ulm university universität
uulm

University of Ulm | 89069 Ulm | Germany

Faculty of
Computer Science
Medical Systems
Biology

Conformal Prediction - A Basic Overview

Master Project: Medical Systems Biology
Submitted for the Artificial Intelligence program.

Submitted by:
Michel Lutz
Matriculation number: 1048353

Supervisor:
Prof. Dr. Hans Kestler
Dr. Julian Schwab
Ulm, January 2024

Abstract

Machine learning models often have the disadvantage that they do not provide an intrinsic notation about their uncertainty, meaning that bad predictions with high uncertainty cannot be distinguished from good predictions with high confidence. One way to tackle this problem is Conformal Prediction (CP), an innovative model-agnostic, finite-sample size and distribution free uncertainty estimation framework, which is applicable for all types of models, even pre-trained ones, requiring only the exchangability of the data.

In recent years Conformal Prediction has come into the focus of the machine learning community and has since spread rapidly in the academic world and beyond. The work presented here provides a broad overview of the main conformal methods such as, Full-, Split-, CV+ and Jackknife+ CP, both for regression and for classification tasks. Furthermore, marginal and adaptive coverage guarantees as well as class-balanced and Mondrian conformal methods are discussed. In addition, the article provides a theoretical estimate for the required calibration set size, ways to evaluate conformal methods, an introduction to Venn predictors and an overview of the historical development of CP.

The possibility of applying conformal methods to small and very small data sets, as they frequently occur in the biomedical context, was evaluated in several series of experiments on an artificially down-sampled synthetic and real-life data set. All conformal methods have in common that they fulfil the coverage guarantee in all cases, but the size and thus the informativeness of the resulting prediction sets depends on the quality of the underlying model, which limits their usability for very small data sets.

Contents

1. Introduction

Machine learning models have become more powerful and prominent in recent years and are increasingly being used in critical decision-making processes, for example in medical care. However, it is still unclear to what extent the predictions of these models can be trusted. Often these systems provide only point predictions with no indication about their quality, and even when they do, these are usually heuristic notations without any guarantee. Even if models make good predictions for most data points and demonstrating high accuracy, many systems tend to treat difficult and unusual data points with unjustified confidence. It is therefore impossible for users to distinguish bad from good predictions.

In order to make rational decisions on the basis of machine-generated predictions, a reliable uncertainty quantification is essential, which makes transparent in a valid way whether the system is certain of a prediction or whether it is a wild speculation. In the words of Stanford University Prof. Emmanuel Candès: "Predictive models are used to make decisions that can have enormous consequences for people's lives. It's extremely important to understand the uncertainty about these predictions, so people don't make decisions based on false beliefs" (Candès, 2020).

1.1 Uncertainty Estimation

A meaningful uncertainty estimate should allow statements to be made about the credibility (how likely is the (point) prediction) and the confidence (how likely are the alternative predictions to the (point) prediction) of a system. This makes it possible to distinguish difficult data points with uncertain predictions from simpler ones with more reliable ones. In the interest of algorithmic fairness, it should also be possible to recognize whether the predictions of a system in a certain subset of data, e.g. health data of men and women, are subject to different degrees of uncertainty. This means that it must be made transparent whether, for example, there are greater uncertainties for female patients than for male patients.

Some machine learning models already have a heuristic notation of their uncertainty by providing not only a point prediction but also, for example, a probability distribution over the entire label space. The softmax outputs of neural networks or the predictive posterior distributions of Bayesian models are therefore typical examples. Other heuristic notations for the uncertainty of a system can be obtained by bootstrapping or additionally trained residual models. In models such as random forests, the variance between the trees can be interpret as a measure of their uncertainty. Nevertheless the main problem with all these approaches is that there is no reasonable guarantee that their predictive distributions are calibrated, i.e. that they actually deliver what they promise. So there is no guarantee that the prediction sets or intervals constructed on their basis contain the true labels with certainty. In other words they do not fulfill any coverage guarantee (Niculescu-Mizil & Caruana, 2005) (Lambrou, Papadopoulos, Nouretdinov, & Gammerman, 2012) (Manokhin, 2022b) (Dewolf, Baets, & Waegeman, 2023). For neural networks it is known that they are not well calibrated (Johansson & Gabrielsson, 2019) and for Bayesian approaches there is rarely a good reason to trust the proposed priors. In bootstrapping methods and residual models, it is known that they tend to underestimate the true variance (Hesterberg, 2015) (Manokhin, 2022b). To summarise in the words of Valery Manokhin a former student of Vladimir Vovk: "[..] many machine learning algorithms do not produce class membership probabilities and the ones that do often generate classification scores that do not correspond to class probabilities. In such cases the scores need to be transformed into well-calibrated probabilities that can be combined with utility scores for effective decision-making" (Manokhin, 2022b).

In the past, many approaches have been proposed to calibrate predictive distributions and thus to obtain a rigorous notation for the uncertainty of a system. Conformal Prediction (CP) is, to the best of my knowledge, the only framework that provides a model agnostic way to transform a heuristic notation for uncertainty into a rigorous one in both classification and regression contexts (Figure 1). In other words, the output of a conformal method has a probabilistic guarantee that it covers the true outcome.

1.2 Contribution

In this paper, the theoretical foundations of Conformal Prediction are first presented on an intuitive level to give the reader a first impression of the framework (Chapter 2). The difference between Full and Split Conformal Prediction is then explained in more detail (Chapter 2.3) and the latter is presented as the more practically relevant, special case, of Full Conformal Prediction. All CP methods use the same core steps and differ only in their specific score functions, this core algorithm, the conformal recipe, is formulated

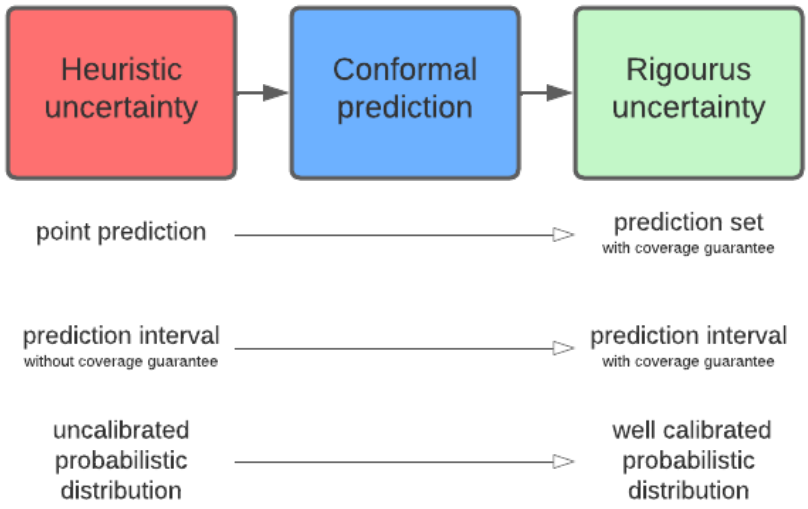


Figure 1: Conformal Prediction uses a purely heuristic notation of uncertainty of any model and transforms it to a rigorous uncertainty estimation with valid coverage guarantee.

in Section 2.3.3. Split conformal methods use a hold-out dataset, often referred to as a calibration set. Theoretical considerations about its size are provided in Chapter 2.4.3.

All Conformal Prediction methods guarantee marginal coverage, but conditional coverage guarantees are more crucial for many scenarios. An introduction to the difference between marginal coverage and conditional coverage is provided in Section 2.4.

Even though conditional coverage cannot be theoretically guaranteed, methods to approach it exist with Mondrian Conformal Prediction 2.4.1 and 2.4.2. CP can be used for both classification and regression tasks, and the Chapters 2.5 and 2.8 provide initial information and specific score functions for regression and classification tasks.

Besides Split and Full Conformal Prediction, CV+ and Jackknife+ are two methods that allow a trade-off between computational effort and data efficiency and are useful for many practically relevant scenarios with only a few available data points. Both methods are described in the Chapter 2.7.

Conformal Prediction can also be used to recognise change points in the underlying distribution of the data. More precisely, the idea is that in an online setting a distribution shift can be controlled with a guaranteed false alarm rate. The basic principle of this method is described in Chapter 3.

Coverage is guaranteed for all conformal methods. However, implementation errors can always occur, or the exchangability of the data is violated. In these cases, coverage is not ensured. It is therefore essential to empirically evaluate the coverage for each new CP method. Furthermore, the adaptivity of each CP method should be checked. Several metrics have been proposed for these correctness checks, which are briefly presented in Chapter 4.

Closely related to CP is the issue of calibrating predictive probability distributions. Many traditional models do not provide well-calibrated distributions and various calibration methods have been proposed in the past. In Chapter 5.1 some of these methods are quickly introduced before Chapter 5.2 describes a framework closely related to CP, the Venn predictors, which allows the calibration of probabilistic predic-

tions with certain guarantees.

In addition to this overview of the different varieties and methods of Conformal Prediction, Chapter 6 provides an overview of the literature a historical account of the development from the initial work of Vladimir Vovk in the 1990s to the latest trends of recent years initiated by researchers such as Michael Jordan and Emanuel Candes.

In the experimental part of this work, the influence of different conformal methods, Split, CV+ and Jackknife+, on small datasets was investigated. For this purpose, both a real life dataset and an artificial dataset were artificially reduced in size and prediction sets with different conformal methods were created and compared with each other (Section 7).

2. Theory

Conformal Prediction (CP) is an innovative distribution-free, non-parametric, model-agnostic framework for conformance estimation with strict coverage guarantees at finite sample size. CP requires minimal assumptions, more specifically the framework only requires data exchangeability, a slightly weaker requirement than the iid requirement needed for many typical machine learning applications. CP comes in several flavors and requires only negligible additional computation time, at least in the inductive setting (Split Conformal Prediction). In this case, CP can be considered as a wrapper for any machine learning model that calibrates any heuristic notation of uncertainty provided by these models, thus guaranteeing mathematically valid coverage.

2.1 Coverage Guarantee

Conformal Prediction guarantees that the output of each CP method contains the ground truth with a predefined probability $1 - \alpha$ on average. This property is usually referred to as coverage guarantee and is ensured by the fact that the method generates a prediction set $\mathbf{C}$ instead of a point predictor. In other words, the set contains the true label with a probability of exactly $1 - \alpha$. For labeled data points $z_i = (x_i, y_i)$, the coverage guarantee is expressed as follows, taking into account a small correction factor for finite sample size:

Definition 1. ***Coverage guarantee***

$$1 - \alpha \quad \leq \quad \mathbb{P}\Big(\mathbf{y}_{test} \in \mathbf{C}(\mathbf{x}_{test})\Big) \quad \leq \quad 1 - \alpha + \frac{1}{n+1} \tag{1}$$

This statement holds for all sample sizes, models and distributions, without the need to make any further assumptions beyond the exchangeability of the data.

2.2 Intuition of Conformal Prediction

In the following, the underlying principle of Conformal Prediction is explained on an intuitive level before a mathematically precise definition of the different conformal methods follows later on. For this purpose, we consider the historically first developed case of Full CP (transductive Conformal Prediction), since in this case the underlying rational becomes particularly clear and all other types of CP can be expressed as special cases of it (Fontana, Zeni, & Vantini, 2023).

2.2.1 Conformity Score

In a nutshell, Conformal Prediction uses past experience to generate precise levels of confidence in new predictions. Therefore, for one data point $z_{n+1} \in \mathbf{Z}$, e.g. the prediction, it is measured how "unusual" it looks compared to the bag of all exchangeable data points $\wr z_1, .., z_{n+1} \wr$. The inventors of the CP framework Gammerman, Vovk, and Vapnik (Gammerman, Vovk, & Vapnik, 1998a) refer to this as "a convenient measure of the evidence found to support this prediction". This concept of nonconformity or "strangeness" is quantified by a nonconformity measure (NCM) function, $S : Z^N \times Z \to \mathbb{R}$, which in principle can be

defined arbitrarily without violating the coverage guarantees of CP. However, for the informativeness of a conformal method an appropriate choice of the NCM function is crucial. The strangeness of a data point $s_i = S(z_i)$ is now a days usually called the conformity score (Angelopoulos & Bates, 2021). Emphasis that the conformity score increases with increasing strangeness. The following chapters discusses how to chose a appropriate conformity score for regression as well as classification problems in more detail.

2.2.2 Constructing The Prediction Set

Thus, the question arises how to construct valid prediction sets from the conformity scores, which guarantee to deliver the true result with a definable certainty. Let us consider labeled exchangeable data $(X_1, Y_1), ..., (X_n, Y_n), (X_{n+1}, Y_{n+1})$ with the goal to predict for the point X_{n+1} the corresponding label Y_{n+1}. Exchangeability can be understood in a simplified way as the swap of any two data points. Meaning, the data obtained after swapping (X_{n+1}, Y_{n+1}) with (X_i, Y_i), cannot be distinguished from the non-exchanged bag of data.

We know that Y_{n+1} must live in the label space $\mathcal{Y}$. If we try every possible label $y \in \mathcal{Y}$ for the data point X_{n+1}, then due to the exchangability of the data points the pair (X_{n+1}, Y_{true}) must be exchangeable to the first n data points, i.e. the data point $(X_{n+1}, Y_{true} = y)$ is not "strange" with respect to all other data points, respectively has a low conformity score.

Full Conformal Prediction now directly exploits this principle by training for each possible label $y \in \mathcal{Y}$ a symmetric model f^y on the augmented dataset $(X_1, Y_1), ..., (X_N, Y_N), (X_{n+1}, y)$. In the next step, for each data point of the dataset, the corresponding conformity score is computed as :

$$\begin{aligned} s_i^y &= S\Big((X_i, Y_i), f^y\Big) \quad \text{,for } i = 1, ..., n \\ s_{n+1}^y &= S\Big((X_{n+1}, y), f^y\Big) \end{aligned} \tag{2}$$

At this point, it should be briefly mentioned that the conformity scores can be interpreted as p-values. This formulation, which is more closely orientated to the original work of Vladimir Vovk, is presented in Chapter 5.

Now we define $\hat{q}$ as the $\frac{\lceil (1-\alpha)(n+1) \rceil}{n}$ quantile of the conformity scores $s_1^y, ..., s_n^y$, which is basically the $1 - \alpha$ quantile, with a small correction for the finite sample size:

$$\hat{q}^y = \text{Quantile}\left(s_1^y, ..., s_n^y; \frac{\lceil (1-\alpha)(n+1) \rceil}{n}\right) \tag{3}$$

The prediction set $\mathbf{C}(X_{test})$ is now constructed by collecting all y that are sufficiently consistent with the previous data $(X_1, Y_1), ..., (X_N, Y_N)$:

$$\mathbf{C}(X_{test}) = \Big\{y : s_{n+1}^y \leq \hat{q}^y\Big\} \tag{4}$$

The set constructed in this way satisfies the coverage guarantee (Equation 1). This can be explained by the fact, that if we order the conformal scores $s_1^y, ..., s_n^y$ of the individual data points by magnitude, the score s_{n+1}^y lies with uniform probability of $\frac{1}{n+1}$ between any two of these points (Figure 2). Thus, the set $\mathbf{C}(X_{test})$ exactly contains $1 - \alpha$ of the probabilistic density and therefore satisfies the coverage guarantee.

For Full CP methods, a model must be trained for each $y \in \mathcal{Y}$, which is not feasible for large or continuous label spaces or for computationally intensive models such as neural networks. For other approaches, such as Split CP, a pre-trained model can be used so that this computational effort is not required. However, new data points that were not used during training would need to be utilized, making this approach less efficient in terms of using existing data. Split CP can be described as a special case of Full CP and is presented in detail in Section 2.3.

2.2.3 Score Function

At this point, the reader may continue to wonder how a statistically valid prediction set can be constructed, even if the underlying model and thus the heuristic notation of uncertainty may be arbitrarily bad. Intuitively, it must be stated that all information about the actual problem, the data and the underlying model is contained in the score function. To illustrate, if the scores s_i correctly classify the errors of the model for a

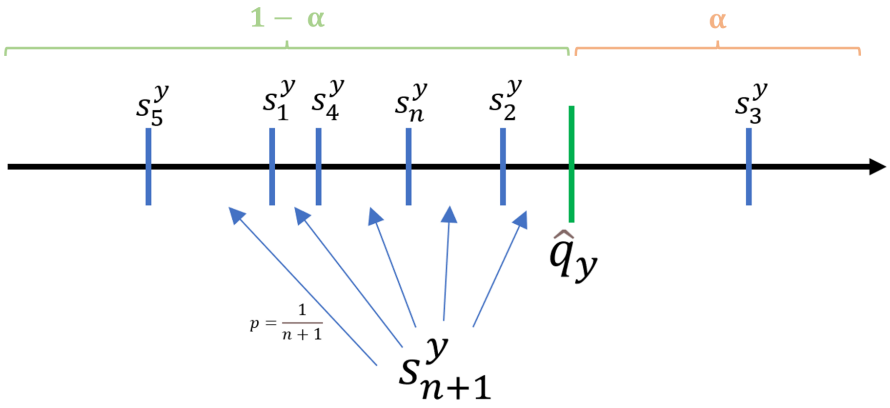


Figure 2: The conformity score $s_{n+1}^y = S\left(X_{n+1}, y, f^y\right)$ for a possible test data point $(X_{n+1}, y), y \in \mathcal{Y}$ lies with uniform probability of $p = \frac{1}{n+1}$ between each of the sorted conformity scores $s_1^y, ..., s_n^y$ of the previous data points $(X_1, Y_1), ..., (X_N, Y_N)$. Thus, all $y : s_{n+1}^y \leq \hat{q}^y$ lies within the $1 - \alpha$ quantile.

given input, this leads to small prediction sets for simple inputs and large prediction sets for difficult inputs. However, if the scores do not correctly reflect this classification of difficult and easy inputs, e.g. because the underlying model only provides an inadequate notation of the uncertainty or because a non-informative score function was selected, the prediction sets become uninformative. In the extreme case, when the scores are just random noise, the resulting prediction sets are also a random sample of the labeling space, but they are large enough to still satisfy the coverage guarantee, i.e. they contain on average $1 - \alpha$ of the labeling space. Such a prediction set is therefore no more informative than guessing without prior information. This intuitive consideration leads to a fundamental property of all conformal methods. Conformal Prediction methods fulfill the coverage guarantee in any case, but the informativeness and thus the usefulness of the prediction sets is determined by the scoring function. Depending on the quality of the scores obtained, which is determined by the choice of the score function and the quality of the underlying model, prediction sets with high informative value (approximate point predictions) or prediction sets with absolutely no informative value (the entire label space) are obtained.

2.3 Split Conformal Prediction

As we have seen, the Full CP approach requires retraining the underlying model for every possible y in the label space $\mathcal{Y}$, which leads to a considerable computational effort. Therefore, recent work on this topic mostly employs the Split CP approach, which can be seen as a special case of Full CP and originally goes back to work by Harris Papadopoulos (Papadopoulos, 2008) (Papadopoulos, Proedrou, Vovk, & Gammerman, 2002) and was introduced under the term inductive Conformal Prediction, in contrast to the Full CP approach, which can be seen as a transductive approach.

2.3.1 Difference Of Inductive And Transductive Learning Systems

The distinction between transductive and inductive learning systems as used here goes back to the work of Wladimir Vapnik (Vapnik, 1998) and should be briefly explained here. Inductive systems generate in a first step from the available training data a general hypothesis which can be understood as a decision rule. With the help of this rule, a prediction for new examples can be deductively derived, without having to consider the training data any further. The advantage is that the derived rule is more compact than the original data, i.e. the information content is compressed, so that it can be stored more efficiently and a prediction based on it can usually be made more quickly. In other words, most of the computational effort is incurred

during learning, also known as eager learning, with predictions being made quickly. Transductive systems, on the other hand, take a shortcut and do not generate a general decision rule but derive the prediction for a new example directly from the training data (Figure 3). Thus, such procedures are computationally inefficient, since they start from scratch for each prediction and, in addition, all training data have to be kept. The actual computational effort is thus incurred in the prediction, while "learning" describes only the storage of the training data (Fontana et al., 2023). Therefore, transductive systems are often referred to as lazy learners.

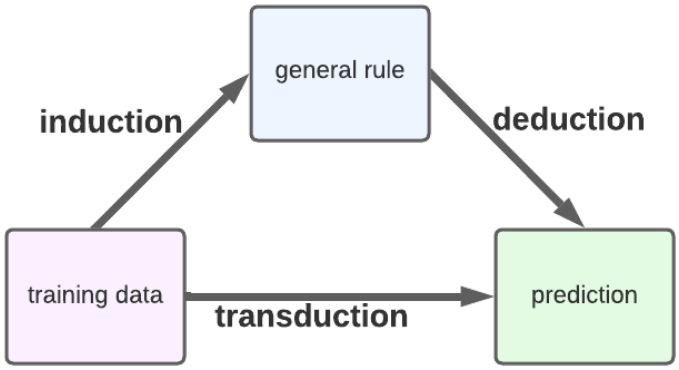


Figure 3: Inductive and transductive learning methods. Inductive learning generates general decision rules from the training data in the learning step and makes new predictions based on them. Transductive learning generates predictions based on the training data and the new data point only at the prediction time.

2.3.2 Principle Of Split Conformal Prediction

In the inductive approach for Conformal Prediction, the training dataset Z is split into two parts, a training dataset Z_{train} and a calibration set $Z_{calib} = (X_1, Y_1), ..., (X_n, Y_n)$. In a first step, an arbitrary underlying model $\hat{f}$ is trained on the training data Z_{train}. It is important to note that the model itself can also be a pre-trained model from any source. Split CP can thus be seen as an additional confidence layer, or wrapper, around any model, providing it with mathematically valid coverage guarantees.

For the application of a Split CP method it is only crucial that an additional calibration dataset exists, which is exchangeable to the used training dataset. A suitable calibration set should generally contain around $|Z_{calib}| = n \sim 1000$ data points. More detailed considerations about the calibration set size can be found in Chapter 11.

In order to construct a prediction set $\mathbf{C}(X_{test})$ with coverage guarantee (Equation 1) from the pre-train model $\hat{f}$, the corresponding conformity scores $s_i = S\Big((X_i, Y_i), \hat{f}\Big)$ are now calculated for each data point $(X_1, Y_1), ..., (X_n, Y_n)$ from the calibration set Z_{calib}. In contrast to Full CP methods, the model $\hat{f}$ is now fixed and thus no longer depends on a test data point X_{n+1}, and therefore all conformity scores s_i can be calculated in advance. Like in the transductive setting, $\hat{q}$ is then calculated as the $\frac{\lceil (1-\alpha)(n+1) \rceil}{n}$ quantile of the empirical conformity scores $s_1, ..., s_n$ and the prediction set for a new data point X_{n+1} is simply calculated as:

$$\mathbf{C}(X_{test}) = \Big\{ y : s(X_{test}, y) \leq \hat{q} \Big\}$$

The prediction sets obtained in this way meet the coverage guarantee, completely independent of the size of the calibration set, the underlying distribution of the data and the correctness of the underlying model (Fontana et al., 2023).

This approach can be derived as a special case from Full CP by using the algorithm for training the model $\hat{f}^y = \text{train}\Big((X_1, Y_1), ..., (X_n, Y_n), (X_{test}, y)\Big)$ in the Full CP method in such a way that it simply ignores the test data point (X_{test}, y). Thus, transductive scores $s_i^y = S((X_i, Y_i), \hat{f}^y)$ are identical to those from the inductive setting $s_i = S((X_i, Y_i), \hat{f})$.

2.3.3 The Conformal Recipe

As described in detail in the previous chapter, each Split CP method follows the same pattern, regardless of the actual underlying problem. Thus, a basic formulation of all Split conformal methods is given here, which will serve as a recipe for all methods presented in the following chapters.

The following steps show the general recipe for all Split CP methods with not necessarily discrete labeled calibration data $(X_1, Y_1), ...(X_n, Y_n)$ and an arbitrary pre-trained model $\hat{f}$ to create a prediction set $\mathbf{C}(X_{test})$ containing the ground truth with probability $1 - \alpha$ (Angelopoulos & Bates, 2021):

1. Identify a heuristic notion of uncertainty provided by $\hat{f}$
2. Define a score function $S\Big(X_i, Y_i; \hat{f}\Big)$ based on the heuristic notion of uncertainty.
3. Compute $\hat{q}$ as the $\frac{\lceil (1-\alpha(n+1) \rceil}{n}$ quantile of the calibration scores
$$s_1 = S\Big(X_1, Y_1; \hat{f}\Big), ..., s_n = S\Big((X_n, Y_n; \hat{f}\Big)$$
4. Calculate the prediction sets for a new data point X_{test} as:
$$\mathbf{C}(X_{test}) = \Big\{y : S(X_{test}, y; \hat{f}) \leq \hat{q}\Big\}$$

2.4 Conditional Coverage

As described in the previous chapters, all conformal methods satisfy the coverage guarantee. To be precise, this means that the prediction set for a new data point X_{test} contains the true label on average with a probability of $1 - \alpha$. This coverage guarantee is also called marginal coverage, since it is not adaptive to the difficulty of a single point X_{test} and can therefore easily be estimated as the percentage of the prediction sets that covers the ground truth for a new dataset (Molnar, 2023). For the sake of readability, the definition for marginal coverage will be repeated here (without considering the finite sample size correction for the upper bound).

Definition 2. ***Marginal coverage***

$$1 - \alpha \quad \leq \quad \mathbb{P}\Big(\mathbf{y}_{test} \in \mathbf{C}(\mathbf{X}_{test})\Big)$$

However, we have no guarantee that the coverage is also $1 - \alpha$ for each individual data point, not even for specific groups such as individual classes. For example, consider a classification task with equal numbers of data points for men and women, where the prediction set for men contains the ground truth in all cases (easy inputs), but for women it contains it on average only in 90% of the cases (difficult inputs). The model still satisfies the coverage guarantee of 95%. Nevertheless, this property is not what is desired in many situations. Rather, we would like the coverage guarantee to apply not only on average, but also to specific subsets of the data. This property is commonly referred to as conditional coverage and can be formalized as follows (Angelopoulos & Bates, 2021) (See also Figure 4 for a intuitive description):

Definition 3. ***Conditional coverage***

$$1 - \alpha \quad \leq \quad \mathbb{P}\Big(\mathbf{y}_{test} \in \mathbf{C}(\mathbf{X}_{test}) \mid X_{test}\Big)$$

This means that for every single input X_{test} we provide prediction sets with $1-\alpha$ coverage. This property is much stronger than marginal coverage and cannot be guaranteed in general. However, there are some practically relevant special cases, such as class or group conditional coverage, for which a guarantee can be given. There are also conforming methods that approximate conditional coverage better than others. Such methods are generally referred to as adaptive conformal methods, as they take into account the higher uncertainty for difficult data points with larger prediction sets.

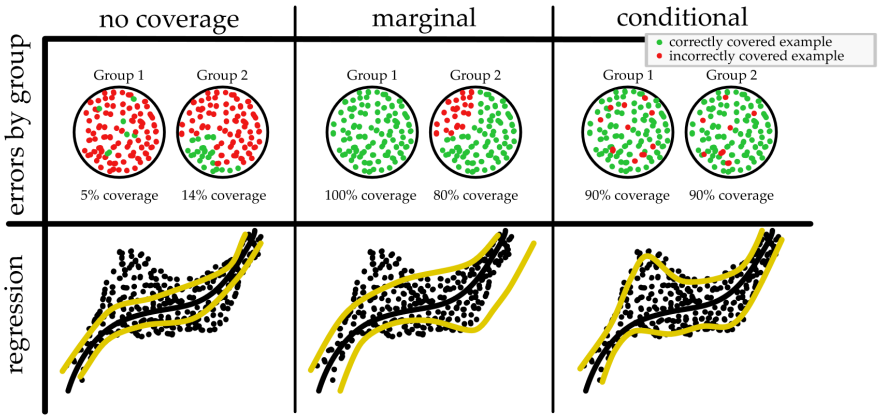


Figure 4: Overview of prediction set with no coverage, marginal and conditional coverage in the classification and regression case. In the marginal coverage case all errors happens in one group of data, or region of X. Conditional coverage enforces evenly distributed errors in all classes. Figure is taken from (Angelopoulos & Bates, 2021).

2.4.1 Mondrian Conformal Prediction

As seen in the previous chapter, conforming predictors within certain subsets of the data do not guarantee coverage. The proportion of errors in one group may be greater than the target significance level, which will result in fewer errors in other groups. The Mondrian Conformal Predictors first proposed by Vovk (Vovk, Lindsay, Nouretdinov, & Gammerman, 2003) solve this problem by first dividing the data Z into certain categories or groups $g_1, ..., g_m \in G$. For this purpose, a measurable function $M : Z \to G$ is formally introduced. A group $g_i = M(Z_i)$ can depend on the other data points, but ignores their order. Such a function is also called Mondrian taxonomy after the Dutch painter Piet Mondrian, because the partitioning of the data Z is reminiscent of his grid-like paintings (Figure 5).

Using Mondrian taxonomy, we can define the group-balance coverage:

Definition 4. ***Group-balanced coverage***

$$1-\alpha \quad \leq \quad \mathbb{P}\Big(\mathbf{y}_{test} \in \mathbf{C}(\mathbf{X}_{test}) \mid X_{test} = g_i\Big) \quad : \quad \forall g \in G$$

This means we have $1-\alpha$ coverage in all groups. This can be easily obtained by first dividing the data points into groups using the Mondrian taxonomy and then applying a standard conformal procedure for each group. That is we compute the scores $s_i^{(g)}$ for each point in the calibration set, where each data point belongs to a group g. We then calculate the $1-\alpha$ quantile $\hat{q}^{(g)}$ for each group:

$$\hat{q}^{(g)} = \text{Quantile}\left(s_1(g), ..., s_n(g); \frac{\lceil (1-\alpha)(n(g)+1) \rceil}{n(g)}\right) \tag{5}$$

where $n(g)$ is the number of data points of the corresponding group. The prediction sets are now formed by using the relevant quantile for the new data point $X_{test}^{(g)}$:

$$\mathbf{C}(X_{test}^{(g)}) = \left\{y : s(X_{test}^{(g)}, y) \leq \hat{q}^{(g)}\right\}$$

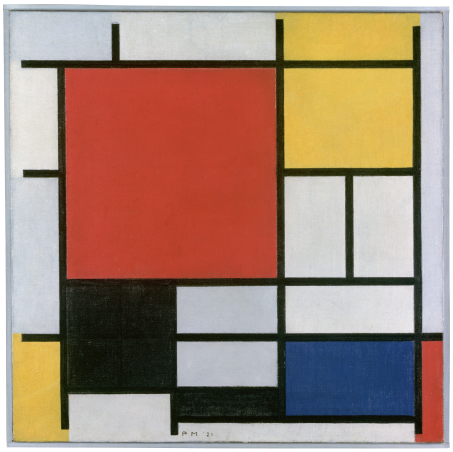


Figure 5: Piet Mondrian. Compositie met groot rood vlak, geel, zwart, grijs en blauw, 1921, Art Museum Den Haag

This method has the not insignificant disadvantage that the calibration set has to be divided and therefore fewer calibration points are available for determining the quantiles, making the estimate less reliable. Even if this does not violate the coverage guarantee (on average), it leads to a significantly higher variance (Angelopoulos & Bates, 2021). In practice, this means that the need for calibration points increases linearly with the number of different groups.

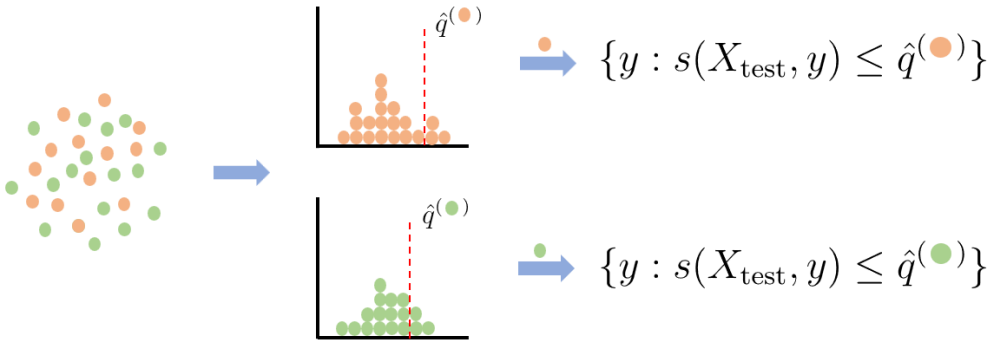


Figure 6: Schematic of the group-balanced Conformal Prediction. To calculate the calibration scores, the calibration set is stratified depending on the group (green, orange). Since the group for a test point X_{test} is known a priori, the corresponding threshold $\hat{q}$ can then be used (Angelopoulos & Bates, 2021).

2.4.2 Class-Balanced Coverage

For classification tasks, coverage within each of the classes to be predicted should often be guaranteed. This leads to class-balanced coverage:

Definition 5. ***Class-balanced coverage***

$$1 - \alpha \quad \leq \quad \mathbb{P}\Big(\mathbf{y}_{test} \in \mathbf{C}(\mathbf{X}_{test}) \mid Y_{test} = y\Big)$$

At first sight this problem seems to be similar to the group-balanced coverage and as we will see the solution is also quite similar. First, the corresponding quantile $\hat{q}^y$ is calculated separately for each class as in the group balanced case. The problem arises when constructing the prediction set for a new data point, because the corresponding class is not known. The solution is to use all class-wise predictors and the union of the resulting sets as the result.

$$\mathbf{C}(X_{test}) = \left\{ y : s(X_{test}, y) \leq \hat{q}^y \right\}$$

This procedure guarantees class-balanced coverage, more precisely, the procedure exactly (Angelopoulos & Bates, 2021) satisfies the coverage for the most difficult class to classify, that is, the class y with the largest quantile $\hat{q}^y$. For all other classes, however, the coverage can be significantly higher than $1 - \alpha$ and thus the resulting prediction sets are usually significantly larger (Molnar, 2023).

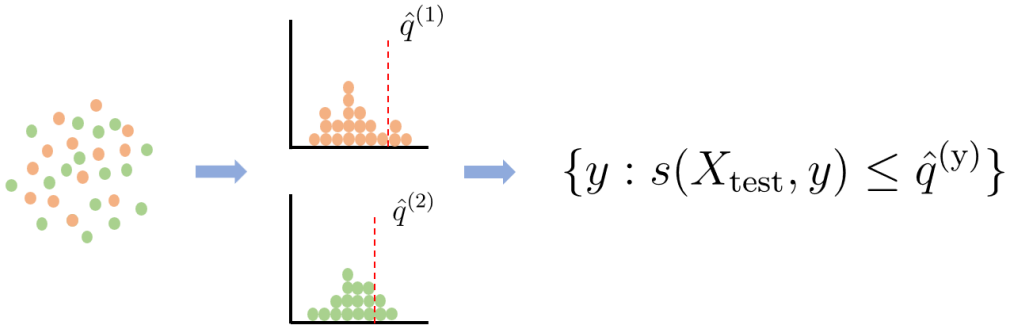


Figure 7: Schematic of the class-balanced Conformal Prediction. To calculate the calibration scores, the calibration set is stratified depending on the class. Since the class for a test point X_{test} is not known a priori, the corresponding prediction set is the union over all of all conformal classes (Derhacobian et al., 2022).

2.4.3 Size Of The Calibration Set

As mentioned above, the coverage guarantee holds regardless of the size of the calibration set. However, it is intuitively clear that larger calibration sets lead to more stable conformal predictors. This intuition is correct, which can be attributed to the fact that the coverage guarantee:

$$1 - \alpha \quad \leq \quad \mathbb{P}\Big(\mathbf{y}_{test} \in \mathbf{C}(\mathbf{x}_{test})\Big) \quad \leq \quad 1 - \alpha + \frac{1}{n+1}$$

holds for a coverage of $1 - \alpha$ on average over the randomness in the calibration set. That is, with a fixed finite calibration set, the coverage will not be exactly $1 - \alpha$ even evaluated on infinite test points, but the deviation from this value will decrease as the calibration set size increases. This insight may seem sobering at first glance, but the fluctuation of coverage can be directly analyzed and controlled, since the coverage of Conformal Prediction conditionally on the calibration set is a random quantity following a Beta distribution (Vovk, 2012) (Figure 8):

$$\mathbb{P}\Big(Y_{test} \in C(X_{test}) \mid \{(X_i, Y_i)\}_{i=1}^{y}\Big) \sim Beta(n + 1 - l, l) \ , \ l = \lfloor (n+1)\alpha \rfloor$$

With this statement we can now precisely determine the expected empirical coverage (evaluated with infinitely many test points). More precisely, it allows us to say how many calibration points are needed for a coverage of $1 - \alpha \pm \epsilon$ with a probability of $1 - \delta$. Table 2.4.3 shows practically relevant example calibration set sizes (Angelopoulos & Bates, 2021):

From this consideration it can be deduced that for most applications calibration sets with about 1000 data points guarantee sufficient coverage.

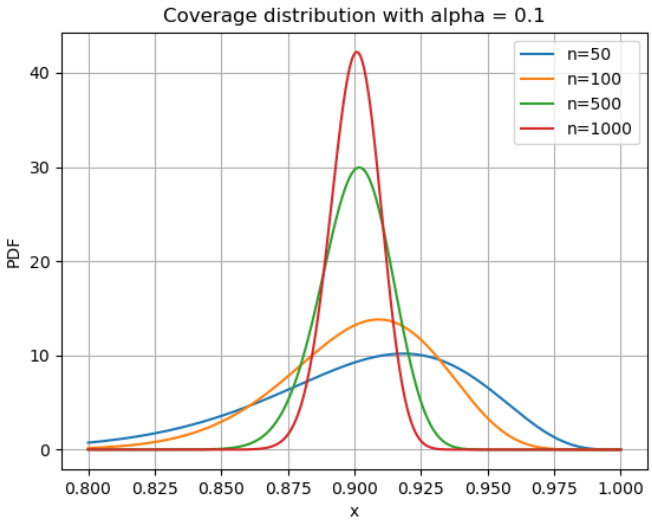


Figure 8: Distribution over the coverage considering a infinite validation set and different calibration set sizes n, following a beta distribution.

ϵ	0.1	0.05	0.01	0.001
$n(\epsilon)$	22	102	9812	244390

Table 1: Required calibrations set size $n(\epsilon)$ for a coverage of $1 - 0.9 \pm \epsilon$ with probability $\delta = 0.1$.

2.5 Classification

Up to this point, the Conformal Prediction framework has been introduced in general. In the following chapter, several concrete conformal methods for classification tasks are presented. All approaches follow the conformal recipe (2.3.3) and differ in the choice of the score function.

2.5.1 Least Ambiguous Set-valued Classifier (lac)

Many classifiers inherently provide a natural notation about the probability of the individual classes, such as the softmax layer of a neural network. These outputs are generally not calibrated, but can be transformed into valid prediction sets using a conformal method. Here we define the score function for the (softmax) outputs of a model $\hat{f}$ as follows:

$$s_i = 1 - \hat{f}^{y=y_i}(x_i)$$

Only the score of the true label (noted as $\hat{f}^{y=y_i}(x_i)$) is used to calculate the scores on the calibration set. Thus, the prediction set includes all classes for which the corresponding softmax output is greater than $1 - \hat{q}$ (Angelopoulos & Bates, 2021):

$$\mathbf{C}(X_{test}) = \left\{ y : \hat{f}(X_{test})_y \geq 1 - \hat{q} \right\}$$

2.6 Adaptive Prediction Sets (aps)

The least ambiguous set-valued classifier score method produces prediction sets with the smallest average size (Sadinle, Lei, & Wasserman, 2019), but it is not adaptive. In practice he method tends to undercover

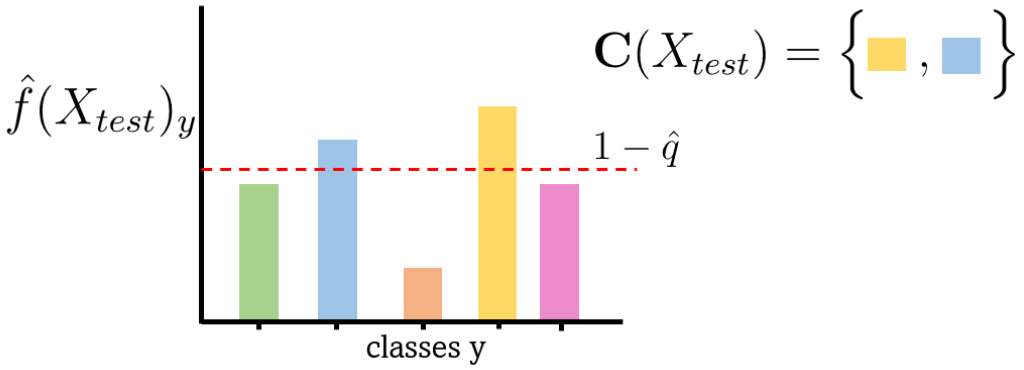


Figure 9: Constructing the prediction set $\mathbf{C}(X_{test})$ for the least ambiguous set-valued method.

hard subgroups and overcover easy subgroups. This is due to the fact that it uses only the outputs of the true class and ignores all others. Adaptive prediction sets, on the other hand, sum up all outputs, starting with the largest score up to the true class. Let $\pi(x)$ the permutation of the classes $1, ..., c$ that sorts $\hat{f}(X_{test})$ from most likely to least likely and y the true label, then the score function can be strongly compressed as follows:(Angelopoulos, Bates, Malik, & Jordan, 2020):

$$s(x, y) = \sum_{j=1}^{c} \hat{f}(x)_{\pi_j(x)} \ , \text{ where } y = \pi_c(x)$$

In an example, a classification algorithm outputs the (softmax) output of cat=0.3, lion=0.6 and dog=0.1 for a cat image (which tends to be hard to classify). The naive method calculates $s = 1 - 0.3 = 0.7$ and ignores the lion output. The adaptive method takes this into account and calculates $s = 1 - 0.6 - 0.3 = 0.9$. For easy to classify examples aps gives comparable scores as the lac method, but for difficult examples larger prediction sets are generated (Molnar, 2023).

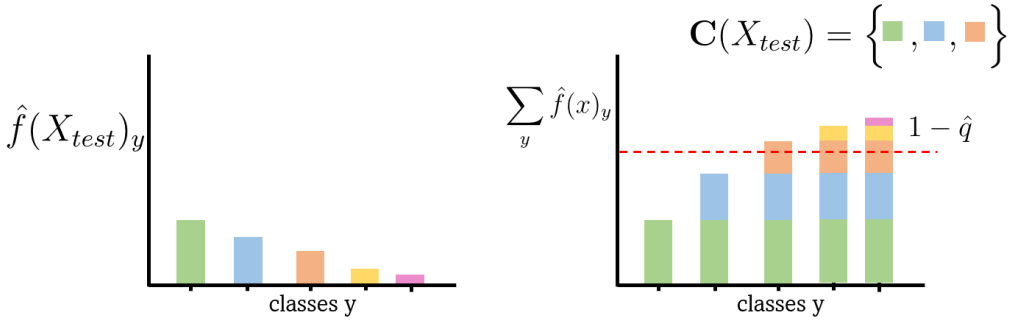


Figure 10: Constructing the prediction set $\mathbf{C}(X_{test})$ for the adaptive prediction set method. The prediction set includes not only the calibration scores of the true label but takes all top-scoring classes into account.

2.6.1 Adaptive classification with Split Conformal Calibration

The description given up to this point was more for an intuitive understanding of how adaptive prediction sets can be constructed. However, some details like tie-breaking to guarantee marginal coverage are still missing for the actual formulation of the algorithm. In the following, a detailed formulation for adaptive classification with Split conformal calibration is given. This approach then leads directly to the CV+ and Jackknife+ methods which use the available data more efficiently (Section 2.7).

Considering an oracle classifier with perfect knowledge of the conditional distribution $\mathbb{P}_{Y|X}$, the construction of optimal prediction sets $C_\alpha^{\text{oracle}}(x_{test})$ would be trivial. Let $f_y(x) = \mathbb{P}[Y = y, X = y]$, $\forall y \in \mathcal{Y}$ with the order statistic $f_{(1)}(x) \geq f_{(2)}(x) \geq ... \geq f_{(C)}(x)$, then, without considering ties by now, we can define the generalized conditional quantile function for an arbitrary $\tau \in [0, 1]$:

$$L(x; f, \tau) = \min\{c \in 1, ..., C \ : \ f_{(1)}(x) + f_1(x) + ... + f_c(x) \geq \tau\} \tag{6}$$

and the resulting prediction set:

$$C_\alpha^{\text{oracle}}(x) = \{\text{corresponding } y \text{ for the } L(x; f, 1-\alpha) \text{ largest } f_y(x)\} \tag{7}$$

With the already used example $f_{(cat)}(x)0.3$, $f_{(lion)}(x)0.3$ and $f_{(dog)}(x)0.3$ we get for $L(x; 0.9) = 2$ the prediction set $C_\alpha^{\text{oracle}}(x) = 1, 2$ and for $L(x; 0.5) = 1$ $C_\alpha^{\text{oracle}}(x) = 2$. The previous formulation does not handle equal values for $f_{(c-1)}(x)$ and $f_{(c)}(x)$. The only way to theoretically guarantee coverage is to randomly add or discard the last label, which will simply break ties at random. For this we define the following function:

$$S(x, u; f, \tau) = \begin{cases} \text{corresponding } y \text{ for the } L(x; f, 1-\alpha) - 1 \text{ largest } f_y(x), & \text{if } u \geq V(x; f, \tau) \\ \text{corresponding } y \text{ for the } L(x; f, 1-\alpha) \text{ largest } f_y(x), & \text{otherwise} \end{cases} \tag{8}$$

where:

$$u \sim \text{Uniform}$$

$$V(x; f, \tau) = \frac{1}{f_{(L(x;f,\tau))}(x)} \left[\sum_{c=1}^{L(x;f,\tau)} f_{(c)}(x) - \tau \right] \tag{9}$$

With this formulation, we obtain tighter, tie breaking, and valid prediction sets (Romano, Sesia, & Candes, 2020):

$$C_\alpha^{\text{oracle}}(x_{test}) = S(x_{test}, U; f, 1-\alpha) \tag{10}$$

It is intuitively clear that any trained model$\hat{f}_y(x)$ can only approximate $f_y(x)$ and even be arbitrarily bad in theory, and thus the oracle approach cannot be used. However, the following method can construct valid prediction sets for any model $\hat{f}_y(x)$ by fitting the threshold τ based on a calibration set. The only restriction that applies to $\hat{f}_y(x)$ is that the algorithm treats the data points interchangeably, i.e. invariant to their order and that the output class probabilities are normalized, that is $\hat{f}_y(x) \in [0, 1]$, $\sum_{y=1}^{C} \hat{f}_y(x) = 1$, $\forall x, y$.

To calibrate τ on the basis of a hold-out set, we define the so-called generalized inverse quantile function $E(x, y, u; \hat{f}$ that computes the smallest possible value for τ such that $S(x, u; f, \tau)$ contains the true label y.

$$E(x, y, u; \hat{f}) = \min\{\tau \in [0, 1] : y \in S(x, u; f, \tau)\} \tag{11}$$

By this construction, the scores computed on the hold-out set (X_i, Y_i) are $E_i = E(X_i, Y_i, U_i; \hat{f})$ are uniformly distributed conditional on X for $f = \hat{f}$ and U_i independent uniform random variable. This property is not present in many other conformity score functions and makes the scores naturally comparable between different samples. In this way, we can now construct prediction sets with a provable marginal coverage guarantee of τ close to the $1 - \alpha$ quantile of $\{E_i\}_{i \in \mathbf{I}_2}$, where $\mathbf{I}_2$ is the hold-out or calibration set that was not used to train $\hat{f}$ (Romano et al., 2020).

The algorithm for calculating adaptive classification with Split conformal calibration is presented in 1.

The algorithms satisfies the marginal coverage guarantee as desired (for approximately distinct values for E_i the upper bound also holds):

$$1 - \alpha \leq \mathbb{P}\Big[Y_{test} \in C_{n,\alpha}^{\text{SC}}(x_{test})\Big] \leq 1 - \alpha + \frac{1}{|X_{calib}| + 1}$$

Algorithm 1 Adaptive Classification with Split Conformal Calibration

Input: data $\{X_i, Y_i\}_{i=1}^n$, X_{test} , model $\hat{f}$, α
X_{train}, $X_{calib} \leftarrow$ train_test_split($\{X_i, Y_i\}_{i=1}^n$)
Train $\hat{f}$ on X_{train}
Compute $E_i = E(x_i, y_i, u_i; \hat{f})$ for each $x_i, y_i \in X_{calib}$ with function 11
Compute $\hat{Q}_{1-\alpha}(\{E_i\}_{i \in X_{calib}})$ as the $\lceil (1-\alpha)(1-|X_{calib}|) \rceil$th largest value in E_i
Output the prediction set:

$$C_{n,\alpha}^{\mathrm{SC}}(x_{test}) = S(x_{test}, u_{test}; \hat{f}, \hat{Q}_{1-\alpha}(\{E_i\}_{i \in X_{calib}}))$$

using the score function S defined in 8.

2.7 Adaptive Classification with Cross-Validation+ and Jackknife+ Calibration

All Split CP methods have in common that they are not efficient, i.e. they do not use all data for the training of the underlying model, but require a hold-out dataset for calibration purposes. Even if for many applications about 1000 data points are sufficient (Angelopoulos & Bates, 2021) these are not always available. Full CP is a method to use all data efficiently, but it requires considerable computational effort and is therefore not suitable for computationally intensive models. However, CV+ and Jackknife+ are two methods that offer a compromise between computational complexity and data efficiency. We first consider CV+ and then introduce Jackknife+ as a special case of CV+.

CV+ uses a cross-validation approach and partitions the data into k distinct splits $\mathcal{I}_1, \mathcal{I}_2, ..., \mathcal{I}_k$. On each of the splits, the corresponding model $\hat{f}^{k(i)} = \hat{f}(\{X_i, Y_i\}_{i \in \{1,...,n\} \setminus \mathcal{I}_k})$ is trained. To form the prediction sets, it is now iterated over each possible label $y \in \mathcal{Y}$ and the y are unified to form the prediction set $C_{n,\alpha}^{\mathrm{CV+}}(x_{test})$ whose score $E(x_{test}, y, u_{test}; \hat{f}^{k(i)})$ is smaller than $\lceil (1-\alpha)(1-|X_{calib}|) \rceil$ hold-out scores $E(x_i, y_i, u_i; \hat{f}^{k(i)})$.

Algorithm 2 Adaptive Classification with CV+ Calibration

Input: data $\{X_i, Y_i\}_{i=1}^n$, X_{test} , model $\hat{f}$, number of splits $K \leq n$, α
Split data into k random distinct subsets $\mathcal{I}_1, \mathcal{I}_2, ..., \mathcal{I}_k$
for $k \in \{1, ..., k\}$:
    Train $\hat{f}^{k(i)}$ on $\{X_i, Y_i\}_{i \in \{1,...,n\} \setminus \mathcal{I}_k}$
Output the prediction set:

$$C_{n,\alpha}^{\mathrm{CV+}}(x_{n+1}) = \Bigg\{ y \in \mathcal{Y} : \sum_{n=1}^{n} \mathbf{1}\Big[E(x_i, y_i, u_i; \hat{f}^{k(i)}) \leq E(x_{n+1}, y_{n+1}, u_{n+1}; \hat{f}^{k(i)}) \Big] \leq \lceil (1-\alpha)(1-|n|) \rceil \Bigg\}$$

where $k(i) \in \{1, .., k\}$ denotes the fold containing the ith sample and using the function E defined in 11.

The algorithm theoretically fulfills a slightly weakened coverage guarantee:

$$\mathbb{P}\Big[Y_{test} \in C_{n,\alpha}^{\mathrm{CV+}}(x_{test}) \Big] \geq 1 - 2\alpha - \min\Big\{ \frac{2(1 - 1 \backslash K}{n \backslash K + 1}, \frac{1 - K \backslash n}{K + 1} \Big\}$$

In the special case for $k = n$ we speak of the Jackknife+ method with simplified bound of:

$$\mathbb{P}\Big[Y_{test} \in C_{n,\alpha}^{\mathrm{CV+}}(x_{test}) \Big] \geq 1 - 2\alpha$$

The authors of the method point out, however, that in practice for these methods mostly an empirical coverage of $1-\alpha$ instead of $1-2\alpha$ is achieved. The following more conservative definition for the prediction set satisfies the coverage guarantee of $1-\alpha$ also in theory but is resulting in larger prediction sets:

$$C_{n,\alpha}^{\text{CV+mm}}(x_{n+1}) = \Bigg\{ y \in \mathcal{Y} : \sum_{n=1}^{n} \mathbf{1}\Big[E(x_i, y_i, u_i; \hat{f}^{k(i)}) \leq \min_{j \in \{1,\ldots,n\}} E(x_{n+1}, y_{n+1}, u_{n+1}; \hat{f}^{k(i)}) \Big] \leq \lceil (1-\alpha)(1-|n|) \rceil \Bigg\}$$

2.7.1 Regularized Adaptive Prediction Sets (RAPS)

In classification tasks with many classes, APS tends to generate large prediction sets, especially when many classes are potentially possible. This is mainly due to the fact that there can be a long tail of (noise) classes with low probability. In these cases it can be helpful to use an additional regularization term. This is achieved by using a regularization term λ to penalize classes with a rank higher than k_{reg}. This has the effect that the prediction sets for classifications tasks with many labels become on average smaller than those produced by APS (Angelopoulos et al., 2020).

2.8 Regression

Although the focus of this thesis is on classification tasks, the basics of conformal regression methods are presented in the following chapter. These provide intervals that, like all conformal methods, adhere to the coverage guarantee 1.

2.8.1 Naive Conformal Regression

In the simplest case, the residuals of a regression model $\hat{f}$ are used directly as a score function:

$$s_i = |y_i - \hat{f}(x_i)|$$

As in all conformal methods, $\hat{q}$ is now calculated as the $\frac{\lceil (1-\alpha(n+1)) \rceil}{n}$ quantile. The prediction interval is then calculated as follows:

$$\mathbf{C}(X_{test}) = \Big[\hat{f}(X_{test}) - \hat{q} \ , \ \hat{f}(X_{test}) + \hat{q} \ \Big]$$

This method has the disadvantage that the resulting intervals all have exactly the same size and the method is therefore not adaptive in any way (Romano, Patterson, & Candes, 2019).

2.8.2 Conformal Regression For Scalar Uncertainty Estimates

To obtain more adaptive intervals, standardized residuals can be used. For this purpose, a scalar notation for the uncertainty of the residuals is employed.

A typical way construct those values is to train a second model $\hat{r}$ that estimates the residuals of the actual regression model $\hat{f}$. The intuition here is that if such a model $\hat{r}$ were a perfect oracle for the uncertainty of the residuals, the interval $\Big[\hat{f}(X_{test}) - \hat{r}(X_{test}) \ , \ \hat{f}(X_{test}) + \hat{r}(X_{test}) \ \Big]$ would have perfect coverage. However, $\hat{r}$ is often a poor estimator in practice and we need to adjust the intervals using Conformal Prediction (Romano et al., 2019).

Generally speaking, we consider a second function $u(x_i)$, where larger values indicate greater uncertainty. This function can, for example, also describe the variance between an ensemble of models or the variance after slight input perturbations. With this heuristic notation of the uncertainty $u(x_i)$ the score function $s(x_i, y_i)$ is defined as follows:

$$s_i = \frac{|y_i - \hat{f}(x_i)|}{u(x_i)}$$

This uncertainty function can be taken as a correction factor for uncertainty $s_i * u(x_i) = |y_i - \hat{f}(x_i)|$. Subsequently, $\hat{q}$ is calculated as the $\frac{\lceil (1-\alpha(n+1)) \rceil}{n}$ quantile of the scores, resulting in the following prediction sets:

$$\mathbf{C}(x_{test}) = \Big[\hat{f}(x_{test}) - u(x_{test})\hat{q} \ , \ \hat{f}(x_{test}) + u(x_{test})\hat{q} \ \Big]$$

These thus fulfill the desired coverage guarantee:

$$\mathbb{P}\Big[s(X_{test}, y_{test}) \leq \hat{q}\Big] \geq 1-\alpha \quad \Rightarrow \quad \mathbb{P}\Big[\Big|y_{test} - \hat{f}(X_{test})\Big| \leq u(X_{test})\hat{q}\Big] \geq 1-\alpha$$

The prediction sets generated in this way are symmetric with respect to individual predictions $\hat{f}$, but it is not necessarily the case that the quantiles of uncertainty (e.g., the variance of the models) are directly related to the quantiles of label distribution. That is, they do not necessarily scale properly with α (Angelopoulos & Bates, 2021). For this reason, the literature points out that the conformalized quantile regression estimates the label distribution directly and thus has the better uncertainty heuristic, which can also be shown empirically (Angelopoulos, Kohli, Bates, Jordan, Malik, Alshaabi, Upadhyayula, & Romano, 2022).

2.8.3 Conformalized Quantile Regression

Many regression models can be easily transformed into quantile regressors by using a quantile loss , also known as pinnball loss. These quantile regression methods not only provide a point estimation but try to determine the γ quantile of a distribution $Y_{test}|X_{test} = x$ for each possible value of x. Conformal methods based on such models often yield better results in practice than the methods presented in the previous chapter, since their heuristic notation of uncertainty is directly related to the label space (Romano et al., 2019).

For the true quantiles $t_\gamma(x)$, the set $[t_\alpha, t_{1-alpha}]$ would have to have exactly a coverage of 2α, since by definition a fraction of α must be above t_α and below $t_{1-alpha}$, respectively. However, there is no guarantee that the calculated quantiles are correct. Thus, Conformal Prediction can be used to calibrate them.

Let $\hat{t}_{\alpha/2}(x_i), \hat{t}_{1-alpha/2}(x_i)$ be the output of any quantile regression for the data point x_i, then the associated score function $s(x_i, y_i)$ can be defined as the distance from y_i to the nearest quantile:

$$s(x_i, y_i) = \max\Big\{\hat{t}_{\alpha/2}(x_i) - y_i \ , \ y_i - \hat{t}_{1-\alpha/2}(x_i)\Big\}$$

As in all conformal methods, $\hat{q}$ is now calculated as the $\frac{\lceil (1-\alpha(n+1)) \rceil}{n}$ quantile of the scores. The associated prediction intervals are then determined as follows:

$$\mathbf{C}(x_{test}) = \Big[\hat{t}_{\alpha/2}(x_i) - \hat{q} \ , \ \hat{t}_{1-\alpha/2}(x_i) + \hat{q}\Big]$$

Conformalized quantile regression can thus be understood as increasing or decreasing the interval provided by the underlying model for certain ranges of the label space to meet the coverage guarantee (Romano et al., 2019).

3. Conformal Change Point Detection

Almost all machine learning models are based on the assumption that the training data comes from the same distribution as the test data. If this assumption is violated, the model's predictions often become useless and the model has to be retrained. In practice, such a change in the distribution, known as distribution shift, can occur unpredictably, so it is essential to have a method to recognise such a change point, i.e. the change in the underlying distribution of the data. In other words, change point detection is about the question of whether the iid assumption holds. One way of recognising a change point in an online setting is to use conformal test martingales (CTM) (Wang, Lu, Wang, Zhuang, & Wang, 2023). These are briefly explained below.

Consider a sequence of test data $(z_1, z_2, ..., z_{n+1})$ and an inductive conformity measure function S that for each data point provides a measure $s_i = S(z_i)$ of how strange the data point is in relation to the other data points. As in all CP methods, S is usually based on a model that depends on the training data. With the help of S we can now calculate the p-value for z_{n+1} as follows:

$$p_{n+1} = \frac{|\{i|s_i > s_{n+1}\}| + \theta_{n+1}|\{i|s_i = s_{n+1}\}|}{n}$$

, where $\theta_{n+1} \in [0, 1]$ is a uniform random number independent of $(z_1, z_2, ..., z_{n+1})$. The core of the CTM is now that if the iid assumption for the data points $(z_1, z_2, ..., z_{n+1})$ is correct, the p-values are independent

of the observations and uniform distributed in $[0, 1]$. The corresponding theorem and detailed theoretical background can be found in (Vovk, Petej, Nouretdinov, Ahlberg, Carlsson, & Gammerman, 2021).

Based on the p-values, we can now formulate the conformal test martingale M_{n+1} as follows:

$$M_{n+1} = F(p_1, p_2, ..., p_{n+1})$$

, where $F : [0, 1]^{n+1} \to [0, \infty]$ is a measurable "betting" function that in a certain sense bets against the assumption that the p-values are uniform distributed. More precisely, it is a so-called betting martingale function for which the following hold by definition for each sequence $p_1, p_2, ..., p_n \ , \in [0, 1]^n$:

$$\int_0^1 F(p_1, ..., p_n, p_{n+1})du = F(p_1, ..., p_n)$$

Thus, like all martingales, CTMs fulfil the following property:

$$\mathbb{E}(M_n|S_1, ..., S_{n+1}) = S_{n-1}$$

Vovk has proposed the following betting martingale function called Simple Jumper for the formulation of the CTM:

$$F(p_1, p_2, ..., p_{n+1}) = \int \prod_{i=1}^{n} f_{\epsilon_i}(p_i)\mu(d(\epsilon_1, \epsilon_2, ...))$$

with

$$f_{\epsilon_i}(p) = 1 + \epsilon(p - 0.5)$$

Here, μ is a probability measure on $\{-1, 0, 1\}^\infty$ defined for a Markov chain with state space $\{-1, 0, 1\}$ and the parameter J. The initial state is $\epsilon_0 = -1, 0, 1$ with equal probability and the transition function remains in the same state with probability $1 - J$ and changes with probability J to a random state of $\{-1, 0, 1\}$. F can be calculated efficiently using algorithm 3 (Vovk et al., 2021):

Algorithm 3 Simple Jumper for CTM

Input: $p_1, p_2, ..., p_{n+1}$, parameter J
Output CTM $M_1, M_2, ..., M_{n+1}$:
  Set: $S_0 = 1, C_{-1} = C_0 = C_1 = \frac{1}{3}$ and $C = 1$
  for $i = [1, n + 1]$ do:
    for $\epsilon \in \{-1, 0, 1\}$ do:
      $C_\epsilon = (1 - J)C_\epsilon + (J/3)C$
  for $\epsilon \in \{-1, 0, 1\}$ **do**
    $C_\epsilon = C_\epsilon f_\epsilon(p_n)$
  $C = C_{-1} + C_0 + C_1$
  $M_n = C$

The Simple Jumper martingales fulfil Ville's inequality:

$$P(\exists i : M_i \geq c) \leq \frac{1}{c}$$

for any $c > 1$. This property can now be used to detect change points. Consider a sequence of random variables $z_1, z_2, ...$ and of which the first $(z_1, ..., z_{\theta-1})$ come from the distribution $f_0(z)$ and the remaining variables $(z_\theta, z_{\theta+1}, ...)$ come from the distribution $f_1(z)$. The distribution thus changes from $f_0(z)$ to $f_1(z)$ at the change-point θ. To check whether the iid assumption can be rejected for $z_1, z_2, ...$ with a false alarm rate of $\alpha = 1/c$, we investigate whether $M_i \geq c$ applies to the CTM M_i after the change point θ. Under the iid assumption, the CTMs should remain constant and not increase significantly.

To the best of my knowledge, CTMs are the only online change point detection method that is equipped with such a validity guarantee for the false alarm rate. However, this guarantee is achieved because the effectiveness, i.e. how quickly a change in the underlying distribution can be detected after the change point, is lower than for other methods such as *CUMSUM* or *Shiryaev-Roberts* procedure (Vovk et al., 2021).

Even if the method described applies in principle to any conformity score function S, the effectiveness, i.e. how quickly the change point can be detected, depends considerably on the function. Vovk has proposed the following conformity measure function for a classic supervised regression problem in which the underlying distribution of the data changes abruptly:

$$s_i = y_i - \hat{y}_i$$

Here y_i is the true label and $\hat{y}_i$ is the prediction provided by the underlying model for the data point x_i. Contrary to intuition, it turned out that for the problem under investigation this conformity score is more effective than $s_i = |y_i - \hat{y}_i|$ (Vovk et al., 2021). In Vovk's work, a classic regression model (e.g. Random Forest, 1NN, MLP) was first trained on a suitable training set and then the CTM was trained online on approximately 2000 data points from the training distribution and used around 1000 data points for calibration. The distribution shift was then performed and the method was able to detect the change point with a false alarm rate of 1% after approx. 30 data points. Another score function was proposed by (Wang et al., 2023) for a domain change in geospartial object detectors. This used a concatenated vector $v_i = V(x_i)$ from the output of the underlying base CNN layer as image-level representation and the logit vector as instance-level representation. The 1-nearest neighbour score of the current test point $v_i = V(x_i)$ and the calibration data then served as the nonconformity measure:

$$s_i = S(z_i) = \min_{j \in \text{cal}} \sqrt{(V(x_i) - V(x_j))^2}$$

4. Evaluating Conformal Prediction Methods

Up to this point, various conformal methods have been described. Even if CP is equipped with a rigid statistical guarantee, every new application should be evaluated. At least the following properties should be checked.

1. **Correctness Check.** Correctness checks test whether the coverage guarantee (Equation 1) is actually adhered to by the implemented method. On the one hand, this may not be the case due to errors in the implementation. On the other hand, the required exchangeability of the data may not be given. This case is severe, as no conventional CP method can be used in this case. The coverage check must take into account the variability of the coverage in relation to the finite available test data points and, in the case of split methods, to the finite calibration set.
2. **Evaluating Adaptivity.** The conformal coverage guarantee is only marginal, i.e. with respect to the variance in the calibration data. In many cases, it turns out that the smallest average prediction sets are not necessarily the most meaningful. A good conformal predictor should provide small prediction sets for simple inputs and large prediction sets for heavy inputs. This property is generally referred to as adaptivity and should also be evaluated.

4.1 Correctness Check

Compliance with the coverage guarantee can obviously be checked by examining the coverage of the conformal method for R different splits of the available data in training and calibration data. The coverage calculated in this way is usually referred to as empirical coverage $\hat{C}$ (Angelopoulos & Bates, 2021). Let $(x_{i,j}^{(\text{val})}, y_{i,j}^{(\text{val})})$ be the ith validation example in train j, and C_j be calibrated using the calibration data from the jth trail. Then the coverage C_j in the split $j \in R$ is:

$$C_j = \frac{1}{n_{\text{val}}} \sum_{i=1}^{n_{\text{val}}} \mathbf{1}\left\{ y_{i,j}^{(\text{val})} \in C_j(x_{i,j}^{(\text{val})}) \right\} \text{ for j=1,...,R} \tag{12}$$

If all C_j are plotted, the coverage should be beta-distributed by $1 - \alpha$ and the mean should be:

$$\hat{C} = \frac{1}{R} \sum_{j=1}^{R} C_j$$

As R must be as large as possible to achieve good results, this method requires considerable effort. However, this can be reduced by first calculating and temporarily storing all conformity scores for all data points before splitting.

4.2 Evaluating Adaptivity

Adaptivity can be simplified as follows: we want small prediction sets for simple data points and larger ones for difficult data points. However, as adaptivity is difficult to describe formally, various metrics have been proposed for the empirical testing of adaptivity (Angelopoulos & Bates, 2021).

1. **Set size**. A histogram of the different predicted set sizes provides two insights. Firstly, large prediction sets are uninformative, i.e. if all sets are very large, either the underlying model is too bad or the conformity score used is insufficient. Secondly, adaptivity means that the sets (depending on the difficulty of the data points) should have different sizes. This means that we expect a broad rather than a narrow distribution. Such a narrow distribution therefore usually indicates only low adaptivity.

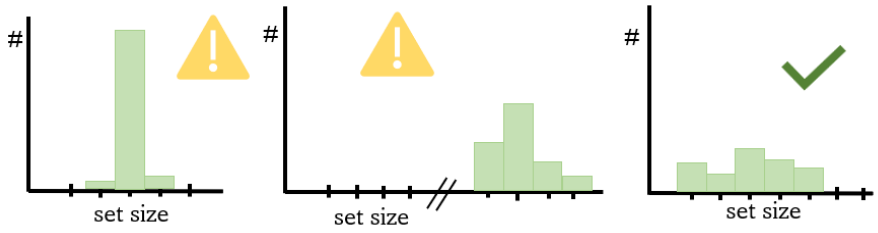


Figure 11: Different options for the distribution of the prediction set size. **Left**: Peak distribution usually indicates a less adaptive method. **Centre**: Very large prediction sets are uninformative. **Right**: Broad distribution with small sets for lighter and larger sets for heavier data points.

2. **Feature-stratified coverage metric (FSC)** The idea behind FSC is to determine the coverage in the worst covered group g. This approach is closely related to the group-balanced CP method (Chapter 2.4.1). For this, we assign all data points of a distinct group $g \in G$ and determine the average coverage within this group. Formal let $\mathcal{I}_g \subset [1, n_{\text{val}}]$ be the index set for which each data point $(x_i, y_i), i \in \mathcal{I}_g$ belongs to group g, then the FSC is calculated as:

$$\textbf{FSC} = \min_{g \in G} \frac{1}{|\mathcal{I}_g|} \sum_{i \in \mathcal{I}_g} \mathbf{1}\left\{ y_i^{(\text{val})} \in C(x_i^{(\text{val})}) \right\}$$

3. **Size-stratified coverage metric (SSC)**. A more general metric that does not require a apriori group membership can be obtained by analysing the empirical coverage for data points with equally large prediction sets. This means that the validation data points are divided into G bins, where $\mathcal{I}_g$ describes the index set for which $|C(x_i)| = g \ , \ i \in \mathcal{I}_g$ holds. Therefore the SSC is calculated as:

$$\textbf{SSC} = \min_{g \in G} \frac{1}{|\mathcal{I}_g|} \sum_{i \in \mathcal{I}_g} \mathbf{1}\left\{ y_i^{(\text{val})} \in C(x_i^{(\text{val})}) \right\}$$

   The idea behind the method is that easier data points produce smaller prediction sets, but the coverage in these is still $1 - \alpha$ and thus these sets are not compensated by fewer errors in other (larger) sets.

5. Related Work

Up to this point, Conformal Predictions has been introduced as a distribution-free, model agnostic, non-asymptotic, finite sample size uncertainty estimation framework. However, Conformal Prediction can also be understood in the context of a calibration method for probabilistic prediction (Manokhin, 2022a). Prediction is concerned with making forecasts about the future, while probabilistic prediction is understood as the attempt to quantify the uncertainty of such predictions (Gneiting & Katzfuss, 2014). As already demonstrated, there are various models, such as Bayesian models or softmax outputs of neural networks,

which already have an intrinsic notation for their uncertainty. However, their notation of uncertainty is error-prone and therefore only heuristic in nature. These probabilistic predictions are therefore to be regarded as non-calibrated. As we will see, methods of the Conformal Prediction framework can also be used to calibrate probabilistic predictions. From Vovk's work on different notations of randomness and the role of exchangeability for the prediction of new data points, not only the framework known today as Conformal Prediction has emerged, but also the Venn predictors closely related to CP, which also have certain guarantees under minimal conditions (Manokhin, 2022a). In a nutshell, Venn predictors calibrate the outputs of classifiers. In the this chapter a brief introduction to Venn predictors and calibration in general is given.

The Full CP method has already been formally introduced in Chapter 2.2.2. However, the original formulation according to Vovk and Gammerman is slightly different and derives p-values from the nonconformity scores. As this notation is necessary for further understanding in the following, it will also be introduced here:

Let $s_i^y = S\Big((X_i, Y_i), f^y\Big)$ be the conformity scores based on the augmented dataset $(X_1, Y_1), ..., (X_N, Y_N), (X_{n+1}, y)$ with $y \in \mathcal{Y}$, then the p-value corresponds to:

$$p_y = \frac{|\{i = 1, ..., n+1 \ : \ s_i \geq s_{n+1}\}|}{n+1} \tag{13}$$

In other words, the p-value is the proportion of the s_i which are at least as large as s_{n+1} , i.e. it is the proportion of examples that appear "stranger than" or "as strange as" the test point. With this notion of p-values the prediction set are defined like:

$$C(X_{n+1}) = \{y \in \mathcal{Y} : p_y > \alpha\}$$

At this point, the following general remarks should be made about the p-values. p-values are not posterior probabilities. This means that they do not express the probability of data point X_{n+1} to have the label y. Rather, the p-value means: "What is the probability of drawing an example that is as or more contrary than the test example to the hypothesis that it comes from the same distribution as the training set?"

With this formulation, we can now turn to the difference between confidence and probability. Confidence corresponds to the p-value, but the question of posterior probability, i.e. what is the probability that X_{n+1} has the label y, is often more decisive in everyday life. A probabilistic prediction answers this question (Figure 12).

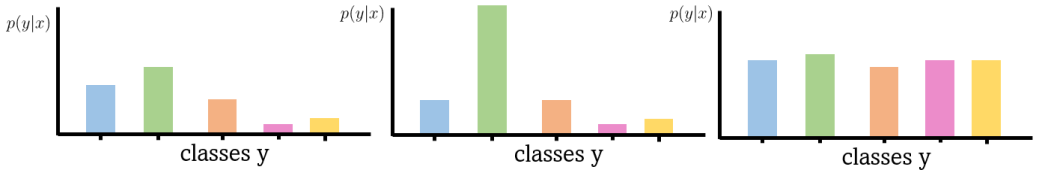


Figure 12: Illustration of a different predictive distributions. Consider the outputs of three different classifiers. All predict the green class as the one most likely, but considering the predictive distribution showcases that the certainty of the predictors is different.

5.1 Calibration

In simplified terms, a model can be described as well calibrated if its output reflects the true probability of the prediction. Formally, a perfect calibrate classifier $s = f(x)$ can be described as follows (Guo, Pleiss, Sun, & Weinberger, 2017):

Definition 6. ***Perfectly calibrated classifier***
A classifier is perfect calibrated if:

$$\mathbf{P}(\hat{Y} = Y|\hat{P} = p) = p \ , \ \forall p \in [0, 1]$$

where $\hat{Y}$ is the class prediction an $\hat{P}$ the associated class prediction.

Intuitively well calibrated means that we are looking for a model whose outputs correspond to the true probability of the classes. However, it must be noted that from a theoretical point of view generally no validity for calibration, which requires more statistical guaranties then only calibration, can be obtained (Manokhin, 2022a). Nevertheless, in the past different attempts, both parametric and non-parametric, have been made to calibrate the outputs of machine learning models. A brief overview of some of the most important methods in this area are discussed in the next section.

One of the most classic calibration methods is 'Platt's scaling', where a sigmoid function is mapped to the calibration set. Although this method and numerous extensions of the original approach provide better outputs than then non calibrated model outputs, the method does not work perfectly (Manokhin, 2022a). Histogram binning can be used for any model, but comes with the disadvantage that the number and size of bins must be defined a priori to achieve good results (Pakdaman Naeini, 2017). Isotonic and smooth isotonic regression, where a non decreasing function is mapped to the calibration set, could empirically show better calibration for logistic regression and SVM models than Platt's scaling, but both methods come with no guarantees and needs more data then the original approach (Manokhin, 2022a). Other methods for calibration that also come without rigorous statistical guarantees are, for example Nested Dichotomies (Leathart, Frank, Pfahringer, & Holmes, 2019), Beta calibration (Kull, Silva Filho, & Flach, 2017) and Scaling-binning (Kumar, Liang, & Ma, 2019). Guo et al. (Guo et al., 2017) showed that convolutional neural networks (CNN) are not well calibrated and therefore he proposed temperature scaling, which can be understood as a simple extension of Platt's scaling. This approach rescales the logit vector s_i of a neural network using a temperature factor T that increases the output entropy. The newly calibrated output is $\hat{q} = \max_k \sigma(s_i/T)^{(k)}$, where k is the class index. For $T \to \infty$ the probability $\hat{q}_i \to 1/K$ resulting in maximum uncertainty, for $T = 1$ in the original scores and for $T \to 0$ the probability converges to a point prediction. The method was later popularized by Hinton (Hinton, Vinyals, & Dean, 2015) and, despite its simplicity and popularity, has the crucial disadvantage that both correct and overconfident incorrect predictions are damped. Furthermore, the method even worsens the calibration under covariate shift (Manokhin, 2022a). Therefore, Mukhoti et al. (Mukhoti, Kulharia, Sanyal, Golodetz, Torr, & Dokania, 2020) proposed to replace the cross-entropy with the so-called focal loss, a method originally developed to address class imbalance problems. Here, overconfident predictions are also attenuated and the entropy of the prediction is down-weighted. The approach of Pereyra et al. (Pereyra, Tucker, Chorowski, Kaiser, & Hinton, 2017) takes a slightly different path. A regularization term penalizes outputs with low entropy, since it can be shown that overconfident and thus poorly calibrated outputs generally tend to have a low entropy. Label smoothing (Szegedy, Vanhoucke, Ioffe, Shlens, & Wojna, 2016) is another widely used approach in the field of image classification and natural language processing to address the issue of overconfident neural networks. Here, a weighted combination of the target labels with a uniform distribution is used to calculate the cross-entropy loss. It has recently been shown that label smoothing not only improves the performance of a net, but also leads to better calibrated results. Although some of these methods achieve good results in some use cases, none have any statistical guarantees. Venn predictors (VP) are, to the best of my knowledge, the only calibration method to obtain well calibrated class probabilities in a strong non-asymptotic case. VPs is based on the idea of Conformal Prediction and assumes also only exchangeability (Manokhin, 2022a). In the following, Venn predictors and various modifications of them will be briefly presented.

5.2 Venn Predictors

Venn Predictors are a form of multi-probabilistic predictors for which we can prove well calibration. Even if valid calibration cannot be achieved, VP guarantees well calibration by specifying multiple probabilities for each label, one of which is the true probability (Vovk, Shafer, & Nouretdinov, 2003). It should be noted at this point that Venn predictors were originally formulated transductive and requires high computational effort. Inductive Venn predictors were later developed, in line with Split Conformal Prediction, witch significantly reduced the required computation time. In the following, the general principle of transductive Venn predictors is described in detail.
The key idea of a Venn predictor is to divide all training examples into a number of k distinct categories and use the relative frequency of the label $Y \in \mathcal{Y}$ in each category to estimate the label probabilities for test points falling into that specific category. The categories are defined using a Venn taxonomy. Since

different taxonomies result in different category memberships, each Venn taxonomy also defines a specific Venn predictor. The category of a test data point X_{n+1} is now determined using the same taxonomy and the frequency of the labels in the training data is used to estimate the distribution over the labels. Since the true label y_{n+1} for the test data point X_{n+1} is not known, a distribution over the labels must be calculated for each potential label $y \in \mathcal{Y}$. Thus, comparable to the prediction sets in CP, we obtain a set of label probability distributions. The pseudo-code for a transductive Venn predictor is formulated in Figure 4.

Algorithm 4 Transductive Venn Predictor

1: **Input:** data $\{X_i, Y_i\}_{i=1}^{n}$, X_{test} , Classifier f, Venn Taxonomy T
2: **for all** $y \in \mathcal{Y}$:
3: Add (X_{n+1}, y) to the training data $\{X_i, Y_i\}_{i=1}^{n}$
4: Find the category k to which (X_{n+1}, y) belongs using the Venn Taxonomy T and the classifier f
5: Let Z_k the set of training instances belonging to category k
6: The probability distribution p_{y_i} of the labels in category k is:

$$p_{y_j}(y) = \frac{|\{(X_i, Y_i (\in Z_k \mid y_i = y\} + 1}{|Z_k| + 1}$$

i.e., for each possible label y we find the relative frequency of test points with the label in category k

It should be noted that the underlying model is not used for the actual prediction but only to determine the category of the training data points. To draw the parallel to conformal classification again, the classifier here corresponds to the underlying model as in CP and the taxonomy is roughly comparable to the score function.

As already mentioned Venn predictor returns a distribution over $|\mathcal{Y}|$ for each possible label $y \in \mathcal{Y}$. In other words, for each potential label, a Venn predictor returns a probability for all labels, of which one is the true label distribution. This means that the output of a Venn predictor corresponds to a $|\mathcal{Y}| \times |\mathcal{Y}|$ matrix (Johansson, Löfström, & Sundell, 2018). Transductive Venn predictors can in principle be built on any underlying classifier that is used to determine the category for each training point. For example, the simplest Venn taxonomy that can be used, and which is compatible with any classifier, is simply to assign each test point the same category as the predicted label. Another possibility is to use transductive predictors with a classical lazy learner like kNN. The obvious Venn taxonomy that can be used in this case is to assign two test data points to the same category if their nearest neighbours have the same label (Johansson et al., 2018). The problem with transductive Venn predictors is that for each new data point X_{n+1} a classifier must be trained anew for each $y \in \mathcal{Y}$. To overcome the limitations of the transductive setting, the data can first be split into a proper training set and a calibration set, just as for Split CP. The underlying model is now trained on the training set, while the calibration set is used to determine the relative label frequencies in the categories (Nouretdinov, Volkhonskiy, Lim, Toccaceli, & Gammerman, 2018). A test data point X_{n+1} is now assigned to a category identically like the calibration points. The relative frequencies of the labels in this category are now used for the probability estimation, but all possible labels for the test data point are also used, resulting in a set of probability intervals (Nguyen, 2021).

So far, the described Venn predictors have returned a set of label distributions. In order to actually arrive at a prediction, the following approach is usually chosen:

$$\hat{y}_{n+1} = \max_{y \in \mathcal{Y}} p_{y_j}(y)$$

In addition, a probability interval $[L(\hat{y}_{n+1}), U(\hat{y}_{n+1}]$ can be specified, where L (lower bound) and U (upper bound) are defined as follows:

$$U(y) = p_{y_j}(y)$$

$$L(y) = \frac{|\{(X_i, Y_i (\in Z_k \mid y_i = y\}}{|Z_k| + 1}$$

The multi-probability predictions provided by a Venn predictor are guaranteed to be well calibrated regardless of the taxonomy used. However, the taxonomy influences the accuracy and the size of the prediction intervals significantly. As in CP methods, the estimated probability's should be as close to 0 or 1 as possible and the interval should be as small as possible for greater significance. Another significant influence is the number of categories used, the more categories are used the more specific the prediction becomes, but too many categories can lead to only a few sample data being in this category and thus the prediction interval becoming larger. Since the choice of a suitable taxonomy can be difficult, the so-called Venn-Abers predictors were introduced. These are Venn predictors and thus inherit their calibration guarantee, but are only applicable to two-class problems. The trick is that they use isotonic regression to automatically optimise the taxonomy. Isotonic regression is a special case of binning and separates the test instances into different bins, which we can understand as categories. The inductive Venn-Abers predictors regard the underlying model as a scoring classifier, i.e. the output of the model is regarded as a prediction score, with higher values indicating a higher confidence that the test instance has the label 1. The method now also uses isotonic regression for calibration, i.e. an isotonic regressor is trained twice on both the calibration set and the test instance. Once with the tentative label 0 and once with the tentative label 1 (Vovk & Petej, 2012). Nevertheless, the calculation can be done very efficiently (Vovk, Petej, & Fedorova, 2015). Venn-Abers predictors have been applied to so-called Probability Estimation Trees (PET). These are decision trees that provide a probabilistic distribution, but are known to be too optimistic in their probability predictions. By combining them with Venn-Abers predictors, we obtain an intrinsically interpretable model that additionally provides a well-calibrated probabilistic interval for each leaf. This approach can thus provide a truly trustworthy model (Johansson, Löfström, & Boström, 2019).

6. History and Recent Development

Conformal Prediction has seen a real explosion of interest particular in the last year, as evidenced by both numerous real-world applications and a veritable flood of publications. Nevertheless, the roots of the framework go back more than 50 years and are based on the work of Andrei Kolmogorov in Moscow. The real fathers of Conformal Prediction are Vladimir Vovk and Alexander Gammermann, who worked out the theoretical foundations together in London in the late 1990s. Especially V. Vovk has made numerous further fundamental applications and theoretical contributions in the following years until today. In the middle of the 2010s, the framework, which until then had lived in the shadows of academia, was given a new impetus by the work of the Professor of Statistics and Data Science at the Carnegie Mellon University Larry Wassermann in America and the group around Ulf Johansson and Henrik Boström in Sweden. Inspired by the work of Larry Wassermann, groups led by Emanuel Candes (Chair Mathematics and Statistics, Stanford), Ryan Tibshirani (Chair Statistics, UC Berkeley) and Michael Jordan (Chair Computer Science, Statistics, UC Berkeley) reformulated and popularized the framework and paved the way for a variety of new approaches and methods in recent years. Of course, this rough outline is highly simplistic and leaves out many researchers who also had significant influence on the development of Conformal Prediction, but should give the reader a sense of the most prominent dynamics in the field. In the following overviews, this rough description will be enriched by a slightly detailed timeline, which shows the most important developments and publications in the field, without claiming to be complete.

- **1960-1980** Andrei Kolmogorov starts at Moscow State University with work on notation of randomness, complexity and probability (Kolmogorov, 1968). Among other things, he studies algorithmically random sequences and finite Bernulli sequences (Kolmogorov, 1983). Vladimir Vovk becomes his student during this period.
- **1988** V. Vovk presents his PhD thesis "Predictability of algorithmically random sequences" under the supervision of A. Kolmogorov. Here he develops the Basis for a first understanding of the role of finite-sample exchangeability in prediction problems for the study of Bernoulli sequences.
- **1996-1999** Vladimir Vovk, Alexander Gammerman and Vladimir Vapnik develop the framework now known as Conformal Prediction together at the Royal Holloway University of London. They first used e-values (Gammerman, Vovk, & Vapnik, 1998b), later p-values (Vovk, Gammerman, & Saunders, 1999).
- **2002** Harris Papadopoulos, together with Vovk, develops what is now known as Split Conformal Prediction (Papadopoulos et al., 2002).

- **2003** Vladimir Vovk and Glen Shafer publish "Algorithmic Learning in a Random World" and coin the term "Conformal Prediction".
- **2003** Vovk and Gammerman lay the foundations for group-balanced CP with the Mondrian Conformal Predictors (Vovk et al., 2003).
- **2003** First Symposium on Conformal Prediction and its Applications (COPA) is organized in Greece by Harris Papadopoulos. This meeting has been held annually and is the most important event for the CP community.
- **2014** The group around Ulf Johansson, Henrik Boström and Henrik Linusson in Sweden take up the work of Vovk and develop CP methods especially for random forests (Johansson, Boström, Löfström, & Linusson, 2014). The group will also publish a number of papers and tutorials in the coming years.
- **2014** Larry Wassermann and Jing Lei begin their work in the field (Lei & Wasserman, 2014), making Vovk's work better known, especially in the United States. Among other things, they have done seminal work on class-balanced CP (Lei, 2014) and formulated a general framework for distribution-free predictive inference in regression (Lei, G'Sell, Rinaldo, Tibshirani, & Wasserman, 2018).
- **2019** The research group led by Emmanuel Candes at Stanford published the first of many papers on CP including. Conformalized Quantile Regression (Romano et al., 2019)
- **2020** Adaptive prediction sets (Romano et al., 2020)
- **2020** In the 2020 U.S. presidential election, The Washington Post estimated the yet-to-be-published election results for individual states using Conformal Prediction (Cherian & Bronner, 2021)
- **2021** Anastasios N. Angelopoulos and Stephen Bates, two PhD students of Michael Jordan show the use of CP on large-scale deep learning classifications tasks (Angelopoulos et al., 2020).
- **2021** General conformal risk control (Angelopoulos, Bates, Candès, Jordan, & Lei, 2021)
- **2021** Conformal Outlier detection (Bates, Candès, Lei, Romano, & Sesia, 2023)
- **2021** Detection of change points in time-series data (Vovk, 2021)
- **2021 and 2022** Dedicated tracks at ICML2021 and ICML2022. Keynote 'Conformal Prediction' at NeurIPS2022 by Emmanuel Candes
- **2022** Valery Manokhin, a PhD student of V. Vovk, created the Git repository "Awesome Conformal Prediction" [1]. which is today the most important overview platform for the rapidly developing field. Here you can find the most important papers, theses, books and many tutorials (Manokhin, 2022a).
- **2022** Release of the scikit-learn compatible package MAPIE with some basic conformal methods.[2]
- **2022** A. Angelopoulos and S. Bates publish "A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification" (Angelopoulos & Bates, 2021), a highly readable introduction to Conformal Prediction including a good video tutorial.[3]
- **2022** CP beyond exchangeability (Barber, Candes, Ramdas, & Tibshirani, 2023)

7. Evaluation of different CP Methods under Small Datasets

Conformal Prediction comes in several flavors, with different properties in terms of required computation time and data efficiency, where higher efficiency is associated with increased computational effort. For large datasets, it is often possible to save a hold-out dataset, with about 1000 data points, for calibration purposes and therefore to use a Split CP method. These have the advantage that the model only has to be trained once or an already pre-trained model can be used. In contrast, Full Conformal Prediction methods require a model to be trained for each possible label in the sense of a lazy learner for each new test data point. This makes these methods impossible to use for many computationally intensive methods. With Jackknife+ and CV+ methods are available, which stand between Full and Split CP and thus represent an interesting alternative for more computationally intensive models and use cases with only few available data points.

1. https://github.com/valeman/awesome-conformal-prediction
2. https://mapie.readthedocs.io/en/stable/
3. https://www.youtube.com/watch?v=nql000Lu_iE&t=1529s&ab_channel=AnastasiosNikolasAngelopoulos

In a first evaluation the extent to which different splits of the data sets differ in terms of the conformal scores and the corresponding quantiles were investigated. In a second experiment the performance of the Split, CV+ and Jackknife+ methods was evaluated on two different multi-class datasets both down sampled artificial into sets of different sizes. In a third experiment, the smallest possible datasets were analysed. For all those experiments both an artificially created and a real life dataset were used.

7.1 Methods

For the evaluation, three traditional classification models were used, namely a Gaussian Naive Bayes (GaussianNB), a random forest model with 500 trees, and a support vector machine with radial basis function kernel (SVM). Both models have the advantage that they are relatively robust to the required hyperparameters and thus their impact on training with different sized data sets is minimal. Another advantage is that the models require limited computational time, making the experiment feasible for the Jackknife+ method. For both models the implementation provided in the **scikit-learn** package[4] was used. For each, a Split CP method and CV+ with $k = 5$ splits were evaluated. For the latter two variants were employed to aggregate the predictions obtained by cross-validation. The first simply calculates the average of the scores of the different models for a new test point (mean) and the second method (crossval) compares the individual conformity scores of the training points of the different models with the new test point as described in algorithm 2. For the Split CP method, 20% of the available data was used as a calibration set for each run. In addition, a Jackknife+ procedure with the cross validation aggregation function was evaluated for each model. All of these experiments were used with the least ambiguous set-valued classifier method (lac) as well as adaptive conformity scores (aps).
The artificial test data set was initialized with the method **make_classification** of the scikit-learn package[5] with 20 features (15 informative, 2 redundant, 3 noise) for 5 different classes in the second and 2 classes in the third experiment. In total, the dataset consists of 10,000 data points. The other dataset used contains the data of 13,611 dry beans, which have to be classified into one of 7 variants based on 8 features such as length, roundness or firmness (Koklu & Ozkan, 2020).
The following experiments were carried out as part of this work.

1. In order to investigate the extent to which different splits of the datasets affect the conformal scores, the artificial dataset was first divided into 5 equally sized splits and the conformal scores and the calculated quantile were calculated separately for each of the splits. Both the least ambiguous set-valued classifier and the adaptive prediction set method were used as scoring methods.
2. To clarify the question of how the coverage of a CP method compares to the accuracy of a non-CP method, a random forest model (100 trees) was analysed on the artificially down sampled artificial data set (16-1000 data points, step size: 1) once with a Split CP method with the lac scoring function and once without the CP method. The experiment was repeated 5 times and the results averaged.
3. For the second experiment, 5000 points were used for both data sets to evaluate the different methods, i.e. they are not used for training or calibration. The remaining data points were dedicated to training, using a randomly distributed subset of these points for each experimental setup. Starting with 100 training points, the set was expanded by 100 points in each run. Thus, there were a total of 8 different strategies for 2 different models, with each resulting combination trained on datasets ranging in size from 100 to 5000 in steps of 100, and thus 800 different compliant runs. The experiment was repeated 5 times for each data set and the results averaged.
4. In the third series of tests, the smallest possible data sets were analysed. It should first be noted that a calibration set must contain at least $\max(\frac{1}{1-\alpha}, \frac{1}{\alpha})$ many data points. The same applies to the CV+ and Jackknife methods. The reason for this is that sufficient conformity scores must be available to determine the $1 - \alpha$ quantile. In addition, at least $|\mathcal{Y}|$ data points must be available for the training. To create an exact example, the artificial data set with 5 different classes and $\alpha = 0.1$ was selected for this experiment. This meant that experiments from $n \geq \frac{1}{1-\alpha} + 1 + |\mathcal{Y}| = 16$ data points could be analysed. From this step on wards, the size of the data set used was increased iteratively by one data

4. `https://scikit-learn.org/stable`
5. `https://scikit-learn.org/stable/modules/generated/sklearn.datasets.make_classification.html`

point up to a maximum of 200 data points. The least ambiguous set-valued classifier (lac) was used for the Split, CV+ and Jackknife+ method.

7.2 PERMAD

In this work, an attempt was made to apply conformal change-point detection to the PERMAD data set. This dataset contains data from 41 oncological patients with a total of 647 measurement points. For each individual patient a different number of measurements (min: 4, max: 46), each with 91 different laboratory values as features, are available. The time points of the measures are distributed irregularly and are different for each patient. At the start of data collection, all patients are classified as non-progressive, i.e. the tumour is not growing any further. CT scans were taken of the patients at irregular intervals and each patient was followed until tumour progression occurred. The aim of the analyses is to be able to predict the change-point, i.e. from non-progression to progression, as early as possible on the basis of the laboratory values and thus to identify certain bio-markers for early detection. The idea here is that every patient experiences a distribution drift from non-progressive to progressive and that there is therefore a distribution change point. Such change points can be recognised, at least in a certain sense, with the help of conformal test martingales. Conformal change point detection was designed to detect a drift of the underlying distribution for an already trained model in online mode with statistical guarantees. In practice, this means that the model is first trained on data from the target distribution. Then the model is used to predict new test data points, also from the training distribution. The corresponding conformity scores are calculated for all these new data points and the corresponding p-values are calculated from them. Using these p-values, the conformal test martingale (CTM) can now be determined for each new data point. If the data points originate from the training distribution, the CTM values remain constant or decrease slightly. If there is now a distribution drift, i.e. the new test points do not originate from the training distribution, the CMT values increase. If the CTMs reach a certain threshold, the drift can then be recognised with a certain degree of certainty (Vovk et al., 2021).

In order to apply this method to the PERMAD data set, a model would have to exist that makes a prediction based on the assumed "non-progression" distribution. Subsequently, all further points in time can also be predicted using this model and the non-conformity scores are calculated on this basis. Vovk has proposed the difference between the real and the predicted label as the non-conformity score, but there are no labels for the PERMAD dataset available, that define the change-points. Only the first data point for each patient could be understood as an approximation of the healthy state. The same applies to the last data point, as this was identified as progression by the CT result.

As part of this work, two autoencoders were trained, one based on the first measurement point ("AE_no_prog") and one based on the last measurement point ("AE_prog") of each patient. Therefore in total, 41 measurement points were available for the first and 41 measurement points for the last data point. Of these, 31 data points were used to train the autoencoder and 10 data points for validation. Each of these models was then applied to the remaining data points of each patient and the corresponding CTMs were calculated. In a first attempt the conformity scores was calculated as the Euclidean distance of the embedding of the current time point with that of the first data point. In a second trail the cross entropy between the autoencoder outputs and the input for each test time point was used. The scores of the two autoencoders were used for both cases both individually and in combination.

Unfortunately, no change-point could be detected with any method. On the other hand, the authors of the CTM method have already described that this method is not particularly efficient, meaning that a possible change-point can usually only be reliably recognised after approx. 20 data points after the distribution shift appears. However, this approximation applies only to the best case, meaning, that the distributions change abruptly and are clearly different. For the PERMAD dataset, however, it is questionable whether there is a uniform non-progression distribution for all patients, and even if there is, whether the change is abrupt or rather a slow change in many covariates. Even worse, there are of course considerable differences between the patients, which can be greater than a non-progression/ progression difference within a patient. Moreover, the available features reflect more than just the oncological status, i.e. within the study period, patients may also vary between other possible conditions, such as general health or other illnesses. It is therefore questionable whether the conformal test point method can be a suitable method for the problem

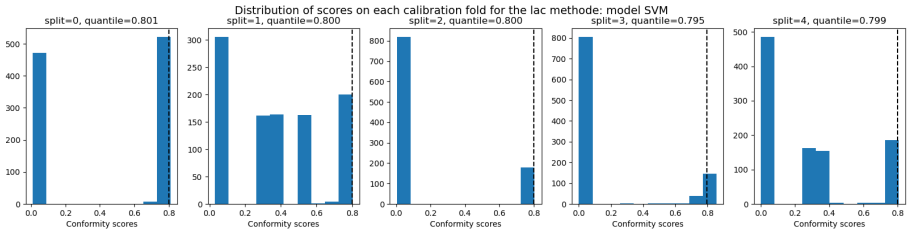
described here in any possible way. The results obtained in the course of this work at least do not point in this direction.

7.3 Results

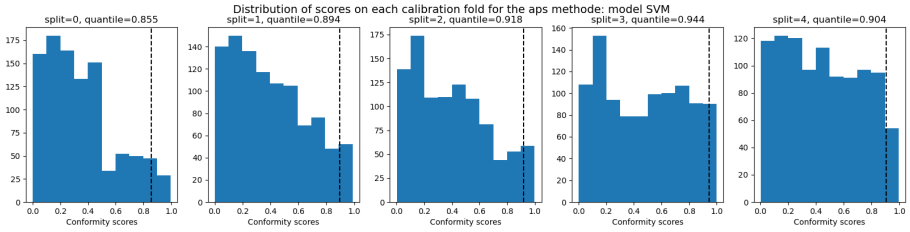
In a first evaluation the extent to which different splits of the data sets differ in terms of the conformal scores and the corresponding quantile was investigated.

7.3.1 Influence of different Splits on the Conformal Scores

Figure 13 shows the results for the distribution of conformal scores and the resulting .95 quantile using the least ambiguous set-valued classifier method (lac) and the adaptive prediction set method (aps) for 5 different splits. The results for the Gaussian Naive Bayes model can be found in the appendix 9.1.



(a) least ambiguous set-valued classifier score



(b) adaptive prediction set score

Figure 13: Distribution over the conformal scores and the resulting .95 quantile on 5 different splits using the least ambiguous set-valued classifier method (lac) and the adaptive prediction set score (aps) based on a support vector machine with radial basis function kernel.

7.3.2 Comparison Coverage and Accuracy

Figure 14 shows the result of the direct comparison between empirical coverage and accuracy.

7.3.3 Influence of small datasets on different conformal methods

In the figures 15, 16 the results for the evaluation of different CP methods under datasets with increasing size for the least ambiguous set-valued classifier method (lac) are presented for the artificial dataset. Since in this set all classes are somewhat equally difficult to determine by design, an adaptive prediction score method shows no real advantage and the resulting sets are only slightly larger and can be found in the appendix 9.2.

The results for the dry beans dataset for the SVM for the least ambiguous set-valued classifier method (lac) are shown in figure 17 and for the adaptive prediction set method (aps) in figure 18. The results for the GaussianNB model can be found in the appendix 9.2.

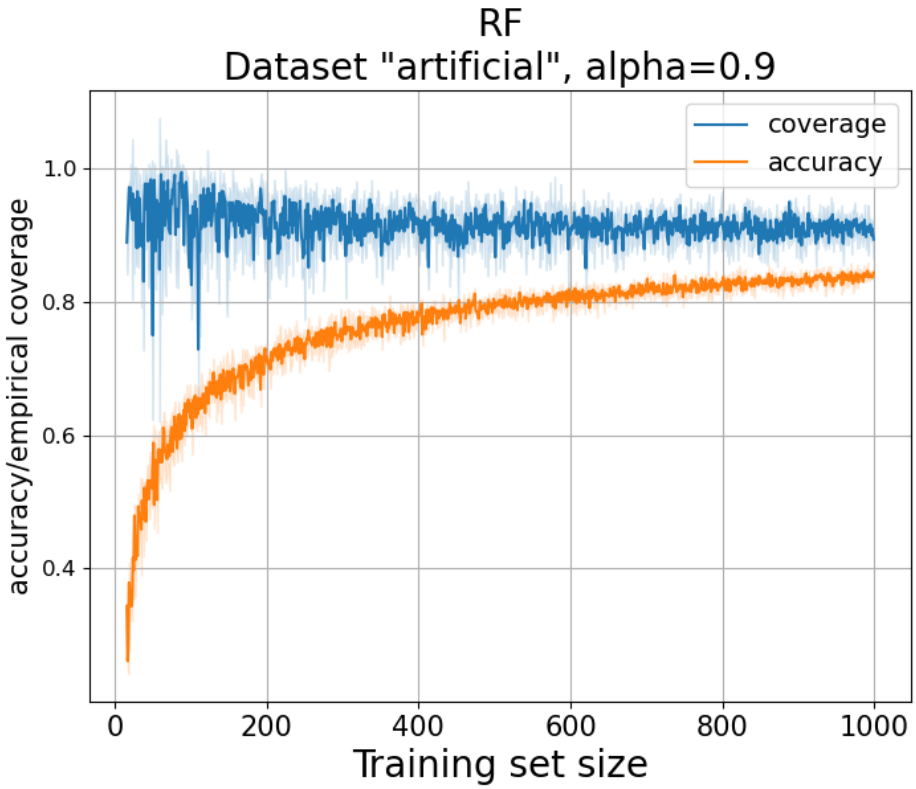


Figure 14: Direct comparison of coverage and accuracy on an artificial dataset with down sampled size. The underlying model is a random forest with 100 individual trees. A least ambiguous set-valued classifier method (lac) is used as the Split Conformal Method. The coloured areas indicate the standard deviation.

7.3.4 Conformal Prediction for the smallest possible datasets

In order to investigate the effect of extremely small data sets (16-200 data points) on the performance of compliant methods in more detail, the sets were analysed with the different methods Split, CV+ and Jackknife+ and the lac scoring function on the artificial data set. The results for the Random Forest model (RF) can be found in Figure 19 and those of the SVM and GaussianNB in Appendix 9.3.

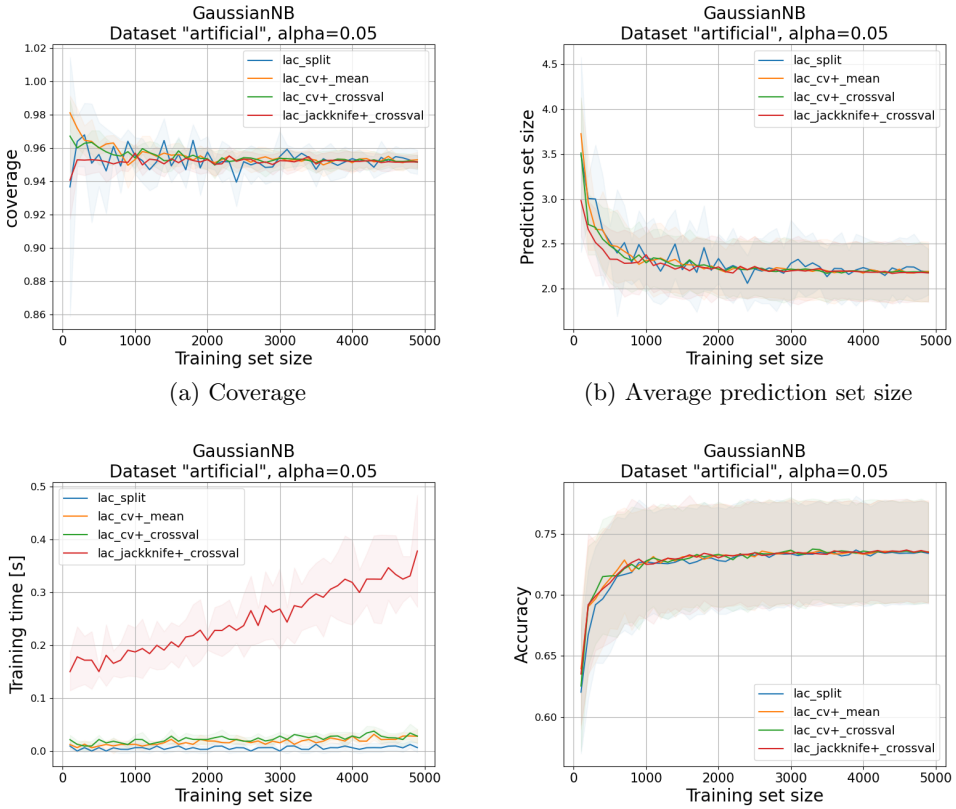
8. Discussion

8.1 Influence of small Prediction Sets on different CP Methods

In a row of experiments, the extent to which data sets with only a few data points influence different conformal methods, namely Split, CV+ and Jackknife+, was investigated.

8.1.1 Influence of different split

It can be shown that even for the homoscedastic artificial data, different splits show a different distribution of scores across the calibration set. This means that a slightly different quantile is calculated for each split. This subsequently leads to fluctuations in the effective coverage. This observation suggests that an unfortunate choice of calibration set can affect the robustness of any Split CP method. Another observation is that the adaptive conformity score function gives a more even distribution of scores and thus separates



(a) Coverage

(b) Average prediction set size

(c) Training time

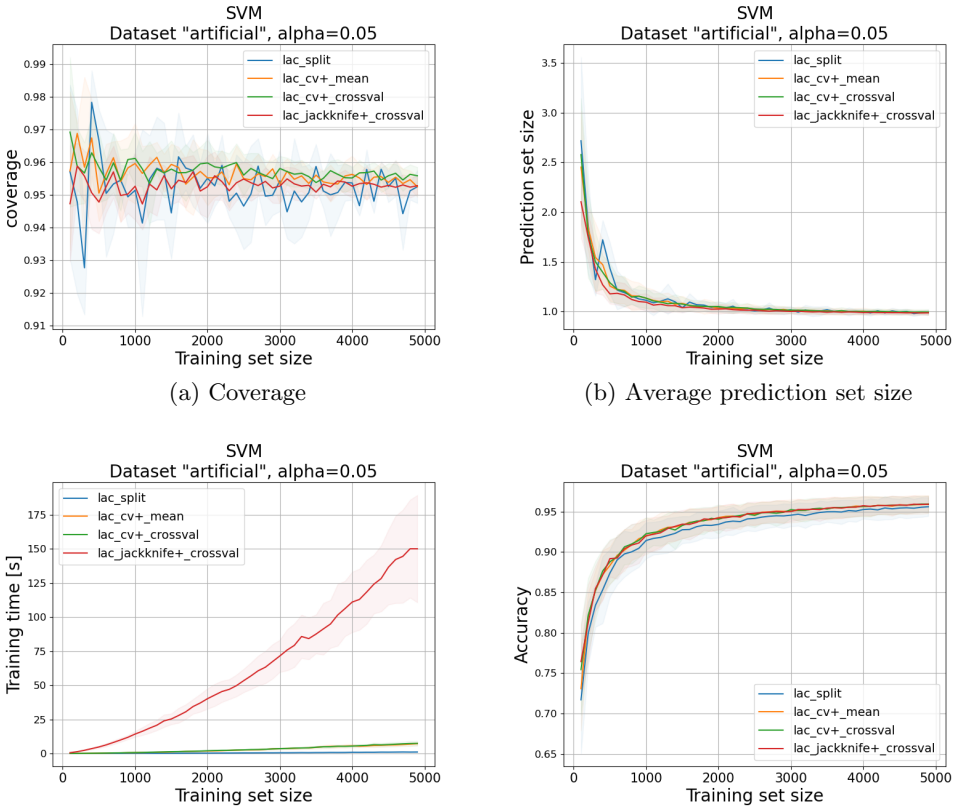
(d) Accuracy

Figure 15: Effective coverage, average prediction set size, training time and accuracy for the different conformal methods Split, CV+ with $k = 5$ and Jackknife+ with increasing training set sizes of an **artificial created dataset** with (20 features and 5 classes), based on a Bayes Naive Gaussian model. CV+ scores are aggregated using an averaging (mean) method and the cross validation approach according to (Romano et al., 2020). Conformal scores are computed using the least ambiguous set-valued classifier method (lac) for $\alpha = 0.05$. The colored areas indicate the standard deviation.

difficult and easier data points more evenly. This effect means that the resulting prediction sets are more adaptive and adjust more to the underlying uncertainty of the individual data points.

8.1.2 Performance of the Base Model

It is intuitively clear that for the Split CP methods, the underlying model can only be trained with less data and therefore, depending on the model and the complexity of the problem, the performance of the model is lower. As expected, the performance, measured as the accuracy of the base model, falls slightly for extremely small data sets for the Split method compared to the other methods, regardless of the underlying model. The Jackknife+ and CV+ methods use all data points and are therefore more performant for small datasets, whereby Jackknife+ has no accuracy advantage over CV+, at least for the simple problems examined here.



(a) Coverage

(b) Average prediction set size

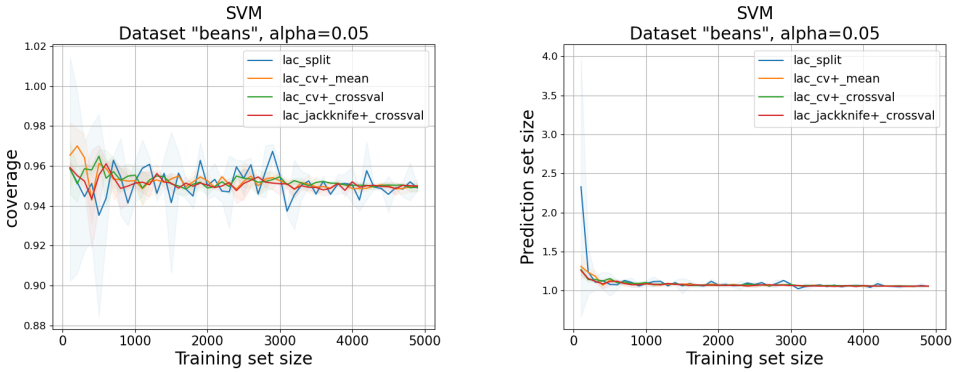
(c) Training time

(d) Accuracy

Figure 16: Effective coverage, average prediction set size, training time and accuracy for the different conformal methods Split, CV+ with $k = 5$ and Jackknife+ with increasing training set sizes of an **artificial created dataset** with (20 features and 5 classes), based on a support vector machine with radial basis function kernel. CV+ scores are aggregated using an averaging (mean) method and the cross validation approach according to (Romano et al., 2020). Conformal scores are computed using the least ambiguous set-valued classifier method (lac) for $\alpha = 0.05$. The colored areas indicate the standard deviation.

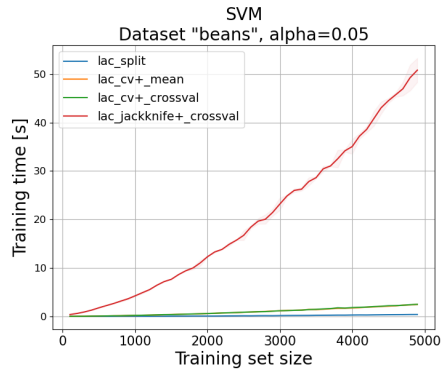
8.1.3 Coverage of the Conformal Methods

For the Split CP methods, theoretical assumptions can be made about the required size of the hold-out set, as described in Chapter 2.4.3. This is due to the fact that the coverage of the randomness in the calibration set follows a beta distribution. This means that 1000 data points should be sufficient for most applications. In this work, 20% of the available data were used as the calibration set for all Split CP experiments. This means that for the largest investigated data set with 5000 data points, exactly 1000 points were used for calibration. As described, all conformal methods fulfill the coverage guarantee, however, smaller calibration sets result in larger fluctuations in the effective coverage. As for the Split methods, this effect is clearly observable and amounts to differences of ±3% for small data sets. CV+ is significantly less affected by this as all data points are taken into account for the calibration, whereby the cross-validation aggregation function is significantly more robust than the mean function. As expected, Jackknife+ shows the most stable results here, as every single available data point can actually be used for calibration.

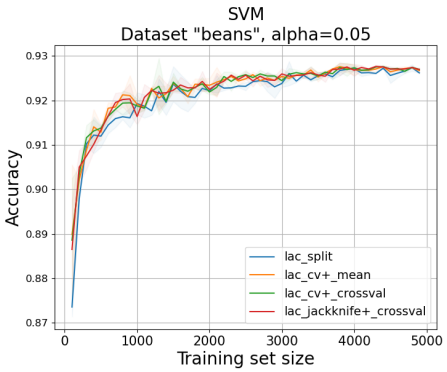


(a) Coverage

(b) Average prediction set size



(c) Training time



(d) Accuracy

Figure 17: Effective coverage, average prediction set size, training time and accuracy for the different conformal methods Split, CV+ with $k = 5$ and Jackknife+ with increasing training set sizes of the **dry beans dataset**, based on a support vector machine with radial basis function kernel. CV+ scores are aggregated using an averaging (**mean**) method and the cross validation approach according to (Romano et al., 2020). Conformal scores are computed using the least ambiguous set-valued classifier method (lac) for $\alpha = 0.05$. The colored areas indicate the standard deviation.

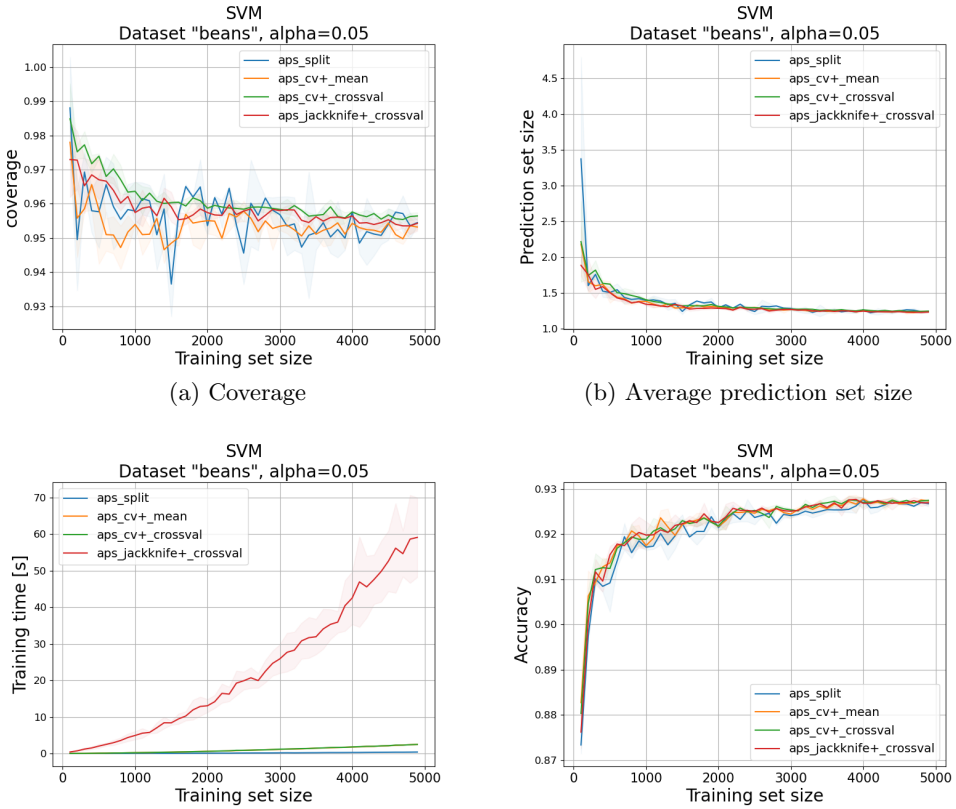
8.1.4 Prediction Set Size

As with the coverage, the average size of the prediction sets fluctuates strongly under the Split CP method, while it remains most stable for the Jackknife+ method. The better, i.e. the more reliable the underlying model is, the smaller the prediction sets obtained, as the notation provided for uncertainty better reflects the true uncertainty in the data.

According to the theory, the least ambiguous set-valued classifier methods provide the smallest prediction sets on average, while they are significantly larger for the adaptive methods. The extent to which the prediction sets vary for the different classes was not investigated in this work and is planned for future work.

8.1.5 Computation time

The Jackknife+ method, which calculates a model for each data point, is therefore extremely computation intensive compared to the other methods and the required training time increases rapidly with the number of data points. The computational effort of the CV+ methods mainly depends on the number of splits used



(a) Coverage

(b) Average prediction set size

(c) Training time

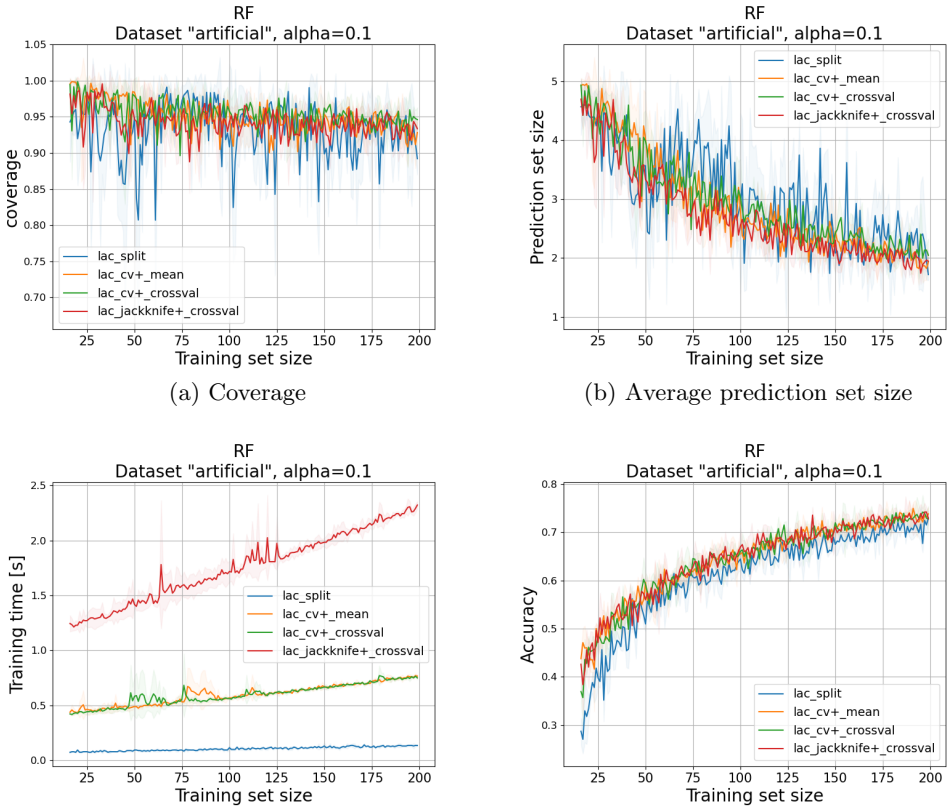
(d) Accuracy

Figure 18: Effective coverage, average prediction set size, training time and accuracy for the different conformal methods Split, CV+ with $k = 5$ and Jackknife+ with increasing training set sizes of of the **dry beans dataset**, based on a support vector machine with radial basis function kernel. CV+ scores are aggregated using an averaging (mean) method and the cross validation approach following (Romano et al., 2020). Conformal scores are computed using the adaptive prediction set method (aps) for $\alpha = 0.05$. The colored areas indicate the standard deviation.

and is therefore adjustable. In the experiments presented here, $k = 5$ splits were always used, whereby the performance using a cross-validation aggregation function only drops negligibly compared to the jackknife+ method. As described, Jackknife+ can be regarded as a special case of CV+ with $k = n$. Thus, CV+ methods with a suitable number of splits represent a good trade-off between performance and computation time and should be preferred for all tasks with models that are reasonably complex to train.

8.1.6 Coverage guarantee

In theory, the CV+ and Jackknife+ methods only fulfill the coverage guarantee of $1 - 2\alpha$. It turns out that for the SVM ($> 90\%$), which generally performs well on the artificial data set, a coverage of $1 - \alpha$ can be maintained empirically. The significantly worse performing GaussianNB models ($\backsim 70\%$ accuracy) experiments on the other hand, are slightly below a .95 coverage for $\alpha = 0.95$. The effect is somewhat more pronounced for the adaptive prediction scores. It is therefore reasonable to assume that for well-performing base models, i.e. those with an accuracy comparable to α, the stricter coverage guarantee of $1 - \alpha$ is adhered



(a) Coverage

(b) Average prediction set size

(c) Training time

(d) Accuracy

Figure 19: Effective coverage, average prediction set size, training time and accuracy for the different conformal methods Split, CV+ with $k = 5$ and Jackknife+ with $16 - 200$ randomly sampled data points of the **artificial dataset**, based on a Random Forest model with 100 trees. CV+ scores are aggregated using an averaging (mean) method and the cross validation approach following (Romano et al., 2020). Conformal scores are computed using the least ambiguous set-valued classifier method (lac) for $\alpha = 0.1$. The colored areas indicate the standard deviation.

to. On the other hand, for base models that are known to perform poorly and under alternative prediction set scores, the choice of α should be better adapted to the theoretically guaranteed $1-2\alpha$ coverage guarantee.

8.2 Conclusion

As this paper shows, Conformal Prediction is an easy-to-use paradigm for creating statistically valid uncertainty predictions for arbitrary models. This is ensured by the fact that instead of a point prediction, the models provide prediction sets or, in the regression case intervals, that contain the true label with a freely adjustable certainty α. This property is known as the conformal coverage guarantee. With Split CP methods are available that can equip pre-trained models with the conformal coverage guarantee only with little additional computational effort. This requires an additional calibration set, the size of which can be estimated by a beta distribution. For most practically applications 100-1000 calibrations points are sufficient. If not enough data is available for a proper calibration set, Full, CV and Jackknife+ methods can be used, which represent a trade-off between computing speed and data efficiency.

The advantage of all conformal methods is that they guarantee certain properties, such as coverage, empirical risk or false alarm rate. These guarantees are model-agnostic, distribution free and also valid for finite sample sizes. The only basic assumption is the exchangeability of the data points. However, the usefulness is strongly dependent on the available data, the performance of the underlying model and the used conformity measure function. If these are insufficient, the method can easily become completely uninformative. For example, the prediction sets can always contain $1 - \alpha$ of the possible labels and thus always comply with a coverage grant of $1 - \alpha$. This means that the prediction set is no better than guessing.

Moreover, we must be carefully interpreting the prediction sets of CP methods. The coverage guarantee holds only on average, meaning we will will make exactly α errors on the long run. An error occurs when the correct label is not in the prediction set (or the prediction set is empty). Therefore we must say, that the coverage guarantee only applies apriori, but once we have seen a specific prediction, we can not say that the probability for that prediction to be wrong is α. To illustrate this consider a a two class example. Many prediction sets will contain both classes. This prediction can not make a error by definition. Therefor all remaining errors must have been made in the singleton predictions (or if occurring, in the empty predictions). Therefore, once we have observed a singleton prediction, the probability for that being incorrect is most likely to be much higher than α. Nevertheless, Conformal Prediction is an extremely useful framework for equipping machine learning models with statistical rigid certainty guarantees. Many methods require only little additional computational effort and the only assumption, the exchangability of the data, is usually a basic condition for the underling models anyway. In addition to pure classifications and regression tasks, various types of CP methods, can be used for change point detection, the calibration of predictive distributions or empirical risk control. More recent work in this field is also aimed at time-series data and allows the use of CP methods beyond exchangability.

To put it in a nutshell, for many applications, Conformal Prediction provides an elegant method to make the uncertainty associated with a model transparent. Based on a freely adjustable uncertainty measure, good models provide predictions that contain only a few labels and thus come close to a point prediction. But even more importantly, poor models provide large and therefore meaningless prediction sets. Consequently, the outcomes of CP methods are easy to interpret, even for non-professionals. Therefore, Conformal Prediction is a powerful framework for constructing trustworthy machine learning models and could even justify their use in sensitive application domains such as medical care.

References

Angelopoulos, A., Bates, S., Malik, J., & Jordan, M. I. (2020). Uncertainty sets for image classifiers using conformal prediction. *arXiv preprint arXiv:2009.14193*, -.

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

Angelopoulos, A. N., Bates, S., Candès, E. J., Jordan, M. I., & Lei, L. (2021). Learn then test: Calibrating predictive algorithms to achieve risk control. *arXiv preprint arXiv:2110.01052*, *51*.

Angelopoulos, A. N., Kohli, A. P., Bates, S., Jordan, M., Malik, J., Alshaabi, T., Upadhyayula, S., & Romano, Y. (2022). Image-to-image regression with distribution-free uncertainty quantification and applications in imaging. In *International Conference on Machine Learning*, pp. 717–730. PMLR.

Barber, R. F., Candes, E. J., Ramdas, A., & Tibshirani, R. J. (2023). Conformal prediction beyond exchangeability. *The Annals of Statistics*, *51*(2), 816–845.

Bates, S., Candès, E., Lei, L., Romano, Y., & Sesia, M. (2023). Testing for outliers with conformal p-values. *The Annals of Statistics*, *51*(1), 149–178.

Candès, E. (2020). Stanford statisticians and washington post data scientists build more honest prediction models..

Cherian, J., & Bronner, L. (2021). How the washington post estimates outstanding votes for the 2020 presidential election..

Derhacobian, A., Guibas, J., Li, L., & Namboothiry, B. (2022). Adaptive prediction sets with class conditional coverage. ,, ,.

Dewolf, N., Baets, B. D., & Waegeman, W. (2023). Valid prediction intervals for regression problems. *Artificial Intelligence Review*, *56*(1), 577–613.

Fontana, M., Zeni, G., & Vantini, S. (2023). Conformal prediction: a unified review of theory and new challenges. *Bernoulli*, *29*(1), 1–23.

Gammerman, A., Vovk, V., & Vapnik, V. (1998a). Learning by transduction. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, UAI'98, p. 148–155, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Gammerman, A., Vovk, V., & Vapnik, V. (1998b). Learning by transduction, vol uai'98..

Gneiting, T., & Katzfuss, M. (2014). Probabilistic forecasting. *Annual Review of Statistics and Its Application*, *1*, 125–151.

Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. In *International conference on machine learning*, pp. 1321–1330. PMLR.

Hesterberg, T. C. (2015). What teachers should know about the bootstrap: Resampling in the undergraduate statistics curriculum. *The american statistician*, *69*(4), 371–386.

Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, ,.

Johansson, U., Boström, H., Löfström, T., & Linusson, H. (2014). Regression conformal prediction with random forests. *Machine learning*, *97*, 155–176.

Johansson, U., & Gabrielsson, P. (2019). Are traditional neural networks well-calibrated?. In *2019 international joint conference on neural networks (IJCNN)*, pp. 1–8. IEEE.

Johansson, U., Löfström, T., & Sundell, H. (2018). Venn predictors using lazy learners. In *The 2018 World Congress in Computer Science, Computer Engineering & Applied Computing, July 30-August 02, Las Vegas, Nevada, USA*, pp. 220–226. CSREA Press.

Johansson, U., Löfström, T., & Boström, H. (2019). Calibrating probability estimation trees using venn-abers predictors. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, pp. 28–36. SIAM.

Koklu, M., & Ozkan, I. A. (2020). Multiclass classification of dry beans using computer vision and machine learning techniques. *Computers and Electronics in Agriculture*, *174*, 105507.

Kolmogorov, A. (1968). Logical basis for information theory and probability theory. *IEEE Transactions on Information Theory*, *14*(5), 662–664.

Kolmogorov, A. N. (1983). Combinatorial foundations of information theory and the calculus of probabilities. *Russian mathematical surveys*, *38*(4), 29.

Kull, M., Silva Filho, T., & Flach, P. (2017). Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. In *Artificial Intelligence and Statistics*, pp. 623–631. PMLR.

Kumar, A., Liang, P. S., & Ma, T. (2019). Verified uncertainty calibration. *Advances in Neural Information Processing Systems*, *32*.

Lambrou, A., Papadopoulos, H., Nouretdinov, I., & Gammerman, A. (2012). Reliable probability estimates based on support vector machines for large multiclass datasets. In *Artificial Intelligence Applications and Innovations: AIAI 2012 International Workshops: AIAB, AIeIA, CISE, COPA, IIVC, ISQL, MHDW, and WADTMB, Halkidiki, Greece, September 27-30, 2012, Proceedings, Part II 8*, pp. 182–191. Springer.

Leathart, T., Frank, E., Pfahringer, B., & Holmes, G. (2019). On calibration of nested dichotomies. In *Advances in Knowledge Discovery and Data Mining: 23rd Pacific-Asia Conference, PAKDD 2019, Macau, China, April 14-17, 2019, Proceedings, Part I 23*, pp. 69–80. Springer.

Lei, J. (2014). Classification with confidence. *Biometrika*, *101*(4), 755–769.

Lei, J., G'Sell, M., Rinaldo, A., Tibshirani, R. J., & Wasserman, L. (2018). Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, *113*(523), 1094–1111.

Lei, J., & Wasserman, L. (2014). Distribution-free prediction bands for non-parametric regression. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *76*(1), 71–96.

Manokhin, V. (2022a). Awesome conformal prediction..

Manokhin, V. (2022b). *Machine Learning for Probabilistic Prediction (PhD thesis, VALERY MANOKHIN)*. Ph.D. thesis, Royal Holloway University of London.

Molnar, C. (2023). *Introduction To Conformal Prediction With Python*. c/o MUCBOOK.

Mukhoti, J., Kulharia, V., Sanyal, A., Golodetz, S., Torr, P., & Dokania, P. (2020). Calibrating deep neural networks using focal loss. *Advances in Neural Information Processing Systems*, *33*, 15288–15299.

Nguyen, K. A. (2021). Venn predictors tutorial at copa 2020..

Niculescu-Mizil, A., & Caruana, R. (2005). Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pp. 625–632.

Nouretdinov, I., Volkhonskiy, D., Lim, P., Toccaceli, P., & Gammerman, A. (2018). Inductive venn-abers predictive distribution. In *Conformal and Probabilistic Prediction and Applications*, pp. 15–36. PMLR.

Pakdaman Naeini, M. (2017). *OBTAINING ACCURATE PROBABILITIES USING CLASSIFIER CALIBRATION*. Ph.D. thesis, University of Pittsburgh.

Papadopoulos, H. (2008). Inductive conformal prediction: Theory and application to neural networks. In *Tools in artificial intelligence*. Citeseer.

Papadopoulos, H., Proedrou, K., Vovk, V., & Gammerman, A. (2002). Inductive confidence machines for regression. In *Machine Learning: ECML 2002: 13th European Conference on Machine Learning Helsinki, Finland, August 19–23, 2002 Proceedings 13*, pp. 345–356. Springer.

Pereyra, G., Tucker, G., Chorowski, J., Kaiser, Ł., & Hinton, G. (2017). Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*, ,.

Romano, Y., Patterson, E., & Candes, E. (2019). Conformalized quantile regression. *Advances in neural information processing systems*, *32*.

Romano, Y., Sesia, M., & Candes, E. (2020). Classification with valid and adaptive coverage. *Advances in Neural Information Processing Systems*, *33*, 3581–3591.

Sadinle, M., Lei, J., & Wasserman, L. (2019). Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association*, *114*(525), 223–234.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826.

Vapnik, V. N. (1998). Adaptive and learning systems for signal processing communications, and control. *Statistical learning theory*, -.

Vovk, V. (2012). Conditional validity of inductive conformal predictors. In *Asian conference on machine learning*, Vol. -, pp. 475–490. PMLR.

Vovk, V. (2021). Testing randomness online. *Statistical Science*, *36*(4), 595–611.

Vovk, V., Lindsay, D., Nouretdinov, I., & Gammerman, A. (2003). Mondrian confidence machine. *Technical Report*, -.

Vovk, V., & Petej, I. (2012). Venn-abers predictors. *arXiv preprint arXiv:1211.0025*, ,.

Vovk, V., Petej, I., & Fedorova, V. (2015). Large-scale probabilistic predictors with and without guarantees of validity. *Advances in Neural Information Processing Systems*, *28*.

Vovk, V., Petej, I., Nouretdinov, I., Ahlberg, E., Carlsson, L., & Gammerman, A. (2021). Retrain or not retrain: Conformal test martingales for change-point detection. In *Conformal and Probabilistic Prediction and Applications*, pp. 191–210. PMLR.

Vovk, V., Shafer, G., & Nouretdinov, I. (2003). Self-calibrating probability forecasting. *Advances in neural information processing systems*, *16*.

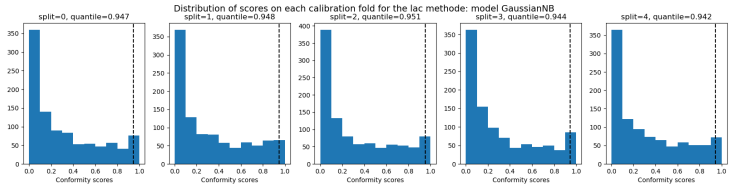
Vovk, V., Gammerman, A., & Saunders, C. (1999). Machine-learning applications of algorithmic randomness. -, -.

Wang, G., Lu, Z., Wang, P., Zhuang, S., & Wang, D. (2023). Conformal test martingale-based change-point detection for geospatial object detectors. *Applied Sciences*, *13*(15), 8647.

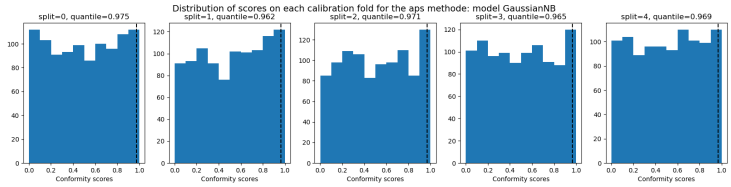
9. Appendix

The following appendix contains the results not presented in the main section for the various experiments, and are listed below for completeness.

9.1 Results: Evaluation of different Splits



(a) least ambiguous set-valued classifier score

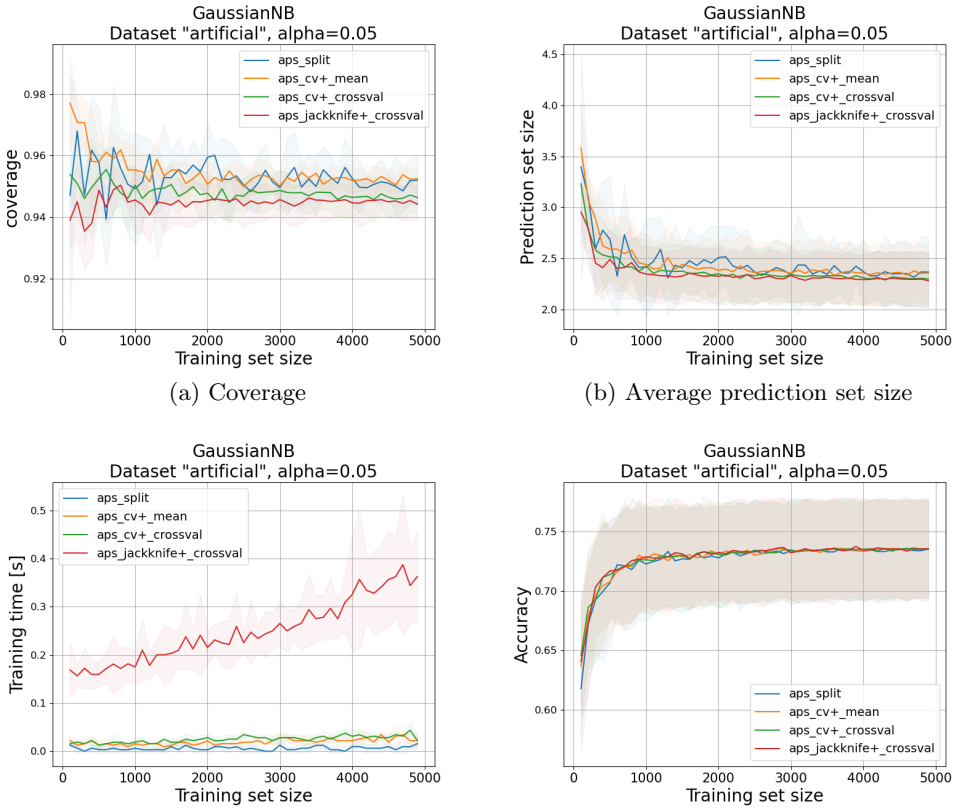


(b) adaptive prediction set score

Figure 20: Distribution over the conformal scores and the resulting .95 quantile on 5 different splits using the least ambiguous set-valued classifier method (lac) and the adaptive prediction set score (aps) based on a Gaussian Naive Bayes.

9.2 Results: Evaluation of different CP Methods under small Datasets

9.2.1 Artificial dataset



(a) Coverage

(b) Average prediction set size

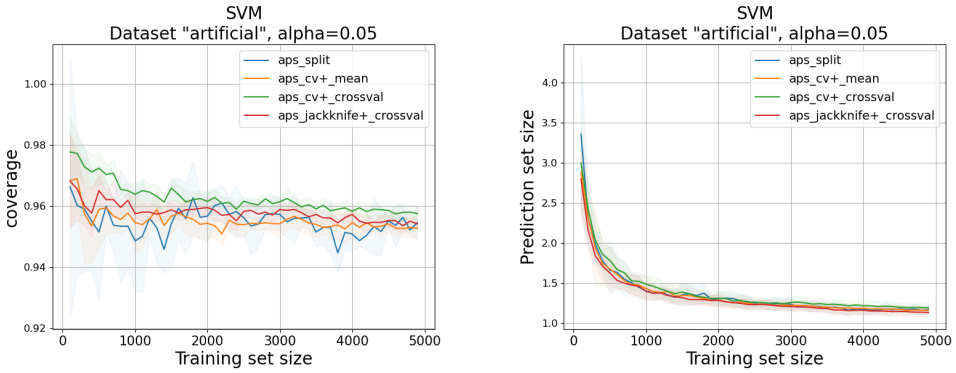
(c) Training time

(d) Accuracy

Figure 21: Effective coverage, average prediction set size, training time and accuracy for the different conformal methods Split, CV+ with $k = 5$ and Jackknife+ with increasing training set sizes of an **artificial created dataset** with (20 features and 5 classes). Based on a Bayes Naive Gaussian model. CV+ scores are aggregated using an averaging (mean) method and the cross validation approach according to (Romano et al., 2020). Conformal scores are computed using the adaptive prediction set method (aps) for $\alpha = 0.05$. The colored areas indicate the standard deviation.

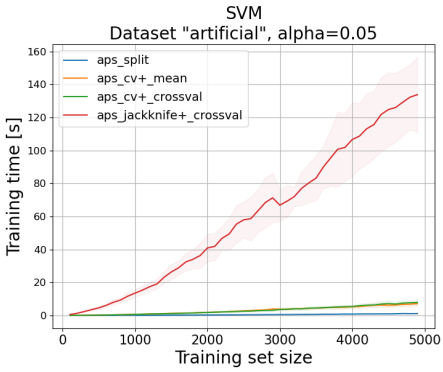
9.2.2 Dry beans dataset

9.3 Results: Conformal Prediction for the smallest possible Datasets

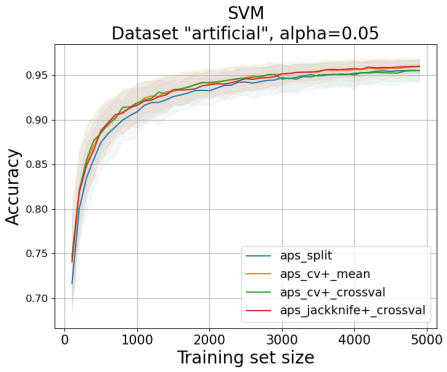


(a) Coverage

(b) Average prediction set size

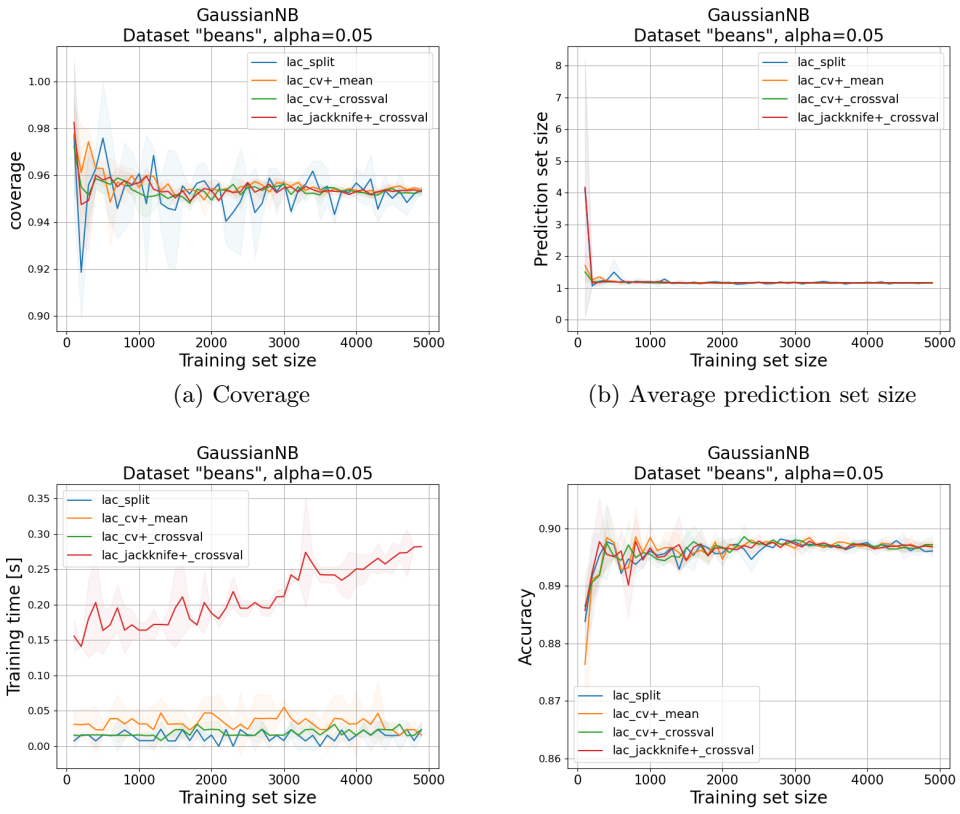


(c) Training time



(d) Accuracy

Figure 22: Effective coverage, average prediction set size, training time and accuracy for the different conformal methods Split, CV+ with $k = 5$ and Jackknife+ with increasing training set sizes of an **artificial created dataset** with (20 features and 5 classes), based on a support vector machine with radial basis function kernel. CV+ scores are aggregated using an averaging (mean) method and the cross validation approach following (Romano et al., 2020). Conformal scores are computed using the adaptive prediction set method (aps) for $\alpha = 0.05$. The colored areas indicate the standard deviation.



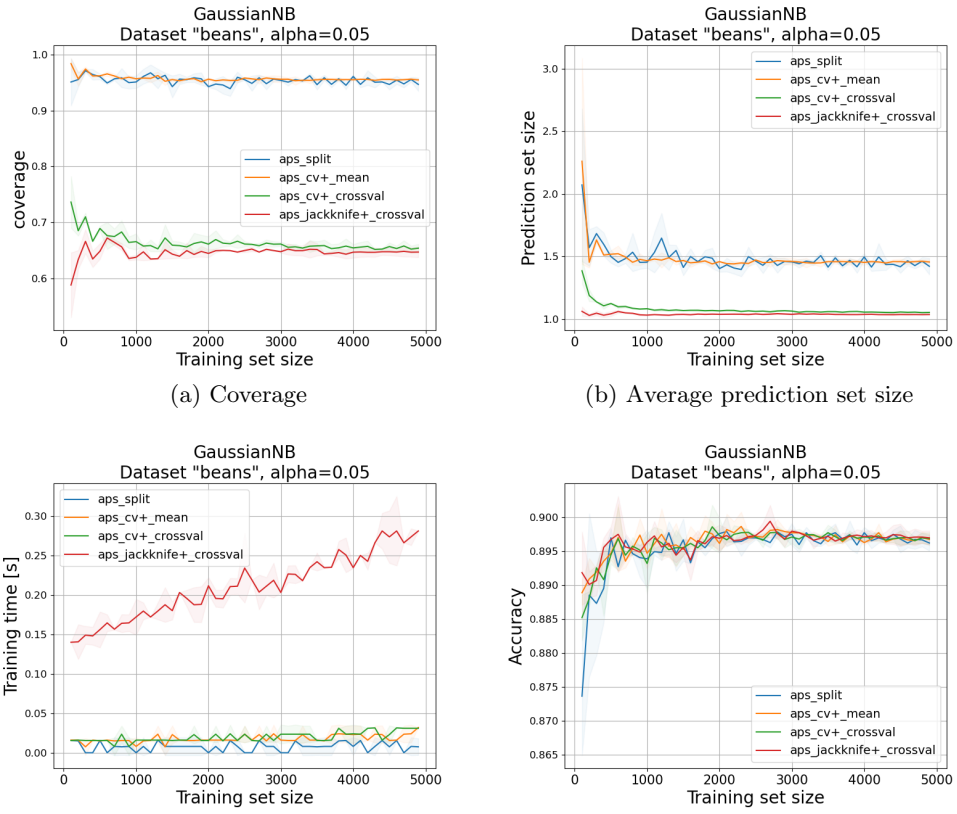
(a) Coverage

(b) Average prediction set size

(c) Training time

(d) Accuracy

Figure 23: Effective coverage, average prediction set size, training time and accuracy for the different conformal methods Split, CV+ with $k = 5$ and Jackknife+ with increasing training set sizes of of the **dry beans dataset**, based on a Bayes Naive Gaussian model. CV+ scores are aggregated using an averaging (mean) method and the cross validation approach following (Romano et al., 2020). Conformal scores are computed using the least ambiguous set-valued classifier method (lac) for $\alpha = 0.05$. The colored areas indicate the standard deviation.



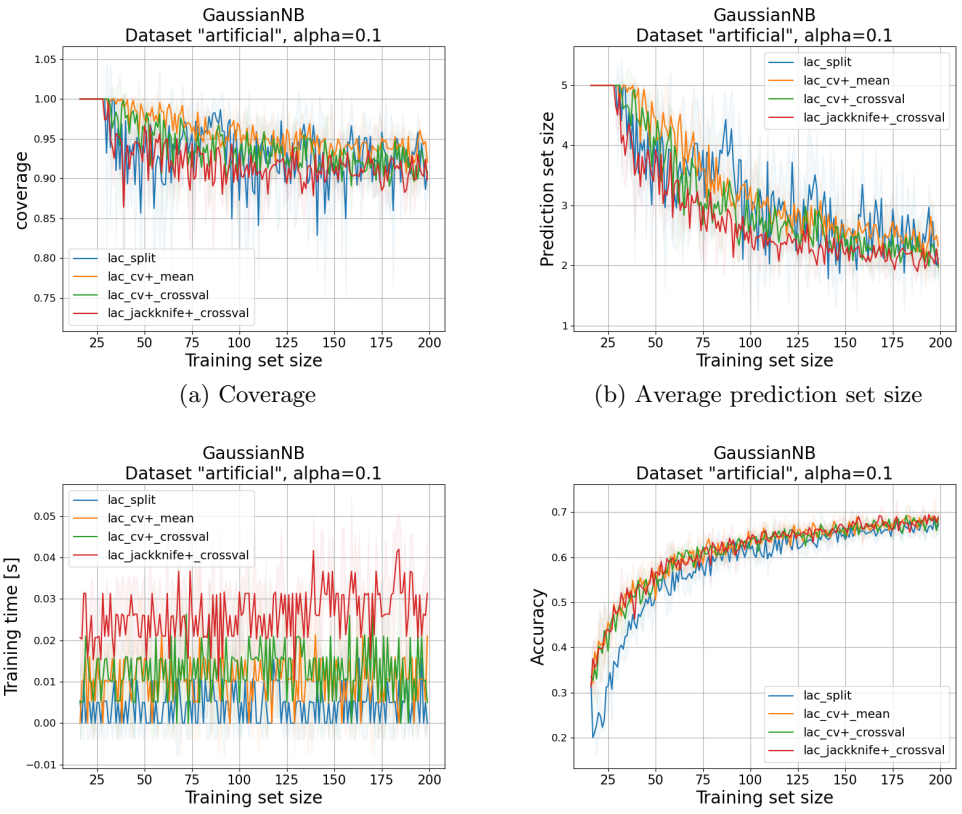
(a) Coverage

(b) Average prediction set size

(c) Training time

(d) Accuracy

Figure 24: Effective coverage, average prediction set size, training time and accuracy for the different conformal methods Split, CV+ with $k = 5$ and Jackknife+ with increasing training set sizes of of the **dry beans dataset**, based on a Bayes Naive Gaussian model. CV+ scores are aggregated using an averaging (mean) method and the cross validation approach following (Romano et al., 2020). Conformal scores are computed using the adaptive prediction set method (aps) for $\alpha = 0.05$. The colored areas indicate the standard deviation.



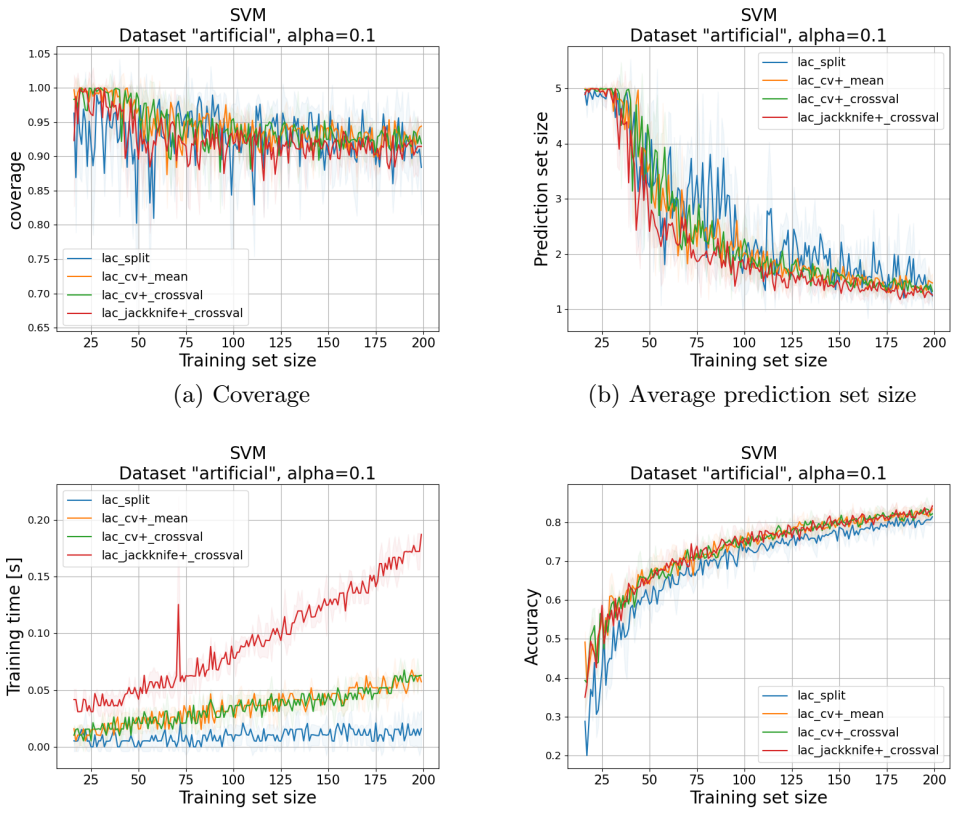
(a) Coverage

(b) Average prediction set size

(c) Training time

(d) Accuracy

Figure 25: Effective coverage, average prediction set size, training time and accuracy for the different conformal methods Split, CV+ with $k = 5$ and Jackknife+ with $16 - 200$ randomly sampled data points of the **artificial dataset**, based on a Bayes Naive Gaussian model. CV+ scores are aggregated using an averaging (mean) method and the cross validation approach following (Romano et al., 2020). Conformal scores are computed using the least ambiguous set-valued classifier method (lac) for $\alpha = 0.1$. The colored areas indicate the standard deviation.



(a) Coverage

(b) Average prediction set size

(c) Training time

(d) Accuracy

Figure 26: Effective coverage, average prediction set size, training time and accuracy for the different conformal methods Split, CV+ with $k = 5$ and Jackknife+ with $16 - 200$ randomly sampled data points of the **artificial dataset**, based on a support vector machine with radial basis function kernel. CV+ scores are aggregated using an averaging (mean) method and the cross validation approach following (Romano et al., 2020). Conformal scores are computed using the least ambiguous set-valued classifier method (lac) for $\alpha = 0.1$. The colored areas indicate the standard deviation.