Online ARIMA Algorithms for Time Series Prediction

Chenghao Liu[1,2], Steven C.H. Hoi[2], Peilin Zhao[3], Jianling Sun[1]
[1]School of Computer Science and Technology, Zhejiang University, China
[2]School of Information Systems, Singapore Management University, Singapore
[3]Institute for Infocomm Research, A*STAR, Singapore
twinsken@zju.edu.cn, chhoi@smu.edu.sg, zhaop@i2r.a-star.edu.sg, sunjl@zju.edu.cn

Abstract

Autoregressive integrated moving average (ARIMA) is one of the most popular linear models for time series forecasting due to its nice statistical properties and great flexibility. However, its parameters are estimated in a batch manner and its noise terms are often assumed to be strictly bounded, which restricts its applications and makes it inefficient for handling large-scale real data. In this paper, we propose online learning algorithms for estimating ARIMA models under relaxed assumptions on the noise terms, which is suitable to a wider range of applications and enjoys high computational efficiency. The idea of our ARIMA method is to reformulate the ARIMA model into a task of full information online optimization (without random noise terms). As a consequence, we can online estimation of the parameters in an efficient and scalable way. Furthermore, we analyze regret bounds of the proposed algorithms, which guarantee that our online ARIMA model is provably as good as the best ARIMA model in hindsight. Finally, our encouraging experimental results further validate the effectiveness and robustness of our method.

Introduction

In the past decades, time series forecasting has played an important role in a wide range of domains including speech analysis (Rabiner and Schafer 2011), noise cancelation (Gao et al. 2010), and financial market analysis (Hamilton 1994; Brockwell and Davis 2009; Rojo-Álvarez et al. 2004; Granger and Newbold 2014; Nerlove, Grether, and Carvalho 2014; Tsay 2005; Li and Hoi 2015). Typically, time series models can collect past observations and uncover their underlying relationship. Among the existing time series models, a fundamental one is the autoregressive moving average (ARMA) model (Hamilton 1994), originated from the autoregressive model (AR) and the moving average model (MA). Theoretically, if there is no missing data (Weigend 1994) for a stationary time series, then this model can learn an identified underlying process to mimic observations for predicting signal in the future. In practice, ARMA can describe the behavior of a noisy linear dynamical system, and is able to represent several different types of time series, due to its flexible modeling capability.

Despite its great success, ARMA assumes the underlying model is linear, which hinders its applications to many challenging real-world time series. To solve this issue, the autoregressive integrated moving average (ARIMA) model has been proposed as an extension of ARMA, which can tackle nonstationary time series forecasting by differencing techniques. Specifically, differencing techniques can eliminate the influences of trend components of data before ARIMA model can be fitted when the observations present trend and heteroscedasticity. However, most of existing ARIMA models still suffer from many limitations. First of all, most of them rely on some strong assumptions with respect to the noise terms (such as i.i.d. assumption (Hamilton 1994), t-distribution (Damsleth and El-Shaarawi 1989);(Tiku et al. 2000)) and loss functions, while many real applications may not fully satisfy these assumptions, which makes such ARIMA models unsuitable to many scenarios. Second, existing algorithms for estimating parameters of ARIMA, such as least squares and maximum likelihood based methods (Hamilton 1994), require to access the entire dataset in advance, which violates the streaming characteristics of time series data and cannot deal with concept-drift issues. In addition, these batch approaches cannot cope with large-scale datasets due to memory-intensive bottleneck.

To solve these issues, we propose online learning algorithms to efficiently estimate parameters of ARIMA by utilizing its recursive formulation in an online learning setting. Our novel approach allows the noise to be arbitrarily or even adversarially generated, making it more general to handle a wider range of time series prediction tasks. Moreover, our online learning approach handles data observations arriving sequentially and updates the models simultaneously, which is more natural for many real-world applications. Finally, the memory cost of our algorithm is independent of the sample size, significantly more scalable to deal with real-time time series forecasting tasks in the era of big data (Shalev-Shwartz et al. 2011; Hoi, Wang, and Zhao 2014).

Our Contributions. We propose a novel online learning method to estimate the parameters of ARIMA models by reformulating it into a full information online optimization task (without random noise terms). Theoretically, we give the regret bounds which show that the solutions produced by our method asymptotically approaches the best ARIMA model in hindsight. Moreover, we show that a recent online

ARMA model (Anava et al. 2013) can be viewed as a special case of our online ARIMA, and our experimental result empirically validates that online ARIMA algorithms considerably outperform the existing online ARMA algorithms.

The rest of this paper is organized as follows. We first review related work, followed by introducing the problem setup of time series prediction. Then we present the proposed method, followed by theoretical analysis. After, we discuss empirical results, and finally conclude this work.

Related Work

Some of the earliest works on ARIMA consider the squared loss and assume the noise follows an i.i.d. random sequence (Hamilton 1994; George 1994; Brockwell and Davis 2009). This assumption allows the use of statistical properties as well as the well-known Box-Jenkins methodology (George 1994) in the model building process. Later, such assumption has been relaxed by using other assumptions such as t-distribution of noise (Damsleth and El-Shaarawi 1989; Tiku et al. 2000) for the squared loss. In addition, the bispectral analysis and the Pade approximation were also utilized to estimate non-Gaussian ARMA models in (Lii 1990; Huang and Shih 2003). Moreover, the ARCH model was proposed in (Engle 1982), which can remove the independence assumption and offer specific dependency model.

In literature, very few study has seriously investigated scalable algorithms for ARIMA models, although some simple methods were attempted. For example, the iterated least-squares approach was developed to consistently estimate autoregressive parameters (Tsay and Tiao 1984). Least squares and gradient algorithms are presented through estimating residuals, for which a convergence analysis is also given by using the martingale convergence theorem (Ding, Shi, and Chen 2006). Nevertheless, none of these has been formally formulated in a standard online learning setting.

The closest related work is the online ARMA model for time series prediction in (Anava et al. 2013). Our online ARIMA model differs from their study in several key aspects. First, unlike online ARMA model that assumes time series data is stationary, online ARIMA model relaxes such assumption and thus can deal with non-stationary time series forecasting with trend or heteroscedasticity more effectively. Second, the theoretical anaysis in (Anava et al. 2013) assumes a restricted constraint on the coefficients β, which is a sufficient condition for a stationary time series process but not necessary. By contrast, we remove such restricted assumption, and thus make our theoretical analysis results more general. Finally, we apply a different analysis method by exploring difference equation techniques (Hamilton 1994), and obtain a regret bound $\mathcal{O}(\log(Tq)\log T)$ that is better than their result $\mathcal{O}(q\log T\log T)$, where q is the number of coefficients for modeling the noise and T is the total of iterations.

Online ARIMA

In this section, we will mainly review the problem setup for time series prediction, and some time series models.

Time Series Modeling

A time series is defined as a sequence of quantitative observations at successive time. We assume time is a discrete variable, X_t denotes the observation at time t, and ϵ_t denotes the zero-mean random noise term at time t. The MA(q) (short for Moving Average) model considers the process: $X_t = \sum_{i=1}^{q} \beta_i \epsilon_{t-i} + \epsilon_t$, where β_i is a coefficient. Similar to $\mathrm{MA}(q)$ models, Autoregression model, denoted by AR(k), satisfies $X_t = \sum_{i=1}^{k} \alpha_i X_{t-i} + \epsilon_t$. In other words, it assumes each X_t is a noisy linear combination of the previous k observations. This is similar to traditional multiple regression model, but X_t is regressed on past values of X_t.

A more sophisticated model is the $\mathrm{ARMA}(k,q)$ (short for autoregressive moving average), which is a combination of $\mathrm{AR}(k)$ and $\mathrm{MA}(q)$ with a compact form and provides a flexible modeling framework. This model assumes that X_t is generated via the formula:

$$X_t = \sum_{i=1}^{q} \beta_i \epsilon_{t-i} + \sum_{i=1}^{k} \alpha_i X_{t-i} + \epsilon_t, \tag{1}$$

where again ϵ_t are zero-mean noise term. If we add some constraint to the weights of $\mathrm{AR}(k)$ part, it can guarantee a stationary process. A stationary and invertible $\mathrm{ARMA}(k,q)$ model may be represented either as an infinite AR model($\mathrm{AR}(\infty)$) or an infinite MA model($\mathrm{MA}(\infty)$). Compared with $\mathrm{AR}(\infty)$ and $\mathrm{MA}(\infty)$, $\mathrm{ARMA}(k,q)$ can generate stationary stochastic processes with only a finite number of parameters (Hamilton 1994).

ARIMA Model

Nevertheless, time series data are usually not realizations of a stationary process. For example, some of them may contain deterministic trends. An effective way to handle such strong serial correlations is to consider the differential method. For example, one can compute the first order differences of X_t by $\nabla X_t = X_t - X_{t-1}$ and the second order differences of X_t by $\nabla^2 X_t = \nabla X_t - \nabla X_{t-1}$.

If the sequence of $\nabla^d X_t$ satisfies an $\mathrm{ARMA}(k,q)$, we say that the sequence of X_t satisfies the $\mathrm{ARIMA}(k,d,q)$ (short for AutoRegressive Integrated Moving Average),

$$\nabla^d X_t = \sum_{i=1}^{q} \beta_i \epsilon_{t-i} + \sum_{i=1}^{k} \alpha_i \nabla^d X_{t-i} + \epsilon_t, \tag{2}$$

which are parameterized by three horizon terms k, d, q and weights vector $\alpha \in \mathbb{R}^k$ and $\beta \in \mathbb{R}^q$. Note that $\mathrm{ARMA}(k,q)$ is a special case of the $\mathrm{ARIMA}(k,d,q)$, where the differences order is zero.

Forecasting with $\mathrm{ARIMA}(k,d,q)$ is a reversion of differential process. Suppose time series sequence X_t satisfies $\mathrm{ARIMA}(k,d,q)$, we can predict the d-th order differential of observation at time $t+1$ as $\nabla^d \tilde{X}_{t+1}$ and then predict the observation at time $t+1$ as $\tilde{X}_t$:

$$\tilde{X}_t = \nabla^d \tilde{X}_t + \sum_{i=0}^{d-1} \nabla^i X_{t-1}. \tag{3}$$

Online ARIMA Algorithms

We follow a typical game-theoretic framework for online learning with ARIMA models, where an online player sequentially commits to a decision and then suffers from a loss which may be unknown to the decision maker ahead of time. It can be adversarial or even depend on the actions taken by the decision maker. In the online setting of ARIMA, we assume coefficient vectors (α, β) are fixed by the adversary. At time t, the adversary chooses the noise ϵ_t and then generates the resulting observation X_t based on Eq. 2 and Eq. 3. It is important to note that the true values of both (α, β) and ϵ_t are not disclosed to the learner at any time.

Consider an online ARIMA iteration at time t, the learner makes a prediction $\tilde{X}_t$, and then the true X_t is disclosed to the learner. As a result, the learner suffers a loss $\ell_t(X_t, \tilde{X}_t)$. More formally, we can define the loss function as follows:

$$f_t(\alpha,\beta)=\ell_t(X_t,\tilde{X}_t(\alpha,\beta))=\ell_t(X_t,(\nabla^d\tilde{X}_t+\sum_{i=0}^{d-1}\nabla^i X_{t-1}))$$

$$=\ell_t(X_t,(\sum_{i=1}^{q}\beta_i\epsilon_{t-i}+\sum_{i=1}^{k}\alpha_i\nabla^d X_{t-i}+\sum_{i=0}^{d-1}\nabla^i X_{t-1})).(4)$$

The goal of online ARIMA learning is to minimize the sum of losses over some number of rounds T. More formally, we can define the regret of the learner after T rounds as:

$$R_T=\sum_{t=1}^{T}\ell_t(X_t,\tilde{X}_t)-\min_{\alpha,\beta}\sum_{t=1}^{T}\ell_t(X_t,\tilde{X}_t(\alpha,\beta)). \quad (5)$$

Our goal is to devise an efficient algorithm that can guarantee the regret grows sublinearly as a function of T, i.e., $R_T \leq o(T)$, implying that the per-round regret of the learner will vanish as T increases.

Given the loss function defined in Eq. 4, one might consider to apply some existing online convex optimization techniques to estimate the coefficient vectors (α, β) for the online ARIMA learning task . However, this is not possible since the noise terms $\{\epsilon_t\}$ are unknown to the learner at any time of the online learning process. As a result, even (α, β) is given, we cannot perform a prediction due to the unknown noise terms. To tackle this challenge, we follow the idea of improper learning principle (Anava et al. 2013) to design a solution where the prediction does not come directly from the original ARIMA model, but from a modified ARIMA model (without the explicit noise terms) that approximates the original model.

Specifically, we propose to approximate the original $\text{ARIMA}(k, d, q)$ model with another $\text{ARIMA}(k+m, d, 0)$ model (without the noise terms), where $m \in \mathbb{N}$ is a properly chosen constant such that the new ARIMA model with an $(m+k)$-dimensional coefficient vector $\gamma \in \mathbb{R}^{m+k}$ is effective enough to approximate the original prediction:

$$\tilde{X}_t(\gamma^t)=\sum_{i=1}^{k+m}\gamma_i\nabla^d X_{t-i}+\sum_{i=0}^{d-1}\nabla^i X_{t-1}. \quad (6)$$

As a result, the loss function becomes

$$\begin{aligned}\ell_t^m(\gamma^t)&=\ell_t(X_t,\tilde{X}_t(\gamma^t))\\&=\ell_t(X_t,(\sum_{i=1}^{k+m}\gamma_i\nabla^d X_{t-i}+\sum_{i=0}^{d-1}\nabla^i X_{t-1})). \quad (7)\end{aligned}$$

The remaining issue is how to choose an appropriate value for parameter m, and what will be the regret with such approximation. We will quantify the result in Theorem 1 later. In the following, we focus on presenting two specific online ARIMA algorithms using two popular online convex optimization solvers (Bubeck 2011): Online Gradient Descent (ODG) method (Zinkevich 2003) and Online Newton Step (ONS) (Hazan, Agarwal, and Kale 2007).

ARIMA Online Newon Step (ARIMA-ONS). We first introduce a few notations. We denote by $\mathcal{K}$ the decision set of candidate $(m+k)$-dimensional coefficient vectors, i.e., $\mathcal{K} = \{\gamma \in \mathbb{R}^{m+k}, |\gamma_j| \leq 1, j = 1, \ldots, m\}$, and $D = 2c \cdot \sqrt{m+k}$ the diameter of $\mathcal{K}$. Further, we denote by G the upper bound of $\|\nabla \ell_t^m(\gamma)\|$ for all t and $\gamma \in \mathcal{K}$, which equals to $2c \cdot \sqrt{m+k}(X_{max})^2$ for the squared loss. Finally, we denote by λ the exp-concavity parameter of the loss functions $\{\ell_t^m\}_{t=1}^T$ which guarantees $e^{-\lambda \ell_t^m(\gamma)}$ is concave for all t. For specific case with the squared loss, $\lambda = \frac{1}{m+k}$.

Algorithm 1 shows the proposed ARIMA-ONS algorithm that iteratively optimizes the coefficient vectors γ^t of the online ARIMA model by applying the Online Newton Step solver (Hazan, Agarwal, and Kale 2007). Note that the projection is done by $\prod_{\mathcal{K}}^{A_t}(y) = \arg\min_{x\in\mathcal{K}}(y-x)^\top A_t(y-x)$ and the inverse of matrix A_t typically can be computed efficiently using the Sherman-Morrison formula. The regret bound of ARIMA-ONS will be analyzed later.

```
Algorithm 1 ARIMA-ONS($k, d, q$)
  Input: parameter $k$, $d$, $m$; learning rate $\eta$; initial $(m+k) \times (m+k)$ matrix $A_0$.
  Set $m = \log_{\lambda_{max}}((TLM_{max}q)^{-1})$.
  for t = 1 to $T-1$ do
      predict $\tilde{X}_t(\gamma^t) = \sum_{i=1}^{k+m}\gamma_i\nabla^d X_{t-i} + \sum_{i=0}^{d-1}\nabla^i X_{t-1}$;

      receive $X_t$ and incur loss $\ell_t^m(\gamma^t)$;
      Let $\nabla_t = \nabla\ell_t^m(\gamma^t)$, update $A_t \leftarrow A_{t-1} + \nabla_t\nabla_t^\top$;
      Set $\gamma^{t+1} \leftarrow \prod_{\mathcal{K}}^{A_t}(\gamma^t - \frac{1}{\eta}A_t^{-1}\nabla_t)$;
  end for
```

ARIMA Online Gradient Descent (ARIMA-OGD). We now apply a more general online convex optimization solver, Online Gradient Descent (Zinkevich 2003), which is applicable to any convex loss functions. Algorithm 2 presents the proposed ARIMA-OGD algorithm for optimizing the coefficient vector using the OGD algorithm. It has a worse regret bound compared as ARIMA-ONS but computationally more efficient. The projection $\prod_{\mathcal{K}}(y)$ refers to the Euclidean projection onto $\mathcal{K}$, i.e., $\prod_{\mathcal{K}}(y) = \arg\min_{x\in\mathcal{K}}\|y-x\|_2$.

Algorithm 2 ARIMA-OGD(k,d,q)

```
Input: parameter $k$, $d$, $q$; learning rate $\eta$.
Set $m = \log_{\lambda_{max}}((TLM_{max}q)^{-1})$.
for t = 1 to $T-1$ do
    predict $\tilde{X}_t(\gamma^t) = \sum_{i=1}^{k+m} \gamma_i \nabla^d X_{t-i} + \sum_{i=0}^{d-1} \nabla^i X_{t-1}$;

    receive $X_t$ and incur loss $\ell_t^m(\gamma^t)$;
    Let $\nabla_t = \nabla \ell_t^m(\gamma^t)$;
    Set $\gamma^{t+1} \leftarrow \prod_{\mathcal{K}}(\gamma^t - \frac{1}{\eta}\nabla_t)$;
end for
```

Main Theoretical Results

We now present our main theoretical results of analyzing the algorithms. We first discuss some necessary assumptions.

1. The coefficients β_i satisfy that a q-th order difference equation with coefficients $|\beta_1|, |\beta_2|, \ldots, |\beta_q|$ is a stationary process (Hamilton 1994); and
2. The noise terms are stochastically and independently generated, which satisfy $\mathbb{E}[|\epsilon_t|] < M_{max} < \infty$ and $\mathbb{E}[\ell_t(X_t, X_t - \epsilon_t)] < \infty$; and
3. The loss function ℓ_t is Lipshitz continuous for some Lipshitz constant $L > 0$; and
4. The coefficients α_i satisfy $|\alpha_i| < c$ for some $c \in \mathbb{R}$.

The following theorem presents our main theoretical result for the proposed ARIMA-ONS in Algorithm 1, which guarantees an $O\big((\log(q) + \log(T)) \log T\big)$ regret bound.

Theorem 1. *Let $k, q \geq 1$, and set $A_0 = \epsilon I_{m+k}, \epsilon = \frac{1}{\eta^2 D^2}, \eta = \frac{1}{2}\min\{4GD, \lambda\}$. Then, for any sequence $\{X_t\}_{t=1}^T$ that satisfies the above assumptions, the online sequence $\{\gamma^t\}_{t=1}^T$ generated by Algorithm 1 guarantees*

$$\sum_{t=1}^T \ell_t^m(\gamma^t) - \min_{\alpha,\beta} \sum_{t=1}^T \mathbb{E}[f_t(\alpha, \beta)] \tag{8}$$

$$= O((GD + \frac{1}{\lambda}) \log T) = O((\log(q) + \log(T)) \log T).$$

Proof. **Step 1:** Relying on the fact that the loss functions $\{\ell_t^m\}_{t=1}^T$ are λ-exp-concave, we can guarantee that

$$\sum_{t=1}^T \ell_t^m(\gamma^t) - \min_{\gamma} \sum_{t=1}^T \ell_t^m(\gamma)$$

$$= O((GD + \frac{1}{\lambda}) \log T) = O((m + k + \frac{1}{\lambda}) \log T),$$

using the ONS result in (Hazan, Agarwal, and Kale 2007).

Step 2: $\nabla^d X_t$ could be regarded as an $\mathrm{ARMA}(k, q)$, an $\mathrm{ARMA}(k, q)$ is equivalent to an $\mathrm{AR}(\infty)$. Thus, we recursively define $\nabla^d X_t^\infty(\alpha, \beta)$ by using the entire past history

$$\nabla^d X_t^\infty(\alpha,\beta) {=} \sum_{i=1}^k \alpha_i \nabla^d X_{t-i} + \sum_{i=1}^q \beta_i (\nabla^d X_{t-i} {-} \nabla^d X_{t-i}^\infty(\alpha, \beta)),$$

$$X_t^\infty(\alpha,\beta) {=} \nabla^d X_t^\infty(\alpha, \beta) + \sum_{i=1}^{d-1} \nabla^i X_{t-1}, \tag{9}$$

with initial condition $\nabla^d X_1^\infty(\alpha, \beta) = \nabla^d X_1$. We then denote by

$$f_t^\infty(\alpha, \beta) = \ell_t(X_t, X_t^\infty(\alpha, \beta)), \tag{10}$$

the loss suffered by the prediction $X_t^\infty(\alpha, \beta)$ at iteration t. It follows that $\nabla^d X_t^\infty(\alpha, \beta)$ is of the form $\nabla^d X_t^\infty(\alpha, \beta) = \sum_{i=1}^{t-1} c_i(\alpha, \beta) \nabla^d X_{t-i}$ where $c_i(\alpha, \beta)$ represent some weight function. The motivation behind the definition of f_t^∞ follows from the idea to replace f_t with a loss function that fits the full information online optimization model. Instead of using the entire past history to make prediction, we consider a fixed-length history. We set $m \in \mathbb{N}$, and define

$$\nabla^d X_t^m(\alpha, \beta) {=} \sum_{i=1}^k \alpha_i \nabla^d X_{t-i} {+} \sum_{i=1}^q \beta_i (\nabla^d X_{t-i} {-} \nabla^d X_{t-i}^{m-i}(\alpha, \beta)),$$

$$X_t^m(\alpha, \beta) = \nabla^d X_t^m(\alpha, \beta) + \sum_{i=1}^{d-1} \nabla^i X_{t-1}, \tag{11}$$

with initial condition $X_t^m(\alpha, \beta) = X_t$ for all t and $m \leq 0$. We then denote by

$$f_t^m(\alpha, \beta) = \ell_t(X_t, X_t^m(\alpha, \beta)), \tag{12}$$

the loss suffered by the prediction $X_t^m(\alpha, \beta)$ at iteration t. Since it is easier to generate predictions using only the last $(m + k)$ observations, and the distance between the loss function is relatively small. Now, let us denote by $(\alpha^\star, \beta^\star) = arg \min_{\alpha,\beta} \sum_{t=1}^T \mathbb{E}[f_t(\alpha, \beta)]$ the best ARIMA model coefficient in hindsight for predicting the observation $\{X_t\}_{t=1}^T$. Then, from Lemma 1, stated and proven below, we have that

$$\min_{\gamma} \sum_{t=1}^T \ell_t^m(\gamma) \leq \sum_{t=1}^T f_t^m(\alpha^\star, \beta^\star),$$

and it follows that

$$\sum_{t=1}^T \ell_t^m(\gamma^t) - \sum_{t=1}^T f_t^m(\alpha^\star, \beta^\star) = O((GD + \frac{1}{\lambda}) \log T).$$

From Lemma 3 we know that

$$|\sum_{t=1}^T \mathbb{E}[f_t^\infty(\alpha^\star, \beta^\star)] - \sum_{t=1}^T \mathbb{E}[f_t^m(\alpha^\star, \beta^\star)]| = O(1),$$

for $m = \log_{\lambda_{min}}((TLM_{max}q)^{-1})$, which implies that

$$\sum_{t=1}^T \ell_t^m(\gamma^t) - \sum_{t=1}^T \mathbb{E}[f_t^\infty(\alpha^\star, \beta^\star)] = O((m + k + \frac{1}{\lambda}) \log T).$$

Finally, from Lemma 4 below we know that

$$|\sum_{t=1}^T \mathbb{E}[f_t^\infty(\alpha^\star, \beta^\star)] - \sum_{t=1}^T \mathbb{E}[f_t(\alpha^\star, \beta^\star)]| = O(1),$$

and thus

$$\sum_{t=1}^T \ell_t^m(\gamma^t) - \sum_{t=1}^T \mathbb{E}[f_t(\alpha^\star, \beta^\star)]$$

$$= O((GD + \frac{1}{\lambda}) \log T) = O((\log(q) + \log(T)) \log T).$$

□

In the following, we will give several important lemmas that are critical to obtaining the bounds as used in the above proof. Due to space limitation, the detailed proofs of Lemma 1, Lemma 3 and Lemma 4 are given in the supplementary file [1]. In our analysis, we adopt difference equation techniques and use a recursive formulation of ARIMA model to eliminate the effect of noise terms and degree of differencing. Lemma 3 and Lemma 4 prove that if we take a length of order $\log(q)+\log(T)$, the distance between the new loss function and the original one is small in expectation.

Lemma 1. *According to Eq. 7 and 12. From any time series sequence satisfies the assumption above, it holds that*

$$\min_{\gamma}\sum_{t=1}^{T}\ell_t^m(\gamma)\leq\sum_{t=1}^{T}f_t^m(\alpha^\star,\beta^\star).$$

Lemma 2. *Given assumption 1 that a q-th order difference equation with coefficients $|\beta_1|\cdots,|\beta_q|$ and observations $\{X_t\}_{t=-(q-1)}^{T}$ is a stationary process, $\lambda_1,\cdots,\lambda_q$ are the q roots of this AR characteristic equation. Let we set $\lambda_{min}=\{|\lambda_1|,\cdots,|\lambda_q|\}$, it holds that*

$$X_t\leq\lambda_{min}^t(X_0+X_1+\cdots+X_{-(q-1)})$$

Proof. We rewrite this q-th order difference equation in a scalar X_t as a first order difference equation in a vector style. Define the initial vector $\psi_{-\mathbf{1}}$, and a $(q\times q)$ matrix $\mathbf{F}$ by

$$\psi_{-1}=\begin{bmatrix}X_{-1}\\X_{-2}\\\vdots\\X_{-q}\end{bmatrix},\mathbf{F}=\begin{bmatrix}\beta_1&\beta_2&\beta_3&\cdots&\beta_{q-1}&\beta_q\\1&0&0&\cdots&0&0\\0&1&0&\cdots&0&0\\\vdots&\vdots&\ddots&\cdots&\vdots&\vdots\\0&0&0&\cdots&1&0\end{bmatrix}.$$

The i-th row matrix $\mathbf{F}$ is denoted by $\mathbf{F}_{i\cdot}$. Thus, we can get an alternative representation $X_t=\mathbf{F}_{1\cdot}^{t+1}\psi_{-1}$. A stationary solution to this difference equation exists if and only if the q roots of the AR characteristic equation each is no more than 1 in absolute value. The eigenvalues of matrix $\mathbf{F}$ are equivalent to the q roots of the AR characteristic equation ((Hamilton 1994) gives a detailed proof). For simplicity, we assume $\lambda_1,\lambda_2,\cdots,\lambda_q$ are distinct. Thus, there exists a $(q\times q)$ matrix $\mathbf{T}$ such that $\mathbf{F}=\mathbf{T}\Lambda\mathbf{T}^{-1}$, where Λ is a $(q\times q)$ matrix with the eigenvalues of $\mathbf{F}$ along the principal diagonal and zeros elsewhere. This enables us to characterize $\mathbf{F}^t$ in terms of the eigenvalues of $\mathbf{F}$ as

$$\mathbf{F}^t=\underbrace{\mathbf{T}\Lambda\mathbf{T}^{-1}\times\mathbf{T}\Lambda\mathbf{T}^{-1}\times\cdots\times\mathbf{T}\Lambda\mathbf{T}^{-1}}_{\text{t terms}}=\mathbf{T}\Lambda^t\mathbf{T}^{-1}.$$

Let us denote by f_{ij}^t the (i,j)-element of $\mathbf{F}^t$, t_{ij} the (i,j)-element of $\mathbf{T}$, t^{ij} the (i,j)-element of $\mathbf{T}^{-1}$. Then, the 1st row, i-th column element of $\mathbf{F}^t$ written out explicitly becomes

$$f_{1i}^t=[t_{i1}t^{i1}]\lambda_1^t+[t_{i2}t^{i2}]\lambda_2^t+\cdots+[t_{iq}t^{iq}]\lambda_q^t\leq\lambda_{min}^t.\quad(13)$$

Since $[t_{i1}t^{i1}]+[t_{i2}t^{i2}]+\cdots+[t_{iq}t^{iq}]$ is equivalent to the (i,i) element of $\mathbf{T}\cdot\mathbf{T}^{-1}$. And $\mathbf{T}\cdot\mathbf{T}^{-1}$ is just the $(q\times q)$ identity matrix, which implies that the $[t_{ij}t^{ij}]$ terms sum to unity. f_{1i}^t could be regarded as a weighted average of each of the q eigenvalues raised to the t-th power.

Therefore, we have

$$\begin{aligned}X_t&=\mathbf{F}_{1\cdot}^{t+1}\psi_{-1}=f_{11}^tX_0+f_{12}^tX_{-1}+\cdots+f_{1q}^tX_{-(q-1)}\\&\leq\lambda_{min}^t(X_0+X_{-1}+\cdots+X_{-(q-1)}).\end{aligned}$$

□

Lemma 3. *According to Eq. 10 and 12. For any time series sequence satisfies the assumption above, it holds that*

$$|\sum_{t=1}^{T}\mathbb{E}[f_t^\infty(\alpha^\star,\beta^\star)]-\sum_{t=1}^{T}\mathbb{E}[f_t^m(\alpha^\star,\beta^\star)]|=O(1),$$

if we choose $m=\log_{\lambda_{min}}((TLM_{max}q)^{-1})$.

Lemma 4. *According to Eq. 4 and 10. For any time series sequence satisfies the assumption above, it holds that*

$$|\sum_{t=1}^{T}\mathbb{E}[f_t^\infty(\alpha^\star,\beta^\star)]-\sum_{t=1}^{T}\mathbb{E}[f_t(\alpha^\star,\beta^\star)]|=O(1).$$

For Algorithm 2, we can prove the following theorem:

Theorem 2. *Let $k,q\geq 1$, and set $\eta=\frac{D}{G\sqrt{T}}$. Then, for any sequence $\{X_t\}_{t=1}^T$ satisfying the above assumptions, the sequence $\{\gamma_t\}_{t=1}^T$ generated by Algorithm 2 guarantees:*

$$\begin{aligned}&\sum_{t=1}^{T}\ell_t^m(\gamma^t)-\min_{\alpha,\beta}\sum_{t=1}^{T}\mathbb{E}[f_t(\alpha,\beta)]\\&=O(GD\sqrt{T})=O((\log(q)+\log(T))\sqrt{T}).\qquad(14)\end{aligned}$$

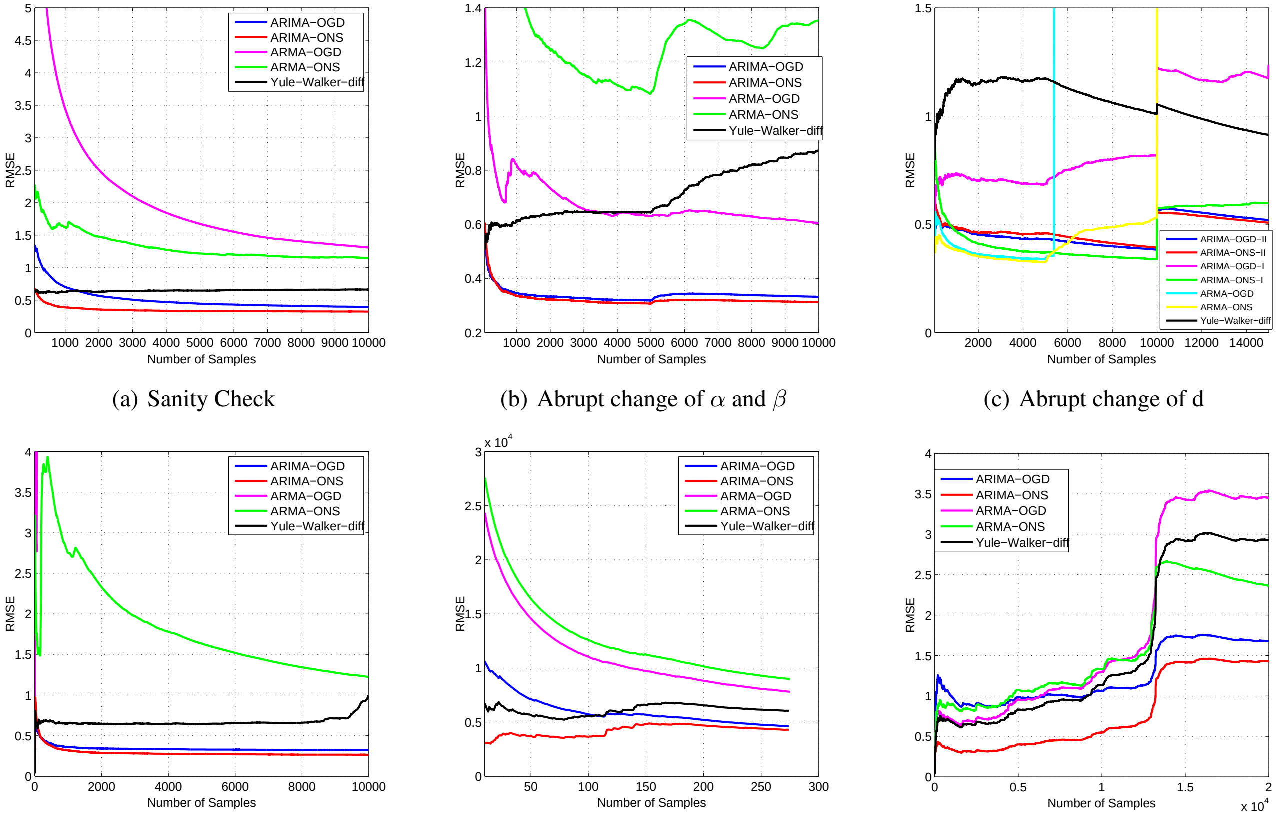
The proof of this theorem is similar to that of Theorem 1, except using the Online Gradient Descent algorithm (Zinkevich 2003) rather than the ONS algorithm.

Remark. In contrast to (Anava et al. 2013), our work has several key advantages. First of all, we do not restrict the parameter β_i with $\sum_{i=1}^q|\beta_i|<1-\epsilon$ for some $\epsilon>0$, which plays a key role in bounding the approximation error. Their assumption is a necessary, but not sufficient condition to our assumption 1. It not only restricts the parameter β_i but also introduces additional new parameter ϵ. Second, even using a more general assumption about β, we can obtain a much smaller value of m in the order of $O\big(\log(q)+\log(T)\big)$, which is much smaller than the result of $m=O(q\log(T))$ in (Anava et al. 2013). Finally, note that our proof method adopts difference equation techniques to approximate the original ARIMA model, which is very different from the analysis techniques used in (Anava et al. 2013).

Experiments

In this section, we conduct experiments on both synthetic and real data to examine the effectiveness and robustness of our online ARIMA algorithms. We compare both the proposed ARIMA-OGD and ARIMA-ONS algorithms with the two existing ARMA algorithms (ARMA-OGD adn ARMA-ONG) proposed in (Anava et al. 2013) based on online root-mean square error (RMSE). Besides, we also

[1] http://OARIMA.stevenhoi.org



(a) Sanity Check

(b) Abrupt change of α and β

(c) Abrupt change of d

(d) Slowly change of α and β

(e) American Vehicles

(f) Dow Jones Industrial Average

Figure 1: Experimental results on six datasets (the results were reported by taking the average results from 20 runs)

compare with the standard Yule-Walker estimation method (Hamilton 1994). Since it is a batch learning method, it utilizes all the previous historical observations to make prediction at each iteration. In light of this, when adapting it in an online setting, this method takes increasingly long time as T increases. To evaluate different algorithms, we design experiments for several settings, in which each experiment was repeated 20 times to yield stable average results and we choose parameter $m + k = 10$ for all the settings [2].

Setting 1. We generate a stationary time series data by assuming the ARIMA model using $\alpha = [0.6, -0.5, 0.4, -0.4, 0.3]$, $\beta = [0.3, -0.2]$ and $d = 1$, the noise terms are normally distributed as $\mathcal{N}(0, 0.3^2)$. As can be seen in Figure 1(a), ARIMA-ONS algorithm outperforms the other online algorithms and quickly approaches the optimum, which verified its theoretical lower regret bound.

Setting 2. We generate a non-stationary time series data by assuming the ARIMA model with two different sets of parameters. The first set is $\alpha = [0.6, -0.5, 0.4, -0.4, 0.3]$, $\beta = [0.3, -0.2]$, $d = 1$, and it is used for generating the first half of the sequence. The second set is $\alpha = [-0.4, -0.5, 0.4, 0.4, 0.1]$, $\beta = [-0.3, 0.2]$, $d = 1$, for generating the second half. The noise terms are distributed $Uni[-0.5, 0.5]$. In Figure 1(b), we show the effectiveness of our proposed method in a context when when parameters α and β abruptly change. Note that ARMA-ONS and ARMA-ONS methods can not fit the abrupt change well and even diverge finally, while our algorithms work well.

Setting 3. We generate a non-stationary time series data by assuming the ARIMA model with $\alpha = [0.6, -0.5, 0.4, -0.4, 0.3]$, $\beta = [0.3, -0.2]$ but different settings about parameter d. $d = 0$ for the first stage, $d = 1$ for the second, and $d = 2$ for the final one. We also compared different settings of d for ARIMA-ONS-II ($d = 2$) and ARIMA-ONS-I ($d = 1$) algorithms (similar to ARIMA-OGD-II and ARIMA-OGD-I). In Figure 1(c), we can clearly see that ARMA-ONS outperforms other algorithms at the first stage due to over-differencing of ARIMA-ONS-I and ARIMA-ONS-II. At the second stage, ARIMA-ONS-I outperforms other algorithms, but ARMA-OGD suddenly diverge due to under-differencing. The final stage shows the superiority of ARIMA-ONS-II method. This experiment demonstrates that differencing is a key factor to successfully model observation sequence.

Setting 4. We generate a non-stationary time series data by assuming the ARIMA model using $\beta =$

[2] All the datasets and source codes for our experiments can be found in our webpage `http://OARIMA.stevenhoi.org`

$[0.32, -0.2]$ and $\alpha(t) = [-0.4, 0.5, 0.4, 0.4, 0.1] \times (\frac{t}{10^4}) + [0.6, -0.4, 0.4, -0.5, 0.4] \times (1 - \frac{t}{10^4})$. In this setting, coefficients change slowly in time. In Figure 1(d), we can clearly see the advantage of our proposed method.

Real-world data. We evaluated our proposed method on some real-world time series data. The first time series data describes monthly registration of private cars during years 1980-1998. Figure 1(e) shows that all algorithms could uncover the pattern behind it. But ARIMA-ONS method significantly outperforms others. The second time series data is daily index of Dow Jones Industrial Average (DJIA) during years 1885-1962. The results in Figure 1(f) indicate that the existence of abrupt change, for which ARIMA-ONS can significantly better adapt it than others.

Conclusion

This paper proposed a novel online learning method with the ARIMA model for time series prediction. We formulated online ARIMA learning as a task of full information online optimization task without noise terms, and theoretically proved that our method attains a sublinear regret bound against the best fixed ARIMA model in hindsight. Moreover, we empirically compared our algorithms with two recent online ARMA algorithms, in which the promising results on both synthetic data and real data validate that our new algorithms are effective and promising for time series prediction.

Acknowledgments

This work was done when the first author visited Prof. Hoi's group at Singapore Management University. This work was supported by Singapore Ministry of Education Academic Research Fund Tier 1 Grant (14-C220-SMU-016).

References

[Anava et al. 2013] Anava, O.; Hazan, E.; Mannor, S.; and Shamir, O. 2013. Online learning for time series prediction. In *Conference on Learning Theory*, 172–184.

[Brockwell and Davis 2009] Brockwell, P. J., and Davis, R. A. 2009. *Time series: theory and methods*. Springer Science & Business Media.

[Bubeck 2011] Bubeck, S. 2011. *Introduction to online optimization*. Princeton University.

[Damsleth and El-Shaarawi 1989] Damsleth, E., and El-Shaarawi, A. 1989. Arma models with double-exponentially distributed noise. *Journal of the Royal Statistical Society. Series B (Methodological)* 61–69.

[Ding, Shi, and Chen 2006] Ding, F.; Shi, Y.; and Chen, T. 2006. Performance analysis of estimation algorithms of nonstationary arma processes. *Signal Processing, IEEE Transactions on* 54(3):1041–1053.

[Engle 1982] Engle, R. F. 1982. Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica: Journal of the Econometric Society* 987–1007.

[Gao et al. 2010] Gao, J.; Sultan, H.; Hu, J.; and Tung, W.-W. 2010. Denoising nonlinear time series by adaptive filtering and wavelet shrinkage: a comparison. *Signal Processing Letters, IEEE* 17(3):237–240.

[George 1994] George, B. 1994. *Time Series Analysis: Forecasting & Control, 3/e*. Pearson Education India.

[Granger and Newbold 2014] Granger, C. W. J., and Newbold, P. 2014. *Forecasting economic time series*. Academic Press.

[Hamilton 1994] Hamilton, J. D. 1994. *Time series analysis*, volume 2. Princeton university press Princeton.

[Hazan, Agarwal, and Kale 2007] Hazan, E.; Agarwal, A.; and Kale, S. 2007. Logarithmic regret algorithms for online convex optimization. *Machine Learning* 69(2-3):169–192.

[Hoi, Wang, and Zhao 2014] Hoi, S. C.; Wang, J.; and Zhao, P. 2014. Libol: A library for online learning algorithms. *The Journal of Machine Learning Research* 15(1):495–499.

[Huang and Shih 2003] Huang, S.-J., and Shih, K.-R. 2003. Short-term load forecasting via arma model identification including non-gaussian process considerations. *Power Systems, IEEE Transactions on* 18(2):673–679.

[Li and Hoi 2015] Li, B., and Hoi, S. C. H. 2015. *Online Portfolio Selection: Principles and Algorithms*. CRC Press.

[Lii 1990] Lii, K.-S. 1990. Identification and estimation of non-gaussian arma processes. *Acoustics, Speech and Signal Processing, IEEE Transactions on* 38(7):1266–1276.

[Nerlove, Grether, and Carvalho 2014] Nerlove, M.; Grether, D. M.; and Carvalho, J. L. 2014. *Analysis of economic time series: a synthesis*. Academic Press.

[Rabiner and Schafer 2011] Rabiner, L., and Schafer, R. 2011. Digital speech processing. *The Froehlich/Kent Encyclopedia of Telecommunications* 6:237–258.

[Rojo-Álvarez et al. 2004] Rojo-Álvarez, J. L.; Martínez-Ramón, M.; de Prado-Cumplido, M.; Artés-Rodríguez, A.; and Figueiras-Vidal, A. R. 2004. Support vector method for robust arma system identification. *Signal Processing, IEEE Transactions on* 52(1):155–164.

[Shalev-Shwartz et al. 2011] Shalev-Shwartz, S.; Singer, Y.; Srebro, N.; and Cotter, A. 2011. Pegasos: Primal estimated sub-gradient solver for svm. *Mathematical programming* 127(1):3–30.

[Tiku et al. 2000] Tiku, M. L.; Wong, W.-K.; Vaughan, D. C.; and Bian, G. 2000. Time series models in non-normal situations: Symmetric innovations. *Journal of Time Series Analysis* 21(5):571–596.

[Tsay and Tiao 1984] Tsay, R. S., and Tiao, G. C. 1984. Consistent estimates of autoregressive parameters and extended sample autocorrelation function for stationary and nonstationary arma models. *Journal of the American Statistical Association* 79(385):84–96.

[Tsay 2005] Tsay, R. S. 2005. *Analysis of financial time series*, volume 543. John Wiley & Sons.

[Weigend 1994] Weigend, A. S. 1994. Time series prediction: forecasting the future and understanding the past. *Santa Fe Institute Studies in the Sciences of Complexity*.

[Zinkevich 2003] Zinkevich, M. 2003. Online convex programming and generalized infinitesimal gradient ascent.