# Sparse Bayesian Learning via Stepwise Regression

**Sebastian Ament** [1]  **Carla Gomes** [1]

## Abstract

Sparse Bayesian Learning (SBL) is a powerful framework for attaining sparsity in probabilistic models. Herein, we propose a coordinate ascent algorithm for SBL termed Relevance Matching Pursuit (RMP) and show that, as its noise variance parameter goes to zero, RMP exhibits a surprising connection to Stepwise Regression. Further, we derive novel guarantees for Stepwise Regression algorithms, which also shed light on RMP. Our guarantees for Forward Regression improve on deterministic and probabilistic results for Orthogonal Matching Pursuit with noise. Our analysis of Backward Regression on determined systems culminates in a bound on the residual of the optimal solution to the subset selection problem that, if satisfied, guarantees the optimality of the result. To our knowledge, this bound is the first that can be computed in polynomial time and depends chiefly on the smallest singular value of the matrix. We report numerical experiments using a variety of feature selection algorithms. Notably, RMP and its limiting variant are both efficient and maintain strong performance with correlated features.

## 1. Introduction

Finding a sparse solution to underdetermined linear systems is a fundamental problem in a diverse array of domains and applications, like network systems (Haupt et al., 2008), materials science (Szameit et al., 2012), medical imaging (Lustig et al., 2007), and more. The problem can be formalized as

$$\min \|\mathbf{x}\|_0 \quad \text{s.t.} \quad \mathbf{\Phi}\mathbf{x} = \mathbf{y}, \tag{1.1}$$

where $\mathbf{\Phi} \in \mathbb{R}^{n \times m}$ is a matrix with potentially more columns than rows, $\mathbf{y}$ is the observation we are trying to represent using an unknown sparse vector $\mathbf{x}$, and $\|\mathbf{x}\|_0$ stands for the number of non-zero elements in $\mathbf{x}$. A column $\boldsymbol{\varphi}_i$ of $\mathbf{\Phi}$ is

variously referred to as a feature or an atom. This problem has been studied extensively, resulting in myriad existing methods and algorithms.

In the present work, we study two seemingly disparate techniques for solving problem (1.1): Sparse Bayesian Learning (SBL) and Stepwise Regression. SBL is based on the Automatic Relevance Determination (ARD) framework (MacKay, 1992a;b) and is concerned with a generative model of the form

$$\mathbf{\Phi}\mathbf{x} = \mathbf{y} + \boldsymbol{\epsilon}, \tag{1.2}$$

where $\mathbf{\Phi}$ is a deterministic matrix, $\epsilon_i \sim \mathcal{N}(0, \sigma)$ are independent noise variables, and the prior distribution over the coefficients is $x_i \sim \mathcal{N}(0, \gamma_i)$ (Tipping, 2001). Each coefficient has an independent prior variance $\gamma_i$, the defining characteristic of the ARD prior. Sparse Bayesian models are usually trained via type-II maximum likelihood estimation, which is the maximization of the marginal likelihood with respect to $\boldsymbol{\gamma}$. The logarithm of the marginal likelihood is

$$\begin{aligned}\mathcal{L}(\boldsymbol{\gamma}) &\stackrel{\text{def}}{=} \log \int p(\mathbf{y}|\mathbf{x})p(\mathbf{x}|\boldsymbol{\gamma})d\mathbf{x} \\ &= -\mathbf{y}^*\mathbf{C}^{-1}\mathbf{y} - \log|\mathbf{C}| - n\log(2\pi),\end{aligned} \tag{1.3}$$

where $\mathbf{C} \stackrel{\text{def}}{=} (\sigma^2\mathbf{I} + \mathbf{\Phi}\mathbf{\Gamma}\mathbf{\Phi}^*)$, and $\mathbf{\Gamma} \stackrel{\text{def}}{=} \text{diag}(\boldsymbol{\gamma})$. Since every coefficient has its own prior variance, the optimization effectively prunes extraneous features of $\mathbf{\Phi}$ if their prior variances approach zero. Since this provably happens frequently (Wipf and Rao, 2005), ARD and SBL are powerful tools for promoting sparsity in a variety of applications.

Stepwise Regression, on the other hand, is a class of well-known greedy algorithms that add and delete features from a candidate solution based on two rules:

$$\arg\min_{i \notin \mathcal{A}} \|\mathbf{r}_{\mathcal{A} \cup i}\|_2, \quad \text{and} \quad \arg\min_{i \in \mathcal{A}} \|\mathbf{r}_{\mathcal{A} \setminus i}\|_2, \tag{1.4}$$

where $\mathbf{r}_{\mathcal{A}}$ is the least-squares residual using only features indexed by $\mathcal{A}$. The intuitive appeal and practical performance of these greedy heuristics have led separate communities to rediscover the same algorithms, leading to a bewildering number of names. The algorithm that selects features based on the left side of (1.4) is known as Forward Regression, Forward Selection (Miller, 2002), Order-Recursive Matching Pursuit (ORMP) (Cotter et al., 1999), Optimized Orthogonal Matching Pursuit (Rebollo-Neira and Lowe, 2002), and

---

[1]Department of Computer Science, Cornell University, Ithaca, NY, USA. Correspondence to: Sebastian Ament <ament@cs.cornell.edu>.

Orthogonal Least-Squares (Cotter et al., 1999). The algorithm that eliminates features based on the right side of (1.4) is known as Backward Regression, Backward Elimination (Miller, 2002), and Backward Optimized Orthogonal Matching Pursuit (Andrle et al., 2004). Unless otherwise noted, we will refer to them simply as the forward and backward algorithm, respectively.

**Contributions** We propose Relevance Matching Pursuit (RMP$_\sigma$), an algorithm which simultaneously maximizes (1.3) locally, and exhibits a surprising relationship to Stepwise Regression. Indeed, by analyzing RMP$_\sigma$ in the noiseless limit in Section 3, we derive RMP$_0$, a combination of the well-known forward and backward algorithms. Having established this connection, we derive novel theoretical guarantees for Stepwise Regression on noisy data in Section 4. The guarantees for Forward Regression improve on existing deterministic and probabilistic results for Orthogonal Matching Pursuit. The bounds of the tolerable perturbation for Backward Regression are, to our knowledge, the first that can be computed in polynomial time and provide an important insight: the backward algorithm returns the *optimal* solution to the subset selection problem on a determined linear system, as long as the residual of the solution is bounded by a quantity that is proportional to the smallest singular value of the matrix. We experimentally verify our prediction that RMP$_0$ exhibits comparable support recovery performance to RMP$_\sigma$ and compare against some of the most potent feature selection algorithms[1] in Section 5. The results demonstrate RMP's combination of performance and efficiency, corroborating our theoretical analysis. The only method with a consistent performance advantage over RMP$_\sigma$ and RMP$_0$ is the ARD-based reweighted $l_1$-norm minimization of Wipf and Nagarajan (2008), though the approach is computationally substantially more expensive.

## 2. Relevant Work

Given the vast amount of work on sparsity-inducing methods, we focus on the most relevant to ours. In the following, we will use the bra-ket notation $\langle \cdot | \cdot \rangle$ to denote an inner product. A feature that has been selected by an algorithm is considered active and we refer to the set $\mathcal{A}$ of all active features as the active set.

**Basis Pursuit** Basis Pursuit (BP) is a framework for solving (1.1) via the following convex relaxation:

$$\min \|\mathbf{x}\|_1 \quad \text{s.t.} \quad \mathbf{\Phi x} = \mathbf{y}. \tag{2.1}$$

Under certain assumptions on the matrix $\mathbf{\Phi}$ and the sparsity level $k = \|\mathbf{x}\|_0$, (2.1) has the same global optimum as (1.1) (Chen et al., 2001; Candès et al., 2006). An im-

portant modification to BP for noisy observations is Basis Pursuit Denoising (BPDN) which replaces the equality constraint with $\|\mathbf{\Phi x} - \mathbf{y}\| \leq \delta$, where $\delta$ is an upper bound on the perturbation of the signal (Chen et al., 2001; Donoho and Elad, 2006). This is closely related to the least absolute shrinkage and selection operator (LASSO) (Tibshirani, 1996; Zhao and Yu, 2007; Bach, 2008). Cawley et al. (2007) studied this principle in the Bayesian framework, where the $l_1$-regularizer is equivalent to a Laplacian prior. Other notable algorithms based on relaxations of (1.1) include FOCUSS, an iterative least-squares scheme (Gorodnitsky and Rao, 1997), and a reweighted $l_1$-norm minimization algorithm proposed in (Candes et al., 2008), both approximating an entropic regularization term. While algorithms based on relaxations can offer strong sparse recovery guarantees and performance, they can be computationally expensive for large problems. Efficient greedy algorithms have been developed for this reason.

**Matching Pursuit** An important family of greedy algorithms for (1.1) are Matching Pursuit (MP) and its variants (Mallat and Zhifeng Zhang, 1993). Matching Pursuit updates a candidate solution one element at a time. The specific element $i$ is chosen by the rule $\arg\max_i |\langle \boldsymbol{\varphi}_i | \mathbf{r} \rangle|$, where $\boldsymbol{\varphi}_i$ is the $i$th column of $\mathbf{\Phi}$ and $\mathbf{r} = \mathbf{y} - \mathbf{\Phi x}$ is the current residual. Orthogonal Matching Pursuit (OMP), also known as Stagewise Regression, uses the same rule to add features, but additionally optimizes all coefficients of the active set in each iteration:

$$\arg\max_i |\langle \boldsymbol{\varphi}_i | \mathbf{r}_\mathcal{A} \rangle|, \tag{2.2}$$

where $\mathbf{r}_\mathcal{A}$ is the least-squares residual given the set of atoms $\mathcal{A}$. Remarkably, OMP has theoretical guarantees for the problem of recovering the support of exactly sparse signals, even for noisy measurements (Davis et al., 1997; Tropp, 2004; Tropp and Gilbert, 2007; Rangan and Fletcher, 2009; Cai and Wang, 2011). Recently, Matching Pursuits have served as inspiration for optimization algorithms: Tibshirani (2015) proposed a general framework for stagewise algorithms with applications to group-structured learning, matrix completion, and image denoising. Locatello et al. (2018) developed a unified analysis of MP and coordinate ascent algorithms and Combettes and Pokutta (2019) proposed Blended Matching Pursuit, combining coordinate descent and gradient steps to compute sparse minimizers of general convex objectives quickly.

**Stepwise Regression** We here focus on known theoretical guarantees of the forward and backward algorithms and existing algorithms that combine them. Das and Kempe (2018) proposed a notion of approximate submodularity and showed that it is satisfied by the coefficient of determination, $R^2$. In this way, they proved approximation guarantees of

---

[1] Code made available at `CompressedSensing.jl`.

the forward algorithm and OMP to the optimal solution of the subset selection problem, which were generalized by Elenberg et al. (2018) for general convex objectives. Couvreur and Bresler (2000) analyzed the backward algorithm and proved the existence of a bound on the perturbation magnitude that guarantees the recovery of the support of sparse solutions of linear systems.

Andrle et al. (2004) proposed running the forward and backward algorithm consecutively, but did not provide theoretical guarantees, nor empirical comparisons against other algorithms. Zhang (2009) proposed FoBa, which combines both forward and backward heuristics into an adaptive algorithm. Similarly, Rao et al. (2015) proposed a forward-backward algorithm for the optimization of convex relaxations of (1.1) based on atomic norms and, most recently, Borboudakis and Tsamardinos (2019) proposed an early-dropping heuristic for forward-backward algorithms for general feature selection problems.

**Sparse Bayesian Learning** The first algorithms for the optimization of (1.3) for SBL were based on expectation-maximization (EM) updates and the fixed-point updates of MacKay (Tipping, 2001). Though these methods are able to obtain sparse solutions to (1.2), they have no convergence guarantees and are slow for large problems, due to the at least quadratic scaling with the number of features (Tipping, 2001). Wipf and Rao (2004) showed how to adapt the EM-based SBL algorithm to the $l_0$-minimization problem (1.1) and proved that, in contrast to BP, the resulting optimization problem has the same global optimum as (1.1) and suffers from fewer local minima than competing non-convex relaxations. Subsequently, Wipf and Nagarajan (2008) showed that the usual type-II approach can be interpreted as a type-I (MAP) approach with a special non-factorial prior. Using this insight, they proposed an algorithm based on reweighted $l_1$-norm minimization, which provably converges to a local maximum of the marginal likelihood, performs at least as well as BP in recovering sparse signals, and usually outperforms techniques based on $l_1$, $l_2$, and entropy regularization (Wipf and Nagarajan, 2009), especially when dictionaries are structured and coherent (Wipf, 2011).

## 3. Relevance Matching Pursuit

This section first recapitulates the derivation of the coordinate ascent updates for SBL derived by Tipping and Faul (2003), subsequently introduces the Relevance Matching Pursuit algorithm, and analyzes the algorithm's behavior as the noise variance approaches zero.

### 3.1. SBL via Coordinate Ascent

Recall from the introduction that $\mathbf{C} = (\sigma^2 \mathbf{I} + \mathbf{\Phi}\mathbf{\Gamma}\mathbf{\Phi}^*)$ is the covariance of the marginal distribution and $\mathbf{\Gamma} = \mathrm{diag}(\boldsymbol{\gamma})$

is the prior variance of the weights $\mathbf{x}$. In the context of SBL, we refer to $\mathcal{A} = \{i|\gamma_i \neq 0\}$ as the active set. Following the analysis of Tipping and Faul (2003) and based on the Woodbury matrix identity, we separate out the contribution of a single prior variance $\gamma_i$ to the marginal likelihood (1.3):

$$\mathcal{L}(\boldsymbol{\gamma}) = \mathcal{L}(\boldsymbol{\gamma}_{-i}) + \frac{1}{2}\left(\frac{q_i^2}{\gamma_i^{-1} + s_i} - \log\frac{1}{1 + \gamma_i s_i}\right)$$
$$= \mathcal{L}(\boldsymbol{\gamma}_{-i}) + l(\gamma_i),$$

where $q_i \stackrel{\mathrm{def}}{=} \boldsymbol{\varphi}_i \mathbf{C}_{\mathcal{A}\backslash i}^{-1}\mathbf{y}$ and $s_i \stackrel{\mathrm{def}}{=} \boldsymbol{\varphi}_i \mathbf{C}_{\mathcal{A}\backslash i}^{-1}\boldsymbol{\varphi}_i$, also termed the "quality" and "sparsity" factors by Faul and Tipping (2002). $\mathbf{C}_{\mathcal{A}\backslash i}$ is as in (1.3) but only includes the features $\boldsymbol{\varphi}_k$ and corresponding prior variances $\gamma_k$ for $k \in \mathcal{A}\backslash i$. Crucially, the argument of the maximum of the marginal likelihood with respect to a single prior variance $\gamma_i$ is unique and has a closed form:

$$\arg\max_{\gamma_i} l(\gamma_i) = \begin{cases} (q_i^2 - s_i)/s_i^2 & q_i^2 > s_i \\ 0 & \text{else} \end{cases}. \quad (3.1)$$

Equation (3.1) is the basis of the efficient coordinate ascent updates put forth in Tipping and Faul (2003). Associated with each coordinate update is a change in the marginal likelihood. If $q_i^2 > s_i$, we denote by $\Delta_{\mathrm{add}}(i)$, respectively $\Delta_{\mathrm{update}}(i)$, the change in the marginal likelihood corresponding to setting a $\gamma_i$ which was previously zero, respectively non-zero, via equation (3.1).

We now make two preliminary observations. Given a subset of features $\mathcal{A}$, and a noise variance $\sigma^2$, the posterior mean $\boldsymbol{\mu}_{\mathcal{A},\sigma}$ and variance $\boldsymbol{\Sigma}_{\mathcal{A},\sigma}$ of the subset of weights $\mathbf{x}_{\mathcal{A}}$ are given by

$$\boldsymbol{\Sigma}_{\mathcal{A},\sigma} = (\mathbf{\Gamma}_{\mathcal{A}}^{-1} + \sigma^{-2}\mathbf{\Phi}_{\mathcal{A}}^T\mathbf{\Phi}_{\mathcal{A}})^{-1},$$
$$\boldsymbol{\mu}_{\mathcal{A},\sigma} = \sigma^{-2}\boldsymbol{\Sigma}_{\mathcal{A}}\mathbf{\Phi}_{\mathcal{A}}^*\mathbf{y},$$

where $\mathbf{\Phi}_{\mathcal{A}}$ and $\mathbf{\Gamma}_{\mathcal{A}}$ are the submatrices of $\mathbf{\Phi}$ and $\mathbf{\Gamma}$ corresponding to $\mathcal{A}$. The Woodbury identity gives

$$\mathbf{R}_{\mathcal{A},\sigma} \stackrel{\mathrm{def}}{=} \sigma^2 \mathbf{C}^{-1} = \mathbf{I} - \mathbf{\Phi}_{\mathcal{A}}[\sigma^{-2}\boldsymbol{\Sigma}_{\mathcal{A}}]\mathbf{\Phi}_{\mathcal{A}}^*,$$

and thus, $\mathbf{r}_{\mathcal{A},\sigma} \stackrel{\mathrm{def}}{=} \mathbf{y} - \mathbf{\Phi}_{\mathcal{A}}\boldsymbol{\mu}_{\mathcal{A},\sigma} = \mathbf{R}_{\mathcal{A},\sigma}\mathbf{y}$. We can now express the following result on the condition which leads SBL to include or exclude a feature.

**Lemma 3.1.** *The optimum of the marginal likelihood with respect to $\gamma_i$ occurs at a non-zero value if and only if*

$$|\langle\tilde{\boldsymbol{\varphi}}_i|\mathbf{r}_{\mathcal{A}\backslash i,\sigma}\rangle| > \sigma,$$

*where $\tilde{\boldsymbol{\varphi}}_i \stackrel{\mathrm{def}}{=} \boldsymbol{\varphi}_i/\|\boldsymbol{\varphi}_i\|_{\mathbf{R}_{\mathcal{A}\backslash i,\sigma}}$, $\|\boldsymbol{\varphi}_i\|_{\mathbf{R}_{\mathcal{A}\backslash i,\sigma}}$ is the energetic norm $(\boldsymbol{\varphi}_i^*\mathbf{R}_{\mathcal{A}\backslash i,\sigma}\boldsymbol{\varphi}_i)$ of $\boldsymbol{\varphi}_i$, and $\mathcal{A}$ is the active set.*

Lemma 3.1 shows the direct and proportional dependency of the acquisition and deletion conditions on $\sigma$. The following result characterizes the inactive feature ($\gamma_i = 0$) that leads to the maximal increase in the marginal likelihood upon its addition to the model.

**Lemma 3.2.** *Let $\Delta_{add}(i)$ be the change in the marginal likelihood upon setting an inactive feature's prior variance $\gamma_i$ to its optimal value via equation* (3.1). *Then*

$$\arg\max_{i \notin \mathcal{A}} \Delta_{add}(i) = \arg\max_{i \notin \mathcal{A}} |\langle \tilde{\varphi}_i | \mathbf{r}_{\mathcal{A}, \sigma} \rangle|.$$

*where $\tilde{\varphi}_i = \varphi_i / \|\varphi_i\|_{\mathbf{R}_{\mathcal{A}, \sigma}}$, and $\mathcal{A}$ is the active set.*

By comparing the right side of the equation in Lemma 3.2 to the acquisition criterion of OMP in (2.2), we see that a feature selection strategy based on the maximal increase in the marginal likelihood is intimately related to the family of Matching Pursuit algorithms. This serves as the inspiration for the name of Relevance Matching Pursuit (RMP), described in the next section.

## 3.2. Algorithm Design

In describing the coordinate ascent updates, Tipping and Faul (2003) purposefully left several choices open: which variance $\gamma_i$ does the algorithm choose to update, add, or delete? In which order should these operations proceed? RMP arises from a particular choice for these design questions, enabling our analysis of the algorithm's behavior, and proving to be closely related to Stepwise Regression. The design principles give rise to Algorithm 1 and are as follows:

1) Add features based on $\arg\max_{i \notin \mathcal{A}} |\langle \tilde{\varphi}_i | \mathbf{r} \rangle|$, until there is no inactive feature with $|\langle \tilde{\varphi}_i | \mathbf{r} \rangle| > \sigma$ left.

2) Remove features based on $\arg\min_{i \notin \mathcal{A}} |\langle \tilde{\varphi}_i | \mathbf{r} \rangle|$, as long as there is a feature with $|\langle \tilde{\varphi}_i | \mathbf{r} \rangle| \leq \sigma$.

3) Update the prior variance of the currently active atom whose update leads to the largest increase in the marginal likelihood, as long as there is a feature with $\Delta_{\text{update}}(i) > \delta_{\mathcal{L}}$, where $\delta_{\mathcal{L}}$ is an input to the algorithm and defines a convergence criterion.

In Algorithm 1, we left the condition of the outer loop imprecise for a reason: in addition to terminating after an improvement to the likelihood fails to exceed $\delta_{\mathcal{L}}$ and no feature is left to add or delete, an implementation might include additional criteria, like a maximum runtime, number of iterations, or change in $\gamma$. We further note that the coordinate ascent updates to $\gamma_i$ do provably converge to a joint maximum, not merely a stationary point (Faul and Tipping, 2002). These facts imply

**Lemma 3.3.** *As $\delta_{\mathcal{L}} \to 0$, the $\gamma$ returned by $RMP_\sigma$ constitutes a local maximum of the marginal likelihood.*

## 3.3. The Noiseless Limit

We now analyze the algorithm's behavior as the noise variance $\sigma^2$ approaches zero. On a high level, this is analogous

---

**Algorithm 1** Relevance Matching Pursuit (RMP$_\sigma$)

1: **Input:** Dictionary $\mathbf{\Phi}$, signal $\mathbf{y}$, noise variance $\sigma^2$, convergence criterion $\delta_{\mathcal{L}}$
2: **Ouput:** Support set $\mathcal{A}$, prior variances $\gamma_i$
3: Initialize $\mathcal{A} \leftarrow \varnothing$
4: Initialize $\gamma_i \leftarrow 0$ for all $i$
5: **while** has not converged **do**
6:     **while** $\exists i \notin \mathcal{A}$ s.t. $|\langle \tilde{\varphi}_i | \mathbf{r}_{\mathcal{A}, \sigma} \rangle| > \sigma$ **do**
7:         $i^* \leftarrow \arg\max_{i \notin \mathcal{A}} |\langle \tilde{\varphi}_i | \mathbf{r}_{\mathcal{A}, \sigma} \rangle|$    {selection}
8:         $\mathcal{A} \leftarrow \mathcal{A} \cup i^*$    {add to active set}
9:         update $\gamma_{i^*}$    {set $\gamma_{i^*}$ via (3.1)}
10:     **end while**
11:     **while** true **do**
12:         **if** $\exists i \in \mathcal{A}$ s.t. $|\langle \tilde{\varphi}_i | \mathbf{r}_{\mathcal{A} \setminus i, \sigma} \rangle| \leq \sigma$ **then**
13:             $i^* \leftarrow \arg\min_{i \in \mathcal{A}} |\langle \tilde{\varphi}_i | \mathbf{r}_{\mathcal{A} \setminus i, \sigma} \rangle|$
14:             $\mathcal{A} \leftarrow \mathcal{A} \setminus i^*$    {elimination}
15:         **else if** $\exists i$ s.t. $\Delta_{\text{update}}(i) > \delta_{\mathcal{L}}$ **then**
16:             $i^* \leftarrow \arg\max_{i \in \mathcal{A}} \Delta_{\text{update}}(i)$    {update}
17:         **else**
18:             break
19:         **end if**
20:         update $\gamma_{i^*}$    {set $\gamma_{i^*}$ via (3.1)}
21:     **end while**
22: **end while**

---

to the approach of Wipf and Rao (2004; 2005) who studied the noiseless limit of the EM-updates for SBL. We make use of the following property of the Moore-Penrose inverse.

**Lemma 3.4.** *Assume the columns of $\mathbf{\Phi}_{\mathcal{A}}$, are linearly independent. Then*

$$\mathbf{R}_{\mathcal{A}} \stackrel{def}{=} (\mathbf{I} - \mathbf{\Phi}_{\mathcal{A}} \mathbf{\Phi}_{\mathcal{A}}^+) = \lim_{\sigma \to 0^+} \mathbf{R}_{\mathcal{A}, \sigma}.$$

Note that $\mathbf{r}_{\mathcal{A}} = \mathbf{R}_{\mathcal{A}} \mathbf{y}$ is the ordinary least-squares residual of the active features $\mathbf{\Phi}_{\mathcal{A}}$ and $\mathbf{y}$. The following technical result is crucial in establishing the connection of RMP$_\sigma$ to Stepwise Regression.

**Lemma 3.5.** *Let $\mathbf{r}_{\mathcal{A}}$ be the least-squares residual associated with a feature set $\mathcal{A}$. Then*

$$\|\mathbf{r}_{\mathcal{A}}\|_2^2 - \|\mathbf{r}_{\mathcal{A} \cup i}\|_2^2 = |\langle \varphi_i, \mathbf{r}_{\mathcal{A}} \rangle|^2 / \|\varphi_i\|_{\mathbf{R}_{\mathcal{A}}}^2, and$$
$$\|\mathbf{r}_{\mathcal{A} \setminus i}\|_2^2 - \|\mathbf{r}_{\mathcal{A}}\|_2^2 = |\langle \varphi_i, \mathbf{r}_{\mathcal{A}} \rangle|^2 / \|\varphi_i\|_{\mathbf{R}_{\mathcal{A} \setminus i}}^2.$$
(3.2)

An immediate corollary of Lemma 3.5 is that $\arg\min_i \|\mathbf{r}_{\mathcal{A} \cup i}\| = \arg\max_i |\varphi_i \cdot \mathbf{r}_{\mathcal{A}}| / \|\varphi_i\|_{\mathbf{R}_{\mathcal{A}}}$, and a similar expression for the second equation in (3.2). As Lemma 3.4 implies that the addition criteria of RMP$_\sigma$ converge to the right-hand side of this expression, the criteria in fact converge to the ones of the forward and backward algorithm in equation (1.4).

It remains to study what happens to the prior-variance-update step in line 16 of Algorithm 1. Notably, $\mathbf{R}_{\mathcal{A}}$ is

**Algorithm 2** RMP$_0$

1: **Input:** Dictionary $\mathbf{\Phi}$, signal $\mathbf{y}$, tolerance $\delta$
2: **Result:** Support set $\mathcal{A}$
3: Initialize $\mathcal{A} \leftarrow \varnothing$
4: **while** has not converged **do**
5:     **while** $\exists i \in \mathcal{A}$ s.t. $\|\mathbf{r}_{\mathcal{A}}\|^2 - \|\mathbf{r}_{\mathcal{A} \cup i}\|^2 > \delta^2$ **do**
6:         $i^* \leftarrow \arg\max_{i \notin \mathcal{A}} \|\mathbf{r}_{\mathcal{A} \cup i}\|$ {selection}
7:         $\mathcal{A} \leftarrow \mathcal{A} \cup i^*$
8:     **end while**
9:     **while** $\exists i \in \mathcal{A}$ s.t. $\|\mathbf{r}_{\mathcal{A} \setminus i}\|^2 - \|\mathbf{r}_{\mathcal{A}}\|^2 \leq \delta^2$ **do**
10:        $i^* \leftarrow \arg\min_{i \in \mathcal{A}} \|\mathbf{r}_{\mathcal{A} \setminus i}\|$ {elimination}
11:        $\mathcal{A} \leftarrow \mathcal{A} \setminus i^*$
12:     **end while**
13: **end while**

independent of $\mathbf{\Gamma}$ under the assumption of Lemma 3.4, and thus, so are the addition and deletion criteria of RMP$_\sigma$ in this limit. Therefore, updating and keeping track of $\mathbf{\Gamma}$ is irrelevant for the execution of the algorithm if the active set is linearly independent.

This observation gives rise to Algorithm 2, which we term RMP$_0$, which models RMP$_\sigma$ when there are no linearly dependent column in the active set. This is not a restrictive assumption since RMP$_0$ starts with an empty $\mathcal{A}$ and stops adding columns when a feasible solution is found since then $\mathbf{r}_{\mathcal{A}} = 0$, which happens at the latest when $n$ columns are added. Inspired by the acquisition criterion in Lemma 3.1, we introduce an acquisition and deletion threshold $\delta$ separate from $\sigma$ to Algorithm 2, which makes the algorithm capable of handling noise. Setting $\delta \leftarrow 0$ corresponds to the noiseless limit of RMP$_\sigma$.

## 4. Stepwise Regression

Having exposed a connection of Relevance Matching Pursuit to Stepwise Regression, we now provide novel theoretical insights about the forward and backward algorithms. To this end, we briefly introduce necessary theoretical tools.

### 4.1. Theoretical Preliminaries

Prior work has established that the performance of many feature selection and sparse recovery algorithms is highly dependent on the correlation of different features. The following definition quantifies this notion.

**Definition 4.1** (Coherence). *The coherence $\mu$ of a matrix $\mathbf{\Phi}$, whose columns have unit norm, is defined as*

$$\mu \stackrel{def}{=} \max_{i \neq j} |\langle \boldsymbol{\varphi}_i | \boldsymbol{\varphi}_j \rangle|.$$

The coherence is a measure of the orthogonality of $\mathbf{\Phi}$. It can lead to pessimistic estimates, as it only considers the

maximal inner product of two columns. Tropp (2004) introduced the Babel function to generalize the coherence by measuring the maximal sum of absolute inner products between a column and *a set* of columns. Its name is inspired by the Tower of Babel, since the function measures "how much the atoms are speaking the same language", and it can be used to derive sharper results.

**Definition 4.2** (Babel Function). *The Babel function $\mu_1$ of a dictionary $\mathbf{\Phi}$ is defined as*

$$\mu_1(k) \stackrel{def}{=} \max_{|\mathcal{I}|=k} \max_{i \notin \mathcal{I}} \sum_{j \in \mathcal{I}} |\langle \boldsymbol{\varphi}_i | \boldsymbol{\varphi}_j \rangle|.$$

*Notably, $\mu_1(1) = \mu$ and $\mu_1(k) \leq \mu k$ (Tropp, 2004).*

Using this notion, Tropp (2004) proved that a necessary and sufficient condition for Orthogonal Matching Pursuit to recover *any* $k$-sparse vector is the Exact Recovery Condition.

**Theorem 4.3** (Tropp (2004)). *Orthogonal Matching Pursuit and Basis Pursuit succeed in recovering the support $\mathcal{S}$ of a $k$-sparse $\mathbf{x}$ from $\mathbf{y} = \mathbf{\Phi}\mathbf{x}$ if*

$$\max_{j \notin \mathcal{S}} \|\mathbf{\Phi}_{\mathcal{S}}^+ \boldsymbol{\varphi}_j\|_1 < 1.$$

*Further, this holds if $\mu_1(k) < 1/2$.*

Soussen et al. (2013) used the connection of the forward algorithm and OMP that was exposed through Lemma 3.5 to jointly analyze the two algorithms, proving that the exact recovery criterion in Theorem 4.3 is necessary and sufficient for both algorithms to retrieve *any* sparse signal with no noise. In particular, for each algorithm, there is a sparse signal which cannot be recovered in $k$ steps, if the inequality doesn't hold. In this sense, the recovery guarantee for the forward algorithm without noise cannot be improved. Tropp (2004) further points out that even the condition on the Babel function is necessary for exact recovery.

### 4.2. Forward Regression

For the results in this subsection, assume that $\mathbf{\Phi}$ has $l_2$-normalized columns. In comparison to the work of Cai and Wang (2011) on OMP with noise, our main theoretical contributions for the analysis of the forward algorithms are related to the necessary recovery condition on the Babel function, $\mu_1(k) < 1/2$, and its implications. Our analysis leads to both tighter deterministic and probabilistic bounds on the tolerable noise magnitude.

**Theorem 4.4.** *Orthogonal Matching Pursuit and Forward Regression recover the support of a $k$-sparse vector in $k$ iterations provided the Babel function $\mu_1$ and the perturbation $\epsilon$ of the target $\mathbf{y}$ satisfy*

$$\frac{1 - 2\mu_1(k)}{\sqrt{2[1 + \mu_1(k)]}} \min_{i \in \mathcal{S}} |x_i| \geq \|\epsilon\|_2.$$

Theorem 4.4 allows for a non-zero amount of noise as long as $\mu_1(k) < 1/2$, which is necessary even in the noiseless case. The tolerable magnitude increases with decreasing Babel function value, and is proportional to the magnitude of the smallest non-zero entry in $\mathbf{x}$. Compared to Thm. 10 of Bruckstein et al. (2009) and Thm. 1 of Cai and Wang (2011), Thm. 4.4 applies to both OMP and Forward Regression (FR) and improves the factor of 2 in both prior results to $\sqrt{2(1 + \mu_1(k))}$. This allows up to around 40% more noise while generalizing the results, and converges to $\sqrt{2}$ as $\mu \to 0$, the provably tightest constant even for orthogonal matrices. We also derived the following novel probabilistic guarantee for perturbations that are normally distributed.

**Theorem 4.5.** *Suppose* $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_n)$, *and let* $\delta = [1/2 - \mu_1(k)] \min_{i \in \mathcal{S}} |x_i|/\sigma > 0$ *and* $d = \lceil m/k \rceil$. *Orthogonal Matching Pursuit and Forward Regression recover the support of a $k$-sparse signal with probability exceeding*

$$1 - \left\lceil \frac{m}{k} \right\rceil \left( \frac{1 + \kappa_1(k)}{1 - \mu_1(2k)} \right)^{k/2} \left( 1 - \mathrm{erf}(\delta/\sqrt{2\kappa_1(k)}) \right)^k.$$

*where* $\kappa_1(k) \stackrel{def}{=} (1 + \mu_1(2k))/(1 - \mu_1(k))$. *For* $\mu_1(2k) < 1/2$, *the result further holds with probability exceeding*

$$1 - \left\lceil \frac{m}{k} \right\rceil \left( \frac{4}{\sqrt{\pi}\delta} e^{-\delta^2/6} \right)^k.$$

By comparison, the existing bounds of Ben-Haim et al. (2010) and Cai and Wang (2011) bound the probability of failure by $2me^{-\delta^2/2}/\delta$. Theorem 4.5 thus guarantees an *earlier* and much *sharper* phase transition (see Fig. 1). Critically, Theorem 4.5 makes use of the approximately isometric structure of subsets of columns of $\boldsymbol{\Phi}$, which is already inherent to the deterministic analysis, to derive a strong bound for $\|\boldsymbol{\Phi}^*\boldsymbol{\epsilon}\|$, for which prior works applied bounds for generic $\boldsymbol{\Phi}$. This allows us to apply much stronger *multiplicative* bounds since $\boldsymbol{\Phi}_{\mathcal{A}}^*\boldsymbol{\epsilon}$, an important quantity in the analysis, has an approximately diagonal covariance.

**Sparse Approximation and Exact Recovery** Elenberg et al. (2018) and Das and Kempe (2018) provide elegant theoretical insights on the performance of OMP and FR. Both works present *approximation* guarantees, but do not provide *exact recovery* guarantees, as Theorems 4.4 and 4.5. The results are thus highly complementary; neither subsumes the other. To elaborate, their results for the $R^2$ score guarantee that FR's result is at most a factor of $(1 - e^{-\gamma})$ from the optimal value, where $\gamma = \sigma_{\min}^2(\boldsymbol{\Phi}_{\mathcal{S}})$. This is a strong result because it holds generally, even for non-sparse vectors and arbitrary noise. On the other hand, if $\gamma > 1/2$ and a sparse vector generated the target with small noise, then Theorem 4.4 guarantees the exact recovery of the support, while the approximation guarantee with $\gamma \approx 1/2$ only ensures FR to explain $\approx 40\%$ of the target variance.
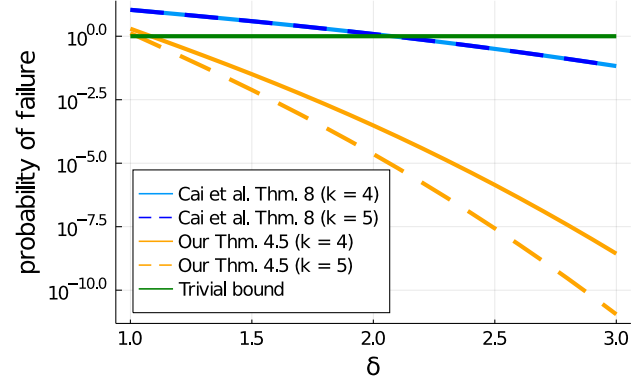


*Figure 1.* Comparing existing probabilistic bounds of (Cai and Wang, 2011) to Theorem 4.5 for a matrix with $m = 16$ columns, where $\delta = [1/2 - \mu_1(k)] \min_{i \in \mathcal{S}} |x_i|/\sigma$.

### 4.3. Backward Regression

We now state our main optimality result for the backward algorithm. The existence of such a result was already proved in Couvreur and Bresler (2000), though their bound is NP-hard to evaluate. In contrast, Theorem 4.6 reveals a proportional dependence of the tolerable noise on the smallest singular value of the matrix. While intuitive, this is to our knowledge the first result that makes this intuition precise.

**Theorem 4.6.** *Suppose* $\boldsymbol{\Phi}$ *has full column rank. Then Backward Regression recovers the support $\mathcal{S}$ of a $k$-sparse $m$-dimensional* $\mathbf{x}$ *in* $m - k$ *iterations if*

$$\frac{\sigma_{\min}(\boldsymbol{\Phi})}{\sqrt{2[2 - \sigma_{\min}(\boldsymbol{\Phi})^2]}} \min_{i \in \mathcal{S}} |x_i| > \|\boldsymbol{\epsilon}\|_2,$$

*where* $\sigma_{\min}(\boldsymbol{\Phi})$ *is the smallest singular value of* $\boldsymbol{\Phi}$.

Remarkably, the Backward Regression (BR) only requires linear independence of the columns, which is implied by $\mu_1(k) < 1$, but only applies to *determined* systems. Despite this necessarily stronger assumption, Thm. 4.6 is the first efficiently evaluable guarantee *for this case*. While the underdetermined case is most challenging, SBL is popularly applied to a kernel regression model for which the corresponding system is determined (see Sec. 5.4).

Still, because of this stronger assumption, it is important to connect the result for the backward algorithm with the already analyzed forward algorithm, the combination of which does apply to underdetermined systems. To this end, suppose a forward algorithm terminates with an arbitrary superset $\mathcal{A}$ of the true support, which is a weaker assumption than that of exact recovery. Then Theorem 4.6 applied to the submatrix $\boldsymbol{\Phi}_{\mathcal{A}}$ guarantees that the backward algorithm only deletes irrelevant features if the noise is not too large. The following corollary formalizes this idea.

**Corollary 4.7.** *Suppose $\Phi_\mathcal{A}$ has full column rank, $|\mathcal{A}| = k$, and $\mathcal{S} \subset \mathcal{A}$. Then Backward Regression recovers the correct support $\mathcal{S}$ in $|\mathcal{A}| - |\mathcal{S}|$ iterations, provided*

$$\sqrt{\frac{1 - \mu_1(k)}{2[1 + \mu_1(k)]}} \min_{i \in \mathcal{S}} |x_i| > \|\epsilon\|_2.$$

Note the striking similarity of the bounds in Corollary 4.7 and Theorem 4.4, though the former is stronger. As Couvreur and Bresler (2000) already established, the backward algorithm is not only capable of recovering the support of an *exactly* sparse vector, but in fact can solve the subset selection to optimality, provided the residual of the optimal solution is small enough. That is, for an arbitrary $\mathbf{y}$, not necessarily generated by $\Phi \mathbf{x}$ with a sparse $\mathbf{x}$, we have

**Theorem 4.8.** *Let $\mathbf{x}_k$ be the vector that achieves the smallest residual norm $\|\mathbf{y} - \Phi \mathbf{x}\|$ among all vectors $\mathbf{x}$ with $k$ or fewer non-zero elements. If the associated residual $\mathbf{r}_k \overset{def}{=} \mathbf{y} - \Phi \mathbf{x}_k$ satisfies the bound in Theorem 4.6 in place of $\epsilon$, Backward Regression recovers $\mathbf{x}_k$, or equivalently, solves the subset selection problem to optimality.*

In an earlier short paper, we provided a precursory result for the backward algorithm (Ament and Gomes, 2021). The full results we present herein provide tighter bounds and connect to the forward algorithm via Corollary 4.7. Similar to the results for the forward algorithm in the present work, the bound in Theorem 4.6 also converges to $\sqrt{2}$ as $\mu \to 0$, the provably tightest constant even for orthogonal matrices.

## 5. Numerical Experiments

The preceding results are powerful in their own right. Further, they guarantee that the backward stage of $RMP_0$ succeeds if the forward stage terminates with a superset of the true support, and the perturbation $\epsilon$ is not too large. Therefore, we would expect RMP to have increasing support recovery performance as the sampling ratio $n/m$ increases. The validation of this hypothesis is the goal of our first experiment. Subsequently, we benchmark the recovery performance of a number of algorithms on uncorrelated and correlated features with noise, a kernel regression task, and end with a discussion of the experiments.

### 5.1. Setup and Implementation

Beside RMP, we implemented OMP, FR, FoBa (Zhang, 2009), and the *steepest* coordinate ascent algorithm for SBL of Tipping and Faul (2003) (FSBL). We compare two versions of $RMP_0$ in the experiments: $RMP_0$, and $RMP_0+$. The former denotes Algorithm 2 with only one iteration of the outer loop, while the latter terminates once the support stabilizes. We also compare against BPDN via constrained $l_1$-norm minimization, and the SBL-based reweighted $l_1$-

norm algorithm of Wipf and Nagarajan (2008) (BP ARD). We implemented all algorithms in Julia (Bezanson et al., 2017), using the JuMP framework (Dunning et al., 2017) to model the BP-approaches as second-order cone programs, and solve them using ECOS (Domahidi et al., 2013) with default settings. All experiments were run on a workstation with an Intel Xeon CPU X5670 and 47 GB of memory.

For the synthetic experiments, the weights $\mathbf{x}$ are random $k$-sparse vectors with $\pm 1$ entries and the targets $\mathbf{y}$ were perturbed by random vectors distributed uniformly on the $10^{-2}$-hypersphere. For all algorithms, we input $\delta = 2\|\epsilon\|$ to simulate a small misspecification of the tolerance parameter that is likely to occur in practice. See also the supplementary materials for how $RMP_\sigma$ could be adapted to infer the tolerance parameter, which however is a non-convex problem. Note that the stopping criteria of some of the algorithms depend differently on $\delta$: For OMP and BPDN, it is a constraint on the residual norm, for $RMP_0$, and FoBa, it is a bound on the marginal improvement in residual norm, and for $RMP_\sigma$, we assign $\sigma \to \delta$ and note that the stopping criterion depends on the more complex expression of Lemma 3.1.

Since the BP cone-programs do not directly return sparse solutions, we determine their support by dropping all entries below $\|\epsilon\|/10$. In setting the threshold below the noise, we highlight that standard BPDN introduces a bias, while the ARD-based approach maintains the same global optimum as the $l_0$-minimization problem (Wipf and Nagarajan, 2008). We stress that BP leads to sparse solutions in theory (Tropp, 2006), but does not yield exactly sparse solutions using numerical LP-solvers, which terminate with many coefficients very close, but not equal to zero, necessitating the thresholding.

### 5.2. Phase-Transitions

First, we study the support recovery performance of Forward Regression (FR), $RMP_0$ and $RMP_\sigma$ as a function of the sampling ratio $n/m$ of the matrix $\Phi$ and the sparsity ratio $k/n$ of the weights $\mathbf{x}$. Figure 2 shows the empirical frequency of support recovery on column-normalized Gaussian random matrices with $m = 128$ columns. Every cell is an average over 256 independent realizations of the experiment.

In accordance with the results for Stepwise Regression in Section 4, the recovery performance the RMP algorithms increases with the sampling ratio, since the likelihood that the forward stage recovers a superset of the true support increases. In contrast, the success of the forward algorithm in isolation is chiefly dependent on the sparsity ratio and apparently independent of $n/m$. The ridges in the bottom left of each plot are due to rounding effects since we set $k = \lceil (k/n)(n/m)m \rceil$. Importantly, the performance of $RMP_0$ is virtually identical to $RMP_\sigma$ and constitutes a first experimental validation of our analysis in Section 3.
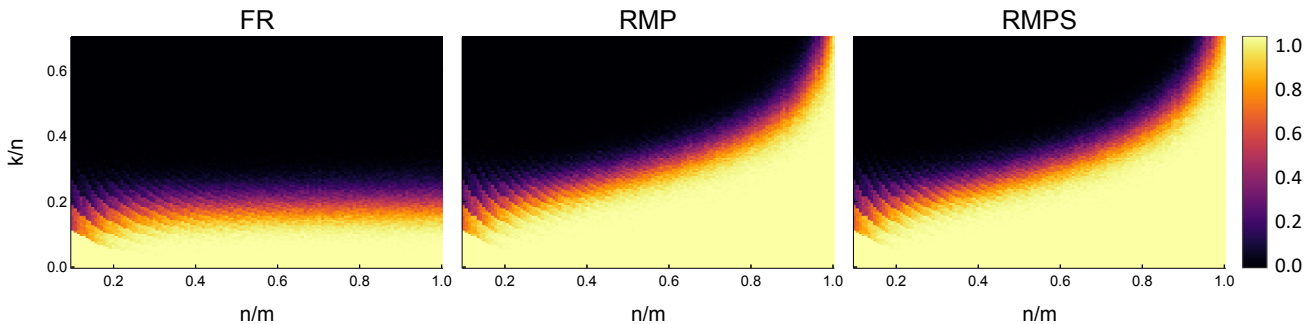
*Figure 2.* Empirical frequency of support recovery as a function of the sampling ratio $n/m$ and sparsity ratio $k/n$ for matrices with $m = 128$ and $\|\epsilon\|_2 = 10^{-2}$ for Forward Regression (left), $\text{RMP}_0$ (middle) and $\text{RMP}_\sigma$ (right).

*Table 1.* Recovery probability for *uncorrelated* features

| Type | Algorithm | Sparsity 12 | 16 | 20 | 24 |
|------|-----------|------|------|------|------|
|      | OMP       | 0.53 | 0.15 | 0.02 | 0.00 |
|      | FR        | 0.54 | 0.14 | 0.01 | 0.00 |
| MP   | FoBa      | **0.99** | **0.82** | **0.31** | **0.04** |
|      | $\text{RMP}_0$ | **0.99** | 0.80 | **0.31** | **0.04** |
|      | $\text{RMP}_0+$ | **0.99** | 0.80 | **0.31** | **0.04** |
| SBL  | FSBL      | **0.99** | 0.80 | **0.31** | **0.04** |
|      | $\text{RMP}_\sigma$ | **0.99** | 0.81 | **0.31** | **0.04** |
| BP   | BP        | 0.26 | 0.05 | 0.01 | 0.00 |
|      | BP ARD    | **1.00** | **1.00** | **0.97** | **0.70** |

*Table 2.* Recovery probability for *correlated* features

| Type | Algorithm | Sparsity 2 | 3 | 4 | 5 |
|------|-----------|------|------|------|------|
|      | OMP       | 0.00 | 0.00 | 0.00 | 0.00 |
|      | FR        | 0.04 | 0.01 | 0.00 | 0.00 |
| MP   | FoBa      | 0.71 | 0.46 | 0.28 | **0.17** |
|      | $\text{RMP}_0$ | **0.72** | 0.45 | 0.28 | 0.14 |
|      | $\text{RMP}_0+$ | **0.72** | **0.48** | **0.32** | **0.17** |
| SBL  | FSBL      | 0.76 | 0.54 | 0.38 | 0.26 |
|      | $\text{RMP}_\sigma$ | **0.81** | **0.58** | **0.45** | **0.30** |
| BP   | BP        | 0.02 | 0.00 | 0.00 | 0.00 |
|      | BP ARD    | **0.96** | **0.91** | **0.83** | **0.70** |

## 5.3. Support Recovery with Noise

The following experiments have been established in the literature as a widespread proxy for performance on a variety of tasks, and thus allow for comparison to other reported results, for example those of Wipf and Rao (2004), Candes et al. (2005; 2008), and He et al. (2017). In particular, we record the empirical frequency of support recovery of a large set of algorithms as a function of the sparsity level $k$ for two types of matrices. First, Gaussian random matrices and second, matrices generated as $\mathbf{\Phi} = \sum_p \frac{1}{p^2} \mathbf{u}\mathbf{v}^*$, where $\mathbf{u}, \mathbf{v}$ have standard normal entries, inspired by the experiments in Xin et al. (2016). The two types of matrices exhibit low and high column correlations, respectively. We used matrices of size 64 by 128 and $l_2$-normalized the columns. The results reported in Tables 1 and 2 are averages over 1024 independent realizations with a 95% confidence interval below 0.02.

BP ARD demonstrates the best recovery performance among all tested methods, followed by $\text{RMP}_\sigma$. Importantly, $\text{RMP}_\sigma$ exhibits similar performance to $\text{RMP}_0$ and virtually identical for uncorrelated features. The differences between $\text{RMP}_0$ and FoBa are marginal and plausibly attributable to statistical fluctuations. However, the differences in recovery

performance of $\text{RMP}_\sigma$ and FSBL is statistically significant for correlated features. BP performs poorly since our experiment was designed to expose that it does not preserve the same optimum as $l_0$-minimization.

Another notable approach is that of Koyejo et al. (2014), who proposed a greedy information-projection algorithm with applications to sparse estimation problems. In our study, the algorithm performed well on Gaussian matrices but deteriorated similarly to OMP and FR on coherent matrices, which is expected since it solely takes forward steps.

Figure 3 shows runtimes of the algorithms for matrices of increasing sizes. As $m$ increases, we kept the ratios $n/m = 1/2$ and $k/m = 1/4$. BP ARD is most time-consuming. RMP and $\text{RMP}_0$ are on average two orders of magnitude faster. The main performance difference between $\text{RMP}_\sigma$ and $\text{RMP}_0$ comes from $\text{RMP}_\sigma$'s $\gamma_i$-update, which it can execute many times before converging. FoBa should in principle scale comparably to $\text{RMP}_0$, but doesn't, as we chose not to let it take advantage of the efficient backward updates discussed in Reeves (1999), but not mentioned in Zhang (2009), which highlights their importance. Last, a limitation of the timings for BP are that we used generic LP solvers, while specialized algorithms exist (Beck and
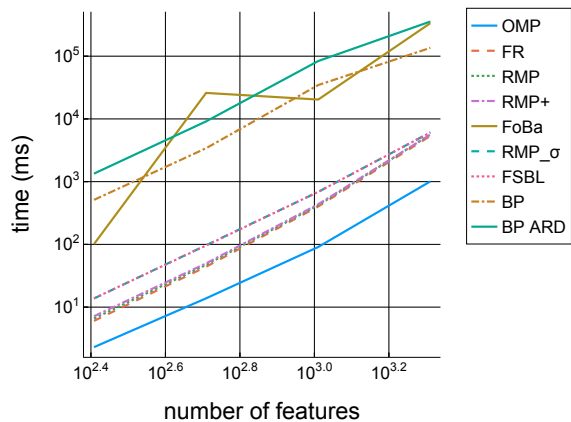
*Figure 3.* Timings for several algorithms on matrices with a fixed sampling $(1/2)$ and sparsity ratio $(1/4)$.



*Figure 4.* Test RMSE for a kernel regression model on the UCI Boston housing data as a function of achieved sparsity.

Teboulle, 2009; Perez et al., 2019) that can accelerate BP approaches. On the other hand, keeping the sparsity ratio $k/n$ fixed while growing $m$ is to the detriment of the greedy algorithms, which would need much fewer iterations if the sparsity level $k$ was fixed instead. In all, the timings are not designed to be fair but to illustrate the inevitable trade-off between performance and efficiency.

### 5.4. Sparse Kernel Regression

Following Tipping (2001), we apply the SBL-related algorithms to a kernel regression model. In particular, given inputs $\{\mathbf{x}_i\}$, we assume the responses are generated according to $y \sim \mathcal{N}(f(\mathbf{x}), \sigma^2)$, where

$$f(\mathbf{x}) \overset{\text{def}}{=} \sum_i k(\mathbf{x}, \mathbf{x}_i) w_i, \tag{5.1}$$

where $k$ is the Matérn-3/2 kernel and weights $w_i$. Given a training set, we optimize $w_i$ using the SBL-related algorithms, and evaluate on a test set using equation (5.1). Figure 4 shows the mean test error as a function of sparsity on the UCI Boston housing data (Dua and Graff, 2017), which contains 506 data points. The results are averaged over 4608 sparsity-error values for each algorithm, generated by evaluations for different tolerance parameters $\delta$ (i.e. $\sigma$) and random 75-25 train-test splits. We conjecture that (F)SBL exhibits larger errors as it does not directly minimize squared errors, but instead the marginal likelihood. While FR can be competitive for highly sparse solutions, it is not as effective as RMP, RMP+ and FoBa, which achieve the best sparsity-error trade-off throughout all sparsity levels.

## 6. Conclusion

We proposed Relevance Matching Pursuit, a coordinate ascent algorithm for SBL whose analysis reveals a surprising connection to Stepwise Regression. The limiting algorithm
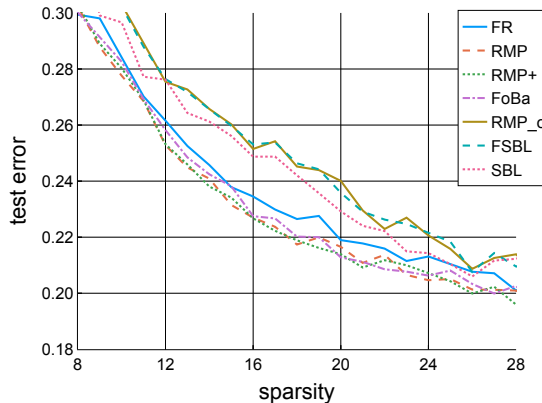
$\text{RMP}_0$ closely tracks the performance of $\text{RMP}_\sigma$ in our empirical evaluation, and is yet remarkably simple. We provided novel theoretical insights for Stepwise Regression, among them, an efficiently computable guarantee for the backward algorithm. Our results further provide theoretical justification to practitioners using Stepwise Regression in one of many statistics packages, prominently in the widely-used SAS. Finally, we hope these insights contribute to providing clarity in this vast space of the literature, and inspire further research on powerful sparsity-inducing algorithms.

## References

Ament, S. and Gomes, C. (2021). On the optimality of backward regression: Sparse recovery and subset selection. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5599–5603.

Andrle, M., Rebollo-Neira, L., and Sagianos, E. (2004). Backward-optimized orthogonal matching pursuit approach. *IEEE Signal Processing Letters*, 11(9):705–708.

Bach, F. R. (2008). Consistency of the group lasso and multiple kernel learning. *Journal of Machine Learning Research*, 9(Jun):1179–1225.

Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202.

Ben-Haim, Z., Eldar, Y. C., and Elad, M. (2010). Coherence-based performance guarantees for estimating a sparse vector under random noise. *IEEE Transactions on Signal Processing*, 58(10):5030–5043.

Bezanson, J., Edelman, A., Karpinski, S., and Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM review*, 59(1):65–98.

Borboudakis, G. and Tsamardinos, I. (2019). Forward-backward selection with early dropping. *The Journal of Machine Learning Research*, 20(1):276–314.

Bruckstein, A. M., Donoho, D. L., and Elad, M. (2009). From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM review*, 51(1):34–81.

Cai, T. T. and Wang, L. (2011). Orthogonal matching pursuit for sparse signal recovery with noise. *IEEE Transactions on Information Theory*, 57(7):4680–4688.

Candes, E., Rudelson, M., Tao, T., and Vershynin, R. (2005). Error correction via linear programming. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, pages 668–681. IEEE.

Candès, E. J., Romberg, J., and Tao, T. (2006). Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509.

Candes, E. J., Wakin, M. B., and Boyd, S. P. (2008). Enhancing sparsity by reweighted l1 minimization. *Journal of Fourier analysis and applications*, 14(5-6):877–905.

Cawley, G. C., Talbot, N. L., and Girolami, M. (2007). Sparse multinomial logistic regression via bayesian l1 regularisation. In *Advances in neural information processing systems*, pages 209–216.

Chen, S. S., Donoho, D. L., and Saunders, M. A. (2001). Atomic decomposition by basis pursuit. *SIAM review*, 43(1):129–159.

Combettes, C. and Pokutta, S. (2019). Blended matching pursuit. In *Advances in Neural Information Processing Systems*, pages 2042–2052.

Cotter, S. F., Adler, R., Rao, R. D., and Kreutz-Delgado, K. (1999). Forward sequential algorithms for best basis selection. *IEE Proceedings - Vision, Image and Signal Processing*, 146(5):235–244.

Couvreur, C. and Bresler, Y. (2000). On the optimality of the backward greedy algorithm for the subset selection problem. *SIAM Journal on Matrix Analysis and Applications*, 21(3):797–808.

Das, A. and Kempe, D. (2018). Approximate submodularity and its applications: subset selection, sparse approximation and dictionary selection. *The Journal of Machine Learning Research*, 19(1):74–107.

Davis, G., Mallat, S., and Avellaneda, M. (1997). Adaptive greedy approximations. *Constructive approximation*, 13(1):57–98.

Domahidi, A., Chu, E., and Boyd, S. (2013). ECOS: An SOCP solver for embedded systems. In *European Control Conference (ECC)*, pages 3071–3076.

Donoho, D. L. and Elad, M. (2006). On the stability of the basis pursuit in the presence of noise. *Signal Processing*, 86(3):511 – 532. Sparse Approximations in Signal and Image Processing.

Dua, D. and Graff, C. (2017). UCI machine learning repository.

Dunning, I., Huchette, J., and Lubin, M. (2017). Jump: A modeling language for mathematical optimization. *SIAM Review*, 59(2):295–320.

Elenberg, E. R., Khanna, R., Dimakis, A. G., Negahban, S., et al. (2018). Restricted strong convexity implies weak submodularity. *The Annals of Statistics*, 46(6B):3539–3568.

Faul, A. C. and Tipping, M. E. (2002). Analysis of sparse bayesian learning. In *Advances in neural information processing systems*, pages 383–389.

Gorodnitsky, I. F. and Rao, B. D. (1997). Sparse signal reconstruction from limited data using focuss: a re-weighted minimum norm algorithm. *IEEE Transactions on Signal Processing*, 45(3):600–616.

Haupt, J., Bajwa, W. U., Rabbat, M., and Nowak, R. (2008). Compressed sensing for networked data. *IEEE Signal Processing Magazine*, 25(2):92–101.

He, H., Xin, B., Ikehata, S., and Wipf, D. (2017). From bayesian sparsity to gated recurrent nets. In *Advances in Neural Information Processing Systems*, pages 5554–5564.

Koyejo, O. O., Khanna, R., Ghosh, J., and Poldrack, R. (2014). On prior distributions and approximate inference for structured variables. *Advances in Neural Information Processing Systems*, 27:676–684.

Locatello, F., Raj, A., Karimireddy, S. P., Raetsch, G., Schölkopf, B., Stich, S., and Jaggi, M. (2018). On matching pursuit and coordinate descent. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3198–3207, Stockholmsmässan, Stockholm Sweden. PMLR.

Lustig, M., Donoho, D., and Pauly, J. M. (2007). Sparse mri: The application of compressed sensing for rapid mr imaging. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 58(6):1182–1195.

MacKay, D. J. (1992a). Bayesian interpolation. *Neural computation*, 4(3):415–447.

MacKay, D. J. (1992b). The evidence framework applied to classification networks. *Neural computation*, 4(5):720–736.

Mallat, S. G. and Zhifeng Zhang (1993). Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415.

Miller, A. (2002). *Subset selection in regression*. CRC Press.

Perez, G., Barlaud, M., Fillatre, L., and Régin, J.-C. (2019). A filtered bucket-clustering method for projection onto the simplex and the $\ell_1$ ball. *Mathematical Programming*.

Rangan, S. and Fletcher, A. K. (2009). Orthogonal matching pursuit from noisy random measurements: A new analysis. In *Advances in Neural Information Processing Systems*, pages 540–548.

Rao, N., Shah, P., and Wright, S. (2015). Forward–backward greedy algorithms for atomic norm regularization. *IEEE Transactions on Signal Processing*, 63(21):5798–5811.

Rebollo-Neira, L. and Lowe, D. (2002). Optimized orthogonal matching pursuit approach. *IEEE signal processing Letters*, 9(4):137–140.

Reeves, S. J. (1999). An efficient implementation of the backward greedy algorithm for sparse signal reconstruction. *IEEE Signal Processing Letters*, 6(10):266–268.

Soussen, C., Gribonval, R., Idier, J., and Herzet, C. (2013). Joint k-step analysis of orthogonal matching pursuit and orthogonal least squares. *IEEE Transactions on Information Theory*, 59(5):3158–3174.

Szameit, A., Shechtman, Y., Osherovich, E., Bullkich, E., Sidorenko, P., Dana, H., Steiner, S., Kley, E. B., Gazit, S., Cohen-Hyams, T., et al. (2012). Sparsity-based single-shot subwavelength coherent diffractive imaging. *Nature materials*, 11(5):455–459.

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

Tibshirani, R. J. (2015). A general framework for fast stagewise algorithms. *The Journal of Machine Learning Research*, 16(1):2543–2588.

Tipping, M. E. (2001). Sparse bayesian learning and the relevance vector machine. *Journal of machine learning research*, 1(Jun):211–244.

Tipping, M. E. and Faul, A. C. (2003). Fast marginal likelihood maximisation for sparse bayesian models. In *AISTATS*.

Tropp, J. A. (2004). Greed is good: algorithmic results for sparse approximation. *IEEE Transactions on Information Theory*, 50(10):2231–2242.

Tropp, J. A. (2006). Just relax: convex programming methods for identifying sparse signals in noise. *IEEE Transactions on Information Theory*, 52(3):1030–1051.

Tropp, J. A. and Gilbert, A. C. (2007). Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 53(12):4655–4666.

Wipf, D. P. (2011). Sparse estimation with structured dictionaries. In Shawe-Taylor, J., Zemel, R. S., Bartlett, P. L., Pereira, F., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 24*, pages 2016–2024. Curran Associates, Inc.

Wipf, D. P. and Nagarajan, S. S. (2008). A new view of automatic relevance determination. In Platt, J. C., Koller, D., Singer, Y., and Roweis, S. T., editors, *Advances in Neural Information Processing Systems 20*, pages 1625–1632. Curran Associates, Inc.

Wipf, D. P. and Nagarajan, S. S. (2009). Sparse estimation using general likelihoods and non-factorial priors. In *Advances in Neural Information Processing Systems*, pages 2071–2079.

Wipf, D. P. and Rao, B. D. (2004). Sparse bayesian learning for basis selection. *IEEE Transactions on Signal processing*, 52(8):2153–2164.

Wipf, D. P. and Rao, B. D. (2005). l0-norm minimization for basis selection. In *Advances in Neural Information Processing Systems*, pages 1513–1520.

Xin, B., Wang, Y., Gao, W., Wipf, D., and Wang, B. (2016). Maximal sparsity with deep networks? In *Advances in Neural Information Processing Systems*, pages 4340–4348.

Zhang, T. (2009). Adaptive forward-backward greedy algorithm for sparse learning with linear models. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 21*, pages 1921–1928. Curran Associates, Inc.

Zhao, P. and Yu, B. (2007). Stagewise lasso. *Journal of Machine Learning Research*, 8(Dec):2701–2726.