Robust Factorization Machine: A Doubly Capped Norms Minimization

Chenghao Liu* Teng Zhang †‡ Jundong Li§ Jianwen Yin†‡ Peilin Zhao¶
Jianling Sun†‡‖ Steven C.H. Hoi*

Abstract

Factorization Machine (FM) is a general supervised learning framework for many AI applications due to its powerful capability of feature engineering. Despite being extensively studied, existing FM methods have several limitations in common. First of all, most existing FM methods often adopt the squared loss in the modeling process, which can be very sensitive when the data for learning contains noises and outliers. Second, some recent FM variants often explore the low-rank structure of the feature interactions matrix by relaxing the low-rank minimization problem as a trace norm minimization, which cannot always achieve a tight approximation to the original one. To address the aforementioned issues, this paper proposes a new scheme of Robust Factorization Machine (RFM) by exploring a doubly capped norms minimization approach, which employs both a capped squared trace norm in achieving a tighter approximation of the rank minimization and a capped ℓ_1-norm loss to enhance the robustness of the empirical loss minimization from noisy data. We develop an efficient algorithm with a rigorous convergence proof of RFM. Experiments on public real-world datasets show that our method outperforms the state-of-the-art FM methods significantly.

1 Introduction

Factorization Machine (FM) [15, 16] represents a family of general-purpose supervised learning techniques in machine learning and data mining, which provides an efficient mechanism for modeling feature interactions in a latent space. Unlike linear models (e.g., Support Vector Machines) that only learn a feature weight vector $\mathbf{w} \in \mathbb{R}^d$, where d is the number of features, FM also learns a pairwise feature interaction matrix $\mathbf{Z} \in \mathbb{R}^{d\times d}$ to model the pairwise interactions between features. Specifically, it models the feature interaction matrix via a factorized formula, $\mathbf{Z} = \mathbf{V}\mathbf{V}^\top$, where $\mathbf{V} \in \mathbb{R}^{d\times k}$ and $k \ll d$ is a hyperparameter indicating the rank of the factorized matrix. FM has been widely explored in various machine learning and data mining applications including classification [15], recommendation [11] and search ranking [10]. Despite its widespread usage, the performance of FM is severely affected by the choice of the low-rank structure of the underlying feature interaction matrix.

To explore a suitable low-rank structure of the feature interaction matrix, many recent studies have improved FM [2, 21, 9] by imposing the trace norm (a.k.a. nuclear norm, the sum of the singular values of the data matrix) regularization $\|\mathbf{Z}\|_*$ [19], which is known to be the tightest convex lower bound on the matrix rank. Specifically, following the recent advances on low-rank optimization with trace norm penalty [17], Blondel et al. [2] presented an efficient greedy coordinate descent algorithm with global convergence; Yamada et al. [21] formulated the objective function as a semidefinite programming and derived an efficient optimization procedure with Hazan's algorithm [4]. Although the trace norm is capable of introducing low-rank structure and learning potential correlations from data samples, the approximation error between the rank minimization and the trace norm constraint cannot be neglected in practical applications [18] . For example, as the non-zero singular values change, the trace norm value will change together but the rank value will remain the same. Thus, it is highly desired to find a more effective way in exploiting the low-rank structure to better approximate the rank minimization problem.

Meanwhile, most existing FM methods often assume the labeled data in training set are clean and precise, and thus adopt the standard loss function to model the relations among the data and the label. In fact, the standard loss function remains agnostic to the unavoidable noise present in the input signals. Consequently, the learnt models have an ambiguous understanding of the

*School of Information Systems, Singapore Management University. {chliu,chhoi}@smu.edu.sg

†School of Computer Science and Technology, Zhejiang University. {faramita,jianwen_yin,sunjl}@zju.edu.cn

‡Alibaba-Zhejiang University Joint Institute of Frontier Technologies

§Computer Science and Engineering, Arizona State University. jundongl@asu.edu

¶School of Software Engineering, South China University of Technology. peilinzhao@hotmail.com

‖Indicates Corresponding Author.

correlation between features and labels and thus exhibit remarkable sensitivity towards small data perturbations. Since the outliers are significantly different from the normal data and could mislead the training process, the learned model will not be optimal and the prediction performance could be jeopardized.

To address the aforementioned issues, we develop a novel framework for Robust Factorization Machine (RFM) based on a Doubly Capped Norms Minimization approach. The key idea is to reformulate the objective function of FM by employing two types of capped norms. First of all, we propose to leverage a capped squared trace norm to replace the trace norm. In this way, it only minimizes the singular values that are smaller than a given threshold and treating other large singular values as a fixed value, which is able to achieve a tighter and more robust approximation of the rank minimization. Second, instead of using the standard square loss or hinge loss, we propose a new capped ℓ_1-norm loss to enhance the robustness of the empirical loss minimization from noisy training data. In addition, in order to tackle the new formulation of the objective function that is non-convex and non-smooth, we develop an efficient optimization algorithm with rigorous convergence analysis. We also empirically validate the efficacy of the proposed method by conducting extensive experiments on real-world datasets, and the results show that our method outperforms the state-of-the-art FM methods in both the binary classification and recommendation tasks.

Our major contributions are summarized as follows:

- We propose a novel formulation of Robust Factorization Machines (RFM) via a Doubly Capped Norms Minimization to tackle the limitations of the vanilla FM;
- We propose an efficient optimization procedure to tackle the non-trivial optimization task of Robust Factorization Machines with proved convergence;
- We evaluate the performance of the proposed RFM on both the binary classification and the recommendation tasks, and validate the efficacy and robustness of the proposed method.

We believe our work sheds light on the Factorization Machine research, especially for robust Factorization Machine. Due to the wide applications of FM, our work is of both theoretical and practical significance.

2 Robust Factorization Machine

Factorization Machine (FM) proposed in [15] have recently gained popularity as an effective learning paradigm for utilizing feature interactions in supervised learning tasks such as classification or regression. Given an input feature vector $\mathbf{x} \in \mathbb{R}^d$, the vanilla FM model predicts the output with the following equation:

$$\hat{y}(\mathbf{x}|\mathbf{w}, \mathbf{V}) = \mathbf{w}^\top \mathbf{x} + \sum_{j=1}^{d} \sum_{j'=j+1}^{d} (\mathbf{V}\mathbf{V}^\top)_{jj'} x_j x_{j'},$$

where $\mathbf{V} \in \mathbb{R}^{d \times k}$ and $\mathbf{w} \in \mathbb{R}^d$ are the model parameters to be estimated. Following the popular practice in designing convex variants of FM [2, 21], we consider a generalized formula by employing a low-rank symmetric matrix $\mathbf{Z} \in \mathbb{R}^{d \times d}$ to model the pairwise feature interactions:

$$\hat{y}(\mathbf{x}|\mathbf{w}, \mathbf{Z}) = \mathbf{w}^\top \mathbf{x} + \sum_{j=1}^{d} \sum_{j'=1}^{d} z_{jj'} x_j x_{j'} = \mathbf{w}^\top \mathbf{x} + \langle \mathbf{Z}, \mathbf{x}\mathbf{x}^\top \rangle,$$

where $z_{jj'}$ is an element of $\mathbf{Z}$. Given a training set $[\mathbf{x}_1, \ldots, \mathbf{x}_n]^\top \in \mathbb{R}^{n \times d}$ and the corresponding targets $[y_1, \ldots, y_n]^\top \in \mathbb{R}^n$, model parameters can be learned by using the principle of empirical risk minimization to solve the following non-convex problem:

$$\min_{\mathbf{w} \in \mathbb{R}^d, \mathbf{Z} \in \mathbb{S}_+^{d \times d}} \sum_{i=1}^{n} \ell(y_i, \hat{y}(\mathbf{x}_i|\mathbf{w}, \mathbf{Z})) + \frac{\alpha}{2} R_1(\mathbf{w}) + \frac{\beta}{2} R_2(\mathbf{Z}),$$

where $R_1(\mathbf{w}), R_2(\mathbf{Z})$ are the regularization terms and ℓ is an incurred loss function.

In the existing FM methods, they usually apply hinge loss for the classification task and squared loss for the regression task. To make the FM model more robust and improve its generalization ability, a natural way is to use the absolute loss function instead. However, if some extreme odd points incur very large residuals, they will still have significantly negative effects on the performance of FM model. Inspired by the recent research work [22, 3, 7], we propose to adopt the capped ℓ_1-norm loss for robust prediction, which has been successfully used to approximate the ℓ_0 norm. For the classification task, the capped hinge loss is defined as:

$$\begin{aligned} &\ell^c_{capped}(y_i, \hat{y}(\mathbf{x}_i|\mathbf{w}, \mathbf{Z})) = \min\{\ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z}), \epsilon_1\} \\ &= \min\{\max\{1 - y_i(\mathbf{w}^\top \mathbf{x}_i + \langle \mathbf{Z}, \mathbf{x}_i \mathbf{x}_i^\top \rangle), 0\}, \epsilon_1\}, \end{aligned} \tag{2.1}$$

where the true label satisfies $y_i \in \{+1, -1\}$. In this term, if the error of a sample is larger than ϵ_1, we consider this sample as an extreme outlier and its error is capped as ϵ_1 such that its effect to the whole FM model is fixed. For other normal samples, our objective will minimize $\max\{1-y_i(\mathbf{w}^\top \mathbf{x}_i+\langle \mathbf{Z}, \mathbf{x}_i \mathbf{x}_i^\top \rangle), 0\}$ directly, which is equivalent to the hinge loss. In this way, the proposed capped ℓ_1-norm is more robust than the traditional ℓ_2-norm loss in FM. Similarly, the capped ϵ-intensive loss for regression task can be written as:

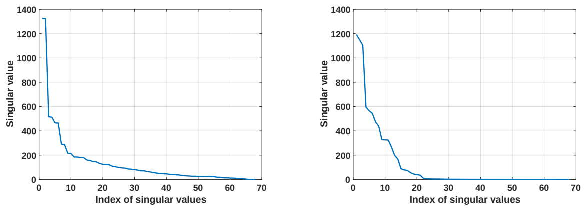
$$\begin{aligned}\ell^r_{capped}(y_i, \hat{y}(\mathbf{x}_i|\mathbf{w}, \mathbf{Z})) &= \min\{\ell_\epsilon(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z}), \epsilon_1\} \\ &= \min\{\max\{|y_i - (\mathbf{w}^\top \mathbf{x}_i + \langle \mathbf{Z}, \mathbf{x}_i\mathbf{x}_i^\top\rangle)| - \epsilon, 0\}, \epsilon_1\}.\end{aligned} \tag{2.2}$$

Recently, trace norm regularization has been employed as the convex relaxation of the rank minimization so as to approximate the low-rank structure of the feature interaction matrix in FM model [2, 21]. However, there is a non-trivial gap between trace norm minimization and the original rank minimization, especially when noises and outliers exist in the data. In particular, if the non-zero singular values of matrix $\mathbf{Z}$ change, the trace norm of $\mathbf{Z}$ will change simultaneously, yet the rank of $\mathbf{Z}$ stays the same.

To achieve a tighter approximation and more robust model, we propose a novel capped trace norm to uncover the low rank structure of the interaction matrix $\mathbf{Z}$, which is defined as:

$$R_{\epsilon_2}(\mathbf{Z}) = \sum_s \min\{\lambda_s^2, \epsilon_2\}, \tag{2.3}$$

where λ_s is the singular value of matrix $\mathbf{Z}$ and $\epsilon_2 > 0$ is a threshold value. In this term, we can approximate the rank function by $\text{rank}(\mathbf{Z}) \approx \sum_s \min\{1, \frac{\lambda_s^2}{\epsilon_2}\}$.The smaller ϵ_2 is, the more accurate the approximation would be. Note that if all the squared singular values of $\mathbf{Z}$ are greater than ϵ_2, then the approximation error will become zero. To illustrate the advantage of employing the capped trace norm, we plot the singular values of the feature interaction matrix $\mathbf{Z}$ with different low-rank optimization methods. From Figure 1, we can see that the capped trace norm only penalizes the singular values that are less than ϵ_2 and ignore other large singular values. Thus, when large singular values vary, it behaves the same as low-rank regularization, which weakens the effect of non-relevant feature interactions and makes FM robust and stable in real-world scenarios. To this end, the proposed Robust Factorization Machine (RFM) can be formulated as:



(a) Trace norm (b) Capped trace norm

Figure 1: Singular values of feature interaction matrix $\mathbf{Z}$ by two low-rank regularization methods on “phishing” dataset.

$$\min_{\mathbf{w},\mathbf{Z}} \sum_{i=1}^n \ell_{capped}(y_i, \hat{y}(\mathbf{x}_i|\mathbf{w}, \mathbf{Z})) + \frac{\alpha}{2}\|\mathbf{w}\|_2^2 + \frac{\beta}{2} R_{\epsilon_2}(\mathbf{Z}), \tag{2.4}$$

where we adopt ℓ_2-norm regularization for $\mathbf{w}$. Clearly, the new objective (2.4) is not convex and not smooth due to the definition of capped ℓ_1-norm loss and capped trace norm regularization which is difficult to optimize. In the next section, we will solve it by an efficient optimization algorithm.

Remark. Note that in literature, there are some existing work on variational trace norms trying to better approximate the rank minimization problem. Specifically, [8, 5] attempted to minimize the k-smallest singular values, which, though avoids the effect of large singular values, suffers from the tedious selection of the best rank parameter. [18, 6] proposed to minimize the sum of singular values which are smaller than a threshold value. However, minimizing the sum of capped singular values would lead to a sparse solution, that is, some small singular values will become zero while others may get large values. Our proposed capped trace norm can alleviate all these issues, since it avoids the cumbersome rank parameter selection process and minimizes the sum of capped squared singular values whose solution will be shrunk near to zero.

Besides, independent of our study, the same problem setting as robust Factorization Machine has been addressed very recently [14]. The key difference between the proposed method and the method by [14] (named as “RFM-PB” for short) is that they model uncertainty as bounded set based variability in the input signals. For each data point $\mathbf{x}$, it associates uncertainty vector $\mathbf{u} \in \mathbb{R}^d s.t. |u_j| \leq \eta_j, \forall j \in \{1, \ldots, d\}$ by characterizing the noises in the linear term and the matrix $\Sigma \in \mathbb{R}^{d\times d} s.t. |\Sigma_{jj'}| \leq \rho_{jj'}, \forall j, j' \in \{1, \ldots, d\}$ is used to capture noises induced by the quadratic term. The prediction function under interval uncertainties takes the form: $\hat{y}(\mathbf{x}|\mathbf{w}, \mathbf{Z}, \mathbf{u}, \Sigma) = \mathbf{w}^\top(\mathbf{x}+\mathbf{u}) + \langle \mathbf{Z}+\Sigma, \mathbf{x}\mathbf{x}^\top\rangle$, which results in the following minimax optimization problem:

$$\min_{\mathbf{w},\mathbf{Z}} \max_{\mathbf{u},\Sigma} \sum_{i=1}^n \ell(y_i, \hat{y}(\mathbf{x}_i|\mathbf{w}, \mathbf{Z}, \mathbf{u}, \Sigma)) + \frac{\alpha}{2} R_1(\mathbf{w}) + \frac{\beta}{2} R_2(\mathbf{Z}).$$

While in contrast, our approach enhances the robustness of FM by simply capping the prediction in the loss function, i.e., $\min\{\ell(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z}), \epsilon_1\}$. Hence, the RFM-PB performs well when the feature is under perturbed setting but our approach is not sensitive to the outliers caused by noisy labels/ratings or features. In addition, the extra parameter space for RFM-PB is $O(d^2)$, while our algorithm only needs $O(1)$ which is easy to implement and avoids the tedious parameter tuning. Finally, our approach implements the low-rank structure with the capped trace norm, while RFM-PB does not consider it. We also prove the convergence of the objective function.

3 Optimization Algorithm

In this section, we develop an efficient optimization scheme to resolve the RFM problem using the principle of re-weighted techniques [22, 23, 12, 13]. Specifically, we first apply the re-weighted method to repeatedly transform the original objective to a convex relaxation, and then the proximal gradient method is applied to solve the subsequent problem. For the classification task, we can transform the original formulation with respect to $\ell_{capped}^{c}(y_i, \hat{y}(\mathbf{x}_i|\mathbf{w}, \mathbf{Z}))$ as follows:

$$\min_{\mathbf{w},\mathbf{Z}} \sum_{i=1}^{n} e_i \ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z})^2 + \frac{\alpha}{2}\|\mathbf{w}\|_2^2 + \sum_{s} \min\{\lambda_s^2, \epsilon_2\}, \tag{3.5}$$

where $e_i = 1/(2\ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}', \mathbf{Z}'))$ if $0 < \text{error} \leq \epsilon_1$; and 0 otherwise. $\mathbf{w}', \mathbf{Z}'$ are the solution obtained from the previous stage. This new objective function can be solved via the iterative re-weighted optimization strategy.

It is obvious that the loss term of the reformulation in (3.5) is similar to the ℓ_2 loss in the original FM except that there are some weights e_i introduced for each data sample. In particular, a sample with a lower residual will have a higher weight, which is consistent with the robustness concern of RFM. To optimize the problem (3.5), we adopt the same strategy as the traditional FM by employing a two-block coordinate descent algorithm [2]. In particular, we develop the optimization algorithm that alternates between minimizing with respect to $\mathbf{w}$ and $\mathbf{Z}$ until convergence. When the algorithm terminates, it returns the final $\mathbf{w}$ and $\mathbf{Z}$. In the following, we will discuss solutions for each subproblem.

w-subproblem: By fixing the feature interaction matrix $\mathbf{Z}$, we have:

$$\min_{\mathbf{w}\in\mathbb{R}^d} \quad \textstyle\sum_{i=1}^{n} e_i \ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z})^2 + \frac{\alpha}{2}\|\mathbf{w}\|_2^2, \tag{3.6}$$

which is similar to the standard linear model except the constant bias term incurred by $\mathbf{Z}$. We can solve (3.6) by applying vanilla gradient descent method [15].

Z-subproblem: By fixing the weight vector $\mathbf{w}$, we obtain:

$$\min_{\mathbf{Z}} \sum_{i=1}^{n} e_i \ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z})^2 + \frac{\beta}{2} \sum_{s} \min\{\lambda_s^2, \epsilon_2\}.$$

Unfortunately, the capped trace norm is non-convex and non-smooth. We propose to solve it by applying the multi-stage convex relaxation technique [22, 23]. Specifically, we define a singluar value decomposition of the symmetric matrix $\mathbf{Z}$ as $\mathbf{Z} = \mathbf{P}\mathbf{\Sigma}\mathbf{P}^\top = \sum_s \lambda_s \mathbf{p}_s \mathbf{p}_s^\top$. $\mathbf{P}$ is an orthogonal matrix with columns $\mathbf{p}_s \in \mathbb{R}^d$ and $\mathbf{\Sigma} = diag(\boldsymbol{\lambda})$ is a diagonal singular matrix in ascending order. Denote the index set whose singular value is smaller than ϵ_2 as $M = \{s|\lambda_s \leq \epsilon_2\}$, and their corresponding eignvectors as $\mathbf{P}_M$. It is easy to verify $Tr(\mathbf{P}_M^\top \mathbf{Z}\mathbf{Z}^\top \mathbf{P}_M) = \sum_{s\in M} \lambda_s^2$. Thus, the Z-subproblem can be formulated as:

$$\min_{\mathbf{Z}} \sum_{i=1}^{n} e_i \ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z})^2 + \frac{\beta}{2} \sum_{s} Tr(\mathbf{P}_M^\top \mathbf{Z}\mathbf{Z}^\top \mathbf{P}_M), \tag{3.7}$$

which is convex if the matrix parameter $\mathbf{P}_M$ is fixed. We use the proximal gradient descent to optimize it iteratively. At each round, the residual of each sample is $r_i = y_i - \mathbf{w}^\top \mathbf{x}_i - \langle \mathbf{Z}, \mathbf{x}_i \mathbf{x}_i^\top \rangle$, and the subgradient with respect to $\mathbf{Z}$ is:

$$\nabla_{\mathbf{Z}} = \sum_{i=1}^{n} e_i \pi_i \mathbf{x}_i \mathbf{x}_i^\top + \beta \mathbf{P}_M \mathbf{P}_M^\top \mathbf{Z}, \tag{3.8}$$

where $\pi_i = \begin{cases} -r_i + \epsilon_1, & r_i \geq \epsilon_1; \\ -r_i - \epsilon_1, & r_i \leq -\epsilon_1; \\ 0 & otherwise \end{cases}$. Thus, $\mathbf{Z}$ is projected onto the positive semidefinite cone with $\mathbf{Z} = \Pi_{\mathbb{S}_+^{d\times d}}(\mathbf{Z} - \eta\nabla_{\mathbf{Z}})$, where η is the step size. Our optimization method is summarized in Algorithm 1.

Algorithm 1 Optimization Algorithm for the Proposed Robust Factorization Machine (RFM).

```
Input: Training data $\{(x_n, y_n\}_{n=1}^{N}$ and parameters $\alpha, \beta, \epsilon_1, \epsilon_2$.
Initialization: $e_i = 1$ for $i = 1, 2, \ldots, n$;
while not converge do
    Update $\mathbf{w}$ according to (3.6);
    Update $\mathbf{Z}$ and $\mathbf{P}_M\mathbf{P}_M^\top$ according to (3.7);
    Compute $e_i = 1$ for $i = 1, 2, \ldots, n$ according to (3.5);
end while
Output: model parameter $w$, and $\mathbf{Z}$.
```

Remark. Note that we only need to compute $\mathbf{P}_M\mathbf{P}_M^\top$ instead of $\mathbf{P}_M$ and $\mathbf{P}_M^\top$. Suppose $\mathbf{P} = [\mathbf{P}_M, \mathbf{P}_{\bar{M}}]$, where $\bar{M} = \{s|\lambda_s > \epsilon_2\}$ and $\mathbf{P}_{\bar{M}}$ are singular values whose corresponding singular vectors are larger than ϵ_2. It is easy to see that $\mathbf{P}_M\mathbf{P}_M^\top = I - \mathbf{P}_{\bar{M}}\mathbf{P}_{\bar{M}}^\top$. Since $\mathbf{Z}$ is a low-rank matrix, by setting a proper ϵ_2, the size of the set $\bar{M}$ would be very small. Thus, we can efficiently compute $\mathbf{P}_{\bar{M}}\mathbf{P}_{\bar{M}}^\top$ via truncated SVD, and then update $\mathbf{P}_M\mathbf{P}_M^\top$ accordingly.

4 Convergence Analysis

In this section, we prove the convergence of our optimization algorithm, where a local optimum can be obtained.

Lemma 4.1. *According to [20], any two hermitian matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n\times n}$ satisfy the inequality ($\lambda_i(\mathbf{A}), \lambda_i(\mathbf{B})$ are singular values sorted in the same order)*

$$\sum_{i=1}^{n} \lambda_i(\mathbf{A})\lambda_{n-i+1}(\mathbf{B}) \leq Tr(\mathbf{A}^\top \mathbf{B}) \leq \sum_{i=1}^{n} \lambda_i(\mathbf{A})\lambda_i(\mathbf{B}).$$

Lemma 4.2. *Let $\mathbf{Z} = \mathbf{P}\boldsymbol{\Sigma}\mathbf{P}^\top$, $\boldsymbol{\Sigma} = diag(\boldsymbol{\lambda})$ be one diagonal singular matrix in ascending order and $M = \{s|\lambda_s \leq \epsilon_2\}$ be the set of index whose singular values are smaller than ϵ_2. Similarly, $\hat{\mathbf{Z}} = \hat{\mathbf{P}}\hat{\boldsymbol{\Sigma}}\hat{\mathbf{P}}^\top$ is the updated parameter after $\mathbf{Z}$ with diagonal singular matrix $\hat{\boldsymbol{\Sigma}} = diag(\hat{\boldsymbol{\lambda}})$ in ascending order and $\hat{M} = \{s|\hat{\lambda}_s \leq \epsilon_2\}$ is the set of index whose singular values are smaller than ϵ_2. Then we have:*

$$\begin{aligned} &\sum_s \min\{\hat{\lambda}_s^2, \epsilon_2\} - Tr(\mathbf{P}_M^\top \hat{\mathbf{Z}}\hat{\mathbf{Z}}^\top \mathbf{P}_M) \\ &\leq \sum_s \min\{\lambda_s^2, \epsilon_2\} - Tr(\mathbf{P}_M^\top \mathbf{Z}\mathbf{Z}^\top \mathbf{P}_M). \end{aligned} \tag{4.9}$$

Proof. According to the definition of $\mathbf{P}_M$ and $\mathbf{Z}$, it is apparent that: $Tr(\mathbf{P}_M^\top \mathbf{Z}\mathbf{Z}^\top \mathbf{P}_M) = Tr(\mathbf{P}_M\mathbf{P}_M^\top \mathbf{P}\boldsymbol{\Sigma}^2\mathbf{P}^\top) = \sum_{s\in M}\lambda_s^2$. The RHS of inequality (4.9) is equivalent to:

$$\sum_{s\in M}\lambda_s^2 + \sum_{s\notin M}\epsilon_2 - \sum_{s\in M}\lambda_s^2 = \sum_{s\notin M}\epsilon_2. \tag{4.10}$$

According to the definition of $\hat{P}, \hat{M}$ and Lemma 4.1, we know that $Tr(\mathbf{P}_M^\top \hat{\mathbf{Z}}\hat{\mathbf{Z}}^\top \mathbf{P}_M) = Tr(\mathbf{P}_M\mathbf{P}_M^\top \hat{\mathbf{P}}\hat{\boldsymbol{\Sigma}}^2\hat{\mathbf{P}}^\top) \geq \sum_{s\in M}\hat{\lambda}_s^2$. Since $\hat{M}$ denotes the total eigenvalues of $\hat{\mathbf{Z}}$ that are smaller than ϵ_2, no matter how the index set $\hat{M}$ varies from M, we could obtain that $\sum_{s\in\hat{M}}\hat{\lambda}_s^2 + \sum_{s\notin\hat{M}}\epsilon_2 \leq \sum_{s\in M}\hat{\lambda}_s^2 + \sum_{s\notin M}\epsilon_2$. Therefore, via the LHS of inequality (4.9), we have:

$$\begin{aligned} &\sum_{s\in\hat{M}}\hat{\lambda}_s^2 + \sum_{s\notin\hat{M}}\epsilon_2 - Tr(\mathbf{P}_M^\top \hat{\mathbf{Z}}\hat{\mathbf{Z}}^\top \mathbf{P}_M) \\ &\leq \sum_{s\in M}\hat{\lambda}_s^2 + \sum_{s\notin M}\epsilon_2 - \sum_{s\in M}\hat{\lambda}_s^2 = \sum_{s\notin M}\epsilon_2. \end{aligned} \tag{4.11}$$

Combining inequality (4.10) and (4.11) completes the proof.

Lemma 4.3. *Given $e_i = \begin{cases} \frac{1}{2d_i}, & 0 < d_i \leq \epsilon_2; \\ 0, & otherwise, \end{cases}$, and $\forall d_i, \hat{d}_i > 0$, we could obtain the following inequality:*

$$\min\{\hat{d}_i, \epsilon_2\} - e_i\hat{d}_i^2 \leq \min\{d_i, \epsilon_2\} - e_i d_i^2 \tag{4.12}$$

Proof. Beginning with an obvious inequality

$$d_i - 2\hat{d}_i + \frac{\hat{d}_i^2}{d_i} = \frac{d_i^2 - 2d_i\hat{d}_i + \hat{d}_i^2}{d_i} = \frac{1}{d_i}(d_i - \hat{d}_i)^2 \geq 0,$$

thus we have $\hat{d}_i - \frac{\hat{d}_i^2}{2d_i} \leq \frac{d_i}{2}$.If $d_i < \epsilon_2$, it is clear that $\min\{\hat{d}_i, \epsilon_2\} \leq \hat{d}_i$, and thus

$$\min\{\hat{d}_i, \epsilon_2\} - \frac{\hat{d}_i^2}{2d_i} \leq \hat{d}_i - \frac{\hat{d}_i^2}{2d_i} \leq \frac{d_i}{2} = \min\{d_i, \epsilon_2\} - \frac{d_i^2}{2d_i}.$$

If $d_i \geq \epsilon_2$, then $e_i = 0$. As $\min\{\hat{d}_i, \epsilon_2\} \leq \epsilon_2 = \min\{d_i, \epsilon_2\}$, Eq. (4.12) also holds. Hence, Eq. (4.12) always holds.

Theorem 1. *By fixing the parameter $\mathbf{w}$, Algorithm 1 decreases the objective value of (2.4) at each iteration until it converges.*

Proof. Let $\hat{\mathbf{Z}}$ be the optimal solution to Eq. (3.7) and $\hat{\lambda}_s$ is its singular value. First of all, the objective function (3.7) is convex and its gradient is Lipschitz continuous. As the projection step is convex and closed, according to [1], if the step size is small enough, the proximal gradient descent method is guaranteed to improve the objective function value at each step. Therefore, after minimizing the objective using the proximal gradient descent at each iteration, it is guaranteed that

$$\begin{aligned} &\sum_{i=1}^{n} e_i\ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \hat{\mathbf{Z}})^2 + \frac{\beta}{2}\sum_s Tr(\mathbf{P}_M^\top \hat{\mathbf{Z}}\hat{\mathbf{Z}}^\top \mathbf{P}_M) \\ &\leq \sum_{i=1}^{n} e_i\ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z})^2 + \frac{\beta}{2}\sum_s Tr(\mathbf{P}_M^\top \mathbf{Z}\mathbf{Z}^\top \mathbf{P}_M). \end{aligned} \tag{4.13}$$

According to the definition of d_i, we apply Lemma 4.3 to the parameters $\mathbf{Z}$ and $\hat{\mathbf{Z}}$:

$$\begin{aligned} &\sum_{i=1}^{n} \min\{\ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \hat{\mathbf{Z}}), \epsilon_1\} - \sum_{i=1}^{n} e_i\ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \hat{\mathbf{Z}})^2 \\ &\leq \sum_{i=1}^{n} \min\{\ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z}), \epsilon_1\} - \sum_{i=1}^{n} e_i\ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z})^2. \end{aligned} \tag{4.14}$$

From Lemma 4.2 and the definition of $\mathbf{P}_M$, we have:

$$\begin{aligned} &\sum_s \min\{\hat{\lambda}_s^2, \epsilon_2\} - Tr(\mathbf{P}_M^\top \hat{\mathbf{Z}}\hat{\mathbf{Z}}^\top \mathbf{P}_M) \\ &\leq \sum_s \min\{\lambda_s^2, \epsilon_2\} - Tr(\mathbf{P}_M^\top \mathbf{Z}\mathbf{Z}^\top \mathbf{P}_M). \end{aligned} \tag{4.15}$$

Combining the inequalities (4.13), (4.14) and (4.15) leads to:

$$\begin{aligned} &\sum_{i=1}^{n} \min\{\ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \hat{\mathbf{Z}}), \epsilon_1\} + \frac{\beta}{2}\sum_s \min\{\hat{\lambda}_s^2, \epsilon_2\} \\ &\leq \sum_{i=1}^{n} \min\{\ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z}), \epsilon_1\} + \frac{\beta}{2}\sum_s \min\{\lambda_s^2, \epsilon_2\}. \end{aligned}$$

Theorem 2. *By fixing the parameter* $\mathbf{Z}$*, Algorithm 1 decreases the objective value of (2.4) at each iteration until it converges.*

Proof. Let $\hat{w}$ be the optimal solution of Eq. (3.6). By minimizing this objective with gradient descent, it is guaranteed that:

$$\sum_{i=1}^{n} e_i \ell_{hinge}(y_i, \mathbf{x}_i, \hat{\mathbf{w}}, \mathbf{Z})^2 + \frac{\alpha}{2}\|\hat{\mathbf{w}}\|_2 \\ \leq \sum_{i=1}^{n} e_i \ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z})^2 + \frac{\alpha}{2}\|\mathbf{w}\|_2.$$

Similar to inequality (4.14), applying Lemma 4.3 to $\mathbf{w}$ and $\hat{\mathbf{w}}$, we have:

$$\sum_{i=1}^{n} \min\{\ell_{hinge}(y_i, \mathbf{x}_i, \hat{\mathbf{w}}, \mathbf{Z}), \epsilon_1\} - \sum_{i=1}^{n} e_i \ell_{hinge}(y_i, \mathbf{x}_i, \hat{\mathbf{w}}, \mathbf{Z})^2 \\ \leq \sum_{i=1}^{n} \min\{\ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z}), \epsilon_1\} - \sum_{i=1}^{n} e_i \ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z})^2.$$

Summing over it on both sides, we can obtain:

$$\sum_{i=1}^{n} \min\{\ell_{hinge}(y_i, \mathbf{x}_i, \hat{\mathbf{w}}, \mathbf{Z}), \epsilon_1\} + \frac{\alpha}{2}\|\hat{\mathbf{w}}\|_2 \\ \leq \sum_{i=1}^{n} \min\{\ell_{hinge}(y_i, \mathbf{x}_i, \mathbf{w}, \mathbf{Z}), \epsilon_1\} + \frac{\alpha}{2}\|\mathbf{w}\|_2.$$

Corollary 1. *Algorithm 1 is guaranteed to converge finally.*

Proof. Algorithm 1 alternatively switches between optimizing $\mathbf{w}$ and $\mathbf{Z}$ while fixing the other factor respectively. Since the objective value of problem (2.4) has a lower bound 0, according to the principle of Alternating Optimization and the results of Theorem 1 and 2, we can obtain this assertion.

5 Experimental Results

In this section, we conduct an extensive set of experiments by evaluating the performance of the proposed RFM method on two popular types of machine learning tasks: classification and recommendation. Our goal is to examine the effectiveness of the proposed method on real-world data sets and evaluate its robustness performance in comparison with the state-of-the-art methods.

5.1 Experimental Testbeds We choose a variety of publicly available datasets to cover different aspects of the experimental testbeds for two types of learning tasks. Specifically, for the classification tasks, we choose six publicly available datasets from LibSVM[1] and UCI repository[2]. For the recommendation tasks, we use the typical Moevielens datasets[3] and the collection of user ratings on Amazon products of music category, instant video category and patio product category[4]. Table 1 gives a summary of the datasets used for our recommendation experiments.

Classification	#Training	#Test	#Features
phishing	7370	3685	68
protein	12263	4298	357
connect-4	40740	20368	126
w8a	49749	14951	300
IJCNN	49990	91701	22
Covtype	387342	193670	54

Recommendation	#Ratings	#Items	#Users
Movielens-100K	100000	1682	943
Amazon-patio	206250	714791	105984
Amazon-video	717651	426922	23965
Amazon-music	6396350	478235	266414

Table 1: Detailed statistics of the datasets used in the classification and the recommendation task.

5.2 Baselines and Experimental Setups We compare the empirical performance of the proposed RFM with the state-of-the-art variants of FM. The comparison between RFM and other methods focused on the aspects of robustness and low-rank structure of the feature interaction matrix. As a summary, the compared algorithms are:

- **FM**: vanilla FM [15] which uses stochastic gradient descent as the optimization algorithm.
- **CFM**: Convex Factorization Machine [2] which uses the trace norm regularization to model the low-rank structure of feature interaction matrix for the recommendation task.
- **RFM-PB**: A recent robust Factorization Machine proposed by [14], which models the data uncertainty in the input signals.
- **RFM(ours)**: the proposed RFM method by exploiting the capped ℓ_1-norm loss and the capped squared trace norm.

All the datasets are normalized to have zero mean and unit variance in each dimension. Due to the severe sparsity of the original Amazon datasets, we select a part of the most active users from the original data. The feature numbers for the music category, instant video category and patio product category

[1] https://www.csie.ntu.edu.tw/cjlin/libsvmtools/datasets

[2] https://archive.ics.uci.edu/ml/datasets.html

[3] http://grouplens.org/datasets/movielens/

[4] http://snap.stanford.edu/data/web-Amazon-links.html

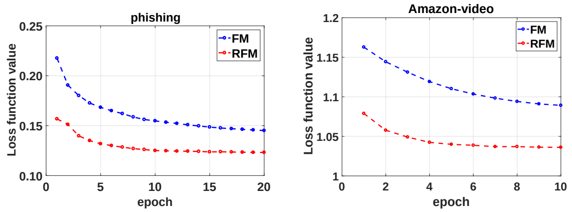


Figure 2: The loss function value during iterations.

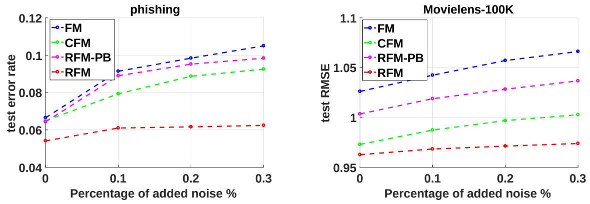
are 23717, 17960 and 16568, respectively. We evaluate the performance of our proposed methods by measuring the accuracy for the classification task and the RMSE for the recommendation task. We randomly split the recommendation datasets into training (90%) and test (10%). To make a fair comparison, all the algorithms are conducted over 5 experimental runs of different random permutations. For parameter settings, we perform grid search to pick the best parameters for each algorithm with 5-fold cross validation. We tune capped parameters ϵ_1 in the range of $[0.1, 0.2, 0.5, 1, 2, 5]$, and ϵ_2 among $[0.01, 0.05, 0.1, 0.5, 1, 5, 10]$. The regularization parameters α is in the range of $[0.001, 0.01, 0.1]$, β is among $[0.001, 0.01, 0.1]$, and the latent dimension d of FM model is among $[5, 10, 20, 30, 40, 50]$.

Amazon-patio	Train RMSE	Test RMSE
FM	1.0604±0.0002	1.1697±0.0006
CFM	1.0248±0.0042	1.1592±0.0058
RFM-PB	1.0492±0.0033	1.1610±0.0036
RFM(ours)	**0.9839±0.0054**	**1.1568±0.0051**
Amazon-video	Train RMSE	Test RMSE
FM	1.0425±0.0013	1.0892±0.0016
CFM	0.9651±0.0022	1.0531±0.0028
RFM-PB	0.8973±0.0031	1.0438±0.0023
RFM(ours)	**0.8511±0.0034**	**1.0361±0.0032**
Amazon-music	Train RMSE	Test RMSE
FM	0.9831±0.0023	0.8866±0.0026
CFM	0.7497±0.0042	0.8514±0.0038
RFM-PB	0.9452±0.0025	0.8725±0.0023
RFM(ours)	**0.8004±0.0038**	**0.8460±0.0038**
Movielens-100K	Train RMSE	Test RMSE
FM	0.9901±0.0023	1.0261±0.0026
CFM	0.9324±0.0033	0.9729±0.0040
RFM-PB	0.9864±0.0025	1.0035±0.0020
RFM(ours)	**0.8461±0.0023**	**0.9626±0.0028**

Table 3: Comparison of different algorithms in terms of train RMSE and test RMSE for recommendation task.

5.3 Results of Recommendation and Classification

Table 2 summaries the performance evaluation results of the proposed RFM and other baseline algorithms on classification tasks. Table 3 gives the evaluation results of the RMSE performance between the proposed RFM method and other baselines for recommendation task. Figure 2 shows the convergence curve of the loss function value with FM and RFM. From the experimental results, we can draw several observations as follows.



(a) phishing

(b) Movielens-100K

Figure 3: Evaluation of robustness performances on noisy datasets: classification (a) and recommendation (b). We gradually inject noise into the original data from 0 to 30%.

First of all, we see that RFM achieves slightly faster convergence and lower loss function values than the vanilla FM. This is consistent with our convergence analysis and reveals the advantage of the capped trace norm and capped ell_1-norm loss. Second, we observe that the CFM method using trace norm regularization outperform the vanilla FM and the improvements seem to be more significant on the recommendation tasks. These results confirm the effectiveness and importance of exploiting the low-rank structure of the feature interaction matrix. Third, comparing with all the methods, the proposed RFM method generally achieves the best performance on both recommendation and classification tasks consistently. The improvements over the state-of-the-art methods have been confirmed to be statistically significant according to the test results. Moreover, by examining the two different robust FM variants (RFM-PB by [14] and our RMF method), we found that both methods improve the performance of vanilla FM for most cases, which validates the advantage of endowing FM model with robustness. However, we notice that the RFM-PB by [14] is not always better than CFM for many cases; and by contrast, our RMF is consistently better than CFM. This encouraging results validate the effectiveness and importance of the proposed doubly capped norms minimization scheme.

5.4 Evaluation of Robustness Performance

For many real-world applications, data often contains noisy features due to varied reasons or even wrong annotation labels/ratings due to malicious attacks. In the previous experiments, the real-world data sets may contain some degree of noise which however is unknown. In this experiment, we aim to explicitly examine the robustness of the proposed RFM method under different levels of noise with controlled experiments.

Specifically, to evaluate the robustness of the RFM model and to validate its immunity to the threat of outliers, we conduct experiments on the phishing dataset

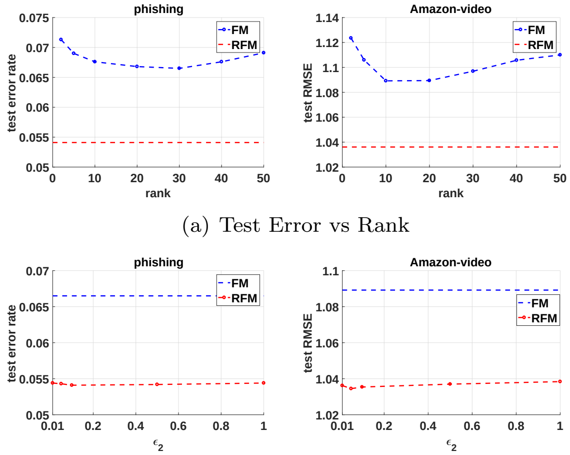
	Train ACC	Test ACC	Train ACC	Test ACC	Train ACC	Test ACC
	phishing		w8a		protein	
FM	94.01 ± 0.06	93.35 ± 0.10	99.12 ± 0.06	98.86 ± 0.07	80.56 ± 0.04	79.75 ± 0.06
CFM	94.57 ± 0.13	93.51 ± 0.10	98.86 ± 0.04	98.79 ± 0.08	80.68 ± 0.10	79.94 ± 0.13
RFM-PB	93.93 ± 0.12	93.57 ± 0.08	99.10 ± 0.12	98.99 ± 0.10	79.92 ± 0.14	79.94 ± 0.12
RFM(ours)	**95.87 ± 0.26**	**94.59 ± 0.33**	**99.56 ± 0.06**	**99.10 ± 0.06**	**82.52 ± 0.20**	**80.34 ± 0.18**
	IJCNN		Covtype		connect-4	
FM	96.45 ± 0.12	96.66 ± 0.09	78.12 ± 0.16	77.31 ± 0.10	89.16 ± 0.18	88.52 ± 0.16
CFM	94.72 ± 0.13	92.77 ± 0.10	78.01 ± 0.20	77.99 ± 0.18	89.21 ± 0.13	88.95 ± 0.23
RFM-PB	96.75 ± 0.12	97.12 ± 0.08	78.61 ± 0.22	78.83 ± 0.14	89.73 ± 0.14	89.39 ± 0.10
RFM(ours)	**98.22 ± 0.16**	**97.83 ± 0.15**	**80.16 ± 0.26**	**79.65 ± 0.25**	**89.88 ± 0.23**	**89.90 ± 0.31**

Table 2: Comparison of different algorithms in terms of train accuracy and test accuracy for classification task.

for the classification task and Movielens-100K for the recommendation task. We inject noise into the original data from 0 to 30% each time and evaluate the robustness of the model via test RMSE for recommendation task and test error rate for classification task. All these noises are set to be the wrong labels/ratings with randomly chosen instances. It is obvious that the objective of the proposed RFM could explicitly deal with the noisy training data and the value of ϵ_1 is used to filter out the outliers. In this way, RFM protects the FM model structure from being distorted.

Figure 3(a) shows the results for the classification task, it is obvious that when the noise increases, the test error rate of all the methods tend to rise. But the performance degradation of RFM is remarkably lower than the baselines. While the percentage of the noise increases from 10% to 30%, the RFM model is much more stable than the baselines. Thus, it validates that RFM model can deal with outliers properly, and keep the right structure in the training model. Figure 3(b) demonstrates the results for the recommendation task. Similar with classification task, the performance degradation of FM gets larger while the performance of RFM still tends to be stable. Note that the RFM-PB does not demonstrate its robustness in this setting. The reason is that RFM-PB only performs well when the feature is under perturbed setting but RFM is immue to the outliers from noisy labels/ratings.

5.5 Sensitivity Evaluation of Rank Parameter d and Capped Trace Norm ϵ_2 We aim to evaluate the sensitivity of the rank parameter d in the vanilla FM and the sensitivity of the threshold value ϵ_2 in the RFM. Both of these hyperparameter control the low-rank extent of the feature interaction matrix. As we can see in Figure 4(a), the vanilla FM is sensitive to the choice of the rank parameter d, but the choice of a good rank value d is often nontrivial. For example, on the protein dataset, when $d < 10$ or $d > 20$ the test RMSE changes dramatically. In order to derive the best performance, it usually requires the tedious tuning. Compared with the sensitivity analysis of the rank parameter d in vanilla FM, Figure 4(b) demonstrates that RFM is more stable with the threshold value ϵ_2. When the cost of hyperparameter tuning is restricted, RFM is more applicable to quickly attain an acceptable performance.



(a) Test Error vs Rank

(b) Test Error vs ϵ_2

Figure 4: Impact of Hyperparameters

6 Conclusion

In this paper, we propose a novel framework of Robust Factorization Machines (RFM) method based on a doubly capped norms minimization approach, where we integrate both a capped ℓ_1 loss and a capped squared trace norm. We show that the proposed capped squared trace norm can approximate the rank minimization problem much tighter than the traditional trace norm, and thus achieve a better rank minimization approximation. The capped ℓ_1 loss is able to enhance the robustness of the FM model. However, the non-convexity and non-smoothness of the new objective function makes the optimization problem non-trivial to solve. We thus propose an efficient optimization procedure to tackle the optimisation task effectively with proved convergence analysis.

The experimental evaluations were conducted on both classification and recommendation tasks in comparison with several FM variants and RFM, in which encouraging results validate the effectiveness of the proposed RFM approach.

Acknowledgements

This research is supported by the National Research Foundation Singapore under its AI Singapore Programme [AISG-RP-2018-001].

References

[1] A. Beck and M. Teboulle. Gradient-based algorithms with applications to signal recovery. *Convex optimization in signal processing and communications*, pages 42–88, 2009.

[2] M. Blondel, A. Fujino, and N. Ueda. Convex factorization machines. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 19–35. Springer, 2015.

[3] H. Gao, F. Nie, W. Cai, and H. Huang. Robust capped norm nonnegative matrix factorization: Capped norm nmf. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 871–880. ACM, 2015.

[4] E. Hazan. Sparse approximate solutions to semidefinite programs. In *Latin American Symposium on Theoretical Informatics*, pages 306–316. Springer, 2008.

[5] Z. Huo, S. Gao, W. Cai, and H. Huang. Video recovery via learning variation and consistency of images. In *AAAI*, pages 4082–4088, 2017.

[6] Z. Huo, F. Nie, and H. Huang. Robust and effective metric learning using capped trace norm: Metric learning via capped trace norm. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1605–1614. ACM, 2016.

[7] W. Jiang, F. Nie, and H. Huang. Robust dictionary learning with capped l1-norm. In *IJCAI*, pages 3590–3596, 2015.

[8] M. T. Law, N. Thome, and M. Cord. Fantope regularization in metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1051–1058, 2014.

[9] X. Lin, W. Zhang, M. Zhang, W. Zhu, J. Pei, P. Zhao, and J. Huang. Online compact convexified factorization machine. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pages 1633–1642. International World Wide Web Conferences S-teering Committee, 2018.

[10] C.-T. Lu, L. He, W. Shao, B. Cao, and P. S. Yu. Multilinear factorization machines for multi-task multi-view learning. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 701–709. ACM, 2017.

[11] T. V. Nguyen, A. Karatzoglou, and L. Baltrunas. Gaussian process factorization machines for context-aware recommendations. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 63–72. ACM, 2014.

[12] F. Nie, H. Huang, X. Cai, and C. H. Ding. Efficient and robust feature selection via joint 2, 1-norms minimization. In *Advances in neural information processing systems*, pages 1813–1821, 2010.

[13] F. Nie, J. Yuan, and H. Huang. Optimal mean robust principal component analysis. In *Proceedings of the 31st international conference on machine learning (ICML-14)*, pages 1062–1070, 2014.

[14] S. Punjabi and P. Bhatt. Robust factorization machines for user response prediction. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pages 669–678. International World Wide Web Conferences Steering Committee, 2018.

[15] S. Rendle. Factorization machines. *Proceedings - IEEE International Conference on Data Mining, ICD-M*, pages 995–1000, 2010.

[16] S. Rendle. Factorization machines with libfm. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 3(3):57, 2012.

[17] S. Shalev-shwartz, A. Gonen, and O. Shamir. Large-scale convex minimization with a low-rank constraint. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 329–336, 2011.

[18] Q. Sun, S. Xiang, and J. Ye. Robust principal component analysis via capped norms. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 311–319. ACM, 2013.

[19] M. Tao and X. Yuan. Recovering low-rank and sparse components of matrices from incomplete and noisy observations. *SIAM Journal on Optimization*, 21(1):57–81, 2011.

[20] C. Theobald. An inequality for the trace of the product of two symmetric matrices. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 77, pages 265–267. Cambridge Univ Press, 1975.

[21] M. Yamada, W. Lian, A. Goyal, J. Chen, K. Wimalawarne, S. A. Khan, S. Kaski, H. Mamitsuka, and Y. Chang. Convex factorization machine for toxicogenomics prediction. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1215–1224. ACM, 2017.

[22] T. Zhang. Multi-stage convex relaxation for learning with sparse regularization. In *Advances in Neural Information Processing Systems*, pages 1929–1936, 2009.

[23] T. Zhang. Analysis of multi-stage convex relaxation for sparse regularization. *Journal of Machine Learning Research*, 11(Mar):1081–1107, 2010.