NONLINEAR BLIND COMPRESSED SENSING UNDER SIGNAL-DEPENDENT NOISE

Rudrajit Das[1] *and Ajit Rajwade*[2]

[1]Dept. of Electrical Engineering, [2]Dept. of Computer Science & Engineering; IIT Bombay

ABSTRACT

In this paper, we consider the problem of nonlinear blind compressed sensing, i.e. jointly estimating the sparse codes and sparsity-promoting basis, under signal-dependent noise. We focus our efforts on the Poisson noise model, though other signal-dependent noise models can be considered. By employing a well-known variance stabilizing transform such as the Anscombe transform, we formulate our task as a non-linear least squares problem with the ℓ_1 penalty imposed for promoting sparsity. We solve this objective function under non-negativity constraints imposed on both the sparse codes and the basis. To this end, we propose a multiplicative update rule, similar to that used in non-negative matrix factorization (NMF), for our alternating minimization algorithm. To the best of our knowledge, this is the first attempt at a formulation for *nonlinear* blind compressed sensing, with and without the Poisson noise model. Further, we also provide some theoretical bounds on the performance of our algorithm.

Index Terms— Blind Compressed Sensing, Anscombe Transform, Multiplicative Update, Performance Bounds.

I. INTRODUCTION

The field of compressed sensing famously presents an excellent confluence between theory and practice. There exist well-known theoretical performance bounds for the estimation of a signal $\boldsymbol{x} \in \mathbb{R}^n$ from its $m \ll n$ noisy compressive measurements of the form $\boldsymbol{y} = \boldsymbol{\Phi x} + \boldsymbol{\eta}$, where $\boldsymbol{y} \in \mathbb{R}^m$ is a noisy vector of measurements, $\boldsymbol{\eta}$ is the noise vector, and $\boldsymbol{\Phi} \in \mathbb{R}^{m \times n}$ is the sensing matrix [1]. These bounds hold under two sufficient conditions: the first pertaining to the sparsity/compressibility of $\boldsymbol{x}$, and the second pertaining to characteristics of $\boldsymbol{\Phi}$ such as the restricted isometry property (RIP). The latter property prohibits sparse signals from lying in the null-space of $\boldsymbol{\Phi}$. The aforementioned bounds also extend to the case where $\boldsymbol{x}$ is not itself sparse/compressible, but instead has a sparse/compressible representation in a (typically but not necessarily orthonormal) basis (also called 'dictionary') $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ [2], [3]. That is $\boldsymbol{x} = \boldsymbol{A\theta}$ where $\boldsymbol{\theta} \in \mathbb{R}^n$ is a sparse/compressible vector. In compressive imaging, typical models for $\boldsymbol{A}$ include the discrete cosine transform (DCT) or various wavelet transforms, since natural images/image patches are known to be compressible in these bases [3]. However the dictionary $\boldsymbol{A}$ can also be learned offline in a manner tuned to a particular class of images [4]. Such a process, however, is necessarily restricted to specific image classes, and also requires the availability of adequate training data. To overcome these limitations, there is interest in inferring $\boldsymbol{A}$ *directly* from the compressive measurements of a group of signals $\{\boldsymbol{x_i}\}_{i=1}^T$ along with their respective sparse codes $\{\boldsymbol{\theta_i}\}_{i=1}^T$. Such a task is termed *blind compressive sensing* (BCS) and has been successfully applied to synthetic [5] and real compressive data [6]. Despite this experimental success, most attempts at a theoretical development for BCS have had limitations. For example, the work in [7] makes very specific/restrictive structural requirements on the dictionary $\boldsymbol{A}$. The very recent work in [8] provides more general theoretical treatment, but the noise model is not fully analyzed.

There are situations, however, when the linear compressive acquisition model $\boldsymbol{y} = \boldsymbol{\Phi x} + \boldsymbol{\eta}$ is not valid. For example, this happens in tomographic acquisition via non-linearity induced by Beer's law [9]. In another scenario, the measurements may themselves be linear, but the noise may be heteroscedastic or signal-dependent. In such a case, a (typically) non-linear transformation f can be applied to the measurement $\boldsymbol{y}$ so that the noise in $f(\boldsymbol{y})$ is homoscedastic and signal-independent. Examples of such transformations are [10] (Sec. 14.6 and 14.7): (1) Square-root, i.e. $f(\boldsymbol{y}) \triangleq \sqrt{\boldsymbol{y}}$ if the noise variance v_n is directly proportional to the mean μ_n (or the underlying signal). The Poisson distribution, ubiquitous in optical and XRay imaging systems, is a special case where $v_n = \mu_n$. (2) Logarithmic, i.e. $f(\boldsymbol{y}) \triangleq \log \boldsymbol{y}$ in case of multiplicative noise or if $\sqrt{v_n} \propto \mu_n$. (3) Reciprocal, i.e. $f(\boldsymbol{y}) \triangleq \boldsymbol{1}./\boldsymbol{y}$ if $\sqrt{v_n} \propto \mu_n^2$. In this paper, we consider the case of non-linear blind compressed sensing in conjunction with the *square-root transformations* on the noisy data, and present rigorous theoretical derivations as well as experimental results. *To the best of our knowledge, ours is the first piece of work to provide algorithms as well as theoretical development for the case of nonlinear BCS of any type.*

[2]AR thanks IITB seed-grant 14IRCCSG012 and NVIDIA Corporation for a GPU donation that was useful for this work.

II. THEORY

Here, we first define the computational problem. Consider T different n-dimensional non-negative signals $\{\boldsymbol{x_i}\}_{i=1}^{T}$ which are all s-sparse in some k-column dictionary such that $s \leq k < n$, i.e. $\boldsymbol{x_i} = \boldsymbol{A\theta_i}$, where $\boldsymbol{A}$ is an $n \times k$ non-negative dictionary, and where for all i from 1 to T, $\boldsymbol{\theta_i}$ is the $k \times 1$ s-sparse vector of coefficients for $\boldsymbol{x_i}$. We consider compressive measurements of the form $\boldsymbol{y_i} \sim \text{Poiss}(\boldsymbol{\Phi_i A \theta_i})$ where $\boldsymbol{y_i}$ is m-dimensional ($m < n$), $\boldsymbol{\Phi_i}$ is a $m \times n$ non-negative sensing matrix (different sensing matrices for all i). Also $\boldsymbol{u} \sim \text{Poiss}(\boldsymbol{v})$ denotes a vector of independent Poisson random variables such that $\forall i, u_i \sim \text{Poiss}(v_i)$. The problem is to estimate $\boldsymbol{A}$ and $\boldsymbol{\Theta} \triangleq \{\boldsymbol{\theta_i}\}_{i=1}^{T}$, given $\{\boldsymbol{y_i}\}_{i=1}^{T}$. Throughout this paper, $\|\boldsymbol{r}\|$ and $\|\boldsymbol{r}\|_1$ refer to the ℓ_2 and ℓ_1 norms of the vector $\boldsymbol{r}$ respectively.

II-A. Objective Function and Algorithm

According to the Anscombe transform, if $u \sim \text{Poiss}(v)$, then $\left(\sqrt{u+c} - \sqrt{v+c}\right)$ with $c \triangleq 3/8$, is approximately $\mathcal{N}(0, 0.25)$ [11]. We use this to convert our problem into a non-linear least squares problem. Imposing the ℓ_1 sparsity prior on $\boldsymbol{\theta_i}$, our objective function (negative log of a quasi-likelihood function) is expressed as:

$$J(\boldsymbol{A}, \boldsymbol{\Theta}) = \sum_{i=1}^{T} \left\{ \left\| \sqrt{\boldsymbol{y_i} + c\boldsymbol{1_m}} - \sqrt{\boldsymbol{\Phi_i A \theta_i} + c\boldsymbol{1_m}} \right\|^2 + \lambda \|\boldsymbol{\theta_i}\|_1 \right\}$$

$$\text{such that } \boldsymbol{A} \succeq \boldsymbol{0_{n \times k}}, \text{ and } \forall i, \|\boldsymbol{A\theta_i}\|_1 \leq I, \boldsymbol{\theta_i} \succeq \boldsymbol{0_k}, \quad (1)$$

where (i) $\mathbf{1}_m$ denotes an $m \times 1$ vector of all ones and $\mathbf{0}_k$ denotes a $k \times 1$ vector of all zeros, (ii) the square root operation is applied element-wise and $\boldsymbol{a} \succeq \mathbf{0}$ means that every element of vector $\boldsymbol{a}$ is non-negative, (iii) λ is a regularization parameter, and (iv) I is an a priori known upper bound on the sum-total value (hereafter termed the 'intensity') of each $\boldsymbol{x_i}$. We advise choosing λ by trying with several discrete values and picking one which satisfies condition $\mathcal{C}$ defined in (8) of Section II-B, if such a value exists. With this in mind, we propose a modified constrained objective function given below in (2). We use measurements of the first $T^{'}$ signals as a 'training set' $\mathcal{T}$ for inferring the dictionary, and the measurements of the remaining signals as a 'validation set' $\mathcal{V}$. That is, $\mathcal{T} \triangleq \{\boldsymbol{y_i}\}_{i=1}^{T'}$ form and $\mathcal{V} \triangleq \{\boldsymbol{y_i}\}_{i=T'+1}^{T}$. This splitting is done for the sake of exploiting certain statistical independence properties for precise characterization of the noise residual in our theoretical performance bounds (refer to proof of Lemma 1 in supplementary material [12]), and to prevent overfitting. For the same reason, we also split up $\boldsymbol{y_i}$ and $\boldsymbol{\Phi_i}$ as follows:

$$\boldsymbol{y_i} = \begin{bmatrix} \boldsymbol{y_i^{(1)}} \\ \boldsymbol{y_i^{(2)}} \end{bmatrix}, \boldsymbol{\Phi_i} = \begin{bmatrix} \boldsymbol{\Phi_i^{(1)}} \\ \boldsymbol{\Phi_i^{(2)}} \end{bmatrix},$$

where $\boldsymbol{y_i^{(1)}}$ and $\boldsymbol{y_i^{(2)}}$ are $m_1 \times 1$ and $m_2 \times 1$ vectors respectively, $\boldsymbol{\Phi_i^{(1)}}$ and $\boldsymbol{\Phi_i^{(2)}}$ are $m_1 \times n$ and $m_2 \times n$ matrices respectively and $m_1 + m_2 = m$.

Define $\boldsymbol{\Theta_{T'}} \triangleq \{\boldsymbol{\theta_i}\}_{i=1}^{T'}$. Then the objective function over the training set, $J(\boldsymbol{A}, \boldsymbol{\Theta_{T'}})$ abbreviated by $J_{\mathcal{T}}$, using the first m_1 measurements becomes:

$$J_{\mathcal{T}} = \sum_{i=1}^{T^{'}} \left\{ \left\| \sqrt{\boldsymbol{y_i^{(1)}} + c\boldsymbol{1_{m_1}}} - \sqrt{\boldsymbol{\Phi_i^{(1)} A \theta_i} + c\boldsymbol{1_{m_1}}} \right\|^2 + \lambda \|\boldsymbol{\theta_i}\|_1 \right\} \quad (2)$$

$$\text{such that } \boldsymbol{A} \succeq \boldsymbol{0_{n \times k}}, \text{ and } \forall i, \boldsymbol{\theta_i} \succeq \boldsymbol{0_k}.$$

The optimization for $J_{\mathcal{T}}$ proceeds only up to the point, when a statistically driven termination condition $\mathcal{C}$ on the validation set, defined in (8), is satisfied. Note that $\boldsymbol{A}$ is estimated by minimizing $J_{\mathcal{T}}$ (i.e. only using the training set), however we need to estimate the sparse codes for the training set ($\mathcal{T}$) as well as the validation set ($\mathcal{V}$). The latter is required for verifying condition $\mathcal{C}$. Using the estimated value of $\boldsymbol{A}$ obtained from (2), we estimate the sparse code $\boldsymbol{\theta_i}$ for the i^{th} ($T^{'} < i \leq T$) signal in $\mathcal{V}$ by minimizing:

$$J_{\mathcal{V}}^{(i)} = \left\| \sqrt{\boldsymbol{y_i^{(1)}} + c\boldsymbol{1_{m_1}}} - \sqrt{\boldsymbol{\Phi_i^{(1)} A \theta_i} + c\boldsymbol{1_{m_1}}} \right\|^2 + \lambda \|\boldsymbol{\theta_i}\|_1 \quad (3)$$

$$\text{such that } \boldsymbol{\theta_i} \succeq \boldsymbol{0_k}.$$

We note the following regarding the optimization of $J_{\mathcal{T}}$ in (2): (i) It is biconvex, i.e. it is convex in $\boldsymbol{\Theta_{T'}}$ if $\boldsymbol{A}$ is fixed and vice-versa. The constraints are also all convex. (ii) We can optimize this objective function using an alternating projected gradient descent algorithm with an adaptive step size. The convergence of such a procedure is guaranteed [13]. (iii) Note that the constraint $\|\boldsymbol{A\theta_i}\|_1 \leq I$ present in (1) has been omitted in (2) and (3) since it is only required for obtaining theoretical performance bounds and is not necessary in practical simulations.

Instead of the adaptive step-size, we also derived multiplicative update rules for $\boldsymbol{\Theta}$ as well as $\boldsymbol{A}$. Details of these can be found in [12]. We mention the update rules here briefly. Before that, let the estimated values of $\boldsymbol{A}$ and $\boldsymbol{\Theta}$ at the end of the t^{th} iteration of gradient descent be $\boldsymbol{A^{(t)}}$ and $\boldsymbol{\Theta^{(t)}} \triangleq \{\boldsymbol{\theta_i^{(t)}}\}_{i=1}^{T}$. Then, under the multiplicative update rules, we have:

$$\boldsymbol{\theta_i^{(t+1)}} = (1 - \beta_i^{(t)})\boldsymbol{\theta_i^{(t)}} + \beta_i^{(t)}\boldsymbol{\theta_i}^{(\text{new})} \text{ where} \quad (4)$$

$$0 < \beta_i^{(t)} \leq 1, \boldsymbol{\theta_i}^{(\text{new})} = \max\left(\mathbf{0}, \left[\frac{\boldsymbol{\theta_i^{(t)}}}{(\boldsymbol{\Phi_i^{(1)} A^{(t)}})^T \boldsymbol{1_{m_1}}} \odot \left\{ (\boldsymbol{\Phi_i^{(1)} A^{(t)}})^T \left(\frac{\sqrt{\boldsymbol{y_i^{(1)}} + c\boldsymbol{1_{m_1}}}}{\sqrt{\boldsymbol{\Phi_i^{(1)} A^{(t)} \theta_i^{(t)}} + c\boldsymbol{1_{m_1}}}} \right) - \lambda \mathbb{1}(\boldsymbol{\theta_i^{(t)}}) \right\} \right]\right), \quad (5)$$

where $0 < \beta_i^{(t)} \leq 1$ and $\mathbb{1}(z) = 1$ if $z > 0$ else $\mathbb{1}(z) = 0$. Also, we have

$$\boldsymbol{A^{(t+1)}} = (1 - \beta^{(t)})\boldsymbol{A^{(t)}} + \beta^{(t)}\boldsymbol{A}^{(\text{new})} \text{ where} \quad (6)$$

$$0 < \beta^{(t)} \leq 1, \boldsymbol{A}^{(\text{new})} = \left[\frac{\boldsymbol{A^{(t)}}}{\sum_{i=1}^{T'} (\boldsymbol{\Phi_i^{(1)}})^T \boldsymbol{1_{m_1}} (\boldsymbol{\theta_i^{(t+1)}})^T} \odot \right.$$

$$\sum_{i=1}^{T'}(\boldsymbol{\Phi}_{\boldsymbol{i}}^{\mathbf{(1)}})^T\left(\frac{\sqrt{\boldsymbol{y}_{\boldsymbol{i}}^{\mathbf{(1)}}+c\mathbf{1}_{\boldsymbol{m_1}}}}{\sqrt{\boldsymbol{\Phi}_{\boldsymbol{i}}^{\mathbf{(1)}}\boldsymbol{A}^{(t)}\boldsymbol{\theta}_{\boldsymbol{i}}^{\boldsymbol{(t+1)}}+c\mathbf{1}_{\boldsymbol{m_1}}}}\right)(\boldsymbol{\theta}_{\boldsymbol{i}}^{\boldsymbol{(t+1)}})^T\Bigg]. \tag{7}$$

In (5) and (7), the $\odot$ operator denotes the element-wise product of two vectors or matrices. The division operations are also performed element-wise. Empirically, we saw that in almost all cases, $\beta_i = 1\ \forall i$ and $\beta = 1$ does indeed reduce the objective function value. However, theoretically, we were not able to establish this (refer to [12]).

II-B. Termination condition for the Algorithm

Here, we provide the following condition for termination:

$$\text{Condition } \mathcal{C} : m_2/4 \leq \widehat{E_v} \leq m_2(1+\varepsilon)/4, \tag{8}$$

$$\widehat{E_v} \triangleq \sum_{i=T'+1}^{T} \frac{\left\|\sqrt{\boldsymbol{y}_{\boldsymbol{i}}^{\mathbf{(2)}}+c\mathbf{1}_{\boldsymbol{m_2}}}-\sqrt{\boldsymbol{\Phi}_{\boldsymbol{i}}^{\mathbf{(2)}}\boldsymbol{A}^{(t)}\boldsymbol{\theta_i}^{(t)}+c\mathbf{1}_{\boldsymbol{m_2}}}\right\|^2}{(T-T')}.$$

The condition in (8) is chosen in order to prevent overfitting. The idea behind (8) is that $\widehat{E_v}$ is the sum of the squares of $m_2(T-T^{'})$ approximately zero-mean Gaussian random variables with variance $1/4$, divided by $(T-T^{'})$. Given enough data in $\mathcal{V}$, i.e. $(T-T^{'})$ is large enough, we expect $\widehat{E_v} \approx 0.25(m_2(T-T^{'}))/(T-T^{'}) = m_2/4$.

We now present a statistical criterion for choice of ε. The criterion is such that $\mathbb{P}\left(\widehat{E_v} > m_2(1+\varepsilon)/4\right) < p$ where $p = \exp\bigl(-\zeta^2(T-T')\bigr)$ where ζ is a non-zero constant of our choice. Observe that p reduces exponentially as the number of validation examples increases. In this case, using concentration inequalities for the chi-squared random variable $\widehat{E_v}$ [14], we get (detailed proof included in [12]):

$$\varepsilon = 2\zeta/\sqrt{m_2}. \tag{9}$$

However, it might not always be possible to converge to a solution which satisfies (8). In that case, we stop when:

$$\widehat{E_t} \leq m_1/4 \text{ where} \tag{10}$$

$$\widehat{E_t} \triangleq \sum_{i=1}^{T'} \frac{\left\|\sqrt{\boldsymbol{y}_{\boldsymbol{i}}^{\mathbf{(1)}}+\boldsymbol{c}\mathbf{1}_{\boldsymbol{m_1}}}-\sqrt{\boldsymbol{\Phi}_{\boldsymbol{i}}^{\mathbf{(1)}}\boldsymbol{A}^{(t)}\boldsymbol{\theta_i}^{(t)}+c\mathbf{1}_{\boldsymbol{m_1}}}\right\|^2}{T'}.$$

It is clear that $\widehat{E_t}$ above is the expected value of the squared loss part of the objective function over the training set (and the first m_1 measurements).

The complete procedure is summarized in Algorithm 1.

II-C. Performance Bounds for NLBCS

We firstly present our main theorem on NLBCS:

Theorem 1. *Consider that the entries of all the sensing matrices $\{\boldsymbol{\Phi_i}\}_{i=1}^{T}$ are independently drawn from a $\{0,1\}$ Bernoulli distribution with probability of drawing 1 (and 0) being 0.5, denoted by Bernoulli(0.5). Let $\boldsymbol{A_e}$ and $\{(\boldsymbol{\theta_i})_{\boldsymbol{e}}\}_{i=T'+1}^{T}$ be the estimates of $\boldsymbol{A}$ and $\{\boldsymbol{\theta_i}\}_{i=T'+1}^{T}$ respectively, upon running NLBCS. Suppose that the condition $\mathcal{C}$ in (8) holds when NLBCS terminates. Also suppose that $\mathcal{V}$ is large enough so that $\widehat{E_v} \approx \mathbb{E}_{val}\left[\left\|\sqrt{\boldsymbol{y}^{\mathbf{(2)}}+c\mathbf{1}_{\boldsymbol{m_2}}}-\sqrt{\boldsymbol{\Phi}^{\mathbf{(2)}}\boldsymbol{A_e}\boldsymbol{\theta_e}+c\mathbf{1}_{\boldsymbol{m_2}}}\right\|^2\right]$ where the symbol E_{val} indicates that the expectation is over signals in $\mathcal{V}$ and over all possible Bernoulli(0.5) sensing matrices. Then with probability $p(\delta, T-T^{'})$, we have the following bound:*

$$\mathbb{E}_{val}\left[\|\boldsymbol{x}-\boldsymbol{x_e}\|^2\right] \leq \mathcal{O}(I\varepsilon/(1-\delta)),$$

where $\boldsymbol{x} = \boldsymbol{A\theta}$ and $\boldsymbol{x_e} = \boldsymbol{A_e\theta_e}$ are the actual and estimated (using the NLBCS Algorithm) values of the signal, I is the upper bound on the ℓ_1 norm of each signal as defined previously, δ is a parameter in $(0,1)$ and $p(\delta, T-T^{'}) \geq 1-\mathcal{O}\Bigl(\exp\Bigl(-\widetilde{c}\delta^2 m_2\sqrt{T-T'}\Bigr)\Bigr)$ with $\widetilde{c} > 0$ for large $(T-T^{'})$ (i.e. the validation set is large enough).

Algorithm 1 Nonlinear Blind Compressed Sensing (**NLBCS**) Algorithm

```
Initialize iteration counter $t = 0$, $\boldsymbol{A}^{\mathbf{(0)}}$ and $\boldsymbol{\theta}_{\boldsymbol{i}}^{\mathbf{(0)}}$'s with
  random non-negative entries. Fix $\varepsilon$ as per (9).
while (8) and (10) is false do
    for $1 \leq i \leq T$ do
        Update $\boldsymbol{\theta}_{\boldsymbol{i}}^{\boldsymbol{(t+1)}}$ as per (4) with a suitable $\beta_i^{(t)}$.
    end for
    Update $\boldsymbol{A}^{\boldsymbol{(t+1)}}$ as per (6) with a suitable $\beta^{(t)}$.
    $t \leftarrow (t+1)$
end while
```

We present the detailed proof of **Theorem 1** and the precise quantification of $p(\delta, T-T^{'})$ in [12]. In order to provide a proof sketch here, we mention the following three key lemmas (which have also been proved in [12]) used in the proof.

Lemma 1. *Under the settings of **Theorem 1**, we have:*

$$\mathbb{E}_{val}\left[\left\|\sqrt{\boldsymbol{\Phi}^{\mathbf{(2)}}\boldsymbol{x}+c\mathbf{1}_{\boldsymbol{m_2}}}-\sqrt{\boldsymbol{\Phi}^{\mathbf{(2)}}\boldsymbol{x_e}+c\mathbf{1}_{\boldsymbol{m_2}}}\right\|^2\right] \leq \frac{m_2\varepsilon}{4}.$$

Lemma 2. *Given that **Lemma 1** holds, we have:*

$$\mathbb{E}_{val}\left[\|\boldsymbol{\Phi}^{\mathbf{(2)}}(\boldsymbol{x}-\boldsymbol{x_e})\|^2\right] \leq \mathcal{O}(Im_2\varepsilon).$$

Lemma 3. *(**Abridged version**) Let $\delta \in (0,1)$. Then with probability $p(\delta, T-T^{'}) \geq 1-\mathcal{O}\Bigl(\exp\Bigl(-\widetilde{c}\delta^2 m_2\sqrt{T-T'}\Bigr)\Bigr)$ $(\widetilde{c} > 0)$ for large $(T-T^{'})$, the following bound holds:*

$$\mathbb{E}_{val}\left[\|\boldsymbol{x}-\boldsymbol{x_e}\|^2\right] \leq \mathbb{E}_{val}\left[\|\boldsymbol{\Phi}^{\mathbf{(2)}}(\boldsymbol{x}-\boldsymbol{x_e})\|^2\right]/(m_2(1-\delta)).$$

[12] contains the full version of **Lemma 3** where $p(\delta, T-T^{'})$ is properly quantified. Combining **Lemma 1**, **Lemma 2** and **Lemma 3**, we can prove **Theorem 1**. We also have the following corollary of **Theorem 1** (proof in [12]):

Corollary 1. *Suppose on running the NLBCS Algorithm under the same settings as described in **Theorem 1**, (8) holds with ε given by (9), then the following bound holds with probability $p(\delta, T-T^{'})$:*

$$\mathbb{E}_{val}\left[\frac{\|\boldsymbol{x}-\boldsymbol{x_e}\|}{\|\boldsymbol{x}\|}\right] \leq \mathcal{O}\left(\sqrt{\frac{\zeta n}{I\sqrt{m_2}(1-\delta)}}\right). \qquad (11)$$

Remarks about the bounds: Observe that the bound improves as m_2 (dependent on the number of measurements) and I increase, which makes intuitive sense. Also the probability $p(\delta, T-T')$ increases in $T-T'$ (see [12]).

Unfortunately, we are unable to provide any bounds on the performance of our algorithm should it terminate when (10) holds. This is because of the non-zero correlation term between the noise and the difference $\sqrt{\boldsymbol{\Phi^{(2)}x}+c\boldsymbol{1_{m_2}}} - \sqrt{\boldsymbol{\Phi^{(2)}x_e}+c\boldsymbol{1_{m_2}}}$ when considering the training set. Please refer to the detailed proof of **Theorem 1** for more details.

III. EXPERIMENTS

In this section, we describe proof-of-concept experiments - two with different synthetic data, and one with patches from an actual image. In practice, we should try with several values of λ and choose the one which results in the best performance. To evaluate the results, we used the $RRMSE$ metric defined as $\mathbb{E}_{val}\left[\|\boldsymbol{x}-\boldsymbol{x_e}\|/\|\boldsymbol{x}\|\right]$ (same as in **Corollary 1**), where $\boldsymbol{x_e}$ is an estimate for the signal $\boldsymbol{x}$. All experiments were performed using $\boldsymbol{\Phi^{(i)}} \sim$ Bernoulli(0.5).

NLBCS on Synthetic Data: Firstly, we considered $n = 80, k = 20, s = 8, T = 4000, T^{'} = 3000$. We set $\boldsymbol{A}$ to contain the element-wise absolute values of the first k columns of the DCT matrix. We drew the non-zero elements of $\boldsymbol{\theta_i}$ (for all i) from an exponential distribution with mean $\alpha (> 0)$ in order to control I. We tried with $m = \{10, 20, 30, 40, 50, 60\}$ and $m_1 = m - 5$ ($m_2 = 5$). We used $\lambda = 1.5m_1$. Table I shows the obtained $RRMSE$ values for $I = \{24, 244\}$.

RRMSE-	m=10	m=20	m=30	m=40	m=50	m=60
$I = 24$	0.2251	0.2218	0.2185	0.2180	0.1954	0.1918
$I = 244$	0.2132	0.2079	0.1959	0.1922	0.1841	0.1828

Table I. $RRMSE$ values for first synthetic dataset.

In the second case, we took $n = 150, k = 25, s = 15, T = 8000, T^{'} = 6000$. We set $\boldsymbol{A}$ to contain the element-wise absolute values of an $n \times k$ random matrix whose entries are drawn from $\mathcal{N}(0, 1)$. We drew the non-zero elements of $\boldsymbol{\theta_i}$ from Unif$[0, \alpha]$ ($\alpha > 0$) to control I. We tried with $m = \{20, 40, 60, 80, 100\}$. For $m = 20, 40$ and 60, we chose $m_1 = m-5$ and $m_2 = 5$, whereas for $m = 80$ and 100, we chose $m_1 = m-10$ and $m_2 = 10$. We set $\lambda = 0.6m_1$. Table II shows the obtained $RRMSE$ values for $I = \{2, 58\}$.

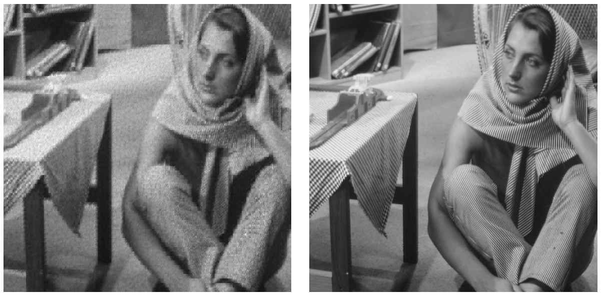
RRMSE-	m=20	m=40	m=60	m=80	m=100
$I = 2$	0.2263	0.2040	0.1896	0.1864	0.1796
$I = 58$	0.1835	0.1792	0.1748	0.1711	0.1673

Table II. $RRMSE$ values for second synthetic dataset.

NLBCS on Image Patches: We performed experiments on the famous Barbara image of size 512×512, divided into $64 \times 64 = 4096 = T$ non-overlapping (to maintain independence of the signals) patches of size 8×8 each, followed by reshaping of each patch to form a $64 (= n) \times 1$ vector. We estimated a dictionary consisting of $k = 16$ columns using $T^{'} = 3600$ signals in our training set and the rest in our validation set. We worked with $m = \{16, 24, 32, 40\}$. For $m = 16$ and 24, we chose $m_1 = m - 4$ and $m_2 = 4$, whereas for $m = 32$ and 40, we chose $m_1 = m - 5$ and $m_2 = 5$. We set the value of $\lambda = 0.375m_1$ in all cases. Table III shows the $RRMSE$ values and Fig. 1 shows the reconstructed image with $m = 40$ alongside the original one.

	m=16	m=24	m=32	m=40
RRMSE-	0.1824	0.1821	0.1751	0.1291

Table III. $RRMSE$ values for the Barbara experiment.



(a) Reconstructed image

(b) Original image

Fig. 1. Barbara experiment with $m = 40$.

It can be observed that in all our experiments, the obtained $RRMSE$ decreases as m and I increase. [12] contains additional experiments to show that learning a data-dependent dictionary using our algorithm is better than just using an off the shelf dictionary (and optimizing only over the sparse codes).

IV. CONCLUSION

In this paper, we have presented a *nonlinear* blind compressed sensing algorithm for a realistic noise model, together with theoretical performance bounds. To the best of our knowledge, this is the first algorithmic or theoretical attempt at this problem. Future work will involve deriving convergence rates for the algorithm, and extending the theory to other non-linear models apart from square-root.

V. REFERENCES

[1] E. Candes, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus de Mathematique*, 2008.

[2] E. Candes, Y. Eldar, D. Needell, and P. Randall, "Compressed sensing with coherent and redundant dictionaries," *Applied and Computational Harmonic Analysis*, vol. 31, no. 1, pp. 59 – 73, 2011.

[3] J. Romberg, "Imaging via compressive sampling," *IEEE Sig. Proc. Mag.*, 2008.

[4] M. Aharon, M. Elad, and A. Bruckstein, "KSVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Sig. Proc.*, 2006.

[5] F. Anaraki and S. Hughes, "Compressive K-SVD," in *ICASSP*, 2013.

[6] A. Rajwade, D. Kittle, T.-H. Tsai, D. Brady, and L. Carin, "Coded hyperspectral imaging and blind compressive sensing," *SIAM J. Imaging Sciences*, 2013.

[7] S. Gleichman and Y. C. Eldar, "Blind compressed sensing," *IEEE Trans. on Information Theory*, 2011.

[8] M. Aghagolzadeh and H. Radha, "Joint estimation of dictionary and image from compressive samples," *IEEE Trans. Computational Imaging*, 2017.

[9] F. Natterer, *The Mathematics of Computerized Tomography*, SIAM, 2001.

[10] J. H. Pollard, *A Handbook of Numerical and Statistical Techniques*, Cambridge University Press, 1979.

[11] J. H. Curtiss, "On transformations used in the analysis of variance," *Ann. Math. Statist.*, 1943.

[12] R. Das and A. Rajwade, "Supplemental material: Nonlinear blind compressed sensing under signal-dependent noise," https://tinyurl.com/y5whl2ws.

[13] J. Gorski, F. Pfeuffer, and K. Klamroth, "Biconvex sets and optimization with biconvex functions: a survey and extensions," *Mathematical methods of operations research*, vol. 66, no. 3, pp. 373–407, 2007.

[14] B. Laurent and P. Massart, "Adaptive estimation of a quadratic functional by model selection," *Annals of Statistics*, 2000.