Annals of Mathematics and Artificial Intelligence
https://doi.org/10.1007/s10472-023-09847-0

Conformal Predictive Distribution Trees

Ulf Johansson[1] · **Tuwe Löfström**[1] · **Henrik Boström**[2]

Accepted: 22 March 2023

Abstract
Being able to understand the logic behind predictions or recommendations on the instance level is at the heart of trustworthy machine learning models. Inherently interpretable models make this possible by allowing inspection and analysis of the model itself, thus exhibiting the logic behind each prediction, while providing an opportunity to gain insights about the underlying domain. Another important criterion for trustworthiness is the model's ability to somehow communicate a measure of confidence in every specific prediction or recommendation. Indeed, the overall goal of this paper is to produce highly informative models that combine interpretability and algorithmic confidence. For this purpose, we introduce conformal predictive distribution trees, which is a novel form of regression trees where each leaf contains a conformal predictive distribution. Using this representation language, the proposed approach allows very versatile analyses of individual leaves in the regression trees. Specifically, depending on the chosen level of detail, the leaves, in addition to the normal point predictions, can provide either cumulative distributions or prediction intervals that are guaranteed to be well-calibrated. In the empirical evaluation, the suggested conformal predictive distribution trees are compared to the well-established conformal regressors, thus demonstrating the benefits of the enhanced representation.

Keywords Conformal predictive distributions · Interpretability · Regression trees · Conformal regression

Mathematics Subject Classification (2010) 68T37

✉ Ulf Johansson
ulf.johansson@ju.se

Tuwe Löfström
tuwe.lofstrom@ju.se

Henrik Boström
bostromh@kth.se

[1] Department of Computing, Jönköping University, Jönköping, Sweden

[2] School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden

Published online: 03 June 2023

1 Introduction

Predictive modeling is increasingly used as a basis for decision support, or even automated decision-making. Making the resulting predictions, decisions, and recommendations trustworthy is a key issue, directly impacting user acceptance and, by extension, how the society in general will adopt AI-based systems. Actually, the importance of trustworthy AI is obvious from the *Ethics Guidelines for Trustworthy AI* [1], released by the European Commission's independent High-Level Expert Group on Artificial Intelligence.

One frequently expressed requirement for predictive models deemed to be trustworthy is that it should be possible to understand the logic behind the predictions or recommendations, typically on an individual (local) level, i.e., for each instance, but preferably also globally for the entire model. Consequently, a predictive model needs to be either *interpretable* or *explainable*. Simply put, an interpretable model can be examined and understood "as is", while an explainable model typically uses an external procedure for the actual explanations. In recent years, many advanced algorithms, e.g., the LIME framework [2], have been developed for offering either local or global explanations. The main advantage of these methods is that they can be applied to all models, including opaque, like ensembles or neural networks. This is important since opaque models typically outperform inherently transparent alternatives like rule sets or decision trees [3]. Unfortunately, techniques producing global explanations that are guaranteed to be faithful to the opaque model are generally lacking. In contrast, inherently interpretable models make it possible to understand the exact reasoning used for every single prediction, while simultaneously allowing inspection and analysis of the model to gain insights about the underlying domain.

Another criterion associated with trustworthiness is *algorithmic confidence*, i.e., the models should not only be accurate, but also able to somehow communicate their confidence in every prediction. Obviously, this requires the confidences to be well-calibrated, if this is not the case, they actually become misleading.

In [4], it was argued that an accurate interpretable classifier, capable of distinguishing between predictions where it is certain and not – and communicating this in an exact way – meets most of the criteria for trustworthy predictive models. Specifically, combining probability estimation trees [5] with so-called Venn-Abers predictors [6] was suggested. The end result was a method for producing decision trees, available for inspection and analysis after the calibration, where each leaf contains a specific prediction, consisting of a label and a well-calibrated probability interval. An interesting property of these models is the information conveyed by the size of the probability intervals; where larger intervals show that the model is less certain in its confidence estimations.

While both inherently interpretable models and explanation techniques, including rule extraction (e.g., approximating a strong opaque model with a weaker but transparent) are more common for classification, there exist several alternatives for predictive regression as well. In this paper, we will look at regression tree models where the leafs are both well-calibrated and more informative than just a point prediction. More specifically, we introduce and evaluate *Conformal predictive distribution trees*, where every leaf contains a conformal predictive system [7], i.e., a well-calibrated cumulative probability distribution over the possible target values. We will contrast and compare the suggested approach to using *conformal regression*, which was suggested and evaluated for creating interpretable and informative tree models in [8].

In the next section, we give an overview of conformal regression and conformal prediction systems, as well as a brief summary of related work. In Section 3, we describe the

experimentation and the evaluation used, including the publicly available data sets. Section 4 demonstrates the merits of the suggested approach, starting with a few detailed examples, before presenting aggregated results from a large number of benchmark data sets. Finally, we summarize the main conclusions and suggest some directions for future work in Section 5.

2 Background

2.1 Conformal regression

The conformal prediction framework [9] has the distinguishing characteristic of producing predictions with guarantees on the error rate under minimal assumptions. In fact, all conformal predictors are *valid*, i.e., given a significance level $\epsilon \in (0, 1)$, the error rate of a conformal predictor will, in the long run, be exactly ϵ. Conformal predictors output *prediction regions*; in classification label sets and in regression prediction intervals. A prediction region not containing the true target is considered an error.

Inductive conformal prediction (ICP) can be applied on top of any predictive model (called the *underlying model*), thus turning it into a conformal predictor. For this step, which is performed only once for each model, ICP requires a labeled data set (the *calibration set*) that was not used for the training of the underlying model. After the calibration, the conformal predictor can be used for prediction on novel (test) data, returning valid prediction regions. Technically, the validity of conformal prediction relies on only one assumption, i.e., that the calibration and test sets are *exchangeable*, which is a slightly weaker property than the standard i.i.d. Here it must be noted that if applied on top of an inherently interpretable underlying model, the conformal predictor and the associated prediction regions can, after the calibration step, be inspected and analyzed, typically providing significantly more information than the underlying model.

While the validity, i.e., the bounded error rate, is guaranteed by the framework, the informativeness of the models may vary. Specifically, the sizes of the prediction regions, i.e., the uncertainty exhibited by the conformal predictor, depend on both the quality of the underlying model, and on design choices and parameter values in the ICP step. For regression, which is the focus of this paper, the most important criterion (which in conformal prediction is referred to as *efficiency*) is that the prediction intervals are as tight as possible.

All conformal predictors utilize so-called *nonconformity functions* which are real-valued functions measuring the strangeness of an instance $(\boldsymbol{x}, y)$. For standard conformal regressors, the nonconformity of an instance $(\boldsymbol{x}_i, y_i)$ is simply defined as the absolute error

$$A(\boldsymbol{x}_i, y_i, h) = |y_i - h(\boldsymbol{x}_i)|, \tag{1}$$

where h is the underlying predictive regressor providing real-valued predictions. Formally, ICP constructs a standard conformal regressor as follows:

1. Divide the training set Z_{tr} into two disjoint subsets:
   a proper training set Z_{t} and a calibration set Z_{c}.
2. Train the underlying model h on Z_{t}.
3. Measure the nonconformity (the absolute errors Eq. 1) of the examples in the calibration set Z_{c} to obtain a list of calibration scores $S = \alpha_1, ..., \alpha_q$ sorted in descending order, where $q = |Z_{\text{c}}|$.

A standard ICP produces a valid prediction interval for a test instance $\boldsymbol{x}_{l+1}$ and a specific confidence level ϵ as follows:

1. Obtain a prediction $h(\boldsymbol{x}_{l+1})$.
2. Find the calibration score α_p where $p = \lfloor \epsilon(q+1) \rfloor$.
3. Using the (partial) inverse of the nonconformity function, obtain the largest nonconformity score that is consistent with ϵ, i.e., $A^{-1}(\alpha_p)$. This is the maximum nonconformity score for h and $\boldsymbol{x}_{l+1}$ with confidence $1-\epsilon$.

If the absolute error in Eq. 1 is used as the nonconformity function, the prediction interval for $\boldsymbol{x}_{l+1}$ thus becomes

$$\hat{Y}_{l+1}^{\epsilon} = h\left(\boldsymbol{x}_{l+1}\right) \pm \alpha_p, \tag{2}$$

with the motivation that the probability for the underlying model h to make an absolute prediction error greater than α_p is exactly ϵ. With this procedure, i.e., using Eqs. 1 and 2, it must be noted that the prediction intervals will be of the same size ($2\alpha_p$) for all test instances. To increase the informativeness, however, we would like the conformal regressors to be more *specific* (or *sharp*), i.e., the interval sizes should differ between easier (where the model is more certain) and harder instances.

The way to make conformal regressors specific is by including a *difficulty estimation* of the instances in the nonconformity function. This addition to the procedure, called *normalization*, will result in the conformal regressor producing tighter intervals for easier instances and larger for harder. In addition to making the predictions specific, previous studies, e.g., [10, 11] show that normalization also leads to tighter prediction intervals on average. So, normalization both makes the conformal regressor more efficient and provides additional information on a per-instance basis.

Several ways of performing the difficulty estimation have been proposed. One early alternative was to use an additional model g trained on the residual errors of h see e.g., [10]. Other options, using just the underlying model, include taking the standard deviation of the predicted values from the members of an ensemble [11] or looking at the spread of true target values in each leaf of a regression tree [8].

With normalization, the nonconformity of an instance is defined as

$$A\left(\boldsymbol{x}_i, y_i, h\right) = \frac{\left|y_i - h\left(\boldsymbol{x}_i\right)\right|}{\sigma_i + \beta}, \tag{3}$$

where σ_i is the difficulty estimation of $\boldsymbol{x}_i$, and β is a sensitivity parameter where lower values put a greater emphasis on the difficulty estimation, relative to the absolute error.

With the normalized nonconformity function, the valid prediction intervals are calculated like:

$$\hat{Y}_{l+1}^{\epsilon} = h\left(\boldsymbol{x}_{l+1}\right) \pm \alpha_p \left(\sigma_{l+1} + \beta\right) . \tag{4}$$

2.2 Conformal predictive systems

Conformal predictive systems [12] generalize conformal regressors in that they output cumulative distribution functions, referred to as *conformal predictive distributions*, instead of prediction intervals. As will be shown, prediction intervals for specified confidence levels may be derived from such distributions, but there are several other usages. A predictive distribution can be used to obtain a threshold value, such that the probability of the true target falling below (or above) it is larger than a specified probability, e.g., what is the highest interest rate such that the probability of the true value exceeding it is less than 0.01. Conversely, the distribution provides probabilities for that the true target falls below (or above) specified thresholds, e.g., what is the probability that the temperature of the engine does not exceed 220

degrees. This is in rather sharp contrast to the prediction intervals in conformal regressors that, even if the are guaranteed to be valid, do not provide information on how values within and outside the intervals are distributed.

The standard, and computationally efficient, approach to forming inductive (often called *split*) conformal predictive systems [7] is very similar to the standard approach of forming inductive conformal regressors. One small, but important, difference is that the nonconformity scores are calculated by considering actual and not absolute values of the residuals:

$$A\left(\boldsymbol{x}_i, y_i, h\right) = \frac{y_i - h\left(\boldsymbol{x}_i\right)}{\sigma_i + \beta}, \tag{5}$$

where σ_i, $\boldsymbol{x}_i$, and β are defined as before. The prediction for a test instance $\boldsymbol{x_i}$, with estimated difficulty σ_i, then becomes the following cumulative distribution function (conformal predictive distribution):

$$Q(y) = \begin{cases} \frac{n+\tau}{q+1}, \text{ if } y \in \left(C_{(n)}, C_{(n+1)}\right), & \text{for } n \in \{0, ..., q\} \\ \frac{n'-1+(n''-n'+2)\tau}{q+1}, \text{if } y = C_{(n)}, & \text{for } n \in \{1, ..., q\} \end{cases} \tag{6}$$

where $C_{(1)}, \ldots, C_{(q)}$ are obtained from the calibration scores $\alpha_1, \ldots, \alpha_q$, sorted in increasing order:

$$C_{(i)} = h\left(\boldsymbol{x}\right) + \sigma\alpha_i$$

and $C_{(0)} = -\infty$ and $C_{(q+1)} = \infty$. τ is sampled from the uniform distribution $\mathcal{U}(0, 1)$ and its role is to allow the p values of target values to be uniformly distributed. n'' is the highest index such that $y = C_{(n'')}$, while n' is the lowest index such that $y = C_{(n')}$ (in case of ties). For a specific value y, the function returns the estimated probability $\mathcal{P}(Y \leq y)$, where Y is a random variable corresponding to the true target.

Given a conformal predictive distribution, a prediction interval for a chosen significance level ϵ can be obtained by $[C_{\lfloor \epsilon(q+1)/2 \rfloor}, C_{\lceil (1-\epsilon/2)(q+1) \rceil}]$. Similarly, a point prediction corresponding to the median of the distribution can be obtained by $C_{\lceil 0.5(q+1) \rceil}$.

2.3 Related work

Recent work in conformal regression proposes many different improvements to the framework. Two specific examples are adaptive and distribution-free prediction intervals for deep neural networks [13] and distribution-free predictive inference [14]. Other important contributions are the utilization of root-finding approaches to efficiently compute conformal prediction sets [15] and conformal histogram regression, which is able to adapt automatically to skewed data [16]. A very interesting paper is [17], presenting an alternate view on conformal regression based on nested sets. Application papers are also frequent, see e.g., [18, 19].

Conformal predictive systems were introduced fairly recently, originally in a symposium paper [20] presented in 2017, which was extended to a journal paper [12]. One of the most important contributions to the area is the computationally efficient approach called *split conformal predictive systems* [7], which was described in the previous section. Other key contributions include techniques for combining multiple such systems into so-called cross-conformal predictive systems [21] and decision procedures to be used on top of conformal predictive systems [22]. The idea of using out-of-bag predictions for calibration, rather than requiring a separate calibration set, has been transferred from the context of conformal regression [23] also to conformal predictive systems [24]. Finally, the idea of employing Mondrian

conformal prediction, which has been used both in the context of conformal classification [25] and conformal regression [26], was recently proposed also for conformal predictive systems [27]. In that study, it was shown that by forming Mondrian categories by binning the predictions of the underlying model, and forming one conformal predictive system per category, predictive performance, as measured by continuous ranked probability score, was significantly improved compared to using one single conformal predictive system.

3 Method

The overall purpose of the empirical study is to demonstrate the usage of conformal prediction systems when applied to regression trees. We also conduct an outright comparison to conformal regression trees, where the four setups evaluated are:

- **CR**: Standard conformal regression, i.e., using no normalization.
- **CRn**: Conformal regression with normalization.
- **CPS**: Standard conformal predictive systems, i.e., without normalization.
- **CPSn**: Conformal predictive systems, with normalization.

For the normalized settings, σ was set to the standard deviation of the true targets (from the training set) in each leaf. It must be noted that with these setups, the result of the calibration is a fixed and interpetable model that can be inspected and analyzed. For the conformal regressors, every leaf would be associated with a prediction interval, and for the conformal predictive systems, every leaf would contain a cumulative distribution function.

All experimentation was performed using *scikit-learn*. The underlying models were regression trees and all parameters were left at the default values, with the exception of $min\_samples\_leaf$, which was set to 25.

The number of calibration instances was selected as $k \cdot 100 - 1$, where k is the largest number making the calibration set less than $1/3$ of the training data available. For the evaluation, standard 10x10-fold cross-validation was used, so all results reported are averaged over the 100 folds.

The 20 publicly available data sets used in the experimentation range from approximately 1400 to 10000 instances. All but one data set, mg from [28], are from the UCI [29], Delve [30] or KEEL [31] repositories. The data sets are described in Table 1 below, where *#inst.* is the number of instances and *#att.* is the number of input attributes. Before the experimentation, all target values were normalized to [0, 1], thus making comparisons over the data sets easier.

4 Results

4.1 Demonstration of the suggested approach

Before presenting experimental results, we demonstrate the flexibility of conformal predictive systems and the suggested approach. Figure 1 illustrates how different prediction intervals – both one-sided and two-sided – can be chosen using the percentiles, and how the probability distribution can be used to find the probability for the true target being higher or lower than a specific threshold value.

The next part of the demonstration uses the comp-activ (comp) data set, which is a collection of a computer systems activity measures. The data was collected from a Sun Sparcstation 20/712 with 128 Mbytes of memory running in a multi-user university department. Users

Table 1 Data set descriptions

Name	#inst.	#att.	Origin	Name	#inst.	#att.	Origin
Abalone	4177	8	UCI	kin8fh	8192	8	Delve
Airfoil	1503	6	UCI	kin8fm	8192	8	Delve
Bank8fh	8192	8	Delve	kin8nh	8192	8	Delve
Bank8fm	8192	8	Delve	kin8nm	8192	8	Delve
Bank8nh	8192	8	Delve	mg	1385	6	Flake
Bank8nm	8192	8	Delve	puma8fh	8192	8	Delve
Comp	8192	12	Delve	puma8fm	8192	8	Delve
DeltaA	7129	5	KEEL	puma8nh	8192	8	Delve
DeltaE	9517	6	KEEL	puma8nm	8192	8	Delve
Friedm	1200	5	KEEL	wizmir	1460	2	KEEL

would typically be doing a large variety of tasks ranging from accessing the internet, editing files or running very cpu-bound programs. The data was collected continuously on two separate occasions. On both occasions, system activity was gathered every 5 seconds. The final data set is taken from both occasions with equal numbers of observations coming from each collection epoch in random order. The target is the portion of time (%) that CPUs run in user mode. The three features used in our example tree are *sread* - number of system *read* calls per second, *swrite* - number of system *write* calls per second, and *runqsz* - process run queue size [30].

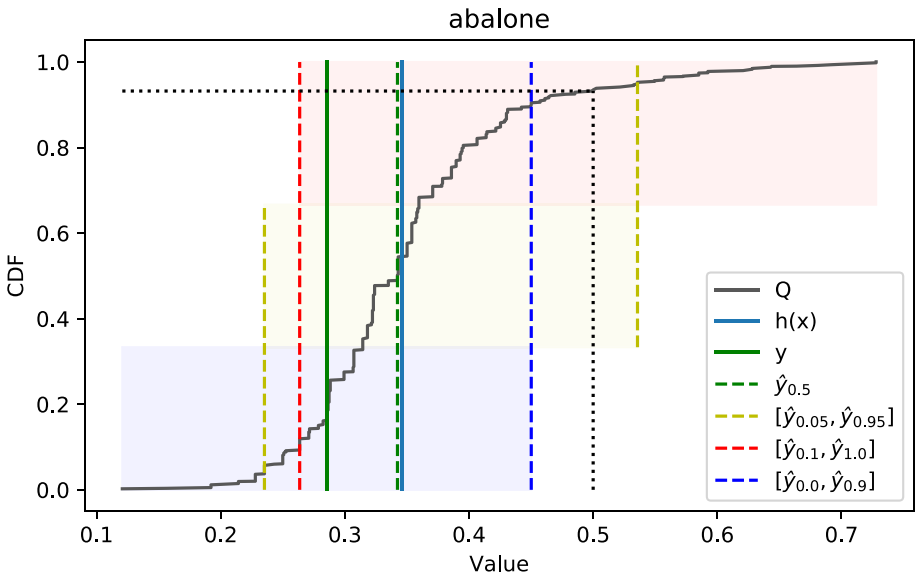


Fig. 1 A Conformal Predictive Distribution with three different intervals defined: red: more than the 10^{th} percentile; yellow: between the 5^{th} and the 95^{th} percentiles; blue: less than the 90^{th} percentile. The black dotted lines indicate how to determine the probability of the true target being smaller than 0.5, which in this case would be 92%

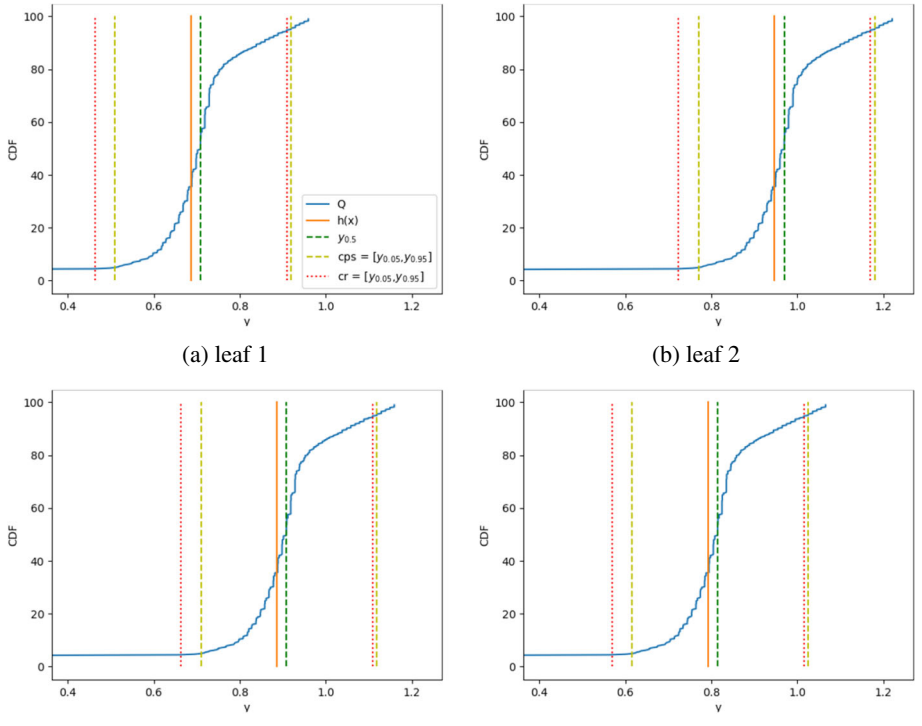
Fig. 2 Regression tree for the comp-active data set

```
runqsz <= 202.50
|   leaf 1: [0.69]
runqsz >  202.50
|   swrite <= 2.10
|   |   sread <= 106.50
|   |   |   leaf 2: [0.95]
|   |   sread >  106.50
|   |   |   leaf 3: [0.89]
|   swrite >  2.10
|   |   leaf 4: [0.79]
```

The data is divided 50/50 into a training and a test set. The training set is further divided into a proper training set (2/3) and a calibration set (1/3). To ensure that the tree is small enough for the demonstration, the *min_samples_leaf* was set to 20%. Figure 2 shows the induced tree used as the underlying model. Obviously, this standard regression tree provides information about the split criteria and the predicted values, but does not provide any further information about the predictions or the associated confidence.

Figure 3 shows the conformal predictive distributions for each of the leaves using standard conformal predictive distributions. Since the tree was forced to be so small, each leaf contains many training and calibration instances, leading to wide intervals and distributions.

Since these conformal predictive distributions are not normalized, all of the distributions are identical, but of course centered around different values in the four leaves. In each subfigure, we have inserted the interval defined by conformal regression using $\epsilon = 0.1$. As a



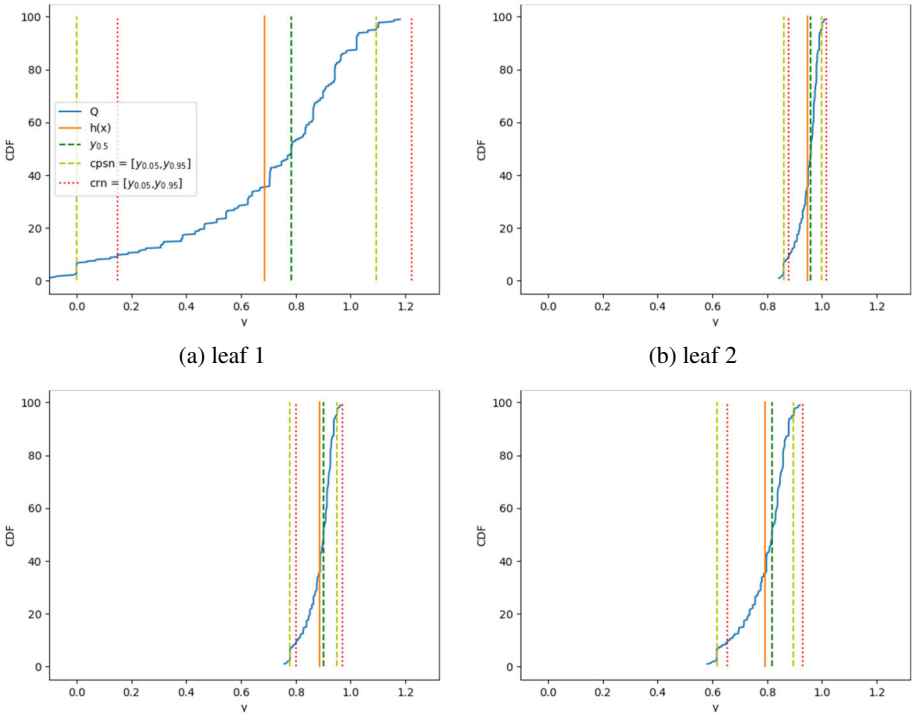
(a) leaf 1 (b) leaf 2

(c) leaf 3 (d) leaf 4

Fig. 3 The conformal predictive distributions for the four different leaves in Fig. 2

comparison, we have also inserted lines for the 5^{th} and 95^{th} percentiles. While the conformal regressor is centered around the prediction $h(x)$, we can see that the corresponding intervals defined from the predictive distribution are adjusted upwards. Furthermore, we can also see that the median of the distribution is higher than the predicted value for the leaf, i.e., the underlying model is most likely underestimating in the predictions. If the conformal predictive distribution is forced to make a point prediction, it would in this case change the prediction upwards, compared to the underlying model.

The normalized conformal predictive distributions for the four leaves are shown in Fig. 4. The most important difference is that the distributions are now adapted to the difficulty of each leaf. Consequently, it is easy to see that both leaves 2 and 3, shown in Fig. 4b and c, have very narrow distributions, providing the user with a very clear picture of what to expect from instances predicted in these leaves. The distribution in leaf 4, shown in Fig. 4d, is also compact, whereas leaf 1, shown in Fig. 4a, is very wide, informing the user that predictions made by that leaf are much more uncertain. The normalized conformal regressors for $\epsilon = 0.1$ are also shown. Interestingly enough, while the intervals defined by the 5^{th} and 95^{th} percentiles are clearly lower than the ones produced by the normalized conformal regressor, the median in the distribution is clearly higher than the point prediction from the underlying model.



(a) leaf 1 (b) leaf 2

(c) leaf 3 (d) leaf 4

Fig. 4 The normalized conformal predictive distributions for the four different leaves in Fig. 2

4.2 Aggregated results

Table 2 shows Mean Absolute Errors (MAE) and tree sizes. The MAE is calculated using:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{7}$$

where $\hat{y}_i$ is the predicted value and y_i the target.

While all setups use the same trees as underlying models, the CPS setups predict the median instead of the tree prediction.

Regarding MAEs, we see that while the differences are very small in absolute numbers, the mean ranks indicate that the CPS variants have slightly smaller errors on most data sets. Looking at the tree sizes, the larger models are arguably too complex to allow for a complete understanding of the underlying relationships. Still, it must be noted that individual leaves could be inspected with ease, and specific predictions analyzed or explained.

As described above, a conformal prediction distribution could provide many different prediction intervals; both one-sided and two-sided. In the following comparison, the most straightforward option is used, i.e., for $\epsilon = 0.1$, the interval is between the 5^{th} and 95^{th} percentiles. Table 3 shows the empirical error rates for $\epsilon = 0.05$ and $\epsilon = 0.1$.

Table 2 MAE and tree size

	MAE			Size
	CPS	CPSn	CR	Tree
Abalone	.058	.057	.058	156
Airfoil	.090	.090	.090	57
Bank8fh	.072	.072	.073	304
Bank8fm	.035	.035	.035	304
Bank8nh	.082	.081	.084	306
Bank8nm	.040	.039	.040	305
Comp	.025	.025	.025	299
DeltaA	.028	.028	.028	264
DeltaE	.042	.042	.042	350
Friedm	.086	.086	.086	47
Kin8fh	.074	.074	.074	301
Kin8fm	.062	.062	.062	300
Kin8nh	.116	.116	.116	304
Kin8nm	.107	.107	.107	305
Mg	.086	.086	.086	52
Puma8fh	.118	.118	.118	305
Puma8fm	.051	.051	.051	302
Puma8nh	.107	.107	.107	305
puma8nm	.046	.046	.046	305
wizmir	.028	.028	.028	55
Mean	**.068**	**.067**	**.068**	**246**
Mean Rank	**2.10**	**1.40**	**2.50**	

Table 3 Error rates

	$\epsilon = 0.05$				$\epsilon = 0.10$			
	CR	CRn	CPS	CPSn	CR	CRn	CPS	CPSn
Abalone	.049	.049	.049	.049	.098	.097	.097	.098
Airfoil	.049	.049	.049	.049	.097	.097	.099	.100
Bank8fh	.050	.050	.050	.050	.099	.099	.100	.100
Bank8fm	.051	.051	.051	.051	.099	.099	.099	.099
Bank8nh	.050	.050	.049	.049	.099	.099	.100	.099
Bank8nm	.052	.051	.052	.052	.100	.101	.101	.101
Comp	.050	.050	.051	.051	.099	.099	.100	.099
DeltaA	.049	.049	.048	.049	.099	.099	.099	.099
DeltaE	.050	.050	.050	.050	.100	.100	.100	.100
Friedm	.050	.052	.049	.049	.098	.100	.102	.102
Kin8fh	.050	.050	.050	.050	.101	.101	.101	.101
Kin8fm	.050	.051	.051	.050	.100	.100	.100	.100
Kin8nh	.049	.050	.049	.049	.101	.100	.101	.101
Kin8nm	.049	.050	.050	.050	.098	.099	.098	.099
Mg	.048	.047	.048	.047	.092	.094	.095	.095
Puma8fh	.049	.049	.049	.049	.099	.099	.099	.100
Puma8fm	.050	.050	.050	.050	.101	.101	.101	.101
Puma8nh	.051	.051	.050	.051	.099	.099	.100	.100
Puma8nm	.050	.050	.050	.050	.100	.099	.100	.100
Wizmir	.053	.053	.052	.051	.096	.097	.100	.099
Mean	**.050**	**.050**	**.050**	**.050**	**.099**	**.099**	**.100**	**.100**

As expected, the observed error rates are very close to the significance level, on each and every data set. While validity is guaranteed for conformal regression and conformal predictive systems, as long as the data set is i.i.d., it is of course important to see that all setups evaluated here are valid not only in theory and in the long run, but also in practice.

Looking finally at the efficiency, Table 4 below shows the mean interval sizes for $\epsilon = 0.05$ and $\epsilon = 0.1$. First of all, it is interesting to see that despite the low significance levels, the intervals are fairly tight. For $\epsilon = 0.1$ and $\epsilon = 0.05$ the intervals cover approximately 28% and 34% of the total range, respectively. When comparing the different setups, there is a clear ordering, showing the importance of the normalization. As a matter of fact, CRn is often the most efficient, followed by CPSn, CR and CPS. Consequently, Friedman tests [32], followed by Bergmann-Hommel's dynamic procedure [33] to establish all pairwise differences at $\alpha = 0.05$, show all these differences to be significant when $\epsilon = 0.05$. For $\epsilon = 0.1$, the only significant differences are between the two normalized and the two standard setups.

While the intervals were marginally larger for the two CPS setups than for the CR counterparts, it must be noted that the intervals used for the comparison were the simplest possible. It would most likely be fairly straightforward to design a heuristics for finding tighter intervals based on the cumulative distributions. Still, it should be remembered that converting the cumulative distribution into an interval was done just for the comparison. In most situations, it is the distribution itself that should be inspected and analyzed.

Table 4 Interval sizes

	$\epsilon = 0.05$				$\epsilon = 0.10$			
	CR	CRn	CPS	CPSn	CR	CRn	CPS	CPSn
Abalone	.351	.326	.357	.332	.263	.246	.271	.255
Airfoil	.469	.449	.471	.450	.390	.377	.386	.372
Bank8fh	.346	.342	.354	.352	.284	.280	.292	.289
Bank8fm	.185	.178	.187	.181	.150	.145	.151	.147
Bank8nh	.366	.358	.413	.414	.294	.286	.325	.325
Bank8nm	.219	.196	.234	.214	.160	.148	.167	.157
Comp	.136	.133	.140	.136	.108	.106	.110	.109
DeltaA	.165	.158	.166	.160	.127	.124	.127	.124
DeltaE	.225	.223	.225	.223	.182	.181	.182	.181
Friedm	.427	.428	.435	.434	.359	.358	.356	.356
Kin8fh	.376	.370	.377	.371	.309	.306	.310	.307
Kin8fm	.315	.309	.315	.309	.259	.255	.259	.256
Kin8nh	.572	.565	.571	.564	.479	.475	.480	.475
Kin8nm	.551	.535	.549	.533	.452	.441	.454	.443
Mg	.486	.447	.491	.450	.393	.360	.390	.357
Puma8fh	.571	.562	.572	.564	.484	.474	.484	.474
Puma8fm	.264	.258	.264	.258	.217	.213	.217	.213
Puma8nh	.545	.525	.547	.527	.454	.439	.454	.439
Puma8nm	.247	.239	.248	.240	.201	.195	.200	.195
Wizmir	.140	.138	.140	.138	.116	.115	.116	.115
Mean	**.348**	**.337**	**.353**	**.343**	**.284**	**.276**	**.286**	**.279**
Mean Rank	**2.95**	**1.15**	**3.80**	**2.10**	**3.25**	**1.45**	**3.50**	**1.80**

Summarizing the experiments, we first showed that the empirical error rates for all setups were very close to the significance levels. When comparing the efficiencies, it was obvious that normalization will not only produce more specific models, but also tighter intervals on average. Finally, the intervals produced from the more informative CPS models, even when using the most straightforward approach, were almost as tight as the ones produced by the conformal regressors.

5 Concluding remarks

We have in this paper introduced conformal predictive distribution trees that combine interpretability with algorithmic confidence to provide highly informative models. As demonstrated, the suggested approach allows very versatile analyses of individual leaves in regression trees. Specifically, a user could be provided with many different, but all valid, prediction intervals; one-sided as well as two-sided. Naturally, the probability distribution can also used to find the probability for the true target being either higher or lower than a specific threshold value.

In the experiments, it was shown that all empirical error rates are very close to the chosen significance levels. Having the validity guarantees from the conformal framework, together

with the inherently interpretable and very versatile representation language, are the key properties of the suggested method.

Finally, the valid prediction intervals produced by the novel regression tree variant were compared to the counterparts in conformal regression trees. Here, the efficiency was found to be comparable, despite the fact that a very straightforward procedure was used to select the intervals from the conformal predictive distribution trees.

For future work, Mondrian CPS, where the guarantees apply locally since a separate calibration set is used for each category, should be applied to tree models using the leaves as the categories. This would potentially produce even more specialized predictions, while providing guarantees for each leaf. Another suggestion is investigating systematic approaches to selecting intervals from the CPS optimizing the efficiency while keeping the validity guarantees.

Funding Open access funding provided by Jönköping University. The authors acknowledge the Swedish Knowledge Foundation, Jönköping University, and the industrial partners for financially supporting the research through the AFAIR project with grant number 20200223, as part of the research and education environment SPARK at Jönköping University, Sweden.

Data Availability The data sets are publicly available. One data set, mg from [28], and the rest from the UCI [29], Delve [30] and KEEL [31] repositories.

Declarations

Conflicts of interest The authors have no competing interests to declare that are relevant to the content of this article.

References

1. The European Commission Independent High-Level Expert Group on Artificial Intelligence: Ethics Guidelines for Trustworthy AI (2019)
2. Ribeiro, M.T., Singh, S., Guestrin, C.: Why should i trust you?: Explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD. KDD'16, pp. 1135–1144. ACM (2016)
3. Freitas, A.A.: A survey of evolutionary algorithms for data mining and knowledge discovery. In: Advances in Evolutionary Computation, Springer (2002)
4. Johansson, U., Löfström, T., Boström, H.: Calibrating probability estimation trees using venn-abers predictors. In: SIAM International Conference on Data Mining, SDM Calgary, Canada, pp. 28–36 (2019)
5. Provost, F., Domingos, P.: Tree induction for probability-based ranking. Mach. Learn. **52**(3), 199–215 (2003)
6. Vovk, V., Petej, I.: Venn-abers predictors. arXiv:1211.0025 (2012)
7. Vovk, V., Petej, I., Nouretdinov, I., Manokhin, V., Gammerman, A.: Computationally efficient versions of conformal predictive distributions. Neurocomputing. **397**, 292–308 (2020)
8. Johansson, U., Linusson, H., Löfström, T., Boström, H.: Interpretable regression trees using conformal prediction. Exp. Syst. Appl. **97**, 394–404 (2018)
9. Vovk, V., Gammerman, A., Shafer, G.: Algorithmic Learning in a Random World. Springer-Verlag, New York Inc (2005)

10. Papadopoulos, H., Haralambous, H.: Neural networks regression inductive conformal predictor and its application to total electron content prediction. In: ICANN. LNCS, vol. 6352, pp. 32–41. Springer (2010)
11. Boström, H., Linusson, H., Löfström, T., Johansson, U.: Accelerating difficulty estimation for conformal regression forests. Ann. Math. Artif. Intell. **81**(1–2), 125–144 (2017)
12. Vovk, V., Shen, J., Manokhin, V., Xie, M.: Nonparametric predictive distributions based on conformal prediction. Mach. Learn. **108**(3), 445–474 (2019)
13. Lei, J., G'Sell, M., Rinaldo, A., Tibshirani, R.J., Wasserman, L.: Distributionfree predictive inference for regression. J. Am. Stat. Assoc. **113**(523), 1094–1111 (2018)
14. Kivaranovic, D., Johnson, K.D., Leeb, H.: Adaptive, distribution-free prediction intervals for deep networks. In: International Conference on Artificial Intelligence and Statistics, pp. 4346–4356. PMLR (2020)
15. Ndiaye, E., Takeuchi, I.: Root-finding approaches for computing conformal prediction set. arXiv:2104.06648 (2021)
16. Sesia, M., Romano, Y.: Conformal histogram regression. arXiv:2105.08747 (2021)
17. Gupta, C., Kuchibhotla, A.K., Ramdas, A.K.: Nested conformal prediction and quantile out-of-bag ensemble methods. arXiv:1910.10562 (2019)
18. Wisniewski, W., Lindsay, D., Lindsay, S.: Application of conformal prediction interval estimations to market makers' net positions. In: Conformal and Probabilistic Prediction and Applications, pp. 285–301. PMLR (2020)
19. Kath, C., Ziel, F.: Conformal prediction interval estimation and applications to day-ahead and intraday power markets. Int. J. Forecast. **37**(2), 777–799 (2021)
20. Vovk, V., Shen, J., Manokhin, V., Xie, M.: Nonparametric predictive distributions based on conformal prediction. In: Conformal and Probabilistic Prediction and Applications, COPA, Stockholm, Sweden. Proceedings of Machine Learning Research, vol. 60, pp. 82–102. PMLR (2017)
21. Vovk, V., Nouretdinov, I., Manokhin, V., Gammerman, A.: Cross-conformal predictive distributions. In: Conformal and Probabilistic Prediction and Applications, COPA 2018, 11-13 June 2018, Maastricht, The Netherlands. Proceedings of Machine Learning Research, vol. 91, pp. 37–51. PMLR (2018)
22. Vovk, V., Bendtsen, C.: Conformal predictive decision making. In: Conformal and Probabilistic Prediction and Applications, COPA. Proceedings of Machine Learning Research, vol. 91, pp. 52–62. PMLR (2018)
23. Johansson, U., Boström, H., Löfström, T., Linusson, H.: Regression conformal prediction with random forests. Mach. Learn. **97**(1–2), 155–176 (2014)
24. Werner, H., Carlsson, L., Ahlberg, E., Boström, H.: Evaluating different approaches to calibrating conformal predictive systems. In: Conformal and Probabilistic Prediction and Applications, COPA. Proceedings of Machine Learning Research, vol. 128, pp. 134–150. PMLR (2020)
25. Löfström, T., Zhao, J., Linnusson, H., Jansson, K.: Predicting adverse drug events with confidence. In: Thirteenth Scandinavian Conference on Artificial Intelligence. IOS Press (2015)
26. Boström, H., Johansson, U.: Mondrian conformal regressors. In: Conformal and Probabilistic Prediction and Applications. Proceedings of Machine Learning Research, vol. 128, pp. 114–133. PMLR (2020)
27. Boström, H., Johansson, U., Löfström, T.: Mondrian conformal predictive distributions. In: Conformal and Probabilistic Prediction and Applications, COPA. Proceedings of Machine Learning Research, vol. 152, pp. 24–38. PMLR (2021)
28. Flake, G.W., Lawrence, S.: Efficient svm regression training with smo. Mach. Learn. **46**(1–3), 271–290 (2002)
29. Bache, K., Lichman, M.: UCI Machine Learning Repository (2013)
30. Rasmussen, C.E., Neal, R.M., Hinton, G., van Camp, D., Revow, M., Ghahramani, Z., Kustra, R., Tibshirani, R.: Delve data for evaluating learning in valid experiments. www.cs.toronto.edu/delve (1996)
31. Alcalá-Fdez, J., Fernández, A., Luengo, J., Derrac, J., García, S.: Keel datamining software tool: Data set repository, integration of algorithms and experimental analysis framework. Multiple-Valued Logic Soft Comput. **17**(2–3), 255–287 (2011)
32. Friedman, M.: The use of ranks to avoid the assumption of normality implicit in the analysis of variance. Journal of American Statistical Association **32**, 675–701 (1937)
33. Bergmann, B., Hommel, G.: Improvements of general multiple test procedures for redundant systems of hypotheses. In: Multiple Hypotheses Testing, pp. 100–115. Springer (1988)