Experimental Design Under the Bradley-Terry Model

Yuan Guo[1*], Peng Tian[1], Jayashree Kalpathy-Cramer[2], Susan Ostmo[3], J.Peter Campbell[3], Michael F.Chiang[3], Deniz Erdoğmuş[1], Jennifer Dy[1] and Stratis Ioannidis[1]

[1] ECE Department, Northeastern University, Boston, MA, USA.

[2] Department of Radiology, Massachusetts General Hospital, Charlestown, MA, USA.

[3] Dept of Ophthalmology, Casey Eye Institute, Oregon Health & Science University, Portland, OR, USA.

[*]yuanee20@ece.neu.edu, [1]{pengtian, erdogmus, jdy, ioannidis}@ece.neu.edu,

[2]kalpathy@nmr.mgh.harvard.edu, [3]{ostmo, campbelp, chiangm}@ohsu.edu

Abstract

Labels generated by human experts via comparisons exhibit smaller variance compared to traditional sample labels. Collecting comparison labels is challenging over large datasets, as the number of comparisons grows quadratically with the dataset size. We study the following experimental design problem: given a budget of expert comparisons, and a set of existing sample labels, we determine the comparison labels to collect that lead to the highest classification improvement. We study several experimental design objectives motivated by the Bradley-Terry model. The resulting optimization problems amount to maximizing submodular functions. We experimentally evaluate the performance of these methods over synthetic and real-life datasets.

1 Introduction

In many supervised learning settings, labels generated by human experts while comparing pairs of samples exhibit smaller variance compared to traditional (i.e., pointwise) sample labels. For example, doctors assessing the presence of a disease in a patient often produce highly variable diagnostic outcomes; in contrast, their answers are less variable when assessing two patients w.r.t. *relative severity* of the disease [Kalpathy-Cramer *et al.*, 2016; Wallace *et al.*, 2008]. In this setting, a learner may only have access to diagnostic labels generated by doctors, who are noisy/subjective, and produce a diagnosis correlated with disease severity. The same severity drives diagnoses as well as comparison outcomes. Similar observations hold for many settings (e.g., recommendation systems) in which humans can easily rank samples but may find direct labeling difficult [Takahama *et al.*, 2016].

Asking an expert to produce, e.g., pairwise comparisons between samples, poses a significant challenge in large datasets. This is precisely because the number of comparisons grows quadratically with the dataset size. Motivated by this observation, we study a scenario in which an experimenter wishes to collect K *comparison* (i.e., pairwise) labels by querying one or more experts. We assume that, in making this decision, the experimenter has access to *absolute* (i.e., pointwise) labels for some of the samples in her dataset, which she wishes to augment with the collected K comparison labels.

We make the following contributions:

- We propose a generative model for both absolute *and* comparison labels, allowing us to take a probabilistic approach to the comparison selection problem. Our probabilistic assumptions are motivated by the so-called Bradley-Terry model [Bradley and Terry, 1952].
- We study several *experimental design* (a.k.a. batch active learning) objectives, and apply them to our specific generative model. All four are *monotone* and *submodular*, and three of them can be efficiently optimized by a greedy approximation algorithm. We also assess computational issues arising from these objectives under greedy optimization.
- Finally, we extensively evaluate these objectives over both synthetic and real-life datasets. We show that as few as 15 comparison labels collected via our proposed algorithms can improve classification measured via AUC by as much as 13%, and that our proposed methods outperform random selection under several different noise settings.

The remainder of this paper is organized as follows. We discuss related work in Section 2. Our problem formulation and a discussion of the optimization of different experimental design objectives can be found in Sections 3 and 4, respectively. Our numerical evaluations are in Section 5, and we conclude in Section 6.

2 Related Work

Integrating regression labels with ranking information was proposed in [Sculley, 2010] as a means to improve regression outcomes in label-imbalanced datasets, and similar approaches have been used to incorporate both "pointwise" and "pairwise" labels in image classification tasks [Chen *et al.*, 2015; Wang *et al.*, 2016]. The generative model we describe in Sec. 3.1 naturally relates to the penalty introduced by Sculley via MAP estimation (see also Sec. 3.4). None of these works deal with the experimental design problem, namely, how to collect comparison (i.e., pairwise) labels.

Experimental design, a.k.a. batch active learning, is a rich and extensively studied area [Delzell *et al.*, 2012; Myerson, 1981; Chaloner and Verdinelli, 1995; Flaherty *et al.*, 2006; Tsilifis *et al.*, 2017; Huan and Marzouk, 2013]. A generative

model based on logistic regression and a Fisher information-based objective is introduced in [Zhang and Oles, 2000; Hoi *et al.*, 2006]; we build upon this work to construct our Fisher information criterion. Mutual information (a.k.a, information gain) is also a commonly used objective [Ryan, 2003; Liepe *et al.*, 2013; Cavagnaro *et al.*, 2010], which is monotone submodular under certain conditions [Krause and Guestrin, 2005; Wei *et al.*, 2015; Guillory and Bilmes, 2011]. Applying this objective to our generative model retains submodularity but, as in other settings [Busetto *et al.*, 2013], both (a) computing the posterior of the model, as well as (b) evaluating the function when having access to this posterior, are intractable. Approximation methods via Monte Carlo sampling have been proposed in [Drovandi and Pettitt, 2013; Drovandi *et al.*, 2014]; we apply these techniques along with variational inference to produce an estimate without directly calculating the posterior.

Finally, [Grasshoff *et al.*, 2003] and [Glickman and Jensen, 2005] study experimental design on the Bradley-Terry model focusing on a single parameter (one-dimensional) learning setting. They use D-Optimal design [Graßhoff and Schwabe, 2008] and KL-divergence [Glickman and Jensen, 2005] as optimization objectives. We depart by studying experimental design over multi-dimensional features, and considering a broader array of objective functions.

3 Problem Formulation

We consider a setting in which data samples are labeled by an expert. Given an individual sample to label, the expert produces a binary *absolute label* indicating a classification result. Given two different samples, the expert produces a *comparison label*. The comparison label is also binary and indicates precedence with respect to the classification outcome. For example, if the sample is a medical diagnosis, the absolute labels indicate the existence of disease, while the comparison label indicates the relative severity between two samples. An experimenter has access to a dataset of noisy absolute labels generated by this expert. At the same time, the experimenter wishes to augment the dataset by adding comparison labels. As comparison labels are numerous and their acquisition is time-consuming, the experimenter only collects a subset of all possible comparison labels.

Formally, the experimenter has access to N samples, indexed by $i \in \mathcal{N} \equiv \{1, 2, \cdots, N\}$. Every sample has a feature vector $\boldsymbol{x}_i \in \mathbb{R}^d$, which is known to the experimenter. For each sample $i \in \mathcal{N}$, there is a noisy binary absolute label $Y_i \in \{+1, -1\}$ generated by the expert. We define $\mathcal{C} \equiv \{(i,j) : i, j \in \mathcal{N}, i < j\}$ to be the set of possible comparisons. For pair $(i,j) \in \mathcal{C}$, the experimenter can collect the comparison label $Y_{i,j} \in \{+1, -1\}$ by querying the expert. We use $\mathcal{A} \subseteq \mathcal{N}$ to represent the subset of absolute labels observed by the experimenter. In the process of augmentation, the experimenter collects K comparison labels. We use $\mathcal{S} \subseteq \mathcal{C}$, where $|\mathcal{S}| = K$, to represent the subset of comparison labels that the experimenter collects.

3.1 Generative Model

We assume that labels are generated according to the following probabilistic model. First, there exists a parameter vector $\boldsymbol{\beta} \in \mathbb{R}^d$, sampled from a Gaussian prior $\mathcal{N}(0, \sigma^2 \boldsymbol{I})$, such that for all $i \in \mathcal{N}$ and all $(i,j) \in \mathcal{C}$ the absolute labels Y_i and comparison labels $Y_{i,j}$ are independent conditioned on $\boldsymbol{\beta}$. Second, the conditional distribution of Y_i given $\boldsymbol{x}_i$ and $\boldsymbol{\beta}$ is given by a logistic model, i.e.,

$$\mathbf{P}(Y_i = +1 | \boldsymbol{x}_i, \boldsymbol{\beta}) = \frac{1}{1+\exp(-\boldsymbol{\beta}^T \boldsymbol{x}_i)}, \quad i \in \mathcal{N}. \tag{1}$$

Finally, the conditional distribution of $Y_{i,j}$ given $\boldsymbol{x}_i, \boldsymbol{x}_j$ and $\boldsymbol{\beta}$ is given by the Bradley-Terry model [Bradley and Terry, 1952]. The Bradley-Terry model assumes that every item $i \in \mathcal{N}$ is associated with a parameter $s_i \in \mathbb{R}^+$ s.t. $\mathbf{P}(Y_{i,j} = +1) = \frac{s_i}{(s_i + s_j)}$ for all $(i,j) \in \mathcal{C}$. We extend this model to incorporate features $\boldsymbol{x}_i \in \mathbb{R}^d, i \in \mathcal{N}$ as follows:

$$\mathbf{P}(Y_{i,j} = +1 | \boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{\beta}) = \frac{s(\boldsymbol{x}_i, \boldsymbol{\beta})}{s(\boldsymbol{x}_i, \boldsymbol{\beta}) + s(\boldsymbol{x}_j, \boldsymbol{\beta})}, \ (i,j) \in \mathcal{C}, \tag{2}$$

where $s(\boldsymbol{x}_i, \boldsymbol{\beta}) = \exp(\boldsymbol{\beta}^T \boldsymbol{x}_i)$.

We note that, although the discussion below focuses on a single expert, the probabilistic nature of the model below allows to naturally incorporate multiple experts, e.g., by having repeated, independent labels over the same comparison pair.

3.2 Experimental Design

As mentioned above, the experimenter augments the dataset by adding comparison labels. It is expensive and time consuming to collect all $|\mathcal{C}| = \frac{N(N-1)}{2}$ comparison labels. The experimenter thus collects K labels in a subset $\mathcal{S} \subseteq \mathcal{C}$, $|\mathcal{S}| = K$. In the experimental design problem, we seek a strategy for determining $\mathcal{S}$. To do so, we introduce objective functions $f : 2^{|\mathcal{C}|} \to \mathbb{R}$, that capture how informative samples in $\mathcal{S}$ are. Given such a function, the optimal subset $\mathcal{S}^*$ is a solution to the following problem:

$$\begin{aligned} &\text{Maximize } \ f(\mathcal{S}), \\ &\quad \text{subj. to } \ \mathcal{S} \subseteq \mathcal{C}, |\mathcal{S}| = K. \end{aligned} \tag{3}$$

We present several candidate objective functions f in Sec. 3.5.

3.3 Greedy Optimization

Unfortunately, for many objective functions of interest, the problem of selecting the most informative subset $\mathcal{S}$ is NP hard. However, we can maximize the objective function by leveraging the theory of submodular functions. A function $f : 2^\Omega \to \mathbb{R}$ is *submodular* if $f(\mathcal{T} \cup \{z\}) - f(\mathcal{T}) \geqslant f(\mathcal{D} \cup \{z\}) - f(\mathcal{D})$ for all $\mathcal{T} \subseteq \mathcal{D} \subseteq \Omega$ and $z \in \Omega$. Function f is called *monotone* if $f(\mathcal{D} \cup \{z\}) - f(\mathcal{D}) \geqslant 0$ for all $\mathcal{D} \subseteq \Omega$ and $z \in \Omega$. For a monotone submodular objective function, we can use the greedy algorithm, described by Alg. 1, which iteratively adds elements to the solution. Then, the following theorem holds:

Algorithm 1 Greedy Algorithm

Initialize $\mathcal{S} = \emptyset$
while $|\mathcal{S}| \leqslant K$ **do**
  $\boldsymbol{r^*} = \underset{(i,j) \notin \mathcal{S}, (i,j) \in \mathcal{C}}{\mathbf{argmax}} \ f(\mathcal{S} \cup (i,j)) - f(\mathcal{S})$
  $\mathcal{S} = \mathcal{S} \cup r^*$
end while
return $\mathcal{S}$

Theorem 3.1. *[Nemhauser* et al.*, 1978]. If f is monotone submodular and $f(\emptyset) = 0$, the greedy algorithm produces a solution $\mathcal{S}_K$ such that $f(\mathcal{S}_K) \geqslant (1 - 1/e)f(\mathcal{S}^*)$, where $\mathcal{S}^*$ is the optimal solution of Eq.* (3)*.*

Note that Alg. 1 is a polynomial time algorithm in the so-called *value-oracle* model, i.e., presuming access to an oracle evaluating objective f for any argument $\mathcal{S} \subseteq \Omega$. Indeed, it requires $O(KN^2)$ accesses to such an oracle. Hence, if oracle $f(S)$ is poly-time in $|\mathcal{S}|$ and $|\Omega|$, so is Alg. 1.

3.4 Maximum a Posteriori Estimation

After the collection of comparison labels, the dataset available to the experimenter consists of absolute labels in subset $\mathcal{A} \subseteq \mathcal{N}$ and comparison labels in subset $\mathcal{S} \subseteq \mathcal{C}$. The experimenter can train a classifier through maximum a posteriori estimation (MAP) over the generative model presented in Section 3.1. Then, the estimation of parameter vector $\boldsymbol{\beta}$ amounts to minimizing the following negative log-likelihood function:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\beta}; \mathcal{A}, \mathcal{S}) =& \textstyle\sum_{i \in \mathcal{A}} \mathbf{log}(1 + e^{-y_i \boldsymbol{\beta}^T \boldsymbol{x_i}}) \\ &+ \textstyle\sum_{(i,j) \in \mathcal{S}} \mathbf{log}(1 + e^{-y_{i,j} \boldsymbol{\beta}^T (\boldsymbol{x_i} - \boldsymbol{x_j})}) + \lambda ||\boldsymbol{\beta}||_2^2,\end{aligned} \tag{4}$$

where the penalty coefficient λ equals $1/\sigma^2$, and in practice is determined through cross-validation.

3.5 Experimental Design Objectives

In what follows, we use the notation $Y_E = \{y_e\}_{e \in E} \in \{-1, +1\}^E$, to denote the vector of labels (absolute or comparison) restricted to set $E \subseteq \mathcal{N} \cup \mathcal{C}$. Following the usual convention, we denote by Y_E the random variable and by y_E a respective sample. Finally, we use the abbreviation $(\cdot|y_E)$ to indicate $(\cdot|Y_E = y_E)$ as in $p(\cdot|y_E)$, $\mathbf{I}(\cdot|y_E)$, etc.

Mutual Information. Recall that the prior distribution is $\boldsymbol{\beta} \sim \mathcal{N}(0, \sigma^2 I_d)$. Our first objective function is to maximize the mutual information between the parameter vector $\boldsymbol{\beta}$ and selected comparison labels $Y_\mathcal{S}$, conditioned on the observed absolute labels, i.e:

$$\begin{aligned}f_1(\mathcal{S}) =& \mathbf{I}(\boldsymbol{\beta}; Y_\mathcal{S} | Y_\mathcal{A} = y_\mathcal{A}) \\ =& \mathbf{H}(Y_\mathcal{S} | Y_\mathcal{A} = y_\mathcal{A}) - \mathbf{H}(Y_\mathcal{S} | \boldsymbol{\beta}, Y_\mathcal{A} = y_\mathcal{A}),\end{aligned} \tag{5}$$

where $\mathbf{I}(\cdot|Y_\mathcal{A} = y_\mathcal{A})$ denotes the mutual information conditioned on the observed absolute labels and $\mathbf{H}(\cdot|Y_\mathcal{A} = y_\mathcal{A})$ denotes the entropy conditioned on the observed absolute labels. We compute the quantities in Eq. (5) using the Bradley-Terry generative model described in Sec. 3.1.

Information Entropy. Recall that given some observed absolute labels $y_\mathcal{A}$, we can estimate the parameter vector $\hat{\boldsymbol{\beta}}$ by:

$$\hat{\boldsymbol{\beta}} = \mathbf{argmax}_{\boldsymbol{\beta}} \, \mathcal{L}(\boldsymbol{\beta}; \mathcal{A}, \emptyset), \tag{6}$$

where the negative log-likelihood function $\mathcal{L}(\boldsymbol{\beta}; \mathcal{A}, \mathcal{S})$ is given by Eq. (4). Assuming that the experimenter estimates the parameter vector $\hat{\boldsymbol{\beta}}$, thus we can use information entropy to measure the unpredictability of $Y_\mathcal{S}$ [Lewis and Gale, 1994; Cohn *et al.*, 1996; Sharma and Bilgic, 2017]. This can be seen as a "point" estimate of the mutual information. Under our generative model, unlabeled samples are independent given $\hat{\boldsymbol{\beta}}$; hence, the information entropy objective can be written as:

$$f_2(\mathcal{S}) = \mathbf{H}(Y_\mathcal{S} | \boldsymbol{\beta} = \hat{\boldsymbol{\beta}}) = \textstyle\sum_{a \in \mathcal{S}} \mathbf{H}(Y_a | \boldsymbol{\beta} = \hat{\boldsymbol{\beta}}). \tag{7}$$

Covariance Matrix/D-Optimal Design. Our third objective is motivated by D-optimal design [Boyd and Vandenberghe, 2004; Horel *et al.*, 2014]. The D-optimal objective is the negative log entropy of a linear regression model under Gaussian noise. We can apply this to our model by treating both logistic models (1) and (2) as regressions; this yields the objective:

$$\begin{aligned}f_3(\mathcal{S}) &= \log\det(cI_d + \textstyle\sum_{i \in \mathcal{A}} \boldsymbol{x}_i \boldsymbol{x}_i^T + \sum_{(i,j) \in \mathcal{S}} \boldsymbol{x}_{i,j} \boldsymbol{x}_{i,j}^T) \\ &= \log\det(F(\mathcal{A}, \mathcal{S})),\end{aligned}$$

where $\boldsymbol{x}_{i,j} = \boldsymbol{x}_i - \boldsymbol{x}_j$, c is a positive value, I_d is the d-dimensional identity matrix, and $F(\mathcal{A}, \mathcal{S}) = cI_d + \sum_{i \in \mathcal{A}} \boldsymbol{x}_i \boldsymbol{x}_i^T + \sum_{(i,j) \in \mathcal{S}} \boldsymbol{x}_{i,j} \boldsymbol{x}_{i,j}^T$.

Fisher Information. The Fisher information measures the amount of information that an observable random feature $\boldsymbol{x}$ carries about an unknown parameter $\boldsymbol{\beta}$ upon which the probability of $\boldsymbol{x}$ depends. The Fisher information matrix can be written as [Zhang and Oles, 2000; Hoi *et al.*, 2006]:

$$I(\boldsymbol{\beta}) = -\int p(y|\boldsymbol{x}, \boldsymbol{\beta}) \tfrac{\partial^2}{\partial \boldsymbol{\beta}^2} \log p(y|\boldsymbol{x}, \boldsymbol{\beta}) \mathrm{d}\boldsymbol{x} \mathrm{d}y. \tag{8}$$

Let $p(\boldsymbol{x})$ be the feature distribution of all unlabeled examples in set $\mathcal{C}$ and $q(\boldsymbol{x})$ be the distribution of unlabeled examples in set $\mathcal{S}$ that are chosen for manual labeling. With the generative model in Section 3.1 and the estimation of parameter vector $\hat{\boldsymbol{\beta}}$ by Eq. (6), the Fisher information matrices for these two distributions can be written as:

$$\begin{aligned}I_p(\hat{\boldsymbol{\beta}}) &= \frac{1}{|\mathcal{C}|} \sum_{(i,j) \in \mathcal{C}} \pi(\boldsymbol{x}_{i,j})(1 - \pi(\boldsymbol{x}_{i,j})) \boldsymbol{x}_{i,j} \boldsymbol{x}_{i,j}^T + \delta I_d, \\ I_q(\mathcal{S}, \hat{\boldsymbol{\beta}}) &= \frac{1}{|\mathcal{S}|} \sum_{(i,j) \in \mathcal{S}} \pi(\boldsymbol{x}_{i,j})(1 - \pi(\boldsymbol{x}_{i,j})) \boldsymbol{x}_{i,j} \boldsymbol{x}_{i,j}^T + \delta I_d,\end{aligned} \tag{9}$$

where $\delta \ll 1$ is to avoid having a singular matrix, $\pi(\boldsymbol{x}_{i,j}) = \frac{1}{1 + \exp(-\boldsymbol{\beta}^T \boldsymbol{x}_{i,j})}$, for $(i,j) \in \mathcal{C}$, and $\boldsymbol{x}_{i,j} = \boldsymbol{x}_i - \boldsymbol{x}_j$. The matrices above relate to variance of the parameter estimate via the so-called Cramer-Rao bound [Rao, 1992]. In particular, maximizing the following objective function:

$$f_4(\mathcal{S}) = -\operatorname{tr}(I_q(\mathcal{S}, \hat{\boldsymbol{\beta}})^{-1} I_p(\hat{\boldsymbol{\beta}})). \tag{10}$$

minimizes the Cramer-Rao bound of the respective $\hat{\boldsymbol{\beta}}$.

4 Optimization of Different Objectives

In this section, we discuss how to optimize different objectives through the greedy algorithm.

Mutual Information. Objective function f_1 is submodular. This follows from a more general statement for graphical models [Krause and Guestrin, 2005]. We provide the proof in Appendix A. Despite its submodularity, function f_1 is hard to compute analytically for two reasons. First, the posterior of $\boldsymbol{\beta}$ is intractable [Robert, 2014], and as such the entropy $\mathbf{H}(Y_\mathcal{S}, Y_{i,j}|y_\mathcal{A})$ and $\mathbf{H}(Y_{i,j}|\boldsymbol{\beta}, y_\mathcal{A})$ are hard to compute. Second, even assuming that the posterior is available, computing these entropies involves a summation over all possible values $y_\mathcal{S}$ of $Y_\mathcal{S}$.

We use variational inference and a Monte-Carlo method to overcome the first challenge. Variational inference can be used to approximate the posterior distribution $p(\boldsymbol{\beta}|y_\mathcal{A})$ via a

Gaussian distribution [Jordan *et al.*, 1999]. We describe how to accomplish this in Appendix B.

Using this we can sample M pseudo random feature vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M \in \mathbb{R}^d$ from $\mathcal{N}(\boldsymbol{\mu}_{\mathcal{A}}, \boldsymbol{\Sigma}_{\mathcal{A}})$, and construct the following estimator:

$$\begin{aligned}&\hat{\mathbf{I}}_{\mathcal{S};\boldsymbol{\beta}|y_{\mathcal{A}}}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M)\\ =&\hat{\mathbf{H}}_{\mathcal{S}|y_{\mathcal{A}}}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M) - \hat{\mathbf{H}}_{\mathcal{S}|\boldsymbol{\beta},y_{\mathcal{A}}}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M),\end{aligned} \tag{11}$$

where

$$\hat{\mathbf{H}}_{\mathcal{S}|y_{\mathcal{A}}}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M) = -\sum_{y_{\mathcal{S}}} \hat{p}_{y_{\mathcal{S}}} \log \hat{p}_{y_{\mathcal{S}}},$$

$$\hat{\mathbf{H}}_{\mathcal{S}|\boldsymbol{\beta},y_{\mathcal{A}}}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M) = \sum_{a\in\mathcal{S}} \sum_{m=1}^{M} \mathbf{H}(Y_a|\boldsymbol{b}_m)/M,$$

and $\hat{p}_{y_{\mathcal{S}}} = \sum_{m=1}^{M} p(y_{\mathcal{S}}|\boldsymbol{b}_m)/M$. The estimator $\hat{\mathbf{I}}_{\mathcal{S};\boldsymbol{\beta}|y_{\mathcal{A}}}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M)$ is also a monotone submodular function of $\mathcal{S}$. Hence, we can still use the greedy algorithm to optimize it. Despite this approximation, computing the mutual information is still very computationally expensive, as it requires computation over all values $y_{\mathcal{S}}$, yielding a complexity of Alg. 1 is $O\left(MN^2(K2^K + d)\right)$.

Information Entropy. Recall that, given the parameter estimation $\hat{\boldsymbol{\beta}}$, labels are independent. As a result, the objective function f_2 is in fact *modular*. A consequence of this is that Alg. 1 is, in this case, optimal.

Line 3 on Alg. 1 involves the following equation:

$$f_2(\mathcal{S} \cup (i,j)) - f_2(\mathcal{S}) = \mathbf{H}(Y_{i,j}|\boldsymbol{\beta} = \hat{\boldsymbol{\beta}}). \tag{12}$$

Note that this does not depend on set $\mathcal{S}$. As a result, Alg. 1 admits the following simple implementation: compute quantity $\mathbf{H}(Y_{i,j}|\boldsymbol{\beta} = \hat{\boldsymbol{\beta}})$, for all $(i,j) \in \mathcal{C}$, and select the top K values. Each evaluation of (12) is $O(d)$; this yields a complexity of $O(N^2(K+d))$ for Alg. 1 under a naïve implementation of top-K selection; this can be further reduced to $O(N^2(\log K + d))$ by using an appropriate data strucure (e.g., a heap).

Covariance Matrix/D-Optimal Design. The monotonicity and submodularity of f_3 is classic (see, e.g., [Horel *et al.*, 2014]). Note that $f_3(\emptyset) = \log\det F(\mathcal{A}, \emptyset)$ is finite but nonzero; as a result, the guarantee provided by Thm. 3.1 applies to the recentered function $f_3(\mathcal{S}) - f_3(\emptyset)$.

We can avoid computing the determinant using the matrix determinant lemma [Harville, 1997], we have that:

$$f_3(\mathcal{S} \cup (i,j)) - f_3(\mathcal{S}) = \log(1 + \boldsymbol{x}_{i,j}^T F(\mathcal{A}, \mathcal{S})^{-1} \boldsymbol{x}_{i,j}), \tag{13}$$

Moreover, to compute this quantity, it is *not* necessary to perform a matrix inversion: by the Sherman–Morrison formula

$$F(\mathcal{A}, \mathcal{S} \cup r^*)^{-1} = F(\mathcal{A}, \mathcal{S})^{-1} - \frac{F(\mathcal{A},\mathcal{S})^{-1} \boldsymbol{x}_{r^*} \boldsymbol{x}_{r^*}^T F(\mathcal{A},\mathcal{S})^{-1}}{1 + \boldsymbol{x}_{r^*}^T F(\mathcal{A},\mathcal{S})^{-1} \boldsymbol{x}_{r^*}},$$

i.e., $F(\mathcal{A}, \mathcal{S})^{-1}$ is computed in $\mathcal{O}(d^2)$ time via matrix-vector multiplications, using the inverse from the previous iteration. This yields an overall complexity of $O(N^2Kd^2)$ for Alg. 1.

Fisher Information. Objective f_4 is not submodular. Several approximations that lead to a submodular function are proposed in [Hoi *et al.*, 2006] under the assumption that feature vectors are normalized. Applying these approximations to our generative model, we get the following objective:

$$\hat{f}_4(\mathcal{S}) = -\sum_{q\notin\mathcal{S}, q\in\mathcal{C}} \frac{\pi(\boldsymbol{x}_q)(1-\pi(\boldsymbol{x}_q))}{\sum_{q'\in\mathcal{S}} \pi(\boldsymbol{x}_{q'})(1-\pi(\boldsymbol{x}_{q'}))(\boldsymbol{x}_q^T \boldsymbol{x}_{q'})^2}, \tag{14}$$

Note that, as $\hat{f}_4(\emptyset) = -\infty$, the greedy algorithm does *not* necessarily attain the approximation guarantee of Theorem 3.1. Nevertheless, we still use Alg. 1 to optimize this objective in our evaluations. Reusing computations as in the case of covariance, the greedy algorithm attains a complexity $O(N^4d + N^2K)$.

5 Evaluation

We use both synthetic and real datasets to evaluate the performance of our experimental design algorithms.

5.1 Datasets

Synthetic Dataset. In our synthetic dataset, $N = 110$ absolute feature vectors $\boldsymbol{x}_i \in \mathbb{R}^d$, $i \in \mathcal{N}$, are sampled from a Gaussian distribution $\mathcal{N}(\mathbf{0}, \sigma_x I_d)$. We also sample a parameter vector $\widetilde{\boldsymbol{\beta}}$ from Gaussian distribution $\mathcal{N}(\mathbf{0}, \sigma_{\boldsymbol{\beta}} I_d)$. We set $\sigma_x = \sigma_{\boldsymbol{\beta}} = 0.8$ in all our experiments. We generate absolute labels y_i, $i \in \mathcal{N}$, using Eq. (1) with $\boldsymbol{\beta} = \widetilde{\boldsymbol{\beta}}/C_a$, where C_a is a positive scalar. Finally, we generate $|\mathcal{C}| = 5995$ comparison labels via Eq. (2), with $\boldsymbol{\beta} = \widetilde{\boldsymbol{\beta}}$. Adjusting parameter C_a allows us to change the variance of absolute labels, and to assess the effect of different noise levels between absolute and comparison labels.

Real Datasets. We use two real-life datasets to verify our algorithms, ROP and SUSHI.

ROP Dataset. Our first dataset consists of 100 images of retinas, labeled by experts w.r.t. the presence of a disease called Retinopathy of Prematurity (ROP) [Hartnett and Penn, 2012]. We represent each image through a vector $\boldsymbol{x}_i \in \mathbb{R}^d$ where $d = 156$, using the feature extraction procedure of [Kalpathy-Cramer *et al.*, 2016], comprising statistics of several indices such as blood vessel curvature, dilation, and tortuosity. Five experts provide diagnostic labels for all 100 images, categorizing them as Plus, Preplus and Normal. We convert these to absolute labels $y_i \in \{-1, +1\}$ by mapping Plus and Preplus as $+1$ and Normal to -1. Finally, these five experts also provide $|\mathcal{C}| = 29705$ comparison labels for 4950 pairs of images in this dataset[1].

It is known that diagnostic labels exhibit a higher variance than comparison labels in this dataset [Kalpathy-Cramer *et al.*, 2016]. Beyond these labels, we also have Reference Standard Diagnosis (RSD) labels for each of these images, which are created via a consensus reached by a committee of 3 experts [Kalpathy-Cramer *et al.*, 2016]. We use these additional labels for testing purposes, as described below in Section 5.3.

SUSHI Dataset. The SUSHI Preference dataset [Kamishima *et al.*, 2009] consists of rankings of $N = 100$ sushi food items by 5000 customers. Each customer ranks 10 items according to her preferences. Each sushi item is associated with a feature vector $\boldsymbol{x}_i \in \mathbb{R}^d$ where $d = 20$, consisting of features such as style, group, heaviness/oiliness in taste, frequency, and normalized price.

We generate comparison labels as follows. For any pair of items $i, j \in \mathcal{N}$ in a customer's ranked list, if i precedes j in the list, we set $y_{i,j} = +1$, otherwise, $y_{i,j} = -1$. We also produce absolute labels via the Elo ranking algorithm [Elo, 1978].

[1] Some experts observe and rate the same pair more than once.

This gives us an individual score for each item; we convert the individual score to an absolute label $y_i \in \{-1, +1\}$ by setting items above (below) the median score to $+1$ (-1).

5.2 Algorithms

We select comparison labels by optimizing, via Alg. 1, the Mutual Information (MI), Covariance (Cov), Information Entropy (Ent), and Fisher Information Method (Fisher) objectives, as described in Section 3.5. We also implement a Random (Ran) strategy, whereby set $\mathcal{S}$ is selected uniformly at random from $\mathcal{C}$. We repeat each execution of the random strategy 10 times, and report the average performance.

Each of the four algorithms listed above have a hyperparameter that needs to be tuned: σ_0 for MI, c for Cov, λ_e for Ent, and λ_f for Fisher. We tune these parameters on a validation set, as described in Section 5.3. We run all algorithm with K ranging from 0 to 100, with the exception of MI, that is the most computation intensive: we execute this for $K = 0$ to 15. We make our code publicly available.[2]

5.3 Experimental Setting

In each experiment, we partition the dataset $\mathcal{N}$ into three datasets: a training set $\mathcal{N}_{\texttt{trn}}$, a test set $\mathcal{N}_{\texttt{tst}}$, and a validation set $\mathcal{N}_{\texttt{val}}$. We denote by $\mathcal{C}_{\texttt{trn}}$, $\mathcal{C}_{\texttt{tst}}$, and $\mathcal{C}_{\texttt{val}}$ the corresponding subsets of $\mathcal{C}$, restricted to pairs of objects in $\mathcal{N}_{\texttt{trn}}$, $\mathcal{N}_{\texttt{tst}}$, and $\mathcal{N}_{\texttt{val}}$, respectively. We ignore comparisons across items in different sets; as a result $\mathcal{C}_{\texttt{trn}} \cup \mathcal{C}_{\texttt{tst}} \cup \mathcal{C}_{\texttt{val}} \subset \mathcal{C}$. To evaluate an experimental design algorithm, we select a random subset $\mathcal{A}$ from $\mathcal{N}_{\texttt{trn}}$. We then execute the design algorithm to select K comparison labels $y_{\mathcal{S}}, \mathcal{S} \subset \mathcal{C}_{\texttt{trn}}$, where $|\mathcal{S}| = K$. We train a model $\boldsymbol{\beta} \in \mathbb{R}^d$ using the labels in $\mathcal{A}$ and $\mathcal{S}$ via MAP estimation (4). We test the performance of the trained model in terms of label prediction on both the test set and the validation set with AUC of both absolute and comparison labels. For ROP, we also measure the performance of RSD label prediction.

For each dataset, we perform 3-fold cross validation, repeating the partition to training and test datasets keeping the validation set fixed. Each 3-fold cross validation is repeated 30 times, i.e., over 30 different random data shuffles. We set each algorithm's hyperparameters as well as λ to be the values that maximize the average AUC on the validation sets. In our result below, we report AUC on test sets.

5.4 Evaluation on Synthetic Data

In the synthetic dataset, the experimenter initially has access to $|\mathcal{A}| = 20$ absolute labels, where $\mathcal{A} \subset \mathcal{N}_{\texttt{trn}}$. These are augmented from $\mathcal{C}_{\texttt{trn}}$ via one of our candidate selection algorithms. Fig. 5.4 shows the average AUC on the test set, assuming that $d = 20$ and that absolute labels have variance 0.65 achieved by setting $C_a = 2$. Fig. 5.4 shows the same result for a lower variance $0.41 (C_a = 1)$. We observe that (a) (less noisy) comparison labels are easier to predict, and that (b) decreasing the noise increases AUC across the board, while also leading to larger differentiation between different methods. MI, which is most computationally intensive, outperforms other methods, followed closely by Cov, while Ran is the worst across all regimes.

[2] https://github.com/neu-spiral/Experimental_Design

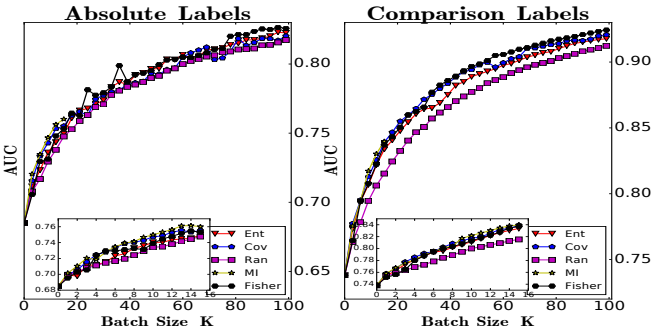

Figure 1: Average AUC for the synthetic data with absolute label variance 0.65, for absolute labels and comparison label prediction. The inset focuses on $K \leq 15$.

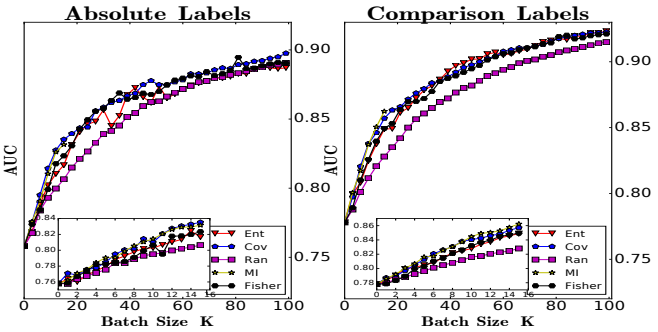

Figure 2: Average AUC for the synthetic data with absolute label variance 0.41, the left one is the classification result for absolute labels, the right one is the classification for comparison labels. The inset focuses on $K \leq 15$.

Table 1: Synthetic Data Classification AUC under different noise factors C_a, for $d = 20$ (Comparison Label Prediction).

	Var(C_a)	Model	K=15	K=30	K=50	K=100
Comparison Label Classification	0.41(1)	Ran	0.828 ± 0.006	0.861 ± 0.005	0.884 ± 0.004	0.916 ± 0.003
		Ent	0.849 ± 0.006	0.878 ± 0.005	0.902 ± 0.004	0.923 ± 0.003
		Cov	0.857 ± 0.005	0.879 ± 0.005	0.899 ± 0.004	0.923 ± 0.003
		Fisher	0.849 ± 0.006	0.872 ± 0.005	0.897 ± 0.004	0.921 ± 0.003
		MI	0.862 ± 0.005			
	0.55(1.5)	Ran	0.814 ± 0.008	0.848 ± 0.007	0.877 ± 0.005	0.909 ± 0.004
		Ent	0.817 ± 0.008	0.86 ± 0.006	0.884 ± 0.005	0.908 ± 0.004
		Cov	0.834 ± 0.007	0.868 ± 0.006	0.895 ± 0.005	0.919 ± 0.004
		Fisher	0.831 ± 0.008	0.868 ± 0.007	0.891 ± 0.005	0.916 ± 0.004
		MI	0.841 ± 0.008			
	0.65(2)	Ran	0.815 ± 0.007	0.851 ± 0.006	0.88 ± 0.005	0.912 ± 0.004
		Ent	0.834 ± 0.007	0.864 ± 0.007	0.891 ± 0.005	0.918 ± 0.003
		Cov	0.839 ± 0.006	0.872 ± 0.005	0.897 ± 0.004	0.921 ± 0.003
		Fisher	0.837 ± 0.006	0.866 ± 0.006	0.897 ± 0.005	0.925 ± 0.003
		MI	0.84 ± 0.007			
	0.72(2.5)	Ran	0.789 ± 0.008	0.833 ± 0.006	0.865 ± 0.005	0.905 ± 0.004
		Ent	0.8 ± 0.008	0.843 ± 0.006	0.872 ± 0.005	0.905 ± 0.004
		Cov	0.817 ± 0.007	0.852 ± 0.006	0.879 ± 0.005	0.909 ± 0.004
		Fisher	0.807 ± 0.007	0.853 ± 0.006	0.882 ± 0.004	0.914 ± 0.003
		MI	0.811 ± 0.007			
	0.78(3)	Ran	0.791 ± 0.008	0.839 ± 0.006	0.877 ± 0.005	0.917 ± 0.003
		Ent	0.797 ± 0.008	0.844 ± 0.007	0.883 ± 0.005	0.911 ± 0.004
		Cov	0.809 ± 0.007	0.856 ± 0.006	0.887 ± 0.005	0.917 ± 0.004
		Fisher	0.804 ± 0.008	0.858 ± 0.007	0.886 ± 0.005	0.923 ± 0.003
		MI	0.818 ± 0.008			

In Tables 1 and 2, we illustrate how the AUC of comparison prediction on the test set is affected by an increase of noise in the absolute labels, for different values of K. We observe that adding comparison labels improves prediction compared to absolute labels alone, and the benefit becomes more pronounced in the high-noise regime. We again observe starker differentiation from Ran in the low-noise regime.

We also illustrate how the AUC of comparison prediction on

Table 2: Synthetic Data Classification AUC under different noise factor C_a, for $d = 20$ (Absolute Label Prediction).

	Var(C_a)	Model	K=15	K=30	K=50	K=100
Absolute Label Classification	0.41(1)	Ran	0.806 ± 0.01	0.839 ± 0.008	0.863 ± 0.008	0.891 ± 0.007
		Ent	0.817 ± 0.009	0.857 ± 0.008	0.863 ± 0.008	0.887 ± 0.007
		Cov	0.835 ± 0.009	0.858 ± 0.008	0.88 ± 0.007	0.894 ± 0.006
		Fisher	0.823 ± 0.009	0.858 ± 0.008	0.87 ± 0.007	0.89 ± 0.007
		MI	0.832 ± 0.009			
	0.55(1.5)	Ran	0.762 ± 0.011	0.791 ± 0.01	0.814 ± 0.01	0.842 ± 0.009
		Ent	0.762 ± 0.012	0.793 ± 0.01	0.813 ± 0.01	0.832 ± 0.009
		Cov	0.78 ± 0.011	0.812 ± 0.009	0.831 ± 0.009	0.85 ± 0.008
		Fisher	0.777 ± 0.011	0.806 ± 0.01	0.816 ± 0.008	0.834 ± 0.009
		MI	0.788 ± 0.011			
	0.65(2)	Ran	0.748 ± 0.011	0.771 ± 0.011	0.79 ± 0.01	0.817 ± 0.009
		Ent	0.75 ± 0.011	0.774 ± 0.01	0.795 ± 0.01	0.822 ± 0.009
		Cov	0.755 ± 0.011	0.777 ± 0.01	0.792 ± 0.009	0.818 ± 0.009
		Fisher	0.753 ± 0.011	0.779 ± 0.011	0.797 ± 0.01	0.826 ± 0.009
		MI	0.76 ± 0.011			
	0.72(2.5)	Ran	0.706 ± 0.011	0.73 ± 0.01	0.746 ± 0.01	0.768 ± 0.01
		Ent	0.712 ± 0.01	0.741 ± 0.011	0.761 ± 0.01	0.768 ± 0.01
		Cov	0.719 ± 0.01	0.735 ± 0.009	0.754 ± 0.009	0.767 ± 0.01
		Fisher	0.714 ± 0.011	0.733 ± 0.01	0.761 ± 0.01	0.774 ± 0.01
		MI	0.718 ± 0.01			
	0.78(3)	Ran	0.686 ± 0.012	0.719 ± 0.011	0.74 ± 0.011	0.766 ± 0.01
		Ent	0.686 ± 0.011	0.723 ± 0.01	0.738 ± 0.01	0.764 ± 0.01
		Cov	0.703 ± 0.012	0.724 ± 0.011	0.746 ± 0.01	0.757 ± 0.01
		Fisher	0.698 ± 0.011	0.727 ± 0.011	0.755 ± 0.01	0.775 ± 0.009
		MI	0.696 ± 0.011			

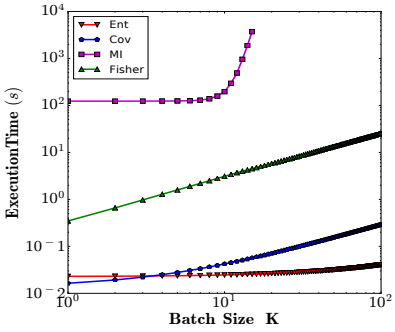

Figure 3: Time to compute S_K as a function of K.

the test set is affected by an increase of feature dimensionality in Tables 3 and 4 in Appendix C. In Fig. 3, we plot the execution time to compute $\mathcal{S}_K$ of different algorithms as a function of K. As expected, Ent is the fastest, with Covariance trailing as second. The slope of the log-log plot indicates that, with the exception of MI that grows exponentially, the remaining algorithms grow almost linearly with K.

5.5 Evaluation on Real Datasets

We repeat the above process in the ROP dataset, reporting the AUC for Reference Standard Diagnosis (RSD) in $\mathcal{N}_{\texttt{tst}}$ and comparison labels in $\mathcal{C}_{\texttt{tst}}$. Fig. 4 shows the corresponding AUC's, as a function of $|\mathcal{S}| = K$, when the experimenter augments $|\mathcal{A}| = 40$ initial absolute labels from $\mathcal{N}_{\texttt{trn}}$. We observe a clear differentiation between policies. Using 100 comparison labels, we can improve AUC by as much as 3.4% (3.8%) for RSD (comparison) prediction by using the Cov (Fisher) method. MI outperforms other methods within the range that we can execute it. Ent significantly underperforms random, especially in RSD label prediction. We repeat the same experiment in the Sushi dataset, giving the experimenter access to $|\mathcal{A}| = 20$ initial labels. We again observe that MI outperforms other methods in the $K = 0$ to 15 range, while the covariance method performs well across the board. The remaining two methods are only marginally better than random in comparison prediction, but not on absolute label prediction.

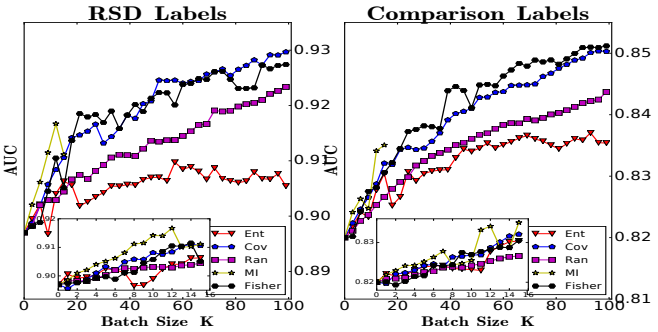

Figure 4: Average AUC for ROP data as a function of K with $|\mathcal{A}| = 40$ initial absolute labels. The inset focuses on $K \leq 15$.

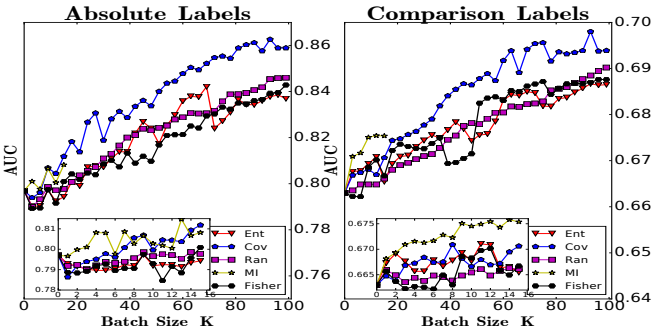

Figure 5: Average AUC for Sushi dataset as a function of K with $|\mathcal{A}| = 20$ initial absolute labels. The inset focuses on $K \leq 15$.

6 Conclusion

We present several methods for collecting comparison labels with the purpose of augmenting a dataset of absolute labels. Several of the objectives we study in modeling experimental design are submodular, and the optimizations involved can be attacked through a greedy algorithm.

We observed in both synthetic and real-life datasets that Mutual Information almost universally outperforms other objectives. On the other hand, computing it is computationally expensive, even when the posterior of the model is approximated with variational inference. Hence, identifying tractable approximations that still yield good estimates is an important open problem. The covariance method performs well–if not optimally in comparison to MI–in terms of prediction, striking a good balance between efficiency and prediction quality. Both for this and alternative methods, greedy selection can grow quadratically in the dataset size N, because of the quadratic nature of set $\mathcal{C}$. Exploiting the underlying geometry of $\mathcal{C}$ to produce sub-quadratic algorithms is also an important open research direction.

7 Acknowledgement

Our work is supported by NIH (R01EY019474, P30EY10572), NSF (SCH-1622542 at MGH; SCH-1622536 at Northeastern; SCH-1622679 at OHSU), and by unrestricted departmental funding from Research to Prevent Blindness (OHSU).

References

[Boyd and Vandenberghe, 2004] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[Bradley and Terry, 1952] R. A. Bradley and M. E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 1952.

[Busetto *et al.*, 2013] A. G. Busetto, A. Hauser, G. Krummenacher, M. Sunnåker, S. Dimopoulos, C. S. Ong, Jö. Stelling, and J. M Buhmann. Near-optimal experimental design for model selection in systems biology. *Bioinformatics*, 2013.

[Cavagnaro *et al.*, 2010] D. R. Cavagnaro, J. I. Myung, M. A. Pitt, and J. V. Kujala. Adaptive design optimization: A mutual information-based approach to model discrimination in cognitive science. *Neural computation*, 2010.

[Chaloner and Verdinelli, 1995] K. Chaloner and I. Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 1995.

[Chen *et al.*, 2015] L. Chen, P. Zhang, and B. Li. Fusing pointwise and pairwise labels for supporting user-adaptive image retrieval. In *ICMR*, 2015.

[Cohn *et al.*, 1996] D. A. Cohn, Z. Ghahramani, and M. I. Jordan. Active learning with statistical models. *JAIR*, 1996.

[Delzell *et al.*, 2012] D. A. Delzell, R. F. Gunst, et al. Key properties of d-optimal designs for event-related functional mri experiments with application to nonlinear models. *Statistics in medicine*, 2012.

[Drovandi and Pettitt, 2013] C. C. Drovandi and A. N. Pettitt. Bayesian experimental design for models with intractable likelihoods. *Biometrics*, 2013.

[Drovandi *et al.*, 2014] C.C. Drovandi, J.M. McGree, and A.N. Pettitt. A sequential monte carlo algorithm to incorporate model uncertainty in bayesian sequential design. *J. C. Graph. Stat*, 2014.

[Elo, 1978] A. E. Elo. *The rating of chessplayers, past and present*. Arco Pub., 1978.

[Flaherty *et al.*, 2006] P. Flaherty, A. Arkin, and M. I. Jordan. Robust design of biological experiments. In *NIPS*, 2006.

[Glickman and Jensen, 2005] Mark E Glickman and Shane T Jensen. Adaptive paired comparison design. *Journal of statistical planning and inference*, 2005.

[Graßhoff and Schwabe, 2008] Ulrike Graßhoff and Rainer Schwabe. Optimal design for the bradley–terry paired comparison model. *Statistical Methods and Applications*, 2008.

[Grasshoff *et al.*, 2003] Ulrike Grasshoff, Heiko Großmann, Heinz Holling, and Rainer Schwabe. Optimal paired comparison designs for first-order interactions. *Statistics*, 2003.

[Guillory and Bilmes, 2011] A. Guillory and J. Bilmes. Active semi-supervised learning using submodular functions. In *UAI*, 2011.

[Hartnett and Penn, 2012] M. E. Hartnett and J. S. Penn. Mechanisms and management of retinopathy of prematurity. *New England Journal of Medicine*, 2012.

[Harville, 1997] D. A. Harville. *Matrix algebra from a statistician's perspective*, volume 1. Springer, 1997.

[Hoi *et al.*, 2006] S. C. Hoi, R. Jin, J. Zhu, and M. R. Lyu. Batch mode active learning and its application to medical image classification. In *ICML*, 2006.

[Horel *et al.*, 2014] T. Horel, S. Ioannidis, and S. Muthukrishnan. Budget feasible mechanisms for experimental design. In *Latin American Symposium on Theoretical Informatics*, 2014.

[Huan and Marzouk, 2013] X. Huan and Y. M. Marzouk. Simulation based optimal bayesian experimental design for nonlinear systems. *Journal of Computational Physics*, 2013.

[Jordan *et al.*, 1999] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine learning*, 1999.

[Kalpathy-Cramer *et al.*, 2016] J. Kalpathy-Cramer, J. P. Campbell, D. Erdogmus, et al. Plus disease in retinopathy of prematurity: improving diagnosis by ranking disease severity and using quantitative image analysis. *Ophthalmology*, 2016.

[Kamishima *et al.*, 2009] T. Kamishima, M. Hamasaki, and S. Akaho. A simple transfer learning method and its application to personalization in collaborative tagging. In *ICDM*, 2009.

[Krause and Guestrin, 2005] A. Krause and C. E. Guestrin. Near-optimal nonmyopic value of information in graphical models. In *UAI*, 2005.

[Lewis and Gale, 1994] D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers. In *SIGIR*, 1994.

[Liepe *et al.*, 2013] J. Liepe, S. Filippi, M. Komorowski, and M. P. Stumpf. Maximizing the information content of experiments in systems biology. *PLoS computational biology*, 2013.

[Myerson, 1981] R. B. Myerson. Optimal auction design. *Mathematics of operations research*, 1981.

[Nemhauser *et al.*, 1978] G. L. Nemhauser, L. A. Wolsey, and M. L Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 1978.

[Rao, 1992] C Radhakrishna Rao. Information and the accuracy attainable in the estimation of statistical parameters. In *Breakthroughs in statistics*. Springer, 1992.

[Robert, 2014] C. Robert. Machine learning, a probabilistic perspective, 2014.

[Ryan, 2003] K. J. Ryan. Estimating expected information gains for experimental designs with application to the random fatigue-limit model. *J. Comput. Graph. Stat*, 2003.

[Sculley, 2010] D. Sculley. Combined regression and ranking. In *SIGKDD*, 2010.

[Sharma and Bilgic, 2017] M. Sharma and M. Bilgic. Evidence-based uncertainty sampling for active learning. *DMKDFD*, 2017.

[Takahama *et al.*, 2016] R. Takahama, T. Kamishima, and H. Kashima. Progressive comparison for ranking estimation. In *IJCAI*, 2016.

[Tsilifis *et al.*, 2017] P. Tsilifis, R. G. Ghanem, and P. Hajali. Efficient bayesian experimentation using an expected information gain lower bound. *SIAM/ASA JUQ*, 2017.

[Wallace *et al.*, 2008] D. K. Wallace, G. E. Quinn, S. F. Freedman, and M. F. Chiang. Agreement among pediatric ophthalmologists in diagnosing plus and pre-plus disease in retinopathy of prematurity. *JAAPOS*, 2008.

[Wang *et al.*, 2016] Y. Wang, S. Wang, J. Tang, H. Liu, and B. Li. PPP: Joint pointwise and pairwise image label prediction. In *CVPR*, 2016.

[Wei *et al.*, 2015] Kai Wei, R. Iyer, and J. Bilmes. Submodularity in data subset selection and active learning. In *ICML*, 2015.

[Zhang and Oles, 2000] T. Zhang and F. Oles. The value of unlabeled data for classification problems. In *ICML*, 2000.

Appendix

A Submodularity of Mutual Information Objective Function

Proof. For every set $\mathcal{T}$ and $\mathcal{D}$ with $\mathcal{T} \subseteq \mathcal{D}$ and every $x \notin \mathcal{D}$, because the conditioning can't increase entropy, we have the following inequality:

$$\mathbf{H}(Y_x|Y_{\mathcal{D}}, y_{\mathcal{A}}) = \mathbf{H}(Y_x|Y_{\mathcal{T}}, Y_{\mathcal{D}-\mathcal{T}}, y_{\mathcal{A}}) \leqslant \mathbf{H}(Y_x|Y_{\mathcal{T}}, y_{\mathcal{A}}). \tag{15}$$

Given the parameter vector $\boldsymbol{\beta}$, the labels are independent, hence:

$$\begin{aligned}\mathcal{E}_\beta \equiv &\mathbf{H}(Y_x, Y_{\mathcal{T}}|\boldsymbol{\beta}, y_{\mathcal{A}}) - \mathbf{H}(Y_{\mathcal{T}}|\boldsymbol{\beta}, y_{\mathcal{A}}) - \mathbf{H}(Y_x, Y_{\mathcal{D}}|\boldsymbol{\beta}, y_{\mathcal{A}}) \\ &+ \mathbf{H}(Y_{\mathcal{D}}|\boldsymbol{\beta}, y_{\mathcal{A}}) = 0.\end{aligned} \tag{16}$$

Combining Eq. 15 and Eq. 16, we get

$$\begin{aligned}&f_1(\mathcal{T} \cup \{x\}) - f_1(\mathcal{T}) - f_1(\mathcal{D} \cup \{x\}) + f_1(\mathcal{D}) \\ =&\mathbf{H}(Y_x, Y_{\mathcal{T}}|y_{\mathcal{A}}) - \mathbf{H}(Y_x, Y_{\mathcal{T}}|\boldsymbol{\beta}, y_{\mathcal{A}}) - \mathbf{H}(Y_{\mathcal{T}}|y_{\mathcal{A}}) + \mathbf{H}(Y_{\mathcal{T}}|\boldsymbol{\beta}, y_{\mathcal{A}}) \\ &- \mathbf{H}(Y_x, Y_{\mathcal{D}}|y_{\mathcal{A}}) + \mathbf{H}(Y_x, Y_{\mathcal{D}}|\boldsymbol{\beta}, y_{\mathcal{A}}) + \mathbf{H}(Y_{\mathcal{D}}|y_{\mathcal{A}}) - \mathbf{H}(Y_{\mathcal{D}}|\boldsymbol{\beta}, y_{\mathcal{A}}) \\ =&\mathbf{H}(Y_x|Y_{\mathcal{T}}, y_{\mathcal{A}}) - \mathbf{H}(Y_x|Y_{\mathcal{D}}, y_{\mathcal{A}}) - \mathcal{E}_\beta \\ =&\mathbf{H}(Y_x|Y_{\mathcal{T}}, y_{\mathcal{A}}) - \mathbf{H}(Y_x|Y_{\mathcal{D}}, y_{\mathcal{A}}) \geqslant 0.\end{aligned}$$

This implies submodularity. Moreover, because conditioning can not increase entropy,

$$\begin{aligned}f_1(\mathcal{S} \cup x) - f_1(\mathcal{S}) =&\mathbf{H}(Y_x|Y_{\mathcal{S}}, Y_{\mathcal{A}}{=}y_{\mathcal{A}}) - \mathbf{H}(Y_x|Y_{\mathcal{S}}, \boldsymbol{\beta}, Y_{\mathcal{A}}{=}y_{\mathcal{A}}) \\ \geqslant&0\end{aligned}$$

Objective function $f_1(\mathcal{S})$ is monotone. □

B Variational Inference

For completeness, we describe here how to use variational inference to approximate the posterior $q(\boldsymbol{\beta})$ in eq.(11) as [Jordan *et al.*, 1999].

If the parameter vector satisfies the prior distribution $\boldsymbol{\beta} \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$. We assume that the posterior distribution satisfies:

$$q(\boldsymbol{\beta}) = \mathcal{N}(\boldsymbol{\mu}_{\mathcal{A}}, \boldsymbol{\Sigma}_{\mathcal{A}}).$$

Here, this equation can be written as:

$$\boldsymbol{\mu}_{\mathcal{A}} = \boldsymbol{\Sigma}_{\mathcal{A}}(\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0 + \sum_{t=1}^{N_{\mathcal{A}}} y_t \boldsymbol{x_t}/2),$$

$$\boldsymbol{\Sigma}_{\mathcal{A}}^{-1} = \boldsymbol{\Sigma}_0^{-1} + 2\sum_{t=1}^{N_{\mathcal{A}}} \lambda(\xi_t)\boldsymbol{x}_t\boldsymbol{x}_t^T,$$

$$\lambda(\xi_t) = \frac{tanh(\xi_t/2)}{4\xi_t},$$

where ξ_t is the variational parameter. We can use EM method to get the Variational Parameters:

At first, we give initial ξ_t, then,

E-Step:

$$\boldsymbol{\mu}_{\mathcal{A}} = \boldsymbol{\Sigma}_{\mathcal{A}}(\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0 + \sum_{t=1}^{N_{\mathcal{A}}} y_t \boldsymbol{x}_t/2)$$

$$\boldsymbol{\Sigma}_{\mathcal{A}}^{-1} = \boldsymbol{\Sigma}_0^{-1} + \sum_{t=1}^{N_{\mathcal{A}}} \lambda(\xi_t^{old})\boldsymbol{x}_t\boldsymbol{x}_t^T$$

$$\lambda(\xi_t^{old}) = \frac{e^{\xi_t^{old}} - 1}{4\xi_t^{old}(e^{\xi_t^{old}} + 1)}$$

M-Step(Re-estimate ξ_t):

$$(\xi_t^{new})^2 = \boldsymbol{x}_t^T(\boldsymbol{\Sigma}_{\mathcal{A}} + \boldsymbol{\mu}_{\mathcal{A}}\boldsymbol{\mu}_{\mathcal{A}}^T)\boldsymbol{x}_t$$

C AUC Tables

In Tables 3 and 4 we indicate now the AUC for comparison prediction and absolute prediction, respectively changes for different values of the dimension d and the set size K.

Table 3: Synthetic Data Classification AUC under different dimensions ($C_a = 2$), (Comparison Label Prediction)

	Dimension	Model	K=0	K=15	K=30	K=50	K=100
Comparison Label Classification	10	Ran	**0.695**	0.811	0.848	0.874	0.898
		Ent	**0.695**	0.817	0.854	0.875	0.891
		Cov	**0.695**	0.839	**0.873**	**0.89**	**0.907**
		Fisher	**0.695**	0.836	0.867	0.879	0.9
		MI	**0.695**	**0.844**			
	15	Ran	**0.720**	0.809	0.849	0.88	0.908
		Ent	**0.720**	0.817	0.859	0.887	0.911
		Cov	**0.720**	**0.836**	**0.875**	0.893	**0.918**
		Fisher	**0.720**	0.825	0.869	**0.894**	0.915
		MI	**0.720**	**0.836**			
	20	Ran	**0.738**	0.815	0.851	0.88	0.912
		Ent	**0.738**	0.834	0.864	0.891	0.918
		Cov	**0.738**	0.839	**0.872**	**0.897**	0.921
		Fisher	**0.738**	0.837	0.866	**0.897**	**0.925**
		MI	**0.738**	**0.84**			
	25	Ran	**0.708**	0.792	0.833	0.865	0.907
		Ent	**0.708**	0.8	0.845	0.873	0.909
		Cov	**0.708**	**0.813**	**0.853**	**0.883**	**0.914**
		Fisher	**0.708**	0.803	0.846	0.879	0.913
		MI	**0.708**	**0.813**			
	30	Ran	**0.693**	0.766	0.81	0.844	0.891
		Ent	**0.693**	0.778	0.818	0.855	0.898
		Cov	**0.693**	0.782	**0.824**	0.859	0.892
		Fisher	**0.693**	0.782	**0.824**	**0.867**	**0.905**
		MI	**0.693**	**0.784**			

Table 4: Synthetic Data Classification AUC under different dimensions ($C_a = 2$)(Absolute Label Prediction)

	Feature Dim	Model	K=0,	K=15,	K=30,	K=50,	K=100
Absolute Label Classification	10	Ran	**0.621**	0.697	0.723	0.735	0.75
		Ent	**0.621**	0.708	0.722	0.742	0.744
		Cov	**0.621**	**0.717**	**0.736**	**0.744**	**0.759**
		Fisher	**0.621**	0.713	0.729	0.736	0.749
		MI	**0.621**	0.713			
	15	Ran	**0.668**	0.723	0.752	0.771	0.788
		Ent	**0.668**	0.734	0.762	0.772	0.79
		Cov	**0.668**	0.737	0.762	**0.777**	**0.795**
		Fisher	**0.668**	0.732	**0.763**	**0.777**	0.781
		MI	**0.668**	**0.738**			
	20	Ran	**0.684**	0.748	0.771	0.79	0.817
		Ent	**0.684**	0.75	0.774	0.795	0.822
		Cov	**0.684**	0.755	0.777	0.792	0.818
		Fisher	**0.684**	0.753	**0.779**	**0.797**	**0.826**
		MI	**0.684**	**0.76**			
	25	Ran	**0.666**	0.735	0.769	0.793	0.824
		Ent	**0.666**	0.732	0.777	0.797	0.822
		Cov	**0.666**	**0.758**	**0.79**	**0.809**	**0.834**
		Fisher	**0.666**	0.735	0.778	0.797	0.823
		MI	**0.666**	0.755			
	30	Ran	**0.653**	0.706	0.742	0.773	0.805
		Ent	**0.653**	0.718	**0.756**	**0.786**	0.814
		Cov	**0.653**	0.721	0.75	0.779	0.812
		Fisher	**0.653**	0.715	0.753	0.784	**0.817**
		MI	**0.653**	**0.726**			