Erasmus University Rotterdam
Erasmus School of Economics
Bachelor Thesis International Bachelor Econometrics and Operations Research

Nonexchangeable and Pseudoexchangeable Conformal Predictions through Random Localised Robust Weighting

Igor Pradhan (598064)

Supervisor:	Sam van Meer, MSc
Second assessor:	Dr. Jesse Hemerik
Date final version:	1st July 2024

The views stated in this thesis are those of the author and not necessarily those of the supervisor, second assessor, Erasmus School of Economics or Erasmus University Rotterdam.

Abstract

This paper aims to generalise the weight selection for nonexchangeable conformal prediction in non-symmetric algorithms. It also extends this notion to a pseudoexchangeable dataset. Conformal Prediction (CP) is a widely used framework that constructs confidence intervals for predictions which are crucial for various applications. Traditional CP methods assume that data points are independent and identically distributed (i.i.d) or exchangeable, which often does not hold in practice due to distributional drifts or correlations. To address this, Barber, Candes, Ramdas, and Tibshirani (2023) provided theoretical coverage guarantees for nonexchangeable data, which rely heavily on the chosen weights. Incorrect weight specification can result in larger coverage gaps and less informative confidence intervals. This paper builds upon these foundations by proposing methods for more robust weight selection, specifically through random localised robust weighting techniques from Hore and Barber (2023). These methods aim to minimise the coverage gap and ensure reliable prediction intervals even in nonexchangeable and pseudoexchangeable settings. Our simulation experiments indicate that while this method does not outperform existing techniques in scenarios involving distributional drift, it performs comparably in real-world datasets.

1 Introduction

A fundamental challenge with current state-of-the-art machine learning models is finding a confidence interval that guarantees a desired coverage level. Confidence intervals are essential in many real-world applications, such as medical diagnosis, financial forecasting, and autonomous driving, where understanding the uncertainty of predictions can significantly impact decision-making processes.

A common approach to achieve these coverage levels with "black box" models involves splitting the dataset into training and test sets. Specifically, the data is partitioned as $Z = Z_{\text{train}} \cup Z_{\text{test}}$, with only Z_{train} used to train the model $\hat{\mu}$ and try to most accurately predict Z_{test}. An increasingly popular framework for this purpose is Conformal Prediction (CP). Given a new input X_{n+1}, the task is to construct a confidence interval (CI) $\hat{C}_n(X_{n+1})$ around the unobserved Y_{n+1}. The seminal work on conformal prediction was introduced by Vovk, Gammerman, and Shafer (2005), where they demonstrated that if the dataset is exchangeable, i.e. $(Z_1, Z_2, \ldots, Z_n) \overset{\Delta}{=} (Z_{\sigma(1)}, Z_{\sigma(2)}, \ldots Z_{\sigma(n)})$ for any permutation σ, then

$$\mathbb{P}(Y_{n+1} \in \hat{C}_n(X_{n+1})) \geqslant 1 - \alpha,$$

for a desired coverage of $1 - \alpha \in (0, 1)$. A crucial assumption for this result is that the data must be exchangeable, and of course this result also holds for i.i.d. data. However, this assumption is often violated in practice due to potential distributional drift or correlation between data points. Barber et al. (2023) were the first to show theoretical coverage guarantees for nonexchangeable data.

A significant metric in evaluating the effectiveness of a prediction set is the *coverage gap*, defined as:

$$\text{Coverage Gap} = (1 - \alpha) - \mathbb{P}(Y_{n+1} \in \hat{C}_n(X_{n+1})).$$

Minimizing the absolute value of this metric is a compelling objective. However, it may not be the most meaningful on its own, as a trivial solution could involve setting the prediction set to $(-\infty, +\infty)$ $(1-\alpha) \times 100\%$ of the time and to $\emptyset$ the remaining $\alpha \times 100\%$ of the time. To enhance the utility of this metric, it is important to also consider the average interval width. By incorporating this additional measure, the metric becomes more informative, as a smaller interval width indicates more efficient and practical outcomes.

Barber et al. (2023) managed to show various upper bounds of the coverag gap for different cases of data types, providing a more practical framework for ensuring reliable prediction intervals in real-world applications.

The motivation for exploring this problem is straightforward. While Conformal Prediction (CP) has gained popularity over the years, there are still gaps to explore. Barber et al. (2023) were the first to show theoretical coverage guarantees for nonexchangeable data, but this heavily relies on the chosen weights. If the weights are misspecified, the coverage gap might be larger than desired, leading to overly broad confidence intervals that provide little useful information.

The key issue is how to choose these "weights", which represent the relationship between two covariates. In time-series data, an autoregressive weight might be appropriate, but selecting weights can be challenging when no clear solution exists. Hore and Barber (2023) found robust weights for localised CP but have not applied them to nonexchangeable data.

In this paper, we will adapt the method from Hore and Barber (2023) on how to find robust weights, and apply a nonexchangeable CP algorithm on top of it. We will then compare the coverage gaps across different datasets to evaluate the performance of these robust weights. As previously mentioned, the weights chosen have a large impact on the width of the CI, and thus also the coverage gap. Moreover, we also want to see how we can adjust the algorithms on a pseudoexchangeable dataset, and will show that the weights used will provide sufficient coverage levels.

Our investigation reveals that the proposed method performs optimally in scenarios where the underlying distribution of the data is unknown. In cases of a clear distributional drift, the use of autoregressive weights significantly outperforms our proposed method. Conversely, when simulating spatial data, our method demonstrates a marginally superior performance. For real-world datasets, our method exhibits the best performance within pseudoexchangeable settings. However, we acknowledge that the evidence supporting this conclusion is not sufficiently robust to make a definitive claim.

Section 2 provides a brief summary of the literature concerning conformal prediction. Section 3 describes the methodology and models that we will use to demonstrate our main results. Section 4 outlines the datasets used in our experiments. Section 5 presents the results of the proposed algorithms. Finally, Section 6 contains a discussion and conclusion of this paper.

2 Literature Review

The groundbreaking work by Vovk et al. (2005) is widely recognized as foundational in the field of Conformal Prediction (CP). Vovk introduced both split and full CP methods, but the traditional approach for full CP, which involves retraining the model for each hypothesized point y, can be computationally intensive. To address this, Fong and Holmes (2021) proposed a Bayesian conformal computation approach that significantly reduces computation time by reweighting the posterior distribution instead of retraining the model entirely.

However, determining the correct weights to minimize the coverage gap can be challenging. Stanton, Maddox, and Wilson (2023) suggest using a Bayesian optimization approach to direct the confidence interval to the correct location, demonstrating that this method can lead to more efficient confidence intervals. Both of these approaches, however, rely on the assumption that the data is exchangeable. In contrast, Barber et al. (2023) offers an algorithm that ensures full coverage regardless of whether the data is exchangeable, given the correct selection of weights.

A comprehensive review of recent advancements in conformal prediction is provided by Fontana, Zeni, and Vantini (2023). An interesting approach mentioned is the Mondrian Conformal Prediction. For example, if we want to create confidence intervals for predictions on different groups in a specific category, such as men and women, and we have a target coverage level of 5%, but we observe an error of 10% for men and 0% for women, then calling this a 5% miscoverage due to the imbalance might be misleading. This issue was first addressed by Vovk et al. (2005) and further extended by Fontana et al. (2023). To tackle this issue, they set the conditional coverage to the target miscoverage level, i.e., the coverage conditional on the next feature being in a certain group should be at least $(1-\alpha) \times 100\%$.

Candès, Lei, and Ren (2023) explore the application of conformal prediction to survival analysis, an area of significant practical importance. Their work extends the conformal prediction framework to handle censored data, which is a common feature in survival analysis. By leveraging the conformal prediction method, they provide valid prediction intervals for survival times, ensuring that the intervals maintain the desired coverage probability despite the presence of censored data. This extension is particularly valuable in medical research and other fields where time-to-event data is prevalent.

Similarly, Lei, Rinaldo, and Wasserman (2015) extend the conformal prediction methodology to functional data, which is data that can be seen as functions rather than scalar values or vectors. Their paper addresses the challenges of applying conformal prediction to data types like time series or curves. By introducing methods to handle the intrinsic complexities of functional data, they provide a robust framework for generating prediction bands that maintain valid coverage rates. This extension broadens the applicability of conformal prediction to a wider array of data types encountered in various scientific disciplines.

Table 1 provides a selection of papers that have developed conformal prediction methods, as well as an overview of our paper as a comparison. Note that this overview is non-exhaustive.

Authors	**Purpose**	**Source Type**	**Summary/Contributions**
Vovk et al. (2005)	Seminal work on conformal prediction. Provides first algorithm on CP.	Book	The proposed method is applied to create prediction sets for a desired outcome Y_{n+1}. They provide various algorithms to do this and minimize complexity over specific prediction algorithms.

Barber et al. (2023)	Proposes a general framework for using conformal prediction on nonexchangeable data	Journal article	The proposed framework extends the original paper by Vovk et al. (2005) and ensures coverage levels for nonexchangeable data. The key results were that weighted quantiles ensure more robustness and are essential to minimize the coverage gap.
Hore and Barber (2023)	Extends the notion of localized conformal prediction by sampling a covariate near the point of interest	Journal article	This paper extends the ideas of Guan (2023). They argue that finding local coverage is theoretically impossible and wish to relax the constraint by sampling a covariate near X_{n+1}, which can ensure better upper bounds for the theoretical coverage for *exchangeable data.* They achieve this by creating a similarity measure, otherwise known as kernels.
Fong and Holmes (2021)	Explores the idea of Bayesian computation for conformal prediction	Journal article	This paper integrates conformal prediction with Bayesian inference to enhance predictive uncertainty estimates. They combine Bayesian posterior predictive checks with conformal scores, ensuring valid coverage without strong distributional assumptions, thereby improving reliability and flexibility in uncertainty quantification for various applications.
Our Paper	Attempts to generalise weight selection for the algorithms proposed by Barber et al. (2023) and adapt it to a pseudoexchangeable setting	Bachelor's Thesis	Applies the methods of Hore and Barber (2023) on random localised predictions to Barber et al. (2023) algorithms for nonexchangeable and pseudoexchangeable data. Finds that it works best when the data is spatially sampled.

Table 1: A selection of papers that develop conformal prediction

3 Methodology

3.1 Full conformal prediction

We will begin by defining some notation. Let $Z_{1:n} = \{(X_i, Y_i)\}_{i \in [n]}$, where Y_i is our outcome of interest, and X_i the row of covariates and $[n] := \{1, 2, \dots, n\}$. Moreover, let $\mathcal{A}$ denote an algorithm that treats all datapoints symmetrically. i.e.:

$$\mathcal{A}\left((X_{\sigma(1)}, Y_{\sigma(1)}), \dots, (X_{\sigma(n)}, Y_{\sigma(n)})\right) = \mathcal{A}\big((X_1, Y_1), \dots, (X_n, Y_n)\big)$$

For any permutation $\sigma \in S_n$. Next, $\forall y \in \mathbb{R}$ we denote the function trained with the pair (X_{n+1}, y) as:

$$\hat{\mu}^y = \mathcal{A}\big((X_1, Y_1, \ldots, (X_n, Y_n), (X_{n+1}, y)\big)$$

In other words, $\hat{\mu}^y$ is the model trained with the hypothesised test point y. In practice looking over the entire space $\mathbb{R}$ is infeasible, so we look at a fine grid $y \in \mathcal{Y}_{cand}$. For some special cases a closed form solution is possible, as an example, in Vovk et al. (2005) provides a closed form solution for the ridge regression case. Moreover, let

$$R_i^y = \begin{cases} |Y_i - \hat{\mu}^y(X_i)| & \text{if } i \in [n] \\ |y - \hat{\mu}^y(X_{n+1})| & \text{if } i = n+1 \end{cases}$$

denote the prediction error of model $\hat{\mu}^y$.

The full conformal prediction proposed by Vovk et al. (2005) relies on a relatively strong assumption, that the data must be exchangeable. Exchangeability defines that the joint of a tuple of random variables is invariant to permutations. Mathematically this states

$$(X_1, X_2, X_3, \ldots, X_n) \triangleq (X_{\sigma(1)}, X_{\sigma(2)}, X_{\sigma(3)}, \ldots, X_{\sigma(n)}), \quad \forall \sigma \in S_n.$$

Theorem 1 (Vovk) If the sequence $Z_{1:n} = \{(X_i, Y_i)\}_{i:n}$ is *i.i.d.*, or exchangeable and if the algorithm $\mathcal{A}$ treats all input data (X_i, Y_i) symmetrically that the following lower bound holds:

$$\mathbb{P}(Y_{n+1} \in \hat{C}_n(X_{n+1})) \geqslant 1 - \alpha,$$

In words, the probability that our outcome of interest falls within the confidence interval is guaranteed to fall within our desired coverage level. A proof can be found in appendix B.1.

3.2 Nonexchangeable Conformal Prediction

Nonexchangeable conformal prediction has been described by Barber et al. (2023), where they use weighted quantiles for robust CP. Weights $w_i \in [0, 1]$ that are closer to 1 argue that Z_i might come from the same distribution as Z_{n+1}. The new prediction set becomes:

$$\hat{C}_n(X_{n+1}) = \left\{ y : R_{n+1}^y \leqslant Q_{1-\alpha}\left(\sum_{i=1}^{n} \tilde{w}_i \cdot \delta_{R_i} + \tilde{w}_{n+1} \cdot \delta_{+\infty} \right) \right\},$$

where

$$\tilde{w}_i = \frac{w_i}{1 + \sum_{1 \leqslant j \leqslant n} w_j}, \quad i \in [n] \qquad \tilde{w}_{n+1} = \frac{1}{1 + \sum_{1 \leqslant j \leqslant n} w_j}$$

are the normalised weights that accommodate the dataset, and δ_a denotes the point mass at point a. As mentioned previously, we assign higher weights if we hypothesise that a point might come from the same (or similar) distribution as the test point. An example might be if we know that there is a distribution drift we pick $w_1 \leqslant w_2 \leqslant \cdots \leqslant w_n$, or we assume the weights to be spatially sampled, we might pick weights based based on a certain distance metric, more on this later.

Nonexchangeable Conformal Prediction with a non-symmetric algorithm To accommodate the non-exchangeable data we must adjust our $\mathcal{A}$ function. To adjust for this difference let us redefine

$$\mathcal{A} : \bigcup_{n \geqslant 0} (\mathcal{X} \times \mathbb{R} \times \mathcal{T})^n \to \{\text{measurable functions } \hat{\mu} : \mathcal{X} \to \mathbb{R}\}.$$

Instead of the pair (X_i, Y_i) we get a triplet (X_i, Y_i, t_i). This t_i term stands for the "tagged" term, this tag can provide various different information, such as the spatial location from where i has been sampled or the weight that we assign point i. The algorithm for full conformal prediction with a non-symmetric algorithm is similar to the one with a symmetric algorithm. We next define our function as

$$\hat{\mu}^{y,k} = \mathcal{A}\big((X_{\sigma_k(i)}, Y^y_{\sigma_k(i)}, t_i) : i \in [n+1]\big) \quad \forall y \in \mathcal{Y}_{cand}.$$

Here σ_k denotes a permutation of only indices k and $n+1$, and where

$$Y_i^y = Y_i \cdot \mathbb{1}\{i \in [n]\} + y \cdot \mathbb{1}\{i = n+1\}.$$

This swap is necessary to establish the coverage guarantees that Barber et al. (2023) further derive. The most important result is

$$\text{Coverage gap} \leqslant \frac{\sum_{i=1}^{n} d_{\mathrm{TV}}(Z, Z^i)}{n+1},$$

where d_{TV} is the statistical total variation distance defined by $d_{\mathrm{TV}} = \sup_{A \in \mathcal{F}} |\mathcal{G}(A) - \mathcal{Q}(A)|$, for some probability space $(\Omega, \mathcal{F})$ and distinct probability measures $\mathcal{G}$ and $\mathcal{Q}$.

3.3 Localised weights

The method of localised conformal prediction was first proposed by Guan (2023) where they emphasise the local region around the test point. This method assumes that data is spatially distributed, whereas Barber et al. (2023) "guess" the distribution of the weights. The suggestion is to use some localised kernel

$$H : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_{\geqslant 0}$$

The localiser function called to itself must be equal to 1, i.e $H(x, x) = 1, \forall x \in \mathcal{X}$.A common kernel for $\mathcal{X} = \mathbb{R}$ is to use the Gaussian kernel

$$H(x, x') = \exp\left(\frac{-\|x - x'\|_2^2}{2h^2}\right).$$

Or the box kernel

$$H(x, x') = \mathbb{1}\{||x - x'||_2 \leqslant h\},$$

for some bandwidth h. We can imagine these kernels as similarity measures. As we weigh observations Z_i more if the covariate is sampled near X_{n+1}. The local quantile is described as follows

$$\hat{q}(X_{n+1})_{1-\alpha} = Q_{1-\alpha}(\sum_{i=1}^{n} w_i \cdot \delta_{R_i} + w_{n+1} \cdot \delta_{+\infty})$$

with the weights now computed as

$$w_i = \frac{H(X_i, X_{n+1})}{1 + \sum_{j=1}^{n} H(X_j, X_{n+1})}.$$

As mentioned in subsection 3.2 we mention that we might use some metric distance for our weighting if we hypothesise that our data is spatially sampled. We conjecture that if we have no prior knowledge on the data distribution or how they are sampled that this is a better way of choosing weights. As localised weights can be seen as a similarity score, if covariates lie close to the covariate of the test point, we would like to weigh these test points more than others.

3.4 Randomly-Localised Conformal Prediction

Hore and Barber (2023) have further developed localised conformal prediction by choosing a point $\tilde{X}_{n+1}$ close to X_{n+1}. By choosing this point, they find that the weights determined by the kernel will lead to more robust weights. The method is similar to the base localised conformal prediction with the added assumption that $H(x, \cdot)$ is a density with respect to some measure ν and space $\mathcal{X}$, i.e.

$$\int_{\mathcal{X}} H(x, x') d\nu(x') = 1 \text{ for all } x \in \mathcal{X}.$$

If we consider ν to be the Lebesgue measure and set $\mathcal{X} \subseteq \mathbb{R}^d$ we can assume the previous kernels with a normalising constant. For the Gaussian kernel with bandwidth h:

$$H(x, x') = \frac{1}{(2\pi h^2)^{\frac{d}{2}}} \exp\left(- \frac{\|x - x'\|_2^2}{2h^2} \right),$$

and for the box kernel including the normalising constant:

$$H(x, x') = \frac{1}{V_d h^d} \mathbb{1}\{\left\|x - x'\right\|_2 \leqslant h\},$$

where V_d is the volume of the unit ball in $\mathbb{R}^d$. Here both functions are symmetric, however this assumption is not assumed. With the function H chosen we can define the randomised localised prediction weights as follows: given the prediction score $R_i^{y,k}$ (so including the permutation), and the test point X_{n+1}, we sample $\tilde{X}_{n+1}$ from the density H and return the prediction set

$$\tilde{C}_n^{RLCP} = \left\{ y \in \mathcal{Y}_{cand} : R_i^{y,k} \leqslant Q_{1-\alpha}(\sum_{1 \leqslant i \leqslant n} \tilde{w}_i \delta_{R_i^{y,k}} + \tilde{w}_{n+1} \delta_{+\infty}) \right\}.$$

Where the weights are:

$$\tilde{w}_i = \frac{H(X_i, \tilde{X}_{n+1})}{\sum_{j=1}^{n+1} H(X_j, \tilde{X}_{n+1})}, \quad \forall i \in [n+1].$$

The most important result of Hore and Barber (2023) is the following theorem:

Theorem 2: Coverage guarantees for RLCP Hore and Barber (2023)
If $(X_i, Y_i) \overset{i.i.d}{\sim} P_X \times P_{Y|X}$ and $B \subseteq \mathcal{X}$ then the following lower bound holds:

$$\mathbb{P}\Big(Y_{n+1} \in \tilde{C}_{n+1}^{RLCP} | X_{n+1} \in B\Big) \geqslant 1 - \alpha - \frac{\inf_{\epsilon > 0} \Big\{ \delta_{(X,\tilde{X})} \{ \Big\| X - \tilde{X} \Big\| > \epsilon \} + \delta_X(bd_{2\epsilon}(B)) \Big\}}{\delta_X(B)},$$

with $bd_r(B) = \{x \in B : \inf_{x' \in B^c} \|x - x'\| \leqslant r\}$ and again $\delta_X(\cdot)$ being the probability mass at the set $(\cdot)$. A complete proof can be found in Hore and Barber (2023).

We can see $\tilde{X}_{n+1}$ as a "synthetic prototype" which is drawn from $H \circ P_X$ where P_X is the distribution of the original feature space. We then centre the kernel around $\tilde{X}_{n+1}$ that is drawn near X_{n+1}, this will also lead to the coverage of X_{n+1}. Further derivations on coverage can be found in Hore and Barber (2023). Furthermore, for the kernels mentioned earlier (as well as other reasonable choices for the kernels), the bandwidth parameter h determines the degree of localisation. Generally, when h is large, the method functions similarly to split conformal, but as h decreases, the method becomes more localised, potentially improving local empirical coverage. However, if h is too small, performance declines due to a reduced effective sample size, and the quantile $q_b(x)$ might even be estimated as $+\infty$.

3.5 Optimal bandwidth selection

As mentioned in subsection 3.5, selecting the optimal bandwidth parameter can be a challenging task. Since we do not know the true probability density function (PDF) of the weights, we estimate it using kernel density functions. The estimated PDF is given by:

$$\hat{f}(x) = \frac{1}{Nh} \sum_{i \in [n]} H(x, x_i),$$

where $H(x, x_i)$ represents the kernel function. The accuracy of this kernel estimation is typically determined using different distance metrics, such as the $\mathcal{L}^1$ norm based integrated absolute measure, the $\mathcal{L}^2$ norm based mean integrated squared error metric (MISE) or the Kullback-Liebler divergence. The optimal bandwidth parameter h is then obtained by minimising the chosen distance measure. One of the most commonly used measures is MISE, also known as IMSE, which was detailed by Silverman (2018). The accuracy of the estimated PDF, using MISE, can be quantified as follows:

$$
\begin{aligned}
MISE(\hat{f}(x)) = \mathbb{E}(ISE(\hat{f}(x))) &= \mathbb{E}\int_\Omega (\hat{f}(x) - f(x))^2 dx \\
&\overset{(*)}{=} \int_\Omega \text{Bias}^2(f(x))dx + \int_\Omega \text{Var}(f(x))dx \\
&= \frac{h^4}{4}(\mu_2(H))^2 R(f'') + \frac{1}{Nh}R(H) + \mathcal{O}(h^4) + \mathcal{O}\left(\frac{h}{N}\right)
\end{aligned}
$$

where $\mu_2(H) = \int_\Omega z^2 H(z)dz$, $R(f'') = \int_\Omega (f''(z))^2 dz$, and $R(H) = \int_\Omega (H(z))^2 dz$ and $(*)$ holds from Fubini's theorem. In an asymptotically large sample N, we can use the approximated mean squared integrated error (AMISE), under the assumption that $\lim_{N\to\infty} h = 0$ and $\lim_{N\to\infty} Nh = \infty$, i.e., h converges to zero at a slower rate than $1/N$.

$$
AMISE = \frac{h^4}{4}(\mu_2(H))^2 R(f'') + \frac{1}{Nh}R(H)
$$

Minimising with respect to h yields:

$$
h_{\text{AMISE}} = \left(\frac{R(H)}{\mu_2(H)R(f")N}\right)^{1/5}.
$$

If we select our kernel H to be the Gaussian kernel, the optimal bandwidth parameter is equal to $h_{\text{AMISE, Guassian}} = (\frac{4\hat{\sigma}^5}{3N}) \approx 1.06\hat{\sigma}N$.

3.6 Weights on Pseudoexchangeable data

Another interesting setting is when considering Pseudoexchangeability, first coined by Aldous (1981). Pseudoexchangeable data envelopes the idea that data might be exchangeable within subgroups. Consider a dataset of mathematics grades from various schools around a country. Within a school the students might be exchangeable, but two students from different schools might not. This can be more formally expressed in the following way. Lets say we observe J groups of data, with each having n_j observations, where again the observed data $Z_{i,j} = (Y_{i,j}, X_{i,j})$. The full dataset can be represented in the sequence $\{Z_{i,j}\}_{i\in[n_j], j\in[J]}$. With pseudoexchangeability a prediction algorithm should treat data within the same group symmetrically, i.e.

$$
\begin{aligned}
&\mathcal{A}\big((X_{1,1}, Y_{1,1}), \dots, (X_{k,i}, Y_{k,i}), \dots, (X_{J,n_J}, Y_{J,n_J})\big) \\
&= \mathcal{A}\left((X_{1,\sigma_1(1)}, Y_{1,\sigma_1(1)}), \dots, (X_{k,\sigma_k(i)}, Y_{k,\sigma_k(i)}), \dots, (X_{J,\sigma_J(n_J)}, Y_{J,\sigma_J(n_J)})\right),
\end{aligned}
$$

with $\sigma_j \in S_{n_j}, j \in [J]$. Recall in subsection 3.2 that we have an algorithm $\mathcal{A} : \bigcup_{n\geqslant 0}(\mathcal{X} \times \mathbb{R} \times \mathcal{T})^n \to \{\text{measurable functions } \hat{\mu} : \mathcal{X} \to \mathbb{R}\}$. where we tag each data point. To adjust for the pseudoexchangeable data we tag the the data from the same group with weight 1, i.e. if we want to find an appropriate prediction set for $Y_{j,k}$ we set $w_{j,i} = 1, \quad \forall i \in [n_j]$. Moreover, we want the weights from the same group to be equal as well. To accomodate for this, we first average the covariates from the same group $\bar{X}_j = \frac{1}{n_j}\sum_{i\in[n_j]} X_{j,i}$, If the next observed covariate belongs to group j, $X_{n+1} = X_{j,n_j+1}$. Next we set the weights from the same group equal to each other $w_{i,k} = w_{i,l} = H(\bar{X}_i, \tilde{X}_{n+1})$, where again $\tilde{X}_{n+1}$ is a sample drawn from density H with test point X_{n+1}. Then the normalised weights look as follows:

$$w_{i,k} = \frac{H(\bar{X}_i, \tilde{X}_{n+1})}{1 + n_1 H(\bar{X}_1, \tilde{X}_{n+1}) + \cdots + n_j + n_J H(\bar{X}_J, \tilde{X}_{n+1})}, \qquad \forall i \in [J] \setminus \{j\}, k \in [n_i],$$

$$w_{j,k} = w_{j,n_j+1} = \frac{1}{1 + n_1 H(\bar{X}_1, \tilde{X}_{n+1}) + \cdots + n_j + n_J H(\bar{X}_J, \tilde{X}_{n+1})},$$

and the prediction set for a pseudoexchangeable dataset becomes:

$$\tilde{C}_n^{\text{RLCP+PsEx}} = \left\{ y \in \mathcal{Y}_{\text{cand}} : R_{i,l}^{y,k} \leqslant Q_{1-\alpha}\Big(\sum_{i \in [J]} \sum_{l \in [n_i]} \tilde{w}_{i,l} \delta_{R_{i,l}^{y,k}} + \tilde{w}_{n+1} \delta_{+\infty} \Big) \right\}.$$

We conjecture that the coverage guarantees of Theorem 2 will also hold for the pseudoexchangeable setting. The condition of $B \subseteq \mathcal{X}$ can be interpreted that B must be contained within some group, $B \subseteq G_j \subset \mathcal{X}$, we can then easily find a subset that contains X_{n+1}.

4 Data

For this paper we would like to replicate the results obtained by Barber et al. (2023) for the simulation with a distribution drift and the **ELEC2** dataset. We want to see how the localised kernels will perform compared to the autoregressive parameter used by Barber et al. (2023), as they assume that there is prior knowledge on distribution drift, with us on the contrary that we do not assume any distribution and only will look at similarity.

4.1 Simulation Study

Setting 1: Distribution drift. We start off by generating $N = 2000$ points (X_i, Y_i) with $X_i \sim \mathcal{N}(0, I_4)$ and $Y_i | X_i = X_i^T \beta^{(i)} + \mathcal{N}(0, 1)$. Here $\beta^{(i)}$ is the coefficient vector, with $\beta^{(1)} = (2, 1, 0, 0)$ and $\beta^{(N)} = (0, 0, 2, 1)$ and every $\beta^{(i)}$ in between can be found by interpolation.

Additionally we will include another simulation study.

Setting 2: Spatial Simulation. We will first generate random variables dependent solely on the preceding random variable. Specifically, $X_i \perp\!\!\!\perp X_{j<i-1} | X_{i-1}$, and $X_i | X_{i-1} \sim \mathcal{N}(x_{i-1}, I_n)$ with $X_1 \sim \mathcal{N}(0, I_n)$. Since we want to generate Y that is spatially dependent on X we will incorporate the some distance metric to generate Y. We do not want Y_i do be too heavily dependent on Y_{i-1}, to account for this we select some permutation $\pi \in S_n$ and let $V_i = X_{\pi(i)}$. Finally we

generate Y in the following manner:

$$Y_0 = X_0^T \beta + \mathcal{N}(0,1)$$
$$Y_i = X_i^T \beta + \sum_{j=0}^{i-1} \frac{d(V_i, V_j)}{\sum_{k=0}^{i-1} d(V_i, V_k)} d(V_i, V_j) + \mathcal{N}(0,1)$$

Where $d(V_i, \cdot)$ to be the euclidean distance. We select $\beta = (2,2)$ and $n = 2$ for this experiment.

4.2 Real World Dataset

ELEC2 Dataset

The ELEC2 dataset from Harries, Nsw-cse tr, and Wales (2003) is a widely recognised benchmark dataset used extensively in the domain of data stream mining and machine learning, specifically for the evaluation of concept drift detection algorithms. This dataset encompasses electricity price data from New South Wales, Australia, collected over a period from 1996 to 1999. It provides a comprehensive set of features, including the day of the week, time of day, total electricity demand, and scheduled electricity supply, along with the corresponding electricity prices. The primary target variable in this dataset is the direction of the price change, which indicates whether the electricity price has increased or decreased relative to the previous time period.

The ELEC2 dataset is particularly valuable for studying concept drift because it reflects real-world scenarios where the underlying data distribution changes over time due to various factors. For instance, electricity prices are influenced by shifts in demand and supply dynamics, regulatory adjustments, and seasonal variations. These changes can cause the predictive models to degrade in performance if they do not adapt to the evolving patterns, making the dataset an ideal test bed for evaluating the robustness and adaptability of different algorithms.

Indonesia dataset

The dataset utilized in the study "Double for Nothing? Experimental Evidence on the Impact of an Unconditional Teacher Salary Increase on Student Performance in Indonesia" from De Ree, Muralidharan, Pradhan, and Rogers (2018) is derived from a comprehensive, large-scale randomized experiment aimed at evaluating the effects of a significant policy change in Indonesia. This policy involved a substantial increase in teacher salaries, providing a unique opportunity to assess its impact on various educational outcomes. The dataset is multifaceted, encompassing detailed information about teachers, students, and schools involved in the experiment.

The teacher data includes extensive details about salaries both before and after the increase. It tracks base salary changes to understand the financial implications for teachers and includes measures of job satisfaction, capturing teachers' sentiments about their income and overall job satisfaction post-raise. The dataset also documents the incidence of teachers holding secondary jobs, which offers insights into whether the salary increase reduced the necessity for supplementary income. Additionally, it includes self-reported levels of financial stress, providing a view into how the salary boost affected teachers' economic well-being and stress levels.

Student data focuses on performance metrics essential for evaluating the impact of the salary increase on educational outcomes. It includes standardised test scores and other academic performance indicators collected over several years, allowing for a longitudinal analysis of changes in student learning outcomes. The dataset also contains demographic information about the students, such as age, gender, and socioeconomic status, which helps control for potential confounding variables and ensures a more accurate analysis of the impact of teacher salary changes on student performance.

School data provides contextual information about the schools participating in the experiment. It details the schools' locations, sizes, and available resources, all of which can influence both teacher performance and student outcomes. The dataset also documents the randomisation process used to assign schools to either the treatment group, which received the salary increase, or the control group, which did not.

See appendix D to see summary statistics tables for both the ELEC2 data and the Indoensia dataset.

5 Results

We begin by replicating the results from Barber et al. (2023), using weights $w_i = \rho^{n+1-i}$ and targeting a coverage of $1 - \alpha = 0.9$.

Following Barber et al. (2023), we employ the following methods:

- **CP + LS:** Conformal prediction with a linear regression model.
- **nexCP + LS:** Nonexchangeable Conformal Prediction with a linear regression model.
- **nexCP + WLS:** Nonexchangeable Conformal Prediction with a weighted least squares model.

For the final proposed method, we consider two different weighting schemes: autoregressive weights and the Gaussian kernel for RLRW with bandwidth $h = 1.06\hat{\sigma}n^{1/5}$.

5.1 Simulation Results

We start with a minimum of 100 training points, running the algorithm for the points $n = 100, 101, \ldots, N-1$. This simulation is repeated 200 times, and the results presented are the average outcomes.

5.1.1 Simulation 1

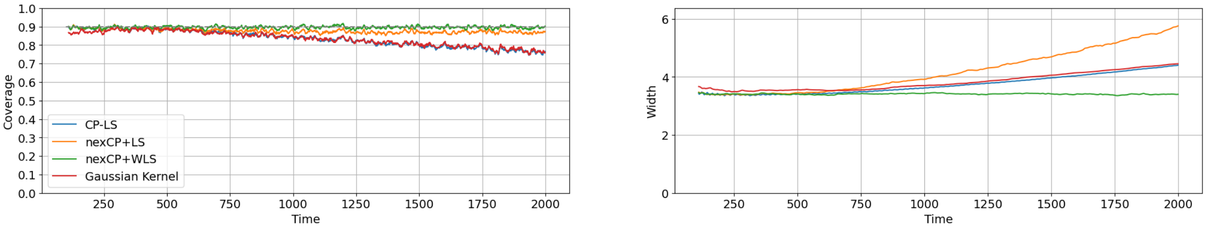

Figure 1: Coverage and Width plot for simulation 1

Method	Avg. Coverage	Avg. Interval Width
CP+LS	0.836	3.730
nexCP+LS	0.876	4.185
nexCP+WLS $\rho = 0.99$	0.896	3.408
nexCP+WLS Gaussian	0.835	3.826

Table 2: Coverage levels compared across different methods

From figure 1 and table 1 it is evident that our proposed method performs the worst, despite using the same algorithm as the best-performing method. We see that the coverage of our proposed method performs very similar to the conformal prediction with a symmetric algorithm. To understand this discrepancy, we examine the weights.

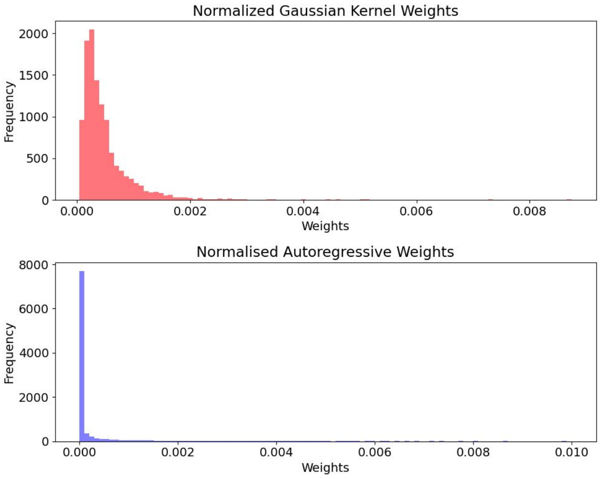

Figure 2: Histogram of the weights used for nexCP+WLS

The plot in figure 2 clarifies the results. Compared to the autoregressive parameter, our weights assign much higher importance to certain observations. The autoregressive parameter exponentially decreases the weight of previous observations, while the Gaussian kernel assigns relatively high weights to some observations because the covariates are sampled near each other. This is not ideal given the distribution drift, and it actually exacerbates the issue.

5.1.2 Simulation 2

If we select X to be two dimensional, we can first inspect the 3 dimensional scatter plot.

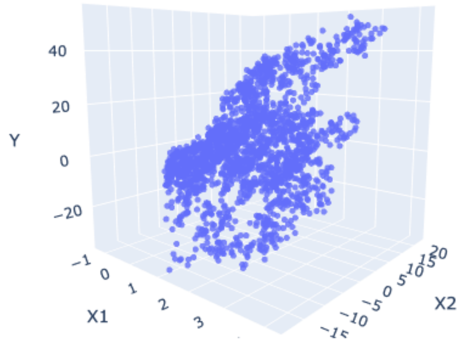

Figure 3: Scatter plot of the simulated data

If we look at figure 3 we can see clearly that if X_i is close to X_j that Y_i will also be close to Y_j. This is very similar to how a linear equations would be formed, however, with the extra component of the distance metric, this is a better simulation for spatial data.

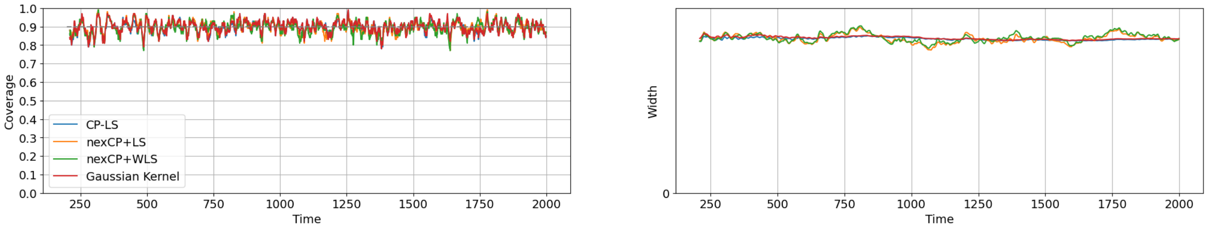

Figure 4: Coverage and Width plot for simulation 2

Method	Avg. Coverage	Avg. Interval Width
CP+LS	0.896	3.43
nexCP+LS	0.896	3.44
nexCP+WLS $\rho = 0.99$	0.894	3.45
nexCP+WLS Gaussian	0.898	3.45

Table 3: Coverage levels compared across different methods

Table 2 presents the results from our second simulation study. We see that over the 200 simulations all models perform very similar, however, our proposed method provides on average the smallest coverage gap. This can also be explained by looking at the histogram of the assigned weight.

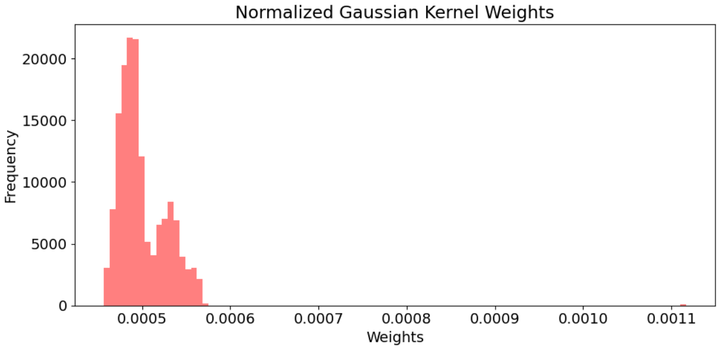

Figure 5: Histogram of assigned weights

Figure 5 shows a histogram of the weights assigned by the Gaussian kernel, as the autoregressive weight assignment will remain the same for this method as well. We can see here again that it gives a higher weighting to more covariates rather than exponentially decreasing. We can also inspect the bar plot to see the weight assignment.

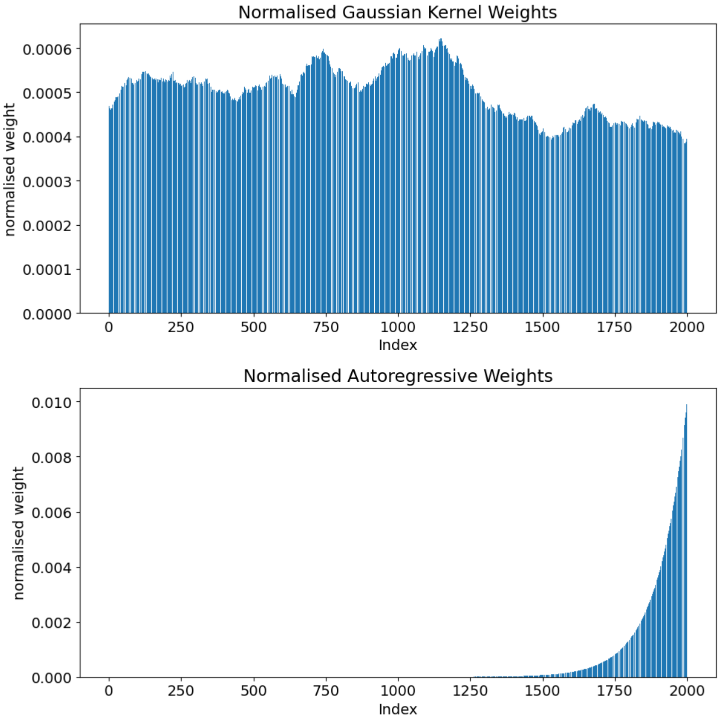

Figure 6: bar plot of assigned weights

Figure 6 shows the bar plot of the assigned weights for the prediction of the 2000^{th} observation, note that this is only one from the 200 experiments as otherwise we would average out over the various experiments and it would converge to uniform assignment of weights. Since we permuted X we see that as the index increases we do not necessarily have increased weights. We can for instance see that around the 1200^{th} has the highest assigned weights, which can explain the better performance.

5.2 Electricity dataset

We will now undertake a comparative analysis between the methods proposed by Barber et al. (2023) and our own method, which incorporates kernel density estimates, using real-world datasets. The initial comparison will be conducted using the dataset employed by Barber et al. (2023), namely, the **ELEC2 Dataset**. This dataset records the electricity usage and pricing in the states of Victoria and New South Wales in Australia.

In alignment with the approach of Barber et al. (2023), we will utilize the covariates **nswprice** and **vsprice**, which represent the electricity prices in New South Wales and Victoria, respectively. Additionally, we will consider the covariates **nswdemand** and **vsdemand**, which correspond to the electricity demand in these states. The response variable for our analysis is **transfer**, which indicates the quantity of electricity transferred between New South Wales and Victoria.

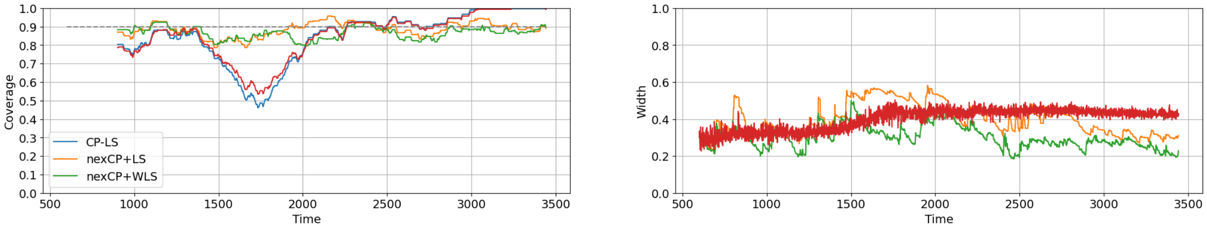

Figure 7: Coverage and Width plot for the ELEC2 Dataset

Table 4: Comparison of Different Prediction Interval Methods

Method	Coverage	Average Prediction Interval Width
CP+LS	0.849	0.402
nexCP+LS	0.883	0.398
nexCP+WLS	0.866	0.299
Gaussian Kernel RLRW	0.854	0.394,

Table 4 and figure 7 show the average coverage levels and the coverage over time respectively. We can clearly see that our proposed weighing algorithm performs relatively poorly, only outperforming the basic CP method.

Just as in Barber et al. (2023) we want to see how much the distribution drift affects the algorithms **CP+LS, nexCP+LS, nexCP + WLS** and the **RLRW**. We then select a random permutation σ and apply it to the dataset to ensure exchangeability, $\{Z_{\sigma(i)}\}_{i \in N}$.

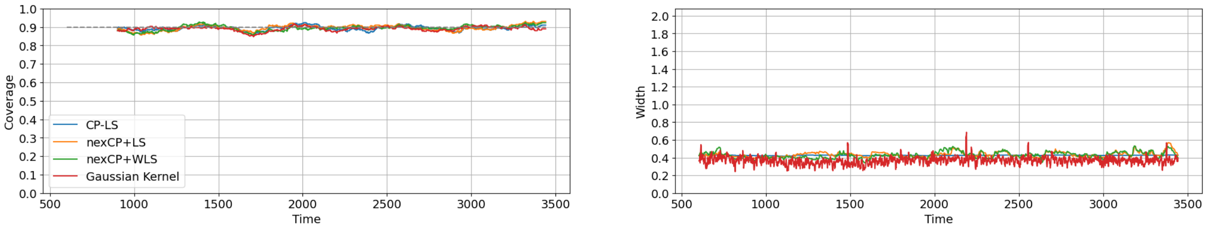

Figure 8: Coverage and Width plot for the randomly permuted ELEC2 Dataset

Table 5: Comparison of Different Prediction Interval Methods with permuted ELEC2 dataset

Method	Coverage	Average Prediction Interval Width
CP+LS	0.894	0.425
nexCP+LS	0.898	0.435
nexCP+WLS	0.896	0.429
Gaussian Kernel RLRW	0.890	0.360

Again, here figure 8 and table 5 show the average coverage levels and coverage over time respectively. We can see that for all models, the permuted data performs significantly better when it comes to finding the appropriate coverage levels. It is not evident enough to make strong conclusions which model performs better. However one thing to note is that our proposed method has the largest coverage gap, however, has an average interval width decrease of 16% comparatively to the other methods. This might lead us to believe that our method might be more efficient. However, it is not evident enough to say that our proposed method definitively performs better.

5.3 Indonesia dataset

In this section we will use the dataset used in De Ree et al. (2018). In their paper they analysed 10 years worth of data to see whether or not unconditional salary increase of teachers in Indonesia would improve test scores of students. This dataset is particularly interesting as it holds the pseudoexchangability property, as it contains data from various schools. For the covariates we use **teacher average base pay, classize, num teachers, num students, school score, SD, BIN score, assets** and **IPA score**. Here classize denotes the classsize of the student, num teachers is the number of teachers that school has, num students being the total number of students in the school, SD is an indicator whether the school is a primary school or not and assets being the "asset" score of each student. Finally, the BIN and IPA score denote the raw Indonesian and raw English test scores. We have as response variable **MAT score**, which is the raw mathematics score obtained by the student, the mathematics score is between 0 and 1. This dataset contains 59462 observations, however due to runtime limitations, we limit ourselves to 10000 observations. For this dataset we use again the Gaussian Kernel with our weights chosen as described in subsection 3.6, and we will compare this to the **CP + LS, neXP + LS, neXP + WLS** with again $w_i = \rho^{n-i+1}$, with $\rho = 0.99$. We choose this as in our previous simulation and empirical studies, this value performs very well.

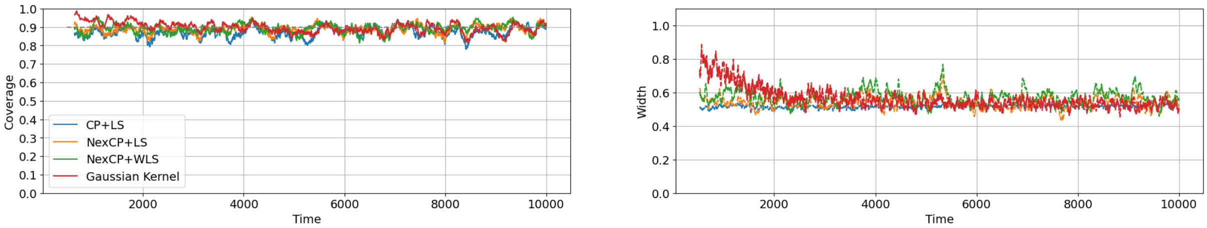

Figure 9: Coverage and Width plot for the ELEC2 Dataset

Table 6: Comparison of indonesia school dataset

Method	coverage	Average Prediction Interval Width
CP+LS	0.871	0.513
nexCP+LS	0.890	0.543
nexCP+WLS	0.893	0.572
GK-PsEx	0.898	0.564

We can see in table 6 that all the methods comes close to the desired coverage level, however, only the basic conformal prediction method is the one that undercovers the most. Moreover, we can see that our proposed method has the smallest coverage gap, however, this difference is not substantial enough to make strong conclusions.

6 Conclusion

In this paper, we have successfully reproduced the results of Barber et al. (2023) and extended the methodology to accommodate weight selection in nonexchangeable settings. Additionally, we introduced an approach for pseudoexchangeable data, where only specific partitions of the dataset exhibit exchangeability.

To validate our generalization of weight selection for nonexchangeable settings, we employed both a simulation framework proposed by Barber et al. (2023) and real-world datasets, namely the ELEC2 dataset from Australia and a dataset from Indonesia on school performance. The ELEC2 dataset is ideal for studying concept drift, as it captures dynamic changes in electricity prices influenced by demand and supply variations, regulatory changes, and seasonal factors Harries et al. (2003). The Indonesia dataset provides a unique perspective on educational outcomes, assuming intra-school exchangeability while considering inter-school differences De Ree et al. (2018).

Our findings suggest that the generalized weight selection method is most effective when there is no prior knowledge of the data distribution. In simulation studies, where there is a clear distribution drift, localized weights did not perform as well as autoregressive weights. However, in the ELEC2 and Indonesia datasets, our proposed method achieved a competitive performance, with minimal coverage gaps observed. Specifically, for the pseudoexchangeable setting, our method demonstrated almost perfect coverage, validating its robustness.

For future research, we recommend exploring the types of data for which this method performs optimally. This would allow for a more conclusive determination of the weight selection method's ability to minimize coverage gaps. Additionally, further investigation into Modrian Conformal Prediction, particularly in the context of pseudoexchangeable datasets, could yield insights into balancing prediction errors across different groups.

In summary, our contribution lies in the successful generalization of weighting schemes for nonexchangeable, non-symmetric algorithms. We provided empirical evidence that, in the absence of prior distributional knowledge, our proposed method performs reliably. Moreover, in pseudoexchangeable settings, it surpasses traditional CP algorithms, offering a promising approach for improving prediction accuracy in real-world applications.

References

Aldous, D. J. (1981). Representations for partially exchangeable arrays of random variables. *Journal of Multivariate Analysis*, *11*(4), 581-598.

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

Candès, E., Lei, L., & Ren, Z. (2023). Conformalized survival analysis. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *85*(1), 24–45.

De Ree, J., Muralidharan, K., Pradhan, M., & Rogers, H. (2018). Double for nothing? experimental evidence on an unconditional teacher salary increase in indonesia. *The Quarterly Journal of Economics*, *133*(2), 993–1039.

Fong, E., & Holmes, C. C. (2021). Conformal bayesian computation. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, & J. W. Vaughan (Eds.), *Advances in neural information processing systems* (Vol. 34, pp. 18268–18279). Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2021/file/97785e0500ad16c18574c64189ccf4b4-Paper.pdf

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

Guan, L. (2023). Localized conformal prediction: A generalized inference framework for conformal prediction. *Biometrika*, *110*(1), 33–50.

Harries, M., Nsw-cse tr, U., & Wales, N. (2003, 05). Splice-2 comparative evaluation: Electricity pricing.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., . . . Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, *585*, 357–362. doi: 10.1038/s41586-020-2649-2

Hore, R., & Barber, R. F. (2023). Conformal prediction with local weights: randomization enables local guarantees. *arXiv preprint arXiv:2310.07850*.

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in science & engineering*, *9*(3), 90–95.

Lei, J., Rinaldo, A., & Wasserman, L. (2015). A conformal prediction approach to explore functional data. *Annals of Mathematics and Artificial Intelligence*, *74*, 29–43.

McKinney, W., et al. (2010). Data structures for statistical computing in python. In *Proceedings of the 9th python in science conference* (Vol. 445, pp. 51–56).

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Silverman, B. W. (2018). *Density estimation for statistics and data analysis.* Routledge.

Stanton, S., Maddox, W., & Wilson, A. G. (2023). Bayesian optimization with conformal prediction sets. In *International conference on artificial intelligence and statistics* (pp. 959–986).

Van Rossum, G., & Drake, F. L. (2009). *Python 3 reference manual.* Scotts Valley, CA: CreateSpace.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., … SciPy 1.0 Contributors (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, *17*, 261–272. doi: 10.1038/s41592-019-0686-2

Vovk, V., Gammerman, A., & Shafer, G. (2005). *Algorithmic learning in a random world* (Vol. 29). Springer.

A Other Simulation results

In this section of the appendix we will include our suggested weights for the other simulation results from Barber et al. (2023) and include some of our own.

Setting 1 Barber et al. (2023): i.i.d. data Again with N $=$ 2000 and $X_i \overset{i.i.d.}{\sim} \mathcal{N}(0, I_4)$ and $Y_i = X_i^T\beta + \mathcal{N}(0,1)$ with $\beta = (2,1,0,0)$.

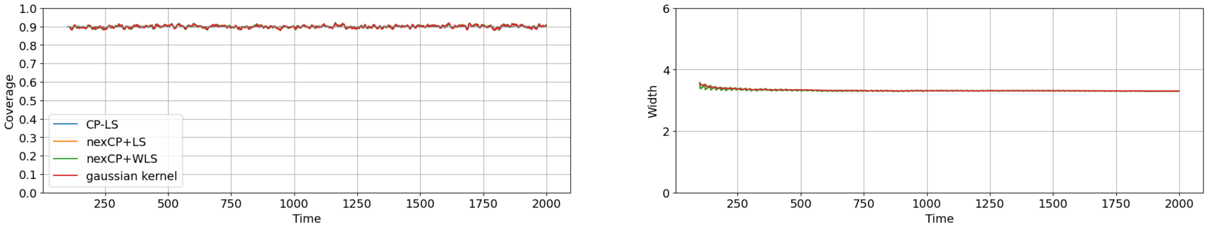

Figure 10: Coverage and Width plot for simulation 1

Table 7: Comparison simulation 1

Method	coverage	Average Prediction Interval Width
CP+LS	0.899	3.32
nexCP+LS	0.899	3.32
nexCP+WLS $\rho = 0.99$	0.899	3.32
nexCP + WLS gaussian kernel	0.899	3.32

We can see table 7 that all methods perform equally for this dataset.

Setting 2 Barber et al. (2023): changepoints N$=$ 2000 and $X_i \overset{i.i.d.}{\sim} \mathcal{N}(0, I_4)$ and $Y_i = X_i^T\beta^{(i)} + \mathcal{N}(0,1)$. Here $\beta^{(i)}$ is the coefficient vector for time (i), β changes two times during the simulation with

$$\beta^{(1)} = \cdots = \beta^{(500)} = (2, 1, 0, 0),$$
$$\beta^{(501)} = \cdots = \beta^{(1500)} = (0, -2, 1, 0),$$
$$\beta^{(1501)} = \cdots = \beta^{(2000)} = (0, 0, 2, 1).$$

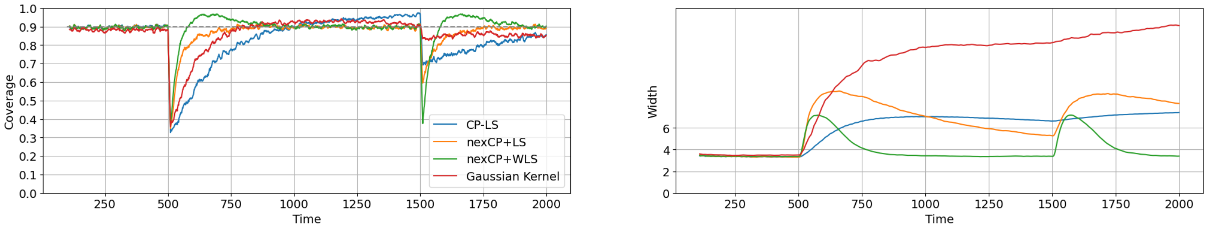

Figure 11: Coverage and Width plot for simulation 2

Table 8: Comparison simulation 2

Method	coverage	Average Prediction Interval Width
CP+LS	0.833	6.00
nexCP+LS	0.872	6.67
nexCP+WLS $\rho = 0.99$	0.895	4.07
nexCP + WLS gaussian kernel	0.863	11.04

Table 8 present the results from simulation 2. What is very noticeable is the coverage width of the gaussian kernel over time. If we look at the plot of the upper and lower bound over time it might provide some more valuable insight.

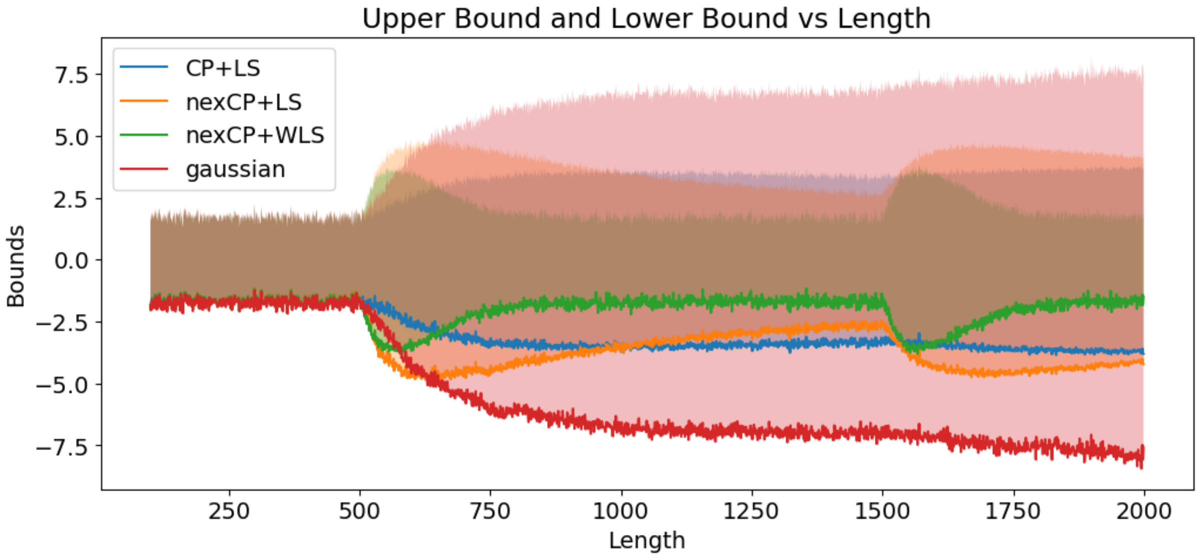

Figure 12: upper and lower bound over time of simulation 2

As every X_i is i.i.d. distributed it still weights observations before and after the changepoints more heavily, rather than selecting $w_i = \rho^{n-i+1}$ which exponentially decreases for each i.

Setting 3: Dependent data. We will first generate random variables dependent solely on the preceding random variable. Specifically, $X_i \perp\!\!\!\perp X_{j<i-1}|X_{i-1}$, and $X_i|X_{i-1} \sim \mathcal{N}(x_{i-1}, I_n)$ with $X_1 \sim \mathcal{N}(0,1)$ and $Y_i = X_i^T\beta + \mathcal{N}(0,1)$.

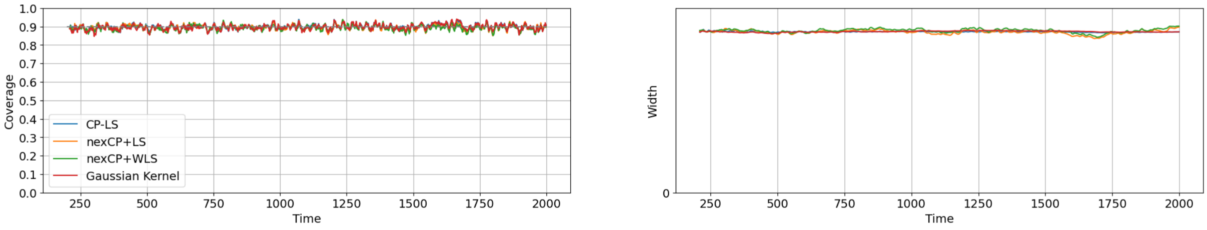

Figure 13: Coverage and Width plot for simulation 3

Table 9: Comparison simulation 4

Method	coverage	Average Prediction Interval Width
CP+LS	0.898	3.30
nexCP+LS	0.895	3.30
nexCP+WLS $\rho = 0.99$	0.893	3.33
nexCP + WLS gaussian kernel	0.898	3.30

The most notable result we find from table 9 is that even though the third model has the largest interval width, it has the smallest coverage, and the second model has the same interval width as method 1 and 4 but has a smaller average coverage.

Setting 4: Trend data. Generate $Y_t = \beta \times \sin(\frac{t}{2}) + \mathcal{N}(0,1)$ with $t \in [N]$

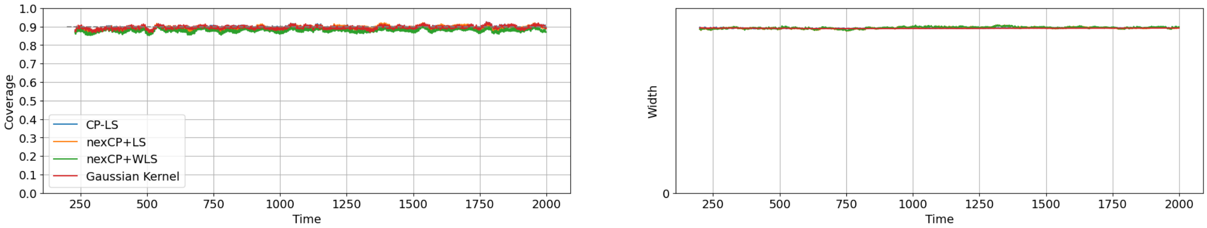

Figure 14: Coverage and Width plot for simulation 4

Table 10: Comparison simulation 4

Method	coverage	Average Prediction Interval Width
CP+LS	0.893	10.89
nexCP+LS	0.892	10.91
nexCP+WLS $\rho = 0.99$	0.878	10.92
nexCP + WLS gaussian kernel	0.894	10.88

Even though in table 10 shows that all models perform very comparable, over the 200 simulations, our proposed method performs the best with the smallest interval width.

B Relevant Proofs

B.1 Outline proof theorem 1 Vovk et al. (2005)

Please note that this proof is directly taken from Barber et al. (2023).
If we denote $R_i = R_i^{Y_{n+1}}$ as the i-th residual for the hypothesised test point $y = Y_{n+1}$ and the data points $(Z_1, Z_2, \dots, Z_n) \overset{\Delta}{=} (Z_{\pi(1)}, Z_{\pi(2)}, \dots, Z_{\pi(n)})$ with $Z_i := (X_i, Y_i)$, for any $\pi \in S_n$ i.e. the data being exchangeable. Then we construct our fitted model $\hat{\mu} = \hat{\mu}^{Y_{n+1}} = \mathcal{A}\big((X_1, Y_1), \dots (X_n, Y_n)\big)$ from an algorithm $\mathcal{A}$ that treats these points symmetrically. Next we define the residuals $R_i = |Y_i - \hat{\mu}(X_i)|$. We can define a set of outliers/strange points as:

$$S(R) = \left\{ i \in n+1 : R_i > Q_{1-\alpha}\left(\sum_{j\in[n+1]} \frac{1}{n+1} \cdot \delta_{R_j} \right) \right\}.$$

By definition how this set is formulated

$$\#(S(R)) \leqslant \alpha(n+1),$$

with $\#(\cdot)$ denoting the cardinality of a set. Next, by the definition of the prediction set, if $Y_{n+1} \notin \hat{C}_{n+1}$ i.f.f $R_{n+1} > Q_{1-\alpha}\big(\sum_{j\in[n+1]} \frac{1}{n+1} \cdot \delta_{R_j}\big)$ or equivalently $n+1 \in S(R)$. Then the outline of the proof is quite straightforward:

$$\mathbb{P}\left\{ Y_{n+1} \notin \hat{C}(X_{n+1}) \right\} = \mathbb{P}\{n+1 \in S(R)\} \overset{(*)}{=} \frac{1}{n+1} \sum_{i\in[n+1]} \mathbb{P}\{i \in S(R)\}$$
$$\overset{(**)}{=} \frac{1}{n+1} \mathbb{E}\left[\sum_{i\in[n+1]} \mathbb{1}\{i \in S(R)\} \right] = \frac{1}{n+1} \mathbb{E}(|S(R)|) \leqslant \frac{1}{n+1} \alpha(n+1) = \alpha$$

The equality $(*)$ holds because of the exchangeability nature of the data, so $\mathbb{P}\{n+1 \in S(R)\} = \mathbb{P}\{i \in S(R)\}$, so finding the the probability of one index being included is equivalent to the average index being included in the set $S(R)$. The equality $(**)$ holds as $\mathbb{E}(\mathbb{1}\{A \in \mathcal{F}\}) = 1 \cdot \mathbb{P}\{A \in \mathcal{F}\} + 0 \cdot \mathbb{P}\{A \notin \mathcal{F}\} = \mathbb{P}\{A \in \mathcal{F}\}$ and the linearity of the expectation.

C Software packages

Table 10 indicates which Python (see Van Rossum and Drake (2009)) package or GitHub repository was used for all experiments.

Model or Method	**Python Package**	**Reference**
Linear Regressions	statmodels, sklearn	Virtanen et al. (2020), Pedregosa et al. (2011)
Data visualization	matplotlib	Hunter (2007)
Ubiquitous	pandas, numpy	McKinney et al. (2010), Harris et al. (2020)

Table 11: A table that indicates which python packages were used

D Summary statistics on Data

	period	nswprice	nswdemand	vicprice	vicdemand	transfer
mean	0.500000	0.057868	0.425418	0.003467	0.422915	0.500526
std	0.294756	0.039991	0.163323	0.010213	0.120965	0.153373
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.250000	0.035127	0.309134	0.002277	0.372346	0.414912
50%	0.500000	0.048652	0.443693	0.003467	0.422915	0.414912
75%	0.750000	0.074336	0.536001	0.003467	0.469252	0.605702
max	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

Table 12: Summary statistics table of ELEC2 dataset

	MAT	IPA	BIN	assets	avg base pay	class size	num teachers	num students	school score
mean	0.41	0.52	0.59	0.54	2.18	23.96	39.39	278.68	0.20
std	0.23	0.21	0.20	0.24	0.40	5.67	18.85	173.00	0.96
min	0.00	0.00	0.00	0.00	0.22	2.83	13.00	14.00	-1.92
25%	0.23	0.36	0.45	0.43	1.97	21.00	24.00	144.00	-0.57
50%	0.36	0.50	0.59	0.57	2.26	24.25	34.00	218.00	0.06
75%	0.55	0.67	0.73	0.71	2.46	26.83	50.00	399.00	0.81
max	1.00	1.00	1.00	1.00	3.16	40.83	113.00	784.00	3.79

Table 13: Summary statistics table for Indonesia dataset