Supplementary Material of VeST

MoonJeong Park[1], Jun-Gi Jang[2], and Lee Sael(✉)[2]

[1] Daegu Gyeongbuk Institute of Science and Technology, Daegu, Korea
moonjeong94@dgist.ac.kr
[2] Department of Computer Science and Engineering, Seoul National University, Seoul, Korea
{elnino4,saellee}@snu.ac.kr

Abstract. In this supplementary material, we describe additional proofs, algorithms, and experimental results of VeST.

1 Proofs

1.1 Update Rules with L_F Regularization

We provide proof of correctness for the entry-wise update rule with L_F regularization.

Lemma 1 (Update rule for factor matrices with L_F regularization).

$$a^{(n)}_{i_n j_n} \longleftarrow \underset{a^{(n)}_{i_n j_n}}{\arg\min}\ \mathcal{L}_F(\mathcal{G}, \mathbf{A}^{(1)}, ..., \mathbf{A}^{(N)}) = \frac{\Big(\sum\limits_{\forall\alpha\in\Omega^{(n)}_{i_n}} \mathcal{X}_\alpha \delta^{(n)}_\alpha(j_n)\Big) - \Big(\sum\limits_{\forall t\neq j_n} \boldsymbol{v}^{(n)}_{i_n j_n}(t)\cdot a^{(n)}_{i_n t}\Big)}{\boldsymbol{v}^{(n)}_{i_n j_n}(j_n) + \lambda}, \tag{1}$$

where $\boldsymbol{v}^{(n)}_{i_n j_n}$ is a length J_n vector whose jth element is

$$\boldsymbol{v}^{(n)}_{i_n j_n}(j) = \sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} \delta^{(n)}_\alpha(j)\delta^{(n)}_\alpha(j_n), \tag{2}$$

$\delta^{(n)}_\alpha$ is a length J_n vector whose jth element is

$$\delta^{(n)}_\alpha(j) = \sum_{\forall\beta_{j_n=j}\in\mathcal{G}} \mathcal{G}_{\beta_{j_n=j}} \prod_{k\neq n} a^{(k)}_{i_k j_k}, \tag{3}$$

$\Omega^{(n)}_{i_n}$ is the subset of Ω whose index of nth mode is i_n, and λ is a regularization parameter. □

Proof. The value that makes partial derivative of L_F regularization loss with regard to the factor matrix entry $a^{(n)}_{i_n j_n}$ to zero is as follows:

$$\frac{\partial\Big(\sum\limits_{\forall\alpha\in\Omega}\Big(\mathcal{X}_\alpha - \sum\limits_{\forall\beta\in\mathcal{G}}\mathcal{G}_\beta\prod\limits_{n=1}^{N} a^{(n)}_{i_n j_n}\Big)^2 + \lambda\big(\sum\limits_{\forall\beta\in\mathcal{G}}{\mathcal{G}_\beta}^2 + \sum\limits_{n=1}^{N}\sum\limits_{(i_n,j_n)\in\mathbf{A}^{(n)}} {a^{(n)}_{i_n j_n}}^2\big)\Big)}{\partial a^{(n)}_{i_n j_n}} = 0$$

$$\Leftrightarrow \sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} \Big(\big(\mathcal{X}_\alpha - \sum_{\forall\beta\in\mathcal{G}}\mathcal{G}_\beta \prod_{n=1}^{N} a^{(n)}_{i_n j_n}\big)\times\big(-\delta^{(n)}_\alpha(j_n)\big)\Big) + \lambda a^{(n)}_{i_n j_n} = 0$$

$$\Leftrightarrow -\sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} \mathfrak{X}_\alpha \delta^{(n)}_\alpha(j_n) + \sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} (\sum_{\forall\beta\in\mathfrak{G}} \mathfrak{G}_\beta \prod_{n=1}^{N} a^{(n)}_{i_n j_n}) \cdot \delta^{(n)}_\alpha(j_n) + \lambda a^{(n)}_{i_n j_n} = 0$$

$$\Leftrightarrow -\sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} \mathfrak{X}_\alpha \delta^{(n)}_\alpha(j_n) + \sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} (\sum_{t=1}^{J_n} \delta^{(n)}_\alpha(t) a^{(n)}_{i_n t}) \cdot (\delta^{(n)}_\alpha(j_n)) + \lambda a^{(n)}_{i_n j_n} = 0$$

$$\Leftrightarrow -\sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} \mathfrak{X}_\alpha \delta^{(n)}_\alpha(j_n) + \sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} (\sum_{t\neq j_n} \delta^{(n)}_\alpha(t) a^{(n)}_{i_n t} + \delta^{(n)}_\alpha(j_n) a^{(n)}_{i_n j_n}) \cdot (\delta^{(n)}_\alpha(j_n)) + \lambda a^{(n)}_{i_n j_n} = 0$$

$$\Leftrightarrow \big(\sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} (\delta^{(n)}_\alpha(j_n))^2 + \lambda \big) a^{(n)}_{i_n j_n} - \sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} \big(\mathfrak{X}_\alpha \delta^{(n)}_\alpha(j_n) - \sum_{t\neq j_n} \delta^{(n)}_\alpha(t) a^{(n)}_{i_n t} \delta^{(n)}_\alpha(j_n) \big) = 0$$

$$\Leftrightarrow a^{(n)}_{i_n j_n} = \frac{\sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} \big(\mathfrak{X}_\alpha \delta^{(n)}_\alpha(j_n) - \sum_{t\neq j_n} \delta^{(n)}_\alpha(t) a^{(n)}_{i_n t} \delta^{(n)}_\alpha(j_n) \big)}{\sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} (\delta^{(n)}_\alpha(j_n))^2 + \lambda}$$

$$\Leftrightarrow a^{(n)}_{i_n j_n} = \frac{\big(\sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} \mathfrak{X}_\alpha \delta^{(n)}_\alpha(j_n) \big) - \big(\sum_{\forall t\neq j_n} \boldsymbol{v}^{(n)}_{i_n j_n}(t) \cdot a^{(n)}_{i_n t} \big)}{\boldsymbol{v}^{(n)}_{i_n j_n}(j_n) + \lambda},$$

where $\boldsymbol{v}^{(n)}_{i_n j_n}$ is a length J_n vector whose jth entry is

$$\boldsymbol{v}^{(n)}_{i_n j_n}(j) = \sum_{\forall\alpha\in\Omega^{(n)}_{i_n}} \delta^{(n)}_\alpha(j) \delta^{(n)}_\alpha(j_n),$$

$\delta^{(n)}_\alpha$ is a length J_n vector whose jth entry is

$$\delta^{(n)}_\alpha(j) = \sum_{\forall\beta_{j_n=j}\in\mathfrak{G}} \mathfrak{G}_{\beta_{j_n=j}} \prod_{k\neq n} a^{(k)}_{i_k j_k},$$

$\Omega^{(n)}_{i_n}$ is the subset of Ω whose index of nth mode is i_n, and λ is a regularization parameter. □

Similar to the derivation of update rules for factor matrices, the entries of the core tensor is updated by making the partial derivative with respect to $\mathfrak{G}_\beta$ to zero.

Lemma 2 (Update rule for core tensor with L_F regularization).

$$\mathfrak{G}_\beta \longleftarrow \frac{\sum_{\forall\alpha\in\Omega} (\mathfrak{X}_\alpha - \sum_{\forall\gamma\neq\beta} \mathfrak{G}_\gamma \prod_{n=1}^{N} a^{(n)}_{i_n j_n}) \cdot \prod_{n=1}^{N} a^{(n)}_{i_n j_n}}{\lambda + \sum_{\forall\alpha\in\Omega} (\prod_{n=1}^{N} a^{(n)}_{i_n j_n})^2} \tag{4}$$

□

Proof.

$$\frac{\partial \Big(\sum_{\forall\alpha\in\Omega} \Big(\mathfrak{X}_\alpha - \sum_{\forall\gamma\in\mathfrak{G}} \mathfrak{G}_\gamma \prod_{n=1}^{N} a^{(n)}_{i_n j_n} \Big)^2 + \lambda \big(\sum_{\forall\gamma\in\mathfrak{G}} {\mathfrak{G}_\gamma}^2 + \sum_{n=1}^{N} \sum_{(i_n, j_n)\in\mathbf{A}^{(n)}} {a^{(n)}_{i_n j_n}}^2 \big) \Big)}{\partial \mathfrak{G}_\beta} = 0$$

$$\Leftrightarrow \sum_{\forall \alpha \in \Omega} (\mathfrak{X}_\alpha - \sum_{\forall \gamma \in \mathfrak{G}} \mathfrak{G}_\gamma \prod_{n=1}^{N} a^{(n)}_{i_n j_n}) \cdot (-\prod_{n=1}^{N} a^{(n)}_{i_n j_n}) + \lambda \mathfrak{G}_\beta = 0$$

$$\Leftrightarrow -\sum_{\forall \alpha \in \Omega} (\mathfrak{X}_\alpha - \sum_{\forall \gamma \neq \beta} \mathfrak{G}_\gamma \prod_{n=1}^{N} a^{(n)}_{i_n j_n} - \mathfrak{G}_\beta \prod_{n=1}^{N} a^{(n)}_{i_n j_n}) \cdot \prod_{n=1}^{N} a^{(n)}_{i_n j_n} + \lambda \mathfrak{G}_\beta = 0$$

$$\Leftrightarrow (\sum_{\forall \alpha \in \Omega} (\prod_{n=1}^{N} a^{(n)}_{i_n j_n})^2 + \lambda) \mathfrak{G}_\beta - \sum_{\forall \alpha \in \Omega} (\mathfrak{X}_\alpha - \sum_{\forall \gamma \neq \beta} \mathfrak{G}_\gamma \prod_{n=1}^{N} a^{(n)}_{i_n j_n}) \cdot \prod_{n=1}^{N} a^{(n)}_{i_n j_n} = 0$$

$$\Leftrightarrow \mathfrak{G}_\beta = \frac{\sum_{\forall \alpha \in \Omega} (\mathfrak{X}_\alpha - \sum_{\forall \gamma \neq \beta} \mathfrak{G}_\gamma \prod_{n=1}^{N} a^{(n)}_{i_n j_n}) \cdot \prod_{n=1}^{N} a^{(n)}_{i_n j_n}}{\lambda + \sum_{\forall \alpha \in \Omega} (\prod_{n=1}^{N} a^{(n)}_{i_n j_n})^2}$$

□

1.2 Update Rules with L_1 Regularization

Lemma 3 (Update rule for factor matrix with L_1 regularization).

$$\underset{a^{(n)}_{i_n j_n}}{\arg\min}\; L_1(\mathfrak{G}, \mathbf{A}^{(1)}, ..., \mathbf{A}^{(N)}) = \begin{cases} (\lambda - g_{fm})/d_{fm} & \text{if } g_{fm} > \lambda \\ -(\lambda + g_{fm})/d_{fm} & \text{if } g_{fm} < -\lambda \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where

$$g_{fm} = -2 \sum_{\forall \alpha \in \Omega^{(n)}_{i_n}} \big(\mathfrak{X}_\alpha \delta^{(n)}_\alpha (j_n) - \sum_{\forall t \neq j_n} \boldsymbol{v}^{(n)}_{i_n j_n}(t)\big), \tag{6}$$

$$d_{fm} = 2 \boldsymbol{v}^{(n)}_{i_n j_n}(j_n), \tag{7}$$

$\boldsymbol{v}^{(n)}_{i_n j_n}$ is a length J_n vector whose jth element is

$$\boldsymbol{v}^{(n)}_{i_n j_n}(j) = \sum_{\forall \alpha \in \Omega^{(n)}_{i_n}} \delta^{(n)}_\alpha (j) \delta^{(n)}_\alpha (j_n), \tag{8}$$

$\delta^{(n)}_\alpha$ is a length J_n vector whose jth element is

$$\delta^{(n)}_\alpha (j) = \sum_{\forall \beta_{j_n = j} \in \mathfrak{G}} \mathfrak{G}_{\beta_{j_n = j}} \prod_{k \neq n} a^{(k)}_{i_k j_k}, \tag{9}$$

$\Omega^{(n)}_{i_n}$ is the subset of Ω whose index of nth mode is i_n, and λ is a regularization parameter. □

Proof. The partial derivative of L_1 regularization loss function with regard to the factor matrix entry $a^{(n)}_{i_n j_n}$ is

$$
\begin{aligned}
&\frac{\partial L_1}{\partial a^{(n)}_{i_n j_n}} \\
&= \left[2 \sum_{\forall \alpha \in \Omega^{(n)}_{i_n}} \left((\mathcal{X}_\alpha - \sum_{\forall t \neq j_n} \delta^{(n)}_\alpha (t) a^{(n)}_{i_n t}) \cdot (-\delta^{(n)}_\alpha (j_n)) \right) \right] + \left[2 \sum_{\forall \alpha \in \Omega^{(n)}_{i_n}} \left(\delta^{(n)}_\alpha (j_n) \right)^2 \cdot a^{(n)}_{i_n j_n} \right] + \lambda \frac{\partial |a^{(n)}_{i_n j_n}|}{\partial a^{(n)}_{i_n j_n}} \\
&= g_{fm} + d_{fm} \cdot a^{(n)}_{i_n j_n} + \lambda \frac{\partial |a^{(n)}_{i_n j_n}|}{\partial a^{(n)}_{i_n j_n}} \\
&= \begin{cases} g_{fm} + d_{fm} \cdot a^{(n)}_{i_n j_n} + \lambda & \text{if } a^{(n)}_{i_n j_n} > 0 \\ g_{fm} + d_{fm} \cdot a^{(n)}_{i_n j_n} - \lambda & \text{if } a^{(n)}_{i_n j_n} < 0 \end{cases}
\end{aligned}
\tag{10}
$$

Case 1. $(g_{fm} > \lambda (> 0))$: if $a^{(n)}_{i_n j_n} > 0$, $\frac{\partial L_1}{\partial a^{(n)}_{i_n j_n}} > 0$ since $\lim_{a^{(n)}_{i_n j_n} \to +0} \frac{\partial L_1}{\partial a^{(n)}_{i_n j_n}} = g_{fm} + \lambda > 0$, and $\frac{\partial^2 L_1}{\partial (a^{(n)}_{i_n j_n})^2} = d_{fm} > 0$. $\frac{\partial L_1}{\partial a^{(n)}_{i_n j_n}} = 0$ when $a^{(n)}_{i_n j_n} = (\lambda - g_{fm})/d_{fm} (> 0)$, so that the value of $\frac{\partial L_1}{\partial a^{(n)}_{i_n j_n}}$ becomes negative when $a^{(n)}_{i_n j_n} < (\lambda - g_{fm})/d_{fm}$. In sum, L_1 decreases if $a_{i_n j_n} < (\lambda - g_{fm})/d_{fm}$, and increases if $a_{i_n j_n} > (\lambda - g_{fm})/d_{fm}$. Thus, the loss becomes minimum when $a^{(n)}_{i_n j_n} = (\lambda - g_{fm})/d_{fm}$.

Case 2. $(g_{fm} < -\lambda (< 0))$: likewise, if $a^{(n)}_{i_n j_n} < 0$, $\frac{\partial L_1}{\partial a^{(n)}_{i_n j_n}} < 0$ since $\lim_{a^{(n)}_{i_n j_n} \to -0} \frac{\partial L_1}{\partial a^{(n)}_{i_n j_n}} = g_{fm} - \lambda < 0$, and $\frac{\partial^2 L_1}{\partial (a^{(n)}_{i_n j_n})^2} = d_{fm} > 0$. $\frac{\partial L_1}{\partial a^{(n)}_{i_n j_n}} = 0$ when $a^{(n)}_{i_n j_n} = -(\lambda + g_{fm})/d_{fm} (< 0)$, so that the value of $\frac{\partial L_1}{\partial a^{(n)}_{i_n j_n}}$ becomes positive when $a^{(n)}_{i_n j_n} > -(\lambda + g_{fm})/d_{fm}$. In sum, the loss decreases if $a_{i_n j_n} < -(\lambda + g_{fm})/d_{fm}$, and increases if $a_{i_n j_n} > -(\lambda + g_{fm})/d_{fm}$. Thus, the loss becomes minimum when $a^{(n)}_{i_n j_n} = -(\lambda + g_{fm})/d_{fm}$ respect to $a^{(n)}_{i_n j_n}$.

Case 3. $(-g_{fm} < \lambda < g_{fm})$: If $a^{(n)}_{i_n j_n} > 0$, $\frac{\partial L}{\partial a^{(n)}_{i_n j_n}} > d_{fm} \cdot a^{(n)}_{i_n j_n} > 0$; if $a^{(n)}_{i_n j_n} < 0$, $\frac{\partial L}{\partial a^{(n)}_{i_n j_n}} < d_{fm} \cdot a^{(n)}_{i_n j_n} < 0$. In sum, the loss decreases if $a_{i_n j_n} < 0$, and increases if $a_{i_n j_n} > 0$. Thus, the loss becomes minimum when $a^{(n)}_{i_n j_n} = 0$.

□

Lemma 4 (Update rule for core tensor with L_1 regularization).

$$
\underset{a^{(n)}_{i_n j_n}}{\arg\min}\ L_1(\mathcal{G}, \mathbf{A}^{(1)}, ..., \mathbf{A}^{(N)}) = \begin{cases} (\lambda - g_c)/d_c & \textit{if } g_c > \lambda \\ -(\lambda + g_c)/d_c & \textit{if } g_c < -\lambda \\ 0 & \textit{otherwise} \end{cases}
\tag{11}
$$

where

$$g_c = -2 \sum_{\forall \alpha \in \Omega} (\mathcal{X}_\alpha - \sum_{\forall \gamma \neq \beta} \mathcal{G}_\gamma \prod_{n=1}^{N} a_{i_n j_n}^{(n)}) \cdot \prod_{n=1}^{N} a_{i_n j_n}^{(n)}, \tag{12}$$

and

$$d_c = 2 \sum_{\forall \alpha \in \Omega} (\prod_{n=1}^{N} a_{i_n j_n}^{(n)})^2. \tag{13}$$

□

Proof. The partial derivative of L_1 regularization loss function with regard to core tensor entry $\mathcal{G}_\beta$ is

$$\begin{aligned}
&\frac{\partial L_1}{\partial \mathcal{G}_\beta} \\
&= 2 \sum_{\forall \alpha \in \Omega} (\mathcal{X}_\alpha - \sum_{\forall \gamma \neq \beta} \mathcal{G}_\gamma \prod_{n=1}^{N} a_{i_n j_n}^{(n)} - \mathcal{G}_\beta \prod_{n=1}^{N} a_{i_n j_n}^{(n)}) \cdot (-\prod_{n=1}^{N} a_{i_n j_n}^{(n)}) + \lambda \frac{\partial |\mathcal{G}_\beta|}{\partial \mathcal{G}_\beta} \\
&= \Big[-2 \sum_{\forall \alpha \in \Omega} (\mathcal{X}_\alpha - \sum_{\forall \gamma \neq \beta} \mathcal{G}_\gamma \prod_{n=1}^{N} a_{i_n j_n}^{(n)}) \cdot (\prod_{n=1}^{N} a_{i_n j_n}^{(n)}) \Big] + \Big[2(\prod_{n=1}^{N} a_{i_n j_n}^{(n)})^2 \cdot \mathcal{G}_\beta \Big] + \lambda \frac{\partial |\mathcal{G}_\beta|}{\partial \mathcal{G}_\beta} \\
&= g_c + d_c \cdot \mathcal{G}_\beta + \lambda \frac{\partial |\mathcal{G}_\beta|}{\partial \mathcal{G}_\beta} \\
&= \begin{cases} g_c + d_c \cdot \mathcal{G}_\beta + \lambda & \text{if } \mathcal{G}_\beta > 0 \\ g_c + d_c \cdot \mathcal{G}_\beta - \lambda & \text{if } \mathcal{G}_\beta < 0 \end{cases}
\end{aligned} \tag{14}$$

The remaining steps are the same as those of update rule for factor matrices with L_1 regularization (Lemma 3). □

1.3 Derivation of responsibility values of factor matrix entries

Proof. From its definition,

$$\begin{aligned}
(RE)^2 ||\mathcal{X}||_F^2 &= \sum_{\forall \alpha \in \Omega} (\mathcal{X}_\alpha - B(\alpha))^2 \\
&= \sum_{\forall \alpha \in \Omega_{i_n}^{(n)}} (\mathcal{X}_\alpha - B(\alpha))^2 + \sum_{\forall \alpha \notin \Omega_{i_n}^{(n)}} (\mathcal{X}_\alpha - B(\alpha))^2 \\
&= \sum_{\forall \alpha \in \Omega_{i_n}^{(n)}} (\mathcal{X}_\alpha - B_{j_n=j}(\alpha) - B_{j_n \neq j}(\alpha))^2 + \sum_{\forall \alpha \notin \Omega_{i_n}^{(n)}} (\mathcal{X}_\alpha - B(\alpha))^2
\end{aligned} \tag{15}$$

Note that

$$\sum_{\forall \alpha \notin \Omega_{i_n}^{(n)}} (\mathcal{X}_\alpha - B(\alpha))^2 = (RE)^2 ||\mathcal{X}||_F^2 - \sum_{\forall \alpha \in \Omega_{i_n}^{(n)}} (\mathcal{X}_\alpha - B_{j_n=j}(\alpha) - B_{j_n \neq j}(\alpha))^2 \tag{16}$$

Thus,

$$
\begin{aligned}
&(RE(a_{ij}^{(n)}))^2 ||\mathcal{X}||_F^2 \\
=&\Big(\sum_{\forall\alpha\in\Omega_{i_n}^{(n)}} (\mathcal{X}_\alpha - B_{j_n\neq j}(\alpha))^2\Big) + \sum_{\forall\alpha\notin\Omega_{i_n}^{(n)}} (\mathcal{X}_\alpha - B(\alpha))^2 \\
=&\Big(\sum_{\forall\alpha\in\Omega_{i_n}^{(n)}} (\mathcal{X}_\alpha - B_{j_n\neq j}(\alpha))^2\Big) + (RE)^2||\mathcal{X}||_F^2 - \sum_{\forall\alpha\in\Omega_{i_n}^{(n)}} (\mathcal{X}_\alpha - B_{j_n=j}(\alpha) - B_{j_n\neq j}(\alpha))^2 \\
=&(RE)^2||\mathcal{X}||_F^2 + \sum_{\forall\alpha\in\Omega_{i_n}^{(n)}} (2\mathcal{X}_\alpha - 2B_{j_n\neq j}(\alpha) - B_{j_n=j}(\alpha)) \cdot B_{j_n=j}(\alpha) \\
=&(RE)^2||\mathcal{X}||_F^2 + \sum_{\forall\alpha\in\Omega_{i_n}^{(n)}} (2\cdot(\mathcal{X}_\alpha - B(\alpha)) + B_{j_n=j}(\alpha)) \cdot B_{j_n=j}(\alpha)
\end{aligned}
\tag{17}
$$

Dividing both sides of Eq. (17) with $||\mathcal{X}||_F^2$, we get

$$
(RE(a_{ij}^{(n)}))^2 = RE^2 + \frac{\sum_{\forall\alpha\in\Omega_{i_n}^{(n)}} \big(2\cdot(\mathcal{X}_\alpha - B(\alpha)) + B_{j_n=j}(\alpha)\big)\cdot(B_{j_n=j}(\alpha))}{||\mathcal{X}||_F^2} \tag{18}
$$

□

2 Core Tensor Update Algorithm

Element-wise update of the core tensor $\mathcal{G}$ using either the L_F or L_1 regularization is detailed in Algorithm 1. Each elements of a core tensor are highly dependent on each other and thus cannot be made parallel. However, considering that typical size $|\mathcal{G}|$ of a core tensor is small, the core tensor updates are minor burden in the computational process.

3 Theoretical Analysis

3.1 Convergence analysis.

We theoretically prove the convergence of VEST update rules.

Theorem 1. VEST$^*_{L_1}$ *and* VEST$^*_{L_F}$ *converges.*

Proof. Both of the loss functions (Eq.(1) and (2) of the main paper) are bounded by 0. Since the proposed element-wise update rule minimizes the loss function at each update sessions, the loss function decreases in every update process and never increases. Thus, VEST converges. □

Algorithm 1: Parallel element-wise core tensor update

Input : Tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$,
factor matrices $A^{(n)} \in \mathbb{R}^{I_n \times J_n} (n = 1, \cdots, N)$, and
core tensor $\mathcal{G} \in J_1 \times ... J_N$.
Output: Updated core tensor $\mathcal{G} \in J_1 \times ... J_N$

for $\alpha = \forall(i_1, ..., i_N) \in \Omega$ **do** ▷ `in parallel`
  calculate $B(\alpha) = \sum_{\forall \beta = (j_1, \cdots, j_N) \in \mathcal{G}} \mathcal{G}_\beta \prod_{n=1}^{N} a_{i_n j_n}^{(n)}$
for $\beta = \forall(j_1, ..., j_N) \in \mathcal{G}$ **do**
  calculate $\sum_{\forall \alpha \in \Omega} (\mathcal{X}_\alpha - B(\alpha) + \mathcal{G}_\beta \prod_{n=1}^{N} a_{i_n j_n}^{(n)}) \cdot \prod_{n=1}^{N} a_{i_n j_n}^{(n)}$
  calculate $\sum_{\forall \alpha \in \Omega} (\prod_{n=1}^{N} a_{i_n j_n}^{(n)})^2$
  update $\mathcal{G}_\beta$ using Eq. (4) for L_F (use Eq. (11) for L_1).

3.2 Complexity Analysis

We analyze time and memory complexities of VEST. Assuming equal mode dimensions for a input tensor and a core tensor, i.e., $I_1 = I_2 = \cdots = I_N$ and $J_1 = J_2 = \cdots = J_N$, time complexity of VEST is $O(N^2 J|\mathcal{G}||\Omega|/T)$, and memory complexity is $O(TJ + J^N + NIJ)$. Note that we calculate time complexities per iteration, and we focus on memory complexities of intermediate data, not for all variables.

Theorem 2. *The time complexity per iteration of* VEST *is*

$$O(N^2 J|\mathcal{G}||\Omega|/T).$$

Proof. Given the $a_{ij}^{(n)}$ that is not pruned (lines 1-5 in Algorithm 4 of the main paper), computing $\delta_\alpha^{(n)}(j)$ (line 6-8) takes $O(N|\mathcal{G}|)$. Calculating $\sum_{\forall \alpha \in \Omega_{i_n}^{(n)}} \mathcal{X}_\alpha \delta_\alpha^{(n)}(j_n)$ and $\boldsymbol{v}_{i_n j_n}^{(n)}$ (line 9) takes $O(N|\Omega_{i_n}^{(n)}||\mathcal{G}|)$. Therefore, the time complexity for updating an element $a_{ij}^{(n)}$ (line 8) is $O(N|\Omega_{i_n}^{(n)}||\mathcal{G}|)$. Updating a i-th row $a_{i:}^{(n)}$ takes $O(NJ|\Omega_{i_n}^{(n)}||\mathcal{G}|)$. Updating all elements of $A^{(n)}$ takes $O(NJ|\mathcal{G}||\Omega|/T)$, using T number of threads. Thus, updating all factor matrices takes $O(N^2 J|\mathcal{G}||\Omega|/T)$. According to Algorithm 1, updating core tensor takes $O(N|\mathcal{G}||\Omega|)$. Calculating $Resp$ of factor matrices and core tensor takes $O(N^2|\mathcal{G}||\Omega|/T)$ and $O(|\Omega|/T)$ respectively. Therefore, the time complexity of VEST is $O(N^2 J|\mathcal{G}||\Omega|/T)$. □

Theorem 3. *The memory complexity of* VEST *is*

$$O(TJ + J^N + NIJ),$$

where T is the number of threads, I is the dimensionality of a mode of $\mathcal{X}$, J is the dimensionality of a mode of $\mathcal{G}$, and N is the order of $\mathcal{X}$.

Proof. VEST generates intermediate data of size $O(J)$ to update an element in a factor matrix. Using T threads, VEST requires $O(TJ)$ space while updating factor matrices.

VEST uses a marking table to indicate pruned and un-pruned elements, which requires $O(J^N+NIJ)$. While pruning, VEST save $Resp$ values which requires $O(J^N+NIJ)$. In sum, the memory complexity of VEST is $O(TJ+J^N+NIJ)$. □

4 Hyperparameter Selection

4.1 Selecting Core Sizes

In the case of $I_n \leq 200$, J_n is assigned to $I_n/10$. When $I_n > 200$, we observed the change of reconstruction error value to determine J_n. Reconstruction Error was measured by VEST$_*^{man}$ while increasing all undetermined J_n from 1 to 10. Then the point where the error value becomes minimum is determined as the rank.

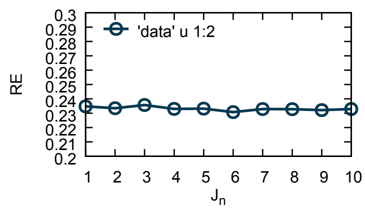


Fig. 1: Reconstruction error value of MovieLens data varying input order.

The graph above shows the change of reconstruction error value of MovieLens data. X-axis represents J_n and y-axis represent Reconstruction error. The data of the graph is the average value of the 3 times run. We note that the change value of RE is substantially small, but has a minimum value when $J_n = 6$. In this way, Rank of MovieLens data is determined as $6\times6\times2\times2$. The Rank for the remaining Real data was determined in the same way. However, Rank $5 \times 5 \times 5$ is used for the sampled-data used for comparison with the competition code. This is because the competing code is implemented only for the $J_1 = J_2 = \cdots = J_n$ case.

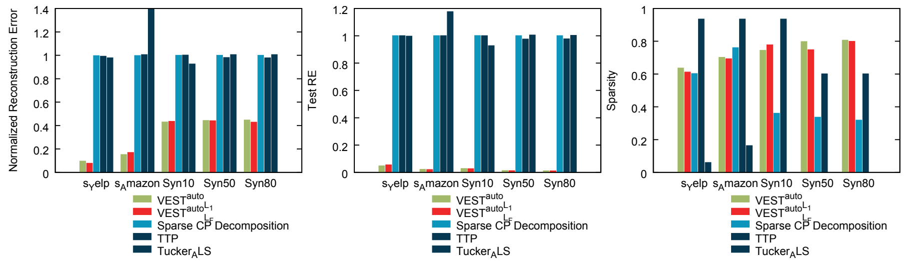
5 Additional Experimental Results

5.1 Comparative Studies

Figure 1 of the main paper is plotted based on the datasets used.

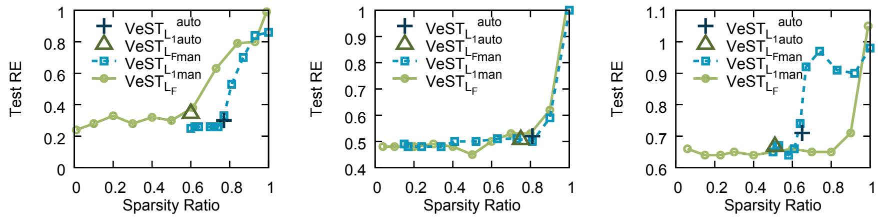
5.2 Sparsity

Figure 3 shows normalized reconstruction error values varing lambda values of VEST$_*^{man}$ on test set (test RE). Results of VEST$_{L_1}^{auto}$ and VEST$_{L_F}^{auto}$ are plotted to show the the auto mode find reasonable sparsity at the elbow point of the curve.



(a) Reconstruction error (b) Test RMSE (c) Sparsity

Fig. 2: Accuracy of VEST compared to standard tensor factorization methods on three real world data. Measurement values are averages of five randomly initialized independent runs.

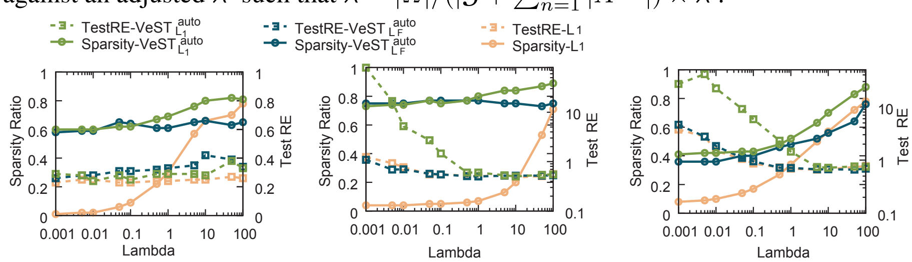


(a) MovieLens (b) Yelp (c) AmazonFood

Fig. 3: Sparsity against test RE of $\text{VEST}_{L_F}^{man}$ and $\text{VEST}_{L_1}^{man}$ with varying target sparsity s from 0 to 0.99. The values are averaged from five randomly initialized independent runs.

5.3 Effect of λ in VEST_{*}^{auto} on Performance.

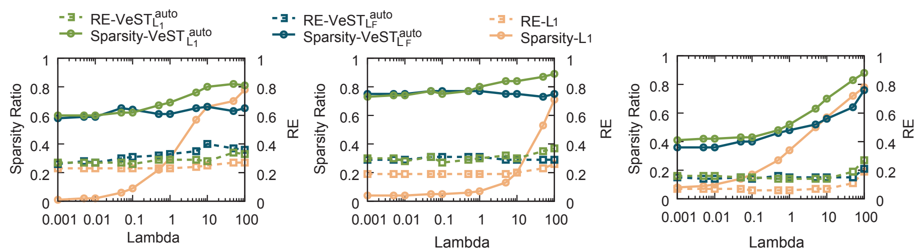
Fig. 4, shows the effect of λ values on normalized reconstruction error on the test set (Test RE) for $\text{VEST}_{L_F}^{auto}$, $\text{VEST}_{L_1}^{auto}$, and L_1, i.e., VEST optimized on L_1 regularizer without pruning. In all three datasets, $\text{VEST}_{L_1}^{auto}$ was able to generate sparse results ($s >= 0.4$) without significant difference in the test RE values. To balance the error term and λ term in the loss functions, Eq.(1) and (2) in the main paper, we plotted against an adjusted λ' such that $\lambda = |\Omega|/(|\mathcal{G} + \sum_{n=1}^{N} |A^{(n)}|) \times \lambda'$.



(a) MovieLens (b) Yelp (c) AmazonFood

Fig. 4: Sparsity and test RE of $\text{VEST}_{L_1}^{man\ p=0}$ and $\text{VEST}_{L_1}^{auto}$ with varying λ values. Measurement values are averaged from five runs.

For $\text{VEST}_{L_F}^{auto}$, sufficiently large λ' values were required to generalize on the test data set (higher test RE values at $\lambda' = 0.001$), while the training set RE values remained relatively consistent over the λ' values (Figure 5). However, sparsity and RE values after



(a) MovieLens (b) Yelp (c) AmazonFood

Fig. 5: Sparsity and RE of $\text{VEST}_{L_1}^{man\ p=0}$ and $\text{VEST}_{L_1}^{auto}$ with varying λ values. Measurement values are averaged from five runs.

$\lambda' >= 5$ remained relatively consistent. For both $\text{VEST}_{L_F}^{auto}$ and $\text{VEST}_{L_1}^{auto}$, sparsity and RE values were more consistent compared to that of L_1, where the sparsity showed high dependency on λ'. Also, as expected, if the λ' is too small, factorization did not generalize well on the test set as shown by the higher test RE values at $\lambda' = 0.001$.