# CLEANING UP TOXIC WASTE: REMOVING NEFARIOUS CONTRIBUTIONS TO RECOMMENDATION SYSTEMS

*Adam Charles, Ali Ahmed, Aditya Joshi, Stephen Conover, Christopher Turnes, Mark Davenport*

Georgia Institute of Technology
Electrical and Computer Engineering

## ABSTRACT

Recommendation systems are becoming increasingly important, as evidenced by the popularity of the Netflix prize and the sophistication of various online shopping systems. With this increase in interest, a new problem of nefarious or false rankings that compromise a recommendation system's integrity has surfaced. We consider such purposefully erroneous rankings to be a form of "toxic waste," corrupting the performance of the underlying algorithm. In this paper, we propose an adaptive reweighted algorithm as a possible approach towards correcting this problem. Our algorithm relies on finding a low-rank-plus-sparse decomposition of the recommendation matrix, where the adaptation of the weights aids in rejecting the malicious contributions. Simulations suggest that our algorithm converges fairly rapidly and produces accurate results.

***Index Terms***— Adaptive optimization, sparsity, convergence, toxic waste

## 1. INTRODUCTION

With expanding storage capabilities, commercial enterprises are gaining access to large amounts of data on consumer preferences. Accordingly, there has been significant interest in developing strong recommendation systems for product marketing. This increased attention has led to a flurry of innovation, a good percentage of which was motivated by the celebrated Netflix prize [1].

Unfortunately, the success of algorithms for recommendation systems has, on occasion, drawn unwanted attention. There are documented instances of enterprises manipulating rating systems to increase the appeal of their products [2, 3, 4]. We refer to the data from such contributions as "toxic waste," as it is in some sense worse than more standard error sources. While normal errors may cause mild ambiguity, these errors not only increase the required storage but also actively contribute misinformation to the system. Toxic waste worsens the performance of recommendation systems at no fault of their underlying algorithms.

In this paper, we explain how such nefarious contributions may be mitigated through statistical techniques. It is well known that recommendation systems are accurately modeled as having two primary components: one that is low-rank, the other sparse. Our approach models the aggregate matrix of product rankings in precisely this manner, with the corrupting agents affecting only a sparse subset of the rows of the system (and therefore contributing to the sparse component of the decomposition). By identifying this subset with an adaptive algorithm, we are able to remove its contribution and improve the quality of the data fed to the recommendation system.

## 2. BACKGROUND

The algorithm we propose combines the ideas of two existing methods. To obtain the desired low-rank-plus-sparse decomposition of the recommendation matrix, we solve a specific optimization problem known as *Principal Components Pursuit* (PCP). This problem may be solved with an optimization tool known as the *Alternating Directions* (AD) algorithm. Our innovation is a variation on the optimization problem that incorporates an adaptive reweighting process, allowing us to more selectively narrow in on the components corresponding to toxic waste.

### 2.1. Low-rank-plus-sparse decompositions

The decomposition of a matrix into sparse and low-rank components is an approach with well established precedence [5, 6]. For a given matrix $\boldsymbol{Y} \in \mathbb{R}^{n_1 \times n_2}$, the decomposition into these components is written as

$$\boldsymbol{Y} = \boldsymbol{L}_0 + \boldsymbol{S}_0, \tag{1}$$

where $\boldsymbol{L}_0$ is the rank-$r$ component (with $r$ small) and $\boldsymbol{S}_0$ is sparse. Let the singular value decomposition (SVD) of $\boldsymbol{L}_0$ be written as

$$\boldsymbol{L}_0 = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*,$$

with $\boldsymbol{U} : n_1 \times r$, $\boldsymbol{\Sigma} : r \times r$, and $\boldsymbol{V} : n_2 \times r$. Following established results, we assume that the SVD obeys three coherence conditions:

$$\max_i \|\boldsymbol{U}\boldsymbol{U}^*\|_\infty^2 \leq \frac{\mu R}{n_1}, \quad \max_i \|\boldsymbol{V}\boldsymbol{V}^*\|_\infty^2 \leq \frac{\mu R}{n_2}, \tag{2}$$

and

$$\|\boldsymbol{U}\boldsymbol{V}^*\|_\infty^2 \leq \frac{\mu R}{n_1 n_2}, \tag{3}$$

where we have used the norm $\|\boldsymbol{Y}\|_\infty = \max_{ij} |X_{ij}|$. The coherence parameter $\mu$ quantifies the dispersion of the singular vectors [7], and is $O(1)$ when the vectors are perfectly spread-out. The coherence conditions effectively ensure that the low-rank matrix is non-sparse, as this would create ambiguity in the decomposition.

Theorem 1 of [6] states that when the sparsity pattern in the matrix $\boldsymbol{S}_0$ is selected uniformly at random[1], $\boldsymbol{L}_0$ obeys the incoherence assumptions of (2) and (3). As a result, we can obtain an *exact recovery* of the components of the decomposition by solving the PCP optimization program

$$\begin{aligned} \text{minimize} \quad & \|\boldsymbol{L}\|_* + \lambda \|\boldsymbol{S}\|_1 \\ \text{subject to} \quad & \boldsymbol{L} + \boldsymbol{S} = \boldsymbol{Y} \end{aligned}$$

with regularity parameter $\lambda = (\max(n_1, n_2))^{-1/2}$. In other words, the matrices $\hat{\boldsymbol{L}}$ and $\hat{\boldsymbol{S}}$ returned by the optimization program will exactly equal $\boldsymbol{L}_0$ and $\boldsymbol{S}_0$ so long as

$$\text{rank}(\boldsymbol{L}_0) \leq \frac{c_1 \min(n_1, n_2)}{\mu \log^2(\max(n_1, n_2))}$$

and

$$\text{sparsity}(\boldsymbol{S}) \leq c_2 n_1 n_2$$

with high probability, where $c_1$ and $c_2$ are fixed constants.

### 2.2. Reweighted $\ell_1$ Optimization

In our work we seek to adapt the parameters of the PCP algorithm to better hone in on malicious corruption. To this end, we can borrow techniques developed for similar optimization problems in sparse signal recovery. The goal of traditional sparse signal recovery is to restore a signal $\boldsymbol{y}$ from measurements $\boldsymbol{z}$ collected as

$$\boldsymbol{z} = \boldsymbol{\Phi}\boldsymbol{y} + \boldsymbol{\epsilon},$$

where $\boldsymbol{\Phi}$ is a measurement matrix (usually taken to be random), $\boldsymbol{\epsilon}$ is the measurement error, and $\boldsymbol{y}$ is assumed to be sparse. When this is the case, $\boldsymbol{y}$ can be recovered by solving the $\ell_1$-regularized least-squares problem

$$\widehat{\boldsymbol{y}} = \arg\min_{\boldsymbol{y}} \|\boldsymbol{z} - \boldsymbol{\Phi}\boldsymbol{y}\|_2^2 + \lambda \|\boldsymbol{y}\|_1, \tag{4}$$

where $\lambda$ is a parameter that controls the trade-off between the sparsity of the solution and the measurement fidelity [8]. While this parameter is usually assumed to be known *a priori*, a growing body of literature suggests that it should instead be adaptively selected [9, 10, 11, 12].

In particular, results on reweighted $\ell_1$ algorithms have shown that recovery from compressive measurements can be improved by defining an adaptable trade-off parameter for each element of the signal $\boldsymbol{y}$ [9]. Under this approach, the optimization problem in (4) is replaced with a series of similar problems where the $\lambda$ parameters are adapted at each iteration. The algorithm alternates between updating the signal estimate by solving

$$\widehat{\boldsymbol{y}} = \arg\min_{\boldsymbol{y}} \|\boldsymbol{z} - \boldsymbol{\Phi}\boldsymbol{y}\|_2^2 + \|\boldsymbol{\Lambda}\boldsymbol{y}\|_1,$$

where $\boldsymbol{\Lambda}$ is the diagonal matrix containing the $\lambda$ parameters for each signal element, and adapting the parameters as

$$\lambda_i = \frac{\beta}{|\widehat{y}_i| + \gamma},$$

where $\alpha$ and $\beta$ control the distribution of $\lambda$. As the algorithm progresses, it "hones in" on active coefficients by consistently weighting active coefficients less heavily and smaller coefficients more heavily.

## 3. ADAPTIVE PCP

### 3.1. General Approach

For our model, the contributions of toxic waste elements are reflected in the matrix $\boldsymbol{S}_0$, which consists of a few non-zero entries spread out over a sparse unknown subset of rows. In effect, we assume corruptions of the low-rank data $\boldsymbol{L}_0$ are present only on a small subset of the rows of the matrix $\boldsymbol{Y}$. Under this assumption, the corruptions are not necessarily sparse across the rows in which they reside, but occur on a *sparse subset* of the system's rows. Our goal is to decompose $\boldsymbol{Y}$ as in (1) to exploit this structure, using the sparse $\boldsymbol{S}_0$ to realize the error sources.

To achieve this goal, we present a novel algorithm that identifies systematic behavior across rows that deviates in an obvious manner from the norm. We can draw upon the ideas taken from reweighted $\ell_1$ algorithms to adapt the parameters for each row of $\boldsymbol{S}_0$ in our optimization procedure. This adaptation will ensure that the algorithm will be able to extract the contributions of the malicious agents.

In our approach, we modify the PCP formulation to adapt the parameters for each row independently. We use a similar reweighted approach, where we alternately seek a row-weighted PCP decomposition

$$\begin{aligned} \text{minimize} \quad & \|\boldsymbol{L}\|_* + |\boldsymbol{\Lambda}\boldsymbol{S}\|_1 \\ \text{subject to} \quad & \boldsymbol{L} + \boldsymbol{S} = \boldsymbol{Y}, \end{aligned}$$

(where the matrix $\boldsymbol{\Lambda}$ in this program is a diagonal weighting matrix indicating how toxic we believe each row of $\boldsymbol{S}_0$ to be), and an adaptive step

$$\boldsymbol{\Lambda}_{i,i} = \frac{\beta}{\|\boldsymbol{S}^{(i)}\|_1 + \gamma}, \tag{5}$$

[1] This assumption ensures the sparse matrix is unlikely to be low-rank.

where $\boldsymbol{S}^{(i)}$ represents the $i^{th}$ row of $\boldsymbol{S}$. This adaptive step looks at the energy in each column of $\boldsymbol{S}$ and re-assigns a parameter that increases or decreases the likelihood that this row is considered toxic.

### 3.2. Implementation

We implement our algorithm *via* a modified version of AD or the augmented Lagrange-multiplier method. The basic AD algorithm and its related extensions have been thoroughly studied over the years [13, 14]. Most notably for our purposes, AD was used in [6] to obtain the basic low-rank-plus-sparse decomposition of (1).

The algorithm we propose uses a variation of this approach to solve the row-weighted PCP optimization, followed by the analytic update in (5) to identify and separate the contributions of nefarious agents. The algorithm minimizes the Lagrangian

$$\begin{aligned}\mathcal{L}(\boldsymbol{L},\boldsymbol{S},\boldsymbol{Z}) = \|\boldsymbol{L}\|_* + \|\boldsymbol{\Lambda}\boldsymbol{S}\|_1 + \frac{\alpha}{2}\|\boldsymbol{Y}-\boldsymbol{L}-\boldsymbol{S}\|_F^2 \\ + <\boldsymbol{Z},\boldsymbol{Y}-\boldsymbol{L}-\boldsymbol{S}>\end{aligned} \tag{6}$$

by alternately minimizing the low-rank and sparse components. In (6), $\boldsymbol{Z}$ is the Lagrangian multiplier, while $\boldsymbol{\Lambda}$ is the re-weighting vector that adjusts the weights applied to rows of $\boldsymbol{S}$ based on the energy they contain. The alternating optimization sequentially minimizes $\mathcal{L}(\boldsymbol{L},\boldsymbol{S},\boldsymbol{Z})$ over $\boldsymbol{L}$ and $\boldsymbol{S}$.

The latter minimization is given by the well known soft thresholding operator

$$\Gamma_{\tau_i}(R_{ij}) = \text{sgn}(R_{ij})\max(|R_{ij}|-\tau_i,0) \quad \forall i,j,$$

where $\tau_i$ is the threshold for the $i^{th}$ row. Similarly, the minimization over $\boldsymbol{L}$ is obtained by soft thresholding the singular values of a given matrix, which is the proximal operator for the nuclear norm. Assuming the SVD of $\boldsymbol{R}$ is given as before, then

$$\mathcal{D}_\tau(\boldsymbol{R}) = \boldsymbol{U}T_\tau(\boldsymbol{\Sigma})\boldsymbol{V}^*$$

where

$$T_\tau(\boldsymbol{\Sigma}) = \max(\boldsymbol{\Sigma}-\tau I,0)$$

is an element-wise thresholding of $\boldsymbol{\Sigma}$.

In Algorithm 1, $\boldsymbol{S}$ converges to an approximation of the sparse error term while $\boldsymbol{L}$ converges to the required low rank matrix.

The convergence of the basic AD algorithm was demonstrated in [14]. Therefore, we know that for any reasonable choice of $\boldsymbol{\Lambda}$ in the outer loop of Algorithm 1, the inner loop will converge. As a result, our adaptive algorithm is expected to be stable, a notion that is reinforced by the knowledge that similar reweighted schemes converge in the vector case.

**Algorithm 1 Principal Component Pursuit by Adaptive Alternating Directions**

**Initialize:** $\boldsymbol{\Lambda}=\boldsymbol{\Lambda}_0, \boldsymbol{S}_0=\boldsymbol{Z}_0=0,$ and $\alpha,\beta,\gamma>0$
**while** not converged **do**
  **while** not converged **do**
    compute $\boldsymbol{L}_{k+1}=\mathcal{D}_{\alpha^{-1}}(\boldsymbol{Y}-\boldsymbol{S}_k+\alpha^{-1}\boldsymbol{Z}_k)$;
    compute $\boldsymbol{S}_{k+1}=\Gamma_{\boldsymbol{\Lambda}\alpha^{-1}}(\boldsymbol{Y}-\boldsymbol{L}_{k+1}+\alpha^{-1}\boldsymbol{Z}_k)$;
    compute $\boldsymbol{Z}_{k+1}=\boldsymbol{Z}_k+\alpha(\boldsymbol{Y}-\boldsymbol{L}_{k+1}-\boldsymbol{S}_{k+1})$;
  **end while**
  **Update:** $\boldsymbol{\Lambda}=\beta/(\|\boldsymbol{S}^{(i)}\|_1+\gamma)$.
**end while**
**Output:** $\boldsymbol{L},\boldsymbol{S}$.

## 4. SIMULATIONS

To test our algorithm, we generated a series of $100\times200$ rank-five matrices. We then added toxic waste by selecting ten rows at random and generating errors on half of the values in those rows. Next, we introduce additional sparse and Gaussian errors to the rest of the matrix, which account for the standard structure of recommendation systems and for minor systematic errors.

With the corrupted systems generated, we recovered the low-rank-plus-sparse decomposition of the matrix using both standard PCP optimization and by adapting $\lambda$ for each row of the sparse component. Each optimization problem was solved with AD outlined in Algorithm 1. The trade-off parameter $\lambda$ for standard PCP was set to $0.1$, while for the reweighted version we set the parameters $\beta=1.5$ and $\gamma=0.01$. Under this setup, we find that we need at most ten iterations for adequate convergence.

Our initial results demonstrate that by adaptively updating the sparse matrix component, our algorithm steers away from spurious sparse errors and narrows in on more systematic deviations corresponding to malicious contributions. Figure 1 depicts the aggregate matrix error (*i.e.*, toxic waste and spurious sparse errors together), the identification of toxic waste using our adaptive algorithm, and the identification using standard AD for an example matrix. The addition of adaptivity to PCP allows our algorithm to better differentiate nefarious contributions from spurious errors.

Since the ultimate goal is to detect sources who contribute misinformation, we then fed the results of the matrix-decomposition algorithms into a detector that compares the energy in each row of the sparse matrix component against a specified threshold. By sweeping the threshold over a range of 0 to 80, we computed estimates of the probabilities of missed detection and of false alarm, which are plotted in Figure 2 as a function of the decision threshold value. For small thresholds, the probability of false alarm is significantly smaller for the adaptive algorithm than for standard PCP. As the threshold increases, the false alarm probability drops to zero and the probability of detection decreases. While

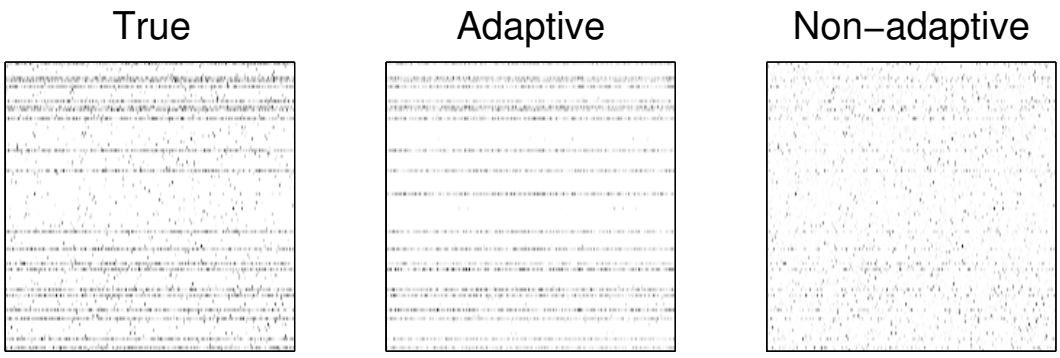


**Fig. 1**. (Left) The total deviation of a dataset from a low-rank model can contain a mixture of sparse noise and small Gaussian deviations in addition to malicious corruption. (Middle) Adaptive PCP can focus on the malicious corruption, ignoring other, natural, deviations. (Right) Standard PCP has no mechanism to differentiate the various errors.

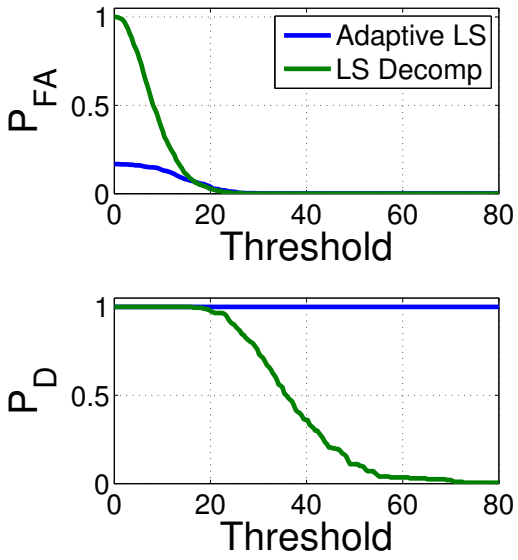


**Fig. 2**. The goal of the adaptive PCP scheme is to identify nefarious activity while ignoring natural deviations. (Top) The probability of false alarms (claiming someone innocent is acting maliciously) for a given decision threshold is lower for adaptive PCP over standard PCP. (Bottom) The probability of detection falls quickly for standard PCP while adaptive PCP retains very high probability of detection.

the probability of detection decreases for both methods, this change is orders of magnitude smaller for the adaptive algorithm. Note that adaptive PCP reaches a minimal probability of false alarm while still retaining a very high probability of detection, which is not the case for standard PCP.

## 5. CONCLUSIONS AND FUTURE WORK

This work outlines an adaptive algorithm that seeks to remove malicious erroneous information from datasets. The corruption is not considered to be a sparse matrix with non-zero entries supported at random locations, but rather is assumed to occur in a structured pattern. Our results indicate that we can "clean" the low-rank component from this structured pattern of errors using an adaptive version of PCP. Moreover, empirical results suggest that our optimization converges in relatively few iterations. We implemented our algorithm using efficient solvers and our result indicate the effectiveness of our approach, especially in comparison to the standard formulation of PCP.

In practical scenarios, we might not have complete information about the entries of matrix $\boldsymbol{Y}$; *i.e.*, only a partial set of the entries of $\boldsymbol{Y}$ may be known. Given a generic set of the entries of $\boldsymbol{Y}$, we can fill in the missing entries using nuclear norm minimization under some incoherence assumptions on $\boldsymbol{Y}$. In addition, we can also decompose matrix $\boldsymbol{Y}$ into low-rank and sparse members with this partial information. The matrix decomposition into structured sparse and low-rank components can also be obtained from the partial information using our adaptive PCP approach.

## 6. REFERENCES

[1] J. Bennett and S. Lanning, "The Netflix prize," in *In KDD Cup and Workshop*, 2007.

[2] A. Harmon, "Amazon glitch unmasks war of reviewers," Available at `www.nytimes.com`, 2004.

[3] D. Streitfeld, "For $2 a star, an online retailer gets 5-star product reviews," Available at `www.nytimes.com`, 2012.

[4] R. Fisman, "Should you trust online reviews?," Available at www.slate.com, 2012.

[5] V. Chandrasekaran, S. Sanghavi, P.A. Parrilo, and A.S. Willsky, "Rank-sparsity incoherence for matrix decomposition," *SIAM J Optimiz.*, vol. 21, no. 2, pp. 572–596, 2011.

[6] Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright, "Robust principal component analysis?," *J. ACM*, vol. 58, no. 3, pp. 11:111:37, June 2011.

[7] E.J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, no. 6, pp. 717–772, 2009.

[8] E.J. Candès, "Compressive sampling," in *Proc. Int. Congr. Mathematics*, 2006, pp. 1433–1452.

[9] E. Candès, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted $\ell_1$ minimization," *J. Fourier Anal. Appl.*, vol. 14, no. 5, pp. 877–905, Dec 2008, Special Issue on Sparsity.

[10] D. Wipf and S. Nagarajan, "Iterative reweighted $\ell_1$ and $\ell_2$ methods for finding sparse solutions," *IEEE J. Sel. Top. Signa.*, vol. 4, no. 2, pp. 317–329, 2010.

[11] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," in *Proc. of ICASSP*. IEEE, 2008, pp. 3869–3872.

[12] D. Needell, "Noisy signal recovery via iterative reweighted l1-minimization," in *Forty-Third Asilomar Conference on Signals, Systems and Computers*. IEEE, 2009, pp. 113–117.

[13] Zhouchen Lin, Minming Chen, and Yi Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *arXiv:1009.5055*, Sept. 2010.

[14] Spyridon Kontogiorgis and Robert R. Meyer, "A variable-penalty alternating directions method for convex optimization," *Mathematical Programming*, vol. 83, no. 1-3, pp. 29–53, Jan. 1998.