ECONSTOR
Make Your Publications Visible.

A Service of

ZBW Leibniz-Informationszentrum Wirtschaft
Leibniz Information Centre for Economics

Paul, Joseph R.; Schaffer, Mark E.

Working Paper

An introduction to conformal inference for economists

Accountancy, Economics, and Finance Working Papers, No. 2024-13

Provided in Cooperation with:
Heriot-Watt University, Department of Accountancy, Economics, and Finance

Suggested Citation: Paul, Joseph R.; Schaffer, Mark E. (2024) : An introduction to conformal inference for economists, Accountancy, Economics, and Finance Working Papers, No. 2024-13, Heriot-Watt University, Department of Accountancy, Economics, and Finance, Edinburgh

This Version is available at:
https://hdl.handle.net/10419/308058

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

WWW.ECONSTOR.EU

Mitglied der Leibniz-Gemeinschaft

HERIOT WATT UNIVERSITY | EDINBURGH BUSINESS SCHOOL

Heriot-Watt University

Accountancy, Economics, and Finance Working Papers

Working Paper 2024-13

AN INTRODUCTION TO CONFORMAL INFERENCE FOR ECONOMISTS

Joseph R. Paul

Mark E. Schaffer

December 2024

Keywords: conformal inference, conformal prediction, distribution-free inference, machine learning

JEL: C12, C14, C53

An Introduction to Conformal Inference for Economists*

Joseph R. Paul
Heriot-Watt University

Mark E. Schaffer
Heriot-Watt University

December 2024

Abstract

This paper introduces conformal inference, a powerful and flexible framework for constructing prediction intervals with guaranteed coverage in finite samples. Unlike conventional methods, conformal inference makes no assumptions about the underlying data distribution other than exchangeability. The paper begins with some simple examples of full and split conformal prediction that highlight the key assumption of exchangeability. We then provide more formal treatments of full and split conformal prediction along with extensions of the basic framework, including the Jackknife+ and CV+ algorithms, both of which offer a better balance between computational and statistical efficiency compared to full and split conformal prediction. The paper then discusses the limitations to achieving exact conditional coverage and several methods that aim to improve conditional coverage in practice. The final section briefly discusses areas of current research the software options for implementing conformal methods.

*Invited paper for the Eighth International Econometric Conference of Vietnam, 'Artificial Intelligence and Machine Learning in Econometrics', Ho-Chi-Minh City, Vietnam, 13-15 January 2025. All errors are our own.

1. Introduction

This paper introduces conformal inference, a powerful and flexible framework for constructing prediction intervals with guaranteed coverage in finite samples. Unlike conventional methods, conformal inference makes no assumptions about the underlying data distribution other than exchangeability, a weaker condition than i.i.d. (independent and identically distributed). This distribution-free nature, coupled with its finite-sample validity, makes conformal inference a potentially attractive tool for empirical research in economics.

The idea behind conformal prediction is to assess the "nonconformity" of a new observation relative to a set of observed data. By comparing the nonconformity of a potential outcome to the nonconformity of the observed data, we can construct prediction intervals that contain the true outcome with a specified probability. This is achieved without imposing any parametric or regularity assumptions on the data generating process or requiring the underlying predictive model to be correctly specified.

The ideas behind conformal prediction originate in work starting in the late 1990s by computer scientists Vladimir Vovk, Alexander Gammerman and Vladimir Vapnik (see, e.g. Volodya Vovk, Alexander Gammerman, and Saunders (1999)). Interest picked up in the 2010s, largely by academics in the statistics community, and in the past 5+ years the number of papers and developments has exploded. Econometricians and applied economists are now also among the academics working in this area.

But despite its theoretical appeal and empirical success in other fields, conformal inference remains relatively unknown and underutilised in economics. This paper aims to bridge this gap by providing an introduction to conformal inference tailored to an economics and econometrics audience. We limit our attention to prediction of a continuous outcome Y, mostly for reasons of space; conformal prediction methods for categorical variables are also available and use the same principles.

The paper begins in Section 2 by illustrating the basic principles of full and split conformal prediction through simple examples, highlighting the key assumption of exchangeability and demonstrating how to construct one-sided and two-sided prediction intervals. We then provide a more formal treatment of the methodology in Section 3, where we introduce the concept of conformity scores and discuss the theoretical guarantees of both full and split conformal inference.

In Section 4 we explore extensions of the basic framework, including the Jackknife+ and CV+ algorithms, both of which offer a better balance between computational and statistical efficiency compared to full and split conformal prediction. We then discuss the issue of conditional coverage in Section 5, where we discuss the limitations of achieving exact conditional coverage and where we introduce four methods that aim to improve conditional coverage in practice.

Finally, in Section 6, we briefly discuss the various directions current research is exploring, and the software options for implementing these methods.

2. Conformal prediction: Some simple examples

In this section we illustrate, using the simplest possible data setup, how the two basic variants of conformal prediction work: "full conformal prediction" and "split conformal prediction". Both methods provide guaranteed marginal coverage in finite samples with no assumptions about the distribution of the data other than exchangeability and a stable data generating process (DGP). The methods can be applied to prediction with any learner or ensemble of learners $f(.)$, with any number of predictors X, and using any measure of "conformity", i.e., how close the prediction $\widehat{f}_{n+1}$ is to the actual outcome Y_{n+1}.

The key assumption, and the key to the intuition behind how conformal inference works, is exchangeability. Exchangeability requires that the joint distribution of our data $\{(X_i, Y_i)\}_{i=1}^n$ is invariant to any permutation – any reordering – of the data. For example, for three observations Y_1, Y_2, Y_3, exchangeability implies $Pr(Y_1, Y_2, Y_3) = Pr(Y_2, Y_1, Y_3) = Pr(Y_3, Y_1, Y_2), \ldots$ and so on.

Exchangeability rarely shows up in basic econometrics textbooks, but in fact our students are exposed to it via different terminology when we teach panel data methods. In the basic error components model, the composite error is $\varepsilon_{it} = \eta_i + \nu_{it}$, where η_i is a time-invariant component specific to panel unit i and ν_{it} is i.i.d. This is an exchangeable error structure: ε_{it} exhibits dependence within panels via the component η_i, but this dependence is invariant to reordering of the observations within the panel unit because η_i is time-invariant.

All i.i.d. sequences are exchangeable, but as the panel data example illustrates, not all exchangeable sequences are i.i.d. Sampling without replacement is another example of an exchangeable sequence that is not i.i.d. (after drawing $n-1$ samples, we know what the last one will be). Note that if a sequence $\{X_i\}$ is exchangeable and $g(.)$ is any measurable function, then the set of Z_i generated by $Z_i = g(X_i)$ is exchangeable[1]. We will make use of this fact below when constructing predictions, residuals, conformal scores, etc.

It is important to emphasize that this means the basic setup for conformal inference can't be used for time series or other structured data without modification. How to accommodate these types of data is an area of current research; more on this later.

The key intuition behind the conformal inference method is to treat the data symmetrically, i.e., in a way that respects exchangeability.

We begin by illustrating how full conformal inference works, and then turn to the more commonly-used split conformal method. We use the simplest possible setting: a small dataset $\{Y_i\}_{i=1}^{10}$ of only 10 observations on an outcome Y_i, and no predictors X_i. The observations are drawn from a standard normal distribution with mean 100 and standard deviation 10 (but of course the researcher doesn't know this). Our learner – our prediction function $f(.)$ – will be the sample mean function $\mu(.)$. Training our learner amounts to calculating the sample mean $\widehat{\mu}$ over a set of values. The error level is α and we want to construct intervals with guaranteed

[1] If we permute the indices of the Z_i's, it's equivalent to permuting the indices of the corresponding X_i's before applying the function g. Because the X_i sequence is exchangeable, this permutation doesn't change the joint distribution, and thus the transformed sequence $\{Z_i\}$ remains exchangeable.

coverage at least $1-\alpha$.[2]

2.1. Full conformal example: one-sided prediction interval

In the first example, we will construct a one-sided prediction interval for $\alpha = 0.5$, i.e., with a minimum 50% coverage guarantee. Our nonconformity measure is just the residual $R_i = Y_i - \widehat{\mu}$. A large value of the residual R_i means low conformity with the data, and a low value means high conformity. We are constructing a one-sided prediction interval, so for very low values of Y_i, the residual R_i will be very negative, implying that observation is highly conformal; a large Y_i and a very positive R_i means that observation is highly nonconformal.

In full conformal prediction, we construct the prediction interval using our dataset and a grid of "trial values". We consider whether each trial value y_{trial} is included in, or excluded from, the prediction interval. For expositional clarity only our trial values will be integers.

The full conformal algorithm is as follows.

For each $y_{trial} \in$ Trial-Grid:

Step 1: Append the trial value to the sample: $D_{trial} = \{Y_i\}_{i=1}^{10} \bigcup \{y_{trial}\}$.

Step 2: Calculate the mean of the augmented sample $\widehat{\mu}_{trial} = mean(D_{trial})$.

Step 3: Calculate the conformity score for each observation in the dataset $R_i = Y_i - \widehat{\mu}_{trial}$, $i = 1, ..., 10$.

Step 4: Calculate the rank of the conformity score for the new observation $Rank(R_{trial}) = \sum_{i=1}^{n+1} 1[R_i \leq R_{trial}]$ where here $n+1 = 11$.

Step 5: Include y_{trial} in the prediction interval if $Rank(R_{trial}) \leq \lceil (1-\alpha) \times (n+1) \rceil$.

And since we are constructing a one-sided interval, the conformity score R_i in Step 3 is the residual $R_i = Y_i - \widehat{\mu}_{trial}$, $i = 1, ..., 10$.

Note that in this example, $(1-\alpha) \times (n+1) = 0.5 \times 11 = 5.5$, so we need to use the ceiling function $\lceil . \rceil$ to round up to 6 before comparing to the rank of the trial value residual $Rank(R_{trial})$.

We use $\{93, ..., 107\}$ as our grid of trial values. This grid is almost entirely arbitrary – y_{trial} need only be in the domain of Y – and is chosen only for expositional clarity: if we wished, we could instead concentrate our search near certain values, we could use non-integer trial values, etc. This flexibility is a key difference between the full and split conformal approaches, as we shall see later.

The example is presented in Table 1.

[2]The inequality in the coverage guarantee is a weak inequality because of the possibility of ties. If the dataset is drawn from a continuous distribution and therefore there are almost surely no ties, then the coverage guarantee can be sharpened to a strict inequality and applies almost surely. There are other methods for dealing with ties. But for simplicity of exposition here we just use a weak inequality in the coverage guarantee.

TABLE 1. Full conformal prediction example: One-sided interval, $n = 10$, $\alpha = 0.5$

Trial number:		Mean over n+1 obs	Observed data Y_i and residuals R_i										Ranked residuals, small to large:										$\lceil (1-\alpha)(n+1) \rceil$ = 6th smallest:	$R_{trial} \leq$ 6th smallest?	Corresp. y_{trial}:
	Trial y	$\widehat{\mu}$											1	2	3	4	5	**6**	7	8	9	10		R_{trial}	
1	93	98.21	3.65	2.94	-2.14	5.66	-20.85	-5.24	18.64	-7.6	12.81	-2.66	-20.85	-7.6	-5.24	-2.66	-2.14	**2.94**	3.65	5.66	12.81	18.64	2.94	-5.21	**TRUE**	93
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
2	94	98.3	3.56	2.85	-2.23	5.57	-20.94	-5.33	18.55	-7.69	12.72	-2.76	-20.94	-7.69	-5.33	-2.76	-2.23	**2.85**	3.56	5.57	12.72	18.55	2.85	-4.3	**TRUE**	94
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
3	95	98.39	3.47	2.76	-2.32	5.48	-21.03	-5.42	18.46	-7.78	12.63	-2.85	-21.03	-7.78	-5.42	-2.85	-2.32	**2.76**	3.47	5.48	12.63	18.46	2.76	-3.39	**TRUE**	95
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
4	96	98.48	3.38	2.67	-2.41	5.38	-21.12	-5.52	18.37	-7.87	12.54	-2.94	-21.12	-7.87	-5.52	-2.94	-2.41	**2.67**	3.38	5.38	12.54	18.37	2.67	-2.48	**TRUE**	96
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
5	97	98.57	3.29	2.58	-2.5	5.29	-21.21	-5.61	18.28	-7.96	12.45	-3.03	-21.21	-7.96	-5.61	-3.03	-2.5	**2.58**	3.29	5.29	12.45	18.28	2.58	-1.57	**TRUE**	97
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
6	98	98.66	3.2	2.49	-2.59	5.2	-21.3	-5.7	18.18	-8.05	12.35	-3.12	-21.3	-8.05	-5.7	-3.12	-2.59	**2.49**	3.2	5.2	12.35	18.18	2.49	-0.66	**TRUE**	98
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
7	99	98.75	3.1	2.4	-2.69	5.11	-21.39	-5.79	18.09	-8.14	12.26	-3.21	-21.39	-8.14	-5.79	-3.21	-2.69	**2.4**	3.1	5.11	12.26	18.09	2.4	0.25	**TRUE**	99
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
8	100	98.84	3.01	2.31	-2.78	5.02	-21.48	-5.88	18	-8.23	12.17	-3.3	-21.48	-8.23	-5.88	-3.3	-2.78	**2.31**	3.01	5.02	12.17	18	2.31	1.16	**TRUE**	100
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
9	101	98.93	2.92	2.21	-2.87	4.93	-21.57	-5.97	17.91	-8.32	12.08	-3.39	-21.57	-8.32	-5.97	-3.39	-2.87	**2.21**	2.92	4.93	12.08	17.91	2.21	2.07	**TRUE**	101
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
10	102	99.02	2.83	2.12	-2.96	4.84	-21.67	-6.06	17.82	-8.41	11.99	-3.48	-21.67	-8.41	-6.06	-3.48	-2.96	**2.12**	2.83	4.84	11.99	17.82	2.12	2.98	FALSE	
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
11	103	99.11	2.74	2.03	-3.05	4.75	-21.76	-6.15	17.73	-8.51	11.9	-3.57	-21.76	-8.51	-6.15	-3.57	-3.05	**2.03**	2.74	4.75	11.9	17.73	2.03	3.89	FALSE	
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
12	104	99.21	2.65	1.94	-3.14	4.66	-21.85	-6.24	17.64	-8.6	11.81	-3.66	-21.85	-8.6	-6.24	-3.66	-3.14	**1.94**	2.65	4.66	11.81	17.64	1.94	4.79	FALSE	
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
13	105	99.3	2.56	1.85	-3.23	4.57	-21.94	-6.33	17.55	-8.69	11.72	-3.76	-21.94	-8.69	-6.33	-3.76	-3.23	**1.85**	2.56	4.57	11.72	17.55	1.85	5.7	FALSE	
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
14	106	99.39	2.47	1.76	-3.32	4.48	-22.03	-6.42	17.46	-8.78	11.63	-3.85	-22.03	-8.78	-6.42	-3.85	-3.32	**1.76**	2.47	4.48	11.63	17.46	1.76	6.61	FALSE	
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
15	107	99.48	2.38	1.67	-3.41	4.38	-22.12	-6.52	17.37	-8.87	11.54	-3.94	-22.12	-8.87	-6.52	-3.94	-3.41	**1.67**	2.38	4.38	11.54	17.37	1.67	7.52	FALSE	
Rank of R_i			7	6	5	8	1	3	10	2	9	4														
Interval endpoint = max(y_{trial} in conformal set):																										**101**

Consider the first trial value in in the grid, $y_{trial} = 93$. The mean of the augmented sample – the dataset of 10 values plus the trial value 93 – is $\widehat{\mu}_{trial} = 98.21$. In the left-hand panel, we have the observed residuals R_i and their rank, from small to large. In the right-hand panel, we repeat the observed residuals in ascending order, i.e., the values of the empirical quantile function $\widehat{q}$ for the residuals from smallest to largest.

As noted above, the choice of $\alpha = 0.5$ means we should be looking at the quantile given in column $\lceil (1-\alpha) \times (n+1) \rceil = 6$. The 6th-largest residual is 2.94, corresponding to $i = 2$ ($R_2 = Y_2 - \widehat{\mu}_{trial} = 101.15 - 98.21 = 2.94$). Since $R_{trial} = y_{trial} - \widehat{\mu}_{trial} = 93 - 98.21 = -5.21$ which is smaller than 2.94, the trial value 93 is inside our prediction interval.

We repeat for the next trial value, 94. We need to re-train our learner using the data plus the new trial value, which in this simple example means recalculating the mean. Now the augmented sample mean is $\widehat{\mu}_{trial} = 98.30$, and the 6th-largest residual is 2.85, again corresponding to the $i = 2$ observation in the actual data. The trial residual is now -4.30, which is again smaller than the 6th-largest residual, and so the trial value 94 is inside the prediction interval.

This continues through trial value 101; all the trial values $93, ..., 101$ are inside our prediction set. Now consider trial value $y_{trial} = 102$. The augmented sample mean is $\widehat{\mu}_{trial} = 99.02$, and the 6th-largest residual is 2.12. The trial residual $R_{trial} = y_{trial} - \widehat{\mu}_{trial} = 102 - 99.02 = 2.98 > 2.12$. Since the trial residual is greater than the 6th-largest residual for the actual data, this trial value is **not** in our prediction set.

The same calculation applies to trial values $103, ..., 107$; all of these trial values turn out to lie outside our prediction set. Hence our final prediction set is $Y \leq 101$. This prediction set has a $1 - \alpha = 50\%$ coverage guarantee: assuming the next actual observation Y_{n+1} is drawn from the same distribution as the original dataset, the probability that the next observation $Y_{n+1} \leq 101$ is at least 50%.[3]

The reason this works is because at every stage we have treated the data, including the trial values, symmetrically, i.e., we have respected exchangeability. When we considered the first trial value, 93, we retrained our learner (i.e., we recalculated the sample mean) on the augmented dataset including the trial value; the trial value was treated as no different from the actual observations in the dataset. Since our learner (the sample mean function) is a symmetric function of the training data, all 11 residuals are exchangeable. That is, the rank of the conformity score for the trial observation is equally likely to be any of the 11 possible positions.

Since the residuals $\{R_1, ..., R_{10}, R_{trial}\}$ are exchangeable, the rank of R_{trial} is uniformly distributed over $\{R_1, ..., R_{10}, R_{trial}\}$. This means that the probability that R_{trial} is among the $\lceil (1-\alpha) \times (n+1) \rceil = 6$ smallest of $\{R_1, ..., R_{10}, R_{trial}\}$ is at least $1 - \alpha = 50\%$. And since R_{trial}can never be larger than itself, this in turn means the probability that R_{trial} is among the $\lceil (1-\alpha) \times (n+1) \rceil = 6$ smallest of $\{R_1, ..., R_{10}\}$ is also at least $1 - \alpha = 50\%$. And given the dataset $\{Y_i\}_{i=1}^{10}$, the residuals $\{R_1, ..., R_{10}, R_{trial}\}$ can be calculated for any trial value Y_{trial} as

[3] As noted above, if we, e.g., rule out ties almost surely, the coverage guarantee can be sharpened to $Y_{n+1} < 101$.

long as we re-train our learner (we recalculate the augmented sample mean), treating all the points in the augmented data D_{trial} symmetrically.

2.2. Full conformal example: two-sided prediction interval

The construction of two-sided full conformal prediction intervals is very similar to the above. The only difference is that we have to change the nonconformity measure so that very small, as well as very large, residuals indicate high nonconformity. A common choice is the absolute residual: $R_i = |Y_i - \widehat{\mu}|$. The algorithm is exactly as before except for the change of definition of the conformity score in Step 3. The only other change we will want to make is to our grid of trial values, because now we are looking for both lower and upper bounds. Since we have almost complete flexibility in the choice of trial values, we select trial values located near the two endpoints: $\{91, ..., 97, 103, ..., 110\}$. Again, we are using integer-valued trial values only for expositional clarity.

Table 2 shows the calculations and results. At the lower end of the trial grid, trial value $y_{trial} = 92$ gives us an augmented mean of $\widehat{\mu}_{trial} = 98.11$. The 6th-largest residual is 5.75, vs the trial value absolute residual of $R_{trial} = 6.11$, so this point lies outside the prediction set. Trial value $y_{trial} = 93$ gives us an augmented mean of $\widehat{\mu}_{trial} = 98.21$, and now the 6th-largest residual is 5.66. The absolute residual for this trial value is 5.21, so now the trial value is inside the prediction set. The same is true of the other lower-end trial values $94, ..., 97$.

TABLE 2. Full conformal prediction example: Two-sided interval, $n = 10$, $\alpha = 0.5$

Trial number:		Mean over n+1 obs	Observed data Y_i and residuals R_i										Ranked residuals, small to large:										$\lceil (1-\alpha)(n+1) \rceil$ = 6th smallest:	$R_{trial} \leq$ 6th smallest?	Corresp. y_{trial}:
	Trial y	$\widehat{\mu}$											1	2	3	4	5	**6**	7	8	9	10		R_{trial}	
1	91	98.02	3.83	3.12	1.96	5.84	20.67	5.06	18.82	7.41	12.99	2.48	1.96	2.48	3.12	3.83	5.06	**5.84**	7.41	12.99	18.82	20.67	5.84	7.02	FALSE	
Rank of R_i			4	3	1	6	10	5	9	7	8	2														
2	92	98.11	3.74	3.03	2.05	5.75	20.76	5.15	18.73	7.51	12.9	2.57	2.05	2.57	3.03	3.74	5.15	**5.75**	7.51	12.9	18.73	20.76	5.75	6.11	FALSE	
Rank of R_i			4	3	1	6	10	5	9	7	8	2														
3	93	98.21	3.65	2.94	2.14	5.66	20.85	5.24	18.64	7.6	12.81	2.66	2.14	2.66	2.94	3.65	5.24	**5.66**	7.6	12.81	18.64	20.85	5.66	5.21	TRUE	93
Rank of R_i			4	3	1	6	10	5	9	7	8	2														
4	94	98.3	3.56	2.85	2.23	5.57	20.94	5.33	18.55	7.69	12.72	2.76	2.23	2.76	2.85	3.56	5.33	**5.57**	7.69	12.72	18.55	20.94	5.57	4.3	TRUE	94
Rank of R_i			4	3	1	6	10	5	9	7	8	2														
5	95	98.39	3.47	2.76	2.32	5.48	21.03	5.42	18.46	7.78	12.63	2.85	2.32	2.76	2.85	3.47	5.42	**5.48**	7.78	12.63	18.46	21.03	5.48	3.39	TRUE	95
Rank of R_i			4	2	1	6	10	5	9	7	8	3														
6	96	98.48	3.38	2.67	2.41	5.38	21.12	5.52	18.37	7.87	12.54	2.94	2.41	2.67	2.94	3.38	5.38	**5.52**	7.87	12.54	18.37	21.12	5.52	2.48	TRUE	96
Rank of R_i			4	2	1	5	10	6	9	7	8	3														
7	97	98.57	3.29	2.58	2.5	5.29	21.21	5.61	18.28	7.96	12.45	3.03	2.5	2.58	3.03	3.29	5.29	**5.61**	7.96	12.45	18.28	21.21	5.61	1.57	TRUE	97
Rank of R_i			4	2	1	5	10	6	9	7	8	3														
8	103	99.11	2.74	2.03	3.05	4.75	21.76	6.15	17.73	8.51	11.9	3.57	2.03	2.74	3.05	3.57	4.75	**6.15**	8.51	11.9	17.73	21.76	6.15	3.89	TRUE	103
Rank of R_i			2	1	3	5	10	6	9	7	8	4														
9	104	99.21	2.65	1.94	3.14	4.66	21.85	6.24	17.64	8.6	11.81	3.66	1.94	2.65	3.14	3.66	4.66	**6.24**	8.6	11.81	17.64	21.85	6.24	4.79	TRUE	104
Rank of R_i			2	1	3	5	10	6	9	7	8	4														
10	105	99.3	2.56	1.85	3.23	4.57	21.94	6.33	17.55	8.69	11.72	3.76	1.85	2.56	3.23	3.76	4.57	**6.33**	8.69	11.72	17.55	21.94	6.33	5.7	TRUE	105
Rank of R_i			2	1	3	5	10	6	9	7	8	4														
11	106	99.39	2.47	1.76	3.32	4.48	22.03	6.42	17.46	8.78	11.63	3.85	1.76	2.47	3.32	3.85	4.48	**6.42**	8.78	11.63	17.46	22.03	6.42	6.61	FALSE	
Rank of R_i			2	1	3	5	10	6	9	7	8	4														
12	107	99.48	2.38	1.67	3.41	4.38	22.12	6.52	17.37	8.87	11.54	3.94	1.67	2.38	3.41	3.94	4.38	**6.52**	8.87	11.54	17.37	22.12	6.52	7.52	FALSE	
Rank of R_i			2	1	3	5	10	6	9	7	8	4														
13	108	99.57	2.29	1.58	3.5	4.29	22.21	6.61	17.28	8.96	11.45	4.03	1.58	2.29	3.5	4.03	4.29	**6.61**	8.96	11.45	17.28	22.21	6.61	8.43	FALSE	
Rank of R_i			2	1	3	5	10	6	9	7	8	4														
14	109	99.66	2.2	1.49	3.59	4.2	22.3	6.7	17.18	9.05	11.35	4.12	1.49	2.2	3.59	4.12	4.2	**6.7**	9.05	11.35	17.18	22.3	6.7	9.34	FALSE	
Rank of R_i			2	1	3	5	10	6	9	7	8	4														
15	110	99.75	2.1	1.4	3.69	4.11	22.39	6.79	17.09	9.14	11.26	4.21	1.4	2.1	3.69	4.11	4.21	**6.79**	9.14	11.26	17.09	22.39	6.79	10.25	FALSE	
Rank of R_i			2	1	3	4	10	6	9	7	8	5														

Interval lower bound = min(y_{trial} in conformal set):	**93**
Interval upper bound = max(y_{trial} in conformal set):	**105**

At the upper end of the grid, for $y_{trial} = 105$ we get $\widehat{\mu}_{trial} = 99.30$, the 6th-largest residual is 6.33, $R_{trial} = 5.70$, so this trial value is again inside the prediction set. For the next trial value $y_{trial} = 106$, the re-trained mean is $\widehat{\mu}_{trial} = 99.39$, the 6th-largest residual is 6.42, $R_{trial} = 6.61$, so $y_{trial} = 106$ is outside the prediction set, and the same is true of the other upper-end trial values $107, ..., 110$. Hence our prediction interval is $[93,\ 105]$ and the coverage guarantee is again that the probability that the next observed observation Y_{n+1} is covered by the interval is at least 50%.

2.3. Full conformal inference in practice

The key limitation to full-conformal inference is its computational cost. In our simple example, we have to recalculate our prediction interval for every different trial value y that we are interested in. We have complete flexibility over the choice of grid points, so we can use as fine a grid as we wish in the neighbourhoods of the prediction interval endpoints. We have to retrain our learner for every gridpoint, but of course if our learner is just the sample mean this is very easy even for large datasets. Using other learners (e.g., the sample median) or some other nonconformity measure doesn't change things.

Now extend to the case where we have some predictors (features) X. The trial value is now (x, y), i.e., a hypothetical combination of a possible value for the outcome and a set of possible values for the predictors. We train the learner $f(.)$ on the union of the observed data $(X_i, Y_i), i = 1, ..., n$ and the trial value (x, y). The point predictions for the observed data are $\widehat{f}_{n,(x,y)}(X_i), i = 1, ..., n$ and $\widehat{f}_{n,(x,y)}(x)$. Everything else goes through as above starting with defining the residuals using the point predictions. But now the computational cost of full conformal prediction is clear. We have to train our learner $f(.)$ **for every combination** (y, x) **we are interested in**. In other words, if we have p predictors X, instead of a grid search of trial values in one dimension on $\mathbb{R}$, we have to do a grid search of trial values in $p + 1$ dimensions on $\mathbb{R}^{p+1}$[4].

This will be computationally very costly unless the p is fairly small and/or we have a good idea of what the grid of trial values should be composed of.

2.4. Split conformal example: One- and two-sided prediction intervals

This is where split conformal predictive inference rides in to the rescue. The algorithm for split conformal looks at first glance rather different from full conformal, but in fact it is closely related. The tradeoff is a computational cost saving in exchange for loss of flexiblity over the grid of trial values. In effect, a portion of the dataset becomes the grid of trial values, and if this is done, then the learner needs to be trained only once instead of for every grid point. The

[4] An alternative approach is to fix $x = X_{n+1}$ and perform a grid search over Y. However we would still need to repeat this procedure for each new X and would thus not be practical if we wished to perform inference for anything more than a small number of points.

coverage guarantee is exactly the same as with full conformal prediction.

To illustrate, we extend our dataset to 20 observations, which we will split into two subsamples of 10 observations each. The dataset is drawn from the same distribution as before (standard normal $N(100, 10)$). Observations $i = 1, ..., 10$ are used for calibration and have the same values the dataset used in the full conformal example. The 10 observations in the training sample are not reported here but have sample mean $\widehat{\mu}_{train} = 102.77$.

The split conformal algorithm is as follows.

Step 1: Split data into training $\mathcal{D}_{train}$ and calibration $\mathcal{D}_{cal}$ sets.

Step 2: Calculate the mean $\widehat{\mu}_{train}$ on $\mathcal{D}_{train}$.

Step 3: Using $\widehat{\mu}_{train}$ from Step 2, calculate the conformity score R_i for each observation in the calibration set $\mathcal{D}_{cal}$, i.e., for observations $i = 1, ..., 10$.

Step 4: Calculate the rank of the conformity score for the residuals in the calibration set.

Step 5: Include Y_i, $i = 1, ..., 10$ in the prediction interval if $Rank(R_i) \leq \lceil (1-\alpha) \times (n+1) \rceil$.

As with full conformal prediction, we use either the residual $R_i = Y_i - \widehat{\mu}$ or the absolute residual $R_i = |Y_i - \widehat{\mu}|$, depending on whether we are constructing a one-sided or two-sided interval. Also as in the full conformal example, $(1-\alpha) \times (n+1) = 5.5$ is a non-integer, so we use the ceiling function $\lceil . \rceil$ when constructing the interval, i.e., we are looking for the 6th-largest residual.

The example is presented in Table 3. The table contains the calculations for both the one-sided and two-sided intervals, again with target coverage of 50%. The learner is the sample mean function applied to the training data observations $i = 1, ..., 10$, which gives us $\widehat{\mu} = 102.77$.

TABLE 3. Split conformal prediction example: One- and two-sided intervals, $n_{cal} = 10$, $\alpha = 0.5$

One-sided interval:																					
Mean over D_{train}: $\widehat{\mu} = 102.77$	Observed calibration data Y_i and residuals R_i										Ranked residuals R_i, small to large, and corresponding Y_i Include in conformal set if $R_i \leq \lceil (1-\alpha)(n+1) \rceil = 6$th smallest										
Rank of R_i:	7	6	5	8	1	3	10	2	9	4	1	2	3	4	5	**6**	7	8	9	10	
Residuals R_i:	-0.92	-1.62	-6.71	1.09	-25.41	-9.81	14.07	-12.16	8.24	-7.23	-25.41	-12.16	-9.81	-7.23	-6.71	**-1.62**	-0.92	1.09	8.24	14.07	
Corresponding Y_i:	101.86	101.15	96.07	103.86	77.36	92.96	116.84	90.61	111.01	95.54	77.36	90.61	92.96	95.54	96.07	101.15	101.86	103.86	111.01	116.84	
In interval?											**TRUE**	**TRUE**	**TRUE**	**TRUE**	**TRUE**	**TRUE**	FALSE	FALSE	FALSE	FALSE	
Interval endpoint = max(Y_i in conformal set):																					**101.15**

Two-sided interval:																					
Mean over D_{train}: $\widehat{\mu} = 102.77$	Observed calibration data Y_i and residuals R_i										Ranked residuals R_i, small to large, and corresp Y_i: Include in conformal set if $R_i \leq \lceil (1-\alpha)(n+1) \rceil = 6$th smallest										
Rank of R_i:	1	3	4	2	10	7	9	8	6	5	1	2	3	4	5	**6**	7	8	9	10	
Absolute residuals R_i:	0.92	1.62	6.71	1.09	25.41	9.81	14.07	12.16	8.24	7.23	0.92	1.09	1.62	6.71	7.23	**8.24**	9.81	12.16	14.07	25.41	
Corresponding Y_i:	101.86	101.15	96.07	103.86	77.36	92.96	116.84	90.61	111.01	95.54	101.86	103.86	101.15	96.07	95.54	111.01	92.96	90.61	116.84	77.36	
In interval?											**TRUE**	**TRUE**	**TRUE**	**TRUE**	**TRUE**	**TRUE**	FALSE	FALSE	FALSE	FALSE	
Interval lower bound = min(Y_i in conformal set):																					**95.54**
Interval upper bound = max(Y_i in conformal set):																					**111.01**

In the first block we construct a one-sided interval. The 6th-largest residual in the calibration set is -1.62, corresponding to observation $i = 2$ (the second observation in the calibration set), $Y_2 = 101.15$. Hence our one-sided prediction set is $Y \leq 101.15$.

The second block in the table shows the construction of the two-sided interval. We are now using absolute residuals as our non-conformity score, and the 6th-largest absolute residual in the calibration set is 8.24, corresponding to observation $i = 9$ (the ninth observation in the calibration set), $Y_9 = 111.01$. But since we are constructing a two-sided interval, we have to examine all the observations in the calibration set with residuals that are smaller than this. The smallest such Y_i in the calibration set with a residual satisfying $R_i \leq 8.24$ (smaller than the 6th-largest residual) is the lower bound of our prediction interval; and the largest such Y_i in the calibration set with a residual satisfying $R_i \leq 8.24$ is the upper bound of our interval. The table shows that these are observations $Y_{10} = 95.54$ and $Y_9 = 111.01$, respectively, and so our two-sided prediction with guaranteed coverage of $\geq 50\%$ is $[95.54,\ 111.01]$.

Why does this work? Because we are still respecting exchangeability. Conditional on the training set $\mathcal{D}_{train}$, the calibration set residuals $R_i,\ i \in \mathcal{D}_{cal}$ and the residual R_{n+1} for a new observation Y_{n+1} are all exchangeable. And because the calibration data now come from the same distribution as the training data (no researcher-selected trial value enters, as in full conformal prediction), we don't have to retrain for every observation in the calibration set.

Adding predictors and learners to this setup is very straightforward. In Step 2 of the algorithm, we just use a learner $f(.)$ of our choice and train it on the observations (X_i, Y_i) in the training set $\mathcal{D}_{train}$, giving us the trained learner $\widehat{f}(.)$ to use in Step 3. The training has to be done only once, so this is computationally much faster than the full-conformal version.

2.5. Full vs. split conformal prediction: summary

Whereas the full conformal algorithm loops through every trial value and has to re-train the learner in each loop, the split conformal algorithm trains the learner once on the training set and then loops through every observation in the calibration set. The split conformal saving in computation cost come from having to train the learner only once. The price for this is twofold: first, setting aside the calibration set means the researcher loses a portion of the data that could be used for training; and second, in effect the trial grid is given to the researcher the calibration set given by the random split in the first step. But as long as the dataset is large enough, these will not be significant problems, and split conformal prediction usually will be preferable in practice to full conformal prediction.

3. Full and Split Conformal Inference: A Formal Presentation

3.1. Notation

We now introduce notation for the general case that will be used throughout the rest of the paper. We consider some dataset $D = \{(X_i, Y_i)\}_{i=1}^n$, where each $Y_i \in \mathcal{Y}$ is an outcome variable and $X_i \in \mathcal{X}$ is a corresponding vector of features (predictors). Unless stated otherwise, we assume the sequence (X_i, Y_i) for $i = 1, \ldots, n$ is an exchangeable sequence. We use $\mathcal{A} : \mathcal{D} \to \mathcal{F}$ to denote a selection procedure that uses a dataset to select some prediction function $\widehat{f} \in \mathcal{F}$.

The score function is notated by $s(x, y; D)$, with the score given by $R_i := s(X_i, Y_i; D)$. A prediction interval is produced by a function $\widehat{C} : \mathcal{X} \to \{[a, b] \subseteq \mathbb{R} \mid a \leq b\}$. Additionally, we will typically suppress the dependence on D and write $R_i := s(X_i, Y_i)$ to denote the conformity score for the i-th observation.

3.2. The Conformal Score Function

A conformal score function $s : \mathcal{X} \times \mathcal{Y} \times \mathcal{D} \to \mathbb{R}$ measures the nonconformity of a data point (x, y) with respect to the data-set $D \in \mathcal{D}$ (Vladimir Vovk, Alexander Gammerman, and Shafer, 2005). The higher the score, the less "conforming" the data point is considered to be.

The choice of conformity score is flexible and can be tailored to specific problems, which we will see in Section 5.3, for example. A common choice for regression problems is the absolute residual:

$$s(x, y) = |y - \widehat{f}(x)|$$

where $\widehat{f}(x)$ is a prediction for y given x, obtained from a fitted model. However, other choices are possible and may be more suitable depending on the context. For instance, if we are interested in constructing one-sided prediction intervals, we might use the signed residual:

$$s(x, y) = y - \widehat{f}(x)$$

In general, the only requirement for a conformity score is that larger scores should reflect greater uncertainty or disagreement between our predictions and the observed outcomes. This flexibility in choosing the conformity score is one of the key strengths of the conformal prediction framework.

3.3. Full and Split Conformal Inference

We now formally define and generalize the full and split conformal inference algorithms and discuss some of their key properties.

In **Full Conformal Prediction** we consider a sequence of "trial values" for the outcome variable Y_{n+1} associated with a new input $X_{n+1} = x$. For each trial value y, we augment the original dataset $D = \{(X_i, Y_i)\}_{i=1}^n$ with the hypothetical data point (x, y), creating an augmented dataset $D_{(x,y)} = D \cup \{(x, y)\}$. We then train a new model on this augmented dataset and compute the conformity score of (x, y) with respect to this model. This process is repeated for every trial value in a chosen set, effectively exploring the entire space of possible outcomes.

Algorithm 1: Full Conformal Prediction Input:

- Data (X_i, Y_i), $i = 1, \ldots, n$
- Target miscoverage level $\alpha \in (0, 1)$
- Regression algorithm $\mathcal{A}$
- A new input x
- A set of trial values $\mathcal{Y}_{trial}$

Output: Prediction set $C(x) \subseteq \mathcal{Y}$

Procedure: Initialize an empty prediction set: $C(x) \leftarrow \emptyset$.
Then, for each trial value $y \in \mathcal{Y}_{trial}$:

Step 1: Augment the dataset with the trial point: $D_{(x,y)} \leftarrow \{(X_1, Y_1), \ldots, (X_n, Y_n), (x, y)\}$.

Step 2: Train the regression algorithm $\mathcal{A}$ on the augmented dataset $D_{(x,y)}$ to obtain the fitted model $\widehat{f}_{(x,y)} = \mathcal{A}(D_{(x,y)})$.

Step 3: Calculate the conformity scores for all data points in the augmented set, including the trial point: $\widehat{R}_{i,(x,y)} = s(X_i, Y_i)$ for $i = 1, ..., n$ and $\widehat{R}_{n+1,(x,y)} = s(x, y)$ using the model $\widehat{f}_{(x,y)}$.

Step 4: Compute the rank of the trial value's conformity score: $\text{Rank}(y) = \sum_{i=1}^{n+1} \mathbb{1}\{\widehat{R}_{i,(x,y)} \leq \widehat{R}_{n+1,(x,y)}\}$.

Step 5: **If** $\text{Rank}(y) \leq \lceil (1-\alpha)(n+1) \rceil$ **then**

- Add the trial value y to the prediction set: $C(x) \leftarrow C(x) \cup \{y\}$.

Return the prediction set $C(x)$.

An alternative but equivalent way to think of the set is through quantiles rather than ranks, in which case we could write the set as

$$\widehat{C}(x) = \{y : R_{n+1}^y \leq \text{Quantile}\left(\sum_{i=1}^{n+1} \delta_{R_i^y};\ (1-\alpha)(n+1)/n\right)\},$$

where $\delta_{R_i^y}$ represents a point mass at the point R_i^y.

The following theorem establishes the validity of the prediction set constructed by the full conformal prediction algorithm:

THEOREM 1 (Full Conformal Prediction Coverage). *If the data points (X_i, Y_i), $i = 1, \ldots, n$ and (X_{n+1}, Y_{n+1}) are exchangeable, and the prediction set $C(x)$ is constructed as in Algorithm 2 for $X_{n+1} = x$, then:*

$$\mathbb{P}(Y_{n+1} \in C(X_{n+1})) \geq 1 - \alpha.$$

Proof (intuition): The key idea is that, under the exchangeability assumption, when we augment the dataset with a trial value y that is equal to the true outcome Y_{n+1}, the augmented dataset remains exchangeable. Consequently, the rank of the trial value's conformity score is uniformly distributed among the ranks of all data points in the augmented set. This implies that the probability of the trial value's score being among the $\lceil (1-\alpha)(n+1) \rceil$ smallest scores is at least $1 - \alpha$. Therefore, the prediction set, which includes all trial values satisfying this condition, will contain the true outcome Y_{n+1} with probability at least $1 - \alpha$.

The **Split Conformal Prediction** procedure, originally proposed by Papadopoulos et al. (2002), proceeds as follows:

Algorithm 1: Split Conformal Prediction

Input:

- Data (X_i, Y_i), $i = 1, \ldots, n$
- Target miscoverage level $\alpha \in (0, 1)$
- Regression algorithm $\mathcal{A}$

Output: Prediction interval $\widehat{C}(\cdot)$

Procedure:

Step 1: Randomly split the data into two disjoint sets $\mathcal{I}_{\text{train}}$ and $\mathcal{I}_{\text{cal}}$ of sizes n_0 and n_1, respectively, such that $n_0 + n_1 = n$.

Step 2: Train the regression algorithm $\mathcal{A}$ on $\mathcal{I}_{\text{train}}$ to obtain the fitted model:

$$\widehat{f} = \mathcal{A}\left(\{(X_i, Y_i) : i \in \mathcal{I}_{\text{train}}\}\right).$$

Step 3: Calculate the conformity scores for the calibration set:

$$\widehat{R}_i = s(X_i, Y_i) \quad \text{for } i \in \mathcal{I}_{\text{cal}}.$$

Step 4: Determine the threshold $\widehat{t}_{n_1}$ as the $\lceil (1-\alpha)(n_1+1) \rceil$-th smallest value in $\{\widehat{R}_i : i \in \mathcal{I}_{\text{cal}}\}$. In other words, $\widehat{t}_{n_1}$ is the $(1-\alpha)$ empirical quantile of the calibration scores, adjusted for finite sample size.

Step 5: Construct the prediction interval for the new input x:

$$\widehat{C}(x) = [\widehat{f}_{n_0}(x) - \widehat{t}_{n_1}, \widehat{f}_{n_0}(x) + \widehat{t}_{n_1}].$$

Return the prediction interval $\widehat{C}(\cdot)$.

The theoretical guarantee of split conformal prediction is given by the following theorem:

THEOREM 2 (Split Conformal Prediction Coverage (From Lei et al. (2018))). *If the data points (X_i, Y_i), $i = 1, \ldots, n$ are exchangeable and $\widehat{C}$ is constructed as in Algorithm 1, then for a new, independent data point (X_{n+1}, Y_{n+1}) also exchangeable with the previous data, we have:*

$$\mathbb{P}(Y_{n+1} \in \widehat{C}(X_{n+1})) \geq 1 - \alpha.$$

Proof (intuition): Under the exchangeability assumption, the conformity score for the new observation, $\widehat{R}_{n+1} = s(X_{n+1}, Y_{n+1})$, is exchangeable with the conformity scores in the calibration set. Therefore, the probability that $\widehat{R}_{n+1}$ is less than or equal to the $\lceil (1-\alpha)(n_1+1) \rceil$-th smallest value in the calibration set is at least $1-\alpha$. Since the prediction interval is constructed such that it contains all points with a conformity score less than or equal to $\widehat{t}_{n_1}$, the coverage guarantee follows.

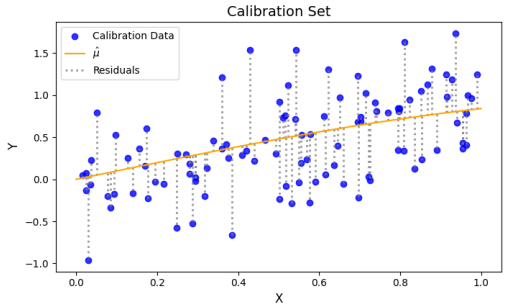
These steps are outlined in the Figure 1 for estimating a 75% prediction interval. In plot A, we estimate the conditional mean of Y given X with the function $\widehat{\mu}(\cdot)$ using the training data. In plot B, we use the calibration data to compute the residuals (dotted lines). Plot C plots a histogram of the absolute residuals and marks the 75th percentile $\widehat{q}$. Finally, in plot D, we construct our prediction set by adding and subtracting $\widehat{q}$ from the estimated conditional expectation.

The above theorems for split and full conformal inference imply:

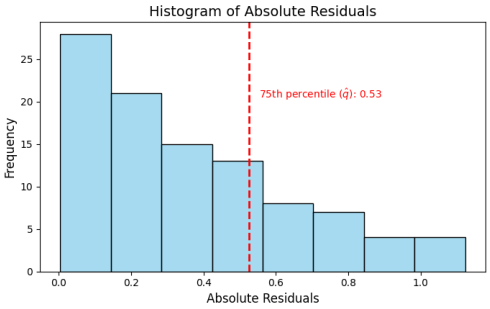
- **Distribution-free coverage guarantee**: The coverage guarantee holds regardless of the underlying distribution of (Y, X), providing sharp bounds on the coverage guarantees under the exchangeability assumption.
- **Finite-sample validity:** The guarantee holds for any sample size n, not just asymptotically.
- **Flexibility:** The choice of the base regression algorithm $\mathcal{A}$ and the conformity score is flexible. Indeed, even if the estimator is severely biased or a poor predictor, full or split conformal prediction will still produce valid predictive confidence intervals.

Under the further mild assumption that the distribution of the residuals is continuous, we get the following sharp bound

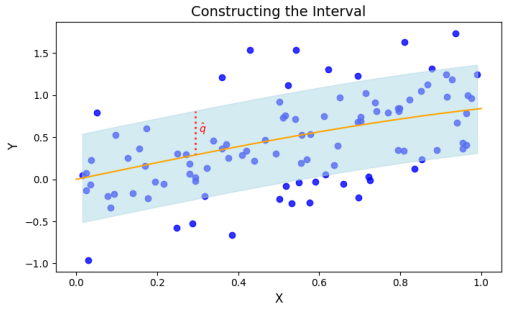
A. Estimate prediction function on training set



B. Calculate residuals on calibration set



C. Calculate $(1-\alpha)$ quantile of residuals



D. Construct Prediction Interval

FIGURE 1. Split Conformal Prediction Steps

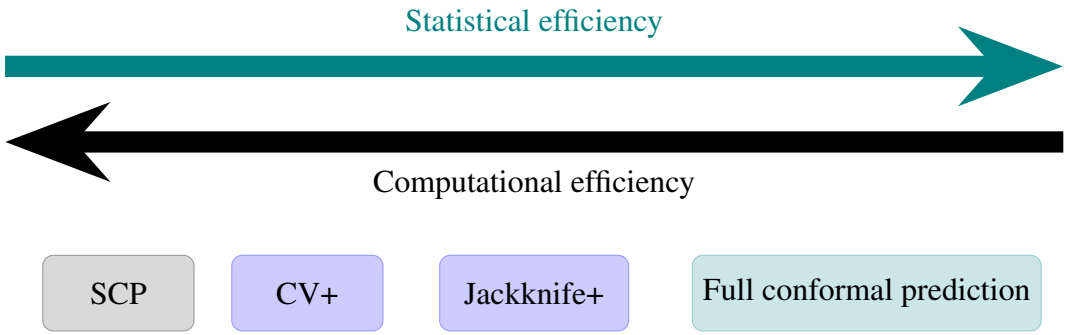
Theorem 2 (Cont.) (Lei et al., 2018) Assuming the residuals $R_i, i \in \mathcal{I}_{cal}$ have a continuous joint distribution, then

$$\mathbb{P}(Y_{n+1} \in \widehat{C}(X_{n+1})) \leq 1-\alpha+\frac{2}{n+2}$$

For split conformal inference, splitting the data equally between the training calibration set is unnecessary. We may choose different size sets if we want to trade off better estimation of the quantile of errors versus better estimation of the trained estimator. The choice of split will depend on the context of the situation and researcher discretion.

4. Extensions

One benefit of split conformal inference is that we only require a single data split and estimation of the model once. However, we sacrifice statistical efficiency due to using less data to estimate both the prediction function and the distribution of the scores. Full conformal prediction, on the other hand, avoids data splitting and fully uses the available data for both estimation and inference. But it is very computationally intensive, as we have to re-estimate the model many times, either for each new X we wish to perform inference on, or for a whole grid across the domain of $\mathcal{X} \times \mathcal{Y}$.



In the following sections, we discuss alternative procedures to better balance the trade-off between computational efficiency and statistical efficiency. These are the Jackknife+ and CV+ algorithms.

4.1. The Jackknife

A first approach might be to use the jackknife method to estimate the residuals for each point out of sample. The idea would be to use the leave-one-out (LOO) residual to estimate the prediction interval and the resulting distribution of the score function.

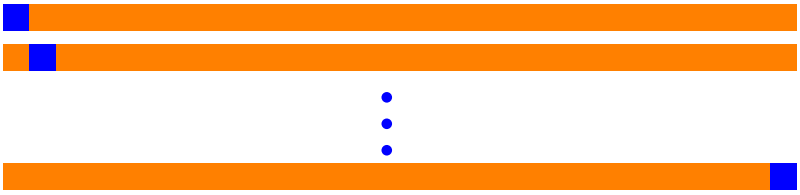


FIGURE 2. Viualisation of the jackknife procedure. Each row represents the data-set with the leave-one-out observation highlighted in blue.

The algorithm is given by:

For each i:

- $D_{-i} = D \setminus (X_i, Y_i)$
- Train $\widehat{f}_{(-i)} = \mathcal{A}(D_{-i})$
- Estimate the score for i: $R_i = s_{(-i)}(X_i, Y_i)$

We would then construct our interval as

$$\widehat{C}_{JK}(Xn+1) = [\widehat{f}(X_{n+1}) \pm \widehat{q}_{1-\alpha}(\{R_i\})]$$

similar to the approach used in split conformal prediction. However, this method does not provide a specific coverage guarantee. Each $\widehat{f}_{(-i)}$ is trained on a dataset that omits a single

data point. If the removal of that single point leads to a substantially different model, then the estimated residuals (and consequently the estimated quantiles $\widehat{q}_{1-\alpha}$) may not accurately reflect the true error distribution of the final model $\widehat{f}$ trained on the full dataset. Consequently, we can no longer construct a valid prediction set with a coverage guarantee (Lei et al., 2018).

However, with a small modification of the underlying algorithm, and how we treat the quantiles, we are able to produce a valid interval.

4.2. Jackknife+

The jackknife+ algorithm (Barber et al., 2021) accounts for variability in the fitted regression function and provides coverage guarantees only under the exchangeability assumption. Before continuing, we need to introduce additional notation as the jackkife+ algorithm treats the quantiles slightly differently from the previous methods. For any values $v_i \in \mathbb{R}$ indexed by $i = 1, \ldots, n$, let

$$\widehat{q}^{+}_{n,\alpha}\{v_i\} = \lceil (1-\alpha)(n+1) \rceil\text{-th smallest quantile of } \{v_i\}$$

and

$$\hat{q}^{-}_{n,\alpha}\{v_i\} = \lfloor \alpha(n+1) \rfloor\text{-th smallest quantile of } \{v_i\}.$$

Note that the jackknife conformal prediction interval described above can be written as $\widehat{C}^{jackknife}_{n,\alpha}(X_{n+1}) = \widehat{f}_n(X_{n+1}) \pm \widehat{q}^{+}_{n,\alpha}\{R^{LOO}_i\}$, where $R^{LOO}_i = s_{(-i)}(X_i, Y_i)$ is the leave-one-out conformity score.

The jackknife+ algorithm works by recentring our interval on the leave-one-out prediction for X_{n+1}. Our prediction set is then given by:

$$\widehat{C}^{jackknife+}_{n,\alpha}(X_{n+1}) = [\hat{q}^{-}_{n,\alpha}\{\, \widehat{f}_{(-i)}(X_{n+1}) - R^{LOO}_i \,\},\ \widehat{q}^{+}_{n,\alpha}\{\, \widehat{f}_{(-i)}(X_{n+1}) + R^{LOO}_i \,\}].$$

If $\widehat{f}_{(-i)}$ and $\widehat{f}_n$ are similar, then the jackknife and jackknife+ algorithms will produce very similar confidence sets. However, when this is not the case, the prediction sets may be quite different, and the jackknife algorithm may not produce a prediction interval with desirable coverage properties.

Interestingly, the coverage guarantee of the jackknife+ algorithm is at the level $1-2\alpha$ (rather than $1-\alpha$)[5] If we want to be certain we get a nominal coverage rate of $1-\alpha$, we can use the jackknife-minimax (Barber et al., 2021) method, where the prediction interval is given by

$$\widehat{C}^{jackknife-minmax}_{n,\alpha}(X_{n+1}) = [\min_i \widehat{f}_{(-i)}(X_{n+1}) - \widehat{q}^{+}_{n,\alpha}\{R^{LOO}_i\},\ \max_i \widehat{f}_{(-i)}(X_{n+1}) + \widehat{q}^{+}_{n,\alpha}\{R^{LOO}_i\}].$$

[5] Although, it will often be close to $1-\alpha$ in practice.

The jackknife-minimax method is more conservative than the jackknife+ method and will always lead to a prediction set that is weakly larger than the size of the set produced by the jackknife algorithm.

4.2.1. Interpreting the interval

The Jackknife+ algorithm is unique in that the interval is no longer symmetric around the point prediction. There is even no guarantee that the prediction will lie within the confidence interval in extreme cases, which complicates interpreting the interval as an uncertainty estimate around our prediction. However, we can interpret this interval as an uncertainty estimate around the median $\widehat{f}_{(-i)}$ from the ensemble of predictions (Angelopoulos, Barber, and Bates, 2024).

4.2.2. Why is jackknife+ coverage $1 - 2\alpha$?

The fact that the coverage guarantee now holds for $1 - 2\alpha$ is an interesting property when performing inference or statistical tests across multiple splits of data, and can also be seen in traditional hypothesis testing settings (e.g. see C. J. DiCiccio, T. J. DiCiccio, and J. P. Romano (2020)).

The reason for the 2α factor is due to potential inconsistencies across the models when trained on different subsets of the data. A transitive ranking of numbers implies that if $A > B$ and $B > C$, then $A > C$. However, the jackknife+ (and more generally, cross-conformal methods) violate transitivity as it constructs its intervals via looking at a collection of models, each trained on a slightly different subset of the data. When we compare residuals across these different models, transitivity can break down. For example,

- Residual A (from model 1)> Residual B (from model 2)
- Residual B (from model 2) > Residual C (from model 3)
- Residual C (from model 3) > Residual A (from model 2)

Recall how standard conformal prediction achieves $1 - \alpha$ coverage. With a single model, we:

Step 1: Calculate residuals $R_i = |Y_i - \widehat{f}(X_i)|$

Step 2: Find a threshold q where $P(R_i > q) \leq \alpha$

Step 3: Construct interval $\widehat{f}(X_{n+1}) \pm q$

This works because we have a single, consistent ranking of residuals - if a point is in the top α of largest residuals, it's equally "extreme" for both the upper and lower bounds of our interval.

Jackknife+ operates differently. For each point i:

Step 1: We train a model without that point: $\widehat{f}_{(-i)}$

Step 2: Calculate its LOO residual: $R_i^{LOO} = |Y_i - \widehat{f}_{(-i)}(X_i)|$

Step 3: Construct intervals using quantiles of $\{\widehat{f}_{(-i)}(X_{n+1}) \pm R_i^{LOO}\}$

The difference is that we're now using different models to determine the upper and lower bounds. This means:

- Point A might have a large residual under model 1 but small under model 2
- Point B might have a large residual under model 2 but small under model 1

In the worst case:

- The α fraction of points determining the upper bound could be completely different from
- The α fraction of points determining the lower bound

...which leads to a total of 2α fraction of points potentially falling outside the interval.

This is why we can only guarantee $1-2\alpha$ coverage, though in practice the coverage is often much closer to $1-\alpha$ as such extreme reorderings are rare.

This also provides intuition for why the minmax version of Jackknife+ covers $1-\alpha$ as we are restoring transitivity by using the same threshold of the residuals for the upper and lower bounds.

4.3. CV+

CV+ or "cross-validation-plus" is very similar to the jackknife+ algorithm but works on a block structure rather than leaving a single observation out (Barber et al., 2021).

To start, split the training set into K disjoint subsets $S_1, \ldots, S_K$, each with size $m = \frac{n}{k}$.[6] Denote by $\widehat{f}_{(-s_k)}$ the prediction function estimated using the dataset excluding S_k for $k = 1, \ldots, K$. The out-of-fold conformity score is given by

$$R_i^{CV} = | Y_i - \widehat{f}_{(-S_{k(i)})}(X_i) |,$$

where $k(i)$ identifies the set in which observation i belongs. The *CV+* prediction interval is then defined as

$$\widehat{C}_{n,\alpha}^{CV+}(X_{n+1}) = [\hat{q}_{n,\alpha}^{-}\{ \widehat{f}_{(-S_{k(i)})}(X_{n+1}) - R_i^{CV} \},\ \widehat{q}_{n,\alpha}^{+}\{ \widehat{f}_{(-S_{k(i)})}(X_{n+1}) + R_i^{CV} \}].$$

We can see that the CV+ algorithm is the same as the jackknife+ algorithm when $K = n$.

[6] We assume without loss of generality that m is an integer.

4.4. Summary

We summarise in Table 4 the guarantees of the methods discussed so far:

Method	Coverage Guarantee	Typical Coverage
Naive	None	$< 1-\alpha$
Split	$1-\alpha$	$\approx 1-\alpha$
Full	$1-\alpha$	$\approx 1-\alpha$ or $> 1-\alpha$ if $\widehat{f}(x)$ is a poor predictor
Jackknife	None	$\approx 1-\alpha$ or $< 1-\alpha$ if $\widehat{f}(x)$ is unstable
Jackknife+	$1-2\alpha$	$\approx 1-\alpha$
Jackknife-minmax	$1-\alpha$	$> 1-\alpha$
K-fold CV+	$1-2\alpha$	$\gtrapprox 1-\alpha$

TABLE 4. Summary of coverage guarantees and typical coverage for different methods. Adapted from Barber et al. (2021)

5. Conditional Coverage

One key consideration in conformal inference under the assumption of exchangeability is conditional validity. The methods discussed so far primarily provide marginal coverage guarantees in finite samples — that is, on average, the prediction interval is expected to contain the true outcome with probability $1-\alpha$ across the joint distribution of $\mathcal{X} \times \mathcal{Y}$.

Formally, we say a prediction set set $\widehat{C}$ satisfies distribution-free conditional coverage at significance level $1-\alpha$ if

$$\mathbb{P}_P\{Y_{n+1} \in \widehat{C}(X_{n+1}) \mid X_{n+1} = x\} \geq 1-\alpha$$

for all distributions P and all measurable $x \in \mathcal{X}$. This is a much stronger requirement than marginal coverage, as it demands that the coverage probability holds for each specific value of the predictors, not just on average.

5.1. Impossibility Results and Fundamental Limitations

To reflect on what we have discussed above, if our new data point (X_{n+1}, Y_{n+1}) is exchangeable with our data-set D, then the rank of R_{n+1} among the points $\{R_i\}_{i \in D} \bigcup \{R_{n+1}\}$ will be distributed uniformly, thus providing us with marginal coverage.

To understand why this approach does not translate to conditional coverage, consider how we use the empirical distribution of the conformity scores to set our thresholds, which

we can denote by $\widehat{F}(R_i)$. While we construct our prediction intervals using the marginal distribution of residuals $\widehat{F}(R_i)$, what is relevant for conditional coverage is the conditional CDF $F(R_i \mid X_i = x)$. When these distributions differ, using quantiles from $\widehat{F}(R_i)$ to set our thresholds will result in inconsistent coverage rates, even though marginal coverage is maintained.

In fact, if $F(R_i \mid X_i = x) = F(R_i)$ for all x – that is, the distribution of the conformity score is independent of X_i – it would be a sufficient condition for perfect conditional coverage. However, this independence condition is strong and rarely holds in practice.

This is formalized in the following impossibility result. Note that in the proposition below, $\widehat{C}$ refers to *any* distribution free prediction set that satisfies conditional coverage, not only intervals from conformal prediction.

PROPOSITION 1 (From Lei et al. (2018) and Foygel Barber et al. (2021)). *Suppose a prediction set $\widehat{C}(\cdot)$ satisfies the distribution-free conditonal coverage property. Then for all distributions P, we have*

$$\mathbb{E}[\mathit{length}(\widehat{C}(x))] = \infty,$$

at almost all points x aside from the atoms of P_X[7]. See A2 in (Lei and Wasserman, 2014) for the full proof.

Proposition 1 tells us that any method claiming to provide exact conditional coverage must necessarily produce intervals with infinite length in expectations, making the coverage guarantee useless in practice.

While we cannot guarantee point-wise coverage for $x \in \mathcal{X}$, we can find some middle ground between the two extremes of marginal coverage and conditional coverage, known as approximate conditional coverage (Foygel Barber et al., 2021). The first two methods discussed (normalized conformal prediction and conformal quantile regression) produce adaptive prediction sets (sets which vary in length) but don't provide any guarantees of conditional coverage in finite samples, but never nevertheless improve measures of conditional coverage in practice. However, the third method (localized conformal prediction) does come with approximate conditional coverage guarantees.

5.2. Normalized Conformal Prediction

One straightforward way to create adaptive prediction intervals that can improve conditional coverage and better reflect local uncertainty is to scale the conformity scores by a measure of how "challenging" the test point is to predict. Normalized conformal prediction adjusts the width of the prediction intervals based on the local difficulty of the prediction task (Papadopoulos, Alex Gammerman, and Volodya Vovk, 2008).

[7] Atoms are points of a distribution with positive probability mass.

The key idea is to modify our conformity score by incorporating a local scaling factor $\widehat{\rho}(X_i)$:

$$R_i = \frac{s(X_i, Y_i)}{\widehat{\rho}(X_i)}.$$

The scaling factor can take various forms, including:

- Estimated conditional standard deviation:

$$\widehat{\rho}(X_i) = \sqrt{\widehat{\mathbb{E}}[(Y - \widehat{\mu}(X))^2 \mid X = X_i]}$$

- Model-based uncertainty:

$$\widehat{\rho}(X_i) = \sqrt{\widehat{V}(\widehat{\mu}(X_i))},$$

  where $\widehat{V}(\cdot)$ estimates prediction variance.

- Local variance estimation:

$$\widehat{\rho}(X_i) = \sqrt{\frac{\sum_{j \in \mathcal{N}_k(i)} (Y_j - \bar{Y}_{\mathcal{N}_k(i)})^2}{k}},$$

  where $\mathcal{N}_k(i)$ denotes the k-nearest neighbors of X_i.

For a new observation X_{n+1}, the prediction interval is:

$$\widehat{C}(X_{n+1}) = [\widehat{\mu}_{n_0}(X_{n+1}) - \widehat{q}\widehat{\sigma}(X_{n+1}), \widehat{\mu}_{n_0}(X_{n+1}) + \widehat{q}\widehat{\sigma}(X_{n+1})]$$

where $\widehat{q}$ is the $(1-\alpha)(1+1/n)$ quantile of the normalized residuals $\{R_i\}_{i=1}^n$.

While this approach doesn't guarantee conditional coverage, it often works well in practice, particularly when the scaling factor accurately captures the true variation in prediction difficulty across the feature space.

5.2.1. Example

Consider the simple regression model

$$Y_i = \beta X_i + X_i e_i$$

where $X_i \sim U(0, 5)$ and $e_i \sim N(0, 1)$, which is plotted in the Figure 3. As we can see, the spread of the data points increase as X increases. Using the split conformal inference algorithm (left diagram), the prediction intervals are constant along the domain of X, not capturing the heteroskedasticity in the error term. This results in areas with with perfect coverage, and areas with less than 95% coverage.

In contrast, the figure on the right illustrates the application of normalized conformal prediction, which adjusts the prediction intervals based on the local difficulty of the prediction task. Here, the conformity scores are modified by a scaling factor $\widehat{\rho}(X_i)$, defined as an estimate of the conditional standard deviation of the error term.

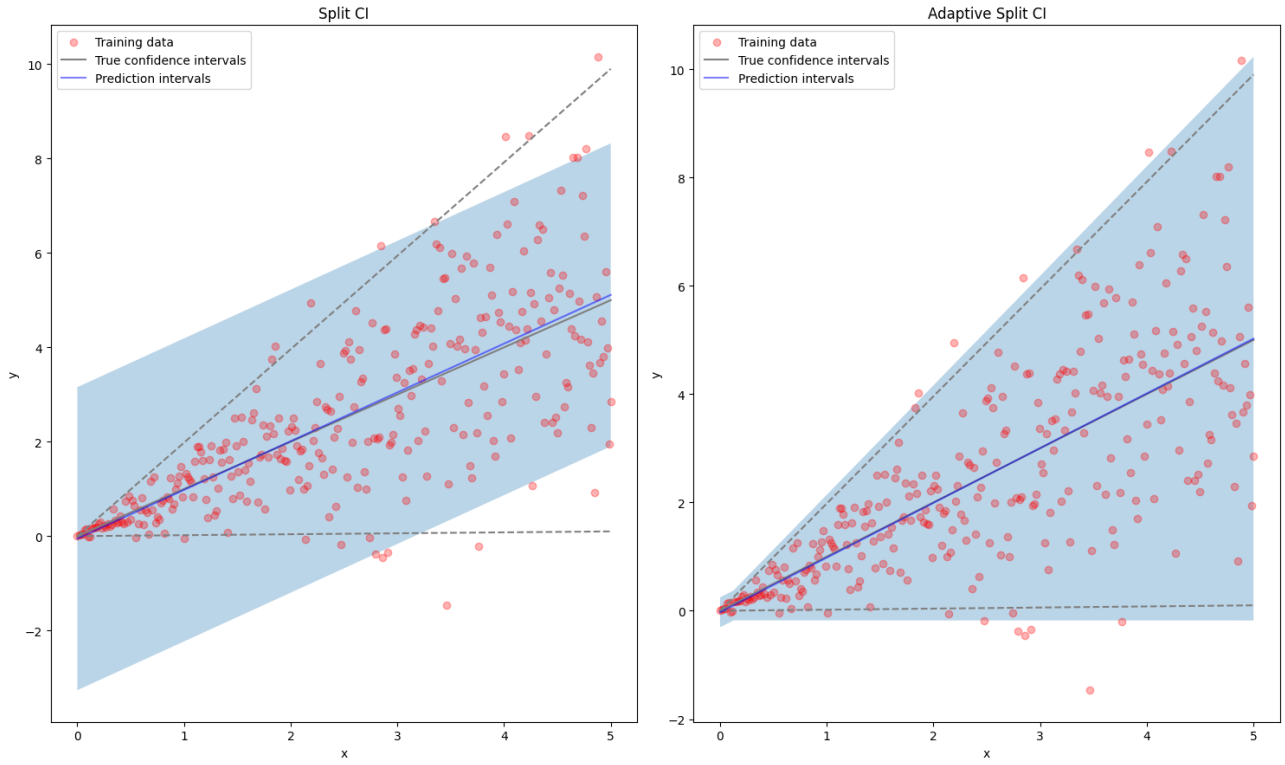


FIGURE 3. Plots of split conformal inference (left) and normalized conformal inference (right) for a DGP with a heteroskedastic error term.

5.3. Conformal Quantile Regression

Conformal quantile regression (CQR) provides a direct approach to handling heterogeneity in the conditional distribution of $Y \mid X$ through a modification of the score function (Y. Romano, Patterson, and Candes, 2019).

Let $\tau_{\text{low}} = \alpha/2$ and $\tau_{\text{high}} = 1 - \alpha/2$. The method proceeds in two steps:

Step 1: Estimate conditional quantile functions $\widehat{q}_{\tau_{\text{low}}}(x)$ and $\widehat{q}_{\tau_{\text{high}}}(x)$ using any consistent quantile regression method. Common specifications include:

- Linear quantile regression:

$$\widehat{q}_\tau(x) = x'\widehat{\beta}_\tau = \arg\min_\beta \sum_{i=1}^{n} \rho_\tau(Y_i - x_i'\beta)$$

  where $\rho_\tau(u) = u(\tau - \mathbb{1}\{u < 0\})$.

- Nonparametric quantile regression:

$$\widehat{q}_\tau(x) = \arg\min_{q \in \mathcal{F}} \sum_{i=1}^{n} \rho_\tau(Y_i - q(x_i))$$

where $\mathcal{F}$ is a flexible function class (e.g., neural networks, random forests).

Step 2: Construct conformity scores for each observation in the calibration set:

$$E_i^{\text{low}} = Y_i - \widehat{q}_{\tau_{\text{low}}}(X_i)$$

$$E_i^{\text{high}} = \widehat{q}_{\tau_{\text{high}}}(X_i) - Y_i$$

The final prediction interval for a new point X_{n+1} is:

$$\widehat{C}(X_{n+1}) = [\widehat{q}_{\tau_{\text{low}}}(X_{n+1}) - \widehat{Q}_{1-\alpha}(E^{\text{low}}), \widehat{q}_{\tau_{\text{high}}}(X_{n+1}) + \widehat{Q}_{1-\alpha}(E^{\text{high}})]$$

where $\widehat{Q}_{1-\alpha}(\cdot)$ is the $(1-\alpha)(1+1/n)$ empirical quantile.

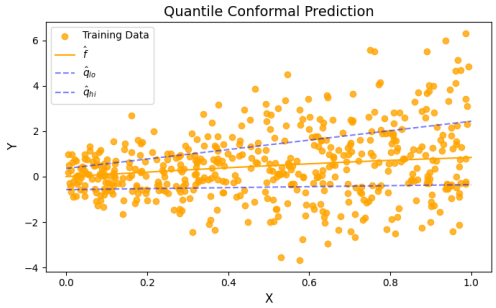
Several theoretical properties are worth noting:

- The method provides valid marginal coverage regardless of the consistency of the quantile estimates:
$$\mathbb{P}(Y_{n+1} \in \widehat{C}(X_{n+1})) \geq 1 - \alpha$$
- Under correct specification of the conditional quantiles, the intervals are asymptotically optimal in terms of expected length:
$$\lim_{n\to\infty} \mathbb{E}[\text{length}(\widehat{C}(X))] = \mathbb{B}[q_{\tau_{\text{high}}}(X) - q_{\tau_{\text{low}}}(X)]$$
- The method is robust to heteroskedasticity and other forms of conditional heterogeneity without requiring explicit modeling of the conditional variance.

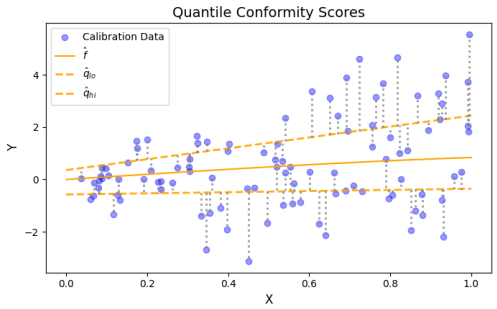
Figure 4 outlines the procedure of CQR for an $\alpha = 0.25$. In plot A, using the training data we estimate the conditional expectation $\widehat{f}$ along with the 12.50th and 87.50th quantiles using quantile regression. Then using the calibration data, we calculate the residuals according to the CQR score function. Plot C plots a histrogram of the conformity scores and marks the 75th percentile $\widehat{q}$. Finally, we construct the interval in plot 6 by adding $\widehat{q}$ to the upper quantile, and substracting $\widehat{q}$ from the lower quantile.

5.4. Weighted Conformal Prediction

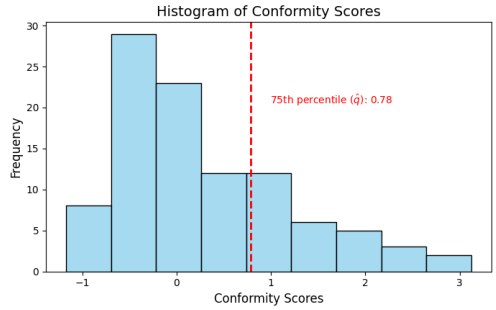
Weighted conformal prediction offers a way to improve conditional coverage by prioritising local information through a weighting scheme, aiming to improve conditional coverage. The idea is to assign higher weights to observations based on their similarity to the test point of interest. This localization is achieved using a kernel function $H : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_{\geq 0}$ that quantifies the proximity between feature vectors. A common choice for H is the Gaussian kernel: $H(x, x') = \exp(-\|x - x'\|^2/2h^2)$, where h is a bandwidth parameter controlling the extent of localization.



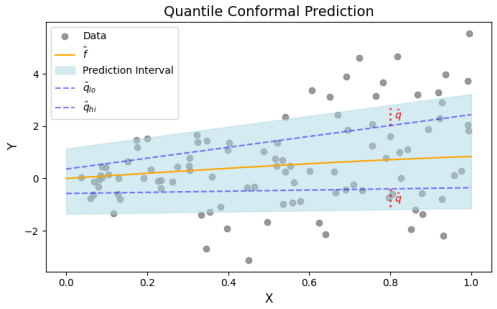
A. Estimate prediction and quantile functions on training set



B. Calculate residuals on calibration set



C. Calculate $(1-\alpha)$ quantile of residuals



D. Construct Prediction Interval

FIGURE 4. Steps to implement Conformal Quantile Regression

5.4.1. Localized Conformal Prediction

A naive approach would be to construct the prediction set as

$$\widehat{C}(X_{n+1}) = \left\{ y \in \mathcal{Y} : R^y_{n+1} \leq \text{Quantile}\left(\sum_{i=1}^{n+1} w_i \delta_{R^y_i}; 1-\alpha \right) \right\}$$

where $w_i = \frac{H(X_i, X_{n+1})}{\sum_{j=1}^{n+1} H(X_j, X_{n+1})}$ are weights based on the proximity to the test point X_{n+1} and R^y_i are the conformity scores. However, this does not guarantee valid coverage.

To address this, localized conformal prediction uses a recalibration step that ensures valid marginal coverage while maintaining some degree of local adaptivity (Guan, 2023). Instead of directly using the weighted empirical quantile, it adapts the threshold based on a weighted quantile of modified conformity scores.

Algorithm for Localized Conformal Prediction:

Input: Training data $\{(X_1, Y_1), \ldots, (X_n, Y_n)\}$, test point X_{n+1}, significance level α, grid of trial values Y_{grid}, and localization kernel H.

Steps:

For each $y \in Y_{grid}$:

Step 1: Compute Conformity Scores and Weights:

$$R_i^y = s^y(X_i, Y_i) \quad \text{for } i = 1, \ldots, n+1$$

$$w_{i,j} = H(X_j, X_i) / \sum_{j'=1}^{n+1} H(X_{j'}, X_i) \text{ for } i, j = 1, \ldots, n+1$$

for $i = 1, \ldots, n+1$.

Step 2: For each i, compute weighted conformity scores:

$$\tilde{R}_i^y = \sum_{j=1}^{n+1} w_{i,j} \mathbb{1}\{R_j^y < R_i^y\}$$

Step 3: Calculate the quantile of the weighted conformity scores:

$$\tilde{q}^y = \text{Quantile}(\tilde{R}_1^y, \ldots, \tilde{R}_{n+1}^y; 1-\alpha)$$

Return the prediction set:

$$\widehat{C}(X_{n+1}) = \{y \in \mathcal{Y} : R_{n+1}^y \leq \tilde{q}^y\}$$

The difference from the naive approach is using the data-dependent threshold $\tilde{q}^y$ instead of a fixed $1-\alpha$ quantile.

5.4.2. Theoretical Considerations and Randomly-Localized Conformal Prediction

While localized conformal prediction offers improved conditional coverage in practice, we can only guarantee marginal coverage. A variation called **randomly-localized conformal prediction** provides a way forward in guaranteeing approximate conditional coverage (Hore and Barber, 2024).

Randomly-Localized Conformal Prediction:

This method introduces a randomization step, sampling a point $\tilde{X}_{n+1}$ from a distribution centered around X_{n+1} (using the kernel H as the density[8]). Weights and quantiles are then calculated based on this sampled point.

Algorithm:

Inputs: Training data $\{(X_i, Y_i)\}_{i=1}^n$, test point X_{n+1}, significance level α, localization kernel H, and score function s.

Steps:

Sample $\tilde{X}_{n+1}$ from a distribution with density $H(X_{n+1}, \cdot)$.

Then, for each $y \in Y_{grid}$:

[8]This requires that our kernel function satisfies $\int_{\mathcal{X}} H(x, x')d\nu(x') = 1$ with respect to some density measure ν

Step 1: Compute Scores and Weights:

$$R_i^y = s^y(X_i, Y_i)$$
$$w_i = H(X_i, \tilde{X}_{n+1}) / \sum_{j=1}^{n+1} H(X_j, \tilde{X}_{n+1})$$

for $i = 1, \ldots, n+1$.

Step 2: Calculate the $1-\alpha$ quantile of the weighted conformity scores:

$$\tilde{q}^y = \text{Quantile}\left(\sum_{i=1}^{n+1} w_i \delta_{R_i^y}; 1-\alpha\right)$$

Return the prediction set:

$$\widehat{C}(X_{n+1}) = \{y \in \mathcal{Y} : R_{n+1}^y \leq \tilde{q}^y\}$$

Theoretical Guarantee:

The randomization in this approach provides us with the following theoretical result:

THEOREM 3. *Suppose $\{(X_i, Y_i)\}_{i=1}^n$ are i.i.d. samples and the score function s is symmetric. Then, for any test point X_{n+1}, the prediction set $\widehat{C}(X_{n+1})$ produced by the randomly-localized conformal prediction method satisfies*

$$P(Y_{n+1} \in \widehat{C}(X_{n+1}) \mid \tilde{X}_{n+1}) \geq 1-\alpha \text{ almost surely.}$$

Interpretation:

This theorem provides a conditional coverage guarantee, but conditional on the *randomly sampled* point $\tilde{X}_{n+1}$, not the test point X_{n+1} itself. This can be interpreted as a guarantee that holds "near" X_{n+1} in a probabilistic sense, as $\tilde{X}_{n+1}$ is drawn from a distribution centered around it. See Angelopoulos, Barber, and Bates (2024) for a proof of this proposition.

6. Conclusion

Conformal inference is a powerful framework for constructing prediction intervals with guaranteed coverage in finite samples. We have demonstrated the basic principles of full and split conformal prediction, highlighting the key assumption of exchangeability and illustrating how to construct one-sided and two-sided prediction intervals. We have also explored extensions such as the Jackknife+ and CV+ algorithms, which offer a better balance between computational and statistical efficiency. Finally, we examined the issue of conditional

coverage, presenting methods that aim to improve conditional coverage in practice, such as normalized conformal prediction, conformal quantile regression, and weighted conformal prediction.

We have touched on only a fraction of this rapidly developing literature. There are many areas we haven't discussed where conformal inference can be used, such as in settings where exchangeability fails, classification, or traditional hypothesis testing and inference.

How limiting the exchangeability assumption is will depend on the context. For example, exchangeability rarely holds in a time-series setting, where the objective is to construct a forecast interval. Seminal works covering conformal inference in a time-series setting include Xu and Xie (2023), Chernozhukov, Wüthrich, and Yinchu (2018), and Xu and Xie (2021). Other areas where conformal inference is being actively developed include outlier detection (Bates et al., 2023), covariate and distribution shift (Tibshirani et al., 2019; Gibbs and Candes, 2021), risk control (Bates et al., 2021), conformal predictive distributions (Vladimir Vovk et al., 2019; Vladimir Vovk et al., 2018), Mondrian conformal prediction (Boström, Johansson, and Löfström, 2021), and length optimization (Kiyani, Pappas, and Hassani, 2024) among others. These are just examples of a large and rapidly-growing literature.

Software. There are a number of good software packages that can implement the methods discussed. In Python, the `MAPIE` package[9](Cordier et al., 2023) and the `crepes` package[10](Boström, 2022) provide comprehensive tools for conformal prediction. For time series forecasting, the Nixtla[11]suite of packages (Garza et al., 2022) offers implementations of conformal inference methods tailored to time series data. In Julia, the `ConformalPrediction.jl` package[12] provides a flexible and efficient framework for conformal inference. In R, there are several packages focusing on specific methods. Two worth noting in particular are `AdaptiveConformal`[13] (Susmann, Chambaz, and Josse, 2023) and `ConformalForecast`[14] (Wang and Hyndman, 2024).

The distribution-free nature and finite-sample validity of conformal inference make it a potentially attractive tool for empirical research in economics. We hope that this paper can serve as a useful starting point for economists interested in exploring and applying these methods in their own work. As the field continues to evolve, we anticipate that conformal inference will play an increasingly important role in empirical economic research, providing a robust and flexible framework for uncertainty quantification and prediction.

[9]https://mapie.readthedocs.io/en/stable/
[10]https://github.com/henrikbostrom/crepes
[11]https://nixtla.github.io/nixtla/
[12]https://github.com/JuliaTrustworthyAI/ConformalPrediction.jl
[13]https://github.com/herbps10/AdaptiveConformal
[14]https://github.com/xqnwang/conformalForecast

References

Angelopoulos, Anastasios N., Rina Foygel Barber, and Stephen Bates (Nov. 2024). *Theoretical Foundations of Conformal Prediction*. arXiv: 2411.11824. (Visited on 11/20/2024).

Barber, Rina Foygel et al. (2021). "Predictive inference with the jackknife+". In: *The Annals of Statistics* 49.1, pp. 486–507. DOI: 10.1214/20-AOS1965. URL: https://doi.org/10.1214/20-AOS1965.

Bates, Stephen et al. (2021). "Distribution-free, risk-controlling prediction sets". In: *Journal of the ACM (JACM)* 68.6, pp. 1–34.

Bates, Stephen et al. (2023). "Testing for outliers with conformal p-values". In: *The Annals of Statistics* 51.1, pp. 149–178.

Boström, Henrik (2022). "crepes: a Python Package for Generating Conformal Regressors and Predictive Systems". In: *Proceedings of the Eleventh Symposium on Conformal and Probabilistic Prediction and Applications*. Ed. by Ulf Johansson et al. Vol. 179. Proceedings of Machine Learning Research. PMLR.

Boström, Henrik, Ulf Johansson, and Tuwe Löfström (Sept. 2021). "Mondrian Conformal Predictive Distributions". In: *Proceedings of the Tenth Symposium on Conformal and Probabilistic Prediction and Applications*. PMLR, pp. 24–38. (Visited on 08/29/2024).

Chernozhukov, Victor, Kaspar Wüthrich, and Zhu Yinchu (July 2018). "Exact and Robust Conformal Inference Methods for Predictive Machine Learning with Dependent Data". In: *Proceedings of the 31st Conference On Learning Theory*. PMLR, pp. 732–749. (Visited on 01/18/2023).

Cordier, Thibault et al. (2023). "Flexible and Systematic Uncertainty Estimation with Conformal Prediction via the MAPIE library". In: *Conformal and Probabilistic Prediction with Applications*.

DiCiccio, Cyrus J, Thomas J DiCiccio, and Joseph P Romano (2020). "Exact tests via multiple data splitting". In: *Statistics & Probability Letters* 166, p. 108865.

Foygel Barber, Rina et al. (2021). "The limits of distribution-free conditional predictive inference". In: *Information and Inference: A Journal of the IMA* 10.2, pp. 455–482.

Garza, Federico et al. (2022). *StatsForecast: Lightning fast forecasting with statistical and econometric models*. PyCon Salt Lake City, Utah, US 2022. URL: https://github.com/Nixtla/statsforecast.

Gibbs, Isaac and Emmanuel Candes (2021). "Adaptive conformal inference under distribution shift". In: *Advances in Neural Information Processing Systems* 34, pp. 1660–1672.

Guan, Leying (2023). "Localized conformal prediction: A generalized inference framework for conformal prediction". In: *Biometrika* 110.1, pp. 33–50.

Hore, Rohan and Rina Foygel Barber (2024). "Conformal prediction with local weights: randomization enables robust guarantees". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology*, qkae103.

Kiyani, Shayan, George Pappas, and Hamed Hassani (Sept. 2024). *Length Optimization in Conformal Prediction*. DOI: 10.48550/arXiv.2406.18814. arXiv: 2406.18814. (Visited on 10/22/2024).

Lei, Jing and Larry Wasserman (Jan. 2014). "Distribution-Free Prediction Bands for Non-parametric Regression". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 76.1, pp. 71–96. ISSN: 1369-7412. DOI: 10.1111/rssb.12021. (Visited on 09/05/2024).

Lei, Jing et al. (2018). "Distribution-free predictive inference for regression". In: *Journal of the American Statistical Association* 113.523, pp. 1094–1111.

Papadopoulos, Harris, Alex Gammerman, and Volodya Vovk (2008). "Normalized nonconformity measures for regression conformal prediction". In: *Proceedings of the IASTED International Conference on Artificial Intelligence and Applications (AIA 2008)*, pp. 64–69.

Papadopoulos, Harris et al. (2002). "Inductive confidence machines for regression". In: *Machine learning: ECML 2002: 13th European conference on machine learning Helsinki, Finland, August 19–23, 2002 proceedings 13*. Springer, pp. 345–356.

Romano, Yaniv, Evan Patterson, and Emmanuel Candes (2019). "Conformalized quantile regression". In: *Advances in neural information processing systems* 32.

Susmann, Herbert, Antoine Chambaz, and Julie Josse (2023). *AdaptiveConformal: An R Package for Adaptive Conformal Inference*. arXiv: 2312.00448 [stat.CO]. URL: https://arxiv.org/abs/2312.00448.

Tibshirani, Ryan J et al. (2019). "Conformal prediction under covariate shift". In: *Advances in neural information processing systems* 32.

Vovk, Vladimir, Alexander Gammerman, and Glenn Shafer (2005). *Algorithmic learning in a random world*. Vol. 29. Springer.

Vovk, Vladimir et al. (June 2018). "Cross-Conformal Predictive Distributions". In: *Proceedings of the Seventh Workshop on Conformal and Probabilistic Prediction and Applications*. PMLR, pp. 37–51. (Visited on 02/14/2023).

Vovk, Vladimir et al. (Mar. 2019). "Nonparametric Predictive Distributions Based on Conformal Prediction". In: *Machine Learning* 108.3, pp. 445–474. ISSN: 1573-0565. DOI: 10.1007/s10994-018-5755-8. (Visited on 08/28/2024).

Vovk, Volodya, Alexander Gammerman, and Craig Saunders (1999). "Machine-learning applications of algorithmic randomness". In: *International Conference on Machine Learning*.

Wang, Xiaoqian and Rob J Hyndman (2024). *Online conformal inference for multi-step time series forecasting*. arXiv: 2410.13115 [stat.ME]. URL: https://arxiv.org/abs/2410.13115.

Xu, Chen and Yao Xie (July 2021). "Conformal Prediction Interval for Dynamic Time-Series". In: *Proceedings of the 38th International Conference on Machine Learning*. PMLR, pp. 11559–11569. (Visited on 01/18/2023).

— (2023). "Conformal prediction for time series". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.10, pp. 11575–11587.