We thank the reviewers for their comments, and address the main concerns below.

Rev 4: Please explicitly write the optimisation objectives in the mathematical programming.

Suppose we observe $S_m = \{(\boldsymbol{x}_{i_t}, \boldsymbol{x}_{j_t}, \boldsymbol{x}_{k_t}, \boldsymbol{x}_{l_t}) \mid t \in [m]\}$, where $D(\boldsymbol{x}_{i_t}, \boldsymbol{x}_{j_t}) \leq D(\boldsymbol{x}_{k_t}, \boldsymbol{x}_{l_t})$ for some unknown similarity function and (i_t, j_t, k_t, l_t) are indices of objects in a countable set $\mathcal{U}$ (e.g. set of people in a social network). To model the underlying similarity function D, we propose a Bregman divergence of the form:

$$\hat{D}(\boldsymbol{x}_i, \boldsymbol{x}_j) \triangleq \hat{\phi}(\boldsymbol{x}_i) - \hat{\phi}(\boldsymbol{x}_j) - \nabla_* \hat{\phi}(\boldsymbol{x}_j), \tag{1}$$

where $\hat{\phi}(\boldsymbol{x}) \triangleq \max_{i \in \mathcal{U}_m} \boldsymbol{a}_i^T(\boldsymbol{x} - \boldsymbol{x}_i) + z_i$, ∇_* is the biggest sub-gradient, $\mathcal{U}_m$ is the set of all observed objects indices $\mathcal{U}_m \triangleq \cup_{t=1}^m \{i_t, j_t, k_t, l_t\}$ and $\boldsymbol{a}_i$'s and z_i's are the solution to the following linear program:

$$\min_{z_i, \boldsymbol{a}_i, L} \sum_{t=1}^{m} \max(\zeta_t, 0) + \lambda L$$

$$\text{s.t.} \begin{cases} z_{i_t} - z_{j_t} - \boldsymbol{a}_{j_t}^T(\boldsymbol{x}_{i_t} - \boldsymbol{x}_{j_t}) + z_{l_t} - z_{k_t} + \boldsymbol{a}_{l_t}^T(\boldsymbol{x}_{k_t} - \boldsymbol{x}_{l_t}) \leq \zeta_t - 1 & t \in [m] \\ z_i - z_j \geq \boldsymbol{a}_j^T(\boldsymbol{x}_i - \boldsymbol{x}_j) & i, j \in \mathcal{U}_m \\ \|\boldsymbol{a}_i\|_1 \leq L & i \in \mathcal{U}_m \end{cases}$$

We will be sure to be explicit about the optimization objective in the final version of the paper.

Rev 4 and Rev 2: Notations, Bounds are meaningless for K=n!, Experiment setup is different from the theory

There are some notational typos which will be corrected. First, the δ in Theorem 1 is just some dummy variable denoting something small which we'll replace by another symbol. Second, n stands for number of unique people in all comparisons. m stands for number of comparisons, i.e: $n = \#\mathcal{U}_m$, so n will increase with m. Resolving these typos lets look at Theorem 2.

Theorem 2. *Consider $S_m = \{(\boldsymbol{x}_{i_t}, \boldsymbol{x}_{j_t}, \boldsymbol{x}_{k_t}, \boldsymbol{x}_{l_t}, y_t), t \in [m]\} \sim \mu^m$, where $D(\boldsymbol{x}_{i_t}, \boldsymbol{x}_{j_t}) \leq D(\boldsymbol{x}_{k_t}, \boldsymbol{x}_{l_t})$. Set $R = \max_i \|\boldsymbol{x}_i\|_\infty$. The generalization error of the learned divergence in (1) when using K hyper-planes satisfies*

$$\begin{aligned} \mathbb{E}\big[1[\hat{D}(\boldsymbol{x}_{i_t}, \boldsymbol{x}_{j_t}) \geq \hat{D}(\boldsymbol{x}_{k_t}, \boldsymbol{x}_{l_t})]\big] \quad &\leq \quad \frac{1}{m}\sum_{t=1}^{m} \max\left(0, 1 + \hat{D}(\boldsymbol{x}_{i_t}, \boldsymbol{x}_{j_t}) - \hat{D}(\boldsymbol{x}_{k_t}, \boldsymbol{x}_{l_t})\right) \\ &+ \quad 32KLR\sqrt{2\ln(2d+2)}/\sqrt{m} + \sqrt{4\ln(4\log_2 L) + \ln(1/\delta)}/\sqrt{m}, \end{aligned}$$

with probability at least $1 - \delta$ for receiving the data S_m.

Case 1: **(K<n)** We discuss details of the algorithm about the case where $K < n$ in appendix A6; this is another approach we have used in our experiments which yielded similar results. Using standard cross-validation to select K is the simplest and most effective way to select a value of K, and also ensures that the theoretical bounds are applicable. Also its almost obvious that doing further cross validation to choose K would result in improvement over the choice of $K = n$. However we liked to use $K = n$ in the reported experiments as it results in a faster algorithm and reduces the time needed for cross validation.

Case 2: **(K=n)** For the theoretical bound to hold we need $n << m$. This could be true in the example of the social network if we extract some kind of similarity information between people. Regardless of this we found acceptable results in our experiments with this setting.

i.i.d setup: The i.i.d setup is correct in how we trained the Bergman divergence in our experiments. We randomly choose the similarity comparisons from the fixed classification data-set $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$. This makes sense in a practical sense too as having or computing relative comparisons between all triplets would make $m = O(n^3)$ which is impractical when n is large. However we test the divergence in different tasks (i.e. ranking and clustering).

Rev 3: In theorem 2, the bound of generalization error is increasing with K, which seems to be strange. Is this correct?

Yes, this is due a to a bias variance trade-off which is more visible in the regression setting. Increasing K will increase the variance and worsens the generalisation gap, however it could lower the empirical risk. In experiments, we have both used cross-validation to choose K as well as using the simpler $K = n$.