Deep Generative Quantile Bayes

Jungeum Kim **Percy S. Zhai** **Veronika Ročková**
The University of Chicago Booth School of Business

Abstract

We develop a multivariate posterior sampling procedure through deep generative quantile learning. Simulation proceeds implicitly through a push-forward mapping that can transform i.i.d. random vectors samples from the posterior. We utilize Monge-Kantorovich depth in multivariate quantiles to directly sample from Bayesian credible sets, a unique feature not offered by typical posterior sampling methods. To enhance training of the quantile mapping, we design a neural network that automatically performs summary statistic extraction. This additional neural network structure has performance benefits including support shrinkage (i.e. contraction of our posterior approximation) as the observation sample size increases. We demonstrate the usefulness of our approach on several examples where the absence of likelihood renders classical MCMC infeasible. Finally, we provide the following frequentist theoretical justifications for our quantile learning framework: consistency of the estimated vector quantile, of the recovered posterior distribution, and of the corresponding Bayesian credible sets.

1 Introduction

The purpose of this work is to develop a generative sampler from a Bayesian posterior for implicit models whose likelihoods can be accessed only through simulation. We develop a new approach based on quantile learning as an alternative to existing adversarial samplers (Wang and Ročková, 2022). Outside Bayes, quantile learning has been useful across a broad spectrum of practical applications, particularly in contexts where the target distribution exhibits skewness or heavy tails, or when the tail behavior is of primary concern (Yu et al., 2003). Recently, there has been a growing interest inside the statistical community in the application of quantile learning to generative modeling, from both Bayesian (Polson and Sokolov, 2023) and frequentist perspectives (Wang et al., 2024). Our work extends this focus from a one-dimensional to a multi-dimensional regime.

Defining a multivariate quantile presents a challenge due to the non-uniqueness of the mapping from a uniform distribution to the target multivariate distribution. Additionally, while the monotonicity of quantile functions can be ensured for univariate variables, this property does not automatically extend to the multivariate context. These ambiguities can be resolved by considering only those mappings that are a gradient of a convex potential function (Carlier et al., 2016).

Following the approach of Wang and Ročková (2022), we obviate the need for MCMC by training our sampler on simulated data obtained from a likelihood simulator (i.e., a forward sampler) and a prior simulator. However, we take a different approach that learns the quantile mapping directly. The training dataset consists of generated triplets $\{\theta_i, X_i, U_i\}_{i=1}^N$, where d-dimensional θ_i's are simulated from the prior $\pi(\theta)$ in the domain $\Theta \subset \mathbb{R}^d$, X_i's are simulated from the implicit likelihood $L(\cdot \mid \theta)$ on the domain $\mathcal{X} \subset \mathbb{R}^{d_X \times n}$, and $U_i \sim F_U$ are i.i.d. random vectors from a source distribution F_U on $\mathcal{U} \subset \mathbb{R}^d$. We train a conditional deep learning mapping $\hat{Q}_{\theta|X} : \mathcal{U} \to \Theta$ that aims to simulate from the posterior distribution $\pi(\theta \mid X)$ by pushing spherically uniform random vectors through itself. In the univariate case with $d = 1$ and uniform F_U, this strategy corresponds to inverse transform sampling using an estimated quantile function $\hat{Q}_{\theta|X}(\cdot)$. The approach by Polson and Sokolov (2023) performs supervised learning on the triplets using the pinball loss. This method does not simply generalize to the multivariate setting, where the relationship between X_i's and θ_i's is more nuanced and cannot be simply aligned through the pinball loss function. Instead, we aim to

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258. Copyright 2025 by the author(s).

compute the 2-Wasserstein distance between the uniform distribution and the conditional distribution of θ given X, which naturally yields a transport map that can be used for posterior sampling. This approach is different from Wang and Ročková (2022) who iteratively estimate and minimize this Wasserstein distance (See, Section A for more detailed comparisons).

A particular contribution of this paper is to incorporate summary statistics into posterior quantile maps. Motivated by the Noise Outsourcing Lemma, this key technical extension not only brings feasibility to learning the convex potential function, but also enables a broad range of quantile learning methods with theoretical guarantee on their consistency. The vital step of summary statistics learning has been broadly studied in the literature. Long Short-Term Memory (LSTM, Hochreiter (1997)) neural networks, for instance, handles correlated observations and is suitable when the order of the data points matters. Meanwhile, the Deep Sets Neural Network (henceforth DeepSet, Zaheer et al. (2017)) is designed for representing summary statistics for exchangeable data. Our approach integrates both architectures for enhanced summary statistic learning.

Since the true posterior shrinks with the increase of dimensionality, the consistent credible sets should also shrink as n increases, a phenomenon that we call *support shrinkage*. We empirically demonstrate that our method that applies DeepSet exhibits support shrinkage. In addition, in the simulation study where the true posterior is known, the credible sets from our proposed method are close to the oracle sets even when the dimension n is high, which agrees with our theoretical findings. While Jiang et al. (2017) also utilize deep learning for automatic summary statistic learning, their approach is more closely aligned with Polson and Sokolov (2023), as they explicitly apply supervised learning to predict θ_i given X_i.

Our approach learns a push-forward map from a spherical uniform distribution. Therefore, a credible set of arbitrary level $\tau \in (0,1)$ can be obtained by applying the map on the inner ball of radius τ. Compared to traditional Bayesian posterior sampling methods like MCMC or ABC that sample indirectly from posterior draws, no resampling is needed for our approach. A formal definition of the credible set relies on the notion of data depth (Hallin et al., 2021). Our choice is Monge-Kantorovich Depth (Chernozhukov et al., 2017), which can be viewed as a byproduct of the vector quantile, interpreted as a potential function in the quantile space. The equipotential surfaces play a role of quantile contours, which can be equivalently regarded as credible sets.

Numerous studies in the literature have explored the theory of deep quantile regression. White (1992) used the method of sieves to establish the consistency of nonparametric conditional quantile estimators based on single hidden layer feed-forward networks. Padilla et al. (2022) demonstrated consistency results of conditional quantile estimate that minimizes the pinball loss. We build upon a more general framework by Chernozhukov et al. (2017), and demonstrate the asymptotic consistency of the estimated vector quantile. We also prove that the recovered posterior of θ converges to the true posterior in terms of 2-Wasserstein distance.

Our contributions can be summarized as follows:

1. We extend the approach of Polson and Sokolov (2023) from a one-dimensional to a d-dimensional parameter θ using two strategies. The first naive strategy consists of learning d univariate samplers exploiting a chain-rule representation of the joint distribution $\pi(\theta \mid X)$. Given a specific ordering of the variables in θ, we learn these samplers sequentially by adding previous parameters (simulated from the previous univariate posterior samplers) into the training data table for the next parameter in the sequence (See, Section D). Next, we develop our quantile learning approach for generative Bayes.

2. As a byproduct, our multivariate quantile learning method enables direct simulation from *multivariate* Bayesian credible sets. Credible sets are fundamental for Bayesian inference and we can target them directly without imposing any strict geometry structure (which would otherwise be the case using an MCMC or ABC approach). A convex hull of the sampled points then provides an estimate of the credible set.

3. Not all deep learning architectures can be equally useful in generative modeling. We design a particular network for automatic summary statistics learning capable of handling both an increasing number of observations and the dependencies between them. Using this approach, we observe contraction of estimated credible sets with increasing sample size, a phenomenon that we call *support shrinkage*. If the credible sets are converging to the oracle sets based on the true posterior, support shrinkage is a necessary sign.

4. We provide frequentist theory for our multivariate quantile learning approach as well as the initial approach of Polson and Sokolov (2023). Existing theoretical results on unidimensional quantile learning cannot be directly extended to the multidimensional case; our work addresses this gap.

> Specifically, we demonstrate that as $N \to \infty$ (1) the estimated vector quantile function achieves uniform consistency, (2) the recovered posterior converges uniformly to the true posterior in terms of 2-Wasserstein distance, and (3) the Bayesian credible sets converge to oracle sets. These generic theoretical results apply to all quantile learning methods that learn the convex potential function based on the summary statistics using a feed-forward neural network.

The rest of the paper is outlined as follows. In Section 2, we review the recent advances in multi-dimensional quantile learning. Section 3 introduces our generative quantile method. The theoretical study in Section 4 demonstrates the consistency of estimated the vector quantile and the posterior recovered from it. We investigate the empirical performance of the proposed method in Section 5. Finally, we conclude the paper in Section 6.

2 Multivariate Quantile Learning

Quantile learning has a long history of literature in Statistics. A brief review of one-dimensional quantile learning is deferred to Section B.1 (Supplement). As meaningful ordering in $\mathbb{R}^d$ is not obvious, so is extending the concept of quantiles, signs, and ranks from a univariate to a multivariate setting. For a comprehensive discussion of the various notions of a multivariate quantile, we refer to Hallin (2022). Our work builds on one of the more recent optimal transport perspectives.

2.1 Optimal Transport for Quantile Learning

The inherent ambiguity of the multivariate quantile arises from the fact that for a target distribution P, there exist multiple mappings Q such that if U follows some source distribution F_U on the domain $\mathcal{U}$, then $Q(U) \sim P$; any of these maps determines a transport from F_U to P (Koenker, 2017). Recently, Carlier et al. (2016) resolves the ambiguity of the mapping by adding an identifiability condition on Q to be a gradient of a convex function, and call such Q a *vector quantile* function. This additional condition ensures the uniqueness of the mapping and can be viewed as a generalization of the monotonicity requirement for the univariate quantile function, as it leads to $[Q(u) - Q(u')]^\top (u - u') \geq 0$ for all $u, u' \in \mathcal{U}$. The concept of vector quantiles was first introduced in the context of quantile regression (Carlier et al., 2016) and later extended by Chernozhukov et al. (2017) and Hallin et al. (2021) to a general multivariate quantile through which data depth, ranks, and signs can be defined. Here, we follow developments in Chernozhukov et al. (2017).

Going forward in this paper, let the domain of U be a d-dimensional unit Euclidean ball, i.e. $\mathcal{U} = S^d(1)$. The source distribution $U \sim F_U$ is defined as follows: let $U = r\phi$, where $r \sim \text{Unif}([0,1])$, and $\phi \in \mathbb{R}^d$ follows uniform distribution on the $(d-1)$-dimensional unit sphere $\mathcal{S}^{d-1}(1)$, with r and ϕ mutually independent. Consider a target distribution P over $\mathbb{R}^d$. The vector quantile mapping, denoted by Q_P, is then defined as a *gradient of a convex function* which transforms F_U onto the target distribution P. It can be shown that such a map uniquely exists.[1] When $d = 1$, Q_P reduces to the standard quantile function. Furthermore, when P has finite moments of order two, Q_P is the Monge-Kantorovich (MK) optimal transport map from F_U to P that minimizes the expected quadratic cost, i.e.

$$Q_P = \underset{Q:Q\#F_U = P}{\arg\min} \ \mathbb{E}_{U \sim F_U} \|Q(U) - U\|^2. \tag{2.1}$$

Here, $Q\#F_U$ denotes the pushforward measure of F_U under the mapping Q, and $\|\cdot\|$ denotes vector 2-norm. We denote by ψ the convex potential function that defines Q_P through $Q_P = \nabla\psi$. By the Kantorovich duality for the problem (2.1), the potential function ψ (and consequently Q_P) can be expressed through

$$\psi = \underset{\varphi:\mathcal{U}\to\mathbb{R}\cup\{+\infty\}}{\arg\inf} \left(\int \varphi \ \mathrm{d}F_U + \int \varphi^* \ \mathrm{d}P \right), \tag{2.2}$$

where φ^* is a convex conjugate defined by $\varphi^*(\theta) := \sup_{u\in\mathcal{U}}[\theta^\top u - \varphi(u)]$. Due to the celebrated Brenier's theorem (Brenier, 1991), we know that the gradient of ψ in (2.2) provides the optimal transport map, i.e., $\nabla\psi = Q_P$, and this gradient is referred to as the Brenier map. In machine learning literature, apart from quantile perspectives, the convex potential ψ in (2.2) has been learned by input convex neural networks (ICNNs) (Amos et al., 2017) for generative modeling (See, Section B.2).

2.1.1 Conditional Vector Quantile

As the focus of our work is posterior sampling, our interest is more related to *conditional* quantile learning by Carlier et al. (2016). Note that our target function is now $\pi(\theta \mid X)$. A conditional quantile map $Q_{\theta|X}(U)$ is defined as a map that satisfies the following properties: (1) for a fixed $X = x$, $Q_{\theta|X=x}(u)$ is a gradient of a convex function with respect to u, and (2) $Q_{\theta|X}(U) \sim \pi(\theta \mid X)$ for $U \mid X \sim F_U$ (given that U is independent to X, it suffices to have $U \sim F_U$).

[1] Theorem 2.1 in Chernozhukov et al. (2017), where the spherical uniform distribution F_U is regarded as the source distribution F therein. The condition is then F is absolutely continuous with respect to the Lebesgue measure on $\mathbb{R}^d$.

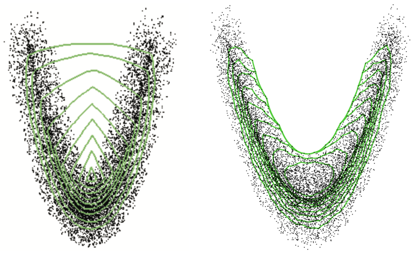

Figure 1: The data depth on a banana shaped distribution. Left: Halfspace depth (Tukey, 1975). Right: Monge-Kantorovich depth (Chernozhukov et al., 2017). The right one aligns more with our natural idea of credible set. Copied from Chernozhukov et al. (2017).

Similar to (2.2), we can find a generative map by $Q_{\theta|X=x}(u) = \nabla_u \psi(u, x)$ (and its inverse map, denoted by $R_{u|X=x}(\theta) = \nabla_\theta \psi^*(\theta, x)$), where

$$\begin{aligned}\psi = \arg\inf_{\varphi} \int \varphi(x, u) F_X(\mathrm{d}x) F_U(\mathrm{d}u) \\ + \int \varphi^*(x, \theta) F_{X,\theta}(\mathrm{d}x, \mathrm{d}\theta),\end{aligned} \tag{2.3}$$

and $\varphi^*(\theta, x) = \sup_{u \in \mathbb{R}^d} [\theta^\top u - \varphi(u, x)]$. Here, F_X is the marginal distribution of X, and $F_{X,\theta}$ is the joint distribution of (X, θ). We can think $\nabla_u \varphi(x, u)$ as a Breinier map extended to a conditional version.

The primal problem of (2.3) is a Monge problem,

$$\min_U \{\mathbb{E}\|\theta - U\|^2 : \; U \sim F_U, \; U \perp\!\!\!\perp X\}, \tag{2.4}$$

which can be seen as a conditional version of (2.1). In Carlier et al. (2017) and Carlier et al. (2016), a relaxed condition on the relationship between X and U and considered instead

$$\min_U \{\mathbb{E}\|\theta - U\|^2 : U \sim F_U, \; \mathbb{E}[X \mid U] = \mathbb{E}[X]\}. \tag{2.5}$$

The additional mean-independence constraint $\mathbb{E}[X \mid U] = \mathbb{E}[X]$ replaces the conditional independence constraint ($U \perp\!\!\!\perp X$) in the Monge problem (2.4). The motivation is a simplified dual problem.

2.1.2 Monge Kantorovich Data Depth

Data depth is a way of giving a weak ordering to a multidimensional data space. Denoting by $D_P(\theta)$ the depth of $\theta \in \mathbb{R}^d$ relative to P, we are given the depth ordering $\geq_{D_P}$ such that $\theta_1 \geq_{D_P} \theta_2$ if and only if $D_P(\theta_1) \geq D_P(\theta_2)$ for $\theta_1, \theta_2 \in \mathbb{R}^d$. The boundary of $\{\theta \in \mathbb{R}^d : D_P(\theta) \geq t\}$ gives us the contour of depth t. The most well-known depth is Tukey's halfspace depth Tukey (1975). To avoid the situation when the depth region is defined outside of the support of the data, Chernozhukov et al. (2017) instead apply Tukey's depth after pushing back P to F_U. See the representative illustration in Figure 1. Here we explain this new depth notion by Chernozhukov et al. (2017), called MK-depth (for the definition of Tukey's halfspace depth, see Chernozhukov et al. (2017)). Denote by ψ^* the convex conjugate of ψ defined in (2.2). Chernozhukov et al. (2017) defines $R_P := \nabla\psi^*$ as the vector rank function, which is the inverse of the quantile function[2], i.e., $R_P = Q_P^{-1}$. Therefore, R_P reverses $\theta \sim P$ back to $U \sim F_U$. The MK depth is defined as the Tukey depth on the reversed distribution. Interestingly, the MK-depth region with probability content $\tau \in (0, 1)$ is identical to $Q_P(S^d(\tau))$. Note that this MK-quantile region is controlled by a single $\tau \in (0, 1)$ and has a nested structure $Q_P(S^d(\tau)) \subset Q_P(S^d(\tau'))$ when $\tau \leq \tau'$. Thus, we may see the MK-quantile region as another generalization of the one-dimensional quantile concept as a function of $\tau \in (0, 1)$.

3 Generative Bayesian Computation

In this section, we extend the conditional vector quantile by Carlier et al. (2016) to adopt summary statistics for Bayesian quantile learning. Then, we present our deep generative Bayes algorithm and implementation, along with the credible set computation.

3.1 Vector Quantile for Generative Bayes

Let $f : \mathcal{X} \to \mathbb{R}^q$ be any feature map such that $f(X)$ is a summary statistic separating X and θ, i.e. $X \perp\!\!\!\perp \theta \mid f(X)$. The dual problem of the relaxed (2.5) has the form

$$\inf_{b,\varphi} \left(\int \varphi(u) F_U(\mathrm{d}u) + \int \tilde{\varphi}(f(x), \theta) F_{X,\theta}(\mathrm{d}x, \mathrm{d}\theta) \right), \tag{3.1}$$

where $\tilde{\varphi}(f(x), \theta) := \max_{u \in \mathcal{U}} [u^\top \theta - \varphi(u) - b(u)^\top f(x)]$, with $b : \mathcal{U} \to \mathbb{R}^q$ and $\varphi : \mathcal{U} \to \mathbb{R}$ being continuous functions. This dual problem, compared to (2.3), enables separation of X and U. In a similar argument to Carlier et al. (2017), given $\mathbb{E}[f(X)] = 0$, for U that solves the primal (2.5) and φ, b that solve (3.1), the following almost-sure relationship holds,

$$\tilde{\varphi}(f(X), \theta) = U^\top \theta - \varphi(U) - b(U)^\top f(X).$$

Thus the potential function takes the form of $\psi(u, x) = \varphi(u) + b(u)^\top f(x)$. If b and φ are smooth, then the partial subdifferential of $\psi(u, x)$ with respect to u becomes degenerate and collapses to the partial gradient $\nabla_u \psi(u, x)$.

[2] due to Brenier-McCann's theorem (Theorem 2.2 in (Chernozhukov et al., 2017))

It is also worth noting that the introduction of arbitrary summary statistics $f(X)$ represents an improvement over the original method by Carlier et al. (2017), which used X in place of $f(X)$. The latter relies on the assumption that the potential function

$$\psi(u,x) = \varphi(u) + b(u)^\top x, \tag{3.2}$$

which is equivalent to the vector quantile $Q_{\theta|X}$ being affine to X. This specific requirement is rarely satisfied in practice. Our analysis, on the other hand, relies on the following assumption.

Assumption 1 (True Quantile Affinity). *There exists some summary statistics $f(X)$ that satisfies $X \perp\!\!\!\perp \theta \mid f(X)$ and*

$$\psi(u,x) = \varphi(u) + b(u)^\top f(x), \tag{3.3}$$

or equivalently, $Q_{\theta|X}$ being affine to $f(X)$.

It is a feasible extension of (3.2), because there exists multiple summary statistics, usually an uncountable class up to some transformations. More importantly, Assumption 1 plays a central role in our methodology, and should not be simply regarded as a mere technical generalization of (3.2). The goal of learning the potential function $\psi(u,x)$ is split into two sub-tasks, learning the summary statistics $f(x)$ and fitting functional coefficients $\varphi(u)$ and $b(u)$ that are convex. This opens the door for a broad class of theoretically-guaranteed quantile learning methods, and turns out to be the cornerstone of our proposed algorithm.

3.2 Deep Generative Quantile Bayes

Consider the parameter space for θ as $\Theta \subset \mathbb{R}^d$ and the data X consists of n observations of d_X-dimensional vectors $X = [x_1, ..., x_n]$, where $x_i \in \mathbb{R}^{d_X}$, and the domain of X is denoted by $\mathcal{X} \subset \mathbb{R}^{d_X \times n}$. We assume that it is possible to simulate (θ, X) from the prior $\pi(\theta)$ and the likelihood $L(X \mid \theta)$ (which could also be implicit, i.e. likelihood-free models). Define $\boldsymbol{\theta} = \{\theta_i\}_{i=1}^N$, $\boldsymbol{X} = \{X_i\}_{i=1}^N$, and $\boldsymbol{U} = \{U_i\}_{i=1}^N$, where $U_i \sim F_U$.

We apply the method of Sun et al. (2022) from the perspective of Wang and Ročková (2022) and Polson and Sokolov (2023). We freshly simulate the training set (mini-batch) for every iteration, which is a set of triples $\{(\theta_i, X_i, U_i)\}_{i=1}^N$.

To train the functions φ, b, and f, we optimize the following objective function

$$\mathcal{L}_1(\varphi, b, f \mid \boldsymbol{X}, \boldsymbol{\theta}, \boldsymbol{U}) = \sum_{i=1}^N \Big(\varphi(U_i) + \max_{j\in\{1,\dots,N\}} \big\{U_j^\top \theta_i - \varphi(U_j) - b(U_j)^\top f(X_i)\big\}\Big), \tag{3.4}$$

which was first designed by Sun et al. (2022) to optimize the dual problem (3.1) in Section 2.1.1. The algorithm is presented in Algorithm 1. The use of the objective function as in (3.4) encourages learning a function $f(X)$ that is linear to the true quantile $Q_{\theta|X}$, based on which the linear coefficients $\varphi(u)$ and $b(u)$ are accurately learned. To ensure that $f(X)$ is a summary statistic, an option is to learn f with DeepSet, the details of which are elaborated in Section 3.3. Meanwhile, φ and b are parametrized using ICNNs as in Sun et al. (2022) to ensure their convexity in u (3 hidden layers of width 512, and CELU activation). Any summary statistic $f(X)$ learned by Algorithm 1 that is not mean-zero will result in b and φ diverging to infinity. To maintain $\mathbb{E}[f(X)] = 0$, we adopt batch normalization as in Sun et al. (2022).

Algorithm 1 Generative Quantile Posterior Sampler

```
Fixed Input: number of iterations $T$, the simulation model $\pi(\theta) \times L(X \mid \theta_i)$
Learnable Input: Learnable networks $\varphi, b$ and $f$
Training
For $t = 1, \dots, T$
    Simulate $\{(\theta_i, X_i)\}_{i=1}^N$ from $\pi(\theta) \times L(X \mid \theta_i)$
    Sample $\{U_i\}_{i=1}^N$ and $U_i \sim F_U$
    Compute $\mathcal{L}_1$ as in (3.4)
    Update $\varphi, b$ and $f$ via a stochastic optimizer
Sampling $\tau$-credible set conditioned on $X = x$
Sample $U \sim F_U$
Compute $\psi(U,x) = \varphi(U) + b(U)^\top f(x)$
Return posterior sample $\nabla_U \psi(U,x)$
```

In our implementation, we use Adam optimizer with default hyperparameter settings, and with a learning rate 0.01. For every epoch (every 100 iterations), we reduce the learning rate multiplicative by 0.99.

3.3 Automatic Learning for Summary Statistics

To learn the feature map $f : \mathcal{X} \to \mathbb{R}^q$ from data to q summary statistics, we consider DeepSet $h_1 : \mathcal{X} \to \mathbb{R}^{q_1}$ and LSTM $h_2 : \mathcal{X} \to \mathbb{R}^{q_2}$. Then, our feature map is defined as $f(X) = [h_1(X), h_2(X)]$, where $q = q_1 + q_2$. Here h_1 is to represent order invariant summary statistics, while for h_2, the order of observation matters. For example, when we know that the data is i.i.d., we set $q_2 = 0$; Otherwise, we can set $q_2 > 0$ to reflect the possible existence of dependence among observations, e.g., sample autocorrelations.

Our design of $f(\cdot)$ through DeepSet and LSTM has a potential for scaling up other deep learning based Bayesian methods including Wang and Ročková (2022) and Kim and Rockova (2023). When the standard fully connected neural network is used, the network size (number of network parameters to optimize) scales up with the input dimension, which is not realistic for a large n. The deep set design was also adopted for neural estimators, e.g., for extreme value analysis (Sainsbury-Dale et al., 2024) and spacial data analysis (Richards et al., 2023).

3.4 Credible Set Computation

Generative quantile posterior learning enables direct sampling from multivariate posterior credible sets. Existing samplers like MCMC and ABC methods compute these sets using posterior draws based on a chosen metric, after committing to certain geometry of the set, e.g. ellipsoid. On the contrary, our approach does not impose any specific restriction on geometric structure, and is able to automatically learn the shape of the credible sets. As proposed by Chernozhukov et al. (2017) and Hallin et al. (2021), the vector quantiles define data depth, from which we can derive the depth region (the deepest set) and quantile contours. Depth region of probability τ then can be used as a credible set of probability τ. Thanks to these desirable properties of the MK depth, we shall see in Section 4 that our approach is asymptotically valid, in the sense that the credible sets converge to the oracle set derived from the true underlying posterior.

4 Theoretical Studies

The simulated training data $\{\theta_i, X_i, U_i\}_{i=1}^N$ consists of i.i.d. observations. However, there is no requirement on the dependence structure within $X_i = (X_{i1}, \cdots, X_{in})^\top$.

Recall that our main goal is simulating from an approximated mapping to the posterior distribution of θ given the observed data vector X. Let $U \sim F_U$ be a random vector that follows the source distribution (as given in Section 2.1) on a d-dimensional unit Euclidean ball $\mathcal{U} = S^d(1)$. We aim to find a mapping $H : \mathcal{U} \times \mathcal{X} \to \Theta$ such that $H(U, X) \sim \pi(\theta \mid X)$. This pursuit is warranted by the Noise Outsourcing Lemma (see Theorem 5.10 in Kallenberg (2002)). In the presence of a summary statistic $f(X)$ such that X and θ are conditionally independent given $f(X)$, this lemma (refined by Bloem-Reddy et al. (2020)) reads as follows:

Lemma 1 (Noise Outsourcing Lemma). *Let $\mathcal{S}$ be a standard Borel space and $f : \mathcal{X} \to \mathcal{S}$ a measurable map. Then $X \perp\!\!\!\perp \theta \mid f(X)$ if and only if there is a measurable function $H : \mathcal{U} \times \mathcal{S} \to \Theta$ such that*

$$(X, \theta) \overset{a.s.}{=} (X, H(U, f(X))), \tag{4.1}$$

where $U \sim F_U$ and $U \perp\!\!\!\perp X$. In particular, $\theta = H(U, f(X))$ has a distribution $\pi(\theta \mid X)$.

A requirement on the source distribution in Lemma 1 is that the support of U must be sufficient to allow $H(U, f(X))$ to attain all values in the support of Θ given the summary statistics $f(X)$. This requirement is satisfied since our source distribution F_U is continuous and has a positive density anywhere in the unit ball $S^d(1)$. Any mapping that satisfies (4.1) can be regarded as a conditional vector quantile function and a conditional generator for $\pi(\theta \mid X)$. Lemma 1 guarantees that such a mapping exists.

In our methodology, the conditional vector quantile $Q_{\theta|X=x}$ stems from a potential function $\psi(u, x)$ that is convex in u. Likewise, the inverse of such a vector quantile, i.e. conditional vector rank denoted by $R_{U|X=x}$ relies on $\psi^*(\theta, x)$, the conjugate of $\psi(u, x)$. Specifically, for each $u \in \mathcal{U}$ and $\theta \in \Theta$,

$$\begin{aligned}\psi(u, x) &= \sup_{\theta \in \Theta} \left[\theta^\top u - \psi^*(\theta, x)\right], \\ \psi^*(\theta, x) &= \sup_{u \in \mathcal{U}} \left[\theta^\top u - \psi(u, x)\right].\end{aligned}$$

The following condition is imposed on the pair of potentials for a rigorous definition of $Q_{\theta|X=x}$ and $R_{U|X=x}$. It ensures that the vector quantile and vector rank are well-defined and are the inverse of each other.

Condition 1 (Similar to Condition (C) in Chernozhukov et al. (2017)). *There exist open, non-empty subsets $\mathcal{U}_0 \subset \mathcal{U}$ and $\Theta_0 \subset \Theta$ such that*

1. *ψ and ψ^* possess gradients $\nabla_u \psi(u, x)$ for all $u \in \mathcal{U}_0$, and $\nabla_\theta \psi^*(\theta, x)$ for all $\theta \in \Theta_0$, respectively,*
2. *the restrictions $\nabla\psi|_{\mathcal{U}_0} : \mathcal{U}_0 \to \Theta_0$ and $\nabla\psi^*|_{\Theta_0} : \Theta_0 \to \mathcal{U}_0$ are homeomorphisms,*
3. *$\nabla\psi|_{\mathcal{U}_0} = (\nabla\psi^*|_{\Theta_0})^{-1}$.*

Under our settings, it is natural to require $\mathcal{U}_0 = \operatorname{int} \mathcal{U}$, i.e. the interior of the unit ball $S^d(1)$, and $\Theta_0 = \operatorname{int} \Theta$. Vector quantiles and vector ranks are thus defined as follows:

$$\begin{aligned}Q_{\theta|X=x}(u) &= \arg\sup_{\theta \in \Theta} \left[\theta^\top u - \psi^*(\theta, x)\right], \quad u \in \mathcal{U}_0; \\ R_{U|X=x}(\theta) &= \arg\sup_{u \in \mathcal{U}} \left[\theta^\top u - \psi(u, x)\right], \quad \theta \in \Theta_0.\end{aligned}$$

By the envelope theorem and Rademacher's theorem (Rademacher, 1919),

$$Q_{\theta|X=x}(u) = \nabla_u \psi(u, x) \quad \text{a.e. on } \mathcal{U}_0, \tag{4.2}$$

$$R_{U|X=x}(\theta) = \nabla_\theta \psi^*(\theta, x) \quad \text{a.e. on } \Theta_0. \tag{4.3}$$

Since the distribution of U is absolutely continuous with respect to the Lebesgue measure on $\mathbb{R}^d$, the results in Brenier (1991) and McCann (1995) show the existence of function $\psi : \mathcal{U} \times \mathcal{X} \to \mathbb{R} \cup \{+\infty\}$ that is convex in u, and that its gradient $\nabla_u \psi$ is unique almost everywhere (see Theorem 2.1 in Chernozhukov et al. (2017)). Similarly, the existence of $\psi^*(\theta, x)$ and the uniqueness of $\nabla_\theta \psi^*$ are also guaranteed.

Recall the affinity assumption (3.3). Our goal is to estimate the coefficients φ, b and the summary statistic f using the generated data flow $\{(X_i, \theta_i, U_i)\}_{i=1}^N$. Specifically,

$$\hat{\psi}_N(u, x) = \hat{\varphi}_N(u) + \hat{b}_N(u)^\top \hat{f}_N(x),$$

where $\hat{\varphi}_N$, $\hat{b}_N$, and $\hat{f}_N$ are learned from a feed-forward neural network. The convex conjugate of $\hat{\psi}_N(u, x)$ is defined as

$$\hat{\psi}_N^*(\theta, x) = \sup_{u \in \mathcal{U}} \{\theta^\top u - \hat{\psi}_N(u, x)\}.$$

Note that the subscripts N corresponds to the estimates stemming from the generated datasets. Our estimate of vector quantile and vector rank functions is defined as follows:

$$\hat{Q}^N_{\theta|X=x}(u) \in \arg\sup_{\theta \in \Theta} \left[\theta^\top u - \hat{\psi}_N^*(\theta, x)\right], \quad u \in \mathcal{U};$$
$$\hat{R}^N_{U|X=x}(\theta) \in \arg\sup_{u \in \mathcal{U}} \left[\theta^\top u - \hat{\psi}_N(u, x)\right], \quad \theta \in \Theta.$$

Thus we have

$$\hat{Q}^N_{\theta|X=x}(u) = \partial_u \hat{\psi}_N(u, x) \quad \text{for } u \in \mathcal{U},$$
$$\hat{R}^N_{U|X=x}(\theta) = \partial_\theta \hat{\psi}^*_N(\theta, x) \quad \text{for } \theta \in \Theta,$$

where ∂ denotes the subdifferential of a convex function.

To quantify the approximation error of $\hat{\psi}_N(u, x)$, we apply the generic result in Farrell et al. (2021) regarding the approximation error of functionals using feed-forward neural networks. The following condition is required.

Condition 2. *There exists an absolute constant $M > 0$, such that*

1. *the true underlying functions satisfy $\|\varphi\|_\infty \le M$, $\|b_k\|_\infty \le M$, $\|f_k\|_\infty \le M$ for all $1 \le k \le q$, and the range $\Theta \subset [-M, M]^d$.*

2. *The estimates of the functions from the feed-forward neural networks are uniformly bounded, i.e. for any $N \in \mathbb{N}$, $\|\hat{\varphi}_N\|_\infty \le 2M$, $\|\hat{b}_{N,k}\|_\infty \le 2M$, $\|\hat{f}_{N,k}\|_\infty \le 2M$ for all $1 \le k \le q$.*

3. *The estimates of the functions from the feed-forward neural networks are uniformly equicontinuous on the compact domain $\mathcal{U}$ or $\mathcal{X}$, i.e. for any $\epsilon > 0$, there exists $\delta > 0$ such that for any $N > 0$, any $u_1, u_2 \in \mathcal{U}$ such that $\|u_1 - u_2\| < \delta$ implies $\|\hat{\varphi}_N(u_1) - \hat{\varphi}_N(u_2)\| < \epsilon$ and $\|\hat{b}_{N,k}(u_1) - \hat{b}_{N,k}(u_2)\| < \epsilon$, and any $x_1, x_2 \in \mathcal{X}$ such that $\|x_1 - x_2\| < \delta$ implies $\|\hat{f}_{N,k}(x_1) - \hat{f}_{N,k}(x_2)\| < \epsilon$.*

These boundedness and compactness assumptions are fairly standard in nonparametrics; see Farrell et al. (2021). The choice of M may be arbitrarily large, and no properties of M is required apart from being finite. We also impose the following condition on the settings of feed-forward networks for the learning of the functions.

Condition 3. *Let the true underlying functions φ, b_k, and f_k for $1 \le k \le q$ lie in classes $\mathcal{F}^\varphi$, $\mathcal{F}^b$, and $\mathcal{F}^f$, respectively. For the feed-forward network classes $\mathcal{F}^\varphi_{DNN}$, $\mathcal{F}^b_{DNN}$, and $\mathcal{F}^f_{DNN}$, let the approximation error be*

$$\epsilon^\varphi_{DNN} := \sup_{\varphi \in \mathcal{F}^\varphi} \inf_{\hat{\varphi}_N \in \mathcal{F}^\varphi_{DNN}, \|\varphi\| \le 2M} \|\hat{\varphi}_N - \varphi\|_\infty;$$
$$\epsilon^b_{DNN} := \max_{1 \le k \le q} \sup_{b \in \mathcal{F}^b} \inf_{\hat{b}_{N,k} \in \mathcal{F}^b_{DNN}, \|b_k\| \le 2M} \|\hat{b}_{N,k} - b_k\|_\infty;$$
$$\epsilon^f_{DNN} := \max_{1 \le k \le q} \sup_{f \in \mathcal{F}^f} \inf_{\hat{f}_{N,k} \in \mathcal{F}^f_{DNN}, \|f_k\| \le 2M} \|\hat{f}_{N,k} - f_k\|_\infty.$$

The network classes are selected such that as $N \to \infty$, all these quantities converge to zero.

This condition assumes that the underlying functions are learned by the feed-forward neural network accurately enough. For example, if the neural network structure is a multi-layer perceptron, a structure known as a good approximator of smooth function, the approximation errors ϵ^φ_{DNN}, ϵ^b_{DNN}, and ϵ^f_{DNN} could have a finite-sample upper bound, given that the true underlying functions belong to certain classes like Sobolev ball; see, e.g. (Giné and Nickl, 2021). The following lemma shows a convergence result for the potential function $\hat{\psi}_N$ and its conjugate, $\hat{\psi}^*_N$.

Lemma 2. *Under Assumption 1, suppose that Conditions 2 and 3 hold. Then almost surely for the generated data flow $\{\theta_i, X_i, U_i\}_{i=1}^N$, the estimated potential function $\hat{\psi}_N(u, x)$ and its conjugate $\hat{\psi}^*_N(\theta, x)$ satisfy the follwing.*

1. *$\hat{\psi}_N(u, x)$ converges uniformly to $\psi(u, x)$ over $\mathcal{U} \times \mathcal{X}$, i.e.*

$$\lim_{N \to \infty} \sup_{(u,x) \in \mathcal{U} \times \mathcal{X}} |\hat{\psi}_N(u, x) - \psi(u, x)| = 0; \quad (4.4)$$

2. *For any compact set $K' \subset \Theta$, $\hat{\psi}^*_N(\theta, x)$ converges*

uniformly to $\psi^*(\theta, x)$ over $K' \times \mathcal{X}$, i.e.

$$\lim_{N\to\infty} \sup_{(\theta,x)\in K'\times\mathcal{X}} |\hat{\psi}^*_N(\theta,x) - \psi^*(\theta,x)| = 0; \quad (4.5)$$

We now present the main theorem of this section, which provides an asymptotic consistency guarantee for the estimated vector quantile $\hat{Q}^N_{\theta|X}$, as well as the recovered posterior $\hat{\pi}^N(\theta \mid X)$. In a slight abuse of notation, denote the 2-Wasserstein distance between the probability measures of the recovered posterior and the true posterior by $W_2(\hat{\pi}^N(\theta \mid X), \pi(\theta \mid X))$.

Theorem 3. *Under Assumption 1, suppose that Conditions 1, 2 and 3 hold. Then almost surely for the generated data flow $\{\theta_i, X_i, U_i\}_{i=1}^N$, the following consistency results hold.*

1. *For any closed subset $K_U \subset \mathcal{U}_0$ and $K_X \subset \mathcal{X}$ as $N \to \infty$,*

$$\sup_{u\in K_U, x\in K_X} \|\hat{Q}^N_{\theta|X=x}(u) - Q_{\theta|X=x}(u)\| \to 0. \quad (4.6)$$

2. *Furthermore, for any closed subset $K_X \subset \mathcal{X}$,*

$$\sup_{x\in K_X} W_2(\hat{\pi}^N(\theta \mid X = x), \pi(\theta \mid X = x)) \to 0. \quad (4.7)$$

Note that when $\mathcal{X}$ itself is closed, the supremum over $x \in K_X$ can be written as over $x \in \mathcal{X}$.

A key corollary of Theorem 3 is the consistency of Bayesian credible sets. Given any $\tau \in (0,1)$, denote the d-dimensional ball with radius τ as $S^d(\tau)$. Recall Section 2.1.2 that the Bayesian credible sets are constructed from the estimated Monge-Kantorovich vector quantile as follows,

$$\hat{C}^N_\tau(\theta \mid X = x) := \hat{Q}^N_{\theta|X=x}(S^d(\tau)), \quad (4.8)$$

while under the true posterior, the oracle set is defined as

$$C_\tau(\theta \mid X = x) := Q_{\theta|X=x}(S^d(\tau)). \quad (4.9)$$

The following corollary verifies that the credible set $\hat{C}^N_\tau(\theta \mid X = x)$ converges to $C_\tau(\theta \mid X = x)$, in terms of Hausdoff distance,

$$d_H(A,B) = \max\left\{\sup_{a\in A}\inf_{b\in B}\|a-b\|, \sup_{b\in B}\inf_{a\in A}\|a-b\|\right\}.$$

Corollary 4. *Suppose that the conditions of Theorem 3 hold. Then the generated data flow $\{X_i, \theta_i, U_i\}_{i=1}^N$ almost surely yields an estimated quantile $\hat{Q}^N_{\theta|X}$ such that for any $\tau \in (0,1)$, the Bayesian credible set in (4.8) satisfies*

$$\sup_{x\in K_X} d_H(\hat{C}^N_\tau(\theta \mid X = x), C_\tau(\theta \mid X = x)) \to 0 \quad (4.10)$$

as $N \to \infty$, for any closed subset $K_X \subset \mathcal{X}$.

An important remark is related to support shrinkage. As the number of observations n increases, contraction occurs on the true underlying posterior, leading to a contraction of the oracle sets. If all the assumptions we have made so far are satisfied, then Corollary 4 implies that the Bayesian credible sets should shrink similarly as the oracle sets do. The violation of these assumptions may be the reason why support shrinkage was not observed in many previous methods, a sign of the credible sets not converging to the oracle sets.

The theoretical analysis in this section is not particularly tailored to our method in Section 3, but rather more generic. As a supplement to the discussion in Section 3.1, we would like to emphasize that Assumption 1 is motivated by Lemma 1, the Noise Outsourcing Lemma. On a broader scale, the wide range of quantile learning methods enabled by this assumption can be theoretically consistent, as long as they learn the summary statistics $f(X)$ and convex functions $\varphi(u)$ and $b(u)$ simultaneously, and the technical conditions are satisfied.

5 Numerical Studies

5.1 Gaussian Conjugate Simulation

Consider a normal-inverse gamma model, where $X \mid \mu, \sigma^2 \sim N(\mu, \sigma^2)$ with priors $\nu_0\sigma_0^2/\sigma^2 \sim \chi^2(\nu_0)$ and $\mu \mid \sigma^2 \sim N(\mu_0, \sigma^2/\kappa)$. We choose this model, as it can be seen as one of the most standard posterior sampling examples, for which we know the true posterior distribution. We set $\mu_0 = 0$, $\sigma_0 = 1$, $\kappa = 2$, and $\nu_0 = 25$. Here, we increased n using a DeepSet feature extractor for a few chosen $X = x$ values. With the DeepSet feature network (order-invariant network design), we can see that our method scales with increasing n values. We highlight that in Figure 2 (the second row), the support shrinkage (contraction of the estimated posterior contour sets) is clearly observed with an increasing n for x relatively near the origin. In this figure, we can also see the effect of using DeepSet in comparison to having no feature extractor ($f(x) = x$) or insufficient statistics ($f(x) = \bar{x}$). In Section E in Appendix, we provide more details of the experiment, comparison with B-GAN (Wang and Ročková, 2022) and the Autoregressive method, the effect of the network choice for $n = 2$.

5.2 Brock Hommes Model

Brock and Hommes (1998) developed an agent-based model to simulate asset trading on an artificial stock market, capturing interactions among heterogeneous traders who follow various trading strategies. The Brock and Hommes model is one of the most foundational economic agent-based models and has been

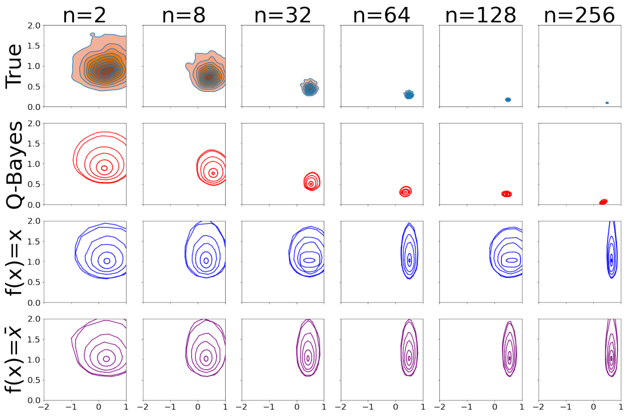

Figure 2: The credible set for the Gaussian example with increasing n for a very small x ($x = 0.5$).

widely used due to its simplicity, while effectively incorporating heterogeneous agents. Recently, Platt (2020) applied this model to evaluate economic agent-based model calibrations. The model is

$$
\begin{aligned}
x_{t+1} &= \frac{1}{R}\sum_{h=1}^{H} n_{h,t+1}(g_h x_t + b_h) + \epsilon_{t+1},\\
n_{h,t+1} &= \frac{\exp(\beta U_{h,t})}{\sum_{h=1}^{H}\exp(\beta U_{h,t})},\\
U_{h,t} &= (x_t - R x_{t-1})(g_h x_{t-2} + b_h - R x_{t-1}),
\end{aligned}
$$

where $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$, and R, β, σ are parameters. Following Platt (2020), we set $\beta = 120$, $H = 4$, $R = 1.01$, $\sigma = 0.04$, $g_1 = b_1 = b_4 = 0$, and $g_4 = 1.01$. Additionally, as in Dyer et al. (2024), we estimate the posterior $p(\theta \mid \mathbf{y})$, where $\theta = (g_2, b_2, g_3, b_3)$, $\mathbf{y} := (y_1, y_2, \ldots, y_T) \sim p(\mathbf{x} \mid \theta^*)$ represents the pseudo-observation, with $T = 100$, and $\theta^* = (g_2^*, b_2^*, g_3^*, b_3^*) = (0.9, 0.2, 0.9, -0.2)$ as the parameters used to generate $\mathbf{y}$. The priors are specified as $g_2, b_2, g_3 \sim \mathcal{U}(0, 1)$, while $b_3 \sim \mathcal{U}(-1, 0)$. For further details on the model, including its interpretation, we refer readers to Platt (2020). The results of our method on this data are presented in Figure 3, which visualizes the posterior contour sets estimated from 10,000 generated samples. It is observed that these contour sets do not overlap. However, as this is an approximation, some overlap could occur. The parameters are contained within the 50% credible set across all dimensions (the innermost contour).

In Section G in Appendix, we also present a comparison with other methods such as the standard ABC (rejection ABC) and sequential Monte Carlo ABC (SMC-ABC, Sisson et al. (2007)), which show the competency of our method in terms of the quality of samples and computational time.

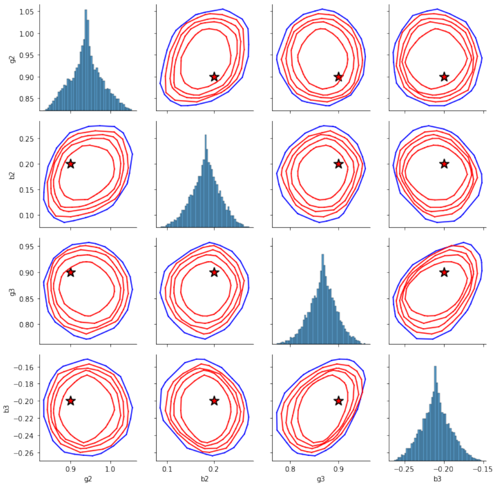

Figure 3: The Brienier map results on Brock Hommes Model, with convex hull corresponding to $\tau \in [0.5, 0.6, 0.7, 0.8, 0.9]$ (red) and $\tau = 1$ (blue). Red stars: the true parameters.

6 Concluding Remarks

This paper develops an implicit sampler from a posterior distribution over multivariate parameters through quantile learning. This methods scales with the number of (potentially dependent) observations and exhibits *support shrinkage*, i.e. shrinkage of the posterior approximation with n. In addition, we provide a tool for estimating contours of the posterior approximation (including posterior credible sets) without imposing a strong geometric structure. Such a structure would be required for multivariate credible sets with more traditional sampling methods such as MCMC or ABC. It is important to note, however, that our method fundamentally differs from both ABC and MCMC approaches. Once our posterior generator is trained, it can be applied to any realization of datasets without the need for retraining. In contrast, both MCMC and ABC must be re-run for each new dataset, which significantly increases their computational burden in practical applications. This reusability makes our approach particularly advantageous for scenarios where multiple datasets need to be analyzed efficiently.

Our work focuses on models with intractable likelihoods and low-dimensional continuous parameter spaces. It would be interesting to extend this work to discrete parameter spaces. The training data is not tailored to the observed data X_0. This could be improved by constructing a dataset with samples more similar to X_0, such as using the approach in O'Hagan et al. (2024), which assigns importance weights to training observations. These weights would then be incorporated into the learning criterion.

Acknowledgments

The authors sincerely acknowledge Nick Polson and Sehwan Kim for their valuable discussions and insights. The authors gratefully acknowledge the support from the National Science Foundation (Grant number DMS 1944740).

References

Amos, B., L. Xu, and J. Z. Kolter (2017). Input convex neural networks. In *International Conference on Machine Learning*, pp. 146–155. PMLR.

Arjovsky, M., S. Chintala, and L. Bottou (2017). Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223. PMLR.

Bloem-Reddy, B., Y. Whye, et al. (2020). Probabilistic symmetries and invariant neural networks. *Journal of Machine Learning Research 21*(90), 1–61.

Brenier, Y. (1991). Polar factorization and monotone rearrangement of vector-valued functions. *Communications on pure and applied mathematics 44*(4), 375–417.

Brock, W. A. and C. H. Hommes (1998). Heterogeneous beliefs and routes to chaos in a simple asset pricing model. *Journal of Economic dynamics and Control 22*(8-9), 1235–1274.

Carlier, G., V. Chernozhukov, and A. Galichon (2016). Vector quantile regression: an optimal transport approach.

Carlier, G., V. Chernozhukov, and A. Galichon (2017). Vector quantile regression beyond the specified case. *Journal of Multivariate Analysis 161*, 96–102.

Chernozhukov, V., A. Galichon, M. Hallin, and M. Henry (2017). Monge–Kantorovich depth, quantiles, ranks and signs.

Dabney, W., G. Ostrovski, D. Silver, and R. Munos (2018). Implicit quantile networks for distributional reinforcement learning. In *International conference on machine learning*, pp. 1096–1105. PMLR.

Diederik, P. K. (2014). Adam: A method for stochastic optimization. *(No Title)*.

Dyer, J., P. Cannon, J. D. Farmer, and S. M. Schmon (2024). Black-box Bayesian inference for agent-based models. *Journal of Economic Dynamics and Control 161*, 104827.

Farrell, M. H., T. Liang, and S. Misra (2021). Deep neural networks for estimation and inference. *Econometrica 89*(1), 181–213.

Giné, E. and R. Nickl (2021). *Mathematical foundations of infinite-dimensional statistical models*. Cambridge university press.

Gneiting, T. and A. E. Raftery (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association 102*(477), 359–378.

Gouttes, A., K. Rasul, M. Koren, J. Stephan, and T. Naghibi (2021). Probabilistic time series forecasting with implicit quantile networks. *arXiv preprint arXiv:2107.03743*.

Gulrajani, I., F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville (2017). Improved training of Wasserstein GANs. *Advances in neural information processing systems 30*.

Hallin, M. (2022). Measure transportation and statistical decision theory. *Annual Review of Statistics and Its Application 9*, 401–424.

Hallin, M., E. Del Barrio, J. Cuesta-Albertos, and C. Matrán (2021). Distribution and quantile functions, ranks and signs in dimension d: A measure transportation approach.

Hochreiter, S. (1997). Long short-term memory. *Neural Computation MIT-Press*.

Huang, C.-W., R. T. Chen, C. Tsirigotis, and A. Courville (2020). Convex potential flows: Universal probability distributions with optimal transport and convex optimization. *arXiv preprint arXiv:2012.05942*.

Jiang, B., T.-y. Wu, C. Zheng, and W. H. Wong (2017). Learning summary statistic for approximate bayesian computation via deep neural network. *Statistica Sinica*, 1595–1618.

Kallenberg, O. (2002). *Foundations of modern probability* (2nd ed.). Springer Science & Business Media.

Kim, J. and V. Rockova (2023). Deep bayes factors. *arXiv preprint arXiv:2312.05411*.

Koenker, R. (2017). Quantile regression: 40 years on. *Annual review of economics 9*, 155–176.

Koenker, R. and Z. Xiao (2006). Quantile autoregression. *Journal of the American statistical association 101*(475), 980–990.

Makkuva, A., A. Taghvaei, S. Oh, and J. Lee (2020). Optimal transport mapping via input convex neural networks. In *International Conference on Machine Learning*, pp. 6672–6681. PMLR.

Matheson, J. E. and R. L. Winkler (1976). Scoring rules for continuous probability distributions. *Management science 22*(10), 1087–1096.

McCann, R. J. (1995). Existence and uniqueness of monotone measure-preserving maps.

O'Hagan, S., J. Kim, and V. Rockova (2024). Tree bandits for generative Bayes. *arXiv preprint arXiv:2404.10436*.

Ostrovski, G., W. Dabney, and R. Munos (2018). Autoregressive quantile networks for generative modeling. In *International Conference on Machine Learning*, pp. 3936–3945. PMLR.

Padilla, O. H. M., W. Tansey, and Y. Chen (2022). Quantile regression with relu networks: Estimators and minimax rates. *Journal of Machine Learning Research 23*(247), 1–42.

Petzka, H., A. Fischer, and D. Lukovnicov (2017). On the regularization of Wasserstein GANs. *arXiv preprint arXiv:1709.08894*.

Platt, D. (2020). A comparison of economic agent-based model calibration methods. *Journal of Economic Dynamics and Control 113*, 103859.

Polson, N. G. and V. Sokolov (2023). Generative AI for Bayesian computation. *arXiv preprint arXiv:2305.14972*.

Rademacher, H. (1919). Über partielle und totale differenzierbarkeit von funktionen mehrerer variabeln und über die transformation der doppelintegrale. *Mathematische Annalen 79*(4), 340–359.

Raynal, L., J.-M. Marin, P. Pudlo, M. Ribatet, C. P. Robert, and A. Estoup (2019). Abc random forests for Bayesian parameter inference. *Bioinformatics 35*(10), 1720–1728.

Richards, J., M. Sainsbury-Dale, A. Zammit-Mangion, and R. Huser (2023). Neural bayes estimators for censored inference with peaks-over-threshold models. *arXiv preprint arXiv:2306.15642*.

Rockafellar, R. T. and R. J.-B. Wets (2009). *Variational analysis*, Volume 317. Springer Science & Business Media.

Sainsbury-Dale, M., A. Zammit-Mangion, and R. Huser (2024). Likelihood-free parameter estimation with neural bayes estimators. *The American Statistician 78*(1), 1–14.

Sisson, S. A., Y. Fan, and M. M. Tanaka (2007). Sequential monte carlo without likelihoods. *Proceedings of the National Academy of Sciences 104*(6), 1760–1765.

Stanczuk, J., C. Etmann, L. M. Kreusser, and C.-B. Schönlieb (2021). Wasserstein GANs work because they fail (to approximate the wasserstein distance). *arXiv preprint arXiv:2103.01678*.

Sun, J., D. Jiang, and Y. Yu (2022). Conditional generative quantile networks via optimal transport. In *ICLR Workshop on Deep Generative Models for Highly Structured Data*.

Taghvaei, A. and A. Jalali (2019). 2-Wasserstein approximation via restricted convex potentials with application to improved training for GANs. *arXiv preprint arXiv:1902.07197*.

Tanaka, A. (2019). Discriminator optimal transport. *Advances in Neural Information Processing Systems 32*.

Tukey, J. W. (1975). Mathematics and the picturing of data. In *Proceedings of the International Congress of Mathematicians, Vancouver, 1975*, Volume 2, pp. 523–531.

Villani, C. (2021). *Topics in optimal transportation*, Volume 58. American Mathematical Soc.

Wang, S., M. Shin, and R. Bai (2024). Generative quantile regression with variability penalty. *Journal of Computational and Graphical Statistics* (just-accepted), 1–21.

Wang, Y. and V. Ročková (2022). Adversarial Bayesian simulation. *arXiv preprint arXiv:2208.12113*.

Wen, R. and K. Torkkola (2019). Deep generative quantile-copula models for probabilistic forecasting. *arXiv preprint arXiv:1907.10697*.

White, H. (1992). Nonparametric estimation of conditional quantiles using neural networks. In *Computing Science and Statistics: Statistics of Many Parameters: Curves, Images, Spatial Models*, pp. 190–199. Springer.

Yu, K., Z. Lu, and J. Stander (2003). Quantile regression: Applications and current research areas. *Journal of the Royal Statistical Society Series D: The Statistician 52*(3), 331–350.

Zaheer, M., S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola (2017). Deep sets. *Advances in neural information processing systems 30*.

Zeng, Z. and T. Wang (2022). Neural copula: A unified framework for estimating generic high-dimensional copula functions. *arXiv preprint arXiv:2205.15031*.

Appendix

A Comparison to Adversarial Bayesian Simulation

Our approach is related to adversarial Bayesian simulation, or Bayesian GAN (B-GAN), by Wang and Ročková (2022). Like B-GAN, our method bypasses the need for posterior sampling by learning a generative joint distribution through forward simulation. While B-GAN employs the 1-Wasserstein GAN framework, our work extends this to the 2-Wasserstein GAN framework, paralleling the relationship between W-GAN (Arjovsky et al., 2017) and 2-Wasserstein GAN (Taghvaei and Jalali, 2019). In the original W-GAN (using the 1-W distance), the critic is trained to compute the distance between the current generating distribution and the data distribution. As the generator is trained to minimize this distance, the critic is iteratively optimized uner an adversarial learning process. In practice, the critic may not be fully optimized due to the iterative adversarial nature (Stanczuk et al., 2021). Taghvaei and Jalali (2019) instead adopt the 2-Wasserstein distance, for which computing the distance only one time provides the generator, which is the derivative of the potential function (the Brenier map). Therefore, the need for iterative distance reduction is eliminated. If the potential function function is designed with fully input-convex networks, no additional regularization to maintain convexity is needed. Conversely, for the original WGAN, ensuring that the critic remains Lipschitz continuous with a constant of 1 is a challenging problem, which has motivated many subsequent works (Gulrajani et al., 2017; Petzka et al., 2017). Another advantage of extending from the 1-Wasserstein to the 2-Wasserstein framework is the ability to separately design a feature map f as described in (3.4), which can be treated as the condition and summary statistics. For the 1-Wasserstein framework, an interesting direction for future research would be to develop an algorithm where both the critic and the generator are conditioned on a learnable feature.

B Additional Literature Review

B.1 One-Dimensional Quantile Learning

For the one-dimensional case, many generative approaches walk around the pinball loss defined by $\Lambda_\tau(q,z) = (\tau - \mathcal{I}_{(z<q)})(z-q)$. Denote the cdf by $F_z(\tau) = P(Z < \tau)$. The key property of this loss is that

$$\arg\min_q \mathbb{E}_{z\sim F_z}[\Lambda_\tau(q,z)] = F_Z^{-1}(\tau).$$

The generative models are trained by minimizing the continuous ranked probability score (CRPS, Matheson and Winkler (1976)), which is the pinball loss integrated over all quantile levels $\tau \in [0,1]$ defined by

$$\mathrm{CRPS}(F^{-1}, z) = \int_0^1 2\Lambda_\tau(F^{-1}(\tau), z)\mathrm{d}\tau.$$

The model trained by it is sometimes called "implicit" quantile function, because τ is not fixed, but also fed into the model (Dabney et al., 2018). CRPS is a proper scoring rule (Gneiting and Raftery, 2007), i.e., $\int g(z)\mathrm{CRPS}(G^{-1}, z)\mathrm{d}z \le \int g(z)\mathrm{CRPS}(F^{-1}, z)\mathrm{d}z$, for any distributions F and G (with g being the density of G).

In the context of quantile regression, where the conditional $F^{-1}_{y|X}(\tau)$ is pursued, Wang et al. (2024) noticed that the trained model $\hat{F}$ may learn to output $\hat{F}^{-1}_{y|X}(\tau) = y$ for all $\tau \in [0,1]$. We think this happens because for each X observation, only one y might be observed, allowing the function to interpolate the training data. Therefore, Wang et al. (2024) propose to use a penalty preventing this quantile collapse to a single point. In settings learning with simulated data, the same problem may not occur, not needing the same preventive penalty.

B.2 Optimal Transport in GAN

While statistical literature provides theoretically rigorous quantile notions and in-depth understandings of them, in machine learning literature, much algorithmic effort has been made to promote the practicality and scalability of quantile learning. In machine learning, multivariate quantile learning has been approached from three different angles: through optimal transport (Makkuva et al., 2020; Sun et al., 2022), copula modeling (Wen and Torkkola, 2019; Zeng and Wang, 2022), and auto-regressive modeling (Koenker and Xiao, 2006; Ostrovski et al., 2018). Here, we focus on the optimal transport perspective.

In developing 2-Wasserstein GAN, Taghvaei and Jalali (2019) paid attention to Villani (2021) Theorem 2.9 (see, Makkuva et al. (2020)),

$$W_2^2(P,Q) = C_{P,Q} - \inf_{f:\ convex}\{\mathbb{E}_P[f(X)] + \mathbb{E}_Q[f^*(X)]\}, \tag{B.1}$$

where $f^*(y) = \sup_x \langle x, y\rangle - f(x)$ is the convex conjugate of $f(\cdot)$. By Brenier's theorem, the optimal transport w.r.t. $W_2^2(P,Q)$ is $\nabla f^*(y)$. The convex potential f has been modeled by using input convex neural networks (ICNNs) (Amos et al., 2017) as a one dimensional potential function, and the corresponding optimal transport has been used either as the generator or refiner of the generator (similar idea was used independently by Tanaka (2019)). Makkuva et al. (2020) stabilized the computation of (B.1), replacing the role of f^* by another convex function g, by $W_2^2(P,Q) = C_{P,Q} - \sup_f \inf_g\{\mathbb{E}_P[f(X)] + \mathbb{E}_Q[\langle Y, \nabla_g(Y)\rangle - f(\nabla_g(Y))]$, where f and g are convex. Huang et al. (2020) further use the gradient of the convex potential ∇f as a Flow model, by developing the inversion algorithm of ∇f and computations needed for the hessian computation for the likelihood calculation. When P is set a uniform $V \sim U(0,1)^d$ or other valid distributions of V in Carlier et al. (2016), these generative approaches can be seen as multidimensional quantile modelling (not conditional).

C Proof of Theorems

This section gives the proof of the results in Section 4. The roadmap to this section is as follows. We start by introducing a useful technical result in Appendix C.1. Appendix C.2 specifies detailed structural assumptions on the feed-forward neural networks and the loss functions. The proof of Lemma 2 is given in Appendix C.3. Appendix C.4 provides the proof of Theorem 3. Finally, Corollary 4 is proved in Appendix C.5.

C.1 A Useful Result in the Literature

In the upcoming analysis, the following lemma (see, e.g. Lemma B.1 of Chernozhukov et al. (2017)) may be useful. It verifies the equivalence between uniform convergence and continuous convergence.

Lemma 5. *[Equivalent condition of uniform convergence] Let $\mathbb{D}$ and $\mathbb{E}$ be complete separable metric spaces, with $\mathbb{D}$ compact. Suppose $f : \mathbb{D} \to \mathbb{E}$ is continuous. Then a sequence of functions $f_N : \mathbb{D} \to \mathbb{E}$ converges to f uniformly if and only if, for any convergent sequence $x_N \to x$ in $\mathbb{D}$, we have that $f_N(x_N) \to f(x)$.*

The proof of Lemma 5 is given in Rockafellar and Wets (2009).

C.2 Structural Assumptions on Neural Networks

We assume the ReLU activation function of the networks (though the results can be extended to piecewise linear activation functions at no notational cost; see Farrell et al. (2021)). Generally, to apply the consistency results in Farrell et al. (2021) to a function g that is learned by a feed-forward neural network by minimizing the empirical loss function $l(g \mid X, \theta, U)$, it is required to assume that the true underlying function g_* minimizes the expectation of its loss function, i.e. $g_* = \arg\min_g \mathbb{E}l(g \mid X, \theta, U)$. For our method in Section 3 specifically, under the affinity assumption (3.3), the true functions φ, b, f (denoted with asterisk in this section) are assumed to satisfy

$$(\varphi_*, b_*, f_*) = \underset{(\varphi,b,f)}{\arg\min}\, \mathbb{E}\mathcal{L}_1(\varphi, b, f \mid \boldsymbol{X}, \boldsymbol{\theta}, \boldsymbol{U}),$$

where the loss function $\mathcal{L}_1$ is given in (3.4).

In general, we assume any loss function $l(g \mid X, \theta, U)$ for the learning procedure of any function g (in our case, φ, b and f) to be Lipschitz in g, i.e.

$$|l(g_1 \mid X, \theta, U) - l(g_2 \mid X, \theta, U)| \le C_l |g_1(X, \theta, U) - g_2(X, \theta, U)| \tag{C.1}$$

for any g_1, g_2 in the same class as function g. Moreover, g obeys a curvature condition around the true function g_*,

$$c_1 \mathbb{E}[(g - g_*)^2] \le \mathbb{E}[l(g \mid X, \theta, U)] - \mathbb{E}[l(g_* \mid X, \theta, U)] \le c_2 \mathbb{E}[(g - g_*)^2]. \tag{C.2}$$

For our method, we assume the loss function $\mathcal{L}_1$ to satisfy (C.1) and (C.2) with respect to (φ, b, f).

For a user-chosen architecture $\mathcal{F}_{\text{DNN}}$, the estimate of a function g is computed using generated data flow $\{X_i, \theta_i, U_i\}_{i=1}^N$ by solving

$$\hat{g}_N := \underset{g \in \mathcal{F}_{\text{DNN}}, \|g\|_\infty \le 2M}{\arg\min} \sum_{i=1}^N l(g \mid X_i, \theta_i, U_i).$$

Our optimization problem (3.4) falls into this category under Condition (2) when N is large enough. Note that in the definition of $\mathcal{L}_1$, the term in the i-th summand, $\max_{j \in \{1, \cdots, N\}} \left\{U_j^\top \theta_i - \varphi(U_j) - b(U_j)^\top f(X_i)\right\}$ is essentially a finite-sample estimate to $\sup_{u \in \mathcal{U}} \left\{u^\top \theta_i - \varphi(u) - b(u)^\top f(X_i)\right\}$. The latter is a function of θ_i and X_i, therefore each summand of $\mathcal{L}_1$ is only related to X_i, θ_i as long as N is large enough.

C.3 Proof of Lemma 2

By triangle inequality, $|\hat{\psi}_N(u, x) - \psi(u, x)|$ is upper bounded by

$$|\hat{\varphi}_N(u) - \varphi(u)| + \sum_{k=1}^q |\hat{b}_{N,k}(u) \hat{f}_{N,k}(x) - b_k(u) f_k(x)|.$$

By Theorem 2 of Farrell et al. (2021), with probability at least $1 - e^{-\gamma}$ over the random sample drawn in data generation process,

$$\mathbb{E}_U(\hat{\varphi}_N(U) - \varphi(U))^2 \le \Phi_\varphi(N, \gamma).$$

Similarly, for any $1 \le k \le q$, with probability at least $1 - e^{-\gamma}$,

$$\mathbb{E}_U(\hat{b}_{N,k}(U) - b_k(U))^2 \le \Phi_b(N, \gamma),$$

and with probability at least $1 - e^{-\gamma}$,

$$\mathbb{E}_X(\hat{f}_{N,k}(X) - b_k(X))^2 \le \Phi_f(N, \gamma).$$

Here, the finite sample rates are defined as follows.

$$\begin{aligned}
&\Phi_\varphi(N,\gamma) = \\
&C_\varphi \left(\frac{W_\varphi L_\varphi \log W_\varphi}{N} \log N + \frac{\log\log N + \gamma}{N} + (\epsilon_{\text{DNN}}^{\varphi})^2 \right); \\
&\Phi_b(N,\gamma) = \\
&C_b \left(\frac{W_b L_b \log W_b}{N} \log N + \frac{\log\log N + \gamma}{N} + (\epsilon_{\text{DNN}}^{b})^2 \right); \\
&\Phi_f(N,\gamma) = \\
&C_f \left(\frac{W_f L_f \log W_f}{N} \log N + \frac{\log\log N + \gamma}{N} + (\epsilon_{\text{DNN}}^{f})^2 \right),
\end{aligned} \tag{C.3}$$

where C_φ, C_b, C_f are positive constants that are independent of N, W_φ, W_b, W_f are the maximum widths of the respective neural networks at a given layer, and L_φ, L_b, L_f are the maximum depths of the respective neural networks. By Cauchy-Schwarz inequality,

$$\begin{aligned}
&(\hat{\psi}_N(u,x) - \psi(u,x))^2 \\
\leq &(2q+1)(\hat{\varphi}_N(u) - \varphi(u))^2 \\
&+ (2q+1)\sum_{k=1}^{q} \hat{f}_{N,k}^2(x)(\hat{b}_{N,k}(u) - b_k(u))^2 \\
&+ (2q+1)\sum_{k=1}^{q} b_k^2(u)(\hat{f}_{N,k}(x) - f_k(x))^2.
\end{aligned}$$

Therefore, with probability no less than $1 - 3qe^{-\gamma}$, we have the following L^2 convergence guarantee of the estimated potential function:

$$\begin{aligned}
&\mathbb{E}_U \mathbb{E}_X (\hat{\psi}_N(U,X) - \psi(U,X))^2 \\
&\leq 3q\Phi_\varphi(N,\gamma) + 12q^2M^2\Phi_b(N,\gamma) + 3q^2M^2\Phi_f(N,\gamma).
\end{aligned} \tag{C.4}$$

We denote the event $\mathcal{A}_N$ hereafter, under which (C.4) holds. By Markov's inequality, this L^2 convergence immediately implies convergence in probability.

Meanwhile, under Condition 2, the estimated function $\{\hat{\psi}_N\}_{N\in\mathbb{N}}$ also satisfies uniform boundedness and uniform equicontinuity on their domain $\mathcal{U}\times\mathcal{X}$, which is also compact. By Arzelà-Ascoli Theorem, there exists a subsequence $\{\hat{\psi}_{N_m}\}_{m\in\mathbb{N}}$ that converges uniformly. The continuity of each function $\hat{\psi}_{N_m}$ implies that the uniform limit $\hat{\psi}^\diamond := \lim_{m\to\infty}\hat{\psi}_{N_m}$ is also continuous. On the other hand, under event $\mathcal{A}_N$, $\hat{\psi}_{N_m}$ converges to ψ in probability. We can thus extract a further subsequence $\{\hat{\psi}_{N_{m_l}}\}_{l\in\mathbb{N}}$, which also uniformly converges to $\hat{\psi}^\diamond$, such that $\hat{\psi}_{N_{m_l}} \to \psi$ almost everywhere on $\mathcal{U}\times\mathcal{X}$. Therefore $\hat{\psi}^\diamond = \psi$ almost everywhere on $\mathcal{U}\times\mathcal{X}$. Since both functions are continuous, $\hat{\psi}^\diamond(u,x) = \psi(u,x)$ holds for all $(u,x)\in\mathcal{U}\times\mathcal{X}$. This ensures uniqueness of the uniform limit, which further implies that the sequence $\hat{\psi}_N$ uniformly converges to ψ, finishing the proof of the the first assertion under the event $\mathcal{A}_N$.

For the second assertion, note that the uniform convergence of $\hat{\psi}_N$ implies that for any $\epsilon > 0$, there exists $N_0 \in \mathbb{N}$ such that for all $N \geq N_0$ and $u \in \mathcal{U}$, $\hat{\psi}_N(u,x) \leq \psi(u,x) + \epsilon$. Therefore,

$$\sup_{u\in\mathcal{U}}\{\theta^\top u - \hat{\psi}_N(u,x)\} \geq \sup_{u\in\mathcal{U}}\{\theta^\top u - \psi(u,x) - \epsilon\},$$

which is equivalent to $\hat{\psi}_N^*(\theta,x) \geq \psi^*(\theta,x) - \epsilon$, and

$$\liminf_{N\to\infty} \hat{\psi}_N^*(\theta,x) \geq \psi^*(\theta,x) - \epsilon.$$

Similarly, we have $\hat{\psi}_N^*(\theta,x) \leq \psi^*(\theta,x) + \epsilon$, and

$$\limsup_{N\to\infty} \hat{\psi}_N^*(\theta,x) \leq \psi^*(\theta,x) + \epsilon.$$

Since $\epsilon > 0$ can be arbitrarily small, we first arrive at pointwise convergence:

$$\lim_{N\to\infty} \hat{\psi}_N^*(\theta,x) = \psi^*(\theta,x). \tag{C.5}$$

Given any compact subset $K' \subset \Theta$, we want to show that the convergence is uniform, i.e.

$$\sup_{\theta\in K', x\in\mathcal{X}} |\hat{\psi}_N^*(\theta,x) - \psi^*(\theta,x)| \to 0.$$

Consider a sequence in $K'\times\mathcal{X}$, $(\theta_N, X_N) \to (\theta, X)$. Since $\mathcal{U}\times\mathcal{X}$ is compact, the supremum in the definition of ψ^* is attained within $\mathcal{U}\times\mathcal{X}$. Therefore, for any $\theta\in\Theta$, there exists $u(\theta)\in\mathcal{U}$ such that

$$\psi^*(\theta,x) = u(\theta)^\top\theta - \psi(u(\theta),x).$$

Similarly, for any $N\in\mathbb{N}$ and $\theta_N\in\Theta$, there exists $u_N(\theta_N)\in\mathcal{U}$ such that

$$\hat{\psi}_N^*(\theta,x_N) = u_N(\theta_N)^\top\theta_N - \hat{\psi}_N(u_N(\theta_N),x_N).$$

Note that $u(\theta)$ is the maximizer in the definition of $\psi^*(\theta,x)$. Therefore we have the inequality $\psi^*(\theta,x) \geq u_N(\theta_N)^\top\theta - \psi(u_N(\theta_N),x)$, thus the difference $\hat{\psi}_N^*(\theta_N,x_N) - \psi^*(\theta,x)$ is upper bounded by

$$u_N(\theta_N)^\top(\theta_N - \theta) - [\hat{\psi}_N(u_N(\theta_N),x_N) - \psi(u_N(\theta_N),x)].$$

The first term converges to zero since $\theta_N \to \theta$ and that $\mathcal{U}$ is a compact set. the two terms in the bracket also converges to zero, by combining uniform convergence of $\hat{\psi}_N$, i.e.

$$\hat{\psi}_N(u_N(\theta_N),x_N) - \psi(u_N(\theta_N),x_N) \to 0,$$

with equicontinuity of ψ, i.e.

$$\psi(u_N(\theta_N),x_N) - \psi(u_N(\theta_N),x) \to 0.$$

Therefore

$$\limsup_{N\to\infty}[\hat{\psi}_N^*(\theta_N, x_N) - \psi^*(\theta, x)] \leq 0. \tag{C.6}$$

Using the same method, $\psi^*(\theta, x) - \hat{\psi}_N^*(\theta_N, x_N)$ is upper bounded by

$$u(\theta)^\top(\theta - \theta_N) + [\psi(u(\theta), x_N) - \hat{\psi}_N(u(\theta), x)],$$

which also converges to zero, yielding

$$\liminf_{N\to\infty}[\hat{\psi}_N^*(\theta_N, x_N) - \psi^*(\theta, x)] \geq 0. \tag{C.7}$$

Combining (C.6) and (C.7), we arrive at

$$\hat{\psi}_N^*(\theta_N, x_N) \to \psi^*(\theta, x) \text{ for all } (\theta_N, x_N) \to (\theta, x). \tag{C.8}$$

The proof of the second assertion under the event $\mathcal{A}_N$ is thus finished by Lemma 5.

Finally, the almost sure argument is proved by noting that $\mathbb{P}(\mathcal{A}_N^c) \leq 3qe^{-\gamma}$. By taking $\gamma = 2\log N$, the rates in (C.3) still converge to zero as $N \to \infty$. Additionally, we have

$$\sum_{N=1}^{\infty} \mathbb{P}(\mathcal{A}_N^c) \leq \sum_{N=1}^{\infty} \frac{3q}{N^2} < \infty.$$

By Borel-Cantelli Lemma, this implies event $\mathcal{A}_N$ occurs indefinitely often.

C.4 Proof of Theorem 3

Recall that

$$\hat{Q}^N_{\theta|X=x}(u) = \arg\sup_{\theta\in\Theta}[\theta^\top u - \hat{\psi}_N^*(\theta, x)],$$

$$Q_{\theta|X=x}(u) = \arg\sup_{\theta\in\Theta}[\theta^\top u - \psi^*(\theta, x)].$$

Define the extension map $g^N_{x,u}(\theta) = \theta^\top u - \hat{\psi}_N^*(\theta, x)$ if $\theta \in \Theta$, and $-\infty$ if $\theta \notin \theta$. Another extension map $g_{x,u}(\theta)$ can be defined analogously with $\psi^*(\theta, x)$. Note that this is a concave mapping from $\mathbb{R}^d$ to $\mathbb{R}$. By Lemma 2, $\hat{\psi}_N^*$ is uniformly consistent over any compact set $K' \subset \Theta$. Under event $\mathcal{A}_N$, for any sequence $\{u_N\}$ such that $u_N \to u \in K_U$, a compact set of $\mathcal{U}_0$, and any $x \in \mathcal{X}_0$,

$$g^N_{x,u_N}(\theta) \to g_{x,u}(\theta)$$

for all $\theta \in \text{int } \Theta$, a dense subset of Θ. By Lemma B.2 of Chernozhukov et al. (2017), we have the consistency of the argsup,

$$\arg\sup_{\theta\in\Theta} g^N_{x,u_N}(\theta) \to \arg\sup_{\theta\in\Theta} g_{x,u}(\theta),$$

which is equivalent to

$$\hat{Q}^N_{\theta|X=x}(u_N) \to Q_{\theta|X=x}(u).$$

Since the sequence $u_N \to u$ is arbitrarily taken over the compact subset K, by Lemma 5, we have

$$\hat{Q}^N_{\theta|X=x}(u) \to Q_{\theta|X=x}(u) \text{ uniformly over } K_U.$$

under event $\mathcal{A}_N$. The first assertion is then proved by letting $\gamma = 2\log N$ and applying Borel-Cantelli Lemma, like in the proof of Lemma 2.

Since our method solves the Monge-Kantorovich problem (2.1), the optimal transport mapping ensure the following identity of 2-Wasserstein distance,

$$\begin{aligned} &W_2^2(\hat{\pi}^N(\theta \mid X = x), \pi(\theta \mid X = x)) \\ &\quad = \int_{\mathcal{U}} \|\hat{Q}^N_{\theta|X=x}(u) - Q_{\theta|X=x}(u)\|^2 \,\mathrm{d}F_U(u). \end{aligned} \tag{C.9}$$

Note that our source distribution F_U features a density that is proportional to r^{-d+1}, which is positive everywhere on the entire unit ball $\mathcal{U} = S^d(1)$ and upper bounded except on the center. Since the choice of compact subset $K_U \subset \mathcal{U}_0 = \text{int } S^d(1)$ is arbitrary, it is possible to find a K_U that contains the center such that its Lebesgue measure is arbitrarily close to that of $\mathcal{U}$. In that case, for any $\epsilon > 0$ there exists such a compact set K_U that

$$F_U(\mathcal{U} \setminus K_U) < \frac{\epsilon}{4dM^2}.$$

In light of this, the right-hand side of (C.9) can be upper bounded by splitting the integral,

$$\begin{aligned} &\int_{K_U} \|\hat{Q}^N_{\theta|X=x}(u) - Q_{\theta|X=x}(u)\|^2 \,\mathrm{d}F_U(u) \\ &+ \int_{\mathcal{U}\setminus K_U} \|\hat{Q}^N_{\theta|X=x}(u) - Q_{\theta|X=x}(u)\|^2 \,\mathrm{d}F_U(u). \end{aligned}$$

The first term converges to zero almost surely. Since the parameter space $\Theta \subset [-M, M]^d$, the second term is upper bounded by $4dM^2 F_U(\mathcal{U} \setminus K_U) < \epsilon$. This proves the convergence of $W_2(\hat{\pi}^N(\theta \mid X = x), \pi(\theta \mid X = x))$ to zero given any fixed $x \in K_X \subset \mathcal{X}$ under the event $\mathcal{A}_N$, which occurs almost surely.

C.5 Proof of Corollary 4

Given any $\tau \in (0, 1)$, the ball $S^d(\tau)$ is a closed subset in the interior of $\mathcal{U} = S^d(1)$. Due to the uniform convergence result of vector quantiles given in the first assertion of Theorem 3, almost surely we have

$$\sup_{x\in K_X} \sup_{u\in S^d(\tau)} \|\hat{Q}^N_{\theta|X=x}(u) - Q_{\theta|X=x}(u)\| \to 0, \tag{C.10}$$

i.e. uniform convergence over the ball $S^d(\tau)$.

We next prove the Hausdorff distance between the credible set $\hat{C}^N_\tau(\theta \mid X = x)$ with the oracle set

$C_\tau(\theta \mid X = x)$. For any $\theta \in C_\tau(\theta \mid X = x)$, since the ball $S^d(\tau)$ is compact, there exists $u_0 \in S^d(\tau)$ such that $\theta = Q_{\theta|X=x}(u_0)$. By uniform convergence (C.10),

$$\sup_{x \in K_X} \|\hat{Q}^N_{\theta|X=x}(u_0) - \theta\| \to 0.$$

Since $u_0 \in S^d(\tau)$, by definition of the oracle set (4.9), we have $\hat{\theta}^N := \hat{Q}^N_{\theta|X=x}(u_0) \in \hat{C}^N_\tau(\theta \mid X = x)$. This means for any $\theta \in C_\tau(\theta \mid X = x)$, there exists $\hat{\theta}^N \in \hat{C}^N_\tau(\theta \mid X = x)$ such that $\|\hat{\theta}^N - \theta\| \to 0$. Thus

$$\sup_{\theta \in C_\tau(\theta|X=x)} \inf_{\hat{\theta}^N \in \hat{C}^N_\tau(\theta|X=x)} \|\hat{\theta}^N - \theta\| \to 0.$$

Conversely, we can prove that for any $\hat{\theta}^N \in \hat{C}^N_\tau(\theta \mid X = x)$, there exists $\theta \in C_\tau(\theta \mid X = x)$ such that $\|\hat{\theta}^N - \theta\| \to 0$, and thus

$$\sup_{\hat{\theta}^N \in \hat{C}^N_\tau(\theta|X=x)} \inf_{\theta \in C_\tau(\theta|X=x)} \|\hat{\theta}^N - \theta\| \to 0.$$

Since all convergences are uniform over $x \in K_X$, we conclude that

$$\sup_{x \in K_X} d_H(\hat{C}^N_\tau(\theta \mid X = x), C_\tau(\theta \mid X = x)) \to 0.$$

D Autoregressive Quantile Learning

Another way to use quantile modeling for multi-dimensional data is through auto-regressive modeling. This idea was first proposed by Koenker and Xiao (2006) and later adopted in the machine learning literature for generative modeling (Ostrovski et al., 2018) and time series forecasting (Gouttes et al., 2021). The key idea is to define the joint quantile by using one-dimensional conditional quantiles as $F_X(x) = P(X_1 \le x_1, \ldots, X_n \le x_n) = \prod_{i=1}^n F_{X_i|X_{i-1},\ldots,X_1}(x_i)$. Now we can define $F_X^{-1}(\tau_1, \ldots, \tau_n) = (F_{X_1}^{-1}(\tau_1), \ldots, F^{-1}_{X_n|X_{n-1},\ldots}(\tau_n))$. Each one-dimensional quantile can be optimized by minimizing the pinball loss integrated over all quantile levels $\tau \in [0, 1]$, and they are often designed with RNNs to encode the conditions. Due to the sequential conditional structure, this approach has received attention in modeling time-dependent latent variables, such as implied volatility.

E More Results for the Gaussian Example

More details for the Gaussian example experiment in Section 5 The model allow us to simulate a pair of parameter and data (of n observations) as $(X, (\mu, \sigma^2))$ for $X \in \mathbb{R}^n$. Due to the conjugate prior, the true posterior samples are available. For the observed data, given the number of observations $n \ge 1$, we set $X = [x, ..., x] \in \mathbb{R}^n$ for a few manually chosen x values. The feature map is chosen as a DeepSet with the output dimension 2. We trained all our network for 150 epochs, with 1 epoch as 100 iterations with mini-batch size 128. We optimize by the Adam optimizer (Diederik, 2014) with default hyperparameter settings, where the learning rate is 0.01 and is reduced by $\times 0.99$ for each epoch. We found that a good initial random initialization is important to learn a meaningful quantile map. We run 10 different random trials and then pick up the model having the lowest loss function.

The case of $n = 2$. Here, we consider the case when ample size $n = 2$. To obtain the credible set of probability τ, we sample $U \sim \tau F_U$, which is equivalent to the source distribution restricted on radius τ, and apply the trained quantile sampler. The boundary of the credible set is obtained by applying a contour plot on the sampled credible sets. The results are shown in Figure 4. We can see very successful posterior sampling, and the credible sets show the nested structure.

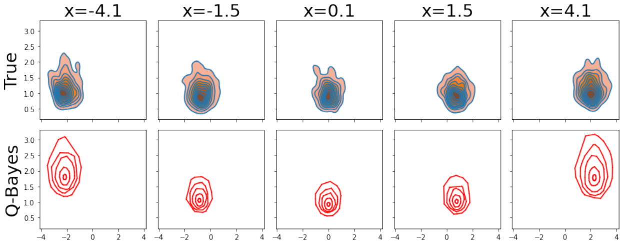

Figure 4: The credible set for the Gaussian example with varying x with $n = 2$. The subplots show the alignment on the first axis (x-axis) between the true posterior sample and the generated samples.

The case of increasing n. We already highlighted in Section 5 that the support shrinkage is clearly observed with an increasing n for x relatively near the origin. A similar phenomenon is observed also for a relatively large x as in Figure 5 albeit some performance decrease probably due to lack of observed data during training.

Comparison with other methods. We train a sequential B-GAN (B-GAN-seq, Wang and Ročková (2022)) and the Autoregressive method from Section D (AutoR) using an off-the-shelf sampling mechanism and compare the results in Figure 6. We believe that the scalability achieved by adopting DeepSets has contributed to the improvement in our algorithm. To examine the effect of DeepSet more clearly, we train our method with three different feature networks f: f as a DeepSet (Q-Bayes-deep), a fully connected network

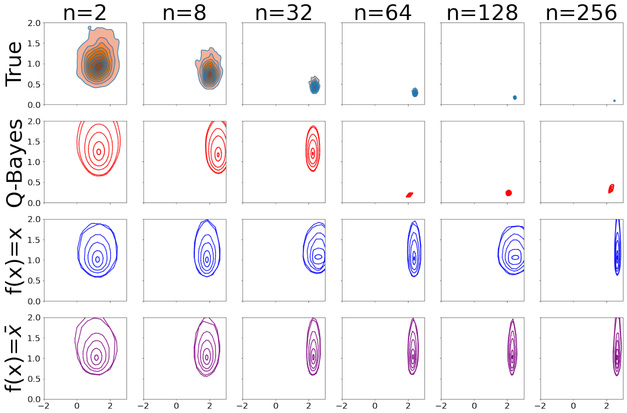

Figure 5: The credible set for the Gaussian example with increasing n for a relatively large x ($x = 2.5$). From $n = 16$, the quality decreases.

(Q-Bayes-mlp), and the mean and standard deviation as manually chosen summary statistics (Q-Bayes-man). Performance is measured by the Maximum Mean Discrepancy (MMD) between the empirical distribution of the true posterior and the generated samples. We also measure Distance to Measure (DTM), which is the average distance between the true parameters and the generated posterior samples conditioned on the data from the true parameters. Specifically, given a trained posterior-generating model $Q(\theta \mid X)$, DTM is defined by $\frac{1}{3\times 10^4}\sum_{j=1}^{100}\sum_{i=1}^{300}|\theta_j^* - \theta_{ji}|$, where $\theta_j^* \sim \pi(\theta)$ and $\theta_{ji} \sim Q(\theta \mid X_j)$ for $X_j \sim L(X \mid \theta_j^*)$. To account for the variability introduced by the randomness of each run, we perform 10 runs for each method and select the model with the best performance measure. As shown in Figure 8, the fully connected network (Q-Bayes-mlp) does not perform as well as the DeepSet (Q-Bayes-deep) as n increases. A similar improvement might be possible for B-GAN with the development of a variation that incorporates a learnable feature network to enable the use of DeepSets. Currently, B-GAN handles increasing n by sequentially applying B-GAN, using a previously learned posterior sampler as a new prior sampler. (For the support shrinkage of B-GAN-seq, see Figure 7.) We also train the autoregressive model with the three feature maps (AutoR-deep, AutoR-mlp, AutoR-man), as shown in Figure 8. Note that in this Gaussian setting, the summary statistics are sufficient summary statistics. The Autoregressive method with the appropriate summary statistics demonstrates the best performance. On the other hand, with other feature f networks, it falls short of our DeepSet-based method as n increases.

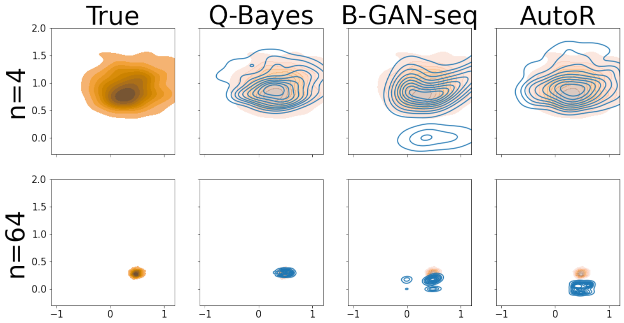

Figure 6: Comparison between Auto Regression (AutoR), Sequential Adversarial Bayesian Simulation (B-GAN-seq), and the Vector Quantile (Brenier) methods for $n = 4$ and $n = 64$. The contour lines in blue represent the KDE approximation.

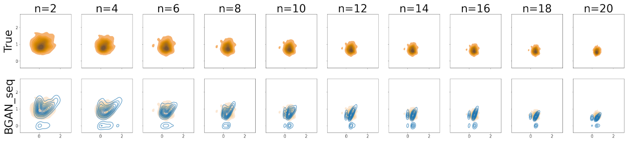

Figure 7: The sequential B-GAN method for increasing n. The network is trained sequentially for every additional two data points. The contour lines in blue represent the KDE approximation.

Note that our method is not the only method that can demonstrate this effect; The sequential B-GAN (B-GAN-seq) and auto-regressive method (AutoR) show support shrinkage. For AutoR, as shown in Figure 8, its performance is only competitive when the true summary statistics are given (the orange line). As a comparison, our proposed method shows about the same performance even when the summary statistics are unknown and learned automatically (the red line). For B-GAN, it had to be adapted in these comparisons to exhibit support shrinkage. While the quality of the samples from B-GAN-seq is very competitive, it trains a sampler for a fixed X_0, not a generic realization of the data.

F Network Architecture

In all our experiments, the DeepSet feature network is implemented with input dimension d_X, a set feature space dimension q_1, a hidden layer width of 16, and two layers of width 16 in the transformation network. We use the LSTM network design by Sun et al. (2022). For the LSTM feature network, the architecture consists of a single LSTM layer with a hidden state size of 512 and an input size of d_X. The final output of the sequence is mapped to a fixed dimensionality q_2 using a fully connected layer. In what follows, we provide the actual code to implement them.

Multilayer Perceptron (MLP)

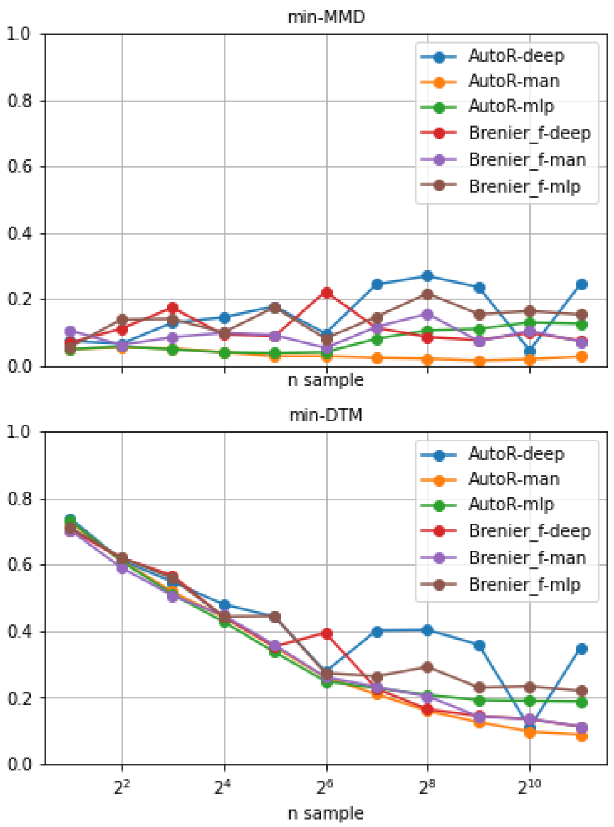

Figure 8: The minimum MMD and DTM (among 10 random runs) for increasing n.

```
import torch
import torch.nn as nn

class MLP(nn.Module):
    def __init__(self, device="cuda", dim
        =2, z_dim=1,
                 leaky=0.1, factor=64,
                   n_layers=2,
                 dropout=0, positive=
                   False):
        super().__init__()
        self.non_linear = nn.LeakyReLU(
            leaky) if leaky > 0 else nn.
            ReLU()
        self.dropout = nn.Dropout(dropout
            )
        self.layers = nn.ModuleList([nn.
            Linear(dim, factor)])
        for _ in range(n_layers):
            self.layers.append(nn.Linear(
                factor, factor))
        self.layers.append(nn.Linear(
            factor, z_dim))
        self.positive = positive
        self.to(device)

    def forward(self, x):
        h = x
        for i, layer in enumerate(self.
            layers):
            h = layer(h)
            if i < len(self.layers) - 1:
                # Apply non-linearity
                except last layer
                h = self.non_linear(h)
                h = self.dropout(h)
        return h.abs() if self.positive
            else h
```

The LSTM Feature Network

```
class BiRNN(nn.Module):
    def __init__(self, input_size,
        hidden_size, num_layers, xdim,
        bn_last=True, device="cuda"):
        super(BiRNN, self).__init__()
        self.hidden_size = hidden_size
        self.num_layers = num_layers
        self.bn_last = bn_last
        self.lstm = nn.LSTM(input_size,
            hidden_size, num_layers,
            batch_first=True,
            bidirectional=False)
        self.fc = nn.Linear(hidden_size,
            xdim)
        self.norm = nn.BatchNorm1d(xdim,
            momentum=1.0, affine=False)
        self.device = device
        self.to(device)

    def forward(self, x):
        # Set initial states
        h0 = torch.zeros(self.num_layers,
             x.size(0), self.hidden_size)
            .to(self.device)
        c0 = torch.zeros(self.num_layers,
             x.size(0), self.hidden_size)
            .to(self.device)
        # Forward propagate LSTM
        out, _ = self.lstm(x, (h0, c0))
        # Decode the hidden state of the
            last time step
        out = self.fc(out[:, -1, :])
        if self.bn_last:
            return self.norm(out)
        return out
```

The DeepSet Feature Network

```
class DeepSets(nn.Module):
    def __init__(self, dim_x, dim_ss,
        factor=16,
                 num_layers=2, device="
                   cuda",
                 bn_last=True):
        super(DeepSets, self).__init__()
        self.common_feature_net = MLP(
            device=device,dim=dim_x,z_dim
            =dim_ss,dropout=0, factor=64,
             n_layers=3)
        self.next_net = MLP(device=device
            , dim=dim_ss, z_dim=dim_ss,
```

Method	Ours	Rejection ABC	SMC ABC	AutoR
DTM	0.177	0.521	0.513	0.179
DPM	0.159	0.432	0.441	0.195
Time	37 mins	29 mins	41 mins	27 mins

Table 1: Comparisons on the Brock-Hommes model

```
            factor=factor, n_layers=
            num_layers)
        self.to(device)
        self.device = device
        self.bn_last = bn_last
        self.norm = nn.BatchNorm1d(dim_ss
            , momentum=1.0, affine=False)

    def forward(self, x):
        shape = x.shape
        assert len(shape) == 3
        phi = self.common_feature_net(x.
            view(-1, shape[-1])).view(x.
            shape[0], x.shape[1], -1).
            mean(1)
        out = self.next_net(phi)
        if self.bn_last:
            return self.norm(out)
        return out
```

G Comparison on the Brock Hommes model

We compare our method with the standard ABC (rejection ABC) and sequential Monte Carlo ABC (SMC-ABC, Sisson et al. (2007)), which is designed to enhance the efficiency of standard ABC methods. For ABC, we allocated a total budget of 10^8 proposals, a generous allocation commonly used for complex ABC simulations (Raynal et al., 2019). Since summary statistics are unknown, we adopted a naive version of ABC by using the entire data vector as the statistic and employing the L_2 distance as the metric for thresholding. To ensure at least 1,000 samples were selected, the threshold was set to 22, resulting in 1,153 proposals accepted out of the total of one hundred million proposals. For SMC-ABC, we use the implementation provided by O'Hagan et al. (2024). For numerical performance metrics, we use DTM and the distance from the posterior mean (DPM), along with the computational time.

The results are given in Table 1. Without a big increase in computational cost, our method produces a qualitative improvement in the posterior samples. It is important to emphasize that ABC sampling must be performed separately for each new observed dataset, making it computationally expensive collectively. On the contrary, our method learns a mapping. Once such a mapping is trained, it can be applied to any observed dataset at a very low computational cost without the need for retraining. In aggregate, this significantly improves efficiency in practice.