A Weighted Average of Sparse Representations is Better than the Sparsest One Alone*

Michael Elad and Irad Yavneh
Department of Computer Science
Technion–Israel Institute of Technology
Technion City, Haifa 32000, Israel
Email: [elad,irad]@cs.technion.ac.il

Abstract

Cleaning of noise from signals is a classical and long-studied problem in signal processing. Algorithms for this task necessarily rely on an a-priori knowledge about the signal characteristics, along with information about the noise properties. For signals that admit sparse representations over a known dictionary, a commonly used denoising technique is to seek the sparsest representation that synthesizes a signal close enough to the corrupted one. As this problem is too complex in general, approximation methods, such as greedy pursuit algorithms, are often employed.

In this line of reasoning, we are led to believe that detection of the sparsest representation is key in the success of the denoising goal. Does this mean that other competitive and slightly inferior sparse representations are meaningless? Suppose we are served with a group of competing sparse representations, each claiming to explain the signal differently. Can those be fused somehow to lead to a better result? Surprisingly, the answer to this question is positive; merging these representations can form a more accurate, yet dense, estimate of the original signal even when the latter is known to be sparse.

In this paper we demonstrate this behavior, propose a practical way to generate such a collection of representations by randomizing the Orthogonal Matching Pursuit (OMP) algorithm, and produce a clear analytical justification for the superiority of the associated Randomized OMP (RandOMP) algorithm. We show that while the Maximum a-posterior Probability (MAP) estimator aims to find and use the sparsest representation, the Minimum Mean-Squared-Error (MMSE) estimator leads to a fusion of representations to form its result. Thus, working with an appropriate mixture of candidate representations, we are surpassing the MAP and tending towards the MMSE estimate, and thereby getting a far more accurate estimation, especially at medium and low SNR.

*This research was supported by the Center for Security Science and Technology – Technion.

Dagstuhl Seminar Proceedings 08492
Structured Decompositions and Efficient Algorithms
http://drops.dagstuhl.de/opus/volltexte/2009/1882

1 Introduction

1.1 Denoising in General

Cleaning of additive noise from signals is a classical and long-studied problem in signal processing. This task, known as denoising, considers a given measurement signal $\mathbf{y} \in \mathbb{R}^n$ obtained from the clean signal $\mathbf{x} \in \mathbb{R}^n$ by a contamination of the form $\mathbf{y} = \mathbf{x} + \mathbf{v}$. In this paper we shall restrict our discussion to noise vectors $\mathbf{v} \in \mathbb{R}^n$, assumed to be zero mean i.i.d. Gaussian, with entries drawn at random from the normal distribution $\mathcal{N}(0, \sigma)$. The denoising goal is to recover $\mathbf{x}$ from $\mathbf{y}$.

In order to design an effective denoising algorithm, we must have at our disposal two pieces of information: The first is a knowledge about the noise characteristics, as described above. Along with it, we must also introduce some knowledge about the class of signals that $\mathbf{x}$ belongs to. Only with these two can one design a scheme to decompose $\mathbf{y}$ into its original components, $\mathbf{x}$ and $\mathbf{v}$. There are numerous algorithms for denoising, as there are numerous ways to describe the a-priori knowledge about the signal characteristics. Among these, a recently emerging model for signals that attracts much attention is one that relies on sparse and redundant representations [18, 2]. This model will be the focus of the work presented here.

1.2 Sparse and Redundant Representations

A signal $\mathbf{x}$ is said to have a sparse representation over a known dictionary $\mathbf{D} \in \mathbb{R}^{n \times m}$ (we typically assume that $m > n$, implying that this is a redundant representation), if there exists a sparse vector $\alpha \in \mathbb{R}^m$ such that $\mathbf{x} = \mathbf{D}\alpha$. The vector α is said to be the representation of $\mathbf{x}$. Referring to the columns of $\mathbf{D}$ as prototype signals or *atoms*, α describes how to construct $\mathbf{x}$ from a few such atoms by a linear combination. The representation is sparse – the number of non-zeros in it, $k = \|\alpha\|_0$, is expected to be much smaller than n. Also, this is a redundant representation – it is longer than the original signal it represents. In this paper we consider the family of signals that admit sparse representations over a known dictionary $\mathbf{D}$ and discuss ways to denoise them. Note that at this stage we do not provide a full and exact definition of this signal family (e.g., we do not specify how the representations are generated) – such a definition will follow at a later stage in the paper, where a rigorous analysis is pursued.

Assuming that $\mathbf{x} = \mathbf{D}\alpha$ with a sparse representation α, how can we denoise a corrupted version of it, $\mathbf{y}$? A commonly used denoising technique is to seek the sparsest representation that synthesizes a signal close enough to the corrupted one [2, 9, 10, 11, 12, 13, 16, 17, 19, 24, 25]. Put formally, one way to define our task is given by

$$\hat{\alpha} = \arg\min_{\alpha} \ \|\alpha\|_0 + \lambda\|\mathbf{y} - \mathbf{D}\alpha\|_2^2. \tag{1}$$

The first penalty directs the minimization task towards the sparsest possible representation, exploiting our a-priori knowledge about the formation of the signal. The second penalty manifests our knowledge about the noise being white and Gaussian. This overall expression is inversely proportional to the posterior probability, $p(\alpha|\mathbf{y})$, and as such, its minimization forms the Maximum A-posteriori Probability (MAP) estimate [2]. The parameter λ should be chosen based on σ and the fine details that model how the signals' representations are generated. As remarked above, there are other ways to formulate our goal – for example, we could replace one of the penalties with a constraint, if their size is known. Once $\hat{\alpha}$ is found, the denoising result is obtained by $\hat{\mathbf{x}} = \mathbf{D}\hat{\alpha}$.

The problem posed in Equation (1) is too complex in general, requiring a combinatorial search that explores all possible sparse supports [20]. Approximation methods are therefore often employed, with the understanding that their result may deviate from the true solution. One such approximation technique is the Orthogonal Matching Pursuit (OMP), a greedy algorithm that accumulates one atom at a time in forming $\hat{\alpha}$, aiming at each step to minimize the representation error $\|\mathbf{y} - \mathbf{D}\alpha\|_2^2$ [2, 3, 5, 6, 19, 21, 23]. When this error falls below some predetermined threshold, or when the number of atoms reaches a destination value, this process stops. While crude, this technique works very fast and can guarantee near-optimal results in some cases.

How good is the denoising obtained by the above approach? Past work provides some preliminary, both theoretical and empirical, answers to this and related questions [2, 8, 9, 10, 12, 13, 16, 17, 24, 25]. Most of this work concentrates on the accuracy with which one can approximate the true representation (rather than the signal itself), adopting a worst-case point of view. Indeed, the only work that targets the theoretical question of denoising performance head-on is reported in [12, 13], providing asymptotic assessments of the denoising performance for very low and very high noise powers, assuming that the original combinatorial problem can be solved exactly.

1.3 This Paper

In the above line of reasoning, we are led to believe that detection of the sparsest representation is key in the success of the denoising goal. Does this mean that other, competitive yet slightly inferior, sparse representations are meaningless? This question is critical, especially due to the often encountered cases where OMP (and other approximation algorithms) fails to find the truly sparsest representation.

Furthermore, past analysis of approximation algorithms indicates that a measure of coherence of the dictionary can predict the tendency of these methods to fail, and this is especially true when the sparsity of the target representation is rather mild [2, 9, 10, 16, 17, 23, 24]. The coherence of a dictionary is defined through the worst pair of atoms exhibiting maximal correlation. If this value is high, it implies that these atoms tend to confuse and mislead any solver. Thus, non-coherent dictionaries necessarily lead to wrong solutions in many cases; are these to be considered as complete failures?

In fact, we should ask a more general question: Suppose we are served with a group of competing sparse representations, each claiming to explain the signal differently. Can those be fused somehow to lead to a better result? Surprisingly, the answer to this question is positive; these representations can definitely be merged to form a more accurate estimate of the original signal. This means that even when the dictionary is non-coherent, one can obtain a reasonable denoising, by exploiting this mixture of representations. Why is this true? How can we exploit this? In this paper we aim to show that *there is life beyond the sparsest representation*. More specifically:

- We propose a practical way to generate a set of sparse representations for a given signal by randomizing the OMP algorithm. This technique samples from the set of sparse solutions that approximate $\mathbf{D}\alpha = \mathbf{y}$;
- We demonstrate the gain in using such a set of representations through a preliminary experiment that fuses these results by a plain averaging; and most important of all,
- We provide a clear explanation for the origin of this strange phenomenon. We develop analytical expressions for the MAP and the Minimum Mean-Squared-Error (MMSE) estimators for the model discussed, and show that while the MAP estimator aims to find and use the sparsest representation, the MMSE estimator fuses a collection of representations to form its result. Thus, working with a set of candidate representations, we are surpassing the MAP and tending towards the MMSE estimate, and thereby getting a more accurate estimation.
- Based on the above rigorous analysis we also provide clear expressions that predict the MSE of the various estimators, and thus obtain a good prediction for the denoising performance of the OMP and its randomized version.

This paper is organized as follows. In Section 2 we build a case for the use of several sparse representations, leaning on intuition and some preliminary experiments that suggests that this idea is worth a closer look. Section 3 contains the analytic part of this paper, which develops the MAP and the MMSE exact estimators and their expected errors, showing how they relate to the use of several representations. We conclude in Section 4 by highlighting the main contribution of this paper, and drawing attention to important open questions to which our analysis points.

2 The Key Idea – A Mixture of Representations

In this section we build a case for the use of several sparse representations. First, we motivate this by drawing intuition from example-based modelling, where several approximations of the corrupted data are used to denoise it. Armed with the desire to generate a set of sparse representations, we present the Randomized Orthogonal Matching Pursuit (RandOMP) algorithm that generates a group of competitive representations for a given signal. Finally, we show that this concept works quite well in practice and provides a benefit over the use of a single representation. In the next section we provide a more rigorous explanation of this phenomenon.

2.1 Why Bother? Some Hand-Waving

Why should we consider the use of more than just one representation? One possible reason is the fact that a single representation we happen to find may be the wrong one, due to the tendency of pursuit techniques to fail from time to time. This suggests that one should possibly seek several representations and choose the best one in some sense, assuming that this is possible. In doing so, we are still in the realm of the MAP estimator, aiming to improve the OMP chances to approximate better the desired single representation solution of Equation (1). As we shall see next, we aim to achieve far more than just that when using a set of representations.

A second and more relevant reason to consider a mixture of representations is an intriguing relation between our model and direct example-based techniques. Our model assumes that signals in the family we handle can be represented as multiplications of the dictionary $\mathbf{D}$ by sparse vectors. What if we allow $\mathbf{D}$ to be more redundant by letting the number of its columns m to grow? The general tendency we expect to see is a decrease in the number of non-zeros required in the representations, that is, they become sparser. At the extreme, when the dictionary contains $m \rightarrow \infty$ columns, reflecting many possible instances of signals, the required sparsity should tend towards $k = 1$, since almost every original signal is available as an atom (possibly up to a scale).

This extreme case is exactly the one practiced in direct example-based methods [1, 14, 15, 22, 7]. Suppose we are given many instances of noisy signals $\{\mathbf{y}_i\}_{i=1}^{N}$. We refer to those as our training data, and form a dictionary $\mathbf{D}$ by simply concatenating them as our atoms. When aiming to denoise a newly obtained signal $\mathbf{y}$, an example-based denoising algorithm suggests that we seek in $\mathbf{D}$ a set of nearby atoms. Each such neighbor found is an extremely sparse representation with cardinality 1, and with the coefficient being 1 as well. We may consider a slightly more general search for neighbors that allows for scaling, which enriches the span of the dictionary signal set.

Using one neighbor atom only as our suggested solution implies that we replace the noise in $\mathbf{y}$ by the noise in this atom, rather than cleaning it, which is of-course useless. Suppose that in the set of neighbors chosen we have managed to find instances of the same original signal with different realizations of noise. In such a case, averaging these solutions leads to an attenuation of the additive noise. Thus, a collection of very sparse representations joins forces to produce a better estimate.

If the above is true for the extreme case, why should it not be relevant for the lower redundancy case as well? The rationale is that each sparse representation found recommends its own way of denoising, and their fusion may lead to a better overall noise removal effect. Could this be true? In order to explore this idea, we must start by finding a practical way to generate a set of candidate representations, which is our next topic.

2.2 RandOMP

Here is a clear definition of our goal: Given a dictionary $\mathbf{D}$ and a signal $\mathbf{y}$, we aim to find a group of sparse representations α_i, such that each satisfies $\|\mathbf{D}\alpha_i - \mathbf{y}\|_2 \leq T$, and all aim to be as sparse as possible yet different from each other. Alternatively, we may desire to find this set such that each has the same pre-specified number of non-zeros, k, and all aim to get residuals, $\|\mathbf{D}\alpha_i - \mathbf{y}\|_2$, that are as low as possible. We shall work in this section with the former option, since it is more relevant to denoising in cases when the noise power is fixed and known, as in the case studied here.

Figure 1 presents the OMP algorithm with a stopping rule that depends on the residual energy [2, 3, 5, 6, 19, 21]. At each iteration, the set $\{\epsilon(j)\}_{j=1}^{m}$ is computed, whose jth term indicates the error that would remain if atom j is added to the current solution. The atom chosen is the one yielding the smallest error. Note that if there are several candidate atoms that show a relatively small residual energy, the smallest one is chosen regardless of the proximity of the others to it. This brings us naturally to the randomization approach we intend to apply.

In order to use this algorithm to generate a set of (probably) distinct sparse representations, all that we need to do is to randomize the choice of the next atom to be added. For example, rather than choose the atom that minimizes $\epsilon(j)$, we can choose it at random with a probability inversely proportional to these error values, or proportional to $|\mathbf{d}_j^T\mathbf{r}^{k-1}|^2/\|\mathbf{d}_j\|_2^2$ (since $\epsilon(j) = \|\mathbf{r}^{k-1}\|_2^2 - |\mathbf{d}_j^T\mathbf{r}^{k-1}|^2/\|\mathbf{d}_j\|_2^2$). For reasons to be explained in detail in the next section, the specific way we choose to draw the next atom is with probability linearly proportional to $\exp\{\frac{c^2}{2\sigma^2} \cdot |\mathbf{d}_j^T\mathbf{r}^{k-1}|^2/\|\mathbf{d}_j\|_2^2\}$, with

$$c^2 = \frac{\sigma_x^2}{\sigma_x^2 + \sigma^2}\,. \tag{2}$$

Here σ_x is the variance of the non-zero entries of the representation of the original signal.

By running this algorithm J_0 times, this randomization leads to J_0 solutions $\{\alpha_i\}_{i=1}^{J_0}$, as desired. Common to all these representations are the facts that (i) their representation error

Task: Approximate the solution of $\min_\alpha \ \|\alpha\|_0$ s.t. $\|\mathbf{D}\alpha - \mathbf{y}\|_2 \leq T$.

Parameters: The matrix $\mathbf{D}$, the signal $\mathbf{y}$, and the error threshold T.

Initialization: Initialize $k = 0$, and set

- The initial solution $\alpha^0 = 0$.
- The initial residual $\mathbf{r}^0 = \mathbf{y} - \mathbf{D}\alpha^0 = \mathbf{y}$.
- The initial solution support $\mathcal{S}^0 = Support\{\alpha^0\} = \emptyset$.

Main Iteration: Increment k by 1 and perform the following steps:

- **Sweep:** Compute the errors $\epsilon(j) = \min_{z_j} \|\mathbf{d}_j z_j - \mathbf{r}^{k-1}\|_2^2$ for all j using the optimal choice $z_j^* = \mathbf{d}_j^T \mathbf{r}^{k-1} / \|\mathbf{d}_j\|_2^2$.
- **Update Support:** Find j_0–the minimizer of $\epsilon(j)$, and update the support, $\mathcal{S}^k = \mathcal{S}^{k-1} \cup \{j_0\}$.
- **Update Solution:** Compute α^k, the minimizer of $\|\mathbf{D}\alpha - \mathbf{y}\|_2^2$ subject to $Support\{\alpha\} = \mathcal{S}^k$.
- **Update Residual:** Compute $\mathbf{r}^k = \mathbf{y} - \mathbf{D}\alpha^k$.
- **Stopping Rule:** If $\|\mathbf{r}^k\|_2 < T$, stop. Otherwise, apply another iteration.

Output: The proposed solution is α^k obtained after k iterations.

Figure 1: The OMP – a greedy algorithm.

$\|\mathbf{D}\alpha_i - \mathbf{y}\|_2$ is below T due to the stopping rule enforced; and (ii) all of them tend to be relatively sparse due to the greedy nature of this algorithm that aims to decrease the residual energy, giving preference to those atoms that serve this goal better. Figure 2 presents this algorithm.

Task: Approximate the solution of $\min_\alpha \ \|\alpha\|_0$ s.t. $\|\mathbf{D}\alpha - \mathbf{y}\|_2 \leq T$.

Parameters: The matrix $\mathbf{D}$, the signal $\mathbf{y}$, and the error threshold T.

Initialization: Initialize $k = 0$, and set

- The initial solution $\alpha^0 = 0$.
- The initial residual $\mathbf{r}^0 = \mathbf{y} - \mathbf{D}\alpha^0 = \mathbf{y}$.
- The initial solution support $\mathcal{S}^0 = \mathit{Support}\{\alpha^0\} = \emptyset$.

Main Iteration: Increment k by 1 and perform the following steps:

- **Sweep:** Compute the errors $\epsilon(j) = \min_{z_j} \|\mathbf{d}_j z_j - \mathbf{r}^{k-1}\|_2^2$ for all j using the optimal choice $z_j^* = \mathbf{d}_j^T \mathbf{r}^{k-1} / \|\mathbf{d}_j\|_2^2$.
- **Update Support:** Draw j_0 at random with probability proportional to $\exp\{\frac{c^2}{2\sigma^2} \cdot |\mathbf{d}_j^T \mathbf{r}^{k-1}|^2 / \|\mathbf{d}_j\|_2^2\}$, and update the support, $\mathcal{S}^k = \mathcal{S}^{k-1} \cup \{j_0\}$.
- **Update Solution:** Compute α^k, the minimizer of $\|\mathbf{D}\alpha - \mathbf{y}\|_2^2$ subject to $\mathit{Support}\{\alpha\} = \mathcal{S}^k$.
- **Update Residual:** Compute $\mathbf{r}^k = \mathbf{y} - \mathbf{D}\alpha^k$.
- **Stopping Rule:** If $\|\mathbf{r}^k\|_2 < T$, stop. Otherwise, apply another iteration.

Output: The proposed solution is α^k obtained after k iterations.

Figure 2: RandOMP – generating random sparse representations.

We demonstrate the behavior of this algorithm by performing the following simple test. First, we build a random dictionary $\mathbf{D}$ of size 100×200 by drawing its entries at random from the normal distribution $\mathcal{N}(0,1)$, and then ℓ_2 normalizing its columns. We then generate a random representation α_0 with $k = 10$ non-zeros chosen at random and with values drawn from $\mathcal{N}(0, \sigma_x)$ with $\sigma_x = 1$. The clean signal is obtained by $\mathbf{x} = \mathbf{D}\alpha$, and its noisy version $\mathbf{y}$ is obtained by adding white Gaussian noise with entries drawn from $\mathcal{N}(0, \sigma)$ with $\sigma = 1$

as well.

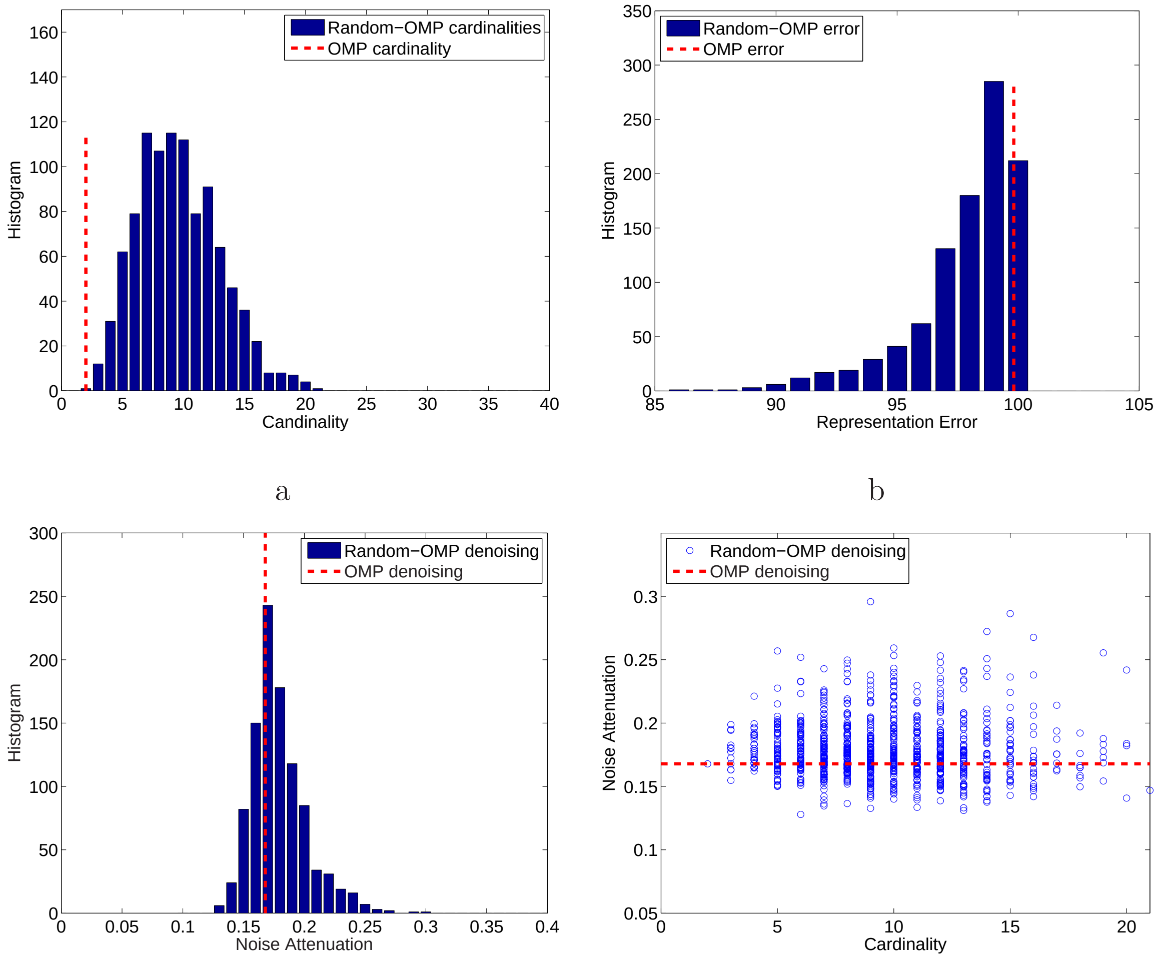
Armed with the dictionary $\mathbf{D}$, the corrupted signal $\mathbf{y}$ and the noise threshold $T = n\sigma^2 = 100$, we first run the plain OMP, and obtain a representation α^{OMP} with cardinality $k = 2$, and with a representation error $\|\mathbf{D}\alpha^{OMP} - \mathbf{y}\| = 99.82$. We can also check the denoising effect obtained by evaluating the expression $\|\mathbf{D}\alpha^{OMP} - \mathbf{x}\|_2^2/\|\mathbf{y} - \mathbf{x}\|_2^2$. The value obtained is 0.168, suggesting that the noise was indeed attenuated nicely by a factor close to 6.

We proceed by running RandOMP $J_0 = 1,000$ times, obtaining $1,000$ candidate representations $\{\alpha_j^{RandOMP}\}_{j=1}^{1,000}$. Among these, there are 999 distinct ones, but we allow repetitions. Figure 3-a presents a histogram of the cardinalities of the results. As can be seen, all the representations obtained are relatively sparse, with cardinalities in the range $[2, 21]$, indicating that the OMP representation is the sparsest. Figure 3-b presents a histogram of the representation errors of the results obtained. As can be seen, all the representations give an error slightly smaller than the threshold chosen, $T = 100$.

We also assess the denoising performance of *each* of these representations as done above for the OMP result. Figure 3-c shows a histogram of the denoising factor $\|\mathbf{D}\alpha_j^{RandOMP} - \mathbf{x}\|_2^2/\|\mathbf{y} - \mathbf{x}\|_2^2$. The results are in the range $[0.128, 0.296]$, with roughly half surpassing the OMP performance and the other half performing more poorly. However, can we detect the better performing representations? Figure 3-d shows the relation between the representations' cardinality and their expected performance, and as can be seen, it is hard to choose the best performing one judging only by their cardinalities. This brings us to the next discussion about a way to fuse the results to get an enhanced overall denoising performance.

2.3 Rule of Fusion

While it is hard to pinpoint the representation that performs best among those created by the RandOMP, their averaging is quite easy to obtain. The questions to be asked are: (i) What weights to use when averaging the various results? and (ii) Will this lead to better



a

b

c

d

Figure 3: Results of the RandOMP algorithm with 1,000 runs: (a) A histogram of the representations' cardinalities; (b) A histogram of the representations' errors; (c) A histogram of the representations' denoising factors; and (d) The denoising performance versus the cardinality.

overall denoising? We shall answer these questions intuitively and experimentally below. In Section 3 we revisit these questions and provide a justification for the choices made.

From an intuitive point of view, we might consider an averaging that gives a precedence to sparser representations. However, our experiments indicate that a plain averaging works even better. Thus, we use the formula[1]

$$\alpha^{AVE} = \frac{1}{J_0} \sum_{j=1}^{J_0} \alpha_j^{RandOMP}. \tag{3}$$

We return to the experiment described in the previous sub-section, and use its core to explore the effect of the averaging described above. We perform $1,000$ different experiments that share the same dictionary but generate different signals α, $\mathbf{x}$ and $\mathbf{y}$ using the same parameters ($\sigma_x = \sigma = 1$ and $k = 10$). For each experiment we generate $J_0 = 100$ RandOMP representations and average them using Equation (3).

Figure 4 presents the results – for each experiment a point is positioned at the denoising performance of the OMP and the corresponding averaged RandOMP. As can be seen, the general tendency suggests much better results with the RandOMP. The average denoising performance over all these experiments is 0.186 for the OMP and 0.105 for the averaged RandOMP method. The mean denoising factor of OMP versus that of RandOMP is denoted by a square symbol.

The above results are encouraging and immediately lead to more questions: (i) How many different representations are enough in order to enjoy the gain of the RandOMP averaging? (ii) How does this gain behave as a function of the input Signal to Noise Ratio (SNR) ? (iii) How does this gain behave for different cardinalities of the original representation? (iv) What is the effect of the dictionary (and its redundancy) on these results? (v) Are these results related to some sort of known estimator? and most important of all, (vi) Why do we get this gain at all? We shall provide experimental answers to questions (i)-(iv) in the next

[1] In Section 3 we show that both the OMP and RandOMP solutions should actually be multiplied by a shrinkage factor, c^2, defined in (2), which is omitted in this experiment.

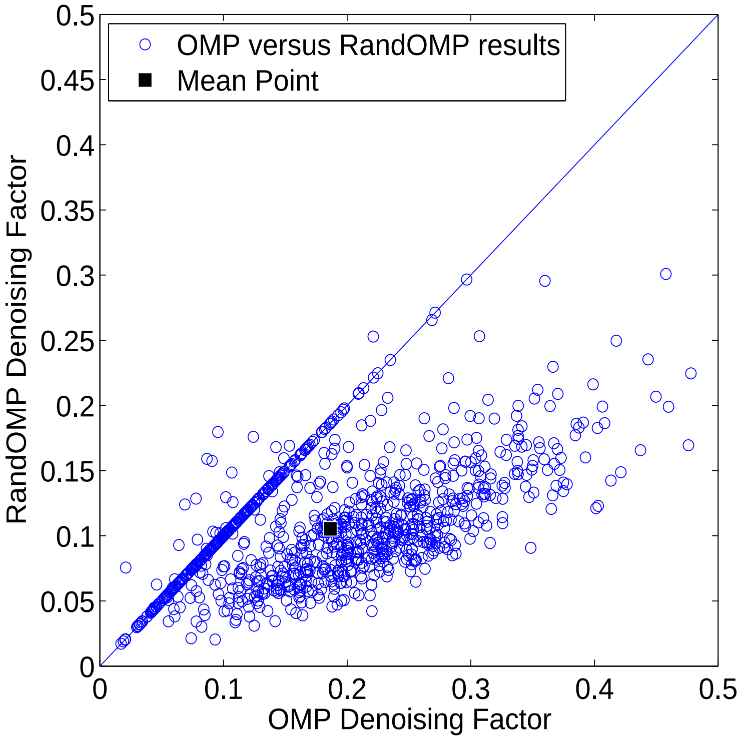


Figure 4: Results of $1,000$ experiments showing the plain OMP denoising performance versus those obtained by the averaged RandOMP with $J_0 = 100$ candidate results.

sub-section, and treat questions (v)-(vi) in Section 3 by providing a detailed analysis of the estimation problem at hand.

Just before leaving this section, we would like to draw attention to the following interesting behavior. When averaging the representations in forming the denoising estimate, we obtain a new representation α^{AVE} that is no longer sparse. Nevertheless, this representation is the one that leads to the improved results. Figure 5 shows the true representation, the OMP one, and α^{AVE} obtained with $1,000$ runs of the RandOMP, in a sequel to the experiment shown in Section 2.2. As can be seen, these three representations are quite distinct, and yet they lead to very similar signals (the denoising factor obtained in this case is 0.168 for the OMP, and 0.06 for the averaged representation). While the OMP uses less atoms than the original one, the averaged representation is dense, using *all* the atoms with appropriate weights.

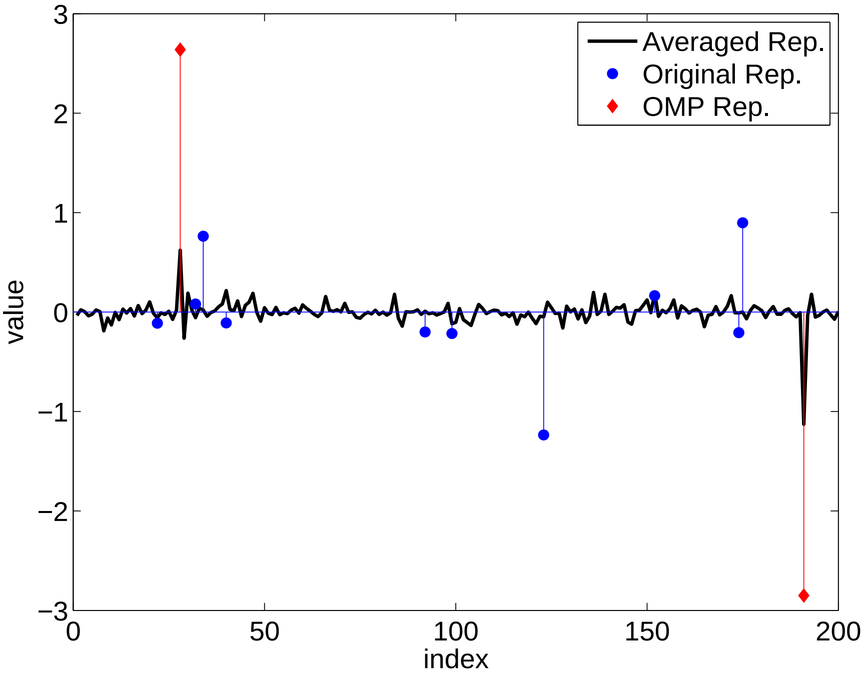


Figure 5: The true (original) representation, the one found by the OMP, and the one obtained by averaging $1,000$ representations created by RandOMP.

2.4 Hey, It Works! Some Experiments and Results

In this section we shall empirically answer some of the questions raised above, with an aim to better map the behavior of the RandOMP averaging method in different scenarios for various settings.

First we address the question of how many different representations to use in order to enjoy the gain of RandOMP averaging. As the complexity of the new estimator with J_0 different representations is about J_0 times higher than that of the plain OMP, there is a strong incentive to reduce J_0 as much as possible without sacrificing performance. Figure 6-a presents the averaged results over $1,000$ experiments, for a varying number of representations in the range $J_0 \in [5, 200]$. We see that while more representations improve the results, the lion's share of the improvement over the OMP is obtained even for small values of J_0.

All the tests done so far assumed $\sigma = \sigma_x = 1$ with $k = 10$. This case corresponds to a very low SNR of k/n and below, since the noise power is $n\sigma^2$, while the signal power is below $k\sigma_x^2$ (depending on the k atoms chosen and their relative orientations.) Thus, we must ask – how is the gain observed affected by the input SNR? In order to explore this, we fix the parameters $\sigma_x = 1$, $k = 10$, $J_0 = 40$, vary the noise power in the range $\sigma \in [0.1, 2]$, and average the denoising results over 200 experiments. Figure 6-b presents the denoising performance of the averaging as a function of σ, and as can be seen, our method is better for all the choices of σ, but the gain it provides is higher for lower SNR.

The next test we perform considers the complexity of the original signal, by varying k in the range $[1, 40]$. The sparser the representation of the original signal, the easier it is supposed to be denoised. Naturally, we desire to find out how the gain of the RandOMP average behaves for different cardinalities of the original representation. Figure 6-c presents the results obtained for $\sigma = \sigma_x = 1$, showing that the OMP is inferior to the averaged results for all cardinalities.

The last test we present studies the effect of the redundancy of the dictionary on the denoising performance. We fix the parameters $\sigma = \sigma_x = 1$, $k = 10$, $J_0 = 40$, the dimension of the signal is set to $n = 100$, and we vary the number of the atoms in the range $m \in [10, 400]$. Averaging the denoising results over 200 experiments we obtain the results as shown in Figure 6-d. These clearly show that for a wide range of redundancies, the gain obtained by the averaging of the RandOMP results remains unchanged, and the denoising factor appears to be independent of m (as opposed to the one obtained by the OMP which deteriorates). The case of underdetermined dictionaries ($m \leq n = 100$) and especially for $m \rightarrow k$ is special, since there the representations found tend to be full, leading to a convergence of the two methods (OMP and RandOMP).

We add that a similar test done on a redundant DCT dictionary[2] led to very similar results, suggesting that the the behavior we observe is robust with respect to the dictionary properties.

2.5 Summary

If we have at our disposal several competing sparse representations of the same noisy signal, they can be averaged to provide better denoising performance. The combined representation is no longer sparse, but this does not reduce its efficiency in attenuating the noise in the signal. In this section we described how to obtain such a group of representations, how to fuse them, and what to expect. Specifically, we found out that the method we propose appears to be very effective, robust with respect to the signal complexity, dictionary type and redundancy, and yields benefits even when we merge only a few representations. We now turn to provide a deeper explanation of these results by a careful modelling of the estimation problem and development of MAP and MMSE estimators.

[2] This dictionary is obtained by assigning

$$d[i, j] = \cos((i-1)(j-1)\pi/m) \quad \text{for} \quad 1 \leq i \leq n \quad \text{and} \quad 1 \leq j \leq m,$$

removing the mean from all the atoms apart from the first, and normalizing each atom to unit ℓ_2-norm.

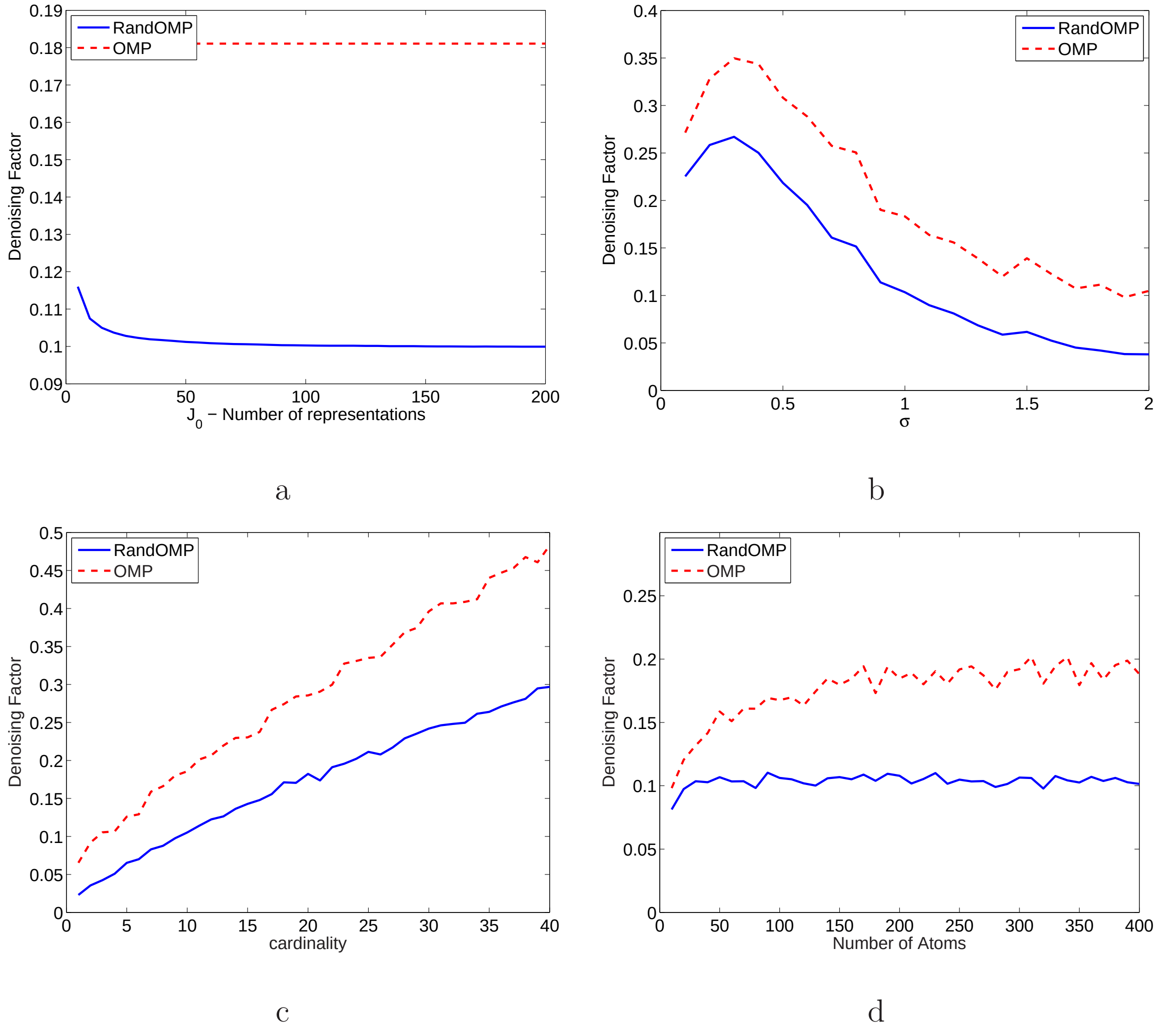


Figure 6: Various tests on the RandOMP algorithm, checking how the denoising is affected by (a) The number of representations averaged; (b) The input noise power; (c) The original representation's cardinality; and (d) The dictionary's redundancy.

3 Why Does it Work? A Rigorous Analysis

In this section we start by modelling the signal source in a complete manner, define the denoising goal in terms of the MSE, and derive several estimators for it. We start with a very general setting of the problem, and then narrow it down to the case discussed above on sparse representations. Our main goal in this section is to show that the MMSE estimator can be written as a weighted averaging of various sparse representations, which explains the results of the previous section. Beyond this, the analysis derives exact expressions for the MSE for various estimators, enabling us to assess analytically their behavior and relative performance, and to explain results that were obtained empirically in Section 2. Towards the end of this section we tie the empirical and the theoretical parts of this work – we again perform simulations and show how the actual denoising results obtained by OMP and RandOMP compare to the analytic expressions developed here.

3.1 A General Setting

3.1.1 Notation

We denote continuous (resp., discrete) vector random variables by small (resp., capital) letters. The probability density function (PDF) of a continuous random variable a over a domain Ω_a is denoted $p(a)$, and the probability of a discrete random variable A by $P(A)$. If $b_1, \ldots, b_k$ is a set of continuous (and/or discrete) random variables, then $p(a|b_1, \ldots, b_k)$ denotes the conditional PDF of a subject to b_1 AND b_2, ... AND b_k. Similarly, $P(A|B_1, \ldots, B_k)$ denotes the conditional probability for discrete event A. With $\mathcal{E}$ denoting expectation, we denote the mean of a by

$$\mathcal{M}(a) = \mathcal{E}(a) = \int_{\Omega_a} ap(a)\, da\,,$$

and the variance by

$$\mathcal{V}(a) = \mathcal{E}\left(\|a - \mathcal{M}(a)\|^2\right) = \int_{\Omega_a} \|a - \mathcal{M}(a)\|^2 p(a)\, da.$$

Similarly, in the discrete case,

$$\mathcal{M}(A) = \mathcal{E}(A) = \sum_{\Omega_A} AP(A)\,,$$

and

$$\mathcal{V}(A) = \mathcal{E}\left(\|A - \mathcal{M}(A)\|^2\right) = \sum_{\Omega_A} \|A - \mathcal{M}(a)\||^2 P(A).$$

Finally, we denote conditional means and variances by

$$\begin{aligned}
\mathcal{M}_{b_1,\ldots,b_k}(a) &= \mathcal{E}(a|b_1,\ldots,b_k) = \int_{\Omega_a} ap(a|b_1,\ldots,b_k)\,da,\\
\mathcal{V}_{b_1,\ldots,b_k}(a) &= \mathcal{E}\left(\|a - \mathcal{M}_{b_1,\ldots,b_k}(a)\|^2|b_1,\ldots,b_k\right)\\
&= \int_{\Omega_a} \|a - \mathcal{M}_{b_1,\ldots,b_k}(a)\|^2 p(a|b_1,\ldots,b_k)\,da\,,\\
\mathcal{M}_{B_1,\ldots,B_k}(A) &= \mathcal{E}(A|B_1,\ldots,B_k) = \sum_{\Omega_A} AP(A|B_1,\ldots,B_k)\,,\\
\mathcal{V}_{B_1,\ldots,B_k}(A) &= \mathcal{E}\left(\|A - \mathcal{M}(A)\|^2|B_1,\ldots,B_k\right)\\
&= \sum_{\Omega_A} \|A - \mathcal{M}_{B_1,\ldots,B_k}(a)\|^2 P(A|B_1,\ldots,B_k)\,.
\end{aligned}$$

3.1.2 Modelling the Problem

Given a dictionary $\mathbf{D} \in \mathbb{R}^{n\times m}$, let Ω denote the set of all 2^m sub-dictionaries, where a sub-dictionary, $\mathbf{S}$, will interchangeably be considered as a subset of the columns of $\mathbf{D}$ or as a matrix comprised of such columns. We assume that a random signal, $\mathbf{x} \in \mathbb{R}^n$, is selected by the following process. With each sub-dictionary, $\mathbf{S} \in \Omega$, we associate a non-negative probability, $P(\mathbf{S})$, with $\sum_{\mathbf{S}\in\Omega} P(\mathbf{S}) = 1$. Furthermore, with each signal $\mathbf{x}$ in the range of $\mathbf{S}$ (that is, such that there exists a vector $\mathbf{z} \in \mathbb{R}^k$ satisfying $\mathbf{Sz} = \mathbf{x}$,) denoted $\mathbf{x} \in \mathcal{R}(\mathbf{S})$, we associate a conditional PDF, $p(\mathbf{x}|\mathbf{S})$. Then, the clean signal $\mathbf{x}$ is assumed to be generated by first randomly selecting $\mathbf{S}$ according to $P(\mathbf{S})$, and then randomly choosing $\mathbf{x} \in \mathbf{S}$ according to $p(\mathbf{x}|\mathbf{S})$. After the signal is generated, an additive random noise term, $\mathbf{v}$, with PDF $p_v(\mathbf{v})$, is introduced, yielding a noisy signal $\mathbf{y} = \mathbf{x} + \mathbf{v}$.

Note that $P(\mathbf{S})$ can be used to represent a tendency towards sparsity. For example, we can choose $P(\mathbf{S})$ to be a strongly decreasing function of the number of elements in $\mathbf{S}$, or we can choose $P(\mathbf{S})$ to be zero for all $\mathbf{S}$'s except those with a particular (small) number of elements, etc.

Given $\mathbf{y}$, and assuming we know $p_v(\mathbf{v})$, $P(\mathbf{S})$ and $p(\mathbf{x}|\mathbf{S})$, our objective is to find an estimator, $\hat{\mathbf{x}}$, that will be as close as possible to the clean signal $\mathbf{x}$ in some sense. In this work we will mainly strive to minimize the conditional mean square error (MSE),

$$\mathrm{MSE}_y = \mathcal{E}\left(\|\hat{\mathbf{x}} - \mathbf{x}\|^2 \,|\mathbf{y}\right). \tag{4}$$

Note that typically one would expect to define the overall MSE without the condition over $\mathbf{y}$. However, this introduces a formidable yet unnecessary complication to the analysis that follows, and we shall avoid it.

3.1.3 Main Derivation

We first write the conditional MSE as the sum

$$\mathrm{MSE}_y = \sum_{\mathbf{S}\in\Omega} \mathrm{MSE}_{S,y} P(\mathbf{S}|\mathbf{y}), \tag{5}$$

with $\mathrm{MSE}_{S,y}$ defined as

$$\mathrm{MSE}_{S,y} = \mathcal{E}\left(\|\hat{\mathbf{x}} - \mathbf{x}\|^2 \,|\mathbf{S}, \mathbf{y}\right). \tag{6}$$

The first factor of the summation in (5) is the MSE subject to a noisy signal $\mathbf{y}$ and a given sub-dictionary $\mathbf{S}$, and the second factor is the probability of $\mathbf{S}$ given a noisy signal $\mathbf{y}$. By Bayes's formula, the latter is given by

$$P(\mathbf{S}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{S})P(\mathbf{S})}{p(\mathbf{y})}, \tag{7}$$

where

$$p(\mathbf{y}|\mathbf{S}) = \int_{\mathbf{x}\in\mathcal{R}(\mathbf{S})} p_v(\mathbf{y} - \mathbf{x}) p(\mathbf{x}|\mathbf{S})\, d\mathbf{x} \tag{8}$$

is the PDF of $\mathbf{y}$ given the sub-dictionary $\mathbf{S}$.

Note that $p(\mathbf{y})$ – the PDF of $\mathbf{y}$ – can be computed directly or, more easily, obtained from the normalization requirement

$$\sum_{\mathbf{S}\in\Omega} P(\mathbf{S}|\mathbf{y}) = 1. \tag{9}$$

Nevertheless, as we shall soon see, it is not explicitly needed in our analysis.

Next, we consider the first factor of the summation in (5), $\text{MSE}_{S,y}$, the MSE for a given $\mathbf{y}$ and sub-dictionary $\mathbf{S}$. Using the fact that $\mathcal{M}_{S,y}(\mathbf{x}) = \mathcal{E}\left(\mathbf{x}\,|\mathbf{S},\mathbf{y}\right)$, we have

$$\begin{aligned}\mathcal{E}\left(\|\mathbf{x}\|^2\,|\mathbf{S},\mathbf{y}\right) &= \mathcal{E}\left(\|\mathcal{M}_{S,y}(\mathbf{x}) + \mathbf{x} - \mathcal{M}_{S,y}(\mathbf{x})\|^2\,|\mathbf{S},\mathbf{y}\right) \\ &= \|\mathcal{M}_{S,y}(\mathbf{x})\|^2 + \mathcal{E}\left(\|\mathbf{x} - \mathcal{M}_{S,y}(\mathbf{x})\|^2\,|\mathbf{S},\mathbf{y}\right) \\ &= \|\mathcal{M}_{S,y}(\mathbf{x})\|^2 + \mathcal{V}_{S,y}(\mathbf{x}).\end{aligned} \tag{10}$$

This property, along with the linearity of the expectation, can be used to rewrite the first factor of the summation in (5) as follows:

$$\begin{aligned}\text{MSE}_{S,y} = \mathcal{E}\left(\|\hat{\mathbf{x}} - \mathbf{x}\|^2\,|\mathbf{S},\mathbf{y}\right) &= \mathcal{E}\left(\|\hat{\mathbf{x}}\|^2 - 2\hat{\mathbf{x}}^T\mathbf{x} + \|\mathbf{x}\|^2\,|\mathbf{S},\mathbf{y}\right) \\ &= \|\hat{\mathbf{x}}\|^2 - 2\hat{\mathbf{x}}^T\mathcal{M}_{S,y}(\mathbf{x}) + \|\mathcal{M}_{S,y}(\mathbf{x})\|^2 + \mathcal{V}_{S,y}(\mathbf{x}) \\ &= \|\hat{\mathbf{x}} - \mathcal{M}_{S,y}(\mathbf{x})\|^2 + \mathcal{V}_{S,y}(\mathbf{x}).\end{aligned} \tag{11}$$

Finally, plugging this into (5) we obtain

$$\begin{aligned}\text{MSE}_y &= \sum_{\mathbf{S}\in\Omega}\left[\|\hat{\mathbf{x}} - \mathcal{M}_{S,y}(\mathbf{x})\|^2 + \mathcal{V}_{S,y}(\mathbf{x})\right] P(\mathbf{S}|\mathbf{y}) \\ &= \mathcal{E}\left(\|\hat{\mathbf{x}} - \mathcal{M}_{S,y}(\mathbf{x})\|^2|\mathbf{y}\right) + \mathcal{E}\left(\mathcal{V}_{S,y}(\mathbf{x})|\mathbf{y}\right)\,,\end{aligned} \tag{12}$$

with $P(\mathbf{S}|\mathbf{y})$ given by (7). As we have already mentioned, the overall MSE is given by

$$\text{MSE} = \mathcal{E}\left(\text{MSE}_y\right) = \int_{\mathbf{y}\in\mathbb{R}^n} \text{MSE}_y p(\mathbf{y})\,d\mathbf{y}\,, \tag{13}$$

but we shall not need this measure here.

3.1.4 The Optimal Estimator

By (12), the optimal $\hat{\mathbf{x}}$ that minimizes MSE_y is, not surprisingly, given by

$$\hat{\mathbf{x}}^{MMSE} = \mathcal{E}\left(\mathcal{M}_{S,y}(\mathbf{x})|\mathbf{y}\right)\,, \tag{14}$$

and, plugged to Equation (12), the resulting optimal conditional MSE is given by

$$\mathrm{MSE}_y^{MMSE} = \mathcal{E}\left(\|\mathcal{M}_{S,y}(\mathbf{x}) - \mathcal{E}\left(\mathcal{M}_{S,y}(\mathbf{x})|\mathbf{y}\right)\|^2|\mathbf{y}\right) + \mathcal{E}\left(\mathcal{V}_{S,y}(\mathbf{x})|\mathbf{y}\right). \tag{15}$$

Finally, from (12) and (14) we obtain for an arbitrary estimator $\hat{\mathbf{x}}$ the conditional MSE

$$\mathrm{MSE}_y = \mathrm{MSE}_y^{MMSE} + \|\hat{\mathbf{x}} - \hat{\mathbf{x}}^{MMSE}\|^2. \tag{16}$$

This can be used to determine how much better the optimal estimator does compared to any other estimator.

3.1.5 The Maximum a-Posteriori (MAP) Estimator

The MAP estimator is obtained by maximizing the probability of $\mathbf{x}$ given $\mathbf{y}$,

$$\begin{aligned}\hat{\mathbf{x}}^{MAP} &= \arg\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}) \\ &= \arg\max_{\mathbf{x}} \frac{p(\mathbf{y}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{y})} \\ &= \arg\max_{\mathbf{x}} p_v(\mathbf{y}-\mathbf{x})p(\mathbf{x}),\end{aligned} \tag{17}$$

with

$$p(\mathbf{x}) = \sum_{\mathbf{S}\in\Omega:\ \mathbf{x}\in\mathcal{R}(\mathbf{S})} p(\mathbf{x}|\mathbf{S})P(\mathbf{S}).$$

At the moment these expressions remain vague, but as we turn to use the specific signal and noise models discussed in Section 3.1.2, these will assume an explicit form.

3.1.6 The Oracle

Suppose that the sub-dictionary $\mathbf{S}$ that was chosen in the generation of $\mathbf{x}$ is revealed to us. Given this information, we clearly minimize MSE_y by setting $\hat{\mathbf{x}} = \mathcal{M}_{S,y}(\mathbf{x})$ for the given $\mathbf{S}$. We call this the oracle estimator. The resulting conditional MSE is evidently given by the last term of (12),

$$\mathrm{MSE}_y^{oracle} = \mathcal{E}\left(\mathcal{V}_{S,y}(\mathbf{x})|\mathbf{y}\right). \tag{18}$$

We shall use this estimator to assess the performance of the various alternatives and see how close we get to this “ideal” performance.

3.2 Back to Our Story – Sparse Representations

Our aim now is to harness the general derivation to the development of a practical algorithm for the sparse representation and white Gaussian noise. Motivated by the sparse-representation paradigm, we concentrate on the case where $P(\mathbf{S})$ depends only on the number of atoms (columns) in $\mathbf{S}$, denoted $|\mathbf{S}|$. We start with the basic case where $P(\mathbf{S})$ vanishes unless $|\mathbf{S}|$ is exactly equal to some particular $0 \le k \le \min(n, m)$, and $\mathbf{S}$ has column rank k. We denote the set of such $\mathbf{S}$'s by Ω_k, and define the uniform distribution

$$P(\mathbf{S}) = \begin{cases} \frac{1}{|\Omega_k|} & \mathbf{S} \in \Omega_k \, , \\ 0 & \text{otherwise.} \end{cases}$$

We assume throughout that the columns of $\mathbf{D}$ are normalized, $\|\mathbf{d}_j\| = 1$, for $j = 1, \ldots, n$. This assumption comes only to simplify the expressions we are about to obtain. Next, we recall that the noise is modelled via a Gaussian distribution with zero mean and variance σ^2, and thus

$$p\left(\mathbf{y}|\mathbf{x}\right) = p_v\left(\mathbf{y} - \mathbf{x}\right) = \frac{1}{\left(2\pi\sigma^2\right)^{n/2}} \cdot \exp\left\{ \frac{-\|\mathbf{y} - \mathbf{x}\|^2}{2\sigma^2} \right\}. \tag{19}$$

Similarly, given the sub-dictionary $\mathbf{S}$ from which $\mathbf{x}$ is drawn, the signal $\mathbf{x}$ is assumed to be generated via a Gaussian distribution with mean zero and variance σ_x^2, thus $p(\mathbf{x}|\mathbf{S})$ is given by

$$p\left(\mathbf{x}|\mathbf{S}\right) = \begin{cases} \frac{1}{\left(2\pi\sigma_x^2\right)^{k/2}} \cdot \exp\left\{ \frac{-\|\mathbf{x}\|^2}{2\sigma_x^2} \right\} & \mathbf{x} \in \mathcal{R}(\mathbf{S}) \\ 0 & \text{otherwise.} \end{cases} \tag{20}$$

Note that this distribution does not align with the intuitive creation of $\mathbf{x}$ as $\mathbf{S}\mathbf{z}$ with a Gaussian vector $\mathbf{z} \in \mathbb{R}^k$ with i.i.d. entries. Instead, we assume that an orthogonalized basis for this sub-dictionary has been created and then multiplied by $\mathbf{z}$. We adopt the latter model for simplicity; the former model has also been worked out in full, but we omit it here because it is significantly more complicated and seems to afford only modest additional insights.

For convenience, we introduce the notation $c^2 = \sigma_x^2/(\sigma^2 + \sigma_x^2)$ (cf. (2)). Also, we denote the orthogonal projection of any vector $\mathbf{a}$ onto the subspace spanned by the columns of $\mathbf{S}$ by

$$\mathbf{a}_S = \mathbf{S}\left(\mathbf{S}^T\mathbf{S}\right)^{-1}\mathbf{S}^T\mathbf{a}.$$

We now follow the general derivation given above. From Equation (8) we can develop a closed-form expression for $p(\mathbf{y}|\mathbf{S})$. By integration and rearrangement we obtain

$$\begin{aligned} p(\mathbf{y}|\mathbf{S}) &= \int_{\mathbf{x}\in\mathcal{R}(\mathbf{S})} \frac{1}{(2\pi\sigma^2)^{n/2}\cdot(2\pi\sigma_x^2)^{k/2}} \cdot \exp\left\{\frac{-\|\mathbf{y}-\mathbf{x}\|^2}{2\sigma^2}+\frac{-\|\mathbf{x}\|^2}{2\sigma_x^2}\right\} d\mathbf{x} \\ &= \frac{(1-c^2)^{k/2}}{|\Omega_k|\,(2\pi\sigma^2)^{n/2}} \cdot \exp\left\{\frac{-(1-c^2)\|\mathbf{y}\|^2}{2\sigma^2}\right\}\cdot\exp\left\{\frac{-c^2\|\mathbf{y}-\mathbf{y}_S\|^2}{2\sigma^2}\right\}. \end{aligned} \tag{21}$$

Since the only dependence of $p(\mathbf{y}|\mathbf{S})$ on $\mathbf{S}$ is through the right-most factor, we immediately obtain by (7) and (9) the simple formula

$$P(\mathbf{S}|\mathbf{y}) = \frac{\exp\left\{-\frac{c^2\|\mathbf{y}-\mathbf{y}_S\|^2}{2\sigma^2}\right\}}{\sum_{\mathbf{S}'\in\Omega_k}\exp\left\{\frac{c^2\|\mathbf{y}-\mathbf{y}_{S'}\|^2}{2\sigma^2}\right\}}. \tag{22}$$

The denominator here is just a normalization. The numerator implies that, given a noisy signal $\mathbf{y}$, the probability that the clean signal was selected from the subspace $\mathbf{S}$ decays at a Gaussian rate with the distance between $\mathbf{y}$ and $\mathbf{S}$, i.e., $\|\mathbf{y}-\mathbf{y}_S\|$. This result is expected, given the Gaussian noise distribution.

Continuing to follow the general analysis, we compute the conditional mean, $\mathcal{M}_{S,y}(\mathbf{x})$, for which we require the conditional probability

$$\begin{aligned} p(\mathbf{x}|\mathbf{S},\mathbf{y}) &= \frac{p(\mathbf{y}|\mathbf{S},\mathbf{x})\,p(\mathbf{x}|\mathbf{S})}{p(\mathbf{y}|\mathbf{S})} \\ &= \frac{1}{p(\mathbf{y}|\mathbf{S})}\cdot\frac{1}{(2\pi\sigma^2)^{n/2}}\cdot\exp\left\{\frac{-\|\mathbf{y}-\mathbf{x}\|^2}{2\sigma^2}\right\}\cdot\frac{1}{(2\pi\sigma_x^2)^{k/2}}\cdot\exp\left\{\frac{-\|\mathbf{x}\|^2}{2\sigma_x^2}\right\}. \end{aligned} \tag{23}$$

By integration, we then obtain the simple result,

$$\mathcal{M}_{S,y}(\mathbf{x}) = \int_{\mathbf{x}\in\mathcal{R}(\mathbf{S})} \mathbf{x}p(\mathbf{x}|\mathbf{S},\mathbf{y})d\mathbf{x} = c^2\mathbf{y}_S. \tag{24}$$

Now the conditional variance can be computed, yielding

$$\mathcal{V}_{S,y}(\mathbf{x}) = \int_{\mathbf{x}\in\mathcal{R}(\mathbf{S})} \|\mathbf{x}-c^2\mathbf{y}_S\|^2 p(\mathbf{x}|\mathbf{S},\mathbf{y})d\mathbf{x} = kc^2\sigma^2, \tag{25}$$

which is independent of $\mathbf{S}$ and $\mathbf{y}$. Thus, the oracle MSE_y in this case is simply

$$\mathrm{MSE}_y^{oracle} = kc^2\sigma^2. \tag{26}$$

The optimal estimator is given by Equation (14),

$$\begin{aligned} \hat{\mathbf{x}}^{MMSE} &= c^2 \sum_{\mathbf{S}\in\Omega_k} \mathbf{y}_S P(\mathbf{S}|\mathbf{y}) \qquad (27) \\ &= \frac{c^2}{\sum_{\mathbf{S}'\in\Omega_k} \exp\left\{-\frac{c^2\|\mathbf{y}-\mathbf{y}_{S'}\|^2}{2\sigma^2}\right\}} \cdot \sum_{\mathbf{S}\in\Omega_k} \exp\left\{\frac{-c^2\|\mathbf{y}-\mathbf{y}_S\|^2}{2\sigma^2}\right\}\mathbf{y}_S, \end{aligned}$$

with $P(\mathbf{S}|\mathbf{y})$ taken from (22). This MMSE estimate is a weighted average of the projections of $\mathbf{y}$ onto all the possible sub-spaces $\mathbf{S} \in \Omega_k$, as claimed. The MSE of this estimate is given by

$$\mathrm{MSE}_y^{MMSE} = kc^2\sigma^2 + \sum_{\mathbf{S}\in\Omega_k} \|\hat{\mathbf{x}}^{MMSE} - c^2\mathbf{y}_S\|^2 P(\mathbf{S}|\mathbf{y}). \tag{28}$$

The latter can also be written as

$$\mathrm{MSE}_y^{MMSE} = kc^2\sigma^2 - \|\hat{\mathbf{x}}^{MMSE}\|^2 + \sum_{\mathbf{S}\in\Omega_k} \|c^2\mathbf{y}_S\|^2 P(\mathbf{S}|\mathbf{y}). \tag{29}$$

We remark that *any* spherically symmetric $p_v(\mathbf{v})$ and $p(\mathbf{x}|\mathbf{S})$ produce a conditional mean, $\mathcal{M}_{S,y}(\mathbf{x})$, that is equal to $\mathbf{y}_S$ times some scalar coefficient. The choice of Gaussian distributions makes the result in (24) particularly simple in that the coefficient, c^2, is independent of $\mathbf{y}$ and $\mathbf{S}$.

Next, we consider the Maximum a Posterior (MAP) estimator, using (17). For simplicity, we shall neglect the fact that some $\mathbf{x}$'s may lie on intersections of two or more sub-dictionaries in Ω_k, and therefore their PDF is higher according to our model. This is a set of measure zero, and it therefore does not influence the MMSE solution, but it does influence somewhat the MAP solution for $\mathbf{y}$'s that are close to such $\mathbf{x}$'s. We can overcome this technical difficulty by modifying our model slightly so as to eliminate the favoring of such $\mathbf{x}$'s. Noting that $P(\mathbf{S})$ is a constant for all $\mathbf{S} \in \Omega_k$, we obtain from (17)

$$\hat{\mathbf{x}}^{MAP} = \arg \max_{\mathbf{x}\in\mathcal{R}(\Omega_k)} \exp\left\{\frac{-\|\mathbf{y}-\mathbf{x}\|^2}{2\sigma^2}\right\} \cdot \exp\left\{\frac{-\|\mathbf{x}\|^2}{2\sigma_x^2}\right\}, \tag{30}$$

where $\mathcal{R}(\Omega_k)$ is defined as the union of the ranges of all $\mathbf{S} \in \Omega_k$. Multiplying through by $\exp(2c^2\sigma^2)$, we find that the maximum is obtained by minimizing $c^2\|\mathbf{y}-\mathbf{x}\|^2 + (1-c^2)\|\mathbf{x}\|^2$, subject to the constraint that $\mathbf{x}$ belongs to some $\mathbf{S} \in \Omega_k$. The resulting estimator is readily found to be given by

$$\hat{\mathbf{x}}^{MAP} = c^2 \mathbf{y}_{S_{MAP}} \,, \tag{31}$$

where $\mathbf{S}_{MAP}$ is the sub-space $\mathbf{S} \in \Omega_k$ which is closest to $\mathbf{y}$, i.e., for which $\|\mathbf{y} - \mathbf{y}_S\|^2$ is the smallest. The resulting MSE_y is given by substituting $\hat{\mathbf{x}}^{MAP}$ for $\hat{\mathbf{x}}$ in (16).

Note that in all the estimators we derive, the oracle, the MMSE, and the MAP, there is a factor of c^2 that performs a shrinking of the estimate. For the model of $\mathbf{x}$ chosen, this is a mandatory step that was omitted in Section 2.

3.3 Combining It All

It is now time to combine the theoretical analysis of section and the estimators we tested in Section 2. We have several goals in this discussion:

- We would like to evaluate both the expressions and the empirical values of the MSE for the oracle, the MMSE, and the MAP estimators, and show their behavior as a function of the input noise power σ,
- We would like to show how the above aligns with the actual OMP and the RandOMP results obtained, and
- This discussion will help explain two choices made in the RandOMP algorithm – the rule for drawing the next atom, and the requirement of a plain averaging of the representations.

We start by building a random dictionary of size 20×30 with ℓ^2-normalized columns. We generate signals following the model described above, by randomly choosing a support with

k columns (we vary in the range $[1,3]$), orthogonalizing the chosen columns, and multiplying them by a random i.i.d. vector with entries drawn from $N(0,1)$ (i.e. $\sigma_x = 1$). We add noise to these signals with σ in the range $[0.1,2]$ and evaluate the following values:

1. **Empirical Oracle** estimation and the MSE it induces. This estimator is simply the projection of $\mathbf{y}$ on the correct support, followed by a multiplication by c^2, as described in Equation (24) .

2. **Theoretical Oracle** estimation error, as given in Equation (26).

3. **Empirical MMSE** estimation and its MSE. We use the formula in Equation (27) in order to compute the estimation, and then assess its error empirically. Note that in applying this formula we gather all the $\binom{30}{k}$ possible supports, compute the projection of $\mathbf{y}$ onto them, and weight them according to the formula. This explains why in the experiment reported here we have restricted the sizes involved.

4. **Theoretical MMSE** estimation error, using Equation (29) directly.

5. **Empirical MAP** estimation and its MSE. We use the analytic solution to (30) as described above, by sweeping through all the possible supports, and searching the one with the smallest projection error. This gives us the MAP estimation, and its error is evaluated empirically.

6. **Theoretical MAP** estimation error, as given in Equation (16), when plugging in the MAP estimation.

7. **OMP** estimation and its MSE. The OMP is the same as described in Section 2, but the stopping rule is based on the knowledge of k, rather than on representation error. Following the MAP analysis done in Section 3, the result is multiplied by c^2 as well.

8. **Averaged RandOMP** estimation and its MSE. The algorithm generates $J_0 = 100$ representations and averages them. As in the OMP, the stopping rule for those is the

number of atoms k, and the result is also multiplied by c^2.

The above process is averaged over $1,000$ signal generations, and the resulting values are shown in Figures 7 for $k = 1,\ 2$, and 3.

First we draw attention to several general observations. As expected, we see in all these graphs that there is a good alignment between the theoretical and the empirical evaluation of the MSE for the oracle, the MMSE, and the MAP estimators. In fact, since the analysis is exact for this experiment, the differences are only due to the finite number of tests per σ. We also see that the denoising performance weakens as k grows. A third and intriguing observation that we will not explore here is the fact that there appears to be a critical input noise power ($\sigma \approx 0.4$) for which the MAP and the MMSE estimators (and their approximations) give their worst denoising performance, as exhibited by the hump in all the MMSE/MAP cases.

The OMP algorithm is an attempt to approximate the MAP estimation, replacing the need for sweeping through all the possible supports by a greedy detection of the involved atoms. As such, we expect it to be competitive and close to the MAP results we get (either analytically or empirically). In fact, for $k = 1$ it aligns perfectly with the empirical MAP, since both are going through the same computational stages. As k grows, there are some differences between the empirical MAP and the OMP, and especially for low noise, but for the cases studied here these differences are relatively small.

Just as OMP is an attempt to approximate the MAP estimation, the RandOMP averaging is approximating the MMSE estimator, thereby yielding much better denoising than OMP. The core idea is to replace the summation over all possible supports with a much smaller selected group of representations that are sampled from the distribution governed by the weights in Equation (27). Indeed, the representations chosen by RandOMP are those that correspond to large weights, since they are built in a way that leads to small projection error $\|\mathbf{y} - \mathbf{y}_P\|^2$ for the k atoms chosen. Since the sampling already mimics approximately the required distribution, all that remains is a simple averaging, as indeed we do in practice. What is required is to tune the sampling to be faithful, and for that we revisit the case of $k = 1$.

Considering the case of $k = 1$, we see from Equation (27) that an atom should be chosen as a candidate representation with a probability proportional to $\exp\{-c^2\|\mathbf{y} - \mathbf{y}_P\|^2/2\sigma^2\}$. This in turn implies that this probability is also proportional to[3] $\exp\{c^2|\mathbf{y}^T\mathbf{d}_i|^2/2\sigma^2\}$. Thus, RandOMP as described in Section 2 is with perfect agreement with this probability, and this explains the goodness of fit of RandOMP with the empirical MSE in Figure 7-a. However, we also see that RandOMP remains close to the empirical MMSE for $k = 2$ and 3, implying that while our sampling strategy is not perfect, it is fair enough. Further investigation is required to better sample the representations in order to get closer to the MSE estimate.

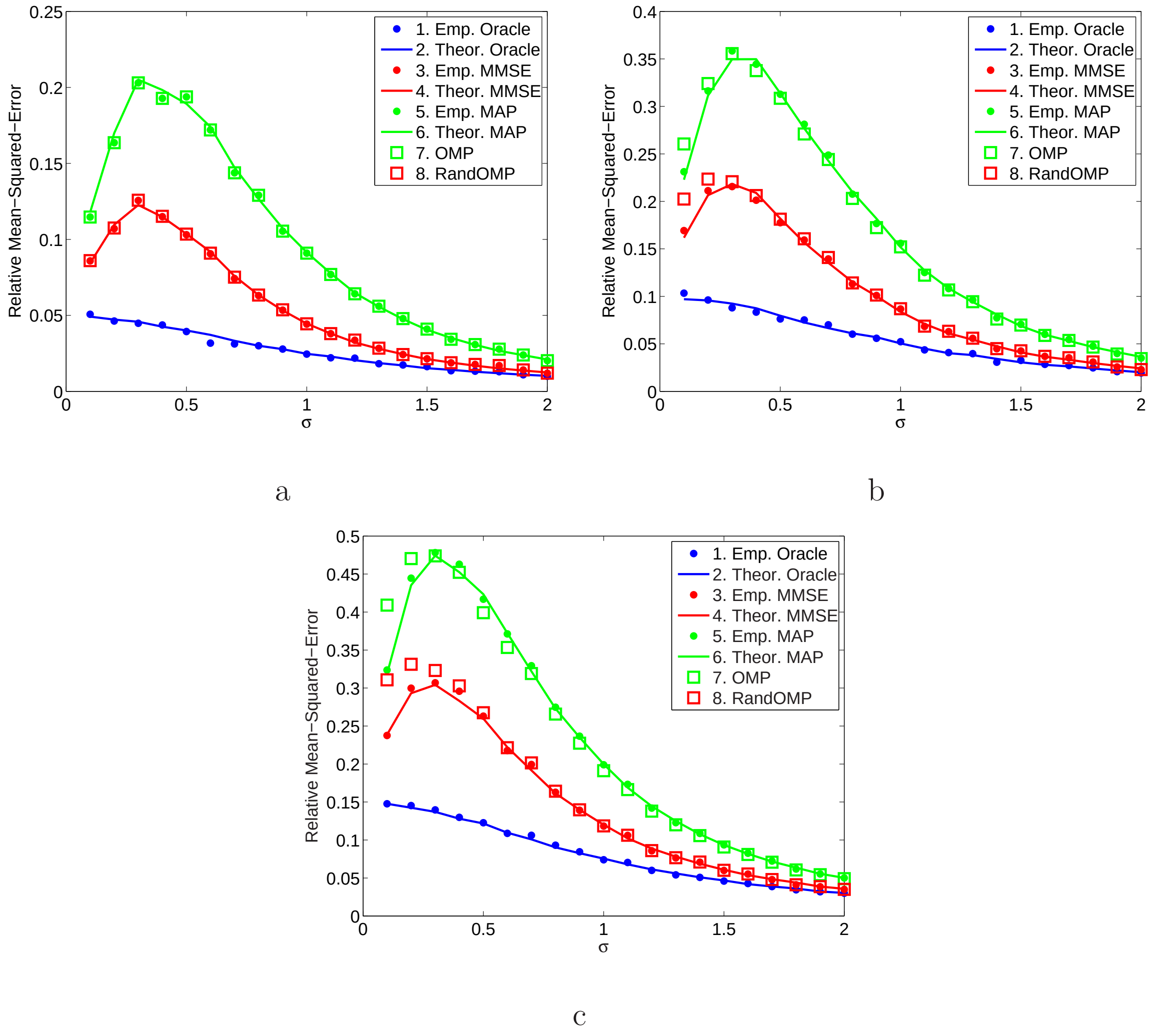
Finally, we note an additional advantage of RandOMP: the MMSE estimator varies continuously with y, whereas the MAP estimator does not, possibly leading to artifacts.

3.4 Summary

Under the assumptions of this section, we obtain simple explicit expressions for the optimal (MMSE) estimator and its resulting MSE_y. The optimal estimator turns out to be a weighted average of the orthogonal projections of the noisy signal on the feasible subspaces, multiplied by a "shrinkage factor" c^2, which tends to zero when the noise variance, σ^2, is large compared to the signal variance, σ_x^2, and to 1 when the opposite is true. The weights in the weighted average depend on the distances between $\mathbf{y}$ and the subspaces, favoring short distances of course, especially when $c^2\|\mathbf{y}\|^2/\sigma^2$ is large.

While the expressions obtained are indeed simple, they involve either an intolerable summations over $\binom{m}{k}$ (for the MMSE estimate), or searching over this amount of sub-spaces (for

[3] Since the columns of the dictionary are normalized, the projection is given by $\mathbf{y}_P = (\mathbf{y}^T\mathbf{d_i}) \cdot \mathbf{d}_i$. Thus, $\|\mathbf{y} - \mathbf{y}_P\|^2 = \|\mathbf{y}\|^2 - (\mathbf{y}^T\mathbf{d}_i)^2$. The term $\exp\{-c^2\|\mathbf{y}\|^2\}$ is therefore a constant that cancels-out in the normalization.



a

b

c

Figure 7: Various empirical and theoretical evaluations of the MSE as a function of the input noise for $k = 1$ (a), $k = 2$ (b), and $k = 3$ (c).

the MAP). Thus, these formulas are impractical for a direct use. In that sense, one should consider the RandOMP approach in Section 2 as a sampler from this huge set of subspaces over which we average. Roughly speaking, since the RandOMP algorithm tends to find near-by sub-spaces that lead to sparse representations, it gives priority to elements in the summation in Equation (27) that are assigned higher weights. We see experimentally that RandOMP samples well from the representations, judging by the proximity of its results to the MMSE error (both empirical and theoretical).

The results of this section can easily be extended to the case where we allow a range of values of k with given probabilities. That is, we can extend these results for the case where

$$P(\mathbf{S}) = f(|\mathbf{S}|), \tag{32}$$

for general non-negative functions f.

4 Summary and Conclusions

The Orthogonal Matching Pursuit is a simple and fast algorithm for approximating the sparse representation for a given signal. It can be used for denoising of signals, as a way to approximate the MAP estimation. In this work we have shown that by running this algorithm several times in a slightly modified version that randomizes its outcome, one can obtain a collection of competing representations, and those can be averaged to lead to far better denoising performance. This work starts by showing how to obtain a set of such representations to merge, how to combine them wisely, and what kind of results to expect. The analytic part of this paper explains this averaging as a way to approximate the MMSE estimate as a sampler of the summation required. Future work on this topic should consider better sampling strategies for better approximation of the MMSE result, an analytical and numerical study of the required number of samples, an assessment of the robustness of this approach with respect to non-Gaussian distribution of signals and limited accuracy in determining their variance, and exploration of special cases for which practical deterministic algorithms are within reach.

References

[1] S. Baker, and T. Kanade, Limits on super-resolution and how to break them, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(9):1167–1183, 2002.

[2] A.M. Bruckstein, D.L. Donoho, and M. Elad, From sparse solutions of systems of equations to sparse modelling of signals and images, to appear in *SIAM Review.*

[3] S. Chen, S.A. Billings, and W. Luo, Orthogonal least squares methods and their application to non-linear system identification, *International Journal of Control*, 50(5):1873–96, 1989.

[4] S.S. Chen, D.L. Donoho, and M.A. Saunders, Atomic decomposition by basis pursuit, *SIAM Journal on Scientific Computing*, 20(1):33–61 (1998).

[5] G. Davis, S. Mallat, and M. Avellaneda, Adaptive greedy approximations, *Journal of Constructive Approximation*, 13:57–98, 1997.

[6] G. Davis, S. Mallat, and Z. Zhang, Adaptive time-frequency decompositions, *Optical-Engineering*, 33(7):2183–91, 1994.

[7] D. Datsenko and M. Elad, Example-based single image super-resolution: a global MAP approach with outlier rejection, *Journal of Multidimensional System and Signal Processing*, 18(2):103–121, September 2007.

[8] D.L. Donoho, For most large underdetermined systems of linear equations, the minimal ℓ_1-norm near-solution approximates the sparsest near-solution, *Communications On Pure And Applied Mathematics*, 59(7):907–934, July 2006.

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

[10] D.L. Donoho, M. Elad, and V. Temlyakov, Stable recovery of sparse overcomplete representations in the presence of noise, *IEEE Trans. On Information Theory*, 52(1):6–18, 2006.

[11] M. Elad and M. Aharon, Image denoising via sparse and redundant representations over learned dictionaries, *IEEE Trans. on Image Processing* 15(12):3736–3745, December 2006.

[12] A.K. Fletcher, S. Rangan, V.K. Goyal, and K. Ramchandran, Analysis of denoising by sparse approximation with random frame asymptotics, IEEE Int. Symp. on Inform. Theory, 2005.

[13] A.K. Fletcher, S. Rangan, V.K. Goyal, and K. Ramchandran, Denoising by sparse approximation: error bounds based on rate-distortion theory, *EURASIP Journal on Applied Signal Processing*, Paper No. 26318, 2006.

[14] W.T. Freeman, T.R. Jones, and E.C. Pasztor, Example-based super-resolution, *IEEE Computer Graphics And Applications*, 22(2):56–65, 2002.

[15] W.T. Freeman, E.C. Pasztor, and O.T. Carmichael, Learning low-level vision, *International Journal of computer Vision*, 40(1):25–47, 2000.

[16] J.J. Fuchs, Recovery of exact sparse representations in the presence of bounded noise, *IEEE Trans. on Information Theory*, 51(10):3601–3608, 2005.

[17] R. Gribonval, R. Figueras, and P. Vandergheynst, A simple test to check the optimality of a sparse signal approximation, *Signal Processing*, 86(3):496–510, March 2006.

[18] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic-Press, 1998.

[19] S. Mallat and Z. Zhang, Matching pursuits with time-frequency dictionaries, *IEEE Trans. Signal Processing*, 41(12):3397–3415, 1993.

[20] B.K. Natarajan, Sparse approximate solutions to linear systems, *SIAM Journal on Computing*, 24:227–234, 1995.

[21] Y.C. Pati, R. Rezaiifar, and P.S. Krishnaprasad, Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition, the twenty seventh Asilomar Conference on Signals, Systems and Computers, 1:40–44, 1993.

[22] L.C. Pickup, S.J. Roberts, and A. Zisserman, A Sampled Texture Prior for Image Super-Resolution, *Advances in Neural Information Processing Systems*, 2003.

[23] J.A. Tropp, Greed is good: Algorithmic results for sparse approximation, *IEEE Trans. On Information Theory*, 50(10):2231–2242, October 2004.

[24] J.A. Tropp, Just relax: Convex programming methods for subset selection and sparse approximation, *IEEE Trans. On Information Theory*, 52(3):1030–1051, March 2006.

[25] B. Wohlberg, Noise sensitivity of sparse signal representations: Reconstruction error bounds for the inverse problem. *IEEE Trans. on Signal Processing*, 51(12):3053–3060, 2003.