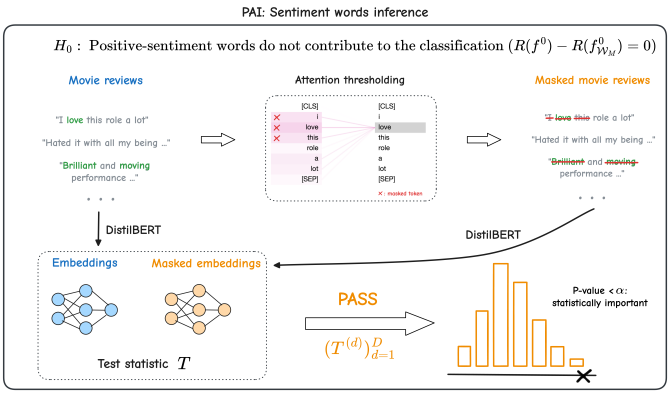


Fig. 5. Depiction of sentiment words inference using PAI. Words under test and their contextual surroundings are masked according to attention thresholds to compute the test statistic; detailed explanation in Section VI-B. PAI operates within the embedding space formulated by DistilBERT; see Algorithm 2 in the supplementary materials for comprehensive steps.

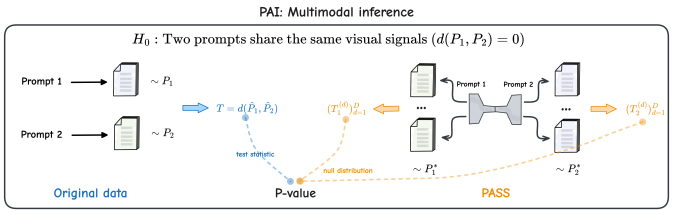


Fig. 6. Illustration of performing multimodal inference using PAI. Simulated test statistics from both prompts using PASS under H_0 are mixed to obtain the estimated null distribution; See Algorithm 3 in the supplementary materials for details.

C. Text-to-Image Generation

Stable Diffusion, a latent diffusion model [26], can generate detailed images given a text prompt. This subsection performs a conditional inference to quantify the statistical certainty of text-to-image generation. Given two text prompts $\boldsymbol{x}^{(1)}$ and $\boldsymbol{x}^{(2)}$, we construct a coherence test for corresponding generated images $\boldsymbol{Y}^{(1)}$ and $\boldsymbol{Y}^{(2)}$ by contrasting their conditional distributions $P(\boldsymbol{y}|\boldsymbol{x}^{(1)})$ and $P(\boldsymbol{y}|\boldsymbol{x}^{(2)})$.

For uncertainty quantification, we use the Inception-V3 embeddings $\mathbf{e}^{(k)}$ [43] for images $\boldsymbol{Y}^{(k)}$; $k = 1, 2$. Under the Gaussian assumption [44], we define the FID score $\text{FID}(P_1, P_2$ between the distributions of two embeddings $\boldsymbol{e}^{(k)}$; $k = 1, 2$, as a coherence measure for hypothesis testing:

$$H_0 : \text{FID}(P_1, P_2) = 0, \quad H_a : \text{FID}(P_1, P_2) > 0. \tag{7}$$

Moreover, we construct $T = \text{FID}(\hat{P}_1, \hat{P}_2)$ as a test statistic, where $\hat{P}_k$ is the corresponding empirical distribution of image embeddings on an inference sample of size n_k; $k = 1, 2$.

For PAI inference, we use the pre-trained Stable Diffusion model [26], a state-of-the-art text-to-image generative model, as our PASS generator. Then, we apply PASS to simulate the null distribution of test statistic T. Given prompt $\mathbf{x}^{(k)}$, for $d = 1, \ldots, D$, PASS generates synthetic samples from P_k, resulting in synthetic embeddings $(\boldsymbol{e}_i^{(k)})_{i=1}^{n_1+n_2}$, of which $(\boldsymbol{e}_i^{(k)})_{i=1}^{n_1}$ and $(\boldsymbol{e}_i^{(k)})_{i=n_1+1}^{n_1+n_2}$ are used to calculate FID score $T_k^{(d)}$, which then renders a sample of the test statistic $(T_d^{(k)})_{k=1,2;d=1,\ldots,D}$ of size $2D$, under the null hypothesis. Under the null that $P_1 = P_2$, there is no difference between the distribution of $\boldsymbol{e}_i^{(1)}$ and that of $\boldsymbol{e}_j^{(2)}$, and thus $T_k^{(d)}$ would be a good estimate for the FID score under H_0, using synthetic samples from PASS. Additionally, $T_k^{(d)}$ also accounts for the symmetry between P_1 and P_2 when calculating FID score. By randomly mixing them, we obtain an estimated null distribution for T; c.f., Algorithm 3 in the supplementary materials for details and Figure 6 for an illustration of this procedure.

VI. Numerical Results

A. Image synthesis

This subsection applies PAI in Section V to hypothesis testing (4) on the quality of image synthesis using the CIFAR-10 benchmark [45]. This dataset consists of $60,000$ images of size $(3 \times 32 \times 32)$ in 10 different classes, with $50,000$ training and $10,000$ and test images.

To synthesize images, we use the CIFAR-10 training set while we use a randomly selected subset of size n of the CIFAR-10 test set for inference. Additionally, we split the CIFAR-10 training set equally into a holdout sample of size $n_h = 25,000$ and another sample of size $n_t = 25,000$, respectively for training a PASS generator (reference) and training competitor generators. In (1), we use a diffusion model (DDPM, [18]) as our baseline generator, denoted by PASS-DDPM. We compare the FID scores of three candidate generators against the baseline generator PASS-DDPM, including DDPM, deep convolutional GAN (DCGAN, [23]), and generative flow (GLOW, [22]); see Fig 7 for samples of the generated images by these generators. To compute the FID scores, we use a 2048-dimensional feature map extracted from an intermediate layer of a pre-trained Inception-V3 model [43] on generated images.

For the hypothesis test in (5), we use PASS-DDPM with $D = 500$ to estimate the null distribution of the FID score and then compute the corresponding P-value for an inference sample, as shown in Table I. Fig 8 illustrates that the empirical null distribution of the FID score varies with the inference sample size n and becomes more concentrated as n increases. This observation highlights the importance of performing uncertainty quantification for the FID score since relying solely on the numerical score could be misleading. Furthermore, we find that DDPM, a generator similar to PASS-DDPM, has a P-value of .78, indicating no significant deviation from the baseline PASS-DDPM. However, DCGAN and GLOW exhibit substantial differences from PASS-DDPM, with P-values of .00 at a significance level of $\alpha = .05$. We confirm this conclusion as the inference sample size increases from $n = 2,050$ to $n = 10,000$.

The experiment result shows that DDPM generators are comparable to the baseline PASS-DDPM, but DCGAN and GLOW show significant differences. It underscores the need to account for uncertainty in the FID score to avoid drawing incorrect conclusions about the generation performance.

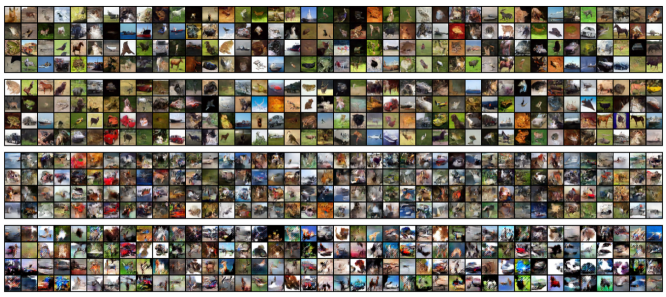


Fig. 7. Generated CIFAR-10 images with dimensions (3, 32, 32), using PASS, DDPM, GLOW, and DCGAN methods (from top to bottom), trained on a dataset of 25,000 images.

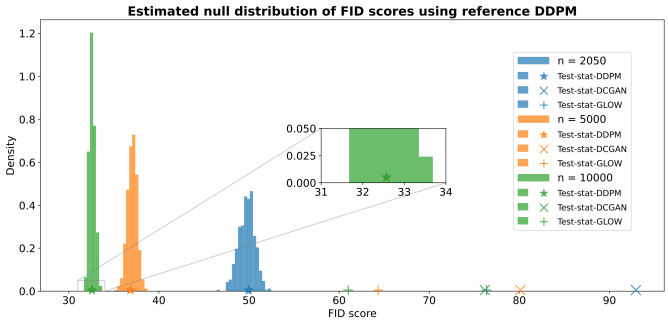


Fig. 8. Empirical FID score distributions with inference sizes $n = 2050, 5000, 10,000$ based on $D = 500$ PASS samples from our PASS-DDPM, represented by colors blue, orange, and green. The FID score is computed using 2048-dimensional features extracted from the Inception-V3 model [43].

TABLE I
FID SCORES AND THEIR P-VALUES FOR TESTING (4), COMPARING THREE GENERATORS, DDPM, DCGAN, AND GLOW, AGAINST THE BASELINE PASS-DDPM. HERE FID SCORES ARE COMPUTED ON 2048-DIMENSIONAL FEATURE MAPS OF THE INCEPTION-V3 MODEL [43] WITH n TEST AND n SYNTHESIZED IMAGES, AND DIST-AVG DENOTES THE AVERAGE FID SCORES OF PASS-DDPM.

Inf size/Generator		DDPM	DCGAN	GLOW	DIST-AVG
$n = 2,050$	FID	49.55	92.93	76.37	49.75
	P-value	.78	.00	.00	
$n = 5,000$	FID	36.83	80.11	64.32	37.04
	P-value	.65	.00	.00	
$n = 10,000$	FID	32.57	76.17	61.01	32.58
	P-value	.72	.00	.00	

B. Sentiment Word Inference

This subsection applies PAI to construct a significance test for quantifying the relevance of sentiment collections of positive, negative, and neutral words, in the context of sentiment classification on the IMDB benchmark [46]. This dataset comprises 50,000 movie reviews labeled as positive or negative. The goal is to determine whether each collection of words contributes significantly to sentiment analysis.

To perform sentiment analysis, we use a pre-trained DistilBERT model [19] to generate text embeddings. Then, we estimate the null distribution of a test statistic using a normalizing flow with a holdout sample of size $n_h = 35,000$, followed by the test (4) in Section V-B with an inference sample of size $n = 5,000$ with a sentiment classifier trained on an independent training set of size $10,000$.

Extraction of Sentiment Words. We extract positive and negative sentiment words of IMDB reviews while treating any remaining words as neutral words based on the opinion lexicon provided by [47]. Then, we extract $|\boldsymbol{W}_M| = 600$ most frequent positive and negative, and neutral words in each collection for inference. Table II displays subsets of these words.

Masking Contexts of Sentiment Words. One main challenge is that BERT-like models [41] have the capability of inferring the context information of sentences via unmarked words due to the use of masked-language modeling for training. As a result, simply masking uni-gram sentiment words does not impact sentiment analysis. To solve this issue, we propose to mask the context of each target word by thresholding attention weights from a pre-trained transformer encoder such that 2% of the context words are masked.

Training via Transfer Learning. To perform sentiment analysis, we construct a classifier by appending a classification head to a pre-trained uncased base DistilBERT model [19], a lighter version of BERT, which permits efficient comprehending of the context. We then fine-tune the model using IMDB review data and obtain fine-tuned embeddings for subsequent tasks. As a result, the model achieves high test accuracy with only a few epochs of fine-tuning.

Learning Embedding Distribution by Normalizing Flows. To train a PASS on the embedding space, we train a RealNVP [31] with affine coupling layer on an independent holdout sample $n_h = 35,000$ to learn the joint distribution of the pair of text embedding and sentiment label. Specifically, we first learn the marginal distribution of sentiment labels and then use normalizing flows to learn the conditional distribution of text embeddings given a sentiment label. The learned flow will be used to emulate the null distribution of the test statistic. For more training details, please refer to Section II-B in the supplementary materials. As Fig 9 suggests, PASS produces an accurate joint null distribution of the word-label pair, evident from the corresponding marginal and conditional distributions given the label.

PAI. We apply PASS to generate $D = 200$ synthetic samples from the null distribution learned from the normalizing flows. Then, we use a training sample of size $n_t = 10,000$ to train a classification model while computing the test statistic on the inference sample of size $n = 5,000$, with the same splitting ratio for all synthetic samples.

Table II and Fig 10 show that positive and negative words have significant P-values of .045 and .015, while neutral words are insignificant with a P-value of .715, at a significance level of $\alpha = .05$. In other words, positive and negative sentiment words, particularly their contexts, are important predictive features for sentiment analysis.

To understand the contribution of PASS for simulating the test statistic null distribution, we note that the joint null distribution of positive, neutral, or negative words does not follow the standard Gaussian with an MC size of $D = 200$, as indicated by Table III. Their distributions differ significantly

TABLE II
DEGREE OF THE IMPORTANCE OF THREE COLLECTIONS OF 600 OF POSITIVE, NEGATIVE, NEUTRAL WORDS, AS MEASURED BY THE P-VALUE AGAINST THE IRRELEVANCE OF EACH COLLECTION BY PAI WITH AN MC SIZE $D = 200$.

	Selected sentiment words	P-value
Positive	*"gratitude, radiant, timely, robust, optimal, thoughtfully, cooperative, calming, assurance, oasis, elegant, remarkable, restored, fantastic, diplomatic, fastest, excellence, precise, brisk, warmly, ..."*	.045
Negative	*"cringed, vomit, excuse, vomiting, fails, ashamed, boring, limp, ridiculous, aground, scrambled, useless, snarl, annoying, bland, unnatural, incorrectly, dire, idiot, leaking, ..."*	.015
Neutral	*"administering, reorganized, curving, gleamed, relinquished, circled, seeded, streamed, curved, scholastic, canning, accommodated, voluntary, cooled, rained, defected, regulated, ousted, straightening, renaming, ..."*	.715

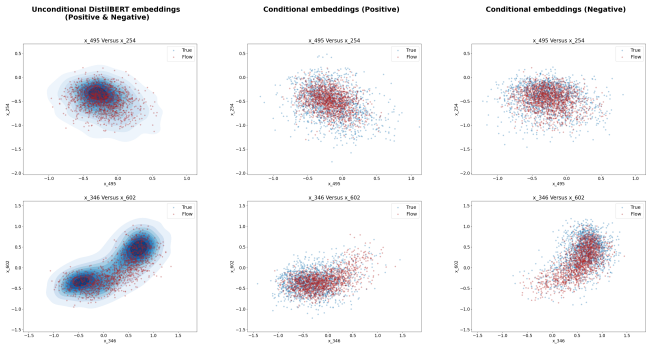


Fig. 9. Two-dimensional projections from null distributions by PASS (blue) from an affine coupling flows trained on a holdout sample of size $n_h = 35,000$ versus the true distribution (red) via 768-dimensional DistilBERT embeddings. The marginal distribution of combined words and conditional distributions for positive and negative reviews are from left to right.

from their asymptotic distributions [13], despite their smooth curves resembling the Gaussian distribution, as shown in Fig 10. As a result, the asymptotic test in [13] is not appropriate in this situation. This result demonstrates the usefulness of PASS when a test statistic's distribution significantly deviates from its asymptotic distribution.

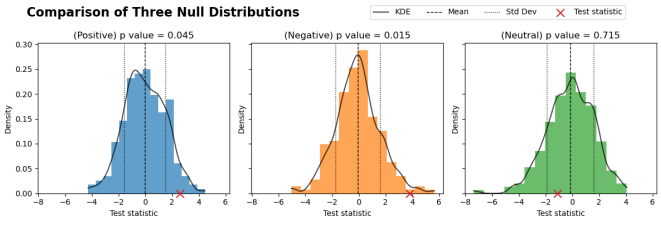


Fig. 10. Empirical null and their kernel smoothed distributions of the test statistic T for positive (blue), negative (orange), and neutral (green) sentiment words, based on PASS with an MC size $D = 200$ for the hypothesis (5). Here, red crosses represent the test statistic's values calculated on an inference sample, while the dashed line and two dotted lines represent the empirical mean and standard error.

C. Text-to-Image Generation

Consider four prompts as follows: Prompt 1 - *"The sun sets behind the mountains"*, Prompt 2 - *"The sun sets behind the mountains"*, Prompt 3 - *"The mountains with sunset behind"*, and Prompt 4 - *"The mountains with a night sky full of shining stars"*. The four prompts have different levels of similarities: Prompts 1 and 2 are identical, Prompt 1 (or 2) is similar to Prompt 3, and Prompt 4 differs from all three above, with the Cosine similarity of 1, .891, .590, and .607 in Table IV. Visually, images from Prompts 1 (or 2) and 3 appear very similar with only slight differences, whereas those from Prompt 4 display stars and look dramatically different, as illustrated in Fig 11. Next, we will confirm the visual impressions through our coherence test in (7).

TABLE III
THE KOLMOGOROV-SMIRNOV TEST FOR THE DISCREPANCY BETWEEN THE TEST STATISTIC'S DISTRIBUTION AND THE STANDARD GAUSSIAN.

	Empirical mean	Std Err	KS test statistic	P-value (two-sided)
Positive	-.028	1.530	.124	.004
Negative	-.083	1.659	.128	.002
Neutral	-.137	1.749	.157	.000

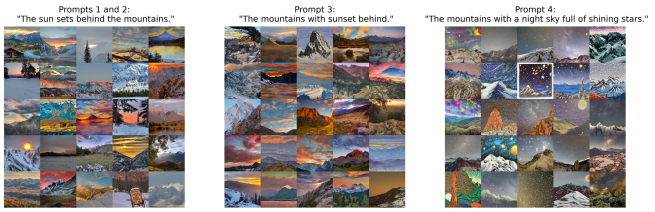


Fig. 11. Generated images given different prompts by Stable Diffusion. The image size is cropped from (512, 512) to (299, 299) to accommodate the input shape for the Inception-V3 model [43].

TABLE IV
COMPARISON OF FOUR PAIRS OF PROMPTS WITH THE COSINE SIMILARITY ON THE CLIP TEXT EMBEDDINGS, THE FID SCORE TEST STATISTIC, AND THE P-VALUE BY PASS WITH $D = 200$, ON 192-DIMENSIONAL EMBEDDINGS FROM THE INCEPTION-V3 MODEL.

	Cosine similarity	FID score	P-value
Prompts 1 and 2 (same)	1.000	.544	.990
Prompts 1 and 3 (similar)	.891	1.010	.124
Prompts 1 and 4 (different)	.590	14.250	.000
Prompts 3 and 4 (different)	.607	14.172	.000

To apply PAI for testing in (7), we construct a PASS generator using a pre-trained stable diffusion model to generate two image sets given two prompts. This pre-trained model is a well-trained state-of-the-art text-to-image model (equivalent to $n_h \to +\infty$). Then, we compute the FID score of 192-dimensional Inception-V3 embeddings between the two sets of images. To simulate the null distribution, we apply PAI to the test statistic with an MC size of $D = 200$ for both image sets, where the effective size of a sample is 400.

Images generated under Prompts 1 and 2, and Prompts 1 and 3, are statistically indistinguishable, given the corresponding P-values of 0.99 and 0.124 at a significance level $\alpha = .05$ in Table IV. In contrast, Prompts 1 and 4, and Prompts 3 and 4 significantly differ in image generation as they have different implications. Moreover, we construct more pairs of prompts to

obtain a spectrum of cosine similarity versus FID score, along with the corresponding test results. As illustrated in Fig 12, a small FID score and a large Cosine similarity imply that two prompts are conceptually equivalent or similar, which can be captured by the test under different significance levels.

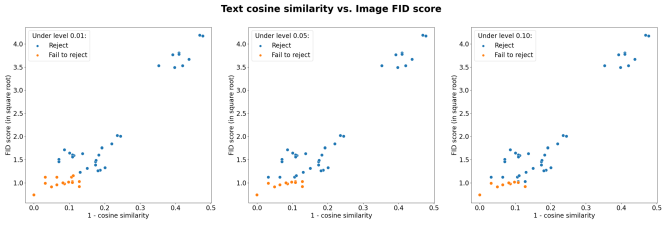


Fig. 12. Pairs of FID score and Cosine similarity on embeddings generated from CLIP versus the FID score (test statistic) computed based on 192-dimensional features from the Inception-V3 model [43], under different significance levels $\alpha = .01, .05, .1$. Each point in the plot represents a pair of prompts.

D. Prediction Interval

We perform a simulation study to evaluate the accuracy and precision of prediction intervals created using PAI with a PASS generator and compare them to those obtained through the conformal method [11]. We use a simulation model where the ground truth is accessible for assessment:

$$Y = 8 + X_1^2 + X_2 X_3 + \cos(X_4) + \exp(X_5 X_6) + 0.1 X_7 + \boldsymbol{\epsilon}, \quad (8)$$

where $\boldsymbol{X} = (X_1, \ldots, X_7)$ follows a uniform distribution over $[0, 1]^7$ (Uniform$(0, 1)^7$), and $\boldsymbol{\epsilon}$ is normally distributed with zero mean and standard deviation $0.4 \times X_1$. We generate $3,200$ samples from (8), dividing them into $3,000$ for training and 200 for testing.

To generate a conditional generative model of $Y|\boldsymbol{X}$, we employ a method suggested by ([48], [49]). Initially, we train a TabDDPM ([50]) as our PASS generator on the training data to model the joint distribution of $(Y, \boldsymbol{X})$. Then, we adjust the reverse process of the diffusion model for conditional generation without re-training. A predictive interval with a coverage level of $1 - \alpha$ can be defined as (l, u), with l and u being the lower $\frac{\alpha}{2}$ and upper $1 - \frac{\alpha}{2}$ quantiles of the conditional distribution, estimated using the MC approach with PAI.

In our experiments, we set $\alpha = 0.05$ and compare the PAI prediction intervals against those from conformal inference. Specifically, for the latter, we split the training dataset further into a modeling sample of $2,400$ and a calibration sample of 600. The former is used to train a CatBoost predictive model [51], while the latter helps construct conformal scores for uncertainty quantification. We evaluate the prediction intervals of both methods on the test sample.

Here, we highlight that the sizes of perturbations do not compromise the validity or accuracy of the learned distribution, due to using distribution-preserving perturbation functions, c.f., (1). This claim is reinforced by the results depicted in Figure 13, demonstrating that the distribution learned by the PASS algorithm remains consistent across various perturbation sizes $\tau \in \{0, 0.2, 0.5, 1\}$, closely matching the true underlying distribution. Additional validation comes from the data presented in Table V, which shows negligible variation in distributional distances under the 1- and 2-Wasserstein distances [2], and Fréchet Inception Distance (FID)[3] for different perturbation sizes, all suggesting comparable generative error rates. In conclusion, the perturbation size only does not affect PAI, which utilizes the MC simulation method.

About 68% of the intervals using PAI are found to be shorter than those obtained via conformal inference, as depicted in Figure 14, where PAI intervals are contrasted with those from conformal inference and the actual values on randomly selected test points. PAI intervals also show a better alignment with the true values, highlighting PAI's effectiveness as a non-parametric inference method.

Furthermore, PAI prediction intervals maintain accurate coverage probabilities. As illustrated in Figure 15, while conformal inference intervals tend to be wider and more conservative, PAI intervals achieve nearly exact coverage: their median coverage probability is 0.95, consistent with the specified level. However, PAI's average coverage probability is slightly lower at 0.9 due to outliers in the underlying model with small variance and some bias in the PASS generator, which slightly mis-aligns the prediction intervals' centers, despite the estimated lengths being close to the actual values.

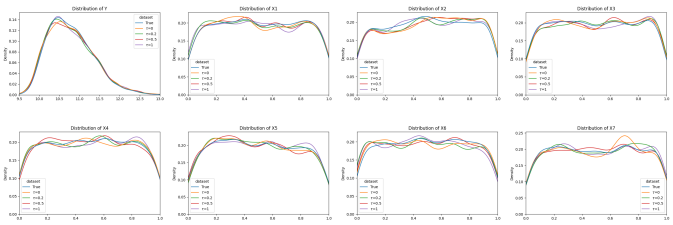


Fig. 13. Kernel density estimates (KDE) of marginal distributions for $(Y, \boldsymbol{X})$ as learned by PASS for perturbation sizes $\tau \in 0, 0.2, 0.5, 1$, compared with those from an independent evaluation sample of size $3,000$ from the underlying true distribution (blue).

TABLE V
DISTRIBUTIONAL DISTANCES BETWEEN THE SYNTHETIC SAMPLE AND AN EVALUATION SAMPLE FROM THE UNDERLYING TRUE DISTRIBUTION, EACH OF SIZE $3,000$. PARENTHETICAL NUMBERS REPRESENT THE STANDARD ERRORS DERIVED FROM REPEATED EXPERIMENTS.

	$\tau = 0.0$	$\tau = 0.2$	$\tau = 0.5$	$\tau = 1.0$
FID	0.024 (0.005)	0.023 (0.005)	0.023 (0.005)	0.024 (0.005)
1-Wasserstein	1.238 (0.004)	1.237 (0.004)	1.238 (0.005)	1.238 (0.005)
2-Wasserstein	1.298 (0.005)	1.296 (0.004)	1.289 (0.005)	1.298 (0.006)

VII. Conclusion

This paper introduces PAI, a novel inference framework grounded in a generative scheme, PASS, which facilitates statistical inference from complex and unstructured data types such as images and texts. PAI addresses the lack of effective uncertainty quantification methods in black-box models like deep neural networks.

[2]https://pythonot.github.io/quickstart.html#computing-wasserstein-distance
[3]Note that FID is 2-Wasserstein distance under Gaussian assumption.

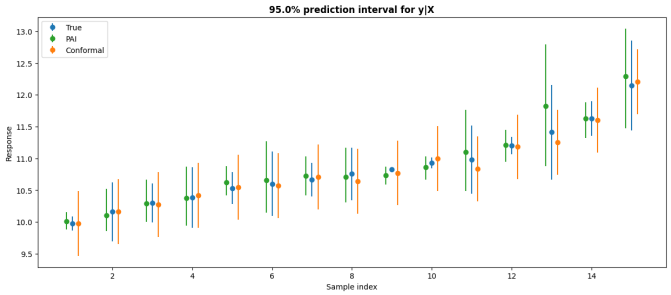


Fig. 14. Comparison of the 95% prediction intervals obtained using PAI (depicted in green), conformal inference (depicted in orange), and the actual observed values (depicted in blue), for a randomly selected subset of 15 data points from the test set.

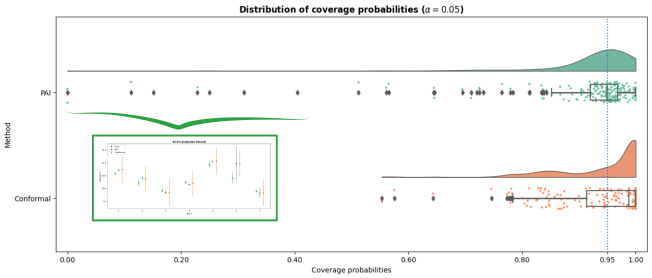


Fig. 15. Comparison of the coverage probability distributions obtained using PAI (in green) and conformal inference (in orange), based on 200 test points. The plot is divided into two sections for each method: the upper section displays the Kernel Density Estimate (KDE) of the probabilities, while the lower section presents the boxplot of the distribution. Additionally, an inset within the plot highlights the prediction intervals for outliers identified by PAI.

The PAI framework, building on PASS, specializes in estimating the distribution of statistics through Monte Carlo experiments, offering a robust method for statistical inference. A key strength of PAI is its theoretical guarantee of inference validity, even in scenarios of scarce data. This paper demonstrates its broad applicability.

Nonetheless, PAI has its limitations. Its primary challenge is the computational demand during Monte Carlo experiments. Also, PAI's performance and accuracy largely depend on the effectiveness of PASS.

On the other hand, PASS utilizes generative models, such as diffusion models and normaliing flows, to mirror the raw data distribution. It can also harness large pre-trained generative models to enhance estimation accuracy. PASS's generator supports data integration and personalization through multivariate rank matching on latent variables, maintaining privacy via data perturbation. Theoretically, we explore PASS's sampling properties, confirming the approximation of latent variable ranks post-data perturbation. Experimental results highlight PASS's superior generation quality.

Our primary goal is to provide researchers with tools that foster reliable and reproducible conclusions from data. These tools have the potential to enhance the credibility and reliability of data-driven discoveries and statistical inferences.

References

[1] E. Gibney, "Is ai fuelling a reproducibility crisis in science," *Nature*, vol. 608, pp. 250–251, 2022.

[2] V. Nemani, L. Biggio, X. Huan, Z. Hu, O. Fink, A. Tran, Y. Wang, X. Zhang, and C. Hu, "Uncertainty quantification in machine learning for engineering design and health prognostics: A tutorial," *Mechanical Systems and Signal Processing*, vol. 205, p. 110796, 2023.

[3] L. Kong, H. Kamarthi, P. Chen, B. A. Prakash, and C. Zhang, "Uncertainty quantification in deep learning," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 5809–5810, 2023.

[4] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, *et al.*, "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," *Information fusion*, vol. 76, pp. 243–297, 2021.

[5] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Advances in neural information processing systems*, vol. 30, 2017.

[6] Y. Song, T. Kim, S. Nowozin, S. Ermon, and N. Kushman, "Pixeldefend: Leveraging generative models to understand and defend against adversarial examples," *arXiv preprint arXiv:1710.10766*, 2017.

[7] C. Liang, W. Wang, J. Miao, and Y. Yang, "Gmmseg: Gaussian mixture based generative semantic segmentation models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 31360–31375, 2022.

[8] L. Kuhn, Y. Gal, and S. Farquhar, "Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation," *arXiv preprint arXiv:2302.09664*, 2023.

[9] Z. Lin, S. Trivedi, and J. Sun, "Generating with confidence: Uncertainty quantification for black-box large language models," *arXiv preprint arXiv:2305.19187*, 2023.

[10] B. Efron, "Bootstrap methods: another look at the jackknife," in *Breakthroughs in statistics*, pp. 569–593, Springer, 1992.

[11] J. Lei, M. G'Sell, A. Rinaldo, R. J. Tibshirani, and L. Wasserman, "Distribution-free predictive inference for regression," *Journal of the American Statistical Association*, vol. 113, no. 523, pp. 1094–1111, 2018.

[12] A. N. Angelopoulos and S. Bates, "A gentle introduction to conformal prediction and distribution-free uncertainty quantification," *arXiv preprint arXiv:2107.07511*, 2021.

[13] B. Dai, X. Shen, and W. Pan, "Significance tests of feature relevance for a black-box learner," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 2, pp. 1898–1911, 2024.

[14] X. Shen, X. Bi, and R. Shen, "Data flush," *Harvard data science review*, vol. 4, no. 2, 2022.

[15] G. Papamakarios, E. T. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, "Normalizing flows for probabilistic modeling and inference.," *J. Mach. Learn. Res.*, vol. 22, no. 57, pp. 1–64, 2021.

[16] Z. Ziegler and A. Rush, "Latent normalizing flows for discrete sequences," in *International Conference on Machine Learning*, pp. 7673–7682, PMLR, 2019.

[17] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International Conference on Machine Learning*, pp. 2256–2265, PMLR, 2015.

[18] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[19] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[20] V. Chernozhukov, D. Chetverikov, M. Demirer, E. Duflo, C. Hansen, W. Newey, and J. Robins, "Double/debiased machine learning for treatment and structural parameters," 2018.

[21] L. Wasserman, A. Ramdas, and S. Balakrishnan, "Universal inference," *Proceedings of the National Academy of Sciences*, vol. 117, no. 29, pp. 16880–16890, 2020.

[22] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," *Advances in neural information processing systems*, vol. 31, 2018.

[23] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[24] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[25] X. Bi and X. Shen, "Distribution-invariant differential privacy," *Journal of Econometrics*, vol. 235, no. 2, pp. 444–453, 2023.

[26] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

[27] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021.

[28] B. Devezer, D. J. Navarro, J. Vandekerckhove, and E. Ozge Buzbas, "The case for formal methodology in scientific reform," *Royal Society open science*, vol. 8, no. 3, p. 200805, 2021.

[29] L. Wasserman and K. Roeder, "High dimensional variable selection," *Annals of statistics*, vol. 37, no. 5A, p. 2178, 2009.

[30] L. Dinh, D. Krueger, and Y. Bengio, "Nice: Non-linear independent components estimation," *arXiv preprint arXiv:1410.8516*, 2014.

[31] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," *arXiv preprint arXiv:1605.08803*, 2016.

[32] X. Shen, W. Pan, Y. Zhu, and H. Zhou, "On constrained and regularized high-dimensional regression," *Annals of the Institute of Statistical Mathematics*, vol. 65, no. 5, pp. 807–832, 2013.

[33] M. Hollander, D. Wolfe, and E. Chicken, *Nonparametric Statistical Methods*. New York: Wiley, 3rd ed., 2013.

[34] J. J. Faraway, "Data splitting strategies for reducing the effect of model selection on inference," *Comput Sci Stat*, vol. 30, pp. 332–41, 1998.

[35] K. Oko, S. Akiyama, and T. Suzuki, "Diffusion models are minimax optimal distribution estimators," *arXiv preprint arXiv:2303.01861*, 2023.

[36] F. Koehler, V. Mehta, and A. Risteski, "Representational aspects of depth and conditioning in normalizing flows," in *International Conference on Machine Learning*, pp. 5628–5636, PMLR, 2021.

[37] J. Ho, C. Saharia, W. Chan, D. J. Fleet, M. Norouzi, and T. Salimans, "Cascaded diffusion models for high fidelity image generation.," *J. Mach. Learn. Res.*, vol. 23, pp. 47–1, 2022.

[38] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," *arXiv preprint arXiv:1809.11096*, 2018.

[39] A. Razavi, A. Van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with vq-vae-2," *Advances in neural information processing systems*, vol. 32, 2019.

[40] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Thirty-first AAAI conference on artificial intelligence*, 2017.

[41] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[42] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proceedings of the Workshop at ICLR*, 2013.

[43] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

[44] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.

[45] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," in *Proceedings of the 2009 conference on computer vision and pattern recognition*, pp. 1378–1385, IEEE, 2009.

[46] A. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pp. 142–150, 2011.

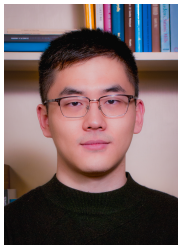
[47] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 168–177, 2004.

[48] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. Van Gool, "Repaint: Inpainting using denoising diffusion probabilistic models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11461–11471, 2022.

[49] H. Zhang, J. Zhang, B. Srinivasan, Z. Shen, X. Qin, C. Faloutsos, H. Rangwala, and G. Karypis, "Mixed-type tabular data synthesis with score-based diffusion in latent space," *arXiv preprint arXiv:2310.09656*, 2023.

[50] A. Kotelnikov, D. Baranchuk, I. Rubachev, and A. Babenko, "Tabddpm: Modelling tabular data with diffusion models," in *International Conference on Machine Learning*, pp. 17564–17579, PMLR, 2023.

[51] A. V. Dorogush, V. Ershov, and A. Gulin, "Catboost: unbiased boosting with categorical features," *arXiv preprint arXiv:1810.11363*, 2018.



Yifei Liu received a B.S. degree in Statistics from University of Science and Technology of China, China, in 2019. He is currently a Ph.D. candidate in Statistics at the University of Minnesota, Minneapolis, USA. His research interests include statistical inference for black-box models, generative models including diffusion models and normalizing flows, statistical learning methods and theory, and deep learning applications.

Rex Shen received a B.S. degree in Mathematics and Computational Science and a M.S. degree in Statistics from Stanford University, USA, in 2022. He is currently a Ph.D. candidate in Statistics at Stanford University. His current research interests span synthetic data generation, Generative AI and its applications.

Xiaotong Shen received a B.S. degree in Applied Mathematics from Peking University, Beijing, China, in 1985, and a Ph.D. degree in Statistics from the University of Chicago, Chicago, USA, in 1991. Currently, He holds the position of the John Black Johnston Distinguished Professor in the School of Statistics at the University of Minnesota, Minneapolis, USA. His research interests include machine learning and data science, high-dimensional inference, non/semi-parametric inference, causal relations, graphical models, explainable Machine Intelligence, personalization, recommender systems, natural language processing, generative modeling, and nonconvex minimization. He is a fellow of the American Statistical Association, the American Association for the Advancement of Science, and the Institute of Mathematical Statistics.