Adaptive Conditional Quantile Neural Processes

Abstract

Neural processes are a family of probabilistic models that inherit the flexibility of neural networks to parameterize stochastic processes. Despite providing well-calibrated predictions, especially in regression problems, and quick adaptation to new tasks, the Gaussian assumption that is commonly used to represent the predictive likelihood fails to capture more complicated distributions such as multimodal ones. To overcome this limitation, we propose Conditional Quantile Neural Processes (CQNPs), a new member of the neural processes family, which exploits the attractive properties of quantile regression in modeling the distributions irrespective of their form. By introducing an extension of quantile regression where the model learns to focus on estimating *informative* quantiles, we show that the sampling efficiency and prediction accuracy can be further enhanced. Our experiments with real and synthetic datasets demonstrate substantial improvements in predictive performance compared to the baselines, and better modeling of heterogeneous distributions' characteristics such as multimodality.

1 INTRODUCTION

Conventionally, regression problems are approached by modeling the relation between inputs and outputs with a deterministic function where the parameters of this function are optimized with respect to a loss function. Nonparametric statistical methods, however, choose a different perspective by viewing the regression function itself as a random object. This allows for fitting a family of functions that are coherent with the data instead of a single one. (Conditional) Neural Processes (C/NPs) [Garnelo et al., 2018a,b] are a class of such models that inherit the computational efficiency of neural networks and integrate it with desirable properties of Gaussian Processes (GPs), specifically uncertainty quantification and rapid adaptation to new observations [Garnelo et al., 2018a,b]. In fact, Rudner et al. [2018] showed that under certain conditions, NPs recover GPs with deep kernels.

NPs can be viewed as the composition of an encoder and a decoder where the encoder embeds a finite collection of observations $\mathcal{E}_{\text{context}} = \{(x_i, y_i)\}_{i=1}^{N}$, also known as context set, into a latent space. Subsequently, the decoder takes in a new target location x^* together with the latent representation of context data to parameterize the conditional distribution $p(\mathrm{y}^* \mid x^*, \mathcal{E}_{\text{context}})$ of the corresponding target output y^*. Several variants of NPs have been introduced. Kim et al. [2019], Nguyen and Grover [2022], Kim et al. [2022], Guo et al. [2023] incorporate attention mechanisms to make NPs less prone to under-fitting. Gordon et al. [2020], Foong et al. [2020] build translation equivariance into NPs by introducing convolutional deep sets. Holderrieth et al. [2021] further extend the translation equivariance to more complicated transformations such as rotations and reflections. Volpp et al. [2021] improve context aggregation by casting it as a Bayesian inference problem. Lee et al. [2020] use the bootstrap technique for inducing the functional uncertainty. Bruinsma et al. [2021], Markou et al. [2022] propose Gaussian Neural Processes to model predictive correlations between different target locations. Wang and Van Hoof [2020] propose Doubly Stochastic Variational Neural Processes to capture target-specific local variation by adding local latent variables in a hierarchical encoder.

Despite their higher expressive power compared to vanilla C/NP, they fail to model multimodality and heavy tails that may be exhibited by the predictive distribution in real-world problems such as those in transportation science, economics, astronomy, and meteorology [Chen et al., 2016]. This is due to the fact that existing models have a common Gaussian likelihood assumption. To circumvent this issue, we propose to employ an infinite mixture of asymmetric Laplace distributions as the likelihood for C/NPs. This likelihood,

Submitted to the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023). **To be used for reviewing only**.

which can be seen as an extension to the well-known quantile regression [Koenker and Bassett, 1978, Koenker, 2005], has shown to be an effective approach for modeling heterogeneous distributions [Dabney et al., 2018, Brando et al., 2019]. To further improve the expressive power of the model, we propose an adaptive extension of our approach where instead of fixing the quantile levels, the model learns to predict the quantiles that are more *informative*. We integrate our model with CNPs resulting in (Adaptive) Conditional Quantile Neural Processes (A/CQNPs) and conduct several experiments on both synthetic and real-world datasets to illustrate the performance enhancements achieved by our method. While, in this work, our focus is on CNPs, we emphasize that the simple, yet generic nature of the proposed approach allows for quick adaptation to other members of the NPs family.

2 PRELIMINARIES

2.1 QUANTILE REGRESSION

In regression analysis, given a set of observations $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ from a pair of random variables $(\mathbf{x}, \mathrm{y})$, the objective is to learn a function f that maps the inputs x_i to the outputs y_i. We further assume that $\mathbf{x}$ and y are vector and scalar random variables, respectively. In the case where $\mathbf{y}$ is a vector, we consider its scalar elements independently. In decision theory, the optimality of an estimator is measured through its risk function $\mathbb{E}_{\mathcal{D}}[\mathcal{L}(y_i,\, f(x_i)]$ where $\mathcal{L}(\cdot\,,\,\cdot)$ is a loss function and an estimator with lower risk is preferred. A common choice for the loss function is the mean squared error. It is well-known that for this loss, the estimator with minimum risk is $f(x) = \mathbb{E}[\mathrm{y} \,|\, \mathbf{x} = x]$ which relates the inputs and outputs through the conditional mean [Casella and Berger, 2021]. However, in more complicated instances where $p(\mathrm{y} \,|\, \mathbf{x} = x)$ is heavy-tailed, asymmetric, or multimodal, the conditional mean is not a sufficient statistic to summarize the distribution characteristics. This can be alleviated by using more robust statistics such as *quantiles*. For $\tau \in (0, 1)$, the τ-th conditional quantile $\mu_\tau(x) = \inf\, \{\mu \mid p(\mathrm{y} \leq \mu \,|\, \mathbf{x} = x) \geq \tau\}$ is obtained by minimizing the asymmetric absolute loss $\mathcal{L}_\tau(y_i, x_i) = \rho_\tau(y_i - \mu_\tau(x_i))$ where $\rho_\tau(y) = \max\, \{y\tau,\, y(\tau - 1)\}$. It is straightforward to show that minimizing $\mathcal{L}_\tau$ is equivalent to maximizing the log-likelihood of an Asymmetric Laplace ($\mathcal{AL}$) distribution [Yu and Moyeed, 2001] with a constant scale parameter. The density function of $\mathcal{AL}$ distribution is defined as the following:

$$\mathcal{AL}\left(y \,|\, \mu_\tau, \sigma_\tau, \tau\right) = \frac{\tau(1-\tau)}{\sigma_\tau} \times \exp\left(-\frac{1}{\sigma_\tau}\rho_\tau(y - \mu_\tau)\right), \quad (1)$$

where $\mu_\tau \in \mathbb{R}$, $\sigma_\tau \in \mathbb{R}_{>0}$, and $\tau \in (0, 1)$ are the location, scale, and skew parameters [Yu and Zhang, 2005]. Note that in vanilla quantile regression τ and σ_τ are fixed.

Like mean, a single quantile may not be sufficient to model heterogeneous distributions. To address this, several works have proposed to predict a set of quantiles instead of a single one [Liu and Wu, 2009, 2011, Sangnier et al., 2016, Dabney et al., 2018, Brando et al., 2019]. Among them, Brando et al. [2019] use an uncountable mixture of $\mathcal{AL}$ distributions to approximate the predictive distribution. More specifically, the predictive distribution is parameterized as follows:

$$p(\mathrm{y} \,|\, x) = \mathbb{E}_{\tau \sim \mathcal{U}(0,1)}\left[\alpha_\tau(x)\, \mathcal{AL}\left(\mathrm{y} \,|\, \mu_\tau(x), \sigma_\tau(x), \tau\right)\right], \quad (2)$$

where $\mathcal{U}(0, 1)$ is the uniform distribution and $\alpha_\tau(x) \geq 0$ is the mixture coefficient such that $\mathbb{E}_{\tau \sim \mathcal{U}(0,1)}[\alpha_\tau(x)] = 1$. The parameters of this mixture distribution are estimated using deep neural networks.

2.2 CONDITIONAL NEURAL PROCESSES

Let $\mathcal{D} = \{(x_i, y_i) \in \mathbb{X} \times \mathbb{Y}\}_{i=1}^N$ be a set of training observations corresponding to a realization of the following stochastic process; let $p(\mathrm{f})$ be a probability distribution over functions $f : \mathbb{X} \to \mathbb{Y}$, then for $f \sim p(\mathrm{f})$, set $y_i = f(x_i)$ [Garnelo et al., 2018a]. CNP is a conditional stochastic process q_υ (where υ is the set of parameters of q) that approximates the distribution over functions $p(\mathrm{f})$. In vanilla CNP, first, each pair (x_i, y_i) of training observations (or a subset of them, i.e. *context* set) is embedded into a latent space. Next, these embeddings are aggregated to form a representation, which is used with target inputs to predict the target outputs. We note that the aggregated representation is invariant to the ordering of context points. More specifically, given an encoder $\varphi_{\mathrm{enc}} \colon \mathbb{X} \times \mathbb{Y} \to \mathbb{R}^{d_e}$ and a decoder $\varphi_{\mathrm{dec}} \colon \mathbb{X} \times \mathbb{R}^{d_e} \to \Theta$, where $\mathbb{R}^{d_e}$ is the embedding space and Θ is the set of parameters of the predictive distribution, CNP formulates the predictive distribution of $f(x^*)$ for a given target x^* as:

$$p_\theta(f(x^*) \,|\, x^*, \mathcal{D}) = p_\theta\left(f(x^*) \,|\, \varphi_{\mathrm{dec}}\big(x^*, \varphi_{\mathrm{enc}}\left(x^*, \mathcal{D}\right)\big)\right). \quad (3)$$

As mentioned earlier, different members of the CNPs family, have different encoder and decoder architectures. In the vast majority of CNP variants, the predictive distribution is chosen to be a simple Gaussian distribution resulting in:

$$p_\theta(f(x^*) \,|\, x^*, \mathcal{D}) = \mathcal{N}(f(x^*) \,|\, \varphi(x^*, \mathcal{D})), \quad (4)$$

where $\varphi(x^*, \mathcal{D}) = \{\varphi^\mu_{\mathrm{dec}}(x^*, \varphi_{\mathrm{enc}}(x^*, \mathcal{D})),\, \varphi^\sigma_{\mathrm{dec}}(x^*, \varphi_{\mathrm{enc}}(x^*, \mathcal{D}))\}$ is the set of functions mapping the embeddings and x^* to the mean and standard deviation of y^*. To the best of our knowledge, none of the existing CNPs are capable of modeling heterogeneous distributions such as multi-modal ones.

3 METHOD

3.1 ADAPTIVE QUANTILE REGRESSION

Although the uncountable mixture of $\mathcal{A}L$s in equation 2 gives us a comprehensive picture of the conditional distribution by fitting the full quantile function, it might lead to some practical inefficiencies that will be discussed below. For the rest of this discussion, we use $\mathcal{B}(c, \epsilon)$ to denote an open interval of length $2\epsilon > 0$ and centered at c, i.e. $\mathcal{B}(c, \epsilon) = (c - \epsilon,\, c + \epsilon)$.

Let's consider a case where there are two equally probable outcomes y_1 and y_2 for an input variable x. It is easy to verify that fitting the mixture distribution in equation 2 will force $\mathcal{A}L$ components with $\tau \in \mathcal{B}(0.5, \epsilon)$ to settle around the median which is $\mu_\tau(x) = (y_1 + y_2)/2$. However, in this scenario, the median is not of high interest as we like our model to concentrate the probability density around y_1 and y_2. This example indicates that depending on the problem at hand, not all quantiles are equally important. To compensate for this, the mixture weights of $\mathcal{A}L$ components corresponding to non-informative quantiles are expected to shrink to zero. Theoretically, where we have infinite samples of τ, this is not an issue, but in practice, the expectation in equation 2 is approximated by a finite number of Monte Carlo samples. In other words, we can approximate the integral up to a certain precision which depends on the number of samples of τ. Hence, it would be more efficient to avoid drawing samples of τ that correspond to non-informative quantiles. In our example, we like to sample τ such that $\mu_\tau \in \mathcal{B}(y_1, \epsilon) \cup \mathcal{B}(y_2, \epsilon)$. This will yield a more accurate approximation of $p(\mathrm{y} \,|\, x)$ around y_1 and y_2 and prevent wasting computing resources.

Another possible issue is regarding making point estimations. In most applications, the final stage involves reporting a set of values predicted by the model for a given input. In the case of using a symmetric unimodal distribution like Gaussian for representing $p(\mathrm{y} \,|\, x)$, we usually report the distribution mean as it coincides with the mode. However, finding the modes of the uncountable mixture in equation 2 is not always straightforward. A naive way to address this is by considering a set of uniformly sampled τ values, calculating their corresponding quantities $\{\alpha_\tau,\, \mu_\tau,\, \sigma_\tau\}$, and finally selecting the most probable quantiles by comparing their mixture weights (or likelihoods). Similar to the previous case, it is quite likely that a small subset of quantiles is of interest. In case of having knowledge of these quantiles, we will be able to select better candidates as our final predictions with less effort which comes as a result of reducing the search space.

Motivated by the above discussion, we propose using an adaptive set of quantiles $\mathcal{T}_x$ for each x where the model learns to approximate the quantiles that are more significant in modeling $p(\mathrm{y} \,|\, x)$. A simple approach for finding such a set incorporates replacing the non-informative uniform distribution $\mathcal{U}(0, 1)$ in equation 2 with $q(\tau \,|\, x)$ (such that $\tau \in (0, 1)$) whose density is mostly concentrated around values corresponding to informative quantiles of $p(\mathrm{y} \,|\, x)$. Note that the dependence of $q(\tau \,|\, x)$ on x can be arbitrarily complex. Therefore, $\mathcal{T}_x$ can be viewed as samples from $q(\tau \,|\, x)$ and the problem of finding $\mathcal{T}_x$ changes to estimation of $q(\tau \,|\, x)$. Notice that conditioning on x is crucial as the conditional distribution $p(\mathrm{y} \,|\, x)$, and, hence, its quantiles (presumably) change at different inputs. Rewriting equation 2 yields:

$$p(\mathrm{y} \,|\, x) = \mathbb{E}_{\tau \sim q(\tau \,|\, x)} \left[\alpha_\tau(x)\, \mathcal{A}L\left(\mathrm{y} \,|\, \mu_\tau(x), \sigma_\tau(x), \tau\right) \right]. \quad (5)$$

The Monte Carlo Approximation of this expectation only requires samples from $q(\tau \,|\, x)$. Hence, having an analytic probability density function for $q(\tau \,|\, x)$ is not necessary as far as samples can be drawn. Various density estimation techniques can be deployed to find $q(\tau \,|\, x)$. Considering the complicated nature of $q(\tau \,|\, x)$, we propose to approximate it with a reparameterizable implicit distribution [Diggle and Gratton, 1984, Mohamed and Lakshminarayanan, 2016]. This means that to sample from $q(\tau \,|\, x)$, we can first draw an auxiliary variable $u \sim \mathcal{U}(0, 1)$ and then set τ as a deterministic function $\psi \colon \mathbb{X} \times (0, 1) \to (0, 1)$ of x and u:

$$u \sim \mathcal{U}(0, 1),\ \tau = \psi(x, u) \quad \equiv \quad \tau \sim q(\tau \,|\, x) \quad (6)$$

When $\psi(x, u)$ is invertible w.r.t. u and for a fixed x, $q(\tau \,|\, x)$ can be calculated by a simple application of the change of variable formula:

$$q(\tau \,|\, x) = \mathbb{1}_{(0,1)}(g_x^{-1}(\tau))\, \frac{d}{d\tau}(g_x^{-1}(\tau)),$$

where $\tau = g_x(u) = \psi(x, u)$. However, this is generally not the case, and hence $q(\tau \,|\, x)$ is implicit. Using equation 6, we can approximate the conditional distribution in equation 5 as follows:

$$p(\mathrm{y} \,|\, x) \approx \mathbb{E}_{u \sim \mathcal{U}(0,1)} \big[\alpha_{\psi(x,u)}(x)\, \mathcal{A}L(\mathrm{y} \,|\, \mu_{\psi(x,u)}(x), \sigma_{\psi(x,u)}(x), \psi(x, u)) \big], \quad (7)$$

where ψ is a fully-connected neural network. The high expressive power of neural networks allows $q(\tau \,|\, x)$ to be highly flexible and capture the dependencies between the elements of x and u.

3.2 CONDITIONAL QUANTILE NEURAL PROCESSES

Despite the attractive properties of likelihood-based models, their expressive power is highly impacted by the form of conditional distribution. Inherently, CNPs with Gaussian likelihood struggle to model more complicated distributions.

We remedy this by adapting the predictive distribution in equation 3 to the compound distribution in equation 7. This requires augmenting the domain of φ_{dec} and ψ as demonstrated below:

$$\psi\colon \mathbb{X} \times (0,1) \to (0,1) \quad \text{to} \quad \psi\colon \mathbb{X} \times \mathbb{R}^{d_e} \times (0,1) \to \mathcal{T}_{\mathbb{X}}$$
$$\varphi_{\text{dec}}\colon \mathbb{X} \times \mathbb{R}^{d_e} \to \Theta \quad \text{to} \quad \varphi_{\text{dec}}\colon \mathbb{X} \times \mathbb{R}^{d_e} \times \mathcal{T}_{\mathbb{X}} \to \Theta$$

Putting all pieces together, the predictive distribution of $f(x^*)$ for a given target input location x^* would be:

$$p(f(x^*) \,|\, x^*, \mathcal{D}) = \mathbb{E}_{u\sim\mathcal{U}(0,1)}[\alpha_\tau(x^*, \mathcal{D}) \times \mathcal{AL}(f(x^*) \,|\, \mu_\tau(x^*, \mathcal{D}), \sigma_\tau(x^*, \mathcal{D}), \tau)] \quad (8)$$

where

$$\tau = \psi(x^*, \varphi_{enc}(x^*, \mathcal{D}), u)$$
$$\{\alpha_\tau(x^*, \mathcal{D}),\, \mu_\tau(x^*, \mathcal{D}),\, \sigma_\tau(x^*, \mathcal{D})\} = \varphi_{dec}(x^*, \varphi_{enc}(x^*, \mathcal{D}), \tau)$$

We refer to this model as Adaptive Conditional Quantile Neural Process (ACQNP). In the case where $\tau = u$, the resulting model is Conditional Quantile Neural Process (CQNP). The expectation in equation 8 is approximated by drawing N_τ Monte Carlo samples from $\mathcal{U}(0,1)$. Unlike Brando et al. [2019], we avoid posing a uniform distribution over the mixing weights α_τ. Instead, we normalize them using a SoftMax function to have a valid convex combination of $\mathcal{AL}$ distributions. The final likelihood can be expressed as follows:

$$p(f(x^*) \,|\, x^*, \mathcal{D}) \approx \sum_{k=1}^{N_\tau} \Big[\frac{e^{\alpha_{\tau_k}(x^*, \mathcal{D})}}{\sum_{k=1}^{N_\tau} e^{\alpha_{\tau_k}(x^*, \mathcal{D})}} \times \mathcal{AL}(f(x^*) \,|\, \mu_{\tau_k}(x^*, \mathcal{D}), \sigma_{\tau_k}(x^*, \mathcal{D}), \tau_k) \Big] \quad (9)$$

Since CNPs and A/CQNPs use the same architectural design for computing the context representation, the computational complexity imposed by the encoder, i.e. $\mathcal{O}(\varphi_{enc})$, remains unchanged. However, the computational complexity of the decoder which comes from estimating N_τ quantiles at each input location is raised from $\mathcal{O}(\varphi_{dec})$ to $\mathcal{O}(N_\tau \varphi_{dec})$ resulting in the overall complexity of $\mathcal{O}(\varphi_{enc} + N_\tau \varphi_{dec})$. This extra computation can be done in parallel as different quantiles are estimated independently and mixed in the final stage.

4 EXPERIMENTS

We evaluate our proposed framework on one and two-dimensional regression tasks. Note that unlike other members of the NPs family that focus on building more expressive encoder-decoder blocks, our work is primarily concerned with the form of conditional likelihood and its effect on the model's performance. Nonetheless, we compare our A/CQNP with Conditional Attentive Neural Process (CANP) [Kim et al., 2019], and Bootstrapping Neural Process (BNP, Lee et al. [2020]) as our baselines to provide a better overview of performance gains obtained by modifying different components of vanilla CNP. Note that, although NPs share fundamental properties with GPs, they usually are not compared directly because of the different training regimes [Kim et al., 2019]. While NPs are trained on different functions sampled from the underlying generative process, GPs are fit to observations corresponding to one realization of the process. In order to compare the goodness of fit across different methods, we report the log-likelihood on context and target data separately. Methods with higher context log-likelihood offer better reconstructions of context points and hence, are less prone to under-fitting, while higher target log-likelihood indicates more accurate predictions [Kim et al., 2019, Lee et al., 2020]. Detailed information on model architectures, training, and testing procedures are included in the supplementary materials.

4.1 SYNTHETIC DATA

We start our study by examination of each model over several synthetically generated datasets. Each collection consists of a handful number of functions sampled from a known stochastic process. In each iteration, a batch of n_b functions $\mathcal{G} = \{g_k\}_{k=1}^{n_b}$ are sampled from a stochastic process such that $g_k : \mathbb{R} \to \mathbb{X} \times \mathbb{Y}$ and $g_k(s) = (g_{k,x}(s),\, g_{k,y}(s))$. For each g_k, a set $\mathcal{S}_k$ of N_{total} random input locations is chosen where $\mathcal{S}_k = \{s_{k,l}\}_{l=1}^{N_{\text{total}}}$ and $s_{k,l} \sim \mathcal{U}[I_{\min},\, I_{\max}]$ ($I_{\min}$ and $I_{\max}$ are fixed constants). Applying g_k to the corresponding $\mathcal{S}_k$ will yield a collection of pairs $\mathcal{E}_k = \{(x_{k,l},\, y_{k,l})\}_{l=1}^{N_{\text{total}}}$ where $(x_{k,l},\, y_{k,l}) = (g_{k,x}(s_{k,l}),\, g_{k,y}(s_{k,l}))$. By repeating this process for each k, we end up with a hierarchical dataset[1] $\mathcal{E} = \{\mathcal{E}_k\}_{k=1}^{n_b}$ [Garnelo et al., 2018a,b]. The variable s

[1]A hierarchical dataset is a collection of observations from many functions sharing some underlying characteristics.

Table 1: Synthetic processes used in multimodal 1D regression experiments.

Process	$g(s) = (g_x(s),\, g_y(s))$
Double Sine	$g_x(s) = s\,,\; g_y(s) = \alpha_1 \sin(\omega_1 s)\, \mathbb{1}_{(0,0.5)}(p(s)) + \alpha_2 \cos(\omega_2 s)\, \mathbb{1}_{[0.5,1)}(p(s))$
Circle	$g_x(s) = \alpha \cos(s) + \delta\,,\; g_y(s) = \alpha \sin(s) + \delta$
Lissajous	$g_x(s) = \alpha_1 \sin(\omega s + \delta)\,,\; g_y(s) = \alpha_2 \sin(s)$

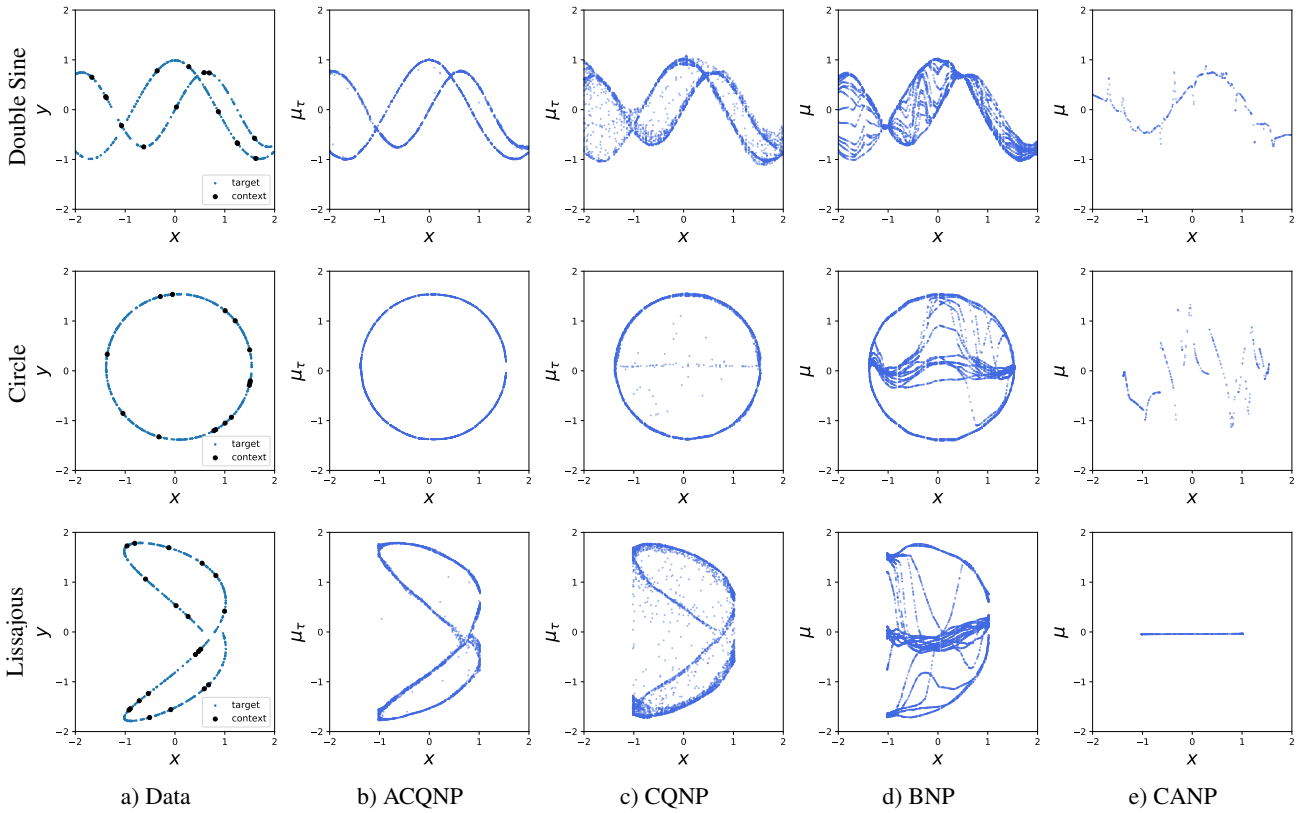
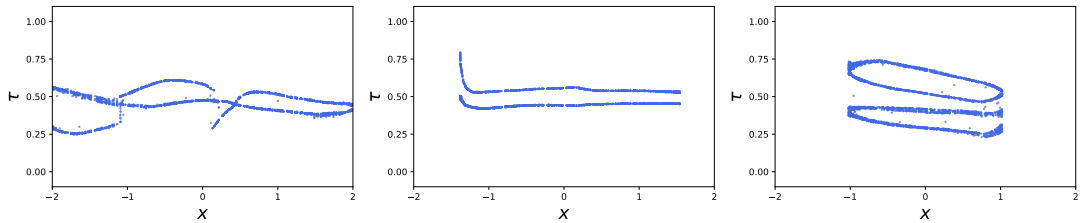


Figure 1: Examples of predictions made by different methods on synthetic datasets. For A/CQNP, 10 randomly chosen conditional quantiles of $p(\mathrm{y} \mid x)$ at each input location x are plotted. For BNP, we plot the conditional means of the Gaussian predictive distributions obtained from 20 different sets of bootstrap contexts. Similarly, we plot the mean of CANP's conditional distribution as its predictions.

is discarded after sampling $\mathcal{E}_k$. In the course of training, the total number of data points N_{total} is randomly chosen for each batch such that $N_{\text{total}} \sim \mathcal{U}[6, 100]$. Lastly, each $\mathcal{E}_k$ is split into context and target sets by choosing a random index $N_{\text{context}} \sim \mathcal{U}[3, N_{\text{total}} - 3]$ and setting $\mathcal{E}_{k,\text{context}} = \{(x_{k,l}, y_{k,l})\}_{l=1}^{N_{\text{context}}}$ and $\mathcal{E}_{k,\text{target}} = \{(x_{k,l}, y_{k,l})\}_{l=N_{\text{context}}+1}^{N_{\text{total}}}$. During testing, however, we fix $N_{\text{total}} = 500$ and select $N_{\text{context}} \sim \mathcal{U}[3, 100]$. Each method is trained and tested over 10^5 and 10^3 batches with $n_b = 128$ and $n_b = 16$, respectively. Note that, unlike training data which is generated in each iteration of training, we fix the testing data across different models by generating them in advance. We consider data arising from the three stochastic processes described in table 1 with the following choice of parameters:

- Double-Sine: $s \sim \mathcal{U}[-2, 2)$, $\alpha_1, \alpha_2 \sim \mathcal{U}[0.5, 1.5)$, $\omega_1, \omega_2 \sim \mathcal{U}[1, 3)$ and $p(s) \sim \mathcal{U}[0, 1)$
- Circle: $s \sim \mathcal{U}[-\pi, \pi)$, $\alpha \sim \mathcal{U}[0.5, 1.5)$, $\delta \sim \mathcal{U}[-0.5, 0.5)$
- Lissajous: $s \sim \mathcal{U}[-\pi, \pi)$, $\alpha_1, \alpha_2 \sim \mathcal{U}[1, 2)$, $\omega \sim [0.5, 2)$, $\delta \sim \mathcal{U}[0, 2)$

Table 2 summarizes the predictive log-likelihood of different methods over testing data. We see that A/CQNP constantly outperforms baselines despite using the vanilla deterministic encoder in CNP which: 1) does not incorporate any latent variable for capturing the functional uncertainty and correlations as in BNP, and 2) does not enjoy the expressive context representations provided by the attention mechanism used in CANP. As depicted in figure 1, CANP fails to capture multimodality and instead tends to make predictions that resemble the average of modes. This happens less severely with BNP. Nonetheless, the sampled curves bounce between the modes which results in unnecessarily wide prediction bands. CQNP, on the other hand, provides decent fits which are further polished by incorporating the quantile adaptation structure in ACQNP. Figure 2 illustrates the quantile levels τ that were adapted by ACQNP for predicting the quantiles shown in figure 1b. We see that ACQNP behaves in line with our motivations behind adaptive quantile regression discussed in section 3.1. The distribution of τ imitates the distribution of the data, in the sense that $p(\mathrm{y} \mid x, \mathcal{D})$ and $p(\tau \mid x, \mathcal{D})$ have the same number of modes almost everywhere. Additional experimental results on unimodal regression tasks are provided in the supplements.



a) Double Sine b) Circle c) Lissajous

Figure 2: Distribution of the adapted τ levels corresponding to the estimated quantiles in figure 1b after applying the transformation ψ in equation 8.

Table 2: Comparison of predictive log-likelihood obtained by different methods over synthetically generated tasks (6 Seeds).

	Double Sine		Circle		Lissajous	
	context	target	context	target	context	target
CNP	$-0.195_{\pm 0.009}$	$-0.520_{\pm 0.019}$	$-2.086_{\pm 0.204}$	$-2.387_{\pm 0.216}$	$-2.212_{\pm 0.146}$	$-2.962_{\pm 0.165}$
CANP	$0.436_{\pm 0.236}$	$-1.742_{\pm 0.222}$	$-0.272_{\pm 0.043}$	$-1.685_{\pm 0.082}$	$-1.112_{\pm 0.499}$	$-2.151_{\pm 0.248}$
BNP	$0.330_{\pm 0.010}$	$0.134_{\pm 0.017}$	$0.150_{\pm 0.010}$	$0.065_{\pm 0.009}$	$-0.314_{\pm 0.011}$	$-0.434_{\pm 0.010}$
CQNP(ours)	$\mathbf{1.448}_{\pm 0.042}$	$\mathbf{1.244}_{\pm 0.049}$	$\mathbf{2.047}_{\pm 0.076}$	$\mathbf{1.932}_{\pm 0.080}$	$\mathbf{0.798}_{\pm 0.020}$	$\mathbf{0.508}_{\pm 0.021}$
ACQNP(ours)	$\mathbf{1.582}_{\pm 0.108}$	$\mathbf{1.349}_{\pm 0.098}$	$\mathbf{2.118}_{\pm 0.059}$	$\mathbf{2.028}_{\pm 0.057}$	$\mathbf{0.929}_{\pm 0.038}$	$\mathbf{0.634}_{\pm 0.034}$

4.2 SPEED-FLOW DATA

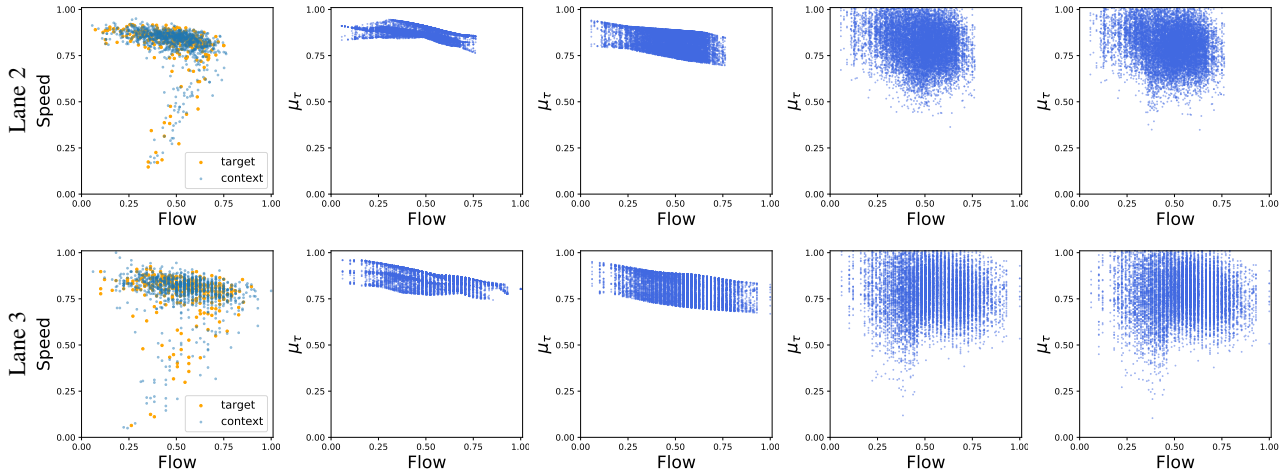
The problem of traffic speed prediction has been widely investigated in transportation science with applications in approximating the expected arrival time [Einbeck and Tutz, 2006]. A key factor in such models is the traffic flow which is usually presented by speed-flow diagrams. As a case study, we consider the speed-flow data collected by Petty et al. [1996]. The dataset contains the speed-flow diagrams of lanes 2 and 3 of a four-lane Californian freeway with 1318 measurements for each lane (Figure 3a) and is included in the R-package `hdrcde` [Hyndman et al., 2022]. This collection can be viewed as an example of a hierarchical dataset where observations from different lanes correspond to different realizations of a random process. The hierarchical structure of this data makes it an ideal fit with NPs as they allow for sharing information among different lanes. We randomly select 75% of each lane's observations (≈ 988) for training and use the rest for testing. The speed and flow values are both scaled to $[0, 1]$. In each training iteration, we split the training data into context and target sets such that $N_{\text{context}} \sim \mathcal{U}[500, 985]$. For testing, however, we take the context and target sets to be the training and testing data, respectively. Table 3 provides a quantitative comparison of different approaches, with A/CQNP being the best-performing model on this real-world task. As shown in figure 3, the Gaussian predictive distributions in BNP and CANP, which resemble the conditional mean regression, lead to wide prediction bands compared to A/CQNP. This is due to the sensitivity of mean estimators in dealing with the less dense cloud of data points at the bottom which can be interpreted as outliers [Feng et al., 2020]. In contrast to the mean which acts on a global level, the local nature of quantiles makes them more robust to the tail behavior.

Table 3: Context and target log-likelihoods from experiments on speed-flow data (6 Seeds).

	Speed-Flow	
	context	target
CNP	$0.845_{\pm 0.010}$	$0.719_{\pm 0.002}$
CANP	$0.887_{\pm 0.010}$	$0.741_{\pm 0.014}$
BNP	$0.879_{\pm 0.005}$	$0.720_{\pm 0.015}$
CQNP(ours)	$\mathbf{1.518}_{\pm 0.013}$	$\mathbf{1.495}_{\pm 0.007}$
ACQNP(ours)	$\mathbf{1.544}_{\pm 0.001}$	$\mathbf{1.507}_{\pm 0.006}$

4.3 IMAGE COMPLETION

A collection of images can also be thought of as a hierarchical dataset where each image is a realization of some random process mapping 2D pixel coordinates to pixel intensities. Motivated by this observation, image inpainting can be framed as a regression problem where conditioned on a set of observed pixels, we are interested in filling the missing regions of the image. MNIST [LeCun et al., 1998], Fashion-MNIST [Xiao et al., 2017], SVHN [Netzer et al., 2011], Omniglot [Lake et al., 2015] (resized to 32 $\times$32) and FreyFace [Roweis et al., 2001] are the datasets that we consider here. Except for FreyFace, we use the default train/test split used by the publishers. For FreyFace, we randomly select 75% of the images for training and keep the rest for testing. For all the benchmarks, the pixel values and pixel



a) Data b) ACQNP c) CQNP d) BNP e) CANP

Figure 3: First column from right includes the ground truth data. Second and third columns show the quantiles of the predictive distribution of A/CQNP at 10 different levels. Fourth column shows the ensembled quantiles at the same levels for 5 bootstrap context sets. Last column shows the conditional quantiles of CANP at the same levels.

coordinates are rescaled to $[0, 1]$ and $[-1,\ 1]$ respectively. In each case, we take $\mathcal{E}_k$ as the set of all image pixels. Similar to section 4.1, the context and target sets in both training and testing are chosen such that $N_{\text{context}} \sim \mathcal{U}[3, N_{\text{total}}/2)$ where N_{total} is the number of image pixels. Table 4 shows that A/CQNP substantially outperforms the baselines across all datsets in terms of predictive log-likelihood. This holds for both context and target sets revealing that A/CQNP yields better reconstruction of context data. Moreover, results from Omniglot experiments suggest that A/CQNP has better generalization capabilities as the default test split has distinct classes from training.

4.4 ABLATION STUDY

Number of quantiles N_τ. As mentioned earlier in section 3.2, we approximate the conditional likelihood by N_τ Monte Carlo samples. While using a larger sample size offers more precise approximation, it demands further computational resources which highlights the importance of a rather fine-grained scheme for sampling τ instead of random draws from a uniform distribution, especially when we are restricted to work with a small N_τ. To check if the adaptive mechanism can alleviate this issue, we compare the predictive performance of ACQNP against CQNP on Lissajous curves studied in section 4.1. Both models have the same architectural design for their encoder/decoder modules and are trained with $N_\tau = 50$. During testing, however, we evaluate each method with $N_\tau \in \{3, 5, 7, 9, 11\}$ as depicted in figure 4a. Note that in addition to testing data, we also fix the input quantile levels u to be $\{u_0, u_0+d, \ldots, u_0+(N_\tau-2)d, u_1\}$ where $u_0 = 0.001$, $u_1 = 0.999$, and $d = \frac{u_1-u_0}{N_\tau-1}$. It can be seen that the log-likelihood of CQNP decreases significantly with smaller sample sizes, whereas ACQNP suffers less as it can manage to use its few shots efficiently and locate informative quantiles.

Flexibility of ψ. The adaptive process that we followed in this paper works by transforming u through some nonlinear

Table 4: Context and target log-likelihoods on 2D regression tasks (6 Seeds).

	MNIST		FashionMNIST		SVHN		Omniglot		FreyFace	
	context	target	context	target	context	target	context	target	context	target
CNP	1.061 (±0.006)	0.938 (±0.001)	0.963 (±0.005)	0.872 (±0.004)	3.554 (±0.014)	3.388 (±0.013)	0.978 (±0.004)	0.874 (±0.009)	0.970 (±0.083)	0.941 (±0.088)
CANP	1.350 (±0.003)	0.913 (±0.006)	1.226 (±0.007)	0.857 (±0.024)	4.112 (±0.002)	3.715 (±0.016)	1.366 (±0.005)	0.974 (±0.004)	1.062 (±0.053)	1.015 (±0.023)
BNP	1.128 (±0.013)	1.061 (±0.009)	1.039 (±0.002)	0.971 (±0.001)	3.679 (±0.008)	3.580 (±0.007)	0.983 (±0.006)	0.950 (±0.004)	1.073 (±0.015)	1.052 (±0.014)
CQNP(ours)	**2.683** (±0.006)	**2.609** (±0.006)	**2.012** (±0.004)	**1.932** (±0.002)	**4.725** (±0.033)	**4.447** (±0.029)	**2.529** (±0.034)	**2.471** (±0.038)	**1.433** (±0.062)	**1.392** (±0.061)
ACQNP(ours)	**2.681** (±0.011)	**2.616** (±0.013)	**2.040** (±0.016)	**1.954** (±0.015)	**4.959** (±0.042)	**4.651** (±0.034)	**2.516** (±0.009)	**2.461** (±0.007)	**1.522** (±0.089)	**1.475** (±0.086)

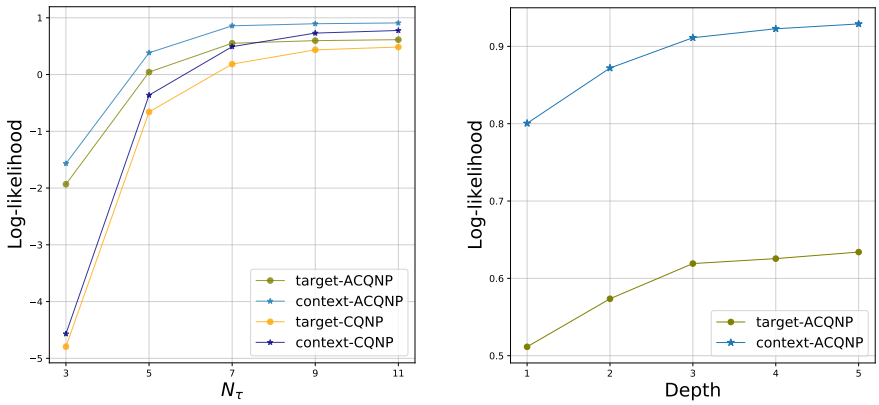


Figure 4: **Left**: Comparison of the predictive log-likelihood over context and target points for A/CQNP with different values of N_τ during testing (6 seeds); **Right**: Predictive performance of ACQNP versus the depth of ψ (6 seeds).

mapping ψ introduced in equation 7; thus, the choice of this mapping is expected to affect the performance. Throughout this work, we modeled ψ with a fully-connected neural network. We investigate the effect of the depth of the neural network as a measure of its expressive power. As illustrated in figure 4, a deeper neural network improves the overall performance of the model. However, the performance gain comes at the cost of additional memory usage and computational complexity which needs to be considered as a tradeoff.

5 RELATED WORKS

Vector quantile regression Quantile regression (QR), introduced by Koenker and Bassett [1978], is a compelling statistical technique that can be used for studying the dependence between random variables by modeling the conditional quantiles of a target variable as a function of some explanatory variables. Unfortunately, the QR framework only considers scalar target variables as the notion of quantile is not well-defined in higher dimensions. We can apply QR to scalar components of a vector-valued response variable by assuming independence among them. This assumption, however, is usually violated. Recently, Carlier et al. [2016], Chernozhukov et al. [2017], Carlier et al. [2017] introduced vector quantiles as extensions to the univariate quantiles which allows for vector quantile regression (VQR). Rosenberg et al. [2022] propose nonlinear vector quantile regression (NL-VQR) which drops the restrictive specification of a linear conditional quantile function. They further introduce a solver allowing for applying their method to large datasets.

Function-space inference There is a growing line of research on using neural networks for direct parameterization of distributions over functions. NPs and their variants are well-known byproducts of this viewpoint. Ma et al. [2019] introduced implicit processes (IPs) as priors over functions by placing an implicit joint distribution over any finite collection of random variables. However, they use GPs for approximating the intractable posterior of IPs which is: 1) computationally expensive, and 2) limited by the Gaussian likelihood assumption. Sparse implicit processes (SIPs, Rodríguez-Santana et al. [2022]) try to address these issues by relying on inducing points [Snelson and Ghahramani, 2005] and using a mixture of Gaussians as the predictive distribution. Yang et al. [2020] use energy-based models (EBMs) to construct a family of expressive stochastic processes for exchangeable data. The additional flexibility, however, comes at the cost of complicated training and inference schemes requiring several approximations. Inspired by contrastive methods [Durkan et al., 2020, Gutmann and Hyvärinen, 2010], Mathieu et al. [2021] drop the explicit likelihood requirement used in NPs which necessitates exact reconstruction of observations. Despite higher tolerance in facing noisy high-dimensional data, their method is incapable of uncertainty quantification.

6 CONCLUSIONS

In this paper, we proposed Conditional Quantile Neural Processes (CQNPs), a new member of the CNP family that leverages advances in quantile regression to increase the expressive power of CNPs in modeling heterogeneous distributions. Furthermore, we introduced an extended framework for quantile regression, named adaptive quantile regression, where instead of fixing the quantile levels, the model gets to choose which quantiles to estimate. Our experiments with several synthetic and real-world datasets showed that A/CQNPs improve the predictive performance of CNPs across regression tasks in terms of log-likelihood, and faithfully model multimodality, skewness, and long tails in predictive distributions.

References

Axel Brando, Jose A Rodriguez, Jordi Vitria, and Alberto Rubio Muñoz. Modelling heterogeneous distributions with an uncountable mixture of asymmetric laplacians. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Wessel Bruinsma, James Requeima, Andrew Y. K. Foong, Jonathan Gordon, and Richard E Turner. The gaussian neural process. In *Third Symposium on Advances in Approximate Bayesian Inference*, 2021.

Guillaume Carlier, Victor Chernozhukov, and Alfred Galichon. Vector quantile regression: an optimal transport approach. *The Annals of Statistics*, 44(3):1165–1192, 2016.

Guillaume Carlier, Victor Chernozhukov, and Alfred Galichon. Vector quantile regression beyond the specified case. *Journal of Multivariate Analysis*, 161:96–102, 2017. ISSN 0047-259X.

George Casella and Roger L Berger. *Statistical inference*. Cengage Learning, 2021.

Yen-Chi Chen, Christopher R Genovese, Ryan J Tibshirani, and Larry Wasserman. Nonparametric modal regression. *The Annals of Statistics*, 44(2):489–514, 2016.

Victor Chernozhukov, Alfred Galichon, Marc Hallin, and Marc Henry. Monge-kantorovich depth, quantiles, ranks and signs. *The Annals of Statistics*, 45(1):223–256, 2017. ISSN 00905364.

Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. In *International conference on machine learning*, pages 1096–1105. PMLR, 2018.

Peter J. Diggle and Richard J. Gratton. Monte carlo methods of inference for implicit statistical models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 46 (2):193–227, 1984. ISSN 00359246.

Conor Durkan, Iain Murray, and George Papamakarios. On contrastive learning for likelihood-free inference. In *International Conference on Machine Learning*, pages 2771–2781. PMLR, 2020.

Jochen Einbeck and Gerhard Tutz. Modelling beyond regression functions: An application of multimodal regression to speed-flow data. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 55(4):461–475, 2006. ISSN 00359254, 14679876.

Yunlong Feng, Jun Fan, and Johan A.K. Suykens. A statistical learning approach to modal regression. *Journal of Machine Learning Research*, 21(2):1–35, 2020.

Andrew Foong, Wessel Bruinsma, Jonathan Gordon, Yann Dubois, James Requeima, and Richard Turner. Meta-learning stationary stochastic process prediction with convolutional neural processes. *Advances in Neural Information Processing Systems*, 33:8284–8295, 2020.

Marta Garnelo, Dan Rosenbaum, Christopher Maddison, Tiago Ramalho, David Saxton, Murray Shanahan, Yee Whye Teh, Danilo Rezende, and S. M. Ali Eslami. Conditional neural processes. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1704–1713. PMLR, 10–15 Jul 2018a.

Marta Garnelo, Jonathan Schwarz, Dan Rosenbaum, Fabio Viola, Danilo J Rezende, SM Eslami, and Yee Whye Teh. Neural processes. *arXiv preprint arXiv:1807.01622*, 2018b.

Jonathan Gordon, Wessel P. Bruinsma, Andrew Y. K. Foong, James Requeima, Yann Dubois, and Richard E. Turner. Convolutional conditional neural processes. In *International Conference on Learning Representations*, 2020.

Zongyu Guo, Cuiling Lan, Zhizheng Zhang, Zhibo Chen, and Yan Lu. Versatile neural processes for learning implicit neural representations. *arXiv preprint arXiv:2301.08883*, 2023.

Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010.

Peter Holderrieth, Michael J Hutchinson, and Yee Whye Teh. Equivariant learning of stochastic fields: Gaussian processes and steerable conditional neural processes. In *International Conference on Machine Learning*, pages 4297–4307. PMLR, 2021.

Rob J Hyndman, Jochen Einbeck, and Matthew P Wand. *hdrcde: Highest Density Regions and Conditional Density Estimation*, 2022. R package version 3.4.

Hyunjik Kim, Andriy Mnih, Jonathan Schwarz, Marta Garnelo, Ali Eslami, Dan Rosenbaum, Oriol Vinyals, and Yee Whye Teh. Attentive neural processes. In *International Conference on Learning Representations*, 2019.

Mingyu Kim, Kyeongryeol Go, and Se-Young Yun. Neural processes with stochastic attention: Paying more attention to the context dataset. *arXiv preprint arXiv:2204.05449*, 2022.

Roger Koenker. *Quantile Regression*. Econometric Society Monographs. Cambridge University Press, 2005. doi: 10.1017/CBO9780511754098.

Roger Koenker and Gilbert Bassett. Regression quantiles. *Econometrica*, 46(1):33–50, 1978. ISSN 00129682, 14680262.

Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Juho Lee, Yoonho Lee, Jungtaek Kim, Eunho Yang, Sung Ju Hwang, and Yee Whye Teh. Bootstrapping neural processes. *Advances in neural information processing systems*, 33:6606–6615, 2020.

Yufeng Liu and Yichao Wu. Stepwise multiple quantile regression estimation using non-crossing constraints. *Statistics and its Interface*, 2(3):299–310, 2009.

Yufeng Liu and Yichao Wu. Simultaneous multiple non-crossing quantile regression estimation using kernel constraints. *Journal of nonparametric statistics*, 23(2):415–437, 2011.

Chao Ma, Yingzhen Li, and Jose Miguel Hernandez-Lobato. Variational implicit processes. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4222–4233. PMLR, 09–15 Jun 2019.

Stratis Markou, James Requeima, Wessel P Bruinsma, Anna Vaughan, and Richard E Turner. Practical conditional neural processes via tractable dependent predictions. *arXiv preprint arXiv:2203.08775*, 2022.

Emile Mathieu, Adam Foster, and Yee Teh. On contrastive representations of stochastic processes. *Advances in Neural Information Processing Systems*, 34:28823–28835, 2021.

Shakir Mohamed and Balaji Lakshminarayanan. Learning in implicit generative models. *arXiv preprint arXiv:1610.03483*, 2016.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

Tung Nguyen and Aditya Grover. Transformer neural processes: Uncertainty-aware meta learning via sequence modeling. *arXiv preprint arXiv:2207.04179*, 2022.

Karl F. Petty, Hisham Noeimi, Kumud Sanwal, Dan Rydzewski, Alexander Skabardonis, Pravin Varaiya, and Haitham Al-deek. The freeway service patrol evaluation project: Database support programs, and accessibility. *Transportation Research Part C: Emerging Technologies*, 4(2):71–85, 1996. ISSN 0968-090X.

Simon Rodríguez-Santana, Bryan Zaldivar, and Daniel Hernandez-Lobato. Function-space inference with sparse implicit processes. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 18723–18740. PMLR, 17–23 Jul 2022.

Aviv A Rosenberg, Sanketh Vedula, Yaniv Romano, and Alex M Bronstein. Fast nonlinear vector quantile regression. *arXiv preprint arXiv:2205.14977*, 2022.

Sam Roweis, Lawrence Saul, and Geoffrey E Hinton. Global coordination of local linear models. *Advances in neural information processing systems*, 14, 2001.

Tim GJ Rudner, Vincent Fortuin, Yee Whye Teh, and Yarin Gal. On the connection between neural processes and gaussian processes with deep kernels. In *Workshop on Bayesian Deep Learning, NeurIPS*, page 14, 2018.

Maxime Sangnier, Olivier Fercoq, and Florence d'Alché-Buc. Joint quantile regression in vector-valued rkhss. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

Edward Snelson and Zoubin Ghahramani. Sparse gaussian processes using pseudo-inputs. *Advances in neural information processing systems*, 18, 2005.

Michael Volpp, Fabian Flürenbrock, Lukas Grossberger, Christian Daniel, and Gerhard Neumann. Bayesian context aggregation for neural processes. In *International Conference on Learning Representations*, 2021.

Qi Wang and Herke Van Hoof. Doubly stochastic variational inference for neural processes with hierarchical latent variables. In *International Conference on Machine Learning*, pages 10018–10028. PMLR, 2020.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017.

Mengjiao Yang, Bo Dai, Hanjun Dai, and Dale Schuurmans. Energy-based processes for exchangeable data. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 10681–10692. PMLR, 13–18 Jul 2020.

Keming Yu and Rana A. Moyeed. Bayesian quantile regression. *Statistics & Probability Letters*, 54(4):437–447, 2001. ISSN 0167-7152.

Keming Yu and Jin Zhang. A three-parameter asymmetric laplace distribution and its extension. *Communications in Statistics - Theory and Methods*, 34(9-10):1867–1879, 2005.