Collaborative Ranking for Local Preferences Supplement

Berk Kapicioglu YP
David S. Rosenberg YP
Robert E. Schapire Princeton University
Tony Jebara Columbia University

1 Problem Formulation

Let $\mathcal{U} = \{1, \ldots, m\}$ be the set of users, let $\mathcal{V} = \{1, \ldots, n\}$ be the set of items, and let $\mathcal{T} = \{1, \ldots, T\}$ indicate the local time. Then, the sample space is defined as

$$\mathcal{X} = \{(u, C, i, t) \mid u \in \mathcal{U}, C \subseteq \mathcal{V}, i \in C, t \in \mathcal{T}\}. \quad (1)$$

Let $P[\cdot]$ denote probability, let C^i be the set C excluding element i, and let $c \overset{U}{\sim} C$ mean that c is sampled uniformly from C. Then, the *local ranking loss* associated with hypothesis g is

$$L_g(u, C, i, t) = \underset{c \overset{U}{\sim} C^i}{P} [g(u, i, t) - g(u, c, t) \leq 0]. \quad (2)$$

2 A Bound on the Generalization Error

We assume that the hypothesis class is based on the set of low-rank matrices. Given a low-rank matrix M, let $g_M \in \mathcal{F}$ be the associated hypothesis, where $g_M(u, i) = M_{u,i}$. Throughout the paper, we abuse notation and use g_M and M interchangeably. We assume that data is generated with respect to $\mathcal{D}$, which is an unknown probability distribution over the sample space $\mathcal{X}$, and we let $\mathbb{E}$ denote expectation. Then, the generalization error of hypothesis M is $\underset{(u,C,i)\sim\mathcal{D}}{\mathbb{E}} L_M(u, C, i)$, which is the quantity we bound below.

We will derive the generalization bound in two steps. In the first step, we will bound the empirical Rademacher complexity of our loss class, defined below, with respect to samples that contain exactly 2 candidates, and in the second step, we will prove the generalization bound with a reduction to the previous step.

Lemma 1. *Let m be the number of users and let n be the number of items. Define $\mathcal{L}_r = \{L_M \mid M \in \mathbb{R}^{m \times n}$ has rank at most $r\}$ as the class of loss functions associated with low-rank matrices. Assume that $S_2 \subseteq \mathcal{X}$ is a set of d samples, where each sample contains exactly 2 candidate items; i.e. if $(u, C, i) \in S_2$, then $|C| = 2$. Let $R_{S_2}(\mathcal{L}_r)$ denote the Rademacher complexity of $\mathcal{L}_r$ with respect to S_2. Then,*

$$R_{S_2}(\mathcal{L}_r) \leq \sqrt{\frac{2r(m+n)\ln\left(\frac{16emn^2}{r(m+n)}\right)}{d}}.$$

Proof. Because each sample in S_2 contains exactly 2 candidates, any hypothesis $L_M \in \mathcal{L}_r$ applied to a sample in S_2 outputs either 0 or 1. Thus, the set of dichotomies that are realized by $\mathcal{L}_r$ on S_2, called $\Pi_{\mathcal{L}_r}(S_2)$, is well-defined. Using Equation (6) from Boucheron et al. [1], we know that $R_{S_2}(\mathcal{L}_r) \leq \sqrt{\frac{2\ln|\Pi_{\mathcal{L}_r}(S_2)|}{d}}$. Let $\mathcal{X}_2 \subseteq \mathcal{X}$ be the set of all samples that contain exactly 2 candidates, $|\Pi_{\mathcal{L}_r}(S_2)| \leq |\Pi_{\mathcal{L}_r}(\mathcal{X}_2)|$, so it suffices to bound $|\Pi_{\mathcal{L}_r}(\mathcal{X}_2)|$.

We bound $|\Pi_{\mathcal{L}_r}(\mathcal{X}_2)|$ by counting the sign configurations of polynomials using proof techniques that are influenced by Srebro et al. [4]. Let $(u, \{i, j\}, i) \in \mathcal{X}_2$ be a sample and let M be a hypothesis matrix. Because M has rank at most r, it can be written as $M = UV^T$, where $U \in \mathbb{R}^{m \times r}$ and $V \in \mathbb{R}^{n \times r}$. Let $[\![\cdot]\!]$ denote an indicator function that is 1 if and only if its argument is true. Then, the loss on the sample can also be rewritten as $L_M(u, \{i, j\}, i) = [\![M_{u,i} - M_{u,j} \leq 0]\!] = [\![(UV^T)_{u,i} - (UV^T)_{u,j} \leq 0]\!] = [\![\sum_{a=1}^{r} U_{u,a}(V_{i,a} - V_{j,a}) \leq 0]\!]$. Since cardinality of $\mathcal{X}_2$ is at most $2m\binom{n}{2} \leq mn^2$, putting it all together, it follows that $|\Pi_{\mathcal{L}_r}(\mathcal{X}_2)|$ is bounded by the number of sign configurations of mn^2 polynomials, each of degree at most 2, over $r(m+n)$ variables. Applying Corollary 3 from Srebro et al. [4], we obtain $|\Pi_{\mathcal{L}_r}(\mathcal{X}_2)| \leq \left(\frac{16emn^2}{r(m+n)}\right)^{r(m+n)}$. Taking logarithms and making basic substitutions yield the desired result. □

We proceed to proving the more general result via a reduction to Lemma 1.

Theorem 1. *Let m be the number of users and let n be the number of items. Assume that S consists of d*

independently and identically distributed samples chosen from $\mathcal{X}$ with respect to a probability distribution $\mathcal{D}$. Let L_M be the loss function associated with a matrix M, as defined in Equation 2. Then, with probability at least $1-\delta$, for any matrix $M \in \mathbb{R}^{m\times n}$ with rank at most r,

$$\mathop{\mathbb{E}}_{(u,C,i)\sim\mathcal{D}} L_M(u,C,i) \leq \mathop{\mathbb{E}}_{(u,C,i)\overset{U}{\sim}S} L_M(u,C,i) + 2\sqrt{\frac{2r(m+n)\ln\left(\frac{16emn}{r}\right)}{d}} + \sqrt{\frac{2\ln\left(\frac{2}{\delta}\right)}{d}}. \quad (3)$$

Proof. We will manipulate the definition of Rademacher complexity [1] in order to use the bound given in Lemma 1:

$$\begin{aligned}
R_S(\mathcal{L}_r) &\doteq \mathop{\mathbb{E}}_{\sigma}\left[\sup_{L_M\in\mathcal{L}_r}\left(\frac{1}{d}\sum_{a=1}^{d}\sigma_a L_M(u_a,C_a,i_a)\right)\right] \\
&= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{L_M\in\mathcal{L}_r}\left(\frac{1}{d}\sum_{a=1}^{d}\sigma_a \mathop{\mathbb{E}}_{j_a\overset{U}{\sim}(C_a\setminus\{i_a\})} L_M(u_a,\{i_a,j_a\},i_a)\right)\right] \\
&= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{L_M\in\mathcal{L}_r}\left(\mathop{\mathbb{E}}_{j_1,\ldots,j_d}\frac{1}{d}\sum_{a=1}^{d}\sigma_a L_M(u_a,\{i_a,j_a\},i_a)\right)\right] \\
&\leq \mathop{\mathbb{E}}_{\sigma}\left[\mathop{\mathbb{E}}_{j_1,\ldots,j_d}\left(\sup_{L_M\in\mathcal{L}_r}\frac{1}{d}\sum_{a=1}^{d}\sigma_a L_M(u_a,\{i_a,j_a\},i_a)\right)\right] \\
&= \mathop{\mathbb{E}}_{j_1,\ldots,j_d}\left[\mathop{\mathbb{E}}_{\sigma}\left(\sup_{L_M\in\mathcal{L}_r}\frac{1}{d}\sum_{a=1}^{d}\sigma_a L_M(u_a,\{i_a,j_a\},i_a)\right)\right] \\
&= \mathop{\mathbb{E}}_{j_1,\ldots,j_d}\left[R_{S_2}(\mathcal{L}_r)\right] \\
&\leq \sqrt{\frac{2r(m+n)\ln\left(\frac{16emn^2}{r(m+n)}\right)}{d}}.
\end{aligned}$$

Plugging the bound to Theorem 3.2 in Boucheron et al. [1] proves the theorem. □

3 Collaborative Local Ranking

Let $h(x) = \max(0, 1-x)$ be the hinge function, let M be the hypothesis matrix with rank at most r, and let $M = UV^T$, where $U \in \mathbb{R}^{m\times r}$ and $V \in \mathbb{R}^{n\times r}$. Then, we can bound the empirical local ranking loss as

$$\begin{aligned}
&\mathop{\mathbb{E}}_{(u,C,i)\overset{U}{\sim}S} L_M(u,C,i) = \frac{1}{|S|}\sum_{(u,C,i)\in S} L_M(u,C,i) \\
&= \frac{1}{|S|}\sum_{(u,C,i)\in S}\mathop{P}_{c\overset{U}{\sim}C^i}\left[M_{u,i} - M_{u,c} \leq 0\right] \\
&= \frac{1}{|S|}\sum_{(u,C,i)\in S}\mathop{\mathbb{E}}_{c\overset{U}{\sim}C^i}\left[\!\left[M_{u,i} - M_{u,c} \leq 0\right]\!\right] \\
&= \frac{1}{|S|}\sum_{(u,C,i)\in S}\frac{1}{|C^i|}\sum_{c\in C^i}\left[\!\left[\left(UV^T\right)_{u,i} - \left(UV^T\right)_{u,c} \leq 0\right]\!\right] \\
&\leq \frac{1}{|S|}\sum_{(u,C,i)\in S}\frac{1}{|C^i|}\sum_{c\in C^i} h\left(\left(UV^T\right)_{u,i} - \left(UV^T\right)_{u,c}\right).
\end{aligned} \quad (4)$$

We note that the CLR and the ranking SVM [2] objectives are closely related. If V is fixed and we only need to minimize U, then each row of V acts as a feature vector for the corresponding item, each row of U acts as a separate linear predictor, and the CLR objective decomposes into solving simultaneous ranking SVM problems. In particular, let $S_u = \{(a,C,i) \in S \mid a = u\}$ be the examples that correspond to user u, let U_u denote row u of U, and let f^{rSVM} denote the objective function of ranking SVM, then

$$\begin{aligned}
f^{\text{CLR}}(S;U,V) &= \frac{\lambda}{2}\|U\|_F^2 \\
&+ \frac{1}{|S|}\sum_{(u,C,i)\in S}\frac{1}{|C^i|}\sum_{c\in C^i} h\left(\left(UV^T\right)_{u,i} - \left(UV^T\right)_{u,c}\right) \\
&= \sum_{u=1}^{m}\frac{\lambda}{2}\|U_u\|_F^2 \\
&+ \sum_{u=1}^{m}\frac{1}{|S|}\sum_{(u,C,i)\in S_u}\frac{1}{|C^i|}\sum_{c\in C^i} h\left(\left(UV^T\right)_{u,i} - \left(UV^T\right)_{u,c}\right) \\
&= \sum_{u=1}^{m} f^{\text{rSVM}}(S_u;U_u,V).
\end{aligned}$$

4 Algorithms

4.1 Derivation

Let $(u,C,i) \in S$ be an example, then the corresponding approximate objective function is

$$f^{\text{CLR}}((u,C,i);U,V) = \frac{\lambda}{2}\|V\|_F^2 + \frac{1}{|C^i|}\sum_{c\in C^i} h\left(\left(UV^T\right)_{u,i} - \left(UV^T\right)_{u,c}\right).$$

We introduce various matrix notation to help us define the approximate subgradients. Given a matrix M, let

Algorithm 1 Alternating minimization for optimizing the CLR objective.

Input: Training data $S \subseteq \mathcal{X}$, regularization parameter $\lambda > 0$, rank constraint r, number of iterations T.

$U_1 \leftarrow$ Sample matrix uniformly at random from $\left[-\frac{1}{\sqrt{\lambda m r}}, \frac{1}{\sqrt{\lambda m r}}\right]^{m \times r}$.
$V_1 \leftarrow$ Sample matrix uniformly at random from $\left[-\frac{1}{\sqrt{\lambda n r}}, \frac{1}{\sqrt{\lambda n r}}\right]^{n \times r}$.
for all t from 1 to $T-1$ **do**
  $U_{t+1} \leftarrow \arg\min_U f^{\mathrm{CLR}}(S; U, V_t)$
  $V_{t+1} \leftarrow \arg\min_V f^{\mathrm{CLR}}(S; U_{t+1}, V)$
return U_T, V_T.

$M_{k,\cdot}$ denote row k of M. Define the matrix $\hat{M}^{p,q,z}$, for $p \neq q$, as

$$\hat{M}^{p,q,z}_{s,\cdot} = \begin{cases} M_{z,\cdot} & \text{for } s = p, \\ -M_{z,\cdot} & \text{for } s = q, \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

and define the matrix $\check{M}^{p,q,z}_{s,\cdot}$ as

$$\check{M}^{p,q,z}_{s,\cdot} = \begin{cases} M_{p,\cdot} - M_{q,\cdot} & \text{for } s = z, \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

Let $[\![\cdot]\!]$ denote an indicator function that is 1 if and only if its argument is true. Then, the subgradient of the approximate objective function with respect to V is

$$\nabla_V f^{\mathrm{CLR}}((u, C, i); U, V) = \lambda V - \frac{1}{|C^i|} \sum_{c \in C^i} [\![\left(UV^T\right)_{u,i} - \left(UV^T\right)_{u,c} < 1]\!] \hat{U}^{i,c,u}. \tag{7}$$

Setting $\eta_t = \frac{1}{\lambda t}$ as the learning rate at iteration t, the approximate subgradient update becomes $V_{t+1} = V_t - \eta_t \nabla_V f^{\mathrm{CLR}}((u, C, i); U, V)$. After the update, the weights are projected onto a ball with radius $\frac{1}{\sqrt{\lambda}}$. The pseudocode for optimizing both convex subproblems is depicted in Algorithms 2 and 3. We prove the correctness of the algorithms and bound their running time in the next subsection.

4.2 Analysis

The convex subproblems we analyze have the general form

$$\min_{X \in D} f(X; \ell) = \min_{X \in D} \frac{\lambda}{2} \|X\|_F^2 + \frac{1}{|S|} \sum_{(u,C,i) \in S} \ell(X; (u, C, i)). \tag{8}$$

Algorithm 2 Projected stochastic subgradient descent for optimizing U.

Input: Factors $V \in \mathbb{R}^{n \times r}$, training data S, regularization parameter λ, rank constraint r, number of iterations T.

$U_1 \leftarrow 0^{m \times r}$
for all t from 1 to $T-1$ **do**
  Choose $(u, C, i) \in S$ uniformly at random.
  $\eta_t \leftarrow \frac{1}{\lambda t}$
  $C^+ \leftarrow \left\{ c \in C^i \mid \left(U_t V^T\right)_{u,i} - \left(U_t V^T\right)_{u,c} < 1 \right\}$
  $U_{t+1} \leftarrow (1 - \eta_t \lambda) U_t + \frac{\eta_t}{|C^i|} \sum_{c \in C^+} \check{V}^{i,c,u}$
  $U_{t+1} \leftarrow \min\left\{1, \frac{1}{\sqrt{\lambda}\|U_{t+1}\|_F}\right\} U_{t+1}$
return U_T.

Algorithm 3 Projected stochastic subgradient descent for optimizing V.

Input: Factors $U \in \mathbb{R}^{m \times r}$, training data S, regularization parameter λ, rank constraint r, number of iterations T.

$V_1 \leftarrow 0^{n \times r}$
for all t from 1 to $T-1$ **do**
  Choose $(u, C, i) \in S$ uniformly at random.
  $\eta_t \leftarrow \frac{1}{\lambda t}$
  $C^+ \leftarrow \left\{ c \in C^i \mid \left(U V_t^T\right)_{u,i} - \left(U V_t^T\right)_{u,c} < 1 \right\}$
  $V_{t+1} \leftarrow (1 - \eta_t \lambda) V_t + \frac{\eta_t}{|C^i|} \sum_{c \in C^+} \hat{U}^{i,c,u}$
  $V_{t+1} \leftarrow \min\left\{1, \frac{1}{\sqrt{\lambda}\|V_{t+1}\|_F}\right\} V_{t+1}$
return V_T.

One can obtain the individual subproblems by specifying the domain D and the loss function ℓ. For example, in case of Algorithm 2, the corresponding minimization problem is specified by

$$\min_{X \in \mathbb{R}^{m \times r}} f(X; \ell_V) \text{ where}$$
$$\ell_V(X; (u, C, i)) = \frac{1}{|C^i|} \sum_{c \in C^i} h\left(\left(XV^T\right)_{u,i} - \left(XV^T\right)_{u,c}\right), \tag{9}$$

and in case of Algorithm 3, it is specified by

$$\min_{X \in \mathbb{R}^{n \times r}} f(X; \ell_U) \text{ where}$$
$$\ell_U(X; (u, C, i)) = \frac{1}{|C^i|} \sum_{c \in C^i} h\left(\left(UX^T\right)_{u,i} - \left(UX^T\right)_{u,c}\right). \tag{10}$$

Let $U^\star = \arg\min_U f(U; \ell_V)$ and $V^\star = \arg\min_V f(V; \ell_U)$ denote the solution matrices of Equations 9 and 10, respectively. Also, given a general convex loss ℓ and domain D, let $\bar{X} \in D$ be an

ϵ-accurate solution for the corresponding minimization problem if $f\left(\bar{X};\ell\right) \leq \min_{X\in D} f\left(X;\ell\right)+\epsilon$.

In the remainder of this subsection, we show that Algorithms 2 and 3 are adaptations of the Pegasos [3] algorithm to the CLR setting. Then, we prove certain properties that are prerequisites for obtaining Pegasos's performance guarantees. In particular, we show that the approximate subgradients computed by Algorithms 2 and 3 are bounded and the loss functions associated with Equations 9 and 10 are convex. In the end, we plug these properties into a theorem proved by Shalev-Shwartz et al. [3] to show that our algorithms reach an ϵ-accurate solution with respect to their corresponding minimization problems in $\tilde{O}\left(\frac{1}{\lambda^2\epsilon}\right)$ iterations.

Lemma 2. *$\|U^\star\| \leq \frac{1}{\sqrt{\lambda}}$ and $\|V^\star\| \leq \frac{1}{\sqrt{\lambda}}$.*

Proof. One can obtain the bounds on the norms of the optimal solutions by examining the dual form of the optimization problems and applying the strong duality theorem. Equations 9 and 10 can both be represented as

$$\min_{v\in D} \frac{1}{2}\|v\|^2 + \sum_{k=1}^{K} e_k h\left(f_k\left(v\right)\right), \tag{11}$$

where $e_k = \frac{1}{\lambda|S||C_k|}$ is a constant, h is the hinge function, D is a Euclidean space, and f_k is a linear function. We rewrite Equation 11 as a constrained optimization problem

$$\begin{aligned} \min_{v\in D,\xi\in\mathbb{R}^K} \quad & \frac{1}{2}\|v\|^2 + \sum_{k=1}^{K} e_k\xi_k && (12)\\ \text{subject to} \quad & \xi_k \geq 1 - f_k\left(v\right), && k=1,\dots K,\\ & \xi_k \geq 0, && k=1,\dots K. \end{aligned}$$

The Lagrangian of this problem is

$$\begin{aligned} L\left(v,\xi,\alpha,\beta\right) &= \frac{1}{2}\|v\|^2 + \sum_{k=1}^{K} e_k\xi_k \\ &\quad + \sum_{k=1}^{K}\alpha_k\left(1-f_k\left(v\right)-\xi_k\right) - \sum_{k=1}^{K}\beta_k\xi_k \\ &= \frac{1}{2}\|v\|^2 + \sum_{k=1}^{K}\xi_k\left(e_k-\alpha_k-\beta_k\right) \\ &\quad + \sum_{k=1}^{K}\alpha_k\left(1-f_k\left(v\right)\right), \end{aligned}$$

and its dual function is

$$g\left(\alpha,\beta\right) = \inf_{v,\xi} L\left(v,\xi,\alpha,\beta\right).$$

Since $L\left(v,\xi,\alpha,\beta\right)$ is convex and differentiable with respect to v and ξ, the necessary and sufficient conditions for minimizing v and ξ are

$$\begin{aligned} \nabla_v L = 0 \quad &\Leftrightarrow \quad v = \sum_{k=1}^{K}\alpha_k\nabla_v f_k\left(v\right), \\ \nabla_\xi L = 0 \quad &\Leftrightarrow \quad e = \alpha+\beta. \end{aligned} \tag{13}$$

We plug these conditions back into the dual function and obtain

$$\begin{aligned} g\left(\alpha,\beta\right) &= \inf_{v,\xi} L\left(v,\xi,\alpha,\beta\right) \\ &= \frac{1}{2}\left\|\sum_{k=1}^{K}\alpha_k\nabla_v f_k\left(v\right)\right\|^2 \\ &\quad + \sum_{k=1}^{K}\alpha_k\left(1-f_k\left(\sum_{k=1}^{K}\alpha_k\nabla_v f_k\left(v\right)\right)\right) \\ &= \frac{1}{2}\left\|\sum_{k=1}^{K}\alpha_k\nabla_v f_k\left(v\right)\right\|^2 + \sum_{k=1}^{K}\alpha_k \\ &\quad - \sum_{k=1}^{K}\alpha_k f_k\left(\sum_{k=1}^{K}\alpha_k\nabla_v f_k\left(v\right)\right). \end{aligned} \tag{14}$$

Since f_k is a linear function, we let $f_k\left(v\right) = \vec{k}\cdot v$, where $\vec{k}$ is a constant vector, and $\nabla_v f_k\left(v\right) = \vec{k}$. Then,

$$\begin{aligned} \left\|\sum_{k=1}^{K}\alpha_k\nabla_v f_k\left(v\right)\right\|^2 &= \left\|\sum_{k=1}^{K}\alpha_k\vec{k}\right\|^2 \\ &= \left(\sum_{k=1}^{K}\alpha_k\vec{k}\right)\cdot\left(\sum_{k=1}^{K}\alpha_k\vec{k}\right) \\ &= \sum_{k=1}^{K}\alpha_k\vec{k}\cdot\left(\sum_{k=1}^{K}\alpha_k\vec{k}\right) \\ &= \sum_{k=1}^{K}\alpha_k f_k\left(\sum_{k=1}^{K}\alpha_k\vec{k}\right) \\ &= \sum_{k=1}^{K}\alpha_k f_k\left(\sum_{k=1}^{K}\alpha_k\nabla_v f_k\left(v\right)\right). \end{aligned} \tag{15}$$

Simplifying Equation 14 using Equation 15 yields

$$\begin{aligned} g\left(\alpha,\beta\right) &= -\frac{1}{2}\left\|\sum_{k=1}^{K}\alpha_k\nabla_v f_k\left(v\right)\right\|^2 + \sum_{k=1}^{K}\alpha_k \\ &= -\frac{1}{2}\left\|\sum_{k=1}^{K}\alpha_k\vec{k}\right\|^2 + \sum_{k=1}^{K}\alpha_k. \end{aligned} \tag{16}$$

Finally, we combine Equations 13 and 16, and obtain the dual form of Equation 12,

$$\begin{aligned} \max_{\alpha} \quad & -\frac{1}{2}\left\|\sum_{k=1}^{K}\alpha_k\vec{k}\right\|^2 + \sum_{k=1}^{K}\alpha_k && (17)\\ \text{subject to} \quad & 0 \leq \alpha_k \leq e_k, && k=1,\dots K. \end{aligned}$$

The primal problem is convex, its constraints are linear, and the domain of its objective is open; thus, Slater's condition holds and strong duality is obtained. Furthermore, the primal problem has differentiable objective and constraint functions, which implies that $(v^\star, \xi^\star)$ is primal optimal and $(\alpha^\star, \beta^\star)$ is dual optimal if and only if these points satisfy the Karush-Kuhn-Tucker (KKT) conditions. It follows that

$$v^\star = \sum_{k=1}^{K} \alpha_k^\star \vec{k}. \tag{18}$$

Note that we defined $e_k = \frac{1}{\lambda |S| |C_k|}$, where $\sum_{k=1}^{K} e_k = \frac{1}{\lambda}$, and the constraints of the dual problem imply $0 \leq \alpha_k \leq e_k$; thus, $\sum_{k=1}^{K} \alpha_k^\star \leq \frac{1}{\lambda}$. Because of strong duality, there is no duality gap, and the primal and dual objectives are equal at the optimum,

$$\begin{aligned}
\frac{1}{2}\|v^\star\|^2 + \sum_{k=1}^{K} e_k \xi_k^\star &= -\frac{1}{2}\left\|\sum_{k=1}^{K} \alpha_k^\star \vec{k}\right\|^2 + \sum_{k=1}^{K} \alpha_k^\star \\
&= -\frac{1}{2}\|v^\star\|^2 + \sum_{k=1}^{K} \alpha_k^\star \qquad \text{(by (18))} \\
&\leq -\frac{1}{2}\|v^\star\|^2 + \frac{1}{\lambda} \\
\Rightarrow \|v^\star\|^2 &\leq \frac{1}{\lambda}.
\end{aligned}$$

This proves the lemma. □

Given the bounds in Lemma 2, it can be verified that Algorithms 2 and 3 are adaptations of the Pegasos [3] algorithm for optimizing Equations 9 and 10, respectively. It still remains to show that Pegasos's performance guarantees hold in our case.

Lemma 3. *In Algorithms 2 and 3, the approximate subgradients have norm at most* $\sqrt{\lambda} + 2\sqrt{\frac{1}{\lambda}}$.

Proof. The approximate subgradient for Algorithm 3 is depicted in Equation 7. Due to the projection step, $\|V\|_F \leq \frac{1}{\sqrt{\lambda}}$, and it follows that $\|\lambda V\|_F \leq \sqrt{\lambda}$. The term $\hat{U}^{i,c,u}$ is constructed using Equation 5, and it can be verified that $\left\|\hat{U}^{i,c,u}\right\|_F \leq \sqrt{2}\|U\|_F \leq \sqrt{\frac{2}{\lambda}}$. Using triangle inequality, one can bound Equation 7 with $\sqrt{\lambda} + \sqrt{\frac{2}{\lambda}}$. A similar argument can be made for the approximate subgradient of Algorithm 2, yielding the slightly higher upper bound given in the lemma statement. □

We combine the lemmas to obtain the correctness and running time guarantees for our algorithms.

Lemma 4. *Let* $\lambda \leq \frac{1}{4}$*, let* T *be the total number of iterations of Algorithm 2, and let* U_t *denote the parameter computed by the algorithm at iteration* t*. Let* $\bar{U} = \frac{1}{T}\sum_{t=1}^{T} U_t$ *denote the average of the parameters produced by the algorithm. Then, with probability at least* $1 - \delta$*,*

$$f\left(\bar{U}; \ell_V\right) \leq f\left(U^\star; \ell_V\right) + \frac{21\left(\sqrt{\lambda} + 2\sqrt{\frac{1}{\lambda}}\right)^2 \ln\left(\frac{T}{\delta}\right)}{\lambda T}.$$

The analogous result holds for Algorithm 3 as well.

Proof. First, for each loss function ℓ_V and ℓ_U, variables are linearly combined, composed with the convex hinge function, and then averaged. All these operations preserve convexity, hence both loss functions are convex. Second, we have argued above that Algorithms 2 and 3 are adaptations of the Pegasos [3] algorithm for optimizing Equations 9 and 10, respectively. Third, in Lemma 3, we proved a bound on the approximate subgradients of both algorithms. Plugging these three results into Corollary 2 in Shalev-Shwartz et al. [3] yields the statement of the theorem. □

The theorem below gives a bound in terms of individual parameters rather than average parameters.

Theorem 2. *Assume that the conditions and the bound in Lemma 4 hold. Let* t *be an iteration index selected uniformly at random from* $\{1, \ldots, T\}$*. Then, with probability at least* $\frac{1}{2}$*,*

$$f\left(U_t; \ell_V\right) \leq f\left(U^\star; \ell_V\right) + \frac{42\left(\sqrt{\lambda} + 2\sqrt{\frac{1}{\lambda}}\right)^2 \ln\left(\frac{T}{\delta}\right)}{\lambda T}.$$

The analogous result holds for Algorithm 3 as well.

Proof. The result follows directly from combining Lemma 4 with Lemma 3 in Shalev-Shwartz et al. [3]. □

Thus, with high probability, our algorithms reach an ϵ-accurate solution in $\tilde{O}\left(\frac{1}{\lambda^2 \epsilon}\right)$ iterations. Since we argued in Subsection 4.1 that the running time of each stochastic update is $O(br)$, it follows that a complete run of projected stochastic subgradient descent takes $\tilde{O}\left(\frac{br}{\lambda^2 \epsilon}\right)$ time, and the running time is *independent* of the size of the training data.

References

[1] Stéphane Boucheron, Olivier Bousquet, and Gabor Lugosi. Theory of classification : A survey of some recent advances. *ESAIM: Probability and Statistics*, 9:323–375, 2005.

[2] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 133–142, New York, NY, USA, 2002. ACM.

[3] Shai S. Shwartz, Yoram Singer, Nathan Srebro, and Andrew Cotter. Pegasos: primal estimated subgradient solver for SVM. *Math. Program.*, 127(1):3–30, March 2011.

[4] Nathan Srebro, Noga Alon, and Tommi Jaakkola. Generalization error bounds for collaborative prediction with Low-Rank matrices. In *NIPS*, 2004.