Scientific
Research

Face Recognition from Incomplete Measurements via ℓ_1-Optimization

Miguel Argaez[1*], Reinaldo Sanchez[2], Carlos Ramirez[2]

[1]Department of Mathematical Sciences, The University of Texas at El Paso, El Paso, USA
[2]Program in Computational Science, The University of Texas at El Paso, El Paso, USA
Email: [*]margaez@utep.edu, rsanchezarias@miners.utep.edu, caramirezvillamarin@miners.utep.edu

ABSTRACT

In this work, we consider a homotopic principle for solving large-scale and dense ℓ_1 underdetermined problems and its applications in image processing and classification. We solve the face recognition problem where the input image contains corrupted and/or lost pixels. The approach involves two steps: first, the incomplete or corrupted image is subject to an inpainting process, and secondly, the restored image is used to carry out the classification or recognition task. Addressing these two steps involves solving large scale ℓ_1 minimization problems. To that end, we propose to solve a sequence of linear equality constrained multiquadric problems that depends on a regularization parameter that converges to zero. The procedure generates a central path that converges to a point on the solution set of the ℓ_1 underdetermined problem. In order to solve each subproblem, a conjugate gradient algorithm is formulated. When noise is present in the model, inexact directions are taken so that an approximate solution is computed faster. This prevents the ill conditioning produced when the conjugate gradient is required to iterate until a zero residual is attained.

Keywords: Sparse Representation; ℓ_1 Minimization; Face Recognition; Sparse Recovery; Interior Point Methods; Sparse Regularization

1. Introduction

Over the last years, new developments in the area of computational harmonic analysis have shown that a wide class of signals can be well represented by linear combinations of only few elements of an appropriate basis. The benefits and applications of this new advance abound and are of extensive research in the present.

The new sampling theory of compressed sensing has unified several insights about wavelets and sparse representation, benefiting several disciplines in sciences including image processing. Practical compressed sensing problems involve solving an optimization problem of the form

$$\min_x \|x\|_0 \text{ subject to } Ax = b, \qquad (1)$$

for decoding a sparse signal $x^* \in \mathbb{R}^n$ that has been significantly sub-sampled by a sampling matrix $A \in \mathbb{R}^{m \times n}$ with $m < n$. Here $\|x\|_0$ counts the number of non-zero entries of the vector x.

Solving (1) is equivalent to finding the sparsest vector x such that $Ax = b$. Nevertheless, finding such a vector x is by nature a combinatorial and generally NP-hard

problem [1]. Efficient numerical algorithms to recover signals under this framework have been developed [2-4], and extensions of this theory have been explored for solving general problems in different areas including statistics, signal processing, geophysics, and others. Significant progress toward understanding this problem has been made in recent years [5-7], and its study has become state-of-the-art interdisciplinary research.

One of the most important characteristics of problem (1) is that under some mild conditions, the input vector x^* can be recovered by solving an ℓ_1-norm underdetermined problem

$$\min_x \|x\|_1 \text{ subject to } Ax = b. \qquad (2)$$

This decoding model in compressed sensing is known as the basis-pursuit problem, first investigated by Chen, Donoho and Saunders [8] and theoretically studied by Donoho and Huo [9]. Candès and Tao [5] proved formally this equivalence provided that x^* is sufficiently sparse, and that A possesses certain properties.

The compressed sensing theory is extended to solve a more general problem that considers noise on the measurements, and almost sparsity for the input vector to be recovered. That is,

---

[*]Corresponding author.

$$\min_x \|x\|_1 \text{ subject to } \|Ax - b\| \le \varepsilon, \qquad (3)$$

where the energy of the noise vector is upper bounded by ε.

Problem (3) can also be reformulated as an unconstrained regularized linear least-squares problem. One strategy consists of replacing the Tikhonov regularization with one that uses the ℓ_1-norm [2,4]. Another equivalent formulation is aimed at minimizing the ℓ_1-norm function regularized by a linear leasts-quares problem [3], known as unconstrained basis pursuit denoising. All these approaches have proved to be successful for solving compressed sensing problems.

We reformulate problem (3) by

$$\min_x \|x\|_1 \text{ subject to } Ax + v = b, \qquad (4)$$

where $v \in \square^m$ is bounded by ε.

In this work we propose a new strategy to obtain an optimal solution to problem (4), and present an application in robust face recognition to demonstrate the effecttiveness of our algorithm. The idea consists of relaxing the nondifferentiable objective function by a sequence of multiquadric, continuously differentiable, strictly convex functions that depend on a positive regularization parameter μ. More precisely, we solve

$$\min_x \sum_{i=1}^n \sqrt{x^2 + \mu} \text{ subject to } Ax + v = b. \qquad (5)$$

The main accomplishment of this idea is that it leads to the generation of a path that converges to an optimal solution of problem (4). This leads us to a path-following algorithm similar to the ones used in primal-dual interior-point methods. The path-following strategy that we are proposing uses inexact fixed-point directions to obtain approximate solutions to problems of the form (5). Such inexact directions are computed via a conjugate gradient algorithm. In order to prevent the procedure from becoming costly, a proximity measure to the central path is introduced for each regularization parameter. The regularization parameter is defined in a dynamic manner that converges to zero as in interior-point methods.

2. Problem Formulation

We study the ℓ_1 underdetermined problem (4), where $x \in \square^n$, $b \in \square^m$, $\|x\|_1 = \sum_{i=1}^n |x_i|$, and A is a full-rank matrix with $m < n$. The Lagrangian function associated with problem (4) is

$$l(x, y) = \|x\|_1 + (Ax + v - b)^\mathrm{T} y,$$

where $y \in \square^m$ is the Lagrange multiplier for the equality constraint. The optimality conditions for (4) are given by

$$X^* = \left\{ (x^*, y^*) \in \square^{n+m} : \begin{cases} (A^\mathrm{T} y^*)_i \in \{-1\} & \text{if } x_i^* > 0, \\ (A^\mathrm{T} y^*)_i \in \{1\} & \text{if } x_i^* < 0, \\ \text{and } Ax^* + v = b \\ (A^\mathrm{T} y^*)_i \in [-1, 1] & \text{if } x_i^* = 0, \end{cases} \right\}.$$

Notice that the main role in the characterization of the optimal conditions for problem (4) is not played by the Lagrange multiplier y^*, but by $A^T y^*$. Using this fact, the complementarity conditions associated with the primal variables x_i^* are determined by

$$\begin{cases} x_i^* z_i^* = 0 \text{ with } z_i^* = 1 + (A^\mathrm{T} y^*)_i \text{ if } x_i^* \ge 0 \\ x_i^* z_i^* = 0 \text{ with } z_i^* = 1 - (A^\mathrm{T} y^*)_i \text{ if } x_i^* < 0. \end{cases}$$

Therefore a necessary condition for a feasible point $(x^*, y^*) \in \square^{n+m}$ to be an optimal solution of (4) is

$$\sum_{i=1}^n \tilde{x}_i \tilde{z}_i = 0, \quad \tilde{x} = |x^*| \ge 0 \text{ and } \tilde{z} = \mathbf{1} - |A^\mathrm{T} y^*| \ge 0.$$

3. A Regularization Path

The nondifferentiability of (4) is overcome by regularizing the ℓ_1-norm with a sum of continuously differentiable functions in the following way: for $\mu > 0$ sufficiently small, $\|x\|_1 \approx \sum_{i=1}^n g(x_i)$ where g is the scalar function defined by

$$g(x_i) = \sqrt{x_i^2 + \mu}, \ x_i \in \square.$$

3.1. Optimality Conditions

We propose to obtain an optimal solution to problem (4) by solving a sequence of subproblems of the form (5) as $\mu \to 0$.

Since each subproblem (5) is strictly convex, then the optimal solution is obtained by solving the associated KKT conditions. The Lagrangian function associated with (5) is

$$l(x, y) = \sum_{i=1}^n \sqrt{x_i^2 + \mu} + (Ax + v - b)^\mathrm{T} y, \qquad (6)$$

where $y \in \square^m$ is the Lagrange multiplier for the equality constraint. Therefore, the KKT conditions are given by the following square system of nonlinear equations:

$$F_\mu(x, y) = \begin{bmatrix} D_\mu(x)^{-1/2} x + A^\mathrm{T} y \\ Ax + v - b \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \qquad (7)$$

where $D_\mu(x) = \text{diag}(x_i^2 + \mu)$ for $i = 1, 2, \cdots, n$ and $\mu > 0$.

3.2. A Fixed Point Problem for the KKT Conditions

We propose to solve (7) using a fixed-point method. To

that effect, we rewrite these nonlinear equations as the augmented system

$$\begin{bmatrix} D_\mu(x)^{-1/2} & A^T \\ A & 0 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ b-v \end{bmatrix}, \qquad (8)$$

where $x \in \square^n$, $y \in \square^m$, $D_\mu(x) = \text{diag}(x_i^2 + \mu)$, $i = 1, 2, \cdots, n$, and $\mu > 0$.

The matrix associated with (8) is nonsingular since A has full rank and $D_\mu^{-1/2}(x)$ is positive definite. In this manner, the nonlinear Equation (7) is posed as a *fixed-point problem*. In order to solve (8) for a fixed $\mu > 0$, we proceed by taking an initial point x_0 and iteratively compute $\chi_k = (x_k, y_k)$ for $k = 0, 1, \cdots$ until two consecutive iterations are less than some stopping criteria.

3.3. Inexact Directions for the Augmented System

For a current point x_-, the following system

$$\begin{bmatrix} D_\mu(x_-)^{-1/2} & A^T \\ A & 0 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ b-v \end{bmatrix}, \qquad (9)$$

is reduced to a weighted normal equation. The first block of equations gives $x + D_\mu^{1/2}(x_-)A^T y = 0$, and since $Ax + v = b$, we obtain the weighted normal equation

$$AD_\mu^{1/2}(x_-)A^T y = -b + v. \qquad (10)$$

With this reduction, we move from an indefinite system of order $n + m$ to a positive definite system of order m. Moreover, the conjugate gradient algorithm applied to (10) converges in at most m iterations in exact arithmetic. The solution x of (9) is computed directly by $x = -D_\mu^{1/2}(x_-)(A^T y)$ once y is obtained.

Taking into account that the values of $A^T y$ characterize the optimality set X^*, we formulate a conjugate gradient algorithm that finds an approximation of $A^T y$ rather than y, see **Figure 1**.

At each iteration, the CG algorithm satisfies the first block of equations $D_\mu^{1/2}(x_-)x + A^T y = 0$, therefore controlling the stopping criteria for solving the augmented system (9) is equivalent to controlling the stopping criteria for solving the linear system $Ax - b + v = 0$. Based on this, we define the vector $(0, r) \in \square^{n+m}$ as the residual vector for the augmented system, where $r = Ax - b + v$. Note that $\|r\| = 0$ implies $\|Ax - b\| = \|v\|$. Now, since $\|v\|$ is bounded by ε, then the conjugate gradient algorithm stops when $\|Ax - b\| \le \varepsilon$. This implies the stopping criterion does not need to be zero, overcoming the ill-conditioning of the weighted normal equation close to the solution.

```
Algorithm 1 - Conjugate Gradient
Task: Return the values of A^T y and x
Parameters: We are given the matrices A, D_μ(x_-)^{1/2},
            the vector b and noise level ε

Step 1. Initialization:
        a. Set A^T y = 0, x = 0
        b. Set r = -b, d = r
        c. Set β_n = r^T r
Step 2. For k = 1, ..., cg_maxiter
Step 3. Step length:
        a. Set q_1 = A^T d, q_2 = D_μ(x_-)^{1/2} q_1
        b. Set q_3 = Aq_2
        c. Compute: α_d = d^T q_3, α = β_n/α_d
Step 4. Updates:
        a. A^T y = A^T y + αq_1
        b. x = x - αq_2
Step 5. Residual: r = r - αq_3
Step 6. Stopping criteria: If ‖r‖ ≤ ε end
Step 7. Update direction:
        a. Set β_d = β_n, β_n = r^T r, β = β_n/β_d
        b. Set d = r + βd
```

Figure 1. Conjugate gradient algorithm.

4. Path Following Method

4.1. Central Path

We define the central path associated with problem (4) by

$$C = \left\{ (x_\mu, y_\mu) \in \square^{n+m} : \begin{array}{l} D_\mu^{-1/2}(x_\mu)x_\mu + A^T y_\mu = 0 \\ Ax_\mu + v - b = 0 \end{array} \right\}. \qquad (11)$$

The set C consists of all the points that are solutions of the subproblem (5) for $\mu > 0$. This set defines a smooth curve called the central path that converges to an optimal solution of problem (4) as the regularization parameter μ tends to zero [10].

4.2. Proximity Measures

Our path-following method follows C in the direction generated by a decreasing sequence of regularization parameters μ. Since moving on C to obtain an optimal solution for (4) could be computationally expensive, we restrict the iterates to some neighborhoods of the central path given by

$$\mathcal{N}_{\mu_k} = \left\{ \chi_j \in \square^{n+m} : \frac{\|\chi_j - \chi_{j-1}\|}{1 + \|\chi_j\|} \le \sqrt{\mu_k} \right\}. \qquad (12)$$

4.3. Updating the Perturbation Parameter

Since $\tilde{x}^T \tilde{z} = 0$ is a necessary condition for obtaining an optimal solution of problem (4), then following the same idea of primal-dual methods we define the regularization parameter by

$$\mu = \sigma \frac{\tilde{x}^T \tilde{z}}{n}, \qquad (13)$$

where $\tilde{x} = |x|$, $\tilde{z} = 1 - |A^T y|$, σ is the centering pa-

---

rameter in $(0,1)$, and n is the size of the primal variable x. To guarantee the decrease of the parameter μ, we update it by $\mu = \min\{\mu, \tau\mu_{\text{prev}}\}$ where $\tau \in (0,1)$ and μ_{prev} denotes the previous value for the regularization parameter.

Now, we present a globalization algorithm for solving the convex and nondifferentiable problem (4). The methodology consists in following the central path C to obtain an optimal solution. To prevent the algorithm from becoming computationally expensive a series of neighborhoods, $\mathcal{N}_{\mu_k}$, around the central path are defined to be used as proximity measures of the path for a decreasing regularization parameters μ. To obtain an approximate solution on the path for a given $\mu > 0$, an inexact fixed-point procedure is applied to (8) until a point $(x, y) \in \mathcal{N}_\mu$. If an optimal solution to (4) is not found, we decrease μ, specify a new neighborhood $\mathcal{N}_\mu$ and repeat the fixed-point procedure. An optimal solution for (4) is found as μ, approaches to zero. For the primal variables $\tilde{x}$ we define their corresponding complementarity variables $\tilde{z}$ such that $\tilde{x}_i \tilde{z}_i = 0$ and $\tilde{z}_i \geq 0$ for $i = 1, \cdots, n$ at the optimal solution of problem (4). This allows us to define the regularization parameter μ in the same manner as in interior-point methods.

4.4. Path Following Algorithm

In this section we present our "Path-Following Signal Recovery" (**PFSR**) algorithm. The pseudo-code form of the algorithm is presented in **Figure 2.**

The PFSR algorithm generates two sequences of iterates. The first sequence (inner loop) generates a series of iterates for obtaining an approximate solution of subproblem (5) for a fixed regularization parameter $\mu > 0$. The second sequence (outer loop) generates a series of approximate solutions for the subproblem (5) that converges to an optimal solution of problem (4) for a sequence of decreasing regularization parameters $\mu > 0$.

5. Sparse Signal Recovery

In this section, we present a set of experimental results that illustrate the performance of the MATLAB implementation for the proposed algorithm.

In the implementation of Algorithm 1 the initial points and the parameters are chosen as follows. In Step 1a, the initial points for $A^T y$ and x are the n-dimensional zero vector. In Step 2, we fix the maximum number of CG iterations by *cg_maxiter* = 10.

In the implementation of Algorithm 2 the parameters are chosen as follows. The initial regularization parameter μ is given by

$$\mu = \sigma \sum_{i=1}^{n} |x_i| \left(1 - |A^T b|_i\right),$$


```
Algorithm 2 -The PFSR Algorithm
Task: Find an approximate solution x to the problem
      min_x ||x||_1 subject to Ax + ν = b and ||ν|| ≤ ε
Parameters: We are given the matrix A and the vector b
Step 1.  Initialization: Set: σ, τ, μ, ε
Step 2.  Initial approximate solution x = A^T b
Step 3.  Outer Loop: for j = 1,...,maxiter
Step 4.  Inner Loop: Set x_- = x
Step 5.  Update weight matrix: D_μ(x_-) = diag(x_-^2 + μ)
Step 6.  Solve the augmented system using Algorithm 1:
         [ D_μ(x_-)^{-1/2}   A^T ] [ x ]   [ 0 ]
         [ A                 0   ] [ y ] = [ b ]
Step 7.  Check proximity to the central path:
         if (x, y) = X ∉ N_μ go to step 4
Step 8.  Set x̃ = |x|, w = A^T y, z̃ = (1 - |w|)
Step 9.  Compute  error_primal = ||Ax - b||, gap = x̃^T z̃ / n
Step 10. Stopping criteria for the problem:
         if (error_primal or gap) > ε
             Update μ = min{σgap, τμ}, go to step 3
         else
             display "x is an optimal solution"
```

Figure 2. PFSR algorithm.

where x is the minimum energy reconstruction. Our numerical experimentation suggests $\sigma = 0.008$ as a good choice. In Step 3, we set *maxiter* = 10 as the maximum number of subproblems of the form (5) to solve. The new regularization parameter μ is updated in Step 10 by $\mu = \min\{\sigma gap, \tau\mu\}$ with $\tau = 0.9$.

5.1. Sparse Signal Recovery Example

This experimentation has the objective of investigating the capability of recovering sparse signals by our PFSR algorithm. The goal in this test is to reconstruct a length-n sparse signal from m observations, where $m < n$. We start with a classical example also considered in [2,4,11]. The problem consists of recovering a signal $x \in \square^{4096}$ with 160 spikes with amplitude ± 1, from $m = 1024$ noisy measurements. We use a partial DCT matrix A whose m rows are chosen randomly from the $n \times n$ discrete cosine transform, without having access to it in explicit form, but using A as a linear operator on $\square^n$. The same for the matrix A^T. Partial DCT matrices are fast transforms for which matrix-vector multiplications cost just $O(n \log n)$ flops, and storage is not required. This case is common for compressed sensing [3,4]. In this problem, we have $b = Ax + \nu \in \square^m$, where the noise vector ν is set according to a Gaussian distribution with mean 0 and standard deviation 0.01. The original and reconstructed signal are shown in **Figure 3**. Moreover, the algorithm was successfully run one hundred times with an average CPU time of 0.2895 seconds, and 0.0491 average relative 2-norm error defined by

$$\text{2-norm error} = \frac{\|x - x^*\|_2}{\|x^*\|_2},$$

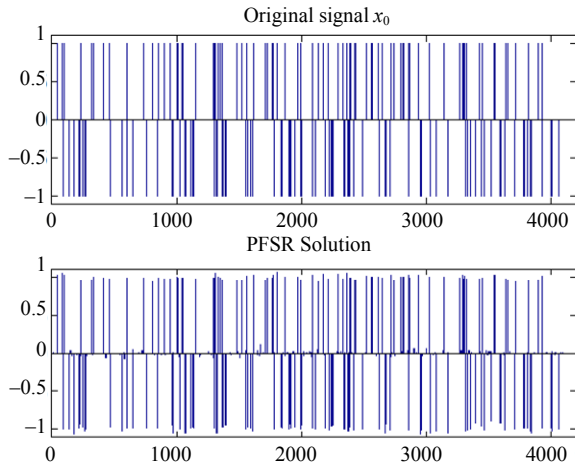being x^* the true solution and x the solution reached by the PFSR algorithm.

Original signal x_0

PFSR Solution

Figure 3. Signal reconstruction.

5.2. The Effect of Noise

For the same test problem described above, we run the algorithm considering the noiseless case, and zero mean Gaussian noise with standard deviations varying from 0.01 to 0.04. The stopping criterion for the conjugate gradient residual is determined by the noise level as explained in Section 3.3. In all cases we successfully recover the original signal after a process of thresholding. **Table 1** reports the results for this experimentation showing that the algorithm is also effective for solving noiseless sparse recovery problems. Moreover, successful recovery was obtained with noise level up to 0.04.

6. Sparse Representation in Classification and Image Processing

There exist a number of areas where the sparse representtation model (4) emerges naturally. In classification problems for instance, an input sample can usually be explained from few other previously trained samples. That is, an incoming sample b can be expressed as a linear combination of only few columns $A_i, i \in \Gamma \subseteq \{1, \cdots, n\}$, where the matrix $A \in \square^{m \times n}$, $m < n$, is a matrix whose columns are previously trained samples.

On the other hand, natural images can be efficiently encoded in an appropriate basis that exploits the hight correlation present between pixels. Among many emerging class of transformations, the Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) are a standard basis where natural images can be sparsely represented. This property has been extensively studied in recent years, leading to a construction of new models for images where the sparsity is incorporated in the model as a regularizer.

6.1. Classification

In pattern recognition and machine learning, a classifica-

Table 1. Performance considering different setups for noise.

Noise	0.00	0.01	0.02	0.03	0.04
2-norm error	0.016	0.063	0.121	0.208	0.294
Recovery	100%	100%	100%	100%	100%
Time [s]	0.4056	0.3432	0.3744	0.4056	0.4836

tion problem consists of finding an algorithm for assigning a given input data to one of several categories. Formally, given an input dataset, $W = \{w_1, \cdots, w_n\}$, a set of labels/classes $T = \{t_1, \cdots, t_n\}$, and a training dataset $D = \{(u_i, t_i) : i = 1, \cdots n\}$ such that t_i is the label/class of the sample u_i, a classifier is a mapping f from W to T, assigning the correct label $t \in T$ to a given input w, that is, $f(D, w) = t$.

Consider a training data set $\{(u_i, t_i) : i = 1, \cdots, n\}$, $u_i \in \square^d$, $t_i \in \{1, 2, \cdots, N\}$, with n being the number of samples and N the number of classes. The vector $u_i \in \square^d$, represents the ith sample, and t_i is the corresponding label.

The sparse representation problem is formulated as follows: For a testing sample $b \in \square^d$, find the sparsest vector $x = [x_1, x_2, \cdots, x_n]^T$ such that

$$b = x_1 u_1 + x_2 u_2 + \cdots + x_n u_n. \tag{14}$$

We show that indeed a valid test sample can be represented using only the training samples from its same class, therefore inducing a natural sparse representation. Let us rearrange the given n_i training samples from the same i-th class as the columns of a submatrix $A_i = \left[u_{i,1}, u_{i,2}, \cdots, u_{i,n_i} \right] \in \square^{d \times n_i}$. In other words, we group all of those samples with the same label into a matrix A_i. Any test sample b from the same class will be represented as a linear combination of the training samples associated with class i:

$$b = x_{i,1} u_{i,1} + x_{i,2} u_{i,2} + \cdots + x_{i,n_i} u_{i,n_i}, \tag{15}$$

for some values of $x_{i,j} \in \square$, $j = 1, \cdots, n_i$. Now, making use of the whole training dataset, we define a $d \times n$ matrix A by concatenating all of the n training samples of the different N classes, that is $A = [A_1, A_2, \cdots, A_N]$. Then, the linear representation of the test sample b that belongs to class i is written by:

$$b = Ax, \tag{16}$$

where $x = \left[0, \cdots, 0, x_{i,1}, x_{i,2}, \cdots, x_{i,n_i}, 0, \cdots, 0 \right]^T \in \square^n$. Thus, the test sample b is expressed by a sparse linear combination of the training samples, more specifically, as a linear combination of only those training samples belonging to the same class. This motivates us to formulate the following problem:

$$\min \|x\|_1 \text{ subject to } Ax = b, \quad (17)$$

which follows the same structure as (2).

One of the advantages of this formulation is that the lack of robustness with respect to outliers can be overcome. Furthermore, we do not need to care for model selection as in support vector machine approaches for classification problems [13].

6.1.1. Discriminant Functions and Classifier

Once the sparse representation vector x is computed, we identify the class to which the testing sample b belongs. The approach consists in associating the nonzero entries of x with the columns of A corresponding to those training samples having the same class of the testing sample b. The solution vector x is decomposed as the sum of d-dimensional vectors $\hat{x}_k$, where $\hat{x}_k$ is obtained by keeping only those entries in x associated with class k and assigning zeros to all the other entries. Then, we define the N discriminant functions

$$g_k(b) = \|b - A\hat{x}_k\|_2, \quad k = 1, \cdots, N. \quad (18)$$

Thus, g_k represents the approximation error when b is assigned to category k. Finally, we assign b to the class with the smallest approximation error. That is,

$$\hat{t} = \arg\min\{g_k(b)\}, \quad k = 1, \cdots, N. \quad (19)$$

In this manner, we identify the class of the test sample b based on how effectively the coefficients associated with the training samples of each class recreate b.

6.1.2. Cross Validation

A classifier performance is commonly measured by its error rate on the entire population. Cross Validation is a statistical method for evaluating performance in which the data is divided in two sets: one used for the training stage, and the second one used for testing (validation). Both training and testing sets should cross-over in consecutive rounds in such a way that each sample in the data set has a chance of being validated.

In the case of K-fold cross validation, a K-fold partition of the dataset is created by splitting the data into K equally (nearly equal) sized subsets (folds), and then for each of the K experiments, $K - 1$ folds are used for training and the remaining one for testing. A common choice for K-Fold cross validation is $K = 10$. The work in [12], compares several approaches for estimating accuracy, and recommends stratified 10-fold cross-validation as the best model selection method because it provides less biased estimation of the actual accuracy. In our numerical experimentation we follow this validation approach to test the performance of the classification algorithm.

6.2. Image Processing

In many practical image processing applications, we are interested in recover a target image $u \in \square^n$, that has been subject to a degradation processes modeled as

$$b = Hu + v, \quad (20)$$

where $b \in \square^m$ is the degraded image, v is additive noise of certain distribution, and H is a linear operator that acts on the target image u. For instance, H can be a convolution operator that models atmospheric turbulence, or defects on the acquisition process due to movement.

From a point of view of parameter estimation, problem (20) corresponds to an inverse problem which is very difficult to solve due to the ill-conditioning nature of H and the large number of degrees of freedom present in the image. Consequently, additional information is required in order to obtain a meaningful solution to (20). This is accomplished with the ℓ_1 norm regularization, and our problem becomes

$$\min_x \|x\|_1 \text{ subject to } H\Psi x + v = b, \quad (21)$$

where Ψ is an sparsifying matrix for u, and x is a sparse vector of coefficients. That is, $u = \Psi x$.

When H models a process of missing information, it receives the name of mask, and is constructed by removing from the $n \times n$ identity matrix, the $n - m$ rows associated with the missing data. In this setting, we consider that $n - m$ pixels has been lost, leading to an incomplete image $b \in \square^m$.

7. Robust Face Recognition

In this section, we demonstrate the effectiveness of the proposed algorithm by showing a real application in robust face recognition. The challenge consists in automatically identify an input human face image within an existing database of several individuals [14]. Furthermore, we assume that the input image has been subject to a data loss process, or that several of its pixels has been severely corrupted.

We consider the database of human faces from the AT&T laboratories in Cambridge (Cambridge University Computer Laboratory)[1] which consists of 400 images of 40 individuals each of which with ten different images. For some individuals, the images were taken at different times, lighting and facial expression. The size of each image is 112×92 pixels with 256 gray levels in a pgm format.

The proposed approach involves two steps. First, the test image containing several corrupted or lost pixels is reconstructed via inpainting. To that end, we solve problem (21) where b is the corrupted test image, Ψ is the wavelet Daubechies level 7 matrix, and H is the matrix

---

[1]http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

(mask) associated with the missing pixels.

Secondly, we exhibit the reconstructed test image to a training dataset in order to find the sparsest linear combination of the training samples that better represents the test image. Both processes are carried out by solving an ℓ_1 minimization problem.

Figure 4 depicts the recognition process where the test image is corrupted by two different kind of masks.

In our numerical experiments, we perform a cross validation scheme in order to assess a recognition rate for a total of 10-fold cross validation runs, each of which solves 40 test problems. None of these experiments included the test image in the training dataset. We consider three different types of corruptive masks: 1) Random missing pixels uniformly distributed over the test image, 2) Corruptive horizontal lines and 3) Corruptive vertical lines. Despite **Figure 4** shows two corrupt test images with a total of 34.12% (top) and 15.18% (bottom) respectively, in our numerical experiments we consider a percentage of lost data ranging from 5 to 40%. The average recognition rate obtained after the cross validation experiments was 97.25%, and **Table 2** reports each of these rates per fold.

In **Figure 5** one can notice that the solution vector x for problem (17) is sparse, and its nonzero components mark those training samples u_i in the dictionary A that are from the same class of the test sample b, that is, those other pictures of the individual b in the training dataset.

8. Concluding Remarks

In this work we present a novel methodology for solving large-scale and dense ℓ_1 underdetermined problems. For solving the large-scale and dense linear systems associated with the problem, a conjugate gradient algorithm is formulated. The regularization parameter is implemented in the same fashion as in interior-point methods by characterizing the complementarity variables associated with the primal variables of the problem. We present a numerical experimentation that shows that our algorithm is capable to perform efficiently when recovering large signals with noisy data. The sparse representation approach is applied to a face recognition problem where the data is incomplete and/or corrupted. An inpainting procedure is carried out for reconstructing the input image, and then a classification process is performed in order to identify the correct individual. Both processes are accomplished with the proposed algorithm for ℓ_1 minimization problems, achieving a high recognition rate.

9. Acknowledgements

Incomplete test image   Inpainted test image   PFSR recognition
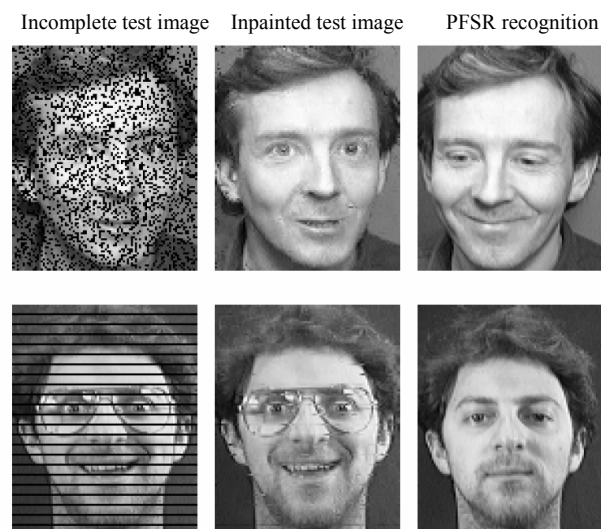
Figure 4. Classification scheme.

Table 2. Recognition rate per fold.

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Recognition rate	100%	97.5%	95%	97.5%	95%
	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10
Recognition rate	95%	100%	95%	100%	97.5%
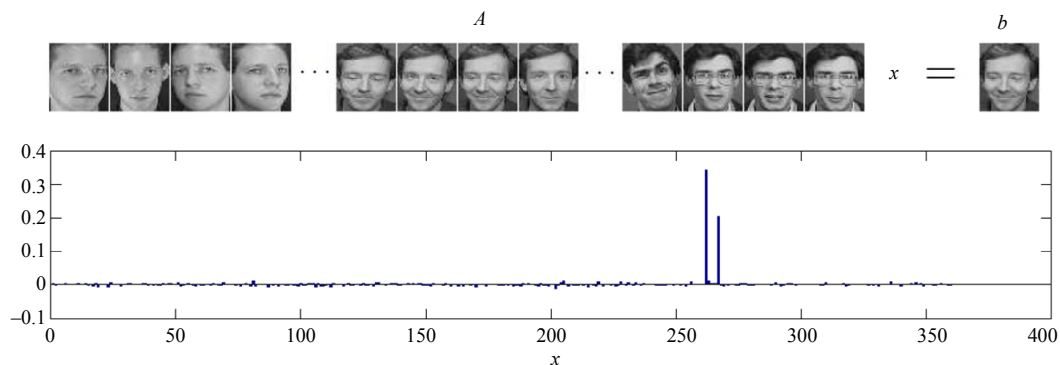
Figure 5. Sparse linear combination.

REFERENCES

[1] B. K. Natarajan, "Sparse Approximate Solutions to Linear Systems," *SIAM Journal on Computing*, Vol. 24, No. 2, 1995, pp. 227-234. doi:10.1137/S0097539792240406

[2] M. Figueiredo, R. Nowak and S. Wright, "Gradient Projection for Sparse Reconstruction: Application to Compressed Sensing and Other Inverse Problems," *IEEE Journal of Selected Topics in Signal Processing*, Vol. 1, No. 4, 2007, pp. 586-597.

[3] E. Hale, W. Yin and Y. Zhang, "A Fixed-Point Continuation Method for ℓ_1 Regularized Minimization with Applications to Compresses Sensing," Technical Report TR07-07, Department of Computational and Applied Mathematics, Rice University, Houston, 2007.

[4] S. Kim, K. Koh, M. Lustig, S. Boyd and D. Gorinvesky, "A Method for Large-Scale ℓ_1-Regularized Least Squares Problems with Applications in Signal Processing and Statistics," *IEEE Journal of Selected Topics in Signal Processing*, 2007. www.stanford.edu/˜boyd/l1_ls.html

[5] E. Candès, J. Romberg and T. Tao, "Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information," *IEEE Transactions on Information Theory*, Vol. 52, No. 2, 2006, pp. 489-509. doi:10.1109/TIT.2005.862083

[6] S. Chen, D. Donoho and M. Saunders, "Atomic Decomposition by Basis Pursuit," *SIAM Review*, Vol. 43, No. 1, 2001, pp. 129-159. doi:10.1137/S003614450037906X

[7] D. Donoho, "Compressed Sensing," *IEEE Transactions on Information Theory*, Vol. 52, No. 4, 2006, pp. 1289-1306. doi:10.1109/TIT.2006.871582

[8] S. Chen, D. Donoho and M. Saunders, "Atomic Decomposition by Basis Pursuit," *SIAM Review*, Vol. 43, No. 1, 2001, pp. 129-159. doi:10.1137/S003614450037906X

[9] D. Donoho and X. Huo, "Uncertainty Principles and Ideal Atomic Decomposition," *IEEE Transactions on Information Theory*, Vol. 47, No. 7, 2001, pp. 2845-2862. doi:10.1109/18.959265

[10] M. Argáez, C. Ramirez and R. Sanchez, "An ℓ_1 Algorithm for Underdetermined Systems and Applications," *IEEE Conference Proceedings on North American Fuzzy Information Processing Society*, El Paso, 18-20 March 2011, pp. 1-6. doi:10.1109/NAFIPS.2011.5752016

[11] S. Wright, R. Nowak and M. Figueiredo, "Sparse Reconstruction by Separable Approximation," *IEEE International Conference on Acoustics*, *Speech and Signal Processing*, ICASSP 2008, Las Vagas, 31 March-4 April 2008. doi:10.1109/ICASSP.2008.4518374

[12] R. Kohavi, "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection," *Proceedings of International Joint Conference on AI*, Quebec, 20-25 August 1995, pp. 1137-1145.

[13] R. Sanchez, M. Argáez and P. Guillen. "Sparse Representation via ℓ_1-Minimization for Underdetermined Systems in Classification of Tumors with Gene Expression Data. *IEEE 33rd Annual International Conference Proceedings of the Engineering in Medicine and Biology Society*, Boston, 30 August-3 September 2011, pp. 3362-3366.

[14] J. Wright, Y. Yang, A. Ganesh, S. Shankar and Y. Ma, "Robust Face Recognition via Sparse Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 31, No. 2, 2009, pp. 210-227. doi:10.1109/TPAMI.2008.79