A Proofs

We present proof for all propositions made in the paper, restating each for convenience. We also include additional discussion on technical aspects of the paper.

A.1 Unified objective for non-exchangeable experiments

Proposition 1 (Generalized total expected information gain)**.** *Consider the data generating distribution $p(h_T|\theta,\pi) = \prod_{t=1:T} p(y_t|\theta,\xi_t,h_{t-1})$, where $\xi_t = \pi(h_{t-1})$ are the designs generated by the policy and, unlike in (4), y_t is allowed to depend on the history h_{t-1}. Then we can write (3) as*

$$\mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log p(h_T|\theta,\pi)\right] - \mathbb{E}_{p(h_T|\pi)}\left[\log p(h_T|\pi)\right]. \tag{6}$$

Proof. Starting with the definition of the total EIG (3) of a policy π:

$$\mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\sum\nolimits_{t=1}^{T} I_{h_{t-1}}(\xi_t)\right] \tag{11}$$

we have by linearity of expectation

$$= \sum\nolimits_{t=1}^{T} \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[I_{h_{t-1}}(\xi_t)\right] \tag{12}$$

and since $I_{h_{t-1}}$ doesn't depend on data acquired after $t-1$ (the future doesn't influence the past)

$$= \sum\nolimits_{t=1}^{T} \mathbb{E}_{p(\theta)p(h_{t-1}|\theta,\pi)}\left[I_{h_{t-1}}(\xi_t)\right] \tag{13}$$

which, applying Bayes rule, is equivalent to

$$= \sum\nolimits_{t=1}^{T} \mathbb{E}_{p(h_{t-1}|\pi)p(\theta|h_{t-1})}\left[I_{h_{t-1}}(\xi_t)\right] \tag{14}$$

Next, using Bayes rule we similarly rearrange $I_{h_{t-1}}$:

$$I_{h_{t-1}}(\xi_t) = \mathbb{E}_{p(\theta|h_{t-1})p(y_t|\theta,\xi_t,h_{t-1})}\left[\log \frac{p(y_t|\theta,\xi_t,h_{t-1})}{p(y_t|\xi_t,h_{t-1})}\right] \tag{15}$$

$$= \mathbb{E}_{p(\theta|h_{t-1})p(y_t|\theta,\xi_t,h_{t-1})}\left[\log \frac{p(\theta|y_t,\xi_t,h_{t-1})}{p(\theta|h_{t-1})}\right] \tag{16}$$

$$= \mathbb{E}_{p(\theta|h_{t-1})p(y_t|\theta,\xi_t,h_{t-1})}\left[\log p(\theta|y_t,\xi_t,h_{t-1})\right] - \mathbb{E}_{p(\theta|h_{t-1})}\left[\log p(\theta|h_{t-1})\right] \tag{17}$$

$$= \mathbb{E}_{p(\theta|y_t,\xi_t,h_{t-1})p(y_t|\xi_t,h_{t-1})}\left[\log p(\theta|y_t,\xi_t,h_{t-1})\right] - \mathbb{E}_{p(\theta|h_{t-1})}\left[\log p(\theta|h_{t-1})\right] \tag{18}$$

and noting $h_t = h_{t-1} \cup \{(\xi_t, y_t)\}$

$$= \mathbb{E}_{p(\theta|h_t)p(y_t|\xi_t,h_{t-1})}\left[\log p(\theta|h_t)\right] - \mathbb{E}_{p(\theta|h_{t-1})}\left[\log p(\theta|h_{t-1})\right] \tag{19}$$

$$= \mathbb{E}_{p(y_t|\xi_t,h_{t-1})}\Big[\mathbb{E}_{p(\theta|h_t)}\left[\log p(\theta|h_t)\right] - \mathbb{E}_{p(\theta|h_{t-1})}\left[\log p(\theta|h_{t-1})\right]\Big] \tag{20}$$

Substituting this in (14), noting that θ has already been integrated out, yields

$$\mathcal{I}_T(\pi) = \sum\nolimits_{t=1}^{T} \mathbb{E}_{p(h_{t-1}|\pi)}\mathbb{E}_{p(y_t|\xi_t,h_{t-1})}\Big[\mathbb{E}_{p(\theta|h_t)}\left[\log p(\theta|h_t)\right] - \mathbb{E}_{p(\theta|h_{t-1})}\left[\log p(\theta|h_{t-1})\right]\Big] \tag{21}$$

$$= \sum\nolimits_{t=1}^{T} \mathbb{E}_{p(h_t|\pi)}\Big[\mathbb{E}_{p(\theta|h_t)}\left[\log p(\theta|h_t)\right] - \mathbb{E}_{p(\theta|h_{t-1})}\left[\log p(\theta|h_{t-1})\right]\Big] \tag{22}$$

$$= \mathbb{E}_{p(h_T|\pi)}\left[\sum\nolimits_{t=1}^{T} \mathbb{E}_{p(\theta|h_t)}\left[\log p(\theta|h_t)\right] - \mathbb{E}_{p(\theta|h_{t-1})}\left[\log p(\theta|h_{t-1})\right]\right], \tag{23}$$

since we have a telescopic sum this simplifies to

$$= \mathbb{E}_{p(h_T|\pi)}\Big[\mathbb{E}_{p(\theta|h_T)}\left[\log p(\theta|h_T)\right] - \mathbb{E}_{p(\theta)}\left[\log p(\theta)\right]\Big] \tag{24}$$

and finally we apply Bayes rule again to rewrite as

$$= \mathbb{E}_{p(h_T|\pi)p(\theta|h_T)}\Big[\log p(\theta|h_T) - \mathbb{E}_{p(\theta)}\left[\log p(\theta)\right]\Big] \tag{25}$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log p(\theta|h_T) - \log p(\theta)\right] \tag{26}$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log p(h_T|\theta,\pi) - p(h_T|\pi)\right] \tag{27}$$

□

A.2 Objective function as a mutual information

We provide some additional discussion on the interpretation of $\mathcal{I}_T(\pi)$ in (6) as a mutual information. First, $\mathcal{I}_T(\pi)$ is not a conventional mutual information between θ and h_T. This is because, for the deterministic policy π considered in this paper, the random variable h_T does not have a density with respect to Lebesgue measure on $\Xi^T \times \mathcal{Y}^T$. Indeed, since the designs $\xi_{1:T}$ are deterministic functions of the observations $y_{1:T}$, to express the sampling distribution of h_T we would have to use Dirac deltas, specifically

$$p(y_{1:T}, \xi_{1:T}|\theta, \pi) = \prod_{t=1}^{T} \delta_{\pi(h_{t-1})}(\xi_t) p(y_t|\theta, \xi_t, h_{t-1}). \tag{28}$$

Due to the presence of Dirac deltas, this is not a conventional probability density, and hence we do not regard $\mathcal{I}_T(\pi)$ as the conventional mutual information between θ and h_T.

We note that we *defined* $p(h_T|\theta, \pi)$ in Proposition 1 differently to $p(y_{1:T}, \xi_{1:T}|\theta, \pi)$ in (28). Specifically, our definition

$$p(h_T|\theta, \pi) = \prod_{t=1}^{T} p(y_t|\theta, \xi_t, h_{t-1}) \tag{29}$$

only involves probability densities for $y_{1:T}$, meaning that our $p(h_T|\theta, \pi)$ is a well-defined probability density on $\mathcal{Y}^T$. Formally, we can treat the designs ξ_t, not as additional random variables, but as part of the density for $y_{1:T}$. Indeed, since the policy π is deterministic, it is possible to reconstruct h_{t-1} and ξ_t from $y_{1:t-1}$ and π, so we could write $p(y_t|\theta, y_{1:t-1}, \pi) := p(y_t|\theta, \xi_t, h_{t-1})$. In this formulation, only $y_{1:T}$ are regarded as random variables. This provides a formal justification for the form of $p(h_T|\theta, \pi)$ that we give in Proposition 1. In this setting, we could formally identify $\mathcal{I}_T(\pi)$ as the mutual information between θ and $y_{1:T}$.

However, it is helpful to think of $\mathcal{I}_T(\pi)$ as a mutual information between θ and h_T, because this naturally leads to critics that have access to θ and h_T, rather than θ and $y_{1:T}$. This way of thinking also connects naturally to the case of stochastic policies, which we now discuss.

If we consider additional noise in the design process so that designs are no longer a deterministic function of past data, then $\mathcal{I}_T(\pi)$ is the mutual information between θ and h_T. In this case, we introduce an additional likelihood for designs $p(\xi|\pi, h)$, leading to the overall sampling distribution for the data

$$p(h_T|\theta, \pi) = \prod_{t=1}^{T} p(\xi_t|\pi, h_{t-1}) p(y_t|\theta, \xi_t, h_{t-1}). \tag{30}$$

Unlike in the deterministic case, this is valid probability density on $\Xi^T \times \mathcal{Y}^T$. If we now consider the mutual information between θ and h_T for a fixed policy π we have

$$I(\theta, h_T) = \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log \frac{\prod_{t=1}^{T} p(\xi_t|\pi, h_{t-1}) p(y_t|\theta, \xi_t, h_{t-1})}{\int_\Theta p(\theta) \prod_{t=1}^{T} p(\xi_t|\pi, h_{t-1}) p(y_t|\theta, \xi_t, h_{t-1})\, d\theta}\right] \tag{31}$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log \frac{\prod_{t=1}^{T} p(\xi_t|\pi, h_{t-1}) \prod_{t=1}^{T} p(y_t|\theta, \xi_t, h_{t-1})}{\prod_{t=1}^{T} p(\xi_t|\pi, h_{t-1}) \int_\Theta p(\theta) \prod_{t=1}^{T} p(y_t|\theta, \xi_t, h_{t-1})\, d\theta}\right] \tag{32}$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log \frac{\prod_{t=1}^{T} p(y_t|\theta, \xi_t, h_{t-1})}{\int_\Theta p(\theta) \prod_{t=1}^{T} p(y_t|\theta, \xi_t, h_{t-1})\, d\theta}\right] \tag{33}$$

noticing that the design likelihood terms cancel out in the integrand, and we reduce to the same integrand given in Proposition 1. Even when the policy is stochastic, the integrand in $I(\theta, h_T)$ only involves terms of the form $p(y_t|\theta, \xi_t, h_{t-1})$, and the likelihood of the design process completely cancels. Thus, the stochasticity of the designs is only present in the sampling distribution $p(h_T|\theta, \pi)$. We therefore see that, as we consider the limiting case of $p(\xi|\pi, h)$ as it approaches a deterministic policy, only the sampling distribution of designs in $I(\theta, h_T)$ changes, with the integrand remaining the same. Under mild assumptions, then, the mutual information between θ and h_T approaches $\mathcal{I}_T(\pi)$ in this limit.

A.3 NWJ and InfoNCE bounds

The next two propositions show that the two bounds—NWJ and InfoNCE—can be applied to the policy-based adaptive BOED setting.

Proposition 2 (NWJ bound for implicit policy-based BOED)**.** *For a design policy π and a critic function $U : \mathcal{H}^T \times \Theta \to \mathbb{R}$, let*

$$\mathcal{L}_T^{NWJ}(\pi, U) := \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[U(h_T,\theta)\right] - e^{-1}\mathbb{E}_{p(\theta)p(h_T|\pi)}\left[\exp(U(h_T,\theta))\right], \tag{7}$$

then $\mathcal{I}_T(\pi) \geq \mathcal{L}_T^{NWJ}(\pi, U)$ holds for any U. Further, the inequality is tight for the optimal critic $U^*_{NWJ}(h_T,\theta) = \log p(h_T|\theta,\pi) - \log p(h_T|\pi) + 1$.

Proof. Let $\pi : \mathcal{H}^* \to \Xi$ be any (deterministic) policy taking histories h_t as inputs and returning a design ξ as output, $U : \mathcal{H}^T \times \Theta \to \mathbb{R}$ be any function and define $g(h_T,\theta) := \frac{\exp(U(h_T,\theta))}{\mathbb{E}_{p(h_T|\pi)}[\exp(U(h_T,\theta))]}$.

First, we multiply the numerator and denominator of the unified objective (6) by $g(h_T,\theta) > 0$

$$\mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log\frac{p(h_T|\theta,\pi)}{p(h_T|\pi)}\right] \tag{34}$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\log\left[\frac{p(h_T|\theta,\pi)}{p(h_T|\pi)}\frac{g(h_T,\theta)}{g(h_T,\theta)}\right] \tag{35}$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log g(h_T,\theta)\right] + \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log\frac{p(h_T|\theta,\pi)}{p(h_T|\pi)g(h_T,\theta)}\right] \tag{36}$$

Next, note that the second term is a KL divergence between two distributions

$$\mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log\frac{p(h_T|\theta,\pi)}{p(h_T|\pi)g(h_T,\theta)}\right] = \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log\frac{p(\theta)p(h_T|\theta,\pi)}{p(\theta)p(h_T|\pi)g(h_T,\theta)}\right] \tag{37}$$

$$= KL(p(\theta)p(h_T|\theta,\pi)||\hat{p}(h_T,\theta)) \geq 0 \tag{38}$$

where $\hat{p}(h_T,\theta) = p(\theta)p(h_T|\pi)g(h_T,\theta)$ is a valid distribution since

$$\int p(\theta)p(h_T|\pi)g(h_T,\theta)d\theta dh_T = \mathbb{E}_{p(\theta)p(h_T|\pi)}\frac{\exp(U(h_T,\theta))}{\mathbb{E}_{p(h_T|\pi)}\left[\exp(U(h_T,\theta))\right]} \tag{39}$$

$$= \mathbb{E}_{p(\theta)}1 = 1. \tag{40}$$

Therefore, we have

$$\mathcal{I}_T(\pi) \geq \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log g(h_T,\theta)\right] \tag{41}$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}[U(h_T,\theta) - \log\mathbb{E}_{p(h_T|\pi)}\exp(U(h_T,\theta))] \tag{42}$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}[U(h_T,\theta)] - \mathbb{E}_{p(\theta)}\left[\log\mathbb{E}_{p(h_T|\pi)}\exp(U(h_T,\theta))\right] \tag{43}$$

Now using the inequality $\log x \leq e^{-1}x$

$$\geq \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}[U(h_T,\theta)] - e^{-1}\mathbb{E}_{p(\theta)p(h_T|\pi)}\left[\exp(U(h_T,\theta))\right] \tag{44}$$

$$= \mathcal{L}_T^{NWJ}(\pi, U) \tag{45}$$

Finally, substituting $U^*(h_T,\theta) = \log\frac{p(h_T|\theta,\pi)}{p(h_T|\pi)} + 1$ in the bound we get

$$\mathcal{L}_T^{NWJ}(\pi, U^*) = \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log\frac{p(h_T|\theta,\pi)}{p(h_T|\pi)} + 1\right] - e^{-1}\mathbb{E}_{p(\theta)p(h_T|\pi)}\left[\frac{p(h_T|\theta,\pi)}{p(h_T|\pi)}e^1\right] \tag{46}$$

$$= \mathcal{I}_T(\pi) + 1 - \mathbb{E}_{p(\theta)p(h_T|\pi)}\left[\frac{p(h_T|\theta,\pi)}{p(h_T|\pi)}\right] \tag{47}$$

$$= \mathcal{I}_T(\pi), \tag{48}$$

where we used $\mathbb{E}_{p(\theta)p(h_T|\pi)}\left[\frac{p(h_T|\theta,\pi)}{p(h_T|\pi)}\right] = \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}[1] = 1$, establishing that the bound is tight for the optimal critic.

□

Proposition 3 (InfoNCE bound for implicit policy-based BOED). *Let $\theta_{1:L} \sim p(\theta_{1:L}) = \prod_i p(\theta_i)$ be a set of contrastive samples where $L \geq 1$. For design policy π and critic function $U : \mathcal{H}^T \times \Theta \to \mathbb{R}$, let*

$$\mathcal{L}_T^{NCE}(\pi, U; L) \coloneqq \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \frac{\exp(U(h_T, \theta_0))}{\frac{1}{L+1} \sum_{i=0}^{L} \exp(U(h_T, \theta_i))} \right], \tag{8}$$

then $\mathcal{I}_T(\pi) \geq \mathcal{L}_T^{NCE}(\pi, U; L)$ for any U and $L \geq 1$. Further, the optimal critic, $U^*_{NCE}(h_T, \theta) = \log p(h_T|\theta, \pi) + c(h_T)$ where $c(h_T)$ is any arbitrary function depending only on the history, recovers the sPCE bound in (5); the inequality is tight in the limit as $L \to \infty$ for this optimal critic.

Proof. Let $\pi : \mathcal{H}^* \to \Xi$ be any (deterministic) policy taking histories h_t as inputs and returning a design ξ as output. Choose any function (critic) $U : \mathcal{H}^T \times \Theta \to \mathbb{R}$.

We introduce the shorthand

$$g(h_T, \theta_{0:L}) \coloneqq \frac{\exp(U(h_T, \theta_0))}{\frac{1}{L+1} \sum_{i=0}^{L} \exp(U(h_T, \theta_i))} \tag{49}$$

Starting with the definition of the unified objective from Equation (6) we multiply its numerator and denominator by $g(h_T, \theta_{0:L}) > 0$ to get

$$\mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)} \right] \tag{50}$$

where $p(\theta_0)p(h_T|\theta_0, \pi) \equiv p(\theta)p(h_T|\theta, \pi)$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)} \right] \tag{51}$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi) g(h_T, \theta_{0:L})}{p(h_T|\pi) g(h_T, \theta_{0:L})} \right] \tag{52}$$

We next split the expectation into two terms one of which does not contain the unknown likelihoods and equals $\mathcal{L}^{NCE}$

$$\begin{aligned} &= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi) g(h_T, \theta_{0:L})} \right] \\ &\quad + \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\log g(h_T, \theta_{0:L}) \right] \\ &= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi) g(h_T, \theta_{0:L})} \right] + \mathcal{L}^{NCE}(\pi, U; L) \end{aligned} \tag{53}$$

We now show that the first term is a KL divergence and hence non-negative. To see why, first write

$$\mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi) g(h_T, \theta_{0:L})} \right] \tag{54}$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\log \frac{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})}{p(\theta_0)p(h_T|\pi)p(\theta_{1:L}) g(h_T, \theta_{0:L})} \right] \tag{55}$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\log \frac{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})}{\hat{p}(\theta_{0:L}, h_T|\pi)} \right] \tag{56}$$

$$= KL(p(h_T|\theta_0, \pi)p(\theta_{0:L}) || \hat{p}(\theta_{0:L}, h_T|\pi)). \tag{57}$$

and $\hat{p}(\theta_{0:L}, h_T|\pi)$ is a valid distribution since

$$\int \hat{p}(\theta_{0:L}, h_T|\pi) d\theta_{0:L} dh_T = \int p(\theta_0)p(h_T|\pi)p(\theta_{1:L}) g(h_T, \theta_{0:L}) d\theta_{0:L} dh_T \tag{58}$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\frac{\exp(U(h_T, \theta_0))}{\frac{1}{L+1} \sum_{i=0}^{L} \exp(U(h_T, \theta_i))} \right], \tag{59}$$

because of the symmetry $\theta_0 \overset{d}{=} \theta_j \; \forall j = 1, \dots, L$

$$= \frac{1}{L+1} \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\frac{\sum_{j=0}^{L} \exp(U(h_T, \theta_j))}{\frac{1}{L+1} \sum_{i=0}^{L} \exp(U(h_T, \theta_i))} \right] \tag{60}$$

$$= 1. \tag{61}$$

Thus we have established

$$\mathcal{I}_T(\pi) = KL(p(h_T|\theta_0, \pi)p(\theta_{0:L})||\hat{p}(\theta_{0:L}, h_T|\pi)) + \mathcal{L}_T^{NCE}(\pi, U; L) \geq \mathcal{L}_T^{NCE}(\pi, U; L). \tag{62}$$

Next, substituting $U^*(h_T, \theta) = \log p(h_T|\theta, \pi) + c(h_T)$ in the definition of $\mathcal{L}^{NCE}(\pi, U; L)$ we obtain

$$\mathcal{L}_T^{NCE}(\pi, U^*; L) = \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\frac{p(h_T|\theta_0, \pi) \exp(c(h_T))}{\frac{1}{L+1} \sum_{i=0}^{L} p(h_T|\theta_i, \pi) \exp(c(h_T))} \right] \tag{63}$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\frac{p(h_T|\theta_0, \pi)}{\frac{1}{L+1} \sum_{i=0}^{L} p(h_T|\theta_i, \pi)} \right], \tag{64}$$

which is exactly the sPCE bound (5), which is monotonically increasing in L and tight in the limit as $L \to \infty$ [see 17, Theorem 2]. □

A.4 A note on optimal critics

An interesting feature of our approach is that, for both the InfoNCE and NWJ bounds, the optimal critics do not depend on the policy. This is because we include the designs as explicit inputs to the critics. Indeed, we have

$$U^*_{\text{NCE}}(h_T, \theta) = \log \left(\prod_{t=1}^{T} p(y_t|\theta, \xi_t, h_{t-1}) \right) + c(h_T), \tag{65}$$

$$U^*_{\text{NWJ}}(h_T, \theta) = \log \left(\frac{\prod_{t=1}^{T} p(y_t|\theta, \xi_t, h_{t-1})}{\int_\Theta p(\theta) \prod_{t=1}^{T} p(y_t|\theta, \xi_t, h_{t-1}) \, d\theta} \right) + 1. \tag{66}$$

In previous work that utilized critics for gradient-based BOED [16, 28], it was typical to not treat the designs $\xi_{1:T}$ as an input to the critic, which renders the optimal critic implicitly dependent on the designs. This makes more sense for static designs, for which the additional design input does not change. Our approach avoids an implicit dependence between policy and optimal critic which may be beneficial for the joint optimization.

B Theoretical Comparison and Additional Bounds

Recently, a number of studies have discussed the challenges of estimating mutual information, an in particular those associated with variational MI estimators [34, 42, 55].

Starting with the InfoNCE bound, it is trivial to show that the bound cannot exceed $\log(L+1)$, where L is the number of contrastive samples used to approximate the marginal in the denominator. Indeed,

$$\mathcal{L}_T^{NCE}(\pi, U; L) = \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \frac{\exp(U(h_T, \theta_0))}{\frac{1}{L+1} \sum_{i=0}^{L} \exp(U(h_T, \theta_i))} \right] \tag{67}$$

$$\leq \log(L+1) + \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \frac{\exp(U(h_T, \theta_0))}{\exp(U(h_T, \theta_0))} \right] \tag{68}$$

$$= \log(L+1) \tag{69}$$

This means that the corresponding Monte Carlo estimator will be highly biases whenever the true mutual information exceeds $\log(L+1)$, regardless of whether we have access to the optimal critic or not. This high bias estimator, however, comes with low variance [see e.g. 42, for discussion]. With the optimal critic we would require exponential (in the MI) number of samples to accurately estimate the true mutual information.

It might appear at first that the NWJ bound might offer a better trade-off between bias and variance. Recall from the proof of Proposition 2, we have for the *optimal* critic

$$\mathcal{L}_T^{NWJ}(\pi, U^*) = \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[\log \frac{p(h_T|\theta,\pi)}{p(h_T|\pi)}\right] + 1 - e^{-1}\mathbb{E}_{p(\theta)p(h_T|\pi)}\left[\frac{p(h_T|\theta,\pi)}{p(h_T|\pi)}e^1\right], \quad (70)$$

of which we form a Monte carlo estimate using N (M) samples for the first (second) term, respectively

$$\approx \frac{1}{N}\sum_{n=1}^{N}\log\frac{p(h_{T,n}|\theta_n,\pi)}{p(h_{T,n}|\pi)} + \left(1 - \frac{1}{M}\sum_{m=1}^{M}\log\frac{p(h_{T,m}|\theta_m,\pi)}{p(h_{T,m}|\pi)}\right), \quad (71)$$

where $\theta_n, h_{T,n} \sim p(\theta)p(h_T|\theta,\pi)$ are samples from the joint distribution and $\theta_m, h_{T,m} \sim p(\theta)p(h_T|\pi)$ are samples from the product of marginals. The first term is a Monte Carlo estimate of the mutual information, while the second has mean zero, meaning that this estimator is unbiased. The second term, however has variance which grows exponentially with the value of the (true) mutual information [see Theorem 2 in 55]. What this means is that even with an optimal critic, we will need an exponential (in the MI) number of samples to control the variance of the NWJ estimator. One might then hope that the variance can be reduced when using a sub-optimal critic at the cost of introducing some (hopefully small) bias. Unfortunately, according to a recent result [see Theorems 3.1 and 4.1 in 34, and the discussion therein], it is not possible to guarantee that a likelihood-free lower bound on the mutual information can exceed $\log(N)$. Indeed, the authors show theoretically and empirically that all high-confidence distribution-free lower bounds on the mutual information require exponential (in the the MI) number of samples.

Constructing a better lower bound on the mutual information—one that does not need exponential number of samples—therefore, requires us to make additional assumptions. Foster et al. [17] propose one such bound, namely the sequential Adaptive Constrative Estimation (sACE). The sACE bound introduces a proposal distribution $q(\theta; h_T)$, which aims to approximate the posterior $p(\theta|h_T)$. Since implicit models were not the focus of the work in [17] the proposed bound, relies on analytically available likelihood. The following proposition shows we can derive a likelihood-free version of the sACE bound.

Proposition 4 (Sequential Likelihood-free ACE)**.** *For a design function π, a critic function U, a number of contrastive samples $L \geq 1$, and a proposal $q(\theta; h_T)$, we have the sequential Likelihood-free Adaptive Contrastive Estimation (sLACE) lower bound*

$$\mathcal{L}_T^{sLACE}(\pi, U, q; L) := \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)}\left[\log\frac{U(h_T,\theta_0)}{\frac{1}{L+1}\sum_{\ell=0}^{L}\frac{U(h_T,\theta_\ell)p(\theta_\ell)}{q(\theta_\ell;h_T)}}\right] \leq \mathcal{I}_T(\pi). \quad (72)$$

The bound is tight as $L \to \infty$ for the optimal critic $U^*(h_T,\theta) = \log p(h_T|\theta,\pi) + c(h_T)$, where $c(h_T)$ is arbitrary. In addition, if $q(\theta; h_T) = p(\theta|h_T)$, the bound is tight for the optimal critic $U^*(h_T,\theta)$ with any $L \geq 0$.

Proof. The proof follows similar arguments to the ones in Propositions 2 and 3. First let

$$g(h_T, \theta_{0:L}) := \frac{U(h_T,\theta_0)}{\frac{1}{L+1}\sum_{\ell=0}^{L}\frac{U(h_T,\theta_\ell)p(\theta_\ell)}{q(\theta_\ell;h_T)}} \quad (73)$$

Starting with the definition of the EIG:

$$\mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)}\left[\log\frac{p(h_T|\theta_0,\pi)}{p(h_T|\pi)}\right] \quad (74)$$

since $q(\theta_i; h_T)$ is a valid density

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)}\left[\log\frac{p(h_T|\theta_0,\pi)}{p(h_T|\pi)}\right] \quad (75)$$

multiplying its numerator and denominator inside the log by $g(h_T, \theta_{0:L}) > 0$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)} \left[\log \frac{p(h_T|\theta_0, \pi)g(h_T, \theta_{0:L})}{p(h_T|\pi)g(h_T, \theta_{0:L})}\right] \tag{76}$$

$$\begin{aligned} &= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)} \left[\log g(h_T, \theta_{0:L})\right] \\ &\quad + \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)g(h_T, \theta_{0:L})}\right] \end{aligned} \tag{77}$$

The first term is exactly the sLACE bound, $\mathcal{L}_T^{\text{sLACE}}(\pi, U, q; L)$. We now show that the second term is a KL divergence between two distribitions and hence non-negative. To see this

$$\mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)g(h_T, \theta_{0:L})}\right] \tag{78}$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)} \left[\log \frac{p(\theta_0)p(h_T|\theta_0, \pi)q(\theta_{1:L};h_T)}{p(h_T|\pi)g(h_T, \theta_{0:L})p(\theta_0)q(\theta_{1:L};h_T)}\right] \tag{79}$$

$$= KL(p(\theta_0)p(h_T|\theta_0, \pi)q(\theta_{1:L};h_T)||\hat{p}(h_T, \theta_{0:L})), \tag{80}$$

since $\hat{p}(h_T, \theta_{0:L}) \coloneqq p(h_T|\pi)g(h_T, \theta_{0:L})p(\theta_0)q(\theta_{1:L};h_T)$ is a valid density. Indeed:

$$\int \hat{p}(h_T, \theta_{0:L})dh_T d\theta_{0:L} = \mathbb{E}_{q(\theta_{1:L};h_T)p(h_T|\pi)} \left[p(\theta_0)g(h_T, \theta_{0:L})\right] \tag{81}$$

$$= \mathbb{E}_{q(\theta_{1:L};h_T)p(h_T|\pi)} \left[p(\theta_0)\frac{U(h_T, \theta_0)}{\frac{1}{L+1}\sum_{\ell=0}^{L} \frac{U(h_T,\theta_\ell)p(\theta_\ell)}{q(\theta_\ell;h_T)}}\right] \tag{82}$$

$$= \mathbb{E}_{q(\theta_{0:L};h_T)p(h_T|\pi)} \left[\frac{\frac{U(h_T,\theta_0)p(\theta_0)}{q(\theta_0;h_T)}}{\frac{1}{L+1}\sum_{\ell=0}^{L} \frac{U(h_T,\theta_\ell)p(\theta_\ell)}{q(\theta_\ell;h_T)}}\right] \tag{83}$$

by symmetry

$$= \mathbb{E}_{q(\theta_{0:L};h_T)p(h_T|\pi)} \left[\frac{\frac{1}{L+1}\sum_{\ell=0}^{L} \frac{U(h_T,\theta_\ell)p(\theta_\ell)}{q(\theta_\ell;h_T)}}{\frac{1}{L+1}\sum_{\ell=0}^{L} \frac{U(h_T,\theta_\ell)p(\theta_\ell)}{q(\theta_\ell;h_T)}}\right] \tag{84}$$

$$= 1. \tag{85}$$

With the optimal critic we recover the sACE bound from [17], which under mild conditions converges to the mutual information $\mathcal{I}_T(\pi)$. To see that start by writing

$$\mathcal{L}_T^{\text{sLACE}}(\pi, U^*, q; L) = \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)} \left[\log \frac{p\left(h_T|\theta_0, \pi\right)}{\frac{1}{L+1}\sum_{\ell=0}^{L} \frac{p(h_T|\theta_\ell,\pi)p(\theta_\ell)}{q(\theta_\ell;h_T)}}\right]. \tag{86}$$

The denominator is a consistent estimator of the marginal, provided that each term in the sum is bounded, and so by the Strong Law of Large Numbers we have

$$\frac{1}{L+1}\sum_{\ell=0}^{L} \frac{p\left(h_T|\theta_\ell, \pi\right)p\left(\theta_\ell\right)}{q\left(\theta_\ell; h_T\right)} \to p(h_T|\pi) \ a.s. \quad \text{as } L \to \infty, \tag{87}$$

which establishes point-wise convergence of the integrand to $p(h_T|\theta_0, \pi)/p(h_T|\pi)$. We can apply Bounded convergence theorem to establish $\mathcal{L}_T^{\text{sACE}}(\pi, U^*, q; L) \to \mathcal{I}_T(\pi)$ as $L \to \infty$.

If in addition $q(\theta; h_T) = p(\theta|h_T)$ we have by Bayes rule:

$$\mathcal{L}_T^{\text{sLACE}}(\pi, U^*, q; L) = \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L}|h_T)} \left[\log \frac{p(h_T|\theta_0, \pi)}{\frac{1}{L+1}\sum_{\ell=0}^{L} \frac{p(h_T|\theta_\ell,\pi)p(\theta_\ell)}{p(\theta_\ell|h_T)}}\right] \tag{88}$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L}|h_T)} \left[\log \frac{p(h_T|\theta_0, \pi)}{\frac{1}{L+1}\sum_{\ell=0}^{L} p(h_T|\pi)}\right] \tag{89}$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)}\right] \tag{90}$$

$$= \mathcal{I}_T(\pi) \quad \forall L \geq 0. \tag{91}$$

□

In practice, we parameterize the policy, the critic and the density of the proposal distribution by neural networks π_ϕ, U_ψ and q_ζ and optimize $\mathcal{L}_T^{\text{sLACE}}$ with respect to the parameters of these networks, ϕ, ψ and ζ with SGA. As before, optimizing with respect to ϕ improves the quality of the designs, proposed by the policy, whilst optimizing with respect to ψ and ζ tightens the bound. If the parametric density q_ζ and the critic U_ψ are expressive enough, so that we can recover the optimal critic and the true posterior, then the bound is tight for any number of contrastive samples L. If, on the other hand, we fix $q_\zeta(\theta; h_T) = p(\theta)$ instead of training it, then we recover the InfoNCE bound. Therefore, as long as q_ζ approximates the posterior better than the prior, then even an imperfect proposal q_ζ can benefit training.

In addition to introducing another set of optimizable parameters, ζ, the sLACE bound assumes that we know the prior $p(\theta)$ and can evaluate its density.

C Neural architecture

C.1 Permutation invariance of the critic for exchangeable experiments

We show that if the BOED problem is exchangeable then the critic function U should be permutation-invariant.

Proposition 5 (Permutation invariance)**.** *Let σ be a permutation acting on a history h_T^1 yielding $h_T^2 = \{(\xi_{\sigma(i)}, y_{\sigma(i)})\}_{i=1}^T$. If the data generating process is conditionally independent of its past given θ, then the optimal critics for both (7) and (8) are invariant under permutations of the history, i.e.*

$$p(\theta)\prod_{t=1}^{T} p(y_t|\theta, \xi_t(h_{t-1}), h_{t-1}) = p(\theta)\prod_{t=1}^{T} p(y_t|\theta, \xi_t) \implies U^*(h_T^1, \theta) = U^*(h_T^2, \theta). \tag{92}$$

Proof. This is a direct consequence from the form of the optimal critics. To see this formally, let h_T^1 be a history and h_T^2 be a permutation of it.

Starting with the InfoNCE bound we have

$$U^*_{\text{NCE}}(h_T^1, \theta) = \log p(h_T^1|\theta, \pi) + c(h_T^1) \tag{93}$$

$$= \log \prod_{t=1}^{T} p(y_t|\theta, \xi_t) + c(\{(\xi_t, y_t)\}_{t=1}^T) \tag{94}$$

since $c(h_T)$ is arbitrary, we can choose it to be permutation invariant

$$= \log \prod_{t=1}^{T} p(y_{\sigma(t)}|\theta, \xi_{\sigma(t)}) + c(\{(\xi_{\sigma(t)}, y_{\sigma(t)})\}_{t=1}^T) \tag{95}$$

$$= \log p(h_T^2|\theta, \pi) + c(h_T^2) \tag{96}$$

$$= U^*_{\text{NCE}}(h_T^2, \theta) \tag{97}$$

Similarly, for the optimal critic of the NWJ bound we have

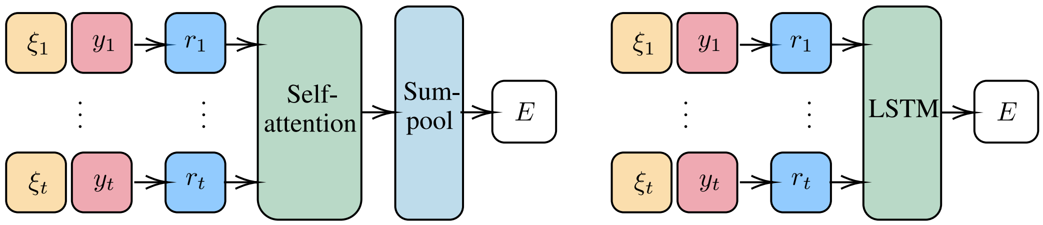
$$U^*_{\text{NWJ}}(h_T^1, \theta) = \log \frac{p(h_T^1|\theta, \pi)}{p(h_T^1|\pi)} + 1 \tag{98}$$

$$= \log \frac{\prod_{t=1}^{T} p(y_t|\theta, \xi_t)}{\mathbb{E}_{p(\theta)}\left[\prod_{s=1}^{T} p(y_s|\theta, \xi_s)\right]} + 1 \tag{99}$$

$$= \log \frac{\prod_{t=1}^{T} p(y_{\sigma(t)}|\theta, \xi_{\sigma(t)})}{\mathbb{E}_{p(\theta)}\left[\prod_{s=1}^{T} p(y_{\sigma(s)}|\theta, \xi_{\sigma(s)})\right]} + 1 \tag{100}$$

$$= \log \frac{p(h_T^2|\theta, \pi)}{p(h_T^2|\pi)} + 1 = U^*_{\text{NWJ}}(h_T^2, \theta). \tag{101}$$

□



(a) Exchangeable models (b) Non-exchangeable models

Figure 6: History encoder architectures for different classes of models. When conditional independence of the experiments holds, we use self-attention, followed by sum-pooling, making the history encoder permutation invariant. When experiments are not conditionally independent we use LSTM and only keep its last hidden state. We train two separate history encoders—one for the design network π_ϕ and one for the critic network U_ψ, although we note that all the weights except those in the head layers can be shared.

To the best of our knowledge, we are the first to propose a critic architecture that is tailored to BOED problems with exchangeable models. Previous work in the static BOED setting, where MI information objective is optimized with variational lower bounds and thus require the training of critics [e.g. 28, 63], did not discuss what an appropriate critic architecture might be. In particular, in all experiments [28, 63] use a generic architecture for both exchangeable and non-exchangeable problems. An expressive enough generic architecture should be able to obtain the optimal critic, and thus achieve a tight bound, however, the optimisation process will be considerably more difficult as the network needs to learn this key invariance structure. We therefore recommend using permutation invariant architectures whenever the model is exchangeable, especially if achieving tight bounds (and therefore learning an optimal critic) is of importance.

C.2 Further details on the history encoder

Figure 6 shows the history encoders we use in the policy network π_ϕ and the critic network U_ψ. First, we encode the individual design-outcome pairs, (ξ_t, y_t), with an MLP, which gives us a vector of representations $r_t \in \mathbb{R}^m$, where m is the encoding dimension we have selected. The representations $\{r_i\}_{i=1}^t$ are row-stacked into a matrix R of dimension $t \times m$, which we then aggregate back to a vector of size m by an appropriate layer(s).

When conditional independence of the experiments holds, we apply 8-head self-attention, based on the Image Transformer [40] and as implemented by [14]. Applying self-attention leaves the dimension of the matrix R unchanged. We then apply sum-pooling across time t, which gives us the final encoding vector $E \in \mathbb{R}^m$.

When experiments are not conditionally independent, we pass the matrix R though an LSTM with two hidden layers and hidden state of size m (see the LSTM module in Pytorch for more details). The LSTM returns hidden state vectors associated with the history h_t for each t; we keep the last hidden state of the last layer, which is our final encoding vector $E \in R^m$.

In both cases the resulting encoding E is a vector of size m. It is passed through final fully connected "head" layers, which output either a design (in the case of the policy) or a vector (in the case of the critic). We train two separate history encoders—one for the design network π_ϕ and one for the critic network U_ψ, although we note that all the weights except those in the head layers can be shared.

D Experiments

D.1 Computational resources

All of the experiments were implemented in Python using open-source software. All estimators and models were implemented in PyTorch [41] (BSD license) and Pyro [6] (Apache License Version 2.0), whilst MlFlow [61] (Apache License Version 2.0) was used for experiment tracking and management. The self-attention architecture from [14] was used to implement the self-attention mechanisms in the design and critic networks. For full details on package versions, environment set-up and commands for running the code, see instructions in the `README.md` file.

Experiments were ran on internal GPU clusters, consisting of GeForce RTX 3090 (24GB memory), GeForce RTX 2080 Ti (11GB memory) and GeForce GTX 1080 Ti GPUs (11GB memory).

The deployment-time of iDAD (Table 4) was estimated on a lightweight CPU machine with the following specifications

Processor	2.8 GHz Quad-Core Intel Core i7
Memory	16 GB
Operating system	macOS Big Sur v11.2.3

D.2 CO2 Emission Related to Experiments

Experiments were conducted using a private infrastructure, which has an estimated carbon efficiency of 0.432 kgCO_2eq/kWh. A cumulative of 160 hours of computation was performed on hardware of type RTX 2080 Ti (TDP of 250W), or similar. The training time of each experiment (including the baselines that require optimization), took on average between 1-3 GPU hours, depending on the number of experiments T.

Total emissions are estimated to be 17.28 kgCO_2eq of which 0% was directly offset.

Estimations were conducted using the Machine Learning Impact calculator presented in [31].

D.3 Traditional sequential BOED with variational posterior estimator

The variational posterior estimator from [15] is based on the Barbar-Agakov lower bound [4], which takes the form

$$\mathcal{L}^{\text{post}}(\xi, q_\psi) = \mathbb{E}_{p(\theta)p(y|\theta,\xi)}\left[\log \frac{q_\psi(\theta; y, \xi)}{p(\theta)}\right] \leq \mathcal{I}(\xi), \tag{102}$$

where q_ψ is any normalized distribution over the parameters θ. The bound is tight when $q_\psi(\theta; y, \xi) = p(\theta|y, \xi)$, i.e. if we can recover the true posterior. We assume mean-field variational family and optimize the parameters ψ by maximizing the bound (102) using stochastic gradient schemes. Simultaneously we optimize the bound with respect to the design variable ξ to select the optimal design ξ^*. At the inference stage, denoting by y^* the outcome of experiment ξ^*, we obtain an approximate posterior by evaluating $q_\psi(\theta; y^*, \xi^*)$, i.e. we reuse the learnt variational posterior. We repeat this process at each stage of the experiments by substituting the the approximate posterior, $q_\psi(\theta; y^*, \xi^*)$, as the prior in (102).

D.4 Location Finding

In this experiment we have K hidden objects (*sources*) in $\mathbb{R}^2$ and we wish to learn their locations, $\theta = \{\theta_1, \dots, \theta_K\}$. The number of sources, K, is assumed to be known. Each source emits a signal with intensity obeying the inverse-square law. Put differently, if a source is located at θ_k and we perform a measurement at a point ξ, the signal strength emitted from that source only will be proportional to $\frac{1}{\|\theta_k - \xi\|^2}$. The total intensity at location ξ, emitted from all K sources, is a superposition of the individual ones

$$\mu(\theta, \xi) = b + \sum_{k=1}^{K} \frac{\alpha_k}{m + \|\theta_k - \xi\|^2}, \tag{103}$$

where α_k can be known constants or random variables, $b > 0$ is a constant background signal and m is a constant, controlling the maximum signal.

We place a standard normal prior on each of the location parameters θ_k and we observe the log-total intensity with some Gaussian noise. We therefore have the following prior and likelihood:

$$\theta_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0_d, I_d) \quad \log y \mid \theta, \quad \xi \sim \mathcal{N}(\log \mu(\theta, \xi), \sigma^2) \tag{104}$$

D.4.1 Training details

All our experiments are performed with the following model hyperparameters

Parameter	Value
Number of sources, K	2
α_k	$1\ \forall k$
Max signal, m	10^{-4}
Base signal, b	10^{-1}
Observation noise scale, σ	0.5

The architecture of the design network π_ϕ used in Table 2 and all its hyperparameters are in the following tables. For the encoder of the design-outcome pairs we used the following:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	ξ, y	3	3	-
H1	Fully connected	64	64	ReLU
H2	Fully connected	512	512	ReLU
Output	Fully connected	64	64	-
Attention	8 heads	64	64	-

The output of the encoder, $R(h_t)$, is fed into an emitter network, for which we used the following:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	$R(h_t)$	64	64	-
H1	Fully connected	256	256	ReLU
H2	Fully connected	64	64	ReLU
Output	Fully connected	2	2	-

The architecture of the critic network U_ψ used in Table 2 and all its hyperparameters are in the tables that follow. First, the encoder network of the latent variables is:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	θ	4	4	-
H1	Fully connected	16	16	ReLU
H2	Fully connected	64	64	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	64	64	-

For the design-outcome pairs encoder we use the same architecture as in the design network, namely:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	ξ, y	3	3	-
H1	Fully connected	64	64	ReLU
H2	Fully connected	512	512	ReLU
Output	Fully connected	64	64	-
Attention	8 heads	64	64	-

The output of the encoder, $R(h_t)$, is fed into fully connected head layers:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	$R(h_t)$	64	64	-
H1	Fully connected	1024	1024	ReLU
H2	Fully connected	512	512	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	64	64	-

The optimisation was performed with Adam [26] with `ReduceLROnPlateau` learning rate scheduler, with the following hyperparameters:

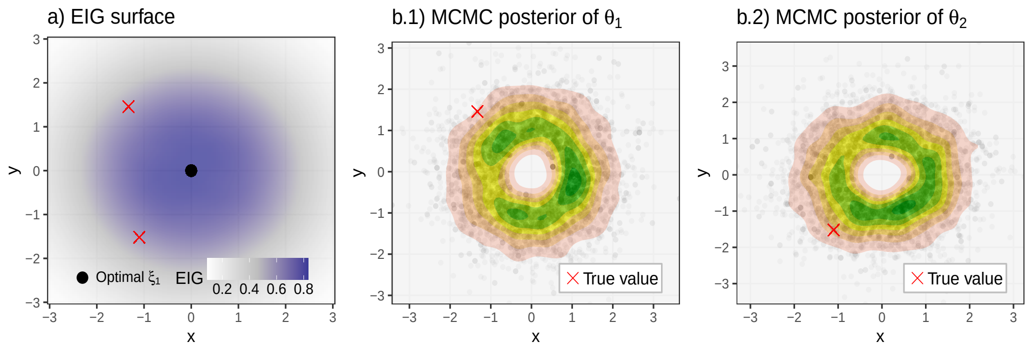


Figure 7: a): EIG surface induced by the prior; b) Samples from $p(\theta|\xi_1, y_1)$—the posterior distribution of the locations, after performing experiment ξ_1 and observing y_1, along with a KDE.

Parameter	iDAD, InfoNCE	iDAD, NWJ
Batch size	2048	2048
Number of contrastive/negative samples	2047	2047
Number of gradient steps	100000	100000
Initial learning rate (LR)	0.0005	0.0005
LR annealing factor	0.8	0.8
LR annealing frequency (if no improvement)	2000	2000

D.4.2 Performance of the variational baseline

As we saw in Table 2, this variational approach to (myopic) adaptive BOED performed very poorly, despite its large computational budget. The likely reason for that is that the mean-field variational approximation cannot adequately capture the complex non-Gaussian posterior of this problem. Figure 7 clearly demonstrates this: before any data is observed it is optimal to sample at the origin (since the prior is centered at it). After observing a low signal (the locations in this example are not close to the origin), we can only conclude that the sources are not within a small radius of the origin, but anywhere outside of it would be a plausible location, as indeed indicated by the fitted posteriors.

D.4.3 Hyperparameter selection

We did not perform extensive hyperparameters search; in particular, the network sizes were guided by two hyperparametes: hidden-dimension ($HD = 512$) and encoding dimension ($ED = 64$). We set-up all the networks to scale up with the number of experiments as follows:

- Design-outcome encoder has three hidden layers of sizes $[64, HD, ED]$.
- Design emitter network has three hidden layers of sizes $[HD/2, ED, 2]$, where 2 is the dimension of the design variable.
- The latent encoder for the critic network has four hidden layers of sizes $[16, 64, HD, ED]$.
- The critic design-outcome encoder's head layer has four hidden layers of sizes $[HD \times \log(T), HD \times \log(T)/2, HD, ED]$.

Since our multi-head attention layer has 8 heads, the encoding dimension we use has to be a multiple of 8. In addition to $ED = 64$ we tried $ED = 32$ which provided marginally worse results. We did not try other values for these hyperparameters.

For the learning rate, we tried 0.001, which was too high, as well as 0.0005 (which we selected) and 0.0001 (which yielded very similar results).

We performed similar level of hyperparameter tuning for all trainable baselines as well (DAD, MINEBED and SG-BOED).

Table 6: Upper bounds on the total information, $\mathcal{I}_{10}(\pi)$, for the location finding experiment in Section 5.1. The bounds were estimated using $L = 5 \times 10^5$ contrastive samples. Errors indicate ± 1 s.e. estimated over 4096 histories (128 for variational). Lower bounds are presented in Table 2.

Method \ θ dim.	4D	6D	10D	20D
Random	4.794 ± 0.041	3.506 ± 0.004	1.895 ± 0.003	0.552 ± 0.001
MINEBED	5.522 ± 0.028	4.229 ± 0.029	2.459 ± 0.029	0.801 ± 0.019
SG-BOED	5.549 ± 0.028	4.220 ± 0.030	2.455 ± 0.029	0.803 ± 0.019
Variational	4.644 ± 0.146	3.626 ± 0.167	2.181 ± 0.152	0.669 ± 0.097
iDAD (NWJ)	7.806 ± 0.050	5.851 ± 0.041	$\mathbf{3.264 \pm 0.039}$	0.877 ± 0.022
iDAD (InfoNCE)	$\mathbf{7.863 \pm 0.043}$	$\mathbf{6.068 \pm 0.039}$	$\mathbf{3.257 \pm 0.040}$	$\mathbf{0.872 \pm 0.020}$
DAD	8.034 ± 0.038	6.310 ± 0.031	3.358 ± 0.040	0.953 ± 0.022

Table 7: Upper and lower bounds on the total information, $\mathcal{I}_{20}(\pi)$, for the location finding experiment in 2D from Section 5.1. The bounds were estimated using $L = 5 \times 10^5$ contrastive samples. Errors indicate ± 1 s.e. estimated over 4096 histories.

Method	Lower bound	Upper bound
Random	7.000 ± 0.034	7.020 ± 0.034
MINEBED	7.672 ± 0.030	7.690 ± 0.031
SG-BOED	7.701 ± 0.030	7.728 ± 0.031
iDAD (NWJ)	$\mathbf{9.961 \pm 0.033}$	$\mathbf{10.372 \pm 0.048}$
iDAD (InfoNCE)	$\mathbf{10.075 \pm 0.032}$	$\mathbf{10.463 \pm 0.043}$
DAD	10.424 ± 0.031	10.996 ± 0.049

D.4.4 Further ablation studies

Scalability with number of experiments. We first demonstrate that iDAD can scale to a larger number of experiments T. We train policy networks to perform $T = 20$ experiments and compare them to baselines in Table 7. We omit the variational baseline as it is too computationally costly to run for a large enough number of histories, and as we saw in the previous subsection, it is not particularly suited to this model.

Training stability. To assess the robustness of the results and the stability of the training process, we trained 5 additional iDAD networks with each of the two bounds, using different seeds but the same hyperparameters (described in Subsection D.4.1) we used to produce the results of the location finding experiment in 2D (Table 2 in the main text). We report upper and lower bounds on the mutual information along with their mean and standard error in the table below.

Estimator	Bound	Run 1	Run 2	Run 3	Run 4	Run 5	**Mean**	**SE**
InfoNCE	Lower	7.826	7.682	7.856	7.713	7.804	**7.776**	**0.034**
InfoNCE	Upper	7.933	7.791	7.856	7.807	7.925	**7.862**	**0.029**
NWJ	Lower	7.820	7.545	7.592	7.555	7.691	**7.641**	**0.052**
NWJ	Upper	7.976	7.640	7.669	7.651	7.800	**7.747**	**0.064**

We can see that the iDAD networks trained with InfoNCE are highly stable, with the 5 additional runs achieving very similar mutual information values to each other and to the iDAD network used the report the results in the main paper. The performance of the iDAD networks trained with the NWJ bound is more variable and empirically achieve slightly lower average value of mutual information. This higher variance is in-line with the discussion in Section B.

We similarly verify the robustness of the static baselines, reporting the results in the table below:

Table 8: Ablation study on the performance of iDAD as a function of training time for the location finding experiment.

Training budget	MI lower bound
0.1%	3.38
1.0%	6.09
2.0%	6.46
4.0%	6.81
8.0%	7.08
16.0%	7.33
32.0%	7.56
64.0%	7.78
100.0%	7.82

Estimator	Bound	Run 1	Run 2	Run 3	Run 4	Run 5	**Mean**	**SE**
SG-BOED	Lower	5.537	5.536	5.473	5.523	5.518	**5.517**	**0.013**
SG-BOED	Upper	5.553	5.548	5.491	5.541	5.531	**5.533**	**0.012**
MINEBED	Lower	5.460	5.506	5.553	5.539	5.565	**5.524**	**0.021**
MINEBED	Upper	5.473	5.526	5.567	5.554	5.574	**5.540**	**0.022**

Performance sensitivity to errors in the policy. Finally, we investigate the effect of slight errors in the design policy network. To this end, we look at the performance achieved by partially trained design networks (there will be some errors or inaccuracies in networks that were not trained until convergence). Table 8 shows the performance of iDAD as a function of training time, demonstrating that small errors in the network only lead to small drops in performance.

In detail, our results show that with just 8% of the total training budget, this slightly inaccurate network still performs relatively well, achieving total mutual information of 7.1, compared to the fully trained network that reached 7.8. We also highlight that iDAD outperforms all baselines with as little as 1% of the total training budget (the best performing baseline achieves mutual information of 5.5, see Table 2).

D.5 PK model

The drug concentration z, measured ξ hours after administering it, and the corresponding noisy observation y are given by

$$z(\xi;\theta) = \frac{D_V}{V}\frac{k_\alpha}{k_\alpha - k_e}[e^{-k_e\xi} - e^{-k_\alpha\xi}], \quad y(\xi;\theta) = z(\xi;\theta)(1+\epsilon)+\eta \tag{105}$$

where $\theta = (k_\alpha, k_e, V)$, $D_V = 400$ is a constant, $\epsilon \sim \mathcal{N}(0, 0.01)$ is multiplicative noise to account for heteroscedasticity and $\eta \sim \mathcal{N}(0, 0.1)$ is an additive observation noise. Since both noise sources are Gaussian, the observation likelihood is also Gaussian i.e.

$$y(\xi;\theta) \sim \mathcal{N}(z(\xi;\theta), 0.01 z(\xi;\theta)^2 + 0.1) \tag{106}$$

The prior for the parameters θ that we used

$$\log\theta \sim \mathcal{N}\left(\begin{bmatrix}\log 1\\ \log 0.1\\ \log 20\end{bmatrix}, \begin{bmatrix}0.05 & 0 & 0\\ 0 & 0.05 & 0\\ 0 & 0 & 0.05\end{bmatrix}\right) \tag{107}$$

D.5.1 Training details

The architecture of the design network π_ϕ used for Figure 3 and 4 and all its hyperparameters are in the following tables. For the encoder of the design-outcome pairs we used the following:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	ξ, y	2	2	-
H1	Fully connected	64	64	ReLU
H2	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-
Attention	8 heads	32	32	-

The outputs of the encoder, $\{R(h_t)\}_{t=1}^{T}$, are summed and the resulting vector (of dimension 32) is fed into an emitter network, for which we used the following:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	$R(h_t)$	32	32	-
H1	Fully connected	256	256	ReLU
H2	Fully connected	32	32	ReLU
Output	Fully connected	1	1	Sigmoid

The architecture of the critic network U_ψ used in Figures 3 and 4 and all its hyperparameters are in the following tables. For the encoder of the design-outcome pairs we used the same architecture as for the design network, namely:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	ξ, y	2	2	-
H1	Fully connected	64	64	ReLU
H2	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-
Attention	8 heads	32	32	-

The resulting pooled representation, $R(h_T)$ is fed into fully connected critic head layers with the following architecture:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	$R(h_T)$	32	32	-
H1	Fully connected	512	512	ReLU
H2	Fully connected	256	256	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-

Finally, for the latent variable encoder network we used:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	θ	3	3	-
H1	Fully connected	8	8	ReLU
H2	Fully connected	64	64	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-

The optimisation was performed with Adam [26] with the following hyperparameters:

Parameter	iDAD, InfoNCE	iDAD, NWJ
Batch size	1024	1024
Number of contrastive/negative samples	1023	1023
Number of gradient steps	100000	100000
Initial learning rate (LR)	0.0001	0.0001
LR annealing factor	0.8	0.5
LR annealing frequency (if no improvement)	2000	2000

Table 9: Upper and lower bounds on the total information, $\mathcal{I}_5(\pi)$, for the pharmacokinetic experiment. Errors indicate ± 1 s.e. estimated over 4096 (126 for variational) histories and $L = 5 \times 10^5$.

Method	Lower bound	Upper bound	Deployment time
Random	2.523 ± 0.033	2.523 ± 0.033	N/A
Equal interval	2.651 ± 0.022	2.651 ± 0.022	N/A
MINEBED	2.955 ± 0.030	2.956 ± 0.030	N/A
SG-BOED	2.985 ± 0.027	2.985 ± 0.027	N/A
Variational	2.683 ± 0.093	2.683 ± 0.093	$505.4 \quad \pm 1\%$
IDAD (NWJ)	$\mathbf{3.163 \pm 0.023}$	$\mathbf{3.163 \pm 0.023}$	$0.007 \pm 7\%$
IDAD (InfoNCE)	$\mathbf{3.200 \pm 0.024}$	$\mathbf{3.200 \pm 0.024}$	$0.007 \pm 8\%$
DAD	3.234 ± 0.023	3.234 ± 0.023	$0.002 \pm 7\%$

Table 10: Upper and lower bounds on the total information, $\mathcal{I}_{10}(\pi)$, for the pharmacokinetic experiment. Errors indicate ± 1 s.e. estimated over 4096 (126 for variational) histories and $L = 5 \times 10^5$.

Method	Lower bound	Upper bound	Deployment time
Random	3.344 ± 0.034	3.345 ± 0.034	N/A
Equal interval	3.422 ± 0.026	3.423 ± 0.026	N/A
MINEBED	3.849 ± 0.034	3.849 ± 0.034	N/A
SG-BOED	3.824 ± 0.034	3.824 ± 0.034	N/A
Variational	3.624 ± 0.099	3.624 ± 0.099	$1055.2 \quad \pm 8\%$
IDAD (NWJ)	$\mathbf{4.034 \pm 0.025}$	$\mathbf{4.034 \pm 0.025}$	$0.007 \pm 6\%$
IDAD (InfoNCE)	$\mathbf{4.045 \pm 0.026}$	$\mathbf{4.045 \pm 0.026}$	$0.007 \pm 5\%$
DAD	4.116 ± 0.024	4.117 ± 0.024	$0.007 \pm 8\%$

D.5.2 Hyperparameter selection

Hyperparameter selection was done in a way similar to the Location Finding experiment (see D.4.3). We tried encodin dimensions $ED = 32, 64$ and selected the smaller size as there were no clear benefits to larger networks (relatively speaking, this is an easier model that the location finding). We used the same hidden dimension, i.e. $HD = 512$. In terms of learning rates, we tried 0.0001, 0.0005 and 0.001; we found 0.0001 to be appropriate, although NWJ bound was exhibiting high variance, so used a smaller learning rate annealing factor for that network (0.5 vs 0.8 for InfoNCE). We performed similar level of hyperparameter tuning for all trainable baselines as well (DAD, MINEBED and SG-BOED).

D.5.3 Further results

Table 9 reports the results shown in Figure 3c), along with the corresponding upper bounds and deployment times, while Table 10 reports the results for $T = 10$.

Training stability. To assess the robustness of the results and the stability of the training process, we trained 5 additional iDAD networks with each of the two bounds, using different seeds but the same hyperparameters we used to produce the results of the pharmacokinetic experiment (Figure 3c) and corresponding Table 9). We report upper and lower bounds on the mutual information along with their mean and standard error in the table below.

Method	Bound	Run 1	Run 2	Run 3	Run 4	Run 5	**Mean**	**SE**
iDAD, InfoNCE	Lower	3.209	3.165	3.198	3.221	3.128	**3.185**	**0.019**
iDAD, InfoNCE	Upper	3.210	3.166	3.201	3.223	3.130	**3.186**	**0.019**
iDAD, NWJ	Lower	3.034	3.049	2.608	3.149	3.082	**3.034**	**0.107**
iDAD, NWJ	Upper	3.034	3.049	2.609	3.150	3.083	**3.034**	**0.107**

We repeat the same procedure for the static baselines. The results reported in the table below demonstrate the training stability of these baselines as well.

Method	Bound	Run 1	Run 2	Run 3	Run 4	Run 5	**Mean**	**SE**
SG-BOED	Lower	2.932	2.452	2.448	2.991	2.962	**2.757**	**0.140**
SG-BOED	Upper	2.932	2.453	2.449	2.992	2.962	**2.757**	**0.140**
MINEBED	Lower	2.912	2.213	3.014	2.092	2.941	**2.634**	**0.221**
MINEBED	Upper	2.914	2.213	3.015	2.092	2.942	**2.635**	**0.222**

D.6 SIR Model

Generally speaking, the SIR model advocates that, within a fixed population of size N, susceptible individuals $S(\tau)$, where τ is time, can become infected and move to an infected state $I(\tau)$. The infected individuals can then recover from the disease and move to the recovered state $R(\tau)$. The dynamics of these events are governed by the infection rate β and recovery rate γ, which define the particular disease in question. In the context of BOED, the aim is generally to estimate these two model parameters by observing state populations at particular measurement times τ, which are the experimental design variables. The SIR model has been studied extensively in the context of BOED, e.g. in [12, 27, 29, 30].

Stochastic versions of the SIR model are usually formulated via continuous-time Markov chains (CTMC), which can be simulated from via the Gillespie algorithm [2], yielding discrete state populations. However, iDAD requires us to differentiate through the sampling path of the state populations to the experimental designs, which is impossible if the simulated data is discrete as gradients are undefined. Thus, we here implement an alternative formulation of the stochastic SIR model that is based on stochastic differential equations (SDEs), as studied in [29], which yields continuous state populations that can be differentiated.

Following [29], let us first define a state population vector $\mathbf{X}(\tau) = (S(\tau), I(\tau))^\top$, where we can safely ignore the population of recovered $R(\tau)$ for modelling purposes because we assume that the total population stays fixed. The system of Itô SDEs that defines the stochastic SIR model is given by

$$\mathrm{d}\mathbf{X}(\tau) = \mathbf{f}(\mathbf{X}(\tau))\mathrm{d}\tau + \mathbf{G}(\mathbf{X}(\tau))\mathrm{d}\mathbf{W}(\tau), \tag{108}$$

where $\mathbf{f}$ is a drift vector, $\mathbf{G}$ is a diffusion matrix and $\mathbf{W}(\tau)$ is a vector of independent Wiener processes. [29] showed that the drift vector and diffusion matrix are given by

$$\mathbf{f}(\mathbf{X}(\tau)) = \begin{pmatrix} -\beta\frac{S(\tau)I(\tau)}{N} \\ \beta\frac{S(\tau)I(\tau)}{N} - \gamma I(\tau) \end{pmatrix} \quad \text{and} \quad \mathbf{G}(\mathbf{X}(t)) = \begin{pmatrix} -\sqrt{\beta\frac{S(\tau)I(\tau)}{N}} & 0 \\ \sqrt{\beta\frac{S(\tau)I(\tau)}{N}} & -\sqrt{\gamma I(\tau)} \end{pmatrix}. \tag{109}$$

Given the system of Itô SDEs in (108), as well as the above drift vector and diffusion matrix, we can then simulate state populations $\mathbf{X}(\tau)$ by solving the SDE using finite-difference methods, such as e.g. the Euler-Maruyama method. See [29] for more information on the SDE-based SIR model, including derivations of the drift vector and diffusion matrix.

Importantly, we note that [29] further used the solutions of (108) as an input to a Poisson observation model, which increases the noise in simulated data. We here opt to simply use the solutions of (108) as data and do not consider an additional Poisson observational model.

D.6.1 Training details

As previously mentioned, the design variable for this model is the measurement time $\tau \in [0, 100]$. When solving the SDE with the Euler-Maruyama method, we discretize the time domain with a resolution of $\Delta\tau = 10^{-2}$. We here only use the number of infected $I(\tau)$ as the observed data, as others might be difficult to measure in reality. The total population is fixed at $N = 500$ and the initial conditions are $\mathbf{X}(\tau = 0) = (0, 2)^\top$. The model parameters β and γ have log-normal priors, i.e. $p(\beta) = \text{Lognorm}(0.50, 0.50^2)$ and $p(\gamma) = \text{Lognorm}(0.10, 0.50^2)$. Importantly, because solving SDEs is expensive, we pre-simulate our data on a time grid, store it in memory and then access the relevant data during training.

We present the network architectures and hyper-parameters corresponding to the $T = 5$ iDAD results shown in Table 5 of the main text. For the encoder of the design-outcome pairs we used:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	ξ, y	2	2	-
H1	Fully connected	8	8	ReLU
H2	Fully connected	64	64	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-

The resulting representations, $\{R(h_t)\}_{t=1}^{T-1}$, are stacked into a matrix (as new design–outcome pairs are obtained) and fed into an emitter network, which contains an LSTM cell with two hidden layers. We only keep the last hidden state of the LSTM's output and pass it through a final FC layer:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	$\{R(h_t)\}_{t=1}^{T-1}$	$32 \times$ t	$32 \times$ t	-
H1 & H2	LSTM	32	32	-
H3	Fully connected	16	16	ReLU
Output	Fully connected	1	1	-

The architecture of the critic network U_ψ used in Table 5 and all its hyper-parameters are in the tables that follow. First, the encoder network of the latent variables is:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	θ	2	2	-
H1	Fully connected	8	8	ReLU
H2	Fully connected	64	64	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-

For the design-outcome pairs encoder we use the same architecture as in the design network, namely:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	ξ, y	2	2	-
H1	Fully connected	8	8	ReLU
H2	Fully connected	64	64	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-

The outputs of the encoder, $\{R(h_t)\}_t$, are stacked and fed into an LSTM cell with two hidden layers. We only keep the last hidden state of the LSTM's output and pass it through a FC layer:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	$\{R(h_t)\}_{t=1}^{T-1}$	$32 \times$ t	$32 \times$ t	-
H1 & H2	LSTM	32	32	-
H3	Fully connected	16	16	ReLU
Output	Fully connected	32	32	-

The optimization was performed with Adam [26] with learning rate annealing with the following hyper-parameters:

Parameter	iDAD InfoNCE	iDAD, NWJ
Batch size	512	512
Number of contrastive/negative samples	511	511
Number of gradient steps	100000	100000
Initial learning rate (LR)	0.0005	0.0005
LR annealing factor	0.96	0.96
LR annealing frequency	1000	1000

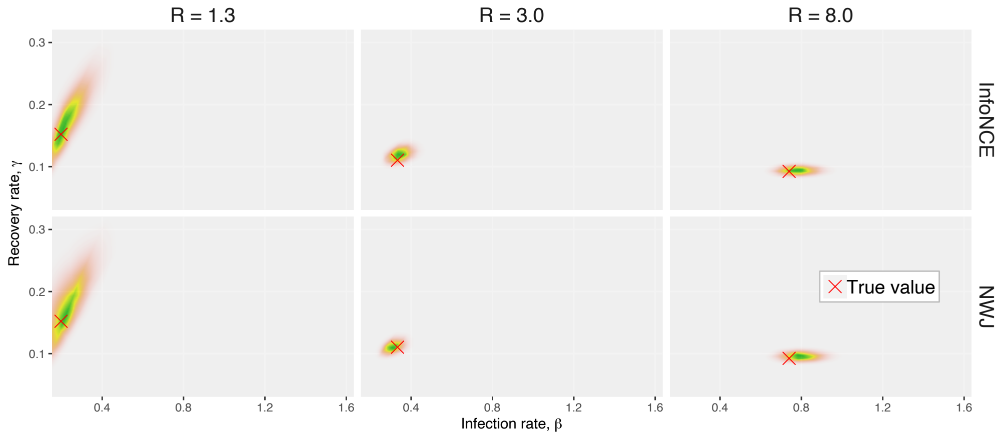


Figure 8: Approximate posteriors for the SIR model.

D.6.2 Further results

Different number of experiments T. In Table 11 we show lower bound estimates when applying iDAD with the InfoNCE lower bound to the SDE-based SIR model for different number of measurements T. The design network and critic architectures are the same as for $T = 5$. Table 11 shows that more measurements yield higher expected information gains, as one might intuitively expect. Furthermore, the increase in expected information gain saturates with increasing T, which is why we presented the results for $T = 5$ in the main text. The biggest increase, however, occurs from $T = 1$ to $T = 2$. This is intuitive, because the SIR model has two model parameters that we wish to estimate but we only gather one data point with one measurement. Hence, in order to accurately estimate both of these parameters, we would need at least 2 measurements, which is reflected in Table 11. We note that all of these numbers, with the exception of $T = 1$, are larger than those found by [29]. This increase in expected information gain may be explained by the fact that [29] use an additional Poisson observation model, which means that the resulting data are inherently noisier and less informative.

Table 11: InfoNCE lower bound estimates ($\pm$ s.e.) when applying iDAD to the SDE-based SIR model for different number of measurements T.

T	iDAD, InfoNCE	iDAD, NWJ
1	1.396 ± 0.018	1.417 ± 0.001
2	2.714 ± 0.019	2.699 ± 0.001
3	3.554 ± 0.021	3.515 ± 0.001
4	3.600 ± 0.018	3.749 ± 0.001
5	3.915 ± 0.020	3.869 ± 0.001
7	4.027 ± 0.019	3.911 ± 0.001
10	4.100 ± 0.020	4.019 ± 0.001

Training stability. To assess the robustness of the results and the stability of the training process, we trained 5 additional iDAD networks with each of the two bounds, using different seeds but the same hyperparameters we used to produce the results of Table 5 in the main text. We report upper and lower bounds on the mutual information along with their mean and standard error in the table below.

Method	Bound	Run 1	Run 2	Run 3	Run 4	Run 5	**Mean**	**SE**
iDAD, InfoNCE	Lower	3.900	3.919	3.919	3.901	3.887	**3.906**	**0.007**
iDAD, NWJ	Lower	3.872	3.838	3.854	3.883	3.848	**3.859**	**0.009**

We repeat the same procedure for the static baselines. The results reported in the table below demonstrate the training stability of these baselines as well.

Method	Bound	Run 1	Run 2	Run 3	Run 4	Run 5	**Mean**	**SE**
SG-BOED	Lower	3.713	3.765	3.767	3.764	3.739	**3.749**	**0.012**
MINEBED	Lower	3.373	3.438	3.376	3.379	3.420	**3.397**	**0.015**

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] Claims we make match theoretical and experimental results. Contributions and the overarching assumptions (e.g. types of models we consider) are clearly stated in the introduction.
   (b) Did you describe the limitations of your work? [Yes] Limitations are discussed throughout the paper (e.g. Section 3 discusses limitations of the bounds used) and in the conclusion.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] We thought about this issue and did not establish potential negative societal impact, ethical or environmental harm that our work could be a cause of.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] Detailed proofs are provided in the Appendix which also includes details on how our results relate to previous results in the field.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] In the supplemental material and includes instructions how to set-up an environment to reproduce the results.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] In the appendix.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] In the appendix, both for our method and baselines.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] Resources used are described in the appendix, together with approximate time required to train a model.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] Deep learning and probabilistic programming frameworks that were used for the experimental part of this work were cited. Details on versions used are available in the appendix.
   (b) Did you mention the license of the assets? [Yes] In appendix along with the details on computational resources.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Link to github repository with the code was provided.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] No actual data was used. All experiments were done using simulators.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

References

[1] Edward J. Allen, Linda J. S. Allen, Armando Arciniega, and Priscilla E. Greenwood. Construction of equivalent stochastic differential equation models. *Stochastic Analysis and Applications*, 26(2):274–297, 2008.

[2] Linda J.S. Allen. A primer on stochastic epidemic models: Formulation, numerical simulation, and analysis. *Infectious Disease Modelling*, 2(2):128–142, 2017. ISSN 2468-0427. doi: https://doi.org/10.1016/j.idm.2017.03.001.

[3] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *arXiv preprint arXiv:1906.00910*, 2019.

[4] David Barber and Felix Agakov. The im algorithm: A variational approach to information maximization. In *Proceedings of the 16th International Conference on Neural Information Processing Systems*, NIPS'03, page 201–208, Cambridge, MA, USA, 2003. MIT Press.

[5] Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of machine learning research*, 18, 2018.

[6] Eli Bingham, Jonathan P Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D Goodman. Pyro: Deep universal probabilistic programming. *Journal of Machine Learning Research*, 2018.

[7] Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, pages 273–304, 1995.

[8] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 13–18 Jul 2020.

[9] Yanzhi Chen, Dinghuai Zhang, Michael U. Gutmann, Aaron Courville, and Zhanxing Zhu. Neural approximate sufficient statistics for implicit models. In *International Conference on Learning Representations*, 2021.

[10] Alex R Cook, Gavin J Gibson, and Christopher A Gilligan. Optimal observation times in experimental epidemic processes. *Biometrics*, 64(3):860–868, 2008.

[11] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 2020.

[12] Mahasen B. Dehideniya, Christopher C. Drovandi, and James M. McGree. Optimal bayesian design for discriminating between models with intractable likelihoods in epidemiology. *Computational Statistics & Data Analysis*, 124:277–297, 2018. ISSN 0167-9473. doi: https://doi.org/10.1016/j.csda.2018.03.004.

[13] Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 4171–4186, 2019.

[14] Yann Dubois, Jonathan Gordon, and Andrew YK Foong. Neural Process Family. http://yanndubs.github.io/Neural-Process-Family/, September 2020.

[15] Adam Foster, Martin Jankowiak, Elias Bingham, Paul Horsfall, Yee Whye Teh, Thomas Rainforth, and Noah Goodman. Variational Bayesian Optimal Experimental Design. In *Advances in Neural Information Processing Systems 32*, pages 14036–14047. Curran Associates, Inc., 2019.

[16] Adam Foster, Martin Jankowiak, Matthew O'Meara, Yee Whye Teh, and Tom Rainforth. A unified stochastic gradient approach to designing bayesian-optimal experiments. In *International Conference on Artificial Intelligence and Statistics*, pages 2959–2969. PMLR, 2020.

[17] Adam Foster, Desi R Ivanova, Ilyas Malik, and Tom Rainforth. Deep adaptive design: Amortizing sequential bayesian experimental design. *Proceedings of the 38th International Conference on Machine Learning (ICML), PMLR 139*, 2021.

[18] Andrew Gelman, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2013.

[19] Matthew Graham and Amos Storkey. Asymptotically exact inference in differentiable generative models. In *Artificial Intelligence and Statistics*, pages 499–508. PMLR, 2017.

[20] PeterJ. Green, Krzysztof Latuszynski, Marcelo Pereyra, and Christian P. Robert. Bayesian computation: a summary of the current state, and samples backwards and forwards. *Statistics and Computing*, 25(4):835–862, 2015. doi: 10.1007/s11222-015-9574-5.

[21] Markus Hainy, Christopher C. Drovandi, and James M. McGree. Likelihood-free extensions for Bayesian sequentially designed experiments. In Joachim Kunert, Christine H. Müller, and Anthony C. Atkinson, editors, *mODa 11 - Advances in Model-Oriented Design and Analysis*, pages 153–161. Springer International Publishing, 2016.

[22] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

[23] Xun Huan and Youssef Marzouk. Gradient-based stochastic optimization methods in bayesian experimental design. *International Journal for Uncertainty Quantification*, 4(6), 2014.

[24] Xun Huan and Youssef M Marzouk. Simulation-based optimal bayesian experimental design for nonlinear systems. *Journal of Computational Physics*, 232(1):288–317, 2013.

[25] Cheng Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Noam Shazeer, Ian Simon, Curtis Hawthorne, Andrew M. Dai, Matthew D. Hoffman, Monica Dinculescu, and Douglas Eck. Music transformer: Generating music with long-term structure, 2019. ISSN 23318422.

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[27] S. Kleinegesse and M.U. Gutmann. Efficient Bayesian experimental design for implicit models. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 89 of *Proceedings of Machine Learning Research*, pages 1584–1592. PMLR, 2019.

[28] Steven Kleinegesse and Michael Gutmann. Bayesian experimental design for implicit models by mutual information neural estimation. In *Proceedings of the 37th International Conference on Machine Learning*, Proceedings of Machine Learning Research, pages 5316–5326. PMLR, 2020.

[29] Steven Kleinegesse and Michael U. Gutmann. Gradient-based bayesian experimental design for implicit models using mutual information lower bounds. *arXiv preprint arXiv:2105.04379*, 2021.

[30] Steven Kleinegesse, Christopher Drovandi, and Michael U. Gutmann. Sequential Bayesian Experimental Design for Implicit Models via Mutual Information. *Bayesian Analysis*, pages 1 – 30, 2021. doi: 10.1214/20-BA1225.

[31] Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*, 2019.

[32] Dennis V Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, pages 986–1005, 1956.

[33] J. Lintusaari, M.U. Gutmann, R. Dutta, S. Kaski, and J. Corander. Fundamentals and recent developments in approximate Bayesian computation. *Systematic Biology*, 66(1):e66–e82, January 2017.

[34] David McAllester and Karl Stratos. Formal limitations on the measurement of mutual information. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 875–884. PMLR, 2020.

[35] Shakir Mohamed, Mihaela Rosca, Michael Figurnov, and Andriy Mnih. Monte carlo gradient estimation in machine learning. *Journal of Machine Learning Research*, 21(132):1–62, 2020.

[36] Jay I Myung, Daniel R Cavagnaro, and Mark A Pitt. A tutorial on adaptive design optimization. *Journal of mathematical psychology*, 57(3-4):53–67, 2013.

[37] Xuanlong Nguyen, Martin J. Wainwright, and Michael I. Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11), 2010. ISSN 00189448. doi: 10.1109/TIT.2010.2068870.

[38] Antony Overstall and James McGree. Bayesian Design of Experiments for Intractable Likelihood Models Using Coupled Auxiliary Models and Multivariate Emulation. *Bayesian Analysis*, 15(1):103 – 131, 2020. doi: 10.1214/19-BA1144.

[39] Emilio Parisotto, Francis Song, Jack Rae, Razvan Pascanu, Caglar Gulcehre, Siddhant Jayakumar, Max Jaderberg, Raphaël Lopez Kaufman, Aidan Clark, Seb Noury, Matthew Botvinick, Nicolas Heess, and Raia Hadsell. Stabilizing transformers for reinforcement learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 7487–7498. PMLR, 2020.

[40] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *International Conference on Machine Learning*, pages 4055–4064. PMLR, 2018.

[41] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[42] Ben Poole, Sherjil Ozair, Aäron van den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In *International Conference on Machine Learning*, pages 5171–5180, 2019.

[43] David J. Price, Nigel G. Bean, Joshua V. Ross, and Jonathan Tuke. On the efficient determination of optimal bayesian experimental designs using abc: A case study in optimal observation of epidemics. *Journal of Statistical Planning and Inference*, 172:1–15, May 2016.

[44] David J Price, Nigel G Bean, Joshua V Ross, and Jonathan Tuke. An induced natural selection heuristic for finding optimal bayesian experimental designs. *Computational Statistics & Data Analysis*, 126:112–124, 2018.

[45] Tom Rainforth. *Automating Inference, Learning, and Design using Probabilistic Programming*. PhD thesis, University of Oxford, 2017.

[46] Tom Rainforth, Rob Cornish, Hongseok Yang, Andrew Warrington, and Frank Wood. On nesting monte carlo estimators. In *International Conference on Machine Learning*, pages 4267–4276. PMLR, 2018.

[47] Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jon Shlens. Stand-alone self-attention in vision models. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[48] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32, pages 1278–1286, 2014.

[49] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[50] Elizabeth G. Ryan, Christopher C. Drovandi, M. Helen Thompson, and Anthony N. Pettitt. Towards bayesian experimental design for nonlinear models that require a large number of sampling times. *Computational Statistics & Data Analysis*, 70:45–60, 2014. ISSN 0167-9473. doi: https://doi.org/10.1016/j.csda.2013.08.017.

[51] Elizabeth G Ryan, Christopher C Drovandi, James M McGree, and Anthony N Pettitt. A review of modern computational algorithms for bayesian optimal design. *International Statistical Review*, 84(1):128–154, 2016.

[52] Ben Shababo, Brooks Paige, Ari Pakman, and Liam Paninski. Bayesian inference and online experimental design for mapping neural microcircuits. In *Advances in Neural Information Processing Systems*, pages 1304–1312, 2013.

[53] Xiaohong Sheng and Yu Hen Hu. Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks. *IEEE Transactions on Signal Processing*, 2005. ISSN 1053587X. doi: 10.1109/TSP.2004.838930.

[54] S.A. Sisson, Y. Fan, and M. Beaumont. *Handbook of Approximate Bayesian Computation*. Chapman & Hall/CRC Handbooks of Modern Statistical Methods. CRC Press, 2018. ISBN 9781351643467.

[55] Jiaming Song and Stefano Ermon. Understanding the limitations of variational mutual information estimators. In *International Conference on Learning Representations*, 2020.

[56] Owen Thomas, Ritabrata Dutta, Jukka Corander, Samuel Kaski, and Michael U Gutmann. Likelihood-free inference by ratio estimation. *arXiv preprint arXiv:1611.10242*, 2016.

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

[58] Joep Vanlier, Christian A Tiemann, Peter AJ Hilbers, and Natal AW van Riel. A Bayesian approach to targeted experiment design. *Bioinformatics*, 28(8):1136–1142, 2012.

[59] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[60] Benjamin T Vincent and Tom Rainforth. The DARC toolbox: automated, flexible, and efficient delayed and risky choice experiments using bayesian adaptive design. *PsyArXiv preprint*, 2017.

[61] M. Zaharia, Andrew Chen, A. Davidson, A. Ghodsi, S. Hong, A. Konwinski, Siddharth Murching, Tomas Nykodym, Paul Ogilvie, Mani Parkhe, Fen Xie, and Corey Zumar. Accelerating the machine learning lifecycle with MLflow. *IEEE Data Eng. Bull.*, 41:39–45, 2018.

[62] Manzil Zaheer, Satwik Kottur, Siamak Ravanbhakhsh, Barnabás Póczos, Ruslan Salakhutdinov, and Alexander J Smola. Deep sets. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, 2017.

[63] Jiaxin Zhang, Sirui Bi, and Guannan Zhang. A stochastic approximate gradient ascent method for Bayesian experimental design with implicit models. In *The 24nd International Conference on Artificial Intelligence and Statistics*, 2021.