Learning Inconsistent Preferences with Gaussian Processes

Siu Lun Chau
University of Oxford

Javier González
Microsoft Research Cambridge

Dino Sejdinovic
University of Oxford

Abstract

We revisit widely used *preferential Gaussian processes* (PGP) by Chu and Ghahramani [2005] and challenge their modelling assumption that imposes rankability of data items via latent utility function values. We propose a generalisation of PGP which can capture more expressive latent preferential structures in the data and thus be used to model inconsistent preferences, i.e. where transitivity is violated, or to discover clusters of comparable items via spectral decomposition of the learned preference functions. We also consider the properties of associated covariance kernel functions and its reproducing kernel Hilbert Space (RKHS), giving a simple construction that satisfies universality in the space of preference functions. Finally, we provide an extensive set of numerical experiments on simulated and real-world datasets showcasing the competitiveness of our proposed method with state-of-the-art. Our experimental findings support the conjecture that violations of rankability are ubiquitous in real-world preferential data.

1 Introduction

Data concerning user preferences for items or services is ubiquitous and is often used to detect patterns in user behaviour and to make recommendations. Moreover, these user preferences are often relative (i.e. based on recording choices between a pair of competing items) and may involve an abundance of ranking inconsistencies, e.g. preference of A over B, B over C, but C over A – sometimes called a *rock-paper-scissors relation*, and reported, e.g. in mating strategies of certain species [Sinervo and Lively, 1996]. Situation like this arises in many domains and is an example of the Condorcet Paradox extensively investigated in social choice theory [Gehrlein, 1983]. Such inconsistencies may arise due to latent structures determining the criteria for preferences, where different item features may be relevant for making each of these three choices. As an example, consider the case where a cue is present in an item description for C, which may be relevant for its comparison to A but not for its comparison to B, and that this cue changes the user's criterion when making the choice. Motivated by such inconsistent preferences, we will propose a Gaussian process (GP) model which can capture such latent structures by seamlessly incorporating all the available context information, i.e. sets of item covariates.

Our main contributions can be summarised as follows:

1. We propose a simple generalisation of PGP by Chu and Ghahramani [2005], allowing to model preferences that do not conform to a consistent ranking. Our method can be integrated directly into many existing probabilistic preference learning algorithms in fields such as rank aggregation [Simpson and Gurevych, 2020], Bayesian optimisation [González et al., 2017], duelling bandits [Zoghi et al., 2015], recommender systems [Nguyen et al., 2014] and reinforcement learning [Zintgraf et al., 2018].

2. The proposed *Generalised Preferential Gaussian Processes* (GPGP) use *Generalised Preferential Kernels* – we give a simple construction of these kernels which we prove to satisfy the appropriate notion of universality, i.e. the corresponding RKHS is rich enough to approximate any bounded continuous skew-symmetric function arbitrarily well. While a weaker form of this result has previously appeared in Waegeman et al. [2012], our proof uses different techniques, building on c_0-universality notions as developed by Sriperumbudur et al. [2011], allowing for more general domains like $\mathbb{R}^d$.

3. We extend ideas from partial ranking [Cheng et al., 2012] and propose a spectral decomposition

Proceedings of the 25[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151.

method to extract *clusters of comparable items* from preferential data using GPGP. This allows us to extract interpretable substructures from a complex network of preferential relationships.

The paper is outlined as follows: in section 2, we outline the problem and overview related work. In section 3, we introduce GPGP, describe universality of the corresponding kernel function and how GPGP can be used to uncover clusters of comparable items. Section 4 provides extensive experiments on synthetic and real-world data. Our results improve performance over PGP on all real-world datasets, giving further evidence for ubiquity of inconsistent preferences. We conclude in section 5.

2 Background

Assume we would like to choose a data item from domain $\mathcal{X}$. The well established paradigm in this context is *preference learning* (PL), which is concerned with predicting and modeling an order relation on a collection of data items [Fürnkranz and Hüllermeier, 2010]. Typical PL models [Chu and Ghahramani, 2005, d'Aspremont et al., 2021, González et al., 2017, Houlsby et al., 2012] assume that there is a latent *utility function* $f : \mathcal{X} \to \mathbb{R}$ to be optimised. We may observe noisy evaluations of f in forms such as item ratings or rankings, but in many cases, an explicit direct feedback from f is scarce or expensive and the quantity of implicit feedback data typically far outweighs the explicit data. Moreover, when the feedback comes from human users, they are better at evaluating relative differences than absolute quantities [Kahneman and Tversky, 1979], and in absence of a reference point explicit feedback may be unreliable and its scale may be ambiguous or difficult to determine. This motivates us to consider the situation where the feedback is *duelling*, i.e. consisting of *binary preferences*. Formally, a pair of items $(x, x') \in \mathcal{X} \times \mathcal{X}$ is presented to the user and we observe a binary outcome which tells us whether x or x' won the duel. For simplicity, we will assume here that no draws are allowed.

Binary preference data are often represented as Directed Aclycic Graphs (DAGs), where items are denoted as nodes and an edge from node $x \to x'$ implies that x won the duel over x' [Pahikkala et al., 2009]. As a result, preference learning can often be seen as learning on DAGs. For example, PageRank [Page et al., 1998] can be seen as an Eigenvector centrality measure on a preference graph. For the rest of the paper, we will use the term preference graph and preferential data interchangeably.

One simple model for the duelling feedback is given by

$$p(y|(x, x')) = \sigma(yg(x, x')), \quad y \in \{-1, +1\} \qquad (1)$$

for some $g : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, logistic function $\sigma(t) = \frac{1}{1+e^{-t}}$ and $y = +1$ denoting that x is preferred over x'. We note that g must be skew-symmetric, i.e. $g(x, x') = -g(x', x)$ to satisfy the natural condition $p(y|(x, x')) + p(y|(x', x)) = 1$ since there are only two outcomes allowed: a win for x or a win for x'. Considering general relations on pairs of items in $\mathcal{X} \times \mathcal{X}$, Pahikkala et al. [2010] term relations which satisfy skew-symmetry *reciprocal*.

An instance of model (1) is *Preferential Gaussian Process* (PGP) introduced by Chu and Ghahramani [2005]. It is assumed therein that g imposes *rankability* on $\mathcal{X}$. If we define $x \preceq x' \iff g(x, x') \leq 0$, then $\preceq$ is a total order on all of $\mathcal{X}$. This corresponds to writing $g(x, x') = f(x) - f(x')$, where f is the utility function which is determined up to a global shift. Pahikkala et al. [2010] consider a similar notion of a reciprocal relation and term it *weakly ranking representable* when such f exists. In the PGP model, a GP prior is imposed on latent f and the likelihood for a given observation $(x_i, x_j, y_{i,j})$ now becomes

$$p(y_{i,j}|(x_i, x_j)) = \sigma\left((f(x_i) - f(x_j))\, y_{i,j}\right). \qquad (2)$$

Inference on f can then proceed similarly as in GP classification, using methods such as Laplace approximation [Williams and Rasmussen, 2006, Section 3.4] or variational methods [Hensman et al., 2015].

A multitude of probabilistic PL algorithms are developed based on PGP. An extension of the model to predict crowd preferences is introduced by Simpson and Gurevych [2020], where a low-rank structure is imposed on the crowd preference matrix and each component is modelled using a GP. On the other hand, González et al. [2017] developed preferential Bayesian optimisation to optimise black-box functions where queries only come in the form of duels. Houlsby et al. [2012] incorporated PGP with unsupervised dimensionality reduction for multi-user recommendation systems. Under a similar setting, Nguyen et al. [2014] applied PGP into a GP factorisation machines to model context-aware recommendations. PGP is also used in the field of reinforcement learning to provide preference elicitation strategies for supporting multi-objective decision making [Zintgraf et al., 2018]. Finally, one can directly incorporate the learned preference function into learning to rank problems [Ailon and Mohri, 2010]. All models mentioned above assume the data to be perfectly rankable and this is the assumption we challenge in this paper.

Other preference learning models also typically assume data to be rankable and that a well defined utility function exists. Classical examples are *random utility model* [Thurstone, 1994], Bradley-Terry-Luce models [Bradley and Terry, 1952, Luce, 1959], the Thurstone-Mosteller model [Mosteller and Nogee, 1951] and many of their variants. Non-probabilistic preference models such as SVM-Rank [Joachims, 2009], Serial-Rank [Fogel et al., 2016], Sync-Rank [Cucuringu, 2016] and SVD-Rank [d'Aspremont et al., 2021] also typically assume rankability in their formulations.

In practice however, total rankability is often too strong of an assumption. There might be many reasons why some "noisy" preferences do not conform to a single overall ranking. For example, it is well studied that cognitive biases often lead to inconsistent human preferences in behavioral economics [Tversky and Kahneman, 1992]. In fact, not until very recently did the ranking community start to challenge this assumption by proposing quantitative metrics on measuring rankability of duelling data: Anderson et al. [2019], Cameron et al. [2020] considered rankability as a metric measuring the difference between the observed preference graph and a perfectly rankable complete dominance graph. This motivates the need to consider a general preference modelling methods without assuming total rankability.

To relax rankability assumptions and thus capture more complex latent structures in preferential data, we will consider a Gaussian process formulation for a general case where no single order can be formed and it is, in particular, possible that transitivity is violated, i.e. $x \preceq x', x' \preceq x''$ but $x'' \preceq x$. We believe that in many cases, such inconsistent relationships are fundamental to the data generating process. In fact, this conjecture is supported by the findings of Zoghi et al. [2015] who consider discrete choice (duelling bandits) problem with the application in ranker evaluation for information retrieval. They concluded that the instances where the Condorcet winner (an item which beats all the others with probability larger than $\frac{1}{2}$) does not exist far outweigh those where it does. Since the existence of a single objective function f with a unique global maximum would imply the existence of the Condorcet winner, we see that inconsistent preferences may, in fact, be prevalent in practice.

A thread of important related work arises in the inference of general (i.e. not necessarily preferential) relations between pairs of data objects [Pahikkala et al., 2010, Waegeman et al., 2012] using *frequentist kernel methods*. In particular, Pahikkala et al. [2010] similarly emphasise the importance of being able to model *intransitive* reciprocal relationships, motivating it using sports games examples. They also introduce the same kernel function we will consider in this work. Waegeman et al. [2012] take this work further, consider more general graded relations, reiterating importance of intransitivity, and study the connections to fuzzy set theory. Waegeman et al. [2012] also prove the theoretical result which is a slightly weaker form of our Theorem 1 on universality. As such, we emphasise that the generalised preferential kernels we will consider are not new, but to the best of our knowledge they have not been used in Gaussian process modelling, nor in discovering richer latent structure behind preferential data, which we propose in this work. There is also work that considers intransitive relations using different types of statistical models – without using item covariates and operating only on the matrix of match outcomes. For example, Causeur and Husson [2005] extend the classical Bradley-Terry model, while Chen and Joachims [2016] introduce so called Blade-Chest model and discover that substantial intransitivity exists in contexts such as online video gaming data.

We will in this paper deliberately adopt both Bayesian and frequentist viewpoints to kernel methods. We consider and implement a new Gaussian process framework, generalising PGP of Chu and Ghahramani [2005] which can hence be integrated in many probabilistic preference learning algorithms that build on PGP. But we also study the properties of the RKHSs associated to the corresponding kernel functions, arriving at conclusions essentially equivalent to those in Pahikkala et al. [2010], Waegeman et al. [2012], although we use different proof techniques which are more grounded in the notions of RKHS universality developed by Sriperumbudur et al. [2011], allowing us to consider more general spaces $\mathcal{X}$ of item covariates. We note that GPs and RKHSs have deep connections, as described in Kanagawa et al. [2018].

3 Methodology

3.1 Generalised preferential kernels

Recall that in PGP we express the preference function $g(x, x')$ as $f(x) - f(x')$ and place a GP prior on f. In fact, one can recast the inference solely in terms of g as f directly induces a GP prior on g by linearity. The corresponding covariance kernel k_E^0 is then given by

$$\begin{aligned} k_E^0((u, u'), (v, v')) &= \mathrm{cov}(f(u) - f(u'), f(v) - f(v')) \\ &= k(u, v) + k(u', v') \\ &\quad - k(u, v') - k(u', v), \end{aligned} \tag{3}$$

where the base kernel k is the covariance structure on f. Houlsby et al. [2012] called k_E^0 the *preference kernel*. This reformulation allows us to directly apply many state-of-the-art GP classification methods.

Now consider a more general case where $g : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ corresponds to any skew-symmetric function. We will consider the following skew-symmetric kernel:

$$k_E((u,u'),(v,v')) = k(u,v)k(u',v') - k(u,v')k(u',v), \quad (4)$$

termed *Generalised Preferential Kernel* and the corresponding GP will be called the *Generalised Preferential Gaussian Processes* (GPGP).

The kernel (4) is not new and was previously studied by Pahikkala et al. [2010], Waegeman et al. [2012] in their work on intransitive relations, as well as in persistent homology analysis to enforce appropriate symmetry conditions [Kwitt et al., 2015, Reininghaus et al., 2015]. In particular, Pahikkala et al. [2010] take a feature mapping ψ on $\mathcal{X} \times \mathcal{X}$ and "skew-symmetrise" it in the following way: $\varphi(x,x') = \psi(x,x') - \psi(x',x)$. Now φ and the corresponding kernel can be used to model skew-symmetric functions and, thus, reciprocal relations. In case where ψ corresponds to the Kronecker product kernel $k \otimes k$, this results exactly in (4). We give some further details of the feature map view of these kernels in the Appendix.

One can interpret both k_E^0 and k_E as kernels between edges in a preference graph. k_E can be extended further to tackle more complex preferential data settings such as *learning from crowd preferences* and *preference learning from distributional data*. We will keep the exposition here simple and a further description of these extensions is included in the Appendix.

For any kernel function κ, denote its RKHS by $\mathcal{H}_\kappa$. $\mathcal{H}_{k_E}$ is clearly more expressive than $\mathcal{H}_{k_E^0}$ as it imposes no rankability assumption on its elements. We next consider how expressive $\mathcal{H}_{k_E}$ is, given suitable regularity conditions on $\mathcal{X}$ and k. In particular, for *any* skew-symmetric bounded continuous function g on $\mathcal{X} \times \mathcal{X}$, can one find a function in $\mathcal{H}_{k_E}$ that arbitrarily well approximates g? We define a suitable notion of ss-c_0-universality below which allows for a very general domain $\mathcal{X}$. There are different notions of universality for kernels and we refer the reader to Micchelli et al. [2006], Sriperumbudur et al. [2011] and references therein for further details.

Definition 1 (ss-c_0-universality)**.** *Let $\mathcal{X}$ be a locally compact Hausdorff space and let $C_{0,ss}(\mathcal{X} \times \mathcal{X})$ be the space of functions $f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ which are continuous, bounded, skew-symmetric and vanish at infinity. A kernel k is said to be ss-c_0-universal on $\mathcal{X} \times \mathcal{X}$ if and only if $\mathcal{H}_k$ is dense in $C_{0,ss}(\mathcal{X} \times \mathcal{X})$ w.r.t. the uniform norm.*

We next prove a theorem which allows us to easily construct ss-c_0-universal kernels k_E by simply selecting k to be c_0-universal [Sriperumbudur et al., 2011]. We note that a weaker form of this result was first proved in [Waegeman et al., 2012, Theorem III.4] using different techniques. Our proof (included in Appendix) builds on the notion of c_0-universality and its relationship with integrally strictly positive definite kernels developed by Sriperumbudur et al. [2011], making the construction applicable to any *locally compact Hausdorff space* $\mathcal{X}$, whereas Waegeman et al. [2012] require *compact metric spaces*, thereby excluding interesting domains such as $\mathbb{R}^d$ or infinite discrete spaces.

Theorem 1 (ss-c_0-universality of k_E)**.** *Assume that the base kernel k is c_0-universal on the locally compact Hausdorff space $\mathcal{X}$. Then the generalised preferential kernel $k_E((u,u'),(v,v')) = k(u,v)k(u',v') - k(u,v')k(u',v)$ is ss-c_0-universal on $\mathcal{X} \times \mathcal{X}$.*

3.2 Clusters of comparable items

Clustering is a popular method to consider latent structures behind preferential data. Many existing methods [Cao et al., 2012, Li et al., 2018, Grbovic et al., 2013, Fogel et al., 2016] cluster items based on their similarity devised from the outcomes of matches. For example, in Fogel et al. [2016] the authors used a two-hop aggregation method on the preference graph to compute the similarity between two items, i.e. $S_{i,j} = \sum_{k=1}^n y_{i,k} y_{j,k}$. In this work, we consider a different notion of clustering for preferential data, which we term *clusters of comparable items*. In particular, we are interested in discovering groups of items that are comparable and thus rankable within clusters but not across. Cases like this might arise when the pairwise comparison is defined indirectly. For example, product preferences are often deduced using product search histories in e-commerce [Karmaker Santu et al., 2017] and products may not always belong to the same categories. A related problem is studied in partial rankings [Cheng et al., 2012], where certain pairs of items can be declared as incomparable by thresholding the probabilities of pairwise preferences between items. In contrast to partial rankings though, we do not need to consider individual probabilities, and by clustering the items, all pairings across clusters are declared as incomparable.

Consider a latent preference function g and assume that it belongs to $\mathcal{H}_{k_E}$. We can associate to g a skew-symmetric Hilbert-Schmidt operator $S_g : \mathcal{H}_k \to \mathcal{H}_k$ which satisfies

$$\langle k(\cdot, x), S_g k(\cdot, x') \rangle_{\mathcal{H}_k} = g(x, x'). \quad (5)$$

For example, if $g(x,x') = f(x) - f(x')$ then S_g is a rank two operator given by $S_g = f \otimes e - e \otimes f$ and $e(x) = 1$ is the constant function. Conversely, if S_g has rank two and one of its top singular functions is constant, a total order can be imposed on $\mathcal{X}$ by the non-constant top singular function. Similar reasoning can also be applied to the match outcomes matrix directly and is the core idea behind SVD-based approaches to ranking [d'Aspremont et al., 2021, Chau et al., 2020].

In general, however, S_g may have a higher rank. Specifically, in the case of the existence of L clusters of comparable items, S_g can be written as an operator of rank $2L$ given by

$$S_g = \sum_{l=1}^{L} (f_l \otimes e_l - e_l \otimes f_l) \tag{6}$$

where f_l is the utility function of the l-th cluster and e_l is the l-th cluster indicator function, i.e. it equals to 1 if item x belongs to cluster l, and 0 otherwise.

We are now interested in extracting clusters of comparable items from a fitted function g. Assuming (6), the true complete preference matrix G with $G_{i,j} = g(x_i, x_j)$ satisfies

$$G = \sum_{l=1}^{L} (\mathbf{f}_l \mathbf{1}_l^\top - \mathbf{1}_l \mathbf{f}_l^\top). \tag{7}$$

$\mathbf{f}_l$ is the vector of evaluations of the l-th cluster utility function f_l and $\mathbf{1}_l$ is the l-th cluster indicator vector, i.e. its j-th entry equals to 1 if item x_j belongs to cluster l, and 0 otherwise.

To recover the clusters, we first estimate the preference matrix $\hat{G}$ using GPGP and treat it as a noisy version of the true low rank matrix G. The clusters can then be recovered by applying standard clustering algorithms (e.g. K-means) to the data representation given by the top $2L$ singular vectors from $\hat{G}$, analogously to classical spectral clustering.

3.3 Data augmentation baseline

It is simple to extend any classification algorithm to model skew-symmetric duelling preferences using data augmentation, without assuming rankability. One example is to take an observation (x_i, x_j, y_{ij}) of the match between x_i and x_j, and concatenate the two sets of item covariates in two different orders, as $x_{i,j} = [x_i, x_j]$ and $x_{j,i} = [x_j, x_i]$ and pass them to a classification model with both $x_{i,j}$, $x_{j,i}$ as inputs and $y_{i,j}$ and $y_{j,i} = -y_{i,j}$ as their respective targets. While such data augmentation does encourage skew-symmetry, the resulting function is not guaranteed to be skew-symmetric on all inputs. Skew-symmetry can then be enforced by averaging the model outputs:

$$\begin{aligned} \hat{p}(y_{ij} = 1 | (x_i, x_j)) &= \frac{1}{2}\hat{p}_{\text{cat}}(y_{ij} = 1 | x_{i,j}) \\ &\quad + \frac{1}{2}\hat{p}_{\text{cat}}(y_{ji} = -1 | x_{j,i}), \end{aligned} \tag{8}$$

where $\hat{p}_{\text{cat}}$ are the probabilities fitted on the concatenated item covariates. Although this ad-hoc augmentation allows us to relax the rankability assumption in preference learning and is applicable to any models, including GPs, its theoretical justification is questionable, and the additional computational cost due to doubling the data size may be problematic.

We note that another approach applicable to linear models would be to impose skew-symmetry via model coefficients directly, but it is not clear how one might extend it to nonparametric methods such as GPs. We provide further discussion of this line of reasoning with its connection to the feature maps of PGP and GPGP in the Appendix.

3.4 Scalability

Since GPGP is formulated on the joint item space $\mathcal{X} \times \mathcal{X}$ of pairs of items, computational considerations need to be taken into account. In the worst case scenario, we may be storing and inverting a $\binom{n}{2} \times \binom{n}{2}$ kernel matrix for n items, if a match is played between every pair of items. This seldom happens in practice, however. In fact, most real-world comparison data is highly sparse, especially if the number of items n is large. Nonetheless, there are a large number of well established ways to scale up GPs that can be readily applied to GPGP, e.g. variational inducing points [Hensman et al., 2015] or conjugate gradient methods [Filippone and Engler, 2015]. In addition, Gardner et al. [2018] proposed techniques to reduce the asymptotic complexity of exact GP inference from cubic to quadratic. One can also use methods such as KISS-GP [Wilson and Nickisch, 2015] exploiting Kronecker and Toeplitz algebra for further speedups. Kronecker structure of kernel matrices, as well as conjugate gradient methods were also exploited by Pahikkala et al. [2013] in the context of regularized least squares with generalised preferential kernel.

4 Experiments

Our experiments demonstrate the key aspect of GPGP: the ability to model cyclic and inconsistent preferences from duelling data. In section 4.1, we study the robustness of GPGP using simulated preferences with different levels of sparsity and inconsistencies. Section 4.2 studies the problem of clusters of comparable items using simulation to further showcase how GPGP can learn complex preferential structures. Finally, we conclude the experiments by testing GPGP against alternative preference prediction methods using 4 real-world datasets with a total of 22 examples. As baselines, we compare GPGP with *Preferential GP* (PGP), *GP with data augmentation* (PAIR-GP) and *Logistic Regression with data augmentation* (PAIR-LOGREG). The latter two baselines use a scheme described in 3.3. For all methods involving kernels, we use the Gaussian radial basis function kernel (RBF) $k(x, x') = \exp\left(-\frac{||x-x'||^2}{2\gamma^2}\right)$

and obtain lengthscale γ by optimising the evidence lower bound. We use Laplace approximation and conjugate gradient methods for inference in GPGP, PGP and PAIR-GP.

4.1 Simulation: Cyclic and inconsistent preferences

Data generation Consider a comparison network with n items and a covariate matrix $X \in \mathbb{R}^{n\times p}$. We assign to each node a latent state $z \in \{1, .., L\}$ and generate a set of utility functions $\{r_{z,z'}\}_{z,z'=1}^{L}$, i.e. there is a different utility function for each pair (z, z') of latent states. We let $r_{z,z'}(x) = \sum_{j=1}^{n} \alpha_j^{z,z'} k(x, x_j)$ with each vector $\alpha^{z,z'} \overset{i.i.d}{\sim} N(0, I_n)$. Comparison between node i and j is then conducted based on the utility selected by their latent states, i.e. $i \preceq j \iff r_{z_i,z_j}(x_i) < r_{z_i,z_j}(x_j)$. This setup brings in cyclic and inconsistent preferences to the overall preference graph. Figure 1a provides a visual illustration of the experiment with $L = 2$ with a cycle indicated in bold. Different colour of the edges indicates that a different criterion, i.e. utility function, is used in pairwise comparisons.

We simulate a preference graph with $n = 30$ players each containing $p = 5$ covariates with different level of graph sparsity and number of latent states ($L = 1, 2, 5$). Latent states are simulated uniformly. Item features are generated conditionally on latent states with $x|z \sim N(z\mathbf{1}, I_5)$, thus allowing the features to encapsulate information about the latent states. We do a $70 - 30$ train-test-split on the data and repeat the experiments 20 times.

Results Figure 2 gives the accuracy of GPGP when predicting preferences on held-out data in comparison with baselines. As L increases, we see a significant decrease in accuracy for PGP and PAIR-LOGREG whereas GPGP and PAIR-GP performed relatively stable. On average GPGP outperforms the other methods, except in the high sparsity regime with $L = 1$, where PGP performed better. In fact, this is not surprising as $L = 1$ corresponds to a perfectly rankable duelling problem since there is only one utility function.

4.2 Simulation: Clusters of comparable items

Data generation Similar to the setup from section 4.1, we assign to each data a latent state and match outcomes follow utility functions dependant on these states. However, when comparisons are made across latent groups, the outcome is a Bernoulli(1/2), independent of all else, due to items being non-comparable. See Figure 1b for a visual illustration. We simulate matches between 30 players each containing 5 features with different level of sparsity and number of latent clusters $L = 2, 3$.

We give three possible approaches of finding the clusters of comparable items,

1. GPGP-CLUS: First recover the latent preference matrix G using GPGP, then run KMeans on the top $2L$ corresponding singular vectors of G.
2. PR-CLUS: First apply the partial ranking with abstention method from Cheng et al. [2012] to remove non-comparable matches. SVD and KMeans are then applied to the trimmed comparison matrix.
3. SVD-CLUS: Apply KMeans to the data representation given by the top 2L singular vectors from the comparison graph directly.

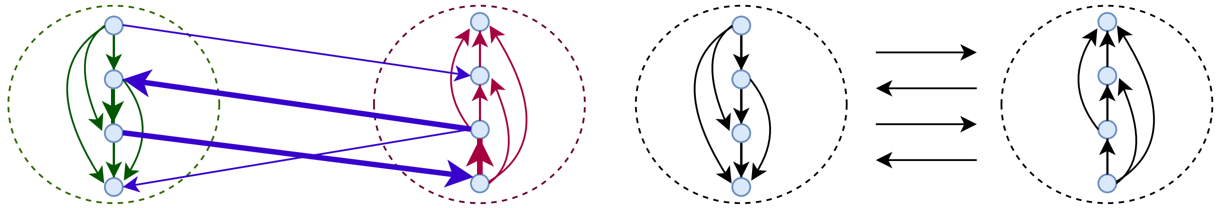
We report the proportion of items which are correctly clustered as a metric of performance. We do not include PGP-CLUS here because PGP performs poorly when there are multiple ranking signals.

Results Figure 3 gives the performance of the methods in recovering clusters of comparable items, comparing the proportion of the items each method clustered correctly. On average GPGP-CLUS performed better than the rest, except at low sparsity, i.e. dense graphs, where it performed similarly to SVD-CLUS. This is expected as for a highly dense preference graph, modelling with GPGP will not gain further additional information about the overall preference structure. On the other hand, PR-CLUS performed consistently poorly because it assumes rankability of the data. In other words, it only removes matches that agree with the sole ranking signal the algorithm recovered.

4.3 Predicting preferences on real data

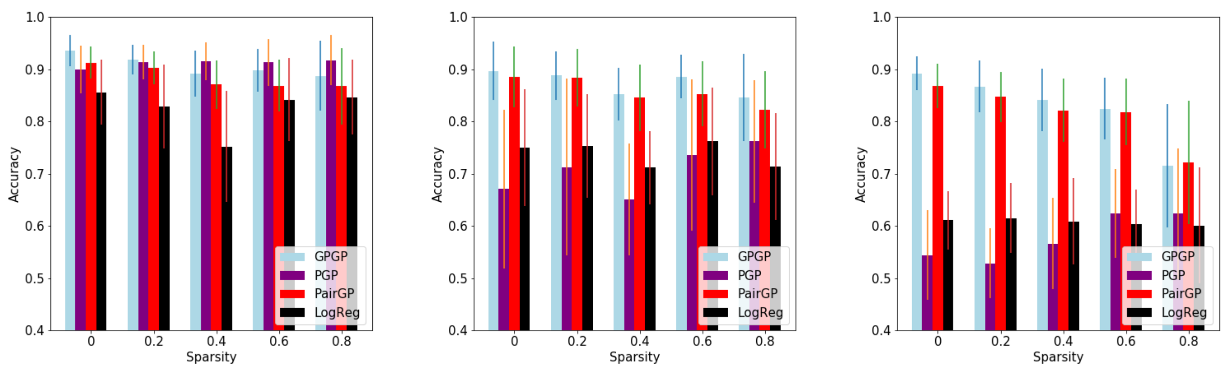
We apply GPGP and baselines to a variety of real-world comparison graphs, and measure outcome by their accuracy in predicting preferences on the test set. A 70-30 train-test split is applied to the data over 20 trials. Table 1 summarises the test results on 4 datasets for preference learning. We report the average network clustering coefficient C_{avg} [Saramäki et al., 2007] as a proxy to illustrate how non-rankable the problem is.

Male Cape Dwarf Chameleons Contest This data is used in the study by Stuart-Fox et al. [2006]. Physical measurements are made on 35 male Cape dwarf chameleons, and the results of 104 contests are recorded. From Table 1, we see that GPGP statistically outperformed all baselines. In particular, PGP was the worst performer due to the moderately high clustering coefficient.



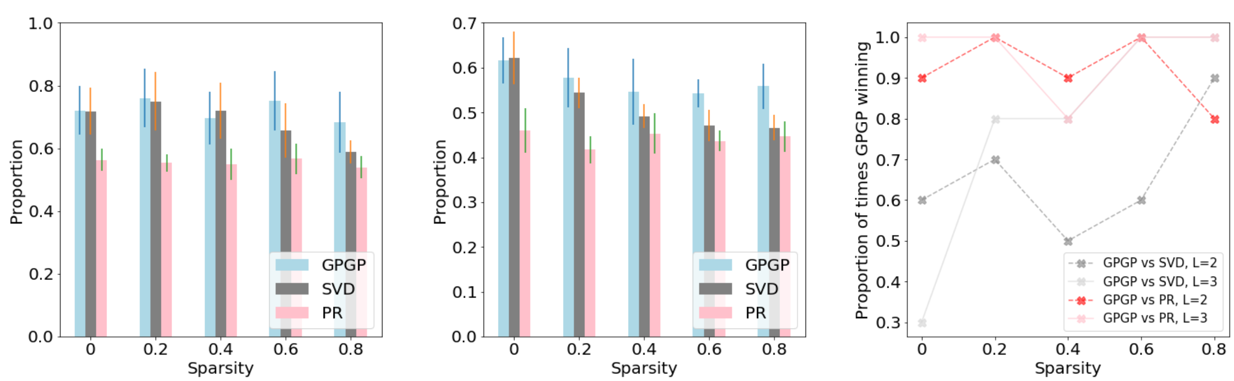
(a) Cyclic and Inconsistent Preference illustration (b) Clusters of comparable items illustration

Figure 1: (a) Items belongs to different groups and preference between items corresponds to the utility function determined by their latent states (different colors indicate that different utility function is used). Overall preferences exhibit cycles (indicated in bold) (b) Items belongs to different groups and items are rankable within the groups but preferences across groups are random.



(a) L = 1 (Perfectly Rankable) (b) L = 2 (c) L = 5

Figure 2: Comparisons of algorithms for simulations at different sparsity and inconsistency level. Accuracies are averaged over 20 runs and error bars of 1 standard deviation are provided.



(a) L = 2 (b) L = 3 (c)

Figure 3: (a, b) Comparisons of algorithms for simulations with different number of clusters and sparsity level. Proportion of items correctly clustered are averaged over 20 runs and error bars of 1 standard deviation are provided. (c) Proportion of times GPGP performed better than baselines.

Flatlizard Competition The data is collected at Augrabies Falls National Park (South Africa) in September-October 2002 [Whiting et al., 2009], on the contest performance and background attributes of 77 male flat lizards (*Platysaurus Broadleyi*). The results of 100 contests were recorded, along with 18

Table 1: Test results on the 4 datasets for preference learning. Accuracy averaged over 20 trials is reported along with its standard deviation. C_{avg} is the average clustering coefficient of a comparison graph. The symbol * indicates when the algorithm's accuracy is significantly worse than that of GPGP. Wilcoxon rank-sum test with level 0.05 was used to determine the statistical significance.

				Accuracy (%)			
DATASET	# Item	# Edge	C_{avg}	GPGP	PGP	PAIRGP	PAIRLOGREG
Chameleon	35	104	0.33	**0.78 ± 0.06**	0.51 ± 0.09*	0.72 ± 0.08*	0.71 ± 0.08*
Flatlizard	77	100	0.07	**0.83 ± 0.06**	0.80 ± 0.09	0.78 ± 0.07*	0.77 ± 0.09*
NFL 2000-18	32	213x19 yrs	0.54	0.59 ± 0.03	0.51 ± 0.02*	0.58 ± 0.03	**0.65 ± 0.03**
ArXiv Graph	1025	1000	0.11	**0.74 ± 0.02**	0.66 ± 0.03*	0.70 ± 0.02*	0.62 ± 0.02*

physical measurements made on each lizard, such as *weight* and *head size*. This comparison graph has the lowest average clustering coefficient thus is the most rankable compared to the rest. On average GPGP still performed better than PGP but the difference is not statistically significant.

NFL Football 2000-2018 The data contains the outcome of National Football League (NFL) matches during the regular season, for the years 2000 - 2018 [1]. In addition, 256 matches per year between 32 teams, along with 18 performance metrics, such as *yards per game* and *number of fumbles* are recorded. We pick the top 5 informative features by applying the BAHSIC feature selection algorithm [Song et al., 2012] and run the algorithm on each year's comparison graph separately and average the results. In this highly non-rankable ($C_{avg} = 0.54$) problem, PAIR-LOGREG outperformed the rest. This is not surprising as the features (e.g. *yards per game*) are expected to be linearly related to the match outcome and a linear model may thus better capture these relationships. Nonetheless, GPGP still outperformed PGP.

ArXiv Citation Network The last dataset we use is from the Open Graph Benchmark [Hu et al., 2020] arXiv Computer Science papers citation network. Each paper represents a node and an edge from node $i \to j$ means paper i cited paper j. We pick an induced subgraph with 1025 nodes and 1000 edges from the full network. Each node contains a 128-dimensional feature vector obtained by averaging the embedding of words in its title and abstract. Again, we see GPGP performed significantly better than the other algorithms. It is interesting to note that PAIR-LOGREG was the worst performing method, indicating that word-embedding features, in contrast to the features from the NFL problem, have a highly non-linear relationship with the match outcome.

[1] data collected from nfl.com

5 Conclusion and Discussion

We proposed *Generalised Preferential Gaussian Processes* (GPGP), a new probabilistic model for preferential data. GPGP relaxed the rankability assumption and comes with a strong theoretical justification in terms of universality of the corresponding kernel function. It can be readily integrated into many existing preference learning algorithms that are based on PGP. Experimental results on simulations and real-world datasets show the superior performance in comparison to PGP, the latter demonstrating the prevalence of inconsistent preferences and the need for relaxing the rankability assumptions in practice. We demonstrated how GPGP can be used to solve a specific problem which goes beyond rankability, i.e. recovering clusters of comparable items. A number of other problems which similarly involve more complex preferential structures can be studied based on the proposed framework.

Relaxing rankability allows to investigate latent structures influencing preferences, including the case where preferences are inconsistent, cyclical or when many items are simply not comparable to each other. Building on the existing preferential Gaussian Process (PGP) model, our approach introduces additional flexibility but preserves the advantages of having a Bayesian probabilistic model and faithful uncertainty quantification. The algorithms we proposed may enable more robust and customised recommendations to users in recommender systems and information retrieval. It is also envisaged that our work will find applications in A/B testing, gaming systems, and Bayesian optimisation with implicit or relative feedback.

Digital trails such as web searches and purchase patterns are often collected for targeted recommendations. It is worth noting that these features might include sensitive personal information and utilising them without careful consideration might be unethical. Therefore, an important practical research direction will be to consider combining GPGP with algorithmic fairness approaches applicable to kernel methods and GPs [Li et al., 2019], or to use differentially private mechanisms for GPs [Smith et al., 2018].

References

Wei Chu and Zoubin Ghahramani. Preference learning with Gaussian processes. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 137–144, 2005.

B. Sinervo and C.M. Lively. The rock-paper-scissors game and the evolution of alternative male strategies. *Nature*, 380(6571):240–243, 1996.

William V Gehrlein. Condorcet's paradox. *Theory and Decision*, 15(2):161–197, 1983.

Edwin Simpson and Iryna Gurevych. Scalable Bayesian preference learning for crowds. *Machine Learning*, 109:689–718, 2020.

Javier González, Zhenwen Dai, Andreas Damianou, and Neil D Lawrence. Preferential Bayesian optimization. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1282–1291, 2017.

Masrour Zoghi, Zohar S Karnin, Shimon Whiteson, and Maarten De Rijke. Copeland dueling bandits. In *Advances in Neural Information Processing Systems*, pages 307–315, 2015.

Trung V Nguyen, Alexandros Karatzoglou, and Linas Baltrunas. Gaussian process factorization machines for context-aware recommendations. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 63–72, 2014.

Luisa M Zintgraf, Diederik M Roijers, Sjoerd Linders, Catholijn M Jonker, and Ann Nowé. Ordered preference elicitation strategies for supporting multi-objective decision making. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 1477–1485, 2018.

Willem Waegeman, Tapio Pahikkala, Antti Airola, Tapio Salakoski, Michiel Stock, and Bernard De Baets. A kernel-based framework for learning graded relations from data. *IEEE Transactions on Fuzzy Systems*, 20(6):1090–1101, 2012.

Bharath K Sriperumbudur, Kenji Fukumizu, and Gert RG Lanckriet. Universality, characteristic kernels and RKHS embedding of measures. *Journal of Machine Learning Research*, 12(Jul):2389–2410, 2011.

Weiwei Cheng, Eyke Hüllermeier, Willem Waegeman, and Volkmar Welker. Label ranking with partial abstention based on thresholded probabilistic models. In *Advances in neural information processing systems*, pages 2501–2509, 2012.

Johannes Fürnkranz and Eyke Hüllermeier. Preference learning and ranking by pairwise comparison. In *Preference learning*, pages 65–82. Springer, 2010.

Alexandre d'Aspremont, Mihai Cucuringu, and Hemant Tyagi. Ranking and synchronization from pairwise measurements via SVD. *Journal of Machine Learning Research*, 22(19):1–63, 2021.

Neil Houlsby, Ferenc Huszar, Zoubin Ghahramani, and Jose M Hernández-Lobato. Collaborative gaussian processes for preference learning. In *Advances in neural information processing systems*, pages 2096–2104, 2012.

Daniel Kahneman and Amos Tversky. On the interpretation of intuitive probability: A reply to Jonathan Cohen. *Cognition*, 7(4):409–411, 1979.

Tapio Pahikkala, Evgeni Tsivtsivadze, Antti Airola, Jouni Järvinen, and Jorma Boberg. An efficient algorithm for learning to rank from preference graphs. *Machine Learning*, 75(1):129–165, 2009.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the Web. In *Proceedings of the 7th International World Wide Web Conference*, pages 161–172, 1998.

Tapio Pahikkala, Willem Waegeman, Evgeni Tsivtsivadze, Tapio Salakoski, and Bernard De Baets. Learning intransitive reciprocal relations with kernel methods. *European Journal of Operational Research*, 206(3):676–685, 2010.

Christopher KI Williams and Carl Edward Rasmussen. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

James Hensman, Alexander Matthews, and Zoubin Ghahramani. Scalable Variational Gaussian Process Classification. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38, pages 351–360, 2015.

Nir Ailon and Mehryar Mohri. Preference-based learning to rank. *Machine Learning*, 80(2-3):189–211, 2010.

Louis L Thurstone. A law of comparative judgment. *Psychological review*, 101(2):266, 1994.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. The method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

R Duncan Luce. On the possible psychophysical laws. *Psychological review*, 66(2):81, 1959.

Frederick Mosteller and Philip Nogee. An experimental measurement of utility. *Journal of Political Economy*, 59(5):371–404, 1951.

Thorsten Joachims. Svm-rank: Support vector machine for ranking. *Cornell University*, 2009.

Fajwel Fogel, Alexandre d'Aspremont, and Milan Vojnovic. Spectral ranking using seriation. *Journal of Machine Learning Research*, 17(1):3013–3057, 2016.

Mihai Cucuringu. Sync-rank: Robust ranking, constrained ranking and rank aggregation via eigenvector and sdp synchronization. *IEEE Transactions on Network Science and Engineering*, 3(1):58–79, 2016.

Amos Tversky and Daniel Kahneman. Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and Uncertainty*, 5(4): 297–323, 1992.

Paul Anderson, Timothy Chartier, and Amy Langville. The rankability of data. *SIAM Journal on Mathematics of Data Science*, 1(1):121–143, 2019.

Thomas R Cameron, Amy N Langville, and Heather C Smith. On the graph laplacian and the rankability of data. *Linear Algebra and its Applications*, 588: 81–100, 2020.

David Causeur and François Husson. A 2-dimensional extension of the Bradley–Terry model for paired comparisons. *Journal of Statistical Planning and Inference*, 135(2):245–259, 2005.

Shuo Chen and Thorsten Joachims. Modeling intransitivity in matchup and comparison data. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, page 227–236, 2016.

Motonobu Kanagawa, Philipp Hennig, Dino Sejdinovic, and Bharath K Sriperumbudur. Gaussian processes and kernel methods: A review on connections and equivalences. *arXiv preprint arXiv:1807.02582*, 2018.

Roland Kwitt, Stefan Huber, Marc Niethammer, Weili Lin, and Ulrich Bauer. Statistical topological data analysis - a kernel perspective. In *Advances in Neural Information Processing Systems 28*, pages 3070–3078. 2015.

Jan Reininghaus, Stefan Huber, Ulrich Bauer, and Roland Kwitt. A stable multi-scale kernel for topological machine learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4741–4748, 2015.

Charles A Micchelli, Yuesheng Xu, and Haizhang Zhang. Universal kernels. *Journal of Machine Learning Research*, 7(Dec):2651–2667, 2006.

Liangliang Cao, Xin Jin, Zhijun Yin, Andrey Del Pozo, Jiebo Luo, Jiawei Han, and Thomas S Huang. Rankcompete: Simultaneous ranking and clustering of information networks. *Neurocomputing*, 95: 98–104, 2012.

Jiyi Li, Yukino Baba, and Hisashi Kashima. Simultaneous clustering and ranking from pairwise comparisons. In *IJCAI*, pages 1554–1560, 2018.

Mihajlo Grbovic, Nemanja Djuric, Shengbo Guo, and Slobodan Vucetic. Supervised clustering of label ranking data using label preference information. *Machine learning*, 93(2-3):191–225, 2013.

Shubhra Kanti Karmaker Santu, Parikshit Sondhi, and ChengXiang Zhai. On application of learning to rank for e-commerce search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 475–484, 2017.

Siu Lun Chau, Mihai Cucuringu, and Dino Sejdinovic. Spectral ranking with covariates. *arXiv preprint arXiv:2005.04035*, 2020.

Maurizio Filippone and Raphael Engler. Enabling scalable stochastic gradient-based inference for gaussian processes by employing the unbiased linear system solver (ULISSE). In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pages 1015–1024, 2015.

Jacob Gardner, Geoff Pleiss, Kilian Q Weinberger, David Bindel, and Andrew G Wilson. GPyTorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. In *Advances in Neural Information Processing Systems*, pages 7576–7586, 2018.

Andrew Wilson and Hannes Nickisch. Kernel interpolation for scalable structured gaussian processes (kiss-gp). In *International Conference on Machine Learning*, pages 1775–1784, 2015.

Tapio Pahikkala, Antti Airola, Michiel Stock, Bernard De Baets, and Willem Waegeman. Efficient regularized least-squares algorithms for conditional ranking on relational data. *Machine Learning*, 93 (2-3):321–356, 2013.

Jari Saramäki, Mikko Kivelä, Jukka-Pekka Onnela, Kimmo Kaski, and Janos Kertesz. Generalizations of the clustering coefficient to weighted complex networks. *Physical Review E*, 75(2):027105, 2007.

Devi M Stuart-Fox, David Firth, Adnan Moussalli, and Martin J Whiting. Multiple signals in chameleon contests: designing and analysing animal contests as a tournament. *Animal Behaviour*, 71(6):1263–1271, 2006.

Martin J Whiting, Jonathan K Webb, and J Scott Keogh. Flat lizard female mimics use sexual deception in visual but not chemical signals. *Proceedings of the Royal Society B: Biological Sciences*, 276(1662): 1585–1591, 2009.

Le Song, Alex Smola, Arthur Gretton, Justin Bedo, and Karsten Borgwardt. Feature selection via dependence maximization. *Journal of Machine Learning Research*, 13(May):1393–1434, 2012.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *arXiv preprint arXiv:2005.00687*, 2020.

Zhu Li, Adrian Perez-Suay, Gustau Camps-Valls, and Dino Sejdinovic. Kernel dependence regularizers and gaussian processes with applications to algorithmic fairness. *arXiv preprint arXiv:1911.04322*, 2019.

Michael T. Smith, Mauricio A. Álvarez, Max Zwiessele, and Neil D. Lawrence. Differentially private regression with Gaussian processes. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, pages 1195–1203, 2018.

Gerald B Folland. *Real analysis: modern techniques and their applications*. John Wiley & Sons, 1999.

Walter Rudin. *Functional analysis*. McGraw-Hill, 1990.

Zoltán Szabó and Bharath K Sriperumbudur. Characteristic and universal tensor product kernels. *Journal of Machine Learning Research*, 18(1):8724–8752, 2017.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, 2008.

Supplementary Material: Learning Inconsistent Preferences with Gaussian Processes

A Proof of ss-c_0-Universality

To prove Theorem 1, we present three propositions to establish the link between ss-c_0-universality and integrally strictly positive definite kernels, following closely the characterisation of c_0-universal kernels from Sriperumbudur et al. [2011, Proposition 4]. The domain $\mathcal{X}$ is assumed to be a locally compact Hausdorff space. We denote by $C_0(\mathcal{X} \times \mathcal{X})$ the space of functions $f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ which are continuous, bounded, and vanish at infinity. We let $C_{0,ss}(\mathcal{X} \times \mathcal{X}) \subset C_0(\mathcal{X} \times \mathcal{X})$ be the subspace of skew-symmetric functions. We denote by $M_{b,ss}(\mathcal{X} \times \mathcal{X})$ the set of finite signed Radon measures on $\mathcal{X} \times \mathcal{X}$ and similarly $M_{b,ss}(\mathcal{X} \times \mathcal{X}) \subset M_b(\mathcal{X} \times \mathcal{X})$ is the subset of skew-symmetric Radon measures.

Proposition 1. *Let $\mathcal{X}$ be a locally compact Hausdorff space and $C'_{0,ss}(\mathcal{X}\times\mathcal{X})$ the topological dual of $C_{0,ss}(\mathcal{X}\times\mathcal{X})$. Then there is a bijective linear isometry $\nu \mapsto T_\nu$ from $M_{b,ss}(\mathcal{X} \times \mathcal{X})$ onto $C'_{0,ss}(\mathcal{X} \times \mathcal{X})$ given by the natural mapping, $T_\nu(f) = \int f d\nu, f \in C_{0,ss}(\mathcal{X} \times \mathcal{X})$. Thus we can identify $C'_{0,ss}(\mathcal{X} \times \mathcal{X}) = M_{b,ss}(\mathcal{X} \times \mathcal{X})$.*

Proof. By the Riesz representation theorem [Folland, 1999, Theorem 7.17], $C'_0(\mathcal{X} \times \mathcal{X}) = M_b(\mathcal{X} \times \mathcal{X})$ for corresponding spaces without enforcing skew-symmetry. Since $C_{0,ss}(\mathcal{X} \times \mathcal{X}) \subseteq C_0(\mathcal{X} \times \mathcal{X})$, for every linear functional $T_\nu \in C'_{0,ss}(\mathcal{X} \times \mathcal{X})$, there is a unique measure $\nu \in M_b(\mathcal{X} \times \mathcal{X})$. We will show that ν must be skew-symmetric.

Assume ν is not skew-symmetric, since $\nu \mapsto T_\nu$ is unique, we have,

$$T_\nu(f) = \int f(x, x')d\nu(x, x') = -\int f(x', x)d\nu^t(x', x) \tag{9}$$

where ν^t is the transpose of ν. Furthermore, we decompose $\nu = \nu^+ + \nu^-$ into a symmetric and skew-symmetric component with $\nu^+ = \frac{1}{2}(\nu + \nu^t)$ and $\nu^- = \frac{1}{2}(\nu - \nu^t)$. Thus, $T_\nu(f) = \int f d\nu = \int f d\nu^+ + \int f d\nu^-$, however,

$$\int f d\nu^+ = \frac{1}{2}\int f d\nu + \frac{1}{2}\int f d\nu^t = 0 \tag{10}$$

by uniqueness. Therefore ν is skew-symmetric and $C'_{0,ss}(\mathcal{X} \times \mathcal{X}) = M_{b,ss}(\mathcal{X} \times \mathcal{X})$. □

Proposition 1 demonstrates the equivalence between the dual of $C_{0,ss}(\mathcal{X}\times\mathcal{X})$ and the measure space $M_{b,ss}(\mathcal{X}\times\mathcal{X})$. This fact is then used along with the Hahn-Banach theorem [Rudin, 1990, Theorem 3.5] to prove a necessary and sufficient condition for k to be ss-c_0-universal.

Proposition 2. *Suppose $\mathcal{X} \times \mathcal{X}$ is a locally compact Hausdorff space with kernel k bounded and $k(\cdot, (x, x')) \in C_{0,ss}(\mathcal{X} \times \mathcal{X}), \forall (x, x') \in \mathcal{X} \times \mathcal{X}$. Then k is ss-c_0-universal if and only if the embedding*

$$\nu \mapsto \int k(\cdot, (x, x'))\, d\nu(x, x') \tag{11}$$

is injective for all $\nu \in M_{b,ss}(\mathcal{X} \times \mathcal{X})$.

Proof. By definition, k is ss-c_0-universal if $\mathcal{H}_k$ is dense in $C_{0,ss}(\mathcal{X} \times \mathcal{X})$. This can be shown directly by applying the Hahn-Banach theorem [Rudin, 1990, Theorem 3.5], which states that $\mathcal{H}_k$ is dense in $C_{0,ss}(\mathcal{X} \times \mathcal{X})$ if and only if $\mathcal{H}_k^\perp := \{T \in C'_{0,ss}(\mathcal{X} \times \mathcal{X}) : \forall f \in \mathcal{H}_k, T(f) = 0\} = \{0\}$. However, $C'_{0,ss}(\mathcal{X} \times \mathcal{X}) = M_{b,ss}(\mathcal{X} \times \mathcal{X})$ by Proposition 1, therefore $\mathcal{H}_k^\perp = \{\nu \in M_{b,ss}(\mathcal{X} \times \mathcal{X}) : \forall f \in \mathcal{H}_k, \int f d\nu = 0\} = \{0\}$. A direct application of the Riesz representation theorem shows that $\mathcal{H}_k^\perp = \{\nu \in M_{b,ss}(\mathcal{X} \times \mathcal{X}) : \int k(\cdot, (x, x'))d\nu(x, x') = 0\}$ thus proving injectivity. □

Finally, we connect ss-c_0-universal kernels to integrally strictly pd kernels as follows:

Proposition 3. *Let $\mathcal{X}$ be a locally compact Hausdorff metric space and k a continuous kernel on the joint space $\mathcal{X} \times \mathcal{X}$. Then, k is ss-c_0-universal if and only if $\mu_k : M_{b,ss}(\mathcal{X} \times \mathcal{X}) \to \mathcal{H}_k$ is a vector space monomorphism, that is,*

$$||\mu_k(\nu)||^2_{\mathcal{H}_k} = \int\int k((u,u'),(v,v'))d\nu((u,u'))d\nu((v,v')) > 0 \quad \forall \nu \in M_{b,ss}(\mathcal{X} \times \mathcal{X})\backslash\{0\}. \tag{12}$$

Proof. ($\Leftarrow$) Suppose k is not ss-c_0-universal. By Proposition 2, there exists $\nu \in M_{b,ss}(\mathcal{X} \times \mathcal{X})\backslash\{0\}$ such that $\int k(\cdot,(x,x'))d\nu(x,x') = 0$, which implies

$$\Big|\Big| \int k(\cdot,(x,x'))d\nu(x,x') \Big|\Big|^2_{\mathcal{H}_k} = \int\int k((u,u'),(v,v'))d\nu(u,u')d\nu(v,v') = 0$$

thus showing k is not integrally strictly pd. Therefore k has to be ss-c_0-universal.

($\Rightarrow$) Suppose there exists $\nu \in M_{b,ss}(\mathcal{X} \times \mathcal{X})\backslash\{0\}$ such that $\Big|\Big|\mu_k(\nu)\Big|\Big|_{\mathcal{H}_k} = 0$. This means,

$$\Big|\Big| \int k(\cdot,(x,x'))d\nu(x,x') \Big|\Big|^2_{\mathcal{H}_k} = 0 \Rightarrow \int k(\cdot,(x,x'))d\nu(x,x') = 0.$$

Therefore, the embedding is not injective, thus a contradiction by Proposition 2. Therefore, if k is ss-c_0-universal, then k satisfies (12). □

Now we can finish the proof for Theorem 1 using the above characterisations for ss-c_0-universal kernels.

Proof of Theorem 1. Pick any $\nu \in M_{b,ss}(\mathcal{X} \times \mathcal{X})\backslash\{0\}$. Consider the corresponding kernel mean embedding of ν i.e $\mu_{K_E}(\nu)$, we have:

$$||\mu_{k_E}(\nu)||^2_{\mathcal{H}_{k_E}} = \int_{(u,u')}\int_{(v,v')} k_E((u,u'),(v,v')) \quad d\nu((u,u'))d\nu((v,v')) \tag{13}$$

$$= \int\int k(u,v)k(u',v')d\nu_{u,u'}d\nu_{v,v'} - \int\int k(u,v')k(u',v)d\nu_{u,u'}d\nu_{v,v'} \tag{14}$$

$$= \int\int k(u,v)k(u',v')d\nu_{u,u'}d\nu_{v,v'} + \int\int k(u,v')k(u',v)d\nu_{u',u}d\nu_{v,v'} \tag{15}$$

$$= 2\int\int k(u,v)k(u',v')d\nu_{u,u'}d\nu_{v,v'} \tag{16}$$

$$= 2||\mu_{k\otimes k}(\nu)||^2_{\mathcal{H}_{k\otimes k}} \tag{17}$$

$$> 0. \tag{18}$$

We flip the sign in (15) because ν is skew symmetric. In the last inequality we used the fact that if k is universal on $\mathcal{X}$, then the product kernel is also universal on the product space $\mathcal{X} \times \mathcal{X}$ [Szabó and Sriperumbudur, 2017] hence they are integrally strictly pd [Sriperumbudur et al., 2011, Proposition 4]. Therefore by Proposition 3, k_E is ss-c_0-universal. □

B Extending the Generalised Preferential Kernel

The kernel we provided in the main paper in fact can be extended to tackle different preference learning situations.

Crowd Preferential Learning Given pairwise labels provided by a crowd, one can model the user-specific preference function $g : \mathcal{X} \times \mathcal{X} \times \mathcal{Z} \to \{0,1\}$ by setting up a RKHS with the following kernel,

$$k_E^z\Big((u,u',z),(v,v',z')\Big) = \Big(k(u,v)k(u',v') - k(u,v')k(u',v)\Big)k_z(z,z') \tag{19}$$

where k, k_z are kernels defined on the item space $\mathcal{X}$ and user space $\mathcal{Z}$ respectively. Appropriate universality can be shown to hold for k_E^z as well, provided k, k_z are universal respectively. If the same set of items are voted on by each user, one can further use tensor algebra to speed up computations since $K_E^z = K_E \otimes K_z$.

Distributional Preferential Learning We now consider situations where we would like to model preferences between groups of items while we only have access to individual level features. Football matches and e-sports tournaments are common examples of this setup. Mathematically this corresponds to the following setup: we have a dataset $\left\{\{x_i^a\}_{i=1}^{N_a}, \{x_i^b\}_{i=1}^{N_b}, y_{a,b}\right\}$ where each $B_a = \{x_i^a\}_{i=1}^{N_a}$ is assumed to be a sample coming from some distribution P_a, and $y_{a,b}$ is the preference outcome when B_a is compared to B_b. Since we only have preferences on the distributional level, we call this *Distributional Preferential Learning* and consider the following generative model

$$p(y_{a,b} = 1 | \{x_i^a\}_{i=1}^{N_a}, \{x_i^b\}_{i=1}^{N_b}) = \sigma(g(\{x_i^a\}_{i=1}^{N_a}, \{x_i^b\}_{i=1}^{N_b})). \tag{20}$$

Once again we consider g as a skew symmetric function corresponding to the RKHS $\mathcal{H}_{k_E^{(B)}}$ with the following kernel,

$$k_E^{(B)}\Big((B_a, B_b), (B_c, B_d)\Big) = k^{(B)}(B_a, B_c)k^{(B)}(B_b, B_d) - k^{(B)}(B_a, B_d)k^{(B)}(B_b, B_c) \tag{21}$$

where $k^{(B)}(B_a, B_c) = k^{(B)}(\{x_i^a\}_{i=1}^{N_a}, \{x_i^c\}_{i=1}^{N_c}) = \frac{1}{N_a N_c}\sum_{i=1}^{N_a}\sum_{j=1}^{N_c} k(x_i, x_j)$ is a linear kernel between the empirical kernel mean embeddings, which are commonly used as feature representations for probability distributions.

C Feature Maps of Preferential Kernels

We here briefly describe the differences in terms of feature space representations between the preferential and generalised preferential kernels. These are very similar to the differences between the sum-kernel and the product-kernel when combining kernels on individual domains to construct a kernel on the product domain.

The feature space of the preferential kernel

$$\kappa\left((u, u'), (v, v')\right) = k(u, v) + k(u', v') - k(u, v') - k(u', v)$$

is given by the direct sum $\mathcal{H}_k \oplus \mathcal{H}_k$ and the feature map is

$$\varphi : (u, u') \mapsto \frac{1}{\sqrt{2}}\left(k(\cdot, u) \oplus k(\cdot, u') - k(\cdot, u') \oplus k(\cdot, u)\right).$$

Indeed,

$$\begin{aligned}
&\langle \varphi(u, u'), \varphi(v, v')\rangle \\
&\quad = \frac{1}{2}\langle k(\cdot, u) \oplus k(\cdot, u') - k(\cdot, u') \oplus k(\cdot, u), k(\cdot, v) \oplus k(\cdot, v') - k(\cdot, v') \oplus k(\cdot, v)\rangle \\
&\quad = \frac{1}{2}\Big\{\langle k(\cdot, u) \oplus k(\cdot, u'), k(\cdot, v) \oplus k(\cdot, v')\rangle + \langle k(\cdot, u') \oplus k(\cdot, u), k(\cdot, v') \oplus k(\cdot, v)\rangle \\
&\qquad - \langle k(\cdot, u') \oplus k(\cdot, u), k(\cdot, v) \oplus k(\cdot, v')\rangle - \langle k(\cdot, u) \oplus k(\cdot, u'), k(\cdot, v') \oplus k(\cdot, v)\rangle\Big\} \\
&\quad = \frac{1}{2}\Big\{\langle k(\cdot, u), k(\cdot, v)\rangle + \langle k(\cdot, u'), k(\cdot, v')\rangle + \langle k(\cdot, u'), k(\cdot, v')\rangle + \langle k(\cdot, u), k(\cdot, v)\rangle \\
&\qquad - \langle k(\cdot, u'), k(\cdot, v)\rangle - \langle k(\cdot, u), k(\cdot, v')\rangle - \langle k(\cdot, u), k(\cdot, v')\rangle - \langle k(\cdot, u'), k(\cdot, v)\rangle\Big\} \\
&\qquad\qquad = k(u, v) + k(u', v') - k(u, v') - k(u', v).
\end{aligned}$$

On the other hand, the feature space of the generalised preferential kernel

$$\kappa\left((u, u'), (v, v')\right) = k(u, v)\, k(u', v') - k(u, v')\, k(u', v)$$

is given by the tensor product $\mathcal{H}_k \otimes \mathcal{H}_k$ and the feature map is

$$\varphi : (u, u') \mapsto \frac{1}{\sqrt{2}}\left(k(\cdot, u) \otimes k(\cdot, u') - k(\cdot, u') \otimes k(\cdot, u)\right).$$

Indeed,

$$
\begin{aligned}
&\langle \varphi(u,u'), \varphi(v,v') \rangle \\
&= \frac{1}{2} \langle k(\cdot,u)\otimes k(\cdot,u') - k(\cdot,u')\otimes k(\cdot,u), k(\cdot,v)\otimes k(\cdot,v') - k(\cdot,v')\otimes k(\cdot,v) \rangle \\
&= \frac{1}{2} \Big\{ \langle k(\cdot,u)\otimes k(\cdot,u'), k(\cdot,v)\otimes k(\cdot,v') \rangle + \langle k(\cdot,u')\otimes k(\cdot,u), k(\cdot,v')\otimes k(\cdot,v) \rangle \\
&\quad - \langle k(\cdot,u')\otimes k(\cdot,u), k(\cdot,v)\otimes k(\cdot,v') \rangle - \langle k(\cdot,u)\otimes k(\cdot,u'), k(\cdot,v')\otimes k(\cdot,v) \rangle \Big\} \\
&= \frac{1}{2} \Big\{ \langle k(\cdot,u), k(\cdot,v) \rangle \langle k(\cdot,u'), k(\cdot,v') \rangle + \langle k(\cdot,u'), k(\cdot,v') \rangle \langle k(\cdot,u), k(\cdot,v) \rangle \\
&\quad - \langle k(\cdot,u'), k(\cdot,v) \rangle \langle k(\cdot,u), k(\cdot,v') \rangle - \langle k(\cdot,u), k(\cdot,v') \rangle \langle k(\cdot,u'), k(\cdot,v) \rangle \Big\} \\
&\qquad = k(u,v)\,k(u',v') - k(u,v')\,k(u',v).
\end{aligned}
$$

These results of course also apply to finite-dimensional feature spaces where the direct sum operation corresponds to concatenation of the individual feature vectors of two players and the tensor product corresponds to an outer product between feature vectors. Namely, if ϕ is an explicit finite-dimensional feature map of kernel k, with explicit feature space $\mathbb{R}^m$, then feature map φ of kernel κ can be constructed as

$$
\varphi : (u,u') \mapsto \frac{1}{\sqrt{2}} \left(\begin{bmatrix} \phi(u) \\ \phi(u') \end{bmatrix} - \begin{bmatrix} \phi(u') \\ \phi(u) \end{bmatrix} \right) \in \mathbb{R}^{2m}
$$

in the case of the preferential kernel, and as

$$
\varphi : (u,u') \mapsto \frac{1}{\sqrt{2}} \left(\phi(u)\,\phi(u')^\top - \phi(u')\,\phi(u)^\top \right) \in \mathbb{R}^{m^2}
$$

in the case of the generalised preferential kernel. These feature maps can be used to construct large scale approximations via explicit feature representations for the preferential or generalised preferential kernel, using, for example, random Fourier features [Rahimi and Recht, 2008] for the base kernel k.

Linear base kernels. Let us consider preferential models with a linear base kernel $k(u,v) = u^\top v$. Here we will assume a logistic model for concreteness, like in the main text, but this of course readily extends to other forms of observation models. Likelihood in the preferential kernel case boils down to

$$
\begin{aligned}
p(y_{ij} = 1 | (x_i, x_j)) &= \sigma\left(\beta^\top \left(\begin{bmatrix} x_i \\ x_j \end{bmatrix} - \begin{bmatrix} x_j \\ x_i \end{bmatrix} \right) \right) \\
&= \sigma\left(\beta_1^\top (x_i - x_j) + \beta_2^\top (x_j - x_i) \right) \\
&= \sigma\left((\beta_1 - \beta_2)^\top (x_i - x_j) \right),
\end{aligned}
$$

where constant $1/\sqrt{2}$ is folded into the coefficient vector and we denoted the two halves of entries in β by β_1 and β_2 respectively. Hence we recover a simple logistic model on the differences between feature vectors with a p-dimensional vector of coefficients $w := \beta_1 - \beta_2$.

In contrast, as we will see, the generalised preferential kernel model starting with a linear base kernel parametrises likelihood using a general skew-symmetric bilinear form of the individual feature vectors. Collating coefficients into a $p \times p$ matrix B, we obtain likelihood given by

$$
\begin{aligned}
p(y_{ij} = 1 | (x_i, x_j)) &= \sigma\left(\operatorname{tr}\left[B \left(x_i x_j^\top - x_j x_i^\top \right)^\top \right] \right) \\
&= \sigma\left(x_i^\top B x_j - x_j^\top B x_i \right).
\end{aligned}
$$

We note that B can decomposed into symmetric and skew-symmetric part with $B^+ = \frac{1}{2}\left(B + B^\top\right)$ and $B^- = \frac{1}{2}\left(B - B^\top\right)$. Hence the likelihood becomes $\sigma\left(x_i^\top W x_j\right)$ where $W = 2B^- = B - B^\top$ is a skew-symmetric matrix.

Enforcing skew-symmetry via the coefficients. One can consider a more immediate way to construct a linear skew-symmetric model in the case of explicit features, by enforcing that the coefficients take particular form. These turn out to be equivalent to using feature maps described above. For example, we could have a model on the concatenation

$$\begin{aligned} p\left(y_{ij}=1 \middle| \left(x_i, x_j\right)\right) &= \sigma\left(\begin{bmatrix} w \\ -w \end{bmatrix}^\top \begin{bmatrix} \phi\left(x_i\right) \\ \phi\left(x_j\right) \end{bmatrix}\right) \\ &= \sigma\left(w^\top\left(\phi\left(x_i\right)-\phi\left(x_j\right)\right)\right). \end{aligned}$$

For a general bilinear model, we write

$$p\left(y_{ij}=1 \middle| \left(x_i, x_j\right)\right) = \sigma\left(\phi\left(x_i\right)^\top W \phi\left(x_j\right)\right),$$

and require that the matrix W is skew-symmetric, i.e. that $a^\top W b = -b^\top W a$.

Hence, we conclude that preferential and generalised preferential feature maps correspond to overparametrised versions of such models, where we parametrise functions using β rather than $\beta_1 - \beta_2$ and using B rather than $B - B^\top$. However, while constraints such as these may be enforceable in finite-dimensional feature spaces, it is not clear whether it is possible to enforce skew-symmetry directly on the dual coefficients in the infinite-dimensional case.