

---

# Online Matrix Completion with Side Information

---

Mark Herbster, Stephen Pasteris, Lisa Tse

Department of Computer Science

University College London

London WC1E 6BT, England, UK

{m.herbster, s.pasteris, l.tse}@cs.ucl.ac.uk

## Abstract

We give an online algorithm and prove novel mistake and regret bounds for online binary matrix completion with side information. The mistake bounds we prove are of the form  $\tilde{O}(\frac{\mathcal{D}}{\gamma^2})$ . The term  $\frac{1}{\gamma^2}$  is analogous to the usual margin term in SVM (perceptron) bounds. More specifically, if we assume that there is some factorization of the underlying  $m \times n$  matrix into  $PQ^\top$ , where the rows of  $P$  are interpreted as “classifiers” in  $\mathbb{R}^d$  and the rows of  $Q$  as “instances” in  $\mathbb{R}^d$ , then  $\gamma$  is the maximum (normalized) margin over all factorizations  $PQ^\top$  consistent with the observed matrix. The quasi-dimension term  $\mathcal{D}$  measures the quality of side information. In the presence of vacuous side information,  $\mathcal{D} = m + n$ . However, if the side information is predictive of the underlying factorization of the matrix, then in an ideal case,  $\mathcal{D} \in \mathcal{O}(k + \ell)$  where  $k$  is the number of distinct row factors and  $\ell$  is the number of distinct column factors. We additionally provide a generalization of our algorithm to the inductive setting. In this setting, we provide an example where the side information is not directly specified in advance. For this example, the quasi-dimension  $\mathcal{D}$  is now bounded by  $\mathcal{O}(k^2 + \ell^2)$ .

## 1 Introduction

We consider the problem of online binary matrix completion with *side information*. In our setting the *learner* receives data sequentially, so that on a trial  $t = 1, \dots, T$ : 1) the learner is queried by the *environment* to predict matrix entry  $(i_t, j_t)$ ; 2) the learner predicts a label  $\hat{y}_t \in \{-1, 1\}$ ; 3) the learner receives a label  $y_t \in \{-1, 1\}$  from the environment and 4) a mistake is incurred if  $y_t \neq \hat{y}_t$ . There are no probabilistic assumptions on how the environment generates its instances or their labels; it is an arbitrary process which in fact may be adversarial. The only restriction on the environment is that it does not “see” the learner’s  $\hat{y}_t$  until after it reveals  $y_t$ . The learner’s aim will be to minimize its *expected regret*,  $\sum_{t=1}^T \mathbb{E}[y_t \neq \hat{y}_t] - \min_U \sum_{t=1}^T [y_t \neq U_{i_t j_t}]$ , where the minimization is over  $U \in \{-1, 1\}^{m \times n}$  and the expectation is with respect to the learner’s internal randomization. We will also consider *mistake bounds* where the aim is to minimize the learner’s mistakes under the assumption (*realizability*) that there exists a  $U$  such that  $U_{i_t j_t} = y_t$  for all  $t \in [T]$ . To aid the learner, side information is associated with each row and column. For instance, in the classic “Netflix challenge” [1], the rows of the matrix correspond to viewers and the columns to movies, with entries representing movie ratings. It is natural to suppose that we have side information in the form of demographic information for each user, and metadata for the movies. In this work, we consider both *transductive* and *inductive* models. In the former model, the side information associated with each row and column is specified completely in advance in the form of a pair of positive definite matrices that inform similarity between row pairs and column pairs. For the inductive model, a pair of kernel functions is specified over potentially continuous domains, where one is for the rows and the other is for the columns. What is not specified is the mapping from the domain of the kernel function to

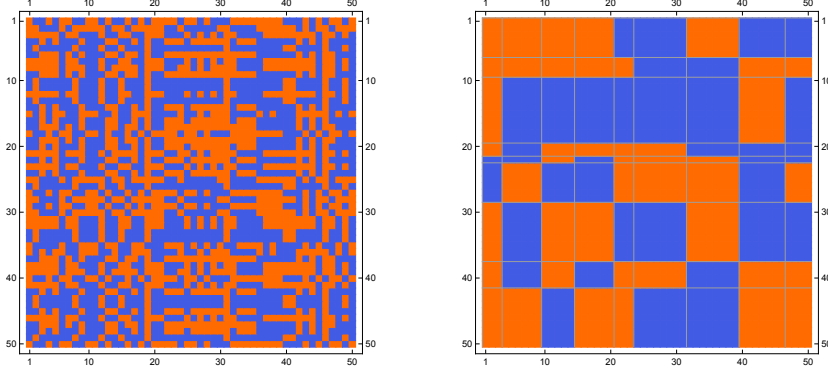


Figure 1: A (9, 9)-biclustered  $50 \times 50$  binary matrix before/after permuting into latent blocks.

specific rows or columns, which is only revealed sequentially. In the Netflix example, the inductive model is especially natural if new users or movies are introduced during the learning process.

In Theorem 1, we will give regret and mistake bounds for online binary matrix completion with side information. Although this theorem has a broader applicability, our interpretation will focus on the case that the matrix has a *latent block structure*. Hartigan [2] introduced the idea of permuting a matrix by both the rows and columns into a few homogeneous blocks. This is equivalent to assuming that each row (column) has an associated row (column) class, and that the matrix entry is completely determined by its corresponding row and column classes. This has since become known as co- or bi-clustering. This same assumption has become the basis for probabilistic models which can then be used to “complete” a matrix with missing entries. The authors of [3] give rate-optimal results for this problem in the batch setting and provide a literature overview. It is natural to compare this assumption to the dominant alternative, which assumes that there exists a low rank decomposition of the matrix to be completed, see for instance [4]. Common to both approaches is that associated with each row and column, there is an underlying *latent* factor so that the given matrix entry is determined by a function on the appropriate row and column factor. The low-rank assumption is that the latent factors are vectors in  $\mathbb{R}^d$  and that the function is the dot product. The latent block structure assumption is that the latent factors are instead categorical and that the function between factors is arbitrary. In this work we prove mistake bounds of the form  $\tilde{\mathcal{O}}(\mathcal{D}/\gamma^2)$ . The term  $1/\gamma^2$  is a parameter of our algorithm which serves as upper estimate of the squared margin complexity  $\text{mc}(\mathbf{U})^2$  of the comparator matrix  $\mathbf{U}$ . The notion of margin complexity in machine learning was introduced in [5], where it was used to study the learnability of concept classes via linear embeddings. It was further studied in [6], and in [7] a detailed study of margin complexity, trace complexity and rank in the context of statistical bounds for matrix completion was given. The squared margin complexity is upper bounded by rank. Furthermore, if our  $m \times n$  matrix has a latent block structure with  $k \times \ell$  homogeneous blocks (for an illustration, see Figure 1), then  $\text{mc}(\mathbf{U})^2 \leq \min(k, \ell)$ . The second term in our bound is the quasi-dimension  $\mathcal{D}$  which, to the best of our knowledge, is novel to this work. The quasi-dimension measures the extent to which the side information is “predictive” of the comparator matrix. In Theorem 3, we provide an upper bound on the quasi-dimension, which measures the predictiveness of the side information when the comparator matrix has a latent block structure. If there is only vacuous side information, then  $\mathcal{D} = m + n$ . However, if there is a  $k \times \ell$  latent block structure and the side information is predictive, then  $\mathcal{D} \in \mathcal{O}(k + \ell)$ ; hence our nomenclature “quasi-dimension.” In this case, we then have that the mistake bound term  $\mathcal{D}/\gamma^2 \in \mathcal{O}(k\ell)$ , which we will later argue is optimal. Although latent block structure may appear to be a “fragile” measure of matrix complexity, our regret bound implies that performance will scale smoothly in the case of adversarial noise.

The paper is organized as follows. First, we discuss related literature. We then introduce preliminary concepts in Section 2. In Section 3, we present our online matrix completion algorithm as well as a theorem (Theorem 1) that characterizes its performance in the transductive setting. In Section 4, we formally introduce the concept of latent block structure (Definition 2) and provide an upper bound (Theorem 3) for the quasi-dimension  $\mathcal{D}$  when the matrix has latent block structure. We then provide an example that bounds  $\mathcal{D}$  when we have graph-based side information (Section 4.1); and a further example (Section 4.2) when the matrix has additionally a “community” structure. Finally, in Section 5, we present an algorithm for the inductive setting, as well as an example illustrating a

bound on  $\mathcal{D}$  when the side information comes as vectors in  $\mathbb{R}^d$  which are separated by a clustering via hyper-rectangles. Proofs as well as an experiment on synthetic data are contained in the appendices.

## Related literature

Matrix completion has been studied extensively in the batch setting, see for example [8; 9; 10; 11] and references therein. Central to these approaches is the aim of finding a low-rank factorization by optimizing a convex proxy to rank, such as the trace norm [12]. The papers [13; 14; 15; 16] are partially representative of methods to incorporate side-information into the matrix completion task. The inductive setting for matrix completion has been studied in [13] through the use of tensor product kernels, and [17] takes a non-convex optimization approach. Some examples in the transductive setting include [15; 16]. The last two papers use graph Laplacians to model the side information, which is similar to our approach. To achieve this, two graph Laplacians are used to define regularization functionals for both the rows and the columns so that rows (columns) with similar side information tend to have the same values. In particular, [16] resembles our approach by applying the Laplacian regularization functionals to the underlying row and column factors directly. An alternate approach is taken in [15], where the regularization is instead applied to the row space (column space) of the “surface” matrix.

In early work, the authors of [18; 19] proved mistake bounds for learning a binary relation which can be viewed as a special case of matrix completion. In the regret setting, with minimal assumptions on the loss function, the regret of the learner is bounded in terms of the *trace-norm* of the underlying comparator matrix in [20]. The authors of [21] provided tight upper and lower bounds in terms of a parameterized complexity class of matrices that include the bounded-trace-norm and bounded-max-norm matrices as special cases. None of the above references considered the problem of side information. The results in [22; 23; 24] are nearest in flavor to the results given here. In [24], a mistake bound of  $\tilde{O}((m+n)\text{mc}(\mathbf{U})^2)$  was given. Latent block structure was also introduced to the online setting in [24]; however, it was treated in a limited fashion and without the use of side information. The papers [22; 23] both used side information to predict a limited complexity class of matrices. In [22], side information was used to predict if vertices in a graph are “similar”; in Section 4.2 we show how this result can be obtained as a special case of our more general bound. In [23], a more general setting was considered, which as a special case addressed the problem of a switching graph labeling. The model in [23] is considerably more limited in its scope than our Theorem 1. To obtain our technical results, we used an adaptation of the matrix exponentiated gradient algorithm [25]. The general form of our regret bound comes from a matricization of the regret bound proven for a Winnow-inspired algorithm [26] for linear classification in the vector case given in [27]. For a more detailed discussion, see Appendix B.2.

## 2 Preliminaries

For any positive integer  $m$ , we define  $[m] := \{1, 2, \dots, m\}$ . For any predicate  $[\text{PRED}] := 1$  if  $\text{PRED}$  is true and equals 0 otherwise, and  $[x]_+ := x[x > 0]$ . We denote the inner product of vectors  $\mathbf{x}, \mathbf{w} \in \mathbb{R}^n$  as  $\langle \mathbf{x}, \mathbf{w} \rangle = \sum_{i=1}^n x_i w_i$  and the norm as  $\|\mathbf{w}\| = \sqrt{\langle \mathbf{w}, \mathbf{w} \rangle}$ . The  $i$ th coordinate  $m$ -dimensional vector is denoted  $\mathbf{e}_m^i := ([j = i])_{j \in [m]}$ ; we will often abbreviate the notation and use  $\mathbf{e}^i$  on the assumption that the dimensionality of the space may be inferred. For vectors  $\mathbf{p} \in \mathbb{R}^m$  and  $\mathbf{q} \in \mathbb{R}^n$  we define  $[\mathbf{p}; \mathbf{q}] \in \mathbb{R}^{m+n}$  to be the concatenation of  $\mathbf{p}$  and  $\mathbf{q}$ , which we regard as a column vector. Hence  $[\mathbf{p}; \mathbf{q}]^\top [\bar{\mathbf{p}}; \bar{\mathbf{q}}] = \mathbf{p}^\top \bar{\mathbf{p}} + \mathbf{q}^\top \bar{\mathbf{q}}$ . We let  $\mathbb{R}^{m \times n}$  be the set of all  $m \times n$  real-valued matrices. If  $\mathbf{X} \in \mathbb{R}^{m \times n}$  then  $\mathbf{X}_i$  denotes the  $i$ -th  $n$ -dimensional row vector and the  $(i, j)^{\text{th}}$  entry of  $\mathbf{X}$  is  $X_{ij}$ . We define  $\mathbf{X}^+$  and  $\mathbf{X}^\top$  to be its pseudoinverse and transpose, respectively. The trace norm of a matrix  $\mathbf{X} \in \mathbb{R}^{m \times n}$  is  $\|\mathbf{X}\|_1 = \text{tr}(\sqrt{\mathbf{X}^\top \mathbf{X}})$ , where  $\sqrt{\cdot}$  indicates the unique positive square root of a positive semi-definite matrix, and  $\text{tr}(\cdot)$  denotes the trace of a square matrix. This is given by  $\text{tr}(\mathbf{Y}) = \sum_{i=1}^n Y_{ii}$  for  $\mathbf{Y} \in \mathbb{R}^{n \times n}$ . The  $m \times m$  identity matrix is denoted  $\mathbf{I}^m$ . In addition, we define  $\mathbf{S}^m$  to be the set of  $m \times m$  symmetric matrices and let  $\mathbf{S}_+^m$  and  $\mathbf{S}_{++}^m$  be the subset of positive semidefinite and strictly positive definite matrices respectively. Recall that the set of symmetric matrices  $\mathbf{S}_+^m$  has the following partial ordering: for every  $\mathbf{M}, \mathbf{N} \in \mathbf{S}_+^m$ , we say that  $\mathbf{M} \preceq \mathbf{N}$  if and only if  $\mathbf{N} - \mathbf{M} \in \mathbf{S}_+^m$ . We also define the squared radius of  $\mathbf{M} \in \mathbf{S}_+^m$  as  $\mathcal{R}_M := \max_{i \in [m]} M_{ii}^+$ .

For every matrix  $\mathbf{U} \in \mathbb{R}^{m \times n}$ , we define  $\text{SP}(\mathbf{U}) = \{\mathbf{V} \in \mathbb{R}^{m \times n} : \forall_{ij} V_{ij} U_{ij} > 0\}$ , the set of matrices which are sign consistent with  $\mathbf{U}$ . We also define  $\text{SP}^1(\mathbf{U}) = \{\mathbf{V} \in \mathbb{R}^{m \times n} : \forall_{ij} V_{ij} \text{sign}(U_{ij}) \geq 1\}$ , that is the set of matrices which are sign consistent with  $\mathbf{U}$  with a margin of at least one. We now

introduce complexity measures for matrices. The max-norm (or  $\gamma_2$  norm [6]) of a matrix  $U \in \mathbb{R}^{m \times n}$  is defined by

$$\|U\|_{\max} := \min_{PQ^T=U} \left\{ \max_{1 \leq i \leq m} \|P_i\| \times \max_{1 \leq j \leq n} \|Q_j\| \right\}, \quad (1)$$

where the minimum is over all matrices  $P \in \mathbb{R}^{m \times d}$ ,  $Q \in \mathbb{R}^{n \times d}$  and every integer  $d$ . Instead of constraining the dimensionality of  $P_i$  and  $Q_j$  as is done with rank, it constrains their maximum  $\ell_2$ -norms. The *margin complexity* of a matrix  $U \in \mathbb{R}^{m \times n}$  is

$$\text{mc}(U) := \min_{V \in \text{SP}^1(U)} \|V\|_{\max} = \min_{PQ^T \in \text{SP}(U)} \max_{ij} \frac{\|P_i\| \|Q_j\|}{|\langle P_i, Q_j \rangle|}. \quad (2)$$

This is the counterpart of the max-norm to learn a sign matrix. Observe that for  $U \in \{-1, 1\}^{m \times n}$ ,  $1 \leq \text{mc}(U) \leq \|U\|_{\max} \leq \min(\sqrt{m}, \sqrt{n})$ , where the lower bound follows from the right hand side of (2) and the upper bound follows since we may decompose  $U = UI^n$  or as  $U = I^m U$ . Note there may be a large gap between the margin complexity and the max-norm. In [6] a matrix in  $U \in \{-1, 1\}^{n \times n}$  was given such that  $\text{mc}(U) = \log n$  and  $\|U\|_{\max} = \Theta(\sqrt{n}/\log n)$ . We denote the classes of  $m \times d$  *row-normalized* and *block expansion* matrices as  $\mathcal{N}^{m,d} := \{\hat{P} \subset \mathbb{R}^{m \times d} : \|\hat{P}_i\| = 1, i \in [m]\}$  and  $\mathcal{B}^{m,d} := \{R \subset \{0, 1\}^{m \times d} : \|R_i\| = 1, i \in [m], \text{rank}(R) = d\}$ , respectively. Block expansion matrices may be seen as a generalization of permutation matrices, additionally duplicating rows (columns) by left (right) multiplication. We define the *quasi-dimension* of a matrix  $U \in \mathbb{R}^{m \times n}$  with respect to  $M \in \mathcal{S}_{++}^m$ ,  $N \in \mathcal{S}_{++}^n$  at margin  $\gamma$  as

$$\mathcal{D}_{M,N}^\gamma(U) := \min_{\hat{P}\hat{Q}^T = \gamma U} \mathcal{R}_M \text{tr}(\hat{P}^T M \hat{P}) + \mathcal{R}_N \text{tr}(\hat{Q}^T N \hat{Q}), \quad (3)$$

where the infimum is over all row-normalized matrices  $\hat{P} \in \mathcal{N}^{m,d}$  and  $\hat{Q} \in \mathcal{N}^{n,d}$  and every integer  $d$ . If the infimum does not exist then  $\mathcal{D}_{M,N}^\gamma(U) := +\infty$ . Note that the infimum exists iff  $\|U\|_{\max} \leq 1/\gamma$ . The quasi-dimension quantifies how aligned the rows of  $\hat{P}$  ( $\hat{Q}$ ) are with the matrices  $M$  ( $N$ ). Note that  $\mathcal{D}_{M,N}^\gamma(U) = m + n$  if  $\|U\|_{\max} \leq 1/\gamma$ ,  $M = I^m$  and  $N = I^n$ .

We now introduce notation specific to the graph setting. Let  $\mathcal{G}$  be an  $m$ -vertex connected, undirected graph with positive weights. The Laplacian  $L$  of  $\mathcal{G}$  is defined as  $D - A$ , where  $D$  is the  $m \times m$  degree matrix and  $A$  is the  $m \times m$  adjacency matrix. Observe that as  $\mathcal{G}$  is connected,  $L$  is a rank  $m - 1$  matrix with 1 in its null space. From  $L$  we define the (strictly) positive definite *PDLaplacian*  $L^\circ := L + (\frac{1}{m}) (\frac{1}{m})^T \mathcal{R}_L^{-1}$ . Observe that if  $u \in [-1, 1]^m$  then  $(u^T L^\circ u) \mathcal{R}_{L^\circ} \leq 2(u^T L u \mathcal{R}_L + 1)$ , and similarly,  $(u^T L u) \mathcal{R}_L \leq \frac{1}{2}(u^T L^\circ u) \mathcal{R}_{L^\circ}$  (see [28] for details of this construction).

### 3 Transductive Matrix Completion

---

**Algorithm 1** Predicting a binary matrix with side information in the transductive setting.

---

**Parameters:** Learning rate:  $0 < \eta$ , quasi-dimension estimate:  $1 \leq \hat{D}$ , margin estimate:  $0 < \gamma \leq 1$ , non-conservative flag [NON-CONSERVATIVE]  $\in \{0, 1\}$  and side information matrices  $M \in \mathcal{S}_{++}^m$ ,  $N \in \mathcal{S}_{++}^n$  with  $m + n \geq 3$

**Initialization:**  $\mathbb{M} \leftarrow \emptyset$ ;  $\tilde{W}^1 \leftarrow \frac{\hat{D}}{(m+n)} I^{m+n}$ .

**For**  $t = 1, \dots, T$

- Receive pair  $(i_t, j_t) \in [m] \times [n]$ .

- Define

$$\tilde{X}^t := x^t (x^t)^\top := \begin{bmatrix} \sqrt{M^+} e_m^{i_t} & \sqrt{N^+} e_n^{j_t} \\ \sqrt{2\mathcal{R}_M} & \sqrt{2\mathcal{R}_N} \end{bmatrix} \begin{bmatrix} \sqrt{M^+} e_m^{i_t} & \sqrt{N^+} e_n^{j_t} \\ \sqrt{2\mathcal{R}_M} & \sqrt{2\mathcal{R}_N} \end{bmatrix}^\top. \quad (4)$$

- Predict

$$Y_t \sim \text{UNIFORM}(-\gamma, \gamma) \times [\text{NON-CONSERVATIVE}]; \bar{y}_t \leftarrow \text{tr}(\tilde{W}^t \tilde{X}^t) - 1; \hat{y}_t \leftarrow \text{sign}(\bar{y}_t - Y_t).$$

- Receive label  $y_t \in \{-1, 1\}$ .

- If  $y_t \neq \hat{y}_t$  then  $\mathbb{M} \leftarrow \mathbb{M} \cup \{t\}$ .

- If  $y_t \hat{y}_t < \gamma \times [\text{NON-CONSERVATIVE}]$  then

$$\tilde{W}^{t+1} \leftarrow \exp(\log(\tilde{W}^t) + \eta y_t \tilde{X}^t).$$

- Else  $\tilde{W}^{t+1} \leftarrow \tilde{W}^t$ .

---

Algorithm 1 corresponds to an adapted MATRIX EXPONENTIATED GRADIENT (MEG) algorithm [25] to perform transductive matrix completion with side information. Although the algorithm is a special case of MEG, the following theorem does not follow as a special case of the analysis in [25]. The underlying proof techniques are discussed in further detail in Appendix B.2. The following theorem includes an expected regret bound and a mistake bound, where the regret bound is the more flexible, noise-tolerant bound. Hence, in practice, the algorithm should always be run non-conservatively to obtain the regret bound, as per the theorem assumptions. We only include the conservative case for further insight as the bound is simpler, being analogous to Novikoff’s (perceptron) bound. In this case, where we also assume realizability and exact tuning ( $\gamma = 1/\text{mc}(\mathbf{U})$ ), the mistakes are bounded by  $\tilde{O}(\mathcal{D}\text{mc}(\mathbf{U})^2)$ . The term  $\mathcal{D}$  evaluates the predictive quality of the side information provided to the algorithm. In order to evaluate  $\mathcal{D}$ , we provide an upper bound in Theorem 3 that is more straightforward to interpret. Examples are given in Sections 4.1 and 5.1, where Theorem 3 is applied to evaluate the quality of side information in idealized scenarios. The  $\text{mc}(\mathbf{U})$  term is analogous to the inverse margin in perceptron, SVM, and other “large margin” classifiers.

**Theorem 1.** *The expected regret of Algorithm 1 with **non-conservative** updates ([NON-CONSERVATIVE] = 1) and parameters  $\gamma \in (0, 1]$ ,  $\hat{\mathcal{D}} \geq \mathcal{D}_{M,N}^\gamma(\mathbf{U})$ ,  $\eta = \sqrt{\frac{\hat{\mathcal{D}} \log(m+n)}{2T}}$ , p.d. matrices  $\mathbf{M} \in \mathbf{S}_{++}^m$  and  $\mathbf{N} \in \mathbf{S}_{++}^n$  is bounded by*

$$\mathbb{E}[|\mathbb{M}|] - \sum_{t \in [T]} [y_t \neq U_{i_t j_t}] \leq 4\sqrt{2(\hat{\mathcal{D}}/\gamma^2) \log(m+n)T} \quad (5)$$

for all  $\mathbf{U} \in \{-1, 1\}^{m \times n}$  with  $\|\mathbf{U}\|_{\max} \leq 1/\gamma$ .

The mistakes in the **realizable** case with **conservative** updates ([NON-CONSERVATIVE] = 0) and parameters  $1/\eta = 1/\gamma \geq \text{mc}(\mathbf{U})$ ,  $\hat{\mathcal{D}} \geq \min_{\mathbf{V} \in \text{SP}^1(\mathbf{U})} \mathcal{D}_{M,N}^\gamma(\mathbf{V})$  are bounded by,

$$|\mathbb{M}| \leq 3.6(\hat{\mathcal{D}}/\gamma^2) \log(m+n), \quad (6)$$

for all  $\mathbf{U} \in \{-1, 1\}^{m \times n}$  with  $\text{mc}(\mathbf{U}) \leq 1/\gamma$  and  $y_t = U_{i_t j_t}$  for all  $t \in \mathbb{M}$ .

If the side information is vacuous, that is  $\mathbf{M} = \mathbf{I}^m$  and  $\mathbf{N} = \mathbf{I}^n$ , then  $\mathcal{D} = m+n$ . In this scenario, we recover a special case<sup>1</sup> of the analysis of [21] up to constant factors, then with the additional assumption of realizability we recover [24, Theorem 3.1]. The term  $\mathcal{D}$  is difficult to directly quantify. In the next section, we specialize our analysis to the case that the matrix  $\mathbf{U}$  has a latent block structure.

## 4 Latent Block Structure

We introduce the concept class of  $(k, \ell)$ -binary-biclustered matrices (previously defined in [24, Section 5]), in the following definition. We then give an upper bound to  $\mathcal{D}_{M,N}^\gamma(\mathbf{U})$  when a matrix has this type of latent structure in Theorem 3. The magnitude of the bound will depend on how “predictive” matrices  $\mathbf{M}$  and  $\mathbf{N}$  are of the latent block structure. In Sections 4.1 and 4.2, we will use a variant of the discrete Laplacian matrix for  $\mathbf{M}$  and  $\mathbf{N}$  to encode side information and illustrate the resultant bounds for idealized scenarios.

**Definition 2.** *The class of  $(k, \ell)$ -binary-biclustered matrices is defined as*

$$\mathbb{B}_{k,\ell}^{m,n} = \{\mathbf{U} \in \{-1, 1\}^{m \times n} : \mathbf{r} \in [k]^m, \mathbf{c} \in [\ell]^n, \mathbf{U}^* \in \{-1, 1\}^{k \times \ell}, U_{ij} = U_{r_i c_j}^*, i \in [m], j \in [n]\}.$$

Thus each row  $r_i$  is associated with a latent factor in  $[k]$  and each column  $c_j$  is associated with a latent factor in  $[\ell]$  and the interaction of factors is determined by a matrix  $\mathbf{U}^* \in \{-1, 1\}^{k \times \ell}$ . More visually, a binary matrix is  $(k, \ell)$ -biclustered if there exists some permutation of the rows and columns into a  $k \times \ell$  grid of blocks each uniformly labeled  $-1$  or  $+1$ , as illustrated in Figure 1. Determining if a matrix is in  $\mathbb{B}_{k,\ell}^{m,n}$ , may be done directly by a greedy algorithm. However, the problem of determining

<sup>1</sup>In [21], a regret bound for general loss functions for matrix completion without side information is given for  $(\beta, \tau)$ -decomposable matrices. When  $\beta$  is at its minimum over all possible decompositions, we recover the bound up to constant factors with respect to the expected 0-1 loss. On the algorithmic level, our works are similar except that the algorithm of [21] contains an additional projection step that dominates the computation time of the update.

if a matrix with missing entries may be completed to a matrix in  $\mathbb{B}_{k,n}^{m,n}$  was shown in [29, Lemma 8] to be NP-COMplete by reducing the problem to CLIQUE COVER.

Many natural functions of matrix complexity are invariant to the presence of block structure. A function  $f : \mathcal{X} \rightarrow \mathfrak{R}$  with respect to a class of matrices  $\mathcal{X}$  is *block-invariant* if for all  $m, k, n, \ell \in \mathbb{N}^+$  with  $m \geq k, n \geq \ell$ ,  $\mathbf{R} \in \mathcal{B}^{m,k}$  and  $\mathbf{C} \in \mathcal{B}^{n,\ell}$  we have that  $f(\mathbf{X}) = f(\mathbf{R}\mathbf{X}\mathbf{C}^\top)$  for any  $k \times \ell$  matrix  $\mathbf{X} \in \mathcal{X}$ . The max-norm, margin complexity, rank and VC-dimension<sup>2</sup> are all block-invariant. From the block-invariance of the max-norm, we may conclude that for  $\mathbf{U} \in \mathbb{B}_{k,\ell}^{m,n}$ ,

$$\text{mc}(\mathbf{U}) \leq \|\mathbf{U}\|_{\max} = \|\mathbf{U}^*\|_{\max} \leq \min(\sqrt{k}, \sqrt{\ell}). \quad (7)$$

This follows since we may decompose  $\mathbf{U} = \mathbf{R}\mathbf{U}^*\mathbf{C}^\top$  for some  $\mathbf{U}^* \in \{-1, 1\}^{k \times \ell}$ ,  $\mathbf{R} \in \mathcal{B}^{m,k}$  and  $\mathbf{C} \in \mathcal{B}^{n,\ell}$  and then use the observation in the preliminaries that the max-norm of any matrix in  $\{-1, 1\}^{m \times n}$  is bounded by  $\min(\sqrt{m}, \sqrt{n})$ .

In the following theorem, we give a bound for the quasi-dimension  $\mathcal{D}_{M,N}^\gamma(\mathbf{U})$  which will scale with the dimensions of the latent block structure and the ‘‘predictivity’’ of  $\mathbf{M}$  and  $\mathbf{N}$  with respect to that block structure. The bound is independent of  $\gamma$  in so far as  $\mathcal{D}_{M,N}^\gamma(\mathbf{U})$  is finite.

**Theorem 3.** *If  $\mathbf{U} \in \mathbb{B}_{k,\ell}^{m,n}$  define*

$$\mathcal{D}_{M,N}^\circ(\mathbf{U}) := \begin{cases} 2 \text{tr}(\mathbf{R}^\top \mathbf{M} \mathbf{R}) \mathcal{R}_M + 2 \text{tr}(\mathbf{C}^\top \mathbf{N} \mathbf{C}) \mathcal{R}_N + 2k + 2\ell & \mathbf{M} \text{ and } \mathbf{N} \text{ are PDLaplacians} \\ k \text{tr}(\mathbf{R}^\top \mathbf{M} \mathbf{R}) \mathcal{R}_M + \ell \text{tr}(\mathbf{C}^\top \mathbf{N} \mathbf{C}) \mathcal{R}_N & \mathbf{M} \in \mathbf{S}_{++}^m \text{ and } \mathbf{N} \in \mathbf{S}_{++}^n \end{cases}, \quad (8)$$

as the minimum over all decompositions of  $\mathbf{U} = \mathbf{R}\mathbf{U}^*\mathbf{C}^\top$  for  $\mathbf{R} \in \mathcal{B}^{m,k}$ ,  $\mathbf{C} \in \mathcal{B}^{n,\ell}$  and  $\mathbf{U}^* \in \{-1, 1\}^{k \times \ell}$ . Thus for  $\mathbf{U} \in \mathbb{B}_{k,\ell}^{m,n}$ ,

$$\begin{aligned} \mathcal{D}_{M,N}^\gamma(\mathbf{U}) &\leq \mathcal{D}_{M,N}^\circ(\mathbf{U}) && \text{(if } \|\mathbf{U}\|_{\max} \leq 1/\gamma) \\ \min_{\mathbf{V} \in \text{SP}^1(\mathbf{U})} \mathcal{D}_{M,N}^\gamma(\mathbf{V}) &\leq \mathcal{D}_{M,N}^\circ(\mathbf{U}) && \text{(if } \text{mc}(\mathbf{U}) \leq 1/\gamma). \end{aligned}$$

The bound  $\mathcal{D}_{M,N}^\gamma(\mathbf{U}) \leq \mathcal{D}_{M,N}^\circ(\mathbf{U})$  allows us to bound the quality of the side information in terms of a hypothetical learning problem. Recall that  $\text{argmin}_{r_i, y_i \geq 1: i \in [m]} (\mathbf{r}^\top \mathbf{M} \mathbf{r}) \mathcal{R}_M$  is the upper bound on the mistakes per Novikoff’s theorem [30] for predicting the elements of vector  $\mathbf{y} \in \{-1, 1\}^m$  with a kernel perceptron using  $\mathbf{M}^{-1}$  as the kernel. Hence the term  $\mathcal{O}(\text{tr}(\mathbf{R}^\top \mathbf{M} \mathbf{R}) \mathcal{R}_M)$  in (8) may be interpreted as a bound for a one-versus-all  $k$ -class kernel perceptron where  $\mathbf{R}$  encodes a labeling from  $[k]^m$  as one-hot vectors. We next show an example where  $\mathcal{D}_{M,N}^\circ(\mathbf{U}) \in \mathcal{O}(k + \ell)$  with ‘‘ideal’’ side information.

#### 4.1 Graph-based Side Information

We may use a pair of separate graph Laplacians to represent the side information on the ‘‘rows’’ and the ‘‘columns.’’ A given row (column) corresponds to a vertex in the ‘‘row graph’’ (‘‘column graph’’). The weight of edge  $(i, j)$  represents our prior belief that row (column)  $i$  and row (column)  $j$  share the same underlying factor. Such graphs may be inherent to the data. For example, we have a social network of users and a network based on shared actors or genres for the movies in a ‘‘Netflix’’ type scenario. Alternatively, as is common in graph-based semi-supervised learning [31; 32] we may build a graph based on vectorial data associated with the rows (columns), for example, user demographics. Although the value of  $\mathcal{D}$  will vary smoothly with the predictivity of  $\mathbf{M}$  and  $\mathbf{N}$  of the factor structure, in the following we give an example to quantify  $\mathcal{D}^\circ$  in a best case scenario.

**Bounding  $\mathcal{D}^\circ$  for ‘‘ideal’’ graph-based side information.** In this ideal case we are assuming that we know the partition of  $[m]$  that maps rows to factors. The rows that share factors have an edge between them and there are no other edges. Given  $k$  factors, we then have a graph that consists of  $k$  disjoint cliques. However, to meet the technical requirement that the side information matrix  $\mathbf{M}(\mathbf{N})$  is positive definite, we need to connect the cliques in a minimal fashion. We achieve this by connecting the cliques like a ‘‘star’’ graph. Specifically, a clique is arbitrarily chosen as the center and a vertex in that clique is arbitrarily chosen as the central vertex. From each of the other cliques, a vertex is chosen arbitrarily and connected to the central vertex. Observe that a property of this construction is that there is a path of length  $\leq 4$  between any pair of vertices. Now we can use the bound from Theorem 3,

$$\mathcal{D}^\circ = 2 \text{tr}(\mathbf{R}^\top \mathbf{M} \mathbf{R}) \mathcal{R}_M + 2 \text{tr}(\mathbf{C}^\top \mathbf{N} \mathbf{C}) \mathcal{R}_N + 2k + 2\ell,$$

<sup>2</sup>Here, a hypothesis class  $\mathcal{H}$  defines a matrix via  $\mathbf{U} := (h(x))_{h \in \mathcal{H}, x \in \mathcal{X}}$ .

to bound  $\mathcal{D} \leq \mathcal{D}^\circ$  in this idealized case. We focus on the rows, as a parallel argument may be made for the side information on the columns. Consider the term  $\text{tr}(\mathbf{R}^\top \mathbf{M} \mathbf{R}) \mathcal{R}_M$ , where  $\mathbf{M} := \mathbf{L}^\circ$  is the PD-Laplacian formed from a graph with Laplacian  $\mathbf{L}$ . Then using the observation from the preliminaries that  $(\mathbf{u}^\top \mathbf{L}^\circ \mathbf{u}) \mathcal{R}_{L^\circ} \leq 2(\mathbf{u}^\top \mathbf{L} \mathbf{u} \mathcal{R}_L + 1)$ , we have that  $\text{tr}(\mathbf{R}^\top \mathbf{M} \mathbf{R}) \mathcal{R}_M \leq 2 \text{tr}(\mathbf{R}^\top \mathbf{L} \mathbf{R}) \mathcal{R}_L + 2k$ . To evaluate this, we use the well-known equality of  $\text{tr}(\mathbf{R}^\top \mathbf{L} \mathbf{R}) = \sum_{(i,j) \in E} \|\mathbf{R}_i - \mathbf{R}_j\|^2$ . Observing that each of the  $m$  rows of  $\mathbf{R}$  is a ‘‘one-hot’’ encoding of the corresponding factor, only the edges between classes then contribute to the sum of the norms, and thus by construction  $\text{tr}(\mathbf{R}^\top \mathbf{L} \mathbf{R}) \leq k - 1$ . We bound  $\mathcal{R}_L \leq 4$ , using the fact that the graph diameter is a bound on  $\mathcal{R}_L$  (see [33, Theorem 4.2]). Combining terms and assuming similar idealized side information on the columns, we obtain  $\mathcal{D}^\circ \in O(k + \ell)$ . Observe then that since the comparator matrix is  $(k, \ell)$ -biclustered, we have in the realizable case (with exact tuning), that  $\text{mc}(\mathbf{U})^2 \leq \min(k, \ell)$  by (7). Thus, the mistakes of the algorithm are bounded by  $\tilde{O}(\text{mc}(\mathbf{U})^2 \mathcal{D}^\circ) = \tilde{O}(k\ell)$ . This upper bound is tight up to logarithmic factors as we may decompose  $\mathbf{U} = \mathbf{R} \mathbf{U}^* \mathbf{C}^\top$  for some  $\mathbf{U}^* \in \{-1, 1\}^{k \times \ell}$ ,  $\mathbf{R} \in \mathcal{B}^{m, k}$  and  $\mathbf{C} \in \mathcal{B}^{n, \ell}$  and force a mistake for each of the  $k\ell$  entries in  $\mathbf{U}^*$ .

Can side information provably help? Unsurprisingly, yes. Consider the set of matrices such that each row is either all ‘+1’ or all ‘-1’. This set is exactly  $\mathbb{B}_{2,1}^{m,n}$ . Clearly, an adversary can force  $m$  mistakes, whereas with ‘‘ideal’’ side information the upper bound is  $\tilde{O}(1)$ .

Similar results to the above can be obtained via alternate positive definite embeddings. For example, consider a  $k$ -partition kernel of  $[m]$  where  $K_{\epsilon, S_1, \dots, S_k}(i, j) := [i, j \in S_r : r \in [k]] + \epsilon[i = j]$  for some partition of  $[m]$  into disjoint sets  $S_1, \dots, S_k$ . By using  $\mathbf{M}^{-1} = (K(i, j))_{i, j \in [m]}$  one can obtain for small  $\epsilon$ , bounds that are tighter than achieved by the Laplacian with respect to constant factors. We have focused on the Laplacian as a method for encoding side information as it is more straightforward to encode [32] ‘‘softer’’ knowledge of relationships.

## 4.2 Online Community Membership Prediction

A special case of matrix completion is the case where there are  $m$  objects which are assumed to lie in  $k$  classes (communities). In this case, the underlying matrix  $\mathbf{U} \in \{-1, 1\}$  is given by  $U_{ij} = 1$  if  $i$  and  $j$  are in the same class and  $U_{ij} = -1$  otherwise. Thus this may be viewed as an online version of community detection or ‘‘similarity’’ prediction. In [22], this problem was addressed when the side information was encoded in a graph and the aim was to perform well when there were few edges between classes (communities).

Observe that this is an example of a  $(k, k)$ -biclustered  $m \times m$  matrix where  $\mathbf{U}^* = 2\mathbf{I}^k - \mathbf{1}\mathbf{1}^\top$  and there exists  $\mathbf{R} \in \mathcal{B}^{m, k}$  such that  $\mathbf{U} := \mathbf{R} \mathbf{U}^* \mathbf{R}^\top$ . Since the max-norm is block-invariant, we have that  $\|\mathbf{U}\|_{\max} = \|\mathbf{U}^*\|_{\max}$ . In the case of a general  $k \times k$  biclustered matrix,  $\|\mathbf{U}^*\|_{\max} \leq \sqrt{k}$  (see (7)). However in the case of ‘‘similarity prediction’’, we have  $\|\mathbf{U}^*\|_{\max} \in O(1)$ . This follows since we have a decomposition  $\mathbf{U}^* = \mathbf{P} \mathbf{Q}^\top$  by  $\mathbf{P}, \mathbf{Q} \in \mathfrak{R}^{k, k+1}$  with  $\mathbf{P} := (P_{ij} = \sqrt{2}[i = j] + [j = k+1])_{i \in [k], j \in [k+1]}$  and  $\mathbf{Q} := (Q_{ij} = \sqrt{2}[i = j] - [j = k+1])_{i \in [k], j \in [k+1]}$ , thus giving  $\|\mathbf{U}^*\|_{\max} \leq 3$ . This example also shows that there may be an arbitrary gap between rank and max-norm of  $\pm 1$  matrices as the rank of  $\mathbf{U}^*$  is  $k$  (in [6] this gap between the max-norm and rank was previously observed). Therefore, if the side-information matrices are taken to be the same PDLaplacian  $\mathbf{M} = \mathbf{N}$  defined from a Laplacian  $\mathbf{L}$ , we have that since  $\|\mathbf{U}\|_{\max} \in O(1)$  and  $\mathcal{D}^\circ \in O(\text{tr}(\mathbf{R}^\top \mathbf{L} \mathbf{R}) \mathcal{R}_L)$ , a mistake bound of  $\tilde{O}(\text{tr}(\mathbf{R}^\top \mathbf{L} \mathbf{R}) \mathcal{R}_L)$  is obtained, which recovers the bound of [22, Proposition 4] up to constant factors. This work extends the results in [22] for similarity prediction to regret bounds, and to the inductive setting with general p.d. matrices. In the next section, we will see how this type of result may be extended to an inductive setting.

## 5 Inductive Matrix Completion

In the previous section, the learner was assumed to have complete foreknowledge of the side information through the matrices  $\mathbf{M}$  and  $\mathbf{N}$ . In the inductive setting, the learner has instead kernel side information functions  $\mathcal{M}^+$  and  $\mathcal{N}^+$ . With complete foreknowledge of the rows (columns) that will be observed, one may use  $\mathcal{M}^+$  ( $\mathcal{N}^+$ ) to compute  $\mathbf{M}$  ( $\mathbf{N}$ ), which corresponds to an inverse of a submatrix of  $\mathcal{M}^+$  ( $\mathcal{N}^+$ ). In the inductive, unlike the transductive setting, we do not have this foreknowledge and thus cannot compute  $\mathbf{M}$  ( $\mathbf{N}$ ) in advance. Notice that the assumption of side information as kernel functions is not particularly limiting, as for instance the side information could be provided by vectors in  $\mathfrak{R}^d$  and the kernel could be the positive definite linear kernel

$\mathbf{K}_\epsilon(\mathbf{x}, \mathbf{x}') := \langle \mathbf{x}, \mathbf{x}' \rangle + \epsilon[\mathbf{x} = \mathbf{x}']$ . On the other hand, despite the additional flexibility of the inductive setting versus the transductive one, there are two limitations. First, only in a technical sense will it be possible to model side information via a PDLaplacian, since  $\mathbf{M}^+$  can only be computed given knowledge of the graph in advance. Second, the bound in Theorem 3 on the quasi-dimension  $\mathcal{D} \leq \mathcal{D}^\circ$  gains additional multiplicative factors  $k$  and  $\ell$ . Nevertheless, we will observe in Section 5.1 that, for a given kernel for which the side information associated with a given row (column) latent factor is “well-separated” from distinct latent factors, we can show that  $\mathcal{D}^\circ \in \mathcal{O}(k^2 + \ell^2)$ .

---

**Algorithm 2** Predicting a binary matrix with side information in the inductive setting.

---

**Parameters:** Learning rate:  $0 < \eta$  quasi-dimension estimate:  $1 \leq \widehat{\mathcal{D}}$ , margin estimate:  $0 < \gamma \leq 1$ , non-conservative flag [NON-CONSERVATIVE]  $\in \{0, 1\}$  and side-information kernels  $\mathcal{M}^+ : \mathcal{I} \times \mathcal{I} \rightarrow \mathfrak{R}$ ,  $\mathcal{N}^+ : \mathcal{J} \times \mathcal{J} \rightarrow \mathfrak{R}$ , with  $\mathcal{R}_{\mathcal{M}} := \max_{i \in \mathcal{I}} \mathcal{M}^+(i, i)$  and  $\mathcal{R}_{\mathcal{N}} := \max_{j \in \mathcal{J}} \mathcal{N}^+(j, j)$ , and maximum distinct rows  $m$  and columns  $n$ , where  $m + n \geq 3$ .

**Initialization:**  $\mathbb{M} \leftarrow \emptyset, \mathbb{U} \leftarrow \emptyset, \mathcal{I}^1 \leftarrow \emptyset, \mathcal{J}^1 \leftarrow \emptyset$ .

**For**  $t = 1, \dots, T$

- Receive pair  $(i_t, j_t) \in \mathcal{I} \times \mathcal{J}$ .

- Define

$$\begin{aligned} (\mathbf{M}^t)^+ &:= (\mathcal{M}^+(i_r, i_s))_{r,s \in \mathcal{I}^t \cup \{i_t\}}; & (\mathbf{N}^t)^+ &:= (\mathcal{N}^+(j_r, j_s))_{r,s \in \mathcal{J}^t \cup \{j_t\}}, \\ \tilde{\mathbf{X}}^t(s) &:= \begin{bmatrix} \frac{(\sqrt{(\mathbf{M}^t)^+})e^{i_s}}{\sqrt{2\mathcal{R}_{\mathcal{M}}}}; \frac{(\sqrt{(\mathbf{N}^t)^+})e^{j_s}}{\sqrt{2\mathcal{R}_{\mathcal{N}}}} \end{bmatrix} \begin{bmatrix} \frac{(\sqrt{(\mathbf{M}^t)^+})e^{i_s}}{\sqrt{2\mathcal{R}_{\mathcal{M}}}}; \frac{(\sqrt{(\mathbf{N}^t)^+})e^{j_s}}{\sqrt{2\mathcal{R}_{\mathcal{N}}}} \end{bmatrix}^\top, \\ \log(\tilde{\mathbf{W}}^t) &\leftarrow \log\left(\frac{\widehat{\mathcal{D}}}{m+n}\right) \mathbf{I}^{|\mathcal{I}^t|+|\mathcal{J}^t|+2} + \sum_{s \in \mathbb{U}} \eta y_s \tilde{\mathbf{X}}^t(s). \end{aligned}$$

- Predict

$$Y_t \sim \text{UNIFORM}(-\gamma, \gamma) \times [\text{NON-CONSERVATIVE}]; \quad \bar{y}_t \leftarrow \text{tr}(\tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t) - 1; \quad \hat{y}_t \leftarrow \text{sign}(\bar{y}_t - Y_t).$$

- Receive label  $y_t \in \{-1, 1\}$ .

- If  $y_t \neq \hat{y}_t$  then  $\mathbb{M} \leftarrow \mathbb{M} \cup \{t\}$ .

- If  $y_t \bar{y}_t < \gamma \times [\text{NON-CONSERVATIVE}]$  then

$$\mathbb{U} \leftarrow \mathbb{U} \cup \{t\}, \quad \mathcal{I}^{t+1} \leftarrow \mathcal{I}^t \cup \{i_t\}, \quad \text{and} \quad \mathcal{J}^{t+1} \leftarrow \mathcal{J}^t \cup \{j_t\}.$$

- Else  $\mathcal{I}^{t+1} \leftarrow \mathcal{I}^t$  and  $\mathcal{J}^{t+1} \leftarrow \mathcal{J}^t$ .

---

Algorithm 2 is prediction-equivalent to Algorithm 1 up to the value of  $\mathcal{R}_{\mathcal{M}}(\mathcal{R}_{\mathcal{N}})$ . In [34], the authors provide very general conditions for the “kernelization” of algorithms with an emphasis on “matrix” algorithms. They sketch a method to kernelize the MATRIX EXPONENTIATED GRADIENT algorithm based on the relationship between the eigensystems of the kernel matrix and the Gram matrix. We take a different, more direct approach, in which we prove its correctness via Proposition 4. The intuition behind the algorithm is that, although we cannot efficiently embed the row and column kernel functions  $\mathcal{M}^+$  and  $\mathcal{N}^+$  as matrices since they are potentially infinite-dimensional, we may instead work with the embedding corresponding to the currently observed rows and columns, recompute the embedding on a per-trial basis, and then “replay” all re-embedded past examples to create the current hypothesis matrix. The following is our proposition of equivalency, proven in Appendix D.

**Proposition 4.** *The inductive and transductive algorithms are equivalent up to  $\mathcal{R}_{\mathcal{M}}$  and  $\mathcal{R}_{\mathcal{N}}$ . Without loss of generality assume  $\mathcal{I}^{T+1} \subseteq [m]$  and  $\mathcal{J}^{T+1} \subseteq [n]$ . Define  $\mathbf{M} := ((\mathcal{M}^+(i', i''))_{i', i'' \in [m]})^+$  and  $\mathbf{N} := ((\mathcal{N}^+(j', j''))_{j', j'' \in [n]})^+$ . Assume that for the transductive algorithm, the matrices  $\mathbf{M}$  and  $\mathbf{N}$  are given whereas for the inductive algorithm, only the strictly positive definite kernel functions  $\mathcal{M}^+$  and  $\mathcal{N}^+$  are provided. Then, if  $\mathcal{R}_{\mathcal{M}} = \mathcal{R}_{\mathcal{M}}$  and  $\mathcal{R}_{\mathcal{N}} = \mathcal{R}_{\mathcal{N}}$ , and if the algorithms receive the same label and index sequences, then the predictions of the algorithms are the same.*

Thus, the only case when the algorithms are different is when  $\mathcal{R}_{\mathcal{M}} \neq \mathcal{R}_{\mathcal{M}}$  or  $\mathcal{R}_{\mathcal{N}} \neq \mathcal{R}_{\mathcal{N}}$ . This is a minor inequivalency, as the only resultant difference is in the term  $\mathcal{D}$ . Alternatively, if one uses a normalized kernel such as the Gaussian, then  $\mathcal{R}_{\mathcal{M}} = \mathcal{R}_{\mathcal{M}} = 1$ .

For Algorithm 1 we have a per trial time complexity of  $\mathcal{O}(\max(m, n)^3)$ ; Algorithm 2 has a per trial complexity of  $\mathcal{O}(\min(\max(m, n)^4, T^3))$ . In the dominant step on every trial (with an update) of the transductive algorithm there is an SVD of a  $(m+n) \times (m+n)$  matrix; thus, the algorithm requires  $\mathcal{O}(\max(m, n)^3)$  time. We split the analysis of the inductive algorithm into two cases. In the case



that  $\max(m, n) \ll T$ , the complexity on every trial is dominated by the sum of up to  $mn$  matrices of size up to  $(m+n) \times (m+n)$  (i.e., in the regret setting we can collapse terms from multiple observations of the same matrix entry) and thus has a per-trial complexity of  $\mathcal{O}(\max(m, n)^4)$ . In the other case on trial  $t$  we need  $\mathcal{O}(t^3)$  time since we need to compute the eigendecomposition of three  $\mathcal{O}(t) \times \mathcal{O}(t)$  matrices as well as sum  $\mathcal{O}(t) \times \mathcal{O}(t)$  matrices up to  $t$  times. Putting together we have a time complexity of  $\mathcal{O}(\min(\max(m, n)^4, T^3))$  per trial. In the following subsection, we describe a scenario where the quasi-dimension bound  $\mathcal{D}^\circ$  scales quadratically with the number of distinct factors.

### 5.1 Side information in $[-r, r]^d$

In the following, we show an example for predicting a matrix  $\mathbf{U} \in \mathbb{B}_{k, \ell}^{m, n}$  such that for online side information in  $[-r, r]^d$  that is well-separated into *boxes*, there exists a kernel for which the quasi-dimension grows no more than quadratically with the number of latent factors (but exponentially with the dimension  $d$ ). For simplicity, we use the *min* kernel, which approximates functions by linear interpolation. In practice, we speculate that similar results may be proven for other universal kernels, but the analysis with the min kernel has the advantage of simplicity.

In the previous section, with the idealized graph-based side information, one may be dissatisfied as the skeleton of the latent structure is essentially encoded into  $\mathbf{M}(\mathbf{N})$ . In the inductive setting, the side information is instead revealed in an online fashion. If such side information may be separated into distinct clusters retrospectively, we will be able to bound  $\mathcal{D}^\circ \in \mathcal{O}(k^2 + \ell^2)$ . In this example, we receive a row and column vector  $\mathbf{v}_t, \mathbf{j}_t \in [-r, r]^d \times [-r', r']^{d'}$  on each trial; these vectors will be the indices to our row and column kernels, and for simplicity we set  $r = r'$  and  $d = d'$ .

**Bounding  $\mathcal{D}^\circ$  for the *min* kernel.** Define the transformation  $s(\mathbf{x}) := \frac{r-1}{2r}\mathbf{x} + \frac{r+1}{2}$  and the *min* kernel  $\mathcal{K} : [0, r]^d \times [0, r]^d \rightarrow \mathfrak{R}$  as  $\mathcal{K}(\mathbf{x}, \mathbf{t}) := \prod_{i=1}^d \min(x_i, t_i)$ . Also define  $\delta(S_1, \dots, S_k) := \min_{1 \leq i < j \leq k} \min_{\mathbf{x} \in S_i, \mathbf{x}' \in S_j} \|\mathbf{x} - \mathbf{x}'\|_\infty$ . A *box* in  $\mathfrak{R}^d$  is a set  $\{\mathbf{x} : a_i \leq x_i \leq b_i, i \in [d]\}$  defined by a pair of vectors  $\mathbf{a}, \mathbf{b} \in \mathfrak{R}^d$ .

**Proposition 5.** *Given  $k$  boxes  $S_1, \dots, S_k \subset [-r, r]^d$ ,  $r \geq 2$ ,  $\delta^* = \min(2, \frac{1}{4}\delta(S_1, \dots, S_k))$ , and  $\mathbf{x}_1, \dots, \mathbf{x}_m \in \cup_{i=1}^k S_i$ , if  $\mathbf{R} = ([\mathbf{x}_i \in S_j])_{i \in [m], j \in [k]}$  and  $\mathbf{K} = (\mathcal{K}(s(\mathbf{x}_i), s(\mathbf{x}_j)))_{i, j \in [m]}$  then  $\text{tr}(\mathbf{R}^\top \mathbf{K}^{-1} \mathbf{R}) \leq k \left(\frac{4}{\delta^*}\right)^d$ .*

Recall the bound (see (8)) on the quasi-dimension for a matrix  $\mathbf{U} \in \mathbb{B}_{k, \ell}^{m, n}$ , where we have  $\mathcal{D} \leq \mathcal{D}^\circ = k \text{tr}(\mathbf{R}^\top \mathbf{M} \mathbf{R}) \mathcal{R}_M + \ell \text{tr}(\mathbf{C}^\top \mathbf{N} \mathbf{C}) \mathcal{R}_N$  for positive definite matrices. If we assume that the side information on the rows (columns) lies in  $[-r, r]^d$ , then  $\mathcal{R}_M \leq \mathcal{R}_M \leq r^d$  ( $\mathcal{R}_N \leq \mathcal{R}_N \leq r^d$ ) for the min kernel. Thus by applying the above proposition separately for the rows and columns and substituting into (8), we have that

$$\mathcal{D} \leq \mathcal{D}^\circ = k^2(4r/\delta^*)^d + \ell^2(4r/\delta^*)^d.$$

We then observe that for this example, with an optimal tuning and well-separated side information on the rows and columns, the mistake bound for a  $(k, \ell)$ -biclustered matrix in the inductive setting is of  $\tilde{\mathcal{O}}(\min(k, \ell) \max(k, \ell)^2)$ . However, our best lower bound in terms of  $k$  and  $\ell$  is just  $k\ell$ , as in the transductive setting. An open problem is to resolve this gap.

## 6 Discussion

We have presented a regret bound with respect to the 0-1 loss for predicting the elements of matrices with latent block structure in the presence of side information. The bound scales as  $\tilde{\mathcal{O}}\left(\sqrt{\mathcal{D}_{M, N}^\gamma(\mathbf{U}) \min(k, \ell) T}\right)$  for  $\mathbf{U} \in \mathbb{B}_{k, \ell}^{m, n}$ . In the case of idealized side information, the term  $\mathcal{D}$  scaled linearly with  $k$  and  $\ell$  in the transductive setting, and quadratically in the inductive setting. Problems for further research include resolving the gap between the upper and lower bounds in the inductive setting and building richer models of side information. In this work, the interrelation was restricted to equivalence relations on the row (column) spaces. A direct generalization would be to consider partial orderings. Such a generalization would be natural for ranking based on concept classes such as the online gambling scenario given in [21]. A broader generalization would allow side information to hint between row and column interrelationships rather than keeping them independent.

## **7 Acknowledgements**

We would like to thank Robin Hirsch for valuable discussions. This research was supported by the U.S. Army Research Laboratory and the U.K. Ministry of Defence under Agreement Number W911NF-16-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon. This research was further supported by the Engineering and Physical Sciences Research Council grant number EP/L015242/1.

## **Broader Impact**

*In general this work does not present any foreseeable specific societal consequence in the authors' joint opinion.*

This is foundational research in *regret-bounded online learning*. As such it is not targeted towards any particular application area. Although this research may have societal impact for good or for ill in the future, we cannot foresee the shape and the extent.

## **Funding Transparency Statement**

### **Funding**

The authors were supported by the U.S. Army Research Laboratory and the U.K. Ministry of Defence under agreement number W911NF-16-3-0001 and by the Engineering and Physical Sciences Research Council grant number EP/L015242/1.

### **Competing Interests**

The authors assert no competing interests.

## References

- [1] J. Bennett and S. Lanning. The netflix prize. In *Proceedings of the KDD Cup Workshop 2007*, pages 3–6, New York, August 2007. ACM.
- [2] J. A. Hartigan. Direct Clustering of a Data Matrix. *Journal of the American Statistical Association*, 67(337):123–129, 1972.
- [3] C. Gao, Y. Lu, Z. Ma, and H. H. Zhou. Optimal estimation and completion of matrices with biclustering structures. *Journal of Machine Learning Research*, 17:161:1–161:29, 2016.
- [4] E. Candès and B. Recht. Exact matrix completion via convex optimization. *Commun. ACM*, 55(6):111–119, June 2012.
- [5] S. Ben-David, N. Eiron, and H. U. Simon. Limitations of learning via embeddings in euclidean half spaces. *Journal of Machine Learning Research*, 3:441–461, 2003.
- [6] N. Linial, S. Mendelson, G. Schechtman, and A. Shraibman. Complexity measures of sign matrices. *Combinatorica*, 27(4):439–463, 2007.
- [7] N. Srebro and A. Shraibman. Rank, trace-norm and max-norm. In *Proceedings of the 18th Annual Conference on Learning Theory*, pages 545–560, 2005.
- [8] N. Srebro, J. D. M. Rennie, and T. S. Jaakkola. Maximum-margin matrix factorization. *Advances in Neural Information Processing Systems 17*, 2005.
- [9] E. J. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Trans. Inf. Theor.*, 56(5):2053–2080, May 2010.
- [10] A. Maurer and M. Pontil. Excess risk bounds for multitask learning with trace norm regularization. In *Proceedings of The 27th Conference on Learning Theory*, pages 55–76, 2013.
- [11] Kai Yang Chiang, Inderjit S. Dhillon, and Cho Jui Hsieh. Using side information to reliably learn low-rank matrices from missing and corrupted observations. *Journal of Machine Learning Research*, 19, 2018.
- [12] M. Fazel, H. Hindi, and S. P. Boyd. A rank minimization heuristic with application to minimum orders system approximation. *Proceedings of the American Control Conference*, 2001.
- [13] J. Abernethy, F. Bach, T. Evgeniou, and J. Vert. Low-rank matrix factorization with attributes. In *ArXiv preprint ArXiv: cs/0611124*, 2006.
- [14] M Xu, R Jin, and Z. H. Zhou. Speedup matrix completion with side information: Application to multi-label learning. In *Advances in Neural Information Processing Systems*, 2013.
- [15] V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst. Matrix completion on graphs. Technical report, EPFL, 2014.
- [16] N. Rao, P. Yu, H.-F.; Ravikumar, and I. Dhillon. Collaborative Filtering with Graph Information: Consistency and Scalable Methods. In *Advances in Neural Information Processing Systems*, 2015.
- [17] X. Zhang, S. S. Du, and Q. Gu. Fast and Sample Efficient Inductive Matrix Completion via Multi-Phase Procrustes Flow. In *Proceedings of Machine Learning Research*, 2018.
- [18] S. A. Goldman, R. L. Rivest, and R. E. Schapire. Learning binary relations and total orders. *SIAM J. Comput.*, 22(5), 1993.
- [19] S. A. Goldman and M. K. Warmuth. Learning binary relations using weighted majority voting. In *Proceedings of the 6th Annual Conference on Computational Learning Theory*, pages 453–462, 1993.
- [20] N. Cesa-Bianchi and O. Shamir. Efficient online learning via randomized rounding. In *Advances in Neural Information Processing Systems 24*, pages 343–351, 2011.

- [21] E. Hazan, S. Kale, and S. Shalev-Shwartz. Near-optimal algorithms for online matrix prediction. In *Proceedings of the 23rd Annual Conference on Learning Theory*, volume 23:38.1-38.13, 2012.
- [22] C. Gentile, M. Herbster, and S. Pasteris. Online similarity prediction of networked data from known and unknown graphs. In *Proceedings of the 26th Annual Conference on Learning Theory*, 2013.
- [23] M. Herbster, S. Pasteris, and S. Pontil. Predicting a switching sequence of graph labelings. *Journal of Machine Learning Research*, 16:2003–2022, 2015.
- [24] M. Herbster, S. Pasteris, and M. Pontil. Mistake bounds for binary matrix completion. In *Advances in Neural Information Processing Systems 29*, pages 3954–3962. 2016.
- [25] K. Tsuda, G. Rätsch, and M.K. Warmuth. Matrix exponentiated gradient updates for on-line learning and bregman projection. *Journal of Machine Learning Research*, 6:995–1018, 2005.
- [26] N. Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2:285–318, April 1988.
- [27] S. Sabato, S. Shalev-Shwartz, N. Srebro, Daniel J. Hsu, and T. Zhang. Learning sparse low-threshold linear classifiers. *Journal of Machine Learning Research*, 16:1275–1304, 2015.
- [28] M. Herbster and M. Pontil. Prediction on a graph with a perceptron. In *Advances in Neural Information Processing Systems 19*, pages 577–584, 2006.
- [29] Robert Galian, Iyad A. Kanj, Sebastian Ordyniak, and Stefan Szeider. Parameterized algorithms for the matrix completion problem. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pages 1642–1651, 2018.
- [30] A.B. Novikoff. On convergence proofs on perceptrons. In *Proceedings of the Symposium on the Mathematical Theory of Automata*, pages 615–622, 1962.
- [31] M. Belkin and P. Niyogi. Semi-supervised learning on riemannian manifolds. *Machine Learning*, 56:209–239, 2004.
- [32] Xiaojin Zhu and Andrew B. Goldberg. Introduction to semi-supervised learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 2009.
- [33] M. Herbster, M. Pontil, and L. Wainer. Online learning over graphs. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 305–312, 2005.
- [34] M. K. Warmuth, W. Kotłowski, and S. Zhou. Kernelization of matrix updates, when and how? In *Algorithmic Learning Theory*, pages 350–364, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.
- [35] R. Bhatia. *Matrix Analysis*. Springer Verlag, New York, 1997.
- [36] S. Shalev-Shwartz. Online Learning and Online Convex Optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2011.

## A Synthetic Experiments

To illustrate the algorithm’s performance, synthetic experiments were performed in the transductive setting with graph side information. In particular, we took  $U$  to be randomly generated square (9,9)-biclustered matrices with i.i.d. noise. A visualization of a noise-free example matrix can be found in Figure 1. The noise process flipped the label of each matrix entry independently with probability  $p = 0.10$ . The side information on the rows and columns were represented by PDLaplacian matrices, for which the underlying graphs were constructed in the manner described in Section 4.1. Varying levels of side information noise  $\beta \in [0.0, 0.5]$  were applied. This was introduced by considering every pair of vertices independently from the constructed graph and flipping the state between EDGE/NOT-EDGE with probability  $\beta$ . A final step is added to ensure the graph is connected. In this step a random pair of components is connected by a random edge, recursively. The process terminates when the graph is connected.

The parameters were chosen so that the expected regret bound in Theorem 1 would apply to our experimental setting. We use the quasi-dimension upper bound  $\widehat{D} := \mathcal{D}_{M,N}^{\circ}(U) = 2 \operatorname{tr}(R^T MR) \mathcal{R}_M + 2 \operatorname{tr}(C^T NC) \mathcal{R}_N + 4k$ , as developed in Theorem 3 for PDLaplacians. The learning rate was set as  $\eta = \sqrt{\frac{\widehat{D} \log(2n)}{2T}}$ . Since each run of the algorithm consisted of predicting all  $n^2$  matrix entries sampled uniformly at random without replacement, we set  $T = n^2$ . As for the margin estimate, due to the requirement that  $\gamma \leq 1/\|U\|_{\max}$ , a suitable value can be extracted from Equation (7), giving  $\gamma = 1/\sqrt{k}$ .

The per trial mistake rate is shown in Fig. 2 for matrix dimension  $n = 20, \dots, 400$ , where each data point is averaged over 10 runs. We observe that for random side information  $\beta = 0.5$ , the term  $\widehat{D}$  could lead to a bound which is vacuous (for small  $n$ ), however, the algorithm’s error rate was in the range of  $[0.30, 0.45]$ , being well below chance. With ideal side information,  $\beta = 0.0$ , the performance improved drastically, as suggested by the bounds, to an error rate in  $[0.10, 0.35]$ . Observe that since there is 10% label noise for all values of  $\beta$ , the curves are converging to an online mistake rate of 10%. The data points for the plot can be found in Table 1. For our specific implementation of  $n = 400$ , each run required approximately 6 hours on a 3.40GHz Intel(R) Xeon(R) CPU E3-1240 v3 with 4GB of RAM. Note that this was run on a shared node on a cluster, meaning that the run time may vary depending on the other jobs that were running.

## B Proof of Theorem 1

The proof of Theorem 1 is organized as follows. We start with the required preliminaries in Subsection B.1, and then proceed to prove the regret statement of the theorem, given by Equation (5), in Subsection B.2. Finally, in Subsection B.3, we provide a proof for the mistake bound in the realizable case, as stated in Equation (6).

### B.1 Preliminaries for Proof

Suppose we have  $M$ ,  $N$  and  $U$  as in Theorem 1. Instead of working with  $U$  directly, we shall work with an embedding of this matrix. We have different treatments for the embedding of  $U$  in the two parts of the proof. For Subsection B.2, let  $\bar{U} \in \mathbb{R}^{m \times n}$  be such that  $\bar{U} = \gamma U$ . Following from the assumption that  $\|U\|_{\max} \leq \frac{1}{\gamma}$ , there exist row-normalized matrices  $\hat{P} \in \mathbb{R}^{n \times d}$  and  $\hat{Q} \in \mathbb{R}^{m \times d}$  that give  $\bar{U} = \hat{P} \hat{Q}^T$ . In Subsection B.3, however, we have that  $\bar{U} = \gamma \operatorname{argmin}_{V \in \mathbb{SP}^1(U)} \mathcal{D}_{M,N}^{\gamma}(V)$ .

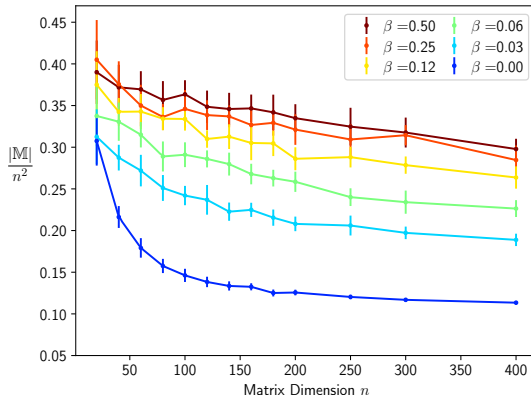


Figure 2: Error rates for predicting a noisy (9,9)-biclustered matrix with side information.

Table 1: Data points used for Fig. 2.

Matrix Dimensions $n$	Noise					
	0.50	0.25	0.125	0.0625	0.03125	0.00
20	0.39± 0.04	0.4± 0.05	0.38± 0.04	0.34± 0.04	0.31± 0.03	0.31± 0.03
40	0.37± 0.03	0.38± 0.03	0.34± 0.02	0.33± 0.02	0.29± 0.02	0.22± 0.01
60	0.37± 0.02	0.35± 0.02	0.34± 0.02	0.32± 0.02	0.27± 0.02	0.18± 0.01
80	0.36± 0.02	0.34± 0.02	0.33± 0.01	0.29± 0.02	0.25± 0.02	0.16± 0.01
100	0.36± 0.02	0.35± 0.02	0.33± 0.01	0.29± 0.01	0.24± 0.01	0.15± 0.01
120	0.35± 0.02	0.34± 0.02	0.31± 0.01	0.29± 0.01	0.24± 0.02	0.14± 0.01
140	0.35± 0.02	0.34± 0.01	0.31± 0.01	0.28± 0.01	0.22± 0.01	0.13± 0.01
160	0.35± 0.02	0.33± 0.02	0.31± 0.02	0.27± 0.01	0.22± 0.01	0.13± 0.0
180	0.34± 0.02	0.33± 0.02	0.3± 0.02	0.26± 0.01	0.22± 0.01	0.13± 0.0
200	0.33± 0.02	0.32± 0.02	0.29± 0.01	0.26± 0.01	0.21± 0.01	0.13± 0.0
250	0.32± 0.02	0.31± 0.02	0.29± 0.01	0.24± 0.01	0.21± 0.01	0.12± 0.0
300	0.32± 0.02	0.31± 0.01	0.28± 0.01	0.23± 0.01	0.2± 0.01	0.12± 0.0
400	0.3± 0.01	0.28± 0.02	0.26± 0.01	0.23± 0.01	0.19± 0.01	0.11± 0.0

In this section we assume  $\text{mc}(\mathbf{U}) \leq \frac{1}{\gamma}$ , which also guarantees a decomposition in terms of the row-normalized matrices  $\hat{\mathbf{P}}$  and  $\hat{\mathbf{Q}}$  such that  $\bar{\mathbf{U}} = \hat{\mathbf{P}}\hat{\mathbf{Q}}^\top$ .

Let us then define the quasi-dimension with respect to a specific factorization

$$\mathcal{D} := \mathcal{R}_M \text{tr} \left( \hat{\mathbf{P}}^\top \mathbf{M} \hat{\mathbf{P}} \right) + \mathcal{R}_N \text{tr} \left( \hat{\mathbf{Q}}^\top \mathbf{N} \hat{\mathbf{Q}} \right).$$

Note that in general for the case that  $\bar{\mathbf{U}} = \gamma \mathbf{U}$ ,  $\mathcal{D} \geq \mathcal{D}_{M,N}^\gamma(\mathbf{U})$ , whereas for the case that  $\bar{\mathbf{U}} \in \gamma \text{SP}^1(\mathbf{U})$ ,  $\mathcal{D} \geq \min_{\mathbf{V} \in \text{SP}^1(\mathbf{U})} \mathcal{D}_{M,N}^\gamma(\mathbf{V})$ . We proceed with the proof assuming that  $(\hat{\mathbf{P}}, \hat{\mathbf{Q}})$  is the optimal factorization for a given  $\bar{\mathbf{U}}$ . That is, for  $\bar{\mathbf{U}} = \gamma \mathbf{U}$ , we have that  $(\hat{\mathbf{P}}, \hat{\mathbf{Q}})$  is the factorization that satisfies  $\mathcal{D} = \mathcal{D}_{M,N}^\gamma(\mathbf{U})$ , and for  $\bar{\mathbf{U}} = \gamma \arg\min_{\mathbf{V} \in \text{SP}^1(\mathbf{U})} \mathcal{D}_{M,N}^\gamma(\mathbf{V})$ ,  $(\hat{\mathbf{P}}, \hat{\mathbf{Q}})$  satisfies

$$\mathcal{D} = \min_{\mathbf{V} \in \text{SP}^1(\mathbf{U})} \mathcal{D}_{M,N}^\gamma(\mathbf{V}).$$

Next, we define  $\tilde{\mathbf{U}}$ , which is a positive semidefinite matrix, used as an embedding for  $\bar{\mathbf{U}}$  in the analysis of the algorithm. Its exact relationship with  $\bar{\mathbf{U}}$  is shown in Lemma 7.

**Definition 6.** Define the  $(m+n) \times (m+n)$  matrix  $\mathbf{Z}$  as

$$\mathbf{Z} := \begin{pmatrix} \sqrt{\mathcal{R}_M} \sqrt{\mathbf{M}} \hat{\mathbf{P}} \\ \sqrt{\mathcal{R}_N} \sqrt{\mathbf{N}} \hat{\mathbf{Q}} \end{pmatrix}. \quad (9)$$

and construct  $\tilde{\mathbf{U}}$  as,

$$\tilde{\mathbf{U}} := \mathbf{Z} \mathbf{Z}^\top = \begin{pmatrix} \mathcal{R}_M \sqrt{\mathbf{M}} \hat{\mathbf{P}} \hat{\mathbf{P}}^\top \sqrt{\mathbf{M}} & \sqrt{\mathcal{R}_M} \mathcal{R}_N \sqrt{\mathbf{M}} \hat{\mathbf{P}} \hat{\mathbf{Q}}^\top \sqrt{\mathbf{N}} \\ \sqrt{\mathcal{R}_M} \mathcal{R}_N \sqrt{\mathbf{N}} \hat{\mathbf{Q}} \hat{\mathbf{P}}^\top \sqrt{\mathbf{M}} & \mathcal{R}_N \sqrt{\mathbf{N}} \hat{\mathbf{Q}} \hat{\mathbf{Q}}^\top \sqrt{\mathbf{N}} \end{pmatrix}.$$

**Lemma 7.** For all trials  $t \in [T]$ ,

$$\bar{U}_{i_t j_t} = \text{tr} \left( \tilde{\mathbf{U}} \tilde{\mathbf{X}}^t \right) - 1$$

where  $\tilde{\mathbf{U}}$  is as constructed from Definition 6.

*Proof.* We have:

$$\text{tr} \left( \tilde{\mathbf{U}} \tilde{\mathbf{X}}^t \right) = (\mathbf{x}^t)^\top \tilde{\mathbf{U}} \mathbf{x}^t \quad (10)$$

$$= (\mathbf{x}^t)^\top \mathbf{Z} \mathbf{Z}^\top \mathbf{x}^t \quad (11)$$

$$= \left\| (\mathbf{x}^t)^\top \mathbf{Z} \right\|^2. \quad (12)$$

Recall that

$$\mathbf{x}^t = \left[ \frac{\sqrt{M^+} \mathbf{e}_m^{it}}{\sqrt{2\mathcal{R}_M}}; \frac{\sqrt{N^+} \mathbf{e}_n^{jt}}{\sqrt{2\mathcal{R}_N}} \right] \text{ and } \mathbf{Z} = \begin{pmatrix} \sqrt{\mathcal{R}_M} \sqrt{M} \hat{\mathbf{P}} \\ \sqrt{\mathcal{R}_N} \sqrt{N} \hat{\mathbf{Q}} \end{pmatrix}$$

Hence,

$$\begin{aligned} (\mathbf{x}^t)^\top \mathbf{Z} &= \frac{(\sqrt{M^+} \mathbf{e}_m^{it})^\top}{\sqrt{2\mathcal{R}_M}} \sqrt{\mathcal{R}_M} \sqrt{M} \hat{\mathbf{P}} + \frac{(\sqrt{N^+} \mathbf{e}_n^{jt})^\top}{\sqrt{2\mathcal{R}_N}} \sqrt{\mathcal{R}_N} \sqrt{N} \hat{\mathbf{Q}} \\ &= \frac{1}{\sqrt{2}} (\mathbf{e}_m^{it})^\top \sqrt{M^+} \sqrt{M} \hat{\mathbf{P}} + \frac{1}{\sqrt{2}} (\mathbf{e}_n^{jt})^\top \sqrt{N^+} \sqrt{N} \hat{\mathbf{Q}} \\ &= \frac{1}{\sqrt{2}} (\hat{\mathbf{P}}_{it} + \hat{\mathbf{Q}}_{jt}) \end{aligned} \quad (13)$$

Thus substituting (13) into (12) gives,

$$\begin{aligned} \text{tr}(\tilde{\mathbf{U}} \tilde{\mathbf{X}}^t) &= \frac{1}{2} \left\| \hat{\mathbf{P}}_{it} + \hat{\mathbf{Q}}_{jt} \right\|^2 \\ &= \frac{1}{2} \left( \left\| \hat{\mathbf{P}}_{it} \right\|^2 + 2 \langle \hat{\mathbf{P}}_{it}, \hat{\mathbf{Q}}_{jt} \rangle + \left\| \hat{\mathbf{Q}}_{jt} \right\|^2 \right) \\ &= \left( 1 + \langle \hat{\mathbf{P}}_{it}, \hat{\mathbf{Q}}_{jt} \rangle \right) \\ &= 1 + \tilde{U}_{i,jt}. \end{aligned}$$

□

In the subsequent proofs, we will also need to make use of the following facts.

**Lemma 8.** For  $\tilde{\mathbf{U}}$  as defined in Definition 6, we have that,

$$\text{tr}(\tilde{\mathbf{U}}) = \mathcal{D}. \quad (14)$$

*Proof.*

$$\begin{aligned} \text{tr}(\tilde{\mathbf{U}}) &= \text{tr}(\mathbf{Z} \mathbf{Z}^\top) = \text{tr} \left( \begin{pmatrix} \sqrt{\mathcal{R}_M} \sqrt{M} \hat{\mathbf{P}} \\ \sqrt{\mathcal{R}_N} \sqrt{N} \hat{\mathbf{Q}} \end{pmatrix} \begin{pmatrix} \sqrt{\mathcal{R}_M} \sqrt{M} \hat{\mathbf{P}} & \sqrt{\mathcal{R}_N} \sqrt{N} \hat{\mathbf{Q}} \end{pmatrix}^\top \right) \\ &= \mathcal{R}_M \text{tr} \left( \sqrt{M} \hat{\mathbf{P}} \hat{\mathbf{P}}^\top \sqrt{M}^\top \right) + \mathcal{R}_N \text{tr} \left( \sqrt{N} \hat{\mathbf{Q}} \hat{\mathbf{Q}}^\top \sqrt{N}^\top \right) \\ &= \mathcal{R}_M \text{tr} \left( \hat{\mathbf{P}}^\top M \hat{\mathbf{P}} \right) + \mathcal{R}_N \text{tr} \left( \hat{\mathbf{Q}}^\top N \hat{\mathbf{Q}} \right) \\ &= \mathcal{D} \end{aligned}$$

□

**Lemma 9.** For all trials  $t$ , all eigenvalues of  $\tilde{\mathbf{X}}^t$  are in  $[0, 1]$ .

*Proof.* Recall that

$$\text{tr}(\tilde{\mathbf{X}}^t) = \text{tr}(\mathbf{x}^t (\mathbf{x}^t)^\top) = \left[ \frac{\sqrt{M^+} \mathbf{e}_m^{it}}{\sqrt{2\mathcal{R}_M}}; \frac{\sqrt{N^+} \mathbf{e}_n^{jt}}{\sqrt{2\mathcal{R}_N}} \right]^\top \left[ \frac{\sqrt{M^+} \mathbf{e}_m^{it}}{\sqrt{2\mathcal{R}_M}}; \frac{\sqrt{N^+} \mathbf{e}_n^{jt}}{\sqrt{2\mathcal{R}_N}} \right].$$

Hence

$$\left\| \mathbf{x}^t \right\|^2 = \left\| \frac{\sqrt{M^+} \mathbf{e}_m^{it}}{\sqrt{2\mathcal{R}_M}} \right\|^2 + \left\| \frac{\sqrt{N^+} \mathbf{e}_n^{jt}}{\sqrt{2\mathcal{R}_N}} \right\|^2$$



and then bounding the first term on the right hand side gives,

$$\left\| \frac{\sqrt{M^+} e_m^{i_t}}{\sqrt{2\mathcal{R}_M}} \right\|^2 = \frac{1}{2\mathcal{R}_M} (e_m^{i_t})^\top (\sqrt{M^+})^\top \sqrt{M^+} e_m^{i_t} \leq \frac{1}{2\mathcal{R}_M} \max_{i \in [m]} (e_m^{i_t})^\top M^+ e_m^{i_t} = \frac{1}{2}.$$

The argument for the second term is parallel. Therefore since it is shown that the trace of  $\tilde{\mathbf{X}}^t$  is bounded by 1 and that  $\tilde{\mathbf{X}}^t$  is positive definite, this implies that all eigenvalues of  $\tilde{\mathbf{X}}^t$  are in  $[0, 1]$ .  $\square$

Next, we introduce the following quantity, which plays a central role in the amortized analysis of our algorithm.

**Definition 10.** *The quantum relative entropy of symmetric positive semidefinite square matrices  $\mathbf{A}$  and  $\mathbf{B}$  is*

$$\Delta(\mathbf{A}, \mathbf{B}) := \text{tr}(\mathbf{A} \log(\mathbf{A}) - \mathbf{A} \log(\mathbf{B}) + \mathbf{B} - \mathbf{A}).$$

An important result that will be used in the subsequent subsections is the well known Golden-Thompson Inequality, whose proof can be found, for example, in [35].

**Lemma 11.** *For any symmetric matrices  $\mathbf{A}$  and  $\mathbf{B}$  we have,*

$$\text{tr}(\exp(\mathbf{A} + \mathbf{B})) \leq \text{tr}(\exp(\mathbf{A}) \exp(\mathbf{B})).$$

## B.2 Proof for the Regret Statement

In this subsection, we prove the regret bound as presented in Theorem 1, which holds for Algorithm 1 with **non-conservative** updates. To do so, we first derive a regret bound in terms of the hinge loss with the deterministic prediction  $\bar{y}_t$ . We then convert this to an expected regret bound in terms of the zero-one loss for the random variable  $\hat{y}_t$ .

Regret bounds for the MEG algorithm were originally proven in [25]. However, that analysis leads to a  $\text{tr}(\tilde{\mathbf{U}})$  dependence, whereas we derive a  $\sqrt{\text{tr}(\tilde{\mathbf{U}})}$  scaling, for our more restrictive setting. Regret bounds with such scaling for linear classification in the vector case have been previously given in [27] (which themselves are generalisations of the bounds from Littlestone [26] for learning  $k$ -literal disjunctions with  $\mathcal{O}(k \log n)$  mistakes). However, to our knowledge, no such regret bounds for MEG are present in the literature for the matrix case. Our proof uses an amortized analysis of the quantum relative entropy, followed by an application of the matricized results in [27]. The original results in [27] are built on the results in [36], which use convex arguments. We eliminate the need to introduce convex concepts to our proof by instead using an amortized analysis. We conclude the proof by using algebraic arguments to transform the regret in terms of the hinge loss to an expected regret in terms of the zero-one loss.

In the following, we define the hinge loss as  $h_\gamma(y, \bar{y}) := \frac{1}{\gamma} [\gamma - y\bar{y}]_+$ . We define

$$\mathbf{H}^t := \nabla_{\tilde{\mathbf{W}}} \gamma h_\gamma(y_t, \bar{y}_t), \quad (15)$$

where  $\nabla$  denotes the subgradient and where  $\bar{y}_t$  is as defined in Algorithm 1. When  $y_t \bar{y}_t = \gamma$ , we will only consider the specific subgradient  $\mathbf{H}^t = \mathbf{0}$ .

**Lemma 12.** *For all  $t \in [T]$ ,*

$$\mathbf{H}^t = -y_t \tilde{\mathbf{X}}^t \left[ \gamma > y_t \left( \text{tr}(\tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t) - 1 \right) \right].$$

*Proof.* Recalling the definition of  $\mathbf{H}^t := \nabla_{\tilde{\mathbf{W}}} \gamma h_\gamma(y_t, \bar{y}_t)$ , observe that when  $\gamma > y_t \left( \text{tr}(\tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t) - 1 \right)$ , we have

$$\nabla_{\tilde{\mathbf{W}}} \gamma h_\gamma(y_t, \bar{y}_t) = \nabla_{\tilde{\mathbf{W}}} \left[ \gamma - y_t \left( \text{tr}(\tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t) - 1 \right) \right]_+ = -y_t (\tilde{\mathbf{X}}^t)^\top = -y_t \tilde{\mathbf{X}}^t, \quad (16)$$

where we used the fact that  $\nabla_{\mathbf{A}} \text{tr}(\mathbf{A}\mathbf{B}) = \mathbf{B}^\top$ . In the case that  $\gamma \leq y_t \left( \text{tr}(\tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t) - 1 \right)$ ,

$$\nabla_{\tilde{\mathbf{W}}} \gamma h_\gamma(y_t, \bar{y}_t) = \mathbf{0}.$$

$\square$

**Lemma 13.** For matrix  $\mathbf{A} \in \mathbf{S}^d$  with eigenvalues no less than  $-1$ ,

$$\mathbf{I} - \mathbf{A} + \mathbf{A}^2 - \exp(-\mathbf{A}) \succeq \mathbf{0}.$$

*Proof.* Let  $\mathbf{B} := \mathbf{I} - \mathbf{A} + \mathbf{A}^2 - \exp(-\mathbf{A})$ . Observing that  $\mathbf{A}$ ,  $\mathbf{A}^2$  and  $\exp(-\mathbf{A})$  share the same set of eigenvectors,

$$\mathbf{I} - \mathbf{A} + \mathbf{A}^2 - \exp(-\mathbf{A}) = \mathbf{U}(\mathbf{I} - \mathbf{\Lambda} + \mathbf{\Lambda}^2 - \exp(-\mathbf{\Lambda}))\mathbf{U}^\top,$$

where  $\mathbf{U}$  is the orthogonal matrix and  $\mathbf{\Lambda}$  is the diagonal matrix in the eigendecomposition of  $\mathbf{A}$ . Therefore, each eigenvalue  $\lambda_{\mathbf{B},i}$  of the resulting matrix  $\mathbf{B}$  can be written in terms of an eigenvalue  $\lambda_{\mathbf{A},i}$  of matrix  $\mathbf{A}$  for all  $i \in [d]$ ,

$$\lambda_{\mathbf{B},i} = 1 - \lambda_{\mathbf{A},i} + \lambda_{\mathbf{A},i}^2 - \exp(-\lambda_{\mathbf{A},i}).$$

**Lemma 14.** For any matrix  $\mathbf{A} \in \mathbf{S}_{++}^d$  and any two matrices  $\mathbf{B}, \mathbf{C} \in \mathbf{S}^d$ ,  $\mathbf{B} \preceq \mathbf{C}$  implies  $\text{tr}(\mathbf{A}\mathbf{B}) \leq \text{tr}(\mathbf{A}\mathbf{C})$ .

*Proof.* This follows a parallel argument to the proof for [25, Lemma 2.2].  $\square$

The positive semidefinite criterion requires that all eigenvalues be non-negative, so that  $\lambda_{\mathbf{B},i} \geq 0$ . This inequality holds true for  $\lambda_{\mathbf{A},i} \geq -1$ .  $\square$

**Lemma 15.** For all trials  $t \in [T]$  in Algorithm 1, we have for  $\eta \in (0, 1]$ :

$$\Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^t) - \Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^{t+1}) \geq \eta \left( \text{tr}((\tilde{\mathbf{W}}^t - \tilde{\mathbf{U}})\mathbf{H}^t) - \eta^2 \text{tr}(\tilde{\mathbf{W}}^t(\mathbf{H}^t)^2) \right). \quad (17)$$

*Proof.* We have:

$$\begin{aligned} \Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^t) - \Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^{t+1}) &= \text{tr} \left( \tilde{\mathbf{U}} \log \tilde{\mathbf{W}}^{t+1} - \tilde{\mathbf{U}} \log \tilde{\mathbf{W}}^t \right) + \text{tr} \left( \tilde{\mathbf{W}}^t \right) - \text{tr} \left( \tilde{\mathbf{W}}^{t+1} \right) \\ &= -\eta \text{tr} \left( \tilde{\mathbf{U}} \mathbf{H}^t \right) + \text{tr} \left( \tilde{\mathbf{W}}^t \right) - \text{tr} \left( e^{\log \tilde{\mathbf{W}}^t - \eta \mathbf{H}^t} \right) \end{aligned} \quad (18)$$

$$\geq -\eta \text{tr} \left( \tilde{\mathbf{U}} \mathbf{H}^t \right) + \text{tr} \left( \tilde{\mathbf{W}}^t \right) - \text{tr} \left( e^{\log \tilde{\mathbf{W}}^t} e^{-\eta \mathbf{H}^t} \right) \quad (19)$$

$$\begin{aligned} &= -\eta \text{tr} \left( \tilde{\mathbf{U}} \mathbf{H}^t \right) + \text{tr} \left( \tilde{\mathbf{W}}^t \left( \mathbf{I} - e^{-\eta \mathbf{H}^t} \right) \right) \\ &\geq -\eta \text{tr} \left( \tilde{\mathbf{U}} \mathbf{H}^t \right) + \text{tr} \left( \tilde{\mathbf{W}}^t \left( \eta \mathbf{H}^t - \eta^2 (\mathbf{H}^t)^2 \right) \right) \end{aligned} \quad (20)$$

where Equation (18) comes from the update of the algorithm and Lemma 12, Equation (19) comes from Lemma 11 and Equation (20) comes from Lemmas 13 and 14.  $\square$

**Lemma 16.** For Algorithm 1, we have for  $\eta \in (0, 1]$ :

$$\sum_{t=1}^T \text{tr} \left( (\tilde{\mathbf{W}}^t - \tilde{\mathbf{U}}) \mathbf{H}^t \right) \leq \frac{1}{\eta} \left( \text{tr} \left( \tilde{\mathbf{U}} \log \left( \frac{\tilde{\mathbf{U}}(m+n)}{e^{\hat{\mathcal{D}}}} \right) \right) + \hat{\mathcal{D}} \right) + \sum_{t=1}^T \eta \text{tr} \left( \tilde{\mathbf{W}}^t (\mathbf{H}^t)^2 \right). \quad (21)$$

Setting the additional assumptions  $\hat{\mathcal{D}} \geq \text{tr} \left( \tilde{\mathbf{U}} \right) \geq 1$  and  $m+n \geq 3$  gives

$$\sum_{t=1}^T \text{tr} \left( (\tilde{\mathbf{W}}^t - \tilde{\mathbf{U}}) \mathbf{H}^t \right) \leq \frac{\hat{\mathcal{D}}}{\eta} \log(m+n) + \sum_{t=1}^T \eta \text{tr} \left( \tilde{\mathbf{W}}^t (\mathbf{H}^t)^2 \right). \quad (22)$$

*Proof.* We start by proving Equation (21). Rearranging Lemma 15 and summing over  $t$ ,

$$\begin{aligned} \sum_{t=1}^T \left( \text{tr}((\tilde{\mathbf{W}}^t - \tilde{\mathbf{U}})\mathbf{H}^t) \right) &\leq \frac{1}{\eta} \left( \sum_{t=1}^T \Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^t) - \Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^{T+1}) + \eta^2 \text{tr} \left( \tilde{\mathbf{W}}^t (\mathbf{H}^t)^2 \right) \right) \\ &\leq \frac{1}{\eta} \left( \Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^1) - \Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^{T+1}) + \sum_{t=1}^T \eta^2 \text{tr} \left( \tilde{\mathbf{W}}^t (\mathbf{H}^t)^2 \right) \right). \end{aligned}$$

Using the fact that  $\Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^{T+1}) \geq 0$  and writing out  $\Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^1)$ , we then obtain Equation (21).

To prove Equation (22), we attempt to maximize the term  $\text{tr} \left( \tilde{\mathbf{U}} \log \left( \frac{\tilde{\mathbf{U}}(m+n)}{e^{\hat{\mathcal{D}}}} \right) \right) + \hat{\mathcal{D}}$ . Noting that  $\text{tr} \left( \tilde{\mathbf{U}} \log(a\tilde{\mathbf{U}}) \right) \leq \text{tr}(\tilde{\mathbf{U}}) \log \left( \text{tr}(a\tilde{\mathbf{U}}) \right)$  for  $a \geq 0$ , we then have

$$\text{tr} \left( \tilde{\mathbf{U}} \log \left( \frac{\tilde{\mathbf{U}}(m+n)}{e^{\hat{\mathcal{D}}}} \right) \right) + \hat{\mathcal{D}} \leq \text{tr}(\tilde{\mathbf{U}}) \log \left( \frac{\text{tr}(\tilde{\mathbf{U}})(m+n)}{e^{\hat{\mathcal{D}}}} \right) + \hat{\mathcal{D}}.$$

The upper bound in the above equation is convex in  $\text{tr}(\tilde{\mathbf{U}})$  and hence is maximized at either boundary  $\{1, \hat{\mathcal{D}}\}$ . Comparing the terms, we have

$$\text{tr} \left( \tilde{\mathbf{U}} \log \left( \frac{\tilde{\mathbf{U}}(m+n)}{e^{\hat{\mathcal{D}}}} \right) \right) + \hat{\mathcal{D}} \leq \log \left( \frac{m+n}{e^{\hat{\mathcal{D}}}} \right) + \hat{\mathcal{D}}$$

for  $\text{tr}(\tilde{\mathbf{U}}) = 1$  and

$$\text{tr} \left( \tilde{\mathbf{U}} \log \left( \frac{\tilde{\mathbf{U}}(m+n)}{e^{\hat{\mathcal{D}}}} \right) \right) + \hat{\mathcal{D}} \leq \hat{\mathcal{D}} \log(m+n)$$

for  $\text{tr}(\tilde{\mathbf{U}}) = \hat{\mathcal{D}}$ . We then observe that given the assumptions,  $\hat{\mathcal{D}} \log(m+n)$  maximizes, therefore giving the upper bound in Equation (22).  $\square$

**Lemma 17.** *The following condition is satisfied for Algorithm 1:*

$$\text{tr} \left( \tilde{\mathbf{W}}^t (\mathbf{H}^t)^2 \right) \leq \gamma h_\gamma(y_t, \bar{y}_t) + \gamma + 1. \quad (23)$$

*Proof.* The proof splits into two cases.

Case 1)  $\gamma \leq y_t \bar{y}_t = y_t \left( \text{tr} \left( \tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t \right) - 1 \right)$ :

Observe that  $\mathbf{H}^t = \mathbf{0}$  due to Lemma 12, giving  $\text{tr} \left( \tilde{\mathbf{W}}^t (\mathbf{H}^t)^2 \right) = 0$ , which demonstrates (23) in this case.

Case 2)  $\gamma > y_t \bar{y}_t = y_t \left( \text{tr} \left( \tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t \right) - 1 \right)$ :

We have that

$$\text{tr} \left( \tilde{\mathbf{W}}^t (\mathbf{H}^t)^2 \right) = \text{tr} \left( \tilde{\mathbf{W}}^t (\tilde{\mathbf{X}}^t)^2 \right) \leq \text{tr} \left( \tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t \right), \quad (24)$$

where the first equality comes from the fact  $\mathbf{H}^t = -y_t \tilde{\mathbf{X}}^t$  from Lemma 12 and the second inequality comes from Lemma 14 and the fact that  $(\tilde{\mathbf{X}}^t)^2 \preceq \tilde{\mathbf{X}}^t$  due to Lemma 9.

We split case 2 into two further subcases.

Sub-case 1)  $\text{tr} \left( \tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t \right) < \gamma + 1$  (Prediction smaller than margin):

Since we have

$$\text{tr} \left( \tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t \right) < \gamma h_\gamma(y_t, \bar{y}_t) + \gamma + 1,$$

lower bounding the L.H.S. by (24) demonstrates (23).

Sub-case 2)  $\text{tr} \left( \tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t \right) \geq \gamma + 1$  (Prediction larger than margin with mistake):

We have

$$\text{tr} \left( \tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t \right) \leq \left[ \text{tr} \left( \tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t \right) + \gamma - 1 \right]_+ - (\gamma - 1) \leq \left[ \text{tr} \left( \tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t \right) + \gamma - 1 \right]_+ + (\gamma + 1).$$

By the case 2 and sub-case 2 conditions we have that  $y_t = -1$ , with

$$\gamma h_\gamma(-1, \bar{y}_t) = \left[ \gamma + \text{tr} \left( \tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t \right) - 1 \right]_+.$$

Thus we have

$$\text{tr} \left( \tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t \right) \leq \gamma h_\gamma(-1, \bar{y}_t) + (\gamma + 1)$$

and by lower bounding L.H.S. by (24) we demonstrate (23) and thus the lemma.  $\square$

Now we are ready to introduce the regret bound in terms of the hinge loss for the deterministic  $\bar{y}_t$ .

**Lemma 18.** *The hinge loss of Algorithm 1 with parameters  $\gamma \in (0, 1]$ ,  $\hat{\mathcal{D}} \geq \mathcal{D} \geq 1$ ,  $\eta = \sqrt{\frac{\hat{\mathcal{D}} \log(m+n)}{2T}}$ ,  $T \geq 2\hat{\mathcal{D}} \log(m+n)$  and  $m+n \geq 3$ , is bounded by*

$$\sum_{t \in [T]} h_\gamma(y_t, \bar{y}_t) \leq \sum_{t \in [T]} h_\gamma(y_t, \bar{\mathbf{U}}_{i_t, j_t}) + \frac{4}{\gamma} \sqrt{2\hat{\mathcal{D}} \log(m+n)T} + \frac{4}{\gamma} \hat{\mathcal{D}} \log(m+n), \quad (25)$$

where  $\bar{\mathbf{U}}$ ,  $\mathcal{D}$  and their relationship are defined in the preliminaries of the proof.

*Proof.* Substituting for  $\bar{y}_t$  gives,

$$h_\gamma(y_t, \bar{y}_t) = \frac{1}{\gamma} [\gamma - y_t (\text{tr}(\tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t) - 1)]_+.$$

Lemma 7 gives,

$$h_\gamma(y_t, \bar{\mathbf{U}}_{i_t, j_t}) = \frac{1}{\gamma} [\gamma - y_t (\text{tr}(\tilde{\mathbf{U}} \tilde{\mathbf{X}}^t) - 1)]_+.$$

Define

$$f_t(\mathbf{Z}) := \frac{1}{\gamma} [\gamma - y_t (\text{tr}(\mathbf{Z} \tilde{\mathbf{X}}^t) - 1)]_+$$

Since  $h_\gamma(y_t, \cdot)$  is convex and the fact that a convex function applied to a linear function is again convex we have that  $f(\cdot)$  is convex. We have

$$\begin{aligned} \sum_{t=1}^T (h_\gamma(y_t, \bar{y}_t) - h_\gamma(y_t, \bar{\mathbf{U}}_{i_t, j_t})) &= \sum_{t=1}^T (f_t(\tilde{\mathbf{W}}^t) - f_t(\tilde{\mathbf{U}})) \\ &\leq \sum_{t=1}^T \text{tr} \left( (\tilde{\mathbf{W}}^t - \tilde{\mathbf{U}})^\top \nabla f_t(\tilde{\mathbf{W}}) \right) \\ &= \sum_{t=1}^T \text{tr} \left( (\tilde{\mathbf{W}}^t - \tilde{\mathbf{U}})^\top \nabla_{\tilde{\mathbf{W}}} h_\gamma(y_t, \bar{y}_t) \right) \\ &= \frac{1}{\gamma} \sum_{t=1}^T \text{tr} \left( (\tilde{\mathbf{W}}^t - \tilde{\mathbf{U}}) \mathbf{H}^t \right), \end{aligned} \quad (26)$$

where (26) follows from the fact that  $f(\mathbf{A}) - f(\mathbf{B}) \leq \text{tr}((\mathbf{A} - \mathbf{B})^\top \nabla f(\mathbf{A}))$  for a convex function  $f$  and (27) comes from the definition of  $\mathbf{H}^t = \nabla_{\tilde{\mathbf{W}}} \gamma h_\gamma(y_t, \bar{y}_t)$  (see (15)) and the fact that  $\tilde{\mathbf{W}}^t$  and  $\tilde{\mathbf{U}}$  are symmetric.

Therefore, we only need an upper bound to  $\sum_{t=1}^T \text{tr} \left( (\tilde{\mathbf{W}}^t - \tilde{\mathbf{U}}) \mathbf{H}^t \right)$ . We observe that  $\eta \in (0, \frac{1}{2}]$  due to the definition of  $\eta$  and the assumption on  $T$ . Thus we can apply (22) and (23) to obtain

$$\sum_{t=1}^T \text{tr} \left( (\tilde{\mathbf{W}}^t - \tilde{\mathbf{U}}) \mathbf{H}^t \right) \leq \frac{1}{\eta} \hat{\mathcal{D}} \log(m+n) + \eta \gamma \sum_{t=1}^T h_\gamma(y_t, \bar{y}_t) + \eta(1 + \gamma)T.$$

Substituting the above into (27) gives,

$$\begin{aligned} \sum_{t=1}^T h_\gamma(y_t, \bar{y}_t) &\leq \sum_{t=1}^T h_\gamma(y_t, \bar{\mathbf{U}}_{i_t, j_t}) + \frac{1}{\gamma} \left( \frac{1}{\eta} \hat{\mathcal{D}} \log(m+n) + \eta \gamma \sum_{t=1}^T h_\gamma(y_t, \bar{y}_t) + \eta(1 + \gamma)T \right) \\ &= \left( \frac{1}{1 - \eta} \right) \left( \sum_{t=1}^T h_\gamma(y_t, \bar{\mathbf{U}}_{i_t, j_t}) + \frac{1}{\eta \gamma} \hat{\mathcal{D}} \log(m+n) + \frac{\eta}{\gamma} (1 + \gamma)T \right) \end{aligned}$$

$$\leq \left( \frac{1}{1-\eta} \right) \left( \sum_{t=1}^T h_\gamma(y_t, \bar{U}_{i_t, j_t}) + \frac{1}{\eta\gamma} \widehat{\mathcal{D}} \log(m+n) + \frac{2\eta}{\gamma} T \right),$$

where the final inequality follows since  $\gamma \in (0, 1]$ .

We apply  $(1/(1-x)) \leq 1+2x$  for  $x \in [0, 1/2]$  to obtain

$$\begin{aligned} \sum_{t=1}^T h_\gamma(y_t, \bar{y}_t) &\leq (1+2\eta) \left( \sum_{t=1}^T h_\gamma(y_t, \bar{U}_{i_t, j_t}) + \frac{1}{\eta\gamma} \left( \widehat{\mathcal{D}} \log(m+n) \right) + \frac{2\eta}{\gamma} T \right) \\ &= \underbrace{(1+2\eta) \sum_{t=1}^T h_\gamma(y_t, \bar{U}_{i_t, j_t})}_{(1)} + \underbrace{\frac{1}{\eta\gamma} \widehat{\mathcal{D}} \log(m+n)}_{(2)} + \underbrace{\frac{2\eta}{\gamma} T}_{(3)} + \underbrace{\frac{2}{\gamma} \widehat{\mathcal{D}} \log(m+n)}_{(4)} + \underbrace{\frac{4\eta^2}{\gamma} T}_{(5)} \end{aligned}$$

Then, substituting the value for  $\eta$ ,

$$\begin{aligned} \sum_{t=1}^T h_\gamma(y_t, \bar{y}_t) &\leq \underbrace{\sum_{t=1}^T h_\gamma(y_t, \bar{U}_{i_t, j_t})}_{(a)} + \underbrace{\sqrt{\frac{2\widehat{\mathcal{D}} \log(m+n)}{T}} \sum_{t=1}^T h_\gamma(y_t, \bar{U}_{i_t, j_t})}_{(b)} \\ &\quad + \underbrace{\frac{4}{\gamma} \widehat{\mathcal{D}} \log(m+n)}_{(c)} + \underbrace{\frac{2}{\gamma} \sqrt{2\widehat{\mathcal{D}} \log(m+n)} T}_{(d)} \end{aligned}$$

where (1) = (a) + (b), (2) + (3) = (d), and (4) + (5) = (c). Recalling  $\bar{U}_{i_t, j_t} = \gamma \tilde{U}_{i_t, j_t}$  from the preliminaries of the proof, we then have  $h_\gamma(y_t, \bar{U}_{i_t, j_t}) \leq 2 \leq \frac{2}{\gamma}$ , giving

$$\sum_{t=1}^T h_\gamma(y_t, \bar{y}_t) - \sum_{t=1}^T h_\gamma(y_t, \bar{U}_{i_t, j_t}) \leq \frac{4}{\gamma} \sqrt{2\widehat{\mathcal{D}} \log(m+n)} T + \frac{4}{\gamma} \widehat{\mathcal{D}} \log(m+n).$$

□

Before we can compute the desired regret bound in terms of the 0-1 loss, we first need to introduce the following relationships.

**Lemma 19.** For  $y_t \in \{-1, 1\}$ ,  $\bar{y}_t \in \mathfrak{R}$ ,  $Y_t \sim \text{UNIFORM}(-\gamma, \gamma)$ ,  $\gamma \in (0, 1]$  and  $\hat{y}_t := \text{sign}(\bar{y}_t - Y_t)$ ,

$$2\mathbb{E}[y_t \neq \hat{y}_t] \leq h_\gamma(y_t, \bar{y}_t).$$

*Proof.* We have

$$p(\hat{y}_t = 1) = \begin{cases} 0 & \text{if } \bar{y}_t \leq -\gamma \\ \frac{1}{2} + \frac{\bar{y}_t}{2\gamma} & \text{if } -\gamma < \bar{y}_t \leq \gamma \\ 1 & \text{if } \bar{y}_t > \gamma \end{cases}$$

and

$$p(\hat{y}_t = -1) = \begin{cases} 1 & \text{if } \bar{y}_t \leq -\gamma \\ \frac{1}{2} - \frac{\bar{y}_t}{2\gamma} & \text{if } -\gamma < \bar{y}_t \leq \gamma \\ 0 & \text{if } \bar{y}_t > \gamma. \end{cases}$$

The possible cases are as follows.

1. If  $|\bar{y}_t| < \gamma$ ,  $2\mathbb{E}[y_t \neq \hat{y}_t] = h_\gamma(y_t, \bar{y}_t)$ . This is since if  $y_t = 1$ ,  $\mathbb{E}[y_t \neq \hat{y}_t] = \frac{1}{2} - \frac{\bar{y}_t}{2\gamma}$  and  $h_\gamma(y_t, \bar{y}_t) = \frac{1}{\gamma}(\gamma - \bar{y}_t)$ . Similarly if  $y_t = -1$ ,  $\mathbb{E}[y_t \neq \hat{y}_t] = \frac{1}{2} + \frac{\bar{y}_t}{2\gamma}$  and  $h_\gamma(y_t, \bar{y}_t) = \frac{1}{\gamma}(\gamma + \bar{y}_t)$ .
2. If  $|\bar{y}_t| \geq \gamma$  and  $\mathbb{E}[y_t \neq \hat{y}_t] = 0$ , then  $h_\gamma(y_t, \bar{y}_t) = \frac{1}{\gamma}[\gamma - |\bar{y}_t|]_+ = 0$ .
3. If  $|\bar{y}_t| \geq \gamma$  and  $\mathbb{E}[y_t \neq \hat{y}_t] = 1$ ,  $h_\gamma(y_t, \bar{y}_t) = \frac{1}{\gamma}[\gamma + |\bar{y}_t|]_+ \geq \frac{2\gamma}{\gamma} = 2\mathbb{E}[y_t \neq \hat{y}_t]$ .

□

**Lemma 20.** Suppose we have  $\mathbf{U}$  as in Theorem 1. Recalling that  $\bar{\mathbf{U}} = \gamma\mathbf{U}$  from the preliminaries of the proof, we have that

$$h_\gamma(y_t, \bar{\mathbf{U}}_{i_t, j_t}) \leq 2[y_t \neq U_{i_t, j_t}].$$

*Proof.* Recall that  $\mathbf{U} \in \{-1, 1\}^{m \times n}$ . The hinge loss is then given by  $h_\gamma(y_t, \bar{\mathbf{U}}_{i_t, j_t}) = \frac{1}{\gamma}[\gamma - y_t \bar{\mathbf{U}}_{i_t, j_t}]_+$ . In the case that  $y_t = \bar{\mathbf{U}}_{i_t, j_t}$ ,  $h_\gamma(y_t, \bar{\mathbf{U}}_{i_t, j_t}) = \frac{1}{\gamma}[\gamma - \gamma]_+ = 0$ . Otherwise,  $h_\gamma(y_t, \bar{\mathbf{U}}_{i_t, j_t}) = \frac{1}{\gamma}[\gamma + \gamma]_+ = 2$ . □

We proceed by giving a sharper bound for Algorithm 1 than is stated in Theorem 1. This, however, only holds under the additional assumption that  $T \geq 2\hat{\mathcal{D}} \log(m+n)$ .

**Theorem 21.** The expected regret of Algorithm 1 with **non-conservative** updates and parameters  $\gamma \in (0, 1]$ ,  $\hat{\mathcal{D}} \geq \mathcal{D}_{\mathbf{M}, \mathbf{N}}^\gamma(\mathbf{U})$ ,  $\eta = \sqrt{\frac{\hat{\mathcal{D}} \log(m+n)}{2T}}$ , p.d. matrices  $\mathbf{M} \in \mathbf{S}_{++}^m$  and  $\mathbf{N} \in \mathbf{S}_{++}^n$  and for  $T \geq 2\hat{\mathcal{D}} \log(m+n)$  is bounded by

$$\mathbb{E}[|\mathbb{M}|] - \sum_{t \in [T]} [y_t \neq U_{i_t, j_t}] \leq \frac{2}{\gamma} \sqrt{2\hat{\mathcal{D}} \log(m+n)T} + \frac{2}{\gamma} \hat{\mathcal{D}} \log(m+n) \quad (28)$$

for all  $\mathbf{U} \in \{-1, 1\}^{m \times n}$  with  $\|\mathbf{U}\|_{\max} \leq 1/\gamma$ .

*Proof.* Starting from Lemma 18, we observe that we can apply Lemma 19, to bound the expected mistakes of the latter by the cumulative hinge loss of the former. Combining this with Lemma 20 then gives the desired regret bound in terms of the 0-1 loss. Note that  $\hat{\mathcal{D}} \geq \mathcal{D} = \mathcal{D}_{\mathbf{M}, \mathbf{N}}^\gamma(\mathbf{U})$ , where the inequality comes from Lemma 18 and the equality is stated in the preliminaries of the proof, which assumes that  $\|\mathbf{U}\|_{\max} \leq 1/\gamma$ . □

We will now prove the first part of Theorem 1. We split into two cases,

Case 1)  $T < 2\hat{\mathcal{D}} \log(m+n)$ :

We have that

$$\mathbb{E}[|\mathbb{M}|] - \sum_{t \in [T]} [y_t \neq U_{i_t, j_t}] \leq T < 2\hat{\mathcal{D}} \log(m+n)$$

for all  $\eta > 0$ .

Case 2)  $T \geq 2\hat{\mathcal{D}} \log(m+n)$ :

From (28) we have

$$\begin{aligned} \mathbb{E}[|\mathbb{M}|] - \sum_{t \in [T]} [y_t \neq U_{i_t, j_t}] &\leq \frac{2}{\gamma} \sqrt{2\hat{\mathcal{D}} \log(m+n)T} + \frac{2}{\gamma} \hat{\mathcal{D}} \log(m+n) \\ &= \frac{2}{\gamma} \sqrt{2\hat{\mathcal{D}} \log(m+n)T} + \frac{2}{\gamma} \sqrt{(\hat{\mathcal{D}} \log(m+n))^2} \\ &\leq \frac{2}{\gamma} \sqrt{2\hat{\mathcal{D}} \log(m+n)T} + \frac{2}{\gamma} \sqrt{\frac{1}{2} \hat{\mathcal{D}} \log(m+n)T} \\ &= \frac{2}{\gamma} \left( \sqrt{2} + \sqrt{\frac{1}{2}} \right) \sqrt{\hat{\mathcal{D}} \log(m+n)T} \end{aligned} \quad (29)$$

where in Equation (29) we used the assumption on  $T$ .

Combining both cases, we have that the following holds for all  $T$

$$\mathbb{E}[|\mathbb{M}|] - \sum_{t \in [T]} [y_t \neq U_{i_t, j_t}] \leq \frac{2}{\gamma} \left( \sqrt{2} + \sqrt{\frac{1}{2}} \right) \sqrt{\hat{\mathcal{D}} \log(m+n)T} + \min(2\hat{\mathcal{D}} \log(m+n), T)$$

$$\begin{aligned}
&= \frac{2}{\gamma} \left( \sqrt{2} + \sqrt{\frac{1}{2}} \right) \sqrt{\widehat{\mathcal{D}} \log(m+n)T} + \sqrt{\min(2\widehat{\mathcal{D}} \log(m+n), T)^2} \\
&\leq \frac{2}{\gamma} \left( \sqrt{2} + \sqrt{\frac{1}{2}} \right) \sqrt{\widehat{\mathcal{D}} \log(m+n)T} + \sqrt{2\widehat{\mathcal{D}} \log(m+n)T} \\
&\leq \frac{2}{\gamma} \left( \sqrt{2} + \sqrt{\frac{1}{2}} \right) \sqrt{\widehat{\mathcal{D}} \log(m+n)T} + \frac{2}{\gamma} \sqrt{\frac{1}{2} \widehat{\mathcal{D}} \log(m+n)T} \quad (30) \\
&= \frac{4}{\gamma} \sqrt{2\widehat{\mathcal{D}} \log(m+n)T}
\end{aligned}$$

where we used the fact that  $\frac{1}{\gamma} \geq 1$  in Equation (30). Thus we have demonstrated (5) proving Theorem 1.  $\blacksquare$

### B.3 Proof for the Realizable Case

In this subsection, we prove the second part of Theorem 1. Recall from the theorem statement that  $y_t = \mathbf{U}_{i_t, j_t}$  for all  $t \in \mathbb{M}$ ,  $\text{mc}(\mathbf{U})^{-1} \geq \gamma$ ,  $\widehat{\mathcal{D}} \geq \min_{\mathbf{V} \in \text{SP}^1(\mathbf{U})} \mathcal{D}_{M, N}^\gamma(\mathbf{V})$ ,  $\eta = \gamma$  and that we have **conservative** updates. Recall from the preliminaries of the proof that given  $\mathbf{U}$ ,  $\tilde{\mathbf{U}} \in \mathbb{R}^{m \times n}$  is such that  $\tilde{\mathbf{U}} = \gamma \operatorname{argmin}_{\mathbf{V} \in \text{SP}^1(\mathbf{U})} \mathcal{D}_{M, N}^\gamma(\mathbf{V})$ , meaning that  $\min_{i \in [m], j \in [n]} |\tilde{\mathbf{U}}_{ij}| \geq \gamma$ . Also recall that  $\mathcal{D} = \min_{\mathbf{V} \in \text{SP}^1(\mathbf{U})} \mathcal{D}_{M, N}^\gamma(\mathbf{V})$ .

**Lemma 22.** [25, Lemma 2.1] If  $\mathbf{A} \in \mathbf{S}_+^d$  with eigenvalues in  $[0, 1]$  and  $a \in \mathbb{R}$  then:

$$(1 - e^a) \mathbf{A} \preceq \mathbf{I} - \exp(a\mathbf{A})$$

**Lemma 23.** For all trials  $t \in \mathbb{M}$ , we have:

$$\Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^t) - \Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^{t+1}) \geq \eta y_t \operatorname{tr}(\tilde{\mathbf{U}} \tilde{\mathbf{X}}^t) + (1 - e^{\eta y_t}) \operatorname{tr}(\tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t). \quad (31)$$

*Proof.* We have:

$$\begin{aligned}
\Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^t) - \Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^{t+1}) &= \operatorname{tr}(\tilde{\mathbf{U}} \log \tilde{\mathbf{W}}^{t+1} - \tilde{\mathbf{U}} \log \tilde{\mathbf{W}}^t) + \operatorname{tr}(\tilde{\mathbf{W}}^t) - \operatorname{tr}(\tilde{\mathbf{W}}^{t+1}) \\
&= \eta y_t \operatorname{tr}(\tilde{\mathbf{U}} \tilde{\mathbf{X}}^t) + \operatorname{tr}(\tilde{\mathbf{W}}^t) - \operatorname{tr}(e^{\log \tilde{\mathbf{W}}^t + \eta y_t \tilde{\mathbf{X}}^t}) \quad (32)
\end{aligned}$$

$$\geq \eta y_t \operatorname{tr}(\tilde{\mathbf{U}} \tilde{\mathbf{X}}^t) + \operatorname{tr}(\tilde{\mathbf{W}}^t) - \operatorname{tr}(e^{\log \tilde{\mathbf{W}}^t} e^{\eta y_t \tilde{\mathbf{X}}^t}) \quad (33)$$

$$\begin{aligned}
&= \eta y_t \operatorname{tr}(\tilde{\mathbf{U}} \tilde{\mathbf{X}}^t) + \operatorname{tr}(\tilde{\mathbf{W}}^t (\mathbf{I} - e^{\eta y_t \tilde{\mathbf{X}}^t})) \\
&\geq \eta y_t \operatorname{tr}(\tilde{\mathbf{U}} \tilde{\mathbf{X}}^t) + (1 - e^{\eta y_t}) \operatorname{tr}(\tilde{\mathbf{W}}^t \tilde{\mathbf{X}}^t), \quad (34)
\end{aligned}$$

where Equation (32) comes from the update of the algorithm, Equation (33) comes from Lemma 11 and Equation (34) comes from Lemma 22 which applies since, by Lemma 9 all eigenvalues of  $\tilde{\mathbf{X}}^t$  are in  $[0, 1]$ .  $\square$

**Lemma 24.** [24, Lemma A.5] For  $x \in [-1, 1]$ ,

$$x^2 + x + 1 - e^x \geq (3 - e)x^2.$$

We proceed by showing that the ‘‘progress’’  $\Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^t) - \Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^{t+1})$  of  $\tilde{\mathbf{W}}^t$  towards  $\tilde{\mathbf{U}}$  may be further lower bounded by  $c\gamma$  (see Lemma 25).

**Lemma 25.** Let  $c := 3 - e$ . For all trials  $t$  with  $t \in \mathbb{M}$  (under the conditions of Lemma 28) we have:

$$\Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^t) - \Delta(\tilde{\mathbf{U}}, \tilde{\mathbf{W}}^{t+1}) \geq c\gamma^2$$

*Proof.* By Lemma 7,  $\bar{U}_{i_t j_t} = \text{tr}(\tilde{U} \tilde{X}^t) - 1$  so since  $y_t = \text{sign}(\bar{U}_{i_t j_t})$ , and  $\gamma \leq |\bar{U}_{i_t j_t}|$  we have  $\gamma \leq y_t (\text{tr}(\tilde{U} \tilde{X}^t) - 1)$ . So when  $y_t = 1$  we have  $\text{tr}(\tilde{U} \tilde{X}^t) \geq 1 + \gamma$  and when  $y_t = -1$  we have  $\text{tr}(\tilde{U} \tilde{X}^t) \leq 1 - \gamma$ . We use these inequalities as follows.

First suppose that  $y_t = 1$ . By Lemma 23 we have:

$$\begin{aligned} \Delta(\tilde{U}, \tilde{W}^t) - \Delta(\tilde{U}, \tilde{W}^{t+1}) &\geq \gamma \text{tr}(\tilde{U} \tilde{X}^t) + (1 - e^\gamma) \text{tr}(\tilde{W}^t \tilde{X}^t) \\ &\geq \gamma(1 + \gamma) + (1 - e^\gamma) \text{tr}(\tilde{W}^t \tilde{X}^t) \\ &\geq \gamma(1 + \gamma) + (1 - e^\gamma) \end{aligned} \quad (35)$$

$$\begin{aligned} &= (\gamma + \gamma^2) + 1 - e^\gamma \\ &\geq c\gamma^2, \end{aligned} \quad (36)$$

where Equation (36) comes from Lemma 24 and Equation (35) comes from the fact that  $\hat{y}^t = -1$  and hence, by the algorithm,  $\text{tr}(\tilde{W}^t \tilde{X}^t) \leq 1$ .

Now suppose that  $y_t = -1$ . By Lemma 23 we have:

$$\begin{aligned} \Delta(\tilde{U}, \tilde{W}^t) - \Delta(\tilde{U}, \tilde{W}^{t+1}) &\geq -\gamma \text{tr}(\tilde{U} \tilde{X}^t) + (1 - e^{-\gamma}) \text{tr}(\tilde{W}^t \tilde{X}^t) \\ &\geq -\gamma(1 - \gamma) + (1 - e^{-\gamma}) \text{tr}(\tilde{W}^t \tilde{X}^t) \\ &\geq -\gamma(1 - \gamma) + (1 - e^{-\gamma}) \end{aligned} \quad (37)$$

$$\begin{aligned} &= -\gamma + \gamma^2 + 1 - e^{-\gamma} \\ &\geq c\gamma^2, \end{aligned} \quad (38)$$

where Equation (38) comes from Lemma 24 and Equation (37) comes from the fact that  $\hat{y}^t = 1$  and hence, by the algorithm,  $\text{tr}(\tilde{W}^t \tilde{X}^t) \geq 1$ .  $\square$

**Lemma 26.** *We have,*

$$c\gamma^2 |\mathbb{M}| \leq \Delta(\tilde{U}, \tilde{W}^1).$$

*Proof.* Suppose that we have  $T$  trials. Then we have:

$$\begin{aligned} \Delta(\tilde{U}, \tilde{W}^1) &\geq \Delta(\tilde{U}, \tilde{W}^1) - \Delta(\tilde{U}, \tilde{W}^{T+1}) \\ &= \sum_{t \in [T]} \left( \Delta(\tilde{U}, \tilde{W}^t) - \Delta(\tilde{U}, \tilde{W}^{t+1}) \right) \\ &= \sum_{t \in \mathbb{M}} \left( \Delta(\tilde{U}, \tilde{W}^t) - \Delta(\tilde{U}, \tilde{W}^{t+1}) \right) \end{aligned} \quad (39)$$

$$\begin{aligned} &\geq \sum_{t \in \mathbb{M}} c\gamma^2 \\ &= c\gamma^2 |\mathbb{M}|, \end{aligned} \quad (40)$$

where (40) follows from (39) using Lemma 25.  $\square$

**Lemma 27.** *Given that  $\tilde{W}^1 = \hat{\mathcal{D}} \frac{I}{m+n}$  we have*

$$\Delta(\tilde{U}, \tilde{W}^1) \leq \text{tr}(\tilde{U}) \log(m+n) + \text{tr}(\tilde{U}) \log \frac{\text{tr}(\tilde{U})}{\hat{\mathcal{D}}} + \hat{\mathcal{D}} - \text{tr}(\tilde{U})$$

*Proof.* We have:

$$\Delta(\tilde{U}, \tilde{W}^1) = \text{tr}(\tilde{U} \log \tilde{U}) - \text{tr}(\tilde{U} \log \tilde{W}^1) + \text{tr}(\tilde{W}^1) - \text{tr}(\tilde{U})$$



$$\begin{aligned}
&= \text{tr}(\tilde{\mathbf{U}} \log \tilde{\mathbf{U}}) - \text{tr}\left(\tilde{\mathbf{U}} \log\left(\frac{\hat{\mathcal{D}}}{m+n} \mathbf{I}\right)\right) + \text{tr}\left(\frac{\hat{\mathcal{D}}}{m+n} \mathbf{I}\right) - \text{tr}(\tilde{\mathbf{U}}) \\
&= \text{tr}(\tilde{\mathbf{U}} \log \tilde{\mathbf{U}}) - \text{tr}\left(\tilde{\mathbf{U}} \log\left(\frac{\hat{\mathcal{D}}}{m+n} \mathbf{I}\right)\right) + \hat{\mathcal{D}} - \text{tr}(\tilde{\mathbf{U}}) \\
&= \text{tr}(\tilde{\mathbf{U}} \log \tilde{\mathbf{U}}) - \text{tr}\left(\tilde{\mathbf{U}} \left(\mathbf{I} \log\left(\frac{\hat{\mathcal{D}}}{m+n}\right)\right)\right) + \hat{\mathcal{D}} - \text{tr}(\tilde{\mathbf{U}}) \\
&= \text{tr}(\tilde{\mathbf{U}} \log \tilde{\mathbf{U}}) - \text{tr}\left(\tilde{\mathbf{U}} \log\left(\frac{\hat{\mathcal{D}}}{m+n}\right)\right) + \hat{\mathcal{D}} - \text{tr}(\tilde{\mathbf{U}}) \tag{41}
\end{aligned}$$

$$\begin{aligned}
&\leq \text{tr}(\tilde{\mathbf{U}} \log(\text{tr}(\tilde{\mathbf{U}}))) - \text{tr}\left(\tilde{\mathbf{U}} \log\left(\frac{\hat{\mathcal{D}}}{m+n}\right)\right) + \hat{\mathcal{D}} - \text{tr}(\tilde{\mathbf{U}}) \tag{42} \\
&= \left(\log(\text{tr}(\tilde{\mathbf{U}})) - \log\left(\frac{\hat{\mathcal{D}}}{m+n}\right)\right) \text{tr}(\tilde{\mathbf{U}}) + \hat{\mathcal{D}} - \text{tr}(\tilde{\mathbf{U}}) \\
&= \log\left(\frac{\text{tr}(\tilde{\mathbf{U}})(m+n)}{\hat{\mathcal{D}}}\right) \text{tr}(\tilde{\mathbf{U}}) + \hat{\mathcal{D}} - \text{tr}(\tilde{\mathbf{U}}),
\end{aligned}$$

where (42) follows from (41), since  $\tilde{\mathbf{U}} := \mathbf{V} \mathbf{\Lambda} \mathbf{V}^{-1}$  where  $\mathbf{\Lambda}$  is a diagonal matrix of the eigenvalues of  $\tilde{\mathbf{U}}$ . This holds since,

$$\begin{aligned}
\text{tr}(\tilde{\mathbf{U}} \log \tilde{\mathbf{U}}) &= \text{tr}(\mathbf{V} \mathbf{\Lambda} \mathbf{V}^{-1} \mathbf{V} \log \mathbf{\Lambda} \mathbf{V}^{-1}) \\
&= \text{tr}(\mathbf{V} \mathbf{\Lambda} \log \mathbf{\Lambda} \mathbf{V}^{-1}) \\
&= \text{tr}(\mathbf{\Lambda} \log \mathbf{\Lambda}) \\
&= \sum_{i=1}^{m+n} \lambda_i \log(\lambda_i) \\
&\leq \left(\sum_{i=1}^{m+n} \lambda_i\right) \log\left(\sum_{i=1}^{m+n} \lambda_i\right) \\
&= \text{tr}(\tilde{\mathbf{U}} \log(\text{tr}(\tilde{\mathbf{U}}))).
\end{aligned}$$

□

**Lemma 28.** *The mistakes,  $|\mathbb{M}|$ , of Algorithm 1 with the assumption that  $y_t = \text{sign}(\bar{U}_{i_t, j_t})$  for all  $t \in \mathbb{M}$  and with parameters  $\gamma \leq \text{mc}(\mathbf{U})^{-1}$ ,  $1 \leq \hat{\mathcal{D}}$  and  $\eta = \gamma$  and **conservative** updates, is bounded above by:*

$$|\mathbb{M}| \leq 3.6 \frac{1}{\gamma^2} \left( \mathcal{D} \left( \log(m+n) + \log \frac{\mathcal{D}}{\hat{\mathcal{D}}} \right) + \hat{\mathcal{D}} - \mathcal{D} \right) \tag{43}$$

*Proof.* Combining Lemmas 26 and 27 gives us

$$|\mathbb{M}| \leq \frac{1}{c} \frac{1}{\gamma^2} \left( \text{tr}(\tilde{\mathbf{U}}) \log(m+n) + \text{tr}(\tilde{\mathbf{U}}) \log \frac{\text{tr}(\tilde{\mathbf{U}})}{\hat{\mathcal{D}}} + \hat{\mathcal{D}} - \text{tr}(\tilde{\mathbf{U}}) \right)$$

Using Lemma 8 and upper bounding  $1/c$  by 3.6 then gives the result. □

The theorem statement for the realizable case then follows by setting  $\hat{\mathcal{D}} \geq \mathcal{D}$ . ■

### C Proof of Theorem 3

We recall Theorem 3 and then prove it.

**Theorem 3.** If  $U \in \mathbb{B}_{k,\ell}^{m,n}$  define

$$\mathcal{D}_{M,N}^\circ(U) := \begin{cases} 2 \operatorname{tr}(\mathbf{R}^\top \mathbf{M} \mathbf{R}) \mathcal{R}_M + 2 \operatorname{tr}(\mathbf{C}^\top \mathbf{N} \mathbf{C}) \mathcal{R}_N + 2k + 2\ell & \mathbf{M} \text{ and } \mathbf{N} \text{ are PDLaplacians} \\ k \operatorname{tr}(\mathbf{R}^\top \mathbf{M} \mathbf{R}) \mathcal{R}_M + \ell \operatorname{tr}(\mathbf{C}^\top \mathbf{N} \mathbf{C}) \mathcal{R}_N & \mathbf{M} \in \mathbf{S}_{++}^m \text{ and } \mathbf{N} \in \mathbf{S}_{++}^n \end{cases}$$

as the minimum over all decompositions of  $U = \mathbf{R} \mathbf{U}^* \mathbf{C}^\top$  for  $\mathbf{R} \in \mathcal{B}^{m,k}$ ,  $\mathbf{C} \in \mathcal{B}^{n,\ell}$  and  $\mathbf{U}^* \in \{-1, 1\}^{k \times \ell}$ . Thus for  $U \in \mathbb{B}_{k,\ell}^{m,n}$ ,

$$\begin{aligned} \mathcal{D}_{M,N}^\gamma(U) &\leq \mathcal{D}_{M,N}^\circ(U) && \text{(if } \|U\|_{\max} \leq 1/\gamma) \\ \min_{V \in \operatorname{SP}^1(U)} \mathcal{D}_{M,N}^\gamma(V) &\leq \mathcal{D}_{M,N}^\circ(U) && \text{(if } \operatorname{mc}(U) \leq 1/\gamma). \end{aligned}$$

*Proof.* A  $\gamma$ -decomposition of matrix  $U$  is given by a  $\hat{\mathbf{P}} \in \mathcal{N}^{m,d}$  and a  $\hat{\mathbf{Q}} \in \mathcal{N}^{n,d}$  such that  $\hat{\mathbf{P}} \hat{\mathbf{Q}}^\top = \gamma U$ . A block-invariant decomposition of matrix  $U \in \mathbb{B}_{k,\ell}^{m,n}$  is given by a  $\hat{\mathbf{P}} \in \mathcal{N}^{m,d}$  and a  $\hat{\mathbf{Q}} \in \mathcal{N}^{n,d}$  for some  $d$  such that there exists a  $\delta \in (0, 1]$ ,  $\hat{\mathbf{P}}^* \in \mathcal{N}^{k,d}$ , and a  $\hat{\mathbf{Q}}^* \in \mathcal{N}^{\ell,d}$ , so that  $\hat{\mathbf{P}} = \mathbf{R} \hat{\mathbf{P}}^*$ ,  $\hat{\mathbf{Q}} = \mathbf{C} \hat{\mathbf{Q}}^*$  and  $\hat{\mathbf{P}} \hat{\mathbf{Q}}^\top = \delta U$ .

We now prove the following intermediate result,

**Lemma:** If  $U \in \mathbb{B}_{k,\ell}^{m,n}$ , then for every  $\gamma \in (0, 1/\|U\|_{\max})$ , there exists a block-invariant  $\gamma$ -decomposition of  $U$ .

*Proof.* Since  $U \in \mathbb{B}_{k,\ell}^{m,n}$  we have that  $U = \mathbf{R} \mathbf{U}^* \mathbf{C}^\top$  for some  $\mathbf{R} \in \mathcal{B}^{m,k}$ ,  $\mathbf{C} \in \mathcal{B}^{n,\ell}$  and  $\mathbf{U}^* \in \{-1, 1\}^{k \times \ell}$ . Observe by block invariance we have that  $\|U\|_{\max} = \|\mathbf{U}^*\|_{\max}$  and by the definition of  $\|\cdot\|_{\max}$  we have that there exists a  $\left(\frac{1}{\|U\|_{\max}}\right)$ -decomposition of  $\mathbf{U}^*$  via factors  $\hat{\mathbf{P}}^* \in \mathcal{N}^{k,d}$ , and a  $\hat{\mathbf{Q}}^* \in \mathcal{N}^{\ell,d}$ , this implies that  $\hat{\mathbf{P}} := \mathbf{R} \hat{\mathbf{P}}^*$ ,  $\hat{\mathbf{Q}} := \mathbf{C} \hat{\mathbf{Q}}^*$  is a  $\left(\frac{1}{\|U\|_{\max}}\right)$ -block-invariant decomposition of  $U$ .

Now given any  $\gamma \in (0, 1/\|U\|_{\max})$  we construct a  $\gamma$ -block-invariant decomposition of  $U$ . Set  $c := \gamma \|U\|_{\max}$ . We construct new factor matrices  $\hat{\mathbf{P}}' \in \mathcal{N}^{m,d+1}$  and  $\hat{\mathbf{Q}}' \in \mathcal{N}^{n,d+1}$

$$\hat{\mathbf{P}}' := \begin{pmatrix} c \hat{\mathbf{P}} & (\sqrt{1-c^2}) \mathbf{1} \end{pmatrix}; \quad \hat{\mathbf{Q}}' := \begin{pmatrix} \hat{\mathbf{Q}} & \mathbf{0} \end{pmatrix}.$$

Observe that  $(\hat{\mathbf{P}}', \hat{\mathbf{Q}}')$  is the required  $\gamma$ -block-invariant decomposition of  $U$ .  $\square$

Recall (3),

$$\mathcal{D}_{M,N}^\gamma(U) := \min_{\hat{\mathbf{P}} \hat{\mathbf{Q}}^\top = \gamma U} \mathcal{R}_M \operatorname{tr}(\hat{\mathbf{P}}^\top \mathbf{M} \hat{\mathbf{P}}) + \mathcal{R}_N \operatorname{tr}(\hat{\mathbf{Q}}^\top \mathbf{N} \hat{\mathbf{Q}}). \quad (44)$$

Observe that when the feasible set of the optimization that defines  $\mathcal{D}_{M,N}^\gamma(U)$  is non-empty and  $U \in \mathbb{B}_{k,\ell}^{m,n}$ , there exists a member of the feasible set which is a block-invariant decomposition of  $U$  by the lemma above. We proceed by proving an upper bound of

$$\mathcal{R}_M \operatorname{tr}(\hat{\mathbf{P}}^\top \mathbf{M} \hat{\mathbf{P}}) + \mathcal{R}_N \operatorname{tr}(\hat{\mathbf{Q}}^\top \mathbf{N} \hat{\mathbf{Q}})$$

for every block-invariant decomposition of  $U$ .

First we will bound the term  $\operatorname{tr}(\hat{\mathbf{P}}^\top \mathbf{M} \hat{\mathbf{P}})$  for general positive definite matrices and then for PDLaplacians. By symmetry, the bound will also hold for  $\operatorname{tr}(\hat{\mathbf{Q}}^\top \mathbf{N} \hat{\mathbf{Q}})$ .

Suppose  $(\hat{\mathbf{P}}, \hat{\mathbf{Q}})$  is a block-invariant decomposition of  $U$ . Then, we have

$$\begin{aligned} \operatorname{tr}(\hat{\mathbf{P}}^\top \mathbf{M} \hat{\mathbf{P}}) &= \operatorname{tr}((\mathbf{R} \hat{\mathbf{P}}^*)^\top \mathbf{M} \mathbf{R} \hat{\mathbf{P}}^*) = \operatorname{tr}(\hat{\mathbf{P}}^* (\hat{\mathbf{P}}^*)^\top \mathbf{R}^\top \mathbf{M} \mathbf{R}) \\ &\leq \operatorname{tr}(\hat{\mathbf{P}}^* (\hat{\mathbf{P}}^*)^\top) \operatorname{tr}(\mathbf{R}^\top \mathbf{M} \mathbf{R}) = k \operatorname{tr}(\mathbf{R}^\top \mathbf{M} \mathbf{R}), \end{aligned}$$

where the inequality comes from the fact that  $\text{tr}(\mathbf{A}\mathbf{B}) \leq \lambda_{\max}(\mathbf{A}) \text{tr}(\mathbf{B}) \leq \text{tr}(\mathbf{A}) \text{tr}(\mathbf{B})$  for  $\mathbf{A}, \mathbf{B} \in \mathbf{S}_+$ . By symmetry we have demonstrated the inequality for positive definite matrices.

We now consider PDLaplacians. Assume  $\mathbf{M} := \mathbf{L}^\circ = \mathbf{L} + \left(\frac{1}{m}\right) \left(\frac{1}{m}\right)^\top \mathcal{R}_{\mathbf{L}}^{-1}$ , a PDLaplacian. Recall the following two elementary inequalities from the preliminaries: if  $\mathbf{u} \in [-1, 1]^m$ , then

$$(\mathbf{u}^\top \mathbf{L}\mathbf{u})\mathcal{R}_{\mathbf{L}} \leq \frac{1}{2}(\mathbf{u}^\top \mathbf{L}^\circ \mathbf{u})\mathcal{R}_{\mathbf{L}^\circ}, \quad (45)$$

$$(\mathbf{u}^\top \mathbf{L}^\circ \mathbf{u})\mathcal{R}_{\mathbf{L}^\circ} \leq 2(\mathbf{u}^\top \mathbf{L}\mathbf{u} \mathcal{R}_{\mathbf{L}} + 1). \quad (46)$$

Observe that for an  $m \times m$  graph Laplacian  $\mathbf{L}$  with adjacency matrix  $\mathbf{A}$  that for  $\mathbf{X} \in \mathfrak{R}^{m \times d}$ ,

$$\text{tr}(\mathbf{X}^\top \mathbf{L}\mathbf{X}) = \sum_{(i,j) \in E} A_{ij} \|\mathbf{X}_i - \mathbf{X}_j\|^2. \quad (47)$$

Suppose  $(\hat{\mathbf{P}}, \hat{\mathbf{Q}})$  is a block-invariant decomposition of  $\mathbf{U}$  then the row vectors  $\hat{\mathbf{P}}_1, \dots, \hat{\mathbf{P}}_m$  come in at most  $k$  distinct varieties, that is  $|\bigcup_{i \in [m]} \hat{\mathbf{P}}_i| \leq k$ . The same holds for  $\mathbf{R}$  and furthermore  $(\hat{\mathbf{P}}_r = \hat{\mathbf{P}}_s) \iff (\mathbf{R}_r = \mathbf{R}_s)$  for  $r, s \in [m]$ . Observe that given  $r, s \in [m]$  that if  $\mathbf{R}_r \neq \mathbf{R}_s$  then  $\|\mathbf{R}_r - \mathbf{R}_s\|^2 = 2$  and  $\|\hat{\mathbf{P}}_r - \hat{\mathbf{P}}_s\|^2 \leq 4$  since they are coordinate and unit vectors respectively. This then implies,

$$\text{tr}(\hat{\mathbf{P}}^\top \mathbf{L}\hat{\mathbf{P}}) \leq 2 \text{tr}(\mathbf{R}^\top \mathbf{L}\mathbf{R}). \quad (48)$$

Thus we have

$$\begin{aligned} \text{tr}(\hat{\mathbf{P}}^\top \mathbf{M}\hat{\mathbf{P}})\mathcal{R}_{\mathbf{M}} &\leq 2 \text{tr}(\hat{\mathbf{P}}^\top \mathbf{L}\hat{\mathbf{P}})\mathcal{R}_{\mathbf{L}} + 2k && \text{by (46)} \\ &\leq 4 \text{tr}(\mathbf{R}^\top \mathbf{L}\mathbf{R})\mathcal{R}_{\mathbf{L}} + 2k && \text{by (48)} \\ &\leq 2 \text{tr}(\mathbf{R}^\top \mathbf{M}\mathbf{R})\mathcal{R}_{\mathbf{M}} + 2k && \text{by (45)} \end{aligned}$$

By symmetry we have demonstrated the inequality for PDLaplacians.  $\square$

## D Proofs for Section 5

### D.1 Proof of Proposition 4

We now prove Proposition 4, that the transductive and inductive algorithms are equivalent. Recall by assumption that  $\mathcal{R}_{\mathcal{M}} = \mathcal{R}_{\mathbf{M}}$  and  $\mathcal{R}_{\mathcal{N}} = \mathcal{R}_{\mathbf{N}}$ .

#### D.1.1 Equivalence of Traces

Suppose, in this subsection, that we have some given trial  $t$ . In this subsection we analyse the inductive algorithm. We make the following definitions:

**Definition 29.** For all  $s \in \mathbb{U} \cap [t]$

$$\begin{aligned} \mathbf{v}(s) &:= \left[ \frac{(\sqrt{(\mathbf{M}^t)^+})\mathbf{e}^{i_s}}{\sqrt{2\mathcal{R}_{\mathbf{M}}}}; \frac{(\sqrt{(\mathbf{N}^t)^+})\mathbf{e}^{j_s}}{\sqrt{2\mathcal{R}_{\mathbf{N}}}} \right] \\ \bar{\mathbf{v}}(s) &:= \left[ \frac{(\sqrt{(\mathbf{M}^{T+1})^+})\mathbf{e}^{i_s}}{\sqrt{2\mathcal{R}_{\mathbf{M}}}}; \frac{(\sqrt{(\mathbf{N}^{T+1})^+})\mathbf{e}^{j_s}}{\sqrt{2\mathcal{R}_{\mathbf{N}}}} \right] \end{aligned}$$

Note that  $\tilde{\mathbf{X}}^t(s) = \mathbf{v}(s)\mathbf{v}(s)^\top$  and  $\tilde{\mathbf{X}}^{T+1}(s) = \bar{\mathbf{v}}(s)\bar{\mathbf{v}}(s)^\top$  for  $s \in \mathbb{U} \cap [t]$ .

**Lemma 30.** For all  $l \in \mathbb{N}$  and for all  $a_1, a_2, \dots, a_l \in \mathbb{U} \cap [t-1]$  there exists some  $\alpha \in \mathfrak{R}$  such that:

$$\tilde{\mathbf{X}}^t(a_1)\tilde{\mathbf{X}}^t(a_2) \cdots \tilde{\mathbf{X}}^t(a_l) = \alpha \mathbf{v}(a_1)\mathbf{v}(a_l)^\top$$

and

$$\tilde{\mathbf{X}}^{T+1}(a_1)\tilde{\mathbf{X}}^{T+1}(a_2) \cdots \tilde{\mathbf{X}}^{T+1}(a_l) = \alpha \bar{\mathbf{v}}(a_1)\bar{\mathbf{v}}(a_l)^\top$$

*Proof.* We prove by induction on  $l$ . In the case  $l := 1$  the result is clear with  $\alpha := 1$ .

Now suppose the result holds with  $l := q$  for some  $q \in \mathbb{N}$ . We now show that it holds for  $l := q + 1$ . Since it holds for  $l := q$ , choose  $\alpha'$  such that  $\tilde{\mathbf{X}}^t(a_1)\tilde{\mathbf{X}}^t(a_2)\cdots\tilde{\mathbf{X}}^t(a_q) = \alpha'\mathbf{v}(a_1)\mathbf{v}(a_q)^\top$  and  $\tilde{\mathbf{X}}^{T+1}(a_1)\tilde{\mathbf{X}}^{T+1}(a_2)\cdots\tilde{\mathbf{X}}^{T+1}(a_q) = \alpha'\bar{\mathbf{v}}(a_1)\bar{\mathbf{v}}(a_q)^\top$ . Note that we now have:

$$\begin{aligned}\tilde{\mathbf{X}}^t(a_1)\tilde{\mathbf{X}}^t(a_2)\cdots\tilde{\mathbf{X}}^t(a_l) &= \tilde{\mathbf{X}}^t(a_1)\tilde{\mathbf{X}}^t(a_2)\cdots\tilde{\mathbf{X}}^t(a_q)\tilde{\mathbf{X}}^t(a_l) \\ &= \alpha'\mathbf{v}(a_1)\mathbf{v}(a_q)^\top\tilde{\mathbf{X}}^t(a_l) \\ &= \alpha'\mathbf{v}(a_1)\mathbf{v}(a_q)^\top\mathbf{v}(a_l)\mathbf{v}(a_l)^\top \\ &= (\mathbf{v}(a_q)^\top\mathbf{v}(a_l))\alpha'\mathbf{v}(a_1)\mathbf{v}(a_l)^\top \\ &= \left(\frac{\mathcal{M}^+(i_{a_q}, i_{a_l})}{2\mathcal{R}_M} + \frac{\mathcal{N}^+(j_{a_q}, j_{a_l})}{2\mathcal{R}_N}\right)\alpha'\mathbf{v}(a_1)\mathbf{v}(a_l)^\top\end{aligned}$$

Similarly we have:

$$\tilde{\mathbf{X}}^{T+1}(a_1)\tilde{\mathbf{X}}^{T+1}(a_2)\cdots\tilde{\mathbf{X}}^{T+1}(a_l) = \left(\frac{\mathcal{M}^+(i_{a_q}, i_{a_l})}{2\mathcal{R}_M} + \frac{\mathcal{N}^+(j_{a_q}, j_{a_l})}{2\mathcal{R}_N}\right)\alpha'\bar{\mathbf{v}}(a_1)\bar{\mathbf{v}}(a_l)^\top,$$

from which the result follows.  $\square$

**Lemma 31.** For all  $l \in \mathbb{N}$  and for all  $a_1, a_2, \dots, a_l \in \mathbb{U} \cap [t-1]$  we have:

$$\text{tr}\left(\tilde{\mathbf{X}}^t(t)\tilde{\mathbf{X}}^t(a_1)\tilde{\mathbf{X}}^t(a_2)\cdots\tilde{\mathbf{X}}^t(a_l)\right) = \text{tr}\left(\tilde{\mathbf{X}}^{T+1}(t)\tilde{\mathbf{X}}^{T+1}(a_1)\tilde{\mathbf{X}}^{T+1}(a_2)\cdots\tilde{\mathbf{X}}^{T+1}(a_l)\right)$$

*Proof.* By Lemma 30, let  $\alpha$  be such that

$$\tilde{\mathbf{X}}^t(a_1)\tilde{\mathbf{X}}^t(a_2)\cdots\tilde{\mathbf{X}}^t(a_l) = \alpha\mathbf{v}(a_1)\mathbf{v}(a_l)^\top$$

and

$$\tilde{\mathbf{X}}^{T+1}(a_1)\tilde{\mathbf{X}}^{T+1}(a_2)\cdots\tilde{\mathbf{X}}^{T+1}(a_l) = \alpha\bar{\mathbf{v}}(a_1)\bar{\mathbf{v}}(a_l)^\top.$$

Note that:

$$\begin{aligned}\text{tr}\left(\tilde{\mathbf{X}}^t(t)\tilde{\mathbf{X}}^t(a_1)\tilde{\mathbf{X}}^t(a_2)\cdots\tilde{\mathbf{X}}^t(a_l)\right) &= \alpha\text{tr}\left(\tilde{\mathbf{X}}^t(t)\mathbf{v}(a_1)\mathbf{v}(a_l)^\top\right) \\ &= \alpha\text{tr}\left(\mathbf{v}(t)\mathbf{v}(t)^\top\mathbf{v}(a_1)\mathbf{v}(a_l)^\top\right) \\ &= \alpha\text{tr}\left(\mathbf{v}(a_l)^\top\mathbf{v}(t)\mathbf{v}(t)^\top\mathbf{v}(a_1)\right) \\ &= \alpha\left(\mathbf{v}(a_l)^\top\mathbf{v}(t)\right)\left(\mathbf{v}(t)^\top\mathbf{v}(a_1)\right) \\ &= \alpha\left(\frac{\mathcal{M}^+(i_{a_l}, i_t)}{2\mathcal{R}_M} + \frac{\mathcal{N}^+(j_{a_l}, j_t)}{2\mathcal{R}_N}\right)\left(\frac{\mathcal{M}^+(i_t, i_{a_1})}{2\mathcal{R}_M} + \frac{\mathcal{N}^+(j_t, j_{a_1})}{2\mathcal{R}_N}\right)\end{aligned}$$

Similarly we have:

$$\text{tr}\left(\tilde{\mathbf{X}}^{T+1}(t)\tilde{\mathbf{X}}^{T+1}(a_1)\tilde{\mathbf{X}}^{T+1}(a_2)\cdots\tilde{\mathbf{X}}^{T+1}(a_l)\right) \tag{49}$$

$$= \alpha\left(\frac{\mathcal{M}^+(i_{a_l}, i_t)}{2\mathcal{R}_M} + \frac{\mathcal{N}^+(j_{a_l}, j_t)}{2\mathcal{R}_N}\right)\left(\frac{\mathcal{M}^+(i_t, i_{a_1})}{2\mathcal{R}_M} + \frac{\mathcal{N}^+(j_t, j_{a_1})}{2\mathcal{R}_N}\right) \tag{50}$$

The result follows.  $\square$

**Lemma 32.** For any  $q \in \mathbb{N}$ , any  $\kappa \in \mathfrak{R}^+$  and any  $b_1, b_2, \dots, b_{t-1} \in \mathfrak{R}$  we have:

$$\text{tr}\left(\tilde{\mathbf{X}}^t(t)\left(\sum_{s \in \mathbb{U} \cap [t-1]} b_s \tilde{\mathbf{X}}^t(s)\right)^q\right) = \text{tr}\left(\tilde{\mathbf{X}}^{T+1}(t)\left(\sum_{s \in \mathbb{U} \cap [t-1]} b_s \tilde{\mathbf{X}}^{T+1}(s)\right)^q\right)$$

*Proof.* We have:

$$\begin{aligned}
& \text{tr} \left( \tilde{\mathbf{X}}^t(t) \left( \sum_{s \in \mathbb{U} \cap [t-1]} b_s \tilde{\mathbf{X}}^t(s) \right)^q \right) \\
&= \text{tr} \left( \tilde{\mathbf{X}}^t(t) \sum_{a_1 \in \mathbb{U} \cap [t-1]} \sum_{a_2 \in \mathbb{U} \cap [t-1]} \cdots \sum_{a_q \in \mathbb{U} \cap [t-1]} \left( \prod_{i=1}^q b_{a_i} \right) \tilde{\mathbf{X}}^t(a_1) \tilde{\mathbf{X}}^t(a_2) \cdots \tilde{\mathbf{X}}^t(a_q) \right) \\
&= \sum_{a_1 \in \mathbb{U} \cap [t-1]} \sum_{a_2 \in \mathbb{U} \cap [t-1]} \cdots \sum_{a_q \in \mathbb{U} \cap [t-1]} \left( \prod_{i=1}^q b_{a_i} \right) \text{tr} \left( \tilde{\mathbf{X}}^t(t) \tilde{\mathbf{X}}^t(a_1) \tilde{\mathbf{X}}^t(a_2) \cdots \tilde{\mathbf{X}}^t(a_q) \right)
\end{aligned}$$

and similarly,

$$\begin{aligned}
& \text{tr} \left( \tilde{\mathbf{X}}^{T+1}(t) \left( \sum_{s=1}^{t-1} b_s \tilde{\mathbf{X}}^{T+1}(s) \right)^q \right) \\
&= \text{tr} \left( \tilde{\mathbf{X}}^{T+1}(t) \sum_{a_1 \in \mathbb{U} \cap [t-1]} \sum_{a_2 \in \mathbb{U} \cap [t-1]} \cdots \sum_{a_q \in \mathbb{U} \cap [t-1]} \left( \prod_{i=1}^q b_{a_i} \right) \tilde{\mathbf{X}}^{T+1}(a_1) \tilde{\mathbf{X}}^{T+1}(a_2) \cdots \tilde{\mathbf{X}}^{T+1}(a_q) \right) \\
&= \sum_{a_1 \in \mathbb{U} \cap [t-1]} \sum_{a_2 \in \mathbb{U} \cap [t-1]} \cdots \sum_{a_q \in \mathbb{U} \cap [t-1]} \left( \prod_{i=1}^q b_{a_i} \right) \text{tr} \left( \tilde{\mathbf{X}}^{T+1}(t) \tilde{\mathbf{X}}^{T+1}(a_1) \tilde{\mathbf{X}}^{T+1}(a_2) \cdots \tilde{\mathbf{X}}^{T+1}(a_q) \right).
\end{aligned}$$

The result follows by Lemma 31.  $\square$

**Lemma 33.** For any  $\kappa \in \mathfrak{R}^+$  and any  $b_1, b_2, \dots, b_{t-1} \in \mathfrak{R}$  we have:

$$\text{tr} \left( \tilde{\mathbf{X}}^t(t) \exp \left( \kappa \mathbf{I} + \sum_{s \in \mathbb{U} \cap [t-1]} b_s \tilde{\mathbf{X}}^t(s) \right) \right) = \text{tr} \left( \tilde{\mathbf{X}}^{T+1}(t) \exp \left( \kappa \mathbf{I} + \sum_{s \in \mathbb{U} \cap [t-1]} b_s \tilde{\mathbf{X}}^{T+1}(s) \right) \right)$$

*Proof.* Using the fact that  $\exp(\mathbf{A} + \mathbf{B}) = \exp(\mathbf{A}) \exp(\mathbf{B})$  for commuting matrices  $\mathbf{A}$  and  $\mathbf{B}$ , and noting that the multiple of the identity matrix commutes with any matrix, we have that

$$\text{tr} \left( \tilde{\mathbf{X}}^t(t) \exp \left( \kappa \mathbf{I} + \sum_{s \in \mathbb{U} \cap [t-1]} b_s \tilde{\mathbf{X}}^t(s) \right) \right) = \text{tr} \left( \tilde{\mathbf{X}}^t(t) \exp(\kappa \mathbf{I}) \exp \left( \sum_{s \in \mathbb{U} \cap [t-1]} b_s \tilde{\mathbf{X}}^t(s) \right) \right).$$

By the Taylors series expansion we have:

$$\begin{aligned}
\text{tr} \left( \tilde{\mathbf{X}}^t(t) \exp(\kappa \mathbf{I}) \exp \left( \sum_{s \in \mathbb{U} \cap [t-1]} b_s \tilde{\mathbf{X}}^t(s) \right) \right) &= e^\kappa \text{tr} \left( \tilde{\mathbf{X}}^t(t) \sum_{q=0}^{\infty} \frac{1}{q!} \left( \sum_{s \in \mathbb{U} \cap [t-1]} b_s \tilde{\mathbf{X}}^t(s) \right)^q \right) \\
&= e^\kappa \sum_{q=0}^{\infty} \frac{1}{q!} \text{tr} \left( \tilde{\mathbf{X}}^t(t) \left( \sum_{s \in \mathbb{U} \cap [t-1]} b_s \tilde{\mathbf{X}}^t(s) \right)^q \right)
\end{aligned}$$

Similarly, we have

$$\text{tr} \left( \tilde{\mathbf{X}}^{T+1}(t) \exp \left( \kappa \mathbf{I} + \sum_{s \in \mathbb{U} \cap [t-1]} b_s \tilde{\mathbf{X}}^{T+1}(s) \right) \right) = e^\kappa \sum_{q=0}^{\infty} \frac{1}{q!} \text{tr} \left( \tilde{\mathbf{X}}^{T+1}(t) \left( \sum_{s \in \mathbb{U} \cap [t-1]} b_s \tilde{\mathbf{X}}^{T+1}(s) \right)^q \right).$$

The result then follows from Lemma 32.  $\square$

## D.1.2 Equivalence of Algorithms

On a trial  $t$  let  $\bar{z}_t$  be the prediction ( $\bar{y}_t$ ) of the inductive algorithm and let  $\bar{y}_t$  remain the prediction of the transductive algorithm. We fix  $\kappa := \log \left( \widehat{\mathcal{D}} / (m+n) \right)$ .

**Lemma 34.** *On a trial  $t$  the prediction,  $\bar{y}_t$ , of the transductive algorithm is given by:*

$$\bar{y}_t = \text{tr} \left( \tilde{\mathbf{X}}^{T+1}(t) \exp \left( \kappa \mathbf{I} + \sum_{s=1}^{t-1} f_s(\bar{y}_s) \tilde{\mathbf{X}}^{T+1}(s) \right) \right)$$

and the prediction,  $\bar{z}_t$ , of the inductive algorithm is given by:

$$\bar{z}_t = \text{tr} \left( \tilde{\mathbf{X}}^t(t) \exp \left( \kappa \mathbf{I} + \sum_{s=1}^{t-1} f_s(\bar{z}_s) \tilde{\mathbf{X}}^t(s) \right) \right)$$

where  $f_s(x) := \eta y_s$  if  $y_s x \leq [\text{NON-CONSERVATIVE}] \times \gamma$  and  $f_s(x) := 0$  otherwise.

*Proof.* Direct from algorithms, noting that if  $s \notin \mathbb{U} \cap [t-1]$  then  $f_s(\bar{z}_s) = 0$ .  $\square$

**Lemma 35.** *Given a trial  $t$ , if  $\bar{y}_s = \bar{z}_s$  for all  $s < t$ , then  $\bar{y}_t = \bar{z}_t$ .*

*Proof.* Direct from Lemmas 34 and 33 (with  $b_s := f_s(\bar{y}_t) = f_s(\bar{z}_t)$ ), noting that if  $s \notin \mathbb{U} \cap [t-1]$  then  $f_s(\bar{z}_s) = 0$ .  $\square$

Proposition 4 follows by induction over Lemma 35.  $\blacksquare$

## D.2 Proof of Proposition 5

In the following, we define  $\mathcal{K}_x(\cdot) := \mathcal{K}(\mathbf{x}, \cdot)$ . If  $r \geq 2$ ,  $\delta^* := \min \left( 2, \frac{1}{4} \delta(S_1, \dots, S_k) \right)$ . This implies that  $\delta^* \leq \min \left( 2, \frac{r-1}{2r} \delta(S_1, \dots, S_k) \right)$ . Recall that  $s(\mathbf{x}) := \frac{r-1}{2r} \mathbf{x} + \frac{r+1}{2} \mathbf{1}$ . Then observe that, given that the transformation  $\tilde{\mathbf{x}}_i = s(\mathbf{x}_i)$  holds true for all  $i \in [m]$ , requiring  $S_1, \dots, S_k \subset [-r, r]^d$  with  $\mathbf{x}_1, \dots, \mathbf{x}_m \in \cup_{i=1}^k S_i$  and  $\delta^* \leq \min \left( 2, \frac{r-1}{2r} \delta(S_1, \dots, S_k) \right)$  is equivalent to the requirement that  $\tilde{S}_1, \dots, \tilde{S}_k \subset [1, r]^d$  with  $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_m \in \cup_{i=1}^k \tilde{S}_i$  and  $\delta^* \leq \min \left( 2, \delta(\tilde{S}_1, \dots, \tilde{S}_k) \right)$ . Furthermore, for all  $i \in [m]$  and  $j \in [k]$ , we have that  $\mathbf{x}_i \in S_j$  if and only if  $\tilde{\mathbf{x}}_i \in \tilde{S}_j$ . We shall proceed with the latter set of requirements for simplicity. Recall that the RKHS for the  $d = 1$  min kernel  $\mathcal{H}_{\mathcal{K}}^1$  is the set of all absolutely continuous functions from  $[0, \infty)^d \rightarrow \mathfrak{R}$  that satisfy  $f(0) = 0$  and  $\int_0^\infty [f'(x)]^2 dx < \infty$ .

**Lemma 36.** *The inner product for  $f \in \mathcal{H}_{\mathcal{K}}^1$  may be computed by,*

$$\langle f, g \rangle = \int_0^\infty f'(x) g'(x) dx.$$

*Proof.* We show this by the reproducing property:

$$\langle f, \mathcal{K}_x \rangle = f(x).$$

Defining  $\mathbf{1}_x(t)$  as the step function that evaluates to 1 for  $t \leq x$  and 0 otherwise, we note that the derivative of  $\min(x, t)$  with respect to  $t$  is equal to  $\mathbf{1}_x(t)$ . This gives rise to

$$\int_0^\infty f'(t) \mathcal{K}'(x, t) dt = \int_0^\infty f'(t) \mathbf{1}_x(t) dt = \int_0^x f'(t) dt = f(x).$$

Using the condition of  $f(0) = 0$ , we then obtain the reproducing property.  $\square$

**Lemma 37.** *Given  $k$  boxes  $\tilde{S}_1, \dots, \tilde{S}_k \subset [1, r]^d$ ,  $\delta^* \leq \min \left( 2, \delta(\tilde{S}_1, \dots, \tilde{S}_k) \right)$  and  $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_m \in \cup_{i=1}^k \tilde{S}_i$ , there exists a function  $f \in H_{\mathcal{K}}$  for which  $f(\tilde{\mathbf{x}}_j) = [\tilde{\mathbf{x}}_j \in \tilde{S}_1]$  for  $j \in [m]$  and this function has norm*

$$\|f\|^2 = \left( \frac{4}{\delta^*} \right)^d.$$

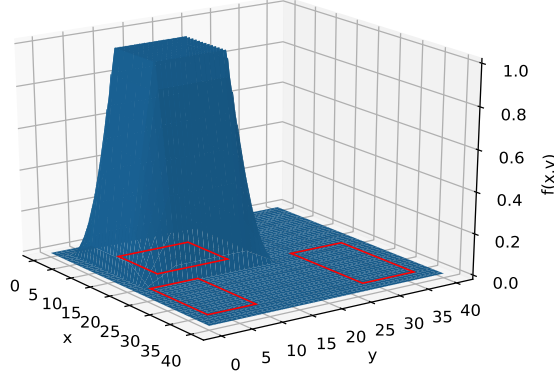


Figure 3: Visualization of the function  $f(x_1, x_2)$  with  $S_1, S_2$  and  $S_3$  represented as red rectangles in the  $x_1 - x_2$  plane.

*Proof.* Recall that a *box* in  $\mathbb{R}^d$  is a set  $\{\mathbf{x} : a_i \leq x_i \leq b_i, i \in [d]\}$  defined by a pair of vectors  $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$ . First, we consider the case of  $d = 1$ , with the coordinates of  $\tilde{S}_1$  defined by  $a$  and  $b$ . Defining the function that interpolates the points  $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_m$  in one dimension as  $f^1 \in \mathcal{H}_{\mathcal{K}}^1$ , we chose  $f^1$  to be the following:

$$f^1(x) = \begin{cases} 0 & \text{for } x \leq a - \frac{\delta^*}{2} \\ \frac{2}{\delta^*}x + 1 - \frac{2}{\delta^*}a & \text{for } a - \frac{\delta^*}{2} < x \leq a \\ 1 & \text{for } a < x \leq b \\ -\frac{2}{\delta^*}x + 1 + \frac{2}{\delta^*}b & \text{for } b < x \leq b + \frac{\delta^*}{2} \\ 0 & \text{for } x > b + \frac{\delta^*}{2}. \end{cases}$$

This function is picked from the space  $\mathcal{H}_{\mathcal{K}}^1$  so that  $\int_0^\infty [(f^1)'(x)]^2 dx$  is minimized with respect to “worst-case” constraints. The condition on  $\delta^*$  implies that  $\delta^* \leq 2$ , so that  $f^1(0) = 0$ . It also implies that  $\delta^* \leq \delta(S_1, \dots, S_k)$  so that for all  $i \in [m]$ ,  $f^1(\tilde{x}_i) = 0$  if  $\tilde{x}_i \notin S_1$ . The norm  $\|f^1\|^2$ , then becomes

$$\begin{aligned} \|f^1\|^2 &= \int_0^\infty [(f^1)'(x)]^2 dx \\ &= \int_{a-\frac{\delta^*}{2}}^a \left(\frac{2}{\delta^*}\right)^2 dx + \int_b^{b+\frac{\delta^*}{2}} \left(\frac{2}{\delta^*}\right)^2 dx \\ &= 2 \left(\frac{2}{\delta^*}\right)^2 \left(\frac{\delta^*}{2}\right) = \frac{4}{\delta^*}. \end{aligned}$$

This can be extended to multiple dimensions by observing that the induced product norm of  $f$  is the product of the norms of  $f^1$  in each dimension, thus giving the required bound. In this case also, the condition on  $\delta^*$  ensures both  $f(\mathbf{0}) = 0$  and  $f(\tilde{\mathbf{x}}_i) = 0$  for  $\tilde{\mathbf{x}}_i \notin \tilde{S}_1$ , where  $i \in [m]$ . For an illustration of this function in two dimensions, see Figure 3.  $\square$

**Lemma 38.** *Given  $k$  boxes  $\tilde{S}_1, \dots, \tilde{S}_k \subset [1, r]^d$ ,  $\delta^* \leq \min(2, \delta(\tilde{S}_1, \dots, \tilde{S}_k))$  and  $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_m \in \cup_{i=1}^k \tilde{S}_i$ , if  $\mathbf{u} = (u_i = [\tilde{\mathbf{x}}_i \in \tilde{S}_1])_{i \in [m]}$  and  $\mathbf{K} = (\mathcal{K}(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j))_{i, j \in [m]}$  then  $\mathbf{u}^\top \mathbf{K}^{-1} \mathbf{u} \leq \left(\frac{4}{\delta^*}\right)^d$ .*

*Proof.* Using Lemma 37 we observe that,

$$\mathbf{u}^\top \mathbf{K}^{-1} \mathbf{u} = \underset{f \in \mathcal{H}_{\mathcal{K}}: f(\tilde{\mathbf{x}}_i) = [\tilde{\mathbf{x}}_i \in \tilde{S}_1], i \in [m]}{\operatorname{argmin}} \|f\|_{\mathcal{K}}^2 \leq \underset{f \in \mathcal{H}_{\mathcal{K}}: f(\tilde{\mathbf{x}}) = [\tilde{\mathbf{x}} \in \tilde{S}_1], \tilde{\mathbf{x}} \in \cup_{i \in [k]} \tilde{S}_i}{\operatorname{argmin}} \|f\|_{\mathcal{K}}^2 \leq \left(\frac{4}{\delta^*}\right)^d,$$

for  $\mathbf{u} := (u_i = [\tilde{\mathbf{x}}_i \in \tilde{S}_1])_{i \in [m]}$ ,  $\mathbf{K} := (\mathcal{K}(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j))_{i, j \in [m]}$  and  $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_m \in \cup_{i \in [k]} \tilde{S}_i$ . Note that the second term in the equation above has a constraint in terms of the given  $m$  points  $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_m$ , whereas the optimization in the third term has a similar constraint, but in terms of any  $\tilde{\mathbf{x}}$  that satisfies  $\tilde{\mathbf{x}} \in \cup_{i \in [k]} \tilde{S}_i$ .  $\square$

Defining  $\mathbf{u}_i = (\mathbf{R}_i^\top)^\top$ , then observe that the term  $\text{tr}(\mathbf{R}^\top \mathbf{K}^{-1} \mathbf{R}) = \sum_{i \in [k]} \mathbf{u}_i^\top \mathbf{K}^{-1} \mathbf{u}_i$ . Thus by applying Lemma 38 to each  $\mathbf{u}_i$ , we have that  $\text{tr}(\mathbf{R}^\top \mathbf{K}^{-1} \mathbf{R}) \leq k \left(\frac{4}{\delta^*}\right)^d$ .  $\blacksquare$