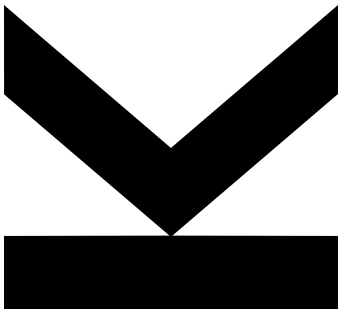
JOHANNES KEPLER
UNIVERSITY LINZ

Author / Eingereicht von
Alexander Krauck
Matriculation number / Matrikelnummer
K11904235

Conditional Density Estimation and Conformal Prediction are Equivalent Tasks in Regression



Master Thesis

to obtain the academic degree of

Master of Science

in the Master's Program

Artificial Intelligence

Submission / Angefertigt am
Institute of Machine Learning

Thesis Supervisor / First Supervisor / BeurteilerIn / ErstbeurteilerIn / ErstbetreuerIn
DI Dr. **Name**

Second Supervisor / ZweitbeurteilerIn / ZweitbetreuerIn
Name

Assistant Thesis Supervisor / Mitbetreuung
Name

Mai 2024

JOHANNES KEPLER UNIVERSITY LINZ
Altenbergerstraße 69
4040 Linz, Austria
www.jku.at
DVR 0093696

Kurzfassung

Kurzfassung auf Deutsch.

Abstract

Abstract in English.

Contents

List of Figures

List of Tables

1 Introduction

Machine Learning (ML) models that can not only estimate single targets accurately, but that are capable of estimating distributional information as well as uncertainty are becoming exceedendly important [Hüllermeier and Waegeman, 2021; Gawlikowski et al., 2023]. The reason therefore is, that most modern ML techniques are mostly black box models that have little intuitive reason behind their predictions but often act on abstract latent representations, especially in the case of Artificial Neural Networks (NN). With a strong focus on regression tasks, in this work we aim to develop a novel understanding of uncertainty estimating methods, where in particular we show that we can combine ideas from multiple different task-types. We show that Conditional Density Estimation (CDE), Conformal Prediction (CP) and Quantile Regression (QR) are fundamentally the same task since they all require to model parts of the conditional probability density function (PDF). Moreover, we develop a novel way to perceive the maximum log likelihood (MLL) objective function, where we show that it is equivalent to the objective function of CP, as we define it. This allows us to split the MLL objective into a constrainted optimization problem, where we intuitively minimize the size of the peaks (we make them as narrow as possible) with the constraint that we maintain calibration. Finally, we aim to introduce a novel way of perceiving epistemic uncertainty in CDE. All details to the theoretical concepts and novel insights are detailed in Section 2.

Before we go into the details of the theoretical background, we want to give a brief overview of the motivation and the exact research questions that we aim to answer in this work. In particular, the practical implications and applications of this work are first discussed below in Section 1.1. Even though this work is not centered around a particular practical application but is more at home in the theoretical part of machine learning the author of this work belives that a motivation in the practical domain is very important to make the work more accessible and to show the relevance of the work. In particular, we aim to show that the methods proposed in this work can be applied to a broad range of

practical tasks and that they can have a significant positive impact on the performance of machine learning models in those tasks.

1.1 Background

Mostly in risk-sensitive practical domains like finance and life science uncertainty estimation is crucial [Abdar et al., 2021; Xia et al., 2020; Ghesu et al., 2021; Mashrur et al., 2020]. Therefore there exist two fundamental approaches to perceive uncertainty. First, there is the uncertainty that is inherent in the data, which means that for a give input, there are multiple possible outputs which are plausible, which also can not be reduced. This is called aleatoric uncertainty and it is the main task of CDE and CP to estimate this kind of uncertainty. Secondly, and less researched in the domain of regression, there is the uncertainty of the model, which could occur if the model is shown a sample that it can not generalize to, based on the training data it was trained on. This uncertainty is termed epistemic uncertainty. In this work we argue that in order to make reliable and informed decisions in high risk tasks, it is crucial to have methods to estimate both kinds of uncertainty, however, most recent works in regression tasks exclusively focus on estimating aleatoric uncertainty [Y. Romano, Patterson, and Candes, 2019; Sesia and Candès, 2020; Angelopoulos and Bates, 2021; Chernozhukov, Wüthrich, and Zhu, 2021; Sesia and Y. Romano, 2021; Oliveira et al., 2022; J. V. Romano, 2022; Izbicki, G. Shimizu, and Stern, 2022; Gupta, Kuchibhotla, and Ramdas, 2022; Auer et al., 2024]. In particular, methods like CDE, CP and QR can only directly estimate the aleatoric uncertainty, which is also the reason why epistemic uncertainty has been out of focus. However, in this work we argue that a type of epistemic uncertainty is already unknowingly being induced into models in many cases, that is with calibration.

1.1.1 Uncertainty in Finance

Energy Price Prediction

A practical task that we particularly focus on in this work is an energy price prediction task, in cooperation with Voestalpine AG, where we attempt to estimate the distribution of

the imbalance energy price[1] of Austria given multiple descriptive input variables/features. In particular, the imbalance energy price we aim to predict is unknown at the time of consumption/production and is only much later revealed.

If an entity on the energy market wants to buy or sell electricity at a certain time, this entity does indicate how much electricity it wants to buy or sell for the dayahead price which is known. Hoever, if this entity produces/consumes more energy than agreed on, the energy imbalance price holds for this over-/underestimation, but this price is only known after the fact and heavily depends on what other entities on the market did. In particular, the imbalance energy price is a very volatile price, making it a relevant use case for uncertainty estimation since it can impact the decision if electricity should be bought or produced at a given time.

Stock Price Prediction

In more traditional finance tasks, we mostly try to predict price-trends of assets like stocks [Ritika Singh and Srivastava, 2017], currencies [Hassanpour, 2023], cryptocurrencies [Alessandretti et al., 2018] and other equities. Those predictions are then used either for assisted decision making of analysists or for automated and potentially high frequency trading. Especially when making decisions with high stakes it is crucial to know exactly the risk that is taken with a certain decision, ideally with certain gurantees. For example, it might be essential not to loose more than a specified amount of money with a trading decision with a certain probabilistic confidence level. A known quantity in trading is the Value at Risk (VaR) as introduced by [Jorion, 2007], which is the maximum amount of money that can be lost with a certain confidence level.

1.1.2 Uncertainty in Healthcare and Life Science

In life science uncertainty aware ML methods have also been of increasing interest [Loftus et al., 2022; Lambert et al., 2024]. Often it is of relevance to estimate some regression targets like from personalized drug dosage prediction [Wu et al., 2023], amniotic fluid volume prediction [Csillag et al., 2023], tumor size quantification [Prasad et al., 2023],

[1] For precise details on this quantity we refer to `https://markttransparenz.apg.at/en/markt/Markttransparenz/Netzregelung/Ausgleichsenergiepreise`

time-to-event prediction [Kvamme, Borgan, and Scheel, 2019; Sloma et al., 2021]. It is crucial for those tasks to not only know the average outcome, but to also be able to see if there are small probability events that could still happen with some plausible probabilitly. For example, if we predict the size of the tumor of a patient, the main probability density peak might be at a certain size, but it might be possible that there is another smaller peak at a much larger size which could lead to a more urgent tratment strategy. In this case, it is crucial to know the full distribution of the target variable and not only the mean. For similar reasons the epistemic uncertainty is also extremely relevant there. The model might not be able to generalize to a certain patient, which could lead to a completely random prediction and thus to a horrible decision if the doctor is not informed about the uncertainty of the model.

1.2 Related Work and Motivation

Many different related works about CDE [Bishop, 1994; Rothfuss, Ferreira, Walther, et al., 2019; Trippe and Turner, 2018; Rothfuss, Ferreira, Boehm, et al., 2019; Ambrogioni et al., 2017], CP [Izbicki, G. Shimizu, and Stern, 2022; Chernozhukov, Wüthrich, and Zhu, 2021; Y. Romano, Patterson, and Candes, 2019; Papadopoulos, 2008; Angelopoulos and Bates, 2021], QR [Chung et al., 2020] aswell as uncertainty estimation in general [Gal and Ghahramani, 2016; Hüllermeier and Waegeman, 2021; Abdar et al., 2021; Klotz et al., 2022] have been published in recent years. Most of those methods essentially attempt to model certain parts of the uncertainty of target variables given descriptive feature variables. Moreover, there exist works that observe that certain concepts can be transferred from one domain of uncertainty estimation to another [Chernozhukov, Wüthrich, and Zhu, 2021]. In this work we aim to show that not only ideas can be borrowed from one task-domain to another one, but that in fact, the tasks of CDE, CP and QR are fundamentally the same and that all concepts of one task-domain can fully be transferred to the other one. This not only gives a theoretical and eye-opening realization, but also makes many improvements possible.

In particular, in the task-domain of CP in recent years many different methods to find the best CP intervals have been proposed [Sesia and Y. Romano, 2021; Chernozhukov, Wüthrich, and Zhu, 2021; Izbicki, G. Shimizu, and Stern, 2022]. However, the produced intervals of those methods are all still fundamentally argnostic to the full distribution of

the target variables and do not consider the fact that it might be better to instead of making the focus interval-size centric to make the focus density/probability mass centric. This is meant in a way that while previous method try to make the intervals as narrow as possible, they ignore the underlying density completely causing that in some cases important and high density events can be missed. We argue, that including those high density events in the CP intervals, even if we obtain a higher coverage level that we intended, can be beneficial for the decision making process, especially in practical tasks. Therefore we argue that using an algorithm that is based on the highest density regions (HDR) as described by [Hyndman, 1996] is a better practice. In particular, proofably HDR is always optimal in terms of interval size if we accept mutliple regions as prediciton, even under the restriction to stay perfectly calibrated. However, we further argue that it is better to instead of ignoring the HDR under a specified confidence level for the sole purpose to find the shortest single intervals is practically worse than to connect the intervals produced by HDR, even if that makes the method slightly overcalibrated.

Moreover, in recent years the possibly most prominent topic of uncertainty in machine learning has been the estimation of epistemic uncertainty [Barber and Bishop, 1998; Neal, 2012; Gal and Ghahramani, 2016; Schweighofer et al., 2023; Gawlikowski et al., 2023]where the main task is to estimate the model's confidence, that it's predictions are accurate, given an unseen sample. This is not to be confused with the conditional PDF produced by CDE methods, which only models the aleatoric uncertainty. While the mentioned methods exist to estimate the epistemic uncertainty primarely while only indirectly estimating the aleatoric uncertainty with hefty limitations, in this work we propose a novel way of doing it the other way around. The new method not only allows the estimation of epistemic uncertainty values for each sample, but we can actually for each sample see where in the conditional PDF the model has epistemic uncertainty. This is a novel way of looking at epistemic uncertainty and can be very beneficial for the decision making process for single sample predictions. Especially, this method, instead of allowing to estimate aleatoric uncertainty in a homogeneous way by substracting the epistemic uncertainty from the total uncertainty as by [Gal and Ghahramani, 2016], we instead predict the aleatoric uncertainty via CDE and are able to infer the epistemic uncertainty from the difference of the total uncertainty and the aleatoric uncertainty, where the total uncertainty is estimated in a novel way that is based on calibration and HDR. We show that calibration as often done in CP, actually infuses epistemic uncertainty into the modeled PDF that describes

only aleatoric uncertainty. This essentially allows us to see both, epistemic and aleatoric uncertainty in a multimodal non-restrictive way.

1.3 Research Questions

A wide variety of research questions are treated within this work. The most essential question is how exactly CP, CDE and QR are related or if they are really equivalent fundamentally. Moreover, if they are equivalent, the question would be how the different methods can benefit from techniques that have been developed for other methods. In particular, one suspicion is that it might improve CP intervals if we can use properties of inductive biases provided by e.g. a mixture of gaussian estimtated by using Mixture Density Networks (MDNs) [Bishop, 1994].

Another fundamental question is whether it actually has practical benefits to have awareness of the multimodality of the target variable. Some real world distributions are clearly multimodal, as we will also explore in the experimental section of this work, however, does it even have any practical implications to have awareness of this multimodality or do we in practice only care about lower/upper bounds of the target variable? We want to develop a list of practical tasks where multimodality is crucial if it exists. Moreover, one may also ask that if we only care about lower/upper bounds, does it even make sense to consider models that can handle multimodality or could we just as well use a single Gaussian distribution to model the target variable with no performance loss?

Another question is that if CDE and CP are equivalent, what optimization strategy should we choose or which ones can we even choose. CDE mostly relies on the MLL objective function, while CP often relies on the pinball loss. However, there are other methods to do both tasks too.

Moreover, we want to analyze the often made statement that CP and QR are fundamentally distribution free methods. In particular, we argue that it is in practice impossible to stay distribution free and that essentially all CP and CDE methods are using underlying distributions, for some more explicit than for others. In particular this goes towards the realization that learning without of assumptions isn't possible [David H. Wolpert, 1996] and that we should rather focus on making the right assumptions instead of being ignorant to assumptions that need to be made.

Furthermore we want to analyze, in problems that do not support strong assumptions, like time series tasks, how we can analyze the transitions of the estimated distributions, especially in relation to calibration. In particular, the question is, when we can calibrate the model on new data that has a different distribution than the training data, what assumptions are we implicitly making and how can we make the model more robust to those assumptions. Especially because if the new data is completely different from the training data, a mere calibration can hardly be enough to make the model work on the new data. In particular, this also goes more into the question what exactly recalibration, especially in CP tasks, actually does.

Moreover, we want to analyze how epistemic and aleatoric uncertainties are both contained in CDE and CP methods. In particular, we want to argue that when calibrating a CP model, we are actually infusing epistemic uncertainty into the existing prediction. This is sourced in the intuition that when we recalibrate, then we are essentially expanding or shrinking the intervals, which could be, under certain regularity conditions, interpreted as making more specific statements about one particular sample or instead making more general statements about the marginal distribution of the targets. We want to argue that the marginal distribution of the targets is actually a very low assumption that we can make, if not the lowest, and that anything more specific to a sample is then basically the certainty or information that we have about a sample. The epistemic uncertainty then would be how much we need to interpolate between the marginal distribution and the specific sample to make a calibrated prediction, in a non-linear way.

Furthermore we want to see how, in practice, evaluated on multiple benchmarks those insights we give improve the performance and what hyperparameters are generally a good choice. In particular, we want to see if we can infer general patterns or rules that can be applied to a wide variety of tasks in the area of CDE and CP.

1.4 Contributions

The main contributions of this work are as follows:

- We show that CDE, CP and QR are fundamentally the same task and that all concepts of one task-domain can fully be transferred to the other one.

- We propose a novel way to perceive the MLL objective function, where we show that it is equivalent to the objective function of CP, as we define it. This allows us to split the MLL objective into a constrainted optimization problem, where we intuitively minimize the size of the peaks (we make them as narrow as possible) with the constraint that we maintain calibration.
- We show that it might improve CP intervals if we can use properties of inductive biases provided by e.g. a mixture of gaussian estimtated by using Mixture Density Networks (MDNs).
- We offer a novel way to estimate epistemic uncertainty in CDE and CP methods, where we show that calibration as often done in CP, actually infuses epistemic uncertainty into the modeled PDF that describes only aleatoric uncertainty.
- Finally we empirically verify our insights on multiple benchmarks and show that our method is competitive with state-of-the-art methods and that it can be applied to a wide variety of tasks. Thereby, we also provide a general overview of hyperapareters that are generally a good choice for CDE and CP tasks. In particular we also propose new hyperparameters and show their impact.

1.5 Structure of the Work

First, in Section 2 we give a thorough introduction into CP, CDE and QR and show they are fundamentally the same task. Moreover, we show how we can infer CP from CDE and how we can estimate epistemic uncertainty in CDE and CP methods. In Section 3 we show how our insights can be applied to a wide variety of tasks and how they improve the performance of CDE, QR and CP methods. Finally, in Section 4 we summarize our insights and give an outlook on future work.

2 Theoretical Analysis

2.1 Preliminaries

In the rest of this work we will unless stated otherwise always be assuming a machine learning task where samples have the assumption of exchangability. Moreover, for the theoretical part of this paper, we also assume that we have unlimited samples unless stated otherwise which is necessary to make certain theoretical statements and in particular with limted data those statements all hold asymptotically.

Thus, unless stated otherwise, let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space for our task. Furthermore, let $\mathbf{Z}_i = (\mathbf{X}_i, \mathbf{Y}_i)$ random variables on the index set $\mathcal{I}$ with $\mathbf{X}_i : \Omega \to \mathbb{R}^n$ and $\mathbf{Y}_i : \Omega \to \mathbb{R}^m$ for all $i \in \mathcal{I}$, where each pair represents one sample. Moreover, let all $(\mathbf{X}_i, \mathbf{Y}_i)$ be exchangable, which means that for any permutation π of the index set $\mathcal{I}$ the join probability distribution remains the same, i.e.

$$\forall \mathbf{x}_i \in \mathbb{R}^n, \mathbf{y}_i \in \mathbb{R}^m : \mathbb{P}\left(\bigcap_{i\in\mathcal{I}} (\mathbf{X}_{\pi(i)} \leq \mathbf{x}_i, \mathbf{Y}_{\pi(i)} \leq \mathbf{y}_i)\right) = \mathbb{P}\left(\bigcap_{i\in\mathcal{I}} (\mathbf{X}_i \leq \mathbf{x}_i, \mathbf{Y}_i \leq \mathbf{y}_i)\right) \quad (2.1)$$

Furthermore, let $I : \Omega \to \mathcal{I}$ be a random variable defined on the probability space that selects an index at random with the further simplification that every index has the same probability to be selected (uniform).

This extensive notation is required mostly in order to define exchangability, however, in many proofs in this paper we will not require the use of the probability space explicitly, in particular with the index set since we often just integrate over the whole sample space $\mathbb{R}^{n+m}$.

The task of the ML methods discussed here is always to predict some property, like the conditional PDF, about the target variable $\mathbf{Y}_I$ given the features $\mathbf{X}_I$. In the more practical case we only have access to a subset of the indices in the index set, which is the observed data set $\mathcal{D} = \{(\mathbf{X}_i, \mathbf{Y}_i)\}_{i \in I_\mathcal{D}}$, with $|I_\mathcal{D}| < \infty$.

Moreover, any model that we discuss here, regardless if it is CDE, CP or QR, will be parameterized by some parameters $\theta \in \Theta$ where Θ is the parameter space. Furthermore, we will make the assumption that the model class can perfectly model the optimal model, which seems like a strong assumption, but considering that we focus on model classes that either can be tweaked to be very expressive or NN's that even are universal function approximators [Hornik, Stinchcombe, and White, 1989], this assumption is not unreasonably strong. The optimal parameter set is indicated by θ^*.

For the above definitions the corresponding PDF of and event $\mathbf{z} = (\mathbf{x}, \mathbf{y}) \sim (\mathbf{X}_I, \mathbf{Y}_I)$ is defined by

$$p(\mathbf{x}, \mathbf{y}) := \frac{d^2 \mathbb{P}(\mathbf{X}_I \leq \mathbf{x}, \mathbf{Y}_I \leq \mathbf{y})}{d\mathbf{x} d\mathbf{y}} \tag{2.2}$$

for $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^m$. It is left as a hint to the reader that $\mathbb{P}(\mathbf{X}_I \leq \mathbf{x}, \mathbf{Y}_I \leq \mathbf{y})$ in the above equation is the cumulative distribution function (CDF). The marginal and conditional PDFs can then be found by integrating out and normalizing with the marginal respectively.

Finally in order to analytically show certain results we require some standard assumptions of the underlying conditional PDF $p(\cdot \mid \mathbf{x})$ which hold $\forall \mathbf{x} \in \mathbb{R}^n$.

Definition 2.1.1 (Standard Assumptions)**.**

1. The PDF p is continously differentiable a.e.
2. $\forall b \in \mathbb{R}^+ : \mathbb{P}(p(\mathbf{Y}) = b \mid \mathbf{x}) = \lambda(p(\mathbf{Y}) = b \mid \mathbf{x}) = 0$
3. $p > 0$ a.e.

Those assumptions are weak in practice since we can approximate any PDF that does not fullfill those assumptions with a PDF that does fullfill those assumptions arbitrary well. We refer to [Klenke, 2013] for a formal argument why this is true.

In the following we will first introduce the three main tasks that we will discuss in this work, namely CDE in Section 2.2, CP in Section 2.4 and QE in Section 2.3. We will then show that those tasks are fundamentally the same in Section 2.5, that optimizing the MLL objective function is equivalent to optimizing the CP objective function as we define it in Section 2.6.1 and finally a rigorous examination of recalibration where we show that it infuses epistemic uncertainty into the modeled conditional PDF that describes only aleatoric uncertainty in Section 2.7 and Section 2.8. It given as a recommendation to the reader to read the sections in order as certain essential concepts that are introduced in the first sections are used in the later sections.

2.2 Conditional Density Estimation

The goal of CDE methods is to estimate the conditional PDF $p(\mathbf{y} \mid \mathbf{x})$ of samples $(\mathbf{x}, \mathbf{y}) \sim \mathbf{Z}_I$. The objective function used for CDE is usually likelihood function, which is given by $p(\mathbf{y} \mid \mathbf{x})$ for one sample and $\mathbb{E}_\Omega \left[\log p(\mathbf{Y}_I \mid \mathbf{X}_I)\right]$ generally, where we take the logarithm of the likelihood function and thereafter can take the integral (the expectation) over the whole sample space, which is valid since the logarithm is a strictly monotonous function.

The reason we do not restrict ourselves here to a finite sample set is that we want to make general statements about the training objective of CDE methods. The objective in a finite sample set setting is analogous, but we can not make general statements about the whole sample space.

Smoothness Assumptions

As we usually in the context of ML like to obtain a model that can generalize, we need to make smoothness assumptions in the context of CDE, as without them it is easy to define the optimal model as the delta function at the observed data points. This is not a useful model as it will not generalize to new data points.

As we can see in the work of [Rothfuss, Ferreira, Boehm, et al., 2019] the objective function of the model equals

$$\arg\max_{\theta\in\Theta}\sum_{i=1}^{n}\log\hat{f}_\theta\left(x_i\right)=\arg\min_{\theta\in\Theta}\mathcal{D}_{KL}\left(p_{\mathcal{D}}\|\hat{f}_\theta\right) \tag{2.3}$$

where $p_{\mathcal{D}}$ is the delta distribution with peaks at the observed target locations. If we consider the full sample space in the optimization problem, then $p_{\mathcal{D}}$ reduces p. Intuitively this equation indicates that MLL estimator is the same as the estimator that has the minimal Kullback-Leibler divergence between the true distribution and itself.

It is easy to see that finding the optimal model for this problem, at least if we limit the number of samples that we can learn from to a finite amount, is meaningless since it will not generalize with the delta function. However, if we make the assumption that the target variable is smooth, then we can assume something like a Gaussian distribution over each observed target and input variable. This is also the apporach that [Rothfuss, Ferreira, Boehm, et al., 2019] introduce in their work where they analytically show that adding noise to the targets and inputs is beneficial for the generalization of the model. In order to gain an intuitive understanding why this is required one needs to imagine the input and output variables as a join probability distribution. If we add noise to each sample than this noise spans thru all dimensions of this distribution and thus we can find reasonable output predictions for unseen input features.

2.3 Quantile Regression

The goal of QR is to estimate specific quantiles of the target variable given the input variables. Formally, that means we want to predict $Q(\mathbf{x})$ for a quantile q and $(\mathbf{x},\mathbf{y})\in\mathbf{Z}_I$ such that $\mathbb{P}(\mathbf{Y}_I\leq Q(\mathbf{X}_I))=q$. The most used objective function for QR is the pinball loss as introduced by [Koenker and Bassett Jr, 1978], which for one sample is defined as $\max((\mathbf{y}-Q(\mathbf{x}))\cdot q,(\mathbf{y}-Q(\mathbf{x}))\cdot(1-q))$ and where we find the optimal parameters at $\min\mathbb{E}_\Omega\left[\max\left((\mathbf{Y}_I-Q(\mathbf{X}_I))\cdot q,(\mathbf{Y}_I-Q(\mathbf{X}_I))\cdot(1-q)\right)\right]$ where we take the expectation over the whole sample space. [Koenker and Bassett Jr, 1978] show that the pinball loss is optimal at the true quantile function. There have also been more recent works with different loss functions, like in the work by [Chung et al., 2021], however, for the scope of this work it is sufficient to use the definition of the pinball loss.

In particular, when estimating a tight grid of quantiles, the QR model can be used to estimated the full conditional CDF of the target variables given the input variables. When estimating a tight grid on limited data, it can happen that two quantiles are switched in position. In this case it is a common practice to simply swap the two quantiles out, i.e. we sort the outputs of the model and only then apply the pinball loss. This is a common practice in QR and is also used in the work of [Sesia and Y. Romano, 2021].

2.4 Conformal Prediction

Conformal Prediction generally is the task of finding sets of possible outcomes that in expectation will contain the true outcome with a certain miscoverage level α. Formally, that means we predict sets of possible outcomes $C(\mathbf{x})$ where $x \sim \mathbf{X}_I$ such that $\mathbb{P}(\mathbf{Y}_I \in C(\mathbf{X}_I)) = 1 - \alpha =: a$, where a is the confidence level that we will often use for the ease of notation. In practice, we will often not be able to fullfill the equality and in this case we generally prefer overcoverage, i.e. $\mathbb{P}(\mathbf{Y}_I \in C(\mathbf{X}_I)) \geq 1 - \alpha$.

This rather general definition has moreover been extended to methods where we aim to find some specific sets of outcomes and not just any kind of set that fullfills the miscoverage level. There have been many different methods proposed to achieve this [Sesia and Y. Romano, 2021; Chernozhukov, Wüthrich, and Zhu, 2021; Balasubramanian, Ho, and Vovk, 2014; Shafer and Vovk, 2008], a very popular method being quantile regression on the central 90% of density. There we aim to predict the invervals given by $[Q(\frac{\alpha}{2}), Q(1 - \frac{\alpha}{2})]$, where Q is the quantile function. However, in more recent works [Sesia and Y. Romano, 2021; Chernozhukov, Wüthrich, and Zhu, 2021] more advanced methods have been proposed that aim to estimate the shortest possible intervals for a given miscoverage level α. However, those approaches both are still aiming to predict single intervals which might not be desireable if the true distribution is multimodal. In particular, [Sesia and Y. Romano, 2021] argue that it is often not desirable to predict multiple intervals for the reason that they are harder to interpret for domain experts. In another work by [Izbicki, G. Shimizu, and Stern, 2022] they propose a method 'hpd-split' that can predict multiple intervals with the highest probability density, but they do not argue why this is actually desireable from a practical point of view.

In this work however, later in Section 2.6.1, we will give argumentations why it is actually desirable to predict multiple intervals in the case of multimodal distributions and how this can be achieved by means of CDE. Moreover, we show that even when predicting single intervals using our method, it is almost as powerful as previous methods and more interpretable. Before we dive into this, we will show in the next section how CDE and CP are fundamentally the same task that will allows us to gain a powerful perspective on CP, QR and CDE methods.

2.5 CP, CDE and QR are Essentially the Same Task

Here we argue that CP, CDE and QR are essentially the same task. Before going into details why this is the case, the motivation behind showing this result is mainly that it gives us a strong foundation on which we can use techniques used for one of the methods also directly for the other methods. In Section 2.7 we will based on that show that we can apply recalibration which is mainly used in CP also for QR and CDE.

Based on this motivation we now show that CDE is the most general of the three tasks and both CP and QR are sub-tasks of CDE. Sub-tasks in this context means that the two methods only model information also used for constructing CDE models and possibly less. For the connection between CDE and QR, we can argue that the QR predicts points on the conditional CDF, which can be fully described by the conditional PDF which is predicted by CDE. For the connection between QR and CP, we can argue that since CP regions need to capture, in expectation over $\mathbf{Z}_I$, a specific proportion of the target variable, the difference between high and low quantiles that produce the borders of the CP regions must in expectation sum up to the desired proportion. This means with a very dense QR grid we can find any CP regions of interest with asymptotic precision.

When we talk about high and low borders of CP regions we mean that any CP method produces for a given sample certain regions that can be described by the borders of the intervals within this region. For example, a region might be described by $[3.4, 5.1] \cup [7.2, 7.3]$ and for those borders there also exist quantiles that describe the borders of the regions, i.e. it could be $Q(0.05) = 3.4$ and $Q(0.5) = 5.1$ etc...

2.5.1 Model Producing Function

In the following we will develop a more formal argumentation for this statement. We argue that any function $g(\mathcal{D}) \in \mathcal{G}$ that can consistently produce models that can accurately do any of the three tasks all need to model the same true probability density function $p(\mathbf{x}, \mathbf{y})$. Here $\mathcal{G}$ is the space of all algorithms that produce models for one of the three tasks and g is the model that is produced. In particular, this formalism is required since the model itself, after it is trained, does not really make any assumptions about the problem.[1]

The task that produces a model is specified often in an optimization problem that is solved to find the model parameters i.e. gradient descent in NNs.

First we will show the theoretical bridge between $g(\mathcal{D}) \in \mathcal{G}$ that can produce any of the three model types. Thereafter we will discuss important practical implications of those insights.

2.5.2 Theoretical Bridge: CDE and CP

By definition, CP is any method that aims to predict regions, that in expectation should contain the true label with a significance level α. Formally, $\mathbb{P}(\mathbf{Y} \in U(\mathbf{X})) = 1 - \alpha$. It is essential to notice here that the PDF is explicitly part of the definition of any CP method.

The evidence for the claim that we can provide here is based on the No Free Lunch theorem [David H Wolpert and Macready, 1997]. The No Free Lunch theorem states that for any learning algorithm to perform well on a broad class of problems, it must necessarily make some implicit or explicit assumptions about the nature of those problems. This theorem tells us that for any algorithm g to work well for CP it must contain some assumptions about the problem. The limitation of the NFL are that we can not directly say that it must be the PDF.

However, it is a reasonable assumption that the PDF or parts of it are part of those assumptions made via the NFL theorem since the CP problem is defined thru it and it would not need any other assumptions to fullfill the requirement. Moreover, it would seem unreasonable that a method that can fullfill the constraint could completely ignore

[1] In non-parametric models that do not need to be trained, one needs to realize that the model generating function lies in the researchers who constructed the model and this needs to adher to the same principles.

the structure of the PDF, since it would basically be random then. It was not possible to find an existing work that provides a general theorem that lets us make statements about that if a model is defined thru a certain property, like the PDF, then it must be biased with this information and it is out of the scope of this work to proof such a general statement, so we will leave it as a very reasonable assumption and for a rather technical mathematical future work to prove this statement.

Morover, knowing the PDF, or parts of it, is not only the requirement for the CP method to make accurate statements about the intervals, but actually knowing the PDF is already enough to make the prediction and in particular any other information would not be useful at all. This can be seen since a PDF contains all information about possible CP regions and densities theirein. As CDE fully models this information, we can conclude that CP is a sub-task of CDE and as practical implication that we can infer any CP intervals from a CDE model.

This does not mean that any CP method existing really models this part of the information, but it means that any method that should generalize and make accurate statements about the intervals, must model this information. In particular one instance where we can see that CP fails to model this and basically ends up making nonsensical and arbitrary predictions is when we observe a quantile inversion, e.g. in the prediction it holds that $Q(\frac{\alpha}{2}) \geq Q(1-\frac{\alpha}{2})$. This is a clear sign that the CP method does not model the required portion of the true conditional density correctly and thus fails to make general and accurate statements about the intervals, hence violating the definition of CP itself.

2.5.3 Theoretical Bridge: CDE and QR

Analogous to CP we know that QR is defined as any method that can predict quantiles q of the target space where we have that $\mathbb{P}(\mathbf{Y} \leq Q(\mathbf{X})) = q$. Again, the PDF is explicitly part of the definition of any QR method and with the NFL and the argumentation in Section 2.5.2 it follows that QR is a sub-task of CDE and that any CDE method can infer any QR quantiles.

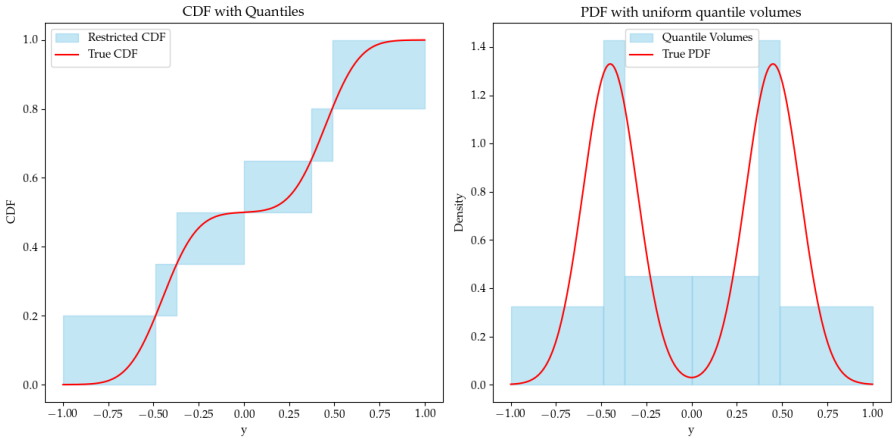
2.5.4 CDE can fully be modeled by CP and QR

We know from Section 2.5.2 and Section 2.5.3 a practical way to infer CP and QR from CDE. In order to rigoriously show that we can infer the CDE from CP and QR a novel concept is required. Moreover, this novel concept will also make it clearer how excatly CDE generally helps estimate CP and QR.

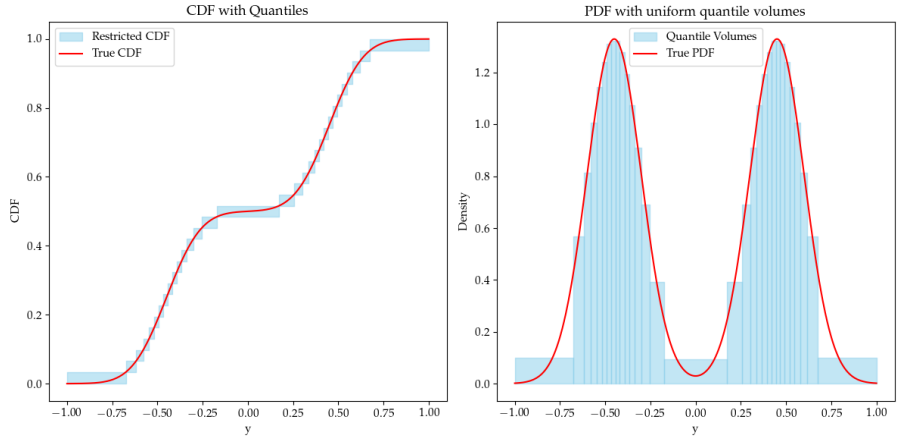
Therefore, let $\mathcal{P}$ be the set off all conditional PDFs, i.e. $\forall p \in \mathcal{P} : p \geq 0 \hat{\int}_{\Omega} p d\mathbb{P}(\omega) = 1$. Then we define $\mathcal{P}_\theta$ as a restriction on this set imposed by a CDE-method which is parameterized by θ. CDE-method refers to any method that imposes a restriction on the conditional PDF which contains CP, QR and CDE. The level of restriction can differ between CDE-methods. In particular, the restriction is always at least to the extent that the desired target type can be obtained uniquely from the restricted set of PDFs. For CDE only a single element is contained in this set and for QR all PDFs that have the integral up to a specific quantile of probability mass and for CP it is a set of PDFs that make it possible to infer that a specific region contains a certain amount of probability mass. In Figure 2.1a and Figure 2.1b we can see how the restriction on a CDF can look like for different quantiles.

We can now infer quite clearly what information is required in a $\mathcal{P}_\theta$ in order to do CP, QR or CDE. For QR, we only need to know that $\forall p \in \mathcal{P}_\theta : \int_{-\infty}^{Q(\mathbf{x})} p d\mathbf{y} = q$ for some q which can be also seen in Figure 2.1a. Even tho we do not know the full PDF when only looking at the restricted PDF we can still infer certain quantile levels precisely. For CP we need to know that $\forall p \in \mathcal{P}_\theta : \int_{U(\mathbf{x})} p d\mathbf{y} = 1 - \alpha$ for some α. In Figure 2.1a we can see that this could simply be an interval between two quantile levels that we predict. For CDE we need to know that $\forall p \in \mathcal{P}_\theta : p = p(\mathbf{y} \mid \mathbf{x}; \theta)$. Since for CDE $|\mathcal{P}_\theta| = 1$ it is quite clear that there is no ambiguity in predicting qunatiles or CP regions and we can just choose any quantile or region from the single PDF in $\mathcal{P}_\theta$.

However, in order to be able to obtain $\mathcal{P}_\theta$ such that it is valid for CDE via CP or QR we need to be able to restrict it to be a single element. Therefore, we need to observe that under our standard assumptions we can just apply multiple QR or CP restrictions to $\mathcal{P}$ by performing multiple CDE-methods that only partially restrict $\mathcal{P}$:



(a) Restricted CDF with 5 quantiles



(b) Restricted CDF with 30 quantiles

Figure 2.1: We can see that if we increase the number of quantiles that we predict, the restriction on the CDF becomes more and more strict and we approach the true PDF on the right side.

$$\mathcal{P}_\theta = \bigcap_{i=1}^{n} \mathcal{P}_{\theta_i} \tag{2.4}$$

Where each $\mathcal{P}_{\theta_i}$ is a restriction on $\mathcal{P}$ that is valid for CDE, CP or QR. Thereby one needs to be careful in the definition of each restriction not to have two restrictions that contradict each other and thus produce the empty set, however this is generally approximately possible in practice. What we mean by approximately possible is that we usually can construct a valid PDF from two restrictions that might contradict each other. For example if we do multiple quantile regression and observe quantile inversion, then it is a common practice to simply swap the two quantiles out. In this case in is common practice to just take this then as the final $\mathcal{P}_\theta$.[2] Moreover, combining restrictions of $\mathcal{P}_\theta$ is already a common method in the literature, e.g. [Sesia and Y. Romano, 2021] where multiple quantile regression is used in order to obtain a grid of quantiles that can be used to infer CP regions.

We can arbitrarily restrict $\mathcal{P}_\theta$ if we can make those restrictions arbitrarily tight with a method as we can also see in Figure 2.1b. In the case of QR we can simply make an infinitely dense quantile grid which essentially accomplishes that. For CP it is a bit less obvious since we could define CP intervals anywhere and it is unclear how we would make sure that we still restrict everywthing even if we increase/decrease the confidence level. However, all CP methods in the literature (e.g. [Sesia and Y. Romano, 2021; Chernozhukov, Wüthrich, and Zhu, 2021]) are based on methods that allow for a nested way of increasing/descreasing CP regions which relates to how calibration works in CP as we can see in Section 2.7. Nested regions essentially allow that we can make as specific statements about the conditional PDF by interpolating the CP regions.

2.5.5 CDE to improve current CP

In this work we argue, that by first estimating the true conditional density with high precision instead of just restricting it partially we can improve the performance of CP and QR. In particular, by restricting $\mathcal{P}$ more we argue that this provides a regulatory effect

[2] This common practice, while it results in a valid $\mathcal{P}_\theta$ does generally and also in literature lack a theoretical foundation an is more a "trick of the trade". We will not analyze the theoretical validity of doing this.

where unreasonable partial predictions are purged. This is conceptually similar to the concept of ensembles if we do this by means of multiple quantile regression.

For example, if we were to predict a quantile at a specific position that might not be perfectly accurate, then by having a tight quantile regression grid over the whole space $(0, 1)$ we will stabilize this since it is likely in that case that quantile inversion appears and by swapping/sorting the quantiles we can obtain a valid PDF. The same applies to CP.

This implicitly also means that given that the CP/QR method only needs to model a subset of the information of the PDF, it will not worsen the performance if we also try to model other parts of the PDF $\mathcal{P}_{\theta'}$ and use it as regularizer in practice. I.e. if we only want the median then it would not hurt the performance in practice if we also predict the 0.1 and 0.9 quantiles and use them as regularizer.

2.5.6 CP methods are practically not distribution free methods

In the literature it is often argued for the benefit of CP that it is distribution free which means that we do not make any distributional assumptions prior to defining the model. Moreover, if we were to infer CP by means of first estimating a CDE e.g. with a MDN then we on first glance directly make distributional assumptions since a MDN is based on distributional components.

However, even if we introduce distributional assumptions, in fact, it can be shown that when using MDNs for CDE with an infinite number of gaussian components, we can predict any arbitrary PDF and also, in fact, CP methods, when limited to a finite number of model parameters as it is always the case in practice, similarly have distributional assumptions introduced by the modeling limitations of NNs. Those modeling limitations of neural networks are present in the initialization of weights with a certain distribution and the distribution of activations after linearities and activation functions. Each activation function imposes a certain distribution on the model. So both, CP and CDE, are not distribution free methods when limiting the number of model parameters dispite common literature suggesting CP is distribution free. It is generally not true and CP is in the same way distribution in the context of those literature works as CDE is.

2.5.7 Limitations of the Bridge between CDE-Methods

While the statements made about CDE, CP and QR being the same task, they all only directly apply under our standard assumptions. However, even without the standard assumptions, without proof, we reasonably suspect that the same statements basically hold. In those cases it might be that the conditional CDF has discontinuities or flat spots for which of course all three methods will struggle. For example QR might behave strangely since we have a jump in the CDF at this point or on points between the quantile level QR is trying to predict. However, since we can approximate any PDF that does not fullfill our standard assumptions with a PDF that does, we can still argue that the same statements hold approximately and in practice anyways.

Moreover, when combining restrictions on the PDF, current methods, like swapping quantiles with quantiles inversion, to combine them while maintaining a valid PDF are not well studied and also impose assumptions on the PDF which are often quite arbitrary and of practical nature. However, as all CDE-methods are asymptotically consistent, cases where the restrictions contradict are an artifact of limited data and there approximate solutions are used anyways making it not really a limitation. However, it would be good to have a more theoretical foundation for those methods.

2.5.8 Conclusion on the Bridge between CDE, CP and QR

By Section 2.5.3 and Section 2.5.3 we can see that in the background of each CDE-method the restriction on all PDFs $\mathcal{P}_\theta$ stands. This has two major implications:

Firstly, any technique on a CDE-method also implicitly acts on $\mathcal{P}_\theta$ and thus can be thru $\mathcal{P}_\theta$ translated to any other CDE-method. For example if we were to decide that the std of a CDE model is to be inceased a bit by a particular function, e.g. $f(p) = 2p$, then this will clearly be reflected in the restricted distribution $\mathcal{P}_\theta$ which means we also could find the generalized form of f which explicitly acts on $\mathcal{P}_\theta$ and thus can be applied to any CDE-method. In this case it would be $f(\mathcal{P}_\theta) = \left\{2p : p \in \mathcal{P} \wedge \frac{p}{2} \in \mathcal{P}_\theta\right\}$, i.e. we just apply the f to each element in $\mathcal{P}_\theta$. So we see that any technique that acts on a CDE-method can be translated to any other CDE-method.

Secondly, any CDE-method goal can be done with any other CDE-method e.g. we can get a full conditional PDF from a method that does QR or CP.

2.6 Optimal Conformal Prediction

In this section we first define what we mean by optimal CP and then show how we can infer optimal CP from CDE by using a novel method. We will give a new perspective on the optimization objective of CDE, which often is likelihood, and show how it relates to the definition of optimal conformal prediction that we give here. This relationship is of theoretical interest and also has practical implications as we will show in Section 2.7.

In order to develop our argument we first define what we mean when we say that a CP method is optimal. We orient ourselves on the work of [Sesia and Y. Romano, 2021] where they define the optimal CP method as the one that predicts the shortest intervals for a given miscoverage level α. However, instead of only predicing single intervals, we argue that it is simpler and more practical to simply estimate the shortest possible regions of the target space that contain the target variable in expectation with the required confidence level.

Let $\alpha \in (0,1)$ be a significance level. Then the goal of conformal prediction is to find a function $C : \mathbb{R}^n \to \mathcal{B}(\mathbb{R}^m)$ that can predict the subsets with the smallest Lebesgue measure λ marginalized over Ω with significance α. That means we want $\mathbb{P}(\mathbf{Y}_I \in C(\mathbf{X}_I)) = 1 - \alpha$ with $\int_\Omega \lambda(U(\mathbf{X}_{I(\omega)}(\omega)))d\mathbb{P}(\omega)$ small.

Formulated as a proper constrained optimization problem, we can rewrite the optimization problem as:

$$\min_U \quad \int_{\mathbb{R}^{m+n}} p(\mathbf{z})\lambda(C(\mathbf{x}))d\mathbf{z} \tag{2.5}$$

$$\text{s.t.} \quad \int_{\mathbb{R}^{m+n}} \mathbb{1}_{\mathbf{y}\in C(\mathbf{x})}d\mathbf{z} = a \tag{2.6}$$

2.6.1 New perspective on MLL and Optimal CP

The method to infer the intervals that will be used by us is the highest density regions method. Using this method we can, given a PDF, infer the set of intervals with the shortest summed Lebesgue measure. In particular, we are shifting our focus from looking at the shortest regions to looking at the regions that contain the most probability mass, even tho that is very similar and mostly the same with HDR there are some delicate differences. This is partiularly beneficial since regions with high densities should usually not be ignored in practice as they often indecate important events. When only focusing on the shortest possible single interval that contains the desired proportion of probability mass, it might happen that a region with high density is ignored even tho this event might have some core interpretation to domain experts. We argue, that if a single interval is desired it is more beneficial in most domains, to simply take the interval from the small edge of the region of the smallest value to the large edge of the largest value region. This might contain more density than $1 - \alpha$ but we argue that it is in most cases still desirable to design intervals this way.

If C is being calculated by using first Conditional Density Estimation (CDE) and then using Highest Density Regions (HDRs) as defined by [Hyndman, 1996] to obtain a significance level of α, then C is a function of the CDE method and the significance level α. HDR is by [Hyndman, 1996] defined as:

$$H(f_a) = \{\mathbf{y} : f(\mathbf{y}) \geq f_a\}$$

with

$$f_a = \max_{f_a} \{f_a \in \mathbb{R}^+ : \mathbb{P}(\mathbf{y} \in H(f_a)) \geq a\}$$

but it can be written equivalently as below. The below formulation is also the one being used in this work from now on.

$$H(f, a) := \left\{\mathbf{y} \in \mathbb{R}^m : f(\mathbf{y}) \geq \max_b \{b \in \mathbb{R}^+ : \mathbb{P}(\{\hat{\mathbf{y}} \in \mathbb{R}^m : f(\hat{\mathbf{y}}) \geq b\}) \geq a\}\right\} \quad (2.7)$$

where f is an arbitrary probability density function (PDF) and $a := 1 - \alpha$ is the confidence level. Note that $\mathbb{P}$ here is different from the one defined in the beginning and only here for defining HDRs. As the CDE method in my case is parametric like mixture density networks it is more reasonable to write C a function of the weights and the significance (and implicitly also the architecture of the CDE of course). So considering that, the initial goal of conformal prediction can be rewritten as:

$$\min_{\theta \in \Theta} \quad \int_{\mathbb{R}^{m+n}} p(\mathbf{z}) \lambda(H(p(\hat{\mathbf{y}} \mid \mathbf{x}; \theta), a)) d\mathbf{z} \tag{2.8}$$

$$\text{s.t.} \quad \int_{\mathbb{R}^{m+n}} \mathbb{1}_{\mathbf{y} \in H(p(\hat{\mathbf{y}}|\mathbf{x};\theta),a)} d\mathbf{z} = a \tag{2.9}$$

where it is important that the $\hat{\mathbf{y}}$ is not the one we integrate over but more a demonstrative artefact we write to denote that $p(\hat{\mathbf{y}} \mid \mathbf{x}; \theta)$ is a conditional density. Moreover Θ is the space of all parameters of the CDE method.

In the following if we write $p(\hat{\mathbf{y}} \mid \mathbf{x}; \theta)$ we mean a conditional density estimator parameterized with θ which implicitly contains the architecture and the weights of the method and if we write $p(\hat{\mathbf{y}} \mid \mathbf{x})$, the true conditional density is meant. Notice, that the goal of this optimization problem is to optimize w.r.t. θ as it is basically the component that completely defines $p(\hat{\mathbf{y}} \mid \mathbf{x}; \theta)$. This in turn means we need to find the argmin of the optimization above.

Let θ^* be the argmin of the equation above, we would like to show that

$$\theta^* = \arg\max_{\theta \in \Theta} \int_{\mathbb{R}^{m+n}} p(\mathbf{z}) \log p(\mathbf{y} \mid \mathbf{x}; \theta) d\mathbf{z} \tag{2.10}$$

which is the maximum likelihood estimator (MLE). This is of great importance if we wish to optimize a CDE method w.r.t. the maximum likelihood objective function in order to implicitly optimize for the conformal prediction objective function when using HDRs. In order to show this powerful statement we first need to develop some insight into the workings of the components in conformal prediction.

First, we require to show that $\lambda(H(p,a))$ is continously differentiable a.e. in order to make sensible statements. Its not that the intuition does not hold if its not continous, however, it complicates things and makes the statemtent more difficult. First in Lemma 2.6.1 below we show that this is fullfilled if the PDF fullfills our standard assumptions in Assumptions 2.1.1.

Note that the second assumption is fullfilled if it holds that p is changing a.e. or the area with derivative 0 is of measure 0 which is left without proof.

Lemma 2.6.1. *Let $p : \mathbb{R}^m \to \mathbb{R}$ be a probability density function for which Assumptions 2.1.1 hold where we neglect the $\mathbf{x}$ for breefity.*

Furthermore let p be for the random variable $\mathbf{Y}$ (for breefity we neglect the index) and let $g(b) := \mathbb{P}(p(\mathbf{Y}) \geq b)$. Moreover, $\lambda_p : [0,1] \to \mathbb{R}$ with $\lambda_p(a) = \lambda(H(p,a))$. Then the following statements hold for $a \in (0,1)$:

1. *$g(b)$ is strictly monotonic on the set $g^{-1}((0,1))$.*
2. *$g(b)$ is continous.*
3. *$g(b)$ is bijective on the set $g^{-1}((0,1))$.*
4. *$g(b)$ is continously differentiable a.e.*
5. *$B(p,a)$ is strictly monotonous on the set $B^{-1}((0,1))$.*
6. *$B(p,a)$ is continous.*
7. *$B(p,a)$ is bijecitve on the set $B^{-1}((0,1))$.*
8. *$B(p,a)$ is continously differentiable a.e.*
9. *$\lambda_p(a)$ is strictly monotonous.*
10. *$\lambda_p(a)$ is continous.*
11. *$\lambda_p(a)$ is bijective on the set $\lambda_p^{-1}((0,\infty))$.*
12. *$\lambda_p(a)$ is continously differentiable a.e.*

It is in particular required to not have any flat-splots in the PDF because otherwise per definition of HDR we can sometimes not obtain the shortest possible intervals for a confidence level a since we would not know which part of the flat-spot to include in the interval and which to leave out. In practice this would hardly be a problem.

Proof. 1. Monotonicity itself is obvious since we can not make the set $\mathbf{Y} \geq b$ larger when increasing b. For strict monotonicity on $g^{-1}((0,1))$ we require that $\forall b \in g^{-1}((0,1)) : \frac{\partial \mathbb{P}(p(\mathbf{Y}) \geq b)}{\partial b} > 0$.

Let b be in $g^{-1}((0,1))$. We essentially need to show that for any $b' > b$ there exists a $\epsilon > 0$ such that $\mathbb{P}(p(\mathbf{Y}) \geq b + \epsilon) = \epsilon + \mathbb{P}(p(\mathbf{Y}) \geq b')$ which essentially means that there is change in g no matter how tight we choose the interval which comes directly from the definition of the derivative. So let b' be like so, and choose any $b'' \in (b, b')$, then continuity of p implies there exists a dense neighborhood around b'' which lies in $p(\mathbf{Y}) \in (b, b')$. In particular the fact that this neighborhood is dense also implies via the Lebesgue Density Theroem that this neighborhood has a positive Lebesuge measure and the assumption that $p > 0$ implies that it does also have positive probability measure. This implies the result of strict monotonicity.

2. To show continuity of g we use that it is because of monotonicity and boundedness of the $\mathbb{P}$ measure, that for any $\tilde{b} \in \mathbb{R}^+$

$$\lim_{b \downarrow \tilde{b}} g(b) = \mathbf{P}(\mathbf{Y} > \tilde{b}) \tag{2.11}$$

and also

$$\lim_{b \uparrow \tilde{b}} g(b) = \mathbf{P}(\mathbf{Y} \geq \tilde{b}) \tag{2.12}$$

where implicitly the continuity of p from above and below are used. This means the two limits are always the same if $\mathbb{P}\mathbf{Y} = \tilde{b} = 0$ which is an assumption and thus continuity of g is shown.

3. Bijectivity follows from strict monotonicity directly. To be exact why actually $g^{-1}((0,1))$ exists for every $a \in (0,1)$ we can use the monotonicty of g and the intermediate value therem since we can clearly find a b such that $g(b)$ is arbitrarily close to 0 and to 1 because of the PDF property of p and the fact that continuity of p implies boundedness of p.

4. Differntiability a.e. follows from the Lebesgue's Theorem for Monotonic Function as g is monotonic. Continous differentiability a.e. follows from the fact that g is uniformly continous due to the Heine-Cantor Theorem and this implies that the derivative is continous a.e. as well.

5. This follows by observing that g is bijective on $g^{-1}((0,1))$ which implies that the maximum of b where g is still greater-equal a will always exactly reach a. Then since g is $g^{-1}((0,1))$ is well defined and g is strictly monotonic we see that increasing a will always increase the possible b.

6. Continuity follows from the fact that g is also continous and bijective.

7. Bijectivity follows directly from 5. and 6. similar to 3.

8. Follows with the same logic as 4.

9. Strict monotonicity of B and the same argument as in 1. (Lebesgue density theorem) imply this.

10. This can also be shown by the same argument as 2. and continuity of B in 6.

11. Bijectivity follows from 9. The fact that the image is $(0,\infty)$ follows from the fact that $p > 0$ and thus in order to go with $a \to 1$ we will require infinite area in the Lebesgue sense. That we also approach 0 in the image follows easily from bijectivity of B on $(0,1)$.

12. Follows with the same logic as 4. □

From this lemma we can see that iff the PDF is continously diffentiable a.e., also the length of the HDR w.r.t. a will be continously differentiable a.e.. This is required because otherwise we can not show the following lemma about convexity of the HDR length w.r.t. a:

Lemma 2.6.2. *Let $p : \mathbb{R}^m \to \mathbb{R}$ fullfill the Assumptions 2.1.1. Then, the function $\lambda_{H_p}(a) := \lambda(H(p,a))$, representing the size of the highest density region (HDR) for PDF p with coverage a, is strictly convex, i.e.,*

$$\frac{\partial^2 \lambda_{H_p}(a)}{\partial a^2} > 0. \tag{2.13}$$

Proof. Without loss of generality, assume $a_1 < a_2$ with both $a_1, a_2 \in (0,1)$. For any $\alpha \in (0,1)$, let $a := \alpha a_1 + (1-\alpha)a_2$. By the definition of H_p in Equation 2.7, the set $H(p,a)$ encompasses points up to the highest densities corresponding to coverage a. Taken from the definition in Equation 2.7 we define this highest density as

$$B(p,a) := \max\left\{b \in \mathbb{R}^+ : \mathbb{P}\left(\{\hat{\mathbf{y}} \in \mathbb{R}^m : p(\hat{\mathbf{y}}) \geq b\}\right) \geq a\right\} \tag{2.14}$$

This implies that $\lambda_{H_p}(a_1) \leq \lambda_{H_p}(a) \leq \lambda_{H_p}(a_2)$. Define $k_1 = a - a_1$, $k_2 = a_2 - a$. Consequently, $H(p,a_1) \subset H(p,a)$, indicating that to transition from $H(p,a_1)$ to $H(p,a)$, only points with density less than $B(p,a_1)$ can be utilized. In contrast, if we go from a to a_2 only points with a density less than $B(p,a)$ can be utilized.

As we know that

$$\lambda_p(a) = \lambda(H(p,a)) = \lambda(H(p,a_1) \cup (H(p,a) \setminus H(p,a_1))) = \lambda(H(p,a_1)) + \lambda(H(p,a) \setminus H(p,a_1)) \tag{2.15}$$

and we know that $\lambda(H(p,a) \setminus H(p,a_1)) \leq k_1 \cdot B(p,a)$ and $\lambda(H(p,a_2) \setminus H(p,a)) \geq k_2 \cdot B(p,a)$ we can finally see that we can approximate the gradient by dividing the change of λ_p by the change of a, which is in k_1 and k_2 we see that the gradients are bounded from above and below respectively for the invervals (a_1, a) and (a, a_2). Moreover strict monotonicity of B as shown in Lemma 2.6.1 implies that the gradient is strictly increasing for λ_p which implies a postive second derivative and thus convexity.

In particular, the second derivative exists by the use of the Lebesgue's Theorem for Monotonic Function a.e. since the first derivative exists continuously a.e. and is strictly

monotonic as we know from Lemma 2.6.1 which implies absolute continous derivative a.e. and thus twice differentiability a.e.. □

Lemma 2.6.3. *For any PDFs $f : \mathbb{R}^m \to \mathbb{R}$ and $p : \mathbb{R}^m \to \mathbb{R}$, where p must fullfill Assumptions 2.1.1, with the same coverage size under confidence levels a_p and a_f, that is,*

$$\lambda(H(f, a_f)) = \lambda(H(p, a_p))$$

it holds that if we measure the coverage of those HDRs with points distributed w.r.t. p, that the coverage measured with $H(p, a)$ will be greater-equal. Moreover, the coverage level of $H(p, a)$ will be exactly a_p. Formally:

$$\int_{\mathbb{R}^m} p(\mathbf{y}) \mathbb{1}_{\mathbf{y} \in H(f, a_f)} d\mathbf{y} \leq \int_{\mathbb{R}^m} p(\mathbf{y}) \mathbb{1}_{\mathbf{y} \in H(p, a_p)} d\mathbf{y} = a_p$$

Note that it is absolutely possible that $a_p = a_f$.

Proof. We split $H(p, a_p)$ and $H(f, a_f)$ into subsets: $H(f, a_f) = A \cup B$ and $H(p, a_p) = A \cup C$ with $H(f, a_f) \cap H(p, a_p) = A$, $H(f, a_f) \setminus A = B$ and $H(p, a_p) \setminus A = C$.

$$\int_{\mathbb{R}^m} p(\mathbf{y}) \mathbb{1}_{\mathbf{y} \in H(f, a_f)} d\mathbf{y} = \int_{H(f, a_f)} p(\mathbf{y}) d\mathbf{y} = \int_A p(\mathbf{y}) d\mathbf{y} + \int_B p(\mathbf{y}) d\mathbf{y} \tag{2.16}$$

$$\int_{\mathbb{R}^m} p(\mathbf{y}) \mathbb{1}_{\mathbf{y} \in H(p, a_p)} d\mathbf{y} = \int_{H(p, a_p)} p(\mathbf{y}) d\mathbf{y} = \int_A p(\mathbf{y}) d\mathbf{y} + \int_C p(\mathbf{y}) d\mathbf{y} \tag{2.17}$$

since the A part of the integrals is equal we can ignore it for comparing $H(f, a_f)$ and $H(p, a_p)$ coverage. So, we need to show:

$$\int_B p(\mathbf{y})d\mathbf{y} \leq \int_C p(\mathbf{y})d\mathbf{y} \tag{2.18}$$

This can be shown by proofing $\forall \mathbf{y}_B \in B \forall \mathbf{y}_C \in C : p(\mathbf{y}_B) \leq p(\mathbf{y}_C)$ because $\lambda(B) = \lambda(C)$. So, let $\mathbf{y}_B$ and $\mathbf{y}_C$ be arbitrary from the corresponding sets. Then we know that $\mathbf{y}_B \in H(p, a_p)$, which means that $p(\mathbf{y}_C) \geq B(p, a_p)$ where $B(p, a_p)$ is the maximum density bound such that the coverage level of a_p is still given w.r.t. p. However, since $\mathbf{y}_C \notin H(p, a_p)$ this means that $p(\mathbf{y}_C) < B(p, a_p)$, which shows the first part of the proof.

The second part of the proof is to show that

$$\int_{\mathbb{R}^m} p(\mathbf{y}) \mathbb{1}_{\mathbf{y} \in H(p, a_p)} d\mathbf{y} = a_p \tag{2.19}$$

which follows from the fact that with our Assumption 2.1.1 $B(p, a_p)$ is bijective and thus $\mathbb{P}(\mathbf{Y} \geq B(p, a_p)) = a_p$. In particular, $H(p, a_p)$ is per construction a set where this is fullfilled. □

In order to finally be able to use this lemma efficiently we need to define something like an inverse of the HDR w.r.t. the significance, which based on the input $\mathbf{x}$ and the length $\lambda(H(p(\mathbf{y} \mid \mathbf{y}), a))$ gives us the significance level that we would need to insert in H together with the true distribution at x to obtain the same size of the distribution but with the above lemma always has a larger or equal coverage.

Definition 2.6.1 (HDR Transform)**.** Let $f, p : \mathbb{R}^m \to \mathbb{R}$ be two probability density functions and let $a \in (0, 1)$ be some coverage level. Then we define the HDR Transform as:

$$H_p(f, a) = \underset{b \in (0,1)}{\arg\max}\, \lambda(H(p, b)) \leq \lambda(H(f, a))$$

In words, the HDR Transform gives us the maximal significance level that we can insert under the distribution p while maintaining a coverage size lower-equal than what we obtain by inserting the significance level a under the distribution f.

In particular, if we evaluate the actual coverage of $H(f,a)$ with samples drawn from p, then the coverage will be always lower-equal a which follows from Lemma 2.6.3. The HDR transform then gives us basically the confidence level that we need to insert into the HDR with the true distribution to obtain the same coverage size but are guranteed to have higher-equal coverage.

Using this definition together with Lemma 2.6.3 establishes a very powerful tool to make proofs related to HDR.

Lemma 2.6.4. *With f, p as in Definition 2.6.1 and Assumptions 2.1.1 on p, it always holds that:*

$$H_p(f,a) = \arg\max_{b\in(0,1)} \lambda(H(p,b)) \leq \lambda(H(f,a)) = \arg\max_{b\in(0,1)} \lambda(H(p,b)) = \lambda(H(f,a)) \quad (2.20)$$

Proof. Bijectivity of $\lambda_p(a)$, under the assumptions, with Lemma 2.6.1 implies that exactly one b exists such that $\lambda(H(p,b)) = \lambda(H(f,a))$ which finishes the proof. □

Now we have all tools in order to proof the important statement. From now on, if the expectation $\mathbb{E}$ is used, it is always w.r.t. the whole space $\mathbb{R}^{m+n}$ and with the random variables $\mathbf{Y}$ and $\mathbf{X}$ which have PDF $p(\mathbf{y},\mathbf{x})$. Moverover, if $\hat{\mathbf{y}}$ is written, it is not an input to the function but a demonstrative artefact to show that the function maps to a conditional density on the same space as $\mathbf{Y}$ is defined.

Theorem 2.6.5 (MLL is equivalent with Optimal Conformal Prediction)**.** *We want to show that if we have definitions as in Section 2.4, the space of all parameters Θ and $\forall \mathbf{x} \in \mathbb{R}^n : p(\mathbf{y} \mid \mathbf{x})$ fullfills Assumptions 2.1.1 and that $\forall \mathbf{x} \in \mathbb{R}^n : \max_\theta p(\mathbf{y} \mid \mathbf{x};\theta) = p(\mathbf{y} \mid \mathbf{x})$, then it holds that:*

$$\arg\max_{\theta\in\Theta} \mathbb{E}\left[\log p(\mathbf{Y} \mid \mathbf{X};\theta)\right] \quad (2.21)$$

equals

$$\arg\min_{\theta\in\Theta} \mathbb{E}\left[\lambda(H(p(\hat{\mathbf{y}} \mid \mathbf{X};\theta), a))\right] \tag{2.22}$$

$$s.t. \quad \mathbb{E}\left[\mathbb{1}_{\mathbf{Y}\in H(p(\hat{\mathbf{y}}|\mathbf{X};\theta),a)}\right] = a \tag{2.23}$$

Proof. For brevity, let $H_\theta := H(p(\hat{\mathbf{y}} \mid \mathbf{X};\theta), a)$ and $H_{\mathbf{\Theta}^*} := H(p(\hat{\mathbf{y}} \mid \mathbf{X};\theta^*), a)$.

In this case let's assume that now we have some $\theta \neq \theta^*$. We can show that if for this θ it holds that if

$$\mathbb{E}\left[\lambda(H_\theta)\right] < \left[\lambda(H_{\mathbf{\Theta}^*})\right] \tag{2.24}$$

it implies

$$\mathbb{E}\left[\mathbb{1}_{\mathbf{Y}\in H_\theta}\right] < a \tag{2.25}$$

which would finish the proof, since it'd show that for any parameter set θ, that produces a smaller average HDR than the the MLE, the constraint would be violated and it is clear that if using the MLE, which is the same as the true PDF, we would fulfill the constraint because of Lemma 2.6.3.

First, we can upper bound the coverage with the IH:

$$\mathbb{E}\left[\mathbb{1}_{\mathbf{Y}\in H_\theta}\right] \leq \mathbb{E}\left[\mathbb{1}_{\mathbf{Y}\in H(p(\hat{\mathbf{y}}|\mathbf{X}),IH(p(\hat{\mathbf{y}}|\mathbf{X};\theta),a))}\right] \tag{2.26}$$

$$= \mathbb{E}\left[IH(p(\hat{\mathbf{y}} \mid \mathbf{X};\theta), a)\right] \tag{2.27}$$

The upper bound follows directly from Lemma 2.6.3 and the monotonicity property of integrals. The equality in Equation 2.27 follows from the fact that if we evaluate the

coverage w.r.t. underlying distribution p we will always get the same coverage level as the one we inserted in the HDR if we fullfill Assumptions 2.1.1.

We know that:

$$\mathbb{E}\left[\lambda(H_{\mathbf{\Theta}^*})\right] > \mathbb{E}\left[\lambda(H_\theta)\right] \tag{2.28}$$

and

$$\mathbb{E}\left[\lambda(H_\theta)\right] = \mathbb{E}\left[\lambda(H(p(\hat{\mathbf{y}} \mid \mathbf{X}), IH(p(\hat{\mathbf{y}} \mid \mathbf{X}; \theta), a)))\right] \tag{2.29}$$

where the second equality follows from Lemma 2.6.4. This upper bound gives via Lemma 2.6.4 the highest coverage level for the same coverage size that is possible. If we can show the desired result for this upper bound, then we have shown the desired result.

Due to convexity and with the Jensen inequality we can now show that then we have appended to the equation in Equation 2.29 the following inequality:

$$\geq \mathbb{E}\left[\lambda(H(p(\hat{\mathbf{y}} \mid \mathbf{X}), \mathbb{E}\left[IH(p(\hat{\mathbf{y}} \mid \mathbf{X}; \theta), a)\right]))\right] \tag{2.30}$$

and thus:

$$\mathbb{E}\left[\lambda(H(p(\hat{\mathbf{y}} \mid \mathbf{X}), a)))\right] > \mathbb{E}\left[\lambda(H(p(\hat{\mathbf{y}} \mid \mathbf{X}), \mathbb{E}\left[IH(p(\hat{\mathbf{y}} \mid \mathbf{X}; \theta), a)\right]))\right] \tag{2.31}$$

In words, what this means is that given the same distribution $p(\hat{\mathbf{y}} \mid \mathbf{x})$, we obtain a strictly larger average coverage size when using coverage a, then when we use coverage $\mathbb{E}\left[IH(p(\hat{\mathbf{y}} \mid \mathbf{x}; \theta), a)\right]$. By monotonicity of the coverage size function, this means a.e. that

$$\lambda(H(p(\hat{\mathbf{y}} \mid \mathbf{X}), a))) > \lambda(H(p(\hat{\mathbf{y}} \mid \mathbf{X}), \mathbb{E}\left[IH(p(\hat{\mathbf{y}} \mid \mathbf{X}; \theta), a)\right])) \tag{2.32}$$

since it can a.e. not be that the inequality is reversed, and thus:

$$a > \mathbb{E}\left[IH(p(\hat{\mathbf{y}} \mid \mathbf{X}; \theta), a)\right] \tag{2.33}$$

which, as we can see in Equation 2.26, upper bounds the coverage of the actual $p(\hat{\mathbf{y}} \mid \mathbf{X}; \theta)$. This completes the proof.

□

This now shows our desired result, that in fact, by maximizing the likelihood of a CDE model to afterwards apply HDR to obtain small but calibrated regions is equivalent to directly trying to minimize the size of the HDR while maintaining calibration. When assuming that our CDE models can reasonably well approximate the true underlying distribution this is a powerful result that allows us to focus fully on optimizing the likelihood for CDE and we get optimal conformal prediction with HDR for free.

Moreover, we can see that this means the MLE can be decomposed into a constrained optimization problem. In particular this is true because not only the MLE is the ideal model for optimal CP, but also the inverse is true, the optimal CP model is also the optimal MLE model which follows directly from the equality that was just shown in Theorem 2.6.5. While in the context of this work we leave it as a theoretical insight, it is possible that this can be used in future work for new relevant findings. Arguably this fact has some mathematical beauty to it, as we realize the MLE inherently is looking for the tightest possible peaks while maintaining calibration.

2.6.2 Focusing on Density instead of the Coverage Level

AS hinted in Section 1.2 in this work we argue that it is practically more reasonable to move the focus to the probability mass/density instead of only the length of the inverval while maintaining calibration. The difference is subtle, but via Figure 2.2 we hope to give some intuitive understanding of the difference with two different arguments.

Firstly, we are when doing CP only focusing on the length of the single inverval being as short as possible in the prediction while maintaining calibration entriely argnostic to the actual shape of the distribution. In particular, if it were possible to obtain significantly

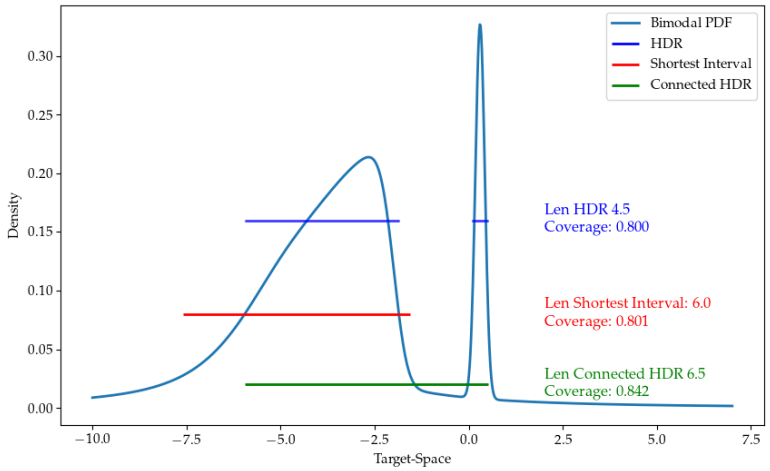


Figure 2.2: Comparison of HDR, connected HDR and Shortest Interval CP for a bimodal distribution. We can see that although the connected HDR slightly overcoveres, we obtain significantly more coverage with only a slightly larger interval and also intutively this interval is more meaningful.

more mass with just a very slight increase the interval size it would be ignored. In the figure this can be observed when comparing the connected HDR with the shortest interval CP. We can see that when we first use HDR but then simply connect it, with the argumentation of [Sesia and Y. Romano, 2021] (multiple intervals can be confusing in practice), we will capture the second peak of the distribution while we do not capture low density parts of the first bigger peak. This is a clear example of where focusing on the length of the interval can be misleading.

Secondly, another intutive reason, even tho we lack empirical evidence of this, why a second predicted peak should be included in the CP prediction is that it appears reasonable to us that if the model learned a second spiky modality at a specific point than there must be a good reason therefore whilst a heavy tail might simply be an artifact of the modeling method. We are looking for further research to confirm this hypothesis.

Primarely for the first but also for the second reason, we believe that when using our approach (first HDR then connect regions) that it is of advantage in practical scenarios like healthcare and fincance.

2.7 Calibration and Recalibration

CDE-methods, while calibrated when optimal, in practice are often not calibrated due to model misspecification, overfitting or underfitting. Calibration if the model is optimal directly follows from the fact that under our assumption the optimal model is the true PDF and the true PDF is calibrated of course. For this section of our work we assume a limited training set $\mathcal{Z} = (\mathcal{X}, \mathcal{Y})$ that is used to train the CDE-method model. The probably most important realization on this is, that even tho the objective functions CDE-methods use in practice are optimal in theory, in practice we mostly need to resort to gradient based optimization methods, which require a step wise optimization and will only find local optima. Especially this insight is important because that means, even tho globally optimal models w.r.t. objective functions of CDE-methods will be calibrated, those objective functions do not gurantee calibration at all times during optimization. The reason is that the constraint of those optimization problems is an implicit one in practice and we will violate it usually with CP.

This is the reason why in practice recalibration methods exist and are at this time mostly applied to CP methods. From Section 2.5 it follows that we can apply the same methods to all CDE-methods that are used currently only for CP. However, since recalibration methods are mostly explored in CP methods, we will first introduce them in the context of CP in Section 2.7.1 and step-wise extend the methods till we reach full generalization of recalibration in all CDE-methods in Section 2.7.4.

2.7.1 Calibration in CP

Generally, calibration of CP refers to the requirement of the defining property of CP to be fullfilled as described in Section 2.4. In particular, since today most CP methods don't aim to estimate any CP intervals but specific ones, during optimization the calibration requirement is often overshadowed. However, calibration in the context of CP is basically the whole point of CP to begin with and because it is in practice often not fullfilled, recalibration methods are employed.

In any case, if doing recalibration we have that the estimated conformal intervals do in expectation not capture the desired $1 - \alpha$ proportion of the target in the calibration

set. However, this can be tackled by recalibration for which a large possible number of methods exist, each depending on the method used to estimate the CP intervals.

One can view the marginal calibration itself as a optimization problem where the objective function is defined as:

$$\min |(1-\alpha) - \mathbb{P}(\mathbf{Y} \in C(\mathbf{X}, \psi))| \tag{2.34}$$

, where C is a map to the CP regions. $\psi \in \Psi$ is a configuration of the method used that can be optimized w.r.t. objective function and can be quite arbitrary. In particular, ψ is not generally the same the parameters of the CDE-method model that has been estimated but rather it is something that we apply after the CDE-method model has been estimated. We usually assume that we already have learned a CDE-method model estimated with parameters θ but that it is possibly not calibrated. One can better understand this by realizing that Section 2.5 implies that any CDE-method can be used to estimate CP intervals and every CDE-method implicitly can fully model the true PDF in the sense that we can restrict $\mathcal{P}_\theta$ arbitrarily close to the true PDF. In particular, we can with that see that we need to reformulate Equation 2.34 to:

$$\min |(1-\alpha) - \mathbb{P}(\mathbf{Y} \in C(\mathcal{P}_\theta, \psi))| \tag{2.35}$$

if we want to be perfectly rigorous since all CP methods necissarly act on a restricted subset of all possible PDFs as discussed in Section 2.5. Now we also observe that C can be very arbitrary, very much dependent on the exact form of $\mathcal{P}_\theta$. For example if from $\mathcal{P}_\theta$ we only know a single quantile in terms of restriction, then we can only define a CP interval between $-\infty$ and this quantile and Ψ is essentially the empty set, i.e. C is not parametric in this case. We can see, that without of further assumptions we will not come very far. In particlar, we have no way of minimizing Equation 2.35 without further assumptions since the only thing we know about $\mathcal{P}_\theta$ is a single quantile and this if we are not calibrated this quantile appears to be wrong. Further, without more assumptions we can not simply change $\mathcal{P}_\theta$ as this is the all we know about the true conditional PDF. Moreover, even if the CDE-method used is very restrictive in terms of $\mathcal{P}_\theta$, for example $|\mathcal{P}_\theta| = 1$, then we still can not simply change $\mathcal{P}_\theta$ if we are not calibrated since the fact that we are not calibrated indicates that the learned $\mathcal{P}_\theta$ is invalid. The question arises what gives us the right to

simply manipulate $\mathcal{P}_\theta$, which we are already doing in practice as an abundance of current literature shows [].

What appears to be a reasonable assumption is that while $\mathcal{P}_\theta$ might be a restriction that is not correct, we might want to assume that it is close to what is correct; Possibly that the KL-divergence is low between each element of $\mathcal{P}_\theta$ and the true conditional PDF, that is lower than for a randomly choosen PDF. The reason why this closeness is reasonably to assume is that all CDE-methods implicitly gurantee asymptotic optimality in the very definitions of their objective functions.

n any case, current literature provides methods that, based on the estimated $\mathcal{P}_\theta$, lets us estimate calibrated CP intervals which are close in some sense to the uncalibrated intervals, i.e. the calibrated $\mathcal{P}_\theta$ is close to the uncalibrated. Before we go into the details in Section 2.7.2 and later sections what it means for a calibrated $\mathcal{P}_\theta$ to be close or how this can be interpreted in the whole scale of the restricted set of all conditional PDFs $\mathcal{P}_\theta$, we will first introduce the methods that are used in practice to recalibrate CP intervals.

In the following we will describe a variation of C and θ where we simplify the optimization problem in Equation 2.35 to one that we can practically easily optimize. The method we focus on here is heavily inspired by [Sesia and Y. Romano, 2021] and can be used if C and θ fullfill certain properties.

1. $U(,\psi)$ must be such that $(U(\cdot,\psi_r))_{r\in\mathbf{R}\subsetneq\mathbb{R}}$ where $\mathbf{R}$ is bounded and $(U(\cdot,\psi_r))$ are strictly nested sets on the target space where the smallest set $(U(,\psi_0)) = \emptyset$ is the empty set and the largest one contains the full target space. I.e. we have $\forall r_1, r_2 \in \mathbf{R} : r_1 < r_2 \implies U(\cdot,\psi_{r_1}) \subset U(\cdot,\psi_{r_2})$.
2. We require that the sequence is continous in a way if we want to be able to gurantee that we can come arbitrarily close the the desired calibration in expectation. I.e. $\forall a \in (0,1) \exists \psi \in \Psi : a = \mathbb{P}(\mathbf{Y} \in C(\mathcal{P}_\theta, \psi))$

If condition 2 in 2.7.1 is not fullfilled we can only gurantee that the calibration in expectation will be larger-equal the desired proportion. When condition one is not fullfilled, we can not gurantee anything about the calibration in expectation, not even that it will be larger-equal the desired proportion. This is because there might be sets that are never part of the sequence but contain probability mass and thus we can never gurantee that we come even close to the desired calibration in expectation there.

If we optimize Equation 2.35 on the calibration set we can gurantee that the model is calibrated in expectation on the calibration set when assuming exchangability. Moreover, if we do not have set-continous nested sets, we can still gurantee in expectation that the CP interval has a larger-equal converage than the desired confidence level by simply taking r such that $(1-\alpha) - \mathbb{P}(\mathbf{Y} \in U(\cdot, \psi))$ is still negative and minimal in absolute value.

The method used by [Sesia and Y. Romano, 2021; Chernozhukov, Wüthrich, and Zhu, 2021] is to basically sort the r that are required for each of the calibration samples to be included in the CP interval. Then we take the upper $(1-\alpha)$ quantile of this sorted list as the r that we use for the CP interval. We refer to [Sesia and Y. Romano, 2021] for a more detailed explanation of the method and of it's validity, but it gurantees in expectation of both, the training and the validation set, that the CP interval will contain the desired proportion of the target. In particular, in this case r acts as the conformity score of the sample as usually defined in CP literature like in the work by [Sesia and Y. Romano, 2021].

It is noteworthy, that if we have only few calibration samples, then the method will not work well, since the quantile will be very noisy. In this case the common choice is to resort to an overestimation, which means we take r larger than the $1-\alpha$ quantile would suggest.

2.7.2 Common Implicit Assumptions on the CP-Region Fuction

It is common practice in the CP literature to do calibration without of considering the implict assumptions that are being actually made when doing so. In particular, in many CP-methods we only predict few quantile limits, often the 5% and 95% quantiles, and if we have a miscalibration we simply recalibrate the model by moving all quantiles by the same amount. Many assumptions are made there, in particular we assume that the distribution is symmetric and that the densities of all samples look similar. In fact those implicite assumption are directly imposed on $\mathcal{P}_\theta$ and thus the whole assumptions of the model. Furthermore, those assumptions might be orthorgonal to the optimization objective of CDE-method, i.e. to precisely model the true PDF asymptotically and we need to be careful to make as few assumptions as possible.

2.7.3 Recalibration of CDE on CP when using HDR

As the HDR for a given confidence level α is actually a set of subintervals which conform with Assumptions 2.7.1 with $\alpha = r \in (0, 1)$, for which it holds that under the assumption of continous differentiablity those subintervals will contain α proportion of the probability mass on the estimated distribution, which implies that we basically have a set of quantile intervals that sum up to α, we can see that a probabilistically calibrated model will actually contain the true value in the HDR with a probability of α. Continous differentiablity is required because otherwise the HDR can by its definition contain higher probability mass. However, this does not make the model uncalibrated but is just a property of the HDR and furthermore it does not make the statement less relevant.

If we are not probabilistically calibrated as described in Section 2.7.3, we can directly utilize the concepts described in Section 2.7.1 when using HDR. This is because HDR can be interpreted as a sequence of nested sets as the first requirement in 2.7.1 when using the a in Equation 2.7 as the index which goes from 0 to 1. Moreover, with continous differntiability of the PDF, HDR also fullfills the second requirement in 2.7.1. This means we can use the recalibration strategies as proposed by [Sesia and Y. Romano, 2021] to recalibrate the model to the desired confidence level.

Calibration of CDE on CP

Probabilistic calibration for a model as defined by [Gneiting, Balabdaoui, and Raftery, 2007] says that for any probability level p, the proportion of probability mass contained below the quantile function at level p for each sample x should approach p as we increase the number of samples. If the CDE model is calibrated this has some major implications. In particular, it implies that any denstiy-proportional subinterval on the target space for each sample x, will really contain the correct amount of probability mass as the number of samples goes to infinity. In particular this means that for any two probability levels p_1 and p_2 it holds that averaged over infinite samples the true proportion of mass between the inverse CDF for each sample for p_1 and p_2 will be $p_2 - p_1$. This means also that any conformal subinterval with some confidence level we take from the CDF will be calibrated marginally (in expectation over the sample space), where we can simply infer a

subinterval for some confidence level α by taking the inverse CDF on two points to obtain $[F^{-1}(k), F^{-1}(k + (1 - \alpha))]$ with $k \in [0, 1 - \alpha]$.

While the fact that we can infer CP-calibration when we have probabilistic calibration in CDE is a nice property even tho we need to aknowledge that CDE calibration is not always given and for such cases we need to recalibrate the model with a recalibration strategy as described in Section 2.7.1.

2.7.4 Calibration of CDE-methods in General

Now that we have seen that it is possible in practice to recalibrate CP models Section 2.5 directly implies that we can generalize this concept to all CDE-methods, i.e. to CDE and QR which is a core contribution of this work. In order to rigoriously utilize the theoretical framework that we have established, we first need to reformulate calibration for CP to a more general form that can be applied to all CDE-methods. Therefore we need to realize what recalibration means in the context of $\mathcal{P}_\theta$, i.e. the constrainted set of PDFs that any CP method is acting on.

When we recalibrate, we essentially always admit that the current model $\mathcal{P}_\theta$ does not describe the true PDF accurately and in order to obtain a certain property, calibration that is, we intend to change $\mathcal{P}_\theta$. In other words, we need to formulate what $U(\cdot, \psi)$ does in Equation 2.35 as part of θ, which defines the model, itself. This means we apply a function to the constrained set of PDFs $\mathcal{P}_\theta$ to obtain a new set of PDFs $\mathcal{P}_{\theta'}$ that is calibrated or more generally that we apply the optimization problem in Equation 2.35 to. However, if we are able to change θ arbitrarely in order to fullfill the optimization problem then it becomes trivial and meaningless (we can just use the marginal distribution of the targets).

In particular, how the practical optimization in Section 2.7.1 is done is by using a function to define the conformity score for each sample and then to sort the samples by this conformity score and to take the upper quantile of the sorted list as the conformity score for the CP interval. Thereby we need to fullfill Assumptions 2.7.1. However, those assumptions are actually are more general and allow us to find a ψ for every desired coverage level $\alpha - 1$. This assumption implicitly assumes that all elements of $\mathcal{P}_\theta$ are the same a.e. since otherwise it would be ambigous at some level ψ which samples are in the CP interval and which are not. By optimizing the objective function in Equation 2.35 we are essentially

transforming this $\mathcal{P}_\theta$ w.r.t. $U(\cdot, \psi)$ such that a certain nested set gets assigned a different probability mass as before, in expectation. Implicitly in doing so, we are lifting all other assumptions from $\mathcal{P}_\theta$ which means we can not make any more gurantees there but only have the restriction on the new $\mathcal{P}_{\theta'}$ that it is calibrated in expectation within this regions. Moreover we have the gurantee that the region is part of a sequence of nested sets which are defined thru $U(\cdot, \psi)$.

Implicit Assumptions when Recalibrating CDE-methods

We can see that without further considerations, the traditional way of calibration seems almost nonsensical. As hinted in Section 2.7.1, the assumption that the estimated restricted set $\mathcal{P}_\theta$ must be close to the model of the true conditional PDF backed up by the reasoning that the CDE-method used to produce $\mathcal{P}_\theta$ can asymptotically model the true PDF is a reasonable one.

In particular tho, the reason why CP actually made the implicit assumption of the monotonicity of the PDF, is because without this, after calibrating a single quantile, we would loose all other knowledge of the model, meaning that we would not know anymore where the other quantiles are or where we can expect them to be. The monotonicity of density assumption basically says that if I shift all bordering quantiles at the same time (or create new borders) the density within them stays in tact which again, is only reasonable at all because of the asymptotic property of the CDE-method model. The same logic can directly be applied then to all CDE-method models; we just shift multiple quantiles at once and we say based on the same assumption that this is legitimate.

While this provides us with a new perspective on recalibration, it also shows that recalibration in CP as it exists right now has some fundamental flaws. In particular, the implicit assumption of the monotonicity of the PDF is not always fullfilled and it is not clear how to actually check if it is fullfilled. This means that recalibration in CP is not always guranteed to work and it is not clear when it will work and when it will not work. This is a very important insight that we have gained here and it is a very powerful tool that we have established here.

Practically to use this for e.g. CDE, one would simply define a theoretically infinitely dense grid of quantiles and then shift them all at once to recalibrate the model which will squeeze

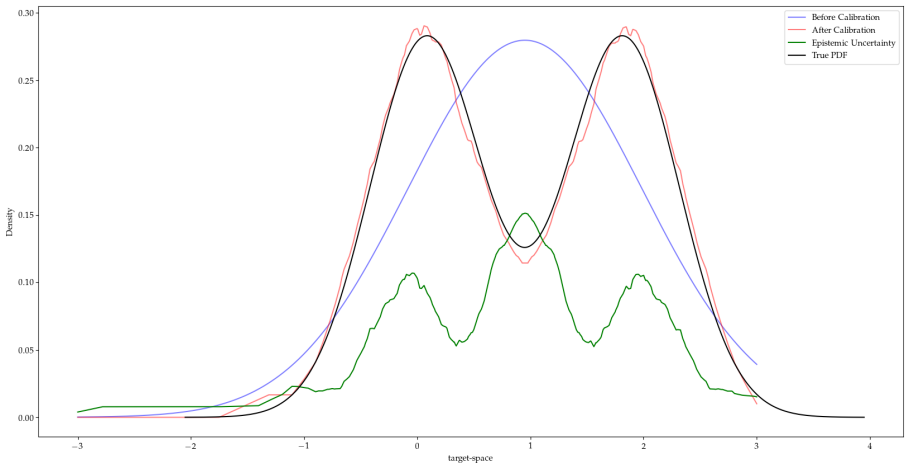


Figure 2.3: Recalibration of a bimodal CDE model. The model is recalibrated by shifting the quantiles of the HDR.

and stretch the model in the right places to obtain calibration. However, we can see here if we look close, and this is not a mistake in deriving the general calibration, that in order to realize a reasonable calibration with an infinitely dense grid of quantiles, we would need to have an infinitely large calibration set, however we can at least asymptotically use this technique to recalibrate whole CDE models.

Moreover, some practical considerations that could be made is that there are more samples at denser areas so we could use a denser grid there and a less dense grid in less dense areas. Quantiles induced by the HDR as the quantile limits for recalibration fit this description. Moreover, due to the limitations in the number of samples, empirically we saw that it is helpful to do a smoothing operation after we recalibrate whole CDE models. Even tho, of course, this imposes another assumption on the model, it is a reasonable one to make (since we usually already assume anyways that its somewhat smooth as we can see in Section 2.2). A visualization of this can be seen in Figure 2.3.

A Pseudo Code where we use HDR to recalibrate a CDE model is provided in Algorithm 3.

2.8 Uncertainty and Calibration: The Connection

The general area of uncertainty estimation has been growing drastically in recent years. This is because of the increasing complexity of models and the need to understand the models better as well as the requirement in risk-sensitive applications to understand when and how much the model can be trusted. Generally, as described by [Hüllermeier and Waegeman, 2021] there exist a lot of different types and perspectives on uncertainty. Most importantly there exists a distinction between so called aleatoric and epistemic uncertainty. Not all defintions of those two fully agree but generally aleatoric uncertainty is the inherent stochtasticity of the data while epistemic uncertainty is the unsureness of the model if only a limited amount of data is observed.

For example, if we try to model the coinflip of a biased coin with 75% probability of landing on the head and 25% on tails, then epistemic uncertainty would be if we did only observe 10 coinflips and we are unsure yet about the exact porbabilities within the coin. Aleatoric uncertainty would be the the inherent randomness of the coin that we might try to model. This means the 75% and 25% themselves are the aleatoric uncertainty. Differences of interpretation of those two kinds of uncertainty are in practice often inherent in how we expect the true model of the data to be and how it really is. For example, if we were to expect for the coinflip experiment that the coin is always landing on the same side with the intention to learn this side,and differences in what we observe is simply noise, then we might not be able to model either uncertainty properly. In particular we might in this case predict that the coin always lands on head and there is simply 25% noise which is of of course not true. A slight variation in definitions between aleatoric and epistemic uncertainty within works [Hüllermeier and Waegeman, 2021] is often whether aleatoric uncertainty is only noise or if it also contains stochtasticity of the data that might be reducible with more features like hidden variables that are not really observable. In this work, we do not want to dive into the philosophical interpretation of this and define that aleatoric uncertainty is always the randomness of the targets, given a fixed set of features, without of considering that there might be hidden variables that actually could reduce this uncertainty.

CDE-methods' task is to model the aleatoric uncertainty when we try to predict targets given features. The CDE-method model $\mathcal{P}_\theta(x)$ that we estimate is supposed to come as closely as possible to the inherent randomness of the targets given the features. One

particular aspect that rarely has been acknowledged is, that those models can, at least with the optimization objective alone, not learn epistemic uncertainty. This implies, that the model might actually give overestimations in the preciseness of outcomes. Moreover, since in this kind of setting model the aleatoric uncertainty itself and usually do not assume that there is such a thing as a second-order aleatoric uncertainty, we actually assume that the conditional distribution of a target given features is deterministic i.e. there is no (second-order) aleatoric uncertainty. This also directly implies that all error of a model which has certain asymptotic gurantees stems from epistemic uncertainty alone.

In this work we provide a novel perspective on how one can estimate full uncertainty which not only includes the directly modeled aleatoric uncertainty but also the epistemic uncertainty. In the context of CDE, aleatoric uncertainty asks the question how the targets are distributed, given the features, while epistemic uncertainty asks to which extent we can actually accurately predict that. Especially for CDE where we need to predict very specific details about the distribution of the data, this is a very important question, as we realistically in real world settings can never model the distribution with full accuracy.

The tool that we propose for this is a novel method that highlights calibration in a new way. In particular, we argue that recalibration of models can be used to accurately infuse the prediction with a lower bound on epistemic uncertainty. Thereby we can reestimate the whole distribution with both epistemic and aleatoric uncertainty which thus gives us a more accurate perspective on what we really know about the distribution of a target.

2.8.1

Our proposed method relies on two key insights. First, all error in CDE-method models is exclusively due to epistemic uncertainty if the model is asymptotically correct and does not over-/underfit. Secondly, if we observe that a model is miscalibrated, it is a direct implication that the model is suboptimal and thus that there is error. In particular what that tells us is, if a model is misaclibrated there is epistemic uncertainty. Moreover, we claim that the amount of miscalibration is in a direct relationship with the amount of epistemic uncertainty and we also claim that if we can recalibrate the whole model, we can estimate a lower bound of the epistemic uncertainty as the difference between the distribution of the uncalibrated and the calibrated model.

In order to solidify those claims and insight we need to define a variant of epistemic uncertainty where we can show those relationships. In particular, in our case we aim to express the amount of epistemic uncertainty in the output space of the model, i.e. in the space of the conditional PDF that we estimate. Therefore we observe that it is a fair statement to say that if the likelihood of the data given the model increases, the model is closer to the true model. The reason why this insight is non-trivial in the context of CDE-methods is that we generally do not have information about the optimal likelihood that could be obtained which is due to the fact that we need to make smoothness assumptions as described in Section 2.2 and have limited data.

Lemma 2.8.1. *If we have a model $\mathcal{P}_\theta$ that is miscalibrated w.r.t. HDR on $\mathbf{Z}_I$, then there must exist a model $\mathcal{P}_{\theta'}$ that is calibrated and where we have a higher likelihood if the true $p(\mathbf{Y} \mid \mathbf{X})$ fullfills Assumptions 2.1.1.*

Proof. □

Lemma 2.8.1 shows us that if we are not calibrated than we know that we still have an error in the model i.e. that we have epistemic uncertainty to some extent. We now aim to construct a method with which we can recalibrate our model such that we have gurantees that it will increase the likelihood.

2.8.2 Recalibration: A new perspective

CDE-method models asymptotically will capture the true conditional PDF perfectly since objective functions used for this purpose, like MLL or minimal Pinball loss gurantee that. The natural question that arises here is why it is then in practice that CDE-method models are usually not calibrated, even on the training data itself. This can be answered by observing that those objective functions are not exclusively looking for calibration but also for the tightest fit. In particular, this can be intuitively interpreted with Section 2.6.1 where we can see that true distribution is equivalent with the model that minimizes the size of the HDR, while maintaining calibration. This means that in CDE-method models, the objective is a constrained optimization problem, where the constraint is implicit and not explicit. This implies that it can happen that the model becomes uncalibrated, even

for the training data, since it might be w.r.t. the optimization method the way of steepest descent to neglect the constraint but instead to minimize the HDR, conceptually.

Thereby, one needs to realize that the calibration problem by itself is not a difficult one as one can simply take the marginal distribution of the targets as the output PDF and it will be calibrated, but relatively meaningless. We claim, that a robust CDE-method model should always to some degree model the marginal distribution, for the reason that it simply is a calibrated educated guess on where the next datapoint might lie, in particular when assuming that we are very unsure about the relationship between features and targets. Even with no modeling knowledge of a datapoint, it is a relatively safe bet to simple estimate the marginal distribution.

Recalibration in CDE essentially makes the difference between high and low density regions either greater or smaller, possibly differently for different density levels or even feature locations possibly with clustering like [Izbicki, G. T. Shimizu, and Stern, 2019]. Intuitively this can be interpreted as the model moving further away or closer to the marginal distribution, at least partially, but there might be other distributional influences. Generally what it means is, that you marginally over the whole distribution of training/calibration samples say, based on calibration feature-target pairs, how accurate the model is, and based on that you adjust your general believe in the models predictions.

Particularly, this is based on the premise that all density levels are reasonablbly estimated, but usual optimization methods gurantee that. For example this means that if I calibrate the model on the calibration set and the expected quantile levels are very different from the empirical ones, for example that in less dense regions, there are lots more samples than there are supposed to be, that means via recalibration we can make denser regions less dense and less dense regions more dense. Moreover, if on the train set we also have strong miscalibration, what that tells is is not necissarely that we have a bad fit on the training data, but only that in optimizing we heavily violated the implicit constraint of the calibration and we can recalibrate that. We have to notice that when recalibrating we essentially lose some gurantees about the optimality of peaks etc., however, in practical applications often calibration is more accurate. This gurantees that we loose happens because we, in practice, always need to recalibrate based on a finite set of calibration data which also might be inaccurate. Especially we also marginally calibrate the model which means that even tho our distribution might be accurate for some samples, even in terms of calibration, it might be not accurate for all and thus we basically calibrate

them in the way such that we change the model the least. Thereby we of course can again apply a clustering approach in the feature space to get different recalibration strategies for different regions of the feature space, but one needs to be careful that we do not have too little calibration samples for each cluster as we can actually overfit on the calibration data or get simply inaccurate recalibration. This is a very important point to consider when recalibrating models.

2.8.3 Mathematical interpretation

As already discussed in Section 2.7, when we need to recalibrate a model due to miscalibration it essentially means that we admit that our model is wrong to some extent. There exist ways to calibrate models based on assumptions given in Assumptions 2.7.1 and we can also apply those to any CDE-method. However, as already stated, without of further assumptions this is rather meaningless. In particular, we assumed border-wise monotonicity of the PDF during the recalibration process.

Here we give a stronger argumentation why this assumption is reasonable and also that when we take the difference between the uncalibrated and the calibrated PDF, we actually can interpret this as the epistemic uncertainty of the model.

Monotonicity of the PDF between borders

For this we need to dive deep into the interpretation of what happens if we make the grid of quantiles tighter infinitely. So, lets do that tomorrow...

Connection to Epistemic Uncertainty

For this we require that for the true PDF Assumptions 2.1.1 hold and moreover we require that the model is asymptotically correct. In particular we require a slightly harder type of convergence gurantee where the function that produces the model $\mathcal{P}_\theta$ for each amount of data always comes as close as possible to the true distribution, i.e. we do not over or underfit. Thereby we need in practice to consider multiple techniques like noise

regularization, dropout, limiting of expressivity or other techniques but generally this is not an issue.

Under those assumptions, if we can find a quantile level $q \in (0,1)$ where we see that the model is not calibrated correctly under the interval finding method which is one of the likes we saw in Section **??** then if the method is H (possibly HDR), then $\mathbb{P}(\mathbf{Y} \in H(\mathcal{P}_\theta, q)) = q' \neq q$. By using the continuity of the true conditional PDF from our assumptions, this implies that there exists a region $\mathcal{R}_q$ around q such that

$$\forall \hat{q} \in \mathcal{R}_q : \mathbb{P}(\mathbf{Y} \in H(\mathcal{P}_\theta, \hat{q})) \neq \hat{q} \tag{2.36}$$

In words, a region around q is uncalibrated and furthermore, it holds that

$$\forall \hat{q} \in \mathcal{R}_q \exists \hat{q}'' \in (0,1) : \mathbb{P}(\mathbf{Y} \in H(\mathcal{P}_\theta, \hat{q}'')) = \hat{q} \tag{2.37}$$

under the assumption that also $\mathcal{P}_\theta$ is continous.

Theorem 2.8.2. *Under the assumptions above, the epistemic uncertainty, up to scaling by a prior distribution on θ can be found with:*

$$\int_{(0,1)} |\mathbb{P}(\mathbf{Y} \in H(\mathcal{P}_\theta, q)) - q| dq = p(\theta \mid \mathcal{D})^{-1} \tag{2.38}$$

Proof. We can decompose $p(\theta \mid \mathcal{D})$ by its definition. In particular, we can write it as:

$$p(\theta \mid \mathcal{D}) = \frac{p(\theta, \mathcal{D})}{p(\mathcal{D})} \tag{2.39}$$

$$= \frac{\frac{\partial^2 \mathbb{P}(\Theta<\theta, \mathbf{Z}<\mathbf{z})}{\partial\theta\partial\mathbf{z}}}{\frac{\partial \mathbb{P}(\mathbf{Z}<\mathbf{z})}{\partial z}} \tag{2.40}$$

where with notational overloading Θ represents the random variable defining the estimated parameters. Now we need to observe that in Equation 2.40 Θ is actually dependent on the data $\mathbf{Z}$ itself because it is a function of the data. This means that we can write the second part as:

$$\frac{\frac{\partial^2 \mathbb{P}(f(\mathbf{Z})<\theta, \mathbf{Z}<\mathbf{z})}{\partial\theta\partial\mathbf{z}}}{\frac{\partial \mathbb{P}(\mathbf{Z}<\mathbf{z})}{\partial z}} \tag{2.41}$$

What we now need to show, is that this quantity is small iff the left hand side of Equation 2.38 is large which would imply that the epistemic uncertainty is large iff the error is large. For this we need to go into the details what it means that there can exist such an error. Generally under our assumptions, if there is an error, that means that there is not enough data to model the quantile points of the errors effectively. This, in turn, comes from the fact that f is asymptotically optimal and does not over/underfit. which means that it is as small as it can be with the data. This implies that at θ the model must change slow in the data i.e. the derivative is small. Formally, this can be shown by contradiction.

Assume that the error is large, but that the derivative is also large. This implies that with a change of the model, which implicitly happens due to a change in $\mathbf{Z}$ i.e. the data, the probability measure $\mathbb{P}(f(\mathbf{Z}) < \theta, \mathbf{Z} < \mathbf{z})$ changes a lot (high derivative). That implies that the model itself is very sensitive to the data and adjusts its fit very precisely at θ, which due to assumptions implies that the model has a strong understanding of the data i.e. low epistemic uncertainty. However, if we had low epistemic uncertainty, then due to asymptotic optimality, also the error would be small. Which implies that f in this case can not be asymptotically optimal. This is a contradiction which implies that the derivative must be small at θ.

$$p(\theta \mid \mathbf{z}) = p(f(\mathbf{z}) \mid \mathbf{z}) = p(f(\mathbf{x},\mathbf{y}) \mid \mathbf{x},\mathbf{y})$$

$$= \frac{p(f(\mathbf{x},\mathbf{y}),\mathbf{y}|\mathbf{x})}{p(\mathbf{y}|\mathbf{x})} = \frac{p(f(\mathbf{x},\mathbf{y}),\mathbf{y},\mathbf{x})}{p(x)p(\mathbf{y}|\mathbf{x})} = \frac{p(\mathbf{y},\mathbf{x}|f(\mathbf{x},\mathbf{y}))p(f(\mathbf{x},\mathbf{y}))}{p(\mathbf{x})p(\mathbf{y}|\mathbf{x})}$$

$$= \frac{p(\mathbf{y}|\mathbf{x},f(\mathbf{x},\mathbf{y}))(\mathbf{x}|f(\mathbf{x},\mathbf{y}))p(f(\mathbf{x},\mathbf{y}))}{p(\mathbf{x})p(\mathbf{y}|\mathbf{x})}$$

$$= \frac{p(\mathbf{y}|\mathbf{x},f(\mathbf{x},\mathbf{y}))p(\mathbf{x}|f(\mathbf{x},\mathbf{y}))p(f(\mathbf{x},\mathbf{y}))}{p(x)p(\mathbf{y}|\mathbf{x})}$$

where we only used rules of basic probability theory. Now if we see that $f(\mathbf{x}, \mathbf{y}) = p_\theta(\cdot \mid \mathbf{x})$ paramerizes a model that approximates $p(\mathbf{y} \mid \mathbf{x})$ and if we assume that it is close to the true distribution we can use it as approxiamtion to obtain:

$$= \frac{p(\mathbf{y}|\mathbf{x},f(\mathbf{x},\mathbf{y}))(\mathbf{x}|f(\mathbf{x},\mathbf{y}))p(f(\mathbf{x},\mathbf{y}))}{p(\mathbf{x})p(\mathbf{y}|\mathbf{x})}$$

$$= \frac{p(\mathbf{y}|\mathbf{x},p_\theta(\cdot|\mathbf{x}))(\mathbf{x}|p_\theta(\cdot|\mathbf{x}))p(p_\theta(\cdot|\mathbf{x}))}{p(\mathbf{x})p(\mathbf{y}|\mathbf{x})}$$

now we can see that in $p(\mathbf{y} \mid \mathbf{x}, p_\theta(\cdot \mid \mathbf{x}))$ we have the density of $\mathbf{y}$ conditioned on $\mathbf{x}$ and a model of $p(\mathbf{y} \mid \mathbf{x})$ that is close to the true distribution. This is approximately equal to $p_\theta(\mathbf{y} \mid \mathbf{x})$. Now we see that:

$$= \frac{p_\theta(\mathbf{y}|\mathbf{x})p(\mathbf{x}|p_\theta(\cdot|\mathbf{x}))p(p_\theta(\cdot|\mathbf{x}))}{p(\mathbf{x})p(\mathbf{y}|\mathbf{x})} = \frac{p(\mathbf{x}|p_\theta(\cdot|\mathbf{x}))p(p_\theta(\cdot|\mathbf{x}))}{p(\mathbf{x})}$$

approximately. Now we have three terms and two of them are basically independent of the connection between $\mathbf{x}$ and $\mathbf{y}$. That means they are approximately constant and are left with only:

$$= p(\mathbf{x} \mid p_\theta(\cdot \mid \mathbf{x}))$$

which is $\mathbf{x}$ conditioned on the model of $\mathbf{y}$ given $\mathbf{x}$. Now for $\mathbf{x}$ we can basically assume that this is calibration data and also we know that for the calibration data we have a corresponding $\mathbf{y}$ to every $\mathbf{x}$ which we can use to approximate the probability. In particular, it is not true that $\mathbf{x}$ is independent from the conditioned quantity since we assume the model to be better at modeling data that it has observed.

□

3 Empirical Study

In order to practically verify the validity of the novel theoretical findings as described in Section ?? and the approach to use the HDR as described in Section ?? for finding the best conformal regions we did a series of experiments with two different stages and a multitude of benchmark datasets. In particular, the first stage was a extensive hyperparameter search with Bayesian optimization on eight datasets with over 1000 hyperparameter configurations on each. Thereby nested cross validation was utilized. The goal of this stage was to get a better understanding of the hyperparameters, including novel hyperparameters and to find a good starting point for the next stage. The second stage then was to do a more detailed hyperparameter search with a smaller grid but with multiple test set splits to get a better understanding of the generalization of the hyperparameters. In this chapter we will describe the results of the first stage and the setup of the second stage. In particular we will discuss mutliple novel hyperparameters that we experimented with and highlight the ones that significantly improved the performance of the models empirically.

3.1 Core Model Classes

In the course of this work we experimented with a multitude of different CDE-method model classes. In particular, insights gained from Section 2.5 establish that we can use any model class in the literature that has been used from CDE, QR or CP which opens up a wide range of options. Model classes experimented with in this work include Mixture Density Networks (MDNs) by [Bishop, 1994], Kernel Mixture Networks (KMNs) [Ambrogioni et al., 2017], Multiple Quantile Regression (MQR) [Gupta, Kuchibhotla, and Ramdas, 2022; Moon et al., 2021], Normalizing Flow Networks (NFNs) by [Trippe and Turner, 2018] and conventional Regression as a baseline. However, in the latter experimental stages we

restricted ourselves to MDNs, KMNs and MQRs as they showed the best performance in the first stage and also have significantly lower computational requirements than NFNs which is also why we restrict the reports to those three model classes in this work.

3.2 Experimental Setup

For the experiments done in the course of this work a python setup with standard libraries like PyTorch [Paszke et al., 2019], NumPy [Harris et al., 2020], Pandas [team, 2020], Scikit-Learn [Pedregosa et al., 2011] and others was utilized. The hardware consisted of NVIDIA TITAN X (Pascal) GPUs, each with 12GB memory.

3.3 Hyperparameters

Architectures in this regime of ML offer an extremely wide range of possible hyperparameter settings. In particular, this is due to the fact that in the output space there is a lot of freedom in how we can model the distribution. While not the main focus of this work, it is an interesting realization that CDE-methods have possibly the most degrees of freedom in their output compared to any other ML task. In particular, it is impossible to fully output in all those degrees of freedom but any output must necissarly be an abstraction of the true PDF. For example we just output model parameters of a mixture of Gaussians instead of the infinitely dense PDF which is very obviously not possible. The elegance now comes in how we decide to make this abstractions and many options with the help of CDE-methods exist.

In this section first we will discuss the known hyperparameters and how the performance seems to be affected by them with a rigorious empirical analysis of good settings for those in the regime of CDE-methods. Then we will discuss novel hyperparameters that we experimented with and how they affected the performance of the models. In particular we found that there is no existing literature that discusses in depth the possible hyperparamters for CDE-methods and their impact. This is a very important contribution of this work as it gives a good starting point for future research in this area as well as a good starting point for practical applications of CDE-methods in the industry.

3.3.1 Known Hyperparameters - General

Learning Rate

A learning rate of around `2e-4` gave us good performance accross all datasets. Moreover, for some of our experiments we utilized a learning rate scheduler `ReduceLROnPlateau` with a patience of 5 epochs, cooldown of 3 and a factor of 0.5. This gave us a slight performance boost when using MDN and KMN models, but not with MQR models.

Batch Size

This hyperparameter varies a lot between datasets. Some datasets had a better performance with a size around 32 and others performed best with as high as 512 with a significant impact on performance. We suggest that this hyperparameter should be tuned for each dataset individually. We did not experiment with a batch size scheduler.

Number of Epochs

We found the models had a rather quick convergence with mostly lower than 50 epochs and performance not improving with more epochs. Furthermore we used early stopping by monitoring the negative log likelihood loss for all models as this loss is the most important one for CDE-methods.

Dropout

We experimented with a wide range of dropout rates and found that the models are extremely sensitive to this hyperparameter. In particular a dropout rate of more than 0.05 will lead to a significant performance decrease for most instances. However, there are some exceptions, in particular when using KMN. Moreover we observed a correlation between the Dropout rate, number of components in MDN, number of layers and number of units. A more expressive architecture allows for a slight increase in dropout which is to be expected. Moreover, we suspect that the higher dropout preference in KMN is due to the reduced degrees of freedom in the KMN model compared to MDN and MQR.

Weight Decay

We initially did experiment with this hyperparameter, tuning it in many ways. However, we found that setting it to 0 consistently yields the best performance.

Base Architecture

The base architecture used was a multi layer perceptron (MLP). We experimented with a wide range of depths and widths and the most performant architecture was one with four hidden layers with sizes $[64, 128, 128, 64]$; however it is possible that with significantly more or complex data a deeper architecture might be beneficial.

Activation Function

The different activation functions we tried were ReLu, Leaky ReLu, TanH, Sigmoid, SELU and ELU. The three best performing ones were ReLu, Leaky ReLu and TanH, however, the differences were not very significant and there are some slight variations between datasets. We decided to use ReLu as it was the most stable one. It is noteworthy that when using ReLu we utilized a He initialization and when using TanH we utilized a Xavier initialization as best practice.

Input-/Output Noise

A hyperparamter that to the best of our knowledge is novel to the CP model literature and was first introduced to CDE methods by [Rothfuss, Ferreira, Boehm, et al., 2019] is the input/output noise to the models. This hyperparameter boosts the performance very significantly and it essential for performance in all CDE-methods. We found an input and output noise of around 0.03 the most consistent, however, tuning this hyperparameter for each dataset can improve the performance even further by a significant margin. In particular when using KMN a higher noise level is sometimes beneficial with values up to 0.3.

Layer Norm

Layer Norm was tried in the later course of the experiments and we found it does reduce the performance of all our models. We did not investigate this further.

Component Entropy Loss

This additional regularization loss which can be used in MDNs and KMNs essentially aims to decrease the entropy of the different mixture components. We found adding a small amount of around 0.125 tends to slightly increase the performance of the models.

Number of Components

The number of components in both MDN and KMN make a significant difference. For MDN a number of components around 35 was the most performant across all datasets. For KMN a higher number of 90 was the best performing one. Moreover, we we had two kernels in the KMN model. This makes an effective number of components of 180. For MQR generally a higher quantile number is better always since with a higher number basically we can model the conditional CDF better. However, we restricted ourselves to 256.

Component Distribution

We experimented with Laplacian and Gaussian mixture components. Both have their advantages and disadvantages depending on the dataset. However, the differences were not very big and since Gaussian components were slightly more performant we decided to use those.

Kernel Width

We initialized the kernel width with 0.3 and 0.7 but decided to make them learnable hyperparameters, which means that we optimize them with the model parameters which slightly boosted the performance.

3.3.2 Novel Hyperparameters

Additional Target Noise

As explained in the theoretical part of this work, in order to be a reasonable prediction that can also be calibrated effectively, certain conditions need to be fullfilled on the distributions. In particular, it is required to have a certain amount of density everywhere. Moreover, we suspected that it might be helpful to enforce the marginal target distribution component on the CDE-models since then during calibration we can be sure that for each possible target there is at least a small amount of density in the model. In particular we decided to implement this by swapping certain targets which implicitly should enforce that the model has a certain amount of density everywhere on the marginal distribution. Furthermore, a theory was that if we have more samples then we have less empistemic uncertainty which implies that we would need to inforce less density on the marginal distribution and thus we made the number of swaps per epoch.

Another suspicion, especially for MDNs, was that some components can never "reach" the density where they want to go during training. In particular this came from the assumption, that if initially all mixture componens are somewhat in the center and we have a smaller density more on an outside location, that in order for a component to move to this location it would need to bridge the gap betweeen the high density in the center to the lower peak on the outside which might be very low density. We suspected that if a component needs to do that it might cause degenerative behaviour since if a component is at a location with low actual density the loss should enforce a lower weight on the component, which, in turn will decrease the general gradient imposed on this component. We suspected that it might happen that a component will then just stay in a gap with a negicible amount of weight so that it will never move again. In order to counteract this we decided to initialize the training with a large amount of uniform noise in the space of the targets and let it decay rather quickly over time. This procedure slightly improved the performance on all tasks but it is hard to say if there might be better ways to do it.

Learn MQR Quantile Distribution Std

MQR asymptotically with an increase in quantile components can as shown in Section 2.5 model the true PDF fully. However, since we have a limited amount of training data and also a limited amount of compute we needed to resrict the number of components to 256. In order to still be able to efficiently calculate the density at a point we decided to treat each quantile as a component with equal weight in a mixture of gaussians. Thereby we decided to learn the standard deviation. However, the standard deviation in this case can not be learned with the Pinball loss that is used for MQR. Thus we decided to use the NLL loss for the standard deviation only but without impacting the quantiles. We did this by detaching the means from the computational graph of the gradient in the loss function.

3.4 Datasets

In the course of this work we experimented with a multitude of different datasets. In particular, we tried to orient ourselves at the literature in CP and CDE [Rothfuss, Ferreira, Boehm, et al., 2019; Sesia and Y. Romano, 2021] and used most of the datasets that were used there. Moreover we tried to have datasets with some different properties to gain more insight into strenghts and weaknesses of different models. Thereby we used `Boston Housing`, `Concrete`, `Energy Efficiency` as smaller datasets in order to elaborate performance with lower number of samples. Moreover, we used larger datasets `Meps19`, `Meps20`, `Meps21`, `CASP`, `Blog`, `Facebook1`, `Facebook2`. Finally, we used two versions of a time series dataset provided by VoestAlpine AG about energy price prediction. In particular `VoestRealistic` and `VoestIdeal` are two variations of the same data where we used realistic features and features as if we could look into the future respectively. This dataset was used to investigate the performance of CDE-methods on time series data.[1] In Table 3.1 we provide an overview of the characteristics of each dataset.

[1] The features used for the Voest datasets were all taken from `https://transparency.entsoe.eu/dashboard/show` in a time windows from November 2022 till November 2023 but will not be directly disclosed.

Table 3.1: Comparison of Different Used Datasets

Dataset	# Samples	# Features	Description
Boston Housing	506	13	Housing prices in Boston
Concrete	1030	8	Concrete compressive strength
Energy	768	8	Energy efficiency of buildings
CASP	45730	9	Protein structure prediction
Blog	52397	280	Blog popularity prediction
Facebook1	40948	53	Facebook user engagement
Facebook2	81311	53	Facebook user engagement
Meps19	15785	139	Medical Expenditure Panel Survey
Meps20	17541	139	Medical Expenditure Panel Survey
Meps21	15656	139	Medical Expenditure Panel Survey
VoestRealistic	35001	42	Realistic Features Voest Dataset
VoestIdeal	35001	42	Ideal Features Voest Dataset

3.5 Calculation of the HDR

Calculating the HDR is a straightforward procedure which is described in the PseudoCode below in Algorithm 1 where we assume that the target grid is spaced equally but that is not required technically. The output are the elements of the target grid that are in the HDR. To obtain the actual intervals we just need to go half the step size to the left and right of each element in the HDR but it is left out the algorithm for inconvenience of writing that down. Moreover it is possible to add an improvement step into Algorithm 1 where we can smooth the HDR via linear interpolation between the consecutive densities. This is a very important step as it can significantly improve the performance of the models in particular if it is expensive to evaluate the density at each point by using the model itself. Moreover when we want a single interval as region than as argued in Section 2.6.2 we just connect the largest and smallest border of the HDR which is guranteed to have more than α probability mass. Note that if we have after the $I_{HDR}+1$ item other items that have the same density as the $I_{HDR}+1$ item technically also should include those in the HDR but we decided to not do that for simplicity.

Algorithm 1 HDR Calculation

Input: CDE-method model f, Features x, Significance Level α, Target Grid y
Output: HDR H
$p \leftarrow f(x, y)$
$p_{\text{normalized}} \leftarrow \frac{p}{\text{sum}(p)}$ We normalize the density
$I_{\text{sorted}} \leftarrow \text{argsort}(p_{\text{normalized}})[:: -1]$ We sort in descending order
$p_{\text{sorted}} \leftarrow p_{\text{normalized}}[I_{\text{sorted}}]$
$p_{\text{cumsum}} \leftarrow \text{cumsum}(p_{\text{sorted}})$
$I_{\text{HDR}} \leftarrow \text{sum}(p_{\text{cumsum}} < 1 - \alpha)$ We look how many elements we need to take
$H \leftarrow y[I_{\text{sorted}}[: I_{\text{HDR}} + 1]]$ We take the elements in the HDR on the overestimated side
return H

3.6 Calculating the Calibrated Conditional PDF

For calibrating the whole PDFs of a dataset we need to find the adjustment for a grid of quantile levels similar to how we would do it for calibrating CP for a sigle level. Therefore we can use Algorithm 3 below which expects calibration samples. In practice we observed that even when inputting the training samples for calibration it increases the general performance. Moreover, it is almost necessary to do smoothing because otherwise the result will be very noisy. In the second for loop we basically reconstruct the PDF from the calibrated HDR by assigning each HDR level the same density. Even tho this algorithm will marginally increase the performance it is possible that on single samples the performance is significantly worse. Moreover, we can obtain a quantification of the epistemic uncertainty by integrating the returned value.

Algorithm 2 Calibrating a HDR at a specific level

Input: Density Grids p, Calibration Targets y, Significance Level α, Target Grid $\bar{y}$
Output: Calibrated Significance Level α'
$p_{\text{normalized}} \leftarrow \frac{p}{\text{sum}(p)}$ We normalize the density
$I_{\text{sorted}} \leftarrow \text{argsort}(p_{\text{normalized}})[:: -1]$ We sort in descending order
$p_{\text{sorted}} \leftarrow p_{\text{normalized}}[I_{\text{sorted}}]$
$\text{cumsum} \leftarrow \text{cumsum}(p_{\text{sorted}})$
$\hat{alpha} \leftarrow \text{cumsum}[\arg\min(|p_{\text{cumsum}} - \alpha|)]$ We find the required quantile levels
$\alpha' \leftarrow 1 - \text{quantile}(\hat{alpha}, 1 - \alpha)$ We find the quantile level of the closest level
return α'

Algorithm 3 Calibrating the Conditional PDF

Input: CDE-method model f, Calibration Features x, Calibration Targets y, Significance Level Grid α, Target Grid $\bar{y}$
Output: Calibrated Conditional PDF grid p', Epistemic Uncertainty p_e
$p \leftarrow f(x, \bar{y})$
$\bar{y}_{\text{spacing}} \leftarrow \bar{y}[1] - \bar{y}[0]$
for α_i in α **do**
  $\alpha_i' \leftarrow$ HDR-Calibrate$(p, y, \bar{y}, \alpha_i)$ Here we use Algorithm 2 to calibrate the HDR
  $H_i \leftarrow \text{HDR}(p, \alpha_i', \bar{y})$
end for
$H_0 \leftarrow \emptyset$
$H_{N+1} \leftarrow \text{ones}(\text{len}(\bar{y}))$
for i in $1, \ldots, \text{len}(\alpha) + 1$ **do**
  $H_i \leftarrow H_{i-1} \cap H_i$ We take the intersection of the HDRs to get the elements for this level
  $p'[H_i] \leftarrow \frac{1}{\text{len}(\alpha) \cdot \text{sum}(H_i) \cdot \bar{y}_{\text{spacing}}}$ We adjust the density for the elements in the HDR
end for
$p_e \leftarrow$ p - p′ We calculate the epistemic uncertainty grid
$p' \leftarrow \text{smooth}(p')$ Optional smoothing because of finite samples and finitely fine grids
$p_e \leftarrow \text{smooth}(p_e)$ Optional smoothing because of finite samples and finitely fine grids
return p', p_e

3.7 Experiment Results

3.7.1 Recalibration of the Whole CDE

We incoperate recalibration of the whole CDE in some of our experiments (not all due to time-constraints) where we can observe also an increase in performance on real world datasets. In particular by choosing the smoothing window appropriately we can consistently increase the performance on validation sets even when calibrating on the train set itself which shows the utility of this method. Figure 3.1 shows examples of recalibration on the concrete dataset where we smooth with $\frac{1}{16}$ of the grid size. In terms of likelihood, the recalibrated method shows a slight but significant increase in likelihood on the test set, in particular -3.202 instead of -3.224 in average log likelihood.

Moreover, we can observe that recalibration of the full CDE can empirically compensate for a significant amount of model misspecification. In particular, as we can also see in Figure 2.3 where the model was specified as an unimodal gaussian distribution but the true distribution is multimodal. In this case the recalibration can compensate for the misspecification as can even recover the bimodal distribution which makes it more than just a nice utility but a powerful tool that can be used to compensate and potentially identify model misspecification.

3.7.2 Main Benchmark Results

Here we show the results of the final experiments on the 12 datasets. The main experimental procedure consisted for each dataset except for the two Voest ones of five-fold nested cross validation to obtain a robust estimate of the actual performance. On the two Voest Datasets instead we decided to always use the same train-test split where we consider the time dependence in the data by using the chronologically first 80% of the data as training data and the last 20% as test data. Moreover, we do cross-validation on a time series split where we expand the training data and use the chronolocially last part for validation. For the exact implementation we refer to our code. The Algorithm used for the evaluation can be seen in Algorithm 4. This algorithm inspired by [Rothfuss, Ferreira, Boehm, et al., 2019] is a nested cross validation algorithm over multiple seeds and hyperparameters that gurantees a robust estimate of the performance of the models on the test set. In particular,

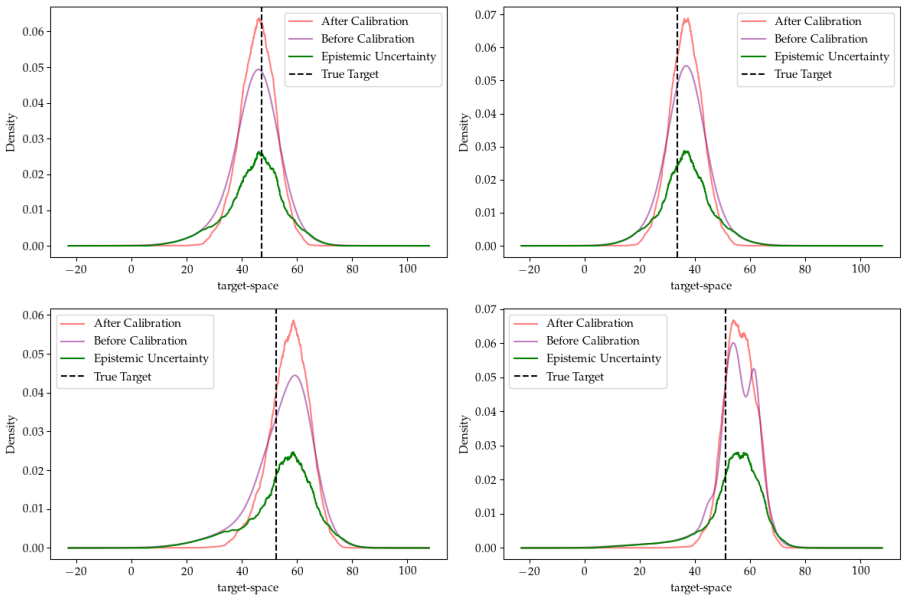


Figure 3.1: Recalibration of the whole estimated conditional PDF on the Concrete dataset. Calibrated on the train dataset and evaluated on the test dataset.

the test set is only used for the final evaluation and the hyperparameters are tuned on the training set which we do CV on.

The results final benchmark results are shown in Table 3.2.

Table 3.2: CDE Experiment Result ALL for Real Data (higher is better)

Dataset	MDN	KMN	NFN	MSE-CDE
Boston Housing	-2.51 ± 0.09	-2.53 ± 0.05	-2.56 ± 0.06	-3.21 ± 0.03
Concrete	-3.09 ± 0.05	-3.20 ± 0.03	-3.14 ± 0.06	-3.79 ± 0.01
Energy	-1.31 ± 0.10	-1.62 ± 0.08	-1.47 ± 0.08	-3.18 ± 0.01
NYC Taxi	5.32 ± 0.03	5.40 ± 0.02	5.18 ± 0.04	5.03 ± 0.05
Voest	-5.40 ± 0.08	-5.30 ± 0.03	-6.42 ± 0.20	-7.09 ± 0.02

Algorithm 4 Evaluation of the Models

Input: Hyperparameter Grid H, Model Class M, Dataset D, Number of Folds K, Number of Nested Folds L
Output: Performance Metrics P
for k in $1, \ldots, K$ **do**
  $D_{\text{train},k}, D_{\text{test},k} \leftarrow \text{split}(D, k)$
  $CVSplits \leftarrow \text{split}(D_{\text{train},k}, L)$
  for h in H **do**
    for $D'_{\text{train}} D'_{\text{val}}$ in $CVSplits$ **do**
      $M_h \leftarrow \text{fit}(M, D'_{\text{train}}, h)$
      $P_h \leftarrow \text{score}(M_h, D'_{\text{val}})$
    end for
  end for
  $h_{\text{best}} \leftarrow H[\text{argmax}(P)]$ Also set calibrated hyperparameters like epoch and α for CP
  $P_k \leftarrow \text{fit}(M, D_{\text{train}}, h_{\text{best}})$
  $P_k \leftarrow \text{score}(M_k, D_{\text{test}})$
end for
$P \leftarrow \text{mean}(P)$
return P

4 Conclusion

5 ToDos

1. Show that calibration is the same as in Gneiting et a. (probabilistic calibration). 1a. Show that there are 2 sub-parts then for CP: fitting-calibration and overfitting-calibration Fitting Calibration: the model is probabilistically calibrated to the training data (is fit well) Overfitting Calibration: The model is basically the same way calibrated to the training data as to the test data -> time series and distributional differences come in here too but is a further subchapter then

2. Then show what my recalibration method does mathematically and how it is different from the other recalibration methods

This all contains that CP is a special case of CDE!

3. I should mention in the thesis the stages of my experiemnts. I first did basically only the stuff I did in my practical work (but not sure if that should even be mentioned). Then I should also mention that I did 1000 runs with bayesian optimization on all datasets with a very high variable range of possible hyperparameters with the goal to get a good understanding of the hyperparameters (as there are some novel ones and complex ones). The next stage that I will now start is the stage where I basically start with reasonably good parameters infered by the previous runs and from there basically a. do very specific experiments for the novel hyperparameters to find out there effectiveness (small grid) b. do hyperparameter runs with a smaller grid but multiple test set spilts (with nested cv).

4. As CP is a special case of CDE, also the synthetic reasoning that we basically need them to make real statements about the performance of methods is also valid for CP. So I should also mention that in the thesis. And from there I can basically start finding out the best hyperparameters for CP methods and then also do the same for CDE methods and also possibly new methods that defeat ceavats on the synthetic sets that I only now can find out.

5. It might be interesting to look into the question if when there is some kind of monotonicity between the hyperparameters and the performance of the model, if that is a good synthetic dataset to infer strategies from. I think that is a very interesting question and I should look into that. => "using hyperparameter performances as a measure of the quality of a synthetic dataset."

6. Can I do more advanced recalibration strategies if the data is not IID? (e.g. time series data)

->->-> After those are done its time to look into the strategy part where I look how those CDE/CP can be practically used.

Bibliography

Abdar, Moloud et al. (2021). "A review of uncertainty quantification in deep learning: Techniques, applications and challenges". In: *Information fusion* 76, pp. 243–297.

Alessandretti, Laura et al. (2018). "Anticipating cryptocurrency prices using machine learning". In: *Complexity* 2018, pp. 1–16.

Ambrogioni, Luca et al. (2017). *The Kernel Mixture Network: A Nonparametric Method for Conditional Density Estimation of Continuous Random Variables*. arXiv: 1705.07111 [stat.ML].

Angelopoulos, Anastasios N and Stephen Bates (2021). "A gentle introduction to conformal prediction and distribution-free uncertainty quantification". In: *arXiv preprint arXiv:2107.07511*.

Auer, Andreas et al. (2024). "Conformal prediction for time series with Modern Hopfield Networks". In: *Advances in Neural Information Processing Systems* 36.

Balasubramanian, Vineeth, Shen-Shyang Ho, and Vladimir Vovk (2014). *Conformal prediction for reliable machine learning: theory, adaptations and applications*. Newnes.

Barber, David and Christopher M Bishop (1998). "Ensemble learning in Bayesian neural networks". In: *Nato ASI Series F Computer and Systems Sciences* 168, pp. 215–238.

Bishop, Christopher M (1994). "Mixture density networks". In.

Chernozhukov, Victor, Kaspar Wüthrich, and Yinchu Zhu (2021). "Distributional conformal prediction". In: *Proceedings of the National Academy of Sciences* 118.48, e2107794118.

Chung, Youngseog et al. (2020). "Beyond Pinball Loss: Quantile Methods for Calibrated Uncertainty Quantification". In: *arXiv preprint arXiv:2011.09588*.

– (2021). "Beyond pinball loss: Quantile methods for calibrated uncertainty quantification". In: *Advances in Neural Information Processing Systems* 34, pp. 10971–10984.

Csillag, Daniel et al. (2023). "AmnioML: amniotic fluid segmentation and volume prediction with uncertainty quantification". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 37. 13, pp. 15494–15502.

Gal, Yarin and Zoubin Ghahramani (20–22 Jun 2016). "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning". In: *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, pp. 1050–1059. URL: https://proceedings.mlr.press/v48/gal16.html.

Gawlikowski, Jakob et al. (2023). "A survey of uncertainty in deep neural networks". In: *Artificial Intelligence Review* 56.Suppl 1, pp. 1513–1589.

Ghesu, Florin C et al. (2021). "Quantifying and leveraging predictive uncertainty for medical image assessment". In: *Medical Image Analysis* 68, p. 101855.

Gneiting, Tilmann, Fadoua Balabdaoui, and Adrian E Raftery (2007). "Probabilistic forecasts, calibration and sharpness". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 69.2, pp. 243–268.

Gupta, Chirag, Arun K Kuchibhotla, and Aaditya Ramdas (2022). "Nested conformal prediction and quantile out-of-bag ensemble methods". In: *Pattern Recognition* 127, p. 108496.

Harris, Charles R. et al. (Sept. 2020). "Array programming with NumPy". In: *Nature* 585.7825, pp. 357–362. DOI: 10.1038/s41586-020-2649-2. URL: https://doi.org/10.1038/s41586-020-2649-2.

Hassanpour, Hamid (2023). "Evaluation of deep neural network in directional prediction of Forex market". In: *Authorea Preprints*.

Hornik, Kurt, Maxwell Stinchcombe, and Halbert White (1989). "Multilayer feedforward networks are universal approximators". In: *Neural Networks* 2.5, pp. 359–366. ISSN: 0893-6080. DOI: https://doi.org/10.1016/0893-6080(89)90020-8. URL: https://www.sciencedirect.com/science/article/pii/0893608089900208.

Hüllermeier, Eyke and Willem Waegeman (Mar. 2021). "Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods". en. In: *Machine Learning* 110.3, pp. 457–506. ISSN: 1573-0565. DOI: 10.1007/s10994-021-05946-3. URL: https://doi.org/10.1007/s10994-021-05946-3 (visited on 10/18/2023).

Hyndman, Rob J (1996). "Computing and graphing highest density regions". In: *The American Statistician* 50.2, pp. 120–126.

Izbicki, Rafael, Gilson Shimizu, and Rafael B Stern (2022). "Cd-split and hpd-split: Efficient conformal regions in high dimensions". In: *Journal of Machine Learning Research* 23.87, pp. 1–32.

Izbicki, Rafael, Gilson T Shimizu, and Rafael B Stern (2019). "Flexible distribution-free conditional predictive bands using density estimators". In: *arXiv preprint arXiv:1910.05575*.

Jorion, Philippe (2007). *Value at risk: the new benchmark for managing financial risk*. McGraw-Hill.

Klenke, Achim (2013). *Probability theory: a comprehensive course*. Springer Science & Business Media.

Klotz, D. et al. (2022). "Uncertainty estimation with deep learning for rainfall–runoff modeling". In: *Hydrology and Earth System Sciences* 26.6, pp. 1673–1693. DOI: 10.5194/hess-26-1673-2022. URL: https://hess.copernicus.org/articles/26/1673/2022/.

Koenker, Roger and Gilbert Bassett Jr (1978). "Regression quantiles". In: *Econometrica: journal of the Econometric Society*, pp. 33–50.

Kvamme, Håvard, Ørnulf Borgan, and Ida Scheel (2019). "Time-to-event prediction with neural networks and Cox regression". In: *Journal of machine learning research* 20.129, pp. 1–30.

Lambert, Benjamin et al. (2024). "Trustworthy clinical AI solutions: a unified review of uncertainty quantification in deep learning models for medical image analysis". In: *Artificial Intelligence in Medicine*, p. 102830.

Loftus, Tyler J et al. (2022). "Uncertainty-aware deep learning in healthcare: a scoping review". In: *PLOS digital health* 1.8, e0000085.

Mashrur, Akib et al. (2020). "Machine learning for financial risk management: a survey". In: *Ieee Access* 8, pp. 203203–203223.

Moon, Sang Jun et al. (2021). "Learning multiple quantiles with neural networks". In: *Journal of Computational and Graphical Statistics* 30.4, pp. 1238–1248.

Neal, Radford M (2012). *Bayesian learning for neural networks*. Vol. 118. Springer Science & Business Media.

Oliveira, Roberto I et al. (2022). "Split conformal prediction for dependent data". In: *arXiv preprint arXiv:2203.15885*.

Papadopoulos, Harris (2008). "Inductive Conformal Prediction: Theory and Application to Neural Networks". In: *Tools in Artificial Intelligence*. Ed. by Paula Fritzsche. Rijeka: IntechOpen. Chap. 18. DOI: 10.5772/6078. URL: https://doi.org/10.5772/6078.

Paszke, Adam et al. (2019). "Pytorch: An imperative style, high-performance deep learning library". In: *Advances in neural information processing systems* 32.

Pedregosa, F. et al. (2011). "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12, pp. 2825–2830.

Prasad, Venkatavara et al. (2023). "Tumor size estimation and 3D model viewing using Deep Learning". In.

Romano, João Vitor (2022). "Conformal Prediction Methods in Finance". In.

Romano, Yaniv, Evan Patterson, and Emmanuel Candes (2019). "Conformalized quantile regression". In: *Advances in neural information processing systems* 32.

Rothfuss, Jonas, Fabio Ferreira, Simon Boehm, et al. (2019). "Noise regularization for conditional density estimation". In: *arXiv preprint arXiv:1907.08982*.

Rothfuss, Jonas, Fabio Ferreira, Simon Walther, et al. (2019). *Conditional Density Estimation with Neural Networks: Best Practices and Benchmarks*. arXiv: 1903.00954 [stat.ML].

Schweighofer, Kajetan et al. (2023). *Quantification of Uncertainty with Adversarial Models*. arXiv: 2307.03217 [cs.LG].

Sesia, Matteo and Emmanuel J Candès (2020). "A comparison of some conformal quantile regression methods". In: *Stat* 9.1, e261.

Sesia, Matteo and Yaniv Romano (2021). "Conformal prediction using conditional histograms". In: *Advances in Neural Information Processing Systems* 34, pp. 6304–6315.

Shafer, Glenn and Vladimir Vovk (2008). "A Tutorial on Conformal Prediction." In: *Journal of Machine Learning Research* 9.3.

Singh, Ritika and Shashi Srivastava (2017). "Stock prediction using deep learning". In: *Multimedia Tools and Applications* 76, pp. 18569–18584.

Sloma, Michael et al. (2021). "Empirical comparison of continuous and discrete-time representations for survival prediction". In: *Survival Prediction-Algorithms, Challenges and Applications*. PMLR, pp. 118–131.

team, The pandas development (Feb. 2020). *pandas-dev/pandas: Pandas*. Version latest. DOI: 10.5281/zenodo.3509134. URL: https://doi.org/10.5281/zenodo.3509134.

Trippe, Brian L and Richard E Turner (2018). *Conditional Density Estimation with Bayesian Normalising Flows*. arXiv: 1802.04908 [stat.ML].

Wolpert, David H and William G Macready (1997). "No free lunch theorems for optimization". In: *IEEE transactions on evolutionary computation* 1.1, pp. 67–82.

Wolpert, David H. (Oct. 1996). "The Lack of A Priori Distinctions Between Learning Algorithms". In: *Neural Computation* 8.7, pp. 1341–1390. ISSN: 0899-7667. DOI: 10.1162/neco.1996.8.7.1341. eprint: https://direct.mit.edu/neco/article-pdf/8/7/1341/813495/neco.1996.8.7.1341.pdf. URL: https://doi.org/10.1162/neco.1996.8.7.1341.

Wu, Yue et al. (2023). "Application of machine learning in personalized medicine". In: *Intelligent Pharmacy*.

Xia, Yingda et al. (2020). "Uncertainty-aware multi-view co-training for semi-supervised medical image segmentation and domain adaptation". In: *Medical image analysis* 65, p. 101766.