Contents lists available at ScienceDirect

Physica D

journal homepage: www.elsevier.com/locate/physd

Sparse methods for automatic relevance determination[☆]

Samuel H. Rudy [*], Themistoklis P. Sapsis

Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, United States of America

ARTICLE INFO

Article history:
Received 15 May 2020
Received in revised form 30 December 2020
Accepted 31 December 2020
Available online 11 January 2021
Communicated by T. Sauer

Keywords:
Sparse regression
Automatic relevance determination
System identification

ABSTRACT

This work considers methods for imposing sparsity in Bayesian regression with applications in nonlinear system identification. We first review automatic relevance determination (ARD) and analytically demonstrate the need to additional regularization or thresholding to achieve sparse models. We then discuss two classes of methods, regularization based and thresholding based, which build on ARD to learn parsimonious solutions to linear problems. In the case of orthogonal features, we analytically demonstrate favorable performance with regard to learning a small set of active terms in a linear system with a sparse solution. Several example problems are presented to compare the set of proposed methods in terms of advantages and limitations to ARD in bases with hundreds of elements. The aim of this paper is to analyze and understand the assumptions that lead to several algorithms and to provide theoretical and empirical results so that the reader may gain insight and make more informed choices regarding sparse Bayesian regression.

1. Introduction

In many modeling and engineering problems it is critical to build statistical models from data which include estimates of the model uncertainty. This is often achieved through non-parametric Bayesian regression in the form of Gaussian processes and similar methods [1]. While these methods offer tremendous flexibility and have seen success in a wide variety of applications they have two significant shortcomings; they are not interpretable and they often fail in high dimensional settings. The simplest case of parametric Bayesian regression is Bayesian ridge regression, where one learns a distribution for model parameters by assuming identical independently distributed (iid) Gaussian priors on model weights. However, Bayesian ridge requires the researcher to provide a single length scale for the prior that stays fixed across dimensions. It is therefore not invariant to changes in units. Furthermore Bayesian ridge regression does not yield sparse models. In a high dimensional setting this may hinder the interpretability of the learned model. Automatic Relevance Determination (ARD) [2–4] addresses both of these problems. ARD learns length scales associated with each free variable in a regression problem. In the context of linear regression, ARD is often referred to as Sparse Bayesian Learning (SBL) [4–6] due to its tendency to learn sparse solutions to linear problems. ARD has been applied to problems in compressed sensing [7], sparse regression [8–12], matrix factorization, [13], classification of gene expression data [14], earthquake detection [15], Bayesian neural networks [2], as well as other fields.

More recently, some works have used ARD for interpretable nonlinear system identification. In this setting a linear regression problem is formulated to learn the equations of motion for a dynamical system from a large collection of candidate functions, called a library. Traditionally, frequentist methods have been applied to select a small set of active terms from the candidate functions. These include symbolic regression [16–18], sequential thresholding [19–21], information theoretic methods [22,23], relaxation methods, [24,25], and constrained sparse optimization [26]. Bayesian methods for nonlinear system identification [27,28] including ARD have been applied to the nonlinear system identification problem for improved robustness in the case of low data [11] and for uncertainty quantification [9,10,12]. The critical challenge for any library method for nonlinear system identification is learning the correct set of active terms.

Motivated by problems such as nonlinear system identification, where accuracy in determining the sparsity pattern on a predictor is paramount, we focus on the ability of ARD to accurately learn a small subset of active terms in a linear system. This is in contrast to convergence in the sense of any norm. Indeed, it was previously shown [11] that ARD converges to the true predictor as noise present in the training data shrinks to zero. However, we show analytically that ARD fails to obtain the true sparsity pattern in the case of an orthonormal design matrix, leading to extraneous terms for arbitrarily small magnitudes of noise. This result motivates further considerations for imposing sparsity on the learned model.

[☆] Python code: https://github.com/snagcliffs/SparseARD.

[*] Corresponding author.
E-mail address: shrudy@mit.edu (S.H. Rudy).

https://doi.org/10.1016/j.physd.2021.132843

This paper explores several intuitive methods for imposing sparsity in the ARD framework. We discuss the assumptions that lead to each technique, any approximations we use to make them tractable, and in some cases provide theoretical results regarding their accuracy with respect to selection of active terms. We stress that while sparse regression is a mature field with many approaches designed to approximate the ℓ^0-penalized least squares problem [22,24,29–31], most of these techniques do not consider uncertainty. We therefore only compare results of the discussed techniques to ARD.

The paper is organized as follows. In Section 2 we provide a brief discussion on the automatic relevance determination method for Bayesian linear regression. Section 3 introduces two regularization-based methods for imposing sparsity of learned predictors from the ARD algorithm. Section 4 introduces various thresholding-based approaches. In each case we provide analytical results for the expected false positive and negative rates with respect to coefficients being set to zero. Section 5 includes a more detailed comparison between methods. Section 6 includes results of each of the methods applied to a variety of problems including a sparse linear system, function fitting, and nonlinear system identification. Discussion and comments towards future work are included in Section 7.

2. Setup

We start with the likelihood model,

$$\begin{aligned} y &= \boldsymbol{\theta}(\mathbf{x})\boldsymbol{\xi} + \nu \\ \nu &\sim \mathcal{N}(0, \sigma^2), \end{aligned} \tag{1}$$

where $\boldsymbol{\theta} : \mathbb{R}^n \to \mathbb{R}^d$ forms a nonlinear basis, y is scalar, $\mathbf{x} \in \mathbb{R}^n$, $\boldsymbol{\xi} \in \mathbb{R}^d$ and ν is normally distributed error with variance σ^2. We assume a prior distribution on weights ξ with variance given by hyper-parameter $\boldsymbol{\gamma}$.

$$\xi_i \sim \mathcal{N}(0, \gamma_i) \tag{2}$$

Automatic relevance determination seeks to learn the value of parameter $\boldsymbol{\gamma}$ that maximizes evidence. This approach is known as evidence maximization, empirical Bayes, or type-II maximum likelihood [32,33]. Given a dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$ we marginalize over $\boldsymbol{\xi}$ to obtain the posterior likelihood of $\boldsymbol{\gamma}$. This gives,

$$\begin{aligned} p(\mathcal{D}|\boldsymbol{\gamma}) &= \int p(\mathcal{D}|\boldsymbol{\xi})p(\boldsymbol{\xi};\boldsymbol{\gamma})\,d\boldsymbol{\xi} \\ &\propto |\boldsymbol{\Sigma}_y|^{-\frac{1}{2}} exp\left(-\frac{1}{2}\mathbf{y}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{y}\right), \end{aligned} \tag{3}$$

where $\boldsymbol{\Sigma}_y = \sigma^2\mathbf{I}_m + \boldsymbol{\theta}(\mathbf{X})\boldsymbol{\Gamma}\boldsymbol{\theta}(\mathbf{X})^T$, $\boldsymbol{\Gamma} = diag(\boldsymbol{\gamma})$, $\mathbf{y}$ is a column vector of all observed outputs and $\boldsymbol{\theta}(\mathbf{X}) \in \mathbb{R}^{d\times m}$ is a matrix whose rows are the nonlinear features of each observed $\mathbf{x}$. We estimate $\boldsymbol{\gamma}$ by maximizing Eq. (3) and the subsequent distribution for $\boldsymbol{\xi}$, given by $\boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{\mu}_\xi, \boldsymbol{\Sigma}_\xi)$. Letting $\boldsymbol{\Theta} = \boldsymbol{\theta}(\mathbf{X})$ this is,

$$\begin{aligned} \boldsymbol{\mu}_\xi &= \sigma^{-2}\boldsymbol{\Sigma}_\xi\boldsymbol{\Theta}^T\mathbf{y} \\ \boldsymbol{\Sigma}_\xi &= \left(\sigma^{-2}\boldsymbol{\Theta}^T\boldsymbol{\Theta} + \boldsymbol{\Gamma}^{-1}\right)^{-1} \end{aligned} \tag{4}$$

In practice $\boldsymbol{\gamma}$ is found by minimizing the negative log of Eq. (3) given by,

$$L(\boldsymbol{\gamma}) = -\log p(\mathcal{D};\boldsymbol{\gamma}) \propto log|\boldsymbol{\Sigma}_y| + \mathbf{y}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{y}. \tag{5}$$

Following [11] (see Appendix A) the second term in (5) is equivalent to,

$$\mathbf{y}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{y} = \min_{\boldsymbol{\xi}} \frac{1}{\sigma^2}\|\mathbf{y} - \boldsymbol{\Theta}\boldsymbol{\xi}\|_2^2 + \boldsymbol{\xi}^T\boldsymbol{\Gamma}^{-1}\boldsymbol{\xi}, \tag{6}$$

which gives the following representation of the loss function (5),

$$L(\boldsymbol{\gamma}) = \min_{\boldsymbol{\xi}} \left(log|\boldsymbol{\Sigma}_y| + \frac{1}{\sigma^2}\|\mathbf{y} - \boldsymbol{\theta}\boldsymbol{\xi}\|_2^2 + \boldsymbol{\xi}^T\boldsymbol{\Gamma}^{-1}\boldsymbol{\xi} \right). \tag{7}$$

To minimize (7) we solve a sequence of ℓ^1-penalized least squares problems developed in [4]. This is shown in Alg. 1.

Algorithm 1 ARD($\boldsymbol{\Theta}, \mathbf{y}, \sigma^2$)

```
1: Initialize $\boldsymbol{\gamma}$
2: while not converged:
3:     $\mathbf{c}^{(k+1)} = \nabla_{\boldsymbol{\gamma}}\left(\log\left|\boldsymbol{\Sigma}_y^{(k)}\right|\right) = diag\left(\boldsymbol{\Theta}^T\boldsymbol{\Sigma}_y^{(k)^{-1}}\boldsymbol{\Theta}\right)$
       where $\boldsymbol{\Sigma}_y^{(k)} = \sigma^2\mathbf{I} + \boldsymbol{\Theta}\boldsymbol{\Gamma}^{(k)}\boldsymbol{\Theta}$
4:     $\boldsymbol{\xi}^{(k+1)} = \arg\min_{\boldsymbol{\xi}} \left\{ \|\mathbf{y} - \boldsymbol{\Theta}\boldsymbol{\xi}\|^2 + \sum_i \eta_i^{(k+1)}|\xi_i| \right\}$
       where $\eta_i^{(k+1)} = 2\sigma^2\sqrt{c_i^{(k+1)}}$
5:     $\gamma_i^{(k+1)} = c_i^{(k+1)^{-1/2}}|\xi_i^{(k+1)}|$
6:     Optional: relearn $\sigma^2$
7: return $\boldsymbol{\gamma}^{(k+1)}$
```

In cases where the variance of the linear model error is unknown, as will be the case for many practical settings, Alg. 1 has the optional step of learning σ^2 using the formula provided by Eq. 46 in [3]. An initial estimate of σ^2 may be set using a specified fraction of $var(\mathbf{y})$ as in [5] or some other plausible value. In this work, we assume σ^2 is known for all examples shown in Sections 3–5, where we are primarily concerned with convergence properties of Alg. 1 and its variations. In Section 6 we treat σ^2 as unknown and learn both σ^2 and $\boldsymbol{\gamma}$.

Some works have used Gamma distribution priors on scale parameters $\boldsymbol{\gamma}$ and precision σ^{-2} [3]. More recent works [4,9,11] have not used this formulation, so much of the following work does not use hierarchical priors. We note however, that the case of a Gamma distribution prior on $\boldsymbol{\gamma}$ with shape parameter $k = 1$ is in fact a Laplace prior. This case has been studied as a Bayesian compressed sensing method [7] and is a special case of the formulation considered in Section 3.

The minimization problem in step 4 of Alg. 1 may be rewritten, after rescaling $\boldsymbol{\Theta}$ and $\boldsymbol{\xi}$, to obtain the commonly used Lagrangian form of the least absolute shrinkage and selection operator (Lasso) [24]. Letting,

$$\boldsymbol{\zeta}^{(k+1)} = \arg\min_{\boldsymbol{\zeta}} \left\|\mathbf{y} - \boldsymbol{\Theta}\, diag\left(\boldsymbol{\eta}^{(k+1)}\right)^{-1}\boldsymbol{\zeta}\right\|_2^2 + \|\boldsymbol{\zeta}\|_1 \tag{8}$$

we get,

$$\boldsymbol{\xi}^{(k+1)} = diag\left(\boldsymbol{\eta}^{(k+1)}\right)^{-1}\boldsymbol{\zeta}. \tag{9}$$

Typical solvers for Eq. (8) include coordinate descent [34], proximal gradient methods [35], alternating direction method of multipliers [36], and least angle regression (LARS) [37]. Several example datasets considered in this manuscript resulted in ill-conditioned $\boldsymbol{\Theta}$ and therefore slow convergence of algorithms for solving the Lasso subroutine. We found empirically that all methods performed equally well on orthogonal $\boldsymbol{\Theta}$ but for ill-conditioned cases LARS outperformed other optimization routines.

Some works [3,5] use expectation–maximization or coordinate descent algorithms to solve Eq. (5). These methods often exhibit rapid convergence but lack theoretical guarantees of convergence to a critical point of Eq. (5). Furthermore, it has been observed that they often exhibit poor performance for high dimensional problems [4]. Alg. 1 is guaranteed to converge to a local minima or saddle point of (5) [4] and is also easily adjusted to account for certain classes of priors on $\boldsymbol{\gamma}$, as will be discussed in Section 3.2. It is plausible that each of the methods discussed in this work could be implemented with a coordinate descent method for solving Eq. (5) rather than Alg. 1, but we leave a more careful comparison to future work.

As we have noted, it is often the case that solutions to Eq. (5) exhibit some degree of sparsity. However, such solutions are only sparse in comparison to those derived by methods such as Bayesian ridge, where all coefficients are nonzero. For problems where we seek to find a very small set of nonzero terms, Alg. 1 must be adjusted to push extraneous terms to zero. In the following two sections we will discuss five methods for doing so.

3. Regularization based methods

We begin with a discussion of two methods for regularizing ARD to obtain more sparse predictors: inflating the variance passed into Alg. 1 and including a prior for the distribution of $\boldsymbol{\gamma}$. In each case the sparse predictor is found as the fixed point of an iterative algorithm. In subsequent sections we will discuss thresholding based methods that alternate between iterative optimization and thresholding operations. In certain cases we refer to the set valued subgradient of a continuous piecewise differentiable function. In cases where the subgradient is a singleton we treat it as a real number.

3.1. Variance inflation

The error variance σ^2 of likelihood model (1) may be intuitively thought of as a level of mistrust for the data $\mathcal{D}$. Extremely large values of σ^2 will push estimates of $\boldsymbol{\xi}$ to be dominated by priors. It is shown in [4] that the ARD prior given by (2) is equivalent to a concave regularization term. We therefore expect large σ^2 to encourage more sparse models. The regularization may be strengthened by passing in an artificially large value of σ^2 to the iterative algorithm for solving Eq. (7) or, if also learning σ^2, by applying an inflated value at each step in the algorithm. We will call this process ARD with variance inflation (ARDvi), shown in Algorithm 2. Note that this differs from Alg. 1 only slightly by treating the variance used in the standard ARD algorithm as a tuning parameter, with a higher variance indicating less trust in the data and a greater regularization.

Algorithm 2 ARDvi($\boldsymbol{\Theta}, \mathbf{y}, \sigma^2, \alpha =$ inflation factor)

```
1: Initialize $\boldsymbol{\gamma}$
2: while not converged:
3:     $\mathbf{c}^{(k+1)} = \nabla_{\boldsymbol{\gamma}}\left(\log\left|\boldsymbol{\Sigma}_y^{(k)}\right|\right) = diag\left(\boldsymbol{\Theta}^T\boldsymbol{\Sigma}_y^{(k)^{-1}}\boldsymbol{\Theta}\right)$
       where $\boldsymbol{\Sigma}_y^{(k)} = \alpha\sigma^2\mathbf{I} + \boldsymbol{\Theta}\boldsymbol{\Gamma}^{(k)}\boldsymbol{\Theta}$
4:     $\boldsymbol{\xi}^{(k+1)} = \underset{\boldsymbol{\xi}}{argmin}\left\{\|\mathbf{y} - \boldsymbol{\Theta}\boldsymbol{\xi}\|^2 + \sum_i \eta_i^{(k+1)}|\xi_i|\right\}$
       where $\eta_i^{(k+1)} = 2\alpha\sigma^2\sqrt{c_i^{(k+1)}}$
5:     $\gamma_i^{(k+1)} = c_i^{(k+1)^{-1/2}}|\xi_i^{(k+1)}|$
6:     Optional: relearn $\sigma^2$
7: return $\boldsymbol{\gamma}^{(k+1)}$
```

3.1.1. Sparsity properties of ARDvi for orthogonal features

To better understand the effect of variance inflation we consider Alg. 2 in the case where columns of $\boldsymbol{\Theta}$ are orthogonal. Note that this implies $m \geq n$. Let $\sqrt{\boldsymbol{\rho}}$ be the vector of column norms of $\boldsymbol{\Theta}$ so that $\boldsymbol{\Theta}^T\boldsymbol{\Theta} = diag(\boldsymbol{\rho})$. Define $\overline{\boldsymbol{\Theta}}$ to be the extension of $\boldsymbol{\Theta}$ to an orthogonal basis of $\mathbb{R}^m$ that $\overline{\boldsymbol{\Theta}}^T\overline{\boldsymbol{\Theta}} = \mathbf{R} = diag(\overline{\boldsymbol{\rho}})$ with the first n entries of $\overline{\boldsymbol{\rho}}$ given by $\boldsymbol{\rho}$. Now let $\boldsymbol{\gamma}^*$ be a fixed point of algorithm 2, $\overline{\boldsymbol{\Gamma}}^* = diag(\boldsymbol{\gamma}^*, \mathbf{0}_{m-n}) \in \mathbb{R}^{m\times m}$, and $\mathbf{c}^*$, $\boldsymbol{\xi}^*$ be defined by steps 3 and 4. The expression in step 3 is given by,

$$\begin{aligned}
c_i^* &= \boldsymbol{\Theta}_i^T\left(\alpha\sigma^2\mathbf{I} + \boldsymbol{\Theta}\boldsymbol{\Gamma}^*\boldsymbol{\Theta}^T\right)^{-1}\boldsymbol{\Theta}_i \\
&= \boldsymbol{\Theta}_i^T\left(\alpha\sigma^2\overline{\boldsymbol{\Theta}}\mathbf{R}^{-1}\overline{\boldsymbol{\Theta}}^T + \overline{\boldsymbol{\Theta}}\overline{\boldsymbol{\Gamma}}^*\overline{\boldsymbol{\Theta}}^T\right)^{-1}\boldsymbol{\Theta}_i \\
&= \boldsymbol{\Theta}_i^T\left(\overline{\boldsymbol{\Theta}}\left(\alpha\sigma^2\mathbf{R}^{-1} + \overline{\boldsymbol{\Gamma}}^*\right)\overline{\boldsymbol{\Theta}}^T\right)^{-1}\boldsymbol{\Theta}_i \\
&= \boldsymbol{\Theta}_i^T\overline{\boldsymbol{\Theta}}\mathbf{R}^{-1}\left(\alpha\sigma^2\mathbf{R}^{-1} + \overline{\boldsymbol{\Gamma}}^*\right)^{-1}\mathbf{R}^{-1}\overline{\boldsymbol{\Theta}}^T\boldsymbol{\Theta}_i \\
&= \mathbf{e}_i^T\left(\alpha\sigma^2\mathbf{R}^{-1} + \overline{\boldsymbol{\Gamma}}^*\right)^{-1}\mathbf{e}_i \\
&= \frac{1}{\alpha\sigma^2\rho_i^{-1} + \boldsymbol{\gamma}_i^*} \\
&= \frac{1}{\alpha\sigma^2\rho_i^{-1} + \frac{|\xi_i^*|}{\sqrt{c_i^*}}} \\
\sqrt{c_i^*} &= \frac{-|\xi_i^*| + \sqrt{\xi_i^{*2} + 4\alpha\sigma^2\rho_i^{-1}}}{2\alpha\sigma^2\rho_i^{-1}}
\end{aligned} \tag{10}$$

The Karush–Kuhn–Tucker (KKT) stationarity condition for the ξ update in step 4 gives,

$$\begin{aligned}
0 &\in \boldsymbol{\Theta}_i^T\left(\boldsymbol{\Theta}\boldsymbol{\xi}^* - \mathbf{y}\right) + \alpha\sigma^2\sqrt{c_i^*}\partial|\xi_i^*| \\
&\in \boldsymbol{\Theta}_i^T\left(\boldsymbol{\Theta}\boldsymbol{\xi}^* - (\boldsymbol{\Theta}\boldsymbol{\xi} + \boldsymbol{\nu})\right) + \alpha\sigma^2\sqrt{c_i^*}\partial|\xi_i^*| \\
&\in \rho_i\xi_i^* - \rho_i\xi_i + \boldsymbol{\Theta}_i^T\boldsymbol{\nu} + \alpha\sigma^2\sqrt{c_i^*}\partial|\xi_i^*|,
\end{aligned} \tag{11}$$

where $\boldsymbol{\xi}$ denotes the true value from Eq. (1). We can find the false positive probability for a term being included in the model by setting $\xi_i = 0$ and finding conditions under which $\xi_i^* \neq 0$. Subbing in the value for $\sqrt{c_i^*}$ from Eq. (10) and dividing by ρ_i gives,

$$\xi_i = 0 \Rightarrow -\rho_i^{-1}\boldsymbol{\Theta}_i^T\boldsymbol{\nu} \in \xi_i^* + \frac{1}{2}\left(\sqrt{\xi_i^{*2} + 4\alpha\sigma^2\rho_i^{-1}} - |\xi_i^*|\right)\partial|\xi_i^*|, \tag{12}$$

where $\partial|\xi_i^*|$ is a set valued function taking value $[-1, 1]$ if $\xi_i^* = 0$ or $\{\text{sgn}(\xi_i^*)\}$ otherwise. If $\xi_i^* = 0$ then,

$$|\rho_i^{-1}\boldsymbol{\Theta}_i^T\boldsymbol{\nu}| \leq \sup_{z\in\partial|\xi_i^*|}\left|z\sqrt{\alpha\sigma^2\rho_i^{-1}}\right| = \sqrt{\alpha\sigma^2\rho_i^{-1}}, \tag{13}$$

while for $\xi^* \neq 0$,

$$\begin{aligned}
|\rho_i^{-1}\boldsymbol{\Theta}_i^T\boldsymbol{\nu}| &= \left|\xi_i^* + \frac{1}{2}\left(\sqrt{\xi_i^{*2} + 4\alpha\sigma^2\rho_i^{-1}} - |\xi_i^*|\right)sgn(\xi_i^*)\right| \\
&= \frac{1}{2}\left(|\xi_i^*| + \sqrt{\xi_i^{*2} + 4\alpha\sigma^2\rho_i^{-1}}\right) \\
&> \sqrt{\alpha\sigma^2\rho_i^{-1}}.
\end{aligned} \tag{14}$$

It follows that $p(\xi_i^* \neq 0|\xi_i = 0) = p(\rho_i^{-1}|\boldsymbol{\Theta}_i^T\boldsymbol{\nu}| > \sqrt{\alpha\sigma^2\rho_i^{-1}})$. Since $\rho_i^{-1}\boldsymbol{\Theta}_i^T\boldsymbol{\nu} \sim \mathcal{N}(0, \rho_i^{-1}\sigma^2)$ we find that the false positive rate is,

$$FP_{VI}(\alpha) = p(\xi_i^* \neq 0|\xi_i = 0) = 1 - \text{erf}\left(\sqrt{\frac{\alpha}{2}}\right), \tag{15}$$

where *erf* is the Gauss error function. Of particular note is that the number of false positives is independent from the variance of the linear model's error term, σ^2. While the mean predictor learned from ARD does converge in any norm to the true solution for $\sigma^2 \to 0$, the expected number of nonzero terms in the learned predictor stays constant. If one desires a sparse predictor, this motivates including a small threshold parameter below which coefficients are ignored, which we will discuss in a subsequent section.

We define a false negative by Algorithm 2 finding some $\gamma_i^* = 0$ (and respectively ξ_i^*) when the true solution $\xi_i \neq 0$ and find the likelihood of such a case in a similar manner. Applying Eq. (10) and the KKT conditions we find,

$$\begin{aligned} 0 &\in -\xi_i + \rho_i^{-1}\boldsymbol{\Theta}_i^T\boldsymbol{\nu} + \sqrt{\alpha\sigma^2\rho_i^{-1}}sgn(\xi_i^*) \\ \rho_i^{-1}\boldsymbol{\Theta}_i^T\boldsymbol{\nu} &\in \xi_i - \sqrt{\alpha\sigma^2\rho_i^{-1}}sgn(\xi_i^*) \\ &\in \left[\xi_i - \sqrt{\alpha\sigma^2\rho_i^{-1}}, \xi_i + \sqrt{\alpha\sigma^2\rho_i^{-1}}\right] \end{aligned} \tag{16}$$

The false negative likelihood is therefore,

$$\begin{aligned} FN_{VI}(\alpha) = p(\xi_i^* = 0|\xi_i \neq 0) = \frac{1}{2}\left(\text{erf}\left(\frac{\xi_i + \sqrt{\alpha\sigma^2\rho_i^{-1}}}{\sigma\sqrt{2\rho_i^{-1}}}\right)\right. \\ \left. -\text{erf}\left(\frac{\xi_i - \sqrt{\alpha\sigma^2\rho_i^{-1}}}{\sigma\sqrt{2\rho_i^{-1}}}\right)\right) \end{aligned} \tag{17}$$

Note that this function vanishes for large $|\xi_i|$, indicating that important terms, as measured by $|\xi_i|$ are far less likely to be missed.

Fig. 1 demonstrates the validity of Eqs. (15) and (17) on a simple test problem. We construct a matrix $\boldsymbol{\Theta} \in \mathbb{R}^{250\times 250}$ with orthogonal columns having random magnitude such that $\rho_i \sim \mathcal{U}([1,3])$ and random $\boldsymbol{\xi}$ with $\|\boldsymbol{\xi}\|_0 = 25$ having non-zero terms distributed according to $\mathcal{N}(0,1)$. The mean number of added and missed nonzero terms across 50 trials are shown and agree very well with the predicted values. As predicted by Eq. (17) the number of missing terms decays to zero as $\sigma \to 0$. However the same is not true for the number of added terms, which only decays as the inflation parameter α is increased. This phenomena is consistent with the predictions of Eq. (17). The failure of ARDvi to converge as $\sigma \to 0$ to the true sparsity pattern for fixed α is certainly troubling, but for sufficiently large α only an arbitrarily small number of terms will be added.

3.2. Regularization via sparsity promoting hierarchical priors

Since the sparsity of $\boldsymbol{\xi}$ is controlled by that of $\boldsymbol{\gamma}$ we can impose sparsity on $\boldsymbol{\xi}$ via regularizing $\boldsymbol{\gamma}$ with a hierarchical prior. Previous approaches to ARD have suggested hierarchical priors on $\boldsymbol{\gamma}$ in the form of Gamma distributions [3]. However, except for certain cases, the general class of Gamma distributions does not impose sparsity. Instead, we consider the use of a sparsity promoting hierarchical prior on the scale parameters $\boldsymbol{\gamma}$. We consider distributions of the form,

$$p(\gamma_i) \propto exp\left(\frac{-g(\gamma_i) - f(\gamma_i)}{2}\right), \tag{18}$$

where f, g are each convex and concave functions in γ_i, respectively. Given data $\mathcal{D}$ we can follow a procedure similar to the one used in Section 2 and find,

$$\begin{aligned} p(\boldsymbol{\gamma}|\mathcal{D}) \propto p(\mathcal{D}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}) &= \int p(\mathcal{D}|\boldsymbol{\xi})p(\boldsymbol{\xi}|\boldsymbol{\gamma})\,d\boldsymbol{\xi}\,p(\boldsymbol{\gamma}) \\ &= (2\pi)^{-m/2}|\boldsymbol{\Sigma}_y|^{-\frac{1}{2}}exp\left(-\frac{1}{2}\mathbf{y}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{y}\right)\prod_{i=1}^{d}e^{\left(\frac{-g(\gamma_i)-f(\gamma_i)}{2}\right)}. \end{aligned} \tag{19}$$

A fully Bayesian approach would estimate $\boldsymbol{\theta}$ through the joint posterior likelihood of pairs $\boldsymbol{\theta}, \boldsymbol{\gamma}$, but this would be computationally expensive. Instead, we approximate $\boldsymbol{\gamma}$ by its maximum-a-posteriori estimate $\boldsymbol{\gamma}_{MAP} = argmax\,p(\boldsymbol{\gamma}|\mathcal{D})$, a process sometimes called type-II MAP [33]. The MAP estimate of $\boldsymbol{\gamma}$ is found by minimizing the negative log of the posterior distribution,

$$\begin{aligned} L_{ARDr}(\boldsymbol{\gamma}) = -\log p(\mathcal{D};\boldsymbol{\gamma}) &\propto log|\boldsymbol{\Sigma}_y| + \mathbf{y}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{y} \\ &\quad + \sum_{i=1}^{d}(f(\gamma_i) + g(\gamma_i)) \\ &= \min_{\boldsymbol{\xi}}\left(log|\boldsymbol{\Sigma}_y| + \frac{1}{\sigma^2}\|\mathbf{y} - \boldsymbol{\Theta}\boldsymbol{\xi}\|_2^2 + \boldsymbol{\xi}^T\boldsymbol{\Gamma}^{-1}\boldsymbol{\xi}\right. \\ &\quad \left. + \sum_{i=1}^{d}(f(\gamma_i) + g(\gamma_i))\right). \end{aligned} \tag{20}$$

As expected, Eq. (20) closely resembles Eq. (7) and may be solved with a similar method. Alg. 3 constructs a sequence $\gamma^{(i)}$ which monotonically increases the likelihood given by Eq. (20). Since $\mathcal{L}_{ARDr}$ is nonconvex, we can only guarantee convergence to a local minimum. We initialize Alg. 3 using the unregularized ARD value of γ.

Algorithm 3: ARDr($\boldsymbol{\Theta}, \mathbf{y}, \sigma^2, f, g$)

```
1: Initialize $\boldsymbol{\gamma}$ using Algorithm 1
2: while not converged:
3:     $\mathbf{c}^{(k+1)} = \nabla_{\boldsymbol{\gamma}}\left(\log\left|\boldsymbol{\Sigma}_y^{(k)}\right| + \sum g\left(\gamma_i^{(k)}\right)\right)$
4:     $\mathbf{c}^{(k+1)} = diag\left(\boldsymbol{\Theta}^T\boldsymbol{\Sigma}_y^{(k)^{-1}}\boldsymbol{\Theta}\right) - \nabla_{\boldsymbol{\gamma}}\sum g\left(\gamma_i^{(k)}\right)$
       where $\boldsymbol{\Sigma}_y^{(k)} = \sigma^2\mathbf{I} + \boldsymbol{\Theta}\boldsymbol{\Gamma}^{(k)}\boldsymbol{\Theta}$
5:     $\boldsymbol{\gamma}^{(k+1)} = \arg\min_{\boldsymbol{\gamma}}\left\{\min_{\boldsymbol{\xi}}\left\{\frac{1}{\sigma^2}\|\mathbf{y} - \boldsymbol{\Theta}\boldsymbol{\xi}\|^2 + \sum_i\left(\frac{\xi_i^2}{\gamma_i} + c_i^{(k+1)}\gamma_i + f(\gamma_i)\right)\right\}\right\}$
6:     Optional: relearn $\sigma^2$
7: return $\boldsymbol{\gamma}^{(k+1)}$
```

Algorithm 3 allows for significant freedom in choosing f and g. The concave component of the prior, g, acts as a sparsity encouraging regularizer on $\boldsymbol{\gamma}$, as is common for concave priors [38]. Examples of concave g include the identity, *tanh*, and approximations of the ℓ^0-norm. We consider functions of the following form;

$$g_{\lambda,\eta}(\gamma_i) = \min\{\lambda\gamma_i, \eta\} \tag{21}$$

where λ is a parameter controlling the strength of the regularizer and η is a width parameter. The convex prior f may be an indicator function restricting γ to a specific domain or left as a constant. In either case implementing the above algorithm is trivial. If f is not a linear or indicator function then step 5 in Alg. 3 will require an internal iterative algorithm.

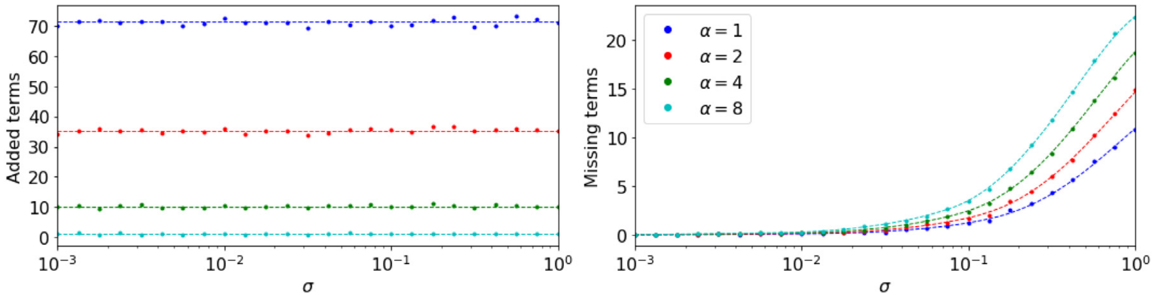

Fig. 1. Number of missing and added terms using ARD with variance inflation. Dots indicate empirical average over 50 trials and dashed lines indicate values predicted by Eqs. (15) and (17). In each case Θ is a 250 × 250 matrix with orthogonal columns and $\|\xi\|_0 = 25$.

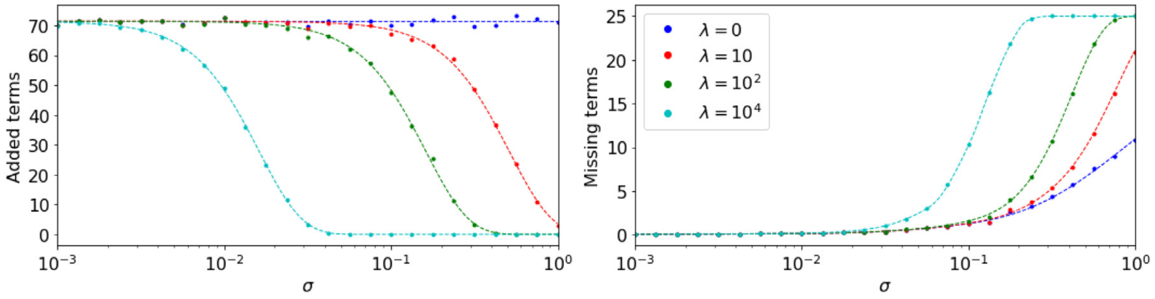

Fig. 2. Number of missing and added terms using ARDr using 250 × 250 orthonormal Θ with $\|\boldsymbol{\xi}\|_0 = 25$. Dots indicate empirical average over 50 trials and dashed lines indicate values predicted by Eqs. (31) and (32).

3.2.1. Sparsity properties of ARDr for orthogonal features

The behavior of Alg. 3 is complicated by the generality of functions f and g. In the simplest case we let f be constant and g be the linear function $g(\gamma_i) = \lambda\gamma_i$. This is the formulation used in [7] and a special case of using a Gamma distribution prior with shape parameter $k = 1$ on γ_i. The update step in line 5 of Alg. 3 gives $\gamma_i^{(k+1)} = |\xi_i^{(k+1)}| c_i^{(k+1)^{-1/2}}$ as in the unregularized case. For a fixed point of Alg. 3 we have,

$$\begin{aligned} c_i^* &= \Theta_i^T \left(\sigma^2 \mathbf{I} + \Theta\Gamma^*\Theta^T\right)^{-1}\Theta_i + \frac{\partial g}{\partial \gamma_i} \\ &= \frac{1}{\sigma^2\rho_i^{-1} + \frac{|\xi_i^*|}{\sqrt{c_i^*}}} + \lambda \qquad (22) \\ 0 &= \rho_i^{-1}\sigma^2 c_i^{*\frac{3}{2}} + |\xi_i^*| c_i^* - \left(\lambda\rho_i^{-1}\sigma^2 + 1\right) c_i^{*\frac{1}{2}} - \lambda|\xi_i^*| \end{aligned}$$

The KKT conditions for the ξ_i update are unchanged from the unregularized case and are given by,

$$0 \in \rho_i\xi_i^* - \rho_i\xi_i + \boldsymbol{\theta}_i^T\boldsymbol{\nu} + \sigma^2\sqrt{c_i^*}sgn(\xi_i^*). \qquad (23)$$

If $\xi_i^* = 0$ then Eq. (22) tells us $\sqrt{c_i^*} = \sqrt{\rho_i\sigma^{-2} + \lambda}$ and therefore,

$$\begin{aligned} \xi_i^* = 0 \Rightarrow \xi_i - \rho_i^{-1}\Theta_i^T\boldsymbol{\nu} \in \\ \times \left[-\sqrt{\rho_i^{-1}\sigma^2 + \lambda\rho_i^{-2}\sigma^4}, \sqrt{\rho_i^{-1}\sigma^2 + \lambda\rho_i^{-2}\sigma^4}\right] \qquad (24) \end{aligned}$$

The converse of Eq. (24) is shown in Appendix B. From this equivalence it follows that the false positive and negative rates for Alg. 3 are given by,

$$FP_r(\lambda) = p(\xi_i^* \neq 0 | \xi_i = 0) = 1 - \text{erf}\left(\sqrt{\frac{1 + \lambda\rho_i^{-1}\sigma^2}{2}}\right) \qquad (25)$$

$$FN_r(\lambda) = \frac{1}{2}\left(\text{erf}\left(\frac{\xi_i + \sqrt{\rho_i^{-1}\sigma^2 + \lambda\rho_i^{-2}\sigma^4}}{\sigma\sqrt{2\rho_i^{-1}}}\right) - \text{erf}\left(\frac{\xi_i - \sqrt{\rho_i^{-1}\sigma^2 + \lambda\rho_i^{-2}\sigma^4}}{\sigma\sqrt{2\rho_i^{-1}}}\right)\right). \qquad (26)$$

These rates are verified empirically by testing 50 trials using 250 × 250 Θ with orthogonal columns and random ρ_i as in Section 3.1. Results are shown in Fig. 2. Note that for fixed $\lambda > 0$ the false negative rate does indeed approach zero as $\sigma \rightarrow 0$, however, the false positive rate increases. This indicates that a linear model with smaller error requires higher regularization to achieve a sparse solution. While counterintuitive, this is explained by Eq. (22). The effect of the linear regularizer appears in Alg. 3 in the computation of c_i with the addition of λ. For small σ, c_i scales as σ^{-2} so for $\sigma^2 \ll \lambda$ the effect of the regularization is negligible. In this case, the algorithm behaves more like unregularized ARD, as can be seen in Fig. 2. We also note that for any λ as $\sigma^2 \rightarrow 0$, $FP_{ARDr}(\lambda) \rightarrow FP_{VI}(1)$ and $FN_{ARDr}(\lambda) \rightarrow FN_{VI}(1)$. For $\lambda\sigma^2$ held fixed as σ varies, the false negative rate still approaches zero and the false positive rate is constant. This latter case is shown in Fig. 3.

Fig. 3 shows a similar convergence pattern to what we observed for ARDvi in Fig. 1. The number of added terms (false positives) remains constant as $\sigma \rightarrow 0$ for any fixed regularization parameter λ. However, we note again that for sufficiently large λ the fixed false positive rate may be made arbitrarily small. In the following section we will construct thresholding methods including one for which the false positive and negative rates converge to zero as $\sigma \rightarrow 0$.

4. Thresholding based methods

As we have shown, automatic relevance determination will not realize the correct non-zero coefficients in a general sparse regression problem, but it will converge in any norm [11]. Therefore, applying an arbitrarily small threshold on $|\xi_i|$ will ensure selection of the correct nonzero coefficients in the limit of low noise. In this section we discuss methods for thresholding the output from Alg. 1 using the mean magnitude of coefficients $|\xi_i|$ or based on the posterior distribution of ξ_i.

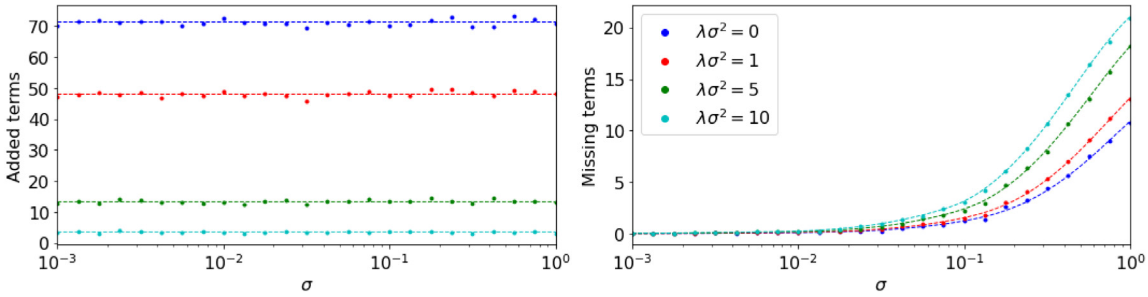

Fig. 3. Number of missing and added terms using ARDr using 250 × 250 orthonormal $\boldsymbol{\theta}$ with $\|\boldsymbol{\xi}\|_0 = 25$ and holding $\lambda\sigma^2$ constant. Dots indicate empirical average over 50 trials and dashed lines indicate values predicted by Eqs. (31) and (32).

4.1. Magnitude based thresholding

Sequential thresholding based on the magnitude of coefficients has been used extensively in regression [19,31] and also in conjunction with automatic relevance determination methods for identifying nonlinear dynamical systems with uncertainty quantification in [9]. Here we consider the method initially proposed in [9], called threshold sparse Bayesian regression (TSBR). To distinguish from other thresholding methods we use the term magnitude sequential threshold sparse Bayesian learning (M-STSBL). We note that the method used in this work (M-STSBL) differs from TSBR in the solver used for the ARD subproblem. This work minimizes Eq. (7) via Alg. 1 whereas [9] uses the Fast Marginal Likelihood Maximization method introduced in [5]. Results, including theoretical estimates of false positive and negative rates, may therefore differ slightly. Magnitude based thresholding assumes that coefficients learned in the ARD algorithm with sufficiently small magnitude, $|\xi_j| < \tau$ are irrelevant and may be treated as zero.

Algorithm 4 M-STSBL($\boldsymbol{\Theta}, \mathbf{y}, \sigma^2, \tau$)

```
$\boldsymbol{\gamma} = \text{ARD}(\boldsymbol{\Theta}, \mathbf{y}, \sigma^2)$
$\boldsymbol{\xi} = \sigma^{-2}\boldsymbol{\Sigma}_\xi \boldsymbol{\Theta}^T \mathbf{y}$
$\mathcal{G} = \{i : |\xi_i| \geq \tau\}$
$\gamma_{\mathcal{G}^c} = 0$
if $\mathcal{G}^c \neq \emptyset$: $\gamma_{\mathcal{G}} = \text{M-STSBL}(\boldsymbol{\Theta}_{\mathcal{G}}, \mathbf{y}, \sigma^2, \tau)$
return $\boldsymbol{\gamma}$
```

The sequential hard-thresholding method for automatic relevance determination is implemented in Alg. 4. Non-zero terms are indexed by $\mathcal{G}$ whose complement $\mathcal{G}^c$ tracks terms removed from the model. At each iteration the algorithm either recursively calls itself with fewer features or terminates if all features are kept non-zero.

4.1.1. Sparsity properties of M-STSBL for orthogonal features

We consider the number of errors using Alg. 4 in a similar context to the analysis of the variance inflation and regularized method. First consider the likelihood of a false non-zero term. Recall from the previous section that the KKT conditions for a fixed point of Alg. 1 imply,

$$\begin{aligned} 0 &\in \xi_i^* - \xi_i + \rho_i^{-1}\boldsymbol{\Theta}_i^T \boldsymbol{\nu} + \rho_i^{-1}\sigma^2\sqrt{c_i^*}\,\partial\|\xi_i^*\|_1 \\ &= \xi_i^* - \xi_i + \rho_i^{-1}\boldsymbol{\Theta}_i^T \boldsymbol{\nu} + \frac{1}{2}\left(\sqrt{\xi_i^{*2} + 4\rho_i^{-1}\sigma^2} - |\xi_i^*|\right) sgn(\xi_i^*), \end{aligned} \tag{27}$$

We can rewrite this as,

$$\phi_{\sigma,\rho}(\xi_i^*) = \xi_i - \rho_i^{-1}\boldsymbol{\Theta}_i^T \boldsymbol{\nu} \sim \mathcal{N}(0, \rho_i^{-1}\sigma^2) \tag{28}$$

where,

$$\begin{aligned} \phi_{\sigma,\rho}(\xi_i^*) &= \frac{\xi_i^*}{2} + \frac{1}{2}\sqrt{\xi_i^2 + 4\rho_i^{-1}\sigma^2}sgn(\xi_i^*) \\ &= \frac{\xi_i^*}{2}\left(1 + \sqrt{1 + 4\rho_i^{-1}\sigma^2\xi_i^{*-2}}\right), \text{ for } \xi_i^* \neq 0, \end{aligned} \tag{29}$$

is invertible on $\mathbb{R} \setminus \{0\}$ and strictly increasing. Therefore,

$$\left|\xi_i - \rho_i^{-1}\boldsymbol{\Theta}_i^T \boldsymbol{\nu}\right| > \phi_{\sigma,\rho}(\tau) \Leftrightarrow |\xi_i^*| > \tau. \tag{30}$$

This gives the likelihood of a false non-zero coefficient as,

$$FP_M(\tau) = p(\xi_i^* \neq 0 | \xi_i = 0) = 1 - \text{erf}\left(\frac{\phi_{\sigma,\rho}(\tau)}{\sigma\sqrt{2\rho_i^{-1}}}\right), \tag{31}$$

and the likelihood for a false zero coefficient as,

$$\begin{aligned} FN_M(\tau) = p(\xi_i^* = 0 | \xi_i \neq 0) = \frac{1}{2}\Bigg(&\text{erf}\left(\frac{\xi_i + \phi_{\sigma,\rho}(\tau)}{\sigma\sqrt{2\rho_i^{-1}}}\right) \\ &-\text{erf}\left(\frac{\xi_i - \phi_{\sigma,\rho}(\tau)}{\sigma\sqrt{2\rho_i^{-1}}}\right)\Bigg). \end{aligned} \tag{32}$$

Eqs. (32) and (31) are verified empirically by testing on 50 trials over a 250 × 250 $\boldsymbol{\Theta}$ using the same experimental design as in Section 3.1. Results are shown in Fig. 4. In contrast to regularization based approaches, we now have the desirable condition where the number of false positive terms each goes to zero as $\sigma \to 0$. However, the number false negatives now only shrinks to a fixed positive number — a consequence of using a hard threshold. This motivates alternative criteria for thresholding. In the next section, we will discuss thresholding based not strictly on magnitude but on the marginal posterior likelihood that a coefficient is zero.

4.2. Likelihood based thresholding

While Alg. 4 was shown to be effective in [9] it is not independent from the units of measurement used for each feature and is not practical in the case where some true coefficients are small. An alternative means of thresholding is to do so based on the marginal likelihood of a coefficient being zero. The marginal posterior distribution of ξ_i is given by,

$$p(\xi_i) \sim \mathcal{N}(\mu_{\xi,i}, \Sigma_{\xi,ii}), \tag{33}$$

where $\mu_{\xi,i}$, $\Sigma_{\xi,ii}$ are given by Eq. (4) and the marginal likelihood that $\xi_i = 0$ is,

$$p(\xi_i = 0) = \mathcal{N}(0\,|\,\mu_{\xi,i}, \Sigma_{\xi,ii}) = \frac{1}{\sqrt{2\pi\,\Sigma_{\xi,ii}}} e^{-\frac{1}{2}\mu_{\xi,i}^2 \Sigma_{\xi,ii}^{-1}}. \tag{34}$$

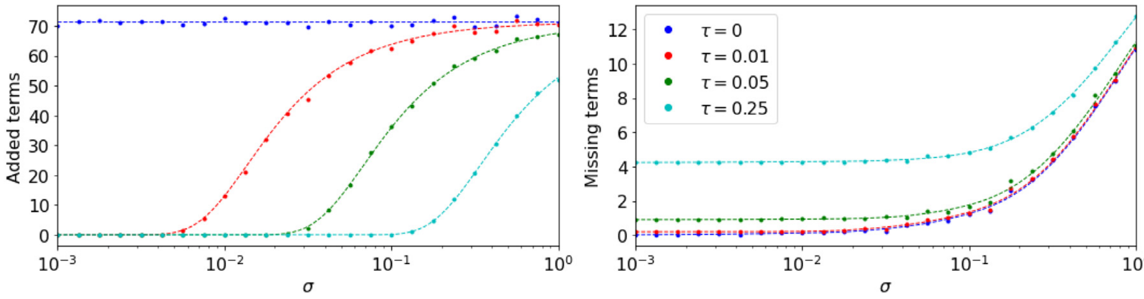

Fig. 4. Number of missing and added terms using M-STSBL using 250 × 250 orthonormal $\boldsymbol{\Theta}$ with $\|\boldsymbol{\xi}\|_0 = 25$. Dots indicate empirical average over 50 trials and dashed lines indicate values predicted by Eqs. (31) and (32).

We can construct a sequential thresholding algorithm shown by Alg. 5 by removing terms whose marginal likelihood evaluated at zero is sufficiently large. The remaining subset of features is then passed recursively to the same procedure until convergence, marked by no change in the number of features. This process is described by Alg. 5 where parameter τ is the threshold marginal likelihood at zero above which features are removed.

Algorithm 5 L-STSBL($\boldsymbol{\Theta}, \mathbf{y}, \sigma^2, \tau$)

```
$\boldsymbol{\gamma} = \text{ARD}(\boldsymbol{\Theta}, \mathbf{y}, \sigma^2)$
$\boldsymbol{\xi} = \sigma^{-2}\boldsymbol{\Sigma}_\xi \boldsymbol{\Theta}^T \mathbf{y}$
$\mathcal{G} = \left\{ i : (2\pi\, \Sigma_{\xi,ii})^{-1/2} exp(-\frac{1}{2}\xi_i^2 \Sigma_{\xi,ii}^{-1}) \leq \tau \right\}$
$\boldsymbol{\gamma}_{\mathcal{G}^c} = 0$
if $\mathcal{G}^c \neq \emptyset$: $\gamma_{\mathcal{G}} = \text{L-STSBL}(\boldsymbol{\Theta}_{\mathcal{G}}, \mathbf{y}, \sigma^2, \tau)$
return $\boldsymbol{\gamma}$
```

4.2.1. *Sparsity properties of L-STSBL for orthogonal features*

We again consider the case where $\boldsymbol{\Theta}^T\boldsymbol{\Theta} = diag(\boldsymbol{\rho})$. Let,

$$h_L(\xi_i, \Sigma_{\xi,ii}) = (2\pi\, \Sigma_{\xi,ii})^{-1/2} exp\left(-\frac{1}{2}\xi_i^2 \Sigma_{\xi,ii}^{-1}\right), \tag{35}$$

so that the thresholding criterion is $h_L(\xi_i, \Sigma_{\xi,ii}) > \tau$. The orthogonality of the columns of $\boldsymbol{\Theta}$ allows us to express the marginal posterior variance $\Sigma_{\xi,ii}$ as a function of $|\xi_i|$. Letting $\boldsymbol{\Sigma}_\xi^*$ be the covariance from a fixed point of Alg. 1 we have,

$$\begin{aligned}
\Sigma_{\xi,ii}^* &= \left(\sigma^{-2}\boldsymbol{\Theta}^T\boldsymbol{\Theta} + \boldsymbol{\Gamma}^{*-1}\right)_{ii}^{-1} \\
&= \left(\sigma^{-2}\,\text{diag}(\boldsymbol{\rho}) + \boldsymbol{\Gamma}^{*-1}\right)_{ii}^{-1} \\
&= \left(\frac{\rho_i}{\sigma^2} + \gamma_i^{*-1}\right)^{-1} \\
\Sigma_{\xi\, ii}^{*-1} &= \frac{\rho_i}{\sigma^2} + \frac{\sqrt{c_i^*}}{|\xi_i^*|} \\
&= \frac{\rho_i}{\sigma^2} + \frac{\sqrt{1 + 4\rho_i^{-1}\sigma^2|\xi_i^*|^{-2}} - 1}{2\rho_i^{-1}\sigma^2} \\
&= \frac{\sqrt{1 + 4\rho_i^{-1}\sigma^2|\xi_i^*|^{-2}} + 1}{2\rho_i^{-1}\sigma^2}.
\end{aligned} \tag{36}$$

This allows us to express,

$$\begin{aligned}
h_L(\xi_i^*, \Sigma_{\xi,ii}^*) &= \tilde{h}_{L,\rho,\sigma}(|\xi_i^*|) \\
&= (2\pi)^{-1/2}\sqrt{\Sigma_{\xi,ii}^{*\;-1}(|\xi_i|)}\, exp\left(-\frac{1}{2}|\xi_i^*|^2 \Sigma_{\xi,ii}^{*\;-1}(|\xi_i|)\right),
\end{aligned} \tag{37}$$

where $\Sigma_{\xi,ii}^{*\;-1}(|\xi_i^*|)$ and $|\xi_i^*|^2\Sigma_{\xi,ii}^{*\;-1}(|\xi_i^*|)$ are strictly decreasing and increasing functions of $|\xi_i^*|$, respectively. It follows that $\tilde{h}_{L,\rho,\sigma}$ is strictly decreasing and therefore invertible with $\tilde{h}_{L,\rho,\sigma}^{-1}$ easily computed by bisection. For $\tau > 0$ there is some $\tilde{h}_{L,\rho,\sigma}^{-1}(\tau)$ such that,

$$|\xi_i^*| > \tilde{h}_{L,\rho,\sigma}^{-1}(\tau) \Leftrightarrow \tilde{h}_{L,\rho,\sigma}(\xi_i^*) \leq \tau, \tag{38}$$

and recalling Eq. (30),

$$|\xi_i - \rho_i^{-1}\boldsymbol{\Theta}_i^T\boldsymbol{\nu}| > \phi_{\sigma,\rho}\left(\tilde{h}_{L,\rho,\sigma}^{-1}(\tau)\right) \Leftrightarrow \tilde{h}_{L,\rho,\sigma}(\xi_i^*) \leq \tau. \tag{39}$$

This gives,

$$FP_L(\tau) = p(\xi_i^* \neq 0|\xi_i = 0) = 1 - \text{erf}\left(\frac{\phi_{\sigma,\rho}\left(\tilde{h}_{L,\rho,\sigma}^{-1}(\tau)\right)}{\sigma\sqrt{2\rho_i^{-1}}}\right), \tag{40}$$

and,

$$\begin{aligned}
FN_L(\tau) = \frac{1}{2}\Bigg(&\text{erf}\left(\frac{\xi_i + \phi_{\sigma,\rho}\left(\tilde{h}_{L,\rho,\sigma}^{-1}(\tau)\right)}{\sigma\sqrt{2\rho_i^{-1}}}\right) \\
&-\text{erf}\left(\frac{\xi_i - \phi_{\sigma,\rho}\left(\tilde{h}_{L,\rho,\sigma}^{-1}(\tau)\right)}{\sigma\sqrt{2\rho_i^{-1}}}\right)\Bigg).
\end{aligned} \tag{41}$$

Eqs. (41) and (40) are verified empirically using the same experimental setup as in previous sections. Results are shown in Fig. 5. Similar to M-STSBL, solutions of L-STSBL converge towards the correct sparsity pattern as $\sigma \to 0$. However, Fig. 5 indicates highly favorable results in the number of missing terms. Eq. (36) indicates that for $\sigma \ll 1$ the marginal variance $\Sigma_{\xi,ii} \sim \mathcal{O}(\sigma^2)$. As a consequence, the exponential in Eq. (35) becomes very small and the algorithms are much more conservative about pruning terms.

4.3. *Thresholding via sparse prior on $\boldsymbol{\xi}$*

Algorithm 5 performs thresholding based on the marginal likelihood of a given coefficient being zero without consideration for the likelihood of the coefficient prior to applying a threshold. We now propose a thresholding method which includes the latter. We consider a prior on $\boldsymbol{\xi}$ which varies from Eq. (2) only where $\|\boldsymbol{\xi}\|_0 < 0$ and use MAP estimates of $\boldsymbol{\xi}$ to prune terms. Consider the same model described in Section 2 but with the following prior on ξ_i,

$$p(\xi_i) = \mathcal{N}(\xi_i|0, \gamma_i)\, e^{\tau\delta_{\xi_i,0}}. \tag{42}$$

Note that this is equivalent to (2) almost everywhere so the integral in (3) is not affected. The posterior for ξ under assumption (42) is then,

$$p(\boldsymbol{\xi}|\mathcal{D}, \tau) \propto \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}_\xi|} e^{-\frac{1}{2}(\boldsymbol{\xi}-\boldsymbol{\mu}_\xi)^T\boldsymbol{\Sigma}_\xi^{-1}(\boldsymbol{\xi}-\boldsymbol{\mu}_\xi) - \tau\|\boldsymbol{\xi}\|_0} \tag{43}$$

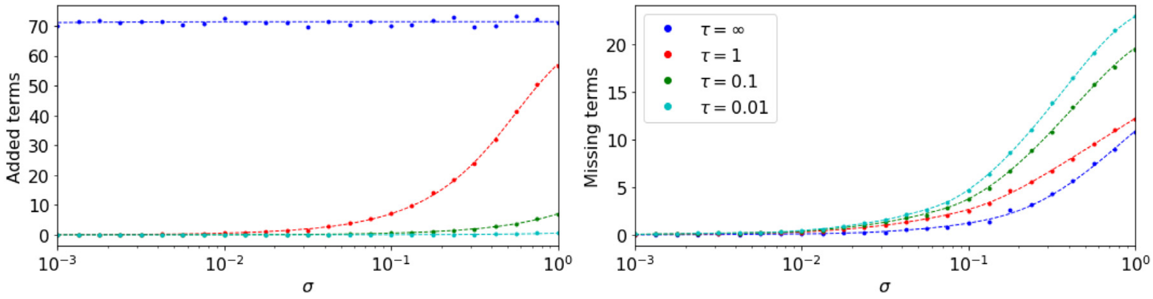

Fig. 5. Number of missing and added terms using L-STSBL using 250 × 250 orthonormal Θ with $\|\boldsymbol{\xi}\|_0 = 25$. Dots indicate points where data was collected as the empirical average over 50 trials.

with $\boldsymbol{\mu}_\xi, \boldsymbol{\Sigma}_\xi$ defined as in Eq. (4). When $\tau = 0$ this reduces to the standard ARD posterior but for $\tau > 0$ the likelihood shrinks exponentially in the number of nonzero terms. Since the two posteriors differ only on a set of measure zero, the inclusion of $exp(\tau\delta_{\xi_i,0})$ in Eq. (42) only affects the solution if we use the MAP estimate of $\boldsymbol{\xi}$ as a means to select active terms. Doing so induces a thresholding operation to find $\boldsymbol{\xi}_{MAP}$.

For a group $S = \{s_1, s_2, \dots, s_q\} \subseteq \{1, \dots, d\}$ where $\boldsymbol{\mu}_{\xi,s_i} \neq 0$ let $\boldsymbol{\xi}_{-S} = \boldsymbol{\mu}_\xi - \sum_q \mu_{\xi,s_i}\mathbf{e}_{s_i}$ where $\mathbf{e}_i$ is the unit vector in the ith coordinate. The likelihood of the thresholded vector $\boldsymbol{\xi}_{-S}$ is given by,

$$
\begin{aligned}
p(\boldsymbol{\xi}_{-S}|\mathcal{D},\tau) &= C\,exp\left(-\frac{1}{2}(\boldsymbol{\xi}_{-S}-\boldsymbol{\mu}_\xi)^T\boldsymbol{\Sigma}_\xi^{-1}(\boldsymbol{\xi}_{-S}-\boldsymbol{\mu}_\xi) - \tau\|\boldsymbol{\xi}_{-S}\|_0\right) \\
&= C\,exp\left(-\frac{1}{2}\left(\sum_q \mu_{\xi,s_i}\mathbf{e}_{s_i}\right)^T \boldsymbol{\Sigma}_\xi^{-1}\left(\sum_q \mu_{\xi,s_i}\mathbf{e}_{s_i}\right)\right. \\
&\qquad \left. -\tau(\|\boldsymbol{\mu}_\xi\|_0 - q)\right) \\
&= p(\boldsymbol{\mu}_\xi|\mathcal{D},\tau)exp\left(-\frac{1}{2}\boldsymbol{\mu}_{\xi,S}^T\boldsymbol{\Sigma}_{\xi,S}^{-1}\boldsymbol{\mu}_{\xi,S} + q\tau\right),
\end{aligned}
\tag{44}
$$

where $\boldsymbol{\Sigma}_{\xi,S}^{-1}$ is the square sub-matrix of $\boldsymbol{\Sigma}_\xi^{-1}$ formed by the rows and columns indexed by S. Then,

$$
p(\boldsymbol{\xi}_{-S}|\mathcal{D},\tau) > p(\boldsymbol{\mu}_\xi|\mathcal{D},\tau) \text{ if } \frac{1}{2}\boldsymbol{\mu}_{\xi,S}^T\boldsymbol{\Sigma}_{\xi,S}^{-1}\boldsymbol{\mu}_{\xi,S} < q\tau, \tag{45}
$$

and the MAP estimate of ξ is given by,

$$
\boldsymbol{\xi}_{MAP} = \boldsymbol{\xi}_{-S} \text{ where } S = \underset{S\in\mathcal{P}([d])}{arg\,max}\; p(\boldsymbol{\xi}_{-S}|\mathcal{D},\tau) \tag{46}
$$

Eq. (46) is combinatorially hard so we approximate it in a manner that makes solution tractable. Most simply we can treat the precision matrix $\boldsymbol{\Sigma}_\xi^{-1}$ as diagonal so that decisions with regard to each variable are decoupled. Alternatively, we can use a greedy algorithm to construct the S maximizing Eq. (46). In this case we iteratively add to S the most likely additional term until no term increases the likelihood. The algorithm may be further refined as a forward–backward greedy algorithm. Here we restrict our attention to the diagonal approximation of the posterior covariance. This gives the simple threshold,

$$
\xi_i = 0 \text{ if } \frac{1}{2}\mu_{\xi,i}^2\Sigma_{\xi,ii}^{-1} < \tau, \tag{47}
$$

which is implemented in Alg. 6. The same pruning technique has also been used for connections in Bayesian neural networks, using the variational approximation of the posterior [39]. We call this technique maximum a-posteriori sequential threshold sparse Bayesian learning (MAP-STSBL).

Algorithm 6 MAP-STSBL($\boldsymbol{\Theta}, \mathbf{y}, \sigma^2, \tau$)

```
1: $\boldsymbol{\gamma} = $ ARD$(\boldsymbol{\Theta}, \mathbf{y}, \sigma^2)$
2: $S \approx \underset{S\in\mathcal{P}([d])}{arg\,max}\; p(\boldsymbol{\xi}_{-S}|\mathcal{D},\tau)$
3: $\boldsymbol{\gamma}_{-S} = 0$
4: if $|S| \neq 0$: $\gamma_S = $ MAP-STSBL$(\boldsymbol{\Theta}_S, \mathbf{y}, \sigma^2, \tau)$
5: return $\boldsymbol{\gamma}$
```

4.4. Sparsity properties of MAP-STSBL for orthogonal features

In the case of orthogonal columns of $\boldsymbol{\Theta}$ we can use the same simplification as in Section 4.2.1 to simplify the thresholding criteria in Eq. (47) to,

$$
\begin{aligned}
h_{MAP}(\xi_i^*, \Sigma_{\xi,ii}^*) &= \frac{1}{2}|\xi_i^*|^2\,\Sigma_{\xi,ii}^{*\;-1} \\
&= \frac{|\xi_i^*|^2\left(\sqrt{1+4\rho_i^{-1}\sigma^2|\xi_i^*|^{-2}}+1\right)}{4\rho_i^{-1}\sigma^2} = \tilde{h}_{MAP}(|\xi_i^*|)
\end{aligned}
\tag{48}
$$

which has inverse given by,

$$
\tilde{h}_{MAP}^{-1}(\tau) = 2\sqrt{\frac{\rho_i^{-1}\sigma^2\tau^2}{1+2\tau}} \tag{49}
$$

Then for $\tau > 0$ there is $\tilde{h}_{MAP}^{-1}(\tau)$ such that,

$$
|\xi_i^*| > \tilde{h}_{MAP}^{-1}(\tau) \Leftrightarrow \tilde{h}_{MAP}(\xi_i^*) \geq \tau, \tag{50}
$$

and,

$$
|\xi_i - \rho_i^{-1}\boldsymbol{\Theta}_i^T\boldsymbol{\nu}| > \phi_{\sigma,\rho}\left(\tilde{h}_{MAP}^{-1}(\tau)\right) = \sigma\sqrt{\rho_i^{-1}(2\tau+1)} \Leftrightarrow \tilde{h}_{MAP}(\xi_i^*) \leq \tau. \tag{51}
$$

This gives,

$$
FP_{MAP}(\tau) = p(\xi_i^* \neq 0|\xi_i = 0) = 1 - \text{erf}\left(\sqrt{\frac{2\tau+1}{2}}\right), \tag{52}
$$

and,

$$
\begin{aligned}
FN_{MAP}(\tau) = \frac{1}{2}\Bigg(&\text{erf}\left(\frac{\xi_i + \sigma\sqrt{\rho_i^{-1}(2\tau+1)}}{\sigma\sqrt{2\rho_i^{-1}}}\right) \\
&-\text{erf}\left(\frac{\xi_i - \sigma\sqrt{\rho_i^{-1}(2\tau+1)}}{\sigma\sqrt{2\rho_i^{-1}}}\right)\Bigg).
\end{aligned}
\tag{53}
$$

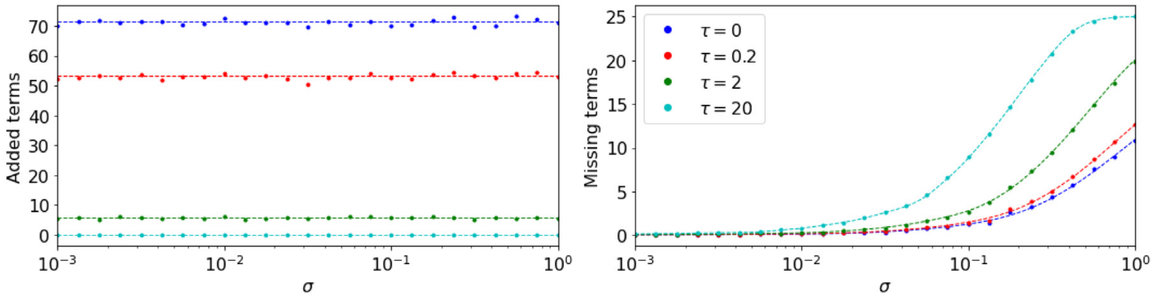

Fig. 6. Number of missing and added terms using MAP-STSBL using 250 × 250 orthonormal Θ with $\|\xi\|_0 = 25$. Dots indicate points where data was collected as the empirical average over 50 trials.

Eqs. (52) and (53) are verified empirically in Fig. 6. The results are very similar to those for ARDvi. Indeed, Eqs. (52) and (53) show that for orthogonal features there is a transformation $\alpha \to 2\tau + 1$ under which ARDvi and MAP-STSBL realize the same sparsity pattern. We will show empirically in a subsequent section that this is not true in the case where columns of Θ are not orthogonal.

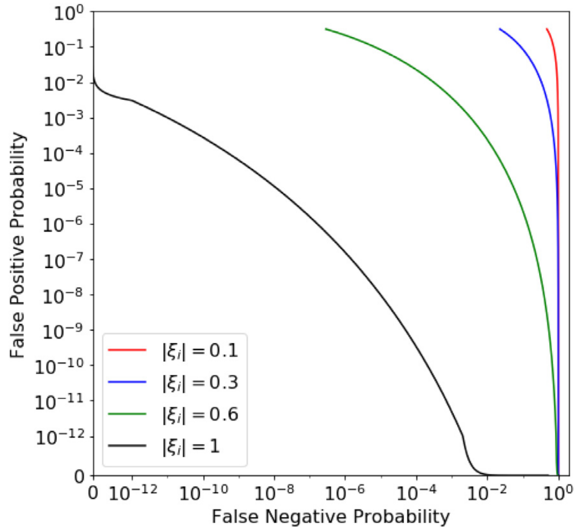

Fig. 7. The FP/FN curve for orthogonal matrices $\Theta^T\Theta = \mathbf{I}$.

5. Comparison

The false positive and negative likelihoods for ξ_i each of the methods described in Sections 3 and 4 are summarized by,

$$
\begin{aligned}
FP_\bullet(\xi_i;\omega) &= 1 - \operatorname{erf}\left(\frac{\psi_\bullet(\omega)}{\sigma\sqrt{2\rho^{-1}}}\right)\\
FN_\bullet(\xi_i;\omega) &= \frac{1}{2}\left(\operatorname{erf}\left(\frac{\xi_i + \sigma\psi_\bullet(\omega)}{\sigma\sqrt{2\rho^{-1}}}\right) - \operatorname{erf}\left(\frac{\xi_i - \sigma\psi_\bullet(\omega)}{\sigma\sqrt{2\rho^{-1}}}\right)\right)
\end{aligned}
\tag{54}
$$

where $\bullet$ refers to the method, ω to the input (α, λ, or τ) and,

$$
\begin{aligned}
\psi_{ARDvi}(\alpha) &= \sigma\sqrt{\alpha\rho_i^{-1}}\\
\psi_{ARDr}(\lambda) &= \sqrt{\rho_i^{-1}\sigma^2 + \lambda\rho_i^{-2}\sigma^4}\\
\psi_{M-STSBL}(\tau) &= \phi_{\sigma,\rho}\,(\tau)\\
\psi_{L-STSBL}(\tau) &= \phi_{\sigma,\rho}\left(\tilde{h}^{-1}_{L,\rho,\sigma}(\tau)\right)\\
\psi_{MAP-STSBL}(\tau) &= \sigma\sqrt{(2\tau+1)\rho_i^{-1}}.
\end{aligned}
\tag{55}
$$

Note that if $\rho_i = \rho_j$ for all i, j then the false positive and negative rates are all equivalent under transformations of the parameters used for each method. Curves $(FP_\bullet(\xi_i;\omega), FN_\bullet(\xi_i;\omega))$ parameterized by $\psi(\omega)$ are shown in Fig. 7 for several values of ξ_i. If ρ_i are unequal then the specific parameter pair that will yield similar results for one column given two different algorithms will not hold for another column. Hence, the methods differ in how they scale with ρ_i. The exception is the pair ARDvi and MAP-STSBL which have the same false positive and negative rates for any ρ_i under the transformation $\alpha = 2\tau + 1$. To visualize the dependence of each false positive and false negative rate on ρ_i we find parameters $\omega_\bullet$ for each method such that the $FP_\bullet(\omega_\bullet) = FN_\bullet(\omega_\bullet)$ when $\rho_i = 1$ and plot the resulting rates over a range of ρ_i. This is shown in Fig. 8. The false negative rate for each method decreases monotonically in ρ_i. This is intuitive, since larger ρ_i corresponds to that term having a larger effect on $\mathbf{y}$. The false negative rate as a function of ρ_i is constant for ARDvi and MAP-STSBL, decreasing for L-STSBL and M-STSBL and increasing for ARDr. These trends are explained by the asymptotic behavior of $\psi_\bullet$ for large ρ. We have,

$$
\begin{aligned}
\psi_{vi/MAP}(\alpha) &\sim \mathcal{O}(\rho^{-1/2})\\
\psi_{ARDr}(\lambda) &\sim \mathcal{O}(\rho^{-1})\\
\psi_{M-STSBL}(\tau) &\sim \mathcal{O}(1)\\
\psi_{L-STSBL}(\tau) &\sim \mathcal{O}(\rho^{q}),\ q \in (-1/2, 0)
\end{aligned}
\tag{56}
$$

where the last statement is inferred from Fig. 8. Since $\psi_\bullet$ is multiplied by $\rho^{1/2}$ in the expression for the false positive rate, which is a decreasing function of $\psi_\bullet$, the trends in Fig. 8 follow from Eq. (56). If we allow ourselves to equate ρ_i with sample size, then M-STSBL and L-STSBL have the desirable property that the false positive rate decreases.

For orthogonal features, ARDvi and MAP-STSBL have equivalent behavior with regard to expected sparsity. However, they begin to yield different results in that case that columns of Θ are correlated. This may be the result of the MAP-threshold criteria no longer aligning with the increased sparsity due to inflated variance, but there is also a fundamental change in the thresholding algorithms which occurs when we move away from orthogonal features. We have shown that when $\Theta^T\Theta$ is diagonal the sparsity of ξ_i depends only on the inner product of the error $\boldsymbol{\nu}$ with Θ_i. Hence, the recursion defined in each thresholding algorithm terminates at a depth of one. This is not true when columns are correlated. For dense $\Theta^T\Theta$ the recursion limit is the number of columns, though the algorithm tends to terminate far earlier. A comparison across matrices with varying condition number is shown in Fig. 9.

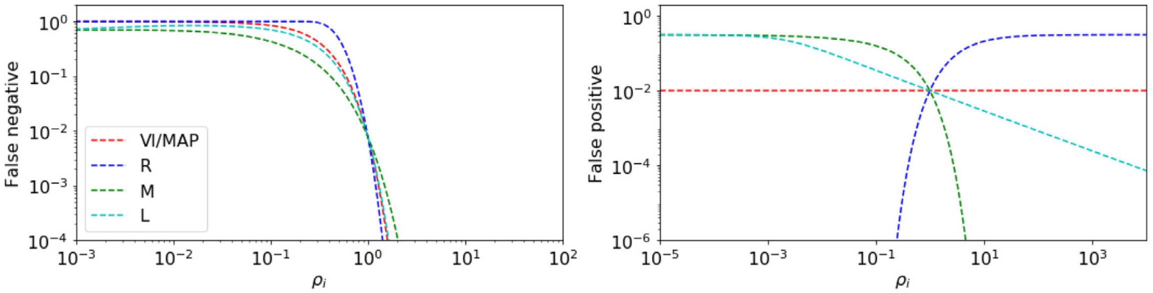

Fig. 8. FP/FN rates for a term with $\|\Theta_i\|^2 = \rho_i$. Parameters for each method have been selected such that $\text{FP}|_{\rho_i=1} = \text{FN}|_{\rho_i=1}$.

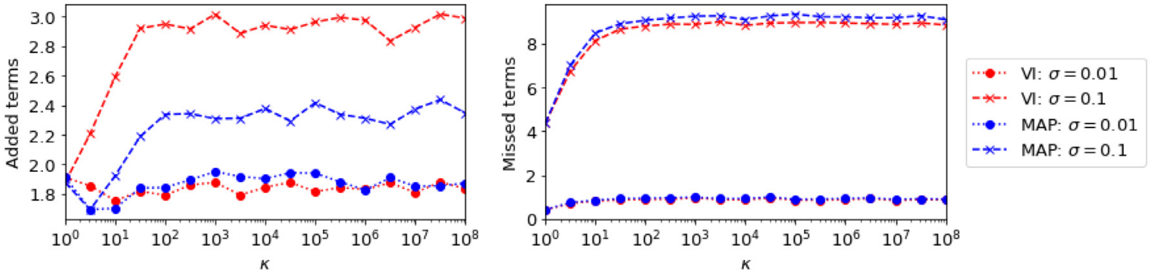

Fig. 9. FP/FN rates ARDvi and MAP-STSBL with two noise magnitudes as a function of $\kappa = \text{Cond}(\Theta)$, the condition number of Θ. Datapoints are averaged over 1000 trials using $\Theta \in \mathbb{R}^{100\times 100}$ and singular values evenly spaced in $[\kappa^{-1}, 1]$ and $\|\xi\|_0 = 25$.

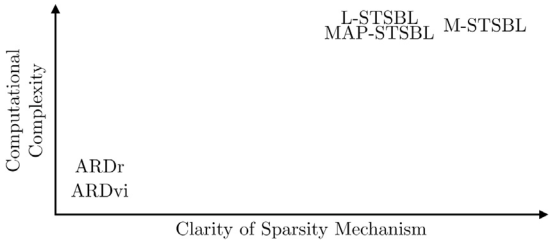

Fig. 10. Simple comparison of the relative merits of the five discussed methods.

An analytical comparison between the algorithms considered in the previous sections for non-orthogonal data is beyond the scope of this work. However, there is one clear trade off between computational complexity and clarity of the algorithm's mechanism for inducing sparsity. Thresholding algorithms offer clear criteria for setting additional terms to zero since we know the magnitude or likelihood at which a coefficient was pruned and at what step. Regularization methods do not provide the same clarity but avoid the cost of increased computational time due to recursion. In particular, for problems with many features, the depth limit in the thresholding algorithms is high. We consider M-STSBL to be slightly more clear than MAP-STSBL and L-STSBL, since the thresholding parameter is a magnitude. We initialize ARDr using ARD, so it is slightly more expensive than ARDvi. This is summarized in Fig. 10. We will present several examples in the following section to compare the algorithms' performance on an empirical basis.

6. Numerical experiments

In this section we compare the performance of each of the methods considered in this work on several test problems. These include a 250 dimensional linear problem, function fitting in a Fourier basis, and system identification for the Lorenz 63, Lorenz 96, and Kuramoto–Sivashinsky equations. We test each of the five methods using a range of input parameters. For the linear and function fitting examples we select optimal parameters for the regularization with the Akaike Information Criterion [40] with small sample size correction (AICc) [41], given by,

$$AIC_c(\boldsymbol{\gamma}) = 2k - 2\ln(p(\boldsymbol{\gamma})) = 2k - \min_{\boldsymbol{\xi}} \left(log|\boldsymbol{\Sigma}_y| + \frac{1}{\sigma^2}\|\mathbf{y} - \Theta\boldsymbol{\xi}\|_2^2 + \boldsymbol{\xi}^T\boldsymbol{\Gamma}^{-1}\boldsymbol{\xi} \right), \tag{57}$$

where $k = \|\boldsymbol{\gamma}\|_0 + 1$ is the number of terms fit by the model including error variance σ^2. For consistency across methods, we do not consider regularization terms when evaluating the likelihood. For examples of nonlinear system identification we found AIC_c selected models with extraneous variables even when the true model was available. This is perhaps due to the errors being non-Gaussian, since we use nonlinear features, and correlated between observations, since numerical differentiation uses adjacent points. We therefore select optimal regularization parameters for the system identification based on minimal mismatch in sparsity to the true solution. This is not practical in an application setting but highlights differences between the algorithms presented in this work without the need for more robust model selection.

Algorithm 3 allows for substantial freedom in the choice of specific regularization functions f and g. For the purposes of comparing with other methods discussed in this work we restrict our attention to the case where f is a constant and $g(\gamma_i) = \lambda\sigma^{-2} min\{\gamma_i, \eta\}$ which is constant for $\gamma_i > \eta$ and linear with positive slope $\lambda\sigma^{-2}$ for $\gamma_i \leq \eta$. We search over parameter λ, keeping η fixed at a value of 0.1. It is reasonable to assume that Alg. 3 may obtain superior results if domain knowledge is available to inform the choice of regularization or if a parameter search is performed over both λ and η.

In each of the following examples, we report the ℓ^1 and ℓ^2 errors between the posterior mean $\boldsymbol{\mu}_\xi$ and the true parameters values used to generate the dataset. We also report the number of added and missed terms, defined by $|\{j : \boldsymbol{\gamma}_j \neq 0 \text{ and } \boldsymbol{\xi}_{true,j} = 0\}|$ and $\{j : \boldsymbol{\gamma}_j = 0 \text{ and } \boldsymbol{\xi}_j \neq 0\}|$, respectively. The ℓ^0 error, which sparse regression methods often seek to minimize, is the sum of the number of added and missed terms.

Table 1
Mean metric and sparsity error for linear system using variations of ARD method. True model has 25 non-zero terms out of 250 dimensions.

Method	ℓ^2 Error	ℓ^1 Error	Added	Missed
ARD	1.2	9.9	65.38	2.15
ARDvi	0.62	4.11	35.13	1.84
ARDr	0.99	7.96	60.44	2.00
M-STSBL	0.35	1.50	3.39	3.21
L-STSBL	0.38	1.76	5.72	2.90
MAP-STSBL	0.47	2.52	12.76	2.74

6.1. Simple linear example

We first consider the methods presented in this work applied to a simple linear regression. We consider a problem with $\mathbf{X} \in \mathbb{R}^{250\times 250}$, $\boldsymbol{\Theta}$ being the identity, and construct random linear maps by setting 25 of 250 coefficients to be Gaussian distributed with unit variance and setting the rest to zero. Since the orthonormal case is explored in Sections 3 and 4 we construct $\mathbf{X}$ (equivalently $\boldsymbol{\theta}(\mathbf{X})$) to have condition number $\kappa(\mathbf{X}) = 10^2$ with singular values spread evenly on a log scale between 10^{-2} and 1. Observations are perturbed by Gaussian noise $\nu \sim \mathcal{N}(0, \sigma^2)$ with $\sigma = 0.1\,\mathrm{std}(\boldsymbol{\Theta\xi})$. That is, σ is set to 10 percent of the standard deviations of the unperturbed output. The magnitude of the noise is not known by the algorithm and is re-estimated after each iteration.

We test each of the methods presented in this work for a total of 100 trials, each with random data, true $\boldsymbol{\xi}$, and noise. Model selection is performed with AIC_c with a wide range of input parameters. Four error metrics are tracked; the ℓ^2 and ℓ^1 differences between the posterior mean estimate and true $\boldsymbol{\xi}$ as well as the number of non-zero terms that the learning algorithm adds and misses. These values are shown in Table 1 and in Fig. 11. Boxes indicate the inter-quartile range and median error across the 100 trials with whiskers indicating maximum and minimum values. Each method gives far higher ℓ^1 than ℓ^2 error indicating these quantities are dominated by the many small terms added by the regression. However, the thresholding based methods exhibit far lower metric error and number of added terms with only a small increase in the number of missed terms.

6.2. Interpolation from few observations

We consider fitting a function defined on $T^2 = [0, \pi]^2$ with sparse representation in a Fourier basis. We let $\mathbf{X} \in [0, \pi)^{250\times 2}$ have rows uniformly sampled on T^2 and $\boldsymbol{\theta} : T^2 \to \mathbb{R}^{900}$ be the mapping to the basis constructed by the first 30 Fourier modes in each direction so that $\boldsymbol{\Theta} \in \mathbb{R}^{250\times 900}$. Similar to the linear example, we set 50 of the 900 coefficients to be Gaussian distributed with unit variance and the rest are set to zero. Noise is again set to have standard deviation equal to 10 percent of the standard deviation of unperturbed values of $\mathbf{y}$ and the magnitude of the noise is re-estimated after each iteration.

Results across 10 trials for fitting a function with a sparse Fourier basis are shown in Fig. 12. Within each trial we test a wide range of input parameters for each technique and select a model using the AIC_c. Regularization and thresholding techniques all exhibit far lower ℓ^1 and ℓ^2 errors and include far fewer extraneous terms. The thresholding methods all show some increase in the number of missed terms. We expect the increase in false negatives would be lower if active terms had magnitudes bounded away from zero.

As was the case for the linear example, all regularized methods exhibit lower metric error than unregularized ARD and add substantially fewer extraneous terms. However, unlike the linear example there is a notable increase in the number of missing terms using thresholding methods and, contrary to intuition, a decrease in the number of missed terms using the regularization based methods.

6.3. Equations of motion for the Lorenz 63 system

Our first example of applying the techniques to a nonlinear system identification problem is the Lorenz 63 system given by,

$$\begin{aligned} \dot{x}_1 &= s(x_2 - x_1) \\ \dot{x}_2 &= x_1(\rho - x_3) - x_2 \qquad (58) \\ \dot{x}_3 &= x_1 x_2 - \beta x_3, \end{aligned}$$

with the standard set of coefficients $s = 10$, $\rho = 28$ and $\beta = \frac{8}{3}$ [42]. We will follow work by [19] for nonlinear system identification and use trajectories from Eq. (58) as data $\mathbf{X}$ and the numerically computed velocity as $\mathbf{y}$.

We construct datasets to test each algorithm by integrating Eq. (58) for 250 steps of length $dt = 0.05$ from an initial condition drawn from $\mathcal{N}\left((0, 0, 15), 5^2\mathbf{I}\right)$ resulting in a times series in $\mathbb{R}^{251\times 3}$. We add Gaussian noise with standard deviation equal to 1 percent of the standard deviation of the time series to get $\mathbf{X}$ and subsequently compute temporal derivatives $y^{(j)} \approx \dot{x}_j$ using a 6th order finite difference scheme applied to the noisy time series. We use the quintic feature map in three variables $\boldsymbol{\theta} : \mathbb{R}^3 \to \mathbb{R}^{\binom{5+3}{5}}$ given by,

$$\boldsymbol{\theta}(x_1, x_2, x_3) = \left(1, x_1, x_2, x_3, x_1^2, x_2^2, x_3^2, x_1x_2, \ldots, x_1^4x_3, x_1^5, x_2^5, x_3^5\right) \qquad (59)$$

This gives a matrix $\boldsymbol{\theta}(\mathbf{X}) \in \mathbb{R}^{251\times 56}$. The system identification problem is then to find sparse solutions to,

$$\dot{x}_j = y^{(j)} = \boldsymbol{\theta}(\mathbf{x})\xi^{(j)} \qquad (60)$$

for each dimension $j = 1, 2, 3$.

Note that since noise is added to the data $\mathbf{X}$ directly rather than to the true $\boldsymbol{\theta}(\mathbf{X})\mathbf{y}$, columns of $\boldsymbol{\theta}(\mathbf{X})$ will be perturbed by nonlinear maps of Gaussian noise. The error in our polynomial regression will therefore be non-Gaussian, violating the likelihood model we start with in Eq. (1). This difference does not significantly affect the regression algorithms but does lead to problems with AIC_c based system identification since the likelihood computed by Eq. (5) makes assumptions regarding error statistics that do not hold. We therefore user oracle model selection, choosing the input parameter that yields the minimal number of added and missed terms compared to the true solution. This of course assumes knowledge of the true solution which would not be the case in an application setting but allows us to focus on comparing sparse regression algorithms rather than on model selection.

We test each of the methods for ten trials, each using the same length of time series but with different random initial conditions and noise instances. Fig. 13 shows error metrics for the coefficients of the learned equations. Since we are solving three distinct problems, the errors shown in Fig. 13 are summed over each of the three dimensions. Thresholding based methods and variance inflation, learn much sparser models than ARD and ARDr, with ARDvi having no increase in the number of missed terms. In this case, M-STSBL outperforms both L-STSBL and MAP-STSBL, possibly due to the fact that none of the true coefficients are small.

6.4. Equations of motion for the Lorenz 96 system

We next the consider the higher dimensional Lorenz 96 system given by,

$$\dot{x}_j = (x_{j+1} - x_{j-2})x_{j-1} - x_j + F \qquad (61)$$

with $n = 40$ and $F = 16$ [43].

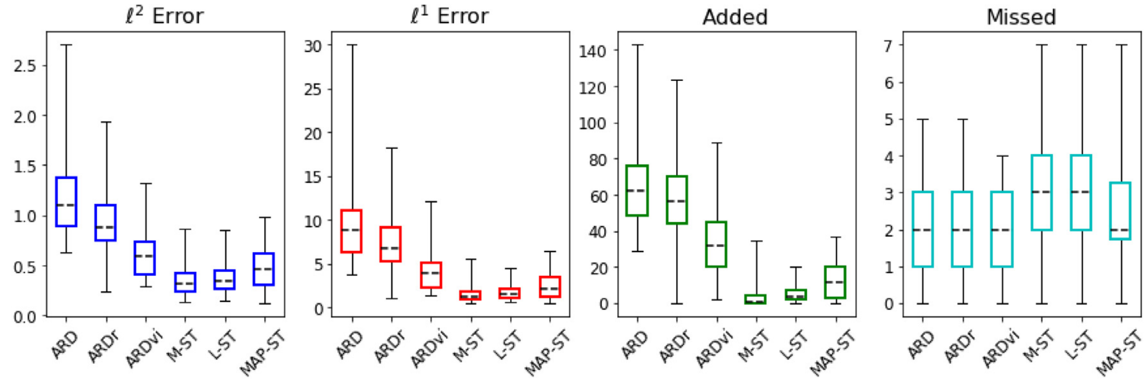

Fig. 11. Error statistics for 100 trials sparse variants of automatic relevance determination with 250 observations of a 250 dimensional problem having 25 non-zero coefficients. Boxes indicate inter-quartile range and median error and whiskers show full range of observed values.

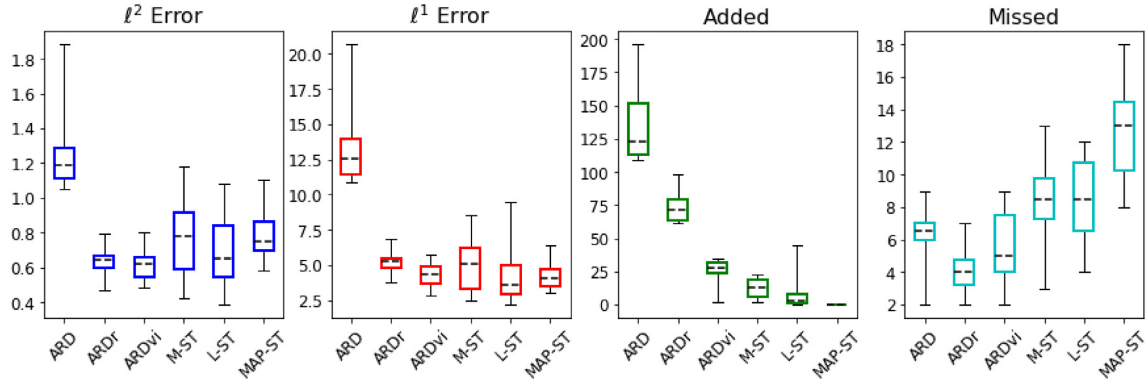

Fig. 12. Error statistics for 10 trials sparse variants of automatic relevance determination with 250 observations in T^2 with two-dimensional Fourier basis $\boldsymbol{\theta} : T^2 \to \mathbb{R}^{900}$ and 50 non-zero coefficients. Boxes indicate inter-quartile range and median error and whiskers show full range of observed values.

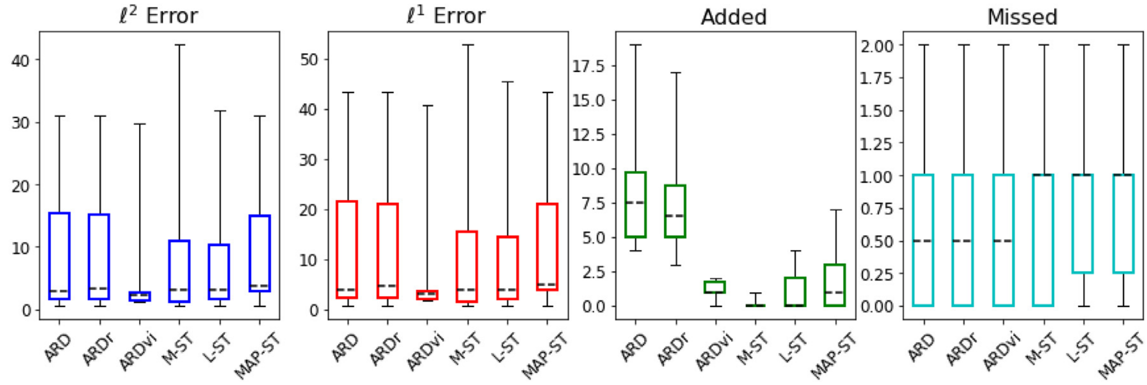

Fig. 13. Error statistics for 10 trials sparse variants of automatic relevance determination with 251 observed timesteps of a single trajectory of the Lorenz 63 system using a quintic polynomial basis. Boxes indicate inter-quartile range and median error and whiskers show full range of observed values. True number of non-zero terms is 7 out of 168 candidate functions.

We construct a dataset to test each algorithm by integrating Eq. (61) with $dt = 0.05$ from an initial condition $x_j = \exp(-\frac{1}{16}(j - 20)^2)$ resulting in a times series in $\mathbb{R}^{200\times 40}$. We add Gaussian noise with standard deviation equal to 1 percent of the standard deviation of the time series to get $\mathbf{X}$ and subsequently compute temporal derivatives $y^{(j)} \approx \dot{x}_j$ using a 6th order finite difference scheme applied to the noisy time series. We use the quadratic feature map in 40 variables $\boldsymbol{\theta} : \mathbb{R}^{40} \to \mathbb{R}^{\binom{2+40}{2}}$ given by,

$$\begin{aligned}\boldsymbol{\theta}(x_1, x_2, \dots, x_{40}) = (&1, x_1, x_2, \dots, x_{40},\\ &x_1^2, x_2^2, \dots, x_{40}^2, x_1x_2, x_1x_3, \dots, x_{39}x_{40})\end{aligned} \tag{62}$$

This gives a matrix $\boldsymbol{\theta}(\mathbf{X}) \in \mathbb{R}^{201\times 861}$. We solve for the equations of motion just as we did in the Lorenz 63 case. Model selection is again performed assuming full knowledge of the true sparsity pattern.

We test each of the methods on a single trial across each of the 40 dimensions. Fig. 14 shows error metrics for the coefficients of the learned equations across the 40 dimensions. Each of the sparse techniques learns a more sparse set of coefficients than ARD with the modal number of added terms for each of the sparse methods being zero. However, metric error is not improved significantly and in the case of MAP-STSBL contains outlier values with substantially increased error where the algorithm

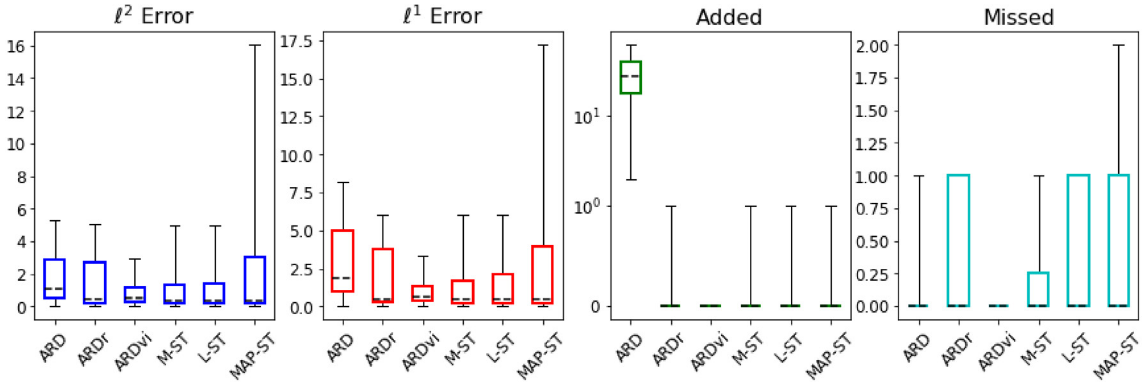

Fig. 14. Error statistics across 40 dimensions using sparse variants of automatic relevance determination with 201 observed timesteps of a single trajectory of the Lorenz 96 system using a quadratic polynomial basis. Boxes indicate inter-quartile range and median error and whiskers show full range of observed values. A symmetric-log scale has been used for the number of added terms due to large differences between ARD and its sparse variants. True number of non-zero terms is 4 out of 861 candidate functions.

failed to include the forcing term $F = 16$. This indicates that extraneous terms in the ARD estimate were generally small. The modal number of missed terms for each method is zero, but all methods except ARDvi added a single term in a some fraction of the dimensions and MAP-STSBL occasionally missed two.

6.5. Equations of motion for the Kuramoto–Sivashinsky equation

Lastly, we test each of the sparse regression methods considered in this work on the Kuramoto–Sivashinsky (KS) equation. The KS equation, given by,

$$u_t + uu_x + u_{xx} + u_{xxxx} = 0, \tag{63}$$

is often used as model for deterministic spatiotemporal chaos and has proved a challenging case for other sparse regression methods [20,44].

We use the ETDRK4 method developed in [45] to solve the KS equation on the domain $(x, t) \in [0, 32\pi] \times [0, 150]$ with periodic boundary conditions, initial condition x, timesteps $dt = 0.14$ and spatial discretization $dx = 32\pi/512$. We add artificial noise to the numerical solution with standard deviation equal to 0.1 percent of the standard deviation of the data. This small magnitude is consistent with previous published works in system identification for the KS equation. Numerical differentiation with respect to time and for the first four spatial derivatives is done by applying sixth order finite difference schemes directly to the noisy data. We take y to be u_t reshaped into a vector and $\boldsymbol{\theta}$ to be the set of powers of u up to 4 multiplied by spatial derivatives up to fourth order so that

$$\boldsymbol{\theta}(\mathbf{X}) = \boldsymbol{\theta}(u(x, t)) = \left(1, u, \dots u^2, u_x, uu_x, \dots u^4 u_{xxxx}\right). \tag{64}$$

With the given discretization of Eq. (63), the feature map (64) gives $\mathbf{y} \in \mathbb{R}^{(1024\cdot 512)\times 1}$ and $\boldsymbol{\theta}(\mathbf{X}) \in \mathbb{R}^{(1024\cdot 512)\times 25}$. Each iteration of Alg. 1 requires storing and inverting $\boldsymbol{\Sigma}_y \in \mathbb{R}^{m\times m}$ where $m 1024\cdot 512$ is the number of observations. Allocating memory for and working with $\boldsymbol{\Sigma}_y$ in this case would be problematic on many standard computers. We instead observe a small fraction of the data through random projections, exploiting the simple fact that,

$$\mathbf{y} = \boldsymbol{\theta}(\mathbf{X})\boldsymbol{\xi} \to \mathbf{C}\mathbf{y} = \mathbf{C}\boldsymbol{\theta}(\mathbf{X})\boldsymbol{\xi}, \tag{65}$$

for any matrix $\mathbf{C}$. We take each column of $\mathbf{C}$ to be a unit direction vector sampled uniformly and without replacement from $\mathbb{R}^{1024\cdot 512}$ so that we are simply sampling rows from the full linear system. To test the effectiveness of each algorithm we take 10 different samples of size 2500 and solve the linear system for each one. Fig. 15 summarized the error of each of the discussed regressions applied to the 10 random subsets given by Eq. (65).

While none of the sparse methods perform well for the task of identifying the Kuramoto–Sivashinsky equation from data, they do learn more parsimonious models than ARD. This comes at the cost of higher ℓ^2 error and a significantly increased number of missing terms. The modal number of missed terms for ARDr is in fact all three of the non-zero terms. While these results are disappointing, they are also unsurprising. The Kuramoto–Sivashinsky equation has proved challenging for past system identification methods [20]. This example showcases some of the limitations of the methodology discussed in this work and the continuing difficulty of sparse regression based methods, both classical and Bayesian, for system identification.

7. Discussion

We have presented several techniques for learning sparse Bayesian methods that build on Automatic Relevance Determination to achieve greater levels of parsimony in the resulting linear model. These methods may be classified in two families; regularization based methods including variance inflation and regularization of $\boldsymbol{\gamma}$, which find the variance coefficients $\boldsymbol{\gamma}$ as the fixed point of a single application of an iterative algorithm, and thresholding based methods, which alternate between solving a smooth optimization problem and simplifying the model via thresholding extraneous terms. For the latter class we tested magnitude based thresholding based on the posterior mean estimate of $\boldsymbol{\xi}$, as well as a likelihood based threshold using the posterior distribution, and adjusting the prior on $\boldsymbol{\xi}$ to find an alternative probabilistic threshold.

For each of these algorithms, we have derived probabilistic estimates for the number of false positive and false negative active terms in the orthogonal case. While most practical problems involve non-orthogonal matrices, these estimates can be taken as guides for the behavior of the algorithms as regularization or thresholding parameters change.

A significant barrier to use of the proposed class of sparse regression methods on many problems is the computational complexity. Each iteration of Alg. 3 requires computing the inverse of an $m \times m$ matrix, where m is the sample size. Future work could explore low-rank approximations of this step, but in the current work this was a computational bottleneck and forced us to only consider small problems. For example, the linear systems tested for the Kuramoto–Sivashinsky equation used subsamples made of only a small fraction of the available data. This technique is common [10,20,29], but may also be exploited for model selection. In particular, the outlier-avoiding subsampling method proposed in [10] could be used in conjunction with any of the

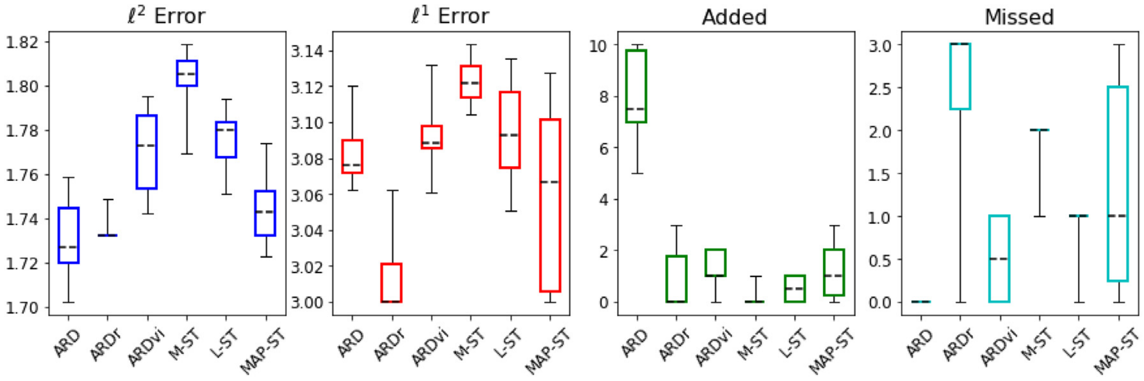

Fig. 15. Error statistics for 10 trials of using 2500 randomly selected rows for finding the Kuramoto–Sivashinsky equation with standard length scales. Boxes indicate inter-quartile range and median error and whiskers show full range of observed values. True number of non-zero terms is 3 out of 25 candidate functions.

sparse variants of ARD discussed in this work and was shown to exhibit increased performance on noisy datasets.

We stress that this work does not attempt to demonstrate the superiority of any particular method for the subset selection problem in sparse Bayesian regression. Ultimately, if one desires a level of sparsity beyond that provided by standard ARD, a choice of additional assumptions should be made with respect to the context of the problem being considered. We have outlined the assumptions that lead to each of the discussed algorithms and demonstrated their accuracy both analytically on orthogonal linear systems as a canonical test case and empirically on several more complicated problems. In application settings, model selection could be performed both over parameter values for each algorithm as well as between algorithms to determine a final result.

CRediT authorship contribution statement

Samuel H. Rudy: Methodology, Software, Formal analysis, Writing - original draft, Funding acquisition. **Themistoklis P. Sapsis:** Project Administration, Supervision, Writing - review & editing, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This material is based upon work supported by the National Science Foundation, United States of America under Award No. 1902972, the Air Force Office of Scientific Research, United States of America (Grant No. FA9550-21-1-0058), the Army Research Office, United States of America (Grant No. W911NF-17-1-0306), and a MathWorks Faculty Research Innovation Fellowship, United States of America.

Appendix A. Proof of Eq. (6)

We show that $\mathbf{y}^T\mathbf{\Sigma}_y^{-1}\mathbf{y} = min_{\boldsymbol{\xi}}\frac{1}{\sigma^2}\|\mathbf{y}-\mathbf{\Theta}\boldsymbol{\xi}\|_2^2+\boldsymbol{\xi}^T\mathbf{\Gamma}^{-1}\boldsymbol{\xi}$. Applying the Woodbury identity to $\mathbf{\Sigma}_y^{-1}$ gives,

$$\begin{aligned}\mathbf{y}^T\mathbf{\Sigma}_y^{-1}\mathbf{y} &= \mathbf{y}^T(\sigma^2\mathbf{I}+\mathbf{\Theta}\mathbf{\Gamma}\mathbf{\Theta}^T)^{-1}\mathbf{y}\\ &= \mathbf{y}^T(\sigma^{-2}\mathbf{I}-\sigma^{-4}\mathbf{\Theta}(\mathbf{\Gamma}^{-1}+\sigma^{-2}\mathbf{\Theta}^T\mathbf{\Theta})^{-1}\mathbf{\Theta}^T)\mathbf{y}\\ &= \mathbf{y}^T(\sigma^{-2}\mathbf{I}-\sigma^{-4}\mathbf{\Theta}\mathbf{\Sigma}_\xi\mathbf{\Theta}^T)\mathbf{y},\end{aligned}$$

On the other hand,

$$\begin{aligned}\underset{\boldsymbol{\xi}}{min}\frac{1}{\sigma^2}\|\mathbf{y}-\mathbf{\Theta}\boldsymbol{\xi}\|_2^2+\boldsymbol{\xi}^T\mathbf{\Gamma}^{-1}\boldsymbol{\xi} &= \frac{1}{\sigma^2}\|\mathbf{y}-\mathbf{\Theta}\boldsymbol{\mu}_\xi\|_2^2+\boldsymbol{\mu}_\xi^T\mathbf{\Gamma}^{-1}\boldsymbol{\mu}_\xi\\ &= \frac{1}{\sigma^2}\|\mathbf{y}-\sigma^{-2}\mathbf{\Theta}\mathbf{\Sigma}_\xi\mathbf{\Theta}^T\mathbf{y}\|_2^2+\sigma^{-4}\mathbf{y}^T\mathbf{\Theta}\mathbf{\Sigma}_\xi\mathbf{\Gamma}^{-1}\mathbf{\Sigma}_\xi\mathbf{\Theta}^T\mathbf{y}\\ &= \frac{1}{\sigma^2}\mathbf{y}^T\mathbf{y}-\frac{2}{\sigma^4}\mathbf{y}^T\mathbf{\Theta}\mathbf{\Sigma}_\xi\mathbf{\Theta}^T\mathbf{y}+\frac{1}{\sigma^6}\mathbf{y}^T\mathbf{\Theta}\mathbf{\Sigma}_\xi\mathbf{\Theta}^T\mathbf{\Theta}\mathbf{\Sigma}_\xi\mathbf{\Theta}^T\mathbf{y}\\ &\quad+\frac{1}{\sigma^4}\mathbf{y}^T\mathbf{\Theta}\mathbf{\Sigma}_\xi\mathbf{\Gamma}^{-1}\mathbf{\Sigma}_\xi\mathbf{\Theta}^T\mathbf{y}\\ &= \frac{1}{\sigma^2}\mathbf{y}^T\mathbf{y}+\frac{1}{\sigma^4}\mathbf{y}^T\mathbf{\Theta}\mathbf{\Sigma}_\xi\\ &\times\left(-2\mathbf{I}+\left(\frac{1}{\sigma^2}\mathbf{\Theta}^T\mathbf{\Theta}+\mathbf{\Gamma}^{-1}\right)\mathbf{\Sigma}_\xi\right)\mathbf{\Theta}^T\mathbf{y}\\ &= \frac{1}{\sigma^2}\mathbf{y}^T\mathbf{y}+\frac{1}{\sigma^4}\mathbf{y}^T\mathbf{\Theta}\mathbf{\Sigma}_\xi\left(-2\mathbf{I}+\mathbf{\Sigma}_\xi^{-1}\mathbf{\Sigma}_\xi\right)\mathbf{\Theta}^T\mathbf{y}\\ &= \frac{1}{\sigma^2}\mathbf{y}^T\mathbf{y}-\frac{1}{\sigma^4}\mathbf{y}^T\mathbf{\Theta}\mathbf{\Sigma}_\xi\mathbf{\Theta}^T\mathbf{y}\\ &= \mathbf{y}^T(\sigma^{-2}\mathbf{I}-\sigma^{-4}\mathbf{\Theta}\mathbf{\Sigma}_\xi\mathbf{\Theta}^T)\mathbf{y},\end{aligned}$$

Appendix B. Converse of Eq. (24)

In this section we show that,

$$\left|\xi_i-\rho_i^{-1}\mathbf{\Theta}_i^T\boldsymbol{\nu}\right| \leq \sqrt{\rho_i^{-1}\sigma^2+\lambda\rho_i^{-2}\sigma^4} \Rightarrow \xi_i^*=0 \tag{66}$$

From the above inequality and the KKT stationarity condition for the ξ we have,

$$\begin{aligned}\sqrt{\rho_i^{-1}\sigma^2+\lambda\rho_i^{-2}\sigma^4} &\geq \left|\xi_i^*+\rho_i^{-1}\sigma^2\sqrt{c_i^*}sgn(\xi_i^*)\right|\\ &= \left|\xi_i^*\right|+\rho_i^{-1}\sigma^2\sqrt{c_i^*}\\ \therefore \sqrt{c_i^*} &\leq \sqrt{\lambda+\rho_i\sigma^{-2}}-\rho_i\sigma^{-2}|\xi_i^*|\\ \text{and } |\xi_i^*| &\leq \sqrt{\rho_i^{-1}\sigma^2+\lambda\rho_i^{-2}\sigma^4}-\rho_i^{-1}\sigma^2\sqrt{c_i^*}\end{aligned} \tag{67}$$

From Eq. (22) $\sqrt{c_i^*}$ is given by the positive valued zero of following cubic,

$$\psi(\omega) = \rho_i^{-1}\sigma^2\omega^3+|\xi_i^*|\omega^2-\left(\lambda\rho_i^{-1}\sigma^2+1\right)\omega-\lambda|\xi_i^*| \tag{68}$$

where $\omega = \sqrt{c_i^*}$ to simplify notation. Note that $\psi(0) \leq 0$ with equality only if λ or $\xi_i^* = 0$, $\psi'(0) < 0$, and the coefficient on the cubic term is positive. This suffices to show there is a unique positive zero of ψ. We also know that $\sqrt{c_i^*}$ is greater than the larger of the two zeros of $\psi'(\omega)$ given by,

$$\sqrt{c_i^*} > \omega^+ = \frac{\rho_i}{3\sigma^2}\left(-|\xi_i^*|+\sqrt{\xi_i^{*2}+3\sigma^2\rho_i^{-1}\left(1+\lambda\sigma^2\rho_i^{-1}\right)}\right). \tag{69}$$

Substituting the lower bound for $\sqrt{c_i^*}$ given by Eq. (69) into Eq. (67) gives,

$$\begin{aligned}
|\xi_i^*| &< \sqrt{\rho_i^{-1}\sigma^2 + \lambda\rho_i^{-2}\sigma^4} \\
&\quad - \frac{1}{3}\left(-|\xi_i^*| + \sqrt{\xi_i^{*2} + 3\sigma^2\rho_i^{-1}\left(1+\lambda\sigma^2\rho_i^{-1}\right)}\right) \\
\frac{2|\xi_i^*|}{3} &< \sqrt{\rho_i^{-1}\sigma^2 + \lambda\rho_i^{-2}\sigma^4} - \frac{1}{3}\sqrt{\xi_i^{*2} + 3\sigma^2\rho_i^{-1}\left(1+\lambda\sigma^2\rho_i^{-1}\right)} \\
|\xi_i^*| &< \frac{3}{2}\sqrt{\rho_i^{-1}\sigma^2 + \lambda\rho_i^{-2}\sigma^4} \\
&\quad - \frac{1}{2}\sqrt{\xi_i^{*2} + 3\sigma^2\rho_i^{-1}\left(1+\lambda\sigma^2\rho_i^{-1}\right)} \\
&\leq \frac{3}{2}\sqrt{\rho_i^{-1}\sigma^2 + \lambda\rho_i^{-2}\sigma^4} - \frac{1}{2}\sqrt{3\sigma^2\rho_i^{-1}\left(1+\lambda\sigma^2\rho_i^{-1}\right)} \\
&= \frac{3-\sqrt{3}}{2}\sqrt{\rho_i^{-1}\sigma^2 + \lambda\rho_i^{-2}\sigma^4}
\end{aligned} \tag{70}$$

From Eq. (67) we know $\sqrt{c_i^*} \leq \sqrt{\lambda + \rho_i\sigma^{-2}} - \rho_i\sigma^{-2}|\xi_i^*|$. Since $\sqrt{c_i^*}$ is the greatest zero of ψ and the cubic coefficient is positive,

$$\begin{aligned}
0 &\leq \psi\left(\sqrt{\lambda + \rho_i\sigma^{-2}} - \rho_i\sigma^{-2}|\xi_i^*|\right) \\
&= \frac{\rho_i|\xi_i^*|}{\sigma^2}\left(|\xi_i^*|\sqrt{\lambda + \rho_i\sigma^{-2}} - 2\lambda\rho_i^{-1}\sigma^2 - 1\right) \\
&\leq \frac{\rho_i|\xi_i^*|}{\sigma^2}\left(\frac{3-\sqrt{3}}{2}(1+\lambda\rho_i^{-1}\sigma^2) - 2\lambda\rho_i^{-1}\sigma^2 - 1\right) \\
&= \frac{\rho_i^{-1}|\xi_i^*|}{\sigma^2}\left(\frac{-1-\sqrt{3}}{2}\lambda\rho_i^{-1}\sigma^2 + \frac{1-\sqrt{3}}{2}\right)
\end{aligned} \tag{71}$$

Note that the quantity inside the parentheses is strictly less than zero. Therefore, for the inequality to hold, $|\xi_i^*| = 0$.

Appendix C. Comparison between L-STSBL and MAP-STSBL

The thresholding operations introduced for Algorithms 5 and 6 bear some similarities but differ in an important manner with regard to how they treat the posterior marginal variance of ξ_i. The thresholding criterion for $\xi_i \to 0$ in Alg. 5 given threshold τ_0 is,

$$h_L(\mu_{\xi,i}, \Sigma_{\xi,ii}) = \frac{1}{\sqrt{2\pi\,\Sigma_{\xi,ii}}}\exp\left(\frac{-\mu_{\xi,i}^2}{2\Sigma_{\xi,ii}}\right) > \tau_0, \tag{72}$$

while for Alg. 6 it is,

$$h_{MAP}(\mu_{\xi,i}, \Sigma_{\xi,ii}) = \frac{\mu_{\xi,i}^2}{2\Sigma_{\xi,ii}} < \tau_1, \tag{73}$$

or equivalently,

$$exp(h_{MAP}(-\mu_{\xi,i}, \Sigma_{\xi,ii})) = \exp\left(\frac{-\mu_{\xi,i}^2}{2\Sigma_{\xi,ii}}\right) > e^{-\tau_1} = \tau_2. \tag{74}$$

The two criteria are related by,

$$h_L(\mu_{\xi,i}, \Sigma_{\xi,ii}) = \frac{exp(-h_{MAP}(\mu_{\xi,i}, \Sigma_{\xi,ii}))}{\sqrt{2\pi\,\Sigma_{\xi,ii}}}. \tag{75}$$

This highlights the difference in assumptions between the two methods. In both algorithms, high uncertainty relative to coefficient magnitude indicates a greater chance of pruning. However, this effect is slightly lessened in Algorithm 5. Coefficients with low uncertainty relative to their magnitude are unlikely to be pruned using either method but the likelihood is higher using 5.

References

[1] Carl Edward Rasmussen, Gaussian processes in machine learning, in: Summer School on Machine Learning, Springer, 2003, pp. 63–71.

[2] Radford M. Neal, Bayesian Learning for Neural Networks, Vol. 118, Springer Science & Business Media, 2012.

[3] Michael E. Tipping, Sparse bayesian learning and the relevance vector machine, J. Mach. Learn. Res. 1 (Jun) (2001) 211–244.

[4] David P. Wipf, Srikantan S. Nagarajan, A new view of automatic relevance determination, in: Advances in Neural Information Processing Systems, 2008, pp. 1625–1632.

[5] Michael E. Tipping, Anita C. Faul, et al., Fast marginal likelihood maximisation for sparse bayesian models, in: AISTATS, 2003.

[6] Yi Wu, David P. Wipf, Dual-space analysis of the sparse linear model, in: Advances in Neural Information Processing Systems, 2012, pp. 1745–1753.

[7] S. Derin Babacan, Rafael Molina, Aggelos K. Katsaggelos, Bayesian compressive sensing using laplace priors, IEEE Trans. Image Process. 19 (1) (2009) 53–63.

[8] D.P. Wipf, B.D. Rao, Sparse bayesian learning for basis selection, IEEE Trans. Signal Process. 52 (8) (2004) 2153–2164.

[9] Sheng Zhang, Guang Lin, Robust data-driven discovery of governing physical laws with error bars, Proc. R. Soc. A 474 (2217) (2018) 20180305.

[10] Sheng Zhang, Guang Lin, Subtsbr to tackle high noise and outliers for data-driven discovery of differential equations, J. Comput. Phys. (2020) 109962.

[11] Ye Yuan, Junlin Li, Liang Li, Frank Jiang, Xiuchuan Tang, Fumin Zhang, Sheng Liu, Jorge Goncalves, Henning U. Voss, Xiuting Li, et al., Machine discovery of partial differential equations from spatiotemporal data, 2019, arXiv preprint arXiv:1909.06730.

[12] R. Fuentes, N. Dervilis, K. Worden, E.J. Cross, Efficient parameter identification and model selection in nonlinear dynamical systems via sparse bayesian learning, in: Journal of Physics: Conference Series, Vol. 1264, IOP Publishing, 2019, 012050.

[13] Vincent Y.F. Tan, Cédric Févotte, Automatic relevance determination in nonnegative matrix factorization with the/spl beta/-divergence, IEEE Trans. Pattern Anal. Mach. Intell. 35 (7) (2012) 1592–1605.

[14] Yi Li, Colin Campbell, Michael Tipping, Bayesian automatic relevance determination algorithms for classifying gene expression data, Bioinformatics 18 (10) (2002) 1332–1339.

[15] Chang Kook Oh, James L. Beck, Masumi Yamada, Bayesian learning using automatic relevance determination prior with an application to earthquake early warning, J. Eng. Mech. 134 (12) (2008) 1013–1020.

[16] Josh Bongard, Hod Lipson, Automated reverse engineering of nonlinear dynamical systems, Proc. Natl. Acad. Sci. USA 104 (24) (2007) 9943–9948.

[17] Michael Schmidt, Hod Lipson, Distilling free-form natural laws from experimental data, Science 324 (5923) (2009) 81–85.

[18] Markus Quade, Markus Abel, Kamran Shafi, Robert K. Niven, Bernd R. Noack, Prediction of dynamical systems by symbolic regression, Phys. Rev. E 94 (1) (2016) 012214.

[19] S.L. Brunton, J.L. Proctor, J.N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, Proc. Natl. Acad. Sci. USA 113 (15) (2016) 3932–3927.

[20] Samuel Rudy, Alessandro Alla, Steven L. Brunton, J. Nathan Kutz, Data-driven identification of parametric partial differential equations, SIAM J. Appl. Dyn. Syst. 18 (2) (2019) 643–660.

[21] Hayden Schaeffer, Learning partial differential equations via data discovery and sparse optimization, in: Proc. R. Soc. A, Vol. 473, The Royal Society, 2017, 20160446.

[22] Abd AlRahman R. AlMomani, Jie Sun, Erik Bollt, How entropic regression beats the outliers problem in nonlinear system identification, Chaos 30 (1) (2020) 013107.

[23] Pileun Kim, Jonathan Rogers, Jie Sun, Erik Bollt, Causation entropy identifies sparsity structure for parameter estimation of dynamic systems, J. Comput. Nonlinear Dyn. 12 (1) (2017).

[24] Robert Tibshirani, Regression shrinkage and selection via the lasso, J. R. Stat. Soc. Ser. B Stat. Methodol. 58 (1) (1996) 267–288.

[25] Peng Zheng, Travis Askham, Steven L. Brunton, J. Nathan Kutz, Aleksandr Y. Aravkin, A unified framework for sparse relaxed regularized regression: Sr3, IEEE Access 7 (2018) 1404–1423.

[26] Giang Tran, Rachel Ward, Exact recovery of chaotic systems from highly corrupted data, Multiscale Model. Simul. 15 (3) (2017) 1108–1129.

[27] Yibo Yang, Mohamed Aziz Bhouri, Paris Perdikaris, Bayesian differential programming for robust systems identification under uncertainty, 2020, arXiv preprint arXiv:2004.06843.

[28] Robert K. Niven, Ali Mohammad-Djafari, Laurent Cordier, Markus Abel, Markus Quade, Bayesian identification of dynamical systems, Multidiscip. Digit. Publ. Inst. Proc. 33 (1) (2020) 33.

[29] Linan Zhang, Hayden Schaeffer, On the convergence of the sindy algorithm, Multiscale Model. Simul. 17 (3) (2019) 948–972.

[30] Tong Zhang, Adaptive forward-backward greedy algorithm for sparse learning with linear models, in: D. Koller, D. Schuurmans, Y. Bengio, L. Bottou (Eds.), Advances in Neural Information Processing Systems, Vol. 21, Curran Associates, Inc., 2009, pp. 1921–1928.

[31] Thomas Blumensath, Mike E. Davies, Iterative hard thresholding for compressed sensing, Appl. Comput. Harmon. Anal. 27 (3) (2009) 265–274.

[32] Christopher M. Bishop, Pattern Recognition and Machine Learning, springer, 2006.

[33] Kevin P. Murphy, Machine Learning: A Probabilistic Perspective, The MIT Press, 2012.

[34] Tong Wu Tong, Kenneth Lange, et al., Coordinate descent algorithms for lasso penalized regression, Ann. Appl. Stat. 2 (1) (2008) 224–244.

[35] Neal Parikh, Stephen Boyd, et al., Proximal algorithms, Found. Trends® Optim. 1 (3) (2014) 127–239.

[36] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al., Distributed optimization and statistical learning via the alternating direction method of multipliers, Found. Trends® Mach. Learn. 3 (1) (2011) 1–122.

[37] Bradley Efron, Trevor Hastie, Iain Johnstone, Robert Tibshirani, et al., Least angle regression, Ann. Statist. 32 (2) (2004) 407–499.

[38] Jianqing Fan, Runze Li, Variable selection via nonconcave penalized likelihood and its oracle properties, J. Amer. Statist. Assoc. 96 (456) (2001) 1348–1360.

[39] Alex Graves, Practical variational inference for neural networks, in: Advances in Neural Information Processing Systems, 2011, pp. 2348–2356.

[40] Hirotugu Akaike, A new look at the statistical model identification, IEEE Trans. Autom. Control 19 (6) (1974) 716–723.

[41] Joseph E. Cavanaugh, Unifying the derivations for the akaike and corrected akaike information criteria, Statist. Probab. Lett. 33 (2) (1997) 201–208.

[42] Edward N. Lorenz, Deterministic nonperiodic flow, J. Atmos. Sci. 20 (2) (1963) 130–141.

[43] Edward N. Lorenz, Predictability: A problem partly solved, in: Proc. Seminar on Predictability, Vol. 1, 1996.

[44] Maziar Raissi, Deep hidden physics models: Deep learning of nonlinear partial differential equations, J. Mach. Learn. Res. 19 (1) (2018) 932–955.

[45] Aly-Khan Kassam, Lloyd N. Trefethen, Fourth-order time-stepping for stiff pdes, SIAM J. Sci. Comput. 26 (4) (2005) 1214–1233.