Synergy Conformal Prediction

Niharika Gauraha NIHARIKA.GAURAHA@FARMBIO.UU.SE
Department of Pharmaceutical Biosciences
Uppsala University
Uppsala, Sweden

Ola Spjuth OLA.SPJUTH@FARMBIO.UU.SE
Department of Pharmaceutical Biosciences
Uppsala University
Uppsala, Sweden

Editor: Lars Carlsson, Zhiyuan Luo, Giovanni Cherubin and Khuong An Nguyen

Abstract

Conformal prediction is a machine learning methodology that produces valid prediction regions. Ensembles of conformal predictors have been proposed to improve the informational efficiency of inductive conformal predictors by combining p-values, however, the validity of such methods has been an open problem. We introduce synergy conformal prediction which is an ensemble method that combines monotonic conformity scores, and is capable of producing valid prediction intervals. We study the applicability in three scenarios; where data is partitioned, where an ensemble of different machine learning methods is used, and where data is unpartitioned. We evaluate the method on 10 data sets and show that the synergy conformal predictor produces valid prediction intervals that on partitioned data performs well compared to the most efficient model trained on individual partitions, making it a viable approach for federated settings when data cannot be pooled. We also show that our method has advantages over current ensembles of conformal predictors by producing valid and efficient results on unpartitioned data, and that it is less computationally demanding.

Keywords: Conformal Prediction, Machine Learning, Synergy Conformal Prediction, Big Data, Federated Learning, Conformal Predictor Ensembles

1. Introduction

Conformal prediction is an established method in the machine learning (ML) landscape that yields valid prediction regions for new objects (Vovk et al., 2005). Transductive conformal prediction (Vovk, 2013) and Inductive conformal prediction (Papadopoulos, 2008) are two approaches for the construction of prediction regions. The transductive approach requires re-training the model on each prediction, and inductive conformal prediction was developed to overcome this. The validity of Transductive Conformal Predictors (TCPs) and Inductive Conformal Predictors (ICPs) is proven in that they produce $1 - \epsilon$ expectation tolerance regions, where ϵ is the selected significance level (Vovk et al., 2005).

The inductive conformal prediction approach comes at the expense that part of the data needs to be set aside as a calibration set, and the model hence becomes less informationally efficient (the predicted regions are larger). Conformal predictor ensembles, such as cross-conformal predictors (Vovk, 2015) and aggregated conformal predictors (Carlsson et al., 2014b), have been developed to improve the informational efficiency of ICPs. The idea is

to train multiple ICPs to make better use of the examples for modeling and calibration, and aggregate the resulting p-values. However, the validity of these ensemble methods has been an open problem for researchers (Linusson et al., 2017). Though it has been shown to achieve empirical validity under certain conditions, its theoretical validity has not been proven yet and it remains a practical problem in many settings. Since the aforementioned conformal predictor ensembles try to aggregate conformal p-values, which usually does not yield standard uniform distribution, the validity is not guaranteed.

Our research tries to answer these two questions: (1) Can we partition a dataset and train multiple models and obtain valid prediction regions with good accuracy when combined? (2) Can we train multiple models on a dataset using different modeling methods, and obtain valid prediction regions with good accuracy when combined?

In this manuscript, we introduce *Synergy Conformal Prediction* (SCP) which is a method that aggregates monotonic conformity scores instead of p-values. This work is inspired by the method of "Synergy of Monotonic Rules" proposed by Vapnik et al. (Vapnik and Izmailov, 2016), in which the authors combine the estimated conditional probabilities to construct the optimal synergy rules for pattern recognition problems. Here we focus on classification problems.

The organization of the paper is as follows. In section 2, we introduce the background concepts and notations used throughout the paper. In Section 3, we present synergy conformal prediction and we prove its useful properties. In Section 4, we perform numerical analysis on a set of real data sets. In Section 5, we discuss the results and the applications of SCP. Finally, in Section 6, we conclude and provide directions for further research.

2. Background

This section gives a brief background about conformal prediction framework and fixes notations and assumptions used throughout the paper.

The object space is denoted by $\mathcal{X} = \mathbb{R}^q$, where q is the number of features, and label space is denoted by $\mathcal{Y} \in \{1, \ldots, K\}$, where K is the number of labels. We assume that each example consists of an object and its label, and its space is given as $\mathcal{Z} := \mathcal{X} \times \mathcal{Y}$. In a typical classification problem, given a training dataset $Z = \{z_1, ..., z_\ell\}$ – where ℓ is the number of examples in the training set, and each example $z_i = (x_i, y_i)$ is labeled – we want to predict the label of a new object x_{new} whose label is unknown. Our experiments are limited to classification problems, and we also assume the exchangeability of examples throughout the paper. The random variables $x_1, \ldots, x_\ell$ are said to be exchangeable under a probability measure p if the joint probability distribution satisfies the following equality condition,

$$p(x_1, \ldots, x_\ell) = p(x_{\pi(1)}, \ldots, x_{\pi(\ell)}),$$

for all permutations π defined on the set $\{1, \ldots, \ell\}$.

In the following, we define conformity measures and conformity scores. The conformity measure is the score from a function that measures the closeness of an example in relation to the previous examples.

Conformal prediction provides a layer on top of an existing machine learning method and uses available data to determine valid prediction regions for new objects (Vovk et al.,

2005). Conformal prediction was primarily defined as an online transductive framework, in which the underlying model must be retrained each time an object is to be predicted and is hence computationally expensive. For further details on the transductive approach, we refer to (Vovk et al., 2005). A more computationally efficient inductive framework was then proposed as an alternative to the transductive approach. In the inductive approach, the training data has to be divided into a proper training set and a calibration set. Since ICP is a key building block for our method, we define it in a bit more detail in the following; for full length details we refer to (Papadopoulos, 2008).

Definition 1 Inductive Conformal Predictor (ICP)
Given a training set of ℓ examples, $Z = \{z_1, ..., z_\ell\}$, drawn from an exchangeable distribution P. The training data is first divided into a proper training set $\{Z_T\}$ and a calibration set $\{Z_C\}$, where (T, C) is a partition of $\{1, ..., \ell\}$. Let $\mathcal{A}$ be a conformity measure, and the conformity score $\mathcal{A}(Z_T, z)$ is used to measure how well the example z conforms to the proper training set Z_T. For example,

$$\mathcal{A}(Z_T, (x, y)) = p(Y = y \mid f(x)), \tag{1}$$

where $f : X \to Y$, is a prediction rule of the model trained on the proper training set Z_T, and $p(Y = y \mid f(x))$ is a calibrated conditional probability. The predictive model trained on the proper training set Z_T, is then used to compute the conformity scores (i.e. class conditional probabilities) for the calibration set, $\alpha_j^y = p(Y = y \mid f(x_j))$, for $j \in C$ and $y \in \mathcal{Y}$. Let x_{new} (follows the same distribution P) be the example we want to predict, and let α_{new}^y be its conformity scores computed using the same function $\mathcal{A}(Z_T, .)$. The ICP p-values are then computed as

$$p_{new}^y = \frac{|j \in C_y; \alpha_{new}^y > \alpha_j^y| + \tau |j \in C_y; \alpha_{new}^y = \alpha_j^y|}{|C_y| + 1}, \tag{2}$$

where C_y denotes the class-wise partition of C (if $j \in C_y$, then $j \in C$ and $y_j = y$), $y \in \mathcal{Y}$, and $\tau \in [0, 1]$ is a random number.
The inductive conformal predictor corresponding to the tuple $(Z_C, \mathcal{A}(Z_T, .))$ is defined as a set predictor

$$\Gamma^\epsilon = \{y \mid p^y > \epsilon\}, \tag{3}$$

where $\epsilon \in (0, 1)$ is a chosen significance level, and $(1 - \epsilon)$ is known as confidence level.

As mentioned before, in ICP some examples are used for modeling only and some are used for calibration only, and this makes ICP less informationally efficient than TCP. To improve the informational efficiency of ICP, ensembles of conformal predictors were proposed. Cross Conformal Predictor (CCP), and Bootstrap Conformal Predictor (BCP) were proposed in (Vovk, 2015) and it was generalized as Aggregated Conformal Predictor (ACP) in (Carlsson et al., 2014a). In CCP, training data is divided into separate folds and each fold is used as calibration set and remaining data is used as a proper training set, and eventually p-values are averaged across all folds. In BCP, the training set is bootstrapped to obtain a proper training set and out-of-bag examples are used as a calibration set. The p-values are then aggregated across bootstrap replications. The ACP is a generalization of CCP and

BCP, where a consistent resampling scheme is used for constructing the calibration set. We refer to (Carlsson et al., 2014a) for details on ACP.

Other ensemble methods in the conformal prediction framework have been proposed, including (Toccaceli and Gammerman, 2018), (Löfström et al., 2013) and (Balasubramanian et al., 2015). In all the above ensemble methods, the main aim was to get more informationally efficient conformal predictors by combining p-values. However, most of the resulting models are not guaranteed to be valid, as the combined p-values need not be uniformly distributed on (0,1). For a detailed study on the validity of such ensemble methods, we refer to (Linusson et al., 2017).

3. Synergy Conformal Prediction

In all the ensemble methods discussed in the previous section, the key idea is to combine p-values computed from various ICPs. We propose a novel method of combining (monotonic) conformity scores for the calibration set and for the test examples computed from various trained models. In the following, we define "Synergy Conformal Prediction" (SCP).

Given ℓ examples, $Z = \{z_1, ..., z_\ell\}$, drawn from an exchangeable distribution P. Akin to the ICP method, here also the training data is first divided into the proper training set $\{Z_T\}$ and the calibration set $\{Z_C\}$, where (T, C) is a partition of $\{1, ..., \ell\}$. But then the proper training data is further divided into M non-empty disjoint subsets and each subset $Z_{T_m}, m = 1, \ldots, M$ is then used as an actual training set for modeling. $(T_1, ..., T_M)$ is a partition of T. The M predictive models trained on individual partitions are then used to compute the monotonic conformity scores (i.e. class conditional probabilities) for the calibration set denoted by α^y_{mj}, for $j \in C$, $m = 1, \ldots, M$, $y \in \mathcal{Y}$. For example,

$$\alpha^y_{mj} = p_m(Y = y \mid f_m(x_j)), \tag{4}$$

where $f_m(x)$ is the prediction rule defined by the predictive model trained on the m^{th} part of the training set. The aggregated conformity scores across models are then defined as

$$\alpha^y_j = \frac{1}{M} \sum_{m=1}^{M} \alpha^y_{mj}$$

Let x_{new} (follows the same distribution P) be the example we want to predict, and let α^y_{new} be the aggregated conformity scores across models. The SCP p-values are then computed using eq. (2). The synergy conformal predictor corresponding to the tuple $(Z_C, \mathcal{A}(Z_{T_1}, .), ..., \mathcal{A}(Z_{T_M}, .))$ is defined as a set predictor as given in eq. (3). The main steps are summarized in Algorithm 1.

The SCP method differs from the other ensemble methods discussed in the previous section by:

1. Traditional ensemble methods combine conformal p-values obtained from different ICPs.

2. Within the inductive conformal prediction framework, SCP tries to improve the informational efficiency by combining monotonic conformity scores (for example, calibrated probability estimates from support vector machines).

Algorithm 1 Synergy Conformal Predictor

Input: training dataset:Z, object to predict:x_{new}, a conformity measure:A

Output: p-values

Step1: Split the training set into two smaller sets, $\{Z_T\}$ and the calibration set $\{Z_C\}$. (T, C) is a partition of $\{1, ..., |Z|\}$.
Split the set Z_T into M proper training sets, $\{Z_{T_m}, m = 1, \ldots, M\}$. $(T_1, ..., T_M)$ is a partition of T.

Step2: For each part Z_{T_m}, train and construct the rule to generate conformity scores.

Step3: Compute the aggregated conformity scores across M models for each example in the calibration set. α_j^y, for $j \in C$ and $y \in \mathcal{Y}$.

Step4: Compute the aggregated conformity scores across M predictive models to the object x_{new}, which results in α_{new}^y, for $y \in \mathcal{Y}$.

Step5: Compute p-values for each of the K classes using eq. 2.

return $(p_{new}^1, \ldots, p_{new}^K)$;

3.1. Properties of Synergy Conformal Predictors

In this section, we discuss the properties of SCP. First we prove that the SCP algorithm produces valid predictions. Then, we also prove that the SCP is more efficient than each individual small ICP of a partitioned dataset.

Proposition 1 The synergy conformal predictor is valid.

Proof:

When we do not partition the proper training set into smaller subsets, in that case SCP is exactly ICP and hence valid. In particular for this case, the whole partition Z_T is used for predictive modeling, and the corresponding calibration set Z_C and the conformity measure based on the prediction rule obtained from Z_T, forms an ICP, and an ICP is proven to be valid (Vovk et al., 2005).

Alternatively, SCP can be viewed as a single ICP and hence valid, where synergy of monotonic conformity scores is considered as one function producing the (aggregated) conformity scores. To illustrate this, let us consider partition of the set Z_T into two subsets, Z_{T_1} and Z_{T_2}, and let their corresponding rules for computing conformity scores be $\mathcal{A}(Z_{T_1}, .)$ and $\mathcal{A}(Z_{T_2}, .)$ respectively. By the definition of ICP, each individual ICP based on the tuple $\{Z_C, \mathcal{A}(Z_{T_m}, z)\}$ for $m = 1, 2$, is valid. Let us define a new conformity score $\mathcal{A}(Z_T, .)$ which aggregated the conformity scores of an example z, $\mathcal{A}(Z_T, z) = \frac{1}{2}\sum_{m=1}^{2} \mathcal{A}(Z_{T_m}, z)$. The pair $\{Z_C, \mathcal{A}(Z_T, .)\}$ forms an ICP corresponding to the new conformity measure $\mathcal{A}(Z_T, .)$, hence valid, and it can be generalized for any number of subsets.

Proposition 2 The synergy conformal predictor corresponding to the tuple $(Z_C, \mathcal{A}(Z_{T_1}, .), ..., \mathcal{A}(Z_{T_M}, .))$ is more informationally efficient that each individual small ICP corresponding to the tuple $(Z_C, \mathcal{A}(Z_{T_m}, .))$ for $m = 1, \ldots, M$.

Proof:

The choice of calibrated conditional probability of support vector machine as conformity measure, is a monotonically increasing function, in the sense that the higher the score of an object x, the higher the probability of x belonging to the positive class. The aggregation

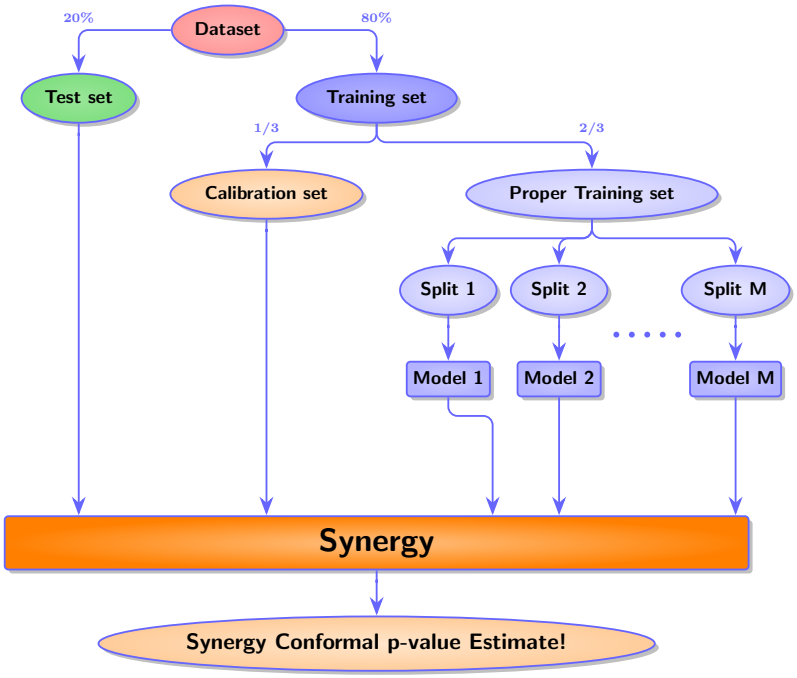


Figure 1: An overview of synergy conformal prediction: the given dataset is randomly partitioned into a test set (20%) and the remaining training set (80%) is then split into a proper training set and a calibration set in proportion 2 : 1. The proper training data set is randomly split into M (almost) equal disjoint subsets, "Split 1", ..., "Split M". A model is trained on each split individually. Finally, the (monotonic) conformity scores for the calibration set and for the test examples computed from various trained models are combined to compute the conformal p-values.

of monotonic estimated conditional probabilities (conformity scores) across various models is monotonic, and has been proven to be more accurate than the individual scores (Vapnik and Izmailov, 2016). Moreover, the subsets used for training different predictive models are independent, thus averaging of M conditional probability values decreases the variance of resulting conditional probability by a factor of M. Therefore the combined scores are more robust than the individual ones. Hence SCP is more informationally efficient than each individual ICP. For further details on synergy of monotonic conditional probability functions we refer to (Vapnik and Izmailov, 2016).

Note that the proof can be generalized for any conformity measure that is a monotonic function.

Table 1: Description of the datasets from UCI repository that are used in the evaluation.

Dataset	Training	Calibration	Test
Spambase	2453	1227	921
Breast Cancer	303	152	114
Phishing Websites	5896	2948	2211
Covertype	9296	4648	3486
Adult	17365	8683	6513
Tic-tac-toe	510	256	192
Australian	368	184	138
MONK's-1	296	148	112
MONK's-2	320	160	121
Bank	2410	1206	905

4. Experiments

We evaluate SCP on ten classification datasets from UCI machine learning repository (Lichman et al., 2013), see Table 1 for specifications. We present 5 experiments: SCP using the same ML algorithm on partitioned data, SCP using different ML algorithms on partitioned data, SCP using different ML algorithms on unpartitioned data, Calibration of SCP and Evaluation of running times. For Experiment 1 and 2 we partition data according to the following procedure (see also Figure 1): Each dataset is randomly partitioned into a test set (20%) and the remaining training set (80%) is then split into a proper training set and a calibration set in proportion $2:1$. The proper training data set is randomly split into three equal disjoint subsets (partitions). Each subset is considered as an actual training set to train an underlying machine learning model and compute conditional probabilities (as conformity measure), and then conditional probabilities are aggregated across all the models to compute the p-values.

In all the experiments, we use the calibrated conditional probability from classifiers such as support vector machine (SVM) and random forest (RF) as the comformity measure. The corresponding hyper-parameters are learned using 5-fold cross-validation, and Platt scaling (Platt et al., 1999) is used to transform SVM output to conditional probability. To compute the corresponding p-values, we use the smoothed Mondrian approach (Vovk et al., 2005), where the taxonomy is defined by the labels. To assess the predictive performance of the conformal predictors we consider validity and efficiency. Validity is empirically assessed in terms of calibration plots, the plot of the percentage of errors against $\epsilon \in (0, 1)$. For a valid prediction, the calibration plots are usually very close to the bisector of the first quadrant. We use observed fuzziness (Vovk et al., 2016) as a measure of efficiency, which is defined as the sum of all p-values for the incorrect class labels (a lower value is advantageous). SVM with linear kernel and SVM with RBF kernel is trained using python implementation of LibSVM (Chang and Lin, 2004) in SciKit Learn (Pedregosa et al., 2011). RF is trained using SciKit Learn (Pedregosa et al., 2011). All experiments are repeated 30 times.

4.1. Experiment 1: SCP using the same ML algorithm on partitioned data

The objective of this experiment is to compare SCP with individual ICPs trained on each data partition, and with an ICP trained on the whole proper training set.

We consider the form of SCP as given in Figure 1 with three equal non-overlapping partitions, $M = 3$. Linear Support Vector Machine (linear SVM) and non-linear Support Vector Machine with an RBF kernel (RBF-SVM) were used as the underlying machine learning methods in this experiment. The average efficiency of SCP across 30 repetitions is reported in the third column of Table 2. The first column is the average efficiency of the best performing ICP on a data partition (the lowest value for observed fuzziness), we call it the 'ICP_p'. The second column is the average efficiency of the ICP where the proper training set as a whole is used for modeling. To illustrate the difference between ICP_p, ICP and SCP, box plots are presented in Figure 2(a) and Figure 2(b) for the Australian dataset. Box plots for the other data sets are available in Appendix A.

Table 2: Results from Experiment 1, SCP using the same ML algorithm on partitioned data. Comparison of efficiency between ICP_p, ICP and SCP, where linear SVM and non-linear SVM with RBF kernel are used as the underlying machine learning algorithms. ICP_p corresponds to the model trained on the partition having the lowest (best) efficiency, ICP refers to the efficiency where the whole dataset is used, and SCP to the synergy conformal prediction method.

	Linear			NonLinear		
Dataset	ICP_p	ICP	SCP	ICP_p	ICP	SCP
Spambase	0.359	0.03	0.357	0.38	0.028	0.385
Breast Cancer Wisconsin	0.35	0.004	0.351	0.052	0.004	0.035
Phishing Websites	0.26	0.022	0.259	0.071	0.009	0.057
Covertype	4.965	0.334	4.965	0.67	0.232	0.594
Adult	0.237	0.151	0.237	0.203	0.103	0.177
Tic-tac-toe	0.444	0.427	0.413	0.634	0.001	0.636
Australian	0.299	0.078	0.161	0.15	0.081	0.141
MONK's-1	0.366	0.252	0.397	0.467	0.008	0.422
MONK's-2	0.435	0.496	0.479	0.391	0.028	0.425
Bank	0.126	0.169	0.118	0.161	0.17	0.199

4.2. Experiment 2: SCP using different ML algorithms on partitioned data

The objective of this experiment is to compare SCP with each ICP trained on individual data partitions, in order to show that SCP is a viable method to integrate models from different machine learning algorithms. We use the same setup as in Fig 1 with $M = 3$, and with three different machine learning algorithms: RBF-SVM, linear SVM and Random Forest (RF), one for each data partition. The results are reported in Table 3. To illustrate the difference between the best ICP$_p$ and SCP, box plots are presented in Figure 2(c) for the Australian dataset. Box plots for the other data sets are available in Appendix A.

Table 3: Results from Experiment 2, SCP using different ML algorithms on partitioned data. We report efficiency of ICPs on partitioned data using the three different algorithms: RBF SVM, linear SVM and RF and the efficiency of SCP for these models.

Dataset	SVM-ICP$_p$	RF-ICP$_p$	RBF-SVM-ICP$_p$	SCP
Spambase	0.36	0.047	0.419	0.051
Breast Cancer Wisconsin	0.349	0.062	0.052	0.066
Phishing Websites	0.262	0.034	0.075	0.044
Covertype	4.958	1.056	0.73	1.137
Adult	0.238	0.14	0.217	0.153
Tic-tac-toe	0.454	0.3	0.634	0.434
Australian	0.319	0.133	0.158	0.11
MONK's-1	0.378	0.199	0.462	0.151
MONK's-2	0.563	0.585	0.428	0.512
Bank	0.124	0.111	0.187	0.115

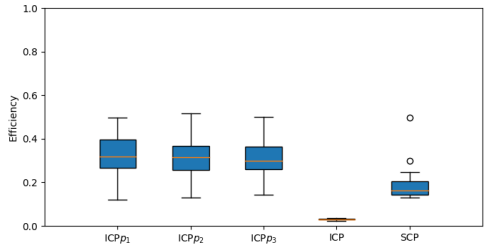
4.3. Experiment 3: SCP using different ML algorithms on unpartitioned data

This objective attempts to explore SCP as an ensemble method on top of different base models with three unpartitioned data sources, and and compare with an ICP on the whole dataset and Cross Conformal Prediction (CCP) with three folds. Three different underlying machine learning algorithms are used: RBF SVM, linear SVM and RF. The efficiencies of all methods are reported in Table 4.

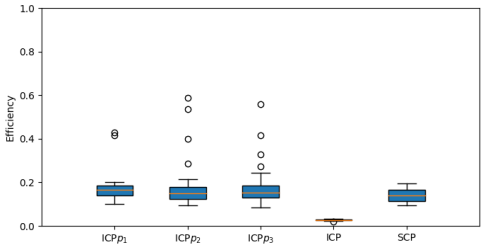
Table 4: Results from Experiment 3, SCP using different ML algorithms on unpartitioned data. Comparison of efficiency between ICP, SCP and CCP. The first three columns show results from the ICP using linear SVM, RBF-SVM and RF as the underlying machine learning algorithms respectively. The fourth column shows results for SCP when ensemble of different machine learning algorithms (RBF SVM, linear SVM and RF) are used on the unpartitioned proper training set, and the fifth, sixth and seventh columns show results for the three fold CCP with linear SVM, RBF-SVM and RF methods used for training.

Dataset	SVM-ICP	RF-ICP	RBF-SVM-ICP	SCP	SVM-CCP	RF-CCP	RBF-SVM-CCP
SB	0.03	0.016	0.028	0.015	0.031	0.016	0.027
BC	0.004	0.01	0.004	0.003	0.006	0.012	0.006
Phishing	0.022	0.006	0.009	0.006	0.022	0.006	0.01
Cover	0.334	0.173	0.232	0.169	0.35	0.173	0.25
Adult	0.151	0.095	0.103	0.096	0.151	0.094	0.103
Tic-tac-toe	0.427	0.013	0.001	0.001	0.434	0.014	0.002
Australian	0.078	0.063	0.081	0.062	0.075	0.067	0.078
Monks-1	0.252	0.0	0.008	0.001	0.248	0.0	0.014
Monks-2	0.496	0.084	0.028	0.034	0.502	0.081	0.037
Bank	0.169	0.1	0.17	0.107	0.167	0.104	0.178

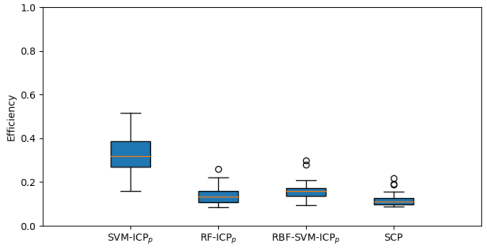
To illustrate the difference between the ICP, SCP and CCP, box plots are presented in Figure 2(d) for Australian dataset. Similar plots for other data sets are available in Appendix A.



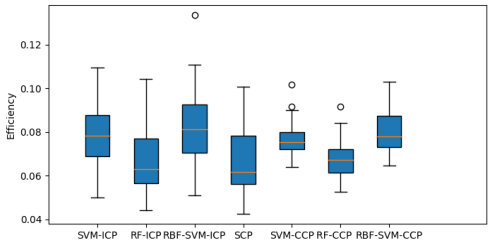
(a) Linear SVM, partitioned data



(b) RBF-SVM, partitioned data



(c) Different ML Models

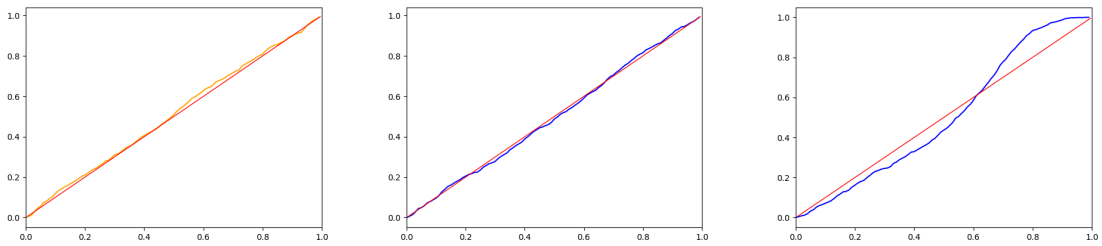


(d) Unpartitioned data

Figure 2: Results from Experiment 1, 2 and 3 comparing the efficiency between different methods for the Australian dataset. (a and b): Experiment 1, comparison of individual ICPs trained on data partitions, ICP on the entire dataset, and SCP where (a) linear SVM is used, and (b) RBF SVM is used as the underlying machine learning algorithm; (c): Experiment 2, comparing individual ICPs on partitions and SCP where three different algorithms (linear SVM, RF and RBF-SVM) are used on partitioned training data; (d): Experiment 3 comparing ICP, SCP and CCP where three methods (linear SVM, RF and RBF-SVM) are used on unpartitioned training data.

4.4. Experiment 4: Calibration of SCP

In this experiment, we compare the calibration of ICP, SCP and CCP. We use the same setup as in Figure 1 with $M = 3$, and with RF using 10 trees as an underlying ML algorithm. We also train a three fold CCP model using RF with 10 trees. Calibration plots for the Spambase dataset are shown in Figure 3. We observe that all the individual ICPs as well as SCP models are valid, in contrast to CCP that shows the same behaviour deviating from validity as reported earlier (Linusson et al., 2017).



(a) ICP (b) SCP (c) CCP

Figure 3: Results from Experiment 4, Calibration of SCP. Calibration plots of ICP, SCP and CCP for the Spambase dataset using random forest with 10 trees for training all the underlying models.

4.5. Experiment 5: Evaluation of running times

In this experiment, we compare the running time of ICP, SCP and CCP. For ICP, we use the whole training set and linear SVM as an underlying machine learning algorithm. For SCP, we use the same setup as in Fig 1 with $M = 3$, and with linear SVM as an underlying ML algorithm. For CCP, we use the whole training set with three fold cross conformal prediction using linear SVM as an underlying machine learning algorithm. The experiment was performed sequentially on one core, and the average running time over 10 runs are reported in Table 5.

Table 5: Results from Experiment 5, Comparison of Running Time. Comparison of average running time over 10 runs (in seconds) between ICP, SCP and CCP.

Dataset	ICP	SCP	CCP
SB	17.39	8.06	47.45
Adult	7654.5	2598.3	21276.2
Covertype	6018.8	2211.8	16121.8

5. Discussions

The purpose of this study was to introduce SCP and demonstrate its properties in different practical settings for predictions with confidence on partitioned and unpartitioned data.

In Experiment 1 (see Table 2), we showed that when using partitioned data, SCP is capable of combining the models and for the case of Linear SVM, in most cases (7 out of 10 datasets) and for Non-Linear SVM (5 out of 10 datasets) obtain improved efficiency than the best individual ICP trained on partitioned data (ICP_p). As expected the efficiency of an ICP trained on the entire dataset is better than SCP and ICP_p, with the exception of the Linear SVM for the two datasets MONKs-2 and Bank. These results support that SCP can be used when data cannot easily be collected into a single dataset. This is the case in a federated setting, where data is located in different locations, and the data cannot be pooled due to privacy or regulatory reasons, or for practical reasons such as long data transfer times. The SCP method allows for efficiently combining such federated data sources, under the condition that a shared calibration set is available in the aggregation process.

Another possibility is to develop new and efficient parallel implementations for predictions, such as in locality-aware Big-data framework. These results open up for the possibility to instead of running long inductive training on a large dataset, to use the SCP method on partitioned data and benefit from reduced modeling time while still getting accurate and valid predictions.

In Experiment 2, we combined multiple modeling methods. Table 3 shows that the model trained with RF method outperform the other models for most of the datasets, but SCP shows improved efficiency for 2 datasets (Australian and MONKs-1). An important observation is that even though the individual models outperform SCP for 8 out of 10 datasets, the SCP efficiency is very close to the best model in all cases apart from the Covertype dataset. This has implications in settings where data cannot be pooled and when models are combined without requiring the same modeling method. Again, this is useful in a federated setting when the different participants in the federation might have different modeling methods and do not wish to disclose details of these. By establishing a shared calibration set and only disclosing nonconformity measures for an object to predict, the SCP method can integrate these results and make predictions.

In Experiment 3 we show on unpartioned training data that SCP for 6 out of 10 datasets outperform ICP and CCP using Linear SVM, RBF-SVM and RF as underlying ML models. Even for the remaining 4 datasets, SCP is very similar to the best performing ICP or CCP. In Experiment 4, we illustrate the theoretically proven validity (proposition 1) by showing that individual small ICPs and SCP are well calibrated (close to the bisector of the first quadrant) even with a non-optimal predictive model (random forest with only 10 trees). In concordance with Linusson et al. (2017) we also note that CCP is poorly calibrated (Figure 3), which is because it (as other previous methods) operate by combining p-values. The validity property of SCP is key to our method, and makes it a practical alternative to especially CCP. SCP also has an additional advantage in being less computationally demanding, as illustrated in Experiment 5, offering faster training times even without parallelization. In most of the experiments, Covertype dataset stands out from others probably due to imbalanced classes.

A drawback of SCP is that it requires a shared calibration set across the individual models, which for the case of partitioned proper training data makes it less efficient as

compared to an ICP of the entire dataset. However for the case when data cannot be pooled, and when using unpartitioned data, SCP offers an ensemble method for conformal predictors as it outputs valid prediction intervals with reasonable to good efficiency depending on the dataset.

6. Conclusions and Future Directions

We presented Synergy Conformal Prediction (SCP), a new ensemble learning method that produces valid prediction intervals for new objects. We demonstrated that the method makes it possible to partition the data and aggregate the resulting predictions, improving the modeling time while retaining reasonable to good accuracy depending on the dataset. We also demonstrated that SCP is a viable approach to the commonly used CCP approach, making it widely applicable as a valid ensemble confidence predictor. Due to its low computational complexity and parallel scalability, we believe the method will be potentially useful for Big-data problems, and due to its isolated and non-sharing data sources and trained models it can have a potential application in federated settings. Future directions when working on partitioned data include (i) studying the effect of the number and size of data partitions, as well as the overlapping partitions (ii) trying with different monotonic (non) conformity scores (using different underlying ML algorithms) with individual partitions.

Acknowledgements

We would like to acknowledge Akshay Chaturvedi for useful discussion. This project received financial support from the Swedish Foundation for Strategic Research (SSF) as part of the HASTE project under the call 'Big Data and Computational Science'. The computations were performed on resources provided by SNIC through Uppsala Multidisciplinary Center for Advanced Computational Science (UPPMAX) under project SNIC 2017/7-273.

References

Vineeth N Balasubramanian, Shayok Chakraborty, and Sethuraman Panchanathan. Conformal predictions for information fusion. *Annals of Mathematics and Artificial Intelligence*, 74(1-2):45–65, 2015.

Lars Carlsson, Martin Eklund, and Ulf Norinder. Aggregated conformal prediction. In Lazaros Iliadis, Ilias Maglogiannis, Harris Papadopoulos, Spyros Sioutas, and Christos Makris, editors, *Artificial Intelligence Applications and Innovations*, pages 231–240, Berlin, Heidelberg, 2014a. Springer Berlin Heidelberg. ISBN 978-3-662-44722-2.

Lars Carlsson, Martin Eklund, and Ulf Norinder. Aggregated conformal prediction. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 231–240. Springer, 2014b.

Chih-Chung Chang and Chih-jen Lin. C.-j.: Libsvm: a library for support vector machines. 2004.

Moshe Lichman et al. Uci machine learning repository, 2013.

Henrik Linusson, Ulf Norinder, Henrik Boström, Ulf Johansson, and Tuve Löfström. On the calibration of aggregated conformal predictors. In Alex Gammerman, Vladimir Vovk, Zhiyuan Luo, and Harris Papadopoulos, editors, *Proceedings of the Sixth Workshop on Conformal and Probabilistic Prediction and Applications*, volume 60 of *Proceedings of Machine Learning Research*, pages 154–173, Stockholm, Sweden, 13–16 Jun 2017. PMLR. URL http://proceedings.mlr.press/v60/linusson17a.html.

Tuve Löfström, Ulf Johansson, and Henrik Boström. Effective utilization of data in inductive conformal prediction. In *International Joint Conference on Neural Networks, Dallas, TX, USA, August 4-9, 2013.* IEEE, 2013.

Harris Papadopoulos. Inductive conformal prediction: Theory and application to neural networks. In *Tools in artificial intelligence*. InTech, 2008.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

Paolo Toccaceli and Alexander Gammerman. Combination of inductive mondrian conformal predictors. *Machine Learning*, pages 1–22, 2018.

Vladimir Vapnik and Rauf Izmailov. Synergy of monotonic rules. *The Journal of Machine Learning Research*, 17(1):4722–4754, 2016.

Vladimir Vovk. Transductive conformal predictors. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 348–360. Springer, 2013.

Vladimir Vovk. Cross-conformal predictors. *Annals of Mathematics and Artificial Intelligence*, 74(1-2):9–28, 2015.

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

Vladimir Vovk, Valentina Fedorova, Ilia Nouretdinov, and Alexander Gammerman. Criteria of efficiency for conformal prediction. In *Symposium on Conformal and Probabilistic Prediction with Applications*, pages 23–39. Springer, 2016.

Appendix A. Results from Experiment 1, 2 and 3

A.1. Results for Spambase dataset

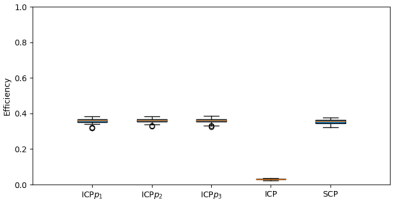


Figure 4: Linear SVM

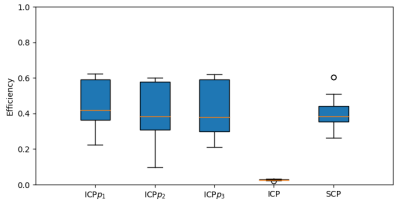


Figure 5: RBF-SVM

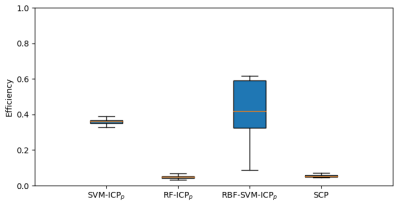


Figure 6: Different ML Models

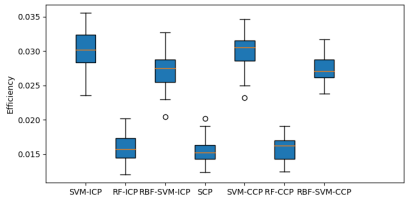


Figure 7: Unpartitioned data

A.2. Results for Breast Cancer Wisconsin dataset

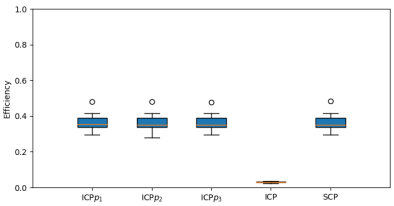


Figure 8: Linear SVM

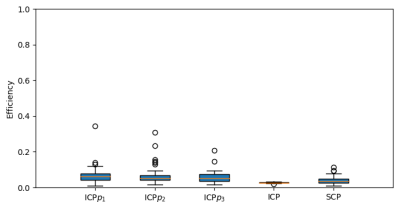


Figure 9: RBF-SVM

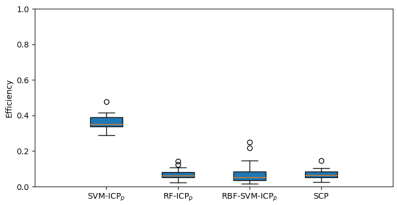


Figure 10: Different ML Models

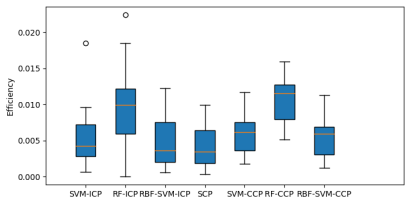


Figure 11: Unpartitioned data

A.3. Results for Phishing Websites dataset

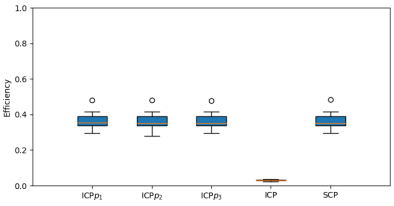


Figure 12: Linear SVM

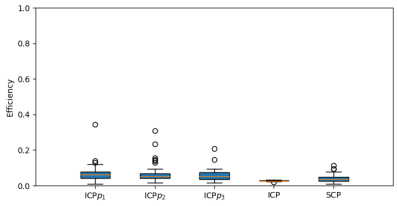


Figure 13: RBF-SVM

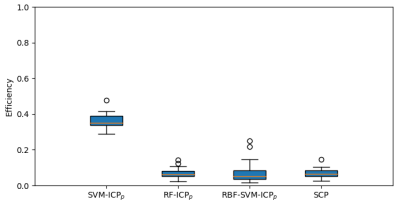


Figure 14: Different ML Models

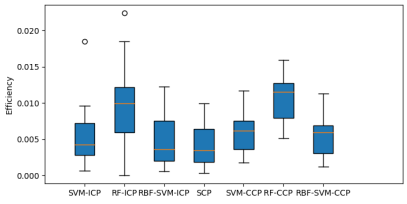


Figure 15: Unpartitioned data

A.4. Results for Covertype dataset

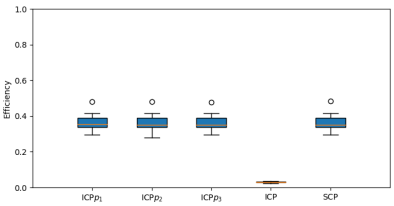


Figure 16: Linear SVM

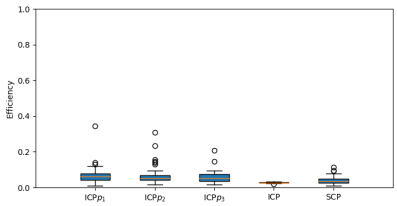


Figure 17: RBF-SVM

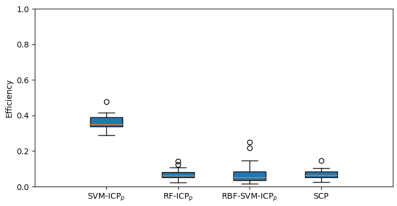


Figure 18: Different ML Models

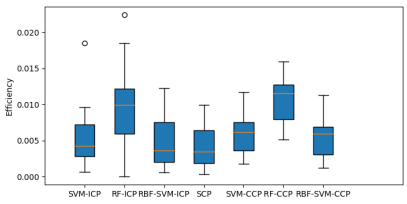


Figure 19: Unpartitioned data

A.5. Results for Adult dataset

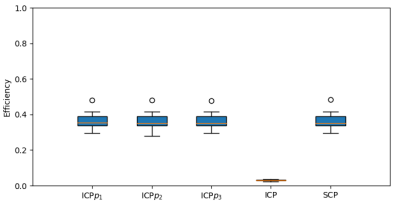


Figure 20: Linear SVM

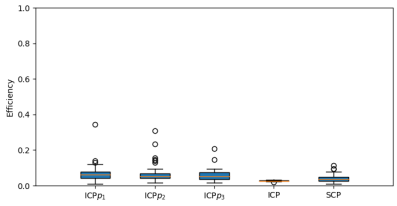


Figure 21: RBF-SVM

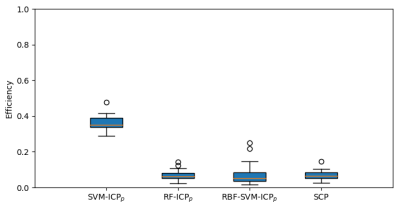


Figure 22: Different ML Models

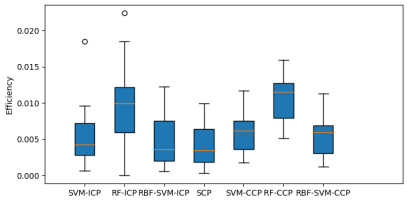


Figure 23: Unpartitioned data

A.6. Results for Tic-tac-toe dataset

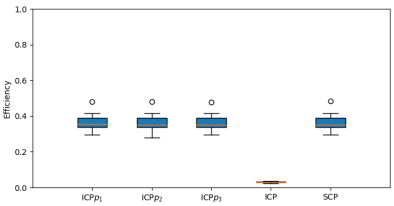


Figure 24: Linear SVM

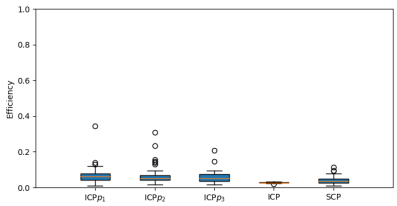


Figure 25: RBF-SVM

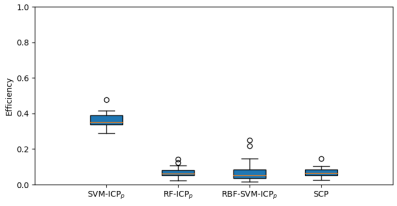


Figure 26: Different ML Models

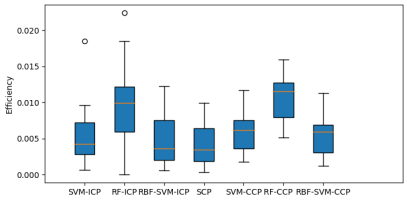


Figure 27: Unpartitioned data

A.7. Results for Australian dataset

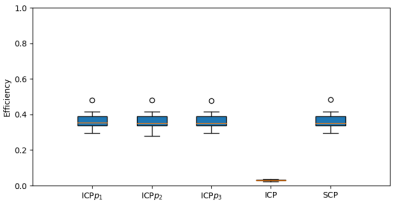


Figure 28: Linear SVM

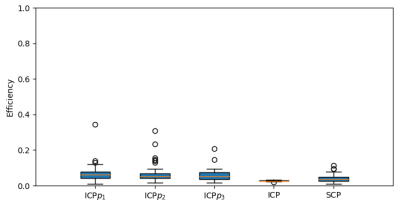


Figure 29: RBF-SVM

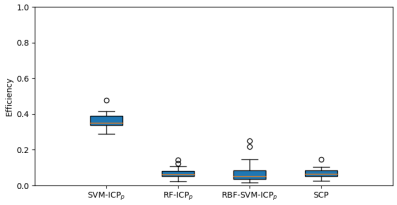


Figure 30: Different ML Models

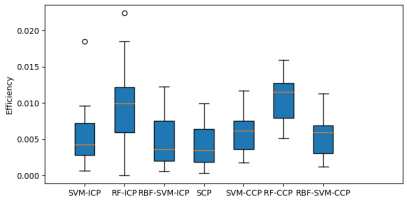


Figure 31: Unpartitioned data

A.8. Results for MONKs-1 dataset

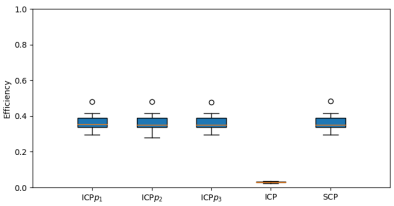


Figure 32: Linear SVM

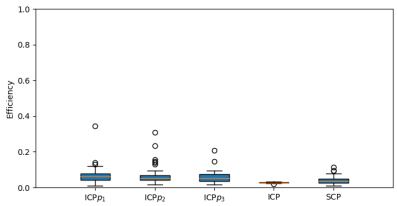


Figure 33: RBF-SVM

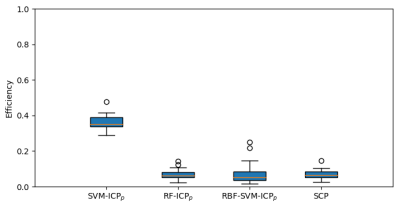


Figure 34: Different ML Models

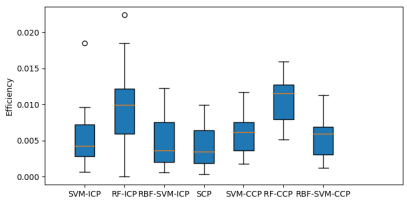


Figure 35: Unpartitioned data

A.9. Results for MONKs-2 dataset

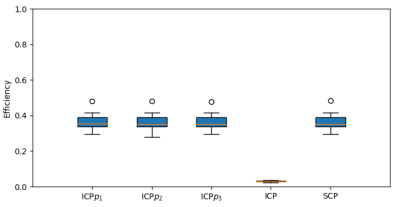


Figure 36: Linear SVM

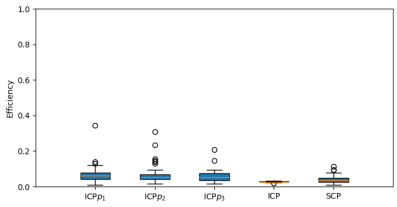


Figure 37: RBF-SVM

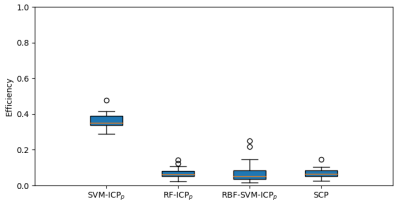


Figure 38: Different ML Models

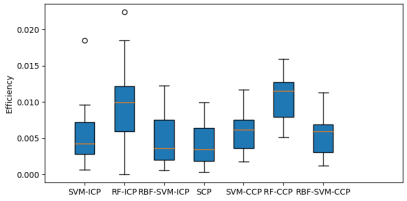


Figure 39: Unpartitioned data

A.10. Results for Bank dataset

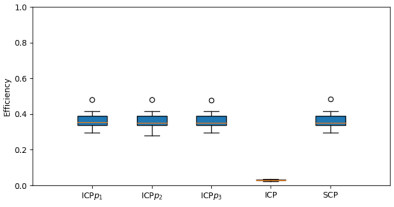


Figure 40: Linear SVM

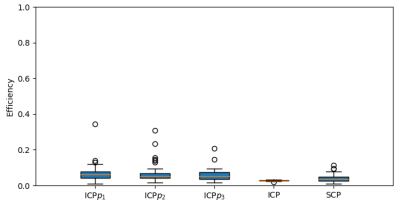


Figure 41: RBF-SVM

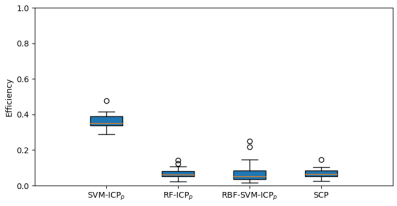


Figure 42: Different ML Models

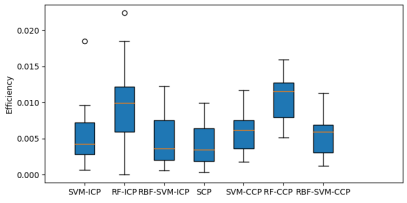


Figure 43: Unpartitioned data