Probabilistic Conformal Prediction Using Conditional Random Samples

Zhendong Wang *†
zhendong.wang@utexas.edu

Ruijiang Gao*†
ruijiang@utexas.edu

Mingzhang Yin*‡
my2674@columbia.edu

Mingyuan Zhou †
mingyuan.zhou@mccombs.utexas.edu

David M. Blei ‡
david.blei@columbia.edu

June 22, 2022

Abstract

This paper proposes probabilistic conformal prediction (PCP), a predictive inference algorithm that estimates a target variable by a discontinuous predictive set. Given inputs, PCP construct the predictive set based on random samples from an estimated generative model. It is efficient and compatible with either explicit or implicit conditional generative models. Theoretically, we show that PCP guarantees correct marginal coverage with finite samples. Empirically, we study PCP on a variety of simulated and real datasets. Compared to existing methods for conformal inference, PCP provides sharper predictive sets.

1 Introduction

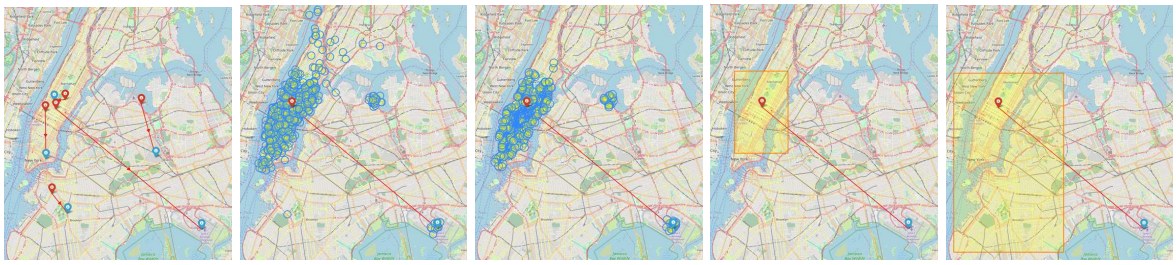
A core problem in supervised machine learning (ML) is to predict a target variable $Y \in \mathcal{Y}$ given a vector of inputs $X \in \mathbb{R}^p$. In this problem, a predictive function $q(Y \mid X)$ is fitted on an observed dataset $\mathbf{D} = \{(X_i, Y_i)\}_{i=1}^N$ and then used to predict the target Y_{N+1} of a new data point with inputs X_{N+1}. While much of machine learning focuses on point predictions of Y, the problem of predictive inference aims at more robust prediction. In predictive inference, our goal is to create a *predictive set* that is likely to contain the unobserved target [12].

In particular, the field of *conformal inference* develops predictive inference algorithms that aim for calibrated coverage probabilities [30, 39]. Assume the data pairs (X_i, Y_i) are sampled independent and identically distributed (iid) from a population distribution $\mathbb{P}(X, Y)$. Given an input X, a conformal inference algorithm provides a set $C_\alpha(X)$ such that

$$\mathbb{P}_{X,Y}(Y \in \hat{C}_\alpha(X)) \geq 1 - \alpha. \tag{1}$$

The scalar $\alpha \in [0, 1]$ is a predefined miscoverage rate and $\hat{C}_\alpha(X) \subset \mathcal{Y}$ is the predictive set. A set that satisfies Equation (1) is called a *valid* predictive set. Since the trivial set $\hat{C}_\alpha(X) = \mathcal{Y}$ is valid, one goal of conformal inference is to keep the size of the predictive set small and (thus) informative. This property is known as *sharpness* [10, 21]. In this paper, we develop a new method for conformal inference that produces valid and sharp predictive sets.

*The first three authors contributed equally.
†University of Texas at Austin
‡Columbia University



(a) Sample Data (b) PCP (Ours) (c) HD-PCP (Ours) (d) CDSplit [16] (e) CHR [33]

Figure 1: NYC Taxi data. The covariates are pickup location (red pin) and other passenger information; The targe is the dropoff location (blue pin). Left to right: Five random samples from Travel Data; Predictive sets output by PCP, HD-PCP, CDSplit [16] and CHR [33] for one travel record.

Existing conformal inference methods often produce a continuous interval as the predictive set [3, 8, 20, 26, 31, 33]. Such intervals are appropriate in some predictive situations. However, consider a target distribution with separated high-density regions. In this setting, validity comes at the cost of sharpness [15]: to ensure validity the set must include all of the high-density regions; but since it's continuous it must also include the low-density regions between them.

For example, consider a prediction problem that estimates the drop-off location of a taxi passenger based on the passenger's information. The target distribution is likely to be multimodal, centered around locations such as tourist attractions and transit centers [37]. A continuous predictive set will have to emcompass these regions, regardless of how far apart they are. A more informative set would contain the regions themselves, but not the areas between them. Other examples of multimodal targets include the effects of a stroke on brain regions [13], and the rewards of actions of a robot [28].

Figure 1 provides an example of the taxi application. Panel (a) illustrates the data, the destinations of New York taxi passengers. Given a new set of inputs, panels (d) and (e) show existing methods for conformal inference, which predict large regions for the possible destination. Panels (b) and (c) show the results of our algorithms, which form sharper predictive sets from distinct subregions of the map. Our method is called *probabilistic conformal prediction* (PCP). Figure 2 illustrates the algorithm.

In more detail, PCP builds on the split conformal prediction framework [20, 30]. It begins by randomly splitting the observed data $\mathbf{D}$ into a preliminary set $\mathbf{D}_{\text{pre}}$ and a calibration set $\mathbf{D}_{\text{cal}}$. It then has three stages. (1) It fits a conditional generative model $q(Y \mid X)$ to the preliminary data $\mathbf{D}_{\text{pre}}$. (2) For each point (X_i, Y_i) in the calibration set $\mathbf{D}_{\text{cal}}$, it generates K independent samples of preditions $\hat{\mathbf{Y}}_{X_i} = \{\hat{Y}_{i1}, \cdots, \hat{Y}_{iK}\}$ from the fitted model $q(Y \mid X_i)$. It then calculates the distance between each sampled predition and the true label Y_i. These quantities are called the *nonconformity scores* and measure the goodness-of-fit of the generative model. (3) Finally, it calculates and records the $(1-\alpha)$ empirical quantile of the nonconformity scores. These will be used to for the predictive sets.

With these calculations in place, PCP can form the predictive set of a new datapoint. First it generates sampled predictions from the fitted target distribution. Then each sample is expanded to a ball that centers at its point and has a radius equal to the quantile computed from the calibration set. Finally, the predictive set is defined as the union of the balls over the samples. Because it is centered at high-density regions, this predictive set is sharp. Further, as we prove below, it is valid.

There are several advantages to PCP (and a related extension, high-density PCP). First, it adapts automatically to the landscape of the target distribution, providing sharp and valid predictive sets regardless of the underlying distribution. Second, the generative model for PCP may have an explicit or implicit density function as long as the random samples can be generated from it. Without requiring an explicit density, PCP is compatible with the likelihood-free prediction [1, 7] and is less prone to model misspecification [27, 42]. Last, (HD-)PCP can be applied to multi-target regression where the target variable $Y \in \mathcal{Y} = \mathbb{R}^T$, $T \geq 1$ [5, 26]. As we shall see, (HD-)PCP scales efficiently with the target

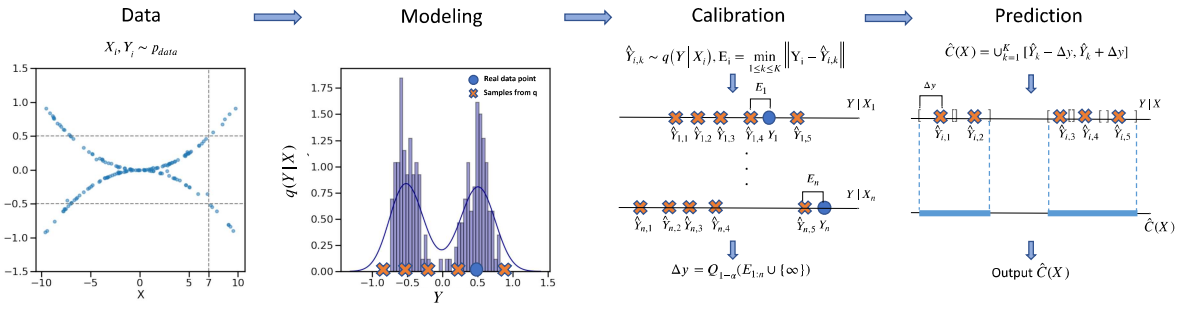


Figure 2: Illustration of the stages of PCP. Data: i.i.d data $\mathbf{D} = \{(X_i, Y_i)\}_{i=1}^N$; Modeling: generate K random samples from a fitted $q(Y \mid X)$; Calibration: compute scores E_i and the quantile Δy; Prediction: create the predictive set $\hat{C}(X)$ for a test data.

dimension and creates a sharp predictive set by capturing the targets' dependencies.

Related Work. PCP provides a contribution to the growing field of conformal inference. Some conformal inference methods are based on predicting summary statistics of the target distribution, for example, by fitting a mean response function [22, 34], conditional quantile functions [31] and approximate histograms [33]. However, these methods produce a single continuous interval as the predictive set, which might be too loose for predicting multimodal targets.

Other conformal inference algorithms estimate the full target distribution. Distributional conformal prediction (DCP) is based on the estimated cumulative density function [8] but its prediction is often sensitive to the tail estimation [33]. CDSplit uses a level set of the estimated probability density function as the predictive set [16]. Similar to PCP, CDSplit can produce discontinuous predictive sets. However, the level set might be loose when the distribution has high dispersion and it has to be computed approximately. Thanks to its sampling-based design, PCP is more computationally efficient than CDSplit, and further it is compatible with likelihood-free predictions [1]. Empirically, across multiple datasets, PCP creates sharper predictive sets than these existing conformal methods.

Finally, there are a few conformal methods for multi-target regression [25, 26, 29]. Compared to these methods, PCP models the target variables jointly and can produce discontinuous predictive sets. As we show in the empirical studies, PCP provides sharper and more interpretable predictions.

2 Probabilistic conformal prediction

2.1 Problem setup

Consider i.i.d. pairs of covariates X_i and a target variable Y_i, i.e. $\mathbf{D} = \{(X_i, Y_i)\}_{i=1}^N$, from an underlying distribution. We observe data $\mathbf{D}$ and the covariates X_{N+1} of a new data point. The goal is to form a predictive set $\hat{C}(X_{N+1})$ for the unobserved target Y_{N+1} with valid uncertainty estimation. Specifically, we create a predictive set $\hat{C}_\alpha(\cdot) : \mathcal{X} \mapsto \mathcal{Y}$ that satifies Equation (1) for $\alpha \in [0, 1]$. Since an arbitrary wide predictive set has valid coverage, a predictive set should be as sharp as possible.

Classic conformal prediction is based on leave-one-out estimation [39], which has high computational cost due to multiple model fitting. In this paper, we adopt the split conformal prediction framework, which improves computational efficiency by data-splitting [22, 30]. It randomly splits the observed data to a preliminary set and a calibration set. The model is fit on the preliminary set and kept fixed in computing the nonconformity scores on the calibration set and the test set.

2.2 Generative model fitting

Our proposed PCP depends on random samples from a conditional generative model $q(Y|X)$ that approximates the target variable distribution $p(Y|X)$. This differs from standard conformal prediction methods that are based on fitting the summary statistics such as the conditional mean and quantiles of the target [20, 31] and that depend on evaluating probability densities [8, 15, 16]. Since the only prerequisite is to sample from $q(Y|X)$, we consider both typical conditional density estimation methods with explicit density function and popular generative models with implicit density.

PCP is compatible with a variety of CGMs, such as the Kernel Mixture Network (KMN) [2], Mixture Density Network (MixD) [4], Quantile Regression Forest (QRF) [24] and implicit generative models. We refer to Appendix D for more details about fitting a CGM and generating random samples from it. We regard CGMs as backbone models for PCP.

2.3 Uncertainty calibration with random samples

Suppose a conditional density model is fit on a preliminary data set $\mathbf{D}_{\text{pre}}$. We use the fitted model $q(Y|X)$ and the calibration data to construct a predictive set for a new test data point. For a data point (X_i, Y_i) in the calibration set, the algorithm first generates K random samples $\hat{Y}_{ik}$, $k = 1, \cdots, K$ independently from $q(Y|X_i)$, denoted as $\hat{\mathbf{Y}}_i = \{\hat{Y}_{i1}, \cdots, \hat{Y}_{iK}\}$. Then, it computes the distance from the observed outcome to this set of samples as

$$E_i = \min_{1 \le k \le K} \left\| Y_i - \hat{Y}_{ik} \right\|. \tag{2}$$

The scalar E_i is set as the nonconformity score. Intuitively, a small score indicates that the speculated outcomes $\hat{Y}_{ik}$ are close to the observed outcome Y_i, where $\hat{Y}_{ik}$ are from the approximate density $q(Y|X_i)$ and Y_i is from the true underlying density $p(Y|X_i)$. Therefore, the scale of E_i reflects the distance between the estimated density and the true density. We use the empirical quantile of the nonconformity scores to construct the predictive set. The α-th empirical quantile is defined as $Q_\alpha(z_{1:n}) = \inf_x\{(\sum_{i=1}^n \mathbb{1}[z_i \le x])/n \ge \alpha\}$ where $\alpha \in [0, 1]$ and $\mathbb{1}[\cdot]$ is the indicator function.

For a new data point with covariates X, we generate $\hat{\mathbf{Y}} = \{\hat{Y}_1, \cdots, \hat{Y}_K\}$ with $\hat{Y}_k \sim q(Y|X)$. Suppose that the desired nominal coverage is $1 - \alpha$. Then, each sample $\hat{Y}_k$ is expanded to a region $R_k = \{y : \left\| y - \hat{Y}_k \right\| \le r\}$ with an arbitrary norm and a radius r as the $(1 - \alpha)$ quantile of the scores $\{E_1, \cdots, E_n\} \cup \{\infty\}$. We call R_k an element region of the data point X. The proposed predictive set is the union of the element regions,

$$\hat{C}(X, \hat{\mathbf{Y}}) = \cup_{k=1}^K R_k = \cup_{k=1}^K \{y : \left\| y - \hat{Y}_k \right\| \le Q_{1-\alpha}(E_{1:n} \cup \{\infty\})\}. \tag{3}$$

As a special case, when the outcome is a scalar, the predictive set can be written explicitly as

$$\hat{C}(X, \hat{\mathbf{Y}}) = \cup_{k=1}^K \Big[\hat{Y}_k - Q_{1-\alpha}(E_{1:n} \cup \{\infty\}),\ \hat{Y}_k + Q_{1-\alpha}(E_{1:n} \cup \{\infty\})\Big]. \tag{4}$$

The proposed PCP algorithm is summarized in Algorithm 1.

As shown in Equation (3), the predictive set can be either continuous or discontinuous. Therefore, it can produce a sharp estimate by automatically adapting to the distributional properties of the target distribution. When the generative model is less well fitted, PCP maintains a valid marginal coverage, properly quantifying the predictive uncertainty. When the generative model fits the target distribution well, the predictive set allocates its volume according to the random samples. For example, if $p(Y|X)$ is multimodal and the multimodality is captured by the estimated $q(Y|X)$, the predictive set would consist of discontinuous sets around the modes where each set is relatively small.

Though in some situations, a continuous interval prediction is preferred in terms of interpretability [33], when the target is multimodal, a discontinuous set might be more interpretable. For example, when predicting a watch price based on its appearance without knowing the brand, a price range

Algorithm 1 Probabilistic Conformal Prediction

Input: Data $\mathbf{D} = \{(X_i, Y_i)\}_{i=1}^N$, model $q(Y|X)$, nominal level α, test point X, sample size K.

Step I: Conditional generative model

1: Split the data into three folds $\mathcal{Z}_{\text{tr}}$, $\mathcal{Z}_{\text{val}}$, $\mathcal{Z}_{\text{cal}}$ with set of index as $\mathbf{I}_{\text{tr}}$, $\mathbf{I}_{\text{val}}$, $\mathbf{I}_{\text{cal}}$ respectively
2: Fit $q(Y|X)$ on $\mathcal{Z}_{\text{tr}}$ with hyper-parameter chosen by cross validation on $\mathcal{Z}_{\text{val}}$

Step II: Predictive set for a test point

1: For $i \in \mathbf{I}_{\text{cal}}$, sample $\hat{Y}_{i1}, \cdots, \hat{Y}_{iK} \sim q(Y|X_i)$
2: For test point X, sample $\hat{Y}_1, \cdots, \hat{Y}_K \sim q(Y|X)$
3: Compute nonconformity score $\{E_i\}_{i \in \mathbf{I}_{\text{cal}}}$ by Equation (2), $E_{N+1} = \infty$, $\tilde{\mathbf{I}}_{\text{cal}} = \mathbf{I}_{\text{cal}} \cup \{N+1\}$
4: Set r as the $(1-\alpha)$ empirical quantile of $\{E_i\}_{i \in \tilde{\mathbf{I}}_{\text{cal}}}$
5: Compute the predictive set $\hat{C}(X, \hat{\mathbf{Y}})$ by Equation (3)

Output: Predictive set $\hat{C}(X, \hat{\mathbf{Y}})$

($\$100, \$200) \cup (\$1000, \$1200)$ might be more informative than $(\$100, \$1200)$. Nevertheless, one can take the convex hull of a discontinuous set to form a continuous interval but not vice versa.

By the construction of the predictive set defined in Equation (3), the estimated density $q(Y|X)$ can be explicit or implicit. This flexibility makes PCP compatible with a wide range of density estimators and generative models. Moreover, the predictive set in Equation (3) can be computed without approximation, making PCP scalable to a high dimensional target variable Y. Finally, PCP has a guaranteed marginal coverage as shown in Theorem 1.

Theorem 1. *Suppose (X_i, Y_i), $i \in \{1, 2, \cdots, n\}$ and (X, Y) are exchangeable, then*

(1) the predictive set in Equation (3) satisfies

$$\mathbb{P}_{X,Y,\hat{\mathbf{Y}}}(Y \in \hat{C}(X, \hat{\mathbf{Y}})) \geq 1 - \alpha; \tag{5}$$

(2) when the scores $E_1, \cdots, E_n$ are distinct almost surely,

$$\mathbb{P}_{X,Y,\hat{\mathbf{Y}}}(Y \in \hat{C}(X, \hat{\mathbf{Y}})) \leq 1 - \alpha + \frac{1}{n+1}. \tag{6}$$

Theorem 1 demonstrates that the marginal coverage of PCP is tight. In particular, the condition of the upper bound is satisfied when $p(Y|X)$ is continuous with respect to the Lebesgue measure.

In practice, we take the quantile of $E_{1:n}$ instead of the inflated scores $E_{1:n} \cup \{\infty\}$ in Equation (3). The following corollary offers the coverage guarantee under such modification.

Corollary 1. *Under the conditions of Theorem 1 and suppose $\alpha \geq 1/(n+1)$, if the quantile in Equation (3) is $Q_{1-\alpha}(E_{1:n})$, then $\mathbb{P}(Y \in \hat{C}(X, \hat{\mathbf{Y}})) \in [1-\alpha-1/(n+1), 1-\alpha+1/(n+1)]$; if the quantile in Equation (3) is $Q_{(1-\alpha)(1+\frac{1}{n})}(E_{1:n})$, then $\mathbb{P}(Y \in \hat{C}(X, \hat{\mathbf{Y}})) \in [1-\alpha, 1-\alpha+1/(n+1)]$*

By Corollary 1, if we take the $1-\alpha$ quantile of $E_{1:n}$, we may lose the coverage probability by $1/(n+1)$ where n is the number of data in the calibration set. Instead, we can take the $(1-\alpha)(1+\frac{1}{n})$ quantile of $E_{1:n}$ to maintain $(1-\alpha)$ coverage when $\alpha \geq 1/(n+1)$.

High Density Probabilistic Conformal Prediction (HD-PCP). Ideally, we may want the predictive sets to contain only high density regions to offer informative predictions. As shown in Appendix B, for different sets covering a specific probability of a multimodal distribution with the same marginal coverage, the high density region has the smallest size.

In PCP, the generated random samples include low density samples. When K increases, the set size of the low density region will decrease. However, PCP may generate many isolated sets and make interpretation difficult for practitioners. To mitigate this problem, we propose High Density Probabilistic Conformal

Prediction (HD-PCP) to filter out β fraction low-density samples to identify the high density regions when $q(Y|X)$ is explicit. Instead of sampling K samples from $q(Y|X)$ like in PCP, we sample $\lceil K/(1-\beta) \rceil$ samples for each X, keep $1-\beta$ fraction of samples with the highest estimated density, and keep all other parts of the algorithm the same as PCP. The HD-PCP algorithm is summarized in Appendix Algorithm 2. The marginal coverage guarantee still holds for HD-PCP, as shown in Corollary 2.

Corollary 2. *Under the conditions of Theorem 1, HD-PCP has the same marginal coverage as PCP.*

The proofs of the theorems and corollaries are presented in Appendix A.

3 Experiments

In this section, we conduct a comprehensive analysis demonstrating the advantages of PCP compared to previously proposed conformal inference methods. We aim to answer the following *questions*: **(a)** how does PCP perform in terms of coverage and predictive set size when compared with baseline models on synthetic datasets, that has multimodal $p(Y|X)$ distributions? **(b)** Does the filtering technique improve the predictive set of PCP? **(c)** How well do PCP and HD-PCP perform on real datasets with a single target? **(d)** How do the backbone models impact the performance of PCP? **(e)** Does PCP provide better predictive sets in tasks with multi-dimensional targets? The code for the simulations are available at `https://github.com/Zhendong-Wang/Probabilistic-Conformal-Prediction`.

Our experiments are structured into three sections. We first conduct experiment on classic 2D synthetic data to answer question **(a)** and **(b)**. Then, we compare PCP and HD-PCP with a full set of baseline methods on several selected real datasets to address question **(b)**, **(c)** and **(d)**. Finally, we conduct experiments on multi-dimensional regression tasks to address question **(e)**. We run all our experiements on machines with AMD EPYC 7763 CPUs.

Baselines. we consider CHR [33], DistSplit [16], CDSplit [16], DCP [8], and CQR [31] as our comparison baselines. For CHR, we use two different conditional density estimation models based on neural network model and random forest model, and we denote them as CHR-NN and CHR-QRF. We evaluate all the mentioned baselines based on their public Github implementation except for CDSplit. We implement python-based CDSplit based on their official R implementation to use the same backbone generative model for a fair comparison, denoted as CDSplit-KMN and CDSplit-MixD.

Choosing the hyperparameter K. We conduct an ablation study on the effect of the sample size K of PCP. As shown in Figure 3, empirically we find when K increases, the average size of the predictive sets reduces fast first and then gets slow. In practice, we set K moderately large to balance the sharpness and the computational cost, i.e., $K = 40$ or $K = 1000$ (two-dimensional targets).

3.1 Synthetic data experiments

To evaluate the effectiveness of proposed methods, we compare the predictive set of PCP and HD-PCP with other baseline methods on classic 2D synthetic data: s-curve, half-moons, 25-Gaussians, 8-Gaussians, circle and swiss-roll. We show the evaluation results of the s-curve and the 25-Gaussians in Figure 4 and place detailed results in Appendix E.

Figure 4 illustrates, for dataset that has multimodal $p(Y|X)$ distribution, models that consider multi-modality, such as CDSplit and (HD-)PCP, works apparently better than the models that can only provide unimodal predictions. This is consistent with our discussion in Section 1. Quantitatively, all models achieve the target marginal coverage $(1-\alpha)$, while the average set sizes of CDSplit and (HD-)PCP are several times smaller than that from CHR. We demonstrate that CDSplit and PCP both can provide sharp and informative predictive sets for these multimodal datasets and PCP is slightly better with respect to the set size. The right two panels show the effect of filtering high density samples. The predictive sets from HD-PCP become cleaner and concentrated on the correct modes. Correspondingly, the average set sizes of HD-PCP drop a lot compared to PCP. The histogram moving from blue bins to orange bins also demonstrates the effectiveness of the filtering.

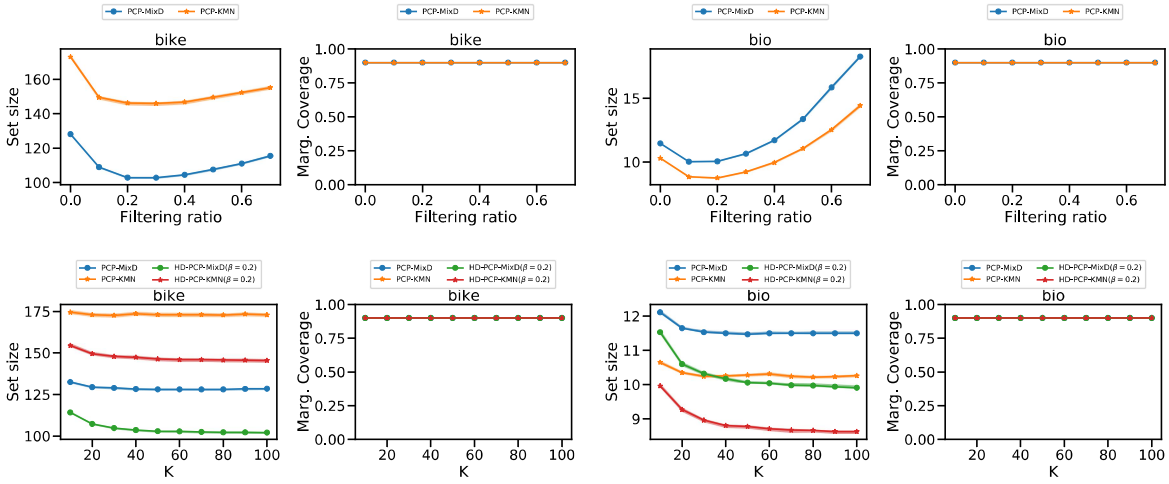


Figure 3: Ablation study on choosing hyperparameter K and filtering ratio β. We run experiments of PCP and HD-PCP with MixD and KMN as backbone models on two datasets, bike and bio. The K grid is $[10, 20, 30, \ldots, 100]$ and the β grid is $[0.1, 0.2, \ldots, 0.7]$. In the first row, we show predictive set sizes with $K = 50$ and varying β. Second row shows how predictive set size varies with K.

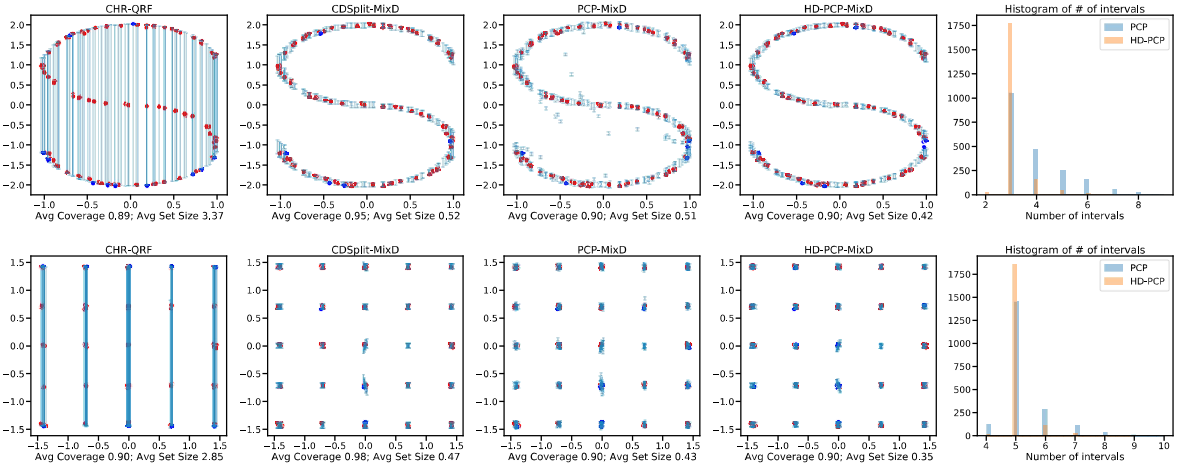


Figure 4: Visualization of predictive sets ($\alpha = 0.1$) on 2D toy datasets: s-curve and 25-Gaussians. We shows the predictive sets on 100 test data samples. Blues lines: the predictive sets from each method; Blue dots: test points that are not covered by the predictive sets; Reds dots: test points covered. We report the marginal coverage and the average set size across test datapoints in the x-axis label. The fifth column shows the histogram of the number of predicted intervals of PCP and HD-PCP. We set $K = 40$ for (HD-)PCP, $\beta = 0.2$. Detailed experiments are in Appendix E.

3.2 Real data experiments

We study regression tasks on several real data sets to evaluate the effectiveness of PCP and HD-PCP [33]. We consider multiple types of generative models $q(Y \mid X)$, including implicit models (GAN [14]) and semi-implicit model [40, 42, 43], explicit models (KMN [2], MixD [4]), and QRF [24, 33]. We denote them as PCP-GAN, PCP-SIVI, PCP-KMN, PCP-MixD and PCP-QRF respectively.

Datasets. We conduct real data experiments on 9 public-domain data sets: bike sharing data (bike), physicochemical properties of protein tertiary structure (bio), blog feedback (blog), and Facebook comment volume, variants one (fb1) and two (fb2), medical expenditure panel survey number 19

(meps19), number 20 (meps20), and number 21 (meps21) [31] and temperature forecast data [9]. Table 3 in Appendix F illustrates the dataset sizes and data splits for training, calibration and testing.

Evaluation Protocol. We evaluate the marginal coverage, conditional coverage (approximated by the worst-slab conditional coverage [6, 32]), and the predictive set size.We report results based on 50 random splits for all datasets.

Table 1 shows numerical results. For our methods, we report PCP-MixD and HD-PCP-MixD; for baselines, we report the backbone model that works generally the best across the 9 datasets with respect to set size in the main paper. See detailed results in Appendix F: Table 4 reports the best results among the variants of each method; Table 5, Table 6 and Table 7 report full experiment results.

We observe that all conformal methods achieve $(1-\alpha)$ marginal coverage and perform well in terms of the worst-slab conditional coverage. Thus, our comparison focuses on the size of predictive sets. As shown in Table 1, HD-PCP-MixD outperforms all the other baselines on 7 out of 9 datasets in terms of the predictive set size. When picking a backbone model for each dataset is allowed, our models outperform baselines on all datasets, as shown in Table 4. Comparing HD-PCP with PCP, we find that the filtering technique leads to consistent performance improvement. Table 4 shows that PCP outperforms the baselines by a large margin especially on blog, facebook1 and facebook2 datasets.

Moreover, note that PCP is flexible for backbone generative models, as long as the model can easily generate random conditional samples. The flexibility of PCP makes it achieve good performance by choosing a proper generative model according to data sets. For example, PCP-SIVI works well in bike and facebook data with implicit backbone conditional generative models, as shown in Table 5.

The limitation of CDSplit may be explained by noting that the method needs to make partitions of data based on K-means algorithm, which is known to be unstable due to local minima. It also needs to approximate the level set on a grid of the target space to form the predictive set. It may be sensitive to the range and coarseness of the grid. Thus, we notice that CDSplit produces large predictive sets on facebook data and meps data. Similar to Sesia and Romano [33], we observe that DCP is sensitive to the estimation of the distribution tails [33], which makes it unstable for some datasets. CHR is more robust because it only needs to estimate a histogram with relatively few bins [33], while it could only provide a single continuous interval and produces a loose predictive set when the data exhibits mulimodality. CQR predicts intervals based on learned lower and upper quantiles, which leads to large intervals when the learned quantiles are not accurate or the data distribution is multimodal.

3.3 Multi-dimensional targets

Beyond single target regression tasks, we study PCP and HD-PCP on multi-target datasets. We use the same strategy in Neeven and Smirnov [29] to adapt previous baselines to multi-target conformal algorithms by fitting each dimension separately with coverage level $(1-\alpha)/d$, where d is the dimension of target Y (this ensures the coverage of the target vector is $1-\alpha$ [23]).

We construct a synthetic dataset to illustrate the benefit of PCP when targets are dependent. Covariates X are sampled from $\mathcal{N}(0,1)^5$ and the target Y is randomly sampled from a bi-modal Gaussian distribution $0.5\mathcal{N}(\mu_1,\Sigma)+0.5\mathcal{N}(\mu_2,\Sigma)$. $\mu_i=\beta_i^T(x,1)$ and β_i is randomly generated from $\mathcal{N}(0,1)^6$, $\Sigma=\begin{pmatrix}10 & \rho\\ \rho & 10\end{pmatrix}$. The synthetic data distribution in Figure 5a shows that the distribution concentrates as ρ increases. As shown in Figure 5b and Figure 5c, PCP achieves the best performance in terms of average predictive set size. We observe when ρ gets higher, only PCP shrinks the predictive set accordingly while the predictions from other methods have little change and become loose. The detailed quantitative results are in Table 8, Appendix E.

We further study conformal methods on two multi-target real datasets. Taxi Data are the taxi trip records of New York City which include the pickup, drop-off locations of each trip and the corresponding time. We sample 10000 records from January 2016 (7800, 2000, 200 for train, calibration and test respectively), and use the pickup time and location as covariates X to predict drop-off locations Y. The locations are represented by latitudes and longitudes.For energy dataset, we predict the heating load and cooling load

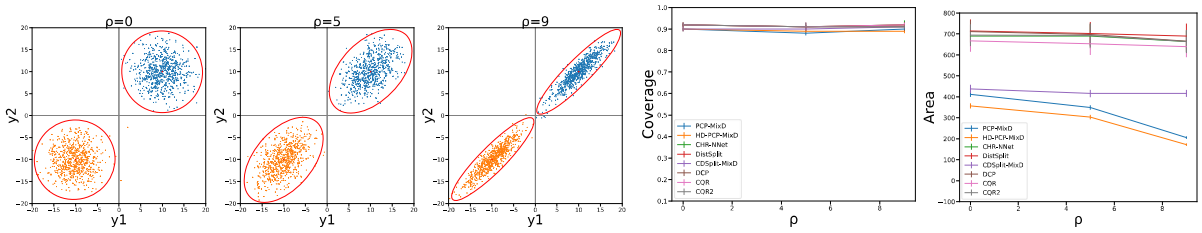
Data	Metric	PCP	HD-PCP	CHR [33]	DistSplit [16]	CDSplit [16]	DCP [8]	CQR [31]
bike	Marg. C	0.90	0.90	0.90	0.90	0.92	0.90	0.90
	Cond. C	0.86	0.88	0.88	0.87	0.91	0.88	0.89
	Set Size	128.13(0.53)	**102.92(0.48)**	204.10(1.03)	423.13(1.51)	115.74(0.50)	443.76(1.36)	403.88(0.86)
bio	Marg. C	0.90	0.90	0.90	0.90	0.90	0.90	0.90
	Cond. C	0.89	0.90	0.89	0.89	0.90	0.89	0.89
	Set Size	11.47(0.04)	10.06(0.05)	10.21(0.04)	13.19(0.04)	**9.58(0.04)**	12.95(0.04)	13.00(0.02)
blogdata	Marg. C	0.89	0.90	0.90	0.90	0.96	0.90	0.90
	Cond. C	0.85	0.87	0.87	0.87	0.95	0.88	0.87
	Set Size	10.78(0.17)	**9.44(0.19)**	10.81(0.17)	16.27(0.23)	39.00(0.40)	1422.36(0.03)	15.15(0.26)
facebook1	Marg. C	0.90	0.90	0.90	0.90	0.95	0.90	0.90
	Cond. C	0.82	0.85	0.86	0.89	0.95	0.89	0.88
	Set Size	9.99(0.14)	**8.93(0.12)**	11.21(0.12)	14.03(0.16)	33.69(0.16)	1303.01(0.04)	13.79(0.15)
facebook2	Marg. C	0.90	0.90	0.90	0.90	0.97	0.90	0.90
	Cond. C	0.82	0.84	0.87	0.89	0.96	0.89	0.89
	Set Size	9.93(0.11)	**8.84(0.10)**	10.81(0.14)	13.48(0.19)	45.75(0.16)	1963.68(0.03)	13.00(0.17)
meps19	Marg. C	0.90	0.90	0.90	0.90	0.93	0.90	0.90
	Cond. C	0.87	0.88	0.89	0.89	0.92	0.88	0.89
	Set Size	19.28(0.16)	**17.78(0.18)**	18.26(0.15)	29.96(0.28)	23.86(0.17)	559.23(0.01)	28.71(0.18)
meps20	Marg. C	0.90	0.90	0.90	0.90	0.92	0.90	0.90
	Cond. C	0.87	0.88	0.90	0.90	0.92	0.88	0.90
	Set Size	19.52(0.16)	18.19(0.17)	**17.94(0.18)**	29.35(0.23)	22.93(0.16)	520.25(0.01)	27.57(0.15)
meps21	Marg. C	0.90	0.90	0.90	0.90	0.93	0.90	0.90
	Cond. C	0.87	0.88	0.90	0.89	0.92	0.88	0.89
	Set Size	19.18(0.12)	**17.91(0.15)**	18.65(0.16)	30.32(0.31)	23.63(0.17)	531.25(0.01)	29.89(0.20)
temperature	Marg. C	0.90	0.90	0.90	0.90	0.92	0.90	0.90
	Cond. C	0.90	0.89	0.89	0.89	0.91	0.88	0.87
	Set Size	2.10(0.01)	**1.85(0.01)**	3.24(0.01)	3.07(0.01)	2.23(0.01)	3.10(0.02)	3.55(0.03)

Table 1: Summary results of real data experiments, where Marg. C and Cond. C denote the marginal coverage and approximated conditional coverage. The results are averaged over 50 random cross-validation splits. We report the set size mean and standard error (inside the parentheses, the default is 0.00) based on the same 50 splits. The nominal coverage rate $(1-\alpha)$ is 90%, the K for (HD-)PCP is set as 40. To save space and keep consistency, here we report PCP-MixD, HD-PCP-MixD, CHR-QRF, CDSplit-MixD and CQR, which include the variant that works generally the best across the 9 datasets. The detailed results are placed in Appendix F.

for energy efficiency analysis [38]. The predictors are building information such as orientation, glazing area and wall area. We use 568, 100, and 100 samples for train, calibration and test set. We report the predictive set sizes of each algorithm. To calculate the predictive set size of PCP, due to the overlapped regions, the set size cannot be calculated exactly. We estimate it by Monte Carlo simulation with a grid size of 100 on each dimension. For (HD-)PCP and CDSplit, we use MixD as the backbone model.

As shown in Table 2, algorithms considering multimodality have significantly better performance compared to other baselines. We visualize the conformal region predicted by (HD-)PCP in Figure 1 (See Appendix G for more qualitative results). We notice that PCP can successfully capture the most popular regions of New York City for drop-off, downtown of Manhattan, LaGuardia airport and JFK airport, while methods with continuous predictive sets would learn a wide bounding box. Furthermore, as shown in Table 2, PCP has similar coverage level but with much smaller predictive set compared to CDSplit. This might be because the joint estimation of the high-dimensional targets can capture dependencies between the target elements. Applying the filtering, HD-PCP provides a cleaner and interpretable predictive set and further reduces the set size.

We include more experiments with two other multi-target real datasets in Appendix H(8 and 3 targets) due to space constraint. Since Monte-Carlo estimation of volume of high-dimensional nonconvex set suffers from curse of dimension, we focus on two dimensional task and evaluate the predictive set generated for every pair of all targets. PCP and HD-PCP offer significantly sharper predictive sets compared to baselines on almost all pairs, and HD-PCP can further improve the performance of PCP.



(a) $p(Y|X)$ with varying ρ (b) Coverage (c) Set Size

Figure 5: (a): Conditional data distribution $p(Y|X)$ for multi-target synthetic dataset. (b), (c): Marginal coverage and set size for baselines. PCP is able to capture the joint coverage more efficiently than other baselines when ρ increases.

Data	Metric	PCP	HD-PCP	CHR [33]	DistSplit [16]	CDSplit [16]	DCP [8]	CQR [31]
Taxi	Marg. C	0.90(0.01)	0.89(0.01)	0.87(0.01)	0.91(0.01)	0.91(0.01)	0.89(0.01)	0.90(0.01)
	Cond. C	0.89(0.03)	0.87(0.02)	0.84(0.04)	0.92(0.03)	0.89(0.03)	0.84(0.03)	0.86(0.04)
	Set Size	0.0089(0.0001)	**0.0064(0.0002)**	0.0245(0.0009)	0.0202(0.0009)	0.0097(0.0023)	0.0354(0.0013)	1.9302(0.5202)
Energy	Marg. C	0.89(0.01)	0.90(0.01)	0.93(0.01)	0.92(0.01)	0.93(0.01)	0.93(0.01)	0.91(0.01)
	Cond. C	0.87(0.04)	0.94(0.03)	0.93(0.05)	0.83(0.10)	0.89(0.06)	0.97(0.02)	0.94(0.03)
	Set Size	19.22(2.55)	**14.13(3.55)**	27.41(2.65)	36.31(3.04)	36.88(3.94)	34.18(3.61)	45.20(3.19)

Table 2: Summary results of Multi-Target Regression experiments, where Marg. C and Cond. C denote the marginal coverage and approximated conditional coverage. Results are averaged over 10 random cross-validation splits. We report the standard error inside the parentheses. The nominal coverage rate $(1-\alpha)$ is 90%, the K for (HD-)PCP is set as 1000.

4 Conclusion and Future Work

We proposed novel conformal inference algorithms, PCP and HD-PCP, that find valid and sharp predictive sets using random samples from a conditional generative model; PCP and HD-PCP can be built from either implicit or explicit models. PCP and HD-PCP outperform existing methods, particularly with multimodal data and multi-dimensional targets.

Like most existing conformal methods, PCP and HD-PCP rely on a reasonable estimation through a fitted conditional generative model. When the generative model is inaccurate, PCP may only provide wide, uninformative predictive sets to ensure the validity. However, given the recent success in conditional generative models, PCP and HD-PCP may be promising in many domains, including precision medicine, quantitative finance and marketing [17, 19, 41].

Here we focused on regression tasks. Future research can consider adapting PCP and HD-PCP to conformal treatment effect estimation and classification problems [11, 32, 44].

References

[1] Justin Alsing, Tom Charnock, Stephen Feeney, and Benjamin Wandelt. Fast likelihood-free cosmology with neural density estimators and active learning. *Monthly Notices of the Royal Astronomical Society*, 488(3):4440–4458, 2019.

[2] Luca Ambrogioni, Umut Güçlü, Marcel AJ van Gerven, and Eric Maris. The kernel mixture network: A nonparametric method for conditional density estimation of continuous random variables. *arXiv preprint arXiv:1705.07111*, 2017.

[3] Rina Foygel Barber, Emmanuel J Candes, Aaditya Ramdas, and Ryan J Tibshirani. Predictive inference with the jackknife+. *arXiv*, May 2019.

[4] Christopher M. Bishop. Mixture density networks. Technical report, Aston University, 1994.

[5] Leo Breiman and Jerome H Friedman. Predicting multivariate responses in multiple linear regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 59(1):3–54, 1997.

[6] Maxime Cauchois, Suyash Gupta, and John C. Duchi. Knowing what you know: valid and validated confidence sets in multiclass and multilabel prediction. *J. Mach. Learn. Res.*, 22:81:1–81:42, 2021.

[7] Jeffrey Chan, Valerio Perrone, Jeffrey Spence, Paul Jenkins, Sara Mathieson, and Yun Song. A likelihood-free inference framework for population genetic data using exchangeable neural networks. *Advances in neural information processing systems*, 31, 2018.

[8] Victor Chernozhukov, Kaspar Wüthrich, and Yinchu Zhu. Distributional conformal prediction. *Proceedings of the National Academy of Sciences*, 118, 2021.

[9] Dongjin Cho, Cheolhee Yoo, Jungho Im, and Dong-Hyun Cha. Comparative assessment of various machine learning-based bias correction methods for numerical weather prediction model forecasts of extreme air temperatures in urban areas. *Earth and Space Science*, 7, 2020.

[10] Jacob Dorn and Kevin Guo. Sharp sensitivity analysis for inverse propensity weighting via quantile balancing. *Journal of the American Statistical Association*, pages 1–28, 2022.

[11] Ruijiang Gao, Max Biggs, Wei Sun, and Ligong Han. Enhancing counterfactual classification via self-training. *arXiv preprint arXiv:2112.04461*, 2021.

[12] Seymour Geisser. *Predictive Inference*, volume 55. CRC Press, 1993.

[13] Christina Gillmann, Lucas Peter, Carlo Schmidt, Dorothee Saur, and Gerik Scheuermann. Visualizing multimodal deep learning for lesion prediction. *IEEE Computer Graphics and Applications*, 41(5): 90–98, 2021.

[14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[15] Peter Hoff. Bayes-optimal prediction with frequentist coverage control. *arXiv*, 2021.

[16] Rafael Izbicki, Gilson Shimizu, and Rafael Stern. Flexible distribution-free conditional predictive bands using density estimators. In *International Conference on Artificial Intelligence and Statistics*, pages 3068–3077. PMLR, 2020.

[17] Adriano Koshiyama, Nick Firoozye, and Philip Treleaven. Generative adversarial networks for financial trading strategies fine-tuning and combination. *Quantitative Finance*, 21(5):797–813, 2021.

[18] Arun Kumar Kuchibhotla. Exchangeability, conformal prediction, and rank tests. *arXiv*, May 2020.

[19] Wonsung Lee, Sungrae Park, Weonyoung Joo, and Il-Chul Moon. Diagnosis prediction via medical context attention networks using deep generative modeling. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 1104–1109. IEEE, 2018.

[20] Jing Lei and Larry Wasserman. Distribution-free prediction bands for non-parametric regression. *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, pages 71–96, 2014.

[21] Jing Lei, Alessandro Rinaldo, and Larry Wasserman. A conformal prediction approach to explore functional data. *Annals of Mathematics and Artificial Intelligence*, 74(1):29–43, 2015.

[22] Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523): 1094–1111, 2018.

[23] Lihua Lei and Emmanuel J Candès. Conformal inference of counterfactuals and individual treatment effects. *Journal of the Royal Statistical Society: Series B*, 2021.

[24] Nicolai Meinshausen. Quantile regression forests. *J. Mach. Learn. Res.*, 7:983–999, 2006.

[25] Soundouss Messoudi, Sébastien Destercke, and Sylvain Rousseau. Conformal multi-target regression using neural networks. In *Conformal and Probabilistic Prediction and Applications*, pages 65–83. PMLR, 2020.

[26] Soundouss Messoudi, Sébastien Destercke, and Sylvain Rousseau. Copula-based conformal prediction for multi-target regression. *Pattern Recognition*, 120:108101, 2021.

[27] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[28] Vivek Myers, Erdem Biyik, Nima Anari, and Dorsa Sadigh. Learning multimodal rewards from rankings. In *Conference on Robot Learning*, pages 342–352. PMLR, 2022.

[29] Jelmer Neeven and Evgueni Smirnov. Conformal stacked weather forecasting. In *Conformal and Probabilistic Prediction and Applications*, pages 220–233. PMLR, 2018.

[30] Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. In *European Conference on Machine Learning*. Springer, 2002.

[31] Yaniv Romano, Evan Patterson, and Emmanuel Candes. Conformalized quantile regression. *Advances in neural information processing systems*, 32, 2019.

[32] Yaniv Romano, Matteo Sesia, and Emmanuel J. Candès. Classification with valid and adaptive coverage. *arXiv: Methodology*, 2020.

[33] Matteo Sesia and Yaniv Romano. Conformal prediction using conditional histograms. *Advances in Neural Information Processing Systems*, 34, 2021.

[34] Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(3), 2008.

[35] Eleftherios Spyromitros-Xioufis, Grigorios Tsoumakas, William Groves, and Ioannis Vlahavas. Multi-target regression via input space expansion: treating targets as inputs. *Machine Learning*, 104(1): 55–98, 2016.

[36] Ryan J Tibshirani, Rina Foygel Barber, Emmanuel Candes, and Aaditya Ramdas. Conformal prediction under covariate shift. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[37] Brian L Trippe and Richard E Turner. Conditional density estimation with bayesian normalising flows. *arXiv preprint arXiv:1802.04908*, 2018.

[38] Athanasios Tsanas and Angeliki Xifara. Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy and buildings*, 49, 2012.

[39] Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer Science & Business Media, 2005.

[40] Zhendong Wang and Mingyuan Zhou. Thompson sampling via local uncertainty. In *International Conference on Machine Learning*, pages 10115–10125. PMLR, 2020.

[41] Lizhen Xu, Jason A Duan, and Andrew Whinston. Path to purchase: A mutually exciting point process model for online advertising and conversion. *Management Science*, 2014.

[42] Mingzhang Yin and Mingyuan Zhou. Semi-implicit variational inference. In *International Conference on Machine Learning*, pages 5660–5669. PMLR, 2018.

[43] Mingzhang Yin and Mingyuan Zhou. Semi-implicit generative model. *Proceedings of the NeurIPS Workshop on Bayesian Deep Learning*, 2018.

[44] Mingzhang Yin, Claudia Shi, Yixin Wang, and David M. Blei. Conformal sensitivity analysis for individual treatment effects. *arXiv*, 2021.

Appendix

A Proof

For the completeness, we first present a lemma adapted from Tibshirani et al. [36], Romano et al. [31].

Lemma 1. *Suppose $Z_1, \cdots, Z_n, Z_{n+1}$ are scalar random variables that are exchangeable and almost surely distinct, then for $\beta \in [0, 1]$*

$$\beta \leq \mathbb{P}(Z_{n+1} \leq Q_\beta(Z_{1:n} \cup \{\infty\})) \leq \beta + \frac{1}{n+1}.$$

Proof of Lemma 1. Denote the inflated $Z_{1:n}$ as

$$\tilde{Z}_{1:n} = Z_{1:n} \cup \{\infty\}. \tag{7}$$

The quantile

$$Q_\beta(Z_{1:n}) := \inf_x (\sum_{i=1}^{n} \mathbb{1}[Z_i \leq x])/n = Z_{(\lceil n\beta \rceil)}.$$

By Lemma 1 of Tibshirani et al. [36], the events

$$Z_{n+1} \leq Q_\beta(\tilde{Z}_{1:n}) \Leftrightarrow Z_{n+1} \leq Q_\beta(Z_{1:n+1}). \tag{8}$$

Furthermore, by exchangeability and the definition of empirical quantile,

$$\beta \leq \mathbb{P}(Z_{n+1} \leq Q_\beta(Z_{1:n+1})) \leq \beta + \frac{1}{n+1}, \tag{9}$$

where the upper bound hold when $Z_{1:n+1}$ are almost surely distinct. By Equation (8), the proof is completed. □

Proof of Theorem 1. Given the estimated conditional density $q(Y|X)$ on an independently sampled training set D_{tr}. By assumption, the calibration set $\{(X_i, Y_i)\}_{i=1}^n$ and the test point (X_{n+1}, Y_{n+1}) are exchangeable. Denote $D_i = (X_i, Y_i, \hat{\mathbf{Y}}_i)$ for $i = 1, \cdots, n, n+1$. Then $D_i \sim p(X, Y)q^K(Y|X_i)$ and $\{D_{1:n+1}\}$ are exchangeable because $\hat{\mathbf{Y}}_i \perp\!\!\!\perp \hat{\mathbf{Y}}_j$.

The nonconformity score E_i in Equation (2) is defined as a deterministic function of D_i. Therefore $\{E_i\}_{i=1}^{n+1}$ are exchangeable [18] and are almost surely distinct. By Lemma 1,

$$1 - \alpha \leq \mathbb{P}(E_{n+1} \leq Q_\alpha(E_{1:n} \cup \{\infty\}) \,|\, D_{\text{tr}}) \leq 1 - \alpha + \frac{1}{n+1}. \tag{10}$$

Next we demonstrate that for

$$\hat{C}(X_{n+1}, \hat{\mathbf{Y}}_{n+1}) = \cup_{k=1}^{K} \{y : \left\| y - \hat{Y}_{n+1,k} \right\| \leq \hat{Q}\},\ \hat{Q} = Q_{1-\alpha}(E_{1:n} \cup \{\infty\}),$$

the following statement holds

$$Y_{n+1} \in \hat{C}(X_{n+1}, \hat{\mathbf{Y}}_{n+1}) \Leftrightarrow E_{n+1} \leq \hat{Q}.$$

Suppose the LHS is true, then $\exists\ m, 1 \leq m \leq K$, s.t. $Y_{n+1} \in \{y : \left\| y - \hat{Y}_{n+1,m} \right\| \leq \hat{Q}\}$. This means $\left\| Y_{n+1} - \hat{Y}_{n+1,m} \right\| \leq \hat{Q}$. Hence $E_{n+1} = \min_k \left\| Y_{n+1} - \hat{Y}_{n+1,k} \right\| \leq \left\| Y_{n+1} - \hat{Y}_{n+1,m} \right\| \leq \hat{Q}$.

On the other hand, suppose the RHS is true, letting $t = \arg\min_k \left\| Y_{n+1} - \hat{Y}_{n+1,k} \right\|$, we have $\left\| Y_{n+1} - \hat{Y}_{n+1,t} \right\| \leq \hat{Q}$, i.e., $Y_{n+1} \in \left\{ y : \left\| y - \hat{Y}_{n+1,t} \right\| \leq \hat{Q} \right\}$. Therefore, $Y_{n+1} \in \hat{C}(X_{n+1}, \hat{\mathbf{Y}}_{n+1})$.

Then by Equation (10), we have

$$1 - \alpha \leq \mathbb{P}(Y_{n+1} \in \hat{C}(X_{n+1}, \hat{\mathbf{Y}}_{n+1}) \,|\, D_{\text{tr}}) \leq 1 - \alpha + \frac{1}{n+1}. \tag{11}$$

Marginalizing out D_{tr} the statement is proved. □

Proof of Corollary 1. Suppose $0 \leq \beta \leq n/(n+1)$, then $\lceil (n+1)\beta \rceil \neq n+1$. We have

$$Q_\beta(\tilde{Z}_{1:n}) = \tilde{Z}_{(\lceil (n+1)\beta \rceil, n+1)} = Z_{(\lceil n \frac{n+1}{n} \beta, n \rceil)} = Q_{(1+\frac{1}{n})\beta}(Z_{1:n}) \tag{12}$$

where $\tilde{Z}_{1:n}$ is defined in Equation (7). By Equations (8) and (9),

$$\beta \leq \mathbb{P}(Z_{n+1} \leq Q_{(1+\frac{1}{n})\beta}(Z_{1:n})) \leq \beta + \frac{1}{n+1}. \tag{13}$$

By Equation (13), we have

$$\beta - \frac{1}{n+1} \leq \mathbb{P}(Z_{n+1} \leq Q_\beta(Z_{1:n})) \leq \beta + \frac{1}{n+1}. \tag{14}$$

□

Proof of Corollary 2. With the notations in the proof of Theorem 1, $D_i = (X_i, Y_i, \hat{\mathbf{Y}}_i)$ are i.i.d. vararibables. For a fixed conditional density function $q(y|x)$, the nonconformity score E_i is fully determined by $X_i, Y_i, \hat{\mathbf{Y}}_i$. So $E_i = g(D_i)$ where $g(\cdot)$ is a deterministic function including the filtering step of HD-PCP. By Kuchibhotla [18], since $\{D_i\}_{i=1}^{n+1}$ are i.i.d., $\{E_i\}_{i=1}^{n+1}$ are exchangeable. The other parts of proof follow the same as the proof of Theorem 1. □

B High Density Probabilistic Conformal Prediction

Figure 6 shows the different predictive sets with 95% coverage when the underlying distribution is bi-mode normal.

We summarize our HD-PCP algorithm in Algorithm 2.

C Hyperparameters

PCP introduces one additional hyperparameter, the sample size K. We conduct ablation study on the effect of K in Figure 3 and find that as long as K is set moderately large, i.e., $K = 40$, the predictive set size is near optimal. K is not a very sensitive hyperparameter that needs much effort for tuning.

D Summary of Conditional generative models

SIVI Model. Following [40], we build a conditional distribution estimator by using semi-implicit variational inference [42] and we call it SIVI model. Specifically, we approximate $\mathbb{P}(Y \,|\, X)$ by an inference distribution $q_{\boldsymbol{\phi}}(Y \,|\, X)$ with respect to parameter $\boldsymbol{\phi}$. We construct the inference distribution as a hierarchical model,

$$y \sim q_{\boldsymbol{\phi}_1}(y|x, \boldsymbol{z}), \boldsymbol{z} \sim q_{\boldsymbol{\phi}_2}(\boldsymbol{z}|x, \boldsymbol{\psi}), \boldsymbol{\psi} \sim p(\boldsymbol{\psi})$$

Here $\boldsymbol{z} \in \mathbb{R}^d$ is an auxiliary latent variable and $p(\boldsymbol{\psi})$ is a known noise distribution, i.e., $\mathcal{N}(\mathbf{0}, \mathbf{I})$. The inference distribution is the marginal distribution of the hierarchical model, $q_{\boldsymbol{\phi}}(y \,|\, x) = \int q_{\boldsymbol{\phi}_1}(y|\boldsymbol{z}) q_{\boldsymbol{\phi}_2}(\boldsymbol{z}|x) d\boldsymbol{z}$

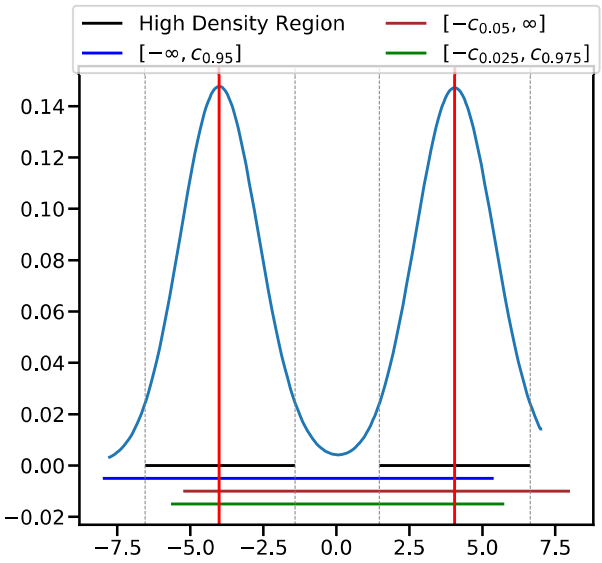


Figure 6: High Density Region can capture multi-mode easier comparing to other predictive set.

Algorithm 2 High Density Probabilistic Conformal Prediction

Input: Data $\mathbf{D} = \{(X_i, Y_i)\}_{i=1}^N$, nominal level α, test point X, generative model class $\mathcal{Q}$, sample size K, β grid B (For HD-PCP).

Step I: Conditional generative model

1: Split the data into three folds $\mathcal{Z}_{\text{tr}}$, $\mathcal{Z}_{\text{val}}$, $\mathcal{Z}_{\text{cal}}$ with set of index as $\mathbf{I}_{\text{tr}}$, $\mathbf{I}_{\text{val}}$, $\mathbf{I}_{\text{cal}}$ respectively
2: Estimate $q(Y|X)$ on $\mathcal{Z}_{\text{tr}}$ with hyper-parameter chosen by cross validation on $\mathcal{Z}_{\text{val}}$

Step II: Predictive set for a test point

1: For $i \in \mathbf{I}_{\text{cal}}$, sample $\hat{Y}_{i1}, \cdots, \hat{Y}_{iK} \sim q(Y|X_i)$
2: For $\beta \in B$, Filtering out β fraction of $\{\hat{Y}_{ik}\}_{k=1}^K$ with the lowest density. Repeat Line 3-7 for $x \in \mathbf{I}_{\text{cal}}$. $\beta_0 = \arg\min_\beta \lambda(\sum_{x \in \mathbf{I}_{\text{cal}}} \hat{C}_\beta(x, \hat{\mathbf{Y}}))$
3: For a test point, sample $\hat{Y}_1, \cdots, \hat{Y}_K \sim q(Y|X)$
4: Filtering out β fraction of $\{\hat{Y}_k\}_{k=1}^K$ with the lowest density
5: Compute nonconformity score $\{E_i\}_{i \in \mathbf{I}_{\text{cal}}}$ by Equation (2), $E_{N+1} = \infty$, $\tilde{\mathbf{I}}_{\text{cal}} = \mathbf{I}_{\text{cal}} \cup \{N+1\}$
6: Set r as the $(1-\alpha)$ empirical quantile of $\{E_i\}_{i \in \tilde{\mathbf{I}}_{\text{cal}}}$
7: Compute the predictive set $\hat{C}_\beta(X, \hat{\mathbf{Y}})$ by Equation (3)

Output: Predictive set $\hat{C}_{\beta_0}(X, \hat{\mathbf{Y}})$

with $\boldsymbol{\phi} = (\boldsymbol{\phi}_1, \boldsymbol{\phi}_2)$. We model $q_{\boldsymbol{\phi}_1}(y|\boldsymbol{z})$ and $q_{\boldsymbol{\phi}_2}(\boldsymbol{z}|x)$ as Gaussian distributions, whose mean and standard deviation are the outputs of neural networks feed with corresponding $(x, \boldsymbol{z})$ and $(x, \boldsymbol{\psi})$. The marginal distribution $q(y \mid x)$ can thus be constructed with flexibility in modeling multimodality, skewness and kurtosis. We learn the $\boldsymbol{\phi}$ by maximizing the ELBO [40],

$$\mathcal{L}_K = \mathbb{E}_{\boldsymbol{\psi}^{(0)}, \ldots, \boldsymbol{\psi}^{(K)} \overset{iid}{\sim} p(\boldsymbol{\psi})} \mathbb{E}_{\boldsymbol{z} \sim q(\cdot \mid \boldsymbol{\psi}^{(0)}, \boldsymbol{x})} \left[\ln q_{\boldsymbol{\phi}_1}(y \mid \boldsymbol{x}, \boldsymbol{z}) + \ln \frac{p(\boldsymbol{z})}{\frac{1}{K+1} \sum_{k=0}^K q_{\boldsymbol{\phi}_2}(\boldsymbol{z} \mid \boldsymbol{\psi}^{(k)}, \boldsymbol{x})} \right],$$

where $p(\boldsymbol{z})$ is a prior distribution for latent variable $\boldsymbol{z}$, i.e.,$\mathcal{N}(\mathbf{0}, \mathbf{I})$ and K is set as 20.

GAN model. To fit a conditional distribution, we follows [27] to build a Conditional GAN model with a generator $G(\boldsymbol{x}, \boldsymbol{z})$ and a discriminator $D(\boldsymbol{x}, y)$. For simplicity, we call it GAN model. Here, the $\boldsymbol{z}$ is a latent variable, which is usually set as $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $G(\boldsymbol{x}, \boldsymbol{z})$ is modeled by a neural network, whose outputs are the samples of y, and $D(\boldsymbol{x}, y)$ is another neural network, which outputs the probability

that the given y is from the true data distribution. We train G and D with the following adversarial loss,

$$\min_G \max_D V(G,D) = \mathbb{E}_{\boldsymbol{x},y\sim p_{data}(\boldsymbol{x},y)}[\log D(\boldsymbol{x},y)] + \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z}),\boldsymbol{x}\sim p_{data}(\boldsymbol{x})}[\log(1-D(\boldsymbol{x},G(\boldsymbol{x},\boldsymbol{z})))]$$

Kernel Mixture Network. Ambrogioni et al. [2] model arbitrarily complex conditional densities as linear combinations of a family of kernel functions centered at a subset of training points. The weights are determined by the outer layer of a deep neural network, trained by minimizing the negative log likelihood. The conditional density function is modeled as follows,

$$q(y|x) = \frac{1}{\sum_{p,j} w_{p,j}(x;W)} \sum_{p,j} w_{p,j}(x;W)\mathcal{K}_j(y,y^{(p)}),$$

where p denotes the index of the observed data points, $\mathcal{K}_j$ is the pre-set kernel function, j is the index of selected bandwidth for $\mathcal{K}_j$, and $w_{p,j}(x;W)$ represents the weight of each kernel. A common choice for $\mathcal{K}_j$ is the Gaussian kernel,

$$\mathcal{K}_j(y,y';\sigma_j) = \frac{1}{\sqrt{2\pi}\sigma_j} \exp^{-\frac{(y-y')^2}{2\sigma_j^2}}.$$

The weights $w_{p,j}(x;W)$ are determined by a deep neural network (DNN), with covariates x as the inputs and W as the parameters. All weights are non negative by applying non-negative activation functions on the output layer of DNN. We train the KMN model by minimizing the loss function,

$$\mathcal{L}(W) = -\sum_q \left[\log \sum_{p,j} w_{p,j}(x;W)\mathcal{K}_j(y,y^{(p)}) - \log \sum_{p,j} w_{p,j}(x;W)\right]$$

Mixture Density Network. Bishop [4] proposes the mixture density network as fellows,

$$q(y|x) = \sum_{k=1}^{K} \pi_k(x)\mathcal{N}(y|\mu_k(x),\sigma_k^2(x))$$

where $\pi_k(\cdot)$, $\mu_k(\cdot)$ and $\sigma_k(\cdot)$ are all modeled by neural networks. $\sum_{k=1}^{K}\pi_k(x)=1$ is guaranteed by using softmax activation function. The model is trained by minimizing the loss function,

$$\mathcal{L} = -\sum_{i=1}^{N} \log\left[\sum_{k=1}^{K} \pi_k(x_i)\mathcal{N}(y_i|\mu_k(x_i),\sigma_k^2(x_i))\right],$$

where $\{(x_i,y_i)\}_{i=1}^N$ are the observed data points.

Quantile Regression Forest Meinshausen [24] shows that random forests provide information about the full conditional distribution of the response variable, not only about the conditional mean. Conditional quantiles can be inferred with quantile regression forests, a generalisation of random forests. Quantile regression forests give a non-parametric and accurate way of estimating conditional quantiles for high-dimensional predictor variables. We refer to [24] for more details about QRF model. PCP needs to get samples from QRF. We first sample a percentile $\tau \sim U[0,1]$, the uniform distribution on the unit interval, and then use QRF to get the estimated conditional quantile value y_τ as a y sample.

E Full synthetic experiment results

We include the Full synthetic experiment results for 2D toy datasets, s-curve, half-moons, 25-Gaussians, 8-Gaussians, circle and swiss-roll, in Figure 7 and Figure 8. We compare conformal prediction with mean estimation (CP-MeanPred), CHR-QRF and CDSplit-MixD with our method (HD-)PCP.

The data used to plot Figure 5b and Figure 5c is included in Table 8. When ρ increases, the set size decreases for PCP and HD-PCP while keeping nearly constant for other baselines, which overlooks the joint relationship between targets.

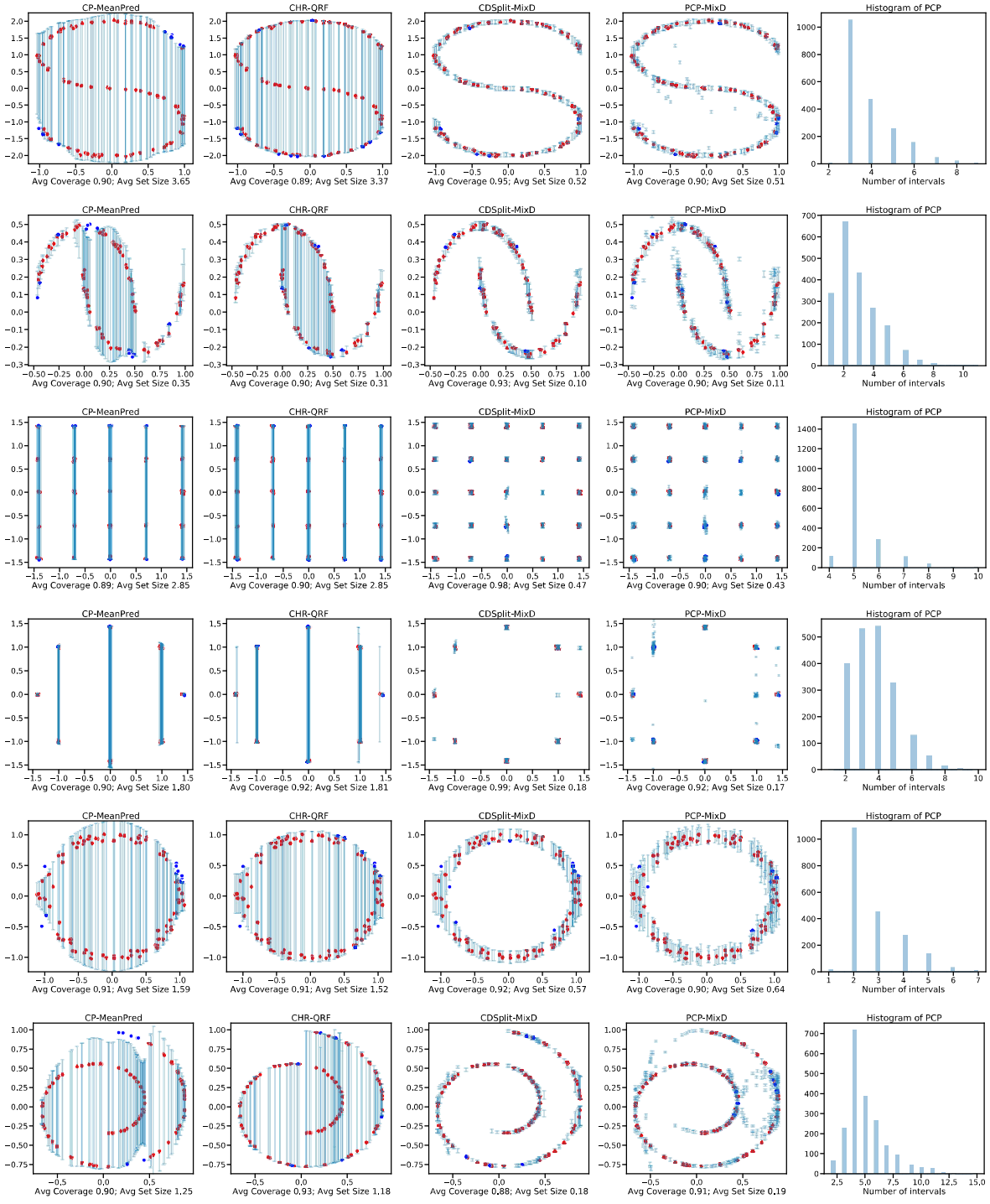


Figure 7: Visualization of predictive sets ($\alpha = 0.1$) on 2D toy datasets: s-curve, half-moons, 25-Gaussians, 8-Gaussians, circle and swiss-roll. For ours, we show the PCP in the last two columns. We show the predictive sets on 100 test data samples, where blues lines represent the predictive sets, blue dots are test points that are not covered by the predictive sets and reds dots are the test points covered. We show the marginal coverage and the average interval length across test datapoints in the x-axis label. The fifth column shows the histogram of the number predicted intervals of PCP.

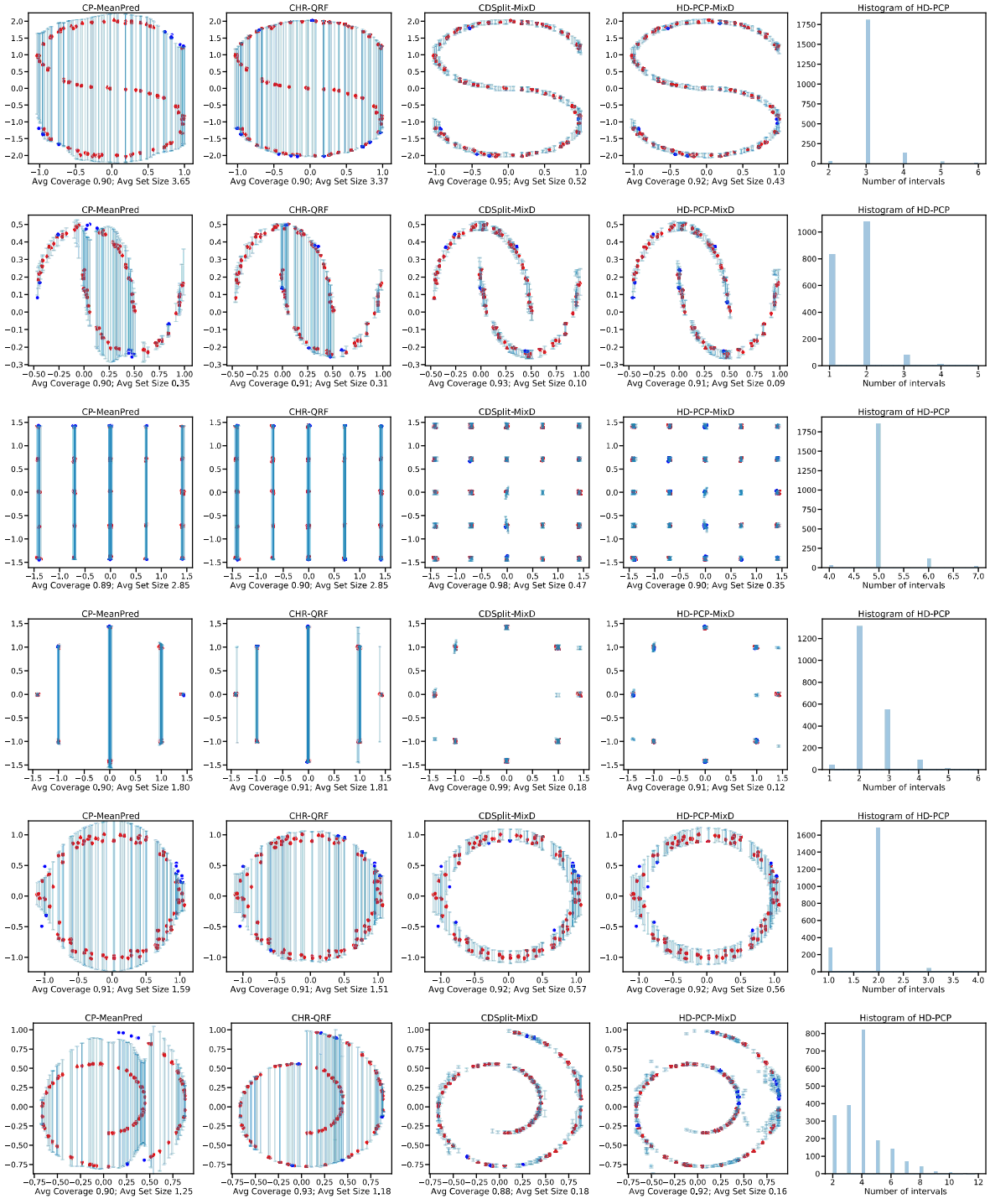


Figure 8: Visualization of predictive sets ($\alpha = 0.1, \beta = 0.2$) on 2D toy datasets: s-curve, half-moons, 25-Gaussians, 8-Gaussians, circle and swiss-roll. For ours, we show the HD-PCP in the last two columns. We show the predictive sets on 100 test data samples, where blues lines represent the predictive sets, blue dots are test points that are not covered by the predictive sets and reds dots are the test points covered. We show the marginal coverage and the average interval length across test datapoints in the x-axis label. The fifth column shows the histogram of the number predicted intervals of HD-PCP.

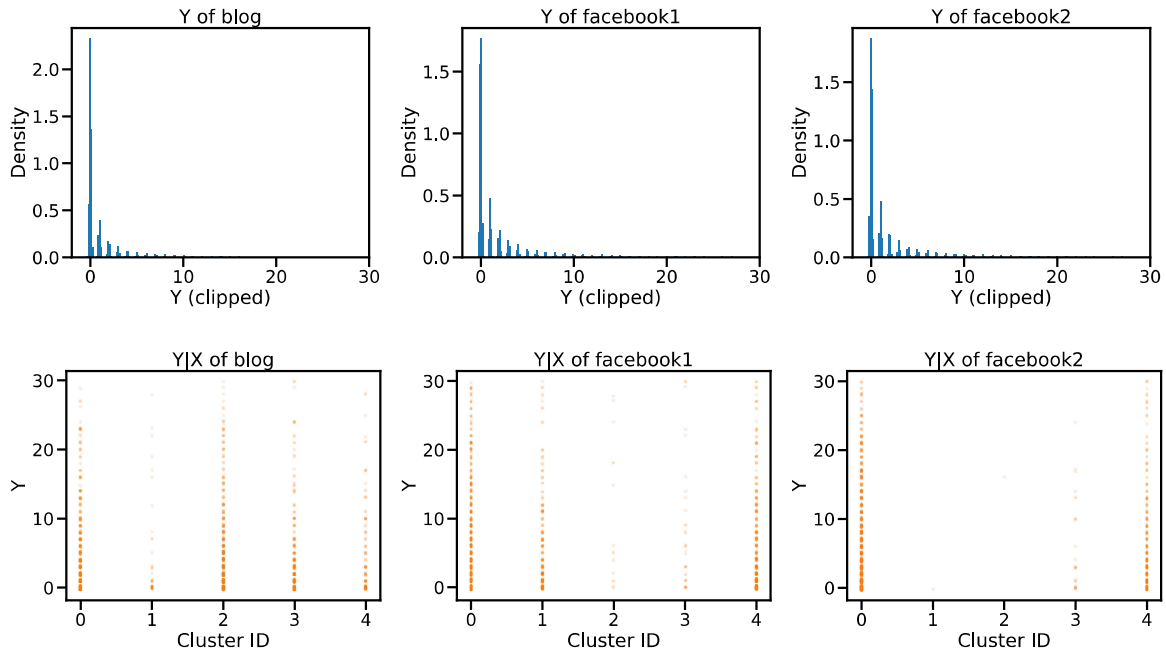


Figure 9: Y response variable of blog, facebook1, and facebook2 data. The first row shows the Y marginal histogram plots. The second row shows an approximation of $Y|X$ distribution, where we fit a K-Means on X to generate the 5 clusters. We could find that multimodality exhibits clearly in the $Y|X$ distribution.

F Full real data experiment results

We report all experiment results for our single target real data regression tasks in Table 5, Table 6 and Table 7. Table 4 illustrates the best performance of each method with best backbone model picked for each dataset repsectively.

G Additional Plots for NYC Taxi Data

Figure 10 shows the additonal plots for NYC Taxi data for PCP, HD-PCP, CHR and CDSplit. Each row representsone individual record in the test set and we show the predictive set generated by each algorithm. Clearly, PCP and HD-PCP generate the most informative predictive set where popular

Data	n_train	n_calib	n_test
bike	6886	2000	2000
bio	41730	2000	2000
blog	48397	2000	2000
facebook1	36948	2000	2000
facebook2	77311	2000	2000
meps_19	11785	2000	2000
meps_20	13541	2000	2000
meps_21	11656	2000	2000
temperature	5314	1138	1138

Table 3: Dataset splits for training, calibration and testing.

Data	Method	Marg. Coverage	Cond. Coverage	Set Size Mean	Set Size SE
bike	PCP-SIVI	0.90	0.86	134.55	1.13
	HD-PCP-MixD($\beta = 0.2$)	**0.90**	**0.88**	**102.92**	**0.48**
	CHR-QRF	0.90	0.88	204.10	1.03
	DistSplit	0.90	0.87	423.13	1.51
	CDSplit-MixD	0.92	0.92	115.74	0.50
	DCP	0.90	0.88	443.76	1.36
	CQR	0.90	0.89	403.88	0.86
bio	PCP-KMN	0.90	0.89	10.30	0.05
	HD-PCP-KMN($\beta = 0.2$)	**0.90**	**0.89**	**8.76**	**0.05**
	CHR-QRF	0.90	0.89	10.21	0.04
	DistSplit	0.90	0.89	13.19	0.04
	CDSplit-KMN	0.9	0.9	9.13	0.04
	DCP	0.90	0.89	12.95	0.04
	CQR2	0.90	0.89	12.88	0.05
blogdata	**PCP-QRF**	**0.89**	**0.85**	**3.68**	**0.65**
	HD-PCP-MixD($\beta = 0.2$)	0.90	0.87	9.44	0.19
	CHR-QRF	0.90	0.87	10.81	0.17
	DistSplit	0.90	0.87	16.27	0.23
	CDSplit-MixD	0.96	0.95	39.00	0.40
	DCP	0.90	0.88	1422.36	0.03
	CQR2	0.90	0.87	13.91	0.27
facebook1	**PCP-QRF**	**0.90**	**0.82**	**4.52**	**0.76**
	HD-PCP-KMN($\beta = 0.2$)	0.90	0.83	8.62	0.06
	CHR-NNet	0.90	0.87	9.96	0.11
	DistSplit	0.90	0.89	14.03	0.16
	CDSplit-MixD	0.95	0.95	33.69	0.16
	DCP	0.90	0.89	1303.01	0.04
	CQR2	0.90	0.88	12.17	0.15
facebook2	**PCP-QRF**	**0.90**	**0.82**	**3.62**	**0.72**
	HD-PCP-KMN($\beta = 0.2$)	0.90	0.82	8.34	0.07
	CHR-NNet	0.90	0.87	10.15	0.13
	DistSplit	0.90	0.89	13.48	0.19
	CDSplit-KMN	0.95	0.94	44.53	0.26
	DCP	0.90	0.89	1963.68	0.03
	CQR2	0.90	0.89	11.41	0.17
meps19	PCP-GAN	0.90	0.86	18.41	0.17
	HD-PCP-MixD($\beta = 0.2$)	**0.90**	**0.88**	**17.78**	**0.18**
	CHR-QRF	0.90	0.89	18.26	0.15
	DistSplit	0.90	0.89	29.96	0.28
	CDSplit-MixD	0.93	0.92	23.86	0.17
	DCP	0.90	0.88	559.23	0.01
	CQR	0.90	0.89	28.71	0.18
meps20	PCP-MixD	0.90	0.87	19.52	0.16
	HD-PCP-MixD($\beta = 0.2$)	**0.90**	**0.88**	**18.19**	**0.17**
	CHR-QRF	**0.90**	**0.90**	**17.94**	**0.18**
	DistSplit	0.90	0.90	29.35	0.23
	CDSplit-MixD	0.92	0.92	22.93	0.16
	DCP	0.90	0.88	520.25	0.01
	CQR	0.90	0.90	27.57	0.15
meps21	PCP-MixD	0.90	0.87	19.18	0.12
	HD-PCP-MixD($\beta = 0.2$)	**0.90**	**0.88**	**17.91**	**0.15**
	CHR-QRF	0.90	0.90	18.65	0.16
	DistSplit	0.90	0.89	30.32	0.31
	CDSplit-MixD	0.93	0.92	23.63	0.17
	DCP	0.90	0.88	531.25	0.01
	CQR	0.90	0.89	29.89	0.2
temperature	PCP-MixD	0.90	0.90	2.10	0.01
	HD-PCP-MixD($\beta = 0.2$)	**0.90**	**0.89**	**1.85**	**0.01**
	CHR-NNet	0.90	0.89	3.17	0.01
	DistSplit	0.90	0.89	3.07	0.01
	CDSplit-MixD	0.92	0.91	2.23	0.01
	DCP	0.90	0.88	3.1	0.02
	CQR2	0.90	0.88	3.14	0.02

Table 4: Best results of real data experiments (the best variant of each method for each dataset is selected).

Data	Method	Marg. Coverage	Cond. Coverage	Set Size Mean	Set Size SE
bike	PCP-SIVI	0.90	0.86	134.55	1.13
	PCP-GAN	0.90	0.88	399.32	2.48
	PCP-QRF	0.90	0.87	241.11	3.35
	PCP-MixD	0.90	0.87	128.13	0.53
	HD-PCP-MixD($\beta = 0.2$)	**0.90**	**0.88**	**102.92**	**0.48**
	PCP-KMN	0.90	0.88	172.92	1.04
	HD-PCP-KMN($\beta = 0.2$)	0.90	0.88	146.24	1.06
	CHR-NNet	0.90	0.89	353.51	1.59
	CHR-QRF	0.90	0.88	204.10	1.03
	DistSplit	0.90	0.87	423.13	1.51
	CDSplit-KMN	0.92	0.91	161.16	0.72
	CDSplit-MixD	0.92	0.92	115.74	0.50
	DCP	0.90	0.88	443.76	1.36
	CQR	0.90	0.89	403.88	0.86
	CQR2	0.90	0.88	416.75	1.57
bio	PCP-SIVI	0.90	0.89	14.08	0.06
	PCP-GAN	0.90	0.89	13.11	0.05
	PCP-QRF	0.90	0.89	11.08	0.16
	PCP-MixD	0.90	0.89	11.47	0.04
	HD-PCP-MixD($\beta = 0.2$)	0.90	0.90	10.06	0.05
	PCP-KMN	0.90	0.89	10.30	0.05
	HD-PCP-KMN($\beta = 0.2$)	**0.90**	**0.89**	**8.76**	**0.05**
	CHR-NNet	0.90	0.89	11.74	0.04
	CHR-QRF	0.90	0.89	10.21	0.04
	DistSplit	0.90	0.89	13.19	0.04
	CDSplit-KMN	0.90	0.90	9.13	0.04
	CDSplit-MixD	0.90	0.90	9.58	0.04
	DCP	0.90	0.89	12.95	0.04
	CQR	0.90	0.89	13.00	0.02
	CQR2	0.90	0.89	12.88	0.05
blog	PCP-SIVI	0.90	0.85	11.21	0.32
	PCP-GAN	0.90	0.86	11.67	0.16
	PCP-QRF	**0.89**	**0.85**	**3.68**	**0.65**
	PCP-MixD	0.90	0.85	10.78	0.17
	HD-PCP-MixD($\beta = 0.2$)	0.90	0.87	9.44	0.19
	PCP-KMN	0.90	0.85	10.67	0.13
	HD-PCP-KMN($\beta = 0.2$)	0.90	0.86	10.51	0.14
	CHR-NNet	0.90	0.88	11.1	0.19
	CHR-QRF	0.90	0.87	10.81	0.17
	DistSplit	0.90	0.87	16.27	0.23
	CDSplit-KMN	0.96	0.95	45.90	0.62
	CDSplit-MixD	0.96	0.95	39.00	0.40
	DCP	0.90	0.88	1422.36	0.03
	CQR	0.90	0.87	15.15	0.26
	CQR2	0.90	0.87	13.91	0.27
facebook1	PCP-SIVI	0.90	0.83	8.8	0.06
	PCP-GAN	0.90	0.85	9.22	0.05
	PCP-QRF	**0.90**	**0.82**	**4.52**	**0.76**
	PCP-MixD	0.90	0.82	9.99	0.14
	HD-PCP-MixD($\beta = 0.2$)	0.90	0.85	8.93	0.12
	PCP-KMN	0.90	0.82	10.60	0.06
	HD-PCP-KMN($\beta = 0.2$)	0.90	0.83	8.62	0.06
	CHR-NNet	0.90	0.87	9.96	0.11
	CHR-QRF	0.90	0.86	11.21	0.12
	DistSplit	0.90	0.89	14.03	0.16
	CDSplit-KMN	0.95	0.94	33.88	0.19
	CDSplit-MixD	0.95	0.95	33.69	0.16
	DCP	0.90	0.89	1303.01	0.04
	CQR	0.90	0.89	13.79	0.15
	CQR2	0.90	0.88	12.17	0.15

Table 5: Detailed results of experiments on data: bike, bio, blog and facebook1.

Data	Method	Marg. Coverage	Cond. Coverage	Set Size Mean	Set Size SE
facebook2	PCP-SIVI	0.90	0.83	8.69	0.17
	PCP-GAN	0.90	0.84	9.47	0.1
	PCP-QRF	**0.90**	**0.82**	**3.62**	**0.72**
	PCP-MixD	0.90	0.82	9.93	0.11
	HD-PCP-MixD($\beta = 0.2$)	0.90	0.84	8.84	0.10
	PCP-KMN	0.90	0.81	10.42	0.07
	HD-PCP-KMN($\beta = 0.2$)	0.90	0.82	8.34	0.07
	CHR-NNet	0.90	0.87	10.15	0.13
	CHR-QRF	0.90	0.87	10.81	0.14
	DistSplit	0.90	0.89	13.48	0.19
	CDSplit-KMN	0.95	0.94	44.53	0.26
	CDSplit-MixD	0.97	0.96	45.75	0.16
	DCP	0.90	0.89	1963.68	0.03
	CQR	0.90	0.89	13	0.17
	CQR2	0.90	0.89	11.41	0.17
meps19	PCP-SIVI	0.90	0.85	26.93	0.3
	PCP-GAN	0.90	0.86	18.41	0.17
	PCP-QRF	0.90	0.86	20.16	0.48
	PCP-MixD	0.90	0.87	19.28	0.16
	HD-PCP-MixD($\beta = 0.2$)	**0.90**	**0.88**	**17.78**	**0.18**
	PCP-KMN	0.90	0.85	23.24	0.21
	HD-PCP-KMN($\beta = 0.2$)	0.90	0.84	23.48	0.20
	CHR-NNet	0.90	0.90	20.17	0.2
	CHR-QRF	0.90	0.89	18.26	0.15
	DistSplit	0.90	0.89	29.96	0.28
	CDSplit-KMN	0.93	0.91	31.10	0.33
	CDSplit-MixD	0.93	0.92	23.86	0.17
	DCP	0.90	0.88	559.23	0.01
	CQR	0.90	0.89	28.71	0.18
	CQR2	0.90	0.89	30.78	0.36
meps20	PCP-SIVI	0.90	0.86	23.87	0.16
	PCP-GAN	0.90	0.86	19.92	0.18
	PCP-QRF	0.90	0.86	20.47	0.52
	PCP-MixD	0.90	0.87	19.52	0.16
	HD-PCP-MixD($\beta = 0.2$)	**0.90**	**0.88**	**18.19**	**0.17**
	PCP-KMN	0.90	0.85	22.96	0.17
	HD-PCP-KMN($\beta = 0.2$)	0.90	0.85	23.35	0.18
	CHR-NNet	0.90	0.9	19.43	0.18
	CHR-QRF	**0.90**	**0.90**	**17.94**	**0.18**
	DistSplit	0.90	0.90	29.35	0.23
	CDSplit-KMN	0.93	0.90	29.05	0.30
	CDSplit-MixD	0.92	0.92	22.93	0.16
	DCP	0.90	0.88	520.25	0.01
	CQR	0.90	0.90	27.57	0.15
	CQR2	0.90	0.90	29.94	0.31
meps21	PCP-SIVI	0.90	0.85	23.74	0.27
	PCP-GAN	0.90	0.86	19.73	0.16
	PCP-QRF	0.89	0.86	18.52	0.45
	PCP-MixD	0.90	0.87	19.18	0.12
	HD-PCP-MixD($\beta = 0.2$)	**0.90**	**0.88**	**17.91**	**0.15**
	PCP-KMN	0.90	0.85	23.13	0.17
	HD-PCP-KMN($\beta = 0.2$)	0.90	0.85	23.70	0.19
	CHR-NNet	0.90	0.90	20.07	0.22
	CHR-QRF	0.90	0.90	18.65	0.16
	DistSplit	0.90	0.89	30.32	0.31
	CDSplit-KMN	0.92	0.91	30.42	0.41
	CDSplit-MixD	0.93	0.92	23.63	0.17
	DCP	0.90	0.88	531.25	0.01
	CQR	0.90	0.89	29.89	0.2
	CQR2	0.90	0.89	31.78	0.36

Table 6: Detailed results of experiments on data: facebook2, meps19, meps20 and meps21.

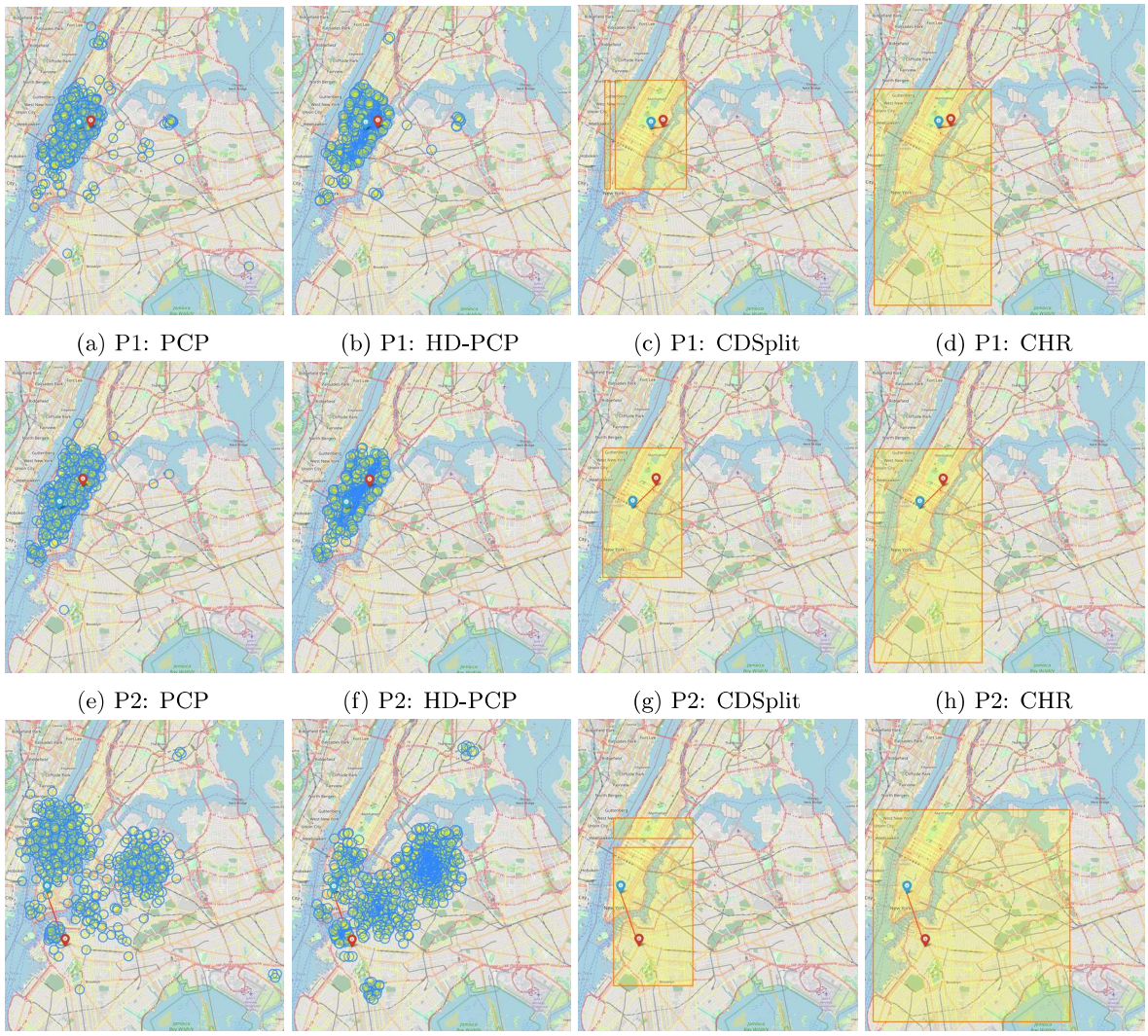
Data	Method	Marg. Coverage	Cond. Coverage	Set Size Mean	Set Size SE
temperature	PCP-SIVI	0.90	0.90	3.27	0.06
	PCP-GAN	0.90	0.89	3.51	0.04
	PCP-QRF	0.88	0.86	3.78	0.09
	PCP-MixD	0.90	0.90	2.10	0.01
	HD-PCP-MixD($\beta = 0.2$)	**0.90**	**0.89**	**1.85**	**0.01**
	PCP-KMN	0.90	0.89	2.68	0.01
	HD-PCP-KMN($\beta = 0.2$)	0.90	0.89	2.43	0.01
	CHR-NNet	0.90	0.89	3.17	0.01
	CHR-QRF	0.90	0.89	3.24	0.01
	DistSplit	0.90	0.89	3.07	0.01
	CDSplit-KMN	0.91	0.90	2.84	0.02
	CDSplit-MixD	0.92	0.91	2.23	0.01
	DCP	0.90	0.88	3.1	0.02
	CQR	0.90	0.87	3.55	0.03
	CQR2	0.90	0.88	3.14	0.02

Table 7: Detailed results of experiments on data: temperature.

Synthetic Data	Method	Cond. Coverage	Marg. Coverage	Set Size Mean	Set Size SE
$\rho = 0$	HD-PCP-MixD	0.92 (0.03)	0.90 (0.01)	356.82	10.78
	PCP-MixD	0.94 (0.02)	0.90 (0.01)	412.40	12.70
	CHR-NNet	0.88 (0.05)	0.92 (0.01)	690.49	51.55
	DistSplit	0.93 (0.01)	0.92 (0.01)	714.92	54.81
	CDSplit-MixD	0.87 (0.03)	0.90 (0.01)	437.98	20.20
	DCP	0.90 (0.03)	0.92 (0.01)	710.64	53.99
	CQR	0.95 (0.02)	0.92 (0.01)	667.01	51.47
	CQR2	0.89 (0.03)	0.92 (0.01)	694.64	52.40
$\rho = 5$	HD-PCP-MixD	0.88 (0.03)	0.89 (0.01)	302.64	9.88
	PCP-MixD	0.87 (0.03)	0.88 (0.01)	348.93	10.70
	CHR-NNet	0.92 (0.03)	0.91 (0.01)	689.22	57.47
	DistSplit	0.93 (0.02)	0.91 (0.01)	702.56	54.02
	CDSplit-MixD	0.87 (0.03)	0.90 (0.01)	415.86	17.49
	DCP	0.89 (0.05)	0.91 (0.02)	697.18	54.01
	CQR	0.87 (0.03)	0.91 (0.02)	653.64	53.59
	CQR2	0.88 (0.05)	0.91 (0.02)	693.10	56.18
$\rho = 9$	HD-PCP-MixD	0.83 (0.07)	0.89 (0.02)	171.61	8.36
	PCP-MixD	0.89 (0.03)	090 (0.01)	205.63	5.76
	CHR-NNet	0.92 (0.02)	0.92 (0.02)	664.31	51.59
	DistSplit	0.91 (0.05)	0.91 (0.01)	689.72	59.96
	CDSplit-MixD	0.92 (0.02)	0.91 (0.01)	416.86	16.96
	DCP	0.90 (0.03)	0.91 (0.02)	666.13	50.61
	CQR	0.93 (0.04)	0.92 (0.01)	639.00	50.19
	CQR2	0.92 (0.02)	0.91 (0.01)	663.54	54.59

Table 8: Detailed results for multidimensional target synthetic dataset. The set size for PCP and HD-PCP decreases when ρ increases while the set sizes for other baselines are similar for different ρ since the marginal distribution remains the same.

neighborhoods and airports are tagged, while HD-PCP offers a more sparse and clean set. As expected, CHR offers a wide predictive set. Both CDSplit and CHR fail to provide predictive set with clear interpretation. Figure 11 shows the predictive set and heatmap for pickup from SOHO and Chinatown. Most popular spots in NYC have higher density, which means passengers are more likely to be dropped off there.



(a) P1: PCP (b) P1: HD-PCP (c) P1: CDSplit (d) P1: CHR

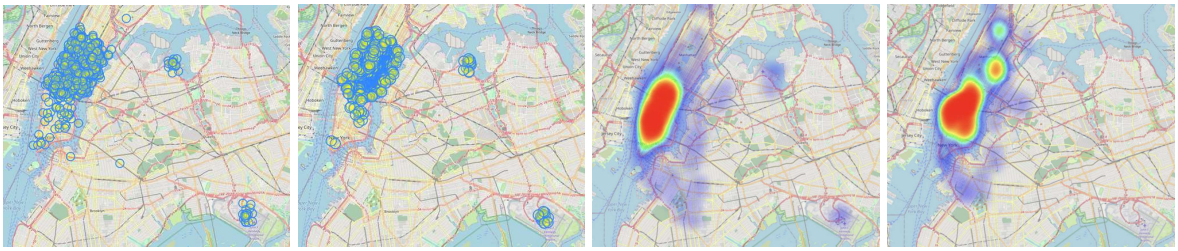
(e) P2: PCP (f) P2: HD-PCP (g) P2: CDSplit (h) P2: CHR

(i) P3: PCP (j) P3: HD-PCP (k) P3: CDSplit (l) P3: CHR

Figure 10: NYC Taxi data. Red Pin: pickup location; Blue Pin: dropoff location. Left to right: Predictive set output by PCP, HD-PCP, CDSplit and CHR. Each row represents a random selected individual record sampled from the dataset.

H Additional Results for Multi-Target Regression Task

We include more experiment results for multi-target regression task in this section. We use two datasets for river flow prediction [35] and stock prediction from StatLib repository. River flow dataset predicts the rivernetwork flows for future 48 hours for 8 sites (8 targets) and the stock dataset has stock price for 10 aerospace companies and we try to predict 3 companies' price using remaining companies'. There are 64 features including past river flow information for river flow predictions. For train, calibration and test size, we use 6925, 2000 and 200 for river flow prediction and 750, 100 and 100 for stock prediction



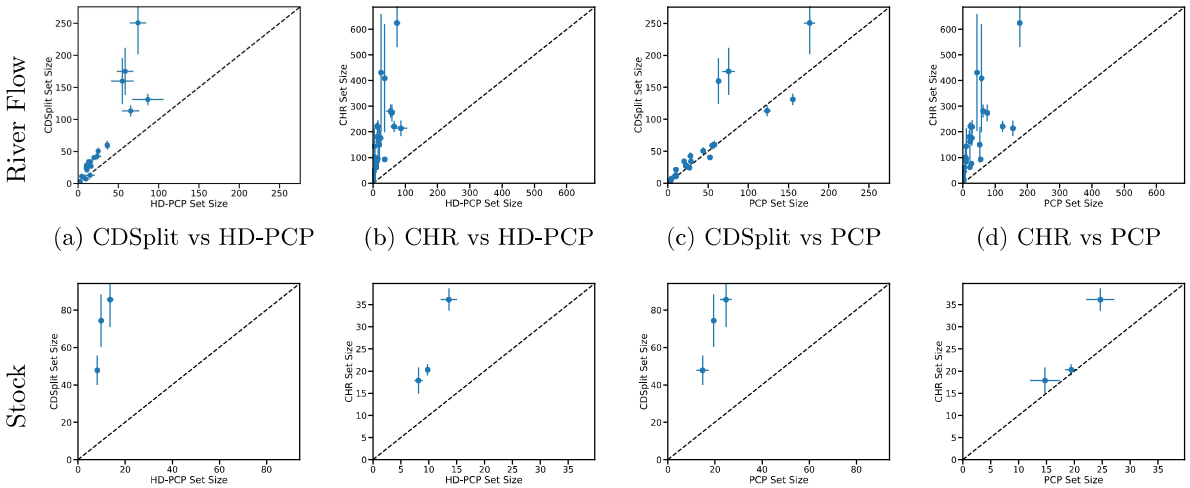
(a) PCP (b) HD-PCP (c) SOHO (d) Chinatown

Figure 11: NYC Taxi data. (a), (b): predictive set for an individual using PCP and HD-PCP; (c): predictive set for riders from SoHo; (d): predictive set for riders from Chinatown.

respectively.

Since the Monte-Carlo estimation of overlapping hypersphere suffers from curse of dimensionality. We convert each dataset into two-dimensional pairwise comparisions to evaluate the robustness of each method (8 targets result in 28 pairs). We plot the pairwise comparison of PCP and HD-PCP against CHR and CDSplit, the two baselines that performs generally the best among other datasets. We use Mixture density Network for CDSplit, PCP and HD-PCP and Neural Netork based CHR, the results are averaged over 5 runs.

For X-axis, we plot the set size of PCP / HD-PCP and for Y-axis, we plot the set size for CDSplit and CHR, and we also show the $Y = X$ line. If all points fall into the left region, it means PCP / HD-PCP outputs a sharper predictive set. For PCP, almost all pointsfall into the left region, which indicates PCP has a better or comparable performance with CDSplit and CHR in all pairwise comparisions. HD-PCP has a much better performance and the points all fall into the far left part in the figure, which shows HD-PCP offers a much sharper predictive set.



(a) CDSplit vs HD-PCP (b) CHR vs HD-PCP (c) CDSplit vs PCP (d) CHR vs PCP

(e) CDSplit vs HD-PCP (f) CHR vs HD-PCP (g) CDSplit vs PCP (h) CHR vs PCP

Figure 12: Additional Results for Multi-Target Regression Task. Each point corresponds to the size of predictive set for two elements of the target vector.