Lovasz Theta Contrastive Learning

Georgios Smyrnis **Matt Jordan** **Ananya Uppal**
Giannis Daras **Alexandros G. Dimakis**
The University of Texas at Austin, Austin, Texas USA

Abstract

We establish a connection between the Lovasz theta function of a graph and the widely used InfoNCE loss. We show that under certain conditions, the minima of the InfoNCE loss are related to that of the Lovasz theta function on the empty similarity graph between samples. Building on this connection, we generalize contrastive learning on weighted similarity graphs between samples. Our Lovasz theta contrastive loss uses a weighted graph that can be learned to take into account similarities between our data. We evaluate our method on image classification tasks, demonstrating an improvement of 1% in the supervised case and up to 4% in the unsupervised case.

1 Introduction

The Lovasz theta function is a fundamental quantity in graph theory. It can be viewed as the natural semidefinite relaxation of the graph independence number and was defined by Laszlo Lovasz to determine the Shannon capacity of the 5-cycle graph [1] solving a problem that had been open in combinatorics for more than 20 years. This work subsequently inspired semidefinite approximation algorithms [2] and perfect graph theory [3]. The Lovasz theta function requires the computation of a graph representation: for a given undirected graph $G(V, E)$ we would like to find unit norm vectors $\boldsymbol{v}_i$ where $i \in V$, such that non-adjacent vertices have orthogonal representations - in other words, $\boldsymbol{v}_i^T \boldsymbol{v}_j = 0$, if $\{i, j\} \notin E$. The Lovasz theta function searches for a graph representation that fits all these vectors in a small spherical cap, which is obtained via a semidefinite program (SDP) [4].

Contrastive learning is also a representation learning technique that has been very successful recently (e.g. [5, 6, 7]). The goal is to learn clustered representations for similar samples, while pulling different ones apart. This can be done in either an unsupervised fashion (i.e. without labels) or in a supervised way [8]. Contrastive learning approaches typically consider samples to be either similar (positive) or different (negative). However, some problems have variability in similarity: Images of cats are closer to dogs compared to airplanes, and this insight can benefit representation learning.

Our Contributions: We establish a connection between contrastive learning and the Lovasz theta function. We prove that the minimizers of the InfoNCE loss are the same (up to rotations) as those of the Lovasz theta optimum graph representation using an empty similarity graph. This allows us to link a classical problem in graph theory with the commonly used technique of contrastive learning. Using this connection, we generalize contrastive learning using Lovasz theta on *weighted* graphs [9]. We define the Lovasz theta contrastive loss for a weighted graph representing similarities between samples in each batch. This way, any image similarity metric can be used to strengthen contrastive learning. For unsupervised contrastive learning, we show that our method can yield a benefit of up to 4% over SimCLR for CIFAR100 using a pre-trained CLIP image encoder to obtain similarities. For supervised contrastive learning (i.e. if class structure is used) our method yields a 1% benefit over supervised contrastive learning in CIFAR100 [8]. We refer the reader to the Appendix for discussion on the related works.

Correspondence at gsmyrnis@utexas.edu.
Code available at https://github.com/GeorgiosSmyrnis/LovaszContrastive.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

Figure 1: **Key idea of our method.** In this figure we show how our proposed method works, with respect to the similarity graph between classes. In regular supervised contrastive learning (on the left), the similarity graph is just the empty graph. While in our method (on the right), the similarity graph reflects that the classes of dogs are similar to each other and different from cats (edge boldness reflects weight magnitude).

2 Lovasz Theta and Contrastive Learning

2.1 The Lovasz Theta Function

Definition 2.1 (Lovasz Theta of a Graph)**.** Let $G = (V, E)$ be a given (unweighted) graph. Then we define as Lovasz theta of this graph, denoted as $\theta(G)$, the optimal value of the following:

$$\begin{array}{cc} \underset{\boldsymbol{u}_1,\ldots,\boldsymbol{u}_N,\boldsymbol{c}\in\mathbb{R}^d}{\text{minimize}} & \max_{i=1,\ldots,N} \frac{1}{(\boldsymbol{c}^T\boldsymbol{u}_i)^2}, \\ \text{s.t.} & \|\boldsymbol{u}_1\|^2 = \|\boldsymbol{u}_2\|^2 = \cdots = \|\boldsymbol{u}_N\|^2 = \|\boldsymbol{c}\|^2 = 1, \\ & \boldsymbol{u}_i^T\boldsymbol{u}_j = 0,\ \forall (i,j) \notin E. \end{array} \tag{1}$$

In the following, we shall use the Delsarte formulation of the Lovasz theta problem [9, 10], replacing $\boldsymbol{u}_i^T\boldsymbol{u}_j = 0$ by $\boldsymbol{u}_i^T\boldsymbol{u}_j \leq 0$ (we will see that this is more amenable to an extension on weighted graphs.

Theorem 2.1 ([4])**.** *The Delsarte formulation of the Lovasz theta problem can be written as:*

$$\begin{array}{cc} \underset{\boldsymbol{u}_1,\ldots,\boldsymbol{u}_N\in\mathbb{R}^d,\ t\in\mathbb{R}}{\text{minimize}} & t, \\ \text{s.t} & \boldsymbol{v}_i^T\boldsymbol{v}_j \leq t,\ \forall (i,j) \notin E, \\ & \|\boldsymbol{u}_1\|^2 = \|\boldsymbol{u}_2\|^2 = \cdots = \|\boldsymbol{u}_N\|^2 = 1. \end{array} \tag{2}$$

The above problem can be converted into a convex SDP by setting $\mathbf{Y} = \left[\boldsymbol{v}_i^T\boldsymbol{v}_j\right]_{i,j}$ (so we have $\mathbf{Y} \succeq 0$). For each vertex in the graph we want to find a unit representation, such that the vertices which are not connected by an edge have dissimilar representations (low inner product).

2.2 Contrastive Learning

One of the most well-known losses used for contrastive learning is the InfoNCE loss [5, 11, 12, 13]:

$$\mathcal{L}_{InfoNCE} = -\sum_{i=1}^{N} \log \frac{\exp(\boldsymbol{v}_+^T\boldsymbol{v}_i/\tau)}{\sum_{j\neq i}\exp(\boldsymbol{v}_j^T\boldsymbol{v}_i/\tau)}, \tag{3}$$

where $\boldsymbol{v}_+$ is the positive sample with respect to $\boldsymbol{v}_i$, and τ is a temperature parameter. For the theoretical analysis, we shall make the following simplifying assumptions: a) the representation vectors $\boldsymbol{v}_I \in \mathbb{R}^d$ are all unit norm and $N \leq d$, and b) the only positive sample with respect to $\boldsymbol{v}_i$ is itself (single positive case). We discuss the necessity and the implications of our assumptions in Appendix B. Under these assumptions, we can reformulate our loss as follows:

$$\mathcal{L}_{InfoNCE} = \text{const} + \tau\sum_{i=1}^{N} \log\left(\sum_{j\neq i}\exp(\boldsymbol{v}_j^T\boldsymbol{v}_i/\tau)\right), \tag{4}$$

where $\|\boldsymbol{v}_i\|_2^2 = 1$, for all i. Here we multiply the loss by $\tau > 0$, which doesn't affect the optimum. Intuitively, if we construct a graph of images where positive examples are connected, then the InfoNCE loss and the objective in the Delsarte formulation of Lovasz theta minimize the dot product between the embeddings of non-adjacent vertices. We formalize this equivalence in the following theorem:

Theorem 2.2 (Equivalence with Lovasz theta)**.** *The above formulation of the InfoNCE loss, for $\tau > 0$, has the same minimizers (up to rotations) as those of the Lovasz theta problem over the empty graph, using the Delsarte formulation.*

We can also consider the case of the Supervised Contrastive Loss [8]. In this setting, the positive samples for each image are all images belonging to the same class:

$$L_{SupCon} = -\frac{1}{\tau}\sum_{i=1}^{N}\frac{1}{|P(i)|}\sum_{p\in P(i)}\boldsymbol{v}_p^T\boldsymbol{v}_i + \sum_{i=1}^{N}\log\left(\sum_{j\in N(i)}\exp(\boldsymbol{v}_j^T\boldsymbol{v}_i/\tau)\right), \tag{5}$$

where $P(i) = \{j : y_i = y_j\}$ and $N(i) = \{j : y_i \neq y_j\}$ the sets of positive and negative samples.

2.3 Extension to Similarity Graphs

Let us assume that we have a weighted graph $G(V, W)$, with vertices being samples and the edge weights w_{ij} being similarities. We use a weighted version of the Lovasz theta problem:

$$\begin{array}{cc} \underset{\boldsymbol{u}_1,\ldots,\boldsymbol{u}_N,\boldsymbol{c}\in\mathbb{R}^d}{\text{minimize}} & \max_{i=1,\ldots,N}\frac{1}{(\boldsymbol{c}^T\boldsymbol{u}_i)^2}, \\ \text{s.t.} & \|\boldsymbol{u}_1\|^2 = \|\boldsymbol{u}_2\|^2 = \cdots = \|\boldsymbol{u}_N\|^2 = \|\boldsymbol{c}\|^2 = 1, \\ & \boldsymbol{u}_i^T\boldsymbol{u}_j \leq w_{ij},\ \forall i \neq j, \end{array} \tag{6}$$

Lemma 2.1. *For $N \leq d$, we can rewrite* (6) *as follows:*

$$\begin{array}{cc} \underset{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N\in\mathbb{R}^d,t\in\mathbb{R}}{\text{minimize}} & t, \\ \text{s.t.} & \|\boldsymbol{v}_1\|^2 = \|\boldsymbol{v}_2\|^2 = \cdots = \|\boldsymbol{v}_N\|^2 = 1, \\ & \boldsymbol{v}_i^T\boldsymbol{v}_j \leq w_{ij} + (1 - w_{ij})t,\ \forall i \neq j. \end{array} \tag{7}$$

Leveraging this graph structure, we can create the following form of contrastive loss, which also takes into account sample similarities:

$$L_{LovaszCon} = -\sum_{i=1}^{N}\frac{1}{|P(i)|}\sum_{p\in P(i)}\boldsymbol{v}_p^T\boldsymbol{v}_i + \tau\sum_{i=1}^{N}\log\left(\sum_{j\neq i}\exp\left(\frac{\boldsymbol{v}_j^T\boldsymbol{v}_i - w_{ij}}{\tau(1 - w_{ij})}\right)\right). \tag{8}$$

Compared to L_{SupCon}, we have multiplied the loss function by $\tau > 0$. For finite $\tau > 0$, this makes no difference in the minimization, but for $\tau \to 0$, this allows us to state the following:

Theorem 2.3 (Informal)**.** *The minimization of the second term of* (8) *for $\tau \to 0$ is a relaxation of the weighted version of the Lovasz theta in* (7)*, if $w_{ij} < 1$ for $i \neq j$.*

We defer all the formal statements and proofs to the Appendix, as well as discussion on the similarity graph used. The above allows us to leverage (2) to transform the loss we use into (8). We always make use of the positives samples (or views in the unsupervised case) available in the first term.

3 Experiments

We now examine the above formulation experimentally. We first train our models using the Lovasz theta contrastive loss, and then freeze the learned representations and learn a linear model on top of them. Our metric is the performance of this linear classifier, as in Khosla et al. [8]. We also employ a technique commonly used in contrastive learning frameworks, where the contrastive loss is applied on the output of a small head using the representations as input, namely on $\boldsymbol{z} = g_{head}(\boldsymbol{v})$ [8, 5], which is then discarded. Further experimental details and results can be found in the Appendix.

Unsupervised Similarity Matrix. In this setting, we derive our similarity matrix using the unsupervised approach mentioned above. We use CLIP to derive the cosine similarities between samples, and use them to perform Lovasz theta contrastive training on our method. Note that we do not train CLIP from scratch, nor do we use data from our task. We assume CLIP to be an externally provided model that could be easily swapped for a different one. The results of our method can be seen in Table 1, where it outperforms regular SimCLR pretraining on CIFAR100. Thus, we can take advantage of sample similarities even if labels aren't immediately available during contrastive pretraining.

Table 1: **Summary of our results on CIFAR100, unsupervised case.** We compare our method with SimCLR, using the implementation provided by the authors of [8].

	SimCLR	Ours
ResNet-50, 300 epochs	64.42	**68.14 ± 0.46**
ResNet-18, 300 epochs	58.64	**60.70 ± 0.73**
ResNet-50, 1000 epochs	68.27	**70.30 ± 0.40**
ResNet-18, 1000 epochs	63.41	**65.24 ± 0.19**

Table 2: **Summary of our results on CIFAR100, supervised case.** Our method outperforms training with just the crossentropy loss (CE) and the supervised contrastive baseline (SupCon). We construct the similarity matrix via the confusion matrix, or via the superclasses in CIFAR100. Numbers with * are from [8].

	CE	SupCon	Ours (Confusion Matrix)	Ours (Superclass)
ResNet-50	75.3*	76.5*	77.15 ± 0.11	**77.60 ± 0.30**
ResNet-34	74.98	75.81	76.06 ± 0.29	**76.55 ± 0.40**
ResNet-18	72.67	73.78	74.63 ± 0.18	**74.91 ± 0.15**

Table 3: **Results on our ImageNet-100 experiment.** We can see that our method yields slightly better accuracy over both classical Cross Entropy loss and Supervised Contrastive Learning.

	CE	SupCon	Ours
ResNet-18	81.08	81.20	**81.58**
ResNet-34	82.52	82.66	**82.90**

Supervised Similarity Matrix. In this setting, we compare our method to two supervised baselines, using a simple crossentropy loss (CE) in a supervised fashion and using supervised contrastive learning (SupCon) [8]. We use the evaluation process we described above, on CIFAR100 and ImageNet-100 (a subset of 100 classes of ImageNet [14]) using 3 different ResNet architectures. For CIFAR100, we use the following two choices for the similarity matrix a) via the confusion matrix, derived from a model trained with SupCon, and b) via the 20 superclasses defined over the 100 classes of CIFAR100 [15], with similarity between two classes in the same superclass being a hyperparameter equal to 0.5 We can see our results in the Tables 2 and 3. From our results on CIFAR100 we can infer that the use of the similarity matrix is helpful to the training process. Intuitively, relaxing the constraint on the negatives allows the loss to achieve a better representation overall. A similar conclusion can be obtained from our results on ImageNet-100, where again our method performs better than the baselines. Results on CIFAR10 can be seen in the Appendix.

Note that the above choices for the similarity matrices are not the only ones we could make. This is especially true for the unsupervised case, where we could either use a different model to extract similarities or bootstrap our own similarities during training. This is an interesting direction for future research.

4 Conclusion

We established a connection between the InfoNCE loss and the Lovasz theta of a graph. This allowed us to generalize contrastive learning using general similarity graphs. Our technique can use any method of measuring similarity between samples to create a problem-specific constrastive loss. Our experiments show benefits over regular contrastive learning, in both the supervised and the unsupervised case, using simple similarity metrics. This is natural since we use additional information provided in the similarity structure. The design decision of how to measure similarity between samples is central for our loss and opens an interesting direction for future research.

Acknowledgements

This research has been supported by NSF Grants CCF 1763702, AF 1901292, CNS 2148141, Tripods CCF 1934932, IFML CCF 2019844 and research gifts by Western Digital, WNCG IAP, UT Austin Machine Learning Lab (MLL), Cisco and the Archie Straiton Endowed Faculty Fellowship. This scientific paper was also supported by the Onassis Foundation - Scholarship ID: F ZS 056-1/2022-2023.

References

[1] László Lovász. On the shannon capacity of a graph. *IEEE Transactions on Information theory*, 25(1):1–7, 1979.

[2] Michel X Goemans and David P Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM)*, 42(6):1115–1145, 1995.

[3] Claude Berge. *The theory of graphs*. Courier Corporation, 2001.

[4] Bernd Gärtner and Jiri Matousek. *Approximation algorithms and semidefinite programming*. Springer Science & Business Media, 2012.

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[6] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[7] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[8] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673, 2020.

[9] Fredrik D Johansson, Ankani Chattoraj, Chiranjib Bhattacharyya, and Devdatt Dubhashi. Weighted theta functions and embeddings with applications to max-cut, clustering and summarization. *Advances in Neural Information Processing Systems*, 28, 2015.

[10] Alexander Schrijver. A comparison of the delsarte and lovász bounds. *IEEE Transactions on Information Theory*, 25(4):425–429, 1979.

[11] Zhirong Wu, Yuanjun Xiong, Stella Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance-level discrimination. *arXiv preprint arXiv:1805.01978*, 2018.

[12] Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *International Conference on Machine Learning*, pages 4182–4192. PMLR, 2020.

[13] Aaron Van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv e-prints*, pages arXiv–1807, 2018.

[14] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *European conference on computer vision*, pages 776–794. Springer, 2020.

[15] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[16] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[17] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, 2021.

[18] Joshua David Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. In *ICLR*, 2021.

[19] Jeff Z HaoChen, Colin Wei, Adrien Gaidon, and Tengyu Ma. Provable guarantees for self-supervised deep learning with spectral contrastive loss. *Advances in Neural Information Processing Systems*, 34:5000–5011, 2021.

[20] Kendrick Shen, Robbie M Jones, Ananya Kumar, Sang Michael Xie, Jeff Z HaoChen, Tengyu Ma, and Percy Liang. Connect, not collapse: Explaining contrastive learning for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 19847–19878. PMLR, 2022.

[21] Peng Cui, Shaowei Liu, and Wenwu Zhu. General knowledge embedded image representation learning. *IEEE Transactions on Multimedia*, 20(1):198–207, 2017.

[22] Shu Zhang, Ran Xu, Caiming Xiong, and Chetan Ramaiah. Use all the labels: A hierarchical multi-label contrastive learning framework. *arXiv preprint arXiv:2204.13207*, 2022.

Appendix

A Related Work

A.1 Lovasz Theta Function

The Lovasz theta function [1] is a quantity which has been used to approximate the chromatic number of a graph. This is done by obtaining a representation for each vertex, which is clustered around a certain "handle" vector. One of the most important aspects of this function is that it is easily computable by solving an SDP [4], despite the fact that the chromatic number it is used to approximate is very difficult to calculate in general.

A.2 Self-supervised Contrastive Learning

There has been a flurry of activity in self-supervision, e.g. [5, 6, 16, 17, 11]. The main goal is to learn general representations that are good for a variety of downstream tasks. Ideally, we want to learn representations that are more general than those obtained by training classifiers. [11] improved classification accuracy on ImageNet by a large margin over the state of the art. This was further simplified and improved by Chen et al. [5] by emphasizing the importance of data augmentations. More recent works [16, 6, 17] extend this by maximizing mutual information, adding a momentum encoder or a Siamese network, respectively. Crucially, these works rely on large datasets of unlabeled data to learn quality representations that can then be applied to supervised learning tasks.

Recently, there has been work on mining hard negative samples for contrastive learning [18]. This line of work considers negative samples that are similar to a given sample as hard negatives, and uses those to perform contrastive learning. While related to our work, this approach is orthogonal, since it does not assume that the similarity between samples is a semantic relationship that we want to preserve.

Finally, a pair of works related to ours are those of HaoChen et al. [19] and Shen et al. [20]. These works formulate the notion of a relationship graph between the given samples, by assuming that two samples are connected via an edge with weight equal to the probability that they correspond to views of the same sample. This is a natural measure of similarity between samples. In our work, we shall explicitly use these similarities to perform contrastive learning.

A.3 Supervised Contrastive Learning

Along this line, Khosla et al. [8] argued that when available we could leverage label information to learn good representations without using large amounts of unlabeled data. They use a contrastive loss to pull together representations of samples belonging to the same class and push them apart from those of other samples. Training under this supervised contrastive loss shows an improvement on the classification accuracy on ImageNet when compared to the cross-entropy loss, without using any extra data.

A.3.1 Multi-Label Contrastive Learning

Several works such as Radford et al. [7] and Cui et al. [21] use contrastive learning along with supervisory multi-labels such as tags to learn good representations. In particular, CLIP [7] maximizes the alignment between representations of an image and those of its captions.

The most closely related to our work is the recent paper Zhang et al. [22] that leverages similarity gleaned from the multi-labels of the data. They define class similarity by using the hierarchical information which can be derived by the multi-label of the sample. They incorporate class similarity information in the contrastive loss by adding a penalty factor for the pairs of images from different classes. The penalty term pushes together representations for images that have similar labels higher in the hierarchy of the multi-label. Thus, they define class similarity by the level at which the labels are similar in the multi-label hierarchy. Note that if the labels are only of a single level, then this loss reduces to regular supervised contrastive loss.

In contrast to this work, our loss formulation is derived from a principled theoretical connection between contrastive loss and the Lovasz theta function of a graph. We define a new generalization

of the contrastive loss that is derived from the Lovasz theta function. Moreover, we can easily incorporate similarities between samples into our setting, either via class similarities as in Zhang et al. [22], or via directly assigning similarity to samples.

B Deferred Theorems & Proofs

In this section, we include further theoretical results, including the proofs omitted from the main body of the paper.

On the assumption of a single positive sample. This assumption allows us to disregard the effect of the positive term and directly link the InfoNCE loss with the Lovasz theta formulation. In order to account for multiple positives, we could also solve the following problem, inspired by the Lovasz theta formulation:

$$\begin{array}{cc} \underset{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N\in\mathbb{R}^d,t,s\in\mathbb{R}}{\text{minimize}} & t-s, \\ \text{s.t.} & \|\boldsymbol{v}_1\|^2=\|\boldsymbol{v}_2\|^2=\cdots=\|\boldsymbol{v}_N\|^2=1, \\ & \boldsymbol{v}_i^T\boldsymbol{v}_j\leq t,\ \forall(i,j)\in N \\ & \boldsymbol{v}_i^T\boldsymbol{v}_j\geq s,\ \forall(i,j)\in P \end{array} \tag{9}$$

where P and N are the sets of positive and negative pairs respectively. The above problem is still convex with respect to $\mathbf{Y}=\mathbf{V}^T\mathbf{V}$, assuming $N\leq d$, even if it doesn't precisely correspond to the InfoNCE loss. However, we elect to keep the single positive assumption to more cleanly demonstrate the connection of our loss to the Lovasz theta problem.

On the assumption $N\leq d$. The assumption on the number of vectors allows us to convert the Lovasz theta problem into a convex one. Indeed, without this assumption, converting the problem into an SDP by setting $\mathbf{Y}=\mathbf{V}^T\mathbf{V}$ would impose constraints on the rank of the matrix $\mathbf{Y}$ (specifically, $\operatorname{rank}(\mathbf{Y})\leq d$). Since we know that $\operatorname{rank}(\mathbf{Y})\leq N$ by construction, then having $N\leq d$ makes this rank constraint trivial. If we had these rank constraints, then the problem would be non-convex, given that the domain of the problem (a set of rank constrained matrices) is non-convex.

In practice, whether this constraint is satisfied is based on our choice for model dimensionality and batch size. It can however be circumvented, by regular optimization methods operating directly on the representations $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N$ (rather than their inner products). While the problem may no longer be convex, we can still apply first order optimization methods to find a good stationary point for the loss.

Proof of Theorem 2.1. The following is an adaptation of the proof found in Gärtner and Matousek [4].

We first show that the following problems:

$$\begin{array}{cc} \underset{\boldsymbol{u}_1,\ldots,\boldsymbol{u}_N,\boldsymbol{c}\in\mathbb{R}^d}{\text{minimize}} & k, \\ \text{s.t.} & \|\boldsymbol{u}_1\|^2=\|\boldsymbol{u}_2\|^2=\cdots=\|\boldsymbol{u}_N\|^2=\|\boldsymbol{c}\|^2=1, \\ & \boldsymbol{u}_i^T\boldsymbol{u}_j=0,\ \forall(i,j)\notin E, \\ & \frac{1}{(\boldsymbol{c}^T\boldsymbol{u}_i)^2}\leq k,\ \forall i, \end{array} \tag{10}$$

and:

$$\begin{array}{cc} \underset{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N\in\mathbb{R}^d,\ k\in\mathbb{R}}{\text{minimize}} & k, \\ \text{s.t.} & \|\boldsymbol{v}_1\|^2=\|\boldsymbol{v}_2\|^2=\cdots=\|\boldsymbol{v}_N\|^2=1, \\ & \boldsymbol{v}_i^T\boldsymbol{v}_j\leq -\frac{1}{k-1},\ \forall(i,j)\notin E, \end{array} \tag{11}$$

are equivalent. To do this, we shall show that an optimal solution of the first problem can be converted to a feasible solution for the second one, and vice versa.

Given an optimal solution to the first problem $\boldsymbol{u}_1,\ldots,\boldsymbol{u}_N,\boldsymbol{c}$ with optimal value k, we can define the following set of vectors:

$$\boldsymbol{z}_i=\frac{1}{\sqrt{k-1}}\left(\frac{\boldsymbol{u}_i}{\boldsymbol{c}^T\boldsymbol{u}_i}-\boldsymbol{c}\right). \tag{12}$$

For these vectors, the following hold:

- $\boldsymbol{z}_i^T \boldsymbol{z}_j = \frac{1}{k-1}\left(\frac{\boldsymbol{u}_i^T \boldsymbol{u}_j}{(\boldsymbol{c}^T \boldsymbol{u}_i)(\boldsymbol{c}^T \boldsymbol{u}_j)} - 1 - 1 + 1\right) \leq -\frac{1}{k-1}$.
- $\|\boldsymbol{z}_i\|^2 = \frac{1}{k-1}\left(\frac{\|\boldsymbol{u}_i\|^2}{(\boldsymbol{c}^T \boldsymbol{u}_i)^2} - 1 - 1 + 1\right) \leq \frac{1}{k-1}(k-1) = 1$.

Let us now define the matrix $\mathbf{Z}$, with columns the vectors $\boldsymbol{z}_i$. Thus, $\mathbf{Z}^T\mathbf{Z}$ is positive semidefinite (by construction) and has diagonal elements which are less than or equal to 1. Thus, we can define the matrix $\mathbf{Y} = \mathbf{Z}^T\mathbf{Z} + \mathbf{D}$, where $\mathbf{D}$ is a diagonal matrix with non-negative elements, such that $y_{ii} = 1$. This matrix $\mathbf{Y}$ is also PSD, which means that we can write it as $\mathbf{Y} = \mathbf{V}^T\mathbf{V}$. The columns of $\mathbf{V}$ are precisely the vectors $\boldsymbol{v}_i$ that form a feasible solution of the second problem, since their inner products (off-diagonal elements of $\mathbf{Y}$) satisfy the required constraints, and their norms (diagonal elements of $\mathbf{Y}$) are equal to 1.

Now, given an optimal solution to the second problem, $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_N$ with optimal value k, we can again define $\mathbf{Y} = \mathbf{V}^T\mathbf{V}$. Here we can make an argument that $\mathbf{Y}$ must have at least one eigenvalue equal to 0 (in other words, the vectors $\boldsymbol{v}_i$ must be linearly dependent). Indeed, if this was not the case, we would have $\mathbf{Y} \succ 0$, so $\mathbf{Y} - \epsilon\mathbf{I} \succeq 0$, where $\epsilon > 0$ sufficiently small. This means that the matrix $\mathbf{Y}' = \frac{1}{1-\epsilon}(\mathbf{Y} - \epsilon\mathbf{I})$ is a feasible solution to the second problem (note that the off diagonal elements of $\mathbf{Y}$ are negative, so multiplying them by a positive constant decreases them) which also has lower value for the objective function, which is a contradiction. Thus $\mathbf{Y}$ is singular, which also implies that there exists a unit vector $\boldsymbol{c} \in \mathbb{R}^d$ that has $\boldsymbol{c}^T\boldsymbol{v}_i = 0$, for all i. We can then define the following vectors:

$$\boldsymbol{u}_i = \frac{1}{\sqrt{k}}(\boldsymbol{c} + \boldsymbol{v}_i\sqrt{k-1}). \tag{13}$$

For these vectors, we have:

- $\boldsymbol{u}_i^T\boldsymbol{u}_j = \frac{1}{k}(1 + 0 + 0 + (k-1)\boldsymbol{v}_i^T\boldsymbol{v}_j) \leq 0$.
- $\|\boldsymbol{u}_i\|^2 = \frac{1}{k}(1 + 0 + 0 + k - 1) = 1$.
- $\boldsymbol{c}^T\boldsymbol{u}_i = \frac{1}{\sqrt{k}} \Rightarrow \frac{1}{(\boldsymbol{c}^T\boldsymbol{u}_i)} = k$.

This means that the vectors $\boldsymbol{u}_i$ along with $\boldsymbol{c}$ form a feasible solution for the first problem, with objective value k.

For the final step, we note that by setting $t = -\frac{1}{k-1}$, then by minimizing k we also minimize t (since the latter is an increasing function of k). This means that our second problem is equivalent to:

$$\begin{array}{cc} \underset{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N \in \mathbb{R}^d, t \in \mathbb{R}}{\text{minimize}} & t, \\ \text{s.t.} & \|\boldsymbol{v}_1\|^2 = \|\boldsymbol{v}_2\|^2 = \cdots = \|\boldsymbol{v}_N\|^2 = 1, \\ & \boldsymbol{v}_i^T\boldsymbol{v}_j \leq t,\ \forall (i,j) \notin E. \end{array} \tag{14}$$

This completes our proof.

Theorem B.1. *Formal version of Theorem 2.2 The problems:*

$$\begin{array}{cc} \text{minimize} & t, \\ \text{s.t.} & \|\boldsymbol{v}_i\|^2 = 1,\ \forall i, \\ & \boldsymbol{v}_i^T\boldsymbol{v}_j \leq t,\ \forall i \neq j, \end{array} \tag{15}$$

and:

$$\begin{array}{cc} \text{minimize} & \tau \sum_{i=1}^n \log \sum_{j \neq i} \exp\left(\frac{\boldsymbol{v}_i^T\boldsymbol{v}_j}{\tau}\right), \\ \text{s.t.} & \|\boldsymbol{v}_i\|^2 = 1,\ \forall i, \end{array} \tag{16}$$

attain their minima at the same values of the matrix $\mathbf{Y} = \mathbf{V}^T\mathbf{V}$, where $\mathbf{V}$ is the matrix which has the vectors $\boldsymbol{v}_i$ as columns.

Proof. Note that by setting $\mathbf{Y} = \mathbf{V}^T\mathbf{V}$, the first problem is converted into:

$$\begin{array}{cc} \text{minimize} & t, \\ \text{s.t.} & y_{ii} = 1,\ \forall i, \\ & y_{ij} \leq t,\ \forall i \neq j, \\ & \mathbf{Y} \succeq 0, \end{array} \tag{17}$$

while the second problem is converted into:

$$\begin{array}{cc} \text{minimize} & \tau \sum_{i=1}^{n} \log \sum_{j \neq i} \exp\left(\frac{y_{ij}}{\tau}\right), \\ \text{s.t.} & y_{ii} = 1, \ \forall i, \\ & \mathbf{Y} \succeq 0. \end{array} \tag{18}$$

Let us assume that we have a feasible solution $\mathbf{Y}$ for the second problem. Let $t = \frac{1}{n(n-1)} \sum_{i,j:i \neq j} y_{ij}$ (the average of the non-diagonal elements of $\mathbf{Y}$). Due to the convexity of the exponential function and the logarithm being an increasing function, we can apply Jensen's inequality in each of the sums inside the logarithms for each given i, thus giving us:

$$\begin{aligned} \tau \sum_{i=1}^{n} \log \sum_{j \neq i} \exp\left(\frac{y_{ij}}{\tau}\right) &\geq \tau \sum_{i=1}^{n} \log \left((n-1) \exp\left(\frac{\frac{1}{n-1} \sum_{j \neq i} y_{ij}}{\tau} \right) \right) \\ &= \tau n \log(n-1) + \frac{1}{n-1} \sum_{i=1}^{n} \sum_{j \neq i} y_{ij} \\ &= \tau n \log(n-1) + nt. \end{aligned} \tag{19}$$

The final step is precisely the value of our objective function, when all y_{ij} , $i \neq j$ are equal to t. This means that, if we replace our solution with the average of the non-diagonal elements, then we will always decrease the value of the objective. Note that the new solution will still be feasible. Indeed, since the original matrix $\mathbf{Y}$ is PSD, we have:

$$\mathbf{1}^T \mathbf{Y} \mathbf{1} = n + n(n-1)t \geq 0 \Rightarrow t \geq -\frac{1}{n-1}. \tag{20}$$

The new matrix we will use is:

$$\mathbf{Y}' = (1-t)\mathbf{I} + t\mathbf{1}, \tag{21}$$

(so the diagonal elements are 1, and the non-diagonal ones are t). This is a matrix with rank-1 difference from the identity, and it has $n-1$ eigenvalues equal to $1-t \geq 0$ (since t is the average of inner products of vectors with unit norm), and one eigenvalue equal to $1-t+tn = 1+(n-1)t \geq 0$ (due to the above). Thus the new matrix is also a feasible solution, so it is always optimal to have $y_{ij} = t$.

Thus, we can rewrite our problem as follows:

$$\begin{array}{cc} \text{minimize} & \tau \sum_{i=1}^{n} \log \sum_{j \neq i} \exp\left(\frac{y_{ij}}{\tau}\right), \\ \text{s.t.} & y_{ii} = 1, \ \forall i, \\ & y_{ij} = t, \ i \neq j, \\ & \mathbf{Y} \succeq 0. \end{array} \tag{22}$$

Given the condition for y_{ij}, we can rewrite our objective function as:

$$\tau \sum_{i=1}^{n} \log \left(n(n-1) \exp\left(\frac{t}{\tau}\right) \right) = \tau n \log\left(n(n-1)\right) + tn. \tag{23}$$

Thus, we can easily see that this problem has the same minimizing matrix $\mathbf{Y}$ as:

$$\begin{array}{cc} \text{minimize} & t, \\ \text{s.t.} & y_{ii} = 1, \ \forall i, \\ & y_{ij} = t, \ i \neq j, \\ & \mathbf{Y} \succeq 0. \end{array} \tag{24}$$

Finally, we need to argue that this problem has the same optimal solution as (17) (or that, in other words, having the constraints $y_{ij} = t$ be $y_{ij} \leq t$ does not change the optimal solution). This can be easily shown using the exact same argument as above - if we assume that there exists an element $y_{ij} < t$ in the optimal solution, then we can replace the non-diagonal elements of $\mathbf{Y}$ with their average $\bar{y} < t$, giving us a feasible solution $\mathbf{Y}' \succeq 0, t' = \bar{y} < t$, which is impossible. Thus, the problem in (17) has the same minimizer matrix $\mathbf{Y}$ as that in (24). This completes the proof, as having the same matrix $\mathbf{Y} = \mathbf{V}^T \mathbf{V}$ and having the vectors all be equal in norm means that the vectors chosen are unique up to rotations. □

Lemma B.1. *The following holds:*

$$\lim_{\tau \to 0^+} \tau \log \sum_{i=1}^{n} \exp \frac{z_i}{\tau} = \max_{i=1}^{n} z_i. \tag{25}$$

Proof. We have:

$$\sum_{i=1}^{n} \exp \frac{z_i}{\tau} \geq \exp\left(\frac{1}{\tau} \max_{i=1}^{n} z_i\right) \Rightarrow \max_{i=1}^{n} z_i \leq \tau \log \sum_{i=1}^{n} \exp \frac{z_i}{\tau}, \tag{26}$$

as well as:

$$\tau \log \sum_{i=1}^{n} \exp \frac{z_i}{\tau} \leq \tau \log \left(n \exp\left(\frac{1}{\tau} \max_{i=1}^{n} z_i\right)\right) = \tau \log n + \max_{i=1}^{n} z_i. \tag{27}$$

Thus, we get:

$$\max_{i=1}^{n} z_i \leq \tau \log \sum_{i=1}^{n} \exp \frac{z_i}{\tau} \leq \tau \log n + \max_{i=1}^{n} z_i, \tag{28}$$

and taking the limit as $\tau \to 0$ gives us the desired result.

We note here that the convergence is **uniform**: for a given τ, the difference between $\tau \log \sum_{i=1}^{n} \exp \frac{z_i}{\tau}$ and $\max_{i=1}^{n} z_i$ is upper bounded by $\tau \log n$, which is independent of $z_1, \ldots, z_n$. □

Lemma B.2 (Formal Version of Lemma 2.1 in the paper)**.** *For $N \leq d$, the weighted Lovasz Theta problem can be rewritten as in* (30)*. In other words, the following formulations of the weighted Lovasz theta problem are equivalent:*

$$\begin{array}{cc} \underset{\boldsymbol{u}_1,\ldots,\boldsymbol{u}_N,\boldsymbol{c}\in\mathbb{R}^d}{\text{minimize}} & \max_{i=1,\ldots,N} \frac{1}{(\boldsymbol{c}^T\boldsymbol{u}_i)^2}, \\ \text{s.t.} & \|\boldsymbol{u}_1\|^2 = \|\boldsymbol{u}_2\|^2 = \cdots = \|\boldsymbol{u}_N\|^2 = \|\boldsymbol{c}\|^2 = 1, \\ & \boldsymbol{u}_i^T\boldsymbol{u}_j \leq w_{ij}, \end{array} \tag{29}$$

and:

$$\begin{array}{cc} \underset{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N\in\mathbb{R}^d,t\in\mathbb{R}}{\text{minimize}} & t, \\ \text{s.t.} & \|\boldsymbol{v}_1\|^2 = \|\boldsymbol{v}_2\|^2 = \cdots = \|\boldsymbol{v}_N\|^2 = 1, \\ & \boldsymbol{v}_i^T\boldsymbol{v}_j \leq w_{ij} + (1 - w_{ij})t. \end{array} \tag{30}$$

Proof. To go from the first problem to the second, we begin by reformulating it as follows:

$$\begin{array}{cc} \underset{\boldsymbol{u}_1,\ldots,\boldsymbol{u}_N,\boldsymbol{c}\in\mathbb{R}^d,\, k\in\mathbb{R}}{\text{minimize}} & k, \\ \text{s.t.} & \|\boldsymbol{u}_1\|^2 = \|\boldsymbol{u}_2\|^2 = \cdots = \|\boldsymbol{u}_N\|^2 = \|\boldsymbol{c}\|^2 = 1, \\ & \boldsymbol{u}_i^T\boldsymbol{u}_j \leq w_{ij}, \\ & (\boldsymbol{c}^T\boldsymbol{u}_i)^2 \geq \frac{1}{k}. \end{array} \tag{31}$$

We can also reformulate the second problem as follows, by setting $t = -\frac{1}{k-1}$ and noticing that t is increasing as k increases:

$$\begin{array}{cc} \underset{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N\in\mathbb{R}^d,\, k\in\mathbb{R}}{\text{minimize}} & k, \\ \text{s.t.} & \|\boldsymbol{v}_1\|^2 = \|\boldsymbol{v}_2\|^2 = \cdots = \|\boldsymbol{v}_N\|^2 = 1, \\ & \boldsymbol{v}_i^T\boldsymbol{v}_j \leq \frac{w_{ij}k-1}{k-1}. \end{array} \tag{32}$$

It is sufficient to show that the problems in (31) and (32) are equivalent. Following the technique used by Gärtner and Matousek [4] the regular Lovasz theta problem, we do so by showing the two problems have the same optimal value. Let p_1 and p_2 be the optimal values of the problems in (31) and (32), respectively.

To show that $p_1 \geq p_2$, consider an optimal solution $\boldsymbol{u}_1^*, \ldots, \boldsymbol{u}_N^*, \boldsymbol{c}^*$. Let us formulate the following matrix $\mathbf{Y}$, with elements:

$$\begin{aligned} y_{ii} &= 1. \\ y_{ij} &= \frac{1}{p_1 - 1}\left(\frac{\boldsymbol{u}_i^*}{\boldsymbol{c}^{*T}\boldsymbol{u}_i^*} - \boldsymbol{c}^*\right)^T\left(\frac{\boldsymbol{u}_j^*}{\boldsymbol{c}^{*T}\boldsymbol{u}_j^*} - \boldsymbol{c}^*\right)^T = \frac{1}{p_1 - 1}\left(\frac{\boldsymbol{u}_i^{*T}\boldsymbol{u}_j^*}{(\boldsymbol{c}^{*T}\boldsymbol{u}_i^*)(\boldsymbol{c}^{*T}\boldsymbol{u}_j^*)} - 1\right) \\ &\leq \frac{p_1 w_{ij} - 1}{p_1 - 1}. \end{aligned} \tag{33}$$

We can also show that this matrix is PSD; since $y_{ii} \geq y_{ij}, \forall j$, $\mathbf{Y}$ can be written as $\mathbf{Y} = \mathbf{D} + \mathbf{U}^T\mathbf{U} \succeq 0$, where $\mathbf{D}$ is a diagonal matrix with non-negative entries and $\mathbf{U}$ is a matrix with columns equal to $\frac{\boldsymbol{u}_i^*}{\boldsymbol{c}^{*T}\boldsymbol{u}_i^*} - \boldsymbol{c}^*$. Consequently, we can use this $\mathbf{Y}$, to create a feasible solution $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_N$ via Cholesky factorization (i.e. $\mathbf{Y} = VV^T$, whenever $N \leq d$ (the constraints arise from the constraints placed on the matrix $\mathbf{Y}$). This feasible solution has objective value p_1 for the second problem. This means that the second problem has optimal value $p_2 \leq p_1$.

To show that $p_2 \geq p_1$, start from an optimal solution of the second problem $\boldsymbol{v}_1^*, \ldots, \boldsymbol{v}_N^*$. Let $\mathbf{Y}^*$ be the matrix with $y_{ij}^* = \boldsymbol{v}_i^{*T}\boldsymbol{v}_j^*$. We note here that this matrix must have at least one eigenvalue equal to 0. If this was not the case, then we would have $\lambda_{min}(\mathbf{Y}^*) > 0$, and we could construct the following matrix:

$$\mathbf{Y}' = \mathbf{Y}^* + \epsilon(\mathbf{I} - \mathbf{1}). \tag{34}$$

where $\mathbf{1}$ is the rank-1 matrix with all of its elements equal to 1. Note that this would strictly decrease all the non-diagonal elements of $\mathbf{Y}^*$, while leaving the diagonal ones intact. Furthermore, $\mathbf{1}$ has $N - 1$ eigenvalues equal to 0, and one of them equal to N (as $\mathbf{1}\boldsymbol{u} = N\boldsymbol{u}$, where u is the all-ones vector). Thus, since $\mathbf{I} - \mathbf{1}$ is a rank-1 difference from the diagonal, $N - 1$ of its eigenvalues being equal to 1 and one of them being equal to $1 - N < 0$ (given the eigenvalues of $\mathbf{I}$). Thus, we would have, for ϵ small enough:

$$\lambda_{min}(\mathbf{Y}') \geq \lambda_{min}(\mathbf{Y}^*) + \epsilon(1 - N) \geq 0. \tag{35}$$

Thus, by Cholesky decomposition again, we would have a feasible solution to problem 32, with strictly smaller off-diagonal elements of y_{ij}, which is a contradiction. Thus, one of the eigenvalues of $\mathbf{Y}$ must be 0, which means that we can find a vector $\boldsymbol{c}$ which is orthogonal to all $\boldsymbol{v}_1^*, \ldots, \boldsymbol{v}_N^*$. We can then define the vectors:

$$\boldsymbol{u}_i = \frac{1}{\sqrt{p_2}}(\boldsymbol{v}_i^*\sqrt{p_2 - 1} + \boldsymbol{c}). \tag{36}$$

These vectors have:

$$\boldsymbol{u}_i^T\boldsymbol{u}_j = \frac{1}{p_2}((p_2 - 1)\boldsymbol{v}_i^{*T}\boldsymbol{v}_j^* + 1). \tag{37}$$

Thus $\|\boldsymbol{u}_i\|^2 = \frac{1}{p_2}(p_2 - 1 + 1) = 1$ and $\boldsymbol{u}_i^T\boldsymbol{u}_j \leq \frac{1}{p_2}(w_{ij}p_2 - 1 + 1) = w_{ij}$. This means that they form a feasible solution for our problem, with objective value p_2. Thus $p_1 \leq p_2$.

Combining all of the above gives us $p_1 = p_2$, making the above problems equivalent.

□

Theorem B.2 (Formal Version of Theorem 2.3)**.** *Consider the following formulation of the weighted Lovasz theta problem:*

$$\begin{array}{ll} \underset{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N \in \mathbb{R}^d,\ t \in \mathbb{R}}{\text{minimize}} & t, \\ \text{s.t.} & \|\boldsymbol{v}_1\|^2 = \|\boldsymbol{v}_2\|^2 = \cdots = \|\boldsymbol{v}_N\|^2 = 1, \\ & \boldsymbol{v}_i^T\boldsymbol{v}_j \leq w_{ij} + (1 - w_{ij})t, \end{array} \tag{38}$$

as well as the minimization of the second term of the Lovasz theta contrastive loss:

$$\begin{array}{ll} \underset{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N \in \mathbb{R}^d}{\text{minimize}} & \tau\sum_{i=1}^N \log\left(\sum_{j=1}^N \exp\left(\frac{\boldsymbol{v}_j^T\boldsymbol{v}_i - w_{ij}}{\tau(1-w_{ij})}\right)\right), \\ \text{s.t.} & \|\boldsymbol{v}_1\|^2 = \|\boldsymbol{v}_1\|^2 = \|\boldsymbol{v}_2\|^2 = \cdots = \|\boldsymbol{v}_N\|^2 = 1. \end{array} \tag{39}$$

Minimizing the limit of the objective of the second problem as $\tau \to 0$ is a relaxation of the first problem, if $w_{ij} < 1$.

Proof. Using the property of Log-Sum-Exp to converge uniformly to the maximum of its arguments as τ goes to 0, in this limit the objective of the second problem becomes:

$$\begin{array}{cc} \underset{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N\in\mathbb{R}^d}{\text{minimize}} & \sum_{i=1}^N \max_{j\neq i} \frac{\boldsymbol{v}_j^T\boldsymbol{v}_i - w_{ij}}{1-w_{ij}}, \\ \text{s.t.} & \|\boldsymbol{v}_1\|^2 = \|\boldsymbol{v}_1\|^2 = \|\boldsymbol{v}_2\|^2 = \cdots = \|\boldsymbol{v}_N\|^2 = 1, \end{array} \tag{40}$$

(note that we don't argue about the behavior of the minimizing solution of the problem as $\tau \to 0$, but rather just the minimization of the limit of the objective function). We now include auxiliary variables t_i, changing the problem into the following:

$$\begin{array}{cc} \underset{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N\in\mathbb{R}^d,\ t_1,\ldots t_N\in\mathbb{R}}{\text{minimize}} & \sum_{i=1}^N t_i, \\ \text{s.t.} & \|\boldsymbol{v}_1\|^2 = \|\boldsymbol{v}_2\|^2 = \cdots = \|\boldsymbol{v}_N\|^2 = 1, \\ & \boldsymbol{v}_i^T\boldsymbol{v}_j \leq w_{ij} + (1-w_{ij})t_i,\ \forall i \neq j. \end{array} \tag{41}$$

This is a relaxation of the following problem:

$$\begin{array}{cc} \underset{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N\in\mathbb{R}^d,\ t_1,\ldots t_N\in\mathbb{R}}{\text{minimize}} & \sum_{i=1}^N t_i, \\ \text{s.t.} & \|\boldsymbol{v}_1\|^2 = \|\boldsymbol{v}_2\|^2 = \cdots = \|\boldsymbol{v}_N\|^2 = 1, \\ & \boldsymbol{v}_i^T\boldsymbol{v}_j \leq w_{ij} + (1-w_{ij})t_i,\ \forall i \neq j, \\ & t_i = t,\ \forall i, \end{array} \tag{42}$$

which is equivalent to precisely the weighted Lovasz theta problem:

$$\begin{array}{cc} \underset{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_N\in\mathbb{R}^d,\ t\in\mathbb{R}}{\text{minimize}} & t, \\ \text{s.t.} & \|\boldsymbol{v}_1\|^2 = \|\boldsymbol{v}_2\|^2 = \cdots = \|\boldsymbol{v}_N\|^2 = 1, \\ & \boldsymbol{v}_i^T\boldsymbol{v}_j \leq w_{ij} + (1-w_{ij})t,\ \forall i \neq j. \end{array} \tag{43}$$

In this case, the lack of symmetry of the problem does not allow us to immediately argue that all the t_i must be equal. □

C Discussion on the similarity matrix

Recall that our loss function is:

$$L_{LovaszCon} = -\sum_{i=1}^N \frac{1}{|P(i)|} \sum_{p\in P(i)} \boldsymbol{v}_p^T\boldsymbol{v}_i + \tau \sum_{i=1}^N \log\left(\sum_{j\neq i} \exp\left(\frac{\boldsymbol{v}_j^T\boldsymbol{v}_i - w_{ij}}{\tau(1-w_{ij})}\right)\right). \tag{44}$$

We can note the following about our loss, regarding the term $\frac{\boldsymbol{v}_j^T\boldsymbol{v}_i - w_{ij}}{\tau(1-w_{ij})}$ in the exponent:

- As w_{ij} approaches 0 (or in other words, as the samples become less and less correlated), the fraction within the exponents in the above expression becomes closer to the one present in the regular supervised contrastive loss.
- As w_{ij} approaches 1 (so the two samples become more and more similar), if $\boldsymbol{v}_i \neq \boldsymbol{v}_j$ (which is satisfied assuming that the representations are not degenerate), the denominator goes to 0, but the numerator is also becomes negative, making the term after the exponentiation negligibly small. In this case, the constraint in (7) becomes trivial (since the vectors have unit norm) and is thus discarded.

The two points above demonstrate that our technique is a natural extension of contrastive learning. Indeed, the latter is a special case of the former, when the similarities are considered to be the identity matrix. We can also graphically see the effect of w_{ij} in each of the terms of the Log-Sum-Exp function, in Figure 2. We can see that w_{ij} varies the slope of the term inside the exponent. The higher w_{ij} is, the less this term is penalized if the inner product is large (so the negative terms are penalized less if the samples are more similar).

One major design choice in the above formulation is that of the similarity graph that we use. We identify two major approaches that can be used to derive the similarity matrix.

- **Supervised case.** If we assume that we have access to the labels of the samples during contrastive training, then we can derive the similarities between samples via the similarities between their respective classes. A simple way to do this is the following:
  1. We obtain a confusion matrix $\mathbf{C} = [c_{ij}]_{i,j=1,\ldots,N_{classes}}$ from a pretrained classifier.
  2. We normalize the confusion matrix across its rows (so that they all sum to 1).
  3. We set $\mathbf{C}' = [c'_{kl}]_{k,l=1,\ldots,N_{classes}}$, where:

$$c'_{kl} = \begin{cases} \frac{1}{2}(c_{kl} + c_{lk}) & k \neq l \\ 1 & k = l \end{cases} \tag{45}$$

  This matrix $\mathbf{C}'$ is our class similarity matrix, and given two samples x_i and x_j, with corresponding classes y_i and y_j, we can define their similarity as $w_{ij} = c'_{y_i y_j}$.

  Moreover, we can make use of domain knowledge for the problem, and derive our similarity matrix in an alternative fashion. If we have a hierarchical structure for the classes of the problem, we can derive similarity by considering classes that belong in the same hierarchical superclass as being similar to each other. We examine both of the above choices in the experimental section.
- **Unsupervised case.** If we assume that we don't have access to the labels of our samples during the contrastive training process (or that only part of them is labeled), then we could also leverage a pre-existing model to derive sample similarities directly. For example, we could use a pretrained image encoder such as CLIP [7], in order to derive a unit norm embedding v_i for each sample x_i, and define the similarity between the two samples as $w_{ij} = v_i^T v_j$.

D Training Details

In our CIFAR experiments, we made use of an A100 GPU to train our models. In the supervised case, our models were trained for 300 epochs, with a batch size of 512, and the same set of hyperparameters as those used in the Supervised Contrastive learning baseline [8]. The architecture used in each of the experiments was a ResNet of varying depth, and the projection head used to perform contrastive learning was a two-layer MLP, reducing the dimension of the features to 128. Evaluation was performed by training a linear classifier on top of the model for 10 epochs, and reporting the best accuracy obtained on the test set across all of the epochs. In the unsupervised case, the architecture is the same, but our models where trained using a batch size of 1024, a learning rate and temperature τ equal to 0.5, and a linear probe trained for 100 epochs over the learned representations (as done in the repository for the code of Khosla et al. [8])

For our ImageNet-100 experiments, we made use of computing nodes with 4 RTX5000 GPUs. For models trained with cross-entropy loss, we employed a batch size of 256. Moreover, we made use of Momentum Contrast, when training these models. During MoCo training, we maintained a batch size of 256 and a memory bank of size 8192 as in Khosla et al. [8]. We trained each model for 200 epochs using the standard hyperparameter choices found in He et al. [6].

Our baseline models were trained using the code directly provided by Khosla et al. [8]. The confusion matrix based similarities were obtained based on the predictions of a model trained with Supervised Contrastive Learning.

The overall code is provided as part of the Supplementary material for review purposes, and will be made publicly available upon acceptance. The code is based upon the publicly available code of Supervised Contrastive Learning and Momentum Contrast, and all the relevant licenses are included.

E Further Experiments

E.1 Ablation on Similarity Matrix

As noted previously, a major design choice in our experiments is the method used to derive the similarity matrix, access to which is assumed by our method. Here, we test this choice on CIFAR100, in the supervised setting. Namely, we consider the following two methods to derive these similarities discussed above: using the superclass similarities provided for CIFAR100 and using the inner products of the CLIP text representations of the labels.

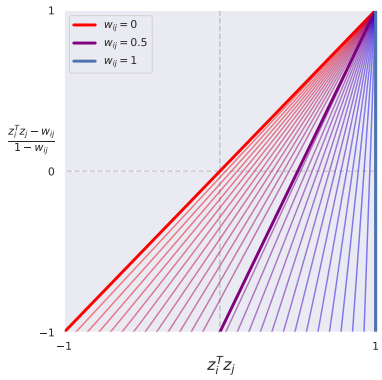


Figure 2: **Illustration of the effect of similarity terms w_{ij} in the Lovasz contrastive loss.** The x-axis denotes the dot product between two features, and the y-axis represents the dot-product after the similarity terms are applied. For completely dissimilar samples, $w_{ij} = 0$ and this reduces to the contrastive supervised loss. For partially similar samples, the contribution to the negative-sample term of the contrastive loss is decreased. As the similarity increases to 1, this inner product vanishes from the Log-Sum-Exp of the uniformity term of the contrastive loss (the second term).

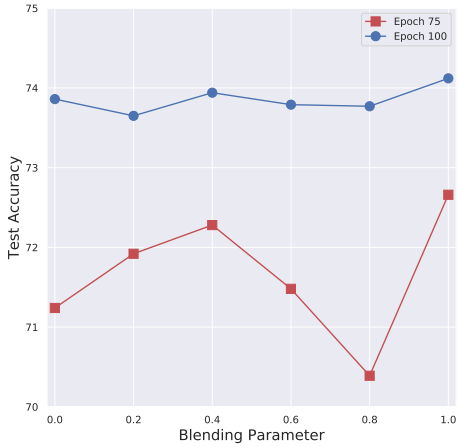


Figure 3: **Evolution of loss based on blending of similarity matrix.** Both plots are based on CIFAR100, with models pretrained for 75 and 100 epochs respectively. The blending parameter varies from 0 (where the similarity matrix is the identity, which is the same as SupCon) to 1 (where the similarity matrix is equal to the confusion matrix). We see that the accuracy of our model has an increasing trend, the closer to using the confusion matrix we get. The fact that there is also a downward spike in the model hints at the usefulness of tuning this blending parameter.

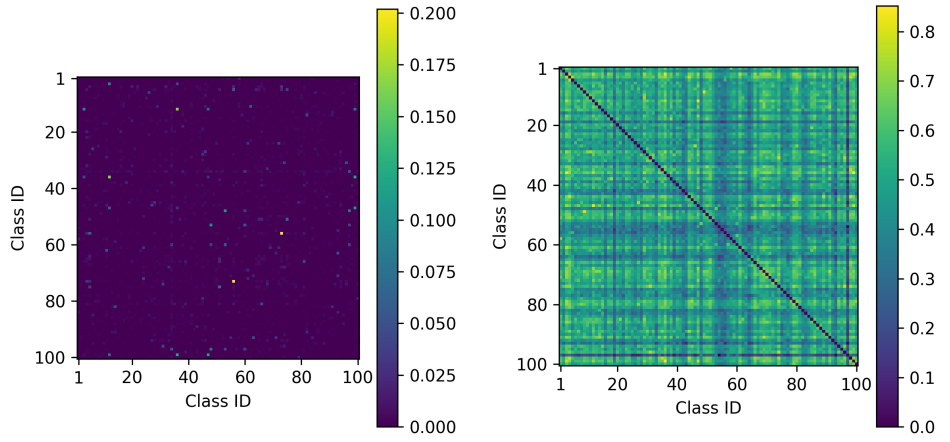
Our results can be seen in Table 4. We can see that the superclass similarity approach achieves better accuracy in all 3 of the considered models. This shows that having access to domain knowledge for the problem is beneficial for our method. However, we should note here that retrieving the similarities through CLIP requires access to only the labels of the problem, and is thus easier to retrieve in cases where further domain knowledge is not available.

E.2 Path from Contrastive Learning to Lovasz Theta Contrastive Learning

A final experiment that we conduct is varying the similarity matrix between the confusion matrix and the identity. For the supervised setting, if $\mathbf{C}$ is our class similarity matrix, we set $\mathbf{C}' = \lambda\mathbf{C} + (1-\lambda)\mathbf{I}$,

Table 4: **Ablation on the choice of the similarity matrix.** We ablate on how we can choose the similarity matrix of our method. Our two proposed choices are retrieving the similarities via the superclasses of the problem, and the other is to examine the inner products of the vectors produced by the text representations of CLIP. We can see that the superclass similarity approach is better, but the CLIP approach gets adequate results, without requiring explicit domain information other than the text labels.

	Superclass Similarity	CLIP
ResNet-50	77.60	76.25
ResNet-34	76.55	75.19
ResNet-18	74.91	73.56



(a) Confusion matrix similarities.

(b) CLIP based similarities.

Figure 4: **Similarity matrices used in the main paper, supervised setting.** On the left, we see the confusion matrix similarities, and on the right the CLIP based similarities. In both cases, the diagonal has been set to 0 for the purposes of visualization. We can see that while the confusion matrix approach is much more selective in which classes are similar to each other, the CLIP based similarities assign much greater similarities to all classes. This, combined with the results of the relevant experiments, shows the importance of the chosen similarity matrix.

as our new similarity matrix, where $\lambda \in [0, 1]$ is a blending parameter. This has the goal of allowing us to visualize the way our performance changes from $\lambda = 0$ (regular SupCon) to $\lambda = 1$ (our method). The results can be seen in Figure 3. We can see that there is a subtle, but overall increasing trend in performance as the similarity matrix moves closer to the confusion matrix that we used, rather than the identity. However, we can also see that the accuracy varies overall, as this parameter is changed. As such, we can infer that this blending hyperparameter can be tuned to obtain better results.

E.3 Confusion Matrix Visualization

In this section, we include a visualization of two similarity matrices used in the main paper. We can see the results in Figure 4. It is evident that the confusion matrix based approach is much more selective than the CLIP based one. Combining this with the fact that both the confusion matrix and the superclass similarities outperforms the CLIP based one, as seen in the experimental section, we can infer the result that the more selective the confusion matrix is, the better the results of our method are (which is also intuitively what we expect to happen).

Table 5: **Summary of our results on CIFAR10.** In this case, our method is competitive with SupCon, but there is a limitation due to the fewer number of classes. The numbers with * are taken from [8].

	CE	SupCon	Ours
ResNet-50	95*	**96***	95.47
ResNet-34	95.15	**95.28**	95.07
ResNet-18	94.37	94.61	**94.69**

Table 6: **Ablation on superclass similarity.** Our method has comparable results, based on how similar samples from the same superclass but different classes are considered to be (0.5 or 0.8 in this case).

	Superclass Similarity 0.5	Superclass Similarity 0.8
ResNet-50	77.60 ± 0.30	77.68 ± 0.49
ResNet-34	76.55 ± 0.40	76.35 ± 0.06
ResNet-18	74.91 ± 0.15	74.99 ± 0.32

E.4 Experiment on CIFAR10

In Table 5, we can see the results of our method on CIFAR10, using the similarity matrix derived via the confusion matrix of another model. We can see that while our method obtains good accuracy, we cannot get significant improvements over regular supervised contrastive learning. We believe that this is a sensible limitation for our method - due to the very small number of classes, it is highly unlikely that any are that similar to begin with. As such, the main benefit of our method, which is being able to leverage sample similarities via their classes during training does not apply. Nevertheless, since many important tasks contain a much larger number of classes, this is only a small limitation.

E.5 Ablation on the Superclass Similarities

As an ablation, we performed the experiments presented in the main paper regarding the superclass similarities on CIFAR, using a different value for the similarity of classes belonging to the same superclass. The results can be seen in Table 6. We can see that we get comparable results with those seen in the main paper, so tuning this hyperparameter may prove useful, in order to improve the performance of the model.

F Broader Impact & Limitations

Our work considers a variant of contrastive learning which takes into account sample similarities when training the model in question. The process of contrastive learning is already well-known in the literature, and our model is a variant of this already widely-used idea. As such, we do not predict any negative societal impact stemming from our work.

While theoretically interesting, our work nevertheless has some limitations. The first of these is the fact that it is as of now unclear whether the techniques we used to obtain the similarity matrices are the best ones available. Indeed, it may be the case that there exist other ways to obtain a similarity matrix, which are better suited for our method. The second one is the fact that further experimentation is required, to fully understand the capabilities of our method.