A Closed-Form Solution to the Regression Problem with Sparse Outliers in the Data

Andrew E. Yagle
Department of EECS, The University of Michigan, Ann Arbor, MI 48109-2122

Abstract—**We show how to solve the regression problem with sparse outliers in the data by solving a linear system of equations. The fraction of data points corrupted by outliers at unknown indices must be less than the reciprocal of the model order plus one. No statistical properties for the outliers or system matrix are required. A simple example of fitting a line to data with outliers added to one-third of its data points, and a Matlab program for the general problem, are given.**

Keywords— **Regression, outliers, sparse.**
Phone: 734-763-9810. Fax: 734-763-1503.
Email: aey@eecs.umich.edu.

I. INTRODUCTION

A. Problem Statement

The regression problem is to fit a model

$$y = \sum_{j=1}^{M} a_j \phi_j(x) + n \tag{1}$$

to N noisy data points $\{(x_i, y_i), i = 1 \ldots N\}$.

The model (1) features

- M unknown parameters $\{a_j, j = 1 \ldots M\}$; and
- M known basis functions $\{\phi_j(x), j = 1 \ldots M\}$.

The data features

- N known values $\{x_i, i = 1 \ldots N\}$ and
- N known values $\{y_i, i = 1 \ldots N\}$ which include
- N noise values $\{n_i, i = 1 \ldots N\}$ for which the
- K noise values $\{n_i, i = 1 1_1 \ldots i_K\} \neq 0$ for the
- K unknown indices $\{n_i, i = i_1 \ldots i_K\}$.

The goal is to compute the M unknown parameters $\{a_j\}$ from the noisy data $\{y_i\}$.

B. Contribution of This Paper

This paper solves the specific case where the noise $\{n_i\}$ is *sparse* (mostly zero), so that the noisy data has K *outliers* added to it at K *unknown* indices i.

Specifically, if the *fraction* $\frac{K}{N}$ of outliers in the data is less than $\frac{1}{M+1}$, then the unknown parameters $\{a_j\}$ are computed by solving an $(N-1)\times(N-1)$ linear system of equations. $M+1$ N-point discrete Fourier transforms are also required.

C. Another Approach: LAD

The usual procedure for solving this problem is to use an LAD (Least Absolute Deviation) criterion

$$LAD = \sum_{i=1}^{N} |y_i - \sum_{j=1}^{M} a_j \phi_j(x_i)|. \tag{2}$$

LAD minimizes the ℓ_1 norm of the error, and thus sparsifies the $\{n_i\}$, if the number of outliers is small.

The LAD solution can be computed eaily using linear programming. However, the linear programming problem has at least $2N$ variables (due to the absolute values in the cost criterion) and only works for a small number of outliers.

The procedure of this paper requires much less computation than LAD, and it is non-iterative. It can also handle more outliers than LAD if M is very small, as if often the case.

II. PROBLEM SOLUTION

A. Matrix Formulation

Define the $N \times M$ matrix H as

$$H_{ij} = \phi_j(x_i), \quad i = 1 \ldots N, \quad j = 1 \ldots M. \tag{3}$$

and assume that the basis functions $\{\phi_j\}$ and $\{x_i\}$ are such that H has full rank. Also define y, n and a as the column vectors

$$\begin{aligned} y &= [y_1, y_2 \ldots y_N]^T \\ n &= [n_1, n_2 \ldots n_N]^T \\ a &= [a_1, a_2 \ldots a_M]^T \end{aligned} \tag{4}$$

Then the problem can be reformulated as solving

$$y = Ha + n \tag{5}$$

where $n_i = 0$ for all i except for the K indices $i = \{i_1, i_2 \ldots i_K\}$ at which n_i is nonzero and unknown. This can be rearranged into

$$-n = Ha - y = [H - y] \begin{bmatrix} a \\ 1 \end{bmatrix}. \tag{6}$$

B. Discrete Fourier Transform Review

Recall that the N-point Discrete Fourier Transform (DFT) of N numbers $\{x_i, i = 1 \ldots N\}$ is the N numbers $\{X_k, k = 1 \ldots N\}$ where $j=\sqrt{-1}$ and

$$X_k = \sum_{i=1}^{N} x_i e^{-j2\pi(i-1)(k-1)/N}, k = 1 \ldots N. \quad (7)$$

The DFT can be regarded as multiplication of the column vector x of $\{x_i\}$ by the $N \times N$ matrix D whose $(k,i)^{th}$ element is $D_{k,i}=e^{-j2\pi(i-1)(k-1)/N}$, yielding the column vector $\hat{x}$ of $\{X_k\}$, i.e., $\hat{x}=Dx$.

The DFT of the product $\{x_i y_i\}$ is the *circular* or *cyclic* convolution of the DFT's of the $\{x_i\}$ and $\{y_i\}$:

$$\begin{aligned} \text{DFT}\{x_i y_i\} &= \sum_{i=1}^{N} x_i y_i e^{-j2\pi(i-1)(k-1)/N} \\ = \frac{1}{N} X_k \copyright Y_k &= \frac{1}{N} \sum_{m=1}^{N} X_m Y_{\substack{k-m \\ \bmod N}} \end{aligned} \quad (8)$$

C. Indicator Function

Let $\{s_i\}$ be the *indicator function* for nonzero $\{n_i\}$:

$$\begin{cases} s_i = 0 & \text{if } n_i \neq 0 \\ s_i \neq 0 & \text{if } n_i = 0 \end{cases} \quad \begin{cases} S_k \neq 0 & 1 \leq k \leq K \\ S_k = 0 & \text{otherwise} \end{cases} \quad (9)$$

where $\{S_k\}$ is the N-point DFT of $\{s_i\}$. Then

$$s_i n_i = 0 \rightarrow \sum_{m=1}^{N} N_m S_{\substack{k-m \\ \bmod N}} = 0, k = 1 \ldots N \quad (10)$$

where $\{N_k\}$ is the N-point DFT of $\{n_i\}$.

This states that multiplication of the column vector $\hat{n}$ of $\{N_k\}$ by a *circulant* matrix $\hat{S}$ whose 1^{st} column is the vector of $\{S_k\}$ is a column vector of zeros:

$$\hat{S}\hat{n} = [0 \ldots 0]^T. \quad (11)$$

The indicator function $\{s_i\}$ exists and is unique to a scale factor. In fact, it can be shown that

$$s_i = \prod_{n=1}^{K} (e^{j2\pi i/N} - e^{j2\pi i_n/N}). \quad (12)$$

D. Derivation of Equations

Taking the N-point DFT of each column of (6), or premultiplying (6) by the DFT matrix D gives

$$-\hat{n} = [\hat{H} - \hat{y}] \begin{bmatrix} a \\ 1 \end{bmatrix} \quad (13)$$

where $\hat{H}$ is the $N \times M$ matrix whose j^{th} column is the DFT of the j^{th} column of H.

Let $\hat{S}$ be the circulant matrix whose 1^{st} column is the vector of $\{S_k\}$. Premultiplying (13) by $\hat{S}$ and using (11) gives

$$[0 \ldots 0]^T = \hat{S}[\hat{H} - \hat{y}] \begin{bmatrix} a \\ 1 \end{bmatrix}. \quad (14)$$

This can be rearranged into Toeplitz-blocks system

$$[\hat{H}_1 | \hat{H}_2 | \ldots | -\hat{Y}]z = 0 \quad (15)$$

with the $(M{+}1)$ $N \times (K{+}1)$ Toeplitz matrices $\hat{H}_k$

$$\hat{H}_k = \begin{bmatrix} \hat{H}_{1,k} & \hat{H}_{N,k} & H_{N-1,k} & \ldots \\ \hat{H}_{2,k} & \hat{H}_{1,k} & H_{N,k} & \ldots \\ \ddots & \ddots & \ddots & \ddots \\ \hat{H}_{N,k} & \hat{H}_{N-1,k} & H_{N-2,k} & \ldots \end{bmatrix} \quad (16)$$

and the $(M{+}1)(K{+}1)$-column vector z is defined as the column concatenation of the column vectors

$$\begin{aligned} z &= [z_1, z_2 \ldots z_{K+1}]^T \\ z_1 &= a_1[S_1, S_2 \ldots S_{K+1}]^T \\ z_2 &= a_2[S_1, S_2 \ldots S_{K+1}]^T \\ z_M &= a_M[S_1, S_2 \ldots S_{K+1}]^T \\ z_{M+1} &= 1[S_1, S_2 \ldots S_{K+1}]^T. \end{aligned} \quad (17)$$

E. Derivation of Algorithm

The system of equations (15) has N equations in $(M{+}1)(K{+}1)$ unknowns. If we have condition

$$N \geq (M+1)(K+1) \quad (18)$$

then the null vector of the matrix in (15) exists and is unique. This condition is equivalent to

$$\frac{K}{N} < \frac{K+1}{N} \leq \frac{1}{M+1} \quad (19)$$

which states that the fraction $\frac{K}{N}$ of outliers in the data should be less than $\frac{1}{M+1}$, the reciprocal of the number of parameters plus one. For example, a line $(M{=}2)$ can be fit correctly to noisy data even when *one-third of the data are corrupted by outliers!*

Then the unknown parameters $\{a_j\}$ can be read off of the null vector z after normalization, as follows:

1. Wrap null vector z into a $(K{+}1)\times(M{+}1)$ array;
2. Read off the top row of this array;
3. Divide this row by its last element, since this is known to be unity by (13). This gives $\{a_j\}$.

Instead of computing a null vector, we can use the last column of the matrix in (15) as the right side of an $(N{-}1)\times(N{-}1)$ linear system of equations.

III. NUMERICAL EXAMPLE

A. Matlab Program

The following Matlab program was used to generate the example to follow, but it is easily adapted to the general problem. The user must supply H and data (x_i, y_i) with outliers added to some of the $\{y_i\}$.

To match the paper, the null vector is computed.

```
clear;M=2;N=24;K=N/(M+1)-1;%Can change
H=[[1:N]' ones(N,1)];A=[3 2]';Y=H*A;
II=[3 6 8 11 13 18 20];%Outlier index
Y(II)=Y(II)+10*[3 1 4 1 5 2 3]';
%GOAL: Compute A from Y+outliers.
FH=fft([H -Y]);GG=[];for J=1:M+1;
G=gallery('circul',FH(:,J)');
G=G';GG=[GG G(:,1:K+1)];end
N2=null(GG);N3=reshape(N2,K+1,M+1);
AHAT=N3(1,1:M)/N3(1,M+1)%Computed A
plot(1:N,Y,'x',1:N,real(H*A),'+')
```

B. Example: Fitting a Line

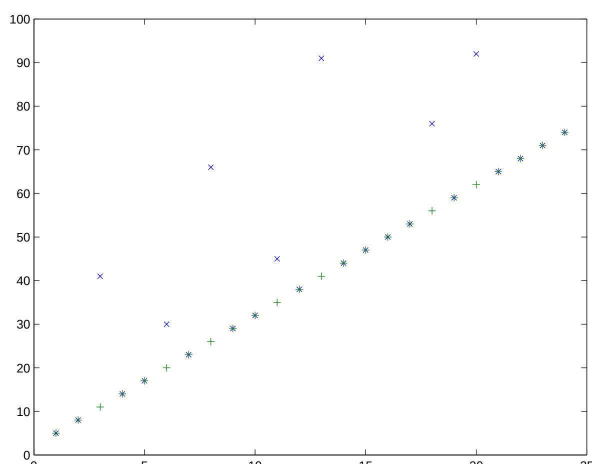
The above program uses the linear model

$$y = 3x + 2 \tag{20}$$

to generate 24 data points for $x_i = 1 \ldots 24$. It then adds outliers of various sizes to 7 of the 24 values of the data y_i. The noisy data points are plotted as x's.

It is apparent that a least-squares fit will be poor. Nor can LAD be expected to work with this many outliers, since the noise $\{n_i\}$ is not very sparse.

The program was then run. The estimated values AHAT are the correct values $[3, 2]$. The (correct) data points generated by the model are plotted as +'s.



IV. SPECIAL CASE: TWO PARAMETERS

A. Derivation of Algorithm

If the model has only two unknown parameters, the following algorithm can be used instead of the above. Recall the basic problem can be written as (6), which in the two-parameter case becomes

$$-n = [H_1 | H_2 | -y] \begin{bmatrix} a_1 \\ a_2 \\ 1 \end{bmatrix} \tag{21}$$

where H_1, H_2, y, n are column N-vectors and $\frac{N}{3}$−1 or fewer values of n are nonzero. Let N be a multiple of 3. Partition (21) into $\frac{N}{3}$ rows of 3×3 blocks:

$$-\begin{bmatrix} n_1 \\ \vdots \\ n_{\frac{N}{3}} \end{bmatrix} = \begin{bmatrix} G_1 \\ \vdots \\ G_{\frac{N}{3}} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ 1 \end{bmatrix} \tag{22}$$

where the $\frac{N}{3}$ 3×3 matrices G_i each have the form

$$G_1 = \begin{bmatrix} H_{11} & H_{12} & -y_1 \\ H_{21} & H_{22} & -y_2 \\ H_{31} & H_{32} & -y_3 \end{bmatrix} \tag{23}$$

Since there are only $\frac{N}{3}$−1 or fewer nonzero values of n, by the pigeonhole principle at least one of the G_i is singular. Computing the determinant of each 3×3 matrix G_i requires 9 multiplications, for a total of $9\frac{N}{3}$=$3N$ multiplications. Once the outlier-free block is determined, a_1 and a_2 can be computed quickly.

B. Example

We wish to fit a model of the form

$$y = ax + bx^2 \tag{24}$$

to the six data points

x	1	2	3	4	5	6
y	5	14	30	44	65	90

$\frac{6}{3}$−1=1 of which is an outlier.
One of these two matrices is singular:

$$G_1 = \begin{bmatrix} 1 & 1 & 5 \\ 2 & 4 & 14 \\ 3 & 9 & 30 \end{bmatrix}; G_2 = \begin{bmatrix} 4 & 16 & 44 \\ 5 & 25 & 65 \\ 6 & 36 & 90 \end{bmatrix}. \tag{25}$$

In fact, the second matrix G_2 is singular, so

$$\begin{bmatrix} 4 & 16 & 44 \\ 5 & 25 & 65 \\ 6 & 36 & 90 \end{bmatrix} \begin{bmatrix} a \\ b \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{26}$$

from which a=3 and b=2. 30 was the outlier.