Conformal multi-target regression using neural networks

Soundouss Messoudi SOUNDOUSS.MESSOUDI@HDS.UTC.FR
Sébastien Destercke SEBASTIEN.DESTERCKE@HDS.UTC.FR
Sylvain Rousseau SYLVAIN.ROUSSEAU@HDS.UTC.FR
HEUDIASYC - UMR CNRS 7253, Université de Technologie de Compiègne, 57 avenue de Landshut, 60203 COMPIEGNE CEDEX - FRANCE

Editor: Alexander Gammerman, Vladimir Vovk, Zhiyuan Luo, Evgueni Smirnov and Giovanni Cherubin

Abstract

Multi-task learning is a domain that is still not fully studied in the conformal prediction framework, and this is particularly true for multi-target regression. Our work uses inductive conformal prediction along with deep neural networks to handle multi-target regression by exploring multiple extensions of existing single-target non-conformity measures and proposing new ones. This paper presents our approaches to work with conformal prediction in the multiple regression setting, as well as the results of our conducted experiments.

Keywords: Inductive conformal prediction - Multi-target regression - Deep neural networks

1. Introduction

Multi-task learning is a natural generalization to single-output supervised learning, where the objective is to predict multiple outputs from the input features characterizing the data set. For instance, multi-label classification focuses on having numerous binary outputs (Zhang and Zhou (2013)) and multi-target regression considers real output values (Borchani et al. (2015)). As we do in single-target tasks, we can improve a multi-output learning algorithm by providing an estimate of the confidence to be placed in its predictions. To achieve this goal, conformal prediction can be applied.

Conformal prediction is a framework that provides partial or set-valued predictions (as a set of labels in the classification case, and as a prediction interval in the case of single-target regression) with a statistical guarantee under the assumption that random variables are exchangeable (a weaker assumption than the i.i.d. one). Initially, conformal prediction was proposed as a transductive online learning method that performs training, learning and prediction simultaneously. Then, an extension to the inductive framework was introduced to learn the model once on the training examples before testing it on future ones. The principle of inductive conformal prediction for regression and its non-conformity measures will be recalled in Section 2.

Conformal prediction was used for the specific multi-task learning that is multi-label classification: Wang et al. (2015) compared three different implementations of multi-label conformal prediction called instance reproduction, binary relevance and power set. However, conformal prediction for multi-target regression is still a largely unexplored area. This paper proposes to examine this aspect of conformal prediction by extending single-output

regression methods in conformal prediction to the multi-target problem, and come up with other non-conformity measures for this problem.

Our work uses inductive conformal prediction applied to multi-target regression. We use neural networks, propose diverse non-conformity measures and conduct experiments on different multi-target regression data sets. Our approach is described in Section 3. The experiments and their results are presented in Section 4, where we try various non-conformity measures inspired from the single-output context of conformal prediction.

2. Conformal prediction

Conformal prediction was originally proposed by Gammerman et al. (1998) as a transductive online learning framework that produces predictions for each new example from the previous examples without generating a prediction rule. This method was then adapted to the inductive approach by Papadopoulos et al. (2002), where a model induced from training instances is used to produce conformal predictions for the new examples. For more details, see the book Vovk et al. (2005). This section will briefly present the inductive approach for regression, which is the groundwork for our paper. It will also review the related work.

2.1. Inductive conformal prediction (ICP) for regression

Let $\wr z_1 = (x_1, y_1), z_2 = (x_2, y_2), \ldots, z_n = (x_n, y_n)\wr$ be successive pairs constituting the observed training examples, with $x_i \in X$ an object and $y_i \in \mathbb{R}$ its label. We assume that the underlying random variables are exchangeable (a weaker condition than the usual i.i.d., meaning that our inference does not depend on their order nor change the underlying joint distribution. Given any new object $x_{n+1} \in X$, the objective is to predict $y_{n+1} \in \mathbb{R}$. To do so, the inductive conformal approach consists of the following steps :

1. Split the original training data set $Z = \wr z_1, \ldots, z_n\wr$ into a *proper training set* $Z^{tr} = \wr z_1, \ldots, z_l\wr$ and a *calibration set* $Z^{cal} = \wr z_{l+1}, \ldots, z_n\wr$, with $|Z^{cal}| = n - l = q$.

2. Train the chosen machine learning algorithm called the *underlying algorithm* on Z^{tr}, and get the *non-conformity measure* A_l. This measure determines how unusual an example is from a bag of other examples, called the non-conformity score. Thus, for an example z_k and a bag of examples $\wr z_1, \ldots, z_l\wr$, we can calculate the non-conformity score α_k of z_k compared to the other examples in the bag, such as:

$$\alpha_k = A_l(\wr z_1, \ldots, z_l\wr, z_k). \tag{1}$$

3. For each example z_i of Z^{cal}, calculate the non-conformity score α_i by applying (1) to get the sequence $\alpha_{l+1}, \ldots, \alpha_n$.

4. For a new example x_{n+1}, associate to any possible prediction $\hat{y}$ its non-conformity score $\alpha_{n+1}^{\hat{y}}$ using the underlying algorithm, and calculate its *p-value* expressing the proportion of less conforming examples than z_{n+1}, with:

$$p(\hat{y}_{n+1}) = \frac{|\{i = l+1, \ldots, n, n+1 : \alpha_i \geq \alpha_{n+1}^{\hat{y}}\}|}{q+1}. \tag{2}$$

5. By considering the desired probability of error $\epsilon \in (0, 1)$, called the *significance level*, a prediction set can be given at a *confidence level* of $1 - \epsilon$, which is the statistical guarantee of coverage of the true value y_{n+1} by this interval, which amounts to provide every $\hat{y}$ such that $p(\hat{y}) > \epsilon$.

When calculating $\alpha_{n+1}^{\hat{y}}$ for each new example x_{n+1}, $\hat{y}$ is replaced with each possible label in classification. However, it is not possible in regression to replace it with all possible values in $\mathbb{R}$. Thus, a prediction interval is instead given by the conformal regressor whose values depend on the confidence level and on the chosen non-conformity measure.

The two fundamental characteristics desired in conformal regressors are (a) *validity*, i.e. the error rate does not exceed ϵ for each chosen confidence level ϵ, and (b) *efficiency*, meaning prediction sets are as small as possible. Therefore, for two valid regressors, a smaller prediction interval will be much more informative and useful than a bigger one.

2.2. Non-conformity measures for regression

Many studies that tackle single-output regression focus on non-conformity measures, so as to treat the issue mentioned above. The most basic non-conformity measure is the absolute difference between the actual value y_i and the predicted value $\hat{y}_i$ by the underlying algorithm as follows:

$$\alpha_i = |y_i - \hat{y}_i|. \tag{3}$$

By applying (3) to Z^{cal}, we get the sequence of non-conformity scores and then sort them in descending order $\alpha_1, \ldots, \alpha_q$. Then, depending on the significance level ϵ, we define the index of the $(1-\epsilon)$-percentile non-conformity score, α_s, such as $s = \lfloor \epsilon(q+1) \rfloor$. Thus, for each new example x_{n+1}, its prediction interval, covering the true output y_{n+1} with probability $1 - \epsilon$, will be :

$$(\hat{y}_{n+1} - \alpha_s, \hat{y}_{n+1} + \alpha_s). \tag{4}$$

Using this standard non-conformity measure means that all prediction intervals have the same size $2\alpha_s$. We can improve this by using a *normalized* non-conformity measure. This latter provides individual bounds for each example by scaling the standard non-conformity measure with an additional term σ_i, which estimates the difficulty of predicting y_i, such as:

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

In this case, the prediction interval is smaller for "easy" examples, and bigger for "hard" examples, making two different examples with the same α_s value using (3) have two distinct interval predictions determined by their difficulty. Thus, for a new example x_{n+1}, the prediction interval becomes :

$$(\hat{y}_{n+1} - \alpha_s \sigma_{n+1}, \hat{y}_{n+1} + \alpha_s \sigma_{n+1}). \tag{6}$$

There are a lot of ways to calculate σ_i. Papadopoulos and Haralambous (2011) propose to train another model to estimate the error of the underlying model by predicting the value $\mu_i = \ln(|y_i - \hat{y}_i|)$. In this matter, the non-conformity score is defined as follows :

$$\alpha_i = \frac{|y_i - \hat{y}_i|}{\exp(\mu_i) + \beta}, \tag{7}$$

where $\beta \geq 0$ is a sensitivity parameter that controls changes in μ_i. The logarithmic scale is used instead of the direct one to guarantee that the estimate is always positive. For a new example x_{n+1}, the prediction interval is :

$$(\hat{y}_{n+1} - \alpha_s(\exp(\mu_{n+1}) + \beta), \hat{y}_{n+1} + \alpha_s(\exp(\mu_{n+1}) + \beta)) . \tag{8}$$

Other non-conformity measures proposed by Papadopoulos et al. (2011) use k-nearest neighbors by calculating two terms λ_i^k and ξ_i^k. The first one is λ_i^k and measures d_i^k, the sum of the distance of the example z_i from its k-nearest neighbors $x_{i_1}, \ldots, x_{i_k}$, and normalizes it with the median of all d_j^k over all training examples, so that we have:

$$\lambda_i^k = \frac{d_i^k}{median(d_j^k, z_j \in Z^{tr})}, \tag{9}$$

where

$$d_i^k = \sum\nolimits_{j=1}^{k} \delta(x_i, x_{i_j}), \tag{10}$$

where δ is a distance.

The second one is ξ_i^k and calculates the standard deviation s_i^k of the outputs of the example k-nearest neighbours, and normalizes it with the median of all standard deviations s_j^k of all training examples, such as :

$$\xi_i^k = \frac{s_i^k}{median(s_j^k, z_j \in Z^{tr})}, \tag{11}$$

where

$$s_i^k = \sqrt{\frac{1}{k}\sum\nolimits_{j=1}^{k}(y_{i_j} - \overline{y_{i_{1\ldots k}}})^2} \qquad \text{and} \qquad \overline{y_{i_{1\ldots k}}} = \frac{1}{k}\sum\nolimits_{j=1}^{k} y_{i_j}. \tag{12}$$

These two terms are then used to calculate many non-conformity measures, including :

$$\alpha_i = \left| \frac{y_i - \hat{y}_i}{\exp(\gamma\lambda_i^k)} \right|, \tag{13}$$

and

$$\alpha_i = \left| \frac{y_i - \hat{y}_i}{\exp(\gamma\lambda_i^k) + \exp(\rho\xi_i^k)} \right|, \tag{14}$$

where $\gamma \geq 0$ and $\rho \geq 0$ are sensitivity parameters that control changes in λ_i^k and ξ_i^k respectively. For a new example x_{n+1}, the prediction interval is respectively :

$$\left(\hat{y}_{n+1} - \alpha_s(\exp(\gamma\lambda_i^k)), \hat{y}_{n+1} + \alpha_s(\exp(\gamma\lambda_i^k))\right), \tag{15}$$

and

$$\left(\hat{y}_{n+1} - \alpha_s(\exp(\gamma\lambda_i^k) + \exp(\rho\xi_i^k)), \hat{y}_{n+1} + \alpha_s(\exp(\gamma\lambda_i^k) + \exp(\rho\xi_i^k))\right). \tag{16}$$

2.3. Related work

Conformal prediction for single-output regression has been studied in many papers with various approaches, such as Ridge regression (Nouretdinov et al. (2001)), regression trees (Johansson et al. (2018)), k-nearest neighbors (Papadopoulos et al. (2011)) and neural networks (Papadopoulos and Haralambous (2011)). Other papers discuss alternative ways to benefit from conformal regressors. For example, Spjuth et al. (2019) aggregates prediction intervals of multiple and independent sources of training data. ICP regression has also been adopted in different applications, for instance, drug discovery (Eklund et al. (2015)), software effort estimation (Papadopoulos et al. (2009)) and student grades (Morsomme and Smirnov (2019)).

To our knowledge, there are only two papers that address conformal prediction for multi-target regression. The first one is by Kuleshov et al. (2018), and applies it in order to get a valid measure of accuracy for Manifold Learning Regression algorithms. This paper is theoretical and does not present any experiments to test the proposed non-conformity measure. It is also limited to manifold learning. The second study is by Neeven and Smirnov (2018) and combines weather forecasting data from different websites on a period of time. To do so, it introduces a straightforward multi-target extension of the conformal single-output k-nearest neighbor regressor (CSkNNR) by Papadopoulos et al. (2011). This extension is a set of m CSkNNR models, one for each target variable Y^m, and its final output is the set of all prediction intervals constructed by each CSkNNR for its related Y^m. This work presents initial results for applying conformal prediction to multi-target regression. However, it focuses on a single non-conformity measure conducted on one single data set that does not always respect the exchangeability assumption of conformal prediction. Moreover, applying m distinctive single-output models on each target Y^m deprives the conformal model of profiting from the possible correlations between the different outputs. In this study, we make a first step towards accounting for such correlations, at least in the normlization of intervals.

3. Conformal multi-target regression

When the objective is to use conformal prediction for a multi-target regression with m targets, the most obvious strategy is to divide the problem into m single-output regression ones and adopt a separate conformal regressor for each one. This approach is indeed simple but does not account for the possible interactions that exist between the m targets.

In this work, we explore other extensions to existing non-conformity measures that take into consideration the possible links between these targets. The first idea is to learn the normalizing coefficients by a multi-target model. Thus, this latter will exploit the information coming from training each task and will share representations between related targets in order to generalize better and give greater performance results (see Ruder (2017) and Caruana (1993)). The second idea is to use a deep fixed-length representation of the data that was learned from trying to predict all the targets at once when calculating the normalizing coefficient in k nearest neighbors based non-conformity measures, thus embedding the correlation information in the deep network layers. This is based on the fact that using representations of the examples instead of the raw instances is a good method to

boost the performance of the neural network on tabular data by helping it generalize better (see Guo and Berkhahn (2016) and De Brébisson et al. (2015)).

3.1. Non-conformity measures (NCMs) for multi-target regression

We extend three existing non-conformity measures in the single-output regression case to the multi-target regression problem as follows :

- **SINGLE :** uses the normalized non-conformity measure described in (7) where each μ_i is estimated by a single deep neural network trained on each output separately.
- **Original k-NN (O-KNN) :** adopts the k-nearest neighbors non-conformity measure (13) based on λ_i^k only, with the distances calculated between the original form of the examples x_i.
- **Original k-NN with Standard Deviation (OS-KNN) :** adopts the k-nearest neighbors non-conformity measure (14) based on λ_i^k and ξ_i^k, with the distances calculated between the original form of the examples x_i.

To these existing non-conformity measures, we add three new ones defined as :

- **MULTI :** trains a single deep neural network to estimate the normalizing coefficients μ_i in (7) for all outputs at the same time.
- **Representation k-NN (R-KNN) :** instead of using the original form of the data, it employs the learned deep representations of the examples extracted from the before last dense layer of the underlying algorithm's neural network to compute d_i^k in λ_i^k for the k-nearest neighbors non-conformity measure (13).
- **Representation k-NN with Standard Deviation (RS-KNN) :** uses the k-nearest neighbors non-conformity measure (14) with λ_i^k and ξ_i^k where the learned deep representations of the examples are used to calculate d_i^k.

3.2. Our approach

Since Transductive Conformal Prediction (TCP) is not computationally efficient when using deep learning architectures, we use the Inductive Conformal Prediction (ICP) framework as described in Section 2.1 in order to only train the underlying deep neural network model once. Hence, we follow these steps :

1. Split the original training set into two smaller subsets: the proper training set and the calibration set.
2. Use the proper training set to train the underlying algorithm, which is a deep neural network, and get the output predictions and the representations of each example.
3. For each non-conformity measure, learn the appropriate normalizing algorithm (deep neural network for SINGLE and MULTI NCMs, and kNN on the original examples for O-KNN and OS-KNN NCMs and their representations for R-KNN and RS-KNN NCMs) using the proper training set.

4. Compute the non-conformity scores for each non-conformity measure by using the calibration set.

5. For each example in the test set, predict $\hat{y}$ and its representation using the underlying neural network, calculate its non-conformity score, and compute its interval prediction depending on the significance level ϵ for each non-conformity measure.

This approach is executed for different non-conformity measures and for various values of ϵ in order to measure the performance of each approach, and verify its validity and efficiency.

4. Evaluation

In this section, we describe the experimental setting (underlying algorithm, data sets and performance metrics) and the results of our study.

4.1. Experiments

Since we are working with neural networks, we normalize all the features and targets to have a mean of 0 and a standard deviation of 1 as a pre-processing step since it makes the deep neural network optimization easier. Then, we conduct all our experiments with 10-folds cross validation, meaning that each data set is split into 10 equally-sized folds and the experiments are repeated for each k fold as the test set and the remaining $k-1$ sets as the training set. This procedure is necessary in order to eliminate biased results caused by a specific split of the data or the examples chosen in the calibration set. The results are thus averaged on all 10 folds.

The overall focus of this paper is to compare between the different non-conformity measures presented above. Thus, for all experiments on all data sets, we keep the same amount of examples in the calibration set, which is 10% of the training examples. We also do not optimize the sensitivity parameters for each data set and use the same values, which are $\beta = 0.1$ for SINGLE and MULTI NCMs, and $\gamma = \rho = 0.5$ for the remaining NCMs. Moreover, we keep the same underlying machine learning algorithm for all experiments, which is a deep neural network. Its architecture is as follows :

- Apply a dense layer to the input (with the number of the continuous features and the number of one hot values for the categorical features), with "selu" activation (scaled exponential linear units Klambauer et al. (2017)).
- Add three other hidden dense layers with dropouts and "selu" activation.
- Add a dense layer with "selu" activation and use it as a feature extractor to produce a representation vector with a fixed size.
- Add a final dense layer with m neurons representing the targets and a linear activation to get the outputs predicted by the model.

The results of this deep neural architecture will enable us to calculate values of the normalizing coefficients for the corresponding non-conformity measure :

- μ_i : estimate the error $\ln(|y_i - \hat{y}_i|)$ which will be learned by the normalizing neural network. This multi-layer perceptron is composed of three hidden dense layers with "selu" activation and dropouts and a final dense layer with one output for each target separately in the case of the SINGLE non-conformity measure, or with m neurons for all targets at once in the case of the MULTI non-conformity measure.
- λ_i^k **and** ξ_i^k : use the deep representations of each example to compute d_i^k for the k-nearest neighbors based non conformity measures R-KNN and RS-KNN.

4.1.1. Performance metrics

In order to adapt the conformal framework to the multi-target regression problem, we calculate all performance measures on a set of hyper-rectangle predictions such as :

$$[\hat{y}_i] = \times_{j=0}^{m} [\underline{\hat{y}}_i^j, \overline{\hat{y}}_i^j], \tag{17}$$

where $\times$ is a Cartesian product, m is the number of targets and $\underline{\hat{y}}_i^j, \overline{\hat{y}}_i^j$ are respectively the lower and upper bounds of the interval predictions given by the non-conformity measure for each target in Y^m. This hyper-rectangle has a volume of $\prod_{j=0}^{t}(\overline{\hat{y}}^j - \underline{\hat{y}}^j)$.

Efficiency is mainly based on the tightness of the hyper-rectangle's volume. This verification is done by calculating the median volume instead of the mean volume in order to avoid the impact of outlier interval predictions for each target, especially as these extreme values can be much more amplified when calculating the hyper-rectangle volume.

Validity is calculated based on the accuracy, i.e. checking whether the observation $y_i \in [\hat{y}_i]$ or not depending on ϵ value. In the single-output case, the confidence level $1 - \epsilon$ is easily verified by checking whether the real output value is in the interval prediction. However, this is harder in the multi-target case, as ϵ is used as a probability error for each target m, and a correctly predicted example must have all of its m observed values y^j in the corresponding interval predictions for each target. In this case, the actual confidence level of the hyper-rectangle corresponding to ϵ is equal to $(1 - \epsilon)^m$. Hence, we need to differentiate between ϵ_t corresponding to the actual significance level for each target and ϵ_h, that of the hyper-rectangle. To adapt the conformal prediction framework to the multi-target regression problem, we can calculate the value of ϵ_t that should be used in order to obtain ϵ_h, so as to get an overall confidence level of the hyper-rectangle $1 - \epsilon_h$. This corrected ϵ_t is defined as follows :

$$\epsilon_t = 1 - \sqrt[m]{1 - \epsilon_h}, \tag{18}$$

and our experiments focus on this corrected validity measure.

4.1.2. Data sets

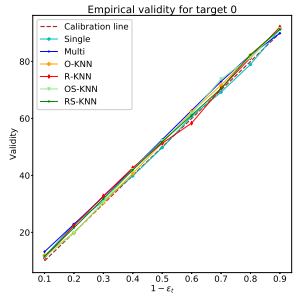
We use eleven data sets with various numbers of targets to examine the effectiveness of the conformal method in the case of multi-target regression. Their information is summarized in Table 1.

Names	Examples	Features	Targets	Source
enb	768	8	2	Mulan (Tsanas and Xifara (2012))
music origin	1059	68	2	UCI (Zhou et al. (2014))
indoor loc	21049	520	3	UCI (Torres-Sospedra et al. (2014))
scpf	1137	23	3	Mulan (Kaggle (2013))
sgemm	241600	14	4	UCI (Nugteren and Codreanu (2015))
rf1	9125	64	8	Mulan (Tsoumakas et al. (2011))
rf2	9125	576	8	Mulan (Tsoumakas et al. (2011))
wq	1060	16	14	Mulan (Džeroski et al. (2000))
scm1d	9803	280	16	Mulan (Tsoumakas et al. (2011))
scm20d	8966	61	16	Mulan (Tsoumakas et al. (2011))
community crime	2215	125	18	UCI (Redmond (2011))

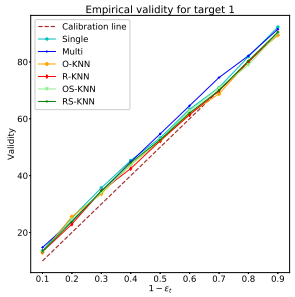
Table 1: Information on the used multi-target regression data sets.

4.2. Results

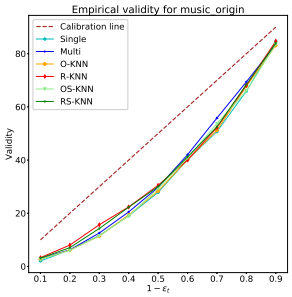
To motivate the choice of using a corrected validity measure to estimate the performance of our non-conformity measures, we show the results of the empirical validity for each target of the "music origin" data set, as well as its uncorrected hyper-rectangle validity in Figure 1. The results show that for each target, the calibration validity is reached or surpassed by all non-conformity measures. However, when computing the validity for the hyper-rectangle, the performance of the conformal predictors for multi-target regression (computed without correction) is less than the calibration line, due to the observation made earlier. This proves the utility of using the corrected validity measure to compute the actual values of ϵ_t to verify a $1 - \epsilon_h$ confidence level on all the hyper-rectangle for each ϵ_h.



(a) Empirical validity for target 0



(b) Empirical validity for target 1

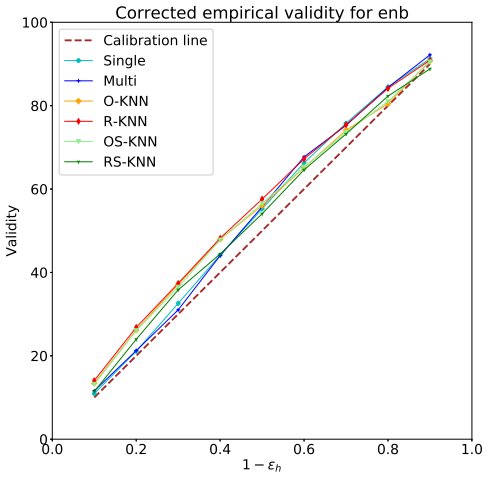


(c) Uncorrected empirical validity for the hyper-rectangle

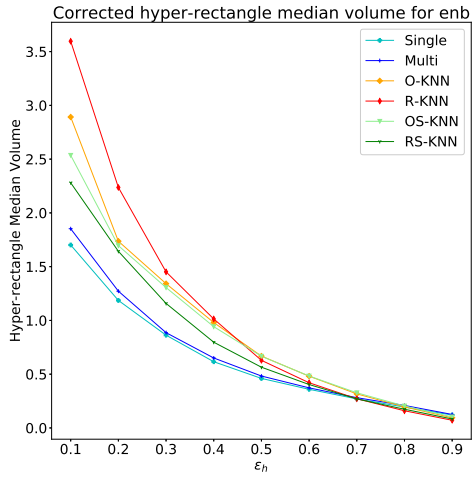
Figure 1: Results per target for music origin.

The results of the corrected empirical validity and the corrected median hyper-rectangle volume that measures efficiency for each data set are shown in Figures 2, 3, 4, 5, 6, 7, 8, 9,

10, 11 and 12. Note that for data sets with more than four targets, we use a logarithmic scale to plot the median volume, as hyper-rectangle volumes quickly decrease when lowering the required confidence.

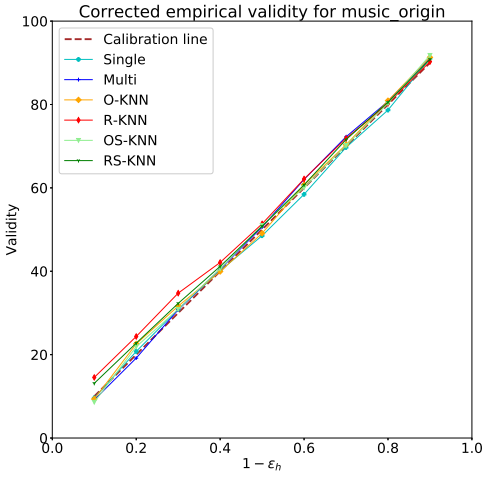


(*a*) Corrected empirical validity

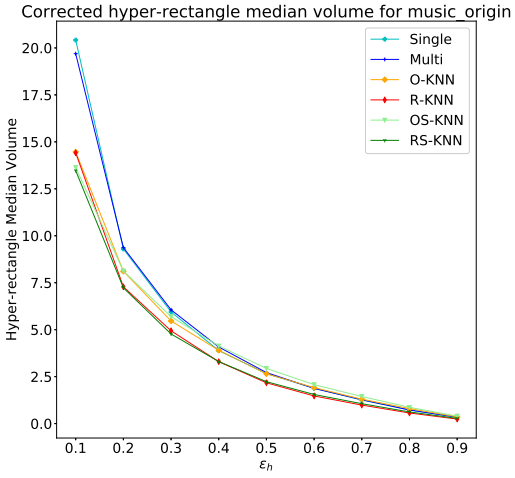


(*b*) Corrected median volume

Figure 2: Results for enb.

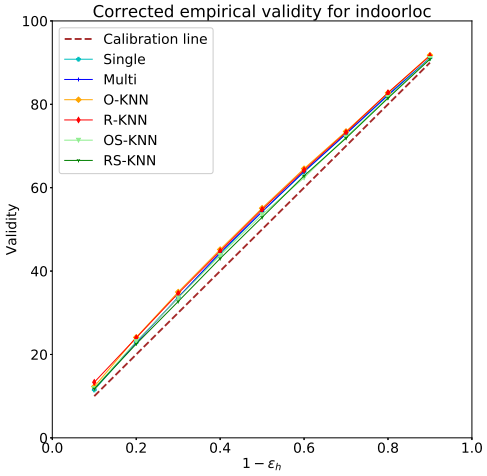


(*a*) Corrected empirical validity

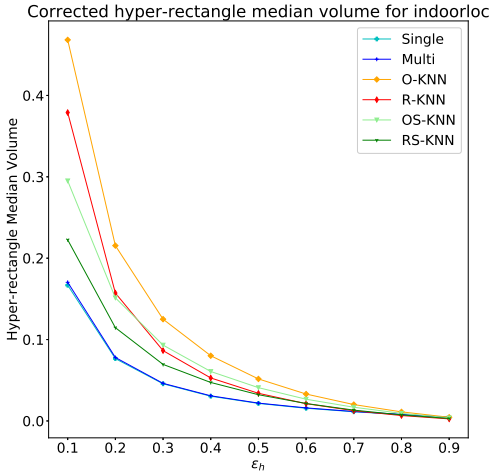


(*b*) Corrected median volume

Figure 3: Results for music origin.

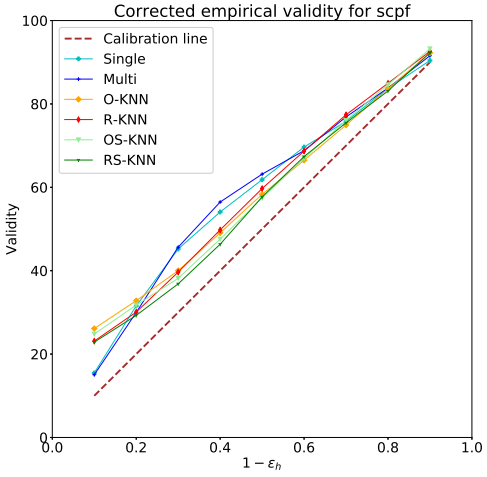


(a) Corrected empirical validity

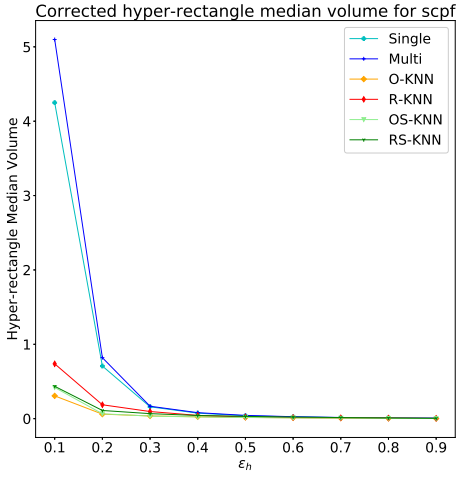


(b) Corrected median volume

Figure 4: Results for indoor loc.

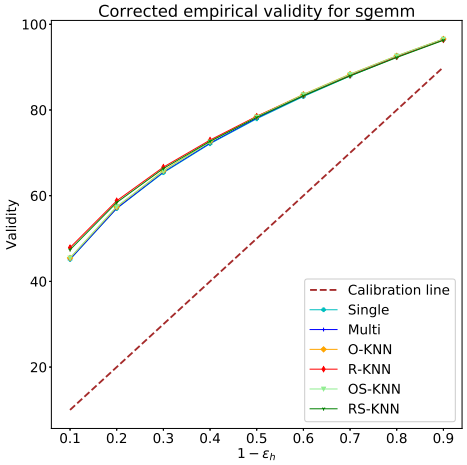


(a) Corrected empirical validity

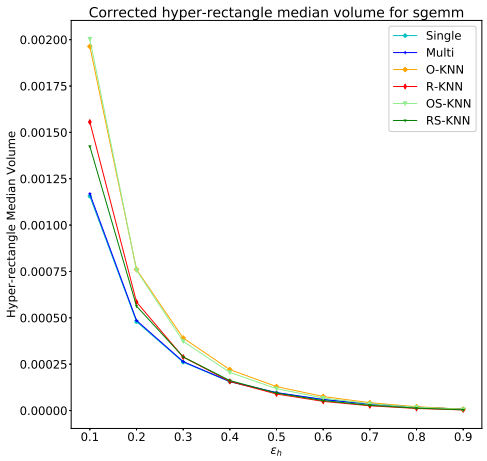


(b) Corrected median volume

Figure 5: Results for scpf.

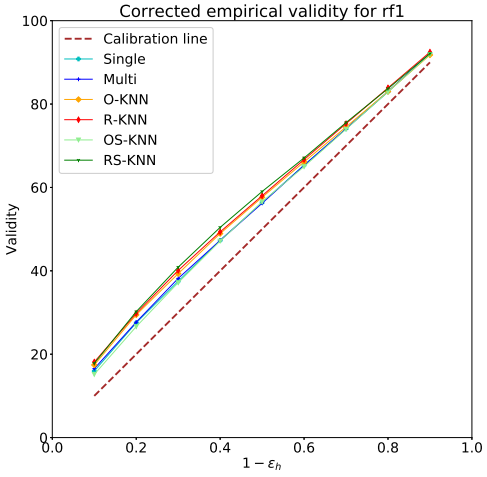


(*a*) Corrected empirical validity

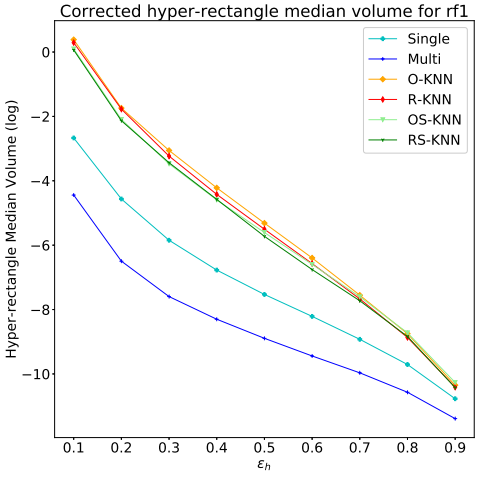


(*b*) Corrected median volume

Figure 6: Results for sgemm.

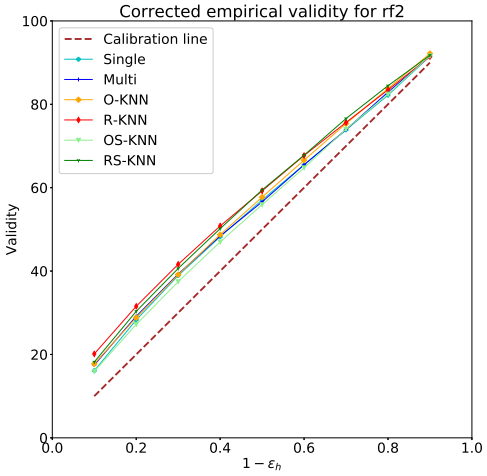


(*a*) Corrected empirical validity

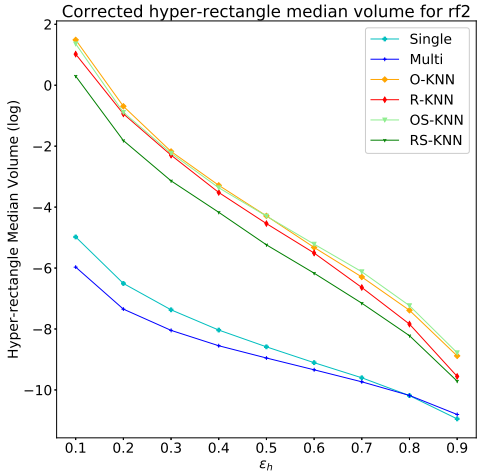


(*b*) Corrected median volume (log)

Figure 7: Results for rf1.

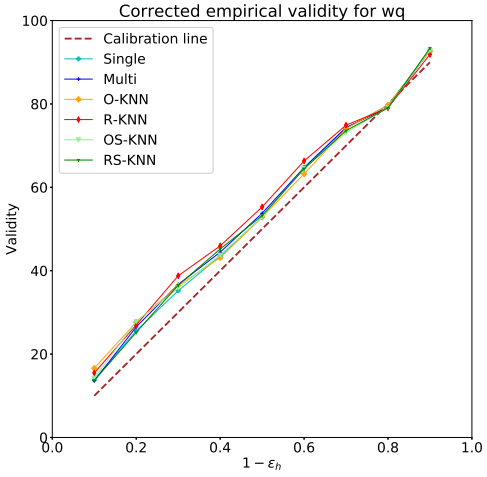


(*a*) Corrected empirical validity

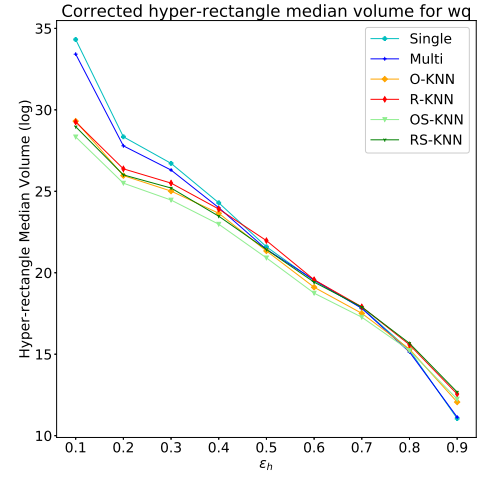


(*b*) Corrected median volume (log)

Figure 8: Results for rf2.

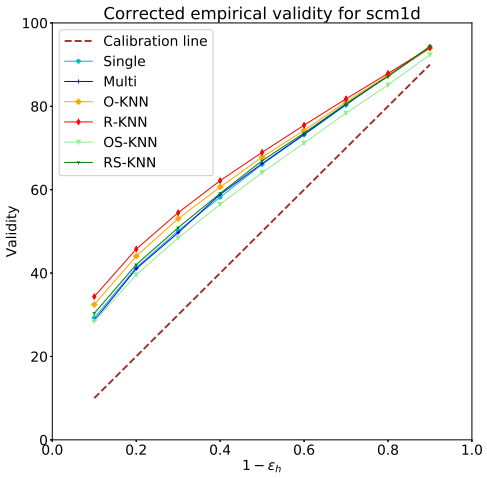


(*a*) Corrected empirical validity

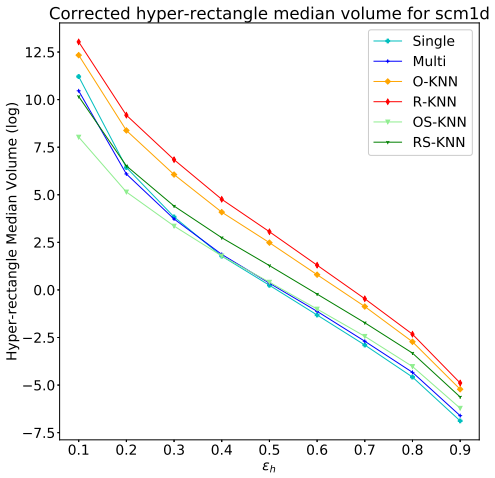


(*b*) Corrected median volume (log)

Figure 9: Results for wq.

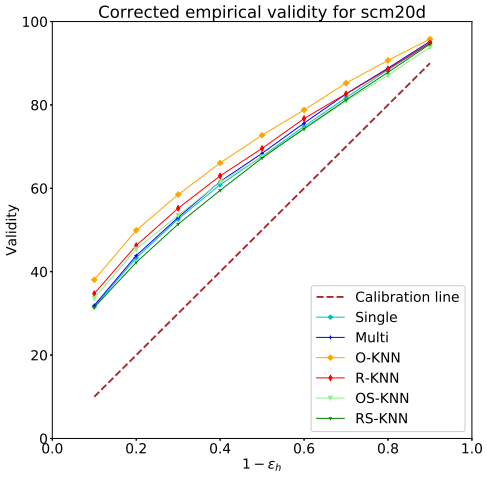


(*a*) Corrected empirical validity

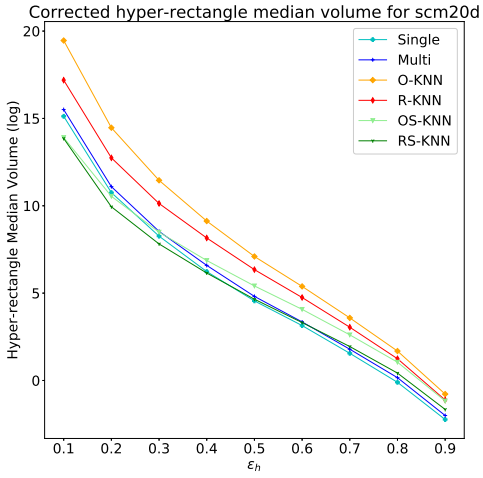


(*b*) Corrected median volume (log)

Figure 10: Results for scm1d.

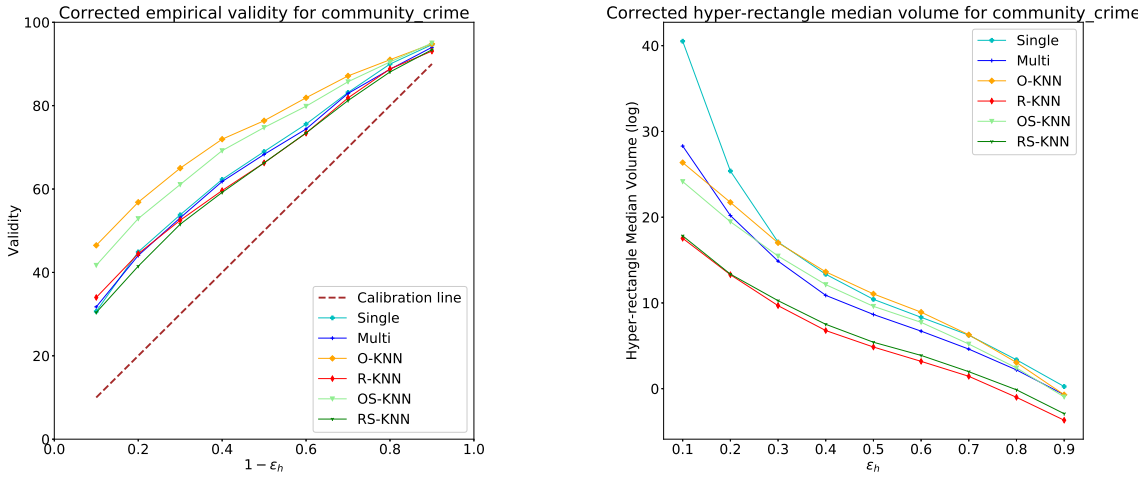


(*a*) Corrected empirical validity



(*b*) Corrected median volume (log)

Figure 11: Results for scm20d.



(a) Corrected empirical validity

(b) Corrected median volume (log)

Figure 12: Results for community crime.

In the case of validity, the results of the empirical validity plots (figure (a) for each data set) show that using corrected values of ϵ_t confirm the validity of conformal predictors in the case of multi-target regression. All NCMs seem to perform well on all data sets, without any obvious best non-conformity measure for all data sets. For "enb", "music origin" and "indoor loc", this validity is close to the calibration line. For the other data sets, the validity is higher than the confidence level defined by ϵ_h values, indicating that the correction may be too strong, even if the NCMs are valid for each individual target (as seen in Figure 1). This may be due to the interactions between the targets, to the number m of targets, or to the distribution of each data set. Further experiments and studies should be conducted to verify this.

In the case of efficiency, the results of the median hyper-rectangle volume (figure (b) for each data set) show that the overall volume decreases with the number of targets. This can be explained by the fact that when ϵ_h grows, the confidence level becomes small, which means that we allow for smaller prediction intervals (as we approach a point prediction), and therefore the conformal regressor tends to give prediction intervals that are smaller and smaller for each target (values less than 1). Then after multiplying these intervals to compute the volume, we find that the hyper-rectangle volume approaches 0 faster when m is large (which justifies the use of a logarithmic scale for data sets with more than four targets).

For all data sets except "scm1d" and "enb", for which results are more ambiguous, we observe that using a MULTI NCM gives predictions with tighter volumes compared to a SINGLE NCM (comparing SINGLE with MULTI, O-KNN with R-KNN and OS-KNN with RS-KNN). The magnitude of this improvement however depends on both the methods and the data set. For instance, it is large for every method in "community crime" (Figure 12), only for MULTI in "rf1" (Figure 7), and only for the k-NN approaches in "indoor loc"

(Figure 4). We also notice that the RS-KNN is in most cases the best non-conformity measure among the ones based on k-nearest neighbors, which shows the advantage of using a deep representation of the examples instead of their original form. However, we cannot observe a best NCM overall as the results differ from one data set to another.

4.3. Computation time

During the experiments, we also computed the time taken in seconds for each non-conformity measure to train and predict for each fold on all data sets. Note that since O-KNN and OS-KNN (respectively R-KNN and RS-KNN) share the same values of parameter λ_i^k, the training of the k-NN is done at the same time for both of them. Thus, the computation time is grouped for both of them. The results of these computation times averaged on 10 folds are shown in the Table 2.

	SINGLE	MULTI	O/OS-KNN	R/RS-KNN
enb	19.18	11.23	0.1	0.24
music origin	39.2	19.78	0.13	0.31
indoor loc	140.64	53.64	33.85	2.19
scpf	30.41	15.88	0.17	0.3
sgemm	2831.4	729.09	33.89	31.12
rf1	124.46	93.12	1.91	2.15
rf2	274.16	96.56	2.36	2.23
wq	187.13	49.31	0.7	0.88
scm1d	245.47	22.28	7.13	4.25
scm20d	193.92	23.54	4.06	4.14
community crime	239.56	45.9	1.38	1.51

Table 2: Average computation time in seconds per NCM for all data sets.

From these results, the non-conformity measures based on k-nearest neighbors O/OS-KNN and R/RS-KNN have similar computation time, showing that using the original form of the examples or a deep representation of them does not affect the computation time. However, when comparing SINGLE to MULTI non-conformity measures, we notice that the first one is much slower than the second one, with an increasing difference between them as the number of targets m grows. This is due to the fact that the SINGLE NCM needs to train additional $m-1$ models compared to the MULTI NCM, with each time corresponding to a training done on one single target separately. This shows another advantage of using a MULTI NCM approach instead of a SINGLE one.

5. Conclusion

In this paper, we applied inductive conformal prediction to multi-target regression using deep neural networks. We extended non-conformity measures from the single-output regression problem to the multi-target regression case and proposed new non-conformity measures. We also introduced a corrected significance level calculation for the whole output space in order

to compute the necessary significance levels for each target to get the overall confidence level. We then performed an empirical study on various data sets, with results showing that our new non-conformity measures generally outperform the existing ones. These results show that using a multi-target non-conformity measure instead of a single-target one is better with regards to the efficiency and complexity time. They also show that using a deep representation of the examples instead of the original form of the data in the distance calculation of k-NN based non-conformity measures is better when it comes to prediction tightness.

This paper is an introductory study of the application of conformal prediction framework to the multi-target regression setting. Our future work will further explore the theoretical definition of non-conformity measures in the multivariate space and on other performance metrics to ensure that the used methods are well-calibrated and efficient not only for each individual target but also on the overall results.

References

Hanen Borchani, Gherardo Varando, Concha Bielza, and Pedro Larrañaga. A survey on multi-output regression. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 5(5):216–233, 2015.

Rich Caruana. Multitask learning: A knowledge-based source of inductive bias icml. *Google Scholar Google Scholar Digital Library Digital Library*, 1993.

Alexandre De Brébisson, Étienne Simon, Alex Auvolat, Pascal Vincent, and Yoshua Bengio. Artificial neural networks applied to taxi destination prediction. *arXiv preprint arXiv:1508.00021*, 2015.

Sašo Džeroski, Damjan Demšar, and Jasna Grbović. Predicting chemical parameters of river water quality from bioindicator data. *Applied Intelligence*, 13(1):7–17, 2000.

Martin Eklund, Ulf Norinder, Scott Boyer, and Lars Carlsson. The application of conformal prediction to the drug discovery process. *Annals of Mathematics and Artificial Intelligence*, 74(1-2):117–132, 2015.

Alex Gammerman, Volodya Vovk, and Vladimir Vapnik. Learning by transduction. *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, page 148155, 1998.

Cheng Guo and Felix Berkhahn. Entity embeddings of categorical variables. *arXiv preprint arXiv:1604.06737*, 2016.

Ulf Johansson, Henrik Linusson, Tuve Löfström, and Henrik Boström. Interpretable regression trees using conformal prediction. *Expert systems with applications*, 97:394–404, 2018.

Kaggle. Kaggle competition: See click predict fix, 2013. URL https://www.kaggle.com/c/see-click-predict-fix.

Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. In *Advances in neural information processing systems*, pages 971–980, 2017.

Alexander Kuleshov, Alexander Bernstein, and Evgeny Burnaev. Conformal prediction in manifold learning. In *Conformal and Probabilistic Prediction and Applications*, pages 234–253, 2018.

Raphaël Morsomme and Evgueni Smirnov. Conformal prediction for students grades in a course recommender system. In *Conformal and Probabilistic Prediction and Applications*, pages 196–213, 2019.

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

Ilia Nouretdinov, Thomas Melluish, and Volodya Vovk. Ridge regression confidence machine. In *ICML*, pages 385–392, 2001.

Cedric Nugteren and Valeriu Codreanu. Cltune: A generic auto-tuner for opencl kernels. In *2015 IEEE 9th International Symposium on Embedded Multicore/Many-core Systems-on-Chip*, pages 195–202. IEEE, 2015.

Harris Papadopoulos and Haris Haralambous. Reliable prediction intervals with regression neural networks. *Neural Networks*, 24(8):842–851, 2011.

Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. In *European Conference on Machine Learning*, pages 345–356. Springer, 2002.

Harris Papadopoulos, Efi Papatheocharous, and Andreas S Andreou. Reliable confidence intervals for software effort estimation. In *AIAI Workshops*, pages 211–220, 2009.

Harris Papadopoulos, Vladimir Vovk, and Alexander Gammerman. Regression conformal prediction with nearest neighbours. *Journal of Artificial Intelligence Research*, 40:815–840, 2011.

M Redmond. Communities and crime unnormalized data set. *UCI Machine Learning Repository. In website: http://www. ics. uci. edu/mlearn/MLRepository. html*, 2011.

Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

Ola Spjuth, Robin Carrión Brännström, Lars Carlsson, and Niharika Gauraha. Combining prediction intervals on multi-source non-disclosed regression datasets. *arXiv preprint arXiv:1908.05571*, 2019.

Joaquín Torres-Sospedra, Raúl Montoliu, Adolfo Martínez-Usó, Joan P Avariento, Tomás J Arnau, Mauri Benedito-Bordonau, and Joaquín Huerta. Ujiindoorloc: A new multi-building and multi-floor database for wlan fingerprint-based indoor localization problems. In *2014 international conference on indoor positioning and indoor navigation (IPIN)*, pages 261–270. IEEE, 2014.

Athanasios Tsanas and Angeliki Xifara. Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy and Buildings*, 49:560–567, 2012.

Grigorios Tsoumakas, Eleftherios Spyromitros-Xioufis, Jozef Vilcek, and Ioannis Vlahavas. Mulan: A java library for multi-label learning. *Journal of Machine Learning Research*, 12 (71):2411–2414, 2011. URL http://jmlr.org/papers/v12/tsoumakas11a.html.

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

Huazhen Wang, Xin Liu, Ilia Nouretdinov, and Zhiyuan Luo. A comparison of three implementations of multi-label conformal prediction. In *International Symposium on Statistical Learning and Data Sciences*, pages 241–250. Springer, 2015.

Min-Ling Zhang and Zhi-Hua Zhou. A review on multi-label learning algorithms. *IEEE transactions on knowledge and data engineering*, 26(8):1819–1837, 2013.

Fang Zhou, Q Claire, and Ross D King. Predicting the geographical origin of music. In *2014 IEEE International Conference on Data Mining*, pages 1115–1120. IEEE, 2014.