arXiv: arXiv:0000.0000

Bootstrap prediction intervals for linear, nonlinear and nonparametric autoregressions

Li Pan and Dimitris N. Politis*

Li Pan
Department of Mathematics
University of California–San Diego
La Jolla, CA 92093-0112, USA
e-mail: lipan@ucsd.edu

Dimitris N. Politis
Department of Mathematics
University of California–San Diego
La Jolla, CA 92093-0112, USA
e-mail: dpolitis@ucsd.edu

Abstract: In order to construct prediction intervals without the cumbersome—and typically unjustifiable—assumption of Gaussianity, some form of resampling is necessary. The regression set-up has been well-studied in the literature but time series prediction faces additional difficulties. The paper at hand focuses on time series that can be modeled as linear, nonlinear or nonparametric autoregressions, and develops a coherent methodology for the construction of bootstrap prediction intervals. Forward and backward bootstrap methods using predictive and fitted residuals are introduced and compared. We present detailed algorithms for these different models and show that the bootstrap intervals manage to capture both sources of variability, namely the innovation error as well as estimation error. In simulations, we compare the prediction intervals associated with different methods in terms of their achieved coverage level and length of interval.

Keywords and phrases: Confidence intervals, forecasting, time series..

1. Introduction

Statistical inference is not considered complete if it is not accompanied by a measure of its inherent accuracy. With point estimators, the accuracy is measured either by a standard error or a confidence interval. With (point) predictors, the accuracy is measured either by the predictor error variance or by a *prediction interval*.

In the setting of an i.i.d. (independent and identically distributed) sample, the problem of prediction is not interesting. However, when the i.i.d. assumption no longer holds, the prediction problem is both important and intriguing; see Geisser (1993)[20] for an introduction. Typical situations where the i.i.d. assumption breaks down include regression and time series.

The literature on predictive intervals in regression is not large; see e.g. Caroll and Ruppert (1991) [12], Patel (1989) [32], Schmoyer (1992)[36] and the references therein. Note that to avoid the cumbersome (and typically unjustifiable) assumption of Gaussianity, some form of resampling is necessary. The residual-based bootstrap in regression is able to capture the predictor variability due to errors in model estimation. Nevertheless, bootstrap prediction intervals in regression are often characterized by finite-sample *undercoverage*. As a remedy, Stine (1985)[37] suggested resampling the studentized residuals but this modification does not fully correct the problem; see the discussion

*Research partialy supported by NSF grants DMS 13-08319 and DMS 12-23137.

in Olive (2007)[30]. Politis(2013)[33] recently proposed the use of *predictive* (as opposed to fitted) residuals to be used in resampling which greatly alleviates the finite-sample undercoverage.

Autoregressive (AR) time series models, be it linear, nonlinear, or nonparametric, have a formal resemblance to the analogous regression models. Indeed, AR models can typically be successfully fitted by the same methods used to estimate a regression, e.g., ordinary Least Square (LS) regression methods for parametric models, and scatterplot smoothing for nonparametric ones. The practitioner has only to be careful regarding the standard errors of the regression estimates but the model-based, i.e., residual-based, bootstrap should in principle be able to capture those.

Therefore, it is not surprising that model-based resampling for regression can be extended to model-based resampling for *auto*-regression. Indeed, standard errors and confidence intervals based on resampling the residuals from a fitted AR model has been one of the first bootstrap approaches for time series; cf. Freedman (1984) [19], Efron and Tibshirani (1986) [15], and Bose (1988) [6].

However, the situation as regards prediction intervals is not as clear; for example, the conditional nature of the predictive inference in time series poses a difficulty. There are several papers on prediction intervals for linear AR models but the literature seems scattered and there are many open questions: (a) how to implement the model-based bootstrap for prediction, i.e., how to generate bootstrap series; (b) how to construct prediction intervals given the availability of many bootstrap series already generated; and lastly (c) how to evaluate asymptotic validity of a prediction interval. In addition, little seems to be known regarding prediction intervals for nonlinear and nonparametric autoregressions.

In the paper at hand we attempt to give answers to the above, and provide a comprehensive approach towards bootstrap prediction intervals for linear, nonlinear, or nonparametric autoregressions. The models we will consider are of the general form:

- **AR model with homoscedastic errors**

$$X_t = m(X_{t-1}, ..., X_{t-p}) + \epsilon_t \tag{1.1}$$

- **AR model with heteroscedastic errors**

$$X_t = m(X_{t-1}, ..., X_{t-p}) + \sigma(X_{t-1}, ..., X_{t-p})\epsilon_t. \tag{1.2}$$

In the above, $m(\cdot)$ and $\sigma(\cdot)$ are unknown; if they can be are assumed to belong to a finite-dimensional, parametric family of functions, then the above describe a linear or nonlinear AR model. If $m(\cdot)$ and $\sigma(\cdot)$ are only assumed to belong to a smoothness class, then the above models describe a nonparametric autoregression. Regarding the errors, the following assumption is made:

$$\epsilon_1, \epsilon_2, \cdots \text{ are i.i.d. } (0, \sigma^2), \text{ and such that } \epsilon_t \text{ is independent from } \{X_s, s < t\} \text{ for all } t; \tag{1.3}$$

in conjuction with model (1.2), we must further assume that $\sigma^2 = 1$ for identifiability. Note, that under either model (1.1) or (1.2), the *causality* assumption (1.3) ensures that $E(X_t|\{X_s, s < t\}) = m(X_{t-1}, ..., X_{t-p})$ gives the optimal predictor of X_t given $\{X_s, s < t\}$; here optimality is with respect to Mean Squared Error (MSE) of prediction.

Section 2 describes the foundations of our approach. Pseudo-series can be generated by either a forward or backward bootstrap, using either fitted or predictive residuals—see Section 2.1 for a discussion. Predictive roots are defined in Section 2.2 while Sections 2.3 and 2.4 discuss notions of asymptotic validity. Section 3 goes in depth as regards bootstrap prediction intervals for linear AR models. Section 4 addresses the nonlinear case using two popular nonlinear models as concrete examples. Finally, Section 5 introduces bootstrap prediction intervals for nonparametric autoregressions. Appendix A contains some technical proofs. A short conclusions section recapitulates the main findings making the point that the forward bootstrap with fitted or predictive residuals serves as the unifying principle across all types of AR models, linear, nonlinear or nonparametric.

2. Bootstrap prediction intervals: laying the foundation

2.1. Forward and backward bootstrap for prediction

As previously mentioned, an autoregression can be formally viewed as regression. However, in prediction with an AR(p) model, linear or nonlinear, an additional difficulty is that the one-step-ahead prediction is done conditionally on the last p observed values that are themselves random.

To fix ideas, suppose $X_1, \cdots, X_n$ are data from the linear AR(1) model: $X_t = \phi_1 X_{t-1} + \epsilon_t$ where $|\phi_1| < 1$ and the ϵ_t are i.i.d. with mean zero. Given the data, the MSE–optimal predictor of X_{n+1} given the data is $\phi_1 X_n$ which is approximated in practice by plugging-in an estimator, say $\hat{\phi}_1$, for ϕ_1. Generating bootstrap series $X_1^*, X_2^*, \cdots$ from the fitted AR model enables us to capture the variability of $\hat{\phi}_1$ when the latter is re-estimated from bootstrap datasets such as $X_1^*, \cdots, X_n^*$.

For the application to prediction intervals, note that the bootstrap also allows us to generate X_{n+1}^* so that the statistical accuracy of the predictor $\hat{\phi}_1 X_n$ can be gauged. However, none of these bootstrap series will have their last value X_n^* *exactly* equal to the original value X_n as needed for prediction purposes. Herein lies the problem, since the behavior of the predictor $\hat{\phi}_1 X_n$ needs to be captured *conditionally* on the original value X_n.

To avoid this difficulty, Thombs and Schucany(1990)[39] proposed to generate the bootstrap data $X_1^*, \cdots, X_n^*$ going backwards from the last value that is fixed at $X_n^* = X_n$. This is the *backward bootstrap* method that was revisited by Breidt, Davis and Dunsmuir(1995)[10] who gave the correct algorithm of finding the backward errors. Note that the generation of X_{n+1}^* is still done in a forward fashion using the fitted AR model conditionally on the value X_n.

Nevertheless, the natural way autoregressions evolve is *forward* in time, i.e., given X_{t-1}, the next observation is generated as $X_t = \phi_1 X_{t-1} + \epsilon_t$, and so on. Thus, it is intuitive to construct bootstrap procedures that run forward in time, i.e., given X_{t-1}^*, the next bootstrap observation is generated as

$$X_t^* = \hat{\phi}_1 X_{t-1}^* + \epsilon_t^*, \tag{2.1}$$

and so on. Indeed, most (if not all) of the literature on bootstrap confidence intervals for AR models uses the natural time order to generate bootstrap series. It would be nice to be able to build upon this large body of work in order to construct prediction intervals. However, recall that predictive inference is to be conducted conditionally on the last value X_n in order to be able to place prediction bounds around the point predictor $\hat{\phi}_1 X_n$. So how can one ensure that $X_n^* = X_n$ so that $X_{n+1}^* = \hat{\phi}_1 X_n + \epsilon_{n+1}^*$?

Aided by the additive structure of the AR model, it is possible to "have our cake and eat it too", i.e., generate bootstrap series forward in time but also ensure that X_{n+1}^* is constructed correctly. This procedure will be called the *forward bootstrap* method for prediction intervals, and comprises of two steps:

A. Choose a starting value X_0^* appropriately, e.g., choose it at random from one of the original data $X_1, \cdots, X_n$. Then, use recursion (2.1) for $t = 1, 2, \ldots, n$ in order to generate bootstrap data $X_1^*, \cdots, X_n^*$. Re-compute the statistic of interest (in this case $\hat{\phi}_1$) from the bootstrap data $X_1^*, \cdots, X_n^*$ to obtain the bootstrap statistic $\hat{\phi}_1^*$.
B. Re-define the last value in the bootstrap world, i.e., let $X_n^* = X_n$. Compute the one-step ahead bootstrap predictor $\hat{X}_{n+1}^* = \hat{\phi}_1^* X_n$, and also generate the future bootstrap observation $X_{n+1}^* = \hat{\phi}_1 X_n + \epsilon_{n+1}^*$.

The above algorithm works because the two constituents of the prediction error $X_{n+1} - \hat{X}_{n+1} = (\phi_1 X_n - \hat{\phi}_1 X_n) + \epsilon_{n+1}$, i.e., estimation error $(\phi_1 X_n - \hat{\phi}_1 X_n)$ and innovation error ϵ_{n+1} are independent, and the same is true in the bootstrap world.

As stated above, the algorithm is specific to an AR(1) model but its extension to higher-order models is straightforward and will be given in the sequel. Indeed, the forward bootstrap is the method that can be immediately generalized to apply for nonlinear and nonparametric autoregressions as well, thus forming a unifying principle for treating all AR models. The forward bootstrap idea has been previously used for prediction intervals in linear AR models by Masarotto(1990)[26] and Pascual *et al.* (2004)[31] but with some important differences; for example, Masarotto(1990)[26] omits the important step B above—see Section 3.8 for a discussion.

Remark 2.1. Both aforementioned bootstrap ideas, backward and forward, hinge on an i.i.d. resampling of the residuals obtained from the fitted model. In the AR(1) case, the *fitted* residuals are obtained as $\hat{\epsilon}_t = X_t - \hat{\phi}_1 X_{t-1}$ for $t = 2, 3, \cdots, n$. Nevertheless, Politis(2013)[33] made a strong case that resampling the *predictive* residuals gives more accurate prediction intervals in regression, be it linear or nonparametric. Section 3 defines a particular notion of predictive residuals in autoregression, and shows their potential benefit in constructing bootstrap prediction intervals.

2.2. Predictive roots and h-step ahead optimal prediction

Given the ability to generate bootstrap datasets using a valid resampling procedure, the question arises as to how to actually construct the prediction interval. Notably, in the related problem of confidence interval construction there are two main approaches: (a) the *percentile* approach along with the associated bias correction and acceleration expounded upon in Efron and Tibshirani (1994) [16]; and (b) the approach based on *pivots* and *roots* as in Bickel and Freedman (1981) [3], Beran (1984) [2], Hall (1992) [22], and Politis, Romano and Wolf (1999) [34]. Both approaches are popular although the latter is more conducive for theoretical analysis.

Politis(2013)[33] gave the definition of *'predictive roots'* to be used in order to construct prediction intervals in regression. We will extend this idea to autoregression. Let $X_1, \cdots, X_n$ be an observed stretch of a time series that follows a stationary autoregressive model with order p, i.e., model (1.1) or (1.2); the autoregression can be linear, nonlinear or nonparametric. The objective is a prediction interval for the h-step ahead value X_{n+h} for some integer $h \geq 1$; the one-step ahead case is, of course, the most basic.

Denote by $\hat{X}_{n+h}$ the point predictor of X_{n+h} based on the data $X_1, \cdots, X_n$; since $\hat{X}_{n+h}$ is a function of the data, we can write $\hat{X}_{n+h} = \Pi(X_1, \cdots, X_n)$. Let $\hat{V}_n^2$ be an estimate of $Var(X_{n+h} - \hat{X}_{n+h}|X_1, \cdots, X_n)$ which is the conditional variance in h-step ahead prediction; since $\hat{V}_n$ is a function of the data, we denote $\hat{V}_n = V(X_1, \cdots, X_n)$.

Definition 2.1. ***Predictive root and studentized predictive root****. The h-step ahead predictive root is defined as $X_{n+h} - \hat{X}_{n+h}$, i.e., it is the error in the h-step ahead prediction. The studentized predictive root is $\frac{X_{n+h} - \hat{X}_{n+h}}{\hat{V}_n}$.*

Given a bootstrap pseudo series $X_1^*, \cdots, X_n^*$, analogs of the aforementioned quantities can be defined, i.e., $\hat{X}_{n+h}^* = \Pi(X_1^*, \cdots, X_n^*)$ and $\hat{V}_n^* = V(X_1^*, \cdots, X_n^*)$.We can then similarly define the bootstrap predictive roots.

Definition 2.2. ***Bootstrap predictive root and studentized bootstrap predictive root****. The bootstrap predictive root is defined as $X_{n+h}^* - \hat{X}_{n+h}^*$. The studentized bootstrap predictive root is $\frac{X_{n+h}^* - \hat{X}_{n+h}^*}{\hat{V}_n^*}$.*

2.3. Prediction intervals and asymptotic validity

Given the data $X_1, \cdots, X_n$, our goal is to construct a prediction interval that will contain the future value X_{n+h} with a prespecified coverage probability. With an AR(p) model, linear or nonlinear, the predictor will be a function of the last p data points, i.e., $X_{n-p+1}, \cdots, X_n$. Hence the prediction interval's coverage probability should be interpreted as *conditional probability* given $X_{n-p+1}, \cdots, X_n$.

Definition 2.3. ***Asymptotic validity of prediction intervals.***
Let L_n, U_n be functions of the data $X_1, \cdots, X_n$. The interval $[L_n, U_n]$ will be called a $(1-\alpha)100\%$ asymptotically valid *prediction interval for X_{n+h} given $X_{n-p+1}, \cdots, X_n$ if*

$$P(L_n \le X_{n+h} \le U_n) \to 1-\alpha \text{ as } n \to \infty \tag{2.2}$$

for all $(X_{n-p+1}, \cdots, X_n)$ in a set that has (unconditional) probability equal to one.

The probability P in (2.2) should be interpreted as *conditional* probability given $X_{n-p+1}, \cdots, X_n$ although it is not explicitly denoted; hence, Definition 2.3 indicates *conditional* validity of the prediction interval $[L_n, U_n]$.

The salient point in all bootstrap algorithms that will be discussed is to use the bootstrap distribution of the (potentially studentized) bootstrap predictive root to estimate the true distribution of the (potentially studentized) predictive root. Bootstrap probabilities and expectations are usually denoted by P^* and E^*, and they are understood to be conditional on the original data $X_1 = x_1, \cdots, X_n = x_n$. Since Definition 2.3 involves conditional validity, we will understand that P^* and E^* are also conditional on $X^*_{n-p+1} = x_{n-p+1}, \cdots, X^*_n = x_n$ when they are applied to 'future' events in the bootstrap world, i.e., events determined by $\{X^*_s \text{ for } s > n\}$; this is not restrictive since we will ensure that our bootstrap algorithms satisfy this requirement. For instance, both P and P^* in Remark 2.2 below represent probabilities conditional on $X_{n-p+1} = x_{n-p+1}, \cdots, X_n = x_n$ and $X^*_{n-p+1} = x_{n-p+1}, \cdots, X^*_n = x_n$ respectively.

Remark 2.2. Suppose the (conditional) probability $P(X_{n+h} - \hat{X}_{n+h} \le a)$ is a continuous function of a in the limit as $n \to \infty$. If one can show that

$$\sup_a |P(X_{n+h} - \hat{X}_{n+h} \le a) - P^*(X^*_{n+h} - \hat{X}^*_{n+h} \le a)| \xrightarrow{P} 0,$$

then standard results imply that the quantiles of $P^*(X^*_{n+h} - \hat{X}^*_{n+h} \le a)$ can be used to consistently estimate the quantiles of $P(X_{n+h} - \hat{X}_{n+h} \le a)$, thus leading to asymptotically valid prediction intervals. Similarly, if one wants to construct asymptotically valid bootstrap prediction intervals based on studentized predictive roots, it suffices to show that

$$\sup_a |P(\frac{X_{n+h} - \hat{X}_{n+h}}{\hat{V}_n} \le a) - P^*(\frac{X^*_{n+h} - \hat{X}^*_{n+h}}{\hat{V}^*_n} \le a)| \xrightarrow{P} 0.$$

2.4. Asymptotic pertinence of bootstrap prediction intervals

Asymptotic validity is a fundamental property but it does not tell the whole story. Prediction intervals are particularly useful if they can also capture the uncertainty involved in model estimation although the latter is asymptotically negligible.

To give a concrete example, consider the simple case where $X_1, X_2, \cdots$ are i.i.d. $N(\mu, \sigma^2)$; this is a special case of an AR model with no dependence present. Given the data $X_1, \cdots, X_n$, we estimate

the unknown μ, σ^2 by the sample mean and variance $\hat{\mu}, \hat{\sigma}^2$ respectively. Then, the exact Normal theory $(1-\alpha)100\%$ prediction interval for X_{n+h} is given by

$$\hat{\mu} \pm t_{n-1}(\alpha/2)\hat{\sigma}\sqrt{1+n^{-1}}. \tag{2.3}$$

One could use the standard normal quantile $z(\alpha/2)$ instead of $t_{n-1}(\alpha/2)$, i.e., construct the prediction interval:

$$\hat{\mu} \pm z(\alpha/2)\hat{\sigma}\sqrt{1+n^{-1}}. \tag{2.4}$$

Since $1+n^{-1} \approx 1$ for large n, an even simpler prediction interval is available:

$$\hat{\mu} \pm z(\alpha/2)\hat{\sigma}. \tag{2.5}$$

Notably, all three above prediction intervals are asymptotically valid in the sense of Definition 2.3. Nevertheless, as discussed in Politis(2013)[33], interval (2.5) can be called *naive* since it fails to take into account the variability that results from the error in estimating the theoretical predictor μ by $\hat{\mu}$. The result is that, although asymptotically valid, interval (2.5) will be characterized by *under-coverage* in finite samples; see Geisser (1993) for an in-depth discussion.

By contrast, interval (2.4) does take into account the variability resulting from estimating the theoretical predictor. Therefore, interval (2.4) deserves to be called something stronger than asymptotically valid; we will call it *pertinent* to indicate that it asymptotically captures *all* three elements of the exact interval (2.3), namely:
(i) the quantile $t_{n-1}(\alpha/2)$ associated with the studentized root;
(ii) the error variance σ^2; and
(iii) the variability associated with the estimated parameters, i.e., the factor $\sqrt{1+n^{-1}}$.

In general, an exact interval analogous to (2.3) will not be available because of non-normality of the errors and/or nonlinearity of the optimal predictor. A 'pertinent' interval such as (2.4) would be something to strive for. Notably, the bootstrap is an attempt to create prediction intervals that are asymptotically pertinent in that (a) they are able to capture the variability due to the estimated quantities—note that in AR(p) models the correction term inside the square root of (2.3) would be $O(p/n)$ not just $1/n$, and in nonparametric AR models it would be $O(\frac{1}{hn})$ with $h \to 0$ as $n \to \infty$, i.e., this correction is not so trivial; and (b) they are able to approximate well the necessary quantiles.

Interestingly, while interval (2.3) is based on the distribution of the studentized predictive root, the bootstrap can also work with nonstudentized roots; in this case, the bootstrap would attempt to estimate the product $t_{n-1}(\alpha/2)\ \hat{\sigma}$ as a whole instead of breaking it up in its two constituent pieces. Nevertheless, it may be the case that the studentized bootstrap may lead to better approximations, and therefore more accurate prediction intervals, although the phenomenon is not as clear-cut as in the case of bootstrap confidence intervals. Finally, note that bootstrap prediction intervals are not restricted to be symmetric around the predictor like (2.3); thus, they may also capture the skewness of the predictive distribution which is valuable in its own right.

To formally define the notion of pertinence, consider the homoscedastic model (1.1), and recall that eq. (1.3) implies that the MSE–optimal predictor of X_{n+1} given $X_1 = x_1, \ldots, X_n = x_n$ is $m(x_n, ..., x_{n-p+1})$. Hence we set $\hat{X}_{n+1} = \hat{m}(x_n, \ldots, x_{n-p+1})$ where $\hat{m}(\cdot)$ is a consistent estimator of $m(\cdot)$; without loss of generality, assume that $\hat{m}(\cdot)$ has rate of convergence a_n, i.e., $a_n(\hat{m}(\cdot) - m(\cdot))$ has a well-defined, non-trivial asymptotic distribution where $a_n \to \infty$ as $n \to \infty$. Then, the predictive root can be written as

$$X_{n+1} - \hat{X}_{n+1} = \epsilon_{n+1} + A_m \tag{2.6}$$

where $A_m = m(x_n, \ldots, x_{n-p+1}) - \hat{m}(x_n, \ldots, x_{n-p+1}) = O_p(1/a_n)$ represents the estimation error.

Similarly, the bootstrap predictive root can be written as

$$X^*_{n+1} - \hat{X}^*_{n+1} = \epsilon^*_{n+1} + A^*_m \tag{2.7}$$

where $A^*_m = \hat{m}(x_n, \ldots, x_{n-p+1}) - \hat{m}^*(x_n, \ldots, x_{n-p+1})$. By construction, the model-based bootstrap should, in principle, be capable of asymptotically capturing both the pure prediction error, i.e., the distribution of ϵ_{n+1}, as well as the estimation error. We are then led to the following definition.

Definition 2.4. ***Asymptotic pertinence of bootstrap prediction intervals under model*** (1.1)**.** *Consider a bootstrap prediction interval for X_{n+1} that is based on approximating the distribution of the predictive root $X_{n+1} - \hat{X}_{n+1}$ of eq.* (2.6) *by the distribution of the bootstrap predictive root $X^*_{n+1} - \hat{X}^*_{n+1}$ of eq.* (2.7)*. The interval will be called* asymptotically pertinent *provided the bootstrap satisfies the following three conditions as $n \to \infty$ conditionally on $X_{n-p+1} = x_{n-p+1}, \cdots, X_n = x_n$.*
(i) $\sup_a |P(\epsilon_{n+1} \le a) - P^*(\epsilon^*_{n+1} \le a)| \xrightarrow{P} 0$*, presupposing that the error distribution is continuous.*
(ii) $|P(a_n A_m \le a) - P^*(a_n A^*_m \le a)| \xrightarrow{P} 0$ *for some sequence $a_n \to \infty$, and for all points a where the assumed nontrivial limit of $P(a_n A_m \le a)$ is continuous.*
(iii) ϵ^*_{n+1} and A^*_m are independent in the bootstrap world—as their analogs are in the real world due to the causality assumption (1.3)*.*
Furthermore, the bootstrap prediction interval for X_{n+1} that is based on the approximating the distribution of the studentized *predictive root $(X_{n+1} - \hat{X}_{n+1})/\hat{V}_n$ by the distribution of the bootstrap studentized predictive root $(X^*_{n+1} - \hat{X}^*_{n+1})/\hat{V}^*_n$ will be called asymptotically pertinent if, in addition to (i)—(iii) above, the following also holds:*
(iv) $\hat{V}_n/\hat{V}^*_n \xrightarrow{P} 0$.

For concreteness, the above focuses on one-step ahead prediction but analogous definitions can be constructed for h-step ahead prediction intervals using studentized or unstudentized predictive roots.

Remark 2.3. Note that asymptotic pertinence is a stronger property than asymptotic validity. In fact, under model (1.1), just part (i) of Definition 2.4 together with the consistency of $\hat{m}(\cdot)$ and $\hat{m}^*(\cdot)$, i.e., the fact that both A_m and A^*_m are $o_p(1)$ due to $a_n \to \infty$, are enough to imply asymptotic validity of the bootstrap prediction interval. Also note that part (ii) of Definition 2.4 is the condition needed in order to show that the bootstrap can yield asymptotically valid *confidence intervals* for the conditional mean $m(\cdot)$. In many cases in the literature, this condition has been already established; we can build upon this for the purpose of constructing pertinent prediction intervals.

Consider now the heteroscedastic model (1.2). Much of the above discussion carries over *verbatim*; for example, our predictor of X_{n+1} given $X_1 = x_1, \ldots, X_n = x_n$ is still $\hat{X}_{n+1} = \hat{m}(x_n, \ldots, x_{n-p+1})$. The only difference is that the predictive root is

$$X_{n+1} - \hat{X}_{n+1} = \sigma(x_n, \ldots, x_{n-p+1})\epsilon_{n+1} + A_m \tag{2.8}$$

and the bootstrap predictive root is

$$X^*_{n+1} - \hat{X}^*_{n+1} = \hat{\sigma}(x_n, \ldots, x_{n-p+1})\epsilon^*_{n+1} + A^*_m \tag{2.9}$$

where $\hat{\sigma}(\cdot)$ is a (consistent) estimator of $\sigma(\cdot)$ that is employed in the bootstrap data generation mechanism. Hence, the following definition is immediate.

Definition 2.5. ***Asymptotic pertinence of bootstrap prediction intervals under model*** (1.2)**.** *Consider a bootstrap prediction interval for X_{n+1} that is based on approximating*

the distribution of the predictive root $X_{n+1} - \hat{X}_{n+1}$ *of eq.* (2.8) *by the distribution of the bootstrap predictive root* $X^*_{n+1} - \hat{X}^*_{n+1}$ *of eq.* (2.9)*. The interval will be called* asymptotically pertinent *provided the bootstrap satisfies conditions (i)—(iii) or Definition 2.4 together with the additional consistency requirement:*

(iv′) $\sigma(x_n, \ldots, x_{n-p+1}) - \hat{\sigma}(x_n, \ldots, x_{n-p+1}) \xrightarrow{P} 0.$

Furthermore, the bootstrap prediction interval for X_{n+1} *that is based on the approximating the distribution of the* studentized *predictive root* $(X_{n+1} - \hat{X}_{n+1})/\hat{V}_n$ *by the distribution of the bootstrap studentized predictive root* $(X^*_{n+1} - \hat{X}^*_{n+1})/\hat{V}^*_n$ *will be called asymptotically pertinent if, in addition condition (iv) or Definition 2.4 also holds.*

Remark 2.4. Taking into account that $A_m = o_p(1)$ as $n \to \infty$, a simple estimator for the (conditional) variance of the predictive root $X_{n+1} - \hat{X}_{n+1}$ under model (1.2) is $\hat{V}_n = \hat{\sigma}(x_n, \ldots, x_{n-p+1})$. Thus, in the case of one-step ahead prediction, condition (iv) or Definition 2.4 can be re-written as $\hat{\sigma}(x_n, \ldots, x_{n-p+1}) - \hat{\sigma}^*(x_n, \ldots, x_{n-p+1}) \xrightarrow{P} 0$, i.e., it is just a bootstrap version of condition (iv′) or Definition 2.5. As a matter of fact, resampling in the heteroscedastic model (1.2) entails using studentized residuals. In this case, the predictive root method becomes tantamount to the studentized predictive root method when the simple estimator $\hat{V}_n = \hat{\sigma}(x_n, \ldots, x_{n-p+1})$ is used; see Section 5.2 for more discussion.

2.5. Prediction interval accuracy: asymptotics vs. finite samples

The notion of 'pertinence' was defined in order to identify a prediction interval that has good finite-sample performance as it imitates all the constituent parts of an 'ideal' interval. Inadvertently, the notion of 'pertinence' was defined via asymptotic requirements. Nevertheless, the asymptotics alone are still not very informative in terms of the interval's finite-sample accuracy, i.e., the attained coverage level.

Going back to the simple interval (2.3), note that we typically have $\hat{\sigma}^2 = \sigma^2 + O_p(1/\sqrt{n})$ while the approximations $z(\alpha/2) \approx t_{n-1}(\alpha/2)$ and $\sqrt{1+n^{-1}} \approx 1$ have a smaller error of order $O(1/n)$. So the determining factor for the asymptotics is the accuracy of the estimator $\hat{\sigma}^2$. However, there are many $\sqrt{n}$–convergent estimators of σ^2 having potentially very different finite-sample behavior. The default estimator is the sample variance of the fitted residuals but this tends to be downwardly biased; the predictive residuals—although asymptotically equivalent to the fitted residuals—lead to improved finite-sample performance as alluded to in Remark 2.1.

Similarly, the quantity (i) of Definition 2.4 is typically of order $O_p(1/\sqrt{n})$. In general, this rate of convergence can not be improved, and dictates the rate of convergence for the coverage probability of eq. (2.2) under the homoscedastic model (1.2). Once again, using the empirical distribution of the predictive—as opposed to the fitted—residuals in order to generate ϵ^*_t makes a huge practical difference in finite samples despite the fact that the two are asymptotically equivalent.

The most challenging situation in practice occurs in the set-up of the heteroscedastic model (1.2) with the conditional variance function $\sigma(\cdot)$ being unknown but assumed smooth. Here the rate of convergence of the coverage probability of eq. (2.2) will be dictated by the nonparametric rate of estimating $\sigma(\cdot)$, i.e., the rate of convergence of the quantity appearing in condition (iv′) of Definition 2.5; see Section 5.2 for more details. This is perhaps the only case where the asymptotics are informative as regards the finite-sample performance of the prediction interval, i.e., the less accurately attained coverage levels in the presence of heteroscedasticity of unknown functional form.

3. Bootstrap Prediction Intervals for Linear Autoregressions

Consider the strictly stationary, causal AR(p) model defined by the recursion

$$X_t = \phi_0 + \sum_{j=1}^{p} \phi_j X_{t-j} + \epsilon_t \tag{3.1}$$

which is a special case of model (1.1) with the ϵ_t being i.i.d. with mean zero, variance σ^2 and distribution F_ϵ. The assumed causality condition (1.3) is now tantamount to $\phi(z) = 1 - \phi_1 z - \cdots - \phi_p z^p \neq 0$ for $|z| \leq 1$. Denote $\underline{\phi} = (\phi_0, \phi_1, \phi_2, \cdots, \phi_p)'$ the vector of autoregressive parameters, and $\hat{\underline{\phi}} = (\hat{\phi}_0, \hat{\phi}_1, \cdots, \hat{\phi}_p)'$ and $\hat{\phi}(z) = 1 - \hat{\phi}_1 z - \cdots - \hat{\phi}_p z^p$ the respective estimates. Let $\hat{X}_t$ be the fitted value of X_t, i.e., $X_t = \hat{\phi}_0 + \sum_{j=1}^{p} \hat{\phi}_j X_{t-j}$. Finally, let $Y_t = (X_t, X_{t-1}, \cdots, X_{t-p+1})'$ be the vector of the last p observations up to X_t.

Stine(1987)[38] used a bootstrap method to estimate the prediction mean squared error of the estimated linear predictor of an AR(p) model with i.i.d. Gaussian errors. Relaxing the assumption of Gaussian errors, Thombs and Schucany(1990)[39] proposed a backward bootstrap method to find prediction intervals for linear autoregressions conditioned on the last p observations; their method was described in Section 2.1. The backward bootstrap method was revisited by Breidt, Davis and Dunsmuir(1995)[10] who gave the correct algorithm of finding the backward errors.

Masarotto(1990)[26] proposed a forward bootstrap method based on the studentized predictive root to obtain prediction intervals for AR(p) models. Notably, his method omits the crucial step B of the Forward bootstrap method defined in Section 2.1. As a result, his intervals are not asymptotically pertinent since the basic premise of Definition 2.4 regarding the construction of the interval is not satisfied; however, his intervals are asymptotically valid because the omitted/distorted term has to do with the estimation error which vanishes asymptotically. Finally, Pascual *et al.* (2004)[31] proposed another forward bootstrap method and applied it to prediction intervals for both autoregressive as well as ARMA models; their intervals are constructed via an analog of the percentile method without considering predictive roots—see Section 3.8 for more discussion.

In the present section, we first give the detailed algorithms for constructing forward bootstrap prediction intervals using fitted and/or predictive residuals, and then prove the consistency of the predictive root method for prediction intervals. We then study the corresponding backward methods. We show how both backward and forward methods can be improved by introducing the predictive residuals. In simulation, we will see that the methods with predictive residuals have improved coverage level compared to the methods with fitted residuals; this result is not unexpected since a similar phenomenon occurs in linear regression—cf. Politis (2013)[33]. In Section 3.8, we review alternative approaches to construct bootstrap prediction intervals, and compare them with ours.

3.1. Forward Bootstrap Algorithm

As described in Section 2.1, the idea of forward bootstrap method is that given observations $X_1 = x_1, \cdots, X_n = x_n$, we can use the fitted AR recursion to generate bootstrap series "forward" in time starting from some initial conditions. This recursion stops when n bootstrap data have been generated; to generate the $(n+1)$th bootstrap point (and beyond), the recursion has to be restarted with different initial values that are fixed to be the last p original observations. The details for estimating the coefficients, generating the bootstrap pseudo-data and constructing the prediction intervals using both fitted and predictive residuals are given below in Sections 3.1.1 and 3.1.2

3.1.1. Forward Bootstrap with Fitted Residuals

Given a sample $\{x_1, \cdots, x_n\}$ from (3.1), the following are the steps needed to construct the prediction interval for future value X_{n+h} based on the predictive root method.

Algorithm 3.1. *Forward bootstrap with fitted residuals (Ff)*

1. *Use all observations $x_1, \cdots, x_n$ to obtain the Least Squares (LS) estimators $\underline{\hat{\phi}} = (\hat{\phi}_0, \hat{\phi}_1, \cdots, \hat{\phi}_p)'$ by fitting the following linear model*

$$\begin{pmatrix} x_n \\ x_{n-1} \\ \vdots \\ x_{p+1} \end{pmatrix} = \begin{bmatrix} 1 & x_{n-1} & \cdots & x_{n-p} \\ 1 & x_{n-2} & \cdots & x_{n-p-1} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_p & \cdots & x_1 \end{bmatrix} \begin{pmatrix} \phi_0 \\ \phi_1 \\ \vdots \\ \phi_p \end{pmatrix} + \begin{pmatrix} \epsilon_n \\ \epsilon_{n-1} \\ \vdots \\ \epsilon_{p+1} \end{pmatrix}. \tag{3.2}$$

2. *For $t = p+1, \cdots, n$, compute the fitted value and fitted residuals:*

$$\hat{x_t} = \hat{\phi}_0 + \sum_{j=1}^{p} \hat{\phi}_j x_{t-j}, \quad \textit{and} \quad \hat{\epsilon}_t = x_t - \hat{x_t}.$$

3. *Center the fitted residuals: let $r_t = \hat{\epsilon}_t - \bar{\hat{\epsilon}}$ for $t = p+1, \cdots, n$, and $\bar{\hat{\epsilon}} = (n-p)^{-1} \sum_{p+1}^{n} \hat{\epsilon}_t$; let the empirical distribution of r_t be denoted by $\hat{F}_n$.*
   (a) *Draw bootstrap pseudo residuals $\{\epsilon_t^*,\ t \geq 1\}$ i.i.d. from $\hat{F}_n$.*
   (b) *To ensure stationarity of the bootstrap series, generate $n+m$ pseudo-data for some large positive m and then discard the first m data. Let $(u_1^*, \cdots, u_p^*)$ be chosen at random from the set of p–tuplets $\{(x_k, \cdots, x_{k+p-1})$ for $k = 1, \cdots, n-p+1\}$; then generate $\{u_t^*, t \geq p+1\}$ by the recursion:*

$$u_t^* = \hat{\phi}_0 + \sum_{j=1}^{p} \hat{\phi}_j u_{t-j}^* + \epsilon_t^*, \ \textit{for } t = p+1, \cdots, n+m.$$

   Then define $x_t^* = u_{m+t}^*$ for $t = 1, 2, \cdots, n$.
   (c) *Based on the pseudo-data $\{x_1^*, \cdots, x_n^*\}$, re-estimate the coefficients $\underline{\phi}$ by the LS estimator $\underline{\hat{\phi}}^* = (\hat{\phi_0}^*, \hat{\phi_1}^*, \cdots, \hat{\phi_p}^*)'$ as in step 1. Then compute the future bootstrap predicted values $\hat{x}_{n+1}^*, \cdots, \hat{x}_{n+h}^*$ by the recursion:*

$$\hat{x}_{n+t}^* = \hat{\phi_0}^* + \sum_{j=1}^{p} \hat{\phi_j}^* \hat{x}_{n+t-j}^* \ \textit{for } t = 1, \cdots, h$$

   where $\hat{x}_{n+t-j}^* = x_{n+t-j}$ when $t \leq j$
   (d) *In order to conduct conditionally valid predictive inference,* re-define *the last p observations to match the original observed values, i.e., let $x_{n-p+1}^* = x_{n-p+1}, \cdots, x_n^* = x_n$. Then, generate the future bootstrap observations $x_{n+1}^*, x_{n+2}^*, \cdots, x_{n+h}^*$ by the recursion:*

$$x_{n+t}^* = \hat{\phi}_0 + \sum_{j=1}^{p} \hat{\phi}_j x_{n+t-j}^* + \epsilon_{n+t}^*, \ \textit{for } t = 1, 2, \cdots, h.$$

(e) *Calculate a bootstrap root replicate as* $x^*_{n+h} - \hat{x}^*_{n+h}$.

4. *Steps (a)-(e) above are repeated* B *times, and the* B *bootstrap replicates are collected in the form of an empirical distribution whose* α*-quantile is denoted* $q(\alpha)$.
5. *Compute the predicted future values* $\hat{x}_{n+1} \cdots, \hat{x}_{n+h}$ *by following recursion:*

$$\hat{x}_{n+t} = \hat{\phi}_0 + \sum_{j=1}^{p} \hat{\phi}_j \hat{x}_{n+t-j} \text{ for } t = 1, \cdots, h$$

where $\hat{x}_{n+t-j} = x_{n+t-j}$ *for* $t \leq j$.
6. *Construct the* $(1-\alpha)100\%$ *equal-tailed prediction interval for* X_{n+h} *as*

$$[\hat{x}_{n+h} + q(\alpha/2), \hat{x}_{n+h} + q(1-\alpha/2)]. \tag{3.3}$$

3.1.2. Forward Bootstrap with Predictive Residuals

Motivated by Politis(2013)[33], we consider using predictive, as opposed to fitted, residuals for the bootstrap. We define the predictive residuals in the AR context as $\hat{\epsilon}_t^{(t)} = x_t - \hat{x}_t^{(t)}$ where $\hat{x}_t^{(t)}$ is computed from the delete-x_t data set, i.e., the available data for the scatterplot of x_k vs. $\{x_{k-p}, \cdots, x_{k-1}\}$ over which the LS fitting takes place excludes the single point that corresponds to $k = t$.

The forward bootstrap procedure with predictive residuals is similar to Algorithm 3.1; the only difference is in step 2.

Algorithm 3.2. *Forward bootstrap with predictive residuals (Fp)*

1 same as step 1 in Algorithm 3.1.

2 Use the delete-x_t *dataset to compute the LS estimator*

$$\underline{\hat{\phi}}^{(t)} = (\hat{\phi_0}^{(t)}, \hat{\phi_1}^{(t)}, \cdots, \hat{\phi_p}^{(t)})'$$

as in step 1, i.e., compute $\underline{\hat{\phi}}^{(t)}$ *by changing regression model (3.2) as follows:* delete the row of x_t in left hand side of (3.2), delete the row $(1, x_{t-1}, \cdots, x_{t-p})$ in the design matrix, delete ϵ_t from the vector of ϵ and change the $\underline{\hat{\phi}}$ to $\underline{\hat{\phi}}^{(t)}$ at the right hand side .
Then, calculate the delete-x_t *fitted values:*

$$\hat{x_t}^{(t)} = \hat{\phi_0}^{(t)} + \sum_{j=1}^{p} \hat{\phi_j}^{(t)} x_{t-j}, \quad \text{for} \quad t = p+1, \cdots, n$$

and the predictive residuals: $\hat{\epsilon}_t^{(t)} = x_t - \hat{x}_t^{(t)}$ *for* $t = p+1, \cdots, n$.

3-6 Change the $\hat{\epsilon}_t$ *into* $\hat{\epsilon}_t^{(t)}$*; the rest is the same as in Algorithm 3.1.*

Remark 3.1. The LS estimator $\underline{\hat{\phi}}$ is asymptotically equivalent to the popular Yule-Walker (YW) estimators for fitting AR models. The advantage of YW estimators is that they almost surely lead to a causal fitted model. By contrast, the LS estimator $\underline{\hat{\phi}}$ is only asymptotically causal but it is completely scatterplot-based, and thus convenient in terms of our notion of predictive residuals. Indeed, for any bootstrap method using fitted residuals (studentized or not), e.g., the forward Algorithm 3.1 or the backward Algorithm 3.5 in the sequel, we could equally employ the Yule-Walker instead of the LS estimators. But for methods using our notion of predictive residuals, it is most convenient to be able to employ the LS estimators. To elaborate, consider the possibility that for our given dataset the LS estimator $\underline{\hat{\phi}}$ turns out to be not causal; we then we have two options:

- Use a bootstrap algorithm with fitted residuals based on the YW estimator $\underline{\tilde{\phi}} = \hat{\Gamma}_p^{-1}\underline{\hat{\gamma}}_p$ where $\hat{\Gamma}_p$ is the $p \times p$ matrix with ij element $\hat{\gamma}(i-j)$, $\underline{\hat{\gamma}}_p$ is a vector with ith element $\hat{\gamma}(i) = n^{-1}\sum_{k=1}^{n-|i|}(X_k - \bar{X})(X_{k+|i|} - \bar{X})$ and $\bar{X} = n^{-1}\sum_{k=1}^{n} X_k$.
- To use bootstrap with predictive residuals, we can define the delete-x_t estimates of the autocovariance as $\hat{\gamma}^{(t)}(i) = n^{-1}\sum_k (X_k - \bar{X}^{(t)})(X_{k+|i|} - \bar{X}^{(t)})$ where the summation runs for $k = 1, \ldots, n-|i|$ but with $k \neq t$ and $k \neq t-|i|$, and $\bar{X}^{(t)} = (n-1)^{-1}\sum_{k\neq t} X_k$. Then define the delete-x_t YW estimator $\underline{\tilde{\phi}}^{(t)} = (\tilde{\phi_0}^{(t)}, \tilde{\phi_1}^{(t)}, \cdots, \tilde{\phi_p}^{(t)})' = (\hat{\Gamma}_p^{(t)})^{-1}\underline{\hat{\gamma}}_p^{(t)}$ where matrix $\hat{\Gamma}_p^{(t)}$ has ij element $\hat{\gamma}^{(t)}(i-j)$, and vector $\underline{\hat{\gamma}}_p^{(t)}$ has ith element $\hat{\gamma}^{(t)}(i)$. Now, barring positive definiteness issues, we can carry out Algorithm 3.2 with $\underline{\tilde{\phi}}^{(t)}$, $\tilde{x_t}^{(t)} = \tilde{\phi_0}^{(t)} + \sum_{j=1}^{p}\tilde{\phi_j}^{(t)} x_{t-j}$ and $\tilde{\epsilon}_t^{(t)} = x_t - \tilde{x}_t^{(t)}$ instead of $\underline{\hat{\phi}}^{(t)}$, $\hat{x_t}^{(t)}$ and $\hat{\epsilon}_t^{(t)}$ respectively.

Note that if the LS estimator $\underline{\hat{\phi}}$ is causal—as it is hopefully the case—we can use either fitted or predictive residuals but will need to discard all bootstrap pseudo-series that lead to a non-causal $\underline{\hat{\phi}}^*$; this is equally important for the Backward Bootstrap methods discussed in Section 3.4.

3.2. Forward Studentized Bootstrap with Fitted Residuals

In the previous two subsections we have described the forward bootstrap based on predictive roots. However, as already mentioned, we can use studentized predictive roots instead; see Definition 2.1 and Remark 2.2. The forward bootstrap procedure with fitted and/or predictive residuals is similar to Algorithm 3.1; the only differences is in step 3(e) and 6.

To describe it, let ψ_j for $j = 0, 1, \cdots$ be the MA(∞) coefficients of the AR(p) model, i.e., ψ_j is the coefficient associated with z^j in the power series expansion of $\phi^{-1}(z)$ for $|z| \leq 1$, defined by $1/\phi(z) = \psi_0 + \psi_1 z + \cdots \equiv \psi(z)$; the power series expansion is guaranteed by the causality assumption (1.3). It is then easy to see that the variance of the h-step ahead predictive root $X_{n+h} - \hat{X}_{n+h}$ is $\sigma^2\sum_{j=0}^{h-1}\psi_j^2$. The latter can be interpreted as either conditional or unconditional variance since the two coincide in a linear AR(p) model.

Similarly, let $1/\hat{\phi}(z) = \hat{\psi}_0 + \hat{\psi}_1 z + \cdots \equiv \hat{\psi}(z)$, and $1/\hat{\phi}^*(z) = \hat{\psi}_0^* + \hat{\psi}_1^* z + \cdots \equiv \hat{\psi}^*(z)$. Denote by $\hat{\sigma}^2$ and $\hat{\sigma}^{*2}$ the sample variances of the fitted residuals and the bootstrap fitted residuals respectively; the latter are defined as $x_t^* - \hat{x}_t^*$ for $t = p+1, \cdots, n$.

Algorithm 3.3. *Forward Studentized bootstrap with fitted residuals (FSf)*
The algorithm is the same as Algorithm 3.1 except for steps 3(e) and 6 that should be replaced by the following steps:

3(e) Calculate a studentized bootstrap root replicate as

$$\frac{x_{n+h}^* - \hat{x}_{n+h}^*}{\hat{\sigma}^*(\sum_{j=0}^{h-1}\hat{\psi}_j^{*2})^{1/2}}.$$

6 Construct the $(1-\alpha)100\%$ *equal-tailed predictive interval for* X_{n+h} *as*

$$[\hat{x}_{n+h} + \hat{\sigma}(\sum_{j=0}^{h-1}\hat{\psi}_j^2)^{1/2} q(\alpha/2), \hat{x}_{n+h} + \hat{\sigma}(\sum_{j=0}^{h-1}\hat{\psi}_j^2)^{1/2} q(1-\alpha/2)] \tag{3.4}$$

where $q(\alpha)$ *is the* α*-quantile of the empirical distribution of the* B *collected studentized bootstrap roots.*

Remark 3.2. For all the algorithms introduced above, in step 3(d) we redefine the last p values of the bootstrap pseudo-series to match the observed values in order to generate out-of-sample bootstrap data and/or predictors. If we calculate the future bootstrap predicted values and observations without fixing the last p values of the bootstrap pseudo-series, i.e., if we omit step 3(d), then Algorithm 3.3 becomes identical to the method proposed by Masarotto(1990)[26] .

3.2.1. Forward Studentized Bootstrap with Predictive Residuals

As mentioned before, we can resample the predictive—as opposed to the fitted—residuals; the algorithm is as follows.

Algorithm 3.4. *Forward Studentized bootstrap with predictive residuals(FSp)*

1 Same as step 1 in Algorithm 3.1.
2 Same as step 2 in Algorithm 3.2
3-6 Change the $\hat{\epsilon}_t$ *into* $\hat{\epsilon}_t^{(t)}$*; the rest is the same as Algorithm 3.3*

Remark 3.3. As in the regression case discussed in Politis(2013)[33], the Fp method yields improved coverage as compared to the Ff method since predictive residuals are inflated as compared to fitted residuals. Interestingly, the FSp method is not much better than the FSf method in finite samples. The reason is that when we studentize the predictive residuals, the aforementioned inflation effect is offset by the simultaneously inflated bootstrap estimator $\hat{\sigma}^*$ in the denominator. In the Monte Carlo simulations of Section 3.7, we will see that the Fp, FSf and FSp methods have similarly good performance while the Ff method is the worst, exhibiting pronounced undercoverage.

3.3. Asymptotic Properties of Forward Bootstrap

We now discuss the asymptotic validity of the aforementioned Forward Bootstrap methods. First note that Step 3(c) of the Algorithm that concerns the construction of $\hat{\underline{\phi}}^*$ is identical to the related construction of the bootstrap statistic routinely used to derive confidence intervals for $\underline{\phi}$; see e.g. Freedman(1984)[19].

Theorem 3.1 (Freedman(1984)[19])**.** *Let* $\{X_t\}$ *be the causal AR(p) process* (3.1) *with* $E\epsilon_t = 0$, $var(\epsilon_t) = \sigma^2 > 0$ *and* $E|\epsilon_t|^4 < \infty$. *Let* $\{x_1, \cdots, x_n\}$ *denote a realization from* $\{X_t\}$. *Then as* $n \to \infty$,

$$d_0(\mathcal{L}^*\left(\sqrt{n}(\hat{\underline{\phi}}^* - \hat{\underline{\phi}})\right), \mathcal{L}\left(\sqrt{n}(\hat{\underline{\phi}} - \underline{\phi})\right) \xrightarrow{P} 0 \tag{3.5}$$

where $\mathcal{L}, \mathcal{L}^*$ *denote probability law in the real and bootstrap world, and* d_0 *is Kolmogorov distance.*

For the next theorem, continuity (and twice differentiability) of the error distribution are assumed.

Theorem 3.2 (Boldin(1982)[4])**.** *Let* $\hat{F}_n$ *be the empirical distribution of fitted residuals* $\{\hat{\epsilon}_t\}$ *centered to mean zero. Let* F_ϵ *be the distribution of errors* ϵ *satisfying the assumptions of Theorem 3.1 and* $\sup_x |F_\epsilon''(x)| < \infty$. *Then, for any integer* $h \geq 1$,

$$\sup_x |\hat{F}_n(x) - F_\epsilon(x)| = O_p(1/\sqrt{n}) \tag{3.6}$$

Recall the notation $Y_t = (X_t, X_{t-1}, \cdots, X_{t-p+1})'$. Then,

$$X_{n+1}^* = \begin{pmatrix} 1 & Y_n' \end{pmatrix} \hat{\underline{\phi}} + \epsilon_{n+1}^*$$

$$X_{n+1} = \begin{pmatrix} 1 & Y_n' \end{pmatrix} \underline{\phi} + \epsilon_{n+1}.$$

Using eq. (3.6) and Slutsky's Lemma (together with induction on h) shows that

$$d_0(\mathcal{L}^* \left(X_{n+h}^* \right), \mathcal{L} \left(X_{n+h} \right)) \xrightarrow{P} 0$$

from which it follows that the Ff prediction interval (3.3) is asymptotically valid. In view of Theorem 3.1, the stronger property of asymptotic pertinence also holds true.

Corollary 3.3. *Under the assumptions of Theorem 3.1 and Theorem 3.2, the Ff prediction interval (3.3) is asymptotically pertinent.*

We now move on to the Fp interval that is based on predictive residuals. The following lemma shows that the difference between fitted and predictive residuals is negligible asymptotically; still, the difference may be important in small samples.

Lemma 3.4. *Under the assumptions of Theorem 3.1,* $\hat{\epsilon}_t - \epsilon_t^{(t)} = O_p(\frac{1}{n})$.

The proof of Lemma 3.4 is given in the Appendix. The asymptotic equivalence of the fitted and predictive residuals immediately implies that the Fp prediction interval is also asymptotically pertinent.

Corollary 3.5. *Under the assumptions of Theorem 3.1 and Theorem 3.2, the Fp prediction interval of Algorithm 3.2 is asymptotically pertinent.*

3.4. Backward Bootstrap: Definition and Asymptotic Properties

The difference of the backward bootstrap to the forward bootstrap is in the way they generate the bootstrap pseudo-data $X_1^*, \cdots, X_n^*$. The idea of starting from the last p observations (that are given) and generate the bootstrap-pseudo data $\{X_{n-p}^*, \cdots, X_1^*\}$ backward in time using the backward representation

$$\phi(B^{-1})X_t = \phi_0 + w_t$$

was first proposed by Thombs and Schucany(1990)[39] and improved/corrected by Breidt, Davis and Dunsmuir(1995)[10]; here, B is the backward shift operator: $B^k X_t = X_{t-k}$, and $\{w_t\}$ is the backward noise defined by

$$w_t = \frac{\phi(B^{-1})}{\phi(B)} \epsilon_t. \tag{3.7}$$

Thombs and Shucany(1990)[39] generated the fitted backward residuals $\hat{w}_t$ as $\hat{w}_t = x_t - \hat{\phi}_0 - \hat{\phi}_1 x_{t+1} - \cdots - \hat{\phi}_p x_{t+p}$, for $t = 1, 2, \cdots, n-p$. Then they fixed the last p values of the data, and generated the pseudo series backwards through the following backwards recursion, $x_t^* = \hat{\phi}_0 + \hat{\phi}_1 x_{t+1}^* + \cdots + \hat{\phi}_p x_{t+p}^* + w_t^*$, for $t = n-p, n-p-1, \cdots, 1$ with w_t^* being generated i.i.d. from $\hat{F}_w$, the empirical distribution of the (centered) $\hat{w}_t$s.

However, as pointed out by Breidt et al. (1995)[10], although the backward errors w_ts are uncorrelated, they are dependent. So it is not advisable to resample $\{w_t^*\}$ as i.i.d. from $\hat{F}_w$. Nevertheless, the forward errors ϵ_t are independent; so we can generate ϵ_t^* i.i.d. from $\hat{F}_n$. After obtaining the ϵ_t^*s, we are then able to generate the bootstrapped backward noise w_t^* using the bootstrap analog of (3.7), i.e.,

$$w_t^* = \frac{\hat{\phi}(B^{-1})}{\hat{\phi}(B)} \epsilon_t^*.$$

3.4.1. Algorithm for Backward Bootstrap with Fitted Residuals

Our algorithm for backward bootstrap with fitted residuals is exactly the same as that of Breidt et al. (1995)[10]. However, we also propose the backward bootstrap with predictive residuals which has better finite sample properties. In addition, we address the construction of prediction intervals via either unstudentized or studentized predictive roots.

Algorithm 3.5. *Backward bootstrap with fitted residuals (Bf)*

1-2. same as the steps in Algorithm 3.1.

3. Center the fitted residuals: let $r_t = \hat{\epsilon}_t - \bar{\hat{\epsilon}}$ *for* $t = p+1, \cdots, n$, *and* $\bar{\hat{\epsilon}} = (n-p)^{-1}\sum_{p+1}^{n}\hat{\epsilon}_t$, *the empirical distribution of* r_t *is denoted by* $\hat{F}_n$.

(a) Choose a large positive integer M *and create the independent bootstrap pseudo-noise* $\epsilon^*_{-M}, \cdots, \epsilon^*_n, \epsilon^*_{n+1}, \cdots$ *from* $\hat{F}_n$; *then generate the bootstrap backward noises* $\{w^*_t, t = -M, \cdots, n\}$ *recursively as follows:*

$$w^*_t = \begin{cases} 0, & t < -M \\ \hat{\phi}_1 w^*_{t-1} + \cdots + \hat{\phi}_p w^*_{t-p} + \epsilon^*_t - \hat{\phi}_1\epsilon^*_{t+1} - \cdots - \hat{\phi}_p\epsilon^*_{t+p}, & t \geq -M \end{cases}$$

(b) Fix the last p *values,i.e.,* $x^*_n = x_n, \cdots, x^*_{n-p+1} = x_{n-p+1}$, *and then generate a bootstrap realization* $\{X^*_t\}$ *by the backward recursion:*

$$x^*_t = \begin{cases} \hat{\phi}_0 + \hat{\phi}_1 x^*_{t+1} + \cdots + \hat{\phi}_p x^*_{t+p} + w^*_t & t = n-p, n-p-1, \cdots, 1 \\ x_t & t = n, n-1, \cdots, n-p+1. \end{cases}$$

(c) Based on the pseudo-data $\{x^*_1, \cdots, x^*_n\}$, *re-estimate the coefficients* $\underline{\phi}$ *by LS estimators* $\underline{\hat{\phi}}^* = (\hat{\phi_0}^*, \hat{\phi_1}^*, \cdots, \hat{\phi_p}^*)'$ *as in step 1. Then compute the future bootstrap predicted values* $\hat{x}^*_{n+1}, \cdots, \hat{x}^*_{n+h}$ *via:*

$$\hat{x}^*_{n+t} = \hat{\phi_0}^* + \sum_{j=1}^{p} \hat{\phi_j}^* \hat{x}^*_{n+t-j} \text{ for } t = 1, \cdots, h$$

where $\hat{x}^*_{n+t-j} = x^*_{n+t-j}$ *when* $t \leq j$.

(d) Compute the future bootstrap observations $x^*_{n+1}, x^*_{n+2}, \cdots, x^*_{n+h}$ *through the last* p *observations by the forward recursion:*

$$x^*_{n+t} = \hat{\phi}_0 + \sum_{j=1}^{p} \hat{\phi}_j x^*_{n+t-j} + \epsilon^*_{n+t} \quad \text{for } t = 1, 2, \cdots, h.$$

(e) Calculate a bootstrap root replicate as

$$x^*_{n+h} - \hat{x}^*_{n+h}$$

4. Steps (a)-(e) in the above are repeated B *times, and the* B *bootstrap replicates are collected in the form of an empirical distribution whose* α*-quantile is denoted* $q(\alpha)$.

5. *Compute the predicted future values* $\hat{x}_{n+1} \cdots, \hat{x}_{n+h}$ *by following recursion:*

$$\hat{x}_{n+t} = \hat{\phi}_0 + \sum_{j=1}^{p} \hat{\phi}_j \hat{x}_{n+t-j} \text{ for } t = 1, \cdots, h.$$

Note that when $t \leq j$, $\hat{x}_{n+t-j} = x_{n+t-j}$

6. *Construct the* $(1-\alpha)100\%$ *equal-tailed predictive interval for* X_{n+h} *as*

$$[\hat{x}_{n+h} + q(\alpha/2), \hat{x}_{n+h} + q(1-\alpha/2)]. \tag{3.8}$$

3.4.2. Algorithm for Backward Bootstrap with Predictive Residuals

Algorithm 3.6. *Backward bootstrap with predictive residuals (Bp)*

1-2. Same as steps 1-2 in Algorithm 3.2

3-6. Change the $\hat{\epsilon}_t$ *into* $\hat{\epsilon}_t^{(t)}$*, the predictive residuals defined in step 2 of Algorithm 3.2; the rest is the same as in Algorithm 3.5.*

3.4.3. Algorithm for Backward Studentized Bootstrap with Fitted Residuals

Algorithm 3.7. *Backward studentized bootstrap with predictive residuals (BSf)*
This algorithm is the same as Algorithm 3.5 except steps 3(e) and 6 that should be taken as steps 3(e) and 6 of Algorithm 3.3.

3.4.4. Algorithm for Backward Studentized Bootstrap with Predictive Residuals

Algorithm 3.8. *Backward bootstrap with predictive residuals (BSp)*
Change the $\hat{\epsilon}_t$ *into* $\hat{\epsilon}_t^{(t)}$*, the predictive residuals defined in step 2 of Algorithm 3.2; the rest is the same as in Algorithm 3.7.*

Remark 3.4. The asymptotic validity of the backward bootstrap prediction interval with fitted residuals, i.e., interval (3.8), has been proven by Breidt et al. (1995)[10]; it is not hard to see that the property of asymptotic pertinence also holds true here. In view of Lemma 3.4, the backward bootstrap prediction interval with predictive residuals is also asymptotically pertinent, and the same is true for the studentized methods.

Remark 3.5. Similarly to the forward bootstrap methods—see Remark 3.3—, the Bp and BSf methods give improved coverage compared to the Bf method but the technique of predictive residuals does not add much in the studentized case (BSp); see the simulations of Section 3.7.

3.5. Generalized Bootstrap Prediction Intervals

Chatterjee and Bose(2005)[14] introduced the generalized bootstrap method for estimators obtained by solving estimating equations. The LS estimators of the AR coefficients is a special case. With a bootstrapped weight $(w_{n1}, \cdots, w_{nn})$ in the estimating equations, the generalized bootstrapped estimators are obtained simply by solving the bootstrapped estimating equations. The generalized bootstrap method is computationally fast because we do not need to generate the

pseudo-series; instead we just resample the weights $(w_{n1}, \cdots, w_{nn})$ from some distribution, e.g., Multinomial$(n; 1/n, \cdots, 1/n)$.

Inspired by the idea of generalized bootstrap, we now propose a new bootstrap approach for bootstrap prediction intervals in linear AR models.

Algorithm 3.9. *Generalized bootstrap with fitted residuals (Gf)*

1-2. Same as the steps in Algorithm 3.1

3. (a) Calculate the bootstrapped estimator of the coefficients

$$\hat{\underline{\phi}}^* = (X'WX)^{-1}X'WY,$$

where $X = \begin{bmatrix} 1 & x_{n-1} & \cdots & x_{n-p} \\ 1 & x_{n-2} & \cdots & x_{n-p-1} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_p & \cdots & x_1 \end{bmatrix}$, $Y = \begin{pmatrix} x_n \\ x_{n-1} \\ \vdots \\ x_{p+1} \end{pmatrix}$

and W is a diagonal matrix whose diagonal elements $(w_1, \cdots, w_{n-p})$ are sampled from Multinomial $(n-p; 1/(n-p), \cdots, 1/(n-p))$.

(b) Compute the future bootstrap predicted values $\hat{X}^*_{n+1}, \cdots, \hat{X}^*_{n+h}$ by the recursion:

$$\hat{X}^*_{n+t} = \hat{\phi_0}^* + \sum_{j=1}^{p} \hat{\phi_j}^* \hat{X}^*_{n+t-j} \text{ for } t = 1, \cdots, h,$$

and the future bootstrap observations $X^*_{n+1}, X^*_{n+2}, \cdots, X^*_{n+h}$ by the recursion:

$$X^*_{n+t} = \hat{\phi_0} + \sum_{j=1}^{p} \hat{\phi_j} X^*_{n+t-j} + \epsilon^*_{n+t}, \text{ for } t = 1, 2, \cdots, h;$$

as usual, $\hat{X}^*_{n+t-j} = X^*_{n+t-j} = x_{n+t-j}$ when $t \leq j$, and $\epsilon^*_{n+1}, \ldots, \epsilon^*_{n+h}$ are sampled i.i.d. from the empirical distribution of the (centered) fitted residuals. Finally, calculate the bootstrap predictive root replicate as $X^*_{n+h} - \hat{X}^*_{n+h}$.

4-6. Same as the corresponding steps from Algorithm 3.1.

The Generalized bootstrap can also be performed using the predictive residuals.

Algorithm 3.10. *Generalized bootstrap with predictive residuals (Gp)*
The algorithm is identical to Algorithm 3.9 with the following changes: replace step 2 of Algorithm 3.9 with step 2 of Algorithm 3.2, and use the predictive residuals instead of the fitted residuals in step 3[b] of Algorithm 3.9.

Under regularity conditions, Chatterjee and Bose(2005)[14] proved the consistency of the Generalized bootstrap in estimating the distribution of $\sqrt{n}(\hat{\underline{\phi}} - \underline{\phi})$, i.e., equation (3.5). Using Theorem 3.2 and Lemma 3.4, it then follows that both Gf and Gp prediction intervals are asymptotically pertinent.

Remark 3.6. Chatterjee and Bose(2005)[14] shows that the resampling conditional variance of $\sqrt{n}\sigma_n^{-1}(\hat{\underline{\phi}}^* - \hat{\underline{\phi}})$ is consistent for the asymptotic variance of $\sqrt{n}(\hat{\underline{\phi}} - \underline{\phi})$. Note that there's a σ_n^{-1} in the first expression, where σ_n^2 is the variance of w_i. If we sample the weight $(w_1, \cdots, w_n)$ from a multinomial distribution and the variance is $1 - 1/(n-p)$, the difference between the conditional variance of $(\hat{\underline{\phi}}^* - \hat{\underline{\phi}})$ and $\sigma_n^{-1}(\hat{\underline{\phi}}^* - \hat{\underline{\phi}})$ is of order $O(\frac{1}{n^2})$,which is negligible.

3.6. Joint Prediction Intervals

Having observed the time series stretch $\{X_1, \cdots, X_n\}$, we may wish to construct joint, i.e., simultaneous, prediction intervals for $\{X_{n+1}, \cdots, X_{n+H}\}$ for some $H \geq 1$. Let a positive integer $h \leq H$, and denote by $g(Y_n; \underline{\phi})$ the theoretical MSE-optimal predictor of X_{n+h} based on $Y_n = (X_n, \cdots, X_{n-p+1})$. The true X_{n+h}, the practical predictor $\hat{X}_{n+h}$, and the predictive root are given as follows:

$$X_{n+h} = g(Y_n; \underline{\phi}) + \sum_{j=0}^{h-1} \psi_j \epsilon_{n+h-j} \tag{3.9}$$

$$\hat{X}_{n+h} = g(Y_n; \hat{\underline{\phi}}) \tag{3.10}$$

$$X_{n+h} - \hat{X}_{n+h} = \big(g(Y_n; \underline{\phi}) - g(Y_n; \hat{\underline{\phi}})\big) + \sum_{j=0}^{h-1} \psi_j \epsilon_{n+h-j}. \tag{3.11}$$

Under the linear AR model (3.1), $g(\cdot)$ is a linear function of Y_n whose coefficients only depend on $\underline{\phi}$; similarly, the ψ_j are the coefficients of the power series expansion of $\phi^{-1}(z)$, i.e., they only depend on $\underline{\phi}$.

Resampling is extremely useful for the construction of univariate prediction intervals but it is absolutely indispensable for joint prediction intervals. To see why, note that the objective is to estimate the distribution of the predictive root (3.11). The first difficulty is in capturing the distribution of the first term $g(Y_n; \underline{\phi}) - g(Y_n; \hat{\underline{\phi}})$ in (3.11); this term is quite small compared to the second term $U_h = \sum_{j=0}^{h-1} \psi_j \epsilon_{n+h-j}$. Nevertheless, even if we ignore the first term, the major difficulty is that the random variables $U_1, \ldots, U_H$ are dependent. If we assume the $\{\epsilon_t\}$ are i.i.d. $\mathrm{N}(0, \sigma^2)$, then $U_1, \ldots, U_H$ has a multivariate normal distribution given $Y_n = y$. One could estimate its covariance matrix, and form the joint prediction intervals for $\{X_{n+1}, \cdots, X_{n+H}\}$ based on Normal theory. However, this method not only ignores the variability from estimating $\underline{\phi}$ by $\hat{\underline{\phi}}$, i.e., the first term $g(Y_n; \underline{\phi}) - g(Y_n; \hat{\underline{\phi}})$ in (3.11), but it relies on the assumption of normal errors that is nowdays unrealistic.

Nevertheless, the bootstrap can construct joint/simultaneous prediction intervals in a straightforward manner (and without resorting to unrealistic assumptions) since the bootstrap can mimic a multivariate distribution as easily as a univariate one. In our case, we use the bootstrap to mimic the multivariate distribution of a collection of predictive roots or studentized predictive roots.

To construct the joint prediction intervals using one of the bootstrap methods based on predictive roots, the easiest procedure is to approximate the distribution of the maximum predictive root $M_H = \max_{h=1,\ldots,H} |X_{n+h} - \hat{X}_{n+h}|$ by that of its bootstrap analog.

Algorithm 3.11. *Joint prediction intervals based on maximum predictive root*

1. *Choose any one of the aforementioned bootstrap methods, i.e., forward, backward or generalized, with fitted or predictive residuals.*
2. *For each of the B bootstrap replications, construct all H bootstrap predictive roots $X^*_{n+h} - \hat{X}^*_{n+h}$ for $h = 1, \ldots, H$, and let $M^*_H = \max_{h=1,\ldots,H} |X^*_{n+h} - \hat{X}^*_{n+h}|$.*
3. *Collect the B replicates of M^*_H in the form of an empirical distribution whose α-quantile is denoted $q_H(\alpha)$.*
4. *Construct the H intervals*

$$[\hat{X}_{n+h} - q_H(1-\alpha), \hat{X}_{n+h} + q_H(1-\alpha)] \text{ for } h = 1, \ldots, H \tag{3.12}$$

where the h^{th} interval is a prediction interval for X_{n+h}; the above H intervals have joint/simultaneous coverage of $(1-\alpha)100\%$ nominally.

5. *Under the necessary regularity conditions that would render each individual prediction interval to be asymptotically valid and/or pertinent, the H simultaneous intervals (3.12) would be likewise asymptotically valid and/or pertinent.*

Recall that the prediction error variance, i.e., the conditional/unconditional variance of $X_{n+h} - \hat{X}_{n+h}$, equals $\sigma^2(\sum_{j=0}^{h-1} \psi_j^2)$, i.e., it is an increasing function of h. Since the intervals (3.12) are of the type plus/minus the same constant for all h, it follows that the intervals (3.12) are *unbalanced* in the sense that the interval for $h = h_1$ would have bigger (individual) coverage as compared to the interval for $h = h_2$ when $h_2 > h_1$. In order to construct balanced prediction intervals the concept of studentized predictive roots comes in handy.

Algorithm 3.12. *Joint prediction intervals based on maximum studentized predictive root*

1. *Choose any one of the aforementioned bootstrap methods, i.e., forward, backward or generalized, with fitted or predictive residuals.*
2. *For each of the B bootstrap replications, construct all H studentized bootstrap predictive roots*

$$\frac{X^*_{n+h} - \hat{X}^*_{n+h}}{\hat{\sigma}^*(\sum_{j=0}^{h-1} \hat{\psi}_j^{*2})^{1/2}} \text{ for } h = 1, \ldots, H,$$

and let

$$\bar{M}^*_H = \max_{h=1,\ldots,H} \frac{|X^*_{n+h} - \hat{X}^*_{n+h}|}{\hat{\sigma}^*(\sum_{j=0}^{h-1} \hat{\psi}_j^{*2})^{1/2}}.$$

3. *Collect the B replicates of $\bar{M}^*_H$ in the form of an empirical distribution whose α-quantile is denoted $\bar{q}_H(\alpha)$.*
4. *Construct the H intervals*

$$[\hat{X}_{n+h} - \hat{\sigma}(\sum_{j=0}^{h-1} \hat{\psi}_j^2)^{1/2} \bar{q}_H(1-\alpha), \hat{X}_{n+h} + \hat{\sigma}(\sum_{j=0}^{h-1} \hat{\psi}_j^2)^{1/2} \bar{q}_H(1-\alpha)] \text{ for } h = 1, \ldots, H; \quad (3.13)$$

the above H intervals are asymptotically balanced, *and have joint/simultaneous coverage of $(1-\alpha)100\%$ nominally.*

5. *Under the necessary regularity conditions that would render each individual prediction interval to be asymptotically valid and/or pertinent, the H simultaneous intervals (3.13) would be likewise asymptotically valid and/or pertinent.*

3.7. Monte Carlo Studies

In this section, we evaluate the performance of all the 10 aforementioned bootstrap methods, i.e., four forward methods with fitted or predictive residuals using nonstudentized or studentized predictive root (Ff, Fp, FSf and FSp), four corresponding backward methods (Bf, Bp, BSf and BSp) and two generalized bootstrap methods (Gf and Gp) in the case of an AR(1) model: $X_{t+1} = \phi_1 X_t + \epsilon_t$ with

(1) $\phi_1 = 0.9$ or 0.5;
(2) errors ϵ_t i.i.d. from N(0,1) or two-sided exponential(Laplace) distribution rescaled to unit variance;
(3) 500 'true' datasets each of size $n = 50$ or 100, and for each 'true' dataset creating $B = 1000$ bootstrap pseudo-series;
(4) prediction intervals with nominal coverage levels of 95% and 90%.

For the ith 'true' dataset, we use one of the bootstrap methods to create $B = 1000$ bootstrap sample paths (step 4 of the algorithms), and construct the prediction interval (step 6 of the algorithms) $[L_i, U_i]$. To assess the corresponding empirical coverage level (CVR) and average length (LEN) of the constructed interval, we also generate 1000 one-step ahead future values $Y_{n+1,j} = \hat{\phi}_1 x_{ni} + \epsilon_j^*$ for $j = 1, 2, \cdots, 1000$ where $\hat{\phi}_1$ is the estimate from the ith 'true' dataset and x_{ni} is the ith dataset's last value. Then, the empirical coverage level and length from the ith dataset are given by

$$CVR_i = \frac{1}{1000} \sum_{j=1}^{1000} 1_{[L_i, U_i]}(Y_{n+1,j}) \text{ and } LEN_i = U_i - L_i$$

where $1_A(x)$ is the indicator function of set A. Note that the ability to generate the future values $Y_{n+1,j}$ independently from the bootstrap datasets allows us to estimate CVR_i in a more refined way as opposed to the usual 0-1 coverage.

Finally, the coverage level and length for each bootstrap method is calculated by the average $\{CVR_i\}$ and $\{LEN_i\}$ over the 500 'true' datasets, i.e.

$$CVR = \frac{1}{500} \sum_{i=1}^{500} CVR_i \text{ and } LEN = \frac{1}{500} \sum_{i=1}^{500} LEN_i.$$

Note, however, that the value of the last observation x_{ni} is different from dataset to dataset; hence the coverage CVR represents an *unconditional* coverage probability, i.e., an average of the conditional coverage probability discussed in the context of asymptotic validity.

normal $\phi_1 = 0.5$	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
Ff	0.930	3.848	0.490		0.881	3.267	0.386
Fp	0.940	4.011	0.506		0.895	3.405	0.406
Bf	0.929	3.834	0.500		0.880	3.261	0.393
Bp	0.941	4.017	0.521		0.896	3.410	0.410
FSf	0.942	4.036	0.501		0.894	3.391	0.395
FSp	0.941	4.028	0.493		0.894	3.393	0.399
BSf	0.941	4.016	0.514		0.894	3.388	0.402
BSp	0.942	4.033	0.500		0.896	3.402	0.398
Gf	0.930	3.847	0.483		0.881	3.264	0.389
Gp	0.940	4.007	0.502		0.895	3.402	0.399
$n = 100$							
Ff	0.940	3.895	0.357		0.892	3.294	0.283
Fp	0.945	3.968	0.377		0.899	3.355	0.281
Bf	0.940	3.895	0.371		0.892	3.286	0.275
Bp	0.945	3.971	0.375		0.899	3.360	0.289
FSf	0.946	3.981	0.358		0.899	3.355	0.282
FSp	0.945	3.977	0.370		0.899	3.350	0.277
BSf	0.945	3.978	0.366		0.898	3.349	0.275
BSp	0.946	3.978	0.366		0.898	3.352	0.283
Gf	0.940	3.891	0.359		0.891	3.289	0.275
Gp	0.944	3.969	0.383		0.897	3.350	0.284

Table 1: Simulation Results of AR(1) with normal innovations and $\phi_1 = 0.5$

normal $\phi_1 = 0.9$	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
Ff	0.933	3.906	0.489		0.884	3.306	0.395
Fp	0.943	4.063	0.513		0.898	3.443	0.411
Bf	0.927	3.824	0.485		0.878	3.255	0.388
Bp	0.939	4.007	0.516		0.893	3.402	0.408
FSf	0.945	4.107	0.515		0.898	3.450	0.411
FSp	0.945	4.099	0.506		0.899	3.444	0.408
BSf	0.939	4.018	0.502		0.892	3.397	0.403
BSp	0.940	4.032	0.501		0.894	3.406	0.399
Gf	0.928	3.838	0.490		0.878	3.261	0.394
Gp	0.938	3.999	0.505		0.892	3.393	0.406
$n = 100$							
Ff	0.941	3.915	0.355		0.893	3.306	0.282
Fp	0.945	3.989	0.371		0.899	3.368	0.282
Bf	0.939	3.892	0.363		0.891	3.284	0.273
Bp	0.945	3.976	0.372		0.898	3.356	0.287
FSf	0.946	4.005	0.355		0.900	3.373	0.286
FSp	0.946	3.997	0.365		0.899	3.369	0.279
BSf	0.945	3.981	0.362		0.898	3.352	0.274
BSp	0.945	3.982	0.355		0.898	3.355	0.282
Gf	0.939	3.890	0.355		0.890	3.286	0.275
Gp	0.944	3.967	0.381		0.897	3.353	0.285

Table 2: Simulation Results of AR(1) with normal innovations and $\phi_1 = 0.9$

Laplace $\phi_1 = 0.5$	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
Ff	0.930	4.175	0.804		0.881	3.270	0.570
Fp	0.937	4.376	0.828		0.892	3.420	0.597
Bf	0.929	4.176	0.815		0.881	3.267	0.571
Bp	0.937	4.376	0.882		0.892	3.415	0.600
FSf	0.940	4.176	0.873		0.894	3.438	0.578
FSp	0.941	4.376	0.851		0.894	3.452	0.583
BSf	0.939	4.457	0.862		0.893	3.436	0.587
BSp	0.941	4.462	0.875		0.895	3.443	0.583
Gf	0.930	4.177	0.774		0.881	3.274	0.577
Gp	0.937	4.367	0.864		0.892	3.420	0.611
$n = 100$							
Ff	0.939	4.208	0.612		0.891	3.274	0.431
Fp	0.943	4.302	0.638		0.897	3.344	0.439
Bf	0.940	4.220	0.616		0.892	3.274	0.429
Bp	0.943	4.290	0.618		0.896	3.340	0.431
FSf	0.945	4.343	0.622		0.898	3.363	0.431
FSp	0.945	4.349	0.629		0.898	3.362	0.429
BSf	0.945	4.338	0.618		0.898	3.362	0.435
BSp	0.945	4.340	0.615		0.898	3.357	0.424
Gf	0.940	4.238	0.627		0.892	3.285	0.424
Gp	0.943	4.305	0.638		0.897	3.355	0.439

Table 3: Simulation Results of AR(1) with Laplace innovations and $\phi_1 = 0.5$

Laplace $\phi_1 = 0.9$	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
Ff	0.930	4.236	0.826		0.886	3.317	0.574
Fp	0.937	4.417	0.833		0.896	3.462	0.597
Bf	0.929	4.179	0.894		0.881	3.280	0.645
Bp	0.937	4.380	0.977		0.892	3.426	0.681
FSf	0.943	4.520	0.885		0.898	3.502	0.595
FSp	0.943	4.510	0.866		0.899	3.510	0.601
BSf	0.939	4.451	0.963		0.894	3.458	0.671
BSp	0.940	4.481	0.987		0.896	3.470	0.685
Gf	0.929	4.159	0.780		0.880	3.260	0.576
Gp	0.936	4.343	0.864		0.891	3.411	0.597
$n = 100$							
Ff	0.940	4.229	0.620		0.892	3.286	0.429
Fp	0.944	4.321	0.636		0.897	3.358	0.428
Bf	0.940	4.225	0.618		0.891	3.270	0.423
Bp	0.943	4.302	0.626		0.896	3.336	0.427
FSf	0.946	4.367	0.620		0.899	3.382	0.428
FSp	0.946	4.365	0.627		0.899	3.380	0.423
BSf	0.945	4.356	0.624		0.898	3.365	0.431
BSp	0.945	4.353	0.617		0.898	3.356	0.416
Gf	0.940	4.233	0.620		0.892	3.285	0.418
Gp	0.943	4.316	0.641		0.896	3.348	0.436

Table 4: Simulation Results of AR(1) with Laplace innovations and $\phi_1 = 0.9$

Tables 1, 2, 3, 4 summarize the findings of our simulation; the entry for st.err is the standard error associated with each average length.

Some important features are as follows:

- As expected, all bootstrap prediction intervals considered are characterized by some degree of under-coverage. It is encouraging that the use of predictive residuals appears to partially correct the under-coverage problem in linear autoregression as was the case in linear regression; see Politis(2013)[33].
- The Fp, Bp and Gp methods using predictive residuals have uniformly improved CVRs as compared to Ff, Bf and Gf using fitted residuals. The reason is that the finite-sample empirical distribution of the predictive residuals is very much like a re-scaled (inflated) version of the empirical distribution of fitted residuals.
- The price to pay for using predictive residuals is the increased variability of the interval length in all unstudentized methods. However this is a finite-sample effect since asymptotically the omission of a finite number of points from the scatterplot makes little difference; see Lemma 3.4.
- The four studentized methods have similar performance to the respective unstudentized methods using predictive residuals. Thus, using predictive residuals is not deemed necessary for the studentized methods although it does not seem to hurt; see also Remark 3.3.
- The coverages of the Gf intervals resemble that of Ff and Bf intervals. Similarly, the coverages of Gp intervals resemble that of Fp and Bp intervals.

3.8. Alternative Approaches to Bootstrap Prediction Intervals for Linear Autoregression Model

In this section, we will discuss other existing methods for constructing the prediction intervals for linear autoregression. We will compare all the methods mentioned in this section with all the methods

previously proposed in this paper in simulation.

3.8.1. Bootstrap Prediction Intervals Based on Studentized Predictive Roots

Box and Jenkins(1976)[9] proposed a widely used prediction interval for an AR(p) model as

$$[\hat{x}_{n+h} + z_{\alpha/2}\hat{\sigma}(\sum_{j=0}^{h-1}\hat{\psi}_j^2)^{1/2}, \hat{x}_{n+h} + z_{1-\alpha/2}\hat{\sigma}(\sum_{j=0}^{h-1}\hat{\psi}_j^2)^{1/2}], \tag{3.14}$$

where $\hat{\psi}_j$, $j = 0, 1, \cdots$ are the coefficients of the power series $\hat{\psi}(B) = \hat{\phi}^{-1}(B)$, z_α is the αth quantile of a standard normal variate, and $\hat{\sigma}$ is an estimate of σ, the standard deviation of the innovations $\{\epsilon_t\}$. This prediction interval only takes into account the variability from the errors but does not account for the variability from the estimation of the model, thus it is not asymptotically pertinent; in fact, it is the analog of the naive interval (2.5). Furthermore, this interval is asymptotically valid only under the assumption of Gaussian errors which is nowdays unrealistic.

To relax the Gaussianity assumption, and to capture the variability from both the errors and the model estimation, Masarotto (1990)[26] proposed a bootstrap method to construct the prediction interval as follows: for each pseudo series $x_1^*, \cdots, x_n^*, \cdots, x_{n+h}^*$, generate the studentized bootstrap predictive root

$$r^* = \frac{x_{n+h}^* - \hat{x}_{n+h}^*}{\hat{\sigma}^*(\sum_{j=0}^{h-1}\hat{\psi}_j^{*2})^{1/2}}, \tag{3.15}$$

where $\hat{\sigma}^*$ and $\hat{\psi}_j^{*2}$ are obtained from the pseudo-series in the same way $\hat{\sigma}$ and $\hat{\psi}_j$ obtained from the true series. Suppose we generate B values of r^*, and order them as $(r_1^*, \cdots, r_B^*)$. Letting $k = \lfloor B\alpha \rfloor$, the $(1-\alpha)100\%$ prediction interval is

$$[\hat{x}_{n+h} + r_k^*\hat{\sigma}(\sum_{j=0}^{h-1}\hat{\psi}_j^2)^{1/2}, \hat{x}_{n+h} + r_{B-k}^*\hat{\sigma}(\sum_{j=0}^{h-1}\hat{\psi}_j^2)^{1/2}]. \tag{3.16}$$

The main difference of the above from our studentized prediction intervals is that we make it a point—either using Backward or Forward bootstrap—to fix the last p bootstrap pseudo values to the values present in the original series with regard to generating out-of-sample bootstrap data and/or predictors. For example, we obtain the bootstrap predicted value $\hat{x}_{n+h}^*$ and future value x_{n+h}^* in Algorithm 3.1 steps 3(c) and 3(d) using the original datapoints $x_{n-p+1}, \ldots, x_n$, thus ensuring the property of asymptotic pertinence, i.e., capturing the estimation error. As already mentioned, Masarotto's interval (3.16) is *not* asymptotically pertinent. A computationally more efficient version of Masarotto's method was proposed by Grigoletto(1998)[21].

3.8.2. Bootstrap Prediction Intervals Based on Percentile Methods

By contrast to the root/predictive root methods adopted in this paper, some authors have chosen to construct bootstrap prediction intervals via a *percentile* method reminiscent of Efron's [16] percentile method for confidence intervals. To elaborate, the percentile method uses the bootstrap distribution of X_{n+h}^* to estimate the distribution of the future value X_{n+h} while we use the distribution of the bootstrap predictive root (studentized or not) to estimate the distribution of the true predictive root. The methods in this subsection are all based on the percentile method.

Cao et al. (1997)[11] proposed a computationally fast bootstrap method in order to relax the Gaussianity assumption implicit in the Box/Jenkins interval (3.14). Conditionally on the last p observations, they only generate the future bootstrap observations only instead of generating the whole bootstrap series up to x^*_{n+h}. i.e. they define $x^*_s = x_s$, for $s = n-p+1, \cdots, n$, and then compute the future pseudo-data by the recursion:

$$x^*_t = \hat{\phi}_0 + \hat{\phi}_1 x^*_{t-1} + \cdots + \hat{\phi}_p x^*_{t-p} + \hat{\epsilon}^*_t \text{ for } t = n+1, \cdots, n+h. \tag{3.17}$$

As was the case with the Box/Jenkins interval (3.14), the prediction interval of Cao et al. (1997)[11] does not make any attempt to capture the variability stemming from model estimation.

Alonso, Peña, Romo(2002)[1] and Pascual, Romo and Ruiz(2004)[31] used a different way to generate the future bootstrap values; they used the recursion

$$x^*_s = \hat{\phi}^*_0 + \hat{\phi}^*_1 x^*_{s-1} + \cdots + \hat{\phi}^*_p x^*_{s-p} + \hat{\epsilon}^*_s \text{ for } t = n+1, \cdots, n+h \tag{3.18}$$

where $x^*_s = x_s$, for $s = n-p+1, \cdots, n$. Notably, recursion (3.18) generates the future pseudo-values using the parameters $\hat{\phi}^*$ instead of $\hat{\phi}$ as is customary; e.g., compare with recursion (3.17). We will call the percentile interval based on (3.18), the APR/PRR bootstrap method; note that the APR/PRR interval does consider the variability from the model estimation albeit in a slightly different than usual fashion.

Alonso, Peña, Romo(2002)[1] also considered the possibility that the order p is not fixed but allowed to increase with the sample size, i.e., the well-known *AR-sieve* bootstrap. Mukhopadhyay and Samaranayake(2010)[28] also used the APR/PRR method in an AR-Sieve context together with an *ad hoc* inflation of the scale of the bootstrap errors in order to improve coverage.

3.8.3. Monte Carlo Studies

In the following two tables, we provide simulation results with an AR(1) model with $\phi_1 = 0.5$ for the aforementioned methods: Box/Jenkins (BJ), Cao et al. (1997), APR/PRR and Masarotto (M). These should be compared to our 10 methods presented in Tables 1 and 3.

normal $\phi_1 = 0.5$	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
BJ	0.934	3.832	0.402		0.880	3.216	0.338
M	0.946	4.510	0.599		0.898	3.792	0.493
Cao	0.917	3.720	0.532		0.871	3.199	0.417
APR/PRR	0.930	3.858	0.498		0.880	3.268	0.390
$n = 100$							
BJ	0.943	3.887	0.275		0.892	3.262	0.231
M	0.948	4.514	0.430		0.898	3.793	0.348
Cao	0.936	3.853	0.392		0.888	3.262	0.291
APR/PRR	0.939	3.893	0.368		0.891	3.283	0.283

Table 5: Simulation Results of AR(1) with normal innovations and $\phi_1 = 0.5$

Laplace $\phi_1 = 0.5$	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
BJ	0.923	3.812	0.603		0.885	3.199	0.506
M	0.942	4.827	0.960		0.897	3.817	0.692
Cao	0.921	4.065	0.863		0.873	3.197	0.605
APR/PRR	0.930	4.211	0.832		0.882	3.279	0.573
$n = 100$							
BJ	0.931	3.877	0.456		0.894	3.254	0.383
M	0.946	4.802	0.668		0.897	3.789	0.479
Cao	0.938	4.198	0.650		0.888	3.245	0.452
APR/PRR	0.940	4.226	0.628		0.892	3.282	0.434

Table 6: Simulation Results of AR(1) with Laplace innovations and $\phi_1 = 0.5$

To summarize our empirical findings:

- The BJ method has similar coverage rates as APR/PRR and our Ff method when the error is normal. However, when the errors have Laplace distribution, the BJ method performs very poorly.
- Our forward and backward methods with fitted residuals (Ff and Bf) outperform both Cao and APR/PRR methods. This conclusion is expected and consistent with the discussion in the previous sections.
- Our methods with predictive residuals (Fp and Bp) and the studentized methods (FSf,FSp,BSf,BSp) are the best performing in terms of coverage.
- Masarotto's (M) method has similar performance to our FSf method; this was somewhat expected in view of Remark 3.2 but it is also deserving of further discussion—see Remark 3.7 in what follows.

3.8.4. More Monte Carlo: AR(2) model

In this subsection, we provide simulation results based on an AR(2) model:

$$X_t = 1.55X_{t-1} - 0.6X_{t-2} + \epsilon_t.$$

Tables 7 and 8 present the coverages using both normal and Laplace innovations; for each type of innovation we generate 500 data sets each of size $n = 50$ or 100; as before, for each data set, we generate $B = 1000$ pseudo-series. The results are qualitatively similar to the AR(1) simulations.

normal errors	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
BJ	0.926	3.771	0.402		0.868	3.165	0.337
M	0.942	4.098	0.515		0.894	3.455	0.413
Cao	0.905	3.640	0.560		0.859	3.150	0.424
APR/PRR	0.926	3.931	0.523		0.875	3.324	0.410
Ff	0.931	3.933	0.521		0.883	3.328	0.417
Fp	0.946	4.171	0.543		0.902	3.527	0.434
Bf	0.926	3.850	0.499		0.874	3.254	0.394
Bp	0.942	4.100	0.527		0.897	3.471	0.424
FSf	0.946	4.185	0.547		0.901	3.521	0.441
FSp	0.945	4.159	0.537		0.899	3.496	0.431
BSf	0.941	4.082	0.517		0.893	3.429	0.408
BSp	0.941	4.085	0.513		0.894	3.443	0.409
Gf	0.926	3.858	0.524		0.875	3.270	0.416
Gp	0.941	4.098	0.541		0.895	3.470	0.430
$n = 100$	CVR	LEN	st.err		CVR	LEN	st.err
BJ	0.939	3.862	0.274		0.885	3.241	0.230
M	0.945	4.002	0.361		0.897	3.370	0.281
Cao	0.931	3.816	0.393		0.879	3.218	0.286
APR/PRR	0.937	3.904	0.361		0.887	3.296	0.282
Ff	0.939	3.908	0.358		0.889	3.295	0.280
Fp	0.945	4.026	0.366		0.899	3.399	0.283
Bf	0.938	3.893	0.367		0.889	3.294	0.277
Bp	0.946	4.026	0.377		0.898	3.384	0.286
FSf	0.946	3.020	0.362		0.898	3.384	0.286
FSp	0.945	4.014	0.360		0.897	3.381	0.278
BSf	0.945	4.003	0.368		0.897	3.379	0.286
BSp	0.945	4.016	0.368		0.897	3.371	0.279
Ff	0.938	3.895	0.369		0.888	3.284	0.280
Fp	0.945	4.007	0.369		0.899	3.389	0.284

Table 7: Simulation Results of AR(2) model $X_t = 1.55X_{t-1} - 0.6X_{t-2} + \epsilon_t$ with normal innovations

Laplace errors	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
BJ	0.918	3.755	0.580		0.877	3.151	0.487
M	0.940	4.488	0.839		0.894	3.489	0.576
Cao	0.915	3.994	0.841		0.864	3.151	0.599
APR/PRR	0.929	4.235	0.800		0.880	3.335	0.596
Ff	0.931	4.252	0.828		0.883	3.322	0.582
Fp	0.942	4.529	0.873		0.898	3.537	0.613
Bf	0.929	4.200	0.814		0.878	3.266	0.570
Bp	0.939	4.448	0.859		0.893	3.469	0.601
FSf	0.943	4.569	0.858		0.899	3.563	0.611
FSp	0.943	4.563	0.871		0.899	3.564	0.620
BSf	0.940	4.507	0.881		0.894	3.493	0.596
BSp	0.940	4.483	0.832		0.894	3.479	0.586
Gf	0.928	4.201	0.825		0.878	3.279	0.592
Gp	0.939	4.456	0.874		0.893	3.486	0.606
$n = 100$	CVR	LEN	st.err		CVR	LEN	st.err
BJ	0.929	3.842	0.440		0.890	3.224	0.369
M	0.944	4.358	0.596		0.896	3.371	0.420
Cao	0.934	4.151	0.644		0.883	3.213	0.443
APR/PRR	0.938	4.219	0.615		0.887	3.256	0.419
Ff	0.939	4.206	0.566		0.889	3.271	0.410
Fp	0.944	4.357	0.645		0.897	3.387	0.430
Bf	0.937	4.194	0.600		0.888	3.259	0.421
Bp	0.943	4.343	0.653		0.896	3.367	0.438
FSf	0.945	4.374	0.591		0.897	3.388	0.421
FSp	0.944	4.377	0.620		0.898	3.393	0.424
BSf	0.944	4.363	0.623		0.896	3.372	0.429
BSp	0.944	4.369	0.637		0.897	3.377	0.432
Gf	0.938	4.199	0.604		0.888	3.258	0.417
Gp	0.944	4.344	0.630		0.896	3.366	0.427

Table 8: Simulation Results of AR(2) model $X_t = 1.55X_{t-1} - 0.6X_{t-2} + \epsilon_t$ with Laplace innovations

3.8.5. More Monte Carlo: conditional coverages in an AR(1) model

We now revert to the AR(1) model in order to investigate the conditional coverage of some intervals of interest. Recall that the nature of the previous simulations resulted in average coverages since each of the 500 'true' datasets had a different last value.

In order to investigate conditional coverages, we now fix the last value of each of the 'true' datasets to some chosen value; the 'true' datasets are then generated using backward bootstrap with X_n fixed. Tables 9 and 10 compare the Masarotto (M) method with our four forward methods. As discussed in Section 3.8.1, Masarotto's method is a forward studentized method; the only difference between Masarotto's method to our FSf method is that Masarotto does not fix the last p values of the bootstrap series to match the ones from the original series.

normal errors	nominal coverage 95%				nominal coverage 90%		
$X_n = 3$	CVR	LEN	st.err		CVR	LEN	st.err
M	0.947	4.108	0.368		0.899	3.450	0.290
FSf	0.951	4.216	0.355		0.905	3.540	0.272
FSp	0.951	4.222	0.337		0.905	3.535	0.264
Ff	0.946	4.125	0.342		0.898	3.466	0.269
Fp	0.951	4.217	0.341		0.904	3.537	0.265
$X_n = 2$	CVR	LEN	st.err		CVR	LEN	st.err
M	0.945	4.002	0.362		0.899	3.384	0.283
FSf	0.947	4.045	0.357		0.902	3.417	0.274
FSp	0.947	4.049	0.349		0.902	3.413	0.263
Ff	0.943	3.959	0.350		0.895	3.350	0.270
Fp	0.947	4.047	0.358		0.902	3.415	0.270
$X_n = 1$	CVR	LEN	st.err		CVR	LEN	st.err
M	0.944	3.960	0.364		0.897	3.340	0.282
FSf	0.944	3.957	0.369		0.897	3.336	0.279
FSp	0.945	3.968	0.366		0.897	3.335	0.269
Ff	0.939	3.877	0.370		0.891	3.273	0.275
Fp	0.944	3.966	0.380		0.898	3.340	0.269
$X_n = 0$	CVR	LEN	st.err		CVR	LEN	st.err
M	0.945	3.956	0.366		0.897	3.329	0.283
FSf	0.944	3.937	0.371		0.895	3.313	0.281
FSp	0.944	3.949	0.374		0.895	3.312	0.272
Ff	0.939	3.861	0.379		0.889	3.252	0.281
Fp	0.943	3.944	0.389		0.896	3.318	0.273

Table 9: Simulation Results of AR(1) model $X_t = 0.5X_{t-1} + \epsilon_t$ with normal innovations when $n = 100$.

Laplace errors	nominal coverage 95%				nominal coverage 90%		
$X_n = 3$	CVR	LEN	st.err		CVR	LEN	st.err
M	0.946	4.475	0.594		0.897	3.473	0.422
FSf	0.948	4.562	0.572		0.901	3.557	0.401
FSp	0.948	4.568	0.581		0.902	3.563	0.410
Ff	0.944	4.424	0.558		0.895	3.465	0.398
Fp	0.947	4.521	0.578		0.900	3.537	0.410
$X_n = 2$	CVR	LEN	st.err		CVR	LEN	st.err
M	0.944	4.366	0.602		0.897	3.392	0.412
FSf	0.946	4.407	0.592		0.899	3.426	0.404
FSp	0.946	4.403	0.597		0.900	3.434	0.407
Ff	0.940	4.265	0.579		0.893	3.338	0.397
Fp	0.944	4.359	0.593		0.898	3.410	0.407
$X_n = 1$	CVR	LEN	st.err		CVR	LEN	st.err
M	0.943	4.326	0.614		0.896	3.343	0.426
FSf	0.944	4.329	0.613		0.896	3.339	0.424
FSp	0.944	4.333	0.607		0.897	3.352	0.422
Ff	0.938	4.188	0.601		0.889	3.259	0.420
Fp	0.942	4.286	0.610		0.894	3.331	0.427
$X_n = 0$	CVR	LEN	st.err		CVR	LEN	st.err
M	0.944	4.324	0.613		0.896	3.336	0.429
FSf	0.943	4.316	0.622		0.895	3.321	0.429
FSp	0.944	4.310	0.606		0.896	3.334	0.431
Ff	0.938	4.172	0.608		0.889	3.238	0.432
Fp	0.941	4.267	0.609		0.894	2.316	0.439

Table 10: Simulation Results of AR(1) model $X_t = 0.5X_{t-1} + \epsilon_t$ with Laplace innovations when $n = 100$.

The CVRs in Tables 9 and 10 represent conditional coverages given the chosen value for X_n. As expected, the worst coverage is associated with the Ff method. Our other three methods, FSf, FSp and Fp all have very accurate CVRs.

Remark 3.7. Recall that Masarotto's intervals are asymptotically valid but not pertinent. However, they appear to have accurate conditional coverages. To further shed light on this phenomenon, recall that the distribution of the bootstrap predictive root depends on $X_n = x_n$ since

$$X^*_{n+1} - \hat{X}^*_{n+1} = (\hat{\phi} - \hat{\phi}^*)x_n + \epsilon^*_{n+1}. \tag{3.19}$$

Since $\hat{\phi} - \hat{\phi}^* = O_p(1/\sqrt{n})$, it is apparent that the term $(\hat{\phi} - \hat{\phi}^*)x_n$ is small compared to the error term ϵ^*_{n+1}; this is why using the wrong x_n—as Masarotto's method does—can still yield accurate coverages. The situation is similar for studentized bootstrap roots since the first term of the numerator contains a term including x_n. Nevertheless, there is no reason to forego using the correct x_n in the bootstrap predictive root (3.19). To elaborate, Masarotto replaces the term $(\hat{\phi} - \hat{\phi}^*)x_n$ in (3.19) with $(\hat{\phi} - \hat{\phi}^*)X^*_n$ where X^*_n is random (with mean 0). If x_n is near zero and X^*_n happens to be near its mean, then the terms match well; but there is an issue of unnecessary variability here that is manifested with higher standard errors of the lengths of the Masarotto intervals, and also with inflated CVRs—but this inflation is due to a fluke, not a *bona fide* capturing of the predictor variability. Now if x_n is large (in absolute value), there is an issue of bias in the centering of the Masarotto intervals but this is again masked by the unnecessary/excess variability of the term $(\hat{\phi} - \hat{\phi}^*)X^*_n$. Thus, adjusting the last p values of the bootstrap series to match the original ones is highly advisable in an AR(p) model. Furthermore, it becomes crucial in situations with heteroscedastic errors as in eq. (1.2) where the scale of the error also depends on these last p values.

4. Bootstrap prediction intervals for nonlinear AR models

The linear AR model (3.1) is, of course, the simplest special case of the additive model (1.1). Nevertheless, there are situations where the autoregression function $m(\cdot)$ is nonlinear. Furthermore, the errors could have (conditional) heteroscedasticity which would bring us to the more general model (1.2).

As it is cumbersome to provide general theory covering all nonlinear AR models we will address in detail two prominent examples. Section 4.1 focuses on the Threshold AutoRegressive (TAR) models in which the autoregression function $m(\cdot)$ is only piecewise linear. Section 4.3 discusses the Autoregressive Conditional Heteroskedasticity (ARCH) models in which the variance of the error ϵ_t conditional on $X_{t-1}, \ldots, X_{t-p}$ is $\sigma^2(X_{t-1}, \ldots, X_{t-p})$ as in (1.2).

The predictive analysis of nonlinear AR models, including the nonparametric AR models of Section 5, differs from the analysis of linear AR models in two fundamental ways:

- There is no immediate way of formulating a Backward Bootstrap procedure in the nonlinear/nonparametric AR case; this is due to the difficulty in propagating the error backwards via the nonlinear autoregression function. However, the Forward Bootstrap applies *verbatim* including the possibility of resampling the predictive residuals.
- It is not easy to derive the h-step ahead optimal predictors in general when $h > 1$.

To elaborate on the last point, note that models (1.1) and (1.2) are tailor-made for one-step ahead prediction; under the causality assumption (1.3), the quantity $m(X_n, \ldots, X_{n-p+1})$ appearing there is nothing other than the the conditional mean $E(X_{n+1}|X_n, \ldots, X_{n-p+1})$ which is the MSE-optimal predictor of X_{n+1} given $\{X_s, s \leq n\}$. For $h > 1$ one might consider iterating the one-step

ahead optimal predictor in order to recursively impute the missing data $X_{n+1}, \cdots, X_{n+h-1}$ required to finally predict X_{n+h}. In a (causal) linear AR model, this imputation procedure indeed leads to the optimal h-step ahead predictor; however, there are no guarantees that iteration will give an optimal—or even reasonable—predictor in the nonlinear case.

Thus, in what follows we focus on the $h = 1$ case paired with the following concrete recommendation: if indeed h-step ahead prediction intervals are desired with $h > 1$, then work directly with the model

$$X_t = m(X_{t-h}, ..., X_{t-h-p+1}) + \sigma(X_{t-h}, ..., X_{t-h-p+1})\epsilon_t \tag{4.1}$$

instead of model (1.2) whether $\sigma(\cdot)$ is constant or not. Notably, all the procedures discussed in this paper are scatterplot-based so they immediately extend to cover the scatterplot of X_t vs. $(X_{t-h}, ..., X_{t-h-p+1})$ that is associated with model (4.1).

4.1. Bootstrap prediction intervals for TAR models

Threshold autoregressive (TAR) models were introduced by H. Tong more than 30 years ago; see Tong (2011) [40] for a review. A TAR(p) model is a special case of the additive model (1.1) where the autoregression function $m(\cdot)$ is piecewise linear. For example, a two-regime TAR(p) is defined by (1.1) letting

$$m(X_{t-1}, \cdots, X_{t-p}) = (\phi_0^L + \phi_1^L X_{t-1} + \cdots + \phi_p^L X_{t-p})\mathbf{1}\{X_{t-d} < C\}$$
$$+ (\phi_0^R + \phi_1^R X_{t-1} + \cdots + \phi_p^R X_{t-p})\mathbf{1}\{X_{t-d} \geq C\} \tag{4.2}$$

where d is an integer in $[1, p]$, C is the threshold of the two regimes, and both AR models $\underline{\phi}^L$ and $\underline{\phi}^R$ are assumed causal; a multiple-regime TAR model is defined analogously.

TAR models can be estimated in a straightforward way from the scatterplot; see e.g., Chan (1993) [13]. For example, in the TAR(p) model (4.2), one can estimate $\phi_0^L, \phi_1^L, \cdots, \phi_p^L$ by Least Squares (LS) using only the points of the scatterplot of X_t vs. $(X_{t-1}, \cdots, X_{t-p})$ that correspond to $X_{t-d} < C$; similarly, one can estimate $\phi_0^R, \phi_1^R, \ldots, \phi_p^R$ by Least Squares using only the points that correspond to $X_{t-d} \geq C$. The asymptotic theory is immediate as long as the number of scatterplot points in either regime increases in proportion to the sample size. If the threshold C is unknown, it can be estimated (also via LS) at a rate that is faster than $\sqrt{n}$ so that the limit distribution of the LS estimators remains unaffected; see Li and Ling (2012) [25] and the references therein.

The Algorithms for the four Forward bootstrap prediction interval methods (Ff, Fp, FSf, and FSp) under model (4.2) are identical to the corresponding ones from Section 3 with the understanding that LS estimation—both in the real and in the bootstrap world—is performed as described above, i.e., using only the points of the scatterplot that correspond to the relevant regime. It is also immediate to show the asymptotic pertinence of all four Forward bootstrap prediction intervals under standard conditions. Sufficient conditions are Conditions 1–3 of Chan (1993) [13], i.e., that X_t satisfying (4.2) is a stationary ergodic Markov process with finite 4th moments, and that the AR innovations ϵ_t possess a uniformly continuous and strictly positive density function.

4.2. Monte Carlo studies: TAR(1) case

We now present some simulation results using the simple TAR(1) model: $X_t = m(X_{t-1}) + \epsilon_t$ where $m(x) = 0.5x$ if $x < 0$ but $m(x) = 0.9x$ if $x \geq 0$. The value of the threshold $C = 0$ was treated as unknown; it was estimated from the data by minimizing Residual Sum of Squares over the range

from the 15th to the 85th percentile of the data. If the LS estimates of either $\underline{\phi}^L$ or $\underline{\phi}^R$ turned out not causal, then the simulation reverted to fitting a linear AR model covering both regimes.

The construction of the simulation parallels the ones in Section 3, and the results are qualitatively similar with the Ff being inferior to the other three: Fp, FSf, and FSp; see Tables 11 and 12.

Normal	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
Ff	0.917	4.061	0.630		0.861	3.403	0.504
Fp	0.937	4.354	0.630		0.889	3.668	0.513
FSf	0.936	4.398	0.717		0.885	3.658	0.563
FSp	0.935	4.332	0.637		0.884	3.614	0.514
$n = 100$							
Ff	0.930	3.957	0.409		0.876	3.334	0.310
Fp	0.940	4.117	0.387		0.890	3.472	0.304
FSf	0.939	4.112	0.425		0.889	3.458	0.326
FSp	0.939	4.112	0.389		0.888	3.451	0.304

Table 11: Simulation Results of TAR(1) with normal innovations when threshold is unknown.

Laplace	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
Ff	0.925	4.332	0.940		0.874	3.420	0.669
Fp	0.940	4.689	0.999		0.895	3.686	0.672
FSf	0.940	4.775	1.088		0.895	3.744	0.756
FSp	0.939	4.721	1.059		0.894	3.689	0.718
$n = 100$							
Ff	0.935	4.227	0.623		0.884	3.304	0.456
Fp	0.943	4.425	0.624		0.896	3.460	0.457
FSf	0.943	4.446	0.672		0.895	3.457	0.485
FSp	0.943	4.457	0.653		0.896	3.469	0.467

Table 12: Simulation Results of TAR(1) with Laplace innovations when threshold is unknown.

4.3. Bootstrap prediction intervals for ARCH models

Autoregressive Conditional Heteroskedasticity (ARCH) models were introduced by Engle(1982)[4] in an effort to model financial returns and the phenomenon of 'volatility clustering'. In an ARCH(p) model, the variance of the error ϵ_t conditional on $X_{t-1}, \ldots, X_{t-p}$ is a function of $(X_{t-1}, \ldots, X_{t-p})$ as in (1.2). So there is a interesting structure in the conditional variance of X_t given $(X_{t-1}, \ldots, X_{t-p})$. By contrast, in ARCH modeling it is customarily assumed that the conditional mean $m(\cdot) \equiv 0$; in practice this means that the data have had their conditional mean estimated and removed at a preliminary step.

Thus, in this subsection, we consider data from a stationary and ergodic process $\{X_t\}$ that satisfies the ARCH(p) model:

$$X_t = \sigma_{t-1}(\beta)\epsilon_t \text{ with } \sigma^2_{t-1}(\beta) = \beta_0 + \beta_1 X^2_{t-1} + \cdots + \beta_p X^2_{t-p}. \tag{4.3}$$

In the above, $\beta = (\beta_0, \beta_1, \cdots, \beta_p)'$ are the unknown parameters to estimated that are assumed nonnegative, and the errors ϵ_t are i.i.d. (0,1) with finite 4th moment, and independent of $\{X_s, s < t\}$. ARCH models are typically estimated using quasi-maximum likelihood estimation (QMLE); see e.g. Weiss(1986)[41], Bollerslev and Wooldridge (1992)[5], and Francq and Zakoian (2010) [17]. The bootstrap prediction intervals of Reeves(2000)[35], Olave Robio (1999)[29] and Miguel and Olave

(1999) [27] are all based on QMLE; they are of the 'percentile'-type as the APR/PRR method discussed in Section 3.8.2, and do not attempt to capture the estimation error similarly to Cao et al. (1997)[11].

Note that eq. (4.3) can be considered as a model with multiplicative i.i.d. error structure. Model-based resampling can be defined in an analogous way using multiplicative errors instead of additive. Interestingly, model (4.3) also implies an additive model for the squared data, namely:

$$X_t^2 = \beta_0 + \beta_1 X_{t-1}^2 + \cdots + \beta_p X_{t-p}^2 + \tau(X_{t-1}, \ldots, X_{t-p})\xi_t \tag{4.4}$$

where ξ_t is a martingale difference, and $\tau(\cdot)$ an appropriate function; for details, see Kokoszka and Politis (2011) [24] and the references therein. Eq.(4.4) suggests that it may be possible to estimate the ARCH parameters by Least Squares on the scatterplot of X_t^2 vs. $(X_{t-1}^2, \cdots, X_{t-p}^2)$. Indeed, this is possible (and consistent) but not optimal. Instead, Bose and Mukherjee(2003) proposed a linear estimator of the ARCH parameter by solving two sets of linear equations. This method does not involve nonlinear optimization and gives a closed form expression, so it is computationally easier to obtain the estimator compared to QMLE. Simulation results in Bose and Mukherjee(2003)[7] also show that the proposed estimator performs better than the QMLE even for small sample sizes such as $n = 30$.

Bose and Mukherjee(2009)[8] further proposed a weighted linear estimator (WLE) to estimate the ARCH parameters, and a corresponding bootstrap weighted linear estimator (BWLE) that is asymptotically valid. In the next subsection, we extend the method of Bose and Mukherjee(2009)[8], and propose an algorithm for bootstrap prediction intervals for ARCH models based on BWLE.

4.3.1. Bootstrap Algorithm Based on BWLE with Fitted Residuals

Let $\{x_1, \cdots, x_n\}$ be the observations from model (4.3), let $y_i = x_i^2$, $z_i = (1, y_i, y_{i-1}, \cdots, y_{i-p+1})'$, $Z = \begin{pmatrix} z_p' \\ z_{p+1}' \\ \vdots \\ z_{n-1}' \end{pmatrix}$ and $Y = \begin{pmatrix} y_{p+1} \\ y_{p+2} \\ \vdots \\ y_n \end{pmatrix}$. Below is the algorithm for constructing bootstrap prediction intervals for ARCH(p) model based on BWLE with fitted residuals (BWLEf).

Algorithm 4.1. *Bootstrap algorithm based on BWLE with fitted residuals (BWLEf)*

(1) Compute the preliminary weighted least squares estimator (PWLS) as $\hat{\beta}_{pr} = (Z'UZ)^{-1}Z'UY$, *where* U *is a* $(n-p) \times (n-p)$ *diagonal matrix whose* i*th diagonal term is*

$$u_i = \frac{1}{[(1+y_i)\cdots(1+y_{i+p-1})]}.$$

And then compute the weighted linear estimator (WLE) of β *as*

$$\hat{\beta}_n = \Big\{ \sum_{i=p}^{n-1} v_{i-p+1}[z_i z_i'/(z_i'\hat{\beta}_{pr})^2] \Big\}^{-1} \Big\{ \sum_{i=p}^{n-1} v_{i-p+1}[z_i y_{i+1}/(z_i'\hat{\beta}_{pr})^2] \Big\},$$

where $v_i = u_i$ *for* $i = 1, 2, \cdots, n-p$.

(2) Compute the residuals as $\hat{\epsilon}_t = x_t/\sqrt{\hat{\beta}_n' z_{t-1}}$. *Then center the residuals:* $\hat{r}_t = \hat{\epsilon}_t - \frac{1}{n-p}\sum_{i=p+1}^{n} \hat{\epsilon}_i$ *for* $t = p+1, \cdots, n$. *Denote the empirical distribution of* $\hat{r}_t$ *as* $\hat{F}_\epsilon$.

(3) (a) *Generate a* $(n-p)\times(n-p)$ *diagonal matrix* W *whose diagonal elements* $(w_1,\cdots,w_{n-p})$ *are a sample from a multinomial* $(n-p,\frac{1}{n-p},\frac{1}{n-p},\cdots,\frac{1}{n-p})$ *distribution.*

(b) *Compute the bootstrapped preliminary weighted least squares estimator (BPWLS) as* $\hat{\beta}^*_{pr}=(Z'WUZ)^{-1}Z'WUY$ *and the bootstrapped weighted linear estimator (BWLE)* $\hat{\beta}^*_n$ *as*

$$\{\sum_{i=p}^{n-1} w_{i-p+1}v_{i-p+1}[z_iz_i'/(z_i'\hat{\beta}^*_{pr})^2]\}^{-1}\{\sum_{i=p}^{n-1} w_{i-p+1}v_{i-p+1}[z_iy_{i+1}/(z_i'\hat{\beta}^*_{pr})^2]\}$$

(c) *Compute* $H_n^2=var(w_i)=1-\frac{1}{n-p}$ *and*

$$x^*_{n+1}=\sigma_n(\hat{\beta}_n+\frac{\hat{\beta}_n-\hat{\beta}^*_n}{H_n})\epsilon^*_{n+1},$$

where $\sigma_n(\beta)=\sqrt{\beta_0+\beta_1X_n^2+\cdots+\beta_pX^2_{n-p+1}}$, *and* ϵ^*_{n+1} *is generated from* $\hat{F}_\epsilon$ *in step(2).*

(4) *Repeat step(3)* B *times and collect* $x^*_{n+1,1},\cdots,x^*_{n+1,B}$ *in the form of an empirical distribution whose* α*-quantile is denoted as* $q(\alpha)$. *Then the* $(1-\alpha)100\%$ *equal-tailed predictive interval for* X_{n+1} *is given by*

$$[q(\alpha),q(1-\alpha/2)] \tag{4.5}$$

Remark 4.1. Note that under the ARCH model (4.3), $E(X_{n+1}|X_s,s\le n)=0$ by construction, so the interval (4.5) is always an interval around zero. However, the width of the interval crucially depends on the last p values $X_n,\cdots,X_{n-p+1}$, thereby capturing the 'volatility' of the process. In addition, interval (4.5) can also capture potential asymmetry/skewness of the process as it will not be exactly centered at zero.

Remark 4.2. There are other choices available for the preliminary weight U, bootstrapped preliminary weight W and the bootstrap weight V as long as they satisfy assumption(W1) and equations (11),(12),(17)-(19),(21) of Bose and Mukherjee(2009)[8]; see also Chatterjee and Bose(2005)[14] for more examples of the bootstrapped preliminary weight W.

4.3.2. Bootstrap Algorithm Based on BWLE with Predictive Residuals

An advantage of using scatterplot-based estimators is that we can obtain the predictive residuals through deleting one data-point from the scatterplot. To get the predictive residual $\hat{\epsilon}^{(t)}_t$, we can exclude the pair (z_{t-1},y_t) from the scatter plot of y_k vs. z_{k-1} in Algorithm 4.1.

Algorithm 4.2. *Bootstrap algorithm based on BWLE with predictive residuals (BWLEp)*
To get the bootstrap algorithm based on BWLE with predictive residuals we only need to substitute $\{\hat{\epsilon}^{(t)}_t,t=p+1,\cdots,n\}$ *for* $\{\hat{\epsilon}_t,t=p+1,\cdots,n\}$ *in step(2) of Algorithm 4.1; the rest is the same.*

The following steps describe how to get the predictive residuals $\{\hat{\epsilon}^{(t)}_t,t=p+1,\cdots,n\}$ *in detail.*

1 *Let* $Z^{(t)}$ *be the matrix* Z *excluding the row* z'_{t-1}, $Y^{(t)}$ *be the vector* Y *excluding* y_t, $U^{(t)}$ *be the diagonal matrix excluding the diagonal element* u_{t-p}.

2 *Compute* $\hat{\beta}^{(t)}_{pr}=(Z'^{(t)}U^{(t)}Z^{(t)})^{-1}Z'^{(t)}U^{(t)}Y^{(t)}$, *and*

$$\hat{\beta}^{(t)}_n=\{\sum_{i=p,i\neq t-1}^{n-1} u_{i-p+1}[z_iz_i'/(z_i'\hat{\beta}^{(t)}_{pr})^2]\}^{-1}\{\sum_{i=p}^{n-1} u_{i-p+1}[z_iy_{i+1}/(z_i'\hat{\beta}^{(t)}_{pr})^2]\}.$$

3 *Finally compute the predictive residual* $\hat{\epsilon}^{(t)}_t=x_t/\sqrt{\hat{\beta}'^{(t)}_n z_{t-1}}$.

4.3.3. Asymptotic Properties of BWLEf and BWLEp

Assume that the process $\{X_t, t \geq 1\}$ is stationary and ergodic and satisfies (4.3) and $E(\epsilon_t^4) < \infty$. We further assume that the assumptions for all the weights V, U and W mentioned in Remark 4.2 are satisfied and the assumptions of Lemma 2 from Bose and Mukherjee(2009) hold. Under these assumptions, Theorem 2 of Bose and Mukherjee(2009)[8] showed

$$\sup\{|F_n^*(x) - F_n(x)|; x \in \mathbb{R}^{p+1}\} = o_p(1) \tag{4.6}$$

where F_n and F_n^* denote the cumulative distribution functions of $\sqrt{n}(\hat{\beta}_n - \beta)$ and $H_n^{-1}\sqrt{n}(\hat{\beta}_n^* - \hat{\beta}_n)$ in the real and bootstrap world respectively.

Recall that in step (3)(c) of Algorithms 4.1 and 4.2, we employed eq. (4.6) to approximate the distribution of $\beta - \hat{\beta}_n$ by the distribution of $\frac{\hat{\beta}_n - \hat{\beta}_n^*}{H_n}$. The result is that the prediction intervals from Algorithms 4.1 and 4.2 are both asymptotically valid; they can also be called asymptotically pertinent with an appropriate modification of the definition of pertinence to account for the multiplicative model (4.3).

4.3.4. Bootstrap Algorithm Based on QMLE

Bootstrap prediction intervals for ARCH models based on QMLE were proposed by Reeves(2000)[35], Olave Robio (1999)[29] and Miguel and Olave (1999) [27]; for easy reference, their algorithm can be described as follows.

Algorithm 4.3. *Bootstrap algorithm based on QMLE*

1 *Fit an ARCH(p) model to the data. Let* $\hat{\beta} = (\hat{\beta}_0, \hat{\beta}_1, \cdots, \hat{\beta}_p)'$ *denote the QMLE estimates.*

2 *Calculate the residuals:* $\hat{\epsilon}_t = x_t/\sigma_{t-1}(\hat{\beta})$, *for* $t = p+1, \cdots, n$. *And then center the residuals:* $r_t = \hat{\epsilon}_t - \bar{\hat{\epsilon}}$ *for* $t = p+1, \cdots, n$, *where* $\bar{\hat{\epsilon}} = (n-p)^{-1}\sum_{t=p+1}^{n}\hat{\epsilon}_t$.

3 (a) *Use* $\hat{\beta}$ *and residuals* $\{\hat{\epsilon}_t\}$, *along with initial conditions* $u_1^* = x_1, \cdots, u_p^* = x_p$ *to generate* $\{u_t^*, t \geq p+1\}$ *by recursion:*

$$u_t^* = \sqrt{\hat{\beta}_0 + \hat{\beta}_1 u_{t-1}^{*2} + \cdots + \hat{\beta}_p u_{t-p}^{*2}}\ \epsilon_t^*,$$

where ϵ_t^* *is a random draw from the pool of centered residuals* $\{r_t, t = p+1, \cdots, n\}$. *To ensure stationarity of the pseudo-series, generate* $n+m$ *pseudo data for some large positive* m, *i.e.* $\{u_1^*, \cdots, u_n^*, u_{n+1}^*, \cdots, u_{n+m}^*\}$, *and discard the first* m *data.*

(b) *Fit an ARCH(p) model to pseudo-data* $\{x_t^* = u_{t+m}^*, t = 1, 2, \cdots, n\}$ *and re-estimate the QMLE* $\hat{\beta}^* = (\hat{\beta}_0^*, \hat{\beta}_1^*, \cdots, \hat{\beta}_p^*)'$.

(c) *Fix the last p pseudo-data to the true data:* $x_{n-p+1}^* = x_{n-p+1}, x_{n-p+2}^* = x_{n-p+2}, \cdots, x_n^* = x_n$ *and generate the future bootstrap value* $\{x_{n+t}^*, t \geq 1\}$ *by the following recursion,*

$$x_{n+t}^* = \sqrt{\hat{\beta}_0^* + \hat{\beta}_1^* x_{n+t-1}^{*2} + \cdots + \hat{\beta}_p^* x_{n+t-p}^{*2}}\ \epsilon_{n+t}^*,$$

where ϵ_{n+t}^* *is a random draw from the centered residuals.*

4 *Repeat steps 3(a)-(c) B times and collect B bootstrap h-step ahead future values in the form of empirical distribution whose* α*-quantile is denoted* $q(\alpha)$. *Construct the* $(1-\alpha)100\%$ *equal-tailed prediction intervals for* X_{n+h} *as*

$$[q(\alpha/2), q(1-\alpha/2)].$$

The above prediction interval is of the 'percentile'-type as discussed in Section 3.8.2.

Remark 4.3. Our bootstrap Algorithm BWLEf and BWLEp are computationally faster and more stable as compared to the bootstrap method based on QMLE because of the following two reasons. First, BWLEf and BLWEp have closed form expressions of the solutions of the two linear equations involved while QMLE requires numerical optimization. Secondly, there is no need to create pseudo-series through recursion in Algorithms for BWLEf and BWLEp.

4.4. Monte Carlo Studies

We use Monte Carlo simulations to assess the performance of our two methods, BWLEf and BWLEp from Algorithms 4.1 and 4.2, and the bootstrap method based on QMLE of Reeves(2000)[35], Olave Robio (1999)[29] and Miguel and Olave (1999) [27]. We create 500 data sets for each of the following scenarios: sample size $n = 50, 100$ or 200; innovations are from standard normal or Laplace (rescaled to unit variance) distribution; data are generated from Model 1 or Model 2 listed below.

- Model 1: ARCH(1), $X_t = \sqrt{0.5 + 0.25X_{t-1}^2}\ \epsilon_t$
- Model 2: ARCH(2), $X_t = \sqrt{0.1 + 0.2X_{t-2}^2}\ \epsilon_t$

normal errors	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
BWLEf	0.927	3.040	0.762		0.877	2.610	0.608
BWLEp	0.941	3.269	0.883		0.890	2.675	0.654
QMLE	0.924	2.964	0.657		0.873	3.507	0.523
$n = 100$	CVR	LEN	st.err		CVR	LEN	st.err
BWLEf	0.940	3.098	0.607		0.890	2.607	0.491
BWLEp	0.946	3.203	0.631		0.896	2.655	0.496
QMLE	0.937	3.055	0.562		0.888	2.582	0.448
$n = 200$	CVR	LEN	st.err		CVR	LEN	st.err
BWLEf	0.945	3.190	0.720		0.896	2.680	0.587
BWLEp	0.948	3.230	0.716		0.898	2.700	0.590
QMLE	0.943	3.152	0.650		0.894	2.661	0.545

Table 13: Simulation Results of ARCH(1) model 1 with normal innovations

Laplace errors	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
BWLEf	0.931	3.428	1.457		0.883	2.645	1.080
BWLEp	0.943	3.811	1.735		0.892	2.773	1.128
QMLE	0.928	3.279	1.211		0.877	2.542	0.892
$n = 100$	CVR	LEN	st.err		CVR	LEN	st.err
BWLEf	0.941	3.466	1.503		0.894	2.662	1.086
BWLEp	0.947	3.615	1.500		0.897	2.717	1.101
QMLE	0.937	3.334	1.228		0.887	2.575	0.919
$n = 200$	CVR	LEN	st.err		CVR	LEN	st.err
BWLEf	0.941	3.320	0.882		0.892	2.559	0.644
BWLEp	0.944	3.378	0.865		0.893	2.577	0.631
QMLE	0.940	3.272	0.778		0.889	2.531	0.569

Table 14: Simulation Results of ARCH(1) model 1 with Laplace innovations

normal errors	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
BWLEf	0.926	1.339	0.337		0.875	1.124	0.275
BWLEp	0.943	1.477	0.407		0.894	1.193	0.295
QMLE	0.926	1.304	0.225		0.875	1.099	0.179
$n = 100$	CVR	LEN	st.err		CVR	LEN	st.err
BWLEf	0.936	1.351	0.264		0.886	1.135	0.212
BWLEp	0.946	1.414	0.277		0.895	1.169	0.220
QMLE	0.936	1.332	0.189		0.887	1.125	0.151
$n = 200$	CVR	LEN	st.err		CVR	LEN	st.err
BWLEf	0.942	1.374	0.237		0.892	1.154	0.198
BWLEp	0.948	1.407	0.252		0.898	1.173	0.202
QMLE	0.943	1.366	0.183		0.892	1.148	0.147

Table 15: Simulation Results of ARCH(2) model 2 with normal innovations

Laplace errors	nominal coverage 95%				nominal coverage 90%		
$n = 50$	CVR	LEN	st.err		CVR	LEN	st.err
BWLEf	0.930	1.599	1.485		0.883	1.242	1.105
BWLEp	0.944	1.852	2.051		0.896	1.324	1.130
QMLE	0.928	1.537	2.224		0.879	1.134	0.982
$n = 100$	CVR	LEN	st.err		CVR	LEN	st.err
BWLEf	0.941	1.454	0.357		0.891	1.116	0.257
BWLEp	0.948	1.549	0.405		0.898	1.156	0.271
QMLE	0.939	1.434	0.419		0.890	1.103	0.245
$n = 200$	CVR	LEN	st.err		CVR	LEN	st.err
BWLEf	0.939	1.486	0.509		0.890	1.145	0.365
BWLEp	0.943	1.523	0.500		0.893	1.165	0.380
QMLE	0.941	1.455	0.391		0.890	1.122	0.212

Table 16: Simulation Results of ARCH(2) model 2 with Laplace innovations

We compare the coverage levels(CVR) and length(LEN) of all the intervals constructed by the three methods mentioned above with nominal coverage levels of 95% and 90%. Tables 13, 14, 15 and 16 show that the BWLEp method outperforms both BWLEf and QMLE with respect to the coverage level but the variability of the interval length is increased as a price to pay for using predictive intervals. Interestingly, BWLEf and QMLE have similar coverage level but the QMLE has the smallest variance of interval length.

5. Bootstrap Prediction Intervals for Nonparametric Autoregression

In this section, we construct bootstrap prediction intervals in a general nonparametric autoregression model fitted via kernel smoothing. As previously mentioned, the forward bootstrap method—in all variations—is the unifying principle for bootstrap all AR models, linear or nonlinear. Thus, for nonparametric AR models we also employ the forward bootstrap method with fitted or predictive residuals as in Sections 3 and 4, and show that it properly estimates the distribution of the future value capturing both the variability of the kernel estimator and the variability of the innovations from the autoregression model.

In Section 5.1, we give the bootstrap procedure for a nonparametric autoregression with i.i.d. errors, i.e., model (1.1). In Section 5.2, we extend the model allowing heteroscedastic errors, i.e., model (1.2). In either case, the functions $m(\cdot)$ and $\sigma(\cdot)$ are unknown but assumed smooth and $\epsilon_t \sim$ i.i.d. $(0, \sigma^2)$; in (1.2), we further assume $\sigma^2 = 1$. Monte Carlo simulations are given in Section 5.3.

5.1. Nonparametric Autoregression with i.i.d Innovations

In this subsection, we consider a stationary and geometrically ergodic process of the form (1.1) with the conditional mean function $m(\cdot)$ being unknown but assumed smooth. We first give the algorithms using fitted residuals and predictive residuals and then discuss their asymptotic validity.

5.1.1. Forward Bootstrap Algorithm with Fitted and Predictive Residuals

Given a sample $\{x_1, x_2, \cdots, x_n\}$, let $y_t = (x_t, x_{t-1}, \cdots, x_{t-p+1})'$ as before. The resampling algorithms for the predictive distribution of future value X_{n+1} are as follows.

Algorithm 5.1. *Forward Bootstrap with Fitted Residuals*

(1) For $y \in \mathbb{R}^p$, construct the Nadaraya-Watson kernel estimator $\hat{m}(\cdot)$ as

$$\hat{m}(y) = \frac{\sum_{t=p}^{n-1} K(\frac{\|y-y_t\|}{h}) x_{t+1}}{\sum_{t=p}^{n-1} K(\frac{\|y-y_t\|}{h})}, \tag{5.1}$$

where $\|\cdot\|$ is a norm in $\mathbb{R}^p$, $K(\cdot)$ is compactly supported, symmetric density function on $\mathbb{R}$ with bounded derivative, and satisfying $\int K(v)dv = 1$. The bandwidth satisfies $h \to 0$ but $hn \to \infty$.

(2) Compute the fitted residuals: $\hat{\epsilon}_i = x_i - \hat{m}(y_{i-1})$, for $i = p+1, \cdots, n$

(3) Center the residuals: $\hat{r}_i = \hat{\epsilon}_i - (n-p)^{-1} \sum_{t=p+1}^{n} \hat{\epsilon}_t$, for $i = p+1, \cdots, n$.

(a) Sample randomly(with replacement) from the values $\hat{r}_{p+1}, \cdots, \hat{r}_n$ to create bootstrap pseudo errors ϵ_i^*, $i = -M+p, \cdots, n+1$ for some large positive M.

(b) Set $(x^*_{-M}, x^*_{-M+1}, \cdots, x_{-M+p-1})$ equal to p consecutive values drawn from $\{x_1, \cdots, x_n\}$. Then generate x_i^*, by the recursion:

$$x_i^* = \hat{m}(y_{i-1}^*) + \epsilon_i^* \text{ for } i = -M+p, \cdots, n.$$

(c) Drop the first M 'burn in' observations to make sure that the starting values have an insignificant effect. Then construct the kernel estimator $\hat{m}^*(\cdot)$ from the bootstrap series $\{x_1^*, \cdots, x_n^*\}$, i.e., let

$$\hat{m}^*(y) = \frac{\sum_{i=p}^{n-1} K(\frac{\|y-y_i^*\|}{h}) x_{i+1}^*}{\sum_{i=p}^{n-1} K(\frac{\|y-y_i^*\|}{h})} \tag{5.2}$$

where $y_t^* = (x_t^*, x_{t-1}^*, \cdots, x_{t-p+1}^*)'$.

(d) Now fix the last p pseudo values to be the true observations, i.e., redefine $y_n^* = y_n$, and then calculate the bootstrap predictor

$$\hat{X}_{n+1}^* = \hat{m}^*(y_n^*) = \hat{m}^*(y_n)$$

and the future bootstrap observation

$$X_{n+1}^* = \hat{m}(y_n^*) + \epsilon_{n+1}^* = \hat{m}(y_n) + \epsilon_{n+1}^*.$$

(e) Calculate the bootstrap predictive root replicate as $X_{n+1}^* - \hat{X}_{n+1}^*$.

(4) steps (a)-(e) in the above are repeated B times, and the B bootstrap predictive root replicates are collected in the form of an empirical distribution whose α-quantile is denoted $q(\alpha)$.

(5) Then, a $(1-\alpha)100\%$ *equal-tailed predictive interval for* X_{n+1} *is given by*

$$[\hat{m}(y_n) + q(\alpha/2), \hat{m}(y_n) + q(1-\alpha/2)] \tag{5.3}$$

Estimating $m(\cdot)$ in the above could in principle be done via different smoothing methods, e.g., local polynomials, splines, etc. We employ the Nadaraya-Watson kernel estimator $\hat{m}(\cdot)$ just for simplicity and concreteness. To define the predictive residuals, however, recall that the chosen estimator must be scatterplot-based.

Algorithm 5.2. *Forward Bootstrap with Predictive Residuals*

(1) same as step(1) of Algorithm 5.1.

(2) Use the delete-x_t *dataset as described in Section 3.1.2 to compute the delete-one kernel estimator*

$$\hat{m}^{(t)}(y) = \frac{\sum_{i=p+1, i\neq t}^{n} K(\frac{\|y-y_{i-1}\|}{h}) x_i}{\sum_{i=p+1, i\neq t}^{n} K(\frac{\|y-y_{i-1}\|}{h})} \text{ for } t = p+1, \cdots, n. \tag{5.4}$$

Then calculate the predictive residuals: $\hat{\epsilon}_t^{(t)} = x_t - \hat{m}_t^{(t)}(y_{t-1})$ *for* $t = p+1, \cdots, n$.

(3)-(5) Replace $\hat{\epsilon}_t$ *by* $\hat{\epsilon}_t^{(t)}$ *in Algorithm 5.1; the remaining steps are the same.*

The studentized versions of Algorithm 5.1 and 5.2 are defined analogously to the ones in Section 3.

Algorithm 5.3. *Forward Studentized bootstrap with fitted residuals (FSf) or predictive residuals (FSp)*
For FSf, define $\hat{\sigma}$ *and* $\hat{\sigma}^*$ *to be the sample standard deviation of the fitted residuals* $\hat{\epsilon}_t$ *and bootstrap residuals* $\hat{\epsilon}_t^*$ *respectively. For FSp, define* $\hat{\sigma}$ *and* $\hat{\sigma}^*$ *to be the sample standard deviation of the predictive residuals* $\hat{\epsilon}_t^{(t)}$ *and their bootstrap analogs* $\hat{\epsilon}_t^{*(t)}$ *respectively.*
Then, replace steps 3(e) and 6 of Algorithm 3.1 and/or 5.2 by the following steps:

3(e) Calculate a studentized bootstrap root replicate as $(X_{n+1}^* - \hat{X}_{n+1}^*/\hat{\sigma}^*$.

(6) Construct the $(1-\alpha)100\%$ *equal-tailed predictive interval for* X_{n+h} *as*

$$[\hat{X}_{n+1} + \hat{\sigma}\ q(\alpha/2), \hat{X}_{n+1} + \hat{\sigma}\ q(1-\alpha/2)] \tag{5.5}$$

where $q(\alpha)$ *is the* α*-quantile of the empirical distribution of the* B *collected studentized bootstrap roots.*

5.1.2. Asymptotic Properties

In this subsection, we focus on a stationary and geometrically ergodic process of order $p = 1$; the general case of order p is similar. Then, the nonparametric autoregression model takes the simple form

$$X_t = m(X_{t-1}) + \epsilon_t \tag{5.6}$$

where the innovations $\{\epsilon_t\}$ are i.i.d. with mean zero, variance σ^2, and distribution F_ϵ that is continuous with density f_ϵ that is strictly positive; as always, we assume the causality assumption (1.3). To ensure $\{X_t\}$ is geometrically ergodic, the following condition is sufficient:

(A) $\{X_t\}$ obeys (5.6) with $|m(x)| \leq C_1 + C_2|x|$ for all x and some $C_1 < \infty$, $C_2 < 1$.

In a very important work, Franke, Kreiss and Mammen (2002) [18] showed the consistency of the bootstrap in constructing confidence bands for the autoregression function in the nonparametric model (5.6).

Theorem 5.1 (Franke, Kreiss and Mammen (2002) [18])**.** *Consider a dataset $X_1 = x_1, \ldots, X_n = x_n$ from model (5.6). Assume assumption (A) given above, as well as assumptions (AB1)–(AB10) of Franke, Kreiss and Mammen (2002) [18]. Also assume $h \to 0$ but $hn \to \infty$.*
If $h = O(n^{-1/5})$, then

$$d_0(F_\epsilon, \hat{F}_n) \xrightarrow{P} 0 \text{ as } n \to \infty$$

where d_0 is Kolmogorov distance, and $\hat{F}_n$ is the empirical distribution of fitted residuals centered at mean zero.
Furthemore, if $h = o(n^{-1/5})$, then

$$d_0(\mathcal{L}^*(\sqrt{nh}\{\hat{m}(x_n) - \hat{m}^*(x_n)\}), \mathcal{L}(\sqrt{nh}\{m(x_n) - \hat{m}(x_n)\}) \xrightarrow{P} 0. \tag{5.7}$$

Below is the analog of Lemma 3.4 in the nonparametric AR case; its proof is in the Appendix.

Lemma 5.2. *Under the assumptions of Theorem 5.1 with $h = O(n^{-1/5})$, we have $\hat{\epsilon}_t - \hat{\epsilon}_t^{(t)} = O_p(\frac{1}{n})$ as $n \to \infty$.*

From the above results, the following corollary is immediate.

Corollary 5.3. *Under the assumptions of Theorem 5.1 with h satisfying $hn^{1/5} \to c \geq 0$, we have: If $c > 0$, then the prediction interval (5.3) is asymptotically valid, and the same is true for its analog using predictive residuals, i.e., the interval of Algorithm 5.2. Similarly, the two studentized intervals of Algorithm 5.3 (based on fitted or predictive residuals) are asymptotically valid.*
If $c = 0$, the four intervals mentioned above are also asymptotically pertinent.

Remark 5.1. The condition $hn^{1/5} \to c > 0$ leads to optimal smoothing in that the large-sample MSE of $\hat{m}(x_n)$ is minimized. In this case, however, the bias of $\hat{m}(x_n)$ becomes of exact order $O(1/\sqrt{hn})$ which is the order of its standard deviation, and (5.7) fails because the bootstrap can not capture the bias term exactly. This is of course important for confidence interval construction—for which (5.7) was originally developed—and is routinely solved via one of three approaches: (a) plugging-in explicit estimates of bias in the two distributions appearing in (5.7); (b) using a bandwidth satisfying $hn^{1/5} \to 0$ leading to *under-smoothing*, i.e., making the bias of $\hat{m}(x_n)$ negligible as compared to the standard deviation; or (c) using the optimal bandwidth $h \sim cn^{-1/5}$ with $c > 0$ but resampling based an over-smoothed estimator. Either of these approaches work—the simplest being under-smoothing—but note that the problem is not as crucial for prediction intervals that remain asymptotically valid in both cases $c > 0$ or $c = 0$. Furthermore, using the optimal bandwidth, the quantity appearing in part (ii) of Definition 2.4 would be $O_p(1)$ instead of $o_p(1)$ so the four intervals mentioned in Corollary 5.3 could be called 'almost' pertinent in the sense that they capture correctly the order of magnitude of the estimation error which is $O(1/\sqrt{hn})$.

Resampling based an over-smoothed estimator will be further explored in the next section.

5.2. Nonparametric Autoregression with Heteroscedastic Innovations

We now consider the nonparametric autoregression model (1.2). Similarly to Section 5.1, we use Nadaraya-Watson estimators to estimate the unknown smooth functions m and σ. In particular, $\hat{m}(y)$ is exactly as given in (5.1) while $\hat{\sigma}^2(y)$ is defined as

$$\hat{\sigma}^2(y) = \frac{\sum_{t=p}^{n-1} K(\frac{\|y-y_t\|}{h})(x_{t+1} - \hat{m}(y_t))^2}{\sum_{t=p}^{n-1} K(\frac{\|y-y_t\|}{h})} \tag{5.8}$$

where, for simplicity, we use the same bandwidth h as the one used for $\hat{m}(y)$.

Remark 5.2. As mentioned in Remark 5.1, in generating the bootstrap pseudo-series it may be advantageous to use over-smoothed estimators of m and σ that will be denoted by $\hat{m}_g$ and $\hat{\sigma}_g$ respectively; these are computed in the exact same way as $\hat{m}$ and $\hat{\sigma}$ but using an over-smoothed bandwidth g instead of h that satisfies

$$g/h \to \infty \text{ with } h \sim cn^{-1/5} \text{ for some } c > 0. \tag{5.9}$$

Such over-smoothing was originally proposed for bootstrap confidence intervals in nonparametric regression by Härdle and Marron(1991)[23]. It can also be useful in the nonparametric AR model (1.1) with i.i.d. innovations but it is particularly helpful in the heteroscedastic model (1.2).

5.2.1. Forward Bootstrap Algorithm with Fitted Residuals

Given a stationary sample $\{x_1, x_2, \cdots, x_n\}$, let $y_t = (x_t, \cdots, x_{t-p+1})'$. The resampling algorithm for the predictive distribution of future value X_{n+1} is as follows:

Algorithm 5.4.

(1) Construct the estimates $\hat{m}(\cdot)$ and $\hat{\sigma}^2(\cdot)$ by formulas (5.1) and (5.8).

(2) Compute the residuals:

$$\hat{\epsilon_i} = \frac{x_i - \hat{m}(y_{i-1})}{\hat{\sigma}(y_{i-1})} \tag{5.10}$$

for $i = p+1, \cdots, n$

(3) Center the residuals: $\hat{r}_i = \hat{\epsilon}_i - (n-p)^{-1}\sum_{t=p+1}^{n}\hat{\epsilon}_t$, for $i = p+1, \cdots, n$.

(a) Sample randomly (with replacement) from the values $r_{p+1}, \cdots, r_n$ to create bootstrap pseudo errors ϵ_i^* for $i = -M+p, \cdots, 1, 2, \cdots, n+1$ where M is some large positive integer.

(b) Set $(x^*_{-M}, x^*_{-M+1}, \cdots, x_{-M+p-1})$ equal to p consecutive values from $\{x_1, \cdots, x_n\}$, and then generate x_i^* by the recursion:

$$x_i^* = \hat{m}_g(y_{i-1}^*) + \hat{\sigma}_g(y_{i-1}^*)\epsilon_i^* \text{ for } i = -M+p, \cdots, n. \tag{5.11}$$

(c) Drop the first M 'burn in' observations to make sure that the starting values have an insignificant effect. And then construct the kernel estimator $\hat{m}^*$ from the bootstrap series $\{x_1^*, \cdots, x_n^*\}$ as in (5.2).

(d) Re-define the last p pseudo values $X_n^* = x_n, \cdots, X_{n-p+1}^* = x_{n-p+1}$, i.e., $y_n^* = y_n$ where $y_i^* = (x_i^*, x_{i-1}^*, \cdots, x_{t-p+1})'$. Then, compute the bootstrap root replicate as $X_{n+1}^* - \hat{X}_{n+1}^*$ where $\hat{X}_{n+1}^* = \hat{m}^*(y_n^*) = \hat{m}^*(y_n)$; recall that $\hat{m}^*$ uses bandwidth h as the original estimator $\hat{m}$. Also let

$$X_{n+1}^* = \hat{m}_g(y_n^*) + \hat{\sigma}_g(y_n^*)\epsilon_{n+1}^* = \hat{m}_g(y_n) + \hat{\sigma}_g(y_n)\epsilon_{n+1}^*.$$

(4) Steps (a)-(d) in the above are repeated B times, and the B bootstrap root replicates are collected in the form of an empirical distribution whose α-quantile is denoted $q(\alpha)$.

(5) Then, a $(1-\alpha)100\%$ *equal-tailed predictive interval for* X_{n+1} *is given by*

$$[\hat{m}(y_n) + q(\alpha/2), \hat{m}(y_n) + q(1-\alpha/2)] \tag{5.12}$$

The corresponding Bootstrap Algorithm with Predictive Residuals is as follows.

5.2.2. Forward Bootstrap Algorithm with Predictive Residuals

Algorithm 5.5.

(1) same as step (1) of Algorithm 5.4.
(2) Use the delete-x_t *dataset to compute the delete-one kernel estimators* $\hat{m}^{(t)}$ *by (5.4) and* $\hat{\sigma}^{(t)}$ *by*

$$\hat{\sigma}^{(t)}(y) = \frac{\sum_{i=p+1, i\neq t}^{n} K(\frac{\|y-y_{i-1}\|}{h})(x_i - \hat{m}^{(t)}(y_{i-1}))^2}{\sum_{i=p+1, i\neq t}^{n} K(\frac{\|y-y_{i-1}\|}{h})}. \tag{5.13}$$

Then, calculate the predictive residuals:

$$\hat{\epsilon}_t^{(t)} = \frac{x_t - \hat{m}^{(t)}(y_{t-1})}{\hat{\sigma}^{(t)}(y_{t-1})} \text{ for } t = p+1, \cdots, n. \tag{5.14}$$

(3)-(5) Replace $\hat{\epsilon}_t$ *by* $\hat{\epsilon}_t^{(t)}$ *in Algorithm 5.4; the remaining steps are the same.*

Remark 5.3. As in all nonparametric methods, $\hat{m}(y)$ and $\hat{\sigma}(y)$ will only be accurate when y is in a dense area of the x_t vs. y_{t-1} scatterplot so that local averaging is effective. Computing these estimates in a sparse area of the scatterplot leads to inaccuracies in general but the problem is compounded when computing predictive residuals. To see why, note that $\sum_{i=p+1, i\neq t}^{n} K(\frac{\|y_{t-1}-y_{i-1}\|}{h})$ in the denominator of (5.4) and (5.13) could be zero or close to zero if y_{t-1} is far from $\{y_i, i \neq t-1\}$, and the estimates $\hat{m}^{(t)}$ and $\hat{\sigma}^{(t)}$ may be ill-defined. Consider the case where $K(\frac{\|y_{t-1}-y_{j-1}\|}{h}) \neq 0$ for some j, and $K(\frac{\|y_{t-1}-y_{i-1}\|}{h})$ is close to 0 for all $i \neq t, j$; then $\hat{m}^{(t)}(y_{t-1}) \approx x_j$ and $\hat{\sigma}^{(t)}(y_{t-1}) \approx x_j - \hat{m}^{(t)}(y_{j-1}) \approx 0$. In this case, the denominator of (5.14) is close to while $\hat{m}^{(t)}(y_{j-1})$ is close to x_j, and $\hat{\epsilon}_t^{(t)}$ will be extremely large. To avoid these situations, we delete all $\hat{\epsilon}_t^{(t)}$ that is not defined or extremely large from the pool of predictive residuals.

5.2.3. Asymptotic Properties

For simplicity, we again focus on a stationary and geometrically ergodic process of order 1. Then the nonparametric autoregression model with heteroscedastic innovations takes the simple form,

$$X_t = m(X_{t-1}) + \sigma(X_{t-1})\epsilon_t \tag{5.15}$$

where the innovations $\{\epsilon_t\}$ are i.i.d. (0,1) and satisfy causality condition (1.3).

Theorem 5.4 (Franke, Kreiss and Mammen(2002) [18])**.** *Consider a dataset* $X_1 = x_1, \ldots, X_n = x_n$ *from model (5.15), and choose the bandwidths* h, g *to satisfy (5.9). Under assumptions (AB1)-(AB12) of Franke, Kreiss and Mammen(2002) [18] we have:*

$$d_0(\mathcal{L}^*(\sqrt{nh}\{\hat{m}_g(x_n) - \hat{m}^*(x_n)\}), (\mathcal{L}(\sqrt{nh}\{m(x_n) - \hat{m}(x_n)\})) \xrightarrow{P} 0,$$

$$d_0(\mathcal{L}^*(\sqrt{nh}\{\hat{\sigma}_g(x_n) - \hat{\sigma}^*(x_n)\}), (\mathcal{L}(\sqrt{nh}\{\sigma(x_n) - \hat{\sigma}(x_n)\})) \xrightarrow{P} 0,$$

and

$$d_0(F_\epsilon, \hat{F}_n) \xrightarrow{P} 0 \text{ as } n \to \infty$$

where d_0 is Kolmogorov distance, and $\hat{F}_n$ is the empirical distribution of fitted residuals centered to mean zero.

As before, we also have:

Lemma 5.5. *Under the assumptions of Theorem 5.4, $\hat{\epsilon}_t - \hat{\epsilon}_t^{(t)} = O_p(\frac{1}{n})$ as $n \to \infty$.*

From Theorem 5.4 and Lemma 5.5, the following corollary is immediate.

Corollary 5.6. *Under the assumptions of Theorem 5.4, , the prediction interval (5.12) is asymptotically pertinent, and the same is true for its analog using predictive residuals, i.e., the interval of Algorithm 5.5.*

Note that we can also define intervals based on studentized predictive roots here as well; however, as mentioned in Remark 2.4, studentization offers little advantage on top of using studentized residuals.

Remark 5.4. To elaborate on the last point, consider the case of fitted residuals; the case of predictive residuals is similar. The predictive root is

$$X_{n+1} - \hat{X}_{n+1} = m(x_n) - \hat{m}(x_n) + \sigma(x_n)\epsilon_{n+1}.$$

As in Remark 2.4, in order to get a simple estimate of the variance of $X_{n+1} - \hat{X}_{n+1}$ it is convenient to omit the term $m(x_n) - \hat{m}(x_n)$ which is asymptotically negligible. This leads to the approximation

$$X_{n+1} - \hat{X}_{n+1} \simeq \sigma(x_n)\epsilon_{n+1},$$

and the corresponding prediction variance estimate $\hat{V}_n^2 = \hat{\sigma}^2(x_n)$. In the bootstrap world, we have

$$X^*_{n+1} - \hat{X}^*_{n+1} = \hat{m}_g(x_n) - \hat{m}^*(x_n) + \hat{\sigma}_g(x_n)\epsilon^*_{n+1} \simeq \hat{\sigma}_g(x_n)\epsilon^*_{n+1}.$$

Thus, the interval based on the predictive root method would be based on the approximation

$$P(X_{n+1} - \hat{X}_{n+1} \le a) \simeq P^*(X^*_{n+1} - \hat{X}^*_{n+1} \le a)$$

for (almost) all a, which is approximately equivalent to

$$P(\sigma(x_n)\epsilon_{n+1} \le a) \simeq P^*(\hat{\sigma}_g(x_n)\epsilon^*_{n+1} \le a). \tag{5.16}$$

By contrast, the studentized predictive roots, real-world and bootstrap, are given by

$$\frac{X_{n+1} - \hat{X}_{n+1}}{\hat{V}_n} = \epsilon_{n+1} \text{ and } \frac{X^*_{n+1} - \hat{X}^*_{n+1}}{\hat{V}^*_n} = \epsilon^*_{n+1}.$$

So, the interval based on the studentized predictive root method would be based on the approximation

$$P(\frac{X_{n+1} - \hat{X}_{n+1}}{\hat{V}_n} \le a) \simeq P^*(\frac{X^*_{n+1} - \hat{X}^*_{n+1}}{\hat{V}^*_n} \le a)$$

which is equivalent to

$$P(\frac{\sigma(x_n)}{\hat{\sigma}(x_n)}\epsilon_{n+1} \leq a) \simeq P^*(\frac{\hat{\sigma}_g(x_n)}{\hat{\sigma}^*(x_n)}\epsilon^*_{n+1} \leq a). \tag{5.17}$$

In the above, the error in the estimates $\hat{\sigma}(x_n)$ and $\hat{\sigma}^*(x_n)$ is $O_p(1/\sqrt{hn}) = O_p(n^{-2/5})$ while the error in the estimate $\hat{\sigma}_g(x_n)$ is $O_p(g^2)$ which is of bigger order because of the suboptimal smoothing employed in constructing $\hat{\sigma}_g(\cdot)$. It is this bigger term of order $O_p(g^2)$ that determines the accuracy of approximation (5.17) making it approximately equivalent to approximation (5.16). Hence, the studentized root method does not promise to offer an advantage here. In fact, as the Monte Carlo simulations of Section 5.3 show, the studentized root intervals FSf and FSp have identical performance in practice as their unstudentized counterparts Ff and Fp.

5.3. Monte Carlo Studies

5.3.1. Simulation Results for Nonparametric Autoregression with i.i.d. Errors

We present Monte Carlo simulations to assess the performance of the bootstrap methods with fitted and predictive residuals through average coverage level (CVR) and length (LEN) as defined in Section 3.7. To evaluate the performance of the bootstrap methods for nonparametric autoregression with i.i.d. innovations, we use the following models, all of order $p = 1$.

- Model 1: $X_{t+1} = \sin(X_t) + \epsilon_{t+1}$
- Model 2: $X_{t+1} = 0.8\log(3X_t^2 + 1) + \epsilon_{t+1}$
- Model 3: $X_{t+1} = -0.5\exp(-50X_t^2)X_t + \epsilon_{t+1}$

Normal innovations	nominal coverage 95%				nominal coverage 90%		
$n = 100$	CVR	LEN	st.err		CVR	LEN	st.err
Ff	0.927	3.860	0.393		0.873	3.255	0.310
Fp	0.943	4.099	0.402		0.894	3.456	0.317
FSf	0.938	4.020	0.403		0.887	3.387	0.314
FSp	0.939	4.030	0.405		0.888	3.390	0.313
$n = 200$							
Ff	0.938	3.868	0.272		0.886	3.263	0.219
Fp	0.948	4.012	0.283		0.899	3.385	0.231
FSf	0.945	3.966	0.280		0.894	3.339	0.222
FSp	0.945	3.970	0.282		0.895	3.344	0.228
Laplace innovations	nominal coverage 95%				nominal coverage 90%		
$n = 100$							
Ff	0.933	4.161	0.648		0.879	3.218	0.452
Fp	0.944	4.430	0.658		0.896	3.445	0.470
FSf	0.942	4.388	0.675		0.892	3.386	0.466
FSp	0.942	4.364	0.641		0.892	3.386	0.465
$n = 200$							
Ff	0.937	4.122	0.460		0.885	3.198	0.329
Fp	0.943	4.275	0.455		0.895	3.341	0.341
FSf	0.943	4.250	0.473		0.893	3.293	0.333
FSp	0.941	4.234	0.447		0.893	3.299	0.327

Table 17: Nonparametric autoregression with i.i.d innovations-model 1

As before, $\{\epsilon_t\}$ are i.i.d. N(0,1) or Laplace rescaled to unit variance; the kernel $K(\cdot)$ was the normal density with bandwidth h chosen by cross validation. Note that the smaller sample size considered here was $n = 100$ due to the reduced rate of convergence in nonparametric estimation.

Tables 17, 18 and 19 summarize the simulation results for each of the four Forward methods (Ff, Fp, FSf, and FSp) under each model. The conclusions are similar as in the parametric cases, namely that Fp, FSf, and FSp are all better than Ff. As before, using predictive residuals is important in the unstudentized case but not so important in the studentized case as FSf and FSp have very similar performance.

Normal innovations	nominal coverage 95%				nominal coverage 90%	
$n = 100$	CVR	LEN	st.err	CVR	LEN	st.err
Ff	0.928	3.870	0.388	0.875	3.260	0.308
Fp	0.946	4.143	0.393	0.900	3.495	0.305
FSf	0.940	4.051	0.393	0.890	3.407	0.309
FSp	0.941	4.049	0.387	0.891	3.407	0. 306
$n = 200$						
Ff	0.936	3.877	0.297	0.883	3.270	0.238
Fp	0.948	4.053	0.295	0.899	3.418	0.238
FSf	0.943	3.992	0.302	0.893	3.359	0.241
FSp	0.944	3.999	0.292	0.894	3.364	0.234
Laplace innovations	nominal coverage 95%				nominal coverage 90%	
$n = 100$	CVR	LEN	st.err	CVR	LEN	st.err
Ff	0.932	4.175	0.639	0.880	3.237	0.449
Fp	0.944	4.458	0.649	0.898	3.472	0.464
FSf	0.942	4.414	0.667	0.893	3.415	0.466
FSp	0.941	4.380	0.631	0.893	3.406	0.452
$n = 200$						
Ff	0.937	4.110	0.455	0.884	3.192	0.322
Fp	0.944	4.284	0.452	0.895	3.347	0.336
FSf	0.942	4.247	0.467	0.892	3.296	0.328
FSp	0.942	4.234	0.453	0.892	3.297	0.327

Table 18: Nonparametric autoregression with i.i.d innovations-model 2

Normal innovations	nominal coverage 95%				nominal coverage 90%		
$n = 100$	CVR	LEN	st.err		CVR	LEN	st.err
Ff	0.932	3.832	0.369		0.881	3.236	0.282
Fp	0.940	3.946	0.380		0.892	3.332	0.296
FSf	0.940	3.945	0.372		0.891	3.323	0.280
FSp	0.940	3.944	0.376		0.891	3.322	0.288
$n = 200$							
Ff	0.941	3.862	0.265		0.890	3.257	0.210
Fp	0.945	3.926	0.274		0.896	3.312	0.214
FSf	0.944	3.921	0.263		0.895	3.302	0.206
FSp	0.945	3.929	0.271		0.896	3.309	0.210
Laplace innovations	nominal coverage 95%				nominal coverage 90%		
$n = 100$	CVR	LEN	st.err		CVR	LEN	st.err
Ff	0.934	4.192	0.636		0.882	3.228	0.447
Fp	0.938	4.295	0.640		0.890	3.322	0.455
FSf	0.941	4.361	0.647		0.891	3.350	0.454
FSp	0.940	4.317	0.622		0.891	3.335	0.443
$n = 200$							
Ff	0.939	4.143	0.478		0.889	3.205	0.327
Fp	0.941	4.190	0.467		0.893	3.269	0.337
FSf	0.943	4.226	0.471		0.894	3.268	0.324
FSp	0.942	4.206	0.460		0.894	3.273	0.329

Table 19: Nonparametric autoregression with i.i.d. innovations-model 3

5.3.2. Simulation Results for Nonparametric Autoregression with Heteroscedastic Errors

To evaluate the performance of the bootstrap methods for nonparametric autoregression with heteroscedastic innovations, we employed the following two models:

- Model 4: $X_t = \sin(X_{t-1}) + \sqrt{0.5 + 0.25X_{t-1}^2}\epsilon_t$
- Model 5: $X_{t+1} = 0.75X_t + 0.15X_t\epsilon_{t+1} + \epsilon_{t+1}$.

Table 20 and 21 summarize the simulation results of model 4 and model 5 using an over-smoothed resampling bandwidth, i.e., letting $g = 2h$ where h is chosen by cross validation as in the previous subsection. Doubling the original bandwidth h is a simple rule-of-thumb used in previous work in nonparametric regression.

The main points from the simulation are as follows:

- As alluded to at the end of Section 5.2.3, the studentized root intervals FSf and FSp have identical performance as their unstudentized counterparts Ff and Fp.
- As in the case of nonparametric regression with heteroscedasticity treated in Politis(2013)[33], resampling predictive residuals gives improved coverage levels as compared to using the fitted residuals.
- It is apparent that the coverages are not as accurate as in the previsously considered cases of time series without heteroscedasticity of nonparametric form. Still the oversmoothing trick seems to be a big part of rendering the CVRs associated with the Fp (or FSp) method reasonable.

To confirm the last point above, we repeat the simulation without an an over-smoothed resampling bandwidth, i.e., letting $g = h$; we omit the FSf and FSp entries as they are indistinguishable from

their Ff and Fp counterparts. Tables 22 and 23 show the results that are characterized by extreme undercoverage that is unacceptable. Using an over-smoothed resampling bandwidth appears to be a *sine qua non* in the presence of conditional heteroscedasticity whose functional form is unknown.

$g = 2h$	nominal coverage 95%				nominal coverage 90%		
normal innovations	CVR	LEN	st.err		CVR	LEN	st.err
$n = 100$							
Ff	0.894	3.015	0.926		0.843	2.566	0.783
Fp	0.922	3.318	1.003		0.868	2.744	0.826
FSf	0.894	3.018	0.934		0.843	2.569	0.790
FSp	0.923	3.337	1.017		0.869	2.761	0.839
$n = 200$							
Ff	0.903	2.903	0.774		0.848	2.537	0.647
Fp	0.921	3.164	0.789		0.863	2.636	0.654
FSf	0.903	2.986	0.779		0.847	2.534	0.652
FSp	0.921	3.168	0.796		0.863	2.638	0.657
Laplace innovations	CVR	LEN	st.err		CVR	LEN	st.err
$n = 100$							
Ff	0.895	3.197	1.270		0.843	2.521	0.909
Fp	0.921	3.662	1.515		0.866	2.740	0.967
FSf	0.894	3.200	1.300		0.843	2.523	0.930
FSp	0.922	3.691	1.553		0.866	2.762	0.989
$n = 200$							
Ff	0.905	3.028	0.955		0.851	2.395	0.747
Fp	0.921	3.285	1.029		0.864	2.514	0.776
FSf	0.904	3.026	0.972		0.850	2.392	0.757
FSp	0.921	3.294	1.041		0.864	2.520	0.783

Table 20: Heteroscedastic model 4 with $g = 2h$

$g = 2h$	nominal coverage 95%				nominal coverage 90%		
normal innovations	CVR	LEN	st.err		CVR	LEN	st.err
$n = 100$							
Ff	0.931	4.063	1.453		0.888	3.465	1.228
Fp	0.950	4.672	1.621		0.908	3.704	1.321
FSf	0.930	4.054	1.462		0.887	3.457	1.232
FSp	0.950	4.481	1.639		0.908	3.711	1.332
$n = 200$							
Ff	0.941	4.034	1.322		0.898	3.427	1.113
Fp	0.952	4.262	1.404		0.909	3.556	1.173
FSf	0.940	4.029	1.326		0.898	3.422	1.117
FSp	0.953	4.262	1.411		0.910	3.557	1.178
Laplace innovations	CVR	LEN	st.err		CVR	LEN	st.err
$n = 100$							
Ff	0.923	4.414	2.283		0.881	3.512	1.811
Fp	0.943	5.081	2.865		0.899	3.827	2.001
FSf	0.922	4.384	2.256		0.880	3.490	1.784
FSp	0.943	5.103	2.951		0.900	3.844	2.076
$n = 200$							
Ff	0.933	4.222	1.484		0.888	3.338	1.151
Fp	0.945	4.594	1.656		0.900	3.525	1.249
FSf	0.932	4.204	1.475		0.887	3.323	1.149
FSp	0.945	4.597	1.661		0.900	3.530	1.255

Table 21: Heteroscedastic model 5 with $g = 2h$

heteroscedastic	nominal coverage 95%				nominal coverage 90%		
normal innovations	CVR	LEN	st.err		CVR	LEN	st.err
$n = 100$							
Ff	0.838	2.630	1.093		0.776	2.238	0.930
Fp	0.871	2.889	1.193		0.804	2.391	0.989
$n = 200$							
Ff	0.864	2.733	1.007		0.801	2.317	0.845
Fp	0.885	2.888	1.024		0.817	2.405	0.846
Laplace innovations	CVR	LEN	st.err		CVR	LEN	st.err
$n = 100$							
Ff	0.847	2.719	1.349		0.785	2.147	1.000
Fp	0.878	3.094	1.571		0.812	2.330	1.063
$n = 200$							
Ff	0.871	2.687	1.098		0.810	2.128	0.873
Fp	0.890	2.914	1.207		0.824	2.233	0.921

Table 22: Heteroscedastic model 4 without over-smoothed bandwidth

heteroscedastic	nominal coverage 95%				nominal coverage 90%		
normal innovations	CVR	LEN	st.err		CVR	LEN	st.err
$n = 100$							
Ff	0.884	3.619	1.575		0.831	3.079	1.317
Fp	0.911	3.969	1.732		0.856	3.290	1.414
$n = 200$							
Ff	0.904	3.706	1.490		0.854	3.146	1.255
Fp	0.919	3.916	1.602		0.866	3.264	1.323
Laplace innovations	CVR	LEN	st.err		CVR	LEN	st.err
$n = 100$							
Ff	0.880	3.886	2.351		0.829	3.085	1.841
Fp	0.906	4.452	3.852		0.850	3.354	2.066
$n = 200$							
Ff	0.900	3.803	1.787		0.846	3.005	1.405
Fp	0.915	4.129	1.962		0.860	3.172	1.485

Table 23: Heteroscedastic model 5 without over-smoothed bandwidth

6. Conclusions

In the paper at hand, we present a comprehensive approach for the construction of prediction intervals in AR models. The construction is based on predictive roots, studentized or not, and notions of validity were defined and discussed. In addition, the usage of predictive residuals in model-based resampling is proposed, and shown to improve coverage levels in finite samples.

There is a lot of previous work in the special case of linear AR models but the literature has been lacking a unifying methodology. We survey the existing approaches and bring them under two umbrellas: Backward vs. Forward bootstrap. The Backward bootstrap has been the most well-known in the literature; we develop further the idea of the Forward bootstrap for prediction intervals, and add the necessary steps needed for it to achieve conditional validity.

To date, little seems to be known concerning prediction intervals for nonlinear and/or nonparametric autoregressions. We show that the Forward bootstrap can be equally applied to such models with some care as regards the particulars; for example, bandwidth considerations are important in the nonparametric case. All in all, it is apparent that the Forward bootstrap with fitted or predictive residuals may serve as the unifying principle for prediction intervals across all types of AR models, linear, nonlinear or nonparametric.

Appendix A: Technical proofs.

Proof of Lemma 3.4.

Proof. Let $M = \begin{bmatrix} 1 & X_{n-1} & \cdots & X_{n-p} \\ 1 & X_{n-2} & \cdots & X_{n-p-1} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & X_p & \cdots & X_1 \end{bmatrix}$, $N = \begin{pmatrix} X_n \\ X_{n-1} \\ \vdots \\ X_{p+1} \end{pmatrix}$;
also let M_t be the matrix after deleting the row $Z_t' = (1, X_{t-1}, \cdots, X_{t-p})$ in M, and N_t be the vector after deleting X_t in N. Then

$$\begin{aligned} \hat{\underline{\phi}} &= (M'M)^{-1}M'N \\ &= [(M_t'M_t)^{-1}Z_tZ_t' + I_{p+1}]^{-1}(M_t'M_t)^{-1}(M_t'N_t + Z_tX_t) \\ &= [I_{p+1}(1 + O_p(\frac{1}{n}))]^{-1}(M_t'M_t)^{-1}M_t'N_t(1 + O_p(\frac{1}{n})) \\ &= \hat{\underline{\phi}}^{(t)} + O_p(\frac{1}{n}) \end{aligned}$$

Here $X_t = O_p(1)$, Z_tZ_t' is a $(p+1) \times (p+1)$ matrix whose entries are of $O_p(1)$ while $M_t'M_t$ is a $(p+1) \times (p+1)$ matrix whose entries are the summation of n bounded in probability terms; hence, $Z_tZ_t' = M_t'M_t \cdot O_p(\frac{1}{n})$. Similarly $Z_tX_t = M_t'N_t \cdot O_p(\frac{1}{n})$. Thus, $\hat{\epsilon}_t - \epsilon_t^{(t)} = Z_t'(\hat{\underline{\phi}}^{(t)} - \hat{\underline{\phi}}) = O_p(\frac{1}{n})$. □

Proof of Lemma 5.2.

Proof.

$$\begin{aligned} \hat{m}(X_{t-1}) &= \frac{\sum_{i=1}^{n-1} K(\frac{X_{t-1}-X_i}{h})X_{i+1}}{\sum_{i=1}^{n-1} K(\frac{X_{t-1}-X_i}{h})} \\ \hat{m}^{(t)}(X_{t-1}) &= \frac{\sum_{i=1}^{n-1} K(\frac{X_{t-1}-X_i}{h})X_{i+1} - K(\frac{X_{t-1}-X_{t-1}}{h})X_t}{\sum_{i=1}^{n-1} K(\frac{X_{t-1}-X_i}{h}) - K(\frac{X_{t-1}-X_{t-1}}{h})} \\ &= \frac{\sum_{i=1}^{n-1} K(\frac{X_{t-1}-X_i}{h})X_{i+1} - K(0)X_t}{\sum_{i=1}^{n-1} K(\frac{X_{t-1}-X_i}{h}) - K(0)} \end{aligned}$$

let $a_n = \sum_{i=1}^{n-1} K(\frac{X_{t-1}-X_i}{h})X_{i+1}$, $\delta_a = K(0)X_t$; $b_n = \sum_{i=1}^{n-1} K(\frac{X_{t-1}-X_i}{h})$, $\delta_b = K(0)$. then

$$\begin{aligned} \hat{m}^{(t)}(X_{t-1}) &= \frac{a_n - \delta_a}{b_n - \delta_b} \\ &= \frac{a_n}{b_n}(\frac{1 - \delta_a/a_n}{1 - \delta_b/b_n}) \\ &= \frac{a_n}{b_n}(1 - \delta_a/a_n)(1 + \delta_b/b_n + (\delta_b/b_n)^2 + o(\delta_b/b_n)^2) \\ &= \frac{a_n}{b_n} + O_p(1/n) \\ &= \hat{m}(X_{t-1}) + O_p(1/n) \end{aligned}$$

It follows that $\hat{\epsilon}_t - \hat{\epsilon}_t^{(t)} = X_t - \hat{m}(X_{t-1}) - (X_t - \hat{m}^{(t)}(X_{t-1})) = O_p(1/n)$ □

Proof of Lemma 5.5.

Proof. We can prove in a similar way as in Lemma 5.2 that $\hat{m}(x) - \hat{m}^{(t)}(x) = O_p(1/n)$ and $\hat{\sigma}(x) - \hat{\sigma}^{(t)}(x) = O_p(1/n)$. Then,

$$\begin{aligned}\hat{\epsilon}_t - \hat{\epsilon}_t^{(t)} &= \frac{X_t - \hat{m}(X_{t-1})}{\hat{\sigma}(X_{t-1})} - \frac{X_t - \hat{m}^{(t)}(X_{t-1})}{\hat{\sigma}^{(t)}(X_{t-1})} \\ &= \frac{X_t - \hat{m}(X_{t-1}) - (X_t - \hat{m}^{(t)}(X_{t-1}))}{\hat{\sigma}(X_{t-1})}(1 + O_p(1/n)) \\ &= O_p(1/n)(1 + O_p(1/n)) = O_p(1/n).\end{aligned}$$

□

References

[1] Andrés M Alonso, Daniel Peña, and Juan Romo. Forecasting time series with sieve bootstrap. *Journal of Statistical Planning and Inference*, 100(1):1–11, 2002.

[2] R. Beran. Bootstrap methods in statistics. *Jahresber. Deutsch. Math.-Verein.*, 86:1430, 1984.

[3] Peter Bickel and D.A. Freedman. Some asymptotic theory for the bootstrap. *The Annals of Statistics*, 9:1196–1217, 1981.

[4] Mixail V. Boldin. The estimate of the distribution of noise in autoregressive scheme. *Teoriya Veroyatnostei i ee Primeneniya*, 27(4):805–810, 1982.

[5] Tim Bollerslev and Jeffrey M Wooldridge. Quasi-maximum likelihood estimation and inference in dynamic models with time-varying covariances. *Econometric reviews*, 11(2):143–172, 1992.

[6] Arup Bose. Edgeworth correction by bootstrap in autoregressions. *The Annals of Statistics*, 16:1345–1741, 1988.

[7] Arup Bose and Kanchan Mukherjee. Estimating the arch parameters by solving linear equations. *Journal of Time Series Analysis*, 24(2):127–136, 2003.

[8] Arup Bose and Kanchan Mukherjee. Bootstrapping a weighted linear estimator of the arch parameters. *Journal of Time Series Analysis*, 30(3):315–331, 2009.

[9] George EP Box and Gwilym M Jenkins. *Time series analysis, control, and forecasting.* San Francisco, CA: Holden Day, 1976.

[10] F. Jay Breidt, Richard A Davis, and William Dunsmuir. Improved bootstrap prediction intervals for autoregressions. *Journal of Time Series Analysis*, 16(2):177–200, 1995.

[11] R. Cao, M. Febrero-Bande, W. González-Manteiga, J.M. Prada-Sánchez, and I. Garcfa-Jurado. Saving computer time in constructing consistent bootstrap prediction intervals for autoregressive processes. *Communications in Statistics-Simulation and Computation*, 26(3):961–978, 1997.

[12] R.J. Carroll and D. Ruppert. *Transformations and Weighting in Regression.* New York: Chapman and Hall, 1988.

[13] K.-S. Chan. Consistency and limiting distribution of the least squares estimator of a threshold autoregressive model. *The Annals of Statistics*, 21:520–533, 1993.

[14] Snigdhansu Chatterjee and Arup Bose. Generalized bootstrap for estimating equations. *Annals of statistics*, pages 414–436, 2005.

[15] B. Efron and R. Tibshirani. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical Science*, 1:1–154, 1986.

[16] B. Efron and R. Tibshirani. *An Introduction to the Bootstrap.* Chapman and Hall/CRC Press, 1994.

[17] C. Francq and J.-M. Zakoian. *GARCH Models: Structure, Statistical Inference and Financial Applications*. New York: Wiley, 2010.
[18] Jürgen Franke, Jens-Peter Kreiss, and Enno Mammen. Bootstrap of kernel smoothing in nonlinear time series. *Bernoulli*, 8(1):1–37, 2002.
[19] David Freedman. On bootstrapping two-stage least-squares estimates in stationary linear models. *The Annals of Statistics*, 12:827–842, 1984.
[20] S. Geisser. *Predictive Inference: An Introduction*. New York: Chapman and Hall, 1993.
[21] Matteo Grigoletto. Bootstrap prediction intervals for autoregressions: some alternatives. *International Journal of Forecasting*, 14(4):447–456, 1998.
[22] P. Hall. *The Bootstrap and Edgeworth Expansion*. New York: Springer, 1992.
[23] W Hardle and JS Marron. Bootstrap simultaneous error bars for nonparametric regression. *The Annals of Statistics*, pages 778–796, 1991.
[24] P. Kokoszka and D.N. Politis. Nonlinearity of arch and stochastic volatility models and bartlett's formula. *Probability and Mathematical Statistics*, 31:47–59, 2011.
[25] D. Li and S. Ling. On the least squares estimation of multiple-regime threshold autoregressive models. *Journal of Econometrics*, 167:240–253, 2012.
[26] Guido Masarotto. Bootstrap prediction intervals for autoregressions. *International Journal of Forecasting*, 6(2):229–239, 1990.
[27] J.A. Miguel and Pilar Olave. Bootstrapping forecast intervals in arch models. *Test*, 8(2):345–364, 1999.
[28] Purna Mukhopadhyay and VA Samaranayake. Prediction intervals for time series: a modified sieve bootstrap approach. *Communications in StatisticsSimulation and Computation®*, 39(3):517–538, 2010.
[29] Pilar Olave Robio. Forecast intervals in arch models: bootstrap versus parametric methods. *Applied Economics Letters*, 6:323–327, 1999.
[30] D.J. Olive. Prediction intervals for regression models. *Comput. Statist. and Data Anal.*, 51:3115–3122, 2007.
[31] Lorenzo Pascual, Juan Romo, and Esther Ruiz. Bootstrap predictive inference for arima processes. *Journal of Time Series Analysis*, 25(4):449–465, 2004.
[32] J.K. Patel. Prediction intervals: a review. *Comm. Statist. Theory Meth.*, 18:2393–2465, 1989.
[33] Dimitris N. Politis. Model-free model-fitting and predictive distributions (with discussion). *Test*, 22(2):183–250, 2013.
[34] D.N. Politis, J.P. Romano, and M. Wolf. *Subsampling*. New York: Springer, 1999.
[35] Jonathan J. Reeves. Bootstrap prediction intervals for arch models. *International Journal of Forecasting*, 21:237–248, 2000.
[36] R.L. Schmoyer. Asymptotically valid prediction intervals for linear models. *The Annals of Statistics*, 34:399–408, 1992.
[37] R.A. Stine. Bootstrap prediction intervals for regression. *J. Amer. Statist. Assoc.*, 80:1026–1031, 1985.
[38] Robert A Stine. Estimating properties of autoregressive forecasts. *Journal of the American statistical association*, 82(400):1072–1078, 1987.
[39] Lori A Thombs and William R Schucany. Bootstrap prediction intervals for autoregression. *Journal of the American Statistical Association*, 85(410):486–492, 1990.
[40] H. Tong. Threshold models in time series analysis–30 years on. *Statistics and its Interface*, 4:107–118, 2011.
[41] Andrew A. Weiss. Asymptotic theory for arch models: Estimation and testing. *Econometric Theory*, 2(1):pp. 107–131, 1986.