Conformal Prediction Interval for Dynamic Time-Series

Chen Xu, Yao Xie

Industrial and Systems Engineering (ISyE)

Georgia Institute of Technology

CREATING THE NEXT®

Layout

- 1. Introduction
- 2. Problem Setup and Algorithm
- 3. Theoretical Analyses
- 4. Experiment
- 5. Conclusion and Extension

1. Introduction

- **Goal**: Construct Prediction Intervals for Time-series to address Uncertainty Quantification.
- **Challenges**:
    1. Dynamic data
    2. Spatio-Temporal Correlation
    3. Ensemble ML Models
    4. Distribution-Free

1. Introduction

- **Conformal Prediction**: Assign conformity score to quantify likeliness of a potential observation (Shafer and Vovk, 2008).

1. Introduction

- **Conformal Prediction**: Assign conformity score to quantify likeliness of a potential observation (Shafer and Vovk, 2008).
    1. Many works follow this logic: Regression (Papadopoulos et al., 2007; Barber et al., 2019) and Classification (Angelopoulos et al., 2020; Romano et al., 2020).
    2. Benefit: have exact marginal coverage guarantee.
    3. Limitation: guarantee only holds for exchangeable data.

1. Introduction

- **Conformal Prediction**: Assign conformity score to quantify likeliness of a potential observation (Shafer and Vovk, 2008).
    1. Many works follow this logic: Regression (Papadopoulos et al., 2007; Barber et al., 2019) and Classification (Angelopoulos et al., 2020; Romano et al., 2020).
    2. Benefit: have exact marginal coverage guarantee.
    3. Limitation: guarantee only holds for exchangeable data.
- **Our Contribution**
    1. Proposes **EnbPI**, an efficient wrapper around any ensemble model that has been trained.
    2. Theoretically, prediction intervals enjoy approximate marginal coverage, which is exact asymptotically.
    3. Empirically, **EnbPI** ensures exact marginal coverage and even conditional coverage on many tasks.
       Modification can handle missing data in graph networks and anomaly detection.

2. Problem Setup—Data and Objective

- Assume $Y_t = f(X_t) + \epsilon_t, t \geq 1$. $Y_t \in \mathbb{R}, X_t \in \mathbb{R}^d, \epsilon_t \sim F$ (CDF).
    1. Training data $\{(X_t, Y_t)\}_{t=1}^{T}$ are available.

2. Problem Setup—Data and Objective

- Assume $Y_t = f(X_t) + \epsilon_t, t \geq 1$. $Y_t \in \mathbb{R}, X_t \in \mathbb{R}^d, \epsilon_t \sim F$ (CDF).
  1. Training data $\{(X_t, Y_t)\}_{t=1}^{T}$ are available.
  2. For each $t > T$, we build $(1-\alpha)$-level prediction intervals $C_{T,t}^{\alpha}$ so that

$$P\left(Y_t \in C_{T,t}^{\alpha}\right) \geq 1 - \alpha, \tag{1}$$

  where $C_{T,t}^{\alpha}$ indicates previous T data are used to construct the interval at level $1-\alpha$.
  We call (1) as marginal coverage.
  Meanwhile, if X_t lies in a subspace $\mathcal{X} \subset \mathbb{R}^d$, conditional coverage requires

$$P\left(Y_t \in C_{T,t}^{\alpha} | X_t \in \mathcal{X}\right) \geq 1 - \alpha, \tag{2}$$

  which has been shown impossible to achieve without distributional assumption.
  Our method can empirically achieve (2) in some cases.

2. Problem Setup—Data and Objective

- Assume $Y_t = f(X_t) + \epsilon_t, t \geq 1$. $Y_t \in \mathbb{R}, X_t \in \mathbb{R}^d, \epsilon_t \sim F$ (CDF).
  1. Training data $\{(X_t, Y_t)\}_{t=1}^{T}$ are available.
  2. For each $t > T$, we build $(1-\alpha)$-level prediction intervals $C_{T,t}^{\alpha}$ so that
  $$P\left(Y_t \in C_{T,t}^{\alpha}\right) \geq 1 - \alpha, \tag{1}$$
  where $C_{T,t}^{\alpha}$ indicates previous T data are used to construct the interval at level $1-\alpha$.
  We call (1) as marginal coverage.
  Meanwhile, if X_t lies in a subspace $\mathcal{X} \subset \mathbb{R}^d$, conditional coverage requires
  $$P\left(Y_t \in C_{T,t}^{\alpha} | X_t \in \mathcal{X}\right) \geq 1 - \alpha, \tag{2}$$
  which has been shown impossible to achieve without distributional assumption.
  Our method can empirically achieve (2) in some cases.
  3. Further assume $\{Y_t\}_{t>T}$ are observable feedback after constructing $\{C_{T,t}^{\alpha}\}_{t>T}$, but not immediately (e.g. construct a batch of s intervals before observing all of these…).

2. Problem Setup—Prediction

- For prediction, fix $\mathscr{A} : \{\mathbb{R}^d \times \mathbb{R}\}^N \to (\mathbb{R}^d \to \mathbb{R})$ as a regression algorithm that trains on N data and outputs a predictor $\hat{f}$ that approximates f.

2. Problem Setup—Prediction

- For prediction, fix $\mathscr{A} : \{\mathbb{R}^d \times \mathbb{R}\}^N \to (\mathbb{R}^d \to \mathbb{R})$ as a regression algorithm that trains on N data and outputs a predictor $\hat{f}$ that approximates f (e.g. neural network architecture).
- In this work, we let $\hat{\epsilon}_i := |Y_i - \hat{f}_{-i}(X_i)|$ be the residual and construct $C^{\alpha}_{T,t}$ as:

$$C^{\alpha}_{T,t} := \hat{f}_{-t}\left(X_t\right) \pm (1-\alpha) \text{ quantile of } \left\{\hat{\epsilon}_i\right\}_{i=t-1}^{t-T}, \tag{3}$$

where $\hat{f}_{-j}$ implies 1) predictor $\hat{f}$ can differ at each index 2) datum (X_j, Y_j) is never used to train $\hat{f}_{-j}$.

2. Problem Setup—Prediction

- For prediction, fix $\mathscr{A} : \{\mathbb{R}^d \times \mathbb{R}\}^N \to (\mathbb{R}^d \to \mathbb{R})$ as a regression algorithm that trains on N data and outputs a predictor $\hat{f}$ that approximates f (e.g. neural network architecture).
- In this work, we let $\hat{\epsilon}_i := |Y_i - \hat{f}_{-i}(X_i)|$ be the residual and construct $C^{\alpha}_{T,t}$ as:

$$C^{\alpha}_{T,t} := \hat{f}_{-t}\left(X_t\right) \pm (1-\alpha) \text{ quantile of } \left\{\hat{\epsilon}_i\right\}_{i=t-1}^{t-T}, \tag{3}$$

  where $\hat{f}_{-j}$ implies 1) predictor $\hat{f}$ can differ at each index 2) datum (X_j, Y_j) is never used to train $\hat{f}_{-j}$.
- In ensemble learning (e.g., bootstrap aggregation), multiple bootstrap models $\hat{f}^b$ are then aggregated via ϕ (e.g., mean, median, trimmed mean) to improve prediction accuracy.

2. Algorithm—Big Picture

- Our ayAlgorithm will:
  1. Efficiently compute each $\hat{f}_{-t}, t \geq 1$ as an ensemble predictor. Importantly, doing so only requires the computational power of training one ensemble.
  2. Under mild assumptions, show $C^{\alpha}_{T,t}$ approximately satisfies (1) at each $t > T$.

$$P\left(Y_t \in C^{\alpha}_{T,t}\right) \geq 1 - \alpha \qquad (1)$$

2. Algorithm—Training

- Our **EnbPI** is designed for bagging estimators, with three parts:
- **Part 1**: Train bootstrap models. Typical bootstrap estimation, but this is the only time $\mathscr{A}$ is ever used.

1: **for** $b = 1, \ldots, B$ **do**
2: Sample with replacement an index set $S_b = (i_1, \ldots, i_T)$ from indices $(1, \ldots, T)$.
3: Compute $\hat{f}^b = \mathcal{A}(\{(x_i, y_i) \mid i \in S_b\})$.
4: **end for**

2. Algorithm—Training

- Our **EnbPI** is designed for bagging estimators, with three parts:
- **Part 1**: Train bootstrap models. Typical bootstrap estimation, but this is the only time $\mathscr{A}$ is ever used.

1: **for** $b = 1, \ldots, B$ **do**
2: Sample with replacement an index set $S_b = (i_1, \ldots, i_T)$ from indices $(1, \ldots, T)$.
3: Compute $\hat{f}^b = \mathcal{A}(\{(x_i, y_i) \mid i \in S_b\})$.
4: **end for**

- **Part 2**: Compute LOO residuals $\hat{\epsilon}_i$ from LOO ensemble models.
- Remarks:
  1. Choice of B: concentration inequality, J+aB (Kim et al. 2020).
  2. Compared with split conformal: no data-splitting or overfitting.
  3. Efficiency: no more fitting $\mathscr{A}$.

5: Initialize $\boldsymbol{\epsilon} = \{\}$
6: **for** $i = 1, \ldots, T$ **do**
7: $\hat{f}^{\phi}_{-i}(x_i) = \phi(\{\hat{f}^b(x_i) \mid i \notin S_b\})$
8: Compute $\hat{\epsilon}^{\phi}_i = |y_i - \hat{f}^{\phi}_{-i}(x_i)|$
9: $\boldsymbol{\epsilon} = \boldsymbol{\epsilon} \cup \{\hat{\epsilon}^{\phi}_i\}$
10: **end for**

2. Algorithm—Interval Construction

- **Part 3**: Construct prediction intervals with sliding.
- Remarks
  1. Adaptive width
  2. When $s \to \infty$
  3. Importance of Sliding
- Overall, **EnbPI** adopts a natural idea with simple implementation for real-world applications.

for $t = T+1, \ldots, T+T_1$ **do**
  Let $\hat{f}^{\phi}_{-t}(x_t) = (1-\alpha)$ quantile of $\{\hat{f}^{\phi}_{-i}(x_t)\}_{i=1}^{T}$
  Let $w_t^{\phi} = (1-\alpha)$ quantile of $\boldsymbol{\epsilon}$.
  Return $C^{\phi,\alpha}_{T,t}(x_t) = [\hat{f}^{\phi}_{-t}(x_t) \pm w_t^{\phi}]$
  if $t - T = 0 \bmod s$ **then**
    for $j = t-s, \ldots, t-1$ **do**
      Compute $\hat{\epsilon}_j^{\phi} = |y_j - \hat{f}^{\phi}_{-j}(x_t)|$
      $\boldsymbol{\epsilon} = (\boldsymbol{\epsilon} - \{\hat{\epsilon}_1^{\phi}\}) \cup \{\hat{\epsilon}_j^{\phi}\}$ and reset index of $\boldsymbol{\epsilon}$.
    end for
  end if
end for

3. Theoretical Guarantee

- WLOG, we analyze the coverage at $t = T + 1$, but Theorem holds at any $t > T$. Drop ϕ for notation simplicity.
- **Assumptions**:
  1. Data Regularity: The error process $\{\epsilon_t\}_{t \geq 1}$ is stationary, strongly mixing, with bounded sum of mixing coefficients. Their common CDF F is Lipschitz with $L > 0$.
  2. Estimation Quality: There exists a real sequence $\{\delta_T\}_{T \geq 1}$ that converges to zero such that $\sum_{t=1}^{T} (\hat{\epsilon}_t - \epsilon_t)^2 / T \leq \delta_T^2$.

3. Theoretical Guarantee

- WLOG, we analyze the coverage at $t = T + 1$, but Theorem holds at any $t > T$. Drop ϕ for notation simplicity.
- **Assumptions**:
  1. Data Regularity: The error process $\{\epsilon_t\}_{t\geq 1}$ is stationary, strongly mixing, with bounded sum of mixing coefficients. Their common CDF F is Lipschitz with $L > 0$.
  2. Estimation Quality: There exists a real sequence $\{\delta_T\}_{T\geq 1}$ that converges to zero such that $\sum_{t=1}^{T} (\hat{\epsilon}_t - \epsilon_t)^2/T \leq \delta_T^2$.
- **Theorem** *(Informally): For any* $\alpha \in (0,1)$, *the prediction interval* $C^{\alpha}_{T,T+1}(X_t)$ *from* ***EnbPI*** *satisfies*

$$\left| P\left(Y_{T+1} \notin C^{\alpha}_{T,T+1}\right) - \alpha \right| \leq C\left((\log T/T)^{1/3} + \delta_T^{2/3} \right)$$

- C can be identified under assumptions above.

3. Theoretical Analyses—Remarks

Theorem *(Informally): For any* $\alpha \in (0,1)$, *the prediction interval* $C^{\alpha}_{T,T+1}(X_t)$ *from* ***EnbPI*** *satisfies*

$$\left|P\left(Y_{T+1} \notin C^{\alpha}_{T,T+1}\right) - \alpha\right| \leq C\left((\log T/T)^{1/3} + \delta_T^{2/3}\right)$$

1. RHS Rate

1. Factor $(\log T/T)^{1/3}$ comes from assuming strongly mixing errors; different error assumptions (e.g., independent, stationary, etc.) yield different rates (Xu and Xie, 2021a, Corollary 1—3).
2. It is a worst-case analysis; empirical coverage is almost always $1-\alpha$ over all test points.

3. Theoretical Analyses—Remarks

Theorem *(Informally): For any* $\alpha \in (0,1)$, *the prediction interval* $C^{\alpha}_{T,T+1}(X_t)$ *from* ***EnbPI*** *satisfies*

$$\left|P\left(Y_{T+1} \notin C^{\alpha}_{T,T+1}\right) - \alpha\right| \leq C\left((\log T/T)^{1/3} + \delta_T^{2/3}\right)$$

1. RHS Rate

1. Factor $(\log T/T)^{1/3}$ comes from assuming strongly mixing errors; different error assumptions (e.g., independent, stationary, etc.) yield different rates (Xu and Xie, 2021a, Corollary 1—3).
2. It is a worst-case analysis; empirical coverage is almost always $1-\alpha$ over all test points.

2. **Assumption Implications:**

1. Assumption 1 allows arbitrary data dependency.
2. Assumption 2 holds true for many classes of f and algorithms $\mathscr{A}$ (Xu and Xie, 2021a, Section 4.2).
   In general, Assumption 2 requires asymptotically exact approximation of $\hat{f}$ to f.

3. Theoretical Analyses—Remarks

Theorem *(Informally): For any* $\alpha \in (0,1)$, *the prediction interval* $C^{\alpha}_{T,T+1}(X_t)$ *from* ***EnbPI*** *satisfies*

$$\left|P\left(Y_{T+1} \notin C^{\alpha}_{T,T+1}\right) - \alpha\right| \leq C\left((\log T/T)^{1/3} + \delta_T^{2/3}\right)$$

1. RHS Rate

1. Factor $(\log T/T)^{1/3}$ comes from assuming strongly mixing errors; different error assumptions (e.g., independent, stationary, etc.) yield different rates (Xu and Xie, 2021a, Corollary 1—3).
2. It is a worst-case analysis; empirical coverage is almost always $1-\alpha$ over all test points.

2. **Assumption Implications:**
   1. Assumption 1 allows arbitrary data dependency.
   2. Assumption 2 holds true for many classes of f and algorithms $\mathscr{A}$ (Xu and Xie, 2021a, Section 4.2).
      In general, Assumption 2 requires asymptotically exact approximation of $\hat{f}$ to f.
3. **Theorem Applicability:** holds for split conformal (Papadopoulos et al., 2007), but
   1. T in the Theorem becomes the size of calibration data.
   2. LOO ensemble predictor $\hat{f}_{-i}$ are in general better function approximators.

3. Theoretical Analyses—Proof Sketch

- Define $\hat{p}_{T+1} := T^{-1} \sum_{i=1}^{T} \mathbf{1}\left\{\hat{\epsilon}_i > \hat{\epsilon}_{T+1}\right\}$, whereby $Y_{T+1} \notin C^{\alpha}_{T,T+1}\left(X_{T+1}\right)$ iff $\hat{p}_{T+1} \leq \alpha$.

  The proof then analyzes how quickly $\hat{F}(x) := T^{-1} \sum_{i=1}^{T} 1\{\hat{\epsilon}_i \leq x\}$ $(\hat{F}(\hat{\epsilon}_{T+1}) = 1 - \hat{p}_{T+1})$ converges to $F(\epsilon_{T+1}) \sim \text{Unif}[0,1]$, whereby $\left| P\left(Y_{T+1} \notin C^{\alpha}_{T,T+1}\right) - \alpha \right|$ is also bounded.

3. Theoretical Analyses—Proof Sketch

- Define $\hat{p}_{T+1} := T^{-1} \sum_{i=1}^{T} \mathbf{1}\left\{\hat{\epsilon}_i > \hat{\epsilon}_{T+1}\right\}$, whereby $Y_{T+1} \notin C^{\alpha}_{T,T+1}\left(X_{T+1}\right)$ iff $\hat{p}_{T+1} \leq \alpha$.

  The proof then analyzes how quickly $\hat{F}(x) := T^{-1} \sum_{i=1}^{T} 1\{\hat{\epsilon}_i \leq x\}$ $(\hat{F}(\hat{\epsilon}_{T+1}) = 1 - \hat{p}_{T+1})$ converges to $F(\epsilon_{T+1}) \sim \text{Unif}[0,1]$, whereby $\left| P\left(Y_{T+1} \notin C^{\alpha}_{T,T+1}\right) - \alpha \right|$ is also bounded.

- **Lemma 1:**

  Define $\tilde{F}(x) := T^{-1} \sum_{i=1}^{T} \mathbf{1}\left\{\epsilon_i \leq x\right\}$. Under Assumption 1, there is an event A_T in the probability space of $\{\epsilon_t\}_{t=1}^{T}$, such that conditional on A_T, $\sup_x |\tilde{F}(x) - F(x)| \leq C(\log T/T)^{1/3}$. Moreover, $P(A_T^C) \leq C(\log T/T)^{1/3}$.

3. Theoretical Analyses—Proof Sketch

- Define $\hat{p}_{T+1} := T^{-1} \sum_{i=1}^{T} \mathbf{1}\left\{\hat{\epsilon}_i > \hat{\epsilon}_{T+1}\right\}$, whereby $Y_{T+1} \notin C^{\alpha}_{T,T+1}\left(X_{T+1}\right)$ iff $\hat{p}_{T+1} \leq \alpha$.

  The proof then analyzes how quickly $\hat{F}(x) := T^{-1} \sum_{i=1}^{T} 1\{\hat{\epsilon}_i \leq x\}$ $(\hat{F}(\hat{\epsilon}_{T+1}) = 1 - \hat{p}_{T+1})$ converges to $F(\epsilon_{T+1}) \sim \text{Unif}[0,1]$, whereby $\left| P\left(Y_{T+1} \notin C^{\alpha}_{T,T+1}\right) - \alpha \right|$ is also bounded.

- **Lemma 1:**

  Define $\tilde{F}(x) := T^{-1} \sum_{i=1}^{T} \mathbf{1}\left\{\epsilon_i \leq x\right\}$. Under Assumption 1, there is an event A_T in the probability space of $\{\epsilon_t\}_{t=1}^{T}$, such that conditional on A_T, $\sup_x |\tilde{F}(x) - F(x)| \leq C(\log T/T)^{1/3}$. Moreover, $P(A_T^C) \leq C(\log T/T)^{1/3}$.

- **Lemma 2:**

  Under Assumption 1 and 2, $\sup_x |\hat{F}(x) - \tilde{F}(x)| \leq C\delta_T^{2/3} + 2 \sup_x |\tilde{F}(x) - F(x)|$

- Together with Lemma 1 & 2, basic algebraic manipulation and triangle inequality prove the Theorem.

Experiment—Setup

- **Key Questions: EnbPI** on marginal coverage, conditional coverage (with missing data on networks), and anomaly detection.
- **Data:** Primarily use renewable energy data (e.g., solar and wind). Each time-series contains hourly data for a whole year. See Fig (a) & (b). We have marginal coverage results on data from other domains (Xu and Xie 2021a, Section 8.4).

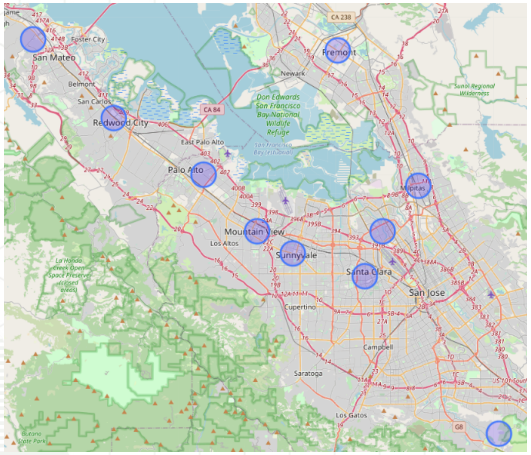


Fig (a): California non-uniform solar radiation sensors

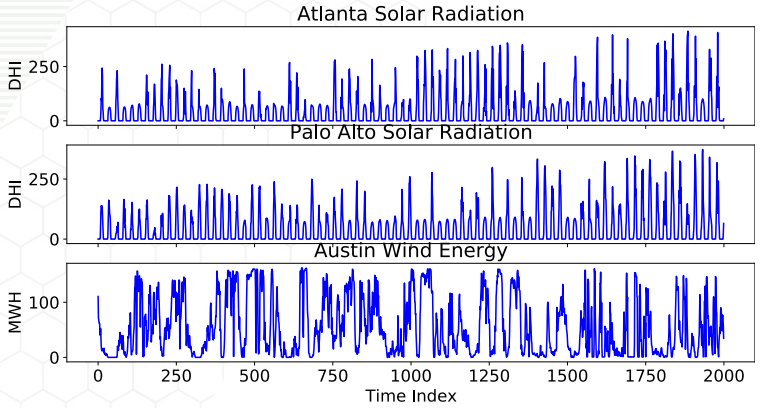


Fig (b): First 2000 Y_t of each time-series

Experiment—Setup

- **Key Questions: EnbPI** on marginal coverage, conditional coverage (with missing data on networks), and anomaly detection.
- **Data:** Primarily use renewable energy data (e.g., solar and wind). Each time-series contains hourly data for a whole year. See Fig (a) & (b).
  We have marginal coverage results on data from other domains (Xu and Xie 2021a, Section 8.4).
- **Features:** X_t is either univariate (past history of Y_t) or multivariate (other time series).
- **Regression** $\mathscr{A}$: Ridge, Random Forest (RF), Neural Network (NN), and RNN.
- **Competing Methods:** ARIMA(10,1,10), ICP, WeightedICP.
- Unless otherwise specified, we set ϕ=mean, B =30, and use the first 20% of total data for training (e.g., $T = 0.2(T + T_1)$). The stride s varies depending on our interests.

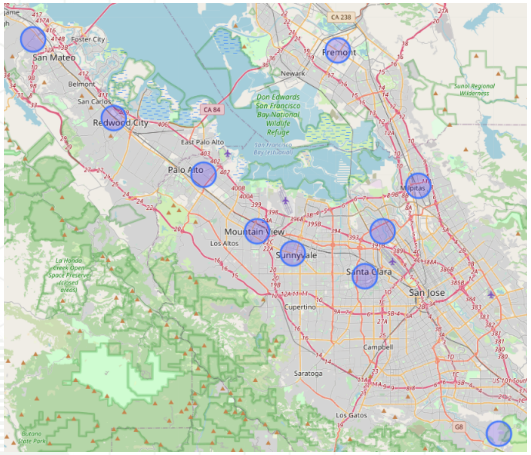


Fig (a): California non-uniform solar radiation sensors

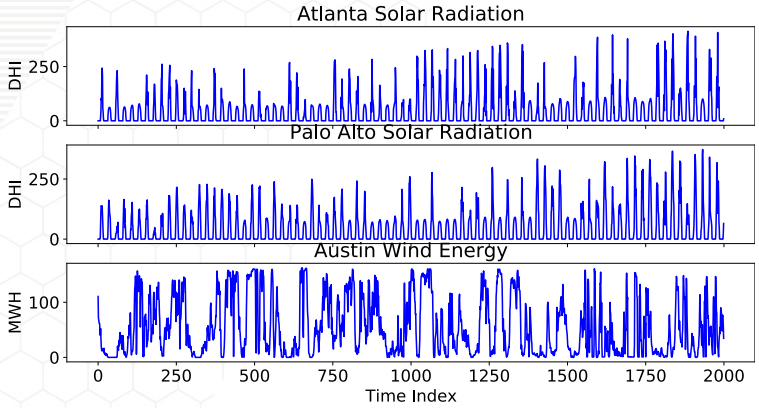
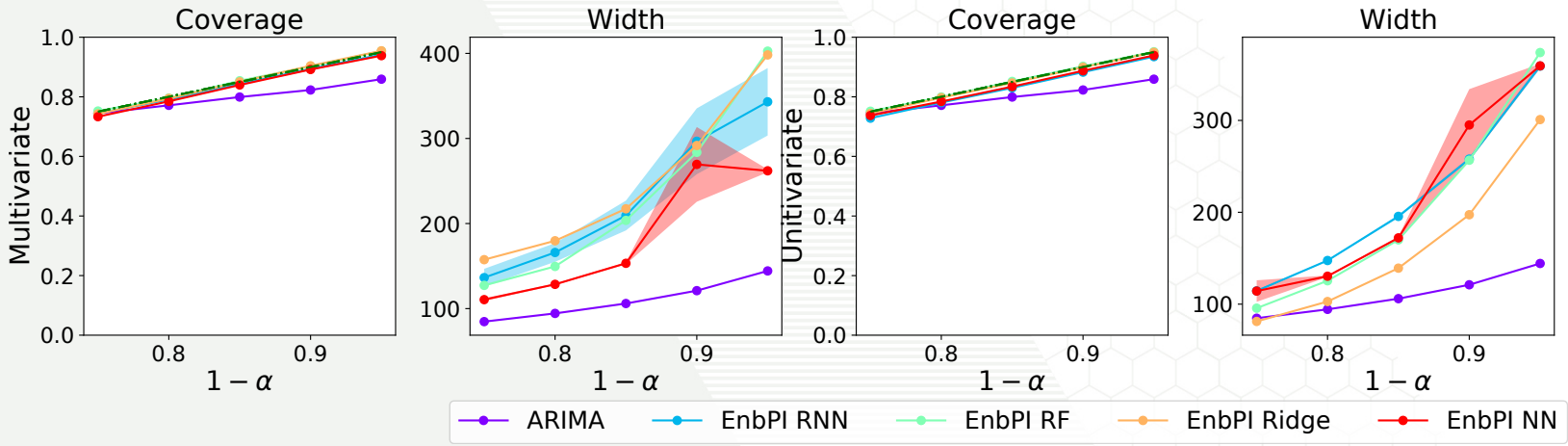
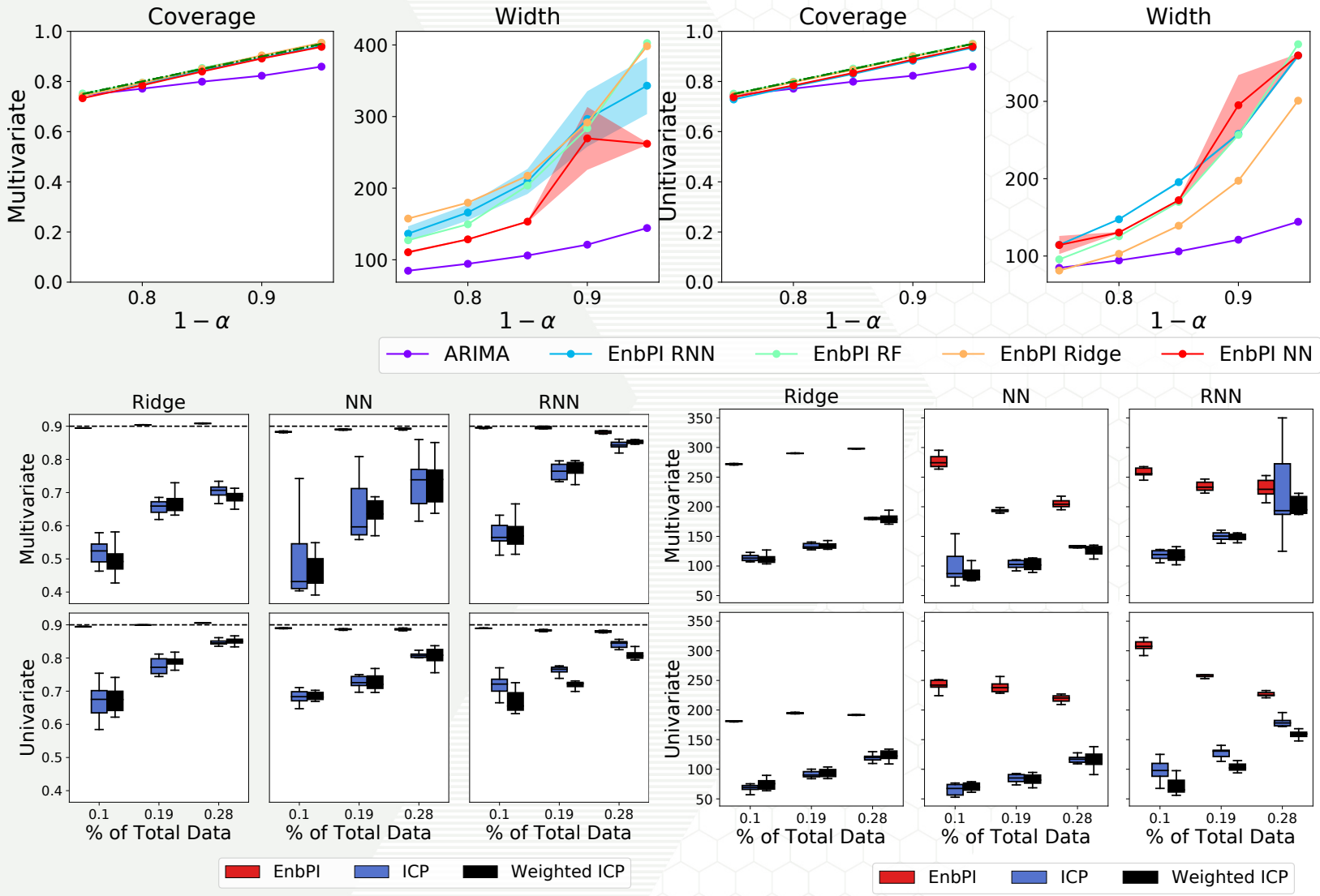


Fig (b): First 2000 Y_t of each time-series

Experiment—Marginal Coverage

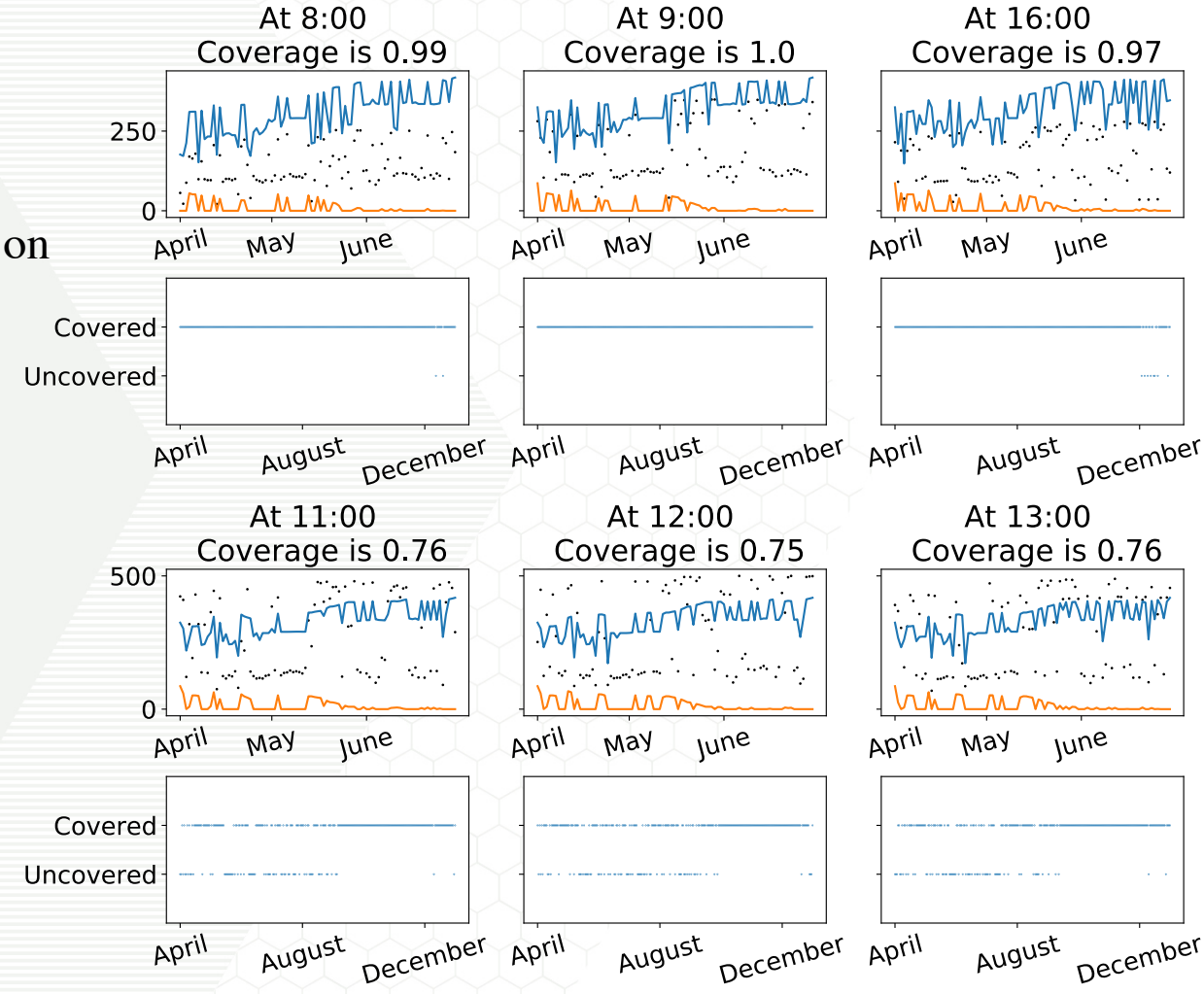


Experiment—Marginal Coverage



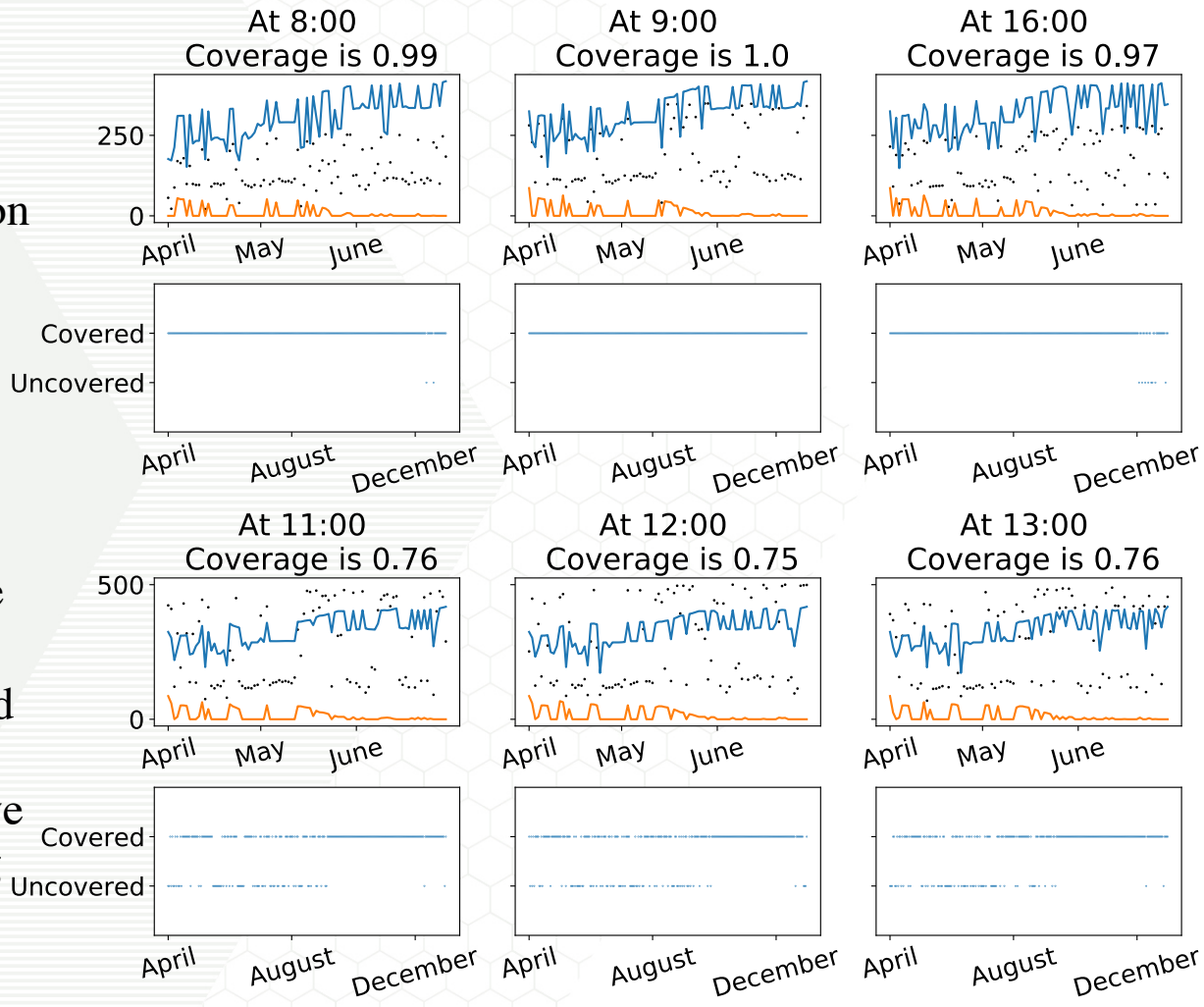
Experiment—Conditional Coverage

- We condition on each hourly coverage, by letting $s = 24$. Missing data are present.
- **Why this is conditional:** train on all hourly data but examine coverage by hour.
- **Handle missing flows:** We sample from empirical flows. More discussions see (Xu and Xie 2021a, Section 3.2)

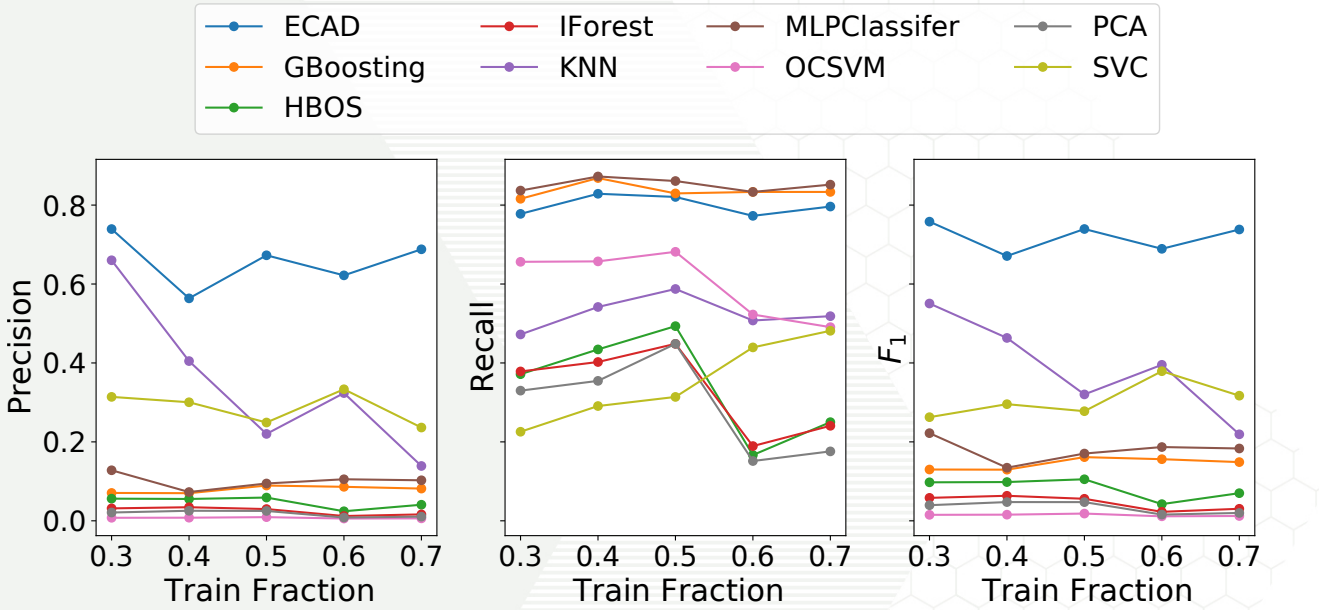


Experiment—Conditional Coverage

- We condition on each hourly coverage, by letting $s = 24$. Missing data are present.
- **Why this is conditional:** train on all hourly data but examine coverage by hour.
- **Handle missing flows:** We sample from empirical flows. More discussions see (Xu and Xie 2021a, Section 3.2)
- **How to improve coverage:** See (Xu and Xie 2021a, Figure 8)
- Results on network data (Xu and Xie 2021a, Section 8.3)
  The primary difference is that we define X_t to contain neighboring values as well.



Experiment—Anomaly Detection



- Use Financial Crime Data.
- The detector wraps around classification algorithms. Details see (Xu and Xie 2021a, Section 8.5).
- Significant improvement over other supervised and unsupervised detectors.

Conclusion

- Present a predictive inference method for dynamic time-series, useful in uncertainty quantification.
- Theoretically, **EnbPI** intervals are approximately marginally valid without assuming data exchangeability.
- Computationally, it is an efficient ensemble-based wrapper for many regression algorithms, including deep neural networks.
- Empirically, it is versatile on a wide range of time-series, including network data and data with missing entries, and maintains validity when traditional methods fail. It can even work well for anomaly detection.

Extension

- **Methodologically**,
  1. Adapt **EnbPI** for classification problems, especially those in computer vision.
  2. Better connect **EnbPI** with other applications, such as anomaly detection and sequential change-point detection
     See our latest work on conformal anomaly detection (Xu and Xie, 2021b).
- **Theoretically**,
  1. How and why LOO ensemble predictors are better function approximators than those in split conformal.
  2. Move towards conditional coverage analyses.
  3. Analyze widths of prediction intervals.

References

Xu, Chen and Xie, Yao (2021a). Conformal prediction interval for dynamic time-series. arXiv: 2010.09107 [stat.ME].

Xu, Chen and Xie, Yao (2021b). Conformal Anomaly Detection on Spatio-Temporal Observations with Missing Data. arXiv: 2105.11886 [stat.AP].

Kim, B., Xu, C., and Barber, R. F. (2020). Predictive inference is free with the jackknife+-after-bootstrap. Part of Advances in Neural Information Processing Systems 33 (NeurIPS 2020)

Angelopoulos, A., Bates, S., Malik, J., and Jordan, M. I. (2020). Uncertainty sets for image classifiers using conformal prediction. ArXiv, abs/2009.14193.

Barber, R. F., Candes, E. J., Ramdas, A., and Tibshirani, R. J. (2019b). Predictive inference with the jackknife+.

Papadopoulos, H., Vovk, V., and Gammerman, A. (2007). Conformal prediction with neural networks. In 19th IEEE International Conference on Tools with Artificial Intelligence(ICTAI 2007), volume 2, pages 388–395.

Romano, Y., Sesia, M., and Cande`s, E. (2020). Classification with valid and adaptive coverage. arXiv: Methodol- ogy.

Shafer, G. and Vovk, V. (2008). A tutorial on conformal prediction. Journal of Machine Learning Research, 9(Mar):371–421.