Time Series Analysis by State Space Learning

André Ramos, Davi Valladão, Alexandre Street

September 27, 2024

Abstract

Time series analysis by state-space models is widely used in forecasting and extracting unobservable components like level, slope, and seasonality, along with explanatory variables. However, their reliance on traditional Kalman filtering frequently hampers their effectiveness, primarily due to Gaussian assumptions and the absence of efficient subset selection methods to accommodate the multitude of potential explanatory variables in today's big-data applications. Our research introduces the State Space Learning (SSL), a novel framework and paradigm that leverages the capabilities of statistical learning to construct a comprehensive framework for time series modeling and forecasting. By utilizing a regularized high-dimensional regression framework, our approach jointly extracts typical time series unobservable components, detects and addresses outliers, and selects the influence of exogenous variables within a high-dimensional space in polynomial time and global optimality guarantees. Through a controlled numerical experiment, we demonstrate the superiority of our approach in terms of subset selection of explanatory variables accuracy compared to relevant benchmarks. We also present an intuitive forecasting scheme and showcase superior performances relative to traditional time series models using a dataset of 48,000 monthly time series from the M4 competition. We extend the applicability of our approach to reformulate any linear state space formulation featuring time-varying coefficients into high-dimensional regularized regressions, expanding the impact of our research to other engineering applications beyond time series analysis. Finally, our proposed methodology is implemented within the Julia open-source package, "StateSpaceLearning.jl".

Keywords: Time Series, State-Space Models, Statistical Learning, Machine Learning, Lasso

1 Introduction

Time series analysis by state space models is a flexible framework for capturing non-observable components such as level, trend, and seasonality, as well as exogenous features. This makes them an attractive choice for modeling temporal data across various domains, including energy, control, telecommunications, signal processing, and business, just to mention a few. According to [9], the state space framework confers enhanced flexibility compared to conventional models, exemplified by ARIMA [6]). This heightened flexibility arises from its capacity to circumvent certain assumptions, such as the imposition of stationarity.

State space modeling, coupled with the Kalman filter technique [14], has emerged as the standard approach for estimating models with unobserved components with or without exogenous variables. Nevertheless, this method has several limitations that restrict its applicability in real-world scenarios. First, the existing methodology relies on non-linear optimization techniques, which may provide estimates that are locally optimal. However, the quality of these estimates is heavily dependent on the user's initial guesses, which may not always be well-informed. In practice, users may resort to trial-and-error or stochastic generation procedures, which can lead to suboptimal and computationally expensive results. Second, despite the state space model's ability to incorporate explanatory variables, it lacks the functionality to conduct built-in subset selection, which can limit its performance in high-dimensional settings characterized by numerous potential explanatory variables (a common feature in today's data-rich environments). Third, although the state space framework is able to detect outliers by studying the auxiliary residuals, as explained in [9], it is done in a heuristic two-step process, which may lead to loss of information.

As such, it is of utmost importance for the time series community to develop novel and effective approaches that can mitigate the limitations of the current methodologies while preserving key features from traditional models. Therefore, the objective of this work is to propose a novel, efficient, and integrated framework for time series in state-space format, named State Space Learning (SSL). Based on this new formulation, a new paradigm is unlocked for non-stationary time-series modeling. The main contributions of the proposed framework concerning the existing technical literature on the subject are the following:

- a novel formulation for structural models (with unobserved components of level, slope, and seasonality) based on a high-dimensional regularized regression on shifted steps, ReLus, and alternating functions;

- a computationally efficient (in polynomial time) estimation method for state-space models based on convex optimization that ensures global optimality;

- an effective and integrated method to detect and handle outliers within the proposed SSL estimation process;

- an optimal subset selection methodology applicable to non-stationary time series;

- a k-step ahead out-of-sample forecasting approach (extrapolation) with improved forecasting capacity compared to traditional state space benchmarks;

- a new interpolation method to handle missing values in non-stationary time series;

- effective way to fully initialize basic Gaussian structural models with high-quality innovations, initial conditions, hyperparameters, and selected explanatory variables;

- a generalization that can be applied to any linear state-space model with time-varying coefficients;

- an open-source package of the proposed method in Julia: "StateSpaceLearning.jl".

We conduct an extensive set of numerical experiments using synthetic and real-world time series that corroborate, with substantial empirical evidence, the robustness, predictive power, and model selection capabilities of the proposed framework. Based on our results, we believe the proposed approach represents a significant advancement over the existing methods and has the potential to substantially enhance the accuracy and reliability of temporal data modeling in diverse fields.

1.1 Illustrative example

The basic idea behind our approach is to establish that state space models can be reformulated as high-dimensional regression models by dismounting the model recursion and be effectively estimated using regularization techniques such as Elastic Net [30].

To illustrate this point, we begin by introducing a simple local level model

$$y_t = \mu_t + \varepsilon_t, \tag{1}$$
$$\mu_t = \mu_{t-1} + \eta_t, \tag{2}$$

where y_t is the dependent variable being estimated, μ_t is the stochastic level component, ε_t is the observation error and η_t is the level component noise.

Now, we reformulate the model (1–2) by iterating the level component over time as

$$y_t = \mu_1 + \sum_{\tau=2}^{t} \eta_\tau + \varepsilon_t, \tag{3}$$

Note that Equation 3 can be written in a vectorized form

$$\mathbf{y} = \mu_1 + \mathbf{X}\eta + \varepsilon, \tag{4}$$

where $\mathbf{y} = (y_1, \ldots, y_T)$, $\eta = (\eta_2, \ldots, \eta_T)$ and

$$\mathbf{X} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 1 & 1 & \cdots & 0 \\ 1 & 1 & \cdots & 1 \end{bmatrix},$$

with each column is referent to one η_τ coefficient. Note that Equation 4 can be interpreted as a high dimension regression and each regressor is a step function that starts at time $\tau = 1, \ldots, T$. To handle the large number of features in the problem, we estimate a regularized model using the following regularized regression formulation

$$\underset{\mu_1, \eta}{\text{minimize}} \ \left\| \mathbf{y} - \mu_1 - \mathbf{X}\eta \right\|_2^2 + \lambda \left[(1-\alpha)\|\eta\|_2^2 / 2 + \alpha\|\eta\|_1 \right], \tag{5}$$

where $\lambda > 0$ and $0 \le \alpha \le 1$.

As delineated in the Introduction, it is noteworthy that the SSL formulation of the local-level model exhibits an underlying high-dimensional regression form that comprises a large collection of step functions as potential regressors. This idea is extended to a more general class of structural models with unobserved components, such as slope, level, seasonality, outliers (as we demonstrate in Section 2), and exogenous explanatory variables. Also, in Section 3.4, we show that this framework can be employed to any additive State Space model, thereby we can use in a large class of engineering and statistical applications.

1.2 Literature Review

State space modeling for time series analysis represents a significant departure from traditional methods like Seasonal Autoregressive Integrated Moving Average (SARIMA). Various classical state space formulations, including the local level, linear trend, and basic structural model, have been extensively elucidated in sources such as [9] and [2]. While many non-parametric methods, such as those depicted by [15], can extract unobserved components, the state space formulation provides a more comprehensive framework for time series analysis and forecasting. This approach, which relies on the Kalman filter for solving, offers robust solutions for modeling time series, including non-stationary ones.

As discussed in [9], the estimation of the Kalman filter and smoother algorithm rely on nonlinear optimization whose performance (accuracy and solution time) is highly dependent on good initialization points. (as discussed in [10]).

Despite their efficacy, traditional state space frameworks face limitations, particularly in accommodating the demands of big data environments, as discussed in Section 1. While applications of these models span diverse domains such as ecology [3], finance [23], and engineering [26], methodological advancements have been sporadic, with few relevant exceptions concentrated on [18], [20], [24], and [17].

Conversely, statistical learning methods have expanded significantly, with researchers pioneering innovative methodologies to bolster predictive accuracy and model interpretability. Widely embraced for their adeptness in handling big data, regularization methods like Lasso [28] and Ridge Regression [11] have emerged as stalwarts in combating overfitting issues inherent in linear regression models. The introduction of techniques such as the Elastic Net [30] and Adaptive Lasso [29] further underscores the dynamic landscape of statistical learning. Recently, a variety of applications have utilized the statistical learning framework for tasks such as prediction, inference, and feature selection, as illustrated in [5, 8]. For an larger collection of examples, see [13].

Specifically in the time series domain, several statistical learning applications have been developed. For instance, [16] introduces the WLadaLASSO, a method employing weighted lag adaptive Lasso penalties in time series analysis. In [19], the authors explore Lasso-based approaches for forecasting macroeconomic variables; they find that combining forecasts from Lasso-based methods with dynamic factor models further reduces forecast error, providing more interpretable and accurate predictions. Furthermore, in [22], the authors thoroughly analyzed the asymptotic properties of the Adaptive Lasso within sparse, high-dimensional, linear time-series models. Notably, their results underscored the Adaptive Lasso's consistent selection of relevant variables as the volume of observations increased, indicative of model selection consistency.

We contend that a symbiotic fusion of statistical learning methodologies with the state-space framework holds immense potential. Such an interdisciplinary approach can invigorate and propel further development in the state-space literature on time series analysis, fostering a fertile ground for future research.

2 State-Space Learning Framework for Time Series Analysis

In this section, we introduce and generalize a comprehensive framework that encompasses critical elements inherent to time series analysis, including stochastic trends, slopes, seasonality patterns, and disturbance terms. Furthermore, we incorporate extensions for robust outlier detection, subset selection strategies, estimation procedures, and model selection criteria (all accomplished without necessitating a stationarity assumption), culminating in the formulation of an easily applicable extrapolation scheme. Additionally, our framework extends its utility beyond traditional time series data by providing a versatile model capable of accommodating diverse datasets, thereby enhancing its adaptability to various analytical contexts.

2.1 Learning the Basic Structure of Time Series

A basic structural model can be formulated as

$$y_t = \mu_t + \gamma_t + \varepsilon_t \tag{6}$$

$$\mu_{t+1} = \mu_t + \nu_t + \xi_{t+1} \tag{7}$$

$$\nu_{t+1} = \nu_t + \zeta_{t+1} \tag{8}$$

$$\gamma_{t+1} = -\sum_{j=1}^{s-1} \gamma_{t+1-j} + \omega_{t+1}, \tag{9}$$

where μ_t, ν_t, γ_t represents the stochastic trend, slope and seasonality respectively and μ_0, ν_0, $\gamma_1, \ldots, \gamma_s$ are the initial states.

In this Section, we will prove that this time series structure with level, trend, and seasonality can be written as a high-dimensional regression framework. This proof will be made separately for each component and then merged into one equation.

Proposition 2.1. *The slope component $\nu_{t+1} = \nu_t + \zeta_{t+1}$ can be formulated as $\nu_{t+1} = \nu_1 + \sum_{\tau=2}^{t+1} \zeta_\tau$.*

Proof. We prove Proposition 2.1 by induction. The statement $\nu_{t+1} = \nu_1 + \sum_{\tau=2}^{t+1} \zeta_\tau$ holds true for our base case $t = 1$ since $\nu_2 = \nu_1 + \zeta_2$. Now, assuming the statement holds true for t, we show that it also holds true for $t + 1$. For that, we show that the induction hypothesis for ν_t, i.e.,

$$\nu_t = \nu_1 + \sum_{\tau=2}^{t} \zeta_\tau, \tag{10}$$

can be confirmed for $t + 1$ by using (10) in (8), i.e.,

$$\nu_{t+1} = \nu_t + \zeta_{t+1} = \nu_1 + \sum_{\tau=2}^{t} \zeta_\tau + \zeta_{t+1} = \nu_1 + \sum_{\tau=2}^{t+1} \zeta_\tau.$$

$\square$

Theorem 2.1. *The trend component $\mu_{t+1} = \mu_t + \nu_t + \xi_{t+1}$ can be formulated as*

$$\mu_{t+1} = \mu_1 + \sum_{\tau=2}^{t+1} \xi_\tau + t\,\nu_1 + \sum_{\tau=2}^{t} ((t+1) - \tau)\zeta_\tau$$

.

Proof. We also prove it by induction. Our base case $\mu_2 = \mu_1 + \nu_1 + \xi_2$ holds true for $t = 2$ and we apply the induction hypothesis

$$\mu_t = \mu_1 + \sum_{\tau=2}^{t} \xi_\tau + (t-1)\nu_1 + \sum_{\tau=2}^{t-1} (t - \tau)\zeta_\tau$$

for $t \geq 2$, in the recursion $\mu_{t+1} = \mu_t + \nu_t + \xi_{t+1}$ and obtain

$$\mu_{t+1} = \mu_1 + \sum_{\tau=2}^{t} \xi_\tau + (t-1)\nu_1 + \sum_{\tau=2}^{t-1} (t - \tau)\zeta_\tau + \nu_t + \xi_{t+1}$$

Now, using Proposition (2.1), we replace ν_t obtain

$$\mu_{t+1} = \mu_1 + \sum_{\tau=2}^{t} \xi_\tau + (t-1)\nu_1 + \sum_{\tau=2}^{t-1} (t - \tau)\zeta_\tau + \nu_1 + \sum_{\tau=2}^{t} \zeta_\tau + \xi_{t+1}$$

and, consequentially,

$$\mu_{t+1} = \mu_1 + \sum_{\tau=2}^{t+1} \xi_\tau + ((t+1) - 1)\nu_1 + \sum_{\tau=2}^{(t+1)-1} ((t+1) - \tau)\zeta_\tau$$

$\square$

Theorem 2.2. *The seasonal component* $\gamma_{t+1} = -\sum_{j=1}^{s-1} \gamma_{t+1-j} + \omega_{t+1}$ *can be formulated as*

$$\gamma_{t+1} = \gamma_{t+1-ks} + \sum_{j=0}^{k-1}(\omega_{t+1-js} - \omega_{t-js}), \quad \forall t \in \{1, \cdots, T-1\}, k \in \left\{1, \ldots, \left\lfloor \frac{t}{s} \right\rfloor\right\},$$

given that $\lfloor \cdot \rfloor$ *is the floor operator.*

Proof. We first define our induction hypothesis as

$$\gamma_{t+1} = \gamma_{t+1-ks} + \sum_{j=0}^{k-1}(\omega_{t+1-js} - \omega_{t-js}), \quad \forall t \in \{1, \cdots, T-1\}, k \in \left\{1, \ldots, \left\lfloor \frac{t}{s} \right\rfloor\right\}.$$

To prove that our base case ($k = 1$) holds, we rewrite Equation 9 and apply it for $t+1$ and t.

$$\sum_{j=1}^{s} \gamma_{t+2-j} = \omega_{t+1} \tag{11}$$

$$\sum_{j=1}^{s} \gamma_{t+1-j} = \omega_t \tag{12}$$

Subtracting 12 from 11, we obtain the seasonal recursion

$$\gamma_{t+1} = \gamma_{t+1-s} + \omega_{t+1} - \omega_t, \quad \forall t \in \{s, \ldots, T-1\}. \tag{13}$$

Now, we prove that if the hypothesis holds for $k = 1$, it also holds for any $k \in \left\{1, \ldots, \left\lfloor \frac{t}{s} \right\rfloor\right\}$. For that, we replace γ_{t+1-ks} in

$$\gamma_{t+1} = \gamma_{t+1-ks} + \sum_{j=0}^{k-1}(\omega_{t+1-js} - \omega_{t-js})$$

by its seasonal recursion $\gamma_{t+1-ks} = \gamma_{t+1-(k+1)s} + \omega_{t+1-ks} - \omega_{t-ks}$ and obtain

$$\gamma_{t+1} = \gamma_{t+1-(k+1)s} + \omega_{t+1-ks} - \omega_{t-ks} + \sum_{j=0}^{k-1}(\omega_{t+1-js} - \omega_{t-js}).$$

Therefore,

$$\gamma_{t+1} = \gamma_{t+1-(k+1)s} + \sum_{j=0}^{k}(\omega_{t+1-js} - \omega_{t-js}).$$

Now, assuming $k = \left\lfloor \frac{t}{s} \right\rfloor$, i.e., its largest possible k value for each time $t+1$, we define $m_{t+1} = t + 1 - s\left\lfloor \frac{t}{s} \right\rfloor$. For instance, if we have a monthly frequency and $s = 12$, then m_{t+1} can be interpreted as the first occurrence of the month associated to time $t+1$. Hence, we define our seasonal component as

$$\gamma_{t+1} = \gamma_{m_{t+1}} + \sum_{j=0}^{\lfloor t/s \rfloor - 1}(\omega_{t+1-js} - \omega_{t-js}).$$

For didactic purposes, we define $\tau = t + 1 - js$ reformulate it as

$$\gamma_{t+1} = \gamma_{m_{t+1}} + \sum_{\tau \in M_{t+1}}(\omega_\tau - \omega_{\tau-1}),$$

where $M_{t+1} = \{t + 1 - js\}_{j=0}^{\lfloor t/s \rfloor - 1} = \{m_{t+1} + s, m_{t+1} + 2s, \ldots, t + 1 - 2s, t + 1 - s, t + 1\}$. Formally speaking, $M_t = \{\tau \in \{m_t + s, \ldots, t\} \mid m_t = m_\tau\}$.

$\square$

Finally, we obtain our final formulation, where the basic structural model is fully represented by a high-dimensional regression.

Corollary 2.2.1. *The basic structural model (6-9) is equivalent to the high dimensional regression*

$$y_1 = \mu_1 + \gamma_1 + \varepsilon_1 \tag{14}$$

$$y_2 = \mu_1 + \xi_2 + \nu_1 + \gamma_2 + \varepsilon_2 \tag{15}$$

$$y_t = \mu_1 + \sum_{\tau=2}^{t} \xi_\tau + (t-1)\nu_1 + \sum_{\tau=2}^{t-1}(t-\tau)\zeta_\tau + \gamma_{m_t} + \varepsilon_t, \quad \forall t = \{3, \ldots, s\} \tag{16}$$

$$y_t = \mu_1 + \sum_{\tau=2}^{t} \xi_\tau + (t-1)\nu_1 + \sum_{\tau=2}^{t-1}(t-\tau)\zeta_\tau + \gamma_{m_t} + \sum_{\tau \in M_t}(\omega_\tau - \omega_{\tau-1}) + \varepsilon_t, \quad \forall t = s+1, \ldots, T \tag{17}$$

Proof. First, (14) is the same as (6) for $t = 1$. Second, we prove (16) by replacing in (6) the definition of μ_t in (7), for $t = 2$. Then, we have $y_2 = \mu_2 + \gamma_2 + \varepsilon_2 = \mu_1 + \nu_1 + \xi_2 + \gamma_2 + \varepsilon_2$. Finally, we prove the corollary for $t \geq 2$ by replacing in the equation $y_t = \mu_t + \gamma_t + \varepsilon_t$ the representations of the trend μ_t and seasonality γ_t obtained in Theorem 2.1 and in Theorem 2.2, respectively. $\square$

2.2 Interpreting the underlying regressors

For an easier understanding of the underlying regressors, we rewrite Corollary 2.2.1 as

$$y_t = \mu_1 + (t-1)\nu_1 + \sum_{k=1}^{s} \mathbb{I}_{\{m_t\}}(k)\, \gamma_k$$

$$+ \sum_{\tau=2}^{T} \left[\mathbb{I}_{[1,t]}(\tau)\, \xi_\tau + (t-\tau)^+ \zeta_\tau + \left(\mathbb{I}_{M_t}(\tau) - \mathbb{I}_{M_{t+1}}(\tau) \right) \omega_\tau \right] + \varepsilon_t, \tag{18}$$

where $x^+ = \max(0, x)$ and $\mathbb{I}_A(x)$ is an indicator function that values 1 when $x \in A$ and 0 otherwise. For a better visualization of the model as a regression, let $Y = (y_1, \ldots, y_T)^\top$, $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_T)^\top$ and $\Theta = (\mu_1, \xi, \nu_1, \zeta, \gamma_1, \ldots, \gamma_s, \omega)^\top$, where $\xi = \xi_2, \ldots, \xi_T$, $\zeta = \zeta_2, \ldots, \zeta_T$ and $\omega = \omega_s, \ldots, \omega_T$, then we can represent it as $Y = X\Theta + \varepsilon$, where X is the regressor matrix. For instance, for $T = 5$ and $s = 2$, the regressor matrix X is defined as

$$\begin{array}{ccccccccccccccc}
\mu_1 & \xi_2 & \xi_3 & \xi_4 & \xi_5 & \nu_1 & \zeta_2 & \zeta_3 & \zeta_4 & \gamma_1 & \gamma_2 & \omega_2 & \omega_3 & \omega_4 & \omega_5
\end{array}$$
$$\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 & 2 & 1 & 0 & 0 & 1 & 0 & -1 & 1 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 3 & 2 & 1 & 0 & 0 & 1 & 0 & -1 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 4 & 3 & 2 & 1 & 1 & 0 & -1 & 1 & -1 & 1
\end{pmatrix}. \tag{19}$$

We now provide a quick interpretation of the components of the proposed high-dimensional regression framework (level, slope, and seasonality). The initial state of the level component, represented by μ_1, corresponds to the model's intercept. The regressors associated with the innovations ξ_t will assume value 1 when $t \geq \tau$ and 0 otherwise. These regressors can be conceptualized as a series of step functions, each starting at distinct times, aiming to capture level changes within the time series. It's important to highlight that a regularized model aims to parsimoniously select the pertinent step functions to elucidate the underlying dynamics of the time series.

For the slope component, it's important to recognize that the initial state ν_1 is the coefficient of a conventional linear trend regressor with values spanning from 0 to $T-1$. Complementing this, the coefficients ζ_t refer to regressors resembling rectified linear unit (ReLU), initiating at each timestamp. Together, they form a piecewise linear function adept at capturing changes in trends across time.

In Figure 1, we present an illustrative example of these components for a time series with 5 observations. The initial states $\gamma_1, \ldots, \gamma_s$ of the seasonal component can be likened to seasonal dummy regressors, representing the deterministic part of the seasonality. Meanwhile, the innovations $\omega_t = \sum_{j=1}^{s} \gamma_{t+1-j}$ can be conceptualized as moving sums of the stochastic seasonal components spanning an entire season. It's worth noting that in a regularized regression model, there's a tendency for ω_t to shrink towards zero. This occurs because seasonal variations tend to cancel out when summed over a whole season, such as twelve months, leading to a diminished impact on the overall model.
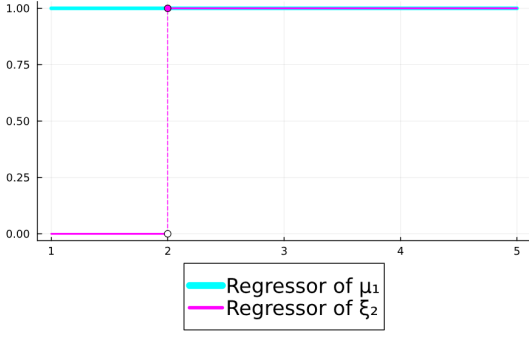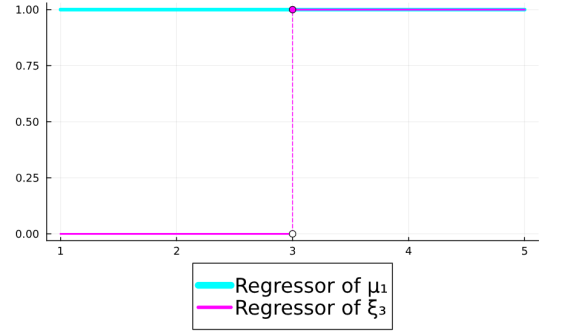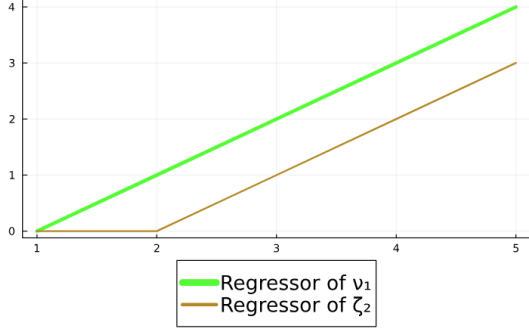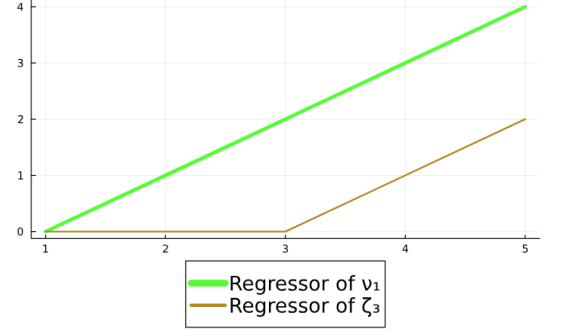
Figure 1: In this example we see in Figures 1a, 1b the components $\mu_1 + \sum_{\tau=2}^{t} \xi_\tau$ and in Figures 1c, 1d the components $(t-1)\nu_1 + \sum_{\tau=2}^{t-1}(t-\tau)\zeta_\tau$.

2.3 Estimation Procedure, Model Selection and Component Extraction

We introduce an estimation procedure featuring adaptive regularization, akin to the Adaptive Lasso method [29]. However, our approach incorporates a distinctive adaptation specifically designed to address the nuances of the present problem. To obtain estimates for Θ, we first solve

$$\underset{\theta_0,\theta}{\text{minimize}} \frac{1}{|\mathcal{T}|} \sum_{t\in\mathcal{T}} [y_t - \hat{y}_t(\theta_0,\theta)]^2 + \lambda \left[(1-\alpha)\frac{1}{2}\|\theta\|_2^2 + \alpha\|\theta\|_1\right], \tag{20}$$

where $\mathcal{T} = \{1,\ldots,T\}$, $\theta_0 = (\mu_1,\nu_1,\gamma_1,\ldots,\gamma_s)$ are the initial states and $\theta = (\xi_2,\ldots,\xi_T,\zeta_2,\ldots,\zeta_T, \omega_s,\ldots,\omega_T)$.

Except for the initial states, note that the regularization manifests as a penalty λ applied to the convex combination of the L1 and L2 norms of θ with weights α and $(1-\alpha)$, respectively. It is crucial to underscore the deliberate exclusion of regularization on the initial states during this phase. This intentional omission allows us to effectively identify and obtain a parsimonious impact of the innovations while preserving the key deterministic components of the model: intercept, a linear deterministic trend, and the seasonal dummies. For a fixed α, we solve (20) for several values of λ and chooses the one the minimizes an information criteria, such as AIC or BIC.

In the second step, we develop adaptive estimation procedure

$$\begin{aligned}
\underset{\theta_0,\theta}{\text{minimize}} \frac{1}{|\mathcal{T}|} \sum_{t\in\mathcal{T}} [y_t - \hat{y}_t(\theta_0,\theta)]^2 + \lambda \Bigg[&w_\xi \left((1-\alpha)\frac{1}{2}\|\xi\|_2^2 + \alpha\|\xi\|_1\right) + \\
&w_\zeta \left((1-\alpha)\frac{1}{2}\|\zeta\|_2^2 + \alpha\|\zeta\|_1\right) + \\
&w_\omega \left((1-\alpha)\frac{1}{2}\|\omega\|_2^2 + \alpha\|\omega\|_1\right) + \\
&w_{\nu_1} \left((1-\alpha)\frac{1}{2}\|\nu_1\|_2^2 + \alpha\|\nu_1\|_1\right) + \\
&\sum_{i=1}^{s} w_{\gamma_i} \left((1-\alpha)\frac{1}{2}\|\gamma_i\|_2^2 + \alpha\|\gamma_i\|_1\right) \Bigg],
\end{aligned} \tag{21}$$

where we introduce different penalty weights $w_\xi, w_\zeta, w_\omega, w_{\nu_1}, w_{\gamma_1}, \ldots, w_{\gamma_s}$ for each regression coefficient. To define these weights, let $\hat{\theta}_0 = (\hat{\mu}_1, \hat{\nu}_1, \hat{\gamma}_1, \ldots, \hat{\gamma}_s)$ and $\hat{\theta} = (\hat{\xi}_2, \ldots, \hat{\xi}_T, \hat{\zeta}_2, \ldots, \hat{\zeta}_T, \hat{\omega}_s, \ldots, \hat{\omega}_T)$ denote the solutions of (20).

Next, we calculate estimates for the standard deviation of each innovation term $\hat{\sigma}_\kappa = \|\kappa\|_2, \forall \kappa \in \{\xi, \zeta, \omega\}$ and derive the associated penalty weights $w_\kappa = \frac{1}{\epsilon + \hat{\sigma}_\kappa}, \forall \kappa \in \{\xi, \zeta, \omega\}$ for the second step of our adaptive procedure. Likewise, we introduce penalty weights for $w_{\nu_1} = \frac{1}{\epsilon + |\hat{\nu}_1|}$ and $w_{\gamma_i} = \frac{1}{\epsilon + |\hat{\gamma}_i|}, \forall i = 1, \ldots, s$. The term ϵ represents a non-negative parameter that handles the case where the estimated coefficients of the first step are 0 [4]. In our numerical experiments, we set $\epsilon = 0.05$. We contend that incorporating a penalty in the second step for the linear trend and seasonal dummies facilitates robust estimation and automated selection of these structures.

For the same α fixed in the first step, we choose the regularization penalty λ such as to minimize the information criteria of choice (e.g., AIC or BIC). Now, let $\theta_0 = (\mu_1, \nu_1, \gamma_1, \ldots, \gamma_s)$ and $\theta = (\xi_2, \ldots, \xi_T, \zeta_2, \ldots, \zeta_T, \omega_s, \ldots, \omega_T)$ denote the solution of (21) for the selected α and λ. Then, we can reconstruct the components using the Theorems in Section 2.1. Formally speaking:

$$\nu_t = \nu_1 + \sum_{\tau=2}^{t} \zeta_\tau \tag{22}$$

$$\mu_t = \mu_1 + \sum_{\tau=2}^{t} \xi_\tau + (t-1)\nu_1 + \sum_{\tau=2}^{t-1} (t-\tau)\zeta_\tau \tag{23}$$

$$\gamma_t = \gamma_{m_t} + \sum_{\tau \in M_t} (\omega_\tau - \omega_{\tau-1}). \tag{24}$$

The extraction of model components constitutes a pivotal phase in structural time series analysis, facilitating the discernment and exploration of trends and patterns that may elude direct observation in raw data. We illustrate component extraction capability of the model and compare it with a traditional benchmark. Specifically, we compare the smoothed components of a Gaussian benchmark estimated via Kalman filter against the SSL $(AIC, \alpha = 0.1)$ model. This comparative analysis is performed on three diverse time series drawn from the M4 dataset, see [27].

Our findings, graphically depicted in Figure 2, reveal notable similarities in the fits produced by both models. Note that the results in for the SSL model presents a less noisy pattern (mainly in the trend component), which indicates that our proposed methodology presents more parsimonious results.
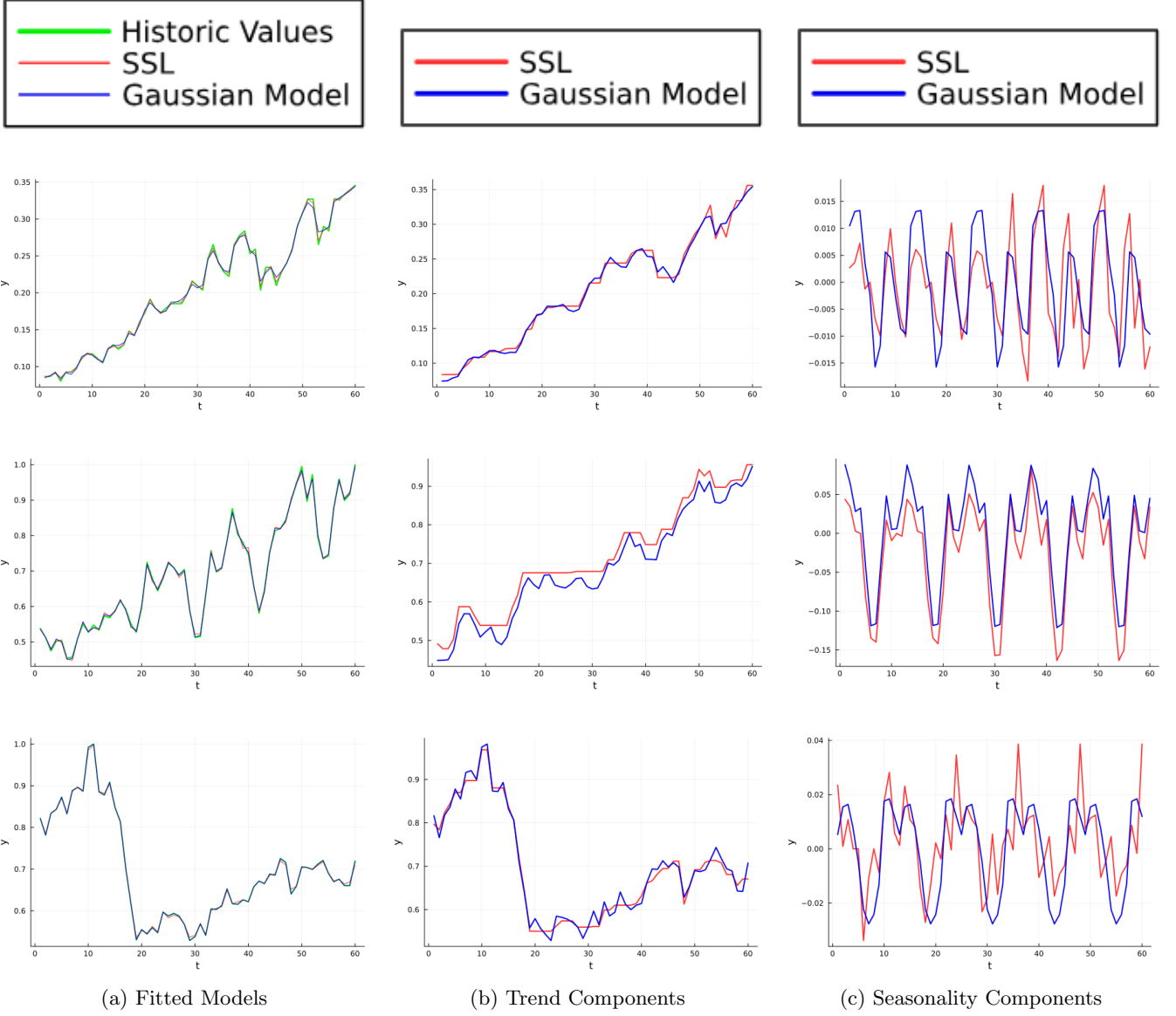
Figure 2: Fitted time series and separated components (each row represents one time series) for both the gaussian and the SSL ($\alpha = 0.1 - AIC$) models.

2.4   Handling Missing Values via Interpolation

The proposed SSL framework has the capability to handle missing values within time-series data through a slight adjustment in the estimation procedure. Specifically, in both Equations (20) and (21), we define the $\mathcal{T} \in \{1, \ldots, T\}$ to be the set of timestamps with valid observations. It is noteworthy that the fitted values of the estimated time series can be employed to populate missing values at those time stamps where observations are absent, as these fitted values serve as interpolations of the underlying time series dynamics. Formally speaking, the interpolated values for $t \notin \mathcal{T}$ is given by $\hat{y}_t = \mu_t + \gamma_t$, where μ_t and γ_t are computed as in (22-24), using the estimate Θ obtained as a solution of (21).

Allow us to elucidate instances of time series exhibiting lacunae in data, alongside the interpolated outcomes for these instances, as facilitated by both the proposed State Space Learning framework and its Gaussian analogue. In this examination, three authentic time series were subjected to the removal of designated observations, thereby categorizing them as absent values, to evaluate the efficacy of our proposed framework. Evidently, our analysis demonstrates that both models yield robust outcomes for the missing observations, as evidenced by comparison with the original values, which were not considered throughout the estimation procedure.
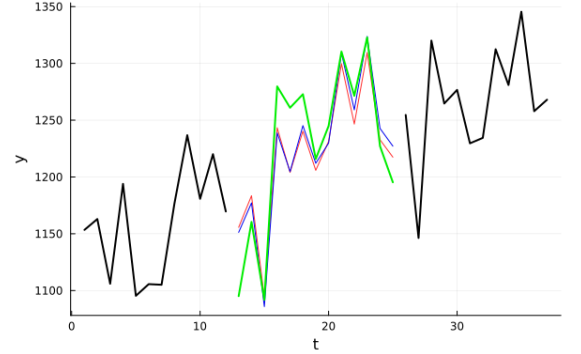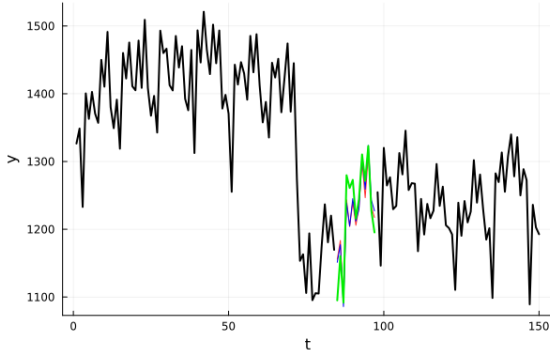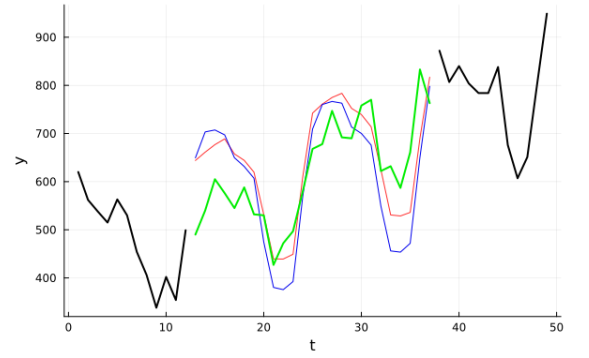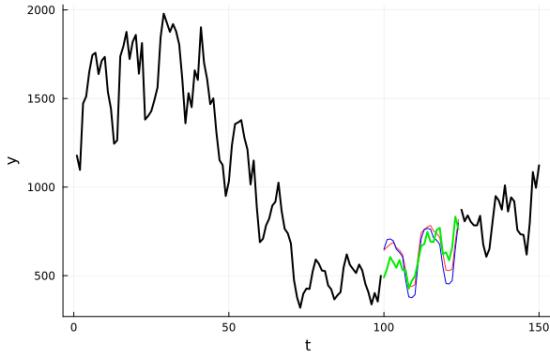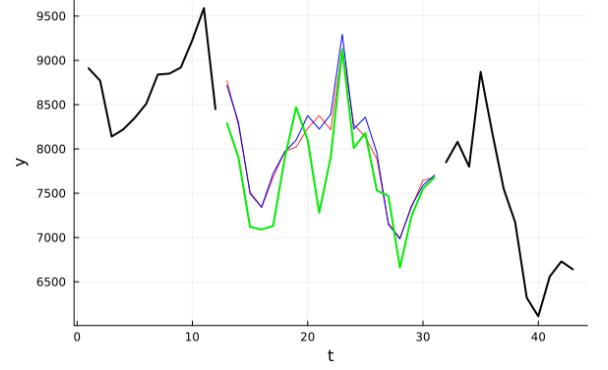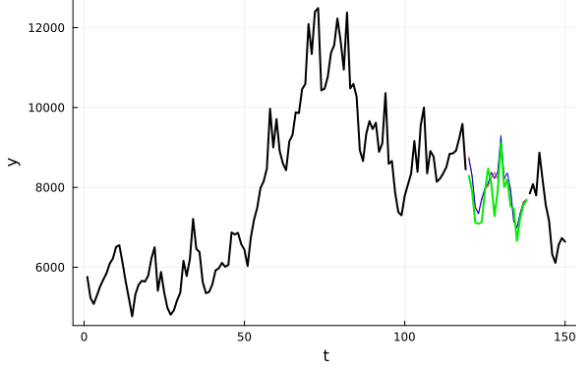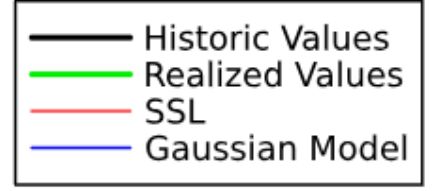
(a) Filled missing values of both the gaussian and the SSL ($\alpha = 0.1 - AIC$) models in three different time series.

(b) Zoom of the filled missing values of both the gaussian and the SSL ($\alpha = 0.1 - AIC$) models in three different time series from the M4 dataset.

Figure 3: Model Forecasts

2.5   Forecasting via Extrapolation

A robust forecasting scheme within a time series analysis framework is paramount for informed decision-making across various domains, including finance, economics, and climate science. Accurate forecasts enable stakeholders to anticipate future trends, allocate resources efficiently, and mitigate potential risks. Moreover, reliable forecasts facilitate proactive planning, aiding businesses in optimizing their decisions. Therefore, the development and implementation of effective forecasting methodologies are essential for fostering resilience and sustainability in dynamic environments. We propose an intuitive forecasting scheme for the SSL framework, relying on the extrapolation of each component of the high-dimensional regression matrix and the utilization of estimated coefficients. This approach not only simplifies the forecasting process but also ensures accuracy and adaptability, making it a valuable tool for addressing the forecasting needs of complex systems.

Proposition 2.2. *Assuming that the values of $\Theta = (\mu_1, \xi_2, \ldots, \xi_T, \nu_1, \zeta_2, \ldots, \zeta_5, \gamma_1, \ldots, \gamma_s, \omega_s, \ldots, \omega_T)^\top$ are given and $E[\varepsilon_t | \Theta] = 0, E[\xi_t | \Theta] = 0, E[\zeta_t | \Theta] = 0,$ and $E[\omega_t | \Theta] = 0$ for every $t > T$, then a forecast for $t > T$ is defined as*

$$\hat{y}_t = \mu_1 + (t-1)\nu_1 + \sum_{\tau=2}^{T-1}(t-\tau)\zeta_\tau + \sum_{\tau=2}^{T}\xi_\tau + \gamma_{m_t} + \sum_{\tau \in M_T}(\omega_\tau - \omega_{\tau-1}), \quad \forall t > T \tag{25}$$

Proof. Given the Equation 18 and that Θ is a given set of coefficients $(\mu_1, \xi_2, \ldots, \xi_T, \nu_1, \zeta_2, \ldots, \zeta_5,$ $\gamma_1, \ldots, \gamma_s, \omega_s, \ldots, \omega_T)^\top$, the value of y_t for $t > T$ is defined as:

$$y_t = \mu_1 + (t-1)\nu_1 + \sum_{\tau=2}^{t-1}(t-\tau)\zeta_\tau + \sum_{\tau=2}^{t}\xi_\tau + \gamma_{m_t} + \sum_{\tau \in M_t}(\omega_\tau - \omega_{\tau-1}) + \varepsilon_t. \tag{26}$$

Therefore, the forecast will be calculated as:

$$\begin{aligned}
\hat{y}_t = E[y_t | \Theta] = {}& \mu_1 + (t-1)\nu_1 + \sum_{\tau=2}^{T}(t-\tau)E[\zeta_\tau | \Theta] + \sum_{\tau=T+1}^{t-1}(t-\tau)E[\zeta_\tau | \Theta] \\
& + \sum_{\tau=2}^{T}E[\xi_\tau | \Theta] + \sum_{\tau=T+1}^{t}E[\xi_\tau | \Theta] + \gamma_{m_t} \\
& + \sum_{\tau \in M_T}\left(E[\omega_\tau | \Theta] - E[\omega_{\tau-1} | \Theta]\right) \\
& + \sum_{\tau \in M_t \setminus M_T}\left(E[\omega_\tau | \Theta] - E[\omega_{\tau-1} | \Theta]\right) + E[\varepsilon_t],
\end{aligned} \tag{27}$$

where the term $M_t \setminus M_T$ specifies the set of elements that belongs to M_t and do not belongs to M_T.

Now, for each innovation $\kappa \in \{\xi, \zeta, \omega\}$, we have $E[\kappa_\tau | \Theta] = \kappa_\tau, \forall \tau \leq T$ and $E[\kappa_\tau | \Theta] = 0, \forall \tau > T$, thereby

$$\hat{y}_t = \mu_1 + (t-1)\nu_1 + \sum_{\tau=2}^{T-1}(t-\tau)\zeta_\tau + \sum_{\tau=2}^{T}\xi_\tau + \gamma_{m_t} + \sum_{\tau \in M_t}(\omega_\tau - \omega_{\tau-1}).$$

$\square$

For a better visualization, we illustrate the forecasting process by showing the process of fit and forecast (for the case where T = 5 and s = 2) and a forecasting process of two steps ahead. The fit process will estimate the $\theta_0 = \mu_1$ and $\theta = [\nu_1, \zeta_1, \zeta_2, \zeta_3, \zeta_4, \xi_1, \xi_2, \xi_3, \xi_4, \gamma_1, \gamma_2, \omega_2, \omega_3, \omega_4]$ parameters.

$$
\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{pmatrix}
=
\begin{array}{c}
\begin{matrix} \mu_1 & \xi_2 & \xi_3 & \xi_4 & \xi_5 & \nu_1 & \zeta_2 & \zeta_3 & \zeta_4 & \gamma_1 & \gamma_2 & \omega_2 & \omega_3 & \omega_4 & \omega_5 \end{matrix} \\
\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 & 2 & 1 & 0 & 0 & 1 & 0 & -1 & 1 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 3 & 2 & 1 & 0 & 0 & 1 & 0 & -1 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 4 & 3 & 2 & 1 & 1 & 0 & -1 & 1 & -1 & 1
\end{pmatrix}
\end{array}
\cdot
\begin{pmatrix} \theta_0 \\ \theta \end{pmatrix} \tag{28}
$$

Finally, the forecast will be obtained by employing the estimated $\hat{\theta}_0$ and $\hat{\theta}$ in the following equation:

$$
\begin{pmatrix} \hat{y}_6 \\ \hat{y}_7 \end{pmatrix}
=
\begin{array}{c}
\begin{matrix} \mu_1 & \xi_2 & \xi_3 & \xi_4 & \xi_5 & \nu_1 & \zeta_2 & \zeta_3 & \zeta_4 & \gamma_1 & \gamma_2 & \omega_2 & \omega_3 & \omega_4 & \omega_5 \end{matrix} \\
\begin{pmatrix}
1 & 1 & 1 & 1 & 1 & 5 & 4 & 3 & 2 & 0 & 1 & 0 & -1 & 1 & -1 \\
1 & 1 & 1 & 1 & 1 & 6 & 5 & 4 & 3 & 1 & 0 & -1 & 1 & -1 & 1
\end{pmatrix}
\end{array}
\cdot
\begin{pmatrix} \hat{\theta}_0 \\ \hat{\theta} \end{pmatrix} \tag{29}
$$

.

We now present in Figure 4 an illustrative example of our proposed methodology's forecasting capabilities and compared it with its Gaussian counterpart across three time series sourced from the M4 dataset.
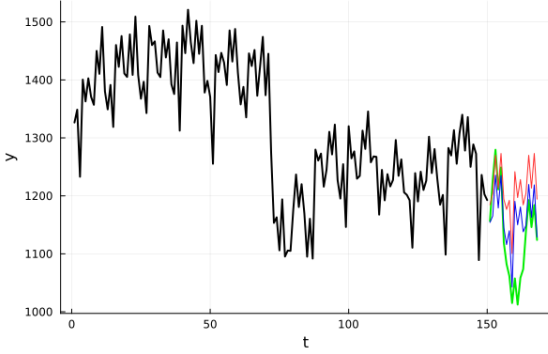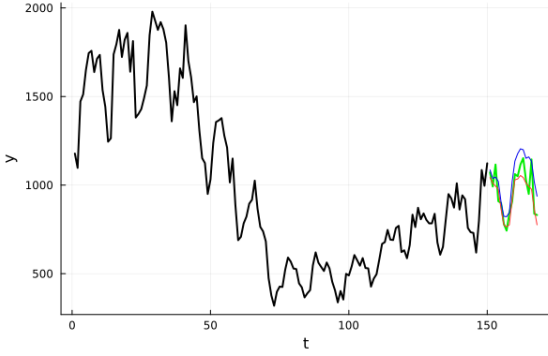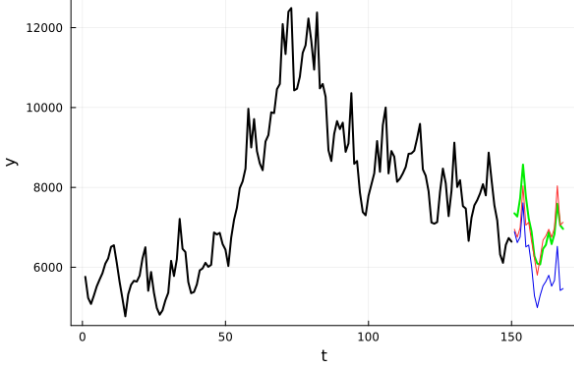
(a) Forecast values of both the gaussian and the SSL ($\alpha = 0.1 - AIC$) models in three different time series.
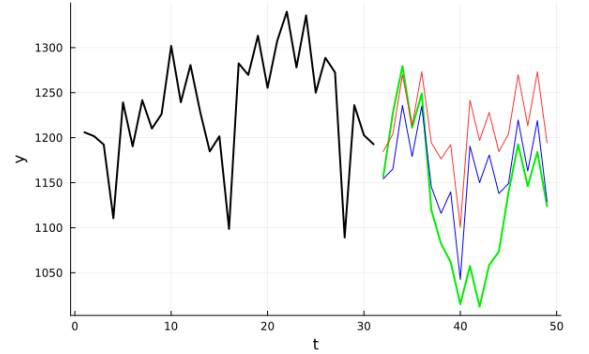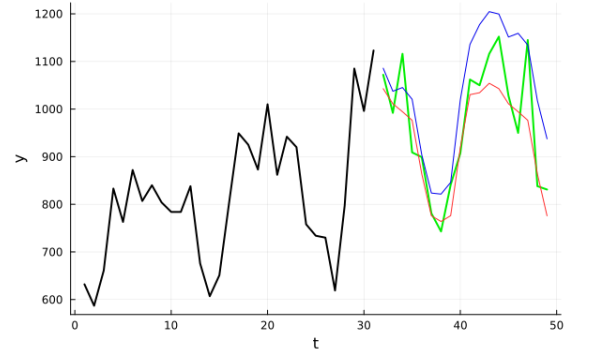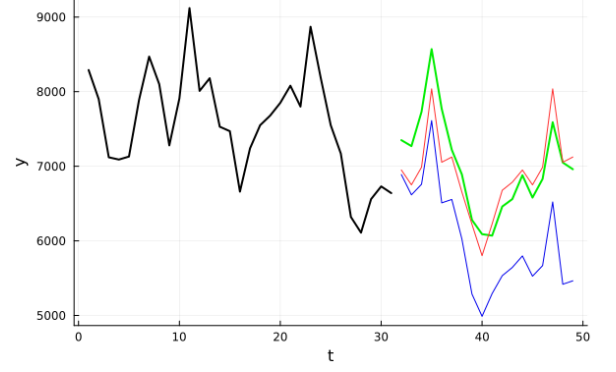
(b) Zoom of the forecast values of both the gaussian and the SSL ($\alpha = 0.1 - AIC$) models in three different time series from the M4 dataset.

Figure 4: Model Forecasts

It is noteworthy that both the Gaussian benchmark and the SSL model illustrate proficiency in capturing level, slope, and seasonal shifts, thereby furnishing credible forecasts. A more comprehensive evaluation of the forecasting performance of SSL is presented in Section 4.1.

3 State Space Learning: Refinements and Extensions

In this section, we explore several extensions and refinements to enhance the capabilities of the state space learning framework. These extensions broaden the applicability and robustness of the framework across various domains and analytical challenges. Firstly, we introduce a subset selection method tailored to accommodate exogenous variables within high-dimensional spaces, enabling more comprehensive modeling of time series in a big-data world. Secondly,

we delve into outlier detection and treatment as well as techniques to avoid overfitting, a crucial development for ensuring the reliability and accuracy of model outputs in the presence of noisy or erroneous data points. Lastly, we discuss the extension of the framework beyond traditional time series analysis, allowing for its utilization in many engineering applications that resort to linear state space formulation.

3.1   Subset Selection of Exogenous Variables

It is worth noting that the proposed methodology and its associated estimation procedure possess the capability to seamlessly adapt to best subset selection with a slight modification in its formulation

$$y_t = \mu_1 + (t-1)\nu_1 + \sum_{\tau=2}^{t-1}(t-\tau)\zeta_\tau + \sum_{\tau=2}^{t}\xi_\tau$$

$$\gamma_{m_t} + \sum_{\tau \in M_t}(\omega_\tau - \omega_{\tau-1}) + X_t\beta + \varepsilon_t. \tag{30}$$

In this adapted framework, a new component $X_t\beta$ emerges, where X_t contains the observations of the exogenous features, and β represents the coefficients associated with these exogenous variables, subject to estimation by the model.

Note that the adapted estimation procedure will be modified. Initially, in the first step (Equation 20), we redefine $\theta = (\xi_2, \ldots, \xi_T, \zeta_2, \ldots, \zeta_T, \omega_s, \ldots, \omega_T, \beta)$. In the second step (Equation 21) there will be the addition of the term $\sum_{i=1}^{p} w_{\beta_i}\left((1-\alpha)\frac{1}{2}\|\beta_i\|_2^2 + \alpha\|\beta_i\|_1\right)$, where $w_{\beta_i} = |\hat{\beta}_i|^{-1}$ and p is the number of exogenous features, $\hat{\beta}_i$ is the estimated value of β_i on the first step of the estimation procedure.

Furthermore, the integration of this term into the forecasting framework is straightforward. To accomplish this, we adapt Equation 2.2:

$$\hat{y}_t = \mu_1 + (t-1)\nu_1 + \sum_{\tau=2}^{T-1}(t-\tau)\zeta_\tau + \sum_{\tau=2}^{T}\xi_\tau + \gamma_{m_t} + \sum_{\tau \in M_T}(\omega_\tau - \omega_{\tau-1}) + \hat{X}_t\beta, \quad \forall t > T, \tag{31}$$

wherein $\hat{X}_t$ denotes the predicted values of the exogenous variables for $t > T$.

By integrating a distinctive framework, the proposed methodology not only concurrently selects and estimates intrinsic time series components but also fills a critical void in the literature by simultaneously addressing the selection and estimation of exogenous features in both stationary and non stationary time series. This enhancement significantly augments the model's versatility and utility in statistical modeling and forecasting applications.

Note that the forecasting via extrapolation procedure can be utilized in the presence of exogenous features as long as a forecast for each variable is provided for the underlying model in order to build their columns in the forecast regression matrix. This leads to an adaptation of Equation 2.2 with the addition of the term $\hat{X}_t$ (the forecast of the exogenous variables at a given time "t").

3.2   Robust Forecasting: Handling Outliers, Identification and Overfitting

In this section we will propose a methodology to handle outlier observations, an identification issue for the extrapolation procedure and a strategy to handle potential overfitting.

With the regression formulation we are able to adapt our model to account for the presence of sporadic values that deviate from the natural behavior of the model and must be disregarded in estimation and forecasting or simulation. These observations, commonly known as outliers, require a distinct treatment to avoid misleading results. To address this, we introduce additional columns of dummy variables to the high-dimensional regression matrix, equivalent to adding an identity matrix to the underlying model. This enables regularization techniques to sparsely identify and isolate outliers while simultaneously estimating the best-fit model for the studied time series. By incorporating this approach, we can enhance the accuracy and reliability of our predictions and simulations, particularly in situations where outliers are prevalent.

The formulation containing the outlier component $D_{\tau,t}\, o_t$ is presented below:

$$y_t = \mu_1 + \sum_{\tau=2}^{t}\xi_\tau + (t-1)\nu_1 + \sum_{\tau=2}^{t-1}(t-\tau)\zeta_\tau + \gamma_{m_t} + \sum_{\tau \in M_t}(\omega_\tau - \omega_{\tau-1}) + X_t\beta + \sum_{\tau=1}^{t}D_{\tau,t}\, o_\tau + \varepsilon_t, \tag{32}$$

where $D_{\tau,t}$ is a dummy term that assumes value equal to 1 when $\tau = t$ and 0 otherwise. Moreover, o_τ represent the coefficient estimated for each outlier component. Note that this procedure is similar to the one proposed in [15] in the L1 filtering context.

Analogous to the exogenous features, the adapted estimation procedure will be modified. In Equation 20), we redefine $\theta = (\xi_2, \ldots, \xi_T, \zeta_2, \ldots, \zeta_T, \omega_s, \ldots, \omega_T, \beta, o_1, \ldots, o_T)$. In Equation 21, we add the term $\sum_{t=1}^{T} w_{o_t}\left((1-\alpha)\frac{1}{2}\|o_t\|_2^2 + \alpha\|o_t\|_1\right)$, where $w_{o_t} = |\hat{o}_t|^{-1}$ and $\hat{o}_t$ is the estimated value of o_t on the first step of the estimation procedure.

Note that the addition of the outlier component in the forecasting procedure is a straightforward process. Specifically, the outlier component is anticipated not to affect the forecast due to the constant 0 value of the dummy term $D_{\tau,t}$ for $t > T$. Therefore, the forecasting equation 2.2 will remain the same.

After handling outliers, we propose small modifications in the estimation to robustify the forecasting procedure. First, it is crucial to recognize that the regression matrix underlying the proposed model, as illustrated in Equation 19, encounters an identification challenge due to the indistinguishable nature of the regressors associated with terms $(\xi_T, \zeta_{T-1}, \omega_T, o_T)$. Note that this distinction has no impact over the estimation procedure. However, for the extrapolation procedure it is extremely relevant as it it will state what time series dynamic was changed at time T. As it is not possible to affirm that a change over a level, slope or seasonal component occurred with a single observation "y_T" we opted to constrain the values of "ξ_T", "ζ_{T-1}" and "ω_T" at 0 and keep o_T unconstrained to robustify the forecast procedure.

More generally, the final terms of the innovation components are estimated based on a limited set of observations, leading to overfitting the model to the most recent data. This is more significant for the slope innovation ζ_t, where spurious slope changes at the end of a time series can degrade forecasting accuracy. We argue that such spurious changes might result from the model using too few observations to infer a slope or seasonal shock, which is not a parsimonious approach. To address this challenge, we propose constraining the latest terms of the ζ_t and ω_t shocks to zero for $t > T - s$. In our numerical experiments, we implement a threshold with $s = 12$ (given that the tested time series have a monthly granularity) for these components. Specifically, we apply the constraints $\zeta_\tau = 0 \; \forall \tau \in (T - 12, T]$ and $\omega_\tau = 0 \; \forall \tau \in (T - 12, T]$.

3.3   SSL as an initialization procedure for for the Gaussian Basic Structural Model

As discussed in Section 1, the traditional state-space model framework depends on good initialization points, given that its estimation procedure relies on nonlinear optimization.

Another use case of the proposed State Space Learning framework is that it leads to a straightforward initialization for its Gaussian counterpart. For the Basic Structural Model case, the initialization values (the variances of innovation terms and residuals) can be obtained as $\hat{\sigma}_\kappa^2 = \|\kappa\|_2^2, \forall \kappa \in \{\xi, \zeta, \omega, \varepsilon\}$.

It is noteworthy that the traditional Basic Structural Model Explanatory, which incorporates exogenous features, lacks the capability to efficiently perform best subset selection. This limitation is particularly significant in the context of today's big data landscape. In contrast, the SSL framework addresses this shortcoming by incorporating an efficient best subset selection process.

Thus, the SSL results can be utilized to reduce the dimensionality of the set of explanatory variables for the Basic Structural Model. In this case, the Basic Structural Model can consider only the pre-selected variables, whose coefficients (β) estimated within the SSL methodology are non-zero. Subsequently, all parameters of the Basic Structural Model can be initialized using the values derived from the SSL estimation methodology.

Expanding the initialization process to accommodate the outlier component detailed in Section 3.2 involves a strategic approach. Specifically, observations (y_τ) characterized by non-zero outlier components (o_τ) are treated as missing values. Leveraging the flexibility of traditional state-space models in handling missing data, this treatment effectively excludes outlier-influenced observations during the initialization phase. By doing so, the proposed initialization process for the Basic Structural Model based on the SSL estimates mitigates the impact of outlier observations, thereby enhancing the robustness of subsequent model estimation and analysis.

3.4   Generalized State-Space Learning

A general additive state-space model can be written as:

$$y_t = Z_t \alpha_t + d_t + \varepsilon_t \tag{33}$$
$$\alpha_t = T_{t-1} \alpha_{t-1} + c_{t-1} + R_{t-1} \eta_{t-1}, \tag{34}$$

where y_t is typically an observed data point in a time series analysis. The term Z_t represents the observation matrix at time t, which maps the state variable α_t to the observed variable y_t. The state variable α_t captures the underlying, unobserved components that evolves over time. The terms d_t and ε_t represent the deterministic and random components, respectively, of the observation equation. α_t represents the state variable at the next time step, which evolves over time. The term T_t denotes the state transition matrix, indicating how the state variable at time t evolves to the state variable at time t+1. The terms c_t represent the deterministic component of the state equation. R_t denotes the transition matrix for the random component or process noise and η_t represents the unobserved, random disturbances or shocks that affect the evolution of the state variable α_t.

We now extend the proposition that a basic structural model can be written as a high dimension regression to any additive state-space model. This overarching generalization holds significant implications, given its potential applicability across various disciplines reliant on state-space models. It introduces an alternative estimation approach for the Kalman Filter, thereby expanding the methodological repertoire available to researchers and practitioners in these fields.

To facilitate the following proof, we start by iterating the state equation as:

$$\alpha_t = T_{t-1}\alpha_{t-1} + c_{t-1} + R_{t-1}\eta_{t-1} \tag{35}$$

$$\alpha_t = T_{t-1}\left(T_{t-2}\alpha_{t-2} + c_{t-2} + R_{t-2}\eta_{t-2}\right) + c_{t-1} + R_{t-1}\eta_{t-1} \tag{36}$$

$$\ldots \tag{37}$$

$$\alpha_t = \sum_{j=1}^{t-1} T_{t-j}\alpha_1 + \sum_{j=1}^{t-1} c_{t-j}\prod_{i=1}^{t-j} T_{t-i} + \sum_{j=1}^{t-1} R_{t-j}\eta_{t-j}\prod_{i=1}^{t-j} T_{t-i}. \tag{38}$$

Proposition 3.1. *Any state-space model of the form of Equations 33-34 can be written as a high dimension regression:* $y_t = Z_t\left(\sum_{j=1}^{t-1} T_{t-j}\alpha_1 + \sum_{j=1}^{t-1} c_{t-j}\prod_{i=1}^{t-j} T_{t-i} + \sum_{j=1}^{t-1} R_{t-j}\eta_{t-j}\prod_{i=1}^{t-j} T_{t-i}\right) + d_t + \varepsilon_t.$

Proof. We also prove Proposition 3.1 by induction of the state equation. Our base case $\alpha_2 = T_1\alpha_1 + c_1 + R_1\eta_1$ holds true for $t = 2$.

The induction hypothesis $y_t = Z_t\left(\sum_{j=1}^{t-1} T_{t-j}\alpha_1 + \sum_{j=1}^{t-1} c_{t-j}\prod_{i=1}^{t-j} T_{t-i} + \sum_{j=1}^{t-1} R_{t-j}\eta_{t-j}\prod_{i=1}^{t-j} T_{t-i}\right) + d_t + \varepsilon_t$ (for $t > 2$) can be confirmed by substituting Equation 38 in the observation Equation 33.

$\square$

Note that as the system variables $(Z_t, T_t, d_t, c_t, R_t)$ are known. Therefore we can create a new y' variable as:

$$y'_t = y_t - Z_t\sum_{j=1}^{t-1} c_{t-j}\prod_{i=1}^{t-j} T_{t-i} - d_t, \tag{39}$$

and finally, we are able to present the general representation of a state-space formulation as a high dimension regression:

$$y'_t = Z_t\sum_{j=1}^{t-1} T_{t-j}\alpha_1 + Z_t\sum_{j=1}^{t-1} R_{t-j}\eta_{t-j}\prod_{i=1}^{t-j} T_{t-i} + \varepsilon_t. \tag{40}$$

This equation can be divided in two terms. The first one $(Z_t\sum_{j=1}^{t-1} T_{t-j}\alpha_1)$ is referent to the estimation of the initial states α_1 (referent to the θ_0 variable described in Section 2.3). The second term $(Z_t\sum_{j=1}^{t-1} R_{t-j}\eta_{t-j}\prod_{i=1}^{t-j} T_{t-i})$ is referent to the estimation noise terms η (referent to the θ variable described in Section 2.3). The terms Z_t, R_t and T_t are known as system matrices and will form the a set of columns in the regression matrix.

Finally, we could employ the exactly proposed estimation procedure 2.3 by grouping the state variables weights and not penalizing an initial state equivalent to an intercept (if existing).

Analogous to traditional state-space models, this versatile framework extends its scope beyond time series analysis, finding application in various domains. Analogous to its Gaussian counterpart, this generalized model may be object of study in vehicle tracking, physics, finance, and more. Additionally, it is instrumental in machine learning, speech recognition, natural language processing, and anomaly detection tasks, enabling the modeling of latent factors. These diverse applications highlight the wide-ranging significance of this generalized structural model, enhancing analytical capabilities across multidisciplinary domains.

4   Numerical Results

In this section, we conduct a comprehensive numerical assessment of the capabilities of the State Space Learning (SSL) framework in the context of time series analysis. Initially, we demonstrate the predictive prowess of SSL through extensive empirical analysis, evaluating its out-of-sample performance across 48,000 monthly time series from the M4 dataset. Through this experiment, we rigorously compare SSL against selected benchmarks, employing various model selection criteria and robust forecasting strategies to address outliers and overfitting issues. Our results underscore the superior performance of SSL over these benchmarks. Furthermore, we present simulation results illustrating SSL's effectiveness in addressing the best subset selection problem, particularly in selecting and estimating the impact of exogenous variables within high-dimensional spaces. Our findings provide compelling evidence of SSL's superior performance in best subset selection when compared to selected benchmarks.

4.1   Empirical Evaluation of SSL Predictive Power

In this section, we undertake a thorough evaluation of the SSL robust forecasting methodology. We compare the performance of two variants of the SSL model, namely SSL and SSL-O (which handles outliers), against the benchmark Gaussian Basic Structural Model, the SSL-initialized Gaussian Basic Structural Model, and the auto Sarima benchmark [12] using the M4 dataset.

The M4 dataset comprises a vast collection of 100,000 time series spanning various frequencies and originating from diverse sectors, including business, finance, and economic forecasting. For our analysis, we focus on a subset of 48,000 monthly time series from this dataset. Notably, the forecasting horizon for these monthly time series extends to 18 steps ahead.

To rigorously assess our results, we utilize standard performance metrics including sMAPE, MASE, and OWA, consistent with those adopted in [27]. The sMAPE measures error scaling using the average between forecasted and actual values. MASE scales error by the average error of a naive predictor, which replicates observations measured s periods in the past, thereby accounting for seasonality. Additionally, we employ a competition-specific metric, Overall Weighted Average (OWA), used to rank competition entries in the M4 competition [27]. OWA is designed such that a seasonally-adjusted naive forecast (Naïve 2 [21]) achieves a score of 1.0. Specifically, the OWA of sMAPE and MASE is computed by dividing their total value by the corresponding value of Naïve 2 to obtain the relative sMAPE and relative MASE, respectively, and then calculating their simple arithmetic mean.

For a given forecast horizon H, a T length time series y, we define

$$sMAPE = \frac{2}{H} \sum_{t=1}^{H} \frac{|y_{T+1} - \hat{y}_{T+t}|}{|y_{T+t}| + |\hat{y}_{T+t}|} \cdot 100(\%) \tag{41}$$

$$MASE = \frac{1}{H} \frac{\sum_{t=T+1}^{T+H} |y_t - \hat{y}_t|}{\frac{1}{T-s} \sum_{t=s+1}^{T} |y_t - y_{t-s}|} \tag{42}$$

$$OWA = \frac{1}{2} \left(\frac{sMAPE}{sMAPE^{NAIVE2}} + \frac{MASE}{MASE^{NAIVE2}} \right), \tag{43}$$

where we denote the time series forecast as $\hat{y}$ and s as the periodicity of the data (12 for monthly series).

To compare the SSL against the benchmark models, we utilized the robust forecasting technique presented in Section 3.2 with different values of alpha (the coefficient that sets the convex combination between Lasso and Ridge Regression) and different information criteria to obtain the λ value. We also utilized the result of the Gaussian Basic Structural Model with the initialization of the best-tested SSL model in the OWA metric (α=0.1 and AIC as information criteria). The presented results were obtained using Julia programming language with the GLMNet.jl [1] package to estimate elastic net models and the python package [25] to estimate the Gaussian Basic Structural Model. For Both SSL and SSL-O we calibrated the number of train observations to the last five years of data (60 observations) as this leads to slightly better results for this particular model, which indicates that it does not need much data to present robust forecasts. We also compare the results against the benchmark results of Auto Sarima (published in the M4 competition results [27]).

Moreover, we present the results of the applied model in two different ways: one considering the outlier component — SSL-O — and another one without it (SSL). We present Table 1 comparing the mean in each metric for the benchmark models and the best performing configurations for the proposed methodology.

Table 1: Performance on the monthly time series of M4 dataset (lower values are better).

Model	MASE	sMAPE	OWA
Gaussian Basic Structural Model	1.012	15.689	1.020
SSL initialized Gaussian Basic Structural Model	0.965	14.894	0.970
AUTO SARIMA	**0.930**	13.443	0.903
SSL-O (AIC, $\alpha = 0.0$)	0.984	13.942	0.946
SSL (AIC, $\alpha = 0.0$)	0.986	14.367	0.961
SSL-O (AIC, $\alpha = 0.1$)	0.936	**12.975**	**0.89**
SSL (AIC, $\alpha = 0.1$)	0.992	14.208	0.959
SSL-O (AIC, $\alpha = 0.3$)	0.954	13.103	0.903
SSL (AIC, $\alpha = 0.3$)	1.004	14.252	0.966
SSL-O (AIC, $\alpha = 0.5$)	0.962	13.168	0.909
SSL (AIC, $\alpha = 0.5$)	1.008	14.25	0.968
SSL-O (AIC, $\alpha = 0.7$)	0.965	13.195	0.911
SSL (AIC, $\alpha = 0.7$)	1.009	14.258	0.969
SSL-O (AIC, $\alpha = 0.9$)	0.966	13.214	0.912
SSL (AIC, $\alpha = 0.9$)	1.01	14.269	0.969
SSL-O (AIC, $\alpha = 1.0$)	0.966	13.223	0.912
SSL (AIC, $\alpha = 1.0$)	1.01	14.272	0.97
SSL-O (BIC, $\alpha = 0.0$)	0.984	13.942	0.946
SSL (BIC, $\alpha = 0.0$)	0.986	14.367	0.961
SSL-O (BIC, $\alpha = 0.1$)	1.183	14.654	1.064
SSL (BIC, $\alpha = 0.1$)	1.007	14.158	0.964
SSL-O (BIC, $\alpha = 0.3$)	0.954	13.106	0.903
SSL (BIC, $\alpha = 0.3$)	1.003	14.244	0.965
SSL-O (BIC, $\alpha = 0.5$)	0.962	13.168	0.909
SSL (BIC, $\alpha = 0.5$)	1.007	14.244	0.967
SSL-O (BIC, $\alpha = 0.7$)	0.965	13.195	0.911
SSL (BIC, $\alpha = 0.7$)	1.008	14.255	0.968
SSL-O (BIC, $\alpha = 0.9$)	0.966	13.214	0.912
SSL (BIC, $\alpha = 0.9$)	1.009	14.265	0.969
SSL-O (BIC, $\alpha = 1.0$)	0.966	13.223	0.913
SSL (BIC, $\alpha = 1.0$)	1.009	14.268	0.969

A noteworthy finding emerges from the analysis, revealing the notable superiority of both the top-performing SSL and SSL-O models in terms of all metrics against its Gaussian Counterpart. These models exhibited remarkable outcomes, surpassing not only the benchmark Basic Structural Model but also against its initialized version. Intriguingly, the initialized Gaussian model demonstrated superior performance across all metrics when compared to the non initialized counterpart. Moreover, the proposed methodology got better results than the auto Sarima benchmark in terms of two of the studied metrics including the official metric of the M4 competition (OWA).

Further, the analysis indicates that the outlier component played a crucial role in enhancing the model's efficacy, as all of the variants of the proposed formulation outperformed their counterparts across all metrics.

Moreover, the performance improvement observed with the initialized Gaussian model, as opposed to its non-initialized counterpart, is particularly significant. These results substantiate the assertion that the SSL model serves as an effective initializer for the conventional Basic Structural Model.

Taken together, these outcomes showcase the potential of the SSL model to outshine its benchmark counterpart in forecasting tasks. The efficacy of this approach highlights its significance in advancing the field of predictive modeling and promoting better decision-making in various domains.

4.2 Best Subset Selection: A Simulation Assessment

In the realm of conventional time series modeling, consensus on the optimal subset selection method remains elusive. While regularization techniques like Lasso offer a promising avenue, they rely on stationary hypotheses that often fail to hold in many real-world applications. Conversely, stepwise approaches such as forward and backward selection, though heuristic, come with significant computational costs.

In this study, we embark on evaluating the efficacy of our proposed methodology for best subset selection. We benchmark our approach against two established methods outlined below:

- SS+FW: A stepwise forward selection (FW) process, where exogenous features are iteratively added to the state-space (SS) Gaussian basic structural model until a predetermined information criterion ceases to improve. This method utilizes a basic structural model estimated with [25].

- SS+AdaLasso: A two-stage procedure involving the estimation of a traditional state space (SS) Gaussian basic structural model (via [25]), followed by the removal of trend and seasonal components. Subsequently, an Adaptive Lasso (AdaLasso) is performed on the remaining values to select the optimal subset (we utilize $\alpha = 1$, where α is the parameter that balances the L2 and L1 penalties).

To commence our simulation experiment, we first outline the data generation process. Our approach begins by constructing the dependent variables, which are derived from time series generated using Basic Structural Models:

$$y_t = \mu_t + \gamma_t + X_t \beta + \varepsilon_t, \qquad\qquad \varepsilon \sim N(0, \sigma_\varepsilon^2) \qquad\qquad (44)$$

$$\mu_t = \mu_{t-1} + \nu_t + \xi_t, \qquad\qquad \xi \sim N(0, \sigma_\xi^2) \qquad\qquad (45)$$

$$\nu_t = \nu_{t-1} + \zeta_t, \qquad\qquad \zeta \sim N(0, \sigma_\zeta^2) \qquad\qquad (46)$$

$$\gamma_t = -\sum_{j=1}^{s-1} \gamma_{t-j} + \omega_t, \qquad\qquad \omega \sim N(0, \sigma_\omega^2), \qquad\qquad (47)$$

where for this experiment we define $\mu_0 = 1$, $\nu_0 = 0.001$, $\gamma_0 =$[1.5, 2.6, 3.0, 2.6, 1.5, 0.0, -1.5, -2.6, -3.0, -2.6, -1.5], $\sigma_\xi = 0.5$, $\sigma_\zeta = 0.001$, $\sigma_\omega = 0.3$ and , and $\sigma_\varepsilon = 0.2$.

Moreover, we defined each exogenous vector X_t as a Seasonal Autoregressive Integrated Moving Average (SARIMA) process with parameters $(p, 0, q)(P, 0, Q)_{12}$. Within this process, each coefficient was stochastically drawn from a uniform distribution $\theta, \Theta, \phi, \Phi \sim N(0, 0.2)$. Furthermore, p and q were selected from $\{0, 1, 2, 3, 4, 5\}$, while P and Q were chosen from $\{0, 1\}$. Additionally, an error term, sampled from a normal distribution $N(0, 3)$, was introduced for each exogenous feature. The coefficients β of the relevant features were assigned as 1.

To conduct a thorough analysis, we explored a range of combinations denoted by the tuple (M, K), where M is the number of candidates and K is the number of true relevant features. Specifically, we varied M within the set $\{3, 5, 8, 10\}$ and K within $\{50, 100\}$. For each configuration of (M, K), we performed a Monte Carlo simulation comprising 500 samples.

An essential consideration arises upon parameter sorting in the described SARIMA framework: the resulting process may exhibit non-stationarity and lack invertibility. To ensure equitable comparisons with the proposed benchmarks, which lack the capability to manage non-stationary exogenous variables, we implement a filtering mechanism. This involves restricting exogenous variables to only include those generated within the SARIMA framework that successfully pass the unit root test [7] for both Autoregressive and Moving Average terms.

We train the SSL model for a fixed value of $\alpha = 1$ (for a fair comparison with SS+AdaLasso benchmark) and test two information criteria (AIC and BIC) for selection of the λ parameter.

For the first benchmark model, we employed a forward selection model to identify significant predictors within a basic structural explanatory (estimated via [25]) framework. Forward selection is a feature selection method commonly used in statistical modeling to iteratively add variables to the model, starting with zero variables. At each step, the variable that provides the greatest improvement in a given information criteria is added, and the process continues until the stopping criterion is met. In our case, we tested both the AIC and BIC as stopping criteria.

For our second benchmark model, we begin by estimating a basic structural model using the methodology described in [25]. Subsequently, we remove the level and seasonal components from the original time series y_t. This process yields a residual series defined as $y_t^* = y_t - \hat{\mu}_t - \hat{\gamma}_t$, where $\hat{\mu}_t$ and $\hat{\gamma}_t$ are the level and seasonal component obtained with a Gaussian structural model. We then apply an Adaptive Lasso technique to y_t^*, facilitating best subset selection. This two-step approach combines state-of-the-art methods for estimating time series basic structural models and performing variable selection through statistical learning.

In contrast, our proposed SSL approach integrates these two steps into a single cohesive process. By co-optimizing both component extraction and exogenous variable selection, SSL streamlines the modeling procedure and potentially enhances performance compared to the separate two-step process. Finally, we run our controlled test and present the results Tables 2-3.

Table 2 depicts the variable selection outcomes for each model selection technique. Various pertinent statistics are presented: Panel (a) showcases the proportion of repetitions where the accurate model was selected, indicating inclusion of all relevant variables and exclusion of all irrelevant regressors from the final model; Panel (b) illustrates the proportion of repetitions where all relevant variables were included; Panel (c) exhibits the proportion of relevant variables that were incorporated; Panel (d) illustrates the proportion of irrelevant variables that were excluded; Panel (e) displays the mean count of variables included; Panel (f) showcases the average count of irrelevant variables included.

M\K	SSL (AIC) 50	SSL (AIC) 100	SSL (BIC) 50	SSL (BIC) 100	SS+AdaLasso (AIC) 50	SS+AdaLasso (AIC) 100	SS+AdaLasso (BIC) 50	SS+AdaLasso (BIC) 100	SS+FW (AIC) 50	SS+FW (AIC) 100	SS+FW (BIC) 50	SS+FW (BIC) 100
Panel (a): Correct Sparsity Pattern												
3	0.652	0.552	0.76	0.682	0.002	0.002	0.178	0.018	0.488	0.458	0.492	0.458
5	0.714	0.634	0.778	0.718	0.0	0.0	0.064	0.002	0.134	0.132	0.132	0.13
8	0.79	0.664	0.856	0.734	0.0	0.0	0.002	0.0	0.004	0.008	0.004	0.002
10	0.796	0.628	0.844	0.674	0.0	0.0	0.0	0.0	0.002	0.002	0.0	0.002
Panel (b): True Model Included												
3	0.996	0.994	0.996	0.994	0.994	0.988	0.99	0.986	0.654	0.63	0.656	0.632
5	0.99	0.986	0.99	0.986	0.964	0.934	0.916	0.93	0.14	0.14	0.138	0.138
8	0.984	0.986	0.984	0.986	0.878	0.824	0.762	0.822	0.004	0.008	0.004	0.002
10	0.984	0.942	0.984	0.942	0.824	0.574	0.662	0.57	0.002	0.002	0.0	0.002
Panel (c): Fraction of Relevant Variables Included												
3	0.997	0.995	0.997	0.995	0.996	0.994	0.994	0.993	0.895	0.883	0.895	0.883
5	0.991	0.99	0.991	0.99	0.982	0.975	0.96	0.973	0.666	0.668	0.665	0.665
8	0.988	0.99	0.988	0.99	0.96	0.962	0.897	0.962	0.462	0.429	0.439	0.408
10	0.989	0.972	0.989	0.972	0.965	0.911	0.856	0.906	0.377	0.321	0.338	0.29
Panel (d): Fraction of Irrelevant Variables Excluded												
3	0.986	0.989	0.991	0.992	0.231	0.457	0.911	0.516	0.996	0.998	0.996	0.998
5	0.99	0.991	0.993	0.994	0.189	0.462	0.83	0.479	0.999	0.999	0.999	0.999
8	0.992	0.992	0.995	0.994	0.184	0.462	0.711	0.467	0.996	0.997	0.997	0.997
10	0.993	0.988	0.995	0.99	0.162	0.473	0.666	0.481	0.994	0.995	0.995	0.995
Panel (e): Number of Included Variables												
3	3.646	4.058	3.398	3.72	39.13	55.626	7.15	49.966	2.89	2.866	2.86	2.84
5	5.402	5.806	5.274	5.53	41.388	55.992	12.47	54.312	3.364	3.398	3.354	3.38
8	8.222	8.674	8.104	8.456	41.968	57.22	19.31	56.74	3.866	3.744	3.66	3.55
10	10.178	10.784	10.092	10.618	43.154	56.516	21.91	55.81	3.998	3.7	3.58	3.316
Panel (f): Number of Included Irrelevant Variables												
3	0.654	1.074	0.406	0.736	36.142	52.644	4.168	46.988	0.204	0.216	0.176	0.192
5	0.446	0.858	0.318	0.582	36.48	51.116	7.67	49.448	0.032	0.06	0.028	0.054
8	0.318	0.752	0.2	0.534	34.286	49.524	12.132	49.048	0.166	0.308	0.144	0.282
10	0.286	1.06	0.2	0.898	33.502	47.404	13.348	46.746	0.23	0.486	0.196	0.416

Table 2: Statistics concerning model selection for each sample configuration. Panel (a), the fraction of instances where the accurate model was chosen is depicted. Panel (b) showcases the proportion of repetitions in which all pertinent variables were incorporated. Panel (c) illustrates the fraction of relevant variables that were included in the selected models. Panel (d) exhibits the ratio of irrelevant variables that were excluded from the chosen models. Panel (e) displays the mean count of variables incorporated across all selected models. In Panel (f), the average number of irrelevant variables included in the chosen models is depicted. Here, "p" denotes the total number of candidate variables, while "q" represents the quantity of relevant variables in consideration.

It is noteworthy that the State Space Learning framework consistently surpassed the benchmarks across all examined scenarios (in terms of Correct Sparsity Pattern), particularly demonstrating remarkable advancements in instances characterized by elevated counts of candidate variables and true features. The State Space Learning framework was able to hold and even increase the ration of correct sparsity pattern in more complex scenarios. It is interesting that both benchmarks got better results when the number of true relevant variables was low (3), but got increasingly bad results as this number increases.

Furthermore, our proposed framework consistently exhibited a parsimonious selection of variables, particularly utilizing BIC as information criteria, yielding an average inclusion of less than one irrelevant variable across almost all tested scenarios. Conversely, the framework inclusively captured over 99% of the relevant variables in all tested scenarios.

Moreover, the benchmark models faltered in maintaining performance efficacy as the numbers of candidates and true features escalated. This disparity underscores the superior adaptability and robustness of the State Space Learning framework compared to traditional benchmarks, especially under conditions of increased complexity and dimensionality.

Table 3 presents the results of the average of the Monte Carlo simulation for MSE (average mean squared error), Bias and execution time for each model. The values of MSE and Bias are calculated as:

$$MSE = \frac{1}{500 \cdot p} \sum_{i=1}^{p} \left(\sum_{j=1}^{500} (\hat{\beta}_{i,j} - \beta_{i,j})^2 \right) \tag{48}$$

$$Bias = \frac{1}{500 \cdot p} \sum_{i=1}^{p} \left(\sum_{j=1}^{500} (\hat{\beta}_{i,j} - \beta_{i,j}) \right) \tag{49}$$

M\K	SSL (AIC)		SSL (BIC)		SS+AdaLasso (AIC)		SS+AdaLasso (BIC)		SS+FW (AIC)		SS+FW (BIC)	
	50	100	50	100	50	100	50	100	50	100	50	100
					Panel (a): Average MSE							
3	2.4e-4	2.0e-4	2.4e-4	2.0e-4	0.046	1.3e7	0.013	84.272	0.024	2.7e7	0.024	2.7e7
5	0.001	6.1e-4	0.001	6.1e-4	3.2e2	97.568	0.03	0.028	0.058	0.034	0.058	0.034
8	0.002	0.001	0.002	0.001	0.131	0.049	0.066	0.049	0.111	0.059	0.113	0.06
10	0.003	0.004	0.003	0.004	4.3e4	0.072	0.09	0.071	2.9e5	0.083	0.157	0.085
					Panel (b): Average Bias							
3	-7.2e-4	-4.3e-4	-7.7e-4	-4.5e-4	-2.0e-2	34.635	-2.2e-2	0.029	-2.4e-2	23.456	-2.4e-2	23.456
5	-1.9e-3	-1.1e-3	-2.0e-3	-1.2e-3	0.115	0.013	-4.0e-2	-2.1e-2	-5.4e-2	-2.7e-2	-5.4e-2	-2.7e-2
8	-4.5e-3	-2.1e-3	-4.6e-3	-2.1e-3	-5.9e-2	-3.7e-2	-6.8e-2	-3.7e-2	-9.9e-2	-5.0e-2	-10.0e-2	-5.1e-2
10	-6.3e-3	-6.1e-3	-6.5e-3	-6.1e-3	-1.0e-1	-5.1e-2	-8.5e-2	-5.2e-2	-3.5e0	-6.8e-2	-1.3e-1	-6.9e-2
					Panel (c): Median MSE							
3	2.6e-5	1.2e-5	2.8e-5	1.3e-5	0.039	0.016	0.009	0.015	0.02	0.01	0.02	0.01
5	5.6e-5	2.7e-5	5.7e-5	2.8e-5	0.068	0.027	0.022	0.027	0.06	0.03	0.06	0.03
8	1.4e-4	6.4e-5	1.4e-4	6.8e-5	0.114	0.047	0.051	0.047	0.108	0.06	0.112	0.06
10	2.3e-4	1.2e-4	2.4e-4	1.2e-4	0.134	0.067	0.072	0.067	0.152	0.082	0.161	0.083
					Panel (d): Median MSE							
3	-4.2e-4	-1.8e-4	-4.4e-4	-1.8e-4	-2.1e-2	-1.2e-2	-2.0e-2	-1.1e-2	-2.0e-2	-1.0e-2	-2.0e-2	-1.0e-2
5	-8.3e-4	-4.4e-4	-8.5e-4	-4.7e-4	-3.8e-2	-2.0e-2	-3.8e-2	-2.0e-2	-5.7e-2	-2.8e-2	-5.7e-2	-2.8e-2
8	-2.0e-3	-1.0e-3	-2.0e-3	-1.1e-3	-5.7e-2	-3.6e-2	-6.2e-2	-3.6e-2	-9.9e-2	-5.0e-2	-9.9e-2	-5.1e-2
10	-3.3e-3	-1.9e-3	-3.4e-3	-1.9e-3	-6.5e-2	-5.0e-2	-7.7e-2	-5.0e-2	-1.3e-1	-6.9e-2	-1.3e-1	-7.0e-2
					Panel (e): Average Time							
3	0.013	0.015	0.013	0.014	0.146	0.15	0.145	0.151	4.9e1	1.0e2	4.7e1	9.6e1
5	0.014	0.016	0.012	0.014	0.154	0.153	0.153	0.152	5.6e1	1.1e2	5.4e1	1.1e2
8	0.013	0.017	0.013	0.015	0.157	0.148	0.157	0.149	6.6e1	1.3e2	6.1e1	1.2e2
10	0.013	0.015	0.013	0.016	0.151	0.151	0.151	0.151	6.6e1	1.3e2	5.7e1	1.1e2

Table 3: Statistics concerning model estimation processes. Panel (a), Average MSE estimated. Panel (b) Average Bias estimated. Panel (c), Median MSE estimated. Panel (f) Median Bias estimated. (e) Average run time.

The proposed framework significantly outperformed the benchmarks in terms of average Mean Squared Error (MSE) and bias across all tested configurations. This substantial improvement suggests that the estimated models of the SSL more accurately reflect the true coefficients of the exogenous features.

Notably, a minority of the test samples produced exceptionally poor results in terms of bias and MSE for the benchmarks. However, even when assessing the median values of these metrics, the proposed framework consistently demonstrated superior performance. This finding indicates that the SSL framework exhibits greater robustness to challenging samples and also for the average scenarios.

Moreover, It is imperative to underscore the substantial enhancement in computational efficiency demonstrated by the proposed framework, surpassing the forward benchmark by over 1000 times in all tested scenarios and surpassing the stepwise Adaptive Lasso by over 5 times in all tested scenarios. This marked high computation cost of the forward selection method can be attributed to the inherent computational demands, necessitating the estimation of a considerable number of models contingent upon the volume of candidate variables.

5 Conclusion

This work introduced State Space Learning (SSL), a comprehensive framework designed to address some relevant limitations of traditional state-space models in time series analysis. More specifically, through a novel formulation that incorporates high-dimensional regularized regression, the proposed SSL approach not only improves the extraction of unobserved components but also efficiently manages outlier detection and optimal subset selection, even in non-stationary time series. Additionally, the regularized regression format allows for a polynomial-time estimation process with global optimality guarantees. Therefore, SSL significantly enhances the flexibility and robustness of time series modeling, particularly in high-dimensional settings, providing a better representation of systems through state-space models that effectively tackle modern data challenges.

The numerical experiments, conducted on both synthetic and real-world datasets, demonstrate the predictive power and robustness of the SSL framework. Based on the results obtained using the M4 competition dataset, the superiority of our approach over traditional benchmarks (such as Kalman filtering and ARIMA models) in both forecasting and subset selection activities corroborates the robustness of our contribution.

By extending state-space modeling to a framework that can be applied to any linear state-space model with time-varying coefficients, SSL paves the way for broader applicability in fields beyond time series analysis. The implementation of our methodology in the open-source package "StateSpaceLearning.jl" ensures that researchers and practitioners alike can access and apply this framework to diverse temporal data problems.

Based on the provided evidence, the proposed SSL framework represents a significant step forward in the evolution of time series modeling, bridging the gap between classical state-space models and modern machine learning techniques. Future research could explore further extensions to non-linear dynamics and scaling improvements, continuing to refine the framework and expand its scope of applications.

References

[1] Glmnet.jl: Lasso and elastic-net regularized generalized linear models. URL `https://github.com/JuliaStats/GLMNet.jl`.

[2] Estimation procedures for structural time series models. *Journal of Forecasting*, 9:89–108. ISSN 0277-6693.

[3] Marie Auger-Méthé, Ken Newman, Diana Cole, Fanny Empacher, Rowenna Gryba, Aaron King, Vianey Leos-Barajas, Joanna Flemming, Anders Nielsen, Giovanni Petris, and Len Thomas. A guide to state–space modeling of ecological time series. *Ecological Monographs*, 91, 06 2021. doi: 10.1002/ecm.1470.

[4] Nadim Ballout, Lola Etievant, and Vivian Viallon. On the use of cross-validation for the calibration of the adaptive lasso. *Biometrical journal. Biometrische Zeitschrift*, 65:e2200047, 03 2023. doi: 10.1002/bimj.202200047.

[5] Sebastian Blanc and Thomas Setzer. Bias–variance trade-off and shrinkage of weights in forecast combination. *Management Science*, 66, 06 2020. doi: 10.1287/mnsc.2019.3476.

[6] George.E.P. Box and Gwilym M. Jenkins. *Time Series Analysis: Forecasting and Control*. Holden-Day, 1976.

[7] Peter J Brockwell and Richard A Davis. *Time Series: Theory and Methods*. SpringerVerlag, New York, 2 edition, 1991.

[8] Liyuan Cui, Yongmiao Hong, Yingxing Li, and Junhui Wang. A regularized high-dimensional positive definite covariance estimator with high-frequency data. *Management Science*, 0(0), 2023. doi: 10.1287/mnsc.2022.04138.

[9] J Durbin and S. J Koopman. *Time Series Analysis by State Space Models*. 2012.

[10] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Nonlinear Optimization: Theory and Examples*. SIAM, 2001.

[11] Arthur Hoerl and Robert Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12:55–67, 04 1970. doi: 10.1080/00401706.1970.10488634.

[12] Rob J. Hyndman and Yeasmin Khandakar. Automatic time series forecasting: The forecast package for r. *Journal of Statistical Software*, 27(3):1–22, 2008. doi: 10.18637/jss.v027.i03. URL `https://www.jstatsoft.org/index.php/jss/article/view/v027i03`.

[13] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, and Jonathan Taylor. Statistical learning. In *An introduction to statistical learning: With applications in Python*, pages 15–67. Springer, 2023.

[14] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35–45, 03 1960. ISSN 0021-9223. doi: 10.1115/1.3662552. URL `https://doi.org/10.1115/1.3662552`.

[15] Seung-Jean Kim, Kwangmoo Koh, Stephen Boyd, and Dimitry Gorinevsky. l1 trend filtering. *Siam Review - SIAM REV*, 51:339–360, 06 2009. doi: 10.1137/070690274.

[16] Evandro Konzen and Flavio A. Ziegelmann. Lasso-type penalties for covariate selection and forecasting in time series. *Journal of Forecasting*, 35(7):592 – 612, 2016. doi: 10.1002/for.2403.

[17] Siem Jan Koopman, André Lucas, and Marcel Scharth. Predicting Time-Varying Parameters with Parameter-Driven and Observation-Driven Models. *The Review of Economics and Statistics*, 98(1):97–110, 03 2016. ISSN 0034-6535. doi: 10.1162/REST_a_00533. URL `https://doi.org/10.1162/REST_a_00533`.

[18] Siem Jan Koopman, Rutger Lit, and Thuy Nguyen. Modified efficient importance sampling for partially non-gaussian state space models: Modified efficient importance sampling. *Statistica Neerlandica*, 73, 03 2018. doi: 10.1111/stan.12128.

[19] Jiahan Li and Weiye Chen. Forecasting macroeconomic time series: Lasso-based approaches and their forecast combinations with dynamic factor models. *International Journal of Forecasting*, 30(4):996 – 1015, 2014. doi: 10.1016/j.ijforecast.2014.03.016.

[20] Mengheng Li and Siem Jan Koopman. Unobserved components with stochastic volatility: Simulation-based estimation and signal extraction. *Journal of Applied Econometrics*, 36, 06 2021. doi: 10.1002/jae.2831.

[21] Spyros Makridakis, S. Wheelwright, and Rob Hyndman. *Forecasting: Methods and Applications*, volume 35. 01 1984. doi: 10.2307/2581936.

[22] Marcelo C. Medeiros and Eduardo F. Mendes. l1-regularization of high-dimensional time-series models with non-gaussian and heteroskedastic errors. *Journal of Econometrics*, 191(1):255–271, 2016. ISSN 0304-4076. doi: https://doi.org/10.1016/j.jeconom.2015.10.011. URL `https://www.sciencedirect.com/science/article/pii/S0304407615002638`.

[23] Sascha Mergner. *Applications of State Space Models in finance: an empirical analysis of the time-varying relationship between macroeconomics, fundamentals and Pan-European industry portfolios.* 01 2009. ISBN 978-3-941875-22-7. doi: 10.17875/gup2009-216.

[24] Caterina Schiavoni, Siem Koopman, Franz Palm, Stephan Smeekes, and Jan Brakel. Time-varying state correlations in state space models and their estimation via indirect inference. *SSRN Electronic Journal*, 01 2021. doi: 10.2139/ssrn.3792448.

[25] Skipper Seabold and Josef Perktold. statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*, 2010.

[26] Jianzhong Sun, Hongfu Zuo, Wenbin Wang, and Michael Pecht. Application of a state space modeling technique to system prognostics based on a health index for condition-based maintenance. *Mechanical Systems and Signal Processing - MECH SYST SIGNAL PROCESS*, 28, 04 2012. doi: 10.1016/j.ymssp.2011.09.029.

[27] M4 Team. M4 competitor's guide: Prizes and rules. Online, 2018. URL `http://www.m4.unic.ac.cy/wp-content/uploads/2018/03/M4-CompetitorsGuide.pdf`.

[28] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996.

[29] Hui Zou. The adaptive lasso ad its oracle properties. *Journal of the American Statistical Association*, 101: 1418–1429, 02 2006. doi: 10.1198/016214506000000735.

[30] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net (vol b 67, pg 301, 2005). *Journal of the Royal Statistical Society Series B*, 67:768–768, 02 2005. doi: 10.1111/j.1467-9868.2005.00527.x.