Cross-Conformal Prediction with Ridge Regression

Harris Papadopoulos(✉)

Computer Science and Engineering Department, Frederick University,
7 Y. Frederickou St., Palouriotisa, 1036 Nicosia, Cyprus
h.papadopoulos@frederick.ac.cy

Abstract. Cross-Conformal Prediction (CCP) is a recently proposed approach for overcoming the computational inefficiency problem of Conformal Prediction (CP) without sacrificing as much informational efficiency as Inductive Conformal Prediction (ICP). In effect CCP is a hybrid approach combining the ideas of cross-validation and ICP. In the case of classification the predictions of CCP have been shown to be empirically valid and more informationally efficient than those of the ICP. This paper introduces CCP in the regression setting and examines its empirical validity and informational efficiency compared to that of the original CP and ICP when combined with Ridge Regression.

Keywords: Conformal prediction · Cross-validation · Inductive conformal prediction · Prediction regions · Tolerance regions

1 Introduction

Conformal Prediction (CP) is a machine learning framework for extending conventional machine learning algorithms and producing methods, called Conformal Predictors (CPs), that produce prediction sets satisfying a given level of confidence. These sets are guaranteed to include the correct labels with a probability equal to or higher than the required confidence level. To date many CPs have been developed and have been shown to produce valid and useful in practice set predictions; see e.g. [2,4,5,8,9,11]. The main drawback of the methods developed following the original version of the CP framework is that they are much more computationally inefficient than the conventional algorithms they are based on. A modification of the framework, called Inductive Conformal Prediction (ICP), overcomes this problem by using different parts of the training set for the two stages of the CP process. This however, has the undesirable side-effect of losing some of the informational efficiency of the original version of the framework. That is the resulting prediction sets are larger than those produced by the original framework.

In an effort to get the best of both worlds a new modification of the framework was proposed in [12], called Cross-Conformal Prediction (CCP), which combines the idea of ICP with that of cross-validation. In [12] CCP was studied

A. Gammerman et al. (Eds.): SLDS 2015, LNAI 9047, pp. 260–270, 2015.
DOI: 10.1007/978-3-319-17091-6_21

in the Classification setting and was shown to produce empirically valid confidence measures with higher informational efficiency than those of the ICP. This, combined with its comparable to ICP computational efficiency makes CCP the best option when the computational time required by the original CP is too much.

This paper introduces CCP in the case of regression and examines its empirical validity and informational efficiency. The particular method examined in this work is based on the well known Ridge Regression algorithm, which is the first algorithm to which regression CP has been applied and one of the only two algorithms for which the original version of CP was developed (the second algorithm is Nearest Neighbours Regression). However the approach described here can be followed with any regression algorithm. This is actually an additional advantage of CCP and ICP over the original CP as the latter can only be combined with particular algorithms in the case of regression. Furthermore unlike the original CP, CCP can be used with all normalized nonconformity measures (see [7,8]), which were shown to improve the informational efficiency of ICP.

The rest of this paper starts with a description of the CP framework and its inductive version in Section 2. Then Section 3 details the CCP approach and explains the way it can be followed in the case of Regression. Section 4 gives the definition of a normalized nonconformity measure, first proposed in [7] for the Ridge Regression ICP, which was used in the experiments performed. This is followed by the experimental examination of CCP and its comparison with the original CP and ICP in Section 5. Finally Section 6 gives the concluding remarks and future directions of this work.

2 Conformal and Inductive Conformal Prediction

We are given a training set of l observations $\{z_1, \dots, z_l\}$, where each $z_i \in \mathbf{Z}$ is a pair (x_i, y_i) consisting of an object $x_i \in \mathbf{X}$ and an associated label $y_i \in \mathbf{Y}$ (dependent variable). We are also given a new object x_{l+1} and our aim is to produce a prediction set, or region that will contain the correct label y_{l+1} with a predefined confidence with the only assumption that all $z_i \in \mathbf{Z}$ are exchangeable.

CP assumes every possible label $\tilde{y} \in \mathbf{Y}$ of x_{l+1} in turn and calculates the *nonconformity scores*

$$\alpha_i^{\tilde{y}} = A(\{z_1, \dots, z_{i-1}, z_{i+1}, \dots, z_l, z_{l+1}^{\tilde{y}}\}, z_i), \quad i = 1, \dots, l, \tag{1a}$$

$$\alpha_{l+1}^{\tilde{y}} = A(\{z_1, \dots, z_l\}, z_{l+1}^{\tilde{y}}), \tag{1b}$$

where $z_{l+1}^{\tilde{y}} = (x_{l+1}, \tilde{y})$ and $A(S_i, z_j)$ is a numerical score indicating how strange it is for the example z_j to belong to the examples in the set $S_i \subset \mathbf{Z}$. In effect the function A, called the *nonconformity measure* of the CP, uses a conventional machine learning algorithm, called the *underlying algorithm* of the CP, to assess the nonconformity of z_j to the set S_i. The resulting nonconformity scores can then be used to calculate the *p-value* of $\tilde{y}$ as

$$p(z_1, \dots, z_l, z_{l+1}^{\tilde{y}}) = \frac{|\{i = 1, \dots, l : \alpha_i^{\tilde{y}} \geq \alpha_{l+1}^{z_{l+1}^{\tilde{y}}}\}| + 1}{l + 1}, \tag{2}$$

also denoted as $p(\tilde{y})$. The p-value function (2) guarantees that $\forall\delta \in [0,1]$ and for all probability distributions P on $\mathbf{Z}$,

$$P^{\tilde{y}}\{((x_1,y_1),\dots,(x_l,y_l),(x_{l+1},y_{l+1})) : p(y_{l+1}) \le \delta\} \le \delta, \tag{3}$$

where y_{l+1} is the true label of x_{l+1}; a proof can be found in [13]. After calculating the p-value of all possible labels $\tilde{y} \in \mathbf{Y}$, the CP outputs the prediction set, or prediction region (PR) in the case of regression,

$$\{\tilde{y} : p(\tilde{y}) > \delta\}, \tag{4}$$

which has at most δ chance of being wrong, i.e. of not containing y_{l+1}.

Of course in the case of regression it would be impossible to explicitly consider every possible label $\tilde{y} \in \mathrm{I\!R}$. A procedure that makes it possible to compute the PR (4) with Ridge Regression for the standard regression nonconformity measure

$$\alpha_i = |y_i - \hat{y}_i| \tag{5}$$

was proposed in [5]. The same idea was followed in [13] and [8] for the k-nearest neighbours regression algorithm. Still however this approach has two important drawbacks

1. it is very computationally inefficient compared to the algorithm the CP is based on, and
2. it cannot be followed with all regression algorithms.

Inductive Conformal Prediction (ICP) overcomes these problems by dividing the training set into two smaller sets, the *proper training set* with m examples and the *calibration set* with $q = l - m$ examples. The proper training set is then used for training the underlying algorithm of the ICP and only the examples in the calibration set are used for calculating the p-value of each possible classification for every test example. More specifically, ICP calculates the p-value of each possible classification $\tilde{y}$ of x_{l+1} as

$$p(\tilde{y}) = \frac{|\{i = m+1,\dots,m+q : \alpha_i \ge \alpha^{\tilde{y}}_{l+1}\}| + 1}{q+1}, \tag{6}$$

where

$$\alpha_i = A(\{z_1,\dots,z_m\}, z_{m+i}), \quad i = 1,\dots,q, \tag{7a}$$

$$\alpha^{\tilde{y}}_{l+1} = A(\{z_1,\dots,z_m\}, z^{\tilde{y}}_{l+1}), \tag{7b}$$

and $z^{\tilde{y}}_{l+1} = (x_{l+1}, \tilde{y})$.

As with the original CP approach in the case of regression it is impossible to explicitly consider every possible label $\tilde{y} \in \mathrm{I\!R}$ and calculate its p-value. However, in this case both the nonconformity scores of the calibration set examples $\alpha_{m+1},\dots,\alpha_{m+q}$ and the underlying algorithm prediction $\hat{y}_{l+1}$ are not affected by the value of $\tilde{y}$, only the nonconformity score α_{l+1} is affected. Therefore $p(\tilde{y})$

changes only at the points where $\alpha^{\tilde{y}}_{l+1} = \alpha_i$ for some $i = m+1, \dots, m+q$. As a result, for a confidence level $1 - \delta$ we only need to find the biggest α_i such that when $\alpha^{\tilde{y}}_{l+1} = \alpha_i$ then $p(\tilde{y}) > \delta$, which will give us the maximum and minimum $\tilde{y}$ that have a p-value bigger than δ and consequently the beginning and end of the corresponding PR. More specifically, we sort the nonconformity scores of the calibration examples in descending order obtaining the sequence

$$\alpha_{(m+1)}, \dots, \alpha_{(m+q)}, \tag{8}$$

and output the PR

$$(\hat{y}_{l+1} - \alpha_{(m+s)}, \hat{y}_{l+1} + \alpha_{(m+s)}), \tag{9}$$

where

$$s = \lfloor \delta(q+1) \rfloor. \tag{10}$$

3 Cross-Conformal Prediction for Regression

As ICP does not include the test example in the training set of its underlying algorithm, the latter needs to be trained only once and in the case of regression the approach can be combined with any underlying algorithm. However the fact that it only uses part of the training set for training its underlying algorithm and for calculating its p-values results in lower informational efficiency, i.e. looser PRs. Cross-Conformal Prediction, which was recently proposed in [12], tries to overcome this problem by combining ICP with cross-validation. Specifically, CCP partitions the training set in K subsets (folds) $S_1, \dots, S_K$ and calculates the nonconformity scores of the examples in each subset S_k and of $(x_{l+1}, \tilde{y})$ for each possible label $\tilde{y}$ as

$$\alpha_i = A(\cup_{m \neq k} S_m, z_i), \quad z_i \in S_k, \quad m = 1, \dots, K, \tag{11a}$$

$$\alpha^{\tilde{y},k}_{l+1} = A(\cup_{m \neq k} S_m, z^{\tilde{y}}_{l+1}), \quad m = 1, \dots, K, \tag{11b}$$

where $z^{\tilde{y}}_{l+1} = (x_{l+1}, \tilde{y})$. Note that for $z^{\tilde{y}}_{l+1}$ K nonconformity scores $\alpha^{\tilde{y},k}_{l+1}$, $k = 1, \dots, K$ are calculated, one with each of the K folds. Now the p-value for each possible label $\tilde{y}$ is computed as

$$p(\tilde{y}) = \frac{\sum_{k=1}^{K} |\{z_i \in S_k : \alpha_i \geq \alpha^{\tilde{y},k}_{l+1}\}| + 1}{l+1}. \tag{12}$$

Again in the case of regression the possible labels $\tilde{y} \in \mathbb{R}$ are infinite. Still though, like in the case of the ICP, only the nonconformity score of z_{l+1} is affected by changes to the value of $\tilde{y}$. As a result $p(\tilde{y})$ can change only at the values of $\tilde{y}$ for which $\alpha^{\tilde{y}}_{l+1} = \alpha_i$ for some $i = 1, \dots, l$. Note that in this case however we have K different predictions $\hat{y}^1_{l+1}, \dots, \hat{y}^K_{l+1}$, where $\hat{y}^k_{l+1}$ was produced by training the underlying algorithm on $\cup_{m \neq k} S_m$, $m = 1, \dots, K$. Specifically nonconformity measure (5) in this case for an example $z_i \in S_k$ is actually

$$\alpha_i = |y_i - \hat{y}^k_i|; \tag{13}$$

for $\alpha_{l+1}^{\tilde{y},k}$ replace y_i with $\tilde{y}$. As a result, each $\tilde{y}$ is associated with K nonconformity scores $\alpha_{l+1}^{\tilde{y},1}, \ldots, \alpha_{l+1}^{\tilde{y},K}$ and each $\alpha_{l+1}^{\tilde{y},k}$ is compared with the nonconformity scores of the examples in S_k. So in order to find the values for which $p(\tilde{y}) > \delta$ we map each nonconformity score α_i to the $\tilde{y}$ values for which $\alpha_{l+1}^{\tilde{y}} = \alpha_i$ generating the lists

$$\nu_i = \hat{y}_{l+1}^k - \alpha_i, \quad z_i \in S_k, \quad i = 1, \ldots, l, \tag{14a}$$

$$\xi_i = \hat{y}_{l+1}^k + \alpha_i, \quad z_i \in S_k, \quad i = 1, \ldots, l. \tag{14b}$$

Now $p(\tilde{y})$ changes only at the points where $\tilde{y} = \nu_i$ or $\tilde{y} = \xi_i$ for some $i = 1, \ldots, l$. So if $\tilde{y}_1, \ldots, \tilde{y}_{2l}$ are the values $\nu_1, \ldots, \nu_l, \xi_1, \ldots, \xi_l$ sorted in ascending order and if $\tilde{y}_0 = -\infty$ and $\tilde{y}_{2l+1} = \infty$, then $p(\tilde{y})$ remains constant in each interval $(\tilde{y}_0, \tilde{y}_1), (\tilde{y}_1, \tilde{y}_2), \ldots, (\tilde{y}_{2l}, \tilde{y}_{2l+1})$.

The p-value in each interval $(\tilde{y}_i, \tilde{y}_{i+1})$ can be calculated as

$$p_i = \frac{|\{i = 1, \ldots, l : \nu_i \leq \tilde{y}_i\}| - |\{i = 1, \ldots, l : \xi_i \leq \tilde{y}_i\}| + 1}{l + 1}. \tag{15}$$

Consequently for any confidence level $1 - \delta$ the resulting PR is:

$$\bigcup_{i : p_i > \delta} [\tilde{y}_i, \tilde{y}_{i+1}]. \tag{16}$$

Note that these PRs may have ‘holes’ in them. This however should happen very rarely, if ever, for the low values of δ we are interested in.

Although CCP needs to train its underlying algorithm K times as opposed to just one for ICP, it is still much more computationally efficient than the original CP. It also can be combined with any underlying algorithm in the case of regression. In comparison with ICP it generates its PRs based on a much richer set of nonconformity scores, resulting from all training examples rather than just the calibration examples. Furthermore, in most cases (depending on K) the underlying algorithm of the CCP is trained on a larger training set compared to the proper training set of the ICP.

4 Normalized Nonconformity Measures

In addition to the typical nonconformity measure (5) some additional nonconformity measures for regression have been proposed in [6–8] for Ridge Regression, Nearest Neighbours Regression and Neural Networks Regression. These measures normalize (5) by the expected accuracy of the underlying algorithm being used. The intuition behind this is that if two examples have the same nonconformity score as defined by (5) and the prediction $\hat{y}$ of the underlying algorithm for one of them was expected to be more accurate than the other, then the former is actually less conforming than the latter. This leads to PRs that are larger for the examples which are more difficult to predict and smaller for the examples which are easier to predict.

Almost all such nonconformity measures however, cannot be used in conjunction with TCP, with the exception of only two out of the six such measures proposed in [8]. This is an additional advantage of CCP over TCP, since the former, like ICP, can be combined with any nonconformity measure.

As this work focuses on Ridge Regression (RR) as underlying algorithm, this section will describe the nonconformity measure proposed in [7] for this algorithm. This measure is defined as

$$\alpha_i = \frac{|y_i - \hat{y}_i|}{\exp(\mu_i)}, \tag{17}$$

where μ_i is the RR prediction of the value $\ln(|y_i - \hat{y}_i|)$. Specifically, after training the RR algorithm on the training set we calculate the residuals $\ln(|y_j - \hat{y}_j|)$ for all training examples and train a linear RR on the pairs $(x_j, \ln(|y_j - \hat{y}_j|))$. The use of the logarithmic instead of the direct scale ensures that the estimate is always positive.

The resulting PRs in the case of CCP can be generated by calculating ν_i and ξ_i as

$$\nu_i = \hat{y}^k_{l+1} - \alpha_i \exp(\mu_i), \quad z_i \in S_k, \quad i = 1, \dots, l, \tag{18a}$$

$$\xi_i = \hat{y}^k_{l+1} + \alpha_i \exp(\mu_i), \quad z_i \in S_k, \quad i = 1, \dots, l, \tag{18b}$$

and following the steps described in Section 3.

5 Experiments and Results

Experiments were performed on four benchmark data sets of different sizes from the UCI [1] and DELVE [10] repositories: Boston Housing, Concrete Compressive Strength, Abalone and Pumadyn (the 8nm variant). Table 1 lists the number of examples and attributes comprising each data set together with the width of the range of its labels. The aim was to examine the validity and informational efficiency of Cross-Conformal Prediction with Ridge Regression as underlying algorithm using nonconformity measures (5) and (17). The informational efficiency was assessed in comparison to those of the corresponding original (Transductive) and Inductive Ridge Regression Conformal Predictors [5,7] under exactly the same setting.

Evaluation was performed on the results obtained from 10 random runs of a cross-validation process. Based on their sizes the two smaller data sets (Boston Housing and Concrete Compressive Strength) were split into 10 folds, whereas the Abalone data set was split into 5 folds and the Pumadyn data set was split into 2 folds; this cross-validation process was for generating the training and test sets the algorithms were evaluated on, not to be confused with the internal cross-validation of CCP. The input attributes of each data set were normalized setting their mean value to 0 and their standard deviation to 1. An RBF kernel was used in the Ridge Regression underlying algorithm, while the kernel parameter (σ) and ridge factor (a) were optimized on each data set

Table 1. Main characteristics and experimental setup for each data set

	Housing	Concrete	Abalone	Pumadyn
Examples	506	1030	4177	8192
Attributes	13	8	8	8
Label range	45	80.27	28	21.17
Folds (evaluation)	10	10	5	2
Calibration size	99	299	1099	1299

Table 2. PR tightness and empirical validity on the Boston Housing data set

Method/		Mean Width			Median Width			Errors (%)		
Measure		90%	95%	99%	90%	95%	99%	90%	95%	99%
ICP	(5)	9.199	12.435	28.469	9.194	12.170	25.743	10.32	5.24	0.97
	(17)	9.099	12.178	26.804	9.037	11.790	24.556	9.82	5.38	1.15
CCP $K = 5$	(5)	9.511	12.573	25.713	9.351	12.420	25.241	8.10	3.52	0.75
	(17)	9.385	12.220	24.936	9.143	11.913	24.457	7.92	3.70	0.61
CCP $K = 10$	(5)	9.044	12.000	24.496	8.991	11.915	24.242	9.19	4.15	0.83
	(17)	8.965	11.583	23.508	8.801	11.336	22.971	9.05	4.23	0.79
CCP $K = 20$	(5)	8.834	11.673	24.194	8.821	11.647	23.909	9.62	4.47	0.85
	(17)	8.742	11.239	22.982	8.592	11.039	22.586	9.64	4.45	0.79
TCP	(5)	10.524	13.448	24.810	7.829	10.036	18.536	9.72	4.80	0.79

by minimizing the radius/margin bound for the first three data sets and (due to its size) the validation error on half the training set for the Pumadyn data set using the gradient descent approach proposed in [3] and the corresponding code provided online[1]. In the case of ICP the calibration set size in was set to $q = 100n - 1$, where n was chosen so that q was the closest value less than one third of the training set; i.e. $n = \lfloor \frac{l}{300} \rfloor$, where l is the training set size. Along with the characteristics of each data set Table 1 gives the number of folds it was split into and calibration set size q used with ICP.

In order to assess the informational efficiency of the Ridge Regression Cross-Conformal Predictor (RR-CCP) Tables 2–5 report the mean and median widths of the PRs produced by RR-CCP with K set to 5, 10 and 20 for the 90%, 95% and 99% confidence levels, along with those produced by the original Transductive version of CP (TCP) and those produced by ICP for each data set. In the case of CCP and ICP the widths obtained with both nonconformity measures (5) and (17) are reported. However with TCP only nonconformity measure (5) can be used. The same Tables report the percentage of errors made by each method

[1] The code is located at http://olivier.chapelle.cc/ams/

Table 3. PR tightness and empirical validity on the Concrete data set

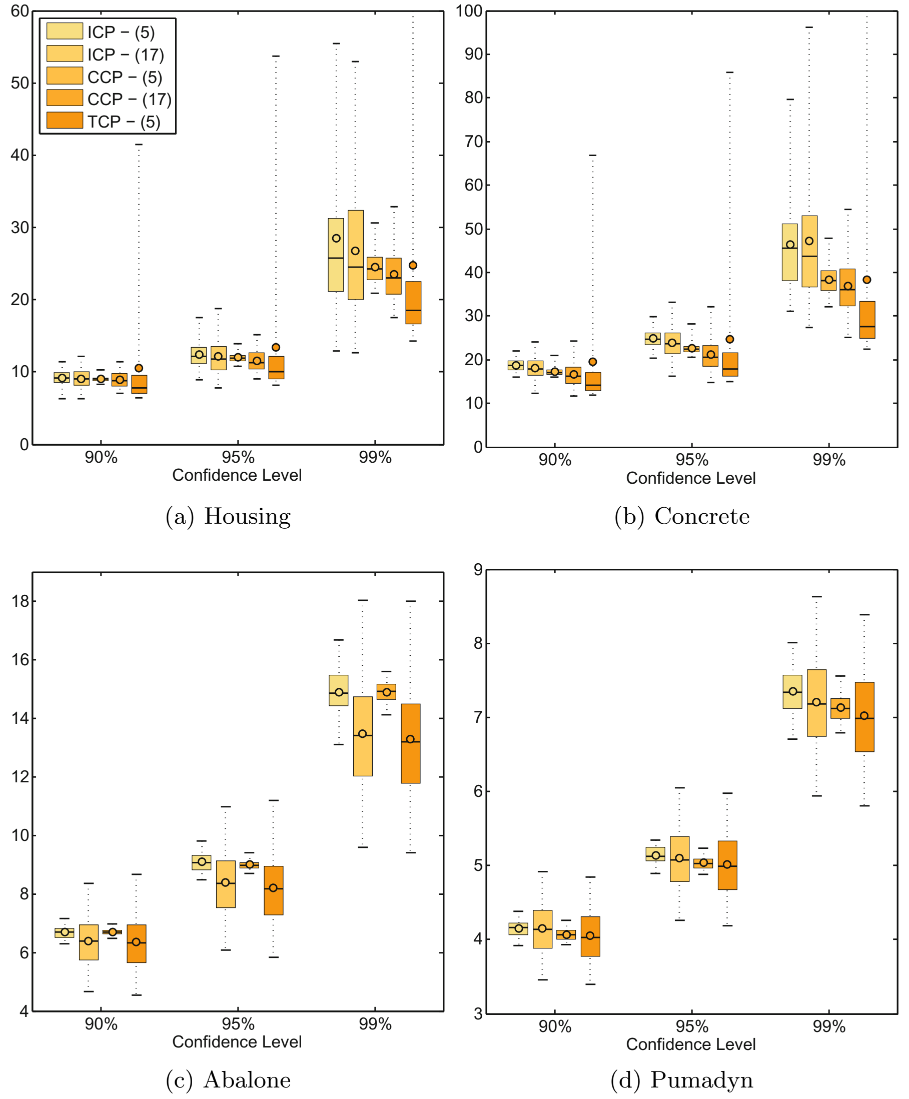
Method/		Mean Width			Median Width			Errors (%)		
Measure		90%	95%	99%	90%	95%	99%	90%	95%	99%
ICP	(5)	18.720	24.879	46.303	18.650	24.784	45.503	9.94	5.03	0.94
	(17)	18.079	23.853	47.210	17.921	23.570	43.691	9.81	4.87	0.92
CCP $K = 5$	(5)	18.516	24.251	41.612	18.082	23.776	41.154	7.06	2.97	0.45
	(17)	17.910	22.990	40.094	17.305	22.236	39.089	7.07	2.55	0.50
CCP $K = 10$	(5)	17.324	22.643	38.411	17.061	22.319	38.134	8.46	3.68	0.54
	(17)	16.675	21.231	36.954	16.255	20.579	36.078	8.39	3.53	0.61
CCP $K = 20$	(5)	16.780	21.878	37.289	16.706	21.700	36.795	9.10	4.14	0.65
	(17)	16.142	20.408	35.610	15.753	19.866	34.752	9.33	4.33	0.76
TCP	(5)	19.513	24.593	38.254	14.176	17.881	27.632	9.82	5.15	0.97

Table 4. PR tightness and empirical validity on the Abalone data set

Method/		Mean Width			Median Width			Errors (%)		
Measure		90%	95%	99%	90%	95%	99%	90%	95%	99%
ICP	(5)	6.700	9.090	14.883	6.682	9.064	14.848	10.06	4.99	1.02
	(17)	6.385	8.380	13.469	6.379	8.370	13.405	10.10	5.01	1.01
CCP $K = 5$	(5)	6.750	9.038	14.961	6.721	9.017	14.983	9.43	4.73	0.93
	(17)	6.425	8.307	13.437	6.404	8.275	13.388	9.48	4.63	0.93
CCP $K = 10$	(5)	6.698	8.994	14.901	6.684	8.980	14.919	9.67	4.85	0.96
	(17)	6.368	8.210	13.276	6.330	8.160	13.193	9.71	4.73	0.96
CCP $K = 20$	(5)	6.673	8.971	14.875	6.660	8.971	14.883	9.80	4.92	0.98
	(17)	6.343	8.152	13.172	6.298	8.088	13.082	9.82	4.87	0.97

Table 5. PR tightness and empirical validity on the Pumadyn data set

Method/		Mean Width			Median Width			Errors (%)		
Measure		90%	95%	99%	90%	95%	99%	90%	95%	99%
ICP	(5)	4.153	5.139	7.355	4.159	5.128	7.337	10.17	5.10	1.02
	(17)	4.148	5.099	7.210	4.137	5.076	7.186	10.16	5.14	1.00
CCP $K = 5$	(5)	4.182	5.167	7.312	4.162	5.148	7.319	8.57	4.11	0.76
	(17)	4.174	5.146	7.196	4.145	5.118	7.163	8.59	3.96	0.69
CCP $K = 10$	(5)	4.070	5.033	7.138	4.064	5.026	7.121	9.26	4.55	0.90
	(17)	4.053	5.012	7.017	4.032	4.990	6.990	9.32	4.43	0.83
CCP $K = 20$	(5)	4.020	4.974	7.049	4.007	4.964	7.025	9.56	4.76	0.98
	(17)	4.006	4.944	6.926	3.986	4.925	6.901	9.66	4.68	0.91



(a) Housing

(b) Concrete

(c) Abalone

(d) Pumadyn

Fig. 1. PR width distribution

for every confidence level to evaluate the empirical validity of the corresponding PRs.

Figure 1 complements the information reported in Tables 2-5 by displaying boxplots for each data set showing the median, upper and lower quartiles and 2nd and 98th percentiles of the PR widths produced by each method for the three confidence levels. The mean widths are also marked with a circle. For CCP the widths obtained with $K = 10$ were used here. In the case of the TCP with the

99% confidence level the 98th percentile of the obtained widths extends much higher than the maximum value displayed in the plots.

In terms of empirical validity the error percentages displayed in Tables 2–5 demonstrate that the error rates of the PRs produced by CCP are always lower than the required significance level. The change of K, at least up to $K = 20$, does not seem to affect validity. In many cases the PRs of CCP seem more conservative than those of the two other methods, since the error percentages of the latter are nearer to the corresponding significance levels. This suggests that there might be room for further improvement.

By comparing the mean and median widths of the PRs produced by CCP with the three different values of K, it seems that increasing K improves informational efficiency. However it is not clear if much larger values of K will still have the same effect. Also the extent of improvement is not always significant. In comparing the widths of the PRs produced by CCP with those of the ICP with the same nonconformity measure, one can see that in all cases, with the exception of the abalone data set with nonconformity measure (5), CCP with K set to 10 and 20 gives overall tighter PRs. In the case of the TCP it seems that while most of its PRs are tighter than those of the CCP and ICP, some are extremely loose and result in a mean width that is higher than that of the CCP.

6 Conclusion

This work introduced Cross-Conformal Prediction in the case of regression and examined its empirical validity and informational efficiency when combined with Ridge Regression as underlying algorithm. CCP does not suffer from the computational inefficiency problem of the original CP approach while it makes a more effective use of the available training data than ICP. Additionally, on the contrary to the original CP it can be combined with any conventional regression algorithm and any regression nonconformity measure.

The experimental results presented show that the PRs produced by CCP are empirically valid while being tighter than those of the ICP. Although the PRs of the original CP approach are tighter for the majority of cases, for some examples they become extremely loose, thing that does not happen with CCP.

A first future direction of this work is to examine the empirical validity and performance of CCP when using one-sided nonconformity measures and combining the resulting upper and lower prediction rays to obtain the overall PR. This makes the computation of the PRs simpler and it is interesting to see if the resulting PRs are also tighter. Moreover it would be interesting to study how the performance of CCP is affected in comparison to that of the TCP and ICP as the size of the data set in question increases. Finally the application of CCP to real world problems where the provision of valid PRs is desirable is an important future aim.

References

1. Bache, K., Lichman, M.: UCI machine learning repository (2013). http://archive.ics.uci.edu/ml
2. Bhattacharyya, S.: Confidence in predictions from random tree ensembles. In: Proceedings of the 11th IEEE International Conference on Data Mining (ICDM 2011), pp. 71–80. Springer (2011)
3. Chapelle, O., Vapnik, V., Bousquet, O., Mukherjee, S.: Choosing multiple parameters for support vector machines. Machine Learning **46** (2002)
4. Lambrou, A., Papadopoulos, H., Gammerman, A.: Reliable confidence measures for medical diagnosis with evolutionary algorithms. IEEE Transactions on Information Technology in Biomedicine **15**(1), 93–99 (2011)
5. Nouretdinov, I., Melluish, T., Vovk, V.: Ridge regression confidence machine. In: Proceedings of the 18th International Conference on Machine Learning (ICML 2001), pp. 385–392. Morgan Kaufmann, San Francisco (2001)
6. Papadopoulos, H., Haralambous, H.: Reliable prediction intervals with regression neural networks. Neural Networks **24**(8), 842–851 (2011). http://dx.doi.org/10.1016/j.neunet.2011.05.008
7. Papadopoulos, H., Proedrou, K., Vovk, V., Gammerman, A.: Inductive confidence machines for regression. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) ECML 2002. LNCS (LNAI), vol. 2430, pp. 345–356. Springer, Heidelberg (2002)
8. Papadopoulos, H., Vovk, V., Gammerman, A.: Regression conformal prediction with nearest neighbours. Journal of Artificial Intelligence Research **40**, 815–840 (2011). http://dx.doi.org/10.1613/jair.3198
9. Proedrou, K., Nouretdinov, I., Vovk, V., Gammerman, A.: Transductive confidence machines for pattern recognition. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) ECML 2002. LNCS (LNAI), vol. 2430, pp. 381–390. Springer, Heidelberg (2002)
10. Rasmussen, C.E., Neal, R.M., Hinton, G.E., Van Camp, D., Revow, M., Ghahramani, Z., Kustra, R., Tibshirani, R.: DELVE: Data for evaluating learning in valid experiments (1996). http://www.cs.toronto.edu/delve/
11. Saunders, C., Gammerman, A., Vovk, V.: Transduction with confidence and credibility. In: Proceedings of the 16th International Joint Conference on Artificial Intelligence, vol. 2, pp. 722–726. Morgan Kaufmann, Los Altos (1999)
12. Vovk, V.: Cross-conformal predictors. Annals of Mathematics and Artificial Intelligence (2013). http://dx.doi.org/10.1007/s10472-013-9368-4
13. Vovk, V., Gammerman, A., Shafer, G.: Algorithmic Learning in a Random World. Springer, New York (2005)