Aggregating Predictions on Multiple Non-disclosed Datasets using Conformal Prediction

Ola Spjuth[a], Lars Carlsson[b], Niharika Gauraha[a]

[a] *Uppsala University*
[b] *AstraZeneca*

Abstract

The flexible framework for machine learning algorithms called conformal prediction, provides region predictions with guaranteed confidence under mild conditions. In this paper, we extend the basic conformal prediction framework to handle multiple data sources that do not require sharing of data. We propose to aggregate conformal predictions from multiple sources, where transductive conformal predictors are applied on the multiple data sources and their individual predictions are aggregated to form a single prediction on a new example. We illustrate the method using simulated and real data sets, and we show that the proposed method produces much more efficient predictions than individual analsys.

Keywords:
Conformal Predition, TCP, ACP

1. Introduction

In the biopharmaceutical sciences, it is not unusual for an experiment to be replicated by different manufacturing groups. However, the pooling (or sharing) of experimental data across various manufacturing groups are not encouraged. Also data security is one of the main concerns that has given rise to DataSHIELD aproaches (secure analyses that do not require sharing of data). In this article we propose to combine results across experiments without sharing the data, by aggregating conformal preditions computed at individual source level. In particular, we propose to combine conformal p-values from multiple data sources using weighted aggregation or fisher's method.

Basically, conformal predictors are confidence predictors, that results prediction sets for all confidence levels. Thus, conformal prediction is a framework that complements the predictions of machine learning algorithms with reliable measures of confidence. Transductive version of conformal predictors have been proven to be valid and more information efficient. In this paper, we extend the basic conformal prediction framework to handle multiple data sources and without sharing of data between sources. We propose to aggregate conformal predictions from multiple sources, where transductive conformal predictors are applied on the multiple data sources and their individual predictions are aggregated to form a single prediction on a new example.

The advantages of this approach of combining conformal predictions across multiple sources are two fold. Firstly, it is more a framework than a method, and it extends the existing framework of conformal prediction for multiple data sources, that do not require sharing of data. Secondly, combined analysis produces much more efficient predictions than individual analsys. This innovative framework is flexible in the sense it supports flexible number and sizes of data sources.

At a high level, our algorithm works as follows. Consider a binary classification problem, and suppose we have a training dataset Z and an external test data x. The training data set is randomly and unequally split into K parts. For example, Let $Z = \{z_1, ..., z_n\}$ be the data set of n observations, then we divide the dataset into $S_1, ..., S_K$ such that $Z = \bigcup_{i=1}^{K} S_i$, and $n = k_1 + ... + k_K$, where $k_i = |S_i|$. We compute p-values under TCP framework using the combined dataset (S_i, x) for each S_i. We have K p-values (for each class), now we aggregate (weighted) the k, p-values to obtain a final p-value for each class for the new example x. We repeat the process say q times by varying number and sizes of the sources. The final analysis consists of analysing q results which accounts for number as well as size of the data sources.

The organization of the paper is as follows. In section 2, we introduce the background concepts and notation, used throughout the paper. In Section 3, we will introduce the concept of aggregating conformal predictions from multiple sources. In Section 4, we discuss the statstical properties of agregated conformal predictions from multiple sources. In Section 5, we perform some numerical analsys on simulated and real datasets. Finally, in Section 6, the summary of the papery is provided. We have also included an appendix that reviews the most relevant aspects about TCP, ICP, CCP and ACP.

2. Notations and Background

2.1. Set Partition

A partion of a set $S = \{1, ..., n\}$ is a family of subsets $S_1, ..., S_k \subset S$, satisfying the following.

1. $S_i \cap S_j = \emptyset$, if $i \neq j$
2. $S = \bigcup_{i=1}^{k} S_i$
3. $S_i \neq \emptyset$, for $i = 1, .., k$

2.2. Transductive Conformity Prediction (TCP)

The object space $\mathcal{X} \in \mathbb{R}^p$, where p is the number of features, and label space $\mathcal{Y} \in (0, 1)$, and example space $\mathcal{Z} := \mathcal{X} \times \mathcal{Y}$ are fixed throughout the article. We assume that each observation consists of an object and its label.

The typical classification problem is, given a training dataset $Z = \{z_1, ..., z_n\}$ – where n is the number of observations in the training set, and each observation $z_i = (x_i, y_i)$ are labeled observations– we want to predict the label of a new observation x whose label is unknown.

First, we define a transductive non-conformity measure and transductive conformity score in the following.

Definition 1 (Transductive nonconformity measure)**.** *A transductive nonconformity measure is a measurable function $\mathcal{A} : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$ such that $\mathcal{A}(Z_1, Z_2)$ does not depend on the ordering of observations in the set Z_1.*

Definition 2 (Transductive nonconformity score)**.** *The transductive nonconformity score measures the lack of conformity of the "test observation" Z_2 to the "training set" Z_1.*

Definition 3 (Transductive Conformity Prediction (TCP))**.** *Given a training dataset Z and a new observation x, the transductive conformal predictor (TCP), corresponding to a nonconformity measure $\mathcal{A}$, checks each of a set of hypothesis (for all possible labels) for the new observation x, and assigns it a p-value at a significance level $\epsilon \in (0, 1)$.*

Algorithm 1: TCP

Input: (training dataset:Z, test data:x, label set:Y, a nonconformity measure:$\mathcal{A}$)
Output: p-values
for *each* $y \in \mathcal{Y}$ **do**
  $z_{n+1} = (x, y)$;
  $Z^* = (Z, z_{n+1})$;
  Compute the transductive nonconformity scores:
  $\alpha_i = \mathcal{A}(Z^*, z_i)$ for each $z_i \in Z^*$;

  Compute p-value: $p(y) = \frac{|i=\{1,..,n+1\}|\alpha_i \geq \alpha_{n+1}|}{n}$;
end
p-values $= \{p(y)|y \in \mathcal{Y}\}$;
return p-values;

3. Multi source aggregated TCP Algorithm

Let us consider a binary classification problem, and suppose we have a training dataset Z and external test data set X, or we randomly partition the given dataset into training (80%) and external test set (20%). The algorithm for aggregated TCP from multiple sources is as follows (see Figure 1).

1. The training data set is randomly split into K parts (disjointly) with varying sizes. For example, Let $Z = \{z_1, ..., z_n\}$ be the data set, then we divide the dataset into $S_1, ..., S_K$ such that $Z = \bigcup_{i=1}^{K} S_i$, $k_i = |S_i|$ and $n = k_1 + ... + k_K$.
2. We compute p-values using (X, S_i) for each S_i, say p_i for each class, then we finally aggregate the k, p-values (weighted average).
3. We repeat the step 1 and step 2 with different values of K and $k_i's$ say for q times.
4. Then we analyze the q results obtained (this part is not clear yet).

Algorithm 2: Multi source aggregated TCP

Input: (training dataset:Z, test dataset:X, label set:Y, a nonconformity measure:$\mathcal{A}$)

Output: Aggregated p-values

Initialization;

Unequal size partition: Partition training.data randomly and unequally into K parts, $S_1, ..., S_K$;

Steps;

for *each* S_i, $i \in \{1, ..., K\}$ **do**

  for *each observation* $x_j \in X$ **do**

    Compute p-values by using **TCP** algorithm:

    $PValues_i = \mathbf{TCP}(S_i, x_i, Y, \mathcal{A})$;

  end

end

Aggregate $PValues_i$ from various sources into a set **p-values**

return p-values

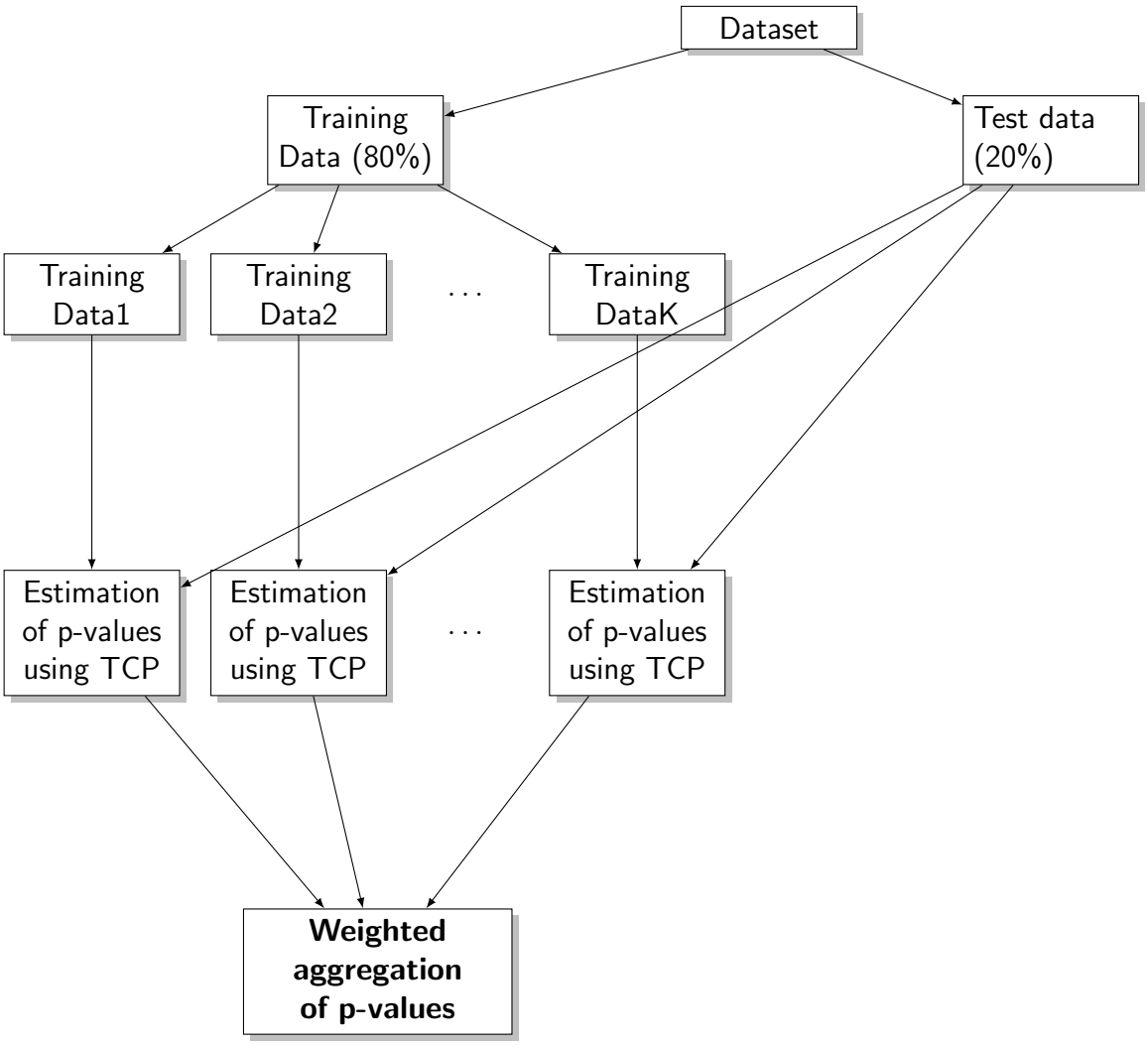

Figure 1: Multi source aggregated TCP Algorithm