An Online Prediction Framework for Non-Stationary Time Series

Christopher Xie
University of Washington
chrisxie@cs.washington.edu

Avleen Bijral
Microsoft
avbijral@microsoft.com

Juan Lavista Ferres
Microsoft
jlavista@microsoft.com

Abstract

We extend an online time series prediction algorithm for ARMA processes to describe a framework for time series prediction that can efficiently handle non-stationarities that exist in many real time series. We show that appropriate transformations to such time series can lead to theoretical and empirical gains. To account for the phenomenon of cointegration in the multivariate case, we present a novel algorithm EC-VARMA-OGD that estimates both the auto-regressive and the cointegrating parameters. Relaxing the assumptions for the analysis, we prove a sublinear regret bound for all the methods described. We note that the theoretical guarantees do not provide a complete picture, thus we provide a data-dependent analysis of the follow-the-leader algorithm for least squares loss that explains the success of using non-stationary transformations. We support all of our results with experiments on simulated and real data.

1 Introduction

In the analysis of time series, AutoRegressive Moving Average (ARMA) models [5, 2, 6] are simple and powerful descriptors of weakly stationary processes. As such, they have found tremendous application in many domains including linear dynamical systems, econometrics, and forecasting resource consumption [5]. They continue to be of immense practical use, especially due to the proliferation of sensors/devices generating time dependent data.

Despite a large amount of literature on model estimation and prediction for these models, most of it remains within the confines of the statistical assumption of Gaussianity. Such assumptions are often unrealistic [15] and lead to poor prediction performance. Moreover, since the noise sequence is not known beforehand, standard methods of ARMA estimation rely on conditional likelihood estimation. These methods usually lead to nonlinear estimation problems and only hold for Gaussian residual sequences and the least squares loss.

In the setting of streaming or high-frequency time series, one would ideally like to have methods that update the model, predict sequentially, and do not rely on any restricting assumptions on the noise sequence or the loss function. This brings attention to the paradigm of online learning [3]. In that vein, Anava et al. [1] recently presented online gradient and Newton methods (ARMA-OGD and ARMA-ONS) for ARMA prediction. Using a truncated auto-regressive (AR) representation of an ARMA process, the authors provide online ARMA prediction algorithms with sublinear regret, where the regret is with respect to the best conditionally expected one-step ARMA prediction loss in hindsight (See [1] for more details).

Anava et al. [1] make no assumption about the stationarity of the generating ARMA process and only assume a bound on the ℓ_1-norm of the coefficients of the moving average part. However we will see in the empirical results section that the performance of ARMA-OGD suffers in the presence of seasonality and/or trends which are very common in real time series [5]. In this paper, we present a general framework for online time series prediction and show that adjusting the data for known characteristics such as trends/seasonality can lead to significant empirical and some theoretical gains. Additionally, we extend this general framework for vector time series to provide extensions for VARMA processes [16, 12]. More precisely, we provide an algorithm for prediction in potentially non-stationary vector valued time series generated by error corrected VARMA (EC-VARMA) processes. Estimating EC-VARMA models are non-trivial in general and

Preliminary work. Under review by AISTATS 2017. Do not distribute.

we provide an efficient algorithm that simultaneously estimates both the error correcting and the VARMA matrix parameters. Please see [16] for more details on error corrected models.

Our regret guarantees and that of [1, 10] do not completely explain why the convergence for these online prediction methods is faster for seasonal/trend adjusted data. We conjecture that these bounds are missing data-dependent terms that capture correlations inherent in many real time series (e.g. with seasonalities and trends). To give a flavor of what a satisfactory data dependent regret bound might look like, we analyze the regret for the Follow-The-Leader (FTL) algorithm in the case of least squares loss and show that these bounds depend on a data dependent term and can be compared across the different spectrum of real time series (stationary/trend/seasonal etc).

1.1 Contributions

Our contributions in this paper can be highlighted as follows:

1. We provide a general framework for time series prediction using Online Gradient Descent (OGD) that allows for appropriate modifications to real time series before making a prediction. Such transformations often result in good empirical convergence properties.

2. Our regret analysis only requires invertibility of the moving average polynomial (See [5] for a discussion of invertibility), while the assumptions in [1] and [10] are less natural. Moreover, we don't require an upper bound on the data as non-stationary data can be unbounded.

3. We also provide a non-trivial online algorithm for potentially non-stationary vector valued time series. This algorithm, refered to as EC-VARMA-OGD, updates both the error correcting and other parameters of the model before making a prediction using online gradient descent.

4. To highlight the effect of the these transformations, we a prove a data dependent regret guarantee for FTL (for least squares loss) that reveals why adjusting for non-stationarities can give faster convergence.

Note that we can easily extend our algorithms and analysis for the online Newton step method of [1] . But for simplicity and efficiency, we limit our analysis to OGD.

2 Preliminaries: Time Series Modeling

In this section, we provide a summary of time series models and other related concepts explored and built upon in our work. For an introduction to online convex optimzation and online gradient descent, please see [14, 17, 7].

2.1 Notation

A time series is an ordered sequence of observations $\{x_t\}_{t\in\mathbb{Z}}$. We denote $x_t \in \mathbb{R}$ as a univariate time series. Let L denote the lag operator, i.e. $Lx_t = x_{t-1}$. We define $\Delta x_t = x_t - x_{t-1}$ to be a non-seasonal differencing operator and $\Delta_s x_t = x_t - x_{t-s}$ to be a seasonal differencing operator. These differencing operators can be compounded: $\Delta^2 x_t = (\Delta x_t - \Delta x_{t-1}) = x_t - 2x_{t-1} + x_{t-2}$.

2.2 SARIMA Processes

Time series exhibiting seasonal patterns can be modeled by Seasonal AutoRegressive Integrated Moving Average (SARIMA) Processes. SARIMA$(p,d,q) \times (P,D,Q)_s$ processes are described by the following equation:

$$\phi(L)\Phi(L^s)\Delta^d\Delta_s^D x_t = \theta(L)\Theta(L^s)\varepsilon_t \tag{1}$$

where $\phi(L) = 1 - \sum_{i=1}^{p}\phi_i L^i, \theta(L^s) = 1 + \sum_{i=1}^{q}\theta_i L^i$, $\Phi(L^s) = 1 - \sum_{i=1}^{P}\Phi_i L^{is}, \Theta(L^s) = 1 + \sum_{i=1}^{Q}\Theta_i L^{is}$ and $\phi, \Phi, \theta, \Theta \in \mathbb{R}$. $\phi(L)$ and $\theta(L)$ are the non-seasonal autoregressive (AR) and moving average (MA) lag polynomials, respectively. Similarly, $\Phi(L^s)$ and $\Theta(L^s)$ are the seasonal AR and the seasonal MA lag polynomials, respectively.

SARIMA processes explicitly model trend and seasonal non-stationarities by assuming that the differenced process $\Delta^d\Delta_s^D x_t$ is an ARMA process with AR lag polynomial $\phi(L)\Phi(L^s)$ and MA lag polynomial $\theta(L)\Theta(L^s)$. We denote the order of the AR and MA lag polynomials as l_a and l_m, respectively. For SARIMA$(p,d,q) \times (P,D,Q)_s$ processes, Eq. 1 gives us that $l_a = p + Ps$ and $l_m = q + Qs$.

If the MA lag polynomial has all of its roots outside of the complex unit circle, then the SARIMA process is defined as invertible. Let β_i be the scalar coefficients of the MA lag polynomial. Invertibility is equivalent

to saying that the companion matrix

$$\mathbf{F} = \begin{bmatrix} -\beta_1 & -\beta_2 & \dots & \dots & -\beta_{l_m} \\ 1 & 0 & \dots & \dots & 0 \\ 0 & 1 & 0 & \dots & \vdots \\ \vdots & 0 & \ddots & \ddots & \vdots \\ 0 & \vdots & \vdots & 1 & 0 \end{bmatrix} \quad (2)$$

has eigenvalues less than 1 in magnitude. If this is the case, then the underlying ARMA process $\Delta^d \Delta_s^D x_t$ can be written as an AR(∞) process. If invertibility holds, then the underlying ARMA process can be approximated quite well with an AR process by truncating the infinity to a large finite number.

A key point to note is that a SARIMA$(p, d, q) \times (P, D, Q)_s$ process can be viewed as an ARIMA$(p + Ps + Ds,\ d,\ q + Qs)$ process with AR lag polynomial $\theta(L)\Theta(L^s)(1-L^s)^D$. Likewise, a SARIMA process can also be viewed as an ARMA$(p+Ps+Ds+d,\ q+Qs)$ process with AR lag polynomial $\theta(L)\Theta(L^s)(1-L)^d(1-L^s)^D$. A SARIMA process is ARIMA process with structure, and is an ARMA process with additional structure. ARMA processes are the most general of the family.

2.3 VARMA Processes

Vector AutoRegressive Moving Average (VARMA) processes provide a parsimonious description of modeling linear multivariate time series. Let $\mathbf{x}_t, \boldsymbol{\varepsilon}_t \in \mathbb{R}^k, \boldsymbol{\Phi}_i \in \mathbb{R}^{k\times k}, \boldsymbol{\Theta}_i \in \mathbb{R}^{k\times k}$. A VARMA$(p, q)$ process is described by:

$$\mathbf{x}_t = \sum_{i=1}^{p} \boldsymbol{\Phi}_i \mathbf{x}_{t-i} + \sum_{i=1}^{q} \boldsymbol{\Theta}_i \boldsymbol{\varepsilon}_{t-i} + \boldsymbol{\varepsilon}_t \quad (3)$$

which can also be written in lag polynomial form:

$$\boldsymbol{\Phi}(L)\mathbf{x}_t = \boldsymbol{\Theta}(L)\boldsymbol{\varepsilon}_t \quad (4)$$

with $\boldsymbol{\Phi}(L) = I - \sum_{i=1}^{p} \boldsymbol{\Phi}_i L^i, \boldsymbol{\Theta}(L) = I + \sum_{i=1}^{q} \boldsymbol{\Theta}_i L^i$. The requirements for invertibility are very similar to the univariate case. We require that $\det(\boldsymbol{\Theta}(L))$ must have all of its roots outside of the complex unit circle. Again, this is equivalent to saying that the companion matrix has eigenvalues less than 1 in magnitude [12, 16]. If the process is invertible, then it can be rewritten as a VAR(∞) process.

2.4 EC-VARMA Model

In many cases, a collection of time series may follow a common trend. This phenomenon, known as cointegration, is ubiquitous in economic times series [16]. Formally let $\mathbf{x}_t$ be described by Eq. 3. Then $\mathbf{x}_t$ is cointegrated if $\Delta\mathbf{x}_t$ is stationary and there exists a vector $\mu \in \mathbb{R}^k$ such that $\mu^\intercal \mathbf{x}_t$ is a stationary process. If $\mathbf{x}_t$ is cointegrated, then we can rewrite the original VARMA representation of $\mathbf{x}_t$ as

$$\Delta\mathbf{x}_t = \Pi\mathbf{x}_{t-1} + \sum_{i=1}^{p-1} \Gamma_i \Delta\mathbf{x}_{t-i} + \sum_{i=1}^{q} \boldsymbol{\Theta}_i \boldsymbol{\varepsilon}_{t-i} + \boldsymbol{\varepsilon}_t \quad (5)$$

where $\Pi = -\boldsymbol{\Phi}(1)$ is low rank, and $\Gamma_j = -(\boldsymbol{\Phi}_{j+1} + \ldots + \boldsymbol{\Phi}_p)$ for $j = 1, \ldots p-1$. Eq. 5 is known as an Error-Corrected VARMA (EC-VARMA) model. Note that this looks like a VARMA$(p-1, q)$ process in the differenced series $\Delta\mathbf{x}_t$, except that there is the error-correction term $\Pi\mathbf{x}_{t-1}$. See [12, 16] for more details.

This family of models is the multivariate analogue of the ARIMA model. However, differencing in multivariate time series means something else entirely and thus we must consider the notion of error correction models. We refer the reader to [16] for a discussion of this issue.

3 Online Time Series Prediction Framework

In this section we present an algorithmic framework for online time series prediction based on the ARMA-OGD algorithm presented in [1], which also includes the extension to trend non-stationarities as presented as ARIMA-OGD in [10].

Precisely, we show that time series with certain characteristics (such as a trend or seasonality) can be appropriately transformed before prediction to give better theoretical and empirical results. To this end, we present a unified template for time series prediction using OGD that allows for prediction of transformed time series. We will show that the choice of the transformation, dependent on the underlying data generation process (DGP), can lead to improved constants in the regret guarantee, partially explaining why these transformations lead to better empirical performance.

This framework includes some of the commonly used transformations of seasonal and non-seasonal differencing [5]. Table 1 shows the explicit form of of such transformations.

3.1 TSP-OGD

We assume the following for the remainder of this section:

U1) x_t is generated by a DGP such that there exists a transformation $\tau(x_t)$ which is an invertible

Algorithm 1 TSP-OGD Framework

Require: DGP parameters l_a, l_m. Horizon T. Learning rate η. Data: $\{x_t\}$. Transformation τ. Inverse Transformation ζ.

Set $M = \log_{\lambda_{\max}} \left(\left(2\kappa TLM_{\max}\sqrt{l_m} \right)^{-1} \right) + l_a$
Transform x_t to get $\tau(x_t)$.
Choose $\boldsymbol{\gamma}^{(1)} \in \mathcal{E}$ arbitrarily.
for $t = 1$ to T **do**
  $\tau\left(\tilde{x}_t\right) = \sum_{i=1}^{M} \boldsymbol{\gamma}_i^{(t)} \tau\left(x_{t-i}\right)$
  Predict $\tilde{x}_t = \zeta(\tau(\tilde{x}_t))$
  Observe x_t and receive loss $\ell_t^M\left(\boldsymbol{\gamma}^{(t)}\right)$
  Set $\boldsymbol{\gamma}^{(t+1)} = \Pi_{\mathcal{E}}\left(\boldsymbol{\gamma}^{(t)} - \eta \nabla \ell_t^M\left(\boldsymbol{\gamma}^{(t)}\right)\right)$
end for

Table 1: DGPs and their Transformations

DGP	$\tau(x_t)$	$\zeta(\tilde{x}_t)$
ARMA	x_t	$\tilde{x}_t$
ARIMA	$\Delta^d x_t$	$\tilde{x}_t + \sum_{i=0}^{d-1} \Delta^i x_{t-1}$
SARIMA	$\Delta^d \Delta_s^D x_t$	$\tilde{x}_t + \sum_{i=0}^{d-1} \Delta^i \Delta_s^D x_{t-1} + \sum_{i=0}^{D-1} \Delta_s^i x_{t-s}$

ARMA process. Moreover, there corresponds an inverse transformation ζ that satisfies $\zeta(\tau(x_t)) = x_t$. Examples of such a process are ARMA, ARIMA, and SARIMA processes.

U2) The noise sequence ε_t of the underlying ARMA process is independent. Also, it satisfies that $\mathbb{E}[|\varepsilon_t|] < M_{\max} < \infty$.

U3) $\ell_t : \mathbb{R}^2 \to \mathbb{R}$ is a convex loss function with Lipschitz constant $L > 0$.

U4) We assume the companion matrix $\mathbf{F}$ (as defined in Eq. 2) of the MA lag polynomial is diagonalizable, i.e. $\mathbf{F} = \mathbf{T}\Lambda\mathbf{T}^{-1}$ where Λ is a diagonal matrix of eigenvalues. Denote $\lambda_{\max}$ as the magnitude of the largest eigenvalue, and $\kappa \in \mathbb{R}$ such that $(\sigma_{\max}(\mathbf{T})/\sigma_{\min}(\mathbf{T})) \leq \kappa$.

In Algorithm 1, the model parameters of the stochastic process are fixed by an adversary. At time t, ε_t and x_t are generated by the DGP. Before x_t is revealed to us the learner (see Algorithm 1) makes a prediction $\tilde{x}_t$ which incurs a prediction loss of $\ell_t(x_t, \tilde{x}_t)$. This prediction is preceded by a transform τ (See Table 1) that may require data points from previous rounds (we suppress that dependence in the notation for convenience). The prediction $\tau(\tilde{x}_t)$ is computed using an AR model of order M to approximate the invertible ARMA process. Then it is inverted with ζ and incurs a loss

$$\ell_t^M(\boldsymbol{\gamma}) = \ell_t\left(x_t, \zeta\left(\tau(\tilde{x}_t)\right)\right) = \ell_t\left(x_t, \zeta\left(\sum_{i=1}^{M} \boldsymbol{\gamma}_i \tau\left(x_{t-i}\right)\right)\right) \tag{6}$$

where $\boldsymbol{\gamma}$ is the vector of parameters of the approximating AR model. The prediction performance is evaluated using an "extended" notion of regret that looks at the prediction loss of the best process in hindsight. Precisely, let $\boldsymbol{\alpha}, \boldsymbol{\beta}$ denote the set of AR and MA parameters, respectively, of the underlying ARMA process $\tau(x_t)$. Define

$$f_t(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \ell_t\left(x_t, \zeta\left(\mathbb{E}\left[\tau(x_t) | \tau\left(\{x_t\}_{t=1}^{t-1}\right); \boldsymbol{\alpha}, \boldsymbol{\beta}\right]\right)\right) \tag{7}$$

Note that f_t depends on the transformations τ, ζ in U1. The extended regret is defined as comparing the accumulated loss in Eq. 6 to the loss of the best process in hindsight:

$$\text{Regret} = \sum_{t=1}^{T} \ell_t^M(\gamma^{(t)}) - \min_{\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathcal{K}} \sum_{t=1}^{T} \mathbb{E}[f_t(\boldsymbol{\alpha}, \boldsymbol{\beta})] \tag{8}$$

where $\mathcal{K}$ is the set of invertible ARMA processes. Note that the randomness in the expectation is w.r.t. the noise sequence ε_t while the data x_t is fixed.

Furthermore, let $\mathcal{E} \subseteq \mathbb{R}^M$ be a convex set of approximating AR models, i.e. $\boldsymbol{\gamma} \in \mathcal{E}$. $\mathcal{E}$ should be chosen to be large enough to include a valid approximation to the DGP described in U1. However, since the DGP is unknown in practice, one usually chooses something simple such as $\mathcal{E} = \{\boldsymbol{\gamma} : \|\boldsymbol{\gamma}\|_\infty \leq 1\}$. Let $D = \sup_{\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2 \in \mathcal{E}} \|\boldsymbol{\gamma}_1 - \boldsymbol{\gamma}_2\|_2$, and $\|\nabla_{\boldsymbol{\gamma}} \ell_t^m(\boldsymbol{\gamma})\|_2 \leq G(T)$ for some monotonically increasing $G(T)$. This assumption essentially stems from the fact that we allow the time series to be potentially unbounded. As an example, the norm of the gradient for the squared loss depends on the bound on the data. Let $\Pi_{\mathcal{E}}$ denote the projection operator onto the set $\mathcal{E}$.

We present a general regret bound for Algorithm 1.

Theorem 3.1. *Let $\eta = \frac{D}{G(T)\sqrt{T}}$. Then for any data sequence $\{x_t\}_{t=1}^T$ that satisfies assumptions U1-U4, Algorithm 1 generates a sequence $\{\boldsymbol{\gamma}^{(t)}\}$ in which*

$$\sum_{t=1}^{T} \ell_t^M(\boldsymbol{\gamma}^{(t)}) - \min_{\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathcal{K}} \sum_{t=1}^{T} \mathbb{E}[f_t(\boldsymbol{\alpha}, \boldsymbol{\beta})] = O\left(DG(T)\sqrt{T}\right)$$

Remark 1: Note that plugging in the ARMA transformation and ARIMA transformation in Table 1 to Algorithm 1 recovers ARMA-OGD as presented in [1] and ARIMA-OGD as presented in [10], respectively.

For the following remarks, assume that ℓ_t is squared loss, the DGP is a SARIMA process, and $|x_t| < C(t) =$

Table 2: Regret Bounds for Different Transformations

$\tau(x_t)$	Regret Bound
x_t	$O\left(M^2 \log^2(T)\sqrt{T}\right)$
$\Delta^d x_t$	$O\left(M^2\sqrt{T}\right)$
$\Delta^d \Delta_s^D x_t$	$O\left(M^2\sqrt{T}\right)$

$O(\log t)$ (note that the log transformation is commonly employed as a variance stabilizer in many time series domains).

Remark 2: With these assumptions, Table 2 shows the regret bounds obtained by using different transformations. Recall that a SARIMA process is also an ARIMA process and an ARMA process, thus plugging in the different transforms result in valid applications of Theorem 3.1 when the DGP is a SARIMA process. The differencing transforms remove any growth trends in the data; as a consequence the transformed time series is bounded by a constant. In our case this implies $|\Delta^d x_t|, |\Delta^d \Delta_s^d x_t| < C_\Delta$ (a constant), which leads to an improvement over the regret bound obtained from ARMA-OGD (no transform) of [1]. This improvement can be seen in the empirical results section of [10].

Remark 3: When the DGP is assumed to be SARIMA, we require that $l_a = p+Ps, l_m = q+Qs$, i.e. l_a, l_m both need to essentially be a multiplicative factor larger than s. This affects the length of the required AR approximation M as described in line 1 of Algorithm 1.

Remark 4: Table 2 suggests that using the ARIMA transformation gives the same regret as when using the SARIMA transformation. However, in this case the SARIMA transformation empirically (Section 6) outperforms the ARIMA transformation. As such, we believe that these results do not paint a complete picture as to why we achieve faster empirical convergence and a more data dependent phenomenon is perhaps at play here. In Section 5 we explore a data dependent analysis of FTL.

4 Error Corrected VARMA Models

Online prediction using multivariate non-stationary models present an additional difficulty due to the notion of cointegration (Section 2). Since the cointegrating relationship is unknown, we need to additionally estimate a low rank cointegrating matrix in order to accurately adapt to the underlying DGP and make predictions.

Due to this unknown error correction term (Eq. 5), we cannot directly plug in the EC-VARMA transform into the TSP-OGD Framework (generalized to the multivariate setting). We instead introduce a multivariate variant of TSP-OGD for potentially cointegrated time series that simultaneously updates both the cointegrating and the approximating VAR matrix parameters.

4.1 Approximating an EC-VARMA Process

Given that an EC-VARMA process starts at some fixed time $t = 0$ with fixed initial values, we can write Eq. 5 in a pure EC-VAR form [13]:

$$\Delta \mathbf{x}_t = \Pi^* \mathbf{x}_{t-1} + \sum_{i=1}^{t-1} \Gamma_i^* \Delta \mathbf{x}_{t-i} + \boldsymbol{\varepsilon}_t, \quad t \in \mathbb{N} \tag{9}$$

This allows us to approximate an EC-VARMA process with an EC-VAR model. To use EC-VARMA as a DGP in Algorithm 1, we generalize Algorithm 1 to the multivariate setting and edit line 5 to be:

$$\Delta \tilde{\mathbf{x}}_t = \tilde{\Pi} \mathbf{x}_{t-1} + \sum_{i=1}^{M} \tilde{\Gamma}_i \Delta \mathbf{x}_{t-i}$$

where $\boldsymbol{\gamma} = \{\tilde{\Pi}, \tilde{\Gamma}_1, \ldots, \tilde{\Gamma}_M\}$ are the approximating EC-VAR parameters.

4.2 Online Prediction for EC-VARMA Models

We generalize the assumptions U1-U4 to the multivariate setting:

M1) The noise sequence $\boldsymbol{\varepsilon}_t$ of the underlying VARMA process is independent. Also, it satisfies that $\mathbb{E}[\|\boldsymbol{\varepsilon}_t\|_2] < M_{\max} < \infty$.

M2) We overload notation for the vector case and let $\ell_t : \mathbb{R}^{2k} \to \mathbb{R}$ be a convex loss function with Lipschitz with constant $L > 0$.

M3) We assume the companion matrix $\mathbf{F}$ of the MA lag polynomial is diagonalizable. $\lambda_{\max}$ and κ are the same as in assumption U4.

We refer to the extension of Algorithm 1 as EC-VARMA-OGD. The setup of this algorithm is the same as described in Section 3. We overload more notation

Algorithm 2 EC-VARMA-OGD

Require: DGP parameters p, q. Horizon T. Learning rate η. Data: $\{\mathbf{x}_t\}$.

Set $M = \log_{\lambda_{\max}}\left(\left(2\kappa TLM_{\max}\sqrt{q}\right)^{-1}\right) + p$

Choose $\boldsymbol{\gamma}^{(1)} \in \mathcal{E}$ arbitrarily.

for $t = 1$ to T **do**

  Predict $\tilde{\mathbf{x}}_t = \mathbf{x}_{t-1} + \tilde{\Pi}\mathbf{x}_{t-1} + \sum_{i=1}^{M} \tilde{\Gamma}_i \Delta\mathbf{x}_{t-i}$

  Observe $\mathbf{x}_t$ and receive loss $\ell_t^M\left(\boldsymbol{\gamma}^{(t)}\right)$

  Set $\tilde{\Gamma}_i^{(t+1)} = \Pi_{\mathcal{E}_\Gamma}\left(\hat{\Gamma}_i - \eta\nabla_{\Gamma_i}\ell_t^M\left(\boldsymbol{\gamma}^{(t)}\right)\right)$, for all i

  Set $\tilde{\Pi}^{(t+1)} = \Pi_{\mathcal{B}(*,\rho)}\left(\hat{\Pi} - \eta_t\nabla_{\Pi}\ell_t^M\left(\boldsymbol{\gamma}^{(t)}\right)\right)$

end for

to generalize Equations 6 and 7:

$$\ell_t^M(\boldsymbol{\gamma}) = \ell_t\left(\mathbf{x}_t,\ \mathbf{x}_{t-1} + \tilde{\Pi}\mathbf{x}_{t-1} + \sum_{i=1}^{M}\tilde{\Gamma}_i\Delta\mathbf{x}_{t-i}\right) \tag{10}$$

$$f_t(\Pi, \Gamma, \boldsymbol{\Theta}) = \ell_t\left(\mathbf{x}_t, \mathbf{x}_{t-1} + \Pi\mathbf{x}_{t-1} + \sum_{i=1}^{p-1}\Gamma_i\Delta\mathbf{x}_{t-i} + \sum_{i=1}^{q}\boldsymbol{\Theta}_i\boldsymbol{\varepsilon}_{t-i}\right) \tag{11}$$

The regret as defined in Eq. 8 can be generalized to

$$\text{Regret} = \sum_{t=1}^{T}\ell_t^M(\boldsymbol{\gamma}_t) - \min_{\Pi,\Gamma,\boldsymbol{\Theta}\in\mathcal{K}}\sum_{t=1}^{T}\mathbb{E}[f_t(\Pi,\Gamma,\boldsymbol{\Theta})] \tag{12}$$

where $\mathcal{K}$ is the set of invertible EC-VARMA processes.

To encourage $\tilde{\Pi}$ to be low rank, we project it onto $\mathcal{B}(*,\rho)$, which is the nuclear norm ball of radius ρ. This involves projecting the singular values of $\tilde{\Pi}$ onto an ℓ_1-ball and can be efficiently done [4]. In our framework, this is handled by letting the convex set $\mathcal{E}$ be described as $\{\boldsymbol{\gamma} : \|\tilde{\Pi}\|_* \le \rho, \|\tilde{\Gamma}_i\|_{\max} \le 1,\ i = 1,\dots,M\}$ and plugging it into OGD where projections are made at each iteration. For convenience of notation, let $\mathcal{E}_\Gamma = \{\tilde{\Gamma} : \|\tilde{\Gamma}_i\|_{\max} \le 1, i = 1,\dots M\}$. As in Section 3, $\mathcal{E}$ should be chosen to be large enough to encompass a valid approximation to the true DGP. In practice one will choose ρ and $\mathcal{E}_\Gamma$ to be something simple.

The resulting algorithm is summarized in Algorithm 2. We present the following regret bound:

Theorem 4.1. *Let $\eta = \frac{D}{G(T)\sqrt{T}}$. Then for any data sequence $\{\mathbf{x}_t\}_{t=1}^T$ that satisfies assumptions M1-M3, Algorithm 2 generates a sequence $\{\boldsymbol{\gamma}_t\}$ in which*

$$\sum_{t=1}^{T}\ell_t^M(\boldsymbol{\gamma}_t) - \min_{\Pi,\Gamma,\boldsymbol{\Theta}\in\mathcal{K}}\sum_{t=1}^{T}\mathbb{E}[f_t(\Pi,\Gamma,\boldsymbol{\Theta})] = O\left(DG(T)\sqrt{T}\right) \tag{13}$$

For the remainder of the section, we assume that ℓ_t is the squared loss and $\|\mathbf{x}_t\|_2 < C(t) = O(\log t)$.

Remark 1: With the above assumptions, the resulting regret bound of EC-VARMA-OGD is $O\left(k^2M^2\log^2(T)\sqrt{T}\right)$.

Remark 2: By setting $\rho = 0$ and using $\mathbf{x}_t$ in place of $\Delta\mathbf{x}_t$ (i.e. not differencing) in Algorithm 2, we can use a VARMA process as the DGP and achieve an equivalent regret bound as in the previous remark. Denote this adaptation as VARMA-OGD. However, if the DGP is EC-VARMA, we expect this to empirically perform worse than EC-VARMA-OGD since the latter exploits a valid transformation of the data.

Remark 3: Assume that the DGP is an EC-VARMA process and $\rho = o(1/\log^2(T))$. Then the regret bound obtained is $O\left(k^2M^2\sqrt{T}\right)$. In Section 6, we find that this choice of ρ works well empirically.

5 Data Dependent Regret Bounds

The transformations discussed in the previous sections essentially diminish the effect of serial correlation in the data due to any existing trends. However, our regret bounds do not account for this adjustment. We conjecture that these bounds are missing data-dependent terms that capture correlations inherent in many non-stationary time series. To give a flavor of what a satisfactory data dependent regret bound might look like, we analyze the regret for the FTL algorithm for the case of least squares loss and show that these bounds depend on a data dependent term.

For simplicity, we consider the univariate case. Specifically we analyze FTL with squared loss:

$$\ell_t(\boldsymbol{\gamma}) = \frac{1}{2}(x_t - \boldsymbol{\gamma}^\intercal\boldsymbol{\psi}_t)^2 \tag{14}$$

We will look at the standard notion of regret, thus the result in this section is much more general than time series prediction and can be applied to general regression problems.

The FTL algorithm follows a simple update [14]:

$$\boldsymbol{\gamma}_{t+1} \in \operatorname*{argmin}_{\boldsymbol{\gamma}}\sum_{i=1}^{t}\ell_t(\boldsymbol{\gamma}) \tag{15}$$

Plugging Eq. 14 in Eq. 15 reveals that the FTL algorithm for least squares loss is just recursive least squares (RLS). Using the relevant RLS update equations [11, 8], we present a data dependent regret bound for FTL with least squares loss.

Theorem 5.1. *Let $\ell_t(\boldsymbol{\gamma})$ be defined in Eq. 14 with*

Lipschitz constant $L > 0$. Then FTL generates a sequence $\{\boldsymbol{\gamma}_t\}$ in which

$$\sum_{t=1}^{T} \ell_t(\boldsymbol{\gamma}_t) - \min_{\boldsymbol{\gamma}} \sum_{t=1}^{T} \ell_t(\boldsymbol{\gamma}) = O\left(\sum_{t=1}^{T} \frac{1}{t\lambda_{\min}(t)}\right)$$

where

$$\lambda_{\min}(t) = \lambda_{\min}\left(\frac{1}{t}\sum_{i=1}^{t} \boldsymbol{\psi}_i \boldsymbol{\psi}_i^{\intercal}\right).$$

At the heart of our framework in Section 3, we are approximating an ARMA process with an AR model. In order to apply Theorem 5.1 to our time series prediction setting for DGPs as described in assumption U1 in Section 3, we use FTL and least squares loss to predict the underlying ARMA process $\tau(x_t)$ with an AR model $\boldsymbol{\gamma}^{\intercal}\tau(\boldsymbol{\xi}_t)$, where $\boldsymbol{\xi}_t = \begin{bmatrix} x_{t-1} & \dots & x_{t-M} \end{bmatrix}^{\intercal}$ and $\tau(\boldsymbol{\xi}_t) = \begin{bmatrix} \tau(x_{t-1}) & \dots & \tau(x_{t-M}) \end{bmatrix}^{\intercal}$. This results in $\lambda_{\min}(t) = \left(\frac{1}{t}\sum_{i=1}^{t} \tau(\boldsymbol{\xi}_i)\tau(\boldsymbol{\xi}_i)^{\intercal}\right)$, which is the empirical non-centered autocovariance of the transformed data. Ideally, we want this quantity to be large, which implies that each direction has a lot of information in it. If this quantity is small, then there are directions where the variance of $\tau(\boldsymbol{\xi}_t)$ is small, meaning the individual samples $\tau(x_t)$ may be highly correlated.

To empirically assess the regret bound across the different spectrum of non-stationarities, we calculate the bound $\sum_{i=1}^{T} 1/(t\lambda_{\min}(t))$ for the three transforms in Table 1. We simulated a SARIMA process 50 times with $T = 10,000$. We then averaged the calculated regret bound across the 50 datasets using the raw data, trend adjusted data, and seasonal/trend adjusted data (i.e. using each transformation). The result is shown in Figure 1. The transformations essentially decrease correlations making the data more like realizations of a stationary ARMA process; we can see that accounting for the appropriate non-stationarities results in tighter regret bounds.

6 Empirical Results

In this section, we show empirically the impact of transformations and methods described in Sections 3 and 4 to synthetic and real datasets. In each scenario, we consider squared loss and plot the log average squared loss vs. iteration. For all experiments, we set $\mathcal{E} = \{\boldsymbol{\gamma} : \|\boldsymbol{\gamma}\|_{\max} \leq 1\}$ and initialize all parameters to 0. For all real world datasets, we log transform the time series. Plots of these datasets can be found in the Appendix in the supplementary material.

6.1 Accounting for Non-stationarities

As described in Section 3, we show that using transformations accounting for appropriate non-stationarities

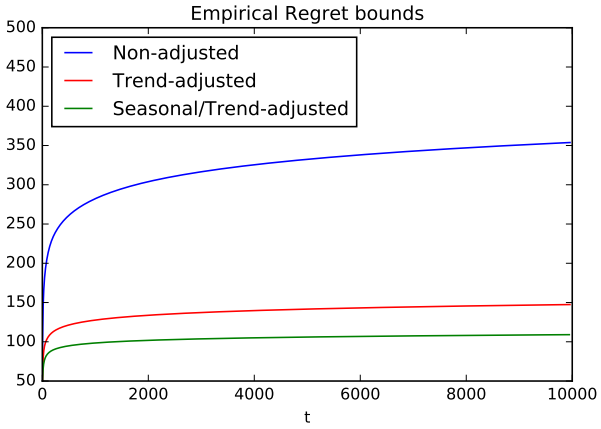


Figure 1: Empirical Regret Bounds for Transformed Data

results in faster empirical convergence. For Algorithm 1, when plugging in the transformation functions shown in Table 1, we fix all other parameters in the algorithm. Recall that M to be at least a multiplicative factor larger than s. We set $M = 2s$ and $d = 1$ for each dataset.

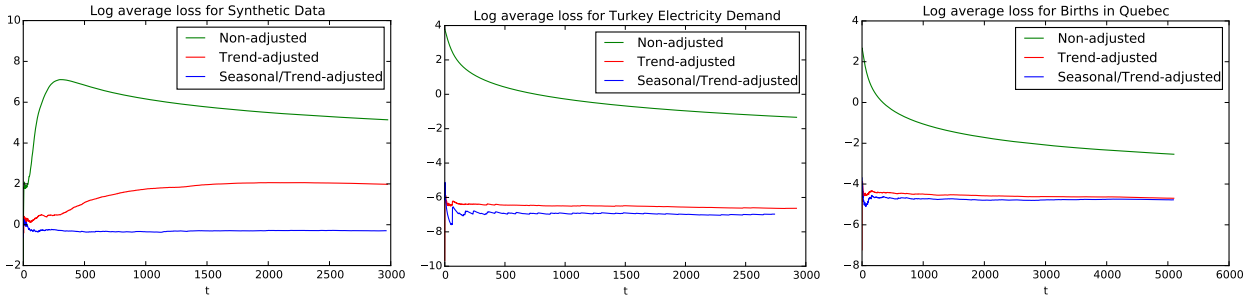
We first simulate a synthetic time series with $T = 3,000$ from the following SARIMA model (obtained from fitting the airlines time series of [5]):

$$\Delta\Delta_{12} x_t = (1 - 0.38L)(1 - 0.57L^{12})\epsilon_t \qquad (16)$$

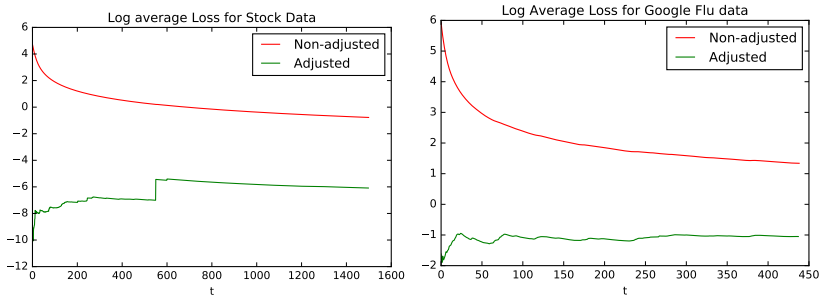
We run Algorithm 1 on the generated series for each transformation of the data. We plotted the log average loss in Figure 2a. As expected, accounting for the appropriate non-stationarities results in faster convergence. Note that the seasonally/trend-adjusted data convergences almost instantly.

Next, we consider a dataset that contains daily electricity demand in Turkey from January 1, 2000, to December 31, 2008. The seasonality in this dataset is biannual (rounded down to $s = 182$ days). This dataset exhibits seasonality and an upwards trend, making it sutable for being modeled by a SARIMA process. The results are shown in Figure 2b. Performing any type of differencing results in faster convergence compared to the non-adjusted transform. Recognizing the trend but ignoring the seasonal trend results in slower convergence compared to recognizing both types of trends.

Lastly, we consider a dataset that contains daily recorded births in Quebec from Jan. 01, 1977 to Dec. 31, 1990. There is a weekly seasonality pattern with $s = 7$. The results are shown in Figure 2c. As shown in the previous example, accounting for any non-stationarity results in faster convergence. However, accounting for the seasonal trend on top of the



(a) Synthetic data (b) Turkey electricity demand (c) Births in Quebec



(d) Stock data (e) Google flu data

Figure 2: Empirical results. The top line has results for univariate data, and the bottom line has results for multivariate data.

trend results in little improvement. Note that ARIMA processes can model seasonality just as well, thus a seasonal tranform is not guaranteed to always improve convergence even when the data does exhibit such patterns. Normally the differencing orders are determined by statistical tests. See [5] for details.

6.2 Multivariate Algorithms

In the multivariate setting we show empirically that accounting for cointegration results in faster convergence. We look at the results of running EC-VARMA-OGD (adjusted) as described in Algorithm 2 vs. VARMA-OGD (non-adjusted) on two real datasets.

We collected 7 time series of stock prices from Yahoo Finance (`http://finance.yahoo.com/`) of large technology companies, and also includes the S&P500 index. By including the S&P500, which is essentially an weighted average of 500 company stock prices, we have partially introduced cointegration into the time series. We set $M = 10, \rho = 0.5$ and ran both algorithms with the resulting plots in Figure 2d. As expected, accounting for cointegration results in better performance. There is a bump in the convergence plot due to a spike in the data (Appendix).

We also evaluated the algorithms on the Google Flu dataset (`https://www.google.com/publicdata/explore/`). We looked at the influenza rates of 28 countries. Plotting them, one can see two distinct seasonality patterns: the northern hemisphere countries have flu incidents that peak in one part of the year while the southern hemisphere countries have flu incidents that peak in the other part of the year. Thus it makes sense to believe that the time series exhibits a cointegrated relationship. This dataset exhibits yearly seasonality (52 weeks), thus we set $M = 60$ to be larger than one seasonal period. We choose $\rho = 0.5$ and plot the results are given in Figure 2e. We see that adjusting for the cointegration dramatically increases performance.

7 Conclusions and Future Work

We developed a framework to account for non-stationary artifacts in both univariate and multivariate time series. We observed in the empirical results section that this leads to faster convergence. Speculating that accounting for non-stationary artifacts like a trend and/or seasonality reduces correlation in the data, we presented a data-dependent bound for FTL for squared loss. In future work, we plan to explore online algorithms that can give data dependent bounds.

References

[1] O. Anava, E. Hzan, S. Mannor, and O. Shamir. Online learning for time series prediction. In *JMLR: Workshop and Conference Proceedings of Conference on Learning Theory*, volume 13, 2013.

[2] P. Brockwell and R. Davis. *Time Series: Theory and Methods*. Springer, 2009.

[3] N. Cesa-Bianchi and G. Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[4] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra. Efficient projections onto the l 1-ball for learning in high dimensions. In *Proceedings of the 25th international conference on Machine learning*, pages 272–279. ACM, 2008.

[5] G. J. George Box and G. Reinsel. *Time Series Analysis: Forecasting and Control*. Prentice-Hall, 1994.

[6] J. D. Hamilton. Time series analysis princeton university press. *Princeton, NJ*, 1994.

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

[8] T. L. Lai and C. Z. Wei. Least squares estimates in stochastic regression models with applications to identification and control of dynamic systems. *The Annals of Statistics*, pages 154–166, 1982.

[9] P. Liang. Cs229t/stat231: Statistical learning theory (winter 2014).

[10] C. Liu, S. C. Hoi, P. Zhao, and J. Sun. Online arima algorithms for time series prediction. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[11] L. Ljung. System identification. In *Signal Analysis and Prediction*, pages 163–173. Springer, 1998.

[12] H. Lütkepohl. *New introduction to multiple time series analysis*. Springer Science & Business Media, 2005.

[13] H. Lütkepohl. Forecasting with varma models. *Handbook of economic forecasting*, 1:287–325, 2006.

[14] S. Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2011.

[15] D. Thompson. Jackknifing multiple-window spectra. In *Proceedings of the 6th ICASSP*, pages 73–76, 1994.

[16] R. Tsay. *Multivariate Time Series Analysis: With R and Financial Applications*. Wiley, 2013.

[17] M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *ICML*, 2003.

8 Appendix

8.1 Proof of Theorem 3.1

We reproduce the proof given in Anava et al. [1] and Liu et al. [10] using our transformation notation, and with the more natural and relaxed assumption of invertibility of the MA process.

Proof. **Step 1**: Assume that $\zeta(\tilde{x}_t)$ is a linear function such as the ones given in Table 1. Then $\{\ell_t^M\}$ are convex loss functions, and we may invoke [17] with a fixed step size $\eta = \frac{D}{G(T)\sqrt{T}}$:

$$\sum_{t=1}^{T} \ell_t^M(\boldsymbol{\gamma}_t) - \min_{\boldsymbol{\gamma}} \sum_{t=1}^{T} \ell_t^M(\boldsymbol{\gamma}) = O\left(DG(T)\sqrt{T}\right)$$

Note that the proof in [17] uses a constant upper bound G on the gradients. Since we assume $G(T)$ is a monotonically increasing function, the proof in [17] follows through straightforwardly.

Step 2: Let $\boldsymbol{\alpha}, \boldsymbol{\beta}$ denote the parameters of the underlying ARMA(l_a, l_m) process. We define a few things:

$$\begin{aligned}\tau\left(x_t^{\infty}(\boldsymbol{\alpha},\boldsymbol{\beta})\right) &= \sum_{i=1}^{l_a} \alpha_i \tau\left(x_{t-i}\right) + \sum_{i=1}^{l_m} \beta_i \left(\tau\left(x_{t-i}\right) - \tau\left(x_{t-i}^{\infty}(\boldsymbol{\alpha},\boldsymbol{\beta})\right)\right) \\ x_t^{\infty}(\boldsymbol{\alpha},\boldsymbol{\beta}) &= \zeta\left(\tau\left(x_t^{\infty}(\boldsymbol{\alpha},\boldsymbol{\beta})\right)\right)\end{aligned}$$

with initial condition $\tau\left(x_t^{\infty}(\boldsymbol{\alpha},\boldsymbol{\beta})\right) = \tau\left(x_t\right)$ for $t < 0$. For convenience, assume that we have fixed data $x_0, \ldots, x_{-h}$ so that $\tau(x_0), \ldots, \tau(x_{-l_a})$ exists. Denote

$$f_t^{\infty}(\boldsymbol{\alpha},\boldsymbol{\beta}) = \ell_t\left(x_t, x_t^{\infty}(\boldsymbol{\alpha},\boldsymbol{\beta})\right)$$

With this definition, we can write $\tau\left(x_t^{\infty}(\boldsymbol{\alpha},\boldsymbol{\beta})\right) = \sum_{i=1}^{t+l_a} c_i(\boldsymbol{\alpha},\boldsymbol{\beta})\tau\left(x_{t-i}\right)$, i.e. as a growing AR process. Next, we define

$$\begin{aligned}\tau\left(x_t^{m}(\boldsymbol{\alpha},\boldsymbol{\beta})\right) &= \sum_{i=1}^{l_a} \alpha_i \tau\left(x_{t-i}\right) + \sum_{i=1}^{l_m} \beta_i \left(\tau\left(x_{t-i}\right) - \tau\left(x_{t-i}^{m-i}(\boldsymbol{\alpha},\boldsymbol{\beta})\right)\right) \\ x_t^{m}(\boldsymbol{\alpha},\boldsymbol{\beta}) &= \zeta\left(\tau\left(x_t^{m}(\boldsymbol{\alpha},\boldsymbol{\beta})\right)\right)\end{aligned}$$

with initial condition $\tau\left(x_t^{m}(\boldsymbol{\alpha},\boldsymbol{\beta})\right) = \tau\left(x_t\right)$ for $m < 0$. We relate M and m with this relation: $M = m + l_a$. With this definition, we can write $\tau\left(x_t^{m}(\boldsymbol{\alpha},\boldsymbol{\beta})\right) = \sum_{i=1}^{M} \tilde{c}_i(\boldsymbol{\alpha},\boldsymbol{\beta})\tau\left(x_{t-i}\right)$, i.e. as a fixed length AR process. Denote

$$f_t^{m}(\boldsymbol{\alpha},\boldsymbol{\beta}) = \ell_t\left(x_t, x_t^{m}(\boldsymbol{\alpha},\boldsymbol{\beta})\right)$$

Let $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = \operatorname{argmin}_{\boldsymbol{\alpha},\boldsymbol{\beta}\in\mathcal{K}} \sum_{t=1}^{T} \mathbb{E}\left[f_t(\boldsymbol{\alpha},\boldsymbol{\beta})\right]$. Recall that the only random part of the expectation is ε_t. x_t is fixed in this quantity.

Lemma 8.1.1 gives us that

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

Lemma 8.1.3 says that choosing $m = \log_{\lambda_{\max}}\left(\left(2\kappa TLM_{\max}\sqrt{l_m}\right)^{-1}\right)$ results in

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

Lemma 8.1.2 gives us that

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

Chaining all of these gives us the final result:

$$\sum_{t=1}^{T} \ell_t^m(\boldsymbol{\gamma}_t) - \min_{\boldsymbol{\alpha},\boldsymbol{\beta}\in\mathcal{K}} \sum_{t=1}^{T} \mathbb{E}[f_t(\boldsymbol{\alpha},\boldsymbol{\beta})] = O\left(DG(T)\sqrt{T}\right)$$

□

Lemma 8.1.1. *For all m and $\{x_t\}$ that satisfies the assumptions U1-U4, we have that*

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

Proof. We simply set $\gamma_i^* = \tilde{c}_i(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)$ and get $\sum_{t=1}^{T} \ell_t^m(\boldsymbol{\gamma}^*) = \sum_{t=1}^{T} f_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)$. Thus, the minimum holds trivially. Note that we assume $\boldsymbol{\gamma}^* \in \mathcal{E}$. □

Lemma 8.1.2. *For any data sequence $\{x_t\}$ that satisfies the assumptions U1-U4, it holds that*

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

Proof. Let $(\boldsymbol{\alpha}',\boldsymbol{\beta}')$ denote the parameters that generated the signal. Thus,

$$f_t(\boldsymbol{\alpha}',\boldsymbol{\beta}') = \ell_t(x_t, x_t - \varepsilon_t)$$

for all t. Since ε_t is independent of $\varepsilon_1,\ldots,\varepsilon_{t-1}$, the best prediction at time t will cause a loss of at least $\mathbb{E}[\ell_t(x_t, x_t-\varepsilon_t)]$. Since $\mathbb{E}[\varepsilon_t] = 0$ and ℓ_t is convex, it follows that $(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*) = (\boldsymbol{\alpha}',\boldsymbol{\beta}')$ and that

$$f_t(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*) = \ell_t(x_t, x_t - \varepsilon_t)$$

We define a few things first. Let

$$y_t = \tau(x_t) - \tau(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)) - \varepsilon_t, \qquad \mathbf{y}_t = \begin{bmatrix} y_t \\ y_{t-1} \\ \vdots \\ y_{t-q+1} \end{bmatrix}$$

WLOG (and by assumption), we can assume that $\mathbb{E}[\|\mathbf{y}_0\|_2] \leq \rho$, where ρ is some positive constant. Next we show that

$$\mathbb{E}[|y_t|] = \mathbb{E}[|\tau(x_t) - \tau(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)) - \varepsilon_t|] \leq \kappa\lambda_{\max}^t\rho$$

We have that

$$\begin{aligned}
\tau(x_t) - \tau(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)) - \varepsilon_t &= \sum_{i=1}^{l_a} \alpha_i^* \tau(x_{t-i}) + \sum_{i=1}^{l_m} \beta_i^* \varepsilon_{t-i} + \varepsilon_t \\
&\quad - \sum_{i=1}^{l_a} \alpha_i^* \tau(x_{t-i}) - \sum_{i=1}^{l_m} \beta_i^* \left(\tau(x_{t-i}) - \tau\left(x_{t-i}^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)\right) - \varepsilon_t \\
&= -\sum_{i=1}^{l_m} \beta_i^* \left(\tau(x_{t-i}) - \tau\left(x_{t-i}^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right) - \varepsilon_{t-i}\right)
\end{aligned}$$

which shows that $y_t = -\sum_{i=1}^{l_m} \beta_i^* y_{t-i}$. The companion matrix to this difference equation is exactly $\mathbf{F}$ as defined in Eq. 2. Thus,

$$\mathbf{y}_t = \mathbf{F}\mathbf{y}_{t-1}$$

Next, we note that

$$\begin{aligned}
|y_t| &\leq \|\mathbf{y}_t\|_2 = \|\mathbf{F}\mathbf{y}_{t-1}\|_2 \\
&= \|\mathbf{F}^2\mathbf{y}_{t-2}\|_2 \\
&= \|\mathbf{F}^t\mathbf{y}_0\|_2 \\
&= \|\mathbf{T}\Lambda^t\mathbf{T}^{-1}\mathbf{y}_0\|_2 \\
&\leq \|\mathbf{T}\|_2\|\mathbf{T}^{-1}\|_2\|\Lambda^t\|_2\|\mathbf{y}_0\|_2 \\
&= \frac{\sigma_{\max}(\mathbf{T})}{\sigma_{\min}(\mathbf{T})}\lambda_{\max}^t\|\mathbf{y}_0\|_2 \\
&\leq \kappa\lambda_{\max}^t\|\mathbf{y}_0\|_2
\end{aligned}$$

Taking the expectation gives us $\mathbb{E}[|y_t|] \leq \kappa\lambda_{\max}^t\mathbb{E}\left[\|\mathbf{y}_0\|_2\right] \leq \kappa\lambda_{\max}^t\rho$.

Now we combine this with the Lipschitz continuity of ℓ_t to get

$$\begin{aligned}
|\mathbb{E}\left[f_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right] - \mathbb{E}\left[f_t(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right]| &= |\mathbb{E}\left[\ell_t(x_t, x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*))\right] - \mathbb{E}\left[\ell_t(x_t, x_t-\varepsilon_t)\right]| \\
&\leq \mathbb{E}\left[|\ell_t(x_t, x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)) - \ell_t(x_t, x_t-\varepsilon_t)|\right] \\
&\leq L\cdot\mathbb{E}\left[|x_t - x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*) - \varepsilon_t|\right] \\
&= L\cdot\mathbb{E}\left[|\tau\left(x_t\right) - \tau\left(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right) - \varepsilon_t|\right] \\
&\leq \kappa L\rho\lambda_{\max}^t
\end{aligned}$$

where we used Jensen's inequality in the first inequality. Note that we also assume $x_t - \tilde{x}_t = \zeta(\tau(x_t)) - \zeta(\tau(\tilde{x}_t)) = \tau(x_t) - \tau(\tilde{x}_t)$. This holds true for the transformations given in Table 1. Summing this from $t=1$ to T gives us the result. □

Lemma 8.1.3. *For any data sequence $\{x_t\}$ that satisfies the assumptions U1-U4, it holds that*

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

if we choose $m = \log_{\lambda_{\max}}\left((2\kappa TLM_{\max}\sqrt{l_m})^{-1}\right)$.

Proof. Fix t. Note that for $m < 0$,

$$\begin{aligned}
|\tau\left(x_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right) - \tau\left(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)| &= |\tau\left(x_t\right) - \tau\left(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)| \\
&\leq |\tau\left(x_t\right) - \tau\left(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right) - \varepsilon_t| + |\varepsilon_t|
\end{aligned}$$

The right hand side of the inequality is simply $|y_t| + |\varepsilon_t|$, where y_t is as defined in Lemma 8.1.2. By assumption, $\mathbb{E}[|\varepsilon_t|] < M_{\max}$. Assume that $M_{\max}$ is large enough such that $\mathbb{E}[|y_t|] \leq M_{\max}$. This is a valid assumption since it is decaying exponentially as proved in Lemma 8.1.2. It is important to note that $\tau\left(x_t^m(\boldsymbol{\alpha},\boldsymbol{\beta})\right)$ and $\tau\left(x_t^\infty(\boldsymbol{\alpha},\boldsymbol{\beta})\right)$ have no randomness in them since τ is deterministic. Thus, for $m < 0$,

$$\begin{aligned}
|\tau\left(x_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right) - \tau\left(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)| &= \mathbb{E}\left[|\tau\left(x_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right) - \tau\left(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)|\right] \\
&= \mathbb{E}\left[|\tau\left(x_t\right) - \tau\left(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)|\right] \\
&\leq \mathbb{E}[|y_t| + |\varepsilon_t|] \\
&\leq 2M_{\max}
\end{aligned}$$

Squaring both sides of the inequality results in

$$\left(\tau\left(x_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right) - \tau\left(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)\right)^2 \leq 4M_{\max}^2$$

Next, we define

$$z_t^m = \tau\left(x_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right) - \tau\left(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right), \qquad \mathbf{z}_t^m = \begin{bmatrix} z_t^m \\ z_{t-1}^{m-1} \\ \vdots \\ z_{t-q+1}^{m-q+1} \end{bmatrix}$$

We have that

$$\begin{aligned}\tau\left(x_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)-\tau\left(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right) &= \sum_{i=1}^{l_a}\alpha_i^*\tau\left(x_{t-i}\right)+\sum_{i=1}^{l_m}\beta_i^*\left(\tau\left(x_{t-i}\right)-\tau\left(x_{t-i}^{m-i}(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)\right)\\ &\quad-\sum_{i=1}^{l_a}\alpha_i^*\tau\left(x_{t-i}\right)-\sum_{i=1}^{l_m}\beta_i^*\left(\tau\left(x_{t-i}\right)-\tau\left(x_{t-i}^{\infty}(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)\right)\\ &= -\sum_{i=1}^{l_m}\beta_i^*\left(\tau\left(x_{t-i}^{m-i}(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)-\tau\left(x_{t-i}^{\infty}(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)\right)\end{aligned}$$

Thus, $z_t^m = -\sum_{i=1}^{l_m}\beta_i^* z_{t-i}^{m-i}$. The companion matrix to this difference equation is exactly $\mathbf{F}$ as defined in Eq. 2. Thus,

$$\mathbf{z}_t^m = \mathbf{F}\mathbf{z}_{t-1}^{m-1}$$

We have that

$$\begin{aligned}|z_t^m| &\le \|\mathbf{z}_t^m\|_2 = \|\mathbf{F}\mathbf{z}_{t-1}^{m-1}\|_2\\ &= \|\mathbf{F}^2\mathbf{z}_{t-2}^{m-2}\|_2\\ &= \|\mathbf{F}^m\mathbf{z}_{t-m}^{0}\|_2\\ &= \|\mathbf{T}\Lambda^m\mathbf{T}^{-1}\mathbf{z}_{t-m}^{0}\|_2\\ &\le \|\mathbf{T}\|_2\|\mathbf{T}^{-1}\|_2\|\Lambda^m\|_2\|\mathbf{z}_{t-m}^{0}\|_2\\ &= \frac{\sigma_{\max}(\mathbf{T})}{\sigma_{\min}(\mathbf{T})}\lambda_{\max}^m\sqrt{\sum_{i=0}^{l_m-1}(z_{t-m-i}^{-i})^2}\\ &\le \kappa\lambda_{\max}^m\sqrt{q4M_{\max}^2}\\ &= \kappa\lambda_{\max}^m 2M_{\max}\sqrt{l_m}\end{aligned}$$

Now we combine this with the Lipschitz continuity of ℓ_t to get

$$\begin{aligned}|\mathbb{E}[f_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)]-\mathbb{E}[f_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)]| &= |\mathbb{E}[\ell_t(x_t,x_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*))]-\mathbb{E}[\ell_t(x_t,x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*))]|\\ &\le \mathbb{E}[|\ell_t(x_t,x_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*))-\ell_t(x_t,x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*))|]\\ &\le L\cdot\mathbb{E}[|x_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)-x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)|]\\ &= L\cdot|\tau\left(x_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)-\tau\left(x_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)\right)|\\ &\le 2\kappa LM_{\max}\sqrt{l_m}\lambda_{\max}^m\end{aligned}$$

where in the first inequality we used Jensen's inequality and we again used the assumption that $x_t - \tilde{x}_t = \tau(x_t) - \tau(\tilde{x}_t)$.

Summing this quantity from $t = 1$ to T gives us the result:

$$\left|\sum_{t=1}^{T}\mathbb{E}[f_t^\infty(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)]-\sum_{t=1}^{T}\mathbb{E}[f_t^m(\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)]\right| \le 2\kappa TLM_{\max}\sqrt{l_m}\lambda_{\max}^m$$

Choosing $m = \log_{\lambda_{\max}}\left((2\kappa TLM_{max}\sqrt{l_m})^{-1}\right)$ gives us the desired $O(1)$ property. □

8.2 Proof of Theorem 4.1

Proof. We again produce a proof of very similar structure to Anava et al. [1] and Liu et al. [10]. We first need to redefine a few things for the vector case. Let $D = \sup_{\boldsymbol{\gamma}_1,\boldsymbol{\gamma}_2\in\mathcal{K}}\|\boldsymbol{\gamma}_1-\boldsymbol{\gamma}_2\|_F$, and $\|\nabla_{\boldsymbol{\gamma}}\ell_t^m(\boldsymbol{\gamma})\|_F \le G(T)$.

Step 1: Since $\{\ell_t^M\}$ are convex loss functions, we may invoke [17] with a fixed step size $\eta = \frac{D}{G(T)\sqrt{T}}$:

$$\sum_{t=1}^{T}\ell_t^M(\boldsymbol{\gamma}_t)-\min_{\boldsymbol{\gamma}}\sum_{t=1}^{T}\ell_t^M(\boldsymbol{\gamma}) = O\left(DG(T)\sqrt{T}\right)$$

Again, we note that the proof in [17] uses a constant upper bound G on the gradients. Since we assume $G(T)$ is a monotonically increasing function, the proof in [17] follows through straightforwardly.

Step 2: Next we define a few things in the same vein as in the proof of Theorem 3.1. Let

$$\begin{aligned}\Delta\mathbf{x}_t^\infty(\Pi,\Gamma,\mathbf{\Theta}) &= \Pi\mathbf{x}_{t-1} + \sum_{i=1}^{p-1}\Gamma_i\Delta\mathbf{x}_{t-i} + \sum_{i=1}^{q}\mathbf{\Theta}_i\left(\Delta\mathbf{x}_{t-i} - \Delta\mathbf{x}_{t-i}^\infty(\Pi,\Gamma,\mathbf{\Theta})\right)\\ \mathbf{x}_t^\infty(\Pi,\Gamma,\mathbf{\Theta}) &= \Delta\mathbf{x}_t^\infty(\Pi,\Gamma,\mathbf{\Theta}) + \mathbf{x}_{t-1}\\ f_t^\infty(\Pi,\Gamma,\mathbf{\Theta}) &= \ell_t\left(\mathbf{x}_t,\ \mathbf{x}_t^\infty(\Pi,\Gamma,\mathbf{\Theta})\right)\end{aligned}$$

with initial condition $\Delta\mathbf{x}_t^\infty(\Pi,\Gamma,\mathbf{\Theta}) = \Delta\mathbf{x}_t$ for all $t<0$. Note that we are assuming that we have fixed data $\mathbf{x}_0,\ldots,\mathbf{x}_{-p}$. With this definition, we can write $\Delta\mathbf{x}_t^\infty(\Pi,\Gamma,\mathbf{\Theta}) = c_0(\Pi,\Gamma,\mathbf{\Theta})\mathbf{x}_{t-1} + \sum_{i=1}^{t+p-1}c_i(\Pi,\Gamma,\mathbf{\Theta})\Delta\mathbf{x}_{t-i}$, i.e. as a growing AR process. This is because we can undo the reparameterization and write $\Delta\mathbf{x}_t$ in its original VARMA process form

$$\begin{aligned}\mathbf{x}_t^\infty(\Pi,\Gamma,\Theta) &= \sum_{i=1}^{p}\mathbf{A}_i\mathbf{x}_{t-i} + \sum_{i=1}^{q}\mathbf{\Theta}_i\left(\mathbf{x}_{t-i} - \mathbf{x}_{t-i}^\infty(\Pi,\Gamma,\Theta)\right)\\ &= \sum_{i=1}^{t+p}c_i(\mathbf{A},\Theta)\mathbf{x}_{t-i}\end{aligned}$$

as shown in the proof of Algorithm 1. Using the error corrected reparameterization here results in

$$\Delta\mathbf{x}_t^\infty(\Pi,\Gamma,\Theta) = c_0(\Pi,\Gamma,\Theta)\mathbf{x}_{t-1} + \sum_{i=1}^{t+p-1}c_i(\Pi,\Gamma,\Theta)\Delta\mathbf{x}_{t-i}$$

Furthermore, we define

$$\begin{aligned}\Delta\mathbf{x}_t^m(\Pi,\Gamma,\mathbf{\Theta}) &= \Pi\mathbf{x}_{t-1} + \sum_{i=1}^{p-1}\Gamma_i\Delta\mathbf{x}_{t-i} + \sum_{i=1}^{q}\mathbf{\Theta}_i\left(\Delta\mathbf{x}_{t-i} - \Delta\mathbf{x}_{t-i}^{m-i}(\Pi,\Gamma,\mathbf{\Theta})\right)\\ \mathbf{x}_t^m(\Pi,\Gamma,\mathbf{\Theta}) &= \Delta\mathbf{x}_t^m(\Pi,\Gamma,\mathbf{\Theta}) + \mathbf{x}_{t-1}\\ f_t^m(\Pi,\Gamma,\mathbf{\Theta}) &= \ell_t(\mathbf{x}_t,\ \mathbf{x}_t^m(\Pi,\Gamma,\mathbf{\Theta}))\end{aligned}$$

with initial condition $\Delta\mathbf{x}_t^m(\Pi,\Gamma,\mathbf{\Theta}) = \Delta\mathbf{x}_t$ for all $m<0$. We relate $M = m+p-1$. With this definition, we can write $\Delta\mathbf{x}_t^m(\Pi,\Gamma,\mathbf{\Theta}) = \tilde{c}_0(\Pi,\Gamma,\mathbf{\Theta})\mathbf{x}_{t-1} + \sum_{i=1}^{M}\tilde{c}_i(\Pi,\Gamma,\mathbf{\Theta})\Delta\mathbf{x}_{t-i}$ by using similar rearrangement arguments as shown above.

Lastly, we define

$$(\Pi^*,\Gamma^*,\mathbf{\Theta}^*) = \operatorname*{argmin}_{\Pi,\Gamma,\mathbf{\Theta}}\sum_{t=1}^{T}\mathbb{E}[f_t(\Pi,\Gamma,\mathbf{\Theta})]$$

Recall that $\mathbf{x}_t$ is fixed in the expectation.

Lemma 8.2.1 gives us that

$$\min_{\boldsymbol{\gamma}}\sum_{t=1}^{T}\ell_t^M(\boldsymbol{\gamma}) \le \sum_{t=1}^{T}f_t^m(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)$$

Lemma 8.2.3 says that choosing $m = \log_{\lambda_{\max}}\left(\left(2\kappa TLM_{\max}\sqrt{q}\right)^{-1}\right)$ results in

$$\left|\sum_{t=1}^{T}\mathbb{E}[f_t^m(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)] - \sum_{t=1}^{T}\mathbb{E}[f_t^\infty(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)]\right| = O(1)$$

Lemma 8.2.2 gives us that

$$\left|\sum_{t=1}^{T}\mathbb{E}[f_t^\infty(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)] - \sum_{t=1}^{T}\mathbb{E}[f_t(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)]\right| = O(1)$$

Chaining all of these gives us the final result:

$$\sum_{t=1}^{T} \ell_t^M(\boldsymbol{\gamma}_t) - \min_{\Pi,\Gamma,\boldsymbol{\Theta}} \sum_{t=1}^{T} \mathbb{E}[f_t(\Pi,\Gamma,\boldsymbol{\Theta})] = O\left(DG(T)\sqrt{T}\right)$$

□

Lemma 8.2.1. *For all m and $\{\mathbf{x}_t\}$ that satisfies assumptions M1-M3, we have that*

$$\min_{\boldsymbol{\gamma}} \sum_{t=1}^{T} \ell_t^m(\boldsymbol{\gamma}) \leq \sum_{t=1}^{T} f_t^m(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)$$

Proof. Recall that $\boldsymbol{\gamma} = \{\tilde{\Pi}, \tilde{\Gamma}_i,\ i = 1,\ldots,M\}$ We simply set $\tilde{\Pi}^* = \tilde{c}_0(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*),\ \tilde{\Gamma}_i^* = \tilde{c}_i(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)$ and let that be denoted by $\boldsymbol{\gamma}^*$. Thus, we get $\sum_{t=1}^{T} \ell_t^M(\boldsymbol{\gamma}^*) = \sum_{t=1}^{T} f_t^m(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)$. Thus, the minimum holds trivially. Note that we assume $\boldsymbol{\gamma}^* \in \mathcal{E}$. □

Lemma 8.2.2. *For any data sequence $\{\mathbf{x}_t\}_{t=1}^T$ that satisfies assumptions M1-M5, it holds that*

$$\left| \sum_{t=1}^{T} \mathbb{E}[f_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)] - \sum_{t=1}^{T} \mathbb{E}[f_t(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)] \right| = O(1)$$

Proof. We start the proof in the same exact way that Anava does. Let $(\Pi',\Gamma',\boldsymbol{\Theta}')$ denote the parameters that generated the signal. Thus,

$$f_t(\Pi',\Gamma',\boldsymbol{\Theta}') = \ell_t(\mathbf{x}_t, \mathbf{x}_t - \boldsymbol{\varepsilon}_t)$$

for all t. Since $\boldsymbol{\varepsilon}_t$ is independent of $\boldsymbol{\varepsilon}_1,\ldots,\boldsymbol{\varepsilon}_{t-1}$, the best prediction at time t will cause a loss of at least $\mathbb{E}[\ell_t(\mathbf{x}_t, \mathbf{x}_t - \boldsymbol{\varepsilon}_t)]$. Since $\mathbb{E}[\boldsymbol{\varepsilon}_t] = 0$ and ℓ_t is convex, it follows that $(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) = (\Pi',\Gamma',\boldsymbol{\Theta}')$ and that

$$f_t(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) = \ell_t(\mathbf{x}_t, \mathbf{x}_t - \boldsymbol{\varepsilon}_t)$$

We define a few things first. Let

$$\mathbf{y}_t = \Delta\mathbf{x}_t - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) - \boldsymbol{\varepsilon}_t, \qquad \mathbf{Y}_t = \begin{bmatrix} \mathbf{y}_t \\ \mathbf{y}_{t-1} \\ \vdots \\ \mathbf{y}_{t-q+1} \end{bmatrix}$$

(note the overloading from previous sections) By assumption, we can assume that $\mathbb{E}\left[\|\mathbf{Y}_0\|_2\right] \leq \rho$, where ρ is some positive constant. Next we show that

$$\mathbb{E}[\|\mathbf{y}_t\|_2] = \mathbb{E}\left[\|\Delta\mathbf{x}_t - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) - \boldsymbol{\varepsilon}_t\|_2\right] \leq \kappa\lambda_{\max}^t\rho$$

We have that

$$\begin{aligned}
\Delta\mathbf{x}_t - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) - \boldsymbol{\varepsilon}_t = \quad & \Pi^*\mathbf{x}_{t-1} + \sum_{i=1}^{p-1} \Gamma_i^* \Delta\mathbf{x}_{t-i} + \sum_{i=1}^{q} \boldsymbol{\Theta}_i^* \boldsymbol{\varepsilon}_{t-i} + \boldsymbol{\varepsilon}_t \\
& - \Pi^*\mathbf{x}_{t-1} - \sum_{i=1}^{p-1} \Gamma_i^* \Delta\mathbf{x}_{t-i} - \sum_{i=1}^{q} \boldsymbol{\Theta}_i^* \left(\Delta\mathbf{x}_{t-i} - \Delta\mathbf{x}_{t-i}^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)\right) - \boldsymbol{\varepsilon}_t \\
= & - \sum_{i=1}^{q} \boldsymbol{\Theta}_i^* \left(\Delta\mathbf{x}_{t-i} - \Delta\mathbf{x}_{t-i}^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) - \boldsymbol{\varepsilon}_{t-i}\right)
\end{aligned}$$

which shows that $\mathbf{y}_t = -\sum_{i=1}^{q} \boldsymbol{\Theta}_i^* \mathbf{y}_{t-i}$. The companion matrix to this difference equation is $\mathbf{F}$. Thus,

$$\mathbf{Y}_t = \mathbf{F}\mathbf{Y}_{t-1}$$

Next, we note that

$$\begin{aligned}\|\mathbf{y}_t\|_2 &\leq \|\mathbf{Y}_t\|_2 = \|\mathbf{F}\mathbf{Y}_{t-1}\|_2 \\ &= \|\mathbf{F}^2\mathbf{Y}_{t-2}\|_2 \\ &= \|\mathbf{F}^t\mathbf{Y}_0\|_2 \\ &= \|\mathbf{T}\Lambda^t\mathbf{T}^{-1}\mathbf{Y}_0\|_2 \\ &\leq \|\mathbf{T}\|_2\|\mathbf{T}^{-1}\|_2\|\Lambda^t\|_2\|\mathbf{Y}_0\|_2 \\ &= \frac{\sigma_{\max}(\mathbf{T})}{\sigma_{\min}(\mathbf{T})}\lambda_{\max}^t\|\mathbf{Y}_0\|_2 \\ &\leq \kappa\lambda_{\max}^t\|\mathbf{Y}_0\|_2\end{aligned}$$

Taking the expectation gives us $\mathbb{E}[\|\mathbf{y}_t\|_2] \leq \kappa(1-\varepsilon)^t\mathbb{E}\left[\|\mathbf{Y}_0\|_2\right] \leq \kappa\lambda_{\max}^t\rho$.

Now we combine this with the Lipschitz continuity of ℓ_t to get

$$\begin{aligned}|\mathbb{E}\left[f_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)\right] - \mathbb{E}\left[f_t(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)\right]| &= |\mathbb{E}\left[\ell_t(\mathbf{x}_t,\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*))\right] - \mathbb{E}\left[\ell_t(\mathbf{x}_t,\mathbf{x}_t-\boldsymbol{\varepsilon}_t)\right]| \\ &\leq \mathbb{E}\left[|\ell_t(\mathbf{x}_t,\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)) - \ell_t(\mathbf{x}_t,\mathbf{x}_t-\boldsymbol{\varepsilon}_t)|\right] \\ &\leq L\cdot\mathbb{E}\left[\|\mathbf{x}_t - \mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) - \varepsilon_t\|_2\right] \\ &= L\cdot\mathbb{E}\left[\|\Delta\mathbf{x}_t - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) - \varepsilon_t\|_2\right] \\ &\leq \kappa L\rho\lambda_{\max}^t\end{aligned}$$

where we used Jensen's inequality in the first inequality. Summing this from $t=1$ to T gives us the result. □

Lemma 8.2.3. *For any data sequence $\{x_t\}_{t=1}^T$ that satisfies assumptions M1-M3, it holds that*

$$\left|\sum_{t=1}^T\mathbb{E}\left[f_t^m(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)\right] - \sum_{t=1}^T\mathbb{E}\left[f_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)\right]\right| = O(1)$$

if we choose $m = \log_{\lambda_{\max}}\left((2\kappa TLM_{\max}\sqrt{q})^{-1}\right)$.

Proof. Fix t. Note that for $m<0$,

$$\begin{aligned}|\Delta\mathbf{x}_t^m(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)| &= |\Delta\mathbf{x}_t - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)| \\ &\leq |\Delta\mathbf{x}_t - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) - \boldsymbol{\varepsilon}_t| + |\boldsymbol{\varepsilon}_t|\end{aligned}$$

The right hand side of the inequality is simply $\|\mathbf{y}_t\|_2 + \|\boldsymbol{\varepsilon}_t\|_2$, where $\mathbf{y}_t$ is as defined in Lemma 8.2.2. By assumption, $\mathbb{E}[\|\boldsymbol{\varepsilon}_t\|_2] < M_{\max}$. Assume that $M_{\max}$ is large enough such that $\mathbb{E}[\|\mathbf{y}_t\|_2] \leq M_{\max}$. This is a valid assumption since it is decaying exponentially as proved in Lemma 8.2.2. It is important to note that $\Delta\mathbf{x}_t^m(\Pi,\Gamma,\boldsymbol{\Theta})$ and $\Delta\mathbf{x}_t^\infty(\Pi,\Gamma,\boldsymbol{\Theta})$ have no randomness in them (recall that they can be written as a linear combination of past values of the realized data sequence $\Delta\mathbf{x}_t$). Thus, for $m<0$,

$$\begin{aligned}\|\Delta\mathbf{x}_t^m(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)\|_2 &= \mathbb{E}\left[\|\Delta\mathbf{x}_t^m(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)\|_2\right] \\ &= \mathbb{E}\left[\|\Delta\mathbf{x}_t - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)\|_2\right] \\ &\leq \mathbb{E}[\|\mathbf{y}_t\|_2 + \|\boldsymbol{\varepsilon}_t\|_2] \\ &\leq 2M_{\max}\end{aligned}$$

Squaring both sides of the inequality results in

$$\|\Delta\mathbf{x}_t^m(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*)\|_2^2 \leq 4M_{\max}^2$$

Next, we define

$$\mathbf{z}_t^m = \Delta\mathbf{x}_t^m(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*) - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\boldsymbol{\Theta}^*), \qquad \mathbf{Z}_t^m = \begin{bmatrix}\mathbf{z}_t^m \\ \mathbf{z}_{t-1}^{m-1} \\ \vdots \\ \mathbf{z}_{t-q+1}^{m-q+1}\end{bmatrix}$$

We have that

$$\begin{aligned}\Delta\mathbf{x}_t^m(\Pi^*,\Gamma^*,\mathbf{\Theta}^*) - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\mathbf{\Theta}^*) = \quad & \Pi^*\mathbf{x}_{t-1} + \sum_{i=1}^k \Gamma_i^*\Delta\mathbf{x}_{t-i} + \sum_{i=1}^q \mathbf{\Theta}_i^*(\Delta\mathbf{x}_{t-i} - \Delta\mathbf{x}_{t-i}^{m-i}(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)) \\ & - \Pi^*\mathbf{x}_{t-1} - \sum_{i=1}^k \Gamma_i^*\Delta\mathbf{x}_{t-i} - \sum_{i=1}^q \mathbf{\Theta}_i^*(\Delta\mathbf{x}_{t-i} - \Delta\mathbf{x}_{t-i}^{\infty}(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)) \\ = & -\sum_{i=1}^q \mathbf{\Theta}_i^*\left(\Delta\mathbf{x}_{t-i}^{m-i}(\Pi^*,\Gamma^*,\mathbf{\Theta}^*) - \Delta\mathbf{x}_{t-i}^{\infty}(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)\right)\end{aligned}$$

Thus, $\mathbf{z}_t^m = -\sum_{i=1}^q \mathbf{\Theta}_i^*\mathbf{z}_{t-i}^{m-i}$. The companion matrix to this difference equation is exactly $\mathbf{F}$ as defined above. Thus,

$$\mathbf{Z}_t^m = \mathbf{F}\mathbf{Z}_{t-1}^{m-1}$$

We have that

$$\begin{aligned}\|\mathbf{z}_t^m\|_2 \le \|\mathbf{Z}_t^m\|_2 &= \|\mathbf{F}\mathbf{Z}_{t-1}^{m-1}\|_2 \\ &= \|\mathbf{F}^2\mathbf{Z}_{t-2}^{m-2}\|_2 \\ &= \|\mathbf{F}^m\mathbf{Z}_{t-m}^{0}\|_2 \\ &= \|\mathbf{T}\Lambda^m\mathbf{T}^{-1}\mathbf{Z}_{t-m}^{0}\|_2 \\ &\le \|\mathbf{T}\|_2\|\mathbf{T}^{-1}\|_2\|\Lambda^m\|_2\|\mathbf{Z}_{t-m}^{0}\|_2 \\ &= \frac{\sigma_{\max}(\mathbf{T})}{\sigma_{\min}(\mathbf{T})}\lambda_{\max}^m\sqrt{\sum_{i=0}^{q-1}\|\mathbf{z}_{t-m-i}^{-i}\|_2^2} \\ &\le \kappa\lambda_{\max}^m\sqrt{q4M_{\max}^2} \\ &= \kappa\lambda_{\max}^m 2M_{\max}\sqrt{q}\end{aligned}$$

Now we combine this with the Lipschitz continuity of ℓ_t to get

$$\begin{aligned}|\mathbb{E}[f_t^m(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)] - \mathbb{E}[f_t^\infty(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)]| &= |\mathbb{E}[\ell_t(\mathbf{x}_t,\mathbf{x}_t^m(\Pi^*,\Gamma^*,\mathbf{\Theta}^*))] - \mathbb{E}[\ell_t(\mathbf{x}_t,\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\mathbf{\Theta}^*))]| \\ &\le \mathbb{E}[|\ell_t(\mathbf{x}_t,\mathbf{x}_t^m(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)) - \ell_t(\mathbf{x}_t,\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\mathbf{\Theta}^*))|] \\ &\le L\cdot\mathbb{E}[\|\mathbf{x}_t^m(\Pi^*,\Gamma^*,\mathbf{\Theta}^*) - \mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)\|_2] \\ &= L\cdot\|\Delta\mathbf{x}_t^m(\Pi^*,\Gamma^*,\mathbf{\Theta}^*) - \Delta\mathbf{x}_t^\infty(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)\|_2 \\ &\le 2\kappa LM_{\max}\sqrt{q}\lambda_{\max}^m\end{aligned}$$

where in the first inequality we used Jensen's inequality.

Summing this quantity from $t=1$ to T gives us the result:

$$\left|\sum_{t=1}^T \mathbb{E}[f_t^m(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)] - \sum_{t=1}^T \mathbb{E}[f_t^\infty(\Pi^*,\Gamma^*,\mathbf{\Theta}^*)]\right| \le 2\kappa TLM_{\max}\sqrt{q}\lambda_{\max}^m$$

Choosing $m = \log_{\lambda_{\max}}\left((2\kappa TLM_{max}\sqrt{q})^{-1}\right)$ gives us the desired $O(1)$ property. □

8.3 Proof of Theorem 5.1

Proof. Recall that for FTL, we have that

$$\boldsymbol{\gamma}_t \in \operatorname*{argmin}_{\boldsymbol{\gamma}} \sum_{i=1}^{t-1}\ell_t(\boldsymbol{\gamma}) = \operatorname*{argmin}_{\boldsymbol{\gamma}} \frac{1}{2}\sum_{i=1}^{t-1}(x_t - \boldsymbol{\gamma}^\intercal\boldsymbol{\psi}_t)^2 = \operatorname*{argmin}_{\boldsymbol{\gamma}} \frac{1}{2}\|X_t - \Psi_t\boldsymbol{\gamma}\|_2^2$$

where $X_t = \begin{bmatrix} x_t & \dots & x_1 \end{bmatrix}^\intercal, \Psi_t = \begin{bmatrix} \boldsymbol{\psi}_t & \dots \boldsymbol{\psi}_1 \end{bmatrix}^\intercal$. Note that this is simply a recursive least squares procedure. This procedure can be computed in a recursive manner using the update equations:

$$\boldsymbol{\gamma}_{t+1} = \boldsymbol{\gamma}_t + \frac{x_t - \boldsymbol{\psi}_t^\intercal \boldsymbol{\gamma}_t}{1 + \boldsymbol{\psi}_t^\intercal V_{t-1} \boldsymbol{\psi}_t} V_{t-1} \boldsymbol{\psi}_t$$
$$V_{t+1} = V_t - \frac{V_t \boldsymbol{\psi}_{t+1} \boldsymbol{\psi}_{t+1}^\intercal V_t}{1 + \boldsymbol{\psi}_{t+1}^\intercal V_t \boldsymbol{\psi}_{t+1}}$$

where $V_t = \left(\sum_{i=1}^t \boldsymbol{\psi}_i \boldsymbol{\psi}_i^\intercal \right)^{-1}$. Using the fact that ℓ_t is Lipschitz, we have that

$$\begin{aligned}
|\ell_t(\boldsymbol{\gamma}_t) - \ell_t(\boldsymbol{\gamma}_{t+1})| &\leq L \|\boldsymbol{\gamma}_{t+1} - \boldsymbol{\gamma}_t\|_2 \\
&= L \left\| \frac{x_t - \boldsymbol{\gamma}_t^\intercal \boldsymbol{\psi}_t}{1 + \boldsymbol{\psi}_t^\intercal V_{t-1} \boldsymbol{\psi}_t} V_{t-1} \boldsymbol{\psi}_t \right\|_2 \\
&\leq L \left| \frac{x_t - \boldsymbol{\gamma}_t^\intercal \boldsymbol{\psi}_t}{1 + \boldsymbol{\psi}_t^\intercal V_{t-1} \boldsymbol{\psi}_t} \right| \|V_{t-1}\|_2 \|\boldsymbol{\psi}_t\|_2 \\
&\leq L^2 \|V_{t-1}\|_2 \\
&= L^2 \lambda_{\max}(V_{t-1}) \\
&= \frac{L^2}{(t-1)\lambda_{\min}(t-1)}
\end{aligned}$$

where we used the fact that $\|\nabla_{\boldsymbol{\gamma}} \ell_t(\boldsymbol{\gamma})\|_2 = |x_t - \boldsymbol{\gamma}^\intercal \boldsymbol{\psi}_t| \|\boldsymbol{\psi}_t\|_2 \leq L$, $\frac{1}{1 + \boldsymbol{\psi}_t^\intercal V_{t-1} \boldsymbol{\psi}_t} \leq 1$.

To complete the proof, we sum this quantity up and invoke Lemma 8.3.1. To avoid the divide-by-zero, simply start the indexing at $t = 2$.

□

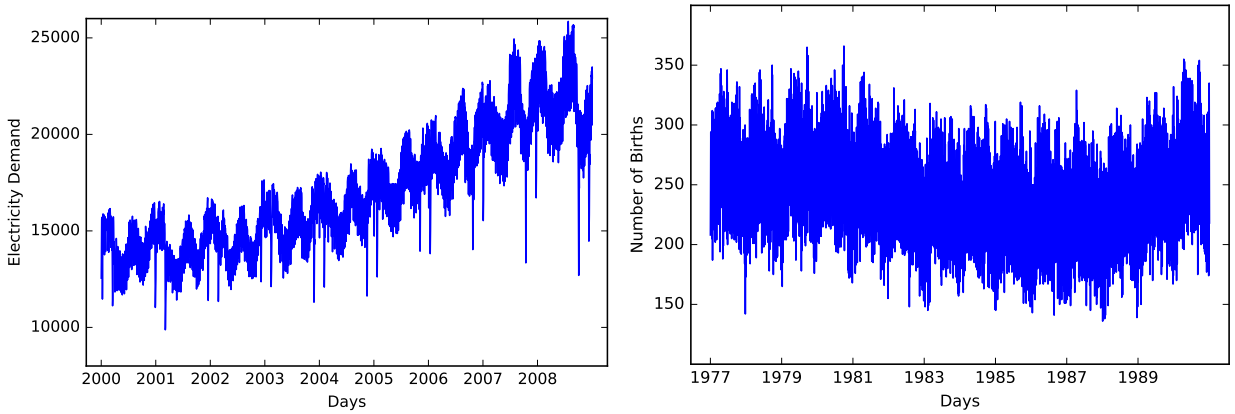
Lemma 8.3.1. *Let $\ell_1, \dots, \ell_T$ be a sequence of loss functions. Let $\boldsymbol{\gamma}_1, \dots, \boldsymbol{\gamma}_t$ be produced by FTL. Then*

$$\sum_{t=1}^T \ell_t(\boldsymbol{\gamma}_t) - \min_{\boldsymbol{\gamma}} \sum_{t=1}^T \ell_t(\boldsymbol{\gamma}) \leq \sum_{t=1}^T \left[\ell_t(\boldsymbol{\gamma}_t) - \ell_t(\boldsymbol{\gamma}_{t+1}) \right]$$

This is fairly standard material. For reference to a proof, see [9].

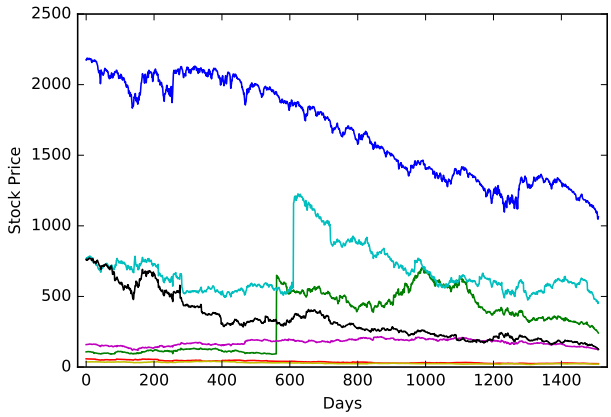
8.4 Data for Experiments

In this section, we display the data we used in Section 6.

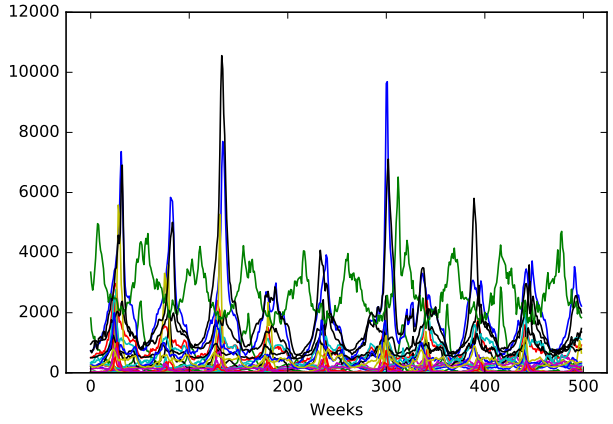


(a) Turkey electricity demand

(b) Births in Quebec



(c) Stock data



(d) Google flu data

Figure 3: Data plots. The top line has plots for univariate data, and the bottom line has plots for multivariate data.