Prediction Inference of Non-linear AR Models with Bootstrap

Advancement to Candidacy

Kejin Wu

University of California, San Diego

Jun. 1 2023

UC San Diego

Contents

Background

- **Time series** is a discrete-time stochastic process, i.e., $\{X_t, t \in \mathbb{Z}\}$. Its realization is called time series data, e.g., heights of ocean tides, and counts of sunspots.

- **Prediction inference** is about determining Optimal Predictor (OP), usually in L_2 or L_1 sense, and Prediction Interval (PI), percentile or centered version, of future value X_{T+k}, $k \geq 1$, based on observed $\{X_0, \ldots, X_T\}$. We are concerned about the Coverage Rate (CVR) and Length (LEN) of PI.

Simple case:

If $\{X_0, \ldots, X_T\}$ are *i.i.d.* with a common distribution. Take sample mean and sample median to be L_2 and L_1 OPs, respectively. Rely on sample quantile values to build PIs.

If it is not, we can apply an invertible function to transform original data to be *i.i.d*, which is one type of Model-free prediction; see the work of Politis (2003); Chen and Politis (2019); Wang and Politis (2022).

Time series model:

We assume that the time series data is generated by some underlying mechanism:

$$X_t = G(\boldsymbol{X}_{t-p}, \epsilon_t), \tag{1}$$

where:

- $G(\cdot, \cdot)$ could be any suitable linear/non-linear function that makes the time series geometrically ergodic.
- ϵ_t is called innovation and assumed to be *i.i.d.* with appropriate moments and independent with X_{t-i}, $i \geq 1$.
- $\boldsymbol{X}_{t-p}$ represents $\{X_{t-1}, \ldots, X_{t-p}\}$.

Casual and invertible linear models, e.g., $X_t = \sum_{i=1}^{p} a_i X_{t-i} + \epsilon_t$; $\epsilon_t \sim N(0, \sigma_\epsilon^2)$.

➢ When the model information is *known*:

- One-step-ahead L_2 conditional OP $X_{T+1}^{L_2}$ is:

$$\mathbb{E}(X_{T+1}|X_T, \ldots, X_0) = \sum_{i=1}^{p} a_i X_{T+1-i}.$$

  Iterating this prediction can get multi-step-ahead L_2 OP $X_{T+k}^{L_2}$.

- A $(1-\alpha)\cdot 100\%$ PI of X_{T+k} centered at $X_{T+k}^{L_2}$ can be expressed as:

$$\left[X_{T+k}^{L_2} - z_{\alpha/2}\sqrt{\mathrm{Var}[e_T(k)]}, X_{T+k}^{L_2} + z_{\alpha/2}\sqrt{\mathrm{Var}[e_T(k)]}\right],$$

  where $e_T(k)$ is $X_{T+k} - X_{T+k}^{L_2}$.

➢ When the model information is *unknown*:

- Plug in the coefficient estimators $\{\hat{a}_i\}_{i=1}^{p}$.
- It fails if the innovation is non-normal.

Limitations: Infeasible to incorporate non-linear models; rely on the normality assumption; hard to extend to multi-step-ahead OP $X_{T+1}^{L_1}$.

Non-linear Autoregressive (NLAR) models, e.g., $X_t = m(\boldsymbol{X}_{t-p}) + \epsilon_t$.

➢ When the model information is *known*:

- One-step-ahead L_2 OP $X_{T+1}^{L_2} = m(\boldsymbol{X}_{T+1-p})$, conditional on observed data.
- For multi-step ahead prediction, the iterative method is no longer L_2 optimal. Pemberton (1987) may be the first attempt to get exact L_2 OP based on numerical integral. Guo et al. (1999) further developed an analytical predictor based on innovation distribution F_ϵ to approximate the exact L_2 OP.

➢ When the model information is *unknown*:

- For above methods, need to replace $m(\cdot)$ and F_ϵ with $\widehat{m}(\cdot)$ and $\widehat{F}_\epsilon$, respectively.

See Lee and Billings (2003) for a review.

Limitations: Hard to find L_1 OP and PI.

MC simulation for k-step-ahead prediction:

In the context of prediction, especially for NLAR models, Monte Carlo (MC) simulation and Bootstrap can give a way to out the dilemma.

Take general model Eq. (1) as an example, when the model is *known* to us:

➢ Apply MC simulation to do predictions:

- Simulate $\{\epsilon_{T+1}^{(i)}, \ldots, \epsilon_{T+k}^{(i)}\}_{i=1}^{M}$ from F_ϵ.
- Compute pseudo $\{X_{T+k}^{(i)}\}_{i=1}^{M}$, i.e., $X_{T+j}^{(i)} = G(\boldsymbol{X}_{T+j-p}, \epsilon_{T+j}^{(i)})$, for $j = 1, \ldots, k$.
- Take sample mean and median of $\{X_{T+k}^{(i)}\}_{i=1}^{M}$ to approximate $X_{T+k}^{L_2}$ and $X_{T+k}^{L_1}$, respectively. Take corresponding quantile values to approximate PIs with arbitrary coverage rates. We call such type of PI Simulation PI (SPI).

Limitations: In practice, model information is generally not known to participators. Thus, this prediction is *Oracle*.

The first step of our work—Bootstrap for k-step-ahead prediction:

When the model information is *unknown* to us, the model $G(\cdot,\cdot)$ and innovation distribution F_ϵ need to be estimated. Assume we can get a consistent estimator $\widehat{G}(\cdot,\cdot)$ and $\widehat{F}_\epsilon$ which is the empirical distribution of residuals, then:

➢ Apply Bootstrap to do predictions:

- Bootstrap $\{\hat{\epsilon}^{(i)}_{T+1},\ldots,\hat{\epsilon}^{(i)}_{T+k}\}_{i=1}^{M}$ from $\widehat{F}_\epsilon$.
- Compute pseudo $\{\widehat{X}^{(i)}_{T+k}\}_{i=1}^{M}$ iteratively, i.e., $\widehat{X}^{(i)}_{T+j}=\widehat{G}(\boldsymbol{X}_{T+j-p},\hat{\epsilon}^{(i)}_{T+j})$, for $j=1,\ldots,k$.
- Take sample mean and median of $\{\widehat{X}^{(i)}_{T+k}\}_{i=1}^{M}$ to approximate $X^{L_2}_{T+k}$ and $X^{L_1}_{T+k}$, respectively. Take corresponding quantile values to approximate PIs with arbitrary coverage rates. We call such type of PI Quantile PI (QPI).

Limitations: In practice with finite samples, this Bootstrap-based PI suffers undercoverage due to several reasons; the requirement to get a consistent model estimation and separate residuals.

Forward bootstrap prediction for a general model

Politis (2015) proposed the *forward bootstrap* method to include the estimation variability into the PI for improving the coverage performance for a small sample size. They also applied the *predictive residuals* to build PI.

We further extend this bootstrap prediction method to the general model Eq. (1) $X_t = G(\boldsymbol{X}_{t-p}, \epsilon_t)$.

Main Algorithm:

- *Step 1*: Do estimations to get $\widehat{G}(\cdot,\cdot)$ and $\widehat{F}_\epsilon$. Then, perform the bootstrap prediction to get $\widehat{X}_{T+k}$.
- *Step 2*: Generate a pseudo series $\{X_0^*, \ldots, X_{T+k}^*\}$ by viewing $\widehat{G}(\cdot,\cdot)$ and $\widehat{F}_\epsilon$ as the true model and innovation distribution in the bootstrap world.
- *Step 3*: Re-estimate model to get $\widehat{G}^*(\cdot,\cdot)$ with $\{X_0^*, \ldots, X_T^*\}$; Re-define $\{X_{T-p+1}^* = X_{T-p+1}, \ldots, X_T^* = X_T\}$. Then do the bootstrap prediction with $\widehat{G}^*(\cdot,\cdot)$ and $\widehat{F}_\epsilon$ to get $\widehat{X}_{T+k}^*$. Record the predictive root $X_{T+k}^* - \widehat{X}_{T+k}^*$ in the bootstrap world.
- *Step 4*: Repeat the above process M times, collect M predictive roots and take its empirical distribution to approximate the distribution of $X_{T+k} - \widehat{X}_{T-k}$.
- *Step 5*: The $(1-\alpha)100\%$ PI for X_{T+k} centered at $\widehat{X}_{T+k}$ can be approximated by $[\widehat{X}_{T+k} + q(\alpha/2), \widehat{X}_{T+k} + q(1-\alpha/2)]$, where $q(\alpha)$ is the α-quantile of the empirical distribution of $X_{T+k}^* - \widehat{X}_{T+k}^*$.

Remark 1: This algorithm provides a framework to do predictions when model information is unknown. It can directly work for the cases where $G(\cdot, \cdot)$ is in a parametric or non-parametric form.

Remark 2: The deployment of Step 3 to get OP in the bootstrap world is necessary since it exactly mimics the prediction process in the real-world bootstrap for NLAR models, so that the center of bootstrap PI is still at the OP.

Remark 3: The model estimation variability is captured due to Step 3 again.

For some specific linear/non-linear models, this *forward bootstrap* method presents better prediction performance according to the empirical CVR and LEN compared to other bootstrap-based methods; see Thombs and Schucany (1990); Pascual et al. (2004, 2006) for other approaches.

The success is due to that we can get a *pertinence* PI (PPI) with this specifically designed algorithm. The pertinence comes from capturing the model estimation variability.

Key components of PPI:

- $\sup_a |\widehat{F}_\epsilon(a) - F_\epsilon(a)| \xrightarrow{p} 0$.
- $\sup_a |\mathbb{P}(\tau_T A_m^* \le a) - \mathbb{P}(\tau_T A_m \le a)| \xrightarrow{p} 0$,

For example, we assume that we can decompose $G(\boldsymbol{X}_{t-p}, \epsilon_t)$ as $M(\boldsymbol{X}_{t-p}) + \epsilon_t$; $A_m^* = \widehat{M}^*(\boldsymbol{x}) - \widehat{M}(\boldsymbol{x})$; $A_m = \widehat{M}(\boldsymbol{x}) - M(\boldsymbol{x})$.

Remark 4: Compared to the recent application of Conformal Prediction on time series; see Xu and Xie (2021); similar preliminary conditions are assumed. However, their prediction focus on *unconditional* PIs which satisfy the minimal coverage rate, i.e., $\mathbb{P}(X_{T+k} \in C_{T+k}^{\alpha}) \ge 1 - \alpha$.

Forward bootstrap prediction for parametric NLAR models

First, we consider the case that we can decompose $G(\boldsymbol{X}_{t-p}, \epsilon_t)$ as a parametric non-linear model and innovation[1]:

$$X_t = G(X_{t-1}, \epsilon_t) = m(X_{t-1}, \theta_1) + \sigma(X_{t-1}, \theta_2)\epsilon_t, \tag{2}$$

where:

- $m(\cdot)$ is the mean function which is Lipschitz continuous w.r.t. the first and second arguments for their domain, respectively.
- $\sigma(\cdot)$ is the positive and bounded variance function which is Lipschitz continuous w.r.t. the first and second arguments for their domains, respectively.
- $\theta_1 \in \Theta_1$ and $\theta_2 \in \Theta_2$, where Θ_1 and Θ_2 are all bounded sets in $\mathbb{R}^d$.
- For ϵ_t, it is mean zero and variance 1; $f_\epsilon(\cdot)$ is continuous and everywhere positive.

We further assume that the time series is geometrically ergodic.

[1] To simplify notation, we consider models with order 1.

Under the consistency of parameter estimations, i.e., $\widehat{\theta_1} \xrightarrow{p} \theta_1$ and $\widehat{\theta_2} \xrightarrow{p} \theta_2$. With additional suitable assumptions on the moments condition of ϵ_t. We provide some results based on $\{X_0, \ldots, X_T\} \in \Omega_T$, where $\mathbb{P}((X_0, \ldots, X_T) \in \Omega_T) = o(1)$ as $T \to \infty$.

Preliminary lemmas:

- *Lemma 1* (according to Theorem 2 of Franke et al. (2004)) For the bootstrap series $\{X_t^*\}_{t=0}^T$ generated by *Step 2* of Main Algorithm, it is also geometrically ergodic.
- *Lemma 2* (according to Theorem 3 of Franke et al. (2004)) For the stationary distribution of the bootstrap series, it can mimic the stationary distribution of the original series closely:

$$\sup_B |\Pi(B) - \Pi^*(B)| = o(1),$$

which holds for all measurable sets B, where $\Pi(B)$ and $\Pi^*(B)$ represent stationary distribution for real series and bootstrap series, respectively.

The consistency of parameter estimations can be guaranteed by applying the Non-linear Least Square technique[2]. We take a two-step estimation process to find $\widehat{\theta_1}$ and $\widehat{\theta_2}$:

- $$\widehat{\theta_1} = \arg\min_{\vartheta \in \Theta_1} L_T(\vartheta) = \arg\min_{\vartheta \in \Theta_1} \frac{1}{T} \sum_{t=1}^{T} (X_t - m(X_{t-1}, \vartheta))^2$$
- $$\widehat{\theta_2} = \arg\min_{\vartheta \in \Theta_2} K_T(\vartheta, \widehat{\theta_1}) = \arg\min_{\vartheta \in \Theta_2} \left| \frac{1}{T} \sum_{t=1}^{T} \left(\frac{X_t - m(X_{t-1}, \widehat{\theta_1})}{\sigma(X_{t-1}, \vartheta)} \right)^2 - 1 \right|.$$

We assume that we can correctly specify the parametric non-linear model, and θ_1, θ_2 uniquely minimize $L(\vartheta)$ and $K(\vartheta, \theta_1)$.

[2] One advantage of applying this estimation method is that the predictive residuals can be performed easily, i.e., we estimate models based on the available data X_i vs. $\{X_{i-p}, \ldots, X_{i-1}\}$ excludes the single point at $i = t$ to get the predictive residual $\hat{\epsilon}_t^p$.

Theorem 1 (Consistency of OP and asymptotic validity of QPI):

Let $\{X_t\}$ satisfy Eq. (2). For $k \geq 1$ we have:

$$\sup_{|x| \leq c_T} \left| F_{X^*_{T+k}|X_T,\ldots,X_0}(x) - F_{X_{T+k}|X_T}(x) \right| \xrightarrow{p} 0, \tag{3}$$

where

- $X^*_{T+k} = \mathcal{G}(X_T; \hat{\epsilon}^*_{T+1}, \ldots, \hat{\epsilon}^*_{T+k}; \widehat{\theta})$. This is computed by $X^*_{T+i} = m(X^*_{T+i-1}, \widehat{\theta}_1) + \sigma(X^*_{T+i-1}, \widehat{\theta}_2)\hat{\epsilon}^*_{T+i}$ iteratively for $i = 1, \ldots, k$. Similar for X_{T+k}.
- $\{\hat{\epsilon}^*_i\}_{i=T+1}^{T+k}$ are $i.i.d. \sim \widehat{F}_\epsilon$.
- c_T is an appropriate sequence converges to infinity as T converges to infinity.
- $F_{X^*_{T+k}|X_T,\ldots,X_0}(x)$ is the distribution of k-step ahead future value in the bootstrap world, i.e., conditional on all observed data.
- $F_{X_{T+k}|X_T}(x)$ is the distribution of k-step ahead future value in the real world.

◁

Theorem 2 (Estimation inference of $\widehat{\theta_1}$ and $\widehat{\theta_2}$, $\widehat{\theta_1^*}$ and $\widehat{\theta_2^*}$):

Based on the realization $\{X_0, \ldots, X_T\} \in \Omega_T$, under other suitable assumptions, we have:

$$\sqrt{T}(\widehat{\theta_1} - \theta_1) \xrightarrow{d} N(0, B_1^{-1}\Omega_1 B_1^{-1}) \; ; \;\; \sqrt{T}(\widehat{\theta_2} - \theta_2) \xrightarrow{d} N(0, B_2^{-1}\Omega_2 B_2^{-1}). \tag{4}$$

where

- $\Omega_1 = 4 \cdot \mathbb{E}(\sigma(X_0, \theta_2) R_1 \sigma(X_0, \theta_2))$; $B_1 = 2 \cdot \mathbb{E}\left(\nabla\phi(X_0, \theta_1)(\nabla\phi(X_0, \theta_1))^\top\right)$; $R_1 = \nabla\phi(X_0, \theta_1)\, \nabla\phi(X_0, \theta_1)^\top$; here ∇ is the gradient operator w.r.t. θ_1.
- $\Omega_2 = 4 \cdot \mathbb{E}(B_3 R_2 B_3^\top)$; $B_3 = \mathbb{E}(\nabla g(X_1, X_0, \theta_2, \theta_1))$; $R_2 = (g(X_1, X_0, \theta_2, \theta_1) - 1)^2$; $B_2 = 2 \cdot (\mathbb{E}(\nabla g(X_1, X_0, \theta_2, \theta_1)) \cdot (\mathbb{E}(\nabla g(X_1, X_0, \theta_2, \theta_1))^\top$; $g(X_1, X_0, \theta_2, \theta_1) = \left(\frac{X_1 - \phi(X_0, \theta_1)}{\sigma(X_0, \theta_2)}\right)^2$; here ∇ is the gradient operator w.r.t. θ_2.

Theorem 2 continued:

To derive the estimation inference of $\widehat{\theta}_1^*$ and $\widehat{\theta}_2^*$, we need to *center* the residuals to be mean 0 and *normalize* its variance to 1. Then, with *Lemma 1* and *Lemma 2*, we further have:

$$\sqrt{T}(\widehat{\theta}_1^* - \widehat{\theta}_1) \xrightarrow{d} N(0, B_1^{-1}\Omega_1 B_1^{-1}) \; ; \;\; \sqrt{T}(\widehat{\theta}_2^* - \widehat{\theta}_2) \xrightarrow{d} N(0, B_2^{-1}\Omega_2 B_2^{-1}). \tag{5}$$

◁

Simulation example (Threshold model):

$$X_t = (0.5 \cdot X_{t-1} + 0.2 \cdot X_{t-2} + 0.1 \cdot X_{t-3})I(X_{t-1} \leq 0) + (0.8 \cdot X_{t-1})I(X_{t-1} > 0) + \epsilon_t.$$
$$\epsilon_t \sim N(0, 1). \tag{6}$$

Simulation setting:

We take the number of bootstrap times $M = 1000$. We repeat simulations $N = 5000$ times. We take $\alpha = 0.05$.

Simulation measurement:

- $$\text{CVR of the } k\text{-th step ahead prediction} = \frac{1}{N}\sum_{n=1}^{N} I_{X_{n,k} \in [L_{n,k}, U_{n,k}]}, \text{ for } k = 1, \ldots, 5. \tag{7}$$
- $$\text{LEN of the } k\text{-th step ahead PI} = \frac{1}{N}\sum_{n=1}^{N}(U_{n,k} - L_{n,k}), \text{ for } k = 1, \ldots, 5, \tag{8}$$

where $[L_{n,k}, U_{n,k}]$ and $X_{n,k}$ represent k-th step ahead prediction intervals and the true future value in the n-th replication, respectively.

Simulation results:

Table: The CVR and LEN of PIs for Model Eq. (6)

Threshold Model:	$X_t = (0.5 \cdot X_{t-1} + 0.2 \cdot X_{t-2} + 0.1 \cdot X_{t-3})I(X_{t-1} \le 0) + (0.8 \cdot X_{t-1})I(X_{t-1} > 0) + \epsilon_t$									
	CVR for each step					LEN for each step				
$T = 400$	1	2	3	4	5	1	2	3	4	5
QPI-f	0.9420	0.9506	0.9468	0.9444	0.9372	3.88	4.68	5.11	5.40	5.58
QPI-p	0.9462	0.9512	0.9502	0.9474	0.9428	3.92	4.72	5.16	5.45	5.64
L_2-PPI-f	0.9446	0.9510	0.9486	0.9470	0.9408	3.90	4.71	5.15	5.44	5.63
L_2-PPI-p	0.9466	0.9542	0.9516	0.9494	0.9434	3.94	4.75	5.20	5.49	5.69
L_1-PPI-f	0.9448	0.9518	0.9478	0.9468	0.9402	3.90	4.71	5.15	5.44	5.62
L_1-PPI-p	0.9470	0.9544	0.9500	0.9486	0.9436	3.94	4.75	5.20	5.49	5.68
SPI	0.9446	0.9534	0.9508	0.9510	0.9454	3.90	4.71	5.16	5.46	5.65
$T = 100$										
QPI-f	0.9270	0.9304	0.9294	0.9272	0.9250	3.81	4.57	4.98	5.23	5.40
QPI-p	0.9370	0.9412	0.9368	0.9372	0.9372	3.98	4.76	5.19	5.46	5.63
L_2-PPI-f	0.9358	0.9352	0.9338	0.9314	0.9298	3.95	4.71	5.13	5.40	5.59
L_2-PPI-p	0.9454	0.9454	0.9444	0.9430	0.9418	4.10	4.90	5.34	5.63	5.83
L_1-PPI-f	0.9364	0.9360	0.9336	0.9310	0.9304	3.95	4.71	5.13	5.39	5.58
L_1-PPI-p	0.9450	0.9456	0.9432	0.9422	0.9412	4.11	4.90	5.33	5.62	5.81
SPI	0.9446	0.9472	0.9498	0.9474	0.9478	3.90	4.71	5.16	5.46	5.65
$T = 50$										
QPI-f	0.8980	0.9054	0.9018	0.8950	0.8926	3.66	4.47	4.87	5.14	5.38
QPI-p	0.9260	0.9314	0.9272	0.9218	0.9212	4.05	4.97	5.42	5.74	5.99
L_2-PPI-f	0.9340	0.9268	0.9214	0.9164	0.9152	4.22	5.10	5.86	6.89	8.97
L_2-PPI-p	0.9522	0.9478	0.9404	0.9400	0.9376	4.60	5.57	6.36	7.33	9.03
L_1-PPI-f	0.9338	0.9268	0.9194	0.9144	0.9130	4.23	5.09	5.82	6.79	8.71
L_1-PPI-p	0.9522	0.9482	0.9384	0.9378	0.9356	4.61	5.55	6.30	7.20	8.71
SPI	0.9494	0.9448	0.9464	0.9458	0.9462	3.90	4.71	5.16	5.46	5.65

Note: "-f" and "-p" represent fitted and predictive residuals, respectively. "L_2" and "L_1" represent the center of PPI is L_2 and L_1 OP, respectively.

Non-parametric forward bootstrap: debiasing and pertinence

When the parametric format of Eq. (2) is unknown, we assume that we only know the data-generating mechanism of time series consists of two parts:

$$X_t = G(X_{t-1}, \epsilon_t) = m(X_{t-1}) + \sigma(X_{t-1})\epsilon_t, \tag{9}$$

we can consider a non-parametric approach to estimate the mean and variance parts of Eq. (9). We focus on Local Constant estimators. Other estimators, such as Local Linear can be deployed similarly.

Local Constant Estimator:

$$\widetilde{m}_h(x) = \frac{\sum_{t=1}^{T} K(\frac{x-X_{t-1}}{h}) X_t}{\sum_{t=1}^{T} K(\frac{x-X_{t-1}}{h})} \text{ and } \widetilde{\sigma}_h(x) = \frac{\sum_{t=1}^{T} K(\frac{x-X_{t-1}}{h})(X_t - \widetilde{m}_h(X_{t-1}))^2}{\sum_{t=1}^{T} K(\frac{x-X_{t-1}}{h})}; \tag{10}$$

For simplifying notation, we use h to represent the bandwidth of kernel functions; h may take a different value for mean and variance estimators. Due to the theoretical and practical issues, we truncate the above naive local constant estimators as below:

$$\widehat{m}_h(x) = \begin{cases} -C_m & \text{if } \widetilde{m}_h(x) < -C_m \\ \widetilde{m}_h(x) & \text{if } |\widetilde{m}_h(x)| \le C_m \\ C_m & \text{if } \widetilde{m}_h(x) > C_m \end{cases}; \widehat{\sigma}_h(x) = \begin{cases} c_\sigma & \text{if } \widetilde{\sigma}_h(x) < c_\sigma \\ \widetilde{\sigma}_h(x) & \text{if } c_\sigma \le \widetilde{\sigma}_h(x) \le C_\sigma \\ C_\sigma & \text{if } \widetilde{\sigma}_h(x) > C_\sigma \end{cases}, \tag{11}$$

where C_m and C_σ are large enough and c_σ is small enough.

Under the assumptions of Franke et al. (2002), we can get some properties about Local Constant estimators.

Preliminary results according to Franke et al. (2002):

- $\sup_{|x|\leq c_T} |\widehat{m}_h(x) - m(x)| \xrightarrow{p} 0$ and $\sup_{|x|\leq c_T} |\widehat{\sigma}_h(x) - \sigma(x)| \xrightarrow{p} 0$.
- $\sup_{x\in\mathbb{R}} |\widehat{F}_\epsilon(x) - F_\epsilon(x)| \xrightarrow{p} 0$,

where

- c_T is a suitable sequence that converges to infinity as T converges to infinity.
- h is determined based on data with optimal rate $O(T^{-1/5})$.

As a result, similar to Theorem 1, we can show the consistency of OP and asymptotic validity of QPI for this non-parametric forward bootstrap prediction.

Debiasing strategies:

It is well known that the center of non-parametric estimation distribution revealed by CLT is asymptotic non-zero if we take the bandwidth with the optimal rate.

Let $\widehat{m}_g(x)$ and $\widehat{\sigma}_g(x)$ be estimated mean and variance functions to generate bootstrap series in the bootstrap world. If we want to mimic the non-parametric estimator distribution by bootstrap, there are three standard approaches to handle the bias:

1. Let $g = h$ and take a bandwidth rate satisfying $hT^{1/5} \to 0$, i.e., under-smoothing bandwidth.
2. Keep using the optimal rate $h = O(T^{-1/5})$, but take over-smoothing g, i.e., $g \neq h$ and $g/h \to \infty$.
3. Perform additional bias correction via estimating this term.

See Politis (2022) for discussions on 1 and 3; see Franke et al. (2002) for a reference on 2.

Debiasing for (multi-step-ahead) non-parametric prediction:

For the one-step-ahead prediction, it has been discussed in Pan and Politis (2016). For analyses of the multi-step ahead case, we take the two-step ahead predictive root as example[3]:

$$\begin{aligned} X_{T+2} - \widehat{X}_{T+2} &= m(X_{T+1}) + \epsilon_{T+2} - \frac{1}{M}\sum_{i=1}^{M}\left(\widehat{m}_h\left(\widehat{m}_h(X_T) + \hat{\epsilon}_{i,T+1}\right) + \hat{\epsilon}_{i,T+2}\right) \\ &\approx m(m(X_T) + \epsilon_{T+1}) + \epsilon_{T+2} - \frac{1}{M}\sum_{i=1}^{M}\widehat{m}_h\left(\widehat{m}_h(X_T) + \hat{\epsilon}_{i,T+1}\right). \end{aligned} \tag{12}$$

$$\begin{aligned} X^*_{T+2} - \widehat{X}^*_{T+2} &= \widehat{m}_g(X^*_{T+1}) + \hat{\epsilon}^*_{T+2} - \frac{1}{M}\sum_{i=1}^{M}\left(\widehat{m}^*_h\left(\widehat{m}^*_h(X_T) + \hat{\epsilon}^*_{i,T+1}\right) + \hat{\epsilon}^*_{i,T+2}\right) \\ &\approx \widehat{m}_g(\widehat{m}_g(X_T) + \hat{\epsilon}^*_{T+1}) + \hat{\epsilon}^*_{T+2} - \frac{1}{M}\sum_{i=1}^{M}\widehat{m}^*_h\left(\widehat{m}^*_h(X_T) + \hat{\epsilon}^*_{i,T+1}\right). \end{aligned} \tag{13}$$

[3] To simplify the notation, we take $\sigma(\cdot) \equiv 1$.

Further simplify Eq. (12) and Eq. (13) by Taylor expansion:

$$X_{T+2} - \widehat{X}_{T+2} = m(m(X_T)) - \widehat{m}_h(\widehat{m}_h(X_T)) + m^{(1)}(\hat{x})\epsilon_{T+1} + \epsilon_{T+2} - \frac{1}{M}\sum_{i=1}^{M} \widehat{m}_h^{(1)}(\hat{\hat{x}}_i)\hat{\epsilon}_{i,T+1};$$

$$X^*_{T+2} - \widehat{X}^*_{T+2} = \widehat{m}_g(\widehat{m}_g(X_T)) - \widehat{m}^*_h(\widehat{m}^*_h(X_T)) + \widehat{m}_g^{(1)}(\hat{x}^*)\hat{\epsilon}^*_{T+1} + \hat{\epsilon}^*_{T+2} - \frac{1}{M}\sum_{i=1}^{M} \widehat{m}_h^{*(1)}(\hat{\hat{x}}^*_i)\hat{\epsilon}^*_{i,T+1}. \tag{14}$$

We can think the r.h.s of Eq. (14) is made up of two components in both real and bootstrap worlds:

- The two-step ahead estimation variability component, $m(m(X_T)) - \widehat{m}_h(\widehat{m}_h(X_T))$ and $\widehat{m}_g(\widehat{m}_g(X_T)) - \widehat{m}^*_h(\widehat{m}^*_h(X_T))$.
- Other terms, which are related to future innovations.

For the second component, the bootstrap can mimic the real-world situation well.

Theorem 3 (Confidence bound for multi-step ahead estimation function):

For $(X_0, \ldots, X_T) \in \Omega_T$, by taking the bandwidth strategy 1, we can build confidence bound for the local constant estimator at k-step by bootstrap:

$$\begin{aligned} \sup_{|x| \leq c_T} \Big| & \mathbb{P}\Big(\sqrt{Th}\Big(\mathcal{M}_k(X_T) - \widehat{\mathcal{M}}_{h,k}(X_T)\Big) \leq x\Big) - \\ & \mathbb{P}\Big(\sqrt{Th}\Big(\mathcal{M}^*_{h,k}(X_T) - \widehat{\mathcal{M}}^*_{h,k}(X_T)\Big) \leq x\Big) \Big| \xrightarrow{p} 0, \text{ for } k \geq 1. \end{aligned} \tag{15}$$

Above convergence result stands true for $X_T \in S$, where S is a large enough interval; $\mathcal{M}_k(X_T)$ can be expressed by computing $X_{T+i} = m(X_{T+i-1})$ iteratively for $i = 1, \ldots, k$, i.e., it has a form in below:

$$\mathcal{M}_k(X_T) = m(m(\cdots m(m(X_T))\cdots)). \tag{16}$$

$\widehat{\mathcal{M}}_{h,k}(X_T)$ can be expressed by computing $X_{T+i} = \widehat{m}_h(X_{T+i-1})$ iteratively for $i = 1, \ldots, k$, i.e., it has a form in below:

$$\widehat{\mathcal{M}}_{h,k}(X_T) = \widehat{m}_h(\widehat{m}_h(\cdots \widehat{m}_h(\widehat{m}_h(X_T))\cdots)). \tag{17}$$

$\mathcal{M}^*_{h,k}(X_T)$ and $\widehat{\mathcal{M}}^*_{h,k}(X_T)$ can be expressed similarly.

Simulation example (NLAR model with heteroscedastic errors):

$$X_t = \sin(X_{t-1}) + \epsilon_t \sqrt{0.5 + 0.25X_{t-1}^2}, \epsilon_t \sim N(0,1). \tag{18}$$

Simulation setting:

We take the number of bootstrap times $M = 500$. We repeat simulations $N = 5000$ times. We take $\alpha = 0.05$. We take the Gaussian kernel to build estimators.

Remark 5: Although there is no difference asymptotically, it is beneficial to apply under-smoothing bandwidth on QPI for small samples.

Remark 6: Once we use the under-smoothing technique to cover the estimation variability for the mean function, we can apply $g = h$ with optimal rate to estimate the variance function.

Simulation results:

Model:	$X_t = \sin(X_{t-1}) + \epsilon_t \sqrt{0.5 + 0.25X_{t-1}^2}, \epsilon_t \sim N(0,1)$									
	CVR for each step					LEN for each step				
$T = 200$	1	2	3	4	5	1	2	3	4	5
QPI-f	0.913	0.918	0.916	0.924	0.924	3.30	3.93	4.07	4.11	4.12
QPI-p	0.935	0.936	0.933	0.941	0.940	3.62	4.29	4.46	4.49	4.51
QPI-f-u	0.904	0.934	0.935	0.943	0.944	3.34	4.25	4.50	4.55	4.57
QPI-p-u	0.926	0.949	0.951	0.958	0.955	3.65	4.62	4.89	4.95	4.97
L_2-PPI-f-opv	0.909	0.938	0.937	0.948	0.946	3.51	4.38	4.60	4.65	4.67
L_2-PPI-p-opv	0.932	0.952	0.951	0.961	0.959	3.87	4.80	5.03	5.08	5.10
L_1-PPI-f-opv	0.912	0.939	0.937	0.949	0.946	3.53	4.38	4.59	4.64	4.66
L_1-PPI-p-opv	0.933	0.951	0.950	0.960	0.960	3.88	4.79	5.02	5.07	5.08
SPI	0.948	0.948	0.940	0.950	0.946	3.37	4.11	4.32	4.38	4.40
$T = 100$										
QPI-f	0.901	0.907	0.912	0.909	0.906	3.28	3.85	3.97	4.01	4.01
QPI-p	0.933	0.931	0.938	0.933	0.938	3.82	4.41	4.55	4.58	4.59
QPI-f-u	0.901	0.923	0.931	0.929	0.932	3.28	4.07	4.29	4.35	4.37
QPI-p-u	0.931	0.943	0.950	0.950	0.947	3.82	4.64	4.85	4.90	4.93
L_2-PPI-f-opv	0.915	0.925	0.935	0.936	0.935	3.52	4.25	4.43	4.48	4.50
L_2-PPI-p-opv	0.941	0.948	0.954	0.955	0.954	4.17	4.90	5.07	5.11	5.13
L_1-PPI-f-opv	0.916	0.926	0.935	0.936	0.936	3.53	4.25	4.43	4.48	4.50
L_1-PPI-p-opv	0.941	0.947	0.954	0.952	0.955	4.17	4.90	5.07	5.12	5.13
SPI	0.951	0.947	0.947	0.946	0.942	3.41	4.13	4.33	4.39	4.40
$T = 50$										
QPI-f	0.844	0.874	0.884	0.883	0.888	3.09	3.68	3.83	3.87	3.89
QPI-p	0.903	0.921	0.929	0.929	0.934	4.01	4.74	4.85	4.93	4.95
QPI-f-u	0.845	0.892	0.907	0.910	0.910	3.09	3.93	4.15	4.23	4.26
QPI-p-u	0.905	0.929	0.934	0.940	0.946	4.03	4.91	5.17	5.23	5.24
L_2-PPI-f-opv	0.871	0.905	0.917	0.918	0.922	3.45	4.19	4.38	4.46	4.47
L_2-PPI-p-opv	0.934	0.941	0.948	0.950	0.954	4.71	5.48	5.60	5.67	5.68
L_1-PPI-f-opv	0.873	0.907	0.920	0.919	0.923	3.46	4.20	4.40	4.47	4.48
L_1-PPI-p-opv	0.934	0.942	0.948	0.950	0.954	4.69	5.44	5.57	5.64	5.64
SPI	0.942	0.946	0.948	0.939	0.950	3.39	4.11	4.33	4.38	4.40

Note: All PPIs with "-opv" symbol are based on applying under-smoothing and optimal bandwidths to estimate mean and variance functions, respectively; All QPIs with "-u" symbol are based on applying under-smoothing estimators.

Future work

Sub-sampling or Bootstrap with DNN

Recently, Deep Neural Network (DNN) with non-smooth activation functions (e.g., ReLU) have been getting more and more attention. Compared to the Shallow Neural Network (SNN) with smooth activation functions which was popular in the last century, this type of DNN has a better empirical performance.

Related literature:

- The forward-bootstrap-type method with SNN on predicting time series; see Giordano et al. (2007) for references. (Heavy computational cost with DNN and a large sample size due to Step 3 in the main algorithm).
- Bootstrap pairs/residuals with NN in the prediction of regression context; see Khosravi et al. (2011) for a review. (Still heavy computational for a large sample size even due to choosing a single random subsample).

Scalable subsampling:

Politis (2021) proposed the idea of choosing non-random subsamples to do estimations. This scalable subagging estimator is more computationally efficient and can be tuned to have the same (or better) rate of convergence compared to a naive estimator with a whole sample.

Error bound for DNN estimations:

Farrell et al. (2021) recently showed that the optimal convergence rate of non-parametric estimation can be achieved by a specific designed deep and wide DNN with some $log(T)$ terms for a class of smooth functions. However, it may not be feasible to take such DNN in practice.

Proposal:

Applying the scalable subsampling technique, we attempt to explore the possibility of killing two birds with one stone: (1) Improve the convergence rate for popular fully-connected DNN; (2) Make prediction inference after building a subagging estimator.

Model-free prediction with normalizing flows

Start with a regression problem, if we observe sample of pairs $\{Y_i, \boldsymbol{X}_i\}_{i=1}^n$, $Y_i \in \mathbb{R}$ and $\boldsymbol{X}_i \in \mathbb{R}^d$. We want to make prediction inference once we observe $\boldsymbol{X}_f$. Here, we hope to take a model-free (non-parametric) approach to describe the conditional distribution $P_{Y|\boldsymbol{X}}$.

Related work in a non-parametric approach:

- Wang and Politis (2021) applied the smooth conditional distribution kernel estimator to approximate the conditional CDF $\widehat{F}(y|\boldsymbol{x}) : \mathbb{P}\{Y \leq y|\boldsymbol{X} = \boldsymbol{x}\}$. Then $\widehat{F}_{Y|\boldsymbol{X}}(y)$ and $\widehat{F}^{-1}_{Y|\boldsymbol{X}}(u)$ represent *normalizing* and *generative* approaches, respectively.
- Zhou et al. (2022) applied a deep generative approach. They define a DNN generator $G(\boldsymbol{Z}, \boldsymbol{X}) : \mathbb{R}^m \times \mathbb{R}^d \to \mathbb{R}$ to estimate $Y|\boldsymbol{X}$, where $\boldsymbol{Z} \sim N(0, I_m)$. They minimize the KL divergence $\mathbb{D}_{KL}\left(p_{G(\boldsymbol{Z},\boldsymbol{X}),\boldsymbol{X}} \parallel p_{Y,\boldsymbol{X}}\right)$ in an adversarial training procedure to get the generator estimator.

Normalizing flow:

The normalizing flow is one kind of generative model, but it is easily invertible due to the specifically structure, e.g. coupling/auto-regressive flows. In short, the normalizing flow is a function $f : \mathbb{R}^d \to \mathbb{R}^d$. If $\boldsymbol{Z} \in \mathbb{R}^d$ has a simple known pdf $p_{\boldsymbol{Z}}$, we can get the pdf of $\boldsymbol{Y} = f(\boldsymbol{Z})$:

$$p_{\boldsymbol{Y}}(\boldsymbol{y}) = p_{\boldsymbol{Z}}(\boldsymbol{z})|\det J_g(\boldsymbol{y})|, \tag{19}$$

where $g(\cdot)$ is the inverse of $f(\cdot)$ and $J_g(\boldsymbol{y})$ is the Jacobian matrix of $g(\cdot)$ by the change of variable formula.

Proposal:

Observing that the normalizing flow coincides with the Model-free prediction idea to some extent, instead of taking the one-way deep generative approach, we attempt to explore a conditional normalizing flow to serve regression prediction purpose, so that we may also be able to keep the pertinence property.

Thank you!

Reference

Chen, J. and Politis, D. N. (2019). Optimal Multi-step-ahead Prediction of ARCH/GARCH Models and NoVaS Transformation. *Econometrics*, 7(3):1–23.

Farrell, M. H., Liang, T., and Misra, S. (2021). Deep neural networks for estimation and inference. *Econometrica*, 89(1):181–213.

Franke, J., Kreiss, J.-P., and Mammen, E. (2002). Bootstrap of kernel smoothing in nonlinear time series. *Bernoulli*, 8(1):1–37.

Franke, J., Neumann, M. H., and Stockis, J.-P. (2004). Bootstrapping nonparametric estimators of the volatility function. *Journal of Econometrics*, 118(1-2):189–218.

Giordano, F., La Rocca, M., and Perna, C. (2007). Forecasting nonlinear time series with neural network sieve bootstrap. *Computational Statistics & Data Analysis*, 51(8):3871–3884.

Guo, M., Bai, Z., and An, H. Z. (1999). Multi-step prediction for nonlinear autoregressive models based on empirical distributions. *Statistica Sinica*, pages 559–570.

Khosravi, A., Nahavandi, S., Creighton, D., and Atiya, A. F. (2011). Comprehensive review of neural network-based prediction intervals and new advances. *IEEE Transactions on neural networks*, 22(9):1341–1356.

Lee, K. and Billings, S. (2003). A new direct approach of computing multi-step ahead predictions for non-linear models. *International Journal of Control*, 76(8):810–822.

Pan, L. and Politis, D. N. (2016). Bootstrap prediction intervals for linear, nonlinear and nonparametric autoregressions. *Journal of Statistical Planning and Inference*, 177:1–27.

Pascual, L., Romo, J., and Ruiz, E. (2004). Bootstrap predictive inference for arima processes. *Journal of Time Series Analysis*, 25(4):449–465.

Reference

Pascual, L., Romo, J., and Ruiz, E. (2006). Bootstrap prediction for returns and volatilities in garch models. *Computational Statistics & Data Analysis*, 50(9):2293–2312.

Pemberton, J. (1987). Exact least squares multi-step prediction from nonlinear autoregressive models. *Journal of Time Series Analysis*, 8(4):443–448.

Politis, D. N. (2003). A normalizing and variance-stabilizing transformation for financial time series. In *Recent Advances and Trends in Nonparametric Statistics*, pages 335–347. Elsevier Inc.

Politis, D. N. (2015). *Model-free prediction in regression*. Springer.

Politis, D. N. (2021). Scalable subsampling: computation, aggregation and inference. *arXiv preprint arXiv:2112.06434*.

Politis, D. N. (2022). Studentization vs. variance stabilization: a simple way out of an old dilemma.

Thombs, L. A. and Schucany, W. R. (1990). Bootstrap prediction intervals for autoregression. *Journal of the American Statistical Association*, 85(410):486–492.

Wang, Y. and Politis, D. N. (2021). Model-free bootstrap and conformal prediction in regression: Conditionality, conjecture testing, and pertinent prediction intervals. *arXiv preprint arXiv:2109.12156*.

Wang, Y. and Politis, D. N. (2022). Model-free bootstrap for a general class of stationary time series. *Bernoulli*, 28(2):744–770.

Xu, C. and Xie, Y. (2021). Conformal prediction interval for dynamic time-series. In *International Conference on Machine Learning*, pages 11559–11569. PMLR.

Zhou, X., Jiao, Y., Liu, J., and Huang, J. (2022). A deep generative approach to conditional sampling. *Journal of the American Statistical Association*, pages 1–12.