Journal of Mathematical Imaging and Vision
https://doi.org/10.1007/s10851-019-00913-z

Adversarial Noise Attacks of Deep Learning Architectures: Stability Analysis via Sparse-Modeled Signals

Yaniv Romano[1] · Aviad Aberdam[2] · Jeremias Sulam[3] · Michael Elad[4]

Received: 20 November 2018 / Accepted: 25 September 2019

Abstract

Despite their impressive performance, deep convolutional neural networks (CNN) have been shown to be sensitive to small adversarial perturbations. These nuisances, which one can barely notice, are powerful enough to fool sophisticated and well performing classifiers, leading to ridiculous misclassification results. In this paper, we analyze the stability of state-of-the-art deep learning classification machines to adversarial perturbations, where we assume that the signals belong to the (possibly multilayer) sparse representation model. We start with convolutional sparsity and then proceed to its multilayered version, which is tightly connected to CNN. Our analysis links between the stability of the classification to noise and the underlying structure of the signal, quantified by the sparsity of its representation under a fixed dictionary. In addition, we offer similar stability theorems for two practical pursuit algorithms, which are posed as two different deep learning architectures—the layered thresholding and the layered basis pursuit. Our analysis establishes the better robustness of the later to adversarial attacks. We corroborate these theoretical results by numerical experiments on three datasets: MNIST, CIFAR-10 and CIFAR-100.

Keywords Theory for deep learning · Adversarial noise · Sparse coding

1 Introduction

Deep learning, and in particular convolutional neural networks (CNN), is one of the hottest topics in data sciences as it has led to many state-of-the-art results spanning across many domains [9,14]. Despite the evident great success of classifying images, it has been recently observed that CNN are highly sensitive to adversarial perturbations in the input signal [10,17,26]. An *adversarial example* is a corrupted version of a valid input (i.e., one that is classified correctly), where the corruption is done by adding a perturbation of a small magnitude to it. This barely noticed nuisance is designed to fool the classifier by maximizing the likelihood of an incorrect class. This phenomenon reveals that state-of-the-art classification algorithms are highly sensitive to noise, so much so that even a single step in the direction of the sign of the gradient of the loss function creates a successful adversarial example [10]. Furthermore, it has been shown that adversarial examples that are generated to attack one network are powerful enough to fool other networks of different architecture and database [17], being the key to the so-called "black-box" attacks that have been demonstrated in some real-world scenarios [13].

Y. Romano and A. Aberdam contributed equally to this work.
The research leading to these results has received funding from the Technion Hiroshi Fujiwara Cyber Security Research Center and the Israel Cyber Directorate, and from Israel Science Foundation (ISF) grant no. 335/18.
Y. R. thanks the Zuckerman Institute, ISEF Foundation and the Viterbi Fellowship, Technion, for supporting this research.

✉ Aviad Aberdam
aaberdam@campus.technion.ac.il

Yaniv Romano
yromano@stanford.edu

Jeremias Sulam
jsulam@jhu.edu

Michael Elad
elad@cs.technion.com

[1] Department of Statistics, Stanford University, Stanford, USA

[2] Electrical Engineering, Technion—Israel Institute of Technology, Haifa, Israel

[3] Biomedical Engineering, Johns Hopkins University, Baltimore, USA

[4] Computer Science, Technion—Israel Institute of Technology, Haifa, Israel

Published online: 05 October 2019

Adversarial training is a popular approach to improve the robustness of a given classifier [10]. It aims to train a robust model by augmenting the data with adversarial examples generated for the specific model and/or transferred from other models. Preprocessing [16] is another defense strategy, suggesting to denoise the input signal first, and then feed this purified version of the signal to the classifier. Indeed, the above defense methods improve the stability of the network; however, these are trained based on adversarial examples that are generated in specific ways. It is quite likely that future work could offer a different generation of adversarial examples that question again the reliability and robustness of such given networks.

In this paper, we provide a principled way to analyze the robustness of a classifier using the vast theory developed in the field of sparse representations. We do so by analyzing the classifier's robustness to adversarial perturbations, providing an upper bound on the permitted energy of the perturbation, while still safely classifying our data. The derived bounds are affected by the classifier's properties and the structure of the signal. Our analysis assumes that the signals of interest belong to the sparse representation model, which is known for its successful regression and classification performance [6, 19], and was recently shown to be tightly connected to CNN [21]. We commence by analyzing a shallow convolutional sparse model and then proceed to its multilayer extension. More concretely, suppose we are given a clean signal that is assigned to the correct class. How much noise of bounded energy can be added to this signal and still guarantee that it would be classified accurately? Our work shows that the bound on the energy of the noise is a function of the sparsity of the signal and the characteristics of the dictionaries (weights).

We proceed by considering specific and practical pursuit algorithms that aim to estimate the signal's representations in order to apply the classification. Our work investigates two such algorithms, the nonnegative layered thresholding (L-THR), which amounts to a conventional feed-forward CNN, and the nonnegative layered basis pursuit (L-BP), which is reminiscent of an RNN (residual neural network) architecture. Our analysis exposes the ingredients of the data model governing the sensitivity to adversarial attacks and clearly shows that the later pursuit (L-BP) is more robust.

The bounds obtained carry in them practical implications. More specifically, our study indicates that a regularization that would take the dictionaries' coherence into account can potentially improve the stability to noise. Interestingly, a regularization that aligns well with our findings was tested empirically by Parseval networks [20] and indeed shown to improve the classification stability. As such, one can consider our work as a theoretical explanation for the empirical success of [20]. Another approach that is tightly connected to our analysis is the one reported in [18,27]. Rather than relying on a simple L-THR, these papers suggested solving a variant of the L-BP algorithm, in an attempt to promote sparse feature maps. Interestingly, it was shown in [18] that the "fooling rate" in the presence of adversarial perturbation is significantly improved, serving as another empirical evidence to our theoretical conclusions. As will be shown in this paper, promoting sparse solutions and incoherent dictionaries is crucial for robust networks, as evidenced empirically in the above two papers [18,20].

We should note that this work does not deal with the learning phase of the networks, as we assume that we have access to the true model parameters. Put on more practical terms, our work analyzes the sensitivity of the chosen inference architectures to malicious noise, by imposing assumptions on the filters/dictionaries and the incoming signals. These architectures follow the pursuit algorithms we explore, and their parameters are assumed to be known, obtained after learning.

Moving to the experimental part, we start by demonstrating the derived theorems on a toy example, in order to better clarify the message of this work. Our simulations carefully illustrate how the L-BP is more stable to adversarial noise, when compared with the regular feed-forward neural network (i.e., the L-THR), and this is shown both in theoretical terms (showing the actual bounds) and in empirical performance. In order to further support the theoretical claims made in this paper, we numerically explore the stability of the L-THR and the L-BP architectures on actual data and learned networks. Note that in these experiments the theoretical assumptions do not hold, as we do not have an access to the true model. In this part, we consider three commonly tested datasets: MNIST [15], CIFAR-10 [12] and CIFAR-100 [12]. Our experiments show that the L-BP is indeed more robust to noise attacks, where those are computed using the fast gradient sign method (FGSM) [10].

This paper is organized as follows: In Sect. 2, we start by reviewing the basics of the convolutional sparse coding model and then proceed to its multilayered version, which is tightly connected to CNN. Then, using Sparseland tools we establish a connection between the stability of the classification to adversarial noise and the underlying structure of the signal, quantified by the sparsity of its representation. We commence by analyzing shallow networks in Sect. 3 and then continue to deeper settings in Sect. 4. In addition, we offer similar stability theorems for two pursuit algorithms, which are posed as two different deep learning architectures—the L-THR and the L-BP. In Sect. 5, we numerically study the stability of these architectures demonstrating the theoretical results, starting with a toy example using simulated data, and then continuing with tests on real data. We conclude in Sect. 6 by delineating further research directions.

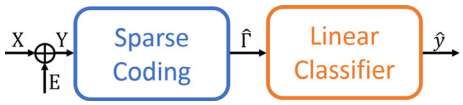


Fig. 1 The classification scheme consists of a sparse coding block and a linear classifier. The adversarial noise $\mathbf{E}$ aims to fail the classification, $\hat{y} \neq y$, while having of the smallest possible energy

2 Background and Problem Setup

Consider a set $\{s^j\}_j = \{(\mathbf{X}^j, y^j)\}_j$ of high-dimensional signals $\mathbf{X}^j \in \mathcal{X} \subseteq \mathbb{R}^N$ and their associated labels $y^j \in \mathcal{Y}$. Suppose that each signal $\mathbf{X}^j = \mathbf{D}\boldsymbol{\Gamma}^j$ belongs to the (possibly multilayer convolutional [21]) sparse representation model, where $\mathbf{D}$ is a dictionary and $\boldsymbol{\Gamma}^j$ is a sparse vector. Suppose further that we are given a linear classifier that operates on the *sparse representation* $\boldsymbol{\Gamma}^j$ and successfully discriminates between the different classes.

Ignoring the superscript j for clarity, given the input $s = (\mathbf{X}, y)$ the adversary's goal is to find an example $\mathbf{Y} = \mathbf{X} + \mathbf{E}$, such that the energy of $\mathbf{E}$ is small, and yet the model would misclassify $\mathbf{Y}$. Figure 1 depicts this classification scheme. We consider the class of ℓ_p bounded adversaries, in the sense that for a given energy ϵ, the adversarial example satisfies $\|\mathbf{Y} - \mathbf{X}\|_p = \|\mathbf{E}\|_p \leq \epsilon$.

How much perturbation $\mathbf{E} \in \mathbb{R}^N$ of bounded energy ϵ can be added to $\mathbf{X}$ so as the measurement $\mathbf{Y} = \mathbf{X} + \mathbf{E}$ will still be assigned to the correct class? What is the effect of the sparsity of the true representation? What is the influence of the dictionary $\mathbf{D}$ on these conclusions? How can we design a system that will be robust to noise based on the answers to the above questions? These questions are the scope of this paper. Before addressing these, in this section we provide the necessary background on several related topics.

2.1 Convolutional Sparse Coding

The convolutional sparse coding (CSC) model assumes that a signal $\mathbf{X} \in \mathbb{R}^N$ can be represented as $\mathbf{X} = \mathbf{D}\boldsymbol{\Gamma}$, where $\mathbf{D} \in \mathbb{R}^{N \times Nm}$ is a given convolutional dictionary and $\boldsymbol{\Gamma} \in \mathbb{R}^{Nm}$ is a sparse vector. The dictionary $\mathbf{D}$ is composed of m local unique filters of length n, where each of these is shifted at every possible location (see Fig. 2 left, here ignore the subscript '1' for clarity). The special structure of this matrix implies that the i-th patch $\mathbf{x}_i \in \mathbb{R}^n$ extracted from the global signal $\mathbf{X}$ has an underlying shift-invariant local model [22]. Concretely, $\mathbf{x}_i = \boldsymbol{\Omega}\mathbf{S}_i\boldsymbol{\Gamma}$, where $\boldsymbol{\Omega}$ is a fixed matrix shared by all the overlapping patches, multiplied by the corresponding stripe vector $\mathbf{S}_i\boldsymbol{\Gamma}_i \in \mathbb{R}^{(2n-1)m}$, where $\mathbf{S}_i \in \mathbb{R}^{(2n-1)m \times mN}$ extracts the stripe from the global $\boldsymbol{\Gamma}$.

Building upon the local structure of this model, it was shown in [22] that measuring the local sparsity of $\boldsymbol{\Gamma}$ rather than the global one is much more informative. The notion of local sparsity is defined by the $\ell_{0,\infty}$ pseudo-norm, expressed by $\|\boldsymbol{\Gamma}\|_{0,\infty}^{\mathrm{s}} = \max_i \|\mathbf{S}_i\boldsymbol{\Gamma}\|_0$, which counts the maximal number of nonzeros in the stripes (and hence the superscript s) of length $(2n-1)m$ extracted from $\boldsymbol{\Gamma}$.

In the setting of this paper, we are given a noisy measurement of $\mathbf{X} = \mathbf{D}\boldsymbol{\Gamma}$, formulated as $\mathbf{Y} = \mathbf{X} + \mathbf{E}$, where $\mathbf{E}$ is an ℓ_p-bounded *adversarial perturbation*. In the ℓ_2 case, the pursuit problem of estimating $\boldsymbol{\Gamma}$ given $\mathbf{Y}$, $\mathbf{D}$ and the energy of $\mathbf{E}$ (denoted by ϵ) is defined as

$$(\mathrm{P}_{0,\infty}^{\boldsymbol{\mathcal{E}}}): \quad \min_{\boldsymbol{\Gamma}} \|\boldsymbol{\Gamma}\|_{0,\infty}^{\mathrm{s}} \ \text{ s.t. } \ \|\mathbf{Y} - \mathbf{D}\boldsymbol{\Gamma}\|_2^2 \leq \epsilon^2. \tag{1}$$

The stability of the above problem and practical algorithms (orthogonal matching pursuit—OMP, and basis pursuit—BP) that aim to tackle it were analyzed in [22]. Under the assumption that $\boldsymbol{\Gamma}$ is "locally sparse enough," it was shown that one can obtain an estimate $\hat{\boldsymbol{\Gamma}}$ that is close to the true sparse vector $\boldsymbol{\Gamma}$ in an ℓ_2-sense. The number of nonzeros in $\boldsymbol{\Gamma}$ that guarantees such a stable recovery is a function of ϵ and the characteristics of the convolutional dictionary $\mathbf{D}$.

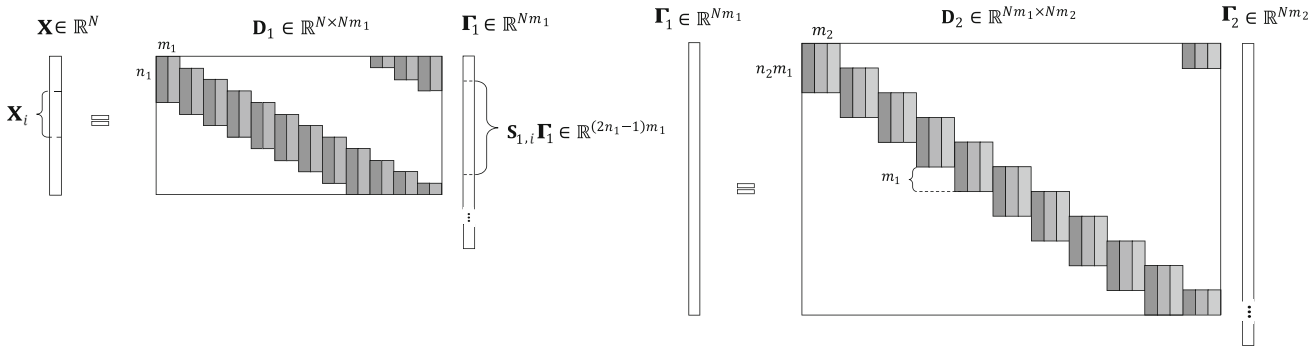


Fig. 2 Left: The global convolutional system $\mathbf{X} = \mathbf{D}_1\boldsymbol{\Gamma}_1$, along with the representation of the i-th patch $\mathbf{S}_{1,i}\boldsymbol{\Gamma}_1$. Right: The second layer of the multilayer CSC model, given by $\boldsymbol{\Gamma}_1 = \mathbf{D}_2\boldsymbol{\Gamma}_2$

Two measures that will serve us in our later analysis are (i) the extension of the restricted isometry property (RIP) [5] to the convolutional case, termed SRIP [22], and (ii) the mutual coherence. The SRIP of a dictionary $\mathbf{D}$ of cardinality k is denoted by δ_k. It measures how much the multiplication of a locally sparse vector $\mathbf{v}$, $\|\mathbf{v}\|_{0,\infty}^{\mathbf{S}} = k$ by $\mathbf{D}$ changes its energy (see definition 14 in [22]). A small value of $\delta_k (\ll 1)$ implies that $\mathbf{D}$ behaves almost like an orthogonal matrix, i.e., $\|\mathbf{D}\mathbf{v}\|_2 \approx \|\mathbf{v}\|_2$.

The second measure that we will rely on is the mutual coherence of a dictionary with ℓ_2 normalized columns, which is formulated as $\mu(\mathbf{D}) = \max_{i \neq j} |\mathbf{v}_i^T \mathbf{d}_j|$, where $\mathbf{d}_j$ stands for the j-th column (atom) from $\mathbf{D}$. In words, $\mu(\mathbf{D})$ is the maximal inner product of two distinct atoms extracted from $\mathbf{D}$.

2.2 Multilayer CSC

The multilayer convolutional sparse coding (ML-CSC) model is a natural extension of the CSC to a hierarchical decomposition. Suppose we are given a CSC signal $\mathbf{X} = \mathbf{D}_1 \Gamma_1$, where $\mathbf{D}_1 \in \mathbb{R}^{N \times Nm_1}$ is a convolutional dictionary and $\Gamma_1 \in \mathbb{R}^{Nm_1}$ is the (local) sparse representation of $\mathbf{X}$ over $\mathbf{D}_1$ (see Fig. 2 left). The ML-CSC pushes this structure forward by assuming that the representation itself is structured, and can be decomposed as $\Gamma_1 = \mathbf{D}_2 \Gamma_2$, where $\mathbf{D}_2 \in \mathbb{R}^{Nm_1 \times Nm_2}$ is another a convolutional dictionary, multiplied by the locally sparse vector $\Gamma_2 \in \mathbb{R}^{Nm_2}$ (see Fig. 2 right). Notice that Γ_1 has two roles, as it is the representation of $\mathbf{X}$, and a signal by itself that has a CSC structure. The second dictionary $\mathbf{D}_2$ is composed of m_2 local filters that skip m_1 entries at a time, where each of the filters is of length $n_2 m_1$. This results in a convolution operation in the spatial domain of Γ_1 but not across channels (Γ_1 has m_1 channels), as in CNN. The above construction is summarized in the following definition (Definition 1 in [21]):

Definition 1 For a global signal $\mathbf{X}$, a set of convolutional dictionaries $\{\mathbf{D}_i\}_{i=1}^K$, and a vector $\boldsymbol{\lambda}$, define the ML-CSC model as:

$$\Gamma_{i-1} = \mathbf{D}_i \Gamma_i, \quad \|\Gamma_i\|_{0,\infty}^{\mathbf{S}} \leq \lambda_i \quad \forall\, 1 \leq i \leq K$$

where $\Gamma_0 = \mathbf{X}$, and the scalar λ_i is the i-th entry in $\boldsymbol{\lambda}$.

Turning to the pursuit problem in the noisy regime, an extension of the CSC pursuit (see Eq. (1)) to the multilayer setting (of depth K) can be expressed as follows:

Definition 2 (*Definition 2 in* [21]) For a global signal $\mathbf{Y}$, a set of convolutional dictionaries $\{\mathbf{D}_i\}_{i=1}^K$, sparsity levels $\boldsymbol{\lambda}$ and noise energy ϵ, the deep coding problem is given by

$$(\mathrm{DCP}_{\boldsymbol{\lambda}}^{\boldsymbol{\mathcal{E}}}) : \text{find } \{\Gamma_i\}_{i=1}^K$$

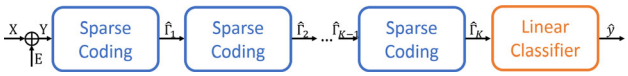


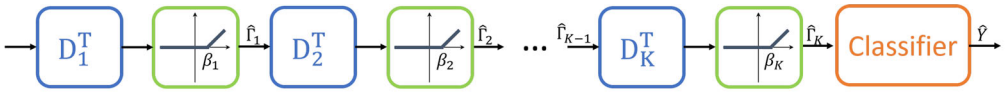
Fig. 3 A deep classification scheme consisting of a chain of sparse coding blocks and a linear classifier

$$\begin{aligned} &\text{s.t. } \|\mathbf{Y} - \mathbf{D}_1 \Gamma_1\|_2 \leq \epsilon, \\ &\quad \Gamma_{i-1} = \mathbf{D}_i \Gamma_i, \\ &\quad \|\Gamma_i\|_{0,\infty}^{\mathbf{S}} \leq \lambda_i, \qquad \forall\, 1 \leq i \leq K. \end{aligned}$$

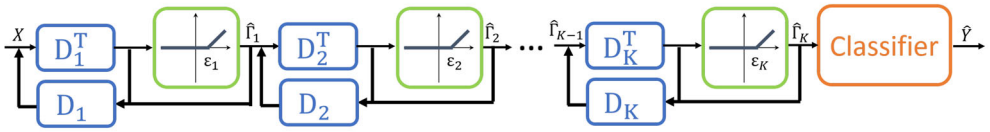
How can one solve this pursuit task? The work reported in [21] has shown that the forward pass of CNN is in fact a pursuit algorithm that is able to estimate the underlying representations $\Gamma_1, \ldots, \Gamma_K$ of a signal $\mathbf{X}$ that belongs to the ML-CSC model. Put differently, the forward pass was shown to be nothing but a nonnegative layered thresholding pursuit, estimating the representations Γ_i of the different layers. To better see this, let us set $\hat{\Gamma}_0 = \mathbf{Y}$ and define the classic thresholding pursuit [6], $\hat{\Gamma}_i = \S_{\beta_i}^+(\mathbf{D}_i^T \hat{\Gamma}_{i-1})$, for $1 \leq i \leq K$. The term $\mathbf{D}_i^T \hat{\Gamma}_{i-1}$ stands for convolving $\hat{\Gamma}_{i-1}$ (the feature map) with the filters of $\mathbf{D}_i$ (the weights), and the soft nonnegative thresholding function $\S_{\beta_i}^+(\mathbf{v}) = \max\{0, \mathbf{v} - \beta_i\}$ is the same as subtracting a bias β_i from $\mathbf{v}$ and applying a ReLU nonlinearity. In a similar fashion, the work in [21] offered to replace the thresholding algorithm in the sparse coding blocks with basis pursuit, exposing a recurrent neural network architecture that emerges from this approach.

This connection of CNN to the pursuit of ML-CSC signals was leveraged [21] to analyze the stability of CNN architectures. Their analysis concentrated only on the feature extraction stage—the pursuit—and ignored the classification step and the role of the labels. In this paper, we build upon this connection of Sparseland to CNN and extend the analysis to cover the stability of layered pursuit algorithms when tackling the classification task in the presence of noise. In Sect. 4 we shall consider a classifier consisting of a chain of sparse coding blocks and a linear classifier at its deepest layer, as depicted in Fig. 3. Our work aims to analyze the stability of such a scheme, suggesting that replacing the pursuit algorithm in the sparse coding blocks from thresholding to basis pursuit yields a more stable architecture with respect to adversarial noise, both theoretically and practically.

More specifically, we first study the stability to adversarial noise of the feed-forward CNN classifier. Or equivalently, where each of the pursuit algorithms in Fig. 3 is chosen to be the Thresholding. This architecture is depicted in Fig. 4a. Then, we switch to the basis pursuit as the sparse coding, serving better the sparse model, and resulting a new deep learning architecture with the same number of parameters



(a) The Layered-Thresholding classifier (L-THR), corresponding to a CNN - a feed-forward convolutional neural network.



(b) The Layered-Basis-Pursuit classifier (L-BP), corresponding to a CNN with additional feedback loops.

Fig. 4 The deep classifier architectures considered in this work

but with additional feedback loops as illustrated in Fig. 4b.[1] We now give more formal definitions of these two schemes.

Definition 3 (*L-THR*) For an ML-CSC signal $\mathbf{Y} = \mathbf{X} + \mathbf{E}$ with convolutional dictionaries $\{\mathbf{D}_i\}_{i=1}^K$, thresholds $\{\beta_i\}_{i=1}^K$, and a classifier $(\mathbf{w}, \omega)$, define the layered thresholding (L-THR) algorithm as: Apply

$$\hat{\Gamma}_i = \S_{\beta_i}^+(\mathbf{D}_i^T \hat{\Gamma}_{i-1}) \quad \text{for} \quad i = 1, 2, \ \ldots \ , K,$$

and assign $y = sign\left(f(\hat{\Gamma}_K)\right)$, where

$$f(\hat{\Gamma}_K) = \mathbf{w}^T \hat{\Gamma}_K + \omega.$$

Definition 4 (*L-BP*) For an ML-CSC signal $\mathbf{Y} = \mathbf{X} + \mathbf{E}$ with convolutional dictionaries $\{\mathbf{D}_i\}_{i=1}^K$, Lagrangian multipliers $\{\epsilon_i\}_{i=1}^K$, and a classifier $(\mathbf{w}, \omega)$, define the layered basis pursuit (L-BP) algorithm as: Apply

$$\hat{\Gamma}_i = \arg\min_{\Gamma_i} \xi_i \|\Gamma_i\|_1 + \frac{1}{2}\|\mathbf{D}_i\Gamma_i - \hat{\Gamma}_{i-1}\|_2^2 \quad \text{for} \quad i = 1, 2, \ \ldots \ , K,$$

and assign $y = sign\left(f(\hat{\Gamma}_K)\right)$, where

$$f(\hat{\Gamma}_K) = \mathbf{w}^T \hat{\Gamma}_K + \omega.$$

3 First Steps: Shallow Sparsity

3.1 Two-Class (Binary) Setting

Herein, we consider a binary classification setting (i.e., $\mathcal{Y} = \{1, -1\}$) in which *a linear classifier is given to us, being part of the generative model*. This classifier is defined by the couple $(\mathbf{w}, \omega)$, where $\mathbf{w} \in \mathbb{R}^{Nm}$ is a weight vector and ω is a bias term (a scalar). Put formally, the model we shall study in this subsection is given by the definition below.

Definition 5 A convolutional Sparseland signal $\mathbf{X} = \mathbf{D}\Gamma$, $\|\Gamma\|_{0,\infty}^{\mathbf{S}} \leq k$ is said to belong to class $y = 1$ when the linear discriminant function $f(\mathbf{X}) = \mathbf{w}^T\Gamma + \omega$ satisfies $f(\Gamma) > 0$, and $y = -1$ otherwise.

The expression $\mathbf{w}^T\Gamma + \omega$ defines a linear discriminant function for which the decision surface $f(\Gamma') = 0$ is a hyperplane in the feature domain (but not in the signal domain), where Γ' is a point that lies on the decision boundary. As such, one can express the distance from the decision boundary *in the feature domain* as $\mathcal{O}_{\mathcal{B}}(\mathbf{X}, y) = yf(\Gamma) = y\left(\mathbf{w}^T\Gamma + \omega\right)$, where the subscript $\mathcal{B}$ stands for Binary. Notice that the larger the value of $\mathcal{O}_{\mathcal{B}}(\mathbf{X}, y)$, the larger the distance to the decision boundary, as it is defined by $\mathcal{O}_{\mathcal{B}}(\mathbf{X}', y) = 0$. Following this rational, $\mathcal{O}_{\mathcal{B}}(\mathbf{X}, y)$ is often termed the *score* or the *output margin*. The measure $\mathcal{O}_{\mathcal{B}}(\mathbf{X}, y)$ is $\mathbf{X}$-dependent, and thus, we have an interest in its extreme value,

$$\mathcal{O}_{\mathcal{B}}^* = \min_{\{\mathbf{X}^j, y^j\}_j} \mathcal{O}_{\mathcal{B}}(\mathbf{X}^j, y^j),$$

being a property of our data and the classifier's parameters, making our claims universal and not signal specific. As will be shown hereafter, classification robustness is directly related to this quantity. Moreover, we emphasize that there are two margins involved in our analysis: (i) the above-described input data margin, which cannot be controlled by any learning scheme, and (ii) the margin that a classifier obtains when operating on a perturbed input signal, resulting in the evaluated representation $\hat{\Gamma}$. The results in this work rely on these two measures, as we aim to make sure that the former margin is not diminished by the practical classifier design or the adversarial noise.

This takes us naturally to the adversarial setting. The problem we consider is defined as follows:

Definition 6 For a signal $\mathbf{Y} = \mathbf{X} + \mathbf{E}$ with a true label y, a convolutional dictionary $\mathbf{D}$, a perturbation energy ϵ, and a classifier $(\mathbf{w}, \omega)$, define the binary classification algorithm as:

[1] Note that in this scheme, the number of iterations for each BP pursuit stage is implicit, hidden by the number of loops to apply. More on this is given in later sections.

$$\text{Solve} \quad \hat{\boldsymbol{\Gamma}} = \arg\min_{\boldsymbol{\Gamma}} \|\boldsymbol{\Gamma}\|_{0,\infty}^{\mathbf{s}}$$
$$\text{s.t.} \quad \|\mathbf{Y} - \mathbf{D}\boldsymbol{\Gamma}\|_2 \leq \epsilon$$
$$\text{and assign } \hat{y} = sign\left(f(\hat{\boldsymbol{\Gamma}})\right)$$

where $f(\hat{\boldsymbol{\Gamma}}) = \mathbf{w}^T\hat{\boldsymbol{\Gamma}} + \omega$.

Notice that the signal is assigned to the correct class if $sign(f(\hat{\boldsymbol{\Gamma}})) = y$, or, equivalently when $\mathcal{O}_{\mathcal{B}}(\mathbf{Y}, y) = yf(\hat{\boldsymbol{\Gamma}}) = y(\mathbf{w}^T\hat{\boldsymbol{\Gamma}} + \omega) > 0$. In words, the pursuit/sparse coding step projects the perturbed signal $\mathbf{Y}$ onto the model by estimating the representation $\hat{\boldsymbol{\Gamma}}$, which in turn is fed to a classifier as formulated by $f(\hat{\boldsymbol{\Gamma}})$. In the remaining of this paper, we shall study the conditions on $\mathbf{X}$, $\mathbf{D}$ and ϵ which guarantee that $\mathcal{O}_{\mathcal{B}}(\mathbf{Y}, y) > 0$, i.e., the input signal $\mathbf{Y}$ will be assigned to the correct class despite of the adversarial perturbation and the limitations of a specific pursuit algorithm. The model assumptions that we are considering allow us to reveal the underlying characteristics of the signal (e.g., the properties of the dictionary and the sparsity level) that are crucial for a successful prediction. Put differently, *we aim to reveal the ingredients that one should consider when designing a robust classification system. Notice that we concentrate on the inference stage only and do not analyze the learning part.* In fact, similarly to [8], we see this as an advantage since it keeps the generality of our findings.

Let us start our discussion by first studying the stability of the binary classification algorithm to noise:

Theorem 7 (Stable binary classification of the CSC model) *Suppose we are given a CSC signal* $\mathbf{X} = \mathbf{D}\boldsymbol{\Gamma}$, $\|\boldsymbol{\Gamma}\|_{0,\infty}^{\mathbf{s}} \leq k$, *contaminated with perturbation* $\mathbf{E}$ *to create the signal* $\mathbf{Y} = \mathbf{X}+\mathbf{E}$, *such that* $\|\mathbf{E}\|_2 \leq \epsilon$. *Suppose further that* $\mathcal{O}_{\mathcal{B}}^* > 0$ *and denote by* $\hat{\boldsymbol{\Gamma}}$ *the solution of the* $P_{0,\infty}^{\boldsymbol{\mathcal{E}}}$ *problem (see Eq. (1)). Assuming that* $\delta_{2k} < 1 - \left(\frac{2\|\mathbf{w}\|_2\epsilon}{\mathcal{O}_{\mathcal{B}}^*}\right)^2$, *then* $sign(f(\boldsymbol{\Gamma})) = sign(f(\hat{\boldsymbol{\Gamma}}))$.

Considering the more conservative bound that relies on $\mu(\mathbf{D})$, *and assuming that*

$$\|\boldsymbol{\Gamma}\|_{0,\infty}^{\mathbf{s}} < k = \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D})}\left[1 - \left(\frac{2\|\mathbf{w}\|_2\epsilon}{\mathcal{O}_{\mathcal{B}}^*}\right)^2\right]\right),$$

then $sign(f(\boldsymbol{\Gamma})) = sign(f(\hat{\boldsymbol{\Gamma}}))$.

The proof of this theorem and the remaining ones are given in the Appendix. Among various implications, the above theorem shows the effect of $\mathbf{D}$ and its properties on the stability of the classifier. A dictionary with $\delta_{2k} \ll 1$ tends to preserve the distance between any pair of *locally* k-sparse vectors (defined by the $\ell_{0,\infty}$ norm), which turns to be crucial for robust classification. The benefit of switching from the SRIP to $\mu(\mathbf{D})$ is that the latter is trivial to compute, but with the cost of weakening the result. The expected and somewhat unsurprising conclusion of Theorem 7 is that the score of the classifier plays a key role for a stable classification—the larger the distance to the decision boundary in the feature space the more robust the classifier is to noise. This stands in a line with the conclusion of [8]. Another alignment with previous work (e.g., [3,23]) is the effect of the norm of $\mathbf{w}$. Notice that in the proposed theorem, $\|\mathbf{w}\|_2$ is multiplied by the noise energy ϵ and both have a negative effect on the stability. As a result, one should promote a weight vector of low energy (this is often controlled via a weight decay regularization) as it is capable of increasing the robustness of the sparsity-inspired classification model to noise.

The added value of Sparseland is that a "well behaved" dictionary, having a small SRIP constant or low mutual coherence, is the key for *stable recovery*, which, in turn, would increase the robustness of the classier. Interestingly, implied from the proof of the obtained results is the fact that *a successful classification can be achieved without recovering the true support* (i.e., the locations of nonzeros in $\boldsymbol{\Gamma}$). This might be counter intuitive, as the support defines the subspace that the signal belongs to. That is, even if the noise in $\mathbf{Y}$ leads to an estimation $\hat{\boldsymbol{\Gamma}}$ that belongs to slightly different subspace than the one of $\mathbf{X}$, the input signal could be still classified correctly as long as the dimension of the subspace that it belongs to is small enough (the sparsity constraint).

Our results and perspective on the problem are very different from previous work that studies the robustness to adversarial noise. Fawzi et al. [8] suggested a measure for the difficulty of the classification task, where in the linear setting this is defined as the distance between the means of the two classes. Our results differ from these as we heavily rely on a generative model, and so are capable of linking the intrinsic properties of the signal—its sparsity and filters' design—to the success of the classification task. This enables us to suggest ways to increase the desired robustness.

A recent work [7] relies on a generative model (similar to ours) that transforms normally-distributed random representation to the signal domain. Their goal was to prove that there exists an upper bound on the robustness that no classifier can exceed. Still, the authors of [7] did not study the effect of the filters nor the network's depth (or the sparsity). Their analysis is very different from ours as we put an emphasis on the stability of sparsity-inspired model and its properties.

As already mentioned, the margin of the data has an impact on the stability as well. Denoting by $\mathbf{X}'$ a point on the boundary decision, the work reported in [23] connected the input margin, given by $\|\mathbf{X}' - \mathbf{X}\|_2$, to the output distance $\|f(\boldsymbol{\Gamma}') - f(\boldsymbol{\Gamma})\|_2 = \|f(\boldsymbol{\Gamma})\|_2$ through the Jacobian of the classifier $f(\cdot)$. This connection is of importance as the input margin is directly related to the generalization error. In the scope of this paper, the distance between the sig-

nal and its cleaned version in the input space is nothing but the energy of the noise perturbation $\|\mathbf{X}-\mathbf{Y}\|_2=\epsilon$. This, in turn, is linked to the score of the classifier distance $\|f(\hat{\mathbf{\Gamma}})-f(\mathbf{\Gamma})\|_2=\|\mathbf{w}^T\hat{\mathbf{\Gamma}}-\mathbf{w}^T\mathbf{\Gamma}\|_2\leq\|\mathbf{w}\|_2\|\hat{\mathbf{\Gamma}}-\mathbf{\Gamma}\|_2$ (refer to the proof of Theorem 7 for more details).

We should clarify the term stability used in this section: This refers to *any solver of the pursuit task* that satisfies the following two constraints (i) $\|\mathbf{D}\mathbf{\Gamma}-\mathbf{Y}\|_2\leq\epsilon$, and (ii) $\|\mathbf{\Gamma}\|_{0,\infty}^{\mathbf{s}}=k$. Later on, we shall refer to actual pursuit methods and extend this result further. Concretely, suppose we run the thresholding pursuit (or the BP) to estimate the representation $\hat{\mathbf{\Gamma}}$ of a given $\mathbf{Y}$. Then, we feed the obtained sparse vector to our linear classifier and hope to assign the input signal to its true label, despite the existence of the adversarial perturbation. Can we guarantee a successful classification in such cases? While this question can be addressed in the CSC case, we shall study the more general multilayer model. Before doing so, however, we expand the above to the multi-class setting.

3.2 Multi-class Setting

In order to provide a complete picture on the factors that affect the stability of the classification procedure, we turn to study multi-class linear classifiers. Formally, the discriminant function is described by the following definition:

Definition 8 A CSC signal $\mathbf{X}=\mathbf{D}\mathbf{\Gamma}$, $\|\mathbf{\Gamma}\|_{0,\infty}^{\mathbf{s}}\leq k$, is said to belong to class $y=u$ if the linear discriminant function satisfies $\forall v\neq u\ \ f_u(\mathbf{\Gamma})=\mathbf{w}_u^T\mathbf{\Gamma}+\omega_u>\mathbf{w}_v^T\mathbf{\Gamma}+\omega_v=f_v(\mathbf{\Gamma})$, where u stands for the index of the true class, and v is in the range of $[1,L]\backslash v$.

Analogously to the binary case, the decision boundary between class u and v is given by $f_u(\mathbf{\Gamma}')=f_v(\mathbf{\Gamma}')$. Therefore, we formalize the distance to the decision boundary of the class $y=u$ in the feature space as $\mathcal{O}_{\mathcal{M}}(\mathbf{X},y)=\min_{v:v\neq u}f_u(\mathbf{\Gamma})-f_v(\mathbf{\Gamma})$, which measures the distance between the classification result of the u-classifier to the rest $L-1$ ones for a given point $\mathbf{X}$. Similarly to the binary setting, we obtain the minimal distance over all the classes and examples by

$$\mathcal{O}_{\mathcal{M}}^{*}=\min_{\{\mathbf{X}^j,y^j\}_j}\mathcal{O}_{\mathcal{M}}(\mathbf{X}^j,y^j),$$

where we assume that $\mathcal{O}_{\mathcal{M}}(\mathbf{X}^j,y^j)>0$. Notice that this assumption aligns with the practice, as the common setting in CNN-based classification assumes that a perfect fit of the data during training is possible. Another measure that will be found useful for our analysis is the distance between the weight vectors. Put formally, we define the multi-class weight matrix $\mathbf{W}$ of size $mN\times L$ as $\mathbf{W}=[\mathbf{w}_1;\mathbf{w}_2;\cdots;\mathbf{w}_L]$, which stores the weight vectors as its columns. The following measure quantifies the mutual Euclidean distance between the classifiers, given by

$$\phi(\mathbf{W})=\max_{u\neq v}\|\mathbf{w}_u-\mathbf{w}_v\|_2.$$

The analogous of this measure in the binary classification, when $L=2$, is the norm of the classifier being $\|\mathbf{w}\|_2$, as in this case one can define $\mathbf{w}_1=-\mathbf{w}_2=\frac{1}{2}\mathbf{w}$.

Theorem 9 (Stable Multi-Class Classification of the CSC Model): *Suppose we are given a CSC signal* $\mathbf{X}=\mathbf{D}\mathbf{\Gamma}$, $\|\mathbf{\Gamma}\|_{0,\infty}^{\mathbf{s}}\leq k$, *contaminated with perturbation* $\mathbf{E}$ *to create the signal* $\mathbf{Y}=\mathbf{X}+\mathbf{E}$, *such that* $\|\mathbf{E}\|_2\leq\epsilon$. *Suppose further that* $f_u(\mathbf{\Gamma})=\mathbf{w}_u^T\mathbf{\Gamma}+\omega_u$ *correctly assigns* $\mathbf{X}$ *to class* $y=u$. *Suppose further that* $\mathcal{O}_{\mathcal{M}}^{*}>0$, *and denote by* $\hat{\mathbf{\Gamma}}$ *the solution of the* $P_{0,\infty}^{\boldsymbol{\mathcal{E}}}$ *problem. Assuming that* $\delta_{2k}<1-\left(\frac{2\phi(\mathbf{W})\epsilon}{\mathcal{O}_{\mathcal{M}}^{*}}\right)^2$, *then* $\mathbf{Y}$ *will be assigned to the correct class.*

Considering the more conservative bound that relies on $\mu(\mathbf{D})$ *and assuming that*

$$\|\mathbf{\Gamma}\|_{0,\infty}^{\mathbf{s}}<k=\frac{1}{2}\left(1+\frac{1}{\mu(\mathbf{D})}\left[1-\left(\frac{2\phi(\mathbf{W})\epsilon}{\mathcal{O}_{\mathcal{M}}^{*}}\right)^2\right]\right),$$

then $\mathbf{Y}$ *will be classified correctly.*

As one might predict, the same ingredients as in Theorem 7 (coherence of $\mathbf{D}$ or its SRIP) play a key role here as well. Moreover, in the two-class setting $\phi(\mathbf{W})=\|\mathbf{w}\|_2$, and so the two theorems align. The difference becomes apparent for $L>2$, when the mutual Euclidean distance between the different classifiers influences the robustness of the system to noise. In the context of multi-class support vector machine, it was shown [4] that the separation margin between the classes u and v is $2/\|\mathbf{w}_u-\mathbf{w}_v\|_2$. This observation motivated the authors of [4] to minimize the distance $\|\mathbf{w}_u-\mathbf{w}_v\|_2,\forall u\neq v$ during the training phase. Interestingly, this quantity serves our bound as well. Notice that our theorem also reveals the effect of the number of classes on the robustness. Since $\phi(\mathbf{W})$ measures the maximal distance between the L weight vectors, it is a monotonically increasing function of L and thereby stiffening our conditions for a successful classification. This phenomenon was observed in practice, indicating that it is easier to "fool" the classifier when the number of classes is large, compared to a binary setting [7].

4 Robustness Bounds for CNN

We now turn to extend the above results to a multilayer setting, and this way shed light on the stability of classic CNN architectures. For simplicity, we return to the binary setting,

as we have seen that the treatment of multiple classes has a similar analysis. We commence by defining the model we aim to analyze in this part:

Definition 10 An ML-CSC signal $\mathbf{X}$ (see Definition 1) is said to belong to class $y = 1$ when the linear discriminant function $f(\mathbf{X}) = \mathbf{w}^T\Gamma_K + \omega$ satisfies $f(\Gamma_K) > 0$, and $y = -1$ otherwise.

Notice that the classifier operates on the representation of the last layer, and so the definition of the signal-dependent score $\mathcal{O}_{\mathcal{B}}(\mathbf{X}, y)$ and the universal $\mathcal{O}^*_{\mathcal{B}}$ are similar to the ones defined in Sect. 3.1. We now turn to the noisy regime, where the adversarial perturbation kicks in:

Definition 11 For a corrupted ML-CSC signal $\mathbf{Y} = \mathbf{X} + \mathbf{E}$ with a true label y, convolutional dictionaries $\{\mathbf{D}_i\}_{i=1}^K$, a perturbation energy ϵ, sparsity levels $\boldsymbol{\lambda}$, and a classifier $(\mathbf{w}, \omega)$, define the multilayer binary classification algorithm as:

find $\{\hat{\Gamma}\}_{i=1}^K$ by solving the DCP$^{\mathcal{E}}_{\boldsymbol{\lambda}}$ problem;

and assign $y = sign\left(f(\hat{\Gamma}_K)\right)$,

where $f(\hat{\Gamma}_K) = \mathbf{w}^T\hat{\Gamma}_K + \omega$.

Above, an accurate classification is achieved when $sign(f(\hat{\Gamma}_K)) = sign(f(\Gamma_K))$. The stability of the multilayer binary classification algorithm can be analyzed by extending the results of [21] as presented in Sect. 2. Therefore, rather than analyzing the properties of the problem, in this section we concentrate on specific algorithms that serve the ML-CSC model—the L-THR algorithm (i.e., the forward pass of CNN), and its L-BP counterpart. To this end, differently from the previous theorems that we presented, we will assume that the noise is locally bounded (rather than globally[2]) as suggested in [21]. Put formally, we use the $\ell_{2,\infty}$-norm to measure the energy of the noise in a vector $\mathbf{E}$, denoted by $\|\mathbf{E}\|^{\mathbf{p}}_{2,\infty}$, which is defined to be the maximal energy of a n_1-dimensional patch extracted from it.

Theorem 12 (Stable binary classification of the L-THR) *Suppose we are given an ML-CSC signal $\mathbf{X}$ contaminated with perturbation $\mathbf{E}$ to create the signal $\mathbf{Y} = \mathbf{X} + \mathbf{E}$, such that $\|\mathbf{E}\|^{\mathbf{p}}_{2,\infty} \le \epsilon_0$. Denote by $|\Gamma_i^{min}|$ and $|\Gamma_i^{max}|$ the lowest and highest entries in absolute value in the vector Γ_i, respectively. Suppose further that $\mathcal{O}^*_{\mathcal{B}} > 0$ and let $\{\hat{\Gamma}_i\}_{i=1}^K$ be the set of solutions obtained by running the layered soft thresholding algorithm with thresholds $\{\beta_i\}_{i=1}^K$, i.e., $\hat{\Gamma}_i = \mathcal{S}_{\beta_i}(\mathbf{D}_i^T\hat{\Gamma}_{i-1})$ where $\mathcal{S}_{\beta_i}$ is the soft thresholding operator and $\hat{\Gamma}_0 = \mathbf{Y}$. Assuming that $\forall\ 1 \le i \le K$*

a. $\|\Gamma_i\|^{\mathbf{S}}_{0,\infty} < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D}_i)}\frac{|\Gamma_i^{min}|}{|\Gamma_i^{max}|}\right) - \frac{1}{\mu(\mathbf{D}_i)}\frac{\epsilon_{i-1}}{|\Gamma_i^{max}|}$;

b. *The threshold β_i is chosen according to*

$$|\Gamma_i^{min}| - C_i - \epsilon_{i-1} > \beta_i > \|\Gamma_i\|^{\mathbf{S}}_{0,\infty}\mu(\mathbf{D}_i)|\Gamma_i^{max}| + \epsilon_{i-1},$$

where

$$C_i = (\|\Gamma_i\|^{\mathbf{S}}_{0,\infty} - 1)\mu(\mathbf{D}_i)|\Gamma_i^{max}|,$$
$$\epsilon_i = \sqrt{\|\Gamma_i\|^{\mathbf{p}}_{0,\infty}}\left(\epsilon_{i-1} + C_i + \beta_i\right);$$

and

c. $\mathcal{O}^*_{\mathcal{B}} > \|\mathbf{w}\|_2\sqrt{\|\Gamma_K\|_0}\left(\epsilon_{K-1} + C_K + \beta_K\right)$,

then $sign(f(\hat{\Gamma}_K)) = sign(f(\Gamma_K))$.

Some of the ingredients of the above theorem are similar to the previous results, but there are several major differences. First, while the discussion in Sect. 3 concentrated on the stability of the problem, here we get that the forward pass is an algorithm that is capable of recovering the true support of the representations Γ_i. Still, this perfect recovery does not guarantee a successful classification, as the error in the deepest layer should be smaller than $\mathcal{O}^*_{\mathcal{B}}$. Second, the forward pass is sensitive to the contrast of the nonzero coefficients in Γ_i (refer to the ratio $|\Gamma_i^{\min}|/|\Gamma_i^{\max}|$), which is a well-known limitation of the thresholding algorithm [6]. Third, we see that without a careful regularization (e.g., promoting the coherence to be small) the noise can be easily amplified throughout the layers (ϵ_i increases as a function of i). Practitioners refer to this as the error amplification effect [16].

In fact, a similar regularization force is used in Parseval networks [20] to increase the robustness of CNN to adversarial perturbations. These promote the convolutional layers to be (approximately) Parseval tight frames, which are extensions of orthogonal matrices to the non-square case. Specifically, the authors in [20] suggested to promote the spectral norm of the weight matrix $\mathbf{D}_i^T$ to be close to 1, i.e., $\|\mathbf{D}_i\mathbf{D}_i^T - \mathbf{I}\|_2^2$, where $\mathbf{I}$ is the identity matrix. This regularization encourages the average coherence of $\mathbf{D}_i$ to be small. As our analysis suggests, it was shown that such a regularization significantly improves the robustness of various models to adversarial examples.

Suppose that our data emerge from the ML-CSC model, can we offer an alternative architecture that is inherently better in handling adversarial perturbations? The answer is positive and is given in the form of the L-BP that was suggested and analyzed in [21]. Consider an ML-CSC model of depth two, the L-BP algorithm suggests estimating Γ_1 and Γ_2 by solving a cascade of basis pursuit problems. The first stage of this method provides an estimate for Γ_1 by minimizing

[2] Locally bounded noise results exist for the CSC as well [22], and can be leveraged in a similar fashion.

$$\hat{\Gamma}_1 = \arg\min_{\Gamma_1} \|\mathbf{Y} - \mathbf{D}_1\Gamma_1\|_2^2 + \xi_1\|\Gamma_1\|_1.$$

Then, an approximation for the deeper representation Γ_2 is given by

$$\hat{\Gamma}_2 = \arg\min_{\Gamma_2} \|\hat{\Gamma}_1 - \mathbf{D}_2\Gamma_2\|_2^2 + \xi_2\|\Gamma_2\|_1.$$

Finally, the recovered $\hat{\Gamma}_2$ is fed into a classifier, resulting in the predicted label.

In what follows we build upon the analysis in [21] and show how our theory aligns with the increased stability that was empirically observed by replacing the L-THR algorithm with the L-BP [18]:

Theorem 13 (Stable Binary Classification of the L-BP): *Suppose we are given an ML-CSC signal* $\mathbf{X}$ *that is contaminated with noise* $\mathbf{E}$ *to create the signal* $\mathbf{Y} = \mathbf{X} + \mathbf{E}$, *such that* $\|\mathbf{E}\|_{2,\infty}^{\mathbf{P}} \leq \epsilon_0$. *Suppose further that* $\mathcal{O}_{\mathcal{B}}^* > 0$, *and let* $\{\hat{\Gamma}_i\}_{i=1}^K$ *be the set of solutions obtained by running the L-BP algorithm with parameters* $\{\xi_i\}_{i=1}^K$, *formulated as* $\hat{\Gamma}_i = \arg\min_{\Gamma_i} \xi_i\|\Gamma_i\|_1 + \frac{1}{2}\|\mathbf{D}_i\Gamma_i - \hat{\Gamma}_{i-1}\|_2^2$, *where* $\hat{\Gamma}_0 = \mathbf{Y}$. *Assuming that* $\forall\ 1 \leq i \leq K$,

a) $\|\Gamma_i\|_{0,\infty}^{\mathbf{S}} \leq \frac{1}{3}\left(1 + \frac{1}{\mu(\mathbf{D}_i)}\right)$;
b) $\xi_i = 4\epsilon_{i-1}$,
where $\epsilon_i = \|\mathbf{E}\|_{2,\infty}^{\mathbf{P}} \cdot 7.5^i \prod_{j=1}^{i}\sqrt{\|\Gamma_j\|_{0,\infty}^{\mathbf{P}}}$; *and*
c) $\mathcal{O}_{\mathcal{B}}^* > 7.5\|\mathbf{w}\|_2\sqrt{\|\Gamma_K\|_0}\,\epsilon_K$,

then $sign(f(\hat{\Gamma}_K)) = sign(f(\Gamma_K))$.

The proof can be derived by relying on the steps of Theorem 12, combined with Theorem 12 from [21]. Note that the conditions for the stable classification of the L-BP are not influenced by the ratio $|\Gamma_i^{\min}|/|\Gamma_i^{\max}|$. Moreover, the condition on the cardinality of the representations in the L-BP case is less strict than the one of the L-THR. As such, while the computational complexity of the BP algorithm is higher than the thresholding one, the former is expected to be more stable than the latter. This theoretical statement is supported in practice [18]. Note that both methods suffer from a similar problem—the noise is propagated thorough the layers. A possible future direction to alleviate this effect could be to harness the projection (onto the ML-CSC model) algorithm [25], whose bound is not cumulative across the layers.

5 Numerical Experiments

Our study of the stability to bounded noise, in particular Theorems 12 and 13, introduces a better guarantee for the L-BP, when compared to the well-known L-THR architecture (=CNN). In this section, we aim to numerically corroborate these findings by exploring the actual robustness to adversarial noise of these two architectures. We achieve this by introducing two sets of experiments: (i) We start with a toy example using synthetically generated data and show the actual behavior of the L-THR and the L-BP versus their theoretical bounds; and (ii) we proceed by testing these two architectures and exploring their robustness to adversarial noise on actual data, exposing the superiority of the L-BP.

As described in [1,21,24] and depicted in Fig. 4b, the L-BP is implemented by unfolding the projected gradient steps of the iterative thresholding algorithm. By setting the number of unfolding iterations to zero, the L-BP becomes equivalent to the L-THR architecture. Note that both pursuit methods contain the same number of filters, and those are of the same dimensions. Therefore, the same number of free parameters govern both their computational paths. Nonetheless, more unfoldings in the L-BP lead to a higher computational complexity when compared to L-THR.

5.1 Synthetic Experiments

We start our experimental section with a *toy* example using synthetically generated data, where we have complete access to the generative model and its parameters. This allows us to (i) compute the theoretical bounds on the permitted noise and (ii) compare these predictions with an evaluation of the practical behavior. Our emphasis in this part is on the single-layer thresholding and basis pursuit classifiers, as synthesizing signals from the multilayered model is far more challenging. Our goal is to show the differences between the theoretical bounds and the measured empirical stability. For completeness, we include experiments with both an undercomplete (having less atoms than the signal dimension) and overcomplete (where the dictionary is redundant) dictionaries.

As already mentioned, we consider the bounds from Theorems 12 and 13 with $K = 1$, corresponding to a one hidden layer neural network with a nonconvolutional (fully connected) dictionary. The two following corollaries describe the bounds of the L-THR and the L-BP in such simplified case.

Corollary 1 (Stability of one hidden layer L-THR) *Suppose that* $\mathbf{Y} = \mathbf{D}\Gamma + \mathbf{E}$, *where* $\mathbf{D}$ *is a general dictionary with normalized atoms,* $\|\mathbf{E}\|_2 \leq \epsilon$, *and* $(\mathbf{w}, \omega)$ *is the linear classifier. Suppose also that*

$$\|\Gamma\|_0 \leq \frac{1}{2}\left(1 + \frac{|\Gamma_{\min}|}{|\Gamma_{\max}|}\frac{1}{\mu(\mathbf{D})}\right) - \frac{\epsilon}{\mu(\mathbf{D})\,|\Gamma_{\max}|},$$

and that the threshold β *set to satisfy:*

$$\|\Gamma\|_0\,\mu(\mathbf{D})\,|\Gamma_{\max}| + \epsilon < \beta$$

$$< |\Gamma_{\min}| - (\|\Gamma\|_0 - 1)\mu(\mathbf{D})\,|\Gamma_{\max}| - \epsilon.$$

Then, the support of $\hat{\Gamma}^{THR}$ *is contained in the support of* Γ, *and*

$$\left\|\hat{\Gamma}^{THR} - \Gamma\right\|_2 \leq \sqrt{\|\Gamma\|_0}\,(\epsilon + (\|\Gamma\|_0 - 1)\mu(\mathbf{D})\,|\Gamma_{\max}| + \beta)\,.$$

Therefore, as long as

$$\epsilon < \frac{\mathcal{O}_{\mathcal{B}}}{\sqrt{\|\Gamma\|_0}\|\mathbf{w}\|_2} - (\|\Gamma\|_0 - 1)\mu(\mathbf{D})\,|\Gamma_{\max}| - \beta, \tag{2}$$

the classification is accurate, i.e., $sign(f(\hat{\Gamma}^{THR})) = sign\,(f(\Gamma))$.

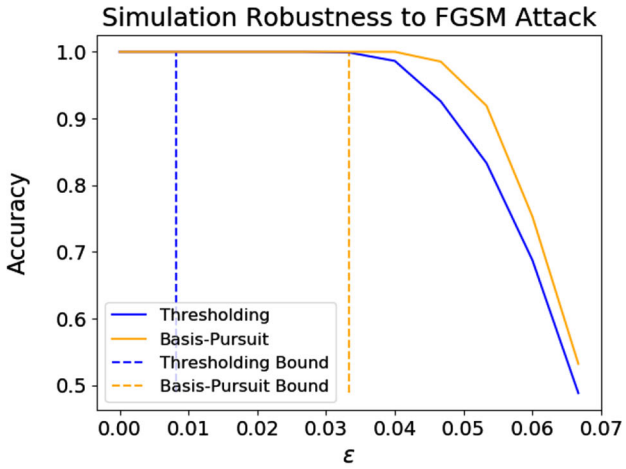
Corollary 2 (Stability of one hidden layer L-BP) *Suppose that* $\mathbf{Y} = \mathbf{D}\Gamma + \mathbf{E}$, *where* $\mathbf{D}$ *is a general dictionary with normalized atoms,* $\|\mathbf{E}\|_2 \leq \epsilon$, *and* $(\mathbf{w}, \omega)$ *is a linear classifier. Suppose also that* $\|\Gamma\|_0 \leq \frac{1}{3}\left(1 + \frac{1}{\mu(\mathbf{D})}\right)$, *and that the Lagrangian multiplier is set to* $\xi = 4\epsilon$. *Then, the support of* $\hat{\Gamma}^{BP}$ *is contained in the support of* Γ, *and* $\left\|\hat{\Gamma}^{BP} - \Gamma\right\|_2 \leq 7.5\epsilon$. *Therefore, as long as*

$$\epsilon < \frac{\mathcal{O}_{\mathcal{B}}}{7.5\,\|\Gamma\|_0\,\|\mathbf{w}\|_2}, \tag{3}$$

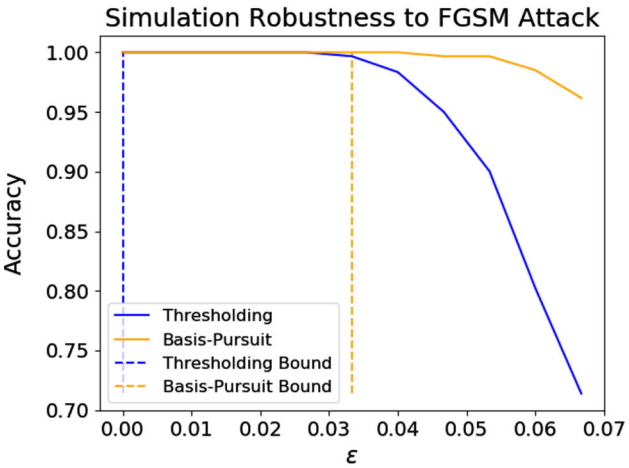
the classification is accurate, i.e., $sign(f(\hat{\Gamma}^{BP})) = sign\,(f(\Gamma))$.

Figure 5 presents the theoretical bounds on the adversarial noise amplitude ϵ for the Thresholding (Eq. (2)) and the basis pursuit (Eq. (3)) classifiers in dash lines. It also shows the empirical stability to adversarial noise under the FGSM (fast gradient sign method) attack [10]. In these simulations, we generate a random normalized and unbiased ($\omega = 0$) classifier $\mathbf{w}$, and a random dictionary with normalized atoms and with a low mutual coherence. Then, we randomly produce sparse representations with four nonzeros in the range $\left[|\Gamma^{\min}|, |\Gamma^{\max}|\right] = [1, 2]$. In order to create a margin $\mathcal{O}_{\mathcal{B}}$ of 1, we project Γ on the classifier $\mathbf{w}$ and keep only the representations satisfying $\left|\mathbf{w}^T\Gamma\right| \geq \mathcal{O}_{\mathcal{B}} = 1$. Figure 5a presents the undercomplete case with $\mathbf{D} \in \mathbb{R}^{100\times 40}$. One can draw three important conclusions from this result: 1) The theoretical stability bound for the BP is better than the THR one; 2) the empirical performance of the BP and THR aligns with the theoretical predictions; and 3) the bounds are not tight due to the worst-case assumptions used in our work. Note that in this experiment the performance of the two methods is quite close—this is due to very low mutual coherence of the chosen dictionary.

This motivates the next experiment, in which we examine a more challenging setting that relies on an overcomplete dictionary. Figure 5b demonstrates this case with $\mathbf{D} \in \mathbb{R}^{100\times 150}$



(a) Simulation with an undercomplete dictionary.



(b) Simulation with an overcomplete dictionary.

Fig. 5 Accuracy of the THR and the BP versus adversarial noise level, computed on synthetic data. Dashed lines: theoretical bounds; solid lines: empirical performance

and with representations having the same properties as before. In this case, the thresholding bound collapses to zero as the mutual coherence is too high, and as can be seen, the practical difference between the THR and the BP classifiers becomes apparent.

5.2 Real Data Experiments

The goal of the following set of experiments is to show that moving from the traditional feed-forward network (i.e., L-THR) to L-BP can potentially improve stability, not only for simulated data (where the dictionaries and the signals are generated exactly to meet the theorem conditions), but also for real data such as MNIST and CIFAR images. Note that one could wonder whether these images we are about to work with belong to the (possibly multilayered) sparse representation model. Our approach for answering this question is to

impose the model on the data and see how the eventual pursuit (such as the forward pass) performs. This line of reasoning stands behind many papers that took the sparse representation model (or any of its many variants) and deployed it to true data in order to address various applications.

The networks we are about to experiment with are obtained as unfoldings of the L-THR (Fig. 4a) and the L-BP (Fig. 4b) pursuit algorithms, and each is trained in a supervised fashion using back-propagation for best classification performance. Our tested architectures are relatively simple and use a small number of parameters in order to isolate the effect of their differences [2,24].

Ideally, in order to demonstrate Theorems 12 and 13, one should require that the same set of dictionaries is used by the two architectures, in a way that fits our multilayered model assumptions. However, such setup leads to various difficulties. First, as obtaining these dictionaries calls for training, we should decide on the loss to use. Trained for representation error, these architectures would lead to inferior classification performance that would render our conclusions irrelevant. The alternative of training for classification accuracy would lead to two very different sets of dictionaries, violating the above desire. In addition, as we know from the analysis in [11], the learned dictionaries are strongly effected by the finite and small number of unfoldings of the pursuit. In the experiments, we report hereafter we chose to let each architecture (e.g., pursuit) to learn the best set of dictionaries for its classification result.

Given the two pre-trained networks, our experiments evaluate the stability by designing noise attacks using the fast gradient sign method (FGSM) [10] with an increasing amplitude ϵ. We perform this evaluation on three popular datasets—MNIST [15], CIFAR-10 [12] and CIFAR-100 [12]. For the MNIST case, we construct an ML-CSC model composed of 3 convolutional layers with 64, 128 and 512 filters, respectively, and kernel sizes of 6×6, 6×6 and 4×4, respectively, with stride of 2 in the first two layers. In addition, the output of the ML-CSC model is followed by a fully connected layer producing the final estimate. Training is done with the stochastic gradient descent (SGD), with a mini-batch size of 64 samples, learning rate of 0.005 and a momentum weight of 0.9. We decrease the learning rate tenfold every 30 epochs.

For CIFAR-10 and CIFAR-100, we define an ML-CSC model as having 3 convolutional layers with 32, 64 and 128 filters, respectively, and kernel sizes of 4×4 with stride of 2. In addition, we used a classifier function as a CNN with 4 layers where the first 3 layers are convolutional and the last layer is fully connected. This effectively results in a 7 layers architecture, out of which the first three are unfolded in the context of the L-BP scheme. As before, all models are trained with SGD and with a decreasing learning rate.

Figure 6 presents the results for the two architectures and the three datasets. It is clear that the L-BP scheme is consistently more robust to adversarial interference. This evidence is in agreement with the theoretical results we introduced earlier, suggesting that the L-THR is more sensitive to bounded noise. We note again, however, that the theoretical guaran-

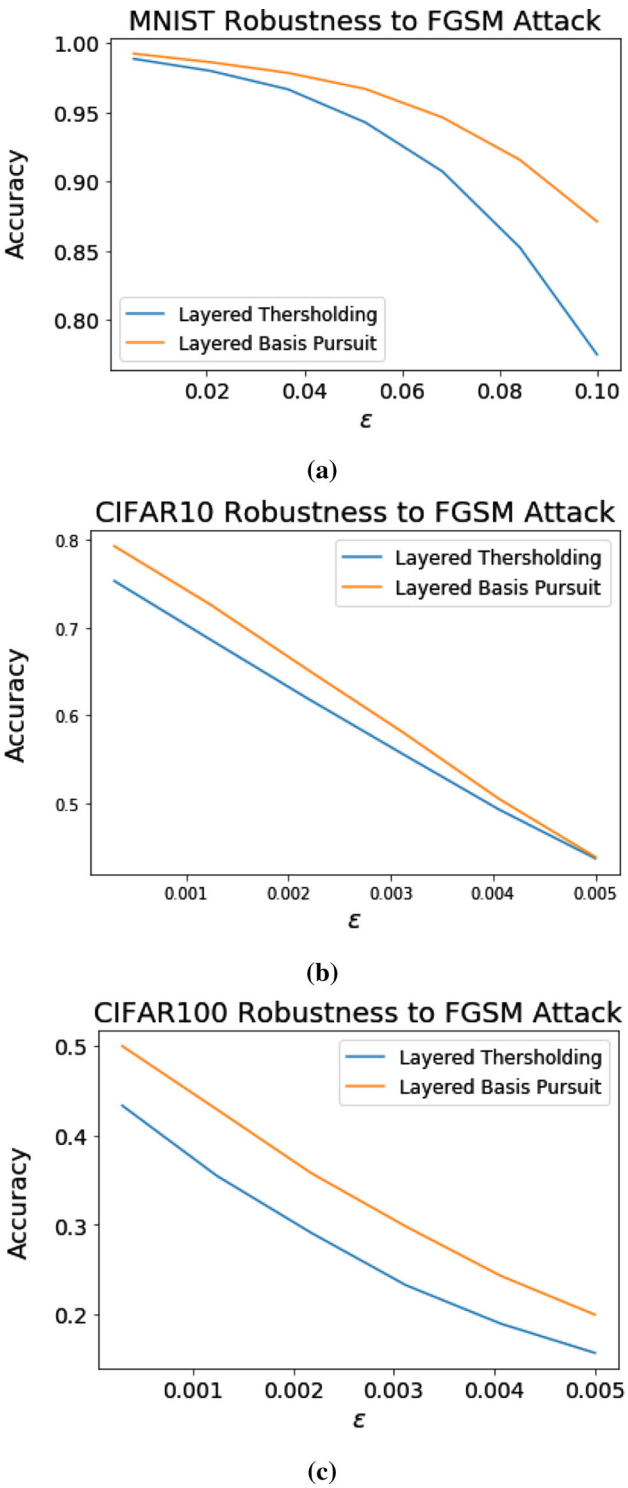


Fig. 6 Comparison of layered thresholding (L-THR) and layered basis pursuit (L-BP) schemes under FGSM attack on **a** MNIST, **b** CIFAR-10 and **c** CIFAR-100 datasets

tees presented earlier are not fully applicable here as the dictionaries of each model are different and as some of the assumptions are violated. For example, the minimal distance between classes $\mathcal{O}^*_{\mathcal{M}}$ is not guaranteed to be nontrivial in real images scenario. However, these experiments do support our earlier analysis about the superiority of the L-BP to handle noise attacks.

6 Conclusions

This paper presents a general theory for the classification robustness when handling sparsity-modeled signals. In the context of CSC, we studied the stability of the classification problem to adversarial perturbations both for the binary- and multi-class settings. Then, we analyzed the stability of a classifier that operates on signals that belong to the ML-CSC model, which was recently shown to be tightly connected to CNN. This leads to a novel analysis of the sensitivity of the classic forward pass algorithm to adversarial perturbations and ways to mitigate its vulnerability (which was empirically validated in [20]). Next, we showed that by relying on the BP algorithm, one can theoretically improve the robustness to such perturbations, a phenomenon that was observed in practice [18].

The bounds obtained are all referring to the case where the dictionaries $\{\mathbf{D}_i\}_{i=1}^K$ and the classification weights $\{(\mathbf{w}_u, \omega_u)\}_{u=1}^L$ are perfectly known, and thus learning is not covered by this theory. As such, the margin for making an error in our work considers only two prime forces. First, the separability of the data, as manifested by $\mathcal{O}^*_{\mathcal{B}}$ (or $\mathcal{O}^*_{\mathcal{M}}$). Second, the chance that our estimated $\mathbf{\Gamma}$ deviates from its true value. This can happen due to noise in the input (getting $\mathbf{Y}$ instead of $\mathbf{X}$), and/or limitation of the pursuit algorithm. Further work is required in order to bring into account distortions in $\mathbf{\Gamma}$ caused by an imperfect estimate of $\mathbf{D}$'s and $\mathbf{w}$'s—this way considering the learning phase as well.

Appendix A: Proof of Theorem 7: Stable Binary Classification of the CSC Model

Theorem 5 (Stable binary classification of the CSC model) *Suppose we are given a CSC signal* $\mathbf{X}$, $\|\mathbf{\Gamma}\|_{0,\infty}^{\mathbf{S}} \le k$, *contaminated with perturbation* $\mathbf{E}$ *to create the signal* $\mathbf{Y} = \mathbf{X} + \mathbf{E}$, *such that* $\|\mathbf{E}\|_2 \le \epsilon$. *Suppose further that* $\mathcal{O}^*_{\mathcal{B}} > 0$ *and denote by* $\hat{\mathbf{\Gamma}}$ *the solution of the* $P^{\boldsymbol{\mathcal{E}}}_{0,\infty}$ *problem. Assuming that* $\delta_{2k} < 1 - \left(\frac{2\|\mathbf{w}\|_2\epsilon}{\mathcal{O}^*_{\mathcal{B}}}\right)^2$, *then* $sign(f(\mathbf{X})) = sign(f(\mathbf{Y}))$.

Considering the more conservative bound that relies on $\mu(\mathbf{D})$, *and assuming that*

$$\|\mathbf{\Gamma}\|_{0,\infty}^{\mathbf{S}} < k = \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D})}\left[1 - \left(\frac{2\|\mathbf{w}\|_2\epsilon}{\mathcal{O}^*_{\mathcal{B}}}\right)^2\right]\right),$$

then $sign(f(\mathbf{X})) = sign(f(\mathbf{Y}))$.

Proof Without loss of generality, consider the case where $\mathbf{w}^T\mathbf{\Gamma} + \omega > 0$, i.e., the original signal $\mathbf{X}$ is assigned to class $y = 1$. Our goal is to show that $\mathbf{w}^T\hat{\mathbf{\Gamma}} + \omega > 0$. We start by manipulating the latter expression as follows:

$$\begin{aligned}\mathbf{w}^T\hat{\mathbf{\Gamma}} + \omega &= \mathbf{w}^T\left(\mathbf{\Gamma} + \hat{\mathbf{\Gamma}} - \mathbf{\Gamma}\right) + \omega \\ &= \left(\mathbf{w}^T\mathbf{\Gamma} + \omega\right) + \mathbf{w}^T\left(\hat{\mathbf{\Gamma}} - \mathbf{\Gamma}\right) \\ &\ge \left(\mathbf{w}^T\mathbf{\Gamma} + \omega\right) - \left|\mathbf{w}^T\left(\hat{\mathbf{\Gamma}} - \mathbf{\Gamma}\right)\right| \\ &\ge \left(\mathbf{w}^T\mathbf{\Gamma} + \omega\right) - \left\|\mathbf{w}^T\right\|_2\left\|\hat{\mathbf{\Gamma}} - \mathbf{\Gamma}\right\|_2, \end{aligned} \tag{4}$$

where the first inequality relies on the relation $a + b \ge a - |b|$ for $a > 0$, and the last derivation leans on the Cauchy-Schwarz inequality. Using the SRIP [22] and the fact that both $\|\mathbf{Y} - \mathbf{D}\mathbf{\Gamma}\|_2 \le \epsilon$ and $\|\mathbf{Y} - \mathbf{D}\hat{\mathbf{\Gamma}}\|_2 \le \epsilon$, we get

$$(1 - \delta_{2k})\|\hat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_2^2 \le \|\mathbf{D}\hat{\mathbf{\Gamma}} - \mathbf{D}\mathbf{\Gamma}\|_2^2 \le 4\epsilon^2.$$

Thus,

$$\|\hat{\mathbf{\Gamma}} - \mathbf{\Gamma}\|_2^2 \le \frac{4\epsilon^2}{1 - \delta_{2k}}.$$

Combining the above with Eq. (4) leads to (recall that y = 1):

$$\mathcal{O}_{\mathcal{B}}(\mathbf{Y}, y) = \mathbf{w}^T\hat{\mathbf{\Gamma}} + \omega \ge \mathbf{w}^T\mathbf{\Gamma} + \omega - \|\mathbf{w}\|_2\frac{2\epsilon}{\sqrt{1 - \delta_{2k}}}.$$

Using the definition of the score of our classifier, satisfying

$$0 < \mathcal{O}_{\mathcal{B}}(\mathbf{X}, y) = \mathbf{w}^T\mathbf{\Gamma} + \omega$$

we get

$$\mathcal{O}_{\mathcal{B}}(\mathbf{Y}, y) \ge \mathcal{O}_{\mathcal{B}}(\mathbf{X}, y) - \|\mathbf{w}\|_2\frac{2\epsilon}{\sqrt{1 - \delta_{2k}}}.$$

We are now after the condition for $\mathcal{O}_{\mathcal{B}}(\mathbf{Y}, y) > 0$, and so we require:

$$\begin{aligned}0 &< \mathcal{O}_{\mathcal{B}}(\mathbf{X}, y) - \|\mathbf{w}\|_2\frac{2\epsilon}{\sqrt{1 - \delta_{2k}}} \\ &\le \mathcal{O}^*_{\mathcal{B}} - \|\mathbf{w}\|_2\frac{2\epsilon}{\sqrt{1 - \delta_{2k}}}.\end{aligned}$$

where we relied on the fact that $\mathcal{O}_{\mathcal{B}}(\mathbf{X}, y) \geq \mathcal{O}^*_{\mathcal{B}}$. The above inequality leads to

$$\delta_{2k} < 1 - \left(\frac{2\|\mathbf{w}\|_2 \epsilon}{\mathcal{O}^*_{\mathcal{B}}}\right)^2. \tag{5}$$

Next we turn to develop the condition that relies on $\mu(\mathbf{D})$. We shall use the relation between the SRIP and the mutual coherence [22], given by $\delta_{2k} \geq (2k-1)\mu(\mathbf{D})$ for all $k < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D})}\right)$. Plugging this bound into Eq. (5) results in

$$0 < \mathcal{O}^*_{\mathcal{B}} - \frac{2\|\mathbf{w}\|_2 \epsilon}{\sqrt{1-(2k-1)\mu(\mathbf{D})}},$$

which completes our proof. □

Appendix B: Proof of Theorem 9: Stable Multi-class Classification of the CSC Model

Theorem 7 (Stable multi-class classification of the CSC model) *Suppose we are given a CSC signal* $\mathbf{X}$, $\|\Gamma\|^{\mathbf{S}}_{0,\infty} \leq k$, *contaminated with perturbation* $\mathbf{E}$ *to create the signal* $\mathbf{Y} = \mathbf{X} + \mathbf{E}$, *such that* $\|\mathbf{E}\|_2 \leq \epsilon$. *Suppose further that* $f_u(\mathbf{X}) = \mathbf{w}_u^T \Gamma + \omega_u$ *correctly assigns* $\mathbf{X}$ *to class* $y = u$. *Suppose further that* $\mathcal{O}^*_{\mathcal{M}} > 0$, *and denote by* $\hat{\Gamma}$ *the solution of the* $P_0^{\mathcal{E}}$ *problem. Assuming that* $\delta_{2k} < 1 - \left(\frac{2\phi(\mathbf{W})\epsilon}{\mathcal{O}^*_{\mathcal{M}}}\right)^2$, *then* $\mathbf{Y}$ *will be assigned to the correct class.*

Considering the more conservative bound that relies on $\mu(\mathbf{D})$ *and assuming that*

$$\|\Gamma\|^{\mathbf{S}}_{0,\infty} < k = \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D})}\left[1 - \left(\frac{2\phi(\mathbf{W})\epsilon}{\mathcal{O}^*_{\mathcal{M}}}\right)^2\right]\right),$$

then $\mathbf{Y}$ *will be assigned to the correct class.*

Proof Given that $f_u(\Gamma) = \mathbf{w}_u^T\Gamma + \omega_u > f_v(\Gamma) = \mathbf{w}_v^T\Gamma + \omega_v$ for all $v \neq u$, i.e., $\mathbf{X}$ belongs to class $y = u$, we shall prove that $f_u(\hat{\Gamma}) > f_v(\hat{\Gamma})$ for all $v \neq u$. Denoting $\Delta = \hat{\Gamma} - \Gamma$, we bound from below the difference $f_u(\hat{\Gamma}) - f_v(\hat{\Gamma})$ as follows:

$$\begin{aligned}
&\left[\mathbf{w}_u^T\hat{\Gamma} + \omega_u\right] - \left[\mathbf{w}_v^T\hat{\Gamma} + \omega_v\right] \\
&= \left[\mathbf{w}_u^T(\Gamma + \Delta) + \omega_u\right] - \left[\mathbf{w}_v^T(\Gamma + \Delta) + \omega_v\right] \\
&= \left[\mathbf{w}_u^T\Gamma + \omega_u\right] - \left[\mathbf{w}_v^T\Gamma + \omega_v\right] + \left(\mathbf{w}_u^T - \mathbf{w}_v^T\right)\Delta \\
&\geq f_u(\Gamma) - f_v(\Gamma) - \left|\left(\mathbf{w}_u^T - \mathbf{w}_v^T\right)\Delta\right| \\
&\geq f_u(\Gamma) - f_v(\Gamma) - \|\mathbf{w}_u^T - \mathbf{w}_v^T\|_2\|\Delta\|_2.
\end{aligned} \tag{6}$$

Similarly to the proof of Theorem 7, the first inequality holds since $a + b \geq a - |b|$ for $a = f_u(\Gamma) - f_v(\Gamma) > 0$, and the last inequality relies on the Cauchy-Schwarz formula. Relying on $\phi(\mathbf{W})$ that satisfies

$$\phi(\mathbf{W}) \geq \|\mathbf{w}_u - \mathbf{w}_v\|_2,$$

and plugging $\|\Delta\|_2^2 \leq \frac{4\epsilon^2}{1-\delta_{2k}}$ into Eq. (6) we get

$$\begin{aligned}
f_u(\hat{\Gamma}) - f_v(\hat{\Gamma}) &\geq f_u(\Gamma) - f_v(\Gamma) - \phi(\mathbf{W})\frac{2\epsilon}{\sqrt{1-\delta_{2k}}} \\
&\geq \mathcal{O}_{\mathcal{M}}(\mathbf{X}, y) - \phi(\mathbf{W})\frac{2\epsilon}{\sqrt{1-\delta_{2k}}} \\
&\geq \mathcal{O}^*_{\mathcal{M}} - \phi(\mathbf{W})\frac{2\epsilon}{\sqrt{1-\delta_{2k}}},
\end{aligned}$$

where the second to last inequality holds since $f_u(\Gamma) - f_v(\Gamma) \geq \mathcal{O}_{\mathcal{M}}(\mathbf{X}, y)$, and the last inequality follows the definition of $\mathcal{O}^*_{\mathcal{M}}$. As such, we shall seek for the following inequality to hold:

$$\begin{aligned}
&0 < \mathcal{O}^*_{\mathcal{M}} - \phi(\mathbf{W})\frac{2\epsilon}{\sqrt{1-\delta_{2k}}} \\
\rightarrow\ &\delta_{2k} < 1 - \left(\frac{2\phi(\mathbf{W})\epsilon}{\mathcal{O}^*_{\mathcal{M}}}\right)^2.
\end{aligned}$$

Similarly to the binary setting, one can readily write the above in terms of $\mu(\mathbf{D})$. □

Appendix C: Proof of Theorem 12: Stable Binary Classification of the L-THR

Theorem 10 (Stable binary classification of the L-THR) *Suppose we are given an ML-CSC signal* $\mathbf{X}$ *contaminated with perturbation* $\mathbf{E}$ *to create the signal* $\mathbf{Y} = \mathbf{X} + \mathbf{E}$, *such that* $\|\mathbf{E}\|^{\mathbf{P}}_{2,\infty} \leq \epsilon_0$. *Denote by* $|\Gamma_i^{min}|$ *and* $|\Gamma_i^{max}|$ *the lowest and highest entries in absolute value in the vector* Γ_i, *respectively. Suppose further that* $\mathcal{O}^*_{\mathcal{B}} > 0$ *and let* $\{\hat{\Gamma}_i\}_{i=1}^K$ *be the set of solutions obtained by running the layered soft thresholding algorithm with thresholds* $\{\beta_i\}_{i=1}^K$, *i.e.,* $\hat{\Gamma}_i = \S_{\beta_i}(\mathbf{D}_i^T\hat{\Gamma}_{i-1})$ *where* $\S_{\beta_i}$ *is the soft thresholding operator and* $\hat{\Gamma}_0 = \mathbf{Y}$. *Assuming that* $\forall\ 1 \leq i \leq K$

a. $\|\Gamma_i\|^{\mathbf{S}}_{0,\infty} < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D}_i)}\frac{|\Gamma_i^{min}|}{|\Gamma_i^{max}|}\right) - \frac{1}{\mu(\mathbf{D}_i)}\frac{\epsilon_{i-1}}{|\Gamma_i^{max}|}$;
b. *The threshold* β_i *is chosen according to*

$$|\Gamma_i^{min}| - C_i - \epsilon_{i-1} > \beta_i > K_i + \epsilon_{i-1},$$

where

$$\begin{aligned} C_i &= (\|\Gamma_i\|_{0,\infty}^{\mathbf{S}} - 1)\mu(\mathbf{D}_i)|\Gamma_i^{max}|, \\ K_i &= \|\Gamma_i\|_{0,\infty}^{\mathbf{S}}\mu(\mathbf{D}_i)|\Gamma_i^{max}|, \\ \epsilon_i &= \sqrt{\|\Gamma_i\|_{0,\infty}^{\mathbf{P}}}\Big(\epsilon_{i-1} + C_i + \beta_i\Big); \end{aligned}$$

and

c. $\mathcal{O}_{\mathcal{B}}^* > \|\mathbf{w}\|_2\sqrt{\|\Gamma_K\|_0}\Big(\epsilon_{K-1} + C_K + \beta_K\Big)$,

then $sign(f(\mathbf{Y})) = sign(f(\mathbf{X}))$.

Proof Following Theorem 10 in [22], if assumptions (a)–(c) above hold $\forall\ 1 \le i \le K$ then

1. The support of the solution $\hat{\Gamma}_i$ is equal to that of Γ_i; and
2. $\|\Gamma_i - \hat{\Gamma}_i\|_{2,\infty}^{\mathbf{P}} \le \epsilon_i$, where ϵ_i defined above.

In particular, the last layer satisfies

$$\|\Gamma_K - \hat{\Gamma}_K\|_\infty \le \epsilon_{K-1} + C_K + \beta_K. \tag{7}$$

Defining $\Delta = \hat{\Gamma}_K - \Gamma_K$, we get

$$\|\Delta\|_2 \le \|\Delta\|_\infty\sqrt{\|\Delta\|_0} = \|\Delta\|_\infty\sqrt{\|\Gamma_K\|_0},$$

where the last equality relies on the successful recovery of the support. Having the upper bound on $\|\Delta\|_2$, one can follow the transition from Eqs. (4) to (5) (see the proof of Theorem 7), leading to the following requirement for accurate classification:

$$\mathcal{O}_{\mathcal{B}}^* - \|\mathbf{w}\|_2\|\Delta\|_\infty\sqrt{\|\Gamma_K\|_0} > 0.$$

Plugging Eq. (7) to the above expression results in the additional condition that ties the propagated error throughout the layers to the output margin, given by

$$\mathcal{O}_{\mathcal{B}}^* > \|\mathbf{w}\|_2\sqrt{\|\Gamma_K\|_0}\Big(\epsilon_{K-1} + C_K + \beta_K\Big).$$

□

References

1. Aberdam, A., Sulam, J., Elad, M.: Multi-layer sparse coding: the holistic way. SIAM J. Math. Data Sci. **1**(1), 46–77 (2019)
2. Bibi, A., Ghanem, B., Koltun, V., Ranftl, R: Deep layers as stochastic solvers. In: International Conference on Learning Representations (2019)
3. Bishop, C.: Neural Networks for Pattern Recognition. Oxford University Press, Oxford (1995)
4. Bredensteiner, E.J., Bennett, K.P.: Multicategory classification by support vector machines. In: Computational Optimization, pp. 53–79. Springer, Berlin (1999)
5. Candes, E.J.: The restricted isometry property and its implications for compressed sensing. C.R. Math. **346**(9–10), 589–592 (2008)
6. Elad, M.: Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing, 1st edn. Springer, Berlin (2010)
7. Fawzi, A., Fawzi, H., Fawzi, O.: Adversarial vulnerability for any classifier. arXiv preprint arXiv:1802.08686 (2018)
8. Fawzi, A., Fawzi, O., Frossard, P.: Analysis of classifiers' robustness to adversarial perturbations. Mach. Learn. **107**(3), 481–508 (2018)
9. Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y.: Deep Learning, vol. 1. MIT Press, Cambridge (2016)
10. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. ICLR (2015)
11. Gregor, K., LeCun, Y.: Learning fast approximations of sparse coding. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10), pp. 399–406 (2010)
12. Krizhevsky, A., Nair, V., Hinton, G.: The CIFAR-10 dataset. online: http://www.cs.toronto.edu/kriz/cifar.html (2014)
13. Kurakin. A., Goodfellow, I.,, Bengio, S.: Adversarial examples in the physical world. arXiv preprint arXiv:1607.02533 (2016)
14. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
15. LeCun, Y., Cortes, C., Burges, C.J.: MNIST handwritten digit database. AT&T Labs [Online]. Available: http://yann.lecun.com/exdb/mnist, 2 (2010)
16. Liao, F., Liang, M., Dong, Y., Pang, T., Zhu, J., Hu, X.: Defense against adversarial attacks using high-level representation guided denoiser. In: IEEE-CVPR (2018)
17. Liu, Y., Chen, X., Liu, C., Song, D.: Delving into transferable adversarial examples and black-box attacks. In: ICLR (2017)
18. Mahdizadehaghdam, S., Panahi, A., Krim, H., Dai, L.: Deep dictionary learning: a parametric network approach. arXiv preprint arXiv:1803.04022 (2018)
19. Mairal, J., Bach, F., Ponce, J.: Sparse modeling for image and vision processing. arXiv preprint arXiv:1411.3230 (2014)
20. Moustapha, C., Piotr, B., Edouard, G., Yann, D., Nicolas, U.: Parseval networks: improving robustness to adversarial examples. In: ICML (2017)
21. Papyan, V., Romano, Y., Elad, M.: Convolutional neural networks analyzed via convolutional sparse coding. J. Mach. Learn. Res. **18**(83), 1–52 (2017)
22. Papyan, V., Sulam, J., Elad, M.: Working locally thinking globally: theoretical guarantees for convolutional sparse coding. IEEE Trans. Signal Process. **65**(21), 5687–5701 (2017)
23. Sokolić, J., Giryes, R., Sapiro, G., Rodrigues, M.R.D.: Robust large margin deep neural networks. IEEE Trans. Signal Process. **65**(16), 4265–4280 (2016)
24. Sulam, J., Aberdam, A., Beck, A., Elad, M.: On multi-layer basis pursuit, efficient algorithms and convolutional neural networks. IEEE Trans. Pattern Anal. Mach. Intell. (2019)
25. Sulam, J., Papyan, V., Romano, Y., Elad, M.: Multilayer convolutional sparse modeling: pursuit and dictionary learning. IEEE Trans. Signal Process. **66**(15), 4090–4104 (2018)
26. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Dumitru, E., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. In: ICLR (2014)
27. Zeiler, M.D.., Krishnan, D., Taylor, G.W., Fergus, R.: Deconvolutional networks. In: IEEE-CVPR (2010)

Yaniv Romano received the B.Sc., M.Sc. and Ph.D. degrees from the Department of Electrical Engineering, Technion—Israel Institute of Technology, Haifa, Israel, in 2012, 2015, and 2017, respectively. He is currently a Postdoctoral Fellow with the Department of Statistics, Stanford University, Stanford, CA, USA. His research interests include machine learning, selective inference, uncertainty estimation, signal and image modeling, inverse problems, and sparse and redundant representations. He is a recipient of the 2015 Zeff fellowship, the 2017 Andrew and Erna Finci Viterbi fellowship, the 2017 Irwin and Joan Jacobs fellowship, the 2018–2020 Zuckerman postdoctoral scholarship, the 2018–2020 ISEF fellowship for postdoctoral studies, and the 2018–2020 Viterbi fellowship for nurturing future faculty members, Technion.

Aviad Aberdam received the B.Sc. degree from the Department of Electrical Engineering, Technion, Israel (2017). He is currently pursuing his Ph.D. at the EE department of the Technion, supervised by Michael Elad. He is the recipient of the 2020–2022 Azrieli fellowship, 2019 Fine fellowship and 2017–2018 Meyer fellowship. His research interests are machine learning, optimization and signal and image processing, in particular inverse problems and sparse representations.

Jeremias Sulam received his Bioengineering degree from the Universidad Nacional de Entre Rios, Argentina (2013), and his Ph.D. (2018) from the Computer Science Department of the Technion—Israel Institute of Technology. Since 2018, he is an Assistant Professor at the Biomedical Engineering Department at Johns Hopkins University, and affiliated with the Center for Imaging Science (CIS) and the Mathematical Institute for Data Science (MINDS). He is the recipient of the Best Graduates award of the Argentinean National Academy of Engineering. His research interests include signal and image processing, sparse representation modeling, inverse problems and machine learning, and their application to biomedical problems.

Michael Elad received the B.Sc., M.Sc. and D.Sc. degrees from the Department of Electrical engineering, Technion, Israel, in 1986, 1988, and 1997, respectively. Since 2003, he has been a faculty member in the Computer Science Department, Technion, and since 2010 he holds a Full-Professorship Position. He works in the field of signal and image processing, specializing in inverse problems and sparse representations. He received numerous teaching awards, the 2008 and 2015 Henri Taub Prizes for Academic Excellence, and the 2010 Hershel-Rich prize for innovation. Michael is an IEEE Fellow (2012) and a SIAM Fellow (2018). He is serving as the Editor-in-Chief for SIAM Journal on Imaging Sciences since January 2016.