

# Conformal prediction and its extensions

## 1 CP: Conformal prediction

Methods for distribution-free prediction.

**Assumption: exchangeability**

- The **data**  $Z_i = (X_i, Y_i)$  are assumed to be exchangeable (for example, i.i.d.).
  - Definition (Vovk, Gammernan, and Shafer 2005; Shafer and Vovk 2008). Suppose that for any collection of  $N$  values, the  $N!$  different orderings are equally likely. Then we say that  $Z_1, \dots, Z_N$  are exchangeable. The exchangeability assumption is slightly weaker than the i.i.d. assumption.
- The **algorithm** which maps data to a fitted model  $\hat{\mu} : \mathcal{X} \rightarrow \mathbb{R}$  is assumed to treat the data points symmetrically.
  - For example, OLS versus WLS.

### 1.1 Split conformal prediction

(inductive conformal prediction)

1. initial training data set: pre-trained model  $\hat{\mu} : \mathcal{X} \rightarrow \mathbb{R}$ .
2. holdout/calibration set: nonconformity scores  $R_i = |Y_i - \hat{\mu}(X_i)|$ ,  $i = 1, \dots, n$ .
3. prediction set:  $\widehat{C}_n(X_{n+1}) = \hat{\mu}(X_{n+1}) \pm Q_{1-\alpha} \left( \sum_{i=1}^n \frac{1}{n+1} \cdot \delta_{R_i} + \frac{1}{n+1} \cdot \delta_{+\infty} \right)$ .  
(the  $\lceil (n+1)(1-\alpha) \rceil$ th smallest of  $R_1, \dots, R_n$ )

Drawback: the loss of accuracy due to sample splitting.

## 1.2 Full conformal prediction

(transductive conformal prediction)

1. training data & a hypothesized test point:  $\hat{\mu}^y = \mathcal{A}((X_1, Y_1), \dots, (X_n, Y_n), (X_{n+1}, y))$  for each  $y \in \mathbb{R}$ .
2. residuals:  $R_i^y = \begin{cases} |Y_i - \hat{\mu}^y(X_i)|, & i = 1, \dots, n \\ |y - \hat{\mu}^y(X_{n+1})|, & i = n + 1 \end{cases}$ .
3. prediction set:  $\widehat{C}_n(X_{n+1}) = \{y \in \mathbb{R} : R_{n+1}^y \leq Q_{1-\alpha}(\sum_{i=1}^{n+1} \frac{1}{n+1} \cdot \delta_{R_i^y})\}$ .

Drawback: a steep computational cost.

*THEOREM:*

$\mathbb{P}\{Y_{n+1} \in \widehat{C}_n(X_{n+1})\} \geq 1 - \alpha$  holds true for both split conformal and full conformal.

## 1.3 Jackknife+

(close to cross-conformal prediction in Vovk (2013), offering a compromise between the computational and statistical costs)

1. training data with  $i$ th point removed:  $\hat{\mu}_{-i} = \mathcal{A}((X_1, Y_1), \dots, (X_{i-1}, Y_{i-1}), (X_{i+1}, Y_{i+1}), \dots, (X_n, Y_n))$ .
2. residuals:  $R_i^{\text{LOO}} = |Y_i - \hat{\mu}_{-i}(X_i)|$ .
3. prediction set:

$$\left[ Q_\alpha \left( \sum_{i=1}^n \frac{1}{n+1} \cdot \delta_{\hat{\mu}_{-i}(X_{n+1}) - R_i^{\text{LOO}}} + \frac{1}{n+1} \cdot \delta_{-\infty} \right), Q_{1-\alpha} \left( \sum_{i=1}^n \frac{1}{n+1} \cdot \delta_{\hat{\mu}_{-i}(X_{n+1}) + R_i^{\text{LOO}}} + \frac{1}{n+1} \cdot \delta_{+\infty} \right) \right]$$

Drawback: while in practice it generally provides coverage close to the target level  $1 - \alpha$ , its theoretical guarantee only ensures  $1 - 2\alpha$  probability of coverage in the worst case.

*THEOREM:*

$\mathbb{P}\{Y_{n+1} \in \widehat{C}_n(X_{n+1})\} \geq 1 - 2\alpha$  holds true for jackknife+.

## 2 Conformal time-series forecasting

Stankeviciute, M Alaa, and Schaar (2021): CF-RNNs

**Multi-horizon time-series forecasting problem**

**Notation:**

- the  $i$ th data point:  $Z_i = (y_{1:t}^{(i)}, y_{t+1:t+H}^{(i)})$ . Note that the label  $y_{t+1:t+H}^{(i)}$  is now an  $H$ -dimensional value, in contrast with the scalar  $y$  value from before.

### Assumption:

- exchangeable time-series observations

## 2.1 Methodology

(Split conformal prediction)

1. training set: train the underlying (auxiliary) model  $\hat{\mu} : \mathbb{R}^t \rightarrow \mathbb{R}^H$ , which produces multi-horizon forecasts **directly** (conditionally independent predictions).
2. calibration set: obtain the  $H$ -dimensional nonconformity scores

$$R_i = \left[ \left| y_{t+1}^{(i)} - \hat{y}_{t+1}^{(i)} \right|, \dots, \left| y_{t+H}^{(i)} - \hat{y}_{t+H}^{(i)} \right| \right]^\top.$$

3. prediction set:  $\Gamma_1^\alpha(y_{1:t}^{(n+1)}), \dots, \Gamma_H^\alpha(y_{1:t}^{(n+1)})$ , where  $\Gamma_h^\alpha(y_{1:t}^{(n+1)}) = [\hat{y}_{t+h}^{(n+1)} - \hat{\varepsilon}_h, \hat{y}_{t+h}^{(n+1)} + \hat{\varepsilon}_h]$ ,  $\forall h \in \{1, \dots, H\}$  with the critical nonconformity scores  $\hat{\varepsilon}_1, \dots, \hat{\varepsilon}_H$  become the  $[(n+1)(1 - \alpha/H)]$ -th smallest residuals in the corresponding nonconformity score distributions. (Bonferroni correction)

*THEOREM:*

- $\mathcal{D} = \left\{ \left( y_{1:t}^{(i)}, y_{t+1:t+H}^{(i)} \right) \right\}_{i=1}^n$ : **exchangeable** time-series observations.
- $\hat{\mu}$ : model predicting  $H$ -step forecasts using **the direct strategy**.

$$\mathbb{P}(\forall h \in \{1, \dots, H\} \cdot y_{t+h} \in [\hat{y}_{t+h} - \hat{\varepsilon}_h, \hat{y}_{t+h} + \hat{\varepsilon}_h]) \geq 1 - \alpha.$$

## 3 NexCP: Conformal prediction beyond exchangeability

Barber et al. (2023)

1. Nonexchangeable conformal with a **symmetric** algorithm (weights)
2. Nonexchangeable conformal with **nonsymmetric** algorithms (weights & swap)

### 3.1 Notation

- the  $i$ th data point  $Z_i = (X_i, Y_i)$
- the full data sequence  $Z = (Z_1, \dots, Z_{n+1})$
- the sequence after swap  $Z^i = (Z_1, \dots, Z_{i-1}, Z_{n+1}, Z_{i+1}, Z_n, Z_i)$

### 3.2 Methodology

1. Choose **fixed (non-data-dependent)** weights  $w_1, \dots, w_n \in [0, 1]$  with the intuition that a higher weight should be assigned to a data point that is “trusted” more.
2. Normalize weights  $\tilde{w}_i = \frac{w_i}{w_1 + \dots + w_{n+1}}, i = 1, \dots, n$  and  $\tilde{w}_{n+1} = \frac{1}{w_1 + \dots + w_{n+1}}$ .
3. Generate “tagged” data points  $(X_i, Y_i, t_i) \in \mathcal{X} \times \mathbb{R} \times \mathcal{T}$ .
4. Swap data set, resulting in  $Z^K$  with  $K \sim \sum_{i=1}^{n+1} \tilde{w}_i \cdot \delta_i$ , i.e., two data points have swapped tags.
5. Apply algorithm  $\mathcal{A}$  to  $Z^K$  in place of  $Z$ .

### 3.3 Split conformal prediction

- prediction set:  $\widehat{C}_n(X_{n+1}) = \hat{\mu}(X_{n+1}) \pm Q_{1-\alpha}(\sum_{i=1}^n \tilde{w}_i \cdot \delta_{R_i} + \tilde{w}_{n+1} \cdot \delta_{+\infty})$

### 3.4 Full conformal prediction

1. training data & a hypothesized test point:  $\hat{\mu}^{y,k} = \mathcal{A}((X_{\pi_k(i)}, Y_{\pi_k(i)}^y, t_i) : i \in [n+1])$  for any  $y \in \mathbb{R}$  and  $k \in [n+1]$ , where  $\pi_k$  is the permutation on  $[n+1]$  swapping indices  $k$  and  $n+1$ , and  $Y_i^y = \begin{cases} Y_i, & i = 1, \dots, n \\ y, & i = n+1 \end{cases}$ .
2. residuals:  $R_i^{y,k} = \begin{cases} |Y_i - \hat{\mu}^{y,k}(X_i)|, & i = 1, \dots, n \\ |y - \hat{\mu}^{y,k}(X_{n+1})|, & i = n+1 \end{cases}$ .
3. prediction set:  $\widehat{C}_n(X_{n+1}) = \{y : R_{n+1}^{y,K} \leq Q_{1-\alpha}(\sum_{i=1}^{n+1} \tilde{w}_i \cdot \delta_{R_i^{y,K}})\}$ .

*THEOREM:*

- Lower bounds on coverage.

$$\mathbb{P}\{Y_{n+1} \in \widehat{C}_n(X_{n+1})\} \geq 1 - \alpha - \sum_{i=1}^n \tilde{w}_i \cdot d_{TV}(R(Z), R(Z^i))$$

- Upper bounds on coverage.

$$\mathbb{P}\{Y_{n+1} \in \widehat{C}_n(X_{n+1})\} < 1 - \alpha + \tilde{w}_{n+1} + \sum_{i=1}^n \tilde{w}_i \cdot d_{TV}(R(Z), R(Z^i)),$$

if  $R_1^{Y_{n+1},K}, \dots, R_n^{Y_{n+1},K}, R_{n+1}^{Y_{n+1},K}$  are distinct with probability 1.

The results hold true for both nonexchangeable split conformal and full conformal.

So, if  $\tilde{w}_{n+1} = \frac{1}{w_1 + \dots + w_{n+1}}$  is small (the effective sample size is large), then mild violations of exchangeability can only lead to mild undercoverage or to mild overcoverage.

### 3.5 Jackknife+

1. training data with  $i$ th point removed and data tag swapped:

$$\hat{\mu}_{-i}^k = \mathcal{A} \left( (X_{\pi_k(j)}, Y_{\pi_k(j)}, t_j) : j \in [n+1], \pi_k(j) \notin \{i, n+1\} \right).$$

2. residuals:  $R_i^{k, \text{LOO}} = |Y_i - \hat{\mu}_{-i}^k(X_i)|$ .

3. prediction set:

$$\left[ Q_\alpha \left( \sum_{i=1}^n \tilde{w}_i \cdot \delta_{\hat{\mu}_{-i}^K(X_{n+1}) - R_i^{K, \text{LOO}}} + \tilde{w}_{n+1} \cdot \delta_{-\infty} \right), Q_{1-\alpha} \left( \sum_{i=1}^n \tilde{w}_i \cdot \delta_{\hat{\mu}_{-i}^K(X_{n+1}) + R_i^{K, \text{LOO}}} + \tilde{w}_{n+1} \cdot \delta_{+\infty} \right) \right]$$

*THEOREM:*

$$\mathbb{P} \left\{ Y_{n+1} \in \widehat{C}_n(X_{n+1}) \right\} \geq 1 - 2\alpha - \sum_{i=1}^n \tilde{w}_i \cdot d_{\text{TV}}(R_{\text{jack}+}(Z), R_{\text{jack}+}(Z^i))$$

## 4 WCP: Conformal prediction under covariate shift

Tibshirani et al. (2019)

A weighted version of conformal prediction, using a quantile of a suitably weighted empirical distribution of nonconformity scores.

### 4.1 Setup/assumption - Covariate shift

Focus on settings in which the data  $(X_i, Y_i), i = 1, \dots, n+1$  are no longer exchangeable. Specifically,

$$\begin{aligned} (X_i, Y_i) &\stackrel{\text{i.i.d.}}{\sim} P = P_X \times P_{Y|X}, \quad i = 1, \dots, n, \\ (X_{n+1}, Y_{n+1}) &\sim \widetilde{P} = \widetilde{P}_X \times P_{Y|X}, \text{ independently.} \end{aligned} \tag{1}$$

the test and training covariate distributions differ, but the likelihood ratio between the two distributions,  $d\widetilde{P}_X/dP_X$ , must be known exactly or well approximated for correct coverage.

## 4.2 Methodology

**Prediction set:**

- CP:  $\widehat{C}_n(X_{n+1}) = \widehat{\mu}(X_{n+1}) \pm Q_{1-\alpha} \left( \sum_{i=1}^n \frac{1}{n+1} \cdot \delta_{R_i} + \frac{1}{n+1} \cdot \delta_{+\infty} \right)$ .
- NexCP:  $\widehat{C}_n(X_{n+1}) = \widehat{\mu}(X_{n+1}) \pm Q_{1-\alpha} \left( \sum_{i=1}^n \tilde{w}_i \cdot \delta_{R_i} + \tilde{w}_{n+1} \cdot \delta_{+\infty} \right)$ .
  - weights  $w$  are fixed
  - $\tilde{w}_i = \frac{w_i}{w_1 + \dots + w_n + 1}, i = 1, \dots, n$  and  $\tilde{w}_{n+1} = \frac{1}{w_1 + \dots + w_n + 1}$
- WCP:  $\widehat{C}_n(X_{n+1}) = \widehat{\mu}(X_{n+1}) \pm Q_{1-\alpha} \left( \sum_{i=1}^n p_i^w(x) \delta_{R_i} + p_{n+1}^w(x) \delta_{+\infty} \right)$ .
  - $w = d\tilde{P}_X/dP_X$  or  $w \propto d\tilde{P}_X/dP_X$
  - $p_i^w(x) = \frac{w(X_i)}{\sum_{j=1}^n w(X_j) + w(x)}, i = 1, \dots, n$ , and  $p_{n+1}^w(x) = \frac{w(x)}{\sum_{j=1}^n w(X_j) + w(x)}$

The **weight function**  $\hat{w}$  can be estimated using logistic regression, random forests, etc.

*THEOREM:*

Assume data from the model Equation 1. Assume  $\tilde{P}_X$  is absolutely continuous with respect to  $P_X$ , and denote  $w = d\tilde{P}_X/dP_X$ . For any score function  $S$ , and any  $\alpha \in (0, 1)$ , define for  $x \in \mathbb{R}^d$ . Then

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

## 4.3 Comparison to NexCP

1. Assumption
  - WCP: covariate shift
  - NexCP: do not make any assumption on the joint distribution of the  $n + 1$  points
2. Weights
  - WCP: a function of the data point  $(X_i, Y_i)$  to compensate for the **known** distribution shift.
  - NexCP: required to be fixed, can compensate for **unknown** violations of the exchangeability assumption, as long as the violations are **small** (to ensure a low coverage gap).
3. Nonsymmetric algorithm
  - WCP: No.
  - NexCP: Yes.
4. For exchangeable data

- WCP: does not have any coverage guarantee.
- NexCP: retains exact coverage.

## 5 ACP: Adaptive conformal inference under distribution shift

Gibbs and Candes (2021)

- No assumptions on the data-generating distribution.
- Modelling the distribution shift as a learning problem in a single parameter whose optimal value is varying over time and must be continuously re-estimated.
- Adjust significance level  $\alpha$  based on rolling coverage of  $Y_t$ .

### 5.1 Methodology

Work with score function  $S(\cdot)$  and quantile function  $\hat{Q}(\cdot)$ .

**Some facts:**

- If the distribution of the data is shifting over time, both functions should be regularly re-estimated to align with the most recent observations.
- The realized miscoverage rate  $M_t(\alpha)$  also varies over time and may not be equal or close to  $\alpha$ .

**Assumptions:**

Assume that there may be an alternative value  $\alpha^* \in [0, 1]$  such that  $M_t(\alpha^*) \cong \alpha$ .

Assume that with probability one,  $\hat{Q}_t(\cdot)$  is continuous, non-decreasing and such that  $\hat{Q}_t(0) = -\infty$  and  $\hat{Q}_t(1) = \infty$ .

**Adaptive conformal inference:**

- Under assumptions,  $M_t(\cdot)$  will be non-decreasing on  $[0, 1]$  with  $M_t(0) = 0$  and  $M_t(1) = 1$ .
- Define  $\alpha_t^* := \sup \{\beta \in [0, 1] : M_t(\beta) \leq \alpha\}$ , then  $M_t(\alpha_t^*) = \alpha$ .
- Use a simple **update process** to perform the calibration.
  - Intuition: after examining the empirical miscoverage frequency of the previous prediction sets, decreasing (increasing) estimate of  $\alpha_t^*$  if the prediction sets were historically under-covering (over-covering)  $Y_t$ .

- Let  $\alpha_1 = \alpha$ , consider the **update**

$$\alpha_{t+1} := \alpha_t + \gamma (\alpha - \text{err}_t)$$

OR

$$\alpha_{t+1} = \alpha_t + \gamma \left( \alpha - \sum_{s=1}^t w_s \text{err}_s \right),$$

where  $\gamma > 0$  is a fixed step size parameter whose choice gives a tradeoff between adaptability and stability.

So the distribution is allowed to shift continuously over time.

*THEOREM:*

With probability one we have that for all  $T \in \mathbb{N}$ ,

$$\left| \frac{1}{T} \sum_{t=1}^T \text{err}_t - \alpha \right| \leq \frac{\max\{\alpha_1, 1 - \alpha_1\} + \gamma}{T\gamma}.$$

In particular,  $\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \text{err}_t \stackrel{\text{a.s.}}{=} \alpha$ .

## 6 PID: Conformal PID control for time series prediction

Angelopoulos, Candes, and Tibshirani (2023)

- PID: proportional-integral-derivative (control theory).
- reactive + forward-looking.



## 6.1 Methodology

The proposed conformal PID controller is given by

$$q_{t+1} = \underbrace{\eta(\text{err}_t - \alpha)}_{\text{P}} + r_t \underbrace{\left( \sum_{i=1}^t (\text{err}_i - \alpha) \right)}_{\text{I}} + \underbrace{g'_t}_{\text{D}}.$$

The goal is to achieve **long-run coverage** in time, i.e.,  $\frac{1}{T} \sum_{t=1}^T \text{err}_t = \alpha + o(1)$ , where  $o(1)$  denotes a quantity that tends to zero as  $T \rightarrow \infty$ .

- P: Quantile tracking. ACI can be regarded as a special case but it can sometimes output infinite or null prediction sets, while quantile tracking on the scale of the original score sequence does not have this behavior.
- I: Error integration.  $r_t$  should be a saturation function to achieve long-run coverage.
- D: Scorecasting (forward-looking). It trains a second model to predict the quantile of the next score.

Reparametrize the PID formula to produce a sequence of quantile estimates  $q_t$ ,  $t \in N$  used in the prediction sets:

let  $\hat{q}_{t+1}$  be any function of the past:  $x_i, y_i, q_i$ , for  $i \leq t$ , then update

$$q_{t+1} = \underbrace{\hat{q}_{t+1}}_{\text{scorecaster}} + r_t \underbrace{\left( \sum_{i=1}^t (\text{err}_i - \alpha) \right)}_{\text{integrator}},$$

which is generally useful and is the main focus of the paper.

The applications only consider one-step-ahead forecast.

## 7 Others

### 7.1 EnbPI: predictive inference method around ensemble estimators

Xu and Xie (2021)

- For Dynamic time series
- Ensemble point forecasts + update residuals

## 7.2 SPCI: Sequential Predictive Conformal Inference

Xu and Xie (2023)

- Adaptively re-estimate the conditional quantile of non-conformity scores, upon leveraging the temporal dependency among residuals. Random Forest for quantile regression is used.

## 7.3 Kath & Ziel (2021, IJF)

Kath and Ziel (2021)

### Method 1.

The steps are:

1. Re-arrange time series data as form  $Z = \{(x_1, y_1), \dots, (x_L, y_L)\}$ .
2. Randomly split observations into subsets  $\pi$  (training set) and  $1 - \pi$  (calibration set).
3. Train a random forecast model using the training set and generate out-of-sample forecasts.
4. Calculate non-conformity score.
5. Obtain final prediction interval estimation.

### Method 2.

The differences compared to Method 1 are:

1. Here an adjusted normalized non-conformity score  $\lambda_{i,h} = \frac{|y_{i,h} - \hat{y}_{i,h}|}{|\hat{\varepsilon}_{i,h}|}$  is used.
2. A second explicit error estimation model is trained to estimate the error associated with the prediction  $\hat{y}_{t,h}$ , i.e.,  $\varepsilon_{t,h}$ .
3. The new interval forecast is now given by  $y_{\alpha,T+1,h} = \hat{y}_{T+1,h} \pm \lambda_{L+1,h}^\alpha |\hat{\varepsilon}_{L+1,h}|$ .

The applications only consider one-step-ahead forecast.

## 7.4 TimeGPT

Garza and Mergenthaler-Canseco (2023)

For uncertainty quantification, they perform rolling forecasts on the latest available data to estimate the model's errors in forecasting the particular target time series.

The  $h$ -step-ahead forecast is generated based on  $h$ -step-ahead non-conformity errors on the calibration set separately.

## 7.5 LCP: Localized conformal prediction

Guan (2022)

The **weight** on data point  $i$  is determined as a **function of the distance**  $\|X_i - X_{n+1}\|_2$ , to enable predictive coverage that holds **locally** (in neighborhoods of  $X$  space, that is, an approximation of prediction that holds conditional on the value of  $X_{n+1}$ ).

## 8 Simulation

### 8.1 Setup

Simulate a time series  $y$  with length  $T = 5000$  from an AR(2) model with  $\phi_1 = 0.8$ ,  $\phi_2 = -0.5$ , and  $\sigma^2 = 1$ .

Only consider one-step-ahead forecasting, i.e.,  $h = 1$ .

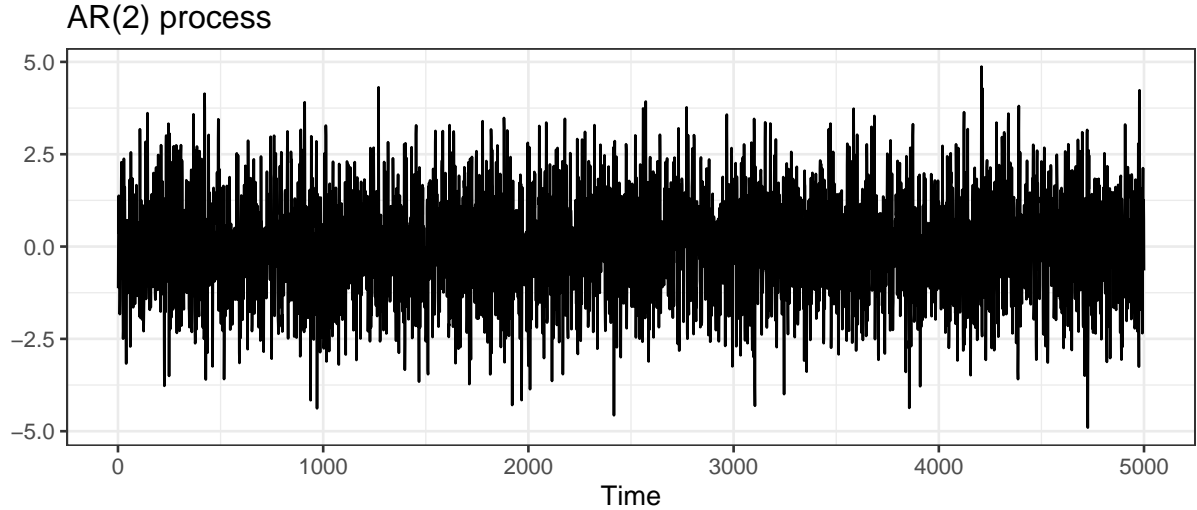


Figure 1: Simulated time series from an AR(2) model.

### 8.2 Split conformal prediction with fixed training set

#### 8.2.1 Details

Let  $n$  be the number of data used to fit an AR(2) model, and  $m$  be the number of data in the calibration set.

- Step 1. Rearrange the time series as  $Z_i = (X_i, Y_i)$ , where  $X_i = y_{i:(i+1)}$  and  $Y_i = y_{i+2}$ ,  $i = 1, 2, \dots, T - 2$ .
- Step 2. Train an AR(2) model  $\hat{\mu}$  based on training set with length  $n$ ,  $Z_{tr} = (Z_1, Z_2, \dots, Z_n)$ .
- Step 3. Calculate nonconformity scores (absolute residuals) based on calibration set with length  $m$ , i.e.,  $R_i = |Y_i - \hat{\mu}(X_i)|$ ,  $i = n + 1, n + 2, \dots, n + m$ .
- Step 4. Generate PI on test set.  $\hat{C}_m(X_i) = \hat{\mu}(X_i) \pm Q_{1-\alpha} \left( \sum_{j=n+1}^{n+m} w_j \cdot \delta_{R_j} + w_i \cdot \delta_{+\infty} \right)$  for  $i = n + m + 1, \dots, T - 2$ .

## 8.2.2 Results

Let  $n = m = 500$  and fit the AR(2) model using the `lm` function.

Consider methods: CP, WCP.LR, WCP.RF, NexCP with  $\alpha = 0.1$ .

Issues:

- PIs have constant width over the test set. We can generate PIs with varying local width by using a function to perform training on the absolute residuals i.e., to produce an estimator of  $E(R|X)$ .
- When generating weights via GLM or RF in WCP method, we need to use all data from the test set, which is not reasonable.
- Regression-based model.

## 8.3 Split conformal prediction with rolling training set

### 8.3.1 Details

Let  $n$  be the number of observations used to fit an AR(2) model, and  $m$  be the number of observations in the calibration set.

For  $i = n + 1, n + 2, \dots, T$ :

- Step 1. Fit an AR(2) model  $\hat{\mu}_{i-1}$  based on observations  $y_{(i-n):(i-1)}$ , generate one-step-ahead forecast  $\hat{y}_i$ .
- When  $i > n + m$ :
  - Step 2. Calculate weights for the updated calibration set with length  $m$  and the updated test set with length 1 using different methods. Here WCP can not be applied because we only have a test set with length equal to one.

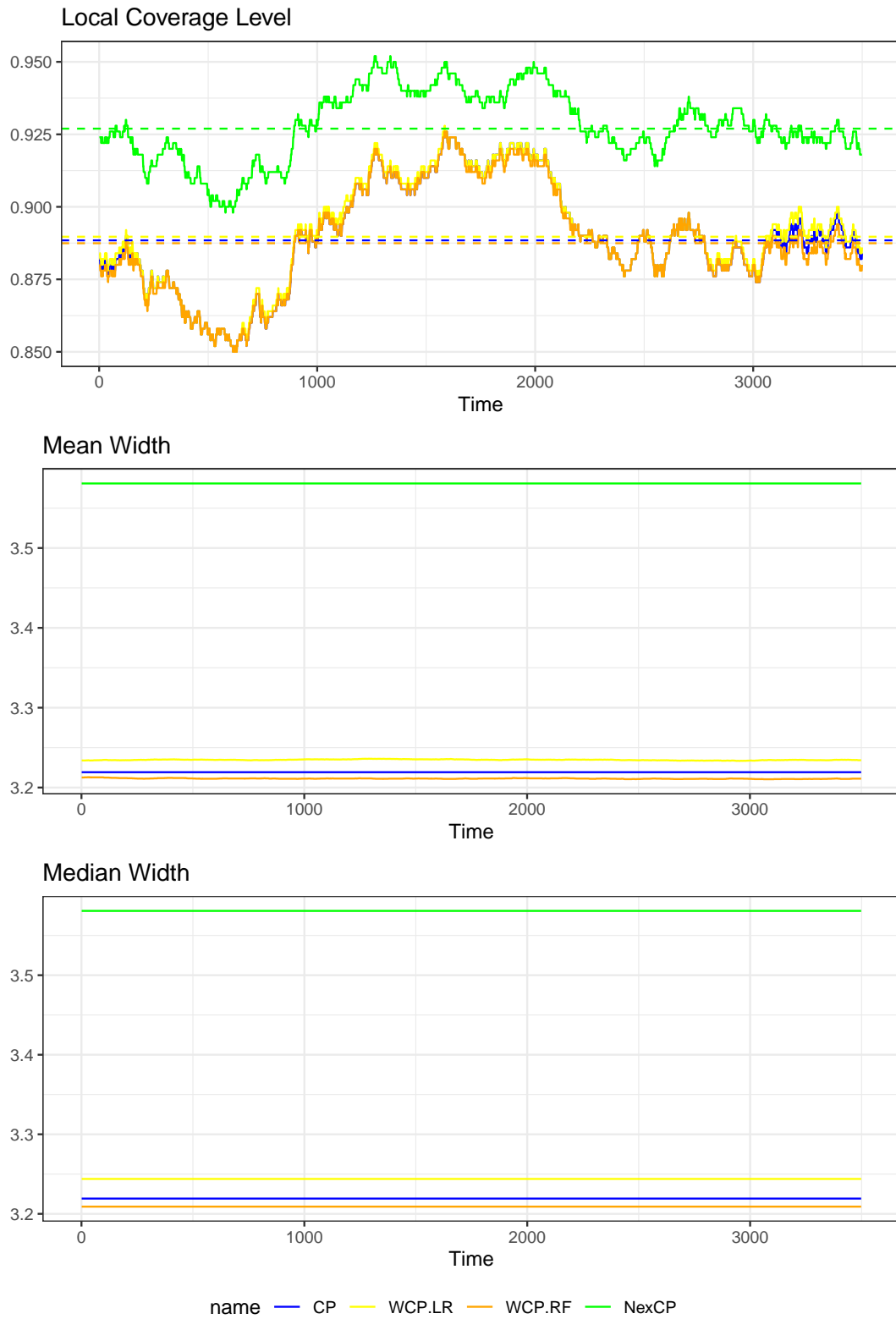


Figure 2: Local coverage frequencies and width for split conformal prediction methods with fixed training set ( $k=500$ ).

- Step 3. Generate PI on test set.  $\hat{y}_i \pm Q_{1-\alpha} \left( \sum_{j=i-m}^{i-1} w_j \cdot \delta_{R_j} + w_i \cdot \delta_{+\infty} \right)$ .
- Setp 4 for ACP. Update  $\alpha$  based on the recent empirical miscoverage frequency.
- Step 5. Calculate nonconformity scores (absolute residuals)  $R_i = |y_i - \hat{y}_i|$ .

### 8.3.2 Results

Let  $n = m = 500$  and fit AR(2) models using the `Arima` function with setting `order = c(2,0,0)`, `include.mean = TRUE`, `method = "CSS"` to make it comparable with the previous result.

Consider methods: AR, CP, NexCP, ACP, PID with  $\alpha = 0.1$ .

For the PID method:

- The AR(2) models are trained using `nfit` observations through a rolling window strategy, as with the other methods.
- Theta method is used as the scorecaster to forecast the conformity scores. An expanding window strategy is used if `ncast = NULL`; a rolling window strategy is used if `ncast != NULL` (here we use rolling windows with `ncast = ncal` for comparison with other methods).
- The first `nburnin` outputs are removed to have output of equal length. (`nburnin` is equal to `ncal` in other methods.)
- Some parameters.
  - Constants  $C_{\text{sat}}$  and  $K_I$  for the integrator  $r_t(x) = K_I \tan(x \log(t) / (t C_{\text{sat}}))$  are determined by some heuristics. See appendix B for more details. Here,  $T = 5000$ ,  $\delta = 0.01$ , and  $K_I = 2$ .
  - The learning rate for the quantile tracking (P):  $\eta = 0.1 \hat{B}_t$ .

Features:

- The fitted model is updating over time.
- PIs have changing width over the test set.
- Other non-regression-based models can be considered.
- Time-consuming because of the rolling window approach (but is still very quick for this simulation).

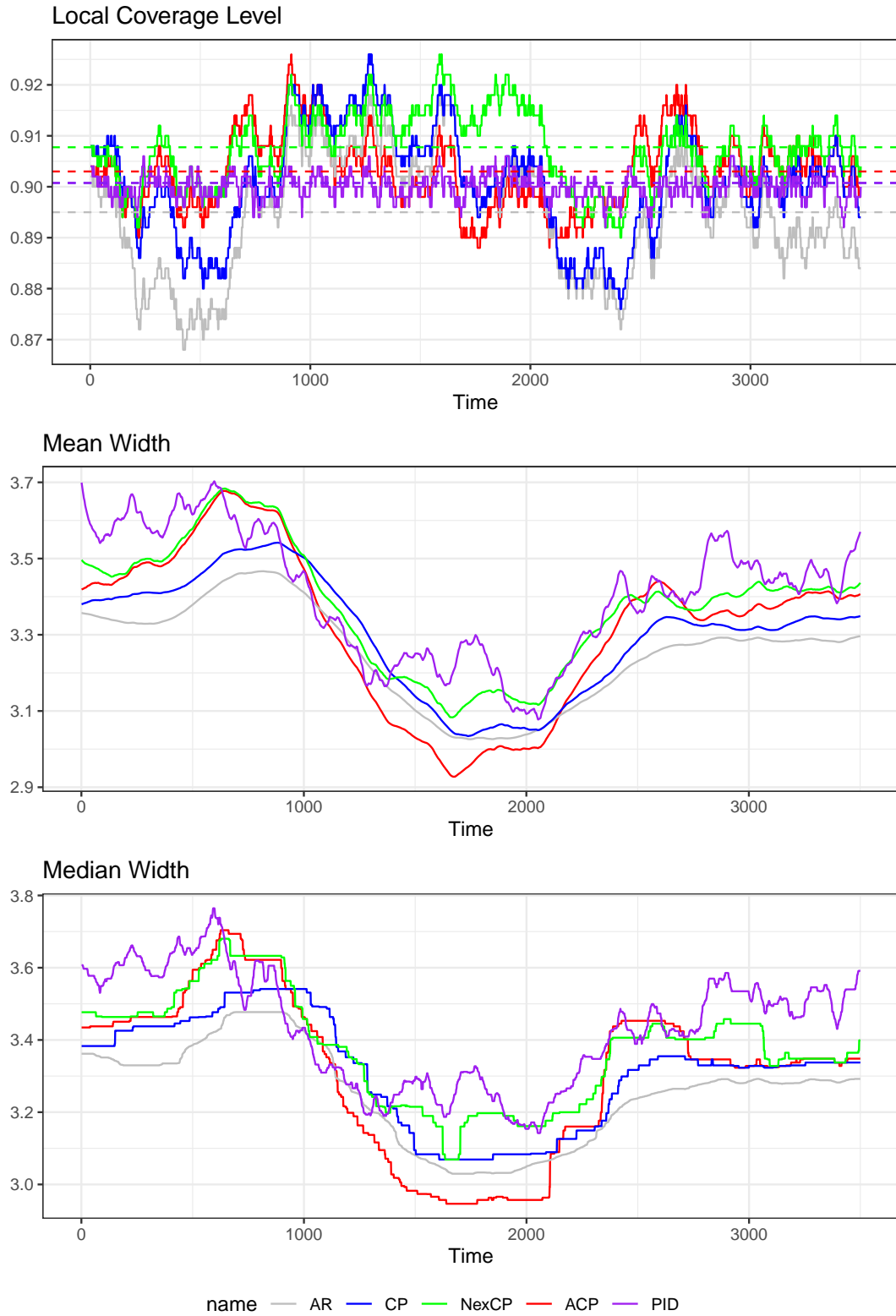


Figure 3: Local coverage frequencies and width for split conformal prediction methods with rolling training set and calibration set ( $k=500$ ).

### 8.3.3 Analysis on PID

#### 8.3.3.1 The effect of constants in integrator

The integrator used in the paper is  $r_t(x) = K_I \tan(x \log(t) / (tC_{\text{sat}}))$ , where  $C_{\text{sat}}$  controls how long the guaranteed coverage will be achieved, and  $K_I$  controls the change scale of the integrator.

Here we set the learning rate for the quantile tracking (P) to  $\eta = 0.1\hat{B}_t$ .

1. The initial wrong setting,  $C_{\text{sat}} = 1$  and  $K_I = 200$ , in Figure 4 gives a **wide variation** in the width of the prediction intervals, thus widening the mean and median values.
2. The heuristic setting,  $C_{\text{sat}} = 0.5618745$  and  $K_I = 2$ , in Figure 3 makes the variation in the width of the prediction intervals more reasonable.

However, upon comparison with other CP methods, it is evident that the relative situation of prediction interval width does not align with the relative situation of coverage.

#### 8.3.3.2 The effect of P, I, and D

To investigate the effect of P, I, and D and identify the source of large variations in prediction interval width, we present boxplots illustrating the width of ACP, P, PI, and PID. A learning rate of 0.005 is applied to ACP, while for P, PI, and PID, the learning rates are set to 0.1B and 0.01B in Figure 5 and Figure 6, respectively.

##### 1. ACP VS. P

- ACP can be considered as a special case of the quantile tracker. The choice of learning rate is crucial for both methods. A higher learning rate results in a larger variation in width. Particularly, ACP may output infinite or null sets when a large learning rate is used.

##### 2. P, PI and PID

- The integrator appears to have minimal impact on the outcomes. Nevertheless, Angelopoulos, Candes, and Tibshirani (2023) claims that it does nicely for most of the time series. We may need to do further exploration using time series with trend and seasonal patterns.
- The substantial variations stem from the scorecaster, it introduces variance into the score sequence that the sets will degrade in stability. Thus, scorecasters should be designed with care, as suggested by Angelopoulos, Candes, and Tibshirani (2023). Their usage is advisable primarily in instances where trends or other patterns show in the score sequence.
- Figure 7 and Figure 8 present results of PID using the learning rate 0.1B and 0.01B respectively.



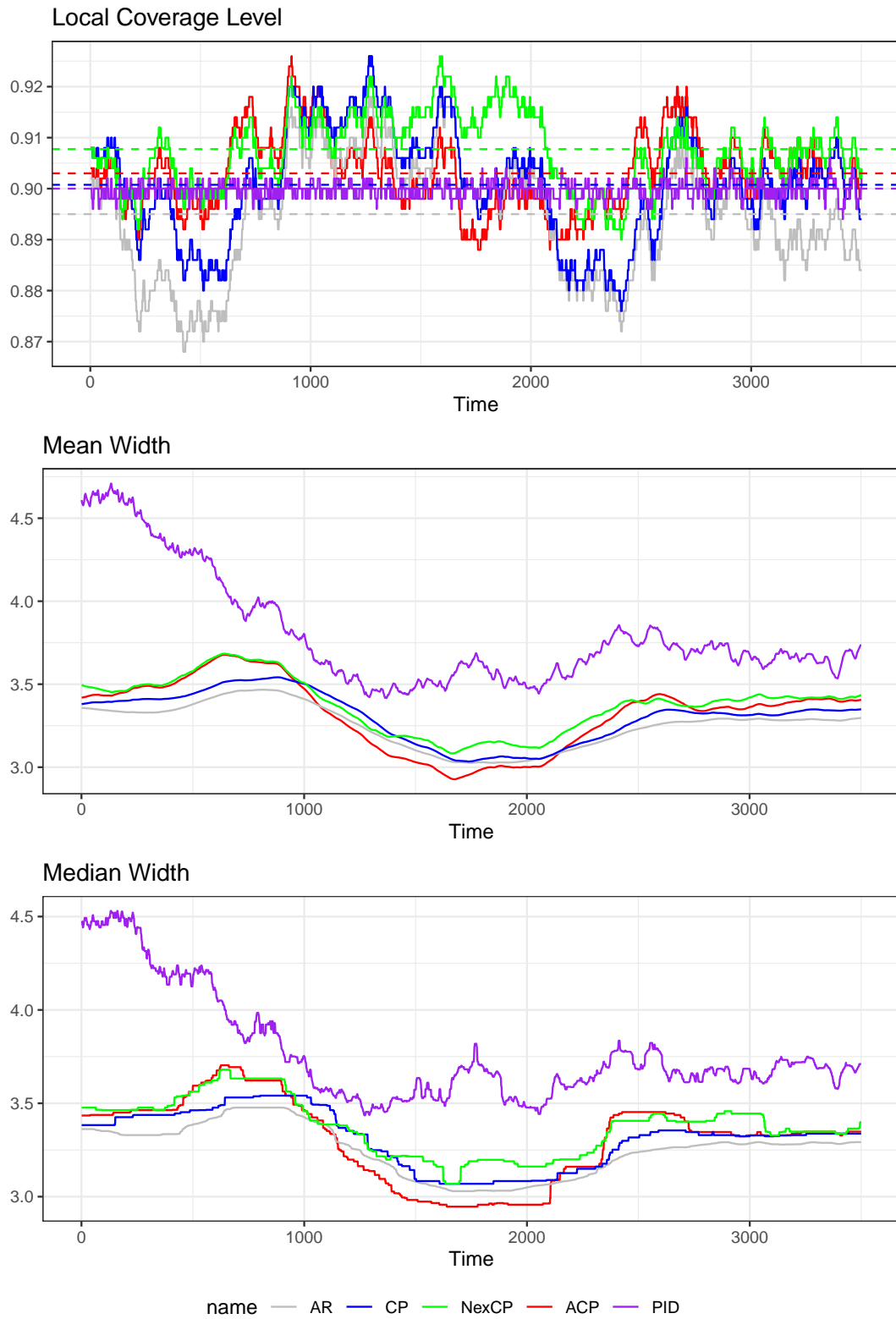


Figure 4: (Wrong setting.) Local coverage frequencies and width for split conformal prediction methods with rolling training set and calibration set ( $k=500$ ).

### 8.3.3.3 Conclusion

- The choice of **learning rate** is crucial for ACP and PID.
- The **constants in the integrator** should be chosen through heuristics or fine-tuned during a burn-in period. Further work is needed to validate the usefulness of the integrator.
- The **scorecaster** should be designed with care, sometimes it can be harmful.

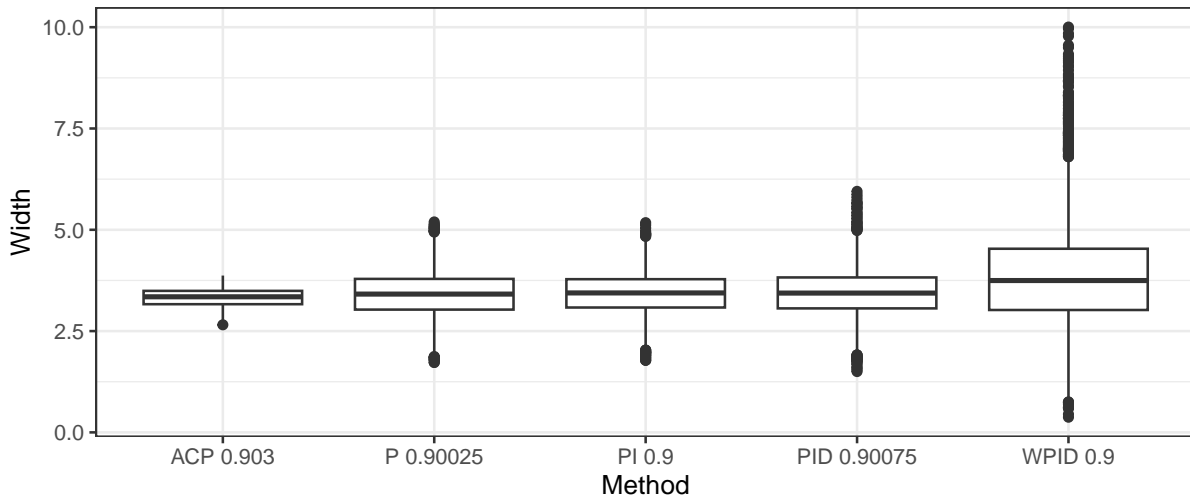


Figure 5: Boxplots for the width of various methods. For ACP, the learning rate is set to 0.005. For P, PI, and PID, the learning rate is set to 0.1B.

## 9 To do list

- ☐ Generalize the existing methods and get a general function for implementation.
  - ☐ Symmetric/asymmetric scores.
  - ☐ Expanding/rolling window strategy.
  - ☐ Include weights in PID (integrator?).
  - ☐ Handle seasonal time series and specify methods used for forecasting and scorecasting.
- ☐ Multi-step ahead forecasting.
  - ☐ Correlation between  $h$ -step-ahead residuals.
  - ☐ Correlation between  $s$ -step-ahead residuals and  $h$ -step-ahead residuals.
  - ☐ Similar strategy as PID?
- ☐ More simulations and real-world data.

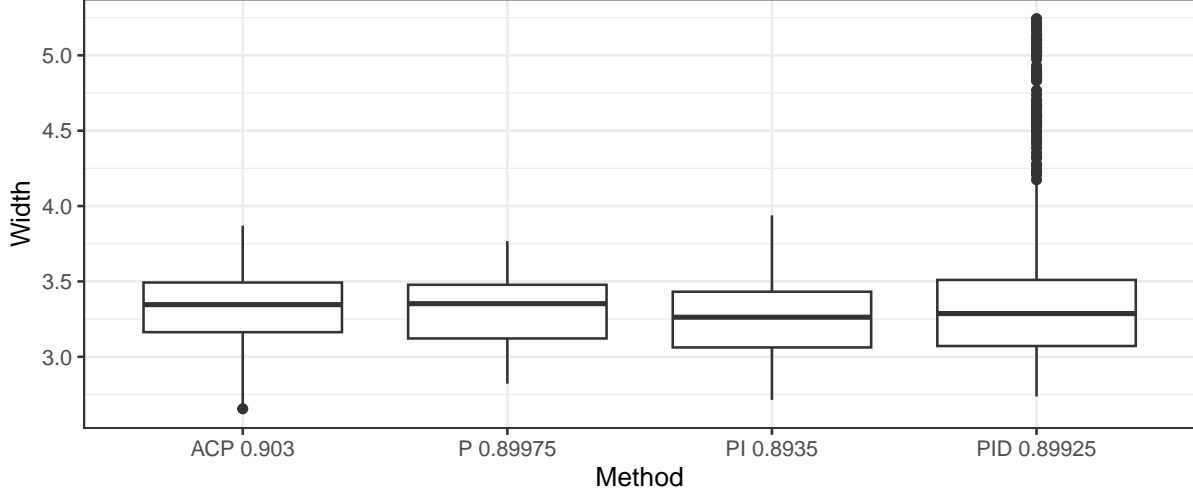


Figure 6: Boxplots for the width of various methods. For ACP, the learning rate is still set to 0.005. But, for P, PI, and PID, the learning rate is set to 0.01B.

- ☐ Theoretical proof.
- ☐ Other data-dependent weight.
- ☐ Hierarchical time series?

## 10 References

- Angelopoulos, Anastasios N, Emmanuel J Candes, and Ryan J Tibshirani. 2023. “Conformal PID Control for Time Series Prediction.” *arXiv Preprint arXiv:2307.16895*.
- Barber, Rina Foygel, Emmanuel J. Candès, Aaditya Ramdas, and Ryan J. Tibshirani. 2023. “Conformal Prediction Beyond Exchangeability.” *The Annals of Statistics* 51 (2). <https://doi.org/10.1214/23-aos2276>.
- Garza, Azul, and Max Mergenthaler-Canseco. 2023. “TimeGPT-1.” *arXiv Preprint arXiv:2310.03589*.
- Gibbs, Isaac, and Emmanuel Candes. 2021. “Adaptive Conformal Inference Under Distribution Shift.” *Advances in Neural Information Processing Systems* 34: 1660–72.
- Guan, Leying. 2022. “Localized Conformal Prediction: A Generalized Inference Framework for Conformal Prediction.” *Biometrika* 110 (1): 33–50. <https://doi.org/10.1093/biomet/asac040>.
- Kath, Christopher, and Florian Ziel. 2021. “Conformal Prediction Interval Estimation and Applications to Day-Ahead and Intraday Power Markets.” *International Journal of Forecasting* 37 (2): 777–99.
- Shafer, Glenn, and Vladimir Vovk. 2008. “A Tutorial on Conformal Prediction.” *Journal of Machine Learning Research* 9 (3).

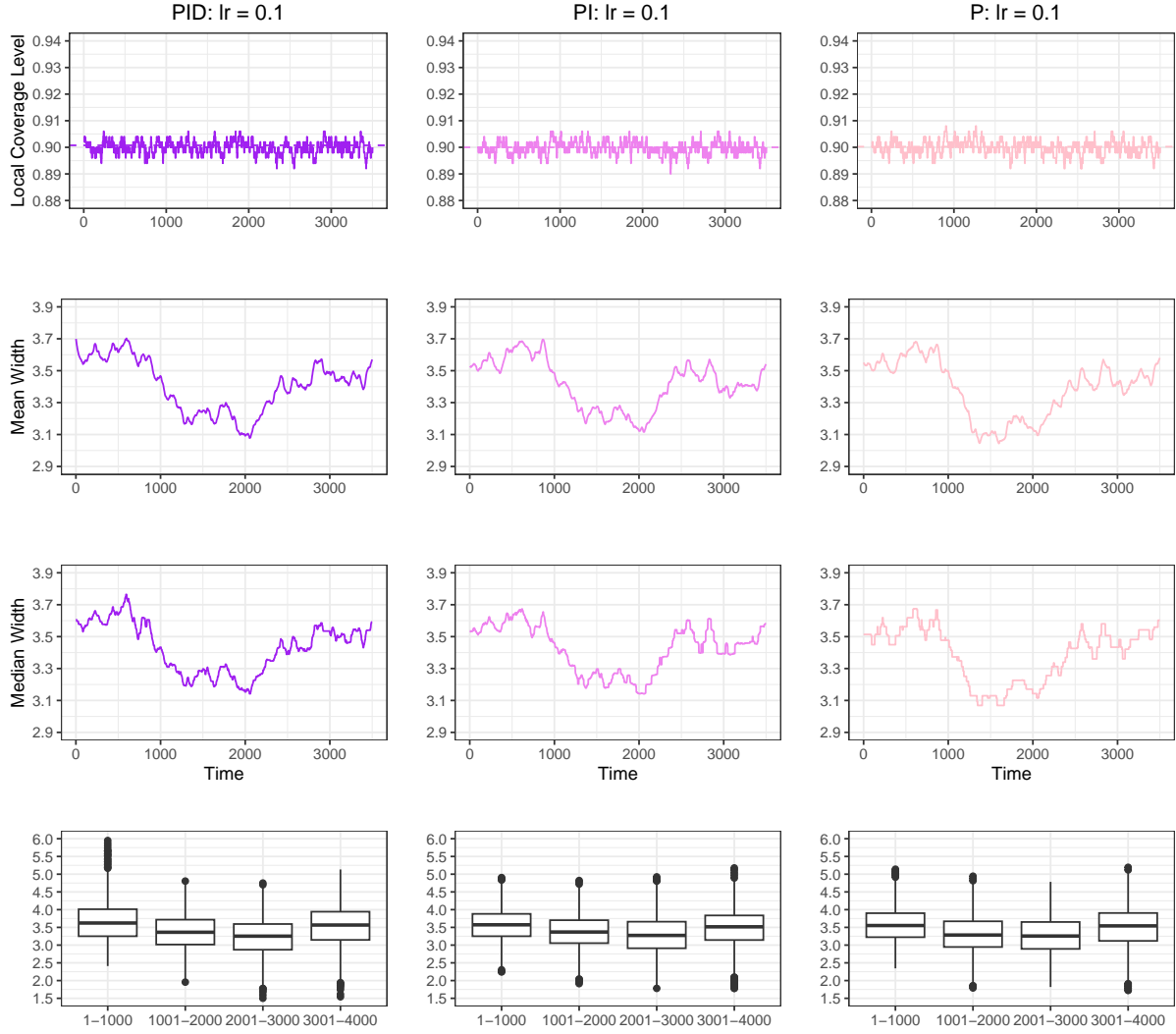


Figure 7: Local coverage frequencies and width for PID with rolling training set and calibration set ( $k=500$ ). The learning rate is set to 0.1B.

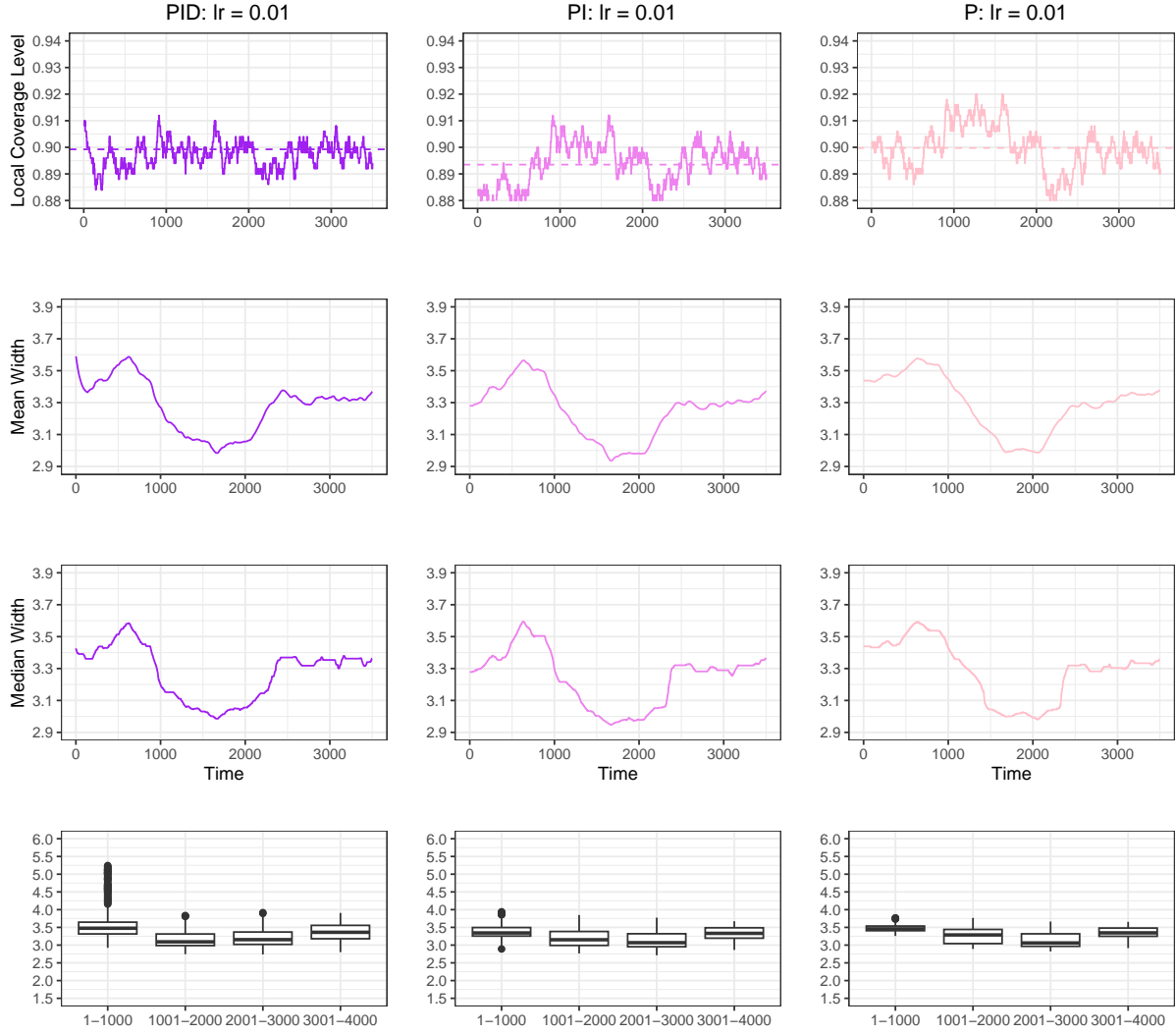


Figure 8: Local coverage frequencies and width for PID with rolling training set and calibration set ( $k=500$ ). The learning rate is set to 0.01B.

- Stankeviciute, Kamile, Ahmed M Alaa, and Mihaela van der Schaar. 2021. “Conformal Time-Series Forecasting.” *Advances in Neural Information Processing Systems* 34: 6216–28.
- Tibshirani, Ryan J, Rina Foygel Barber, Emmanuel Candes, and Aaditya Ramdas. 2019. “Conformal Prediction Under Covariate Shift.” *Advances in Neural Information Processing Systems* 32.
- Vovk, Vladimir. 2013. “Cross-Conformal Predictors.” *Annals of Mathematics and Artificial Intelligence* 74 (1-2): 9–28. <https://doi.org/10.1007/s10472-013-9368-4>.
- Vovk, Vladimir, Alexander Gammerman, and Glenn Shafer. 2005. *Algorithmic Learning in a Random World*. Springer-Verlag. <https://doi.org/10.1007/b106715>.
- Xu, Chen, and Yao Xie. 2021. “Conformal Prediction Interval for Dynamic Time-Series.” In *International Conference on Machine Learning*, 11559–69. PMLR.
- . 2023. “Sequential Predictive Conformal Inference for Time Series.” In *International Conference on Machine Learning*, 38707–27. PMLR.