Abstract—We propose novel tensor decomposition methods that advocate both properties of sparsity and robustness to outliers. The sparsity enables us to extract some essential features from a big data that are easily interpretable. The robustness ensures the resistance to outliers that appear commonly in high-dimensional data. We first propose a method that generalizes the ridge regression in $M$-estimation framework for tensor decompositions. The other approach we propose combines the least absolute deviation (LAD) regression and the least absolute shrinkage operator (LASSO) for the CANDECOMP/PARAFAC (CP) tensor decompositions. We also formulate various robust tensor decomposition methods using different loss functions. The simulation study shows that our robust-sparse methods outperform other general tensor decomposition methods in the presence of outliers.

I. INTRODUCTION

Tensors, being multi-way arrays in a simple definition, accommodate high-dimensional data sets naturally. Various tensor decompositions based on nonlinear models are powerful tools to explore a high-dimensional data in many areas of science and engineering [8].

The well-known regularization methods such as ridge regression or the LASSO [4] [9] can form a sparse estimation for tensor decompositions, which is highly useful in reducing dimensionality, simplifying visualization, and variable selection [15], [6]. It also improves interpretability of models often reducing the estimation uncertainty.

Outliers often occurring in multi-dimensional data obscure the model assumptions and add difficulty in data analysis. Most of the tensor decomposition methods are based on the optimization with a least squares criterion which is severely lacking robustness to outliers. Surprisingly, the use of robust estimators has been largely neglected in the tensor (signal processing) research community. We are aware of only few studies on the robust tensor estimation such as [13], [14], and [3], other than some work in the medical imaging.

Instead of using the least squares estimation which is highly sensitive to outliers, robust alternatives can be obtained with different loss functions such as absolute deviation ($L_1$), least trimmed squares (LTS), or $M$-estimation (e.g., using Huber’s or Tukey’s biweight) loss function [11].

In Section II-A, we first formulate various robust tensor decomposition methods using different loss functions. In Section II-B and II-C, we propose two robust and sparse methods in the context of CP tensor decompositions extending our previous work (encouraging only sparsity) [6].

The first method we propose called the CP alternating ridge regression (CP-RMR) is to minimize the penalized robust objective function formed by an $M$-estimation loss function (Huber’s $p$-functions) with the $L_2$-penalty. This CPA-RMR can be used not only as a stand-alone method but also as a reliable method to facilitate good initializations for any alternating tensor decomposition algorithms.

The second method we propose called the CP alternating LAD-LASSO combines the LAD regression and the LASSO in the context of CP tensor decompositions. To our knowledge, our method is the first that incorporates robustness and sparsity simultaneously in tensor decompositions. Such work can be found recently only in other areas of regression analysis [1].

The simulation study shows that our methods depict good estimation of sparsity of factor matrices and give highly improved performance over the conventional tensor decomposition method especially in the presence of outliers.

Notations: A tensor of order $d$ as a $d$-way array is denoted by a boldface Euler script letter, $\mathbf{A}$ whereas low-order tensors such as matrices by $\mathbf{A}$, vectors by $a$ and scalars by $a$. For a $m \times n$ matrix $\mathbf{A}$, we write its $i$th row (resp. column) vector by $a_i$ (resp. $\mathbf{a}_i$), i.e., a matrix can be expressed as $\mathbf{A} = (\mathbf{a}_1 \ldots \mathbf{a}_m) = (a_1 \ldots a_m)^\top$ and $a_{ij}$ denotes its $(i,j)$th element. Let $\| \cdot \|_2$ (resp. $\| \cdot \|_1$) denote the $\ell_2$-norm (resp. $\ell_1$-norm) defined as $\| \mathbf{A} \|_2^2 = \text{Tr}(\mathbf{A} \mathbf{A}^\top) = \sum_{i,j} a_{ij}^2$ (resp. $\| \mathbf{A} \|_1 = \sum_{i,j} |a_{ij}|$) for any matrix $\mathbf{A}$. Let $\circ$ denote the outer product, i.e. $\mathbf{a} \circ \mathbf{b} = \mathbf{ab}^\top$ and $(\mathbf{a} \circ \mathbf{b} \circ \mathbf{c})_{ijk} = a_ib_jc_k$.

A. Sparse CP decompositions

For simplicity, we illustrate all the methods for 3-way tensors, but extensions to multiway tensors are straightforward.

The CANDECOMP/PARAFAC (CP) decomposition [2], [7] approximates a tensor $\mathbf{X} \in \mathbb{R}_{I \times J \times K}$ by a predicted tensor $\hat{\mathbf{X}}$ consisting of a sum of $R \in \mathbb{N}^+$ rank-1 tensors [8]:

$$\mathbf{X} = \hat{\mathbf{X}} + \mathbf{E} = \sum_{r=1}^{R} \mathbf{\bar{a}}_r \circ \mathbf{\bar{b}}_r \circ \mathbf{\bar{c}}_r + \mathbf{E} \quad (1)$$

where $\mathbf{\bar{a}}_r \in \mathbb{R}^I$, $\mathbf{\bar{b}}_r \in \mathbb{R}^J$ and $\mathbf{\bar{c}}_r \in \mathbb{R}^K$ for $r = 1, \ldots, R$ form the unit-norm column vectors of the factor matrices $\mathbf{A} \in \mathbb{R}_{I \times R}$, $\mathbf{B} \in \mathbb{R}_{J \times R}$ and $\mathbf{C} \in \mathbb{R}_{K \times R}$ and the tensor $\mathbf{E} \in \mathbb{R}_{I \times J \times K}$ contains the error terms.
Unfolding the tensor $\mathbf{X}$ along the first mode yields a $I \times JK$-matrix denoted as $\mathbf{X}_{(1)}$:

$$\mathbf{X}_{(1)} = \mathbf{A} \Gamma (\mathbf{C} \otimes \mathbf{B})^T + \mathbf{E}_{(1)},$$

where $\Gamma = \text{diag} (\gamma_1, \ldots, \gamma_R)$ and $\mathbf{E}_{(1)}$ denotes the unfolded matrix of $\mathbf{E}$. Note that the Khatri-Rao product for $\mathbf{B} \in \mathbb{R}^{R \times R}$ and $\mathbf{C} \in \mathbb{R}^{K \times K}$, $\mathbf{C} \otimes \mathbf{B}$ is a $\mathbf{J} \times \mathbf{R}$ matrix $\mathbf{C} \otimes \mathbf{B} = (\mathbf{c}_1 \otimes \mathbf{b}_1 \cdots \mathbf{c}_R \otimes \mathbf{b}_R)$, where $\otimes$ denotes the Kronecker product. The mode-2 and mode-3 unfoldings of the tensor $\mathbf{X}$ are obtained similarly. For the notational convenience, we write $\mathbf{Z} = \mathbf{C} \otimes \mathbf{B}$, $\mathbf{X} = \mathbf{X}_{(1)}$, $n = JK$ and $m = I$ so that $\mathbf{X}$ is an $m \times n$ matrix and $\mathbf{A}$ is a $m \times R$ matrix. Let $\gamma_r = ||\mathbf{a}_r||$ taking the scales of the columns of $\mathbf{A}$.

The conventional CP decomposition seeks to minimize

$$Q(\mathbf{A}, \mathbf{B}, \mathbf{C}) = ||\mathbf{X} - \mathbf{A} \mathbf{Z}^\top||_2^2 = \sum_{i=1}^{m} \sum_{j=1}^{n} (x_{ij} - z_{ij}^\top \mathbf{a}_j)^2. \tag{3}$$

To enforce sparsity, we add the $L_2$-norm penalty as in [6] to find the minimum of the penalized objective function:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} (x_{ij} - z_{ij}^\top \mathbf{a}_j)^2 + \lambda_1 ||\mathbf{A}||_q^q + \lambda_2 ||\mathbf{B}||_q^q + \lambda_3 ||\mathbf{C}||_q^q, \tag{4}$$

where $\lambda = (\lambda_1, \lambda_2, \lambda_3)^\top$ and $\lambda_i \geq 0$, $i \in \{1, 2, 3\}$ denote the fixed penalty (shrinkage, regularization) parameters and $q \in \{1, 2\}$ is the chosen norm of the penalty. A computationally convenient form of (3) for the alternating algorithm (estimating one factor matrix at a time) is

$$\hat{\mathbf{A}}_i = \min_{\mathbf{a}_i} \left\{ \sum_{j=1}^{n} (x_{ij} - z_{ij}^\top \mathbf{a}_j)^2 + \lambda_1 ||\mathbf{a}_j||_q^q \right\}. \tag{5}$$

Here we first estimate the factor matrix $\mathbf{A}$ keeping other factor matrices ($\mathbf{C}$ and $\mathbf{B}$) fixed. Then, finding the minimum $\hat{\mathbf{A}} = (\hat{\mathbf{a}}_1 \cdots \hat{\mathbf{a}}_m)^\top$ is equivalent to solving the $m$ separate penalized $L_2$-objective functions, for each $i = 1, \ldots, m$:

$$\hat{a}_i = \min_{a_i} \left\{ \sum_{j=1}^{n} (x_{ij} - z_{ij}^\top a_j)^2 + \lambda_1 ||a_j||_q^q \right\}. \tag{6}$$

With the $L_2$-norm ($q = 2$) for the penalty, $\hat{\mathbf{A}}$ has a closed form solution $\hat{\mathbf{A}} = \mathbf{XZ}(\mathbf{Z}^\top \mathbf{Z} + \lambda_1 \mathbf{I})^{-1}$. This method is called the CP alternating Ridge Regression (CP-ARR) in [6]. If $\lambda_1 = 0 \forall i$, then the approach reduces to the well-used CP alternating least squares (CP-ALS) method. With the $L_1$-norm ($q = 1$) penalty, the solution to (5) is the LASSO estimate, which leads to our method called the CP alternating LASSO in [6].

II. ROBUST AND SPARSE CP-ALS METHODS

A. Robust CP decompositions

Disregarding outliers in data sets or heavy-tailed errors can mislead the analysis and inference. One of the most popular robust approaches may be to replace the $L_2$-loss function by the $L_1$-loss as in [13], [3] yielding an objective function

$$Q_{L_1}(\mathbf{A}, \mathbf{B}, \mathbf{C}) = ||\mathbf{X} - \mathbf{A} \mathbf{Z}^\top||_1 = \sum_{i=1}^{m} \sum_{j=1}^{n} |x_{ij} - z_{ij}^\top \mathbf{a}_j|. \tag{7}$$

However, the $L_1$-loss function is not bounded and large outliers can still have a considerable influence on the estimation. To achieve better robustness, we can consider the $M$-estimation [5] type objective function

$$Q_\rho(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \sum_{i=1}^{m} \sum_{j=1}^{n} \rho \left( \frac{x_{ij} - z_{ij}^\top \mathbf{a}_j}{\tilde{\sigma}_i} \right), \tag{8}$$

where $\tilde{\sigma}_i$ is a preliminary robust scale estimate (to be discussed later) and $\rho$ is an even ($\rho(e) = \rho(-e)$) nondecreasing function for $e \geq 0$. The multiplying factor $\tilde{\sigma}_i^2$ is used so that $Q_\rho(\mathbf{A}, \mathbf{B}, \mathbf{C})$ becomes (3) when $\rho(e) = e^2$. A popular robust choice for $\rho(\cdot)$ is Huber’s $\rho$-function defined as

$$\rho_k(e) = \begin{cases} \frac{1}{2} e^2, & \text{for } |e| \leq k \\ k |e| - \frac{1}{2} k^2, & \text{for } |e| > k \end{cases} \tag{9}$$

where $k$ is a user-defined tuning constant that affects robustness and efficiency of the method. Note that $\rho_k(e) \to \rho(e) = e^2$ as $k \to \infty$. Thus, the solution to minimizing (6) converges to the CP-ALS solution as $k \to \infty$. With $k = 0.7317$, 85% asymptotic efficiency is attained for Gaussian errors in the multiple regression model. Since Huber’s $\rho$-function is not bounded, Tukey’s biweight $\rho$-function is often preferred. However, this $\rho$-function is non-convex and causes difficulty in finding a global minimum in the regression setting. A high-breakdown estimator can be derived by utilizing the LTS loss function. For the vector of squared residuals $r_i^2(a) = r_{i1}(a), \ldots, r_{in}(a)$, where $r_{ij}(a) = x_{ij} - z_{ij}^\top a$, for $i = 1, \ldots, m$, $j = 1, \ldots, n$, the LTS objective function is

$$Q_{LTS}(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \sum_{i=1}^{m} \sum_{j=1}^{n} (r_{ij}^2(a_i))_{j,n} \tag{10}$$

where $(r_{ij}^2(a))_{1:n} \leq \cdots \leq (r_{ij}^2(a))_{m,n}$ are the order statistics of the squared residuals and $[n/2] \leq h \leq n$.

B. The CP alternating Ridge M-Regression (CPA-RMR)

We propose a robust and sparse CP tensor decompositions using the alternating ridge M-regression abbreviated as CPA-RMR. The ridge regression inducing sparsity is adjusted to the robust version using the $M$-regression (hence, called RMR). The objective function (6) with the $L_2$-norm penalty is

$$\sum_{i=1}^{m} \sum_{j=1}^{n} \rho \left( \frac{x_{ij} - z_{ij}^\top a_j}{\tilde{\sigma}_i} \right) + \lambda_1 ||\mathbf{a}_j||_2^2 + \lambda_2 ||\mathbf{B}||_2^2 + \lambda_3 ||\mathbf{C}||_2^2. \tag{11}$$

For $\mathbf{B}$ and $\mathbf{C}$ fixed, the minimum $\hat{\mathbf{A}} = (\hat{\mathbf{a}}_1 \cdots \hat{\mathbf{a}}_m)^\top$ can be found by

$$\hat{a}_i = \min_{a_i} \left\{ \tilde{\sigma}_i^2 \sum_{j=1}^{n} \rho \left( \frac{x_{ij} - z_{ij}^\top a_j}{\tilde{\sigma}_i} \right) + \lambda_1 ||a_j||_2^2 \right\} \tag{12}$$

for $i = 1, \ldots, m$. We center $x_i$ and columns of $\mathbf{Z}$ to have median zero to keep the results invariant. Furthermore, the columns of $\mathbf{Z}$ are scaled to have median absolute deviation (MAD) equal to one. After obtaining $a_i$’s for the standardized variables using the efficient “iteratively (re)weighted ridge
regression (IWRR) algorithm explained below, we can transform the solutions back to the original scale. The initial value of \( a_i \) (in the IWRR) is \( \hat{a}_i^{(0)} \) calculated as the LAD regression estimate when regressing \( x_i \) on \( Z \):

\[
\hat{a}_i^{(0)} = \text{LAD}(x_i, Z) = \arg \min_a \sum_{j=1}^n |x_{ij} - z_j a_i|.
\]

The scale estimate \( \hat{\sigma} \) is computed as MAD of residuals:

\[
\hat{\sigma}_i = 1.4826 \cdot \text{median} \{|r_{ij}(\hat{a}_i^{(0)})|\}_{j=1}^n
\]

IWRR algorithm. Let \( \psi(e) = \rho'(e) \) and \( w(e) = \psi(e)/e \) with the convention that \( w(e) = 0 \) for \( e = 0 \). For example, if Huber’s \( \rho \)-function is used in (7), then the respective \( \psi \)-function is \( \psi_k(e) = \max|k|, \min(k, e)| \). By setting the derivatives of the objective function in (9) to zero, shows that \( \hat{a}_i \) solves the following estimating equation:

\[
(Z^T W_i Z + 2\lambda I) \hat{a}_i = Z^T W_i x_i
\]

(10)

where \( W_i = \text{diag}(\{w_{ij}\}_j) \) with \( w_{ij} = w(r_{ij}(\hat{a}_i)/\hat{\sigma}_i) \). This leads us to iterative computation of solving the weighted rige normal equations:

\[
(Z^T W_i Z + 2\lambda I) \hat{a}_i^{(t+1)} = Z^T W_i x_i
\]

(11)

where \( W_i = \text{diag}(\{w_{ij}\}_j) \) and \( w_{ij} = w(r_{ij}(\hat{a}_i^{(t)}/\hat{\sigma}_i) \) and \( t \) indicates the iteration step (until convergence).

Following [11], it can be shown that the objective function (9) descends at each iteration. Thus, for convex problems (e.g., using Huber’s \( \rho \)), the IWRR algorithm can be used to find the global minimum. It is important to note that (11) can be solved efficiently as follows. Let

\[
x_{i,t} = \begin{bmatrix} W_i^{1/2} x_i \\ 0 \end{bmatrix} \quad \text{and} \quad Z_{i,\lambda_1} = \begin{bmatrix} W_i^{1/2} Z \\ \sqrt{2\lambda_1} I \end{bmatrix},
\]

\[
\left( Z_{i,\lambda_1}^T Z_{i,\lambda_1} \right)^{-1} Z_{i,\lambda_1} x_{i,t}
\]

(12)

which is in line with the conventional normal equations. Consequently, \( \hat{a}_i^{(t+1)} \) can be solved efficiently without inverting (nor computing) \( Z_{i,\lambda_1}^T Z_{i,\lambda_1} \).

The procedure for the CPA-RMR method is the same as the one for the CP alternating LAD-LASSO given in Table I except that \( \hat{a}_i, \hat{b}_i, \hat{c}_i \) are computed by the IWRR algorithm.

C. The CP alternating LAD-LASSO method

We propose here another method called the CP alternating LAD-LASSO, which is robust with the \( L_1 \)-penalty inducing sparse (exactly zero) estimates. We note that this method is computationally more demanding than the CPA-RMR due to the \( L_1 \) objective function. The aim is to minimize the objective function \( Q_{L_1}(\cdot) \) with the \( L_1 \)-norm penalty

\[
\sum_{i=1}^m \left\{ \sum_{j=1}^n |x_{ij} - z_j a_i| + \lambda_1 ||a_i||_1 \right\} + \lambda_2 ||B||_1 + \lambda_3 ||C||_1.
\]

The following theorem holds (Lemma 2 in [11]).

\begin{table}[ht]
\centering
\caption{CP Alternating LAD-LASSO method.}
\begin{tabular}{ll}
\hline
Input: & X, rank \( R \), shrinkage parameters \( \lambda_1, \lambda_2 \) and \( \lambda_3 \). \\
\hline
0) Initialize B and C by B and C using CPA-RMR solution. & \\
1) Set X = X(1), Z = C \odot B, n = JK and m = I. Compute & \\
A^T = (a_1 \cdots a_m) by solving \( \hat{a}_i = \text{LADLASSO}(x_i, Z, \lambda_1) \) & \\
(12)
& \quad \text{(using the LAD-LASSO algorithm.)} \\
2) Set X = X(2) and Z = C \odot A, n = IK and & \\
m = J. Compute \( \hat{B} = (b_1 \cdots b_m) \) by solving \( \hat{b}_i = & \\
\text{LADLASSO}(x_i, Z, \lambda_2). \) & \\
3) Set X = X(3), Z = B \odot A, n = IJ and m = K. Compute & \\
\hat{C} = (c_1 \cdots c_m) \text{ by solving } \hat{c}_i = \text{LADLASSO}(x_i, Z, \lambda_3). & \\
4) Repeat steps 1)–3 until \( \delta = ||X - X_{old}||_2/||X||_2 \) & \\
the relative change between the current and previous CP fits is small. & \\
\hline
\end{tabular}
\end{table}

D. Selection of the shrinkage parameter

We estimate the shrinkage parameter \( \lambda \) using the Bayesian information criteria (BIC)

\[
\text{BIC}(\lambda) = 2N \ln \hat{\sigma} + w \cdot \text{df}(\lambda) \cdot \ln N
\]

(14)

where \( N = I \cdot J \cdot K \), \( \hat{\sigma} \) is a scale estimate of the residuals \( r_{ij} = \{X(1) - \hat{A}Z^T\}_{i,j} \) with \( Z = C \odot B \) \( (i = 1, \ldots, I, \)

\[
\text{df}(\lambda) = \text{degrees of freedom of the model and } w \geq 1 \text{ is a weight that can be assigned by the user. The default is } w = 1 \text{ and } w = \sqrt{2} \text{ was used in our simulations. Note that } w > 1 \text{ favors sparsity.}
\]

In the case of LASSO, the scale is estimated by \( \hat{\sigma}^2 = \text{median}_{r_{ij}} \{|r_{ij}|\} \) and \( \text{df}(\lambda) = \text{mean}_{r_{ij}} \{|r_{ij}|\} \), i.e. the mean absolute deviation estimate, and for RMR \( \hat{\sigma}^2 = \frac{1}{4} \text{median}_{r_{ij}} \{|r_{ij}|\} \), the median absolute deviation estimate. Regarding the CP alternating LASSO and LAD-LASSO, \( \text{df}(\lambda) \) is defined as the sum of the number of non-zero elements in factor matrices (A, B and C). In calculating the number of non-zeros in a factor matrix (e.g. A after
normalizing the columns to be of unit norm), the element \( \hat{a}_{ij} \) is considered to be zero if \( \hat{a}_{ij} \leq 10^{-11} \). In respect to the ridge (\( M \)-regression), def \( \lambda = \| Z \|_2^2 + \lambda_1^2 \| Z \|_2^2 + \lambda_2^2 \| Z \|_2^2 + \lambda_3^2 \| Z \|_2^2 \), where \( Z = C \odot A \) and \( Z = B \odot A \).

## III. SIMULATIONS

We consider the CP model whose true noise-free three-way tensor \( X_0 \) is sparse. The observed three-way tensor is generated as \( \tilde{X} = X_0 + \epsilon \), where \( X_0 = \sum_{r=0}^R \lambda_r \circ b_r \circ c_r \), \( \epsilon \) is the noise tensor and the rank \( R \) is assumed to be known. The accuracy of the obtained estimate \( \hat{X} \) is calculated by the normalized mean squared error \( \text{NMSE}(\hat{X}) = \| X_0 - \hat{X} \|_F^2 / \| X_0 \|_F^2 \).

In our simulation we set \( I = 1000, J = 20, K = 20, \) and \( R = 3 \) with one sparse factor matrix \( A \in \mathbb{R}^{1000 \times 3} \) whose half of the elements (1500) are randomly chosen to be zero. The other half of the elements of \( A \) are independent random deviates from \( N(0,1) \). The entries of factor matrices \( B \in \mathbb{R}^{20 \times 3} \) and \( C \in \mathbb{R}^{20 \times 3} \) are independently drawn from the \( N(0,1) \) distribution. The columns of \( A, B \) and \( C \) are then normalized to have unit length and the values of the loadings are \( \gamma_1 = 1000, \gamma_2 = 500 \) and \( \gamma_3 = 500 \). We generated \( M = 50 \) tensors according to the setup above and first added the noise tensor \( \epsilon \in \mathbb{R}^{1000 \times 20 \times 20} \) from \( N(0,1) \). Then, the heavy-tailed noise tensor \( \epsilon \in \mathbb{R}^{1000 \times 20 \times 20} \) from the Cauchy distribution with symmetric center 0 and scale parameter \( 1/2 \) is added in place of the normal noise tensor for comparison.

The penalty parameter is selected by minimizing the BIC in (14) with the weight \( w = \sqrt{2} \) over a grid of \( \lambda = \lambda_1 = \lambda_2 = \lambda_3 \) values. Note that an equal penalty parameter \( \lambda \) for each factor matrix was used simply for the computational feasibility. Naturally, using different penalty parameters for each factor matrix would be more appropriate yielding improved estimates, since only one factor matrix is sparse in this simulation. However, this would require the BIC evaluation over the 3-dimensional grids of all possible \( (\lambda_1, \lambda_2, \lambda_3) \) combinations and adds more computational burden. For our proposed CPA-RMR method, we considered three values of \( \lambda \in \{1, 10, 100\} \) representing mild, medium and hard-shrinkage. A small selection of \( \lambda \) candidates are used for CPA-RMR, since we used this algorithm only to obtain good initializations for the other (LAD)-LASSO methods in the simulation study. In the case of LASSO (resp. LAD-LASSO) we used \( \lambda \in \{0.001\} \cup \{0.01, \ldots, 0.09\} \) (resp. \( \lambda \in \{0.01\} \cup \{1.0, 1.5, \ldots, 5.0\} \)).

The results of the study are summarized in Table II. The RER (recovery rate) is another measure for performance which estimates the rate of correctly classified zero/non-zero elements for a sparse factor matrix \( A \). It is defined as the sum of entries correctly estimated (classified) zeros/non-zeros divided by the number of all entries in \( A \). Thus, \( \text{RER} = 0 \) implies the case of perfect classification.

From Table II, both the CP-ALS and the CP alternating LASSO methods poorly estimate the factor matrices for the tensors with the heavy-tailed Cauchy noise, whereas our robust sparse methods, LAD-LASSO and RMR, show excellent performance. For the tensors with the normal noise, all the sparse methods (CPA-RMR, LASSO, LAD-LASSO) outperform the conventional CP-ALS method. In terms of NMSE values (for the normal noise) the CPA-RMR reveals 2.19 times better accuracy than CP-ALS, the CP alternating LASSO 12.02 times better, and the CP alternating LASSO 4.47 times better. The best (NMSE) performance of the LASSO in the case of Normal noise is expected, since the sum of the squared residuals (in the objective function) is optimal for the Normal distribution. It can be noted that the weight \( w = \sqrt{2} = 1.414 \) on the penalty term produces the higher value of \( \lambda \) (assigning more zeros) for the LAD-LASSO compared to the LASSO.

### REFERENCES