

An approach for finding fully Bayesian optimal designs using normal-based approximations to loss functions

Antony M. Overstall¹ · James M. McGree² · Christopher C. Drovandi²

Received: 20 August 2016 / Accepted: 5 February 2017 / Published online: 20 February 2017
© The Author(s) 2017. This article is published with open access at Springerlink.com

Abstract The generation of decision-theoretic Bayesian optimal designs is complicated by the significant computational challenge of minimising an analytically intractable expected loss function over a, potentially, high-dimensional design space. A new general approach for approximately finding Bayesian optimal designs is proposed which uses computationally efficient normal-based approximations to posterior summaries to aid in approximating the expected loss. This new approach is demonstrated on illustrative, yet challenging, examples including hierarchical models for blocked experiments, and experimental aims of parameter estimation and model discrimination. Where possible, the results of the proposed methodology are compared, both in terms of performance and computing time, to results from using computationally more expensive, but potentially more accurate, Monte Carlo approximations. Moreover, the methodology is also applied to problems where the use of Monte Carlo approximations is computationally infeasible.

Keywords Loss function · Model discrimination · Bayesian optimal design · Parameter estimation · Hierarchical model

1 Introduction

The process of designing a physical experiment fits naturally within the Bayesian approach to statistical inference. Prior information on parameters and models can be represented by prior distributions, and the experimental aim encapsulated in a decision-theoretic framework by the loss function. A Bayesian optimal design is found by minimising the expected loss function over the space of all possible designs, i.e. the design space, where the expectation is with respect to the joint distribution of all unknown quantities, i.e. parameters, models and experimental responses.

Formally, suppose the experiment consists of n runs where the i th run (for $i = 1, \dots, n$) involves measuring response y_i having specified settings $\mathbf{d}_i = (d_{i1}, \dots, d_{ik})$ for the k controllable factors. Let $\mathbf{d} = (\mathbf{d}_1, \dots, \mathbf{d}_n) \in \mathcal{D}$ be the $W \times 1$ vector giving the design where $W = nk$ and $\mathcal{D} \subset \mathbb{R}^W$ denotes the W -dimensional design space. Assume there is a set, \mathcal{M} , of competing statistical models. Model $m \in \mathcal{M}$ posits a probability distribution, F_m , for $\mathbf{y} = (y_1, \dots, y_n)$ which is completely specified up to an unknown $p_m \times 1$ vector of parameters $\boldsymbol{\theta}_m$. Bayesian inference on models and/or parameters is based on their joint posterior distribution given by Bayes' theorem as

$$\pi(\boldsymbol{\theta}_m, m | \mathbf{y}, \mathbf{d}) \propto \pi(\mathbf{y} | \boldsymbol{\theta}_m, m, \mathbf{d}) \pi(\boldsymbol{\theta}_m | m) \pi(m), \quad (1)$$

where $\pi(\mathbf{y} | \boldsymbol{\theta}_m, m, \mathbf{d})$ is the probability mass/density function of F_m ; $\pi(\boldsymbol{\theta}_m | m)$ is the probability density function of the prior distribution of $\boldsymbol{\theta}_m$; and $\pi(m)$ is the prior model probability of model m . Note how $\pi(\mathbf{y} | \boldsymbol{\theta}_m, m, \mathbf{d})$ depends on the design and this induces a dependence of the posterior on the design.

The aim of the experiment is represented by a loss function, which can be tailored to experimental aims of parameter

✉ Antony M. Overstall
A.M.Overstall@soton.ac.uk

James M. McGree
james.mcgree@qut.edu.au

Christopher C. Drovandi
c.drovandi@qut.edu.au

¹ Southampton Statistical Sciences Research Institute,
University of Southampton, Southampton, UK

² School of Mathematical Sciences, Queensland University of
Technology, Brisbane, Queensland, Australia

estimation or model discrimination. In general, the loss function is given by $\lambda(\theta_m, m, \mathbf{y}, \mathbf{d})$. Essentially, it compares a summary of the posterior distribution (e.g. O’Hagan and Forster 2004, pgs 13–14) for θ_m and m (conditional on \mathbf{y} and \mathbf{d}) to the true values of θ_m and m . However, θ_m, m and \mathbf{y} are unknown so a Bayesian optimal design (e.g. Chaloner and Verdinelli 1995) is found by minimising the expected loss

$$L(\mathbf{d}) = E_{\theta_m, m, \mathbf{y} | \mathbf{d}} [\lambda(\theta_m, m, \mathbf{y}, \mathbf{d})] \tag{2}$$

over the design space, \mathcal{D} , where the expectation in (2) is with respect to the joint distribution of θ_m, m and \mathbf{y} .

Robert (2007, Section 2.1) discusses how Bayesian inference relies on the specification of three components: (a) the set of models, \mathcal{M} ; (b) the joint prior distribution, given by $\pi(\theta_m, m)$; and (c) the loss function, $\lambda(\theta_m, m, \mathbf{y}, \mathbf{d})$. Notwithstanding the difficulties faced in specifying these three components (e.g. O’Hagan and Forster 2004, Chapter 6), once in place, a Bayesian optimal design is conceptually straightforward to define. However, finding such a design in practice is hindered by the computational challenge of minimising the expected loss. Typically, the expected loss is given by a multi-dimensional analytically intractable integral, i.e. as given by (2). In the two decades since the seminal review of Bayesian design by Chaloner and Verdinelli (1995), there have been few general-purpose approaches to finding such designs, as recently highlighted by Ryan et al. (2016) and Woods et al. (2017). Note that being able to find the exact optimal design for an arbitrary problem is, at present, an unrealistic goal. Instead, the aim is to find a design “close” to the optimal design, termed a near-optimal design by Hamada et al. (2001).

Existing approaches to approximately finding Bayesian designs can be divided into two broad strategies. First, the simulation-based approach of Müller (1999) arranges a joint distribution for θ_m, m, \mathbf{y} and design \mathbf{d} given by

$$h(\theta_m, m, \mathbf{y}, \mathbf{d}) \propto (c - \lambda(\theta_m, m, \mathbf{y}, \mathbf{d})) \times \pi(\mathbf{y} | \theta_m, m, \mathbf{d}) \pi(\theta_m | m) \pi(m), \tag{3}$$

where $c \geq \sup \lambda(\theta_m, m, \mathbf{y}, \mathbf{d})$. The marginal mode of \mathbf{d} corresponds to the Bayesian optimal design. The so-called Müller algorithm essentially proceeds by using simulation methods to generate a sample from the joint distribution of θ_m, m, \mathbf{y} and \mathbf{d} and to use this sample to estimate the marginal mode of \mathbf{d} . This approach has been further modified by Müller et al. (2004) and Amzal et al. (2006). However, due to the difficulty in implementing efficient sampling methods, the Müller algorithm is difficult to implement for high-dimensional design spaces with the limit typically considered to be just $W = 4$ (e.g. Ryan et al. 2016).

Alternatively, the second broad strategy is the smoothing-based approach reliant on the following Monte Carlo approximation to the expected loss

$$\hat{L}(\mathbf{d}) = \frac{1}{B} \sum_{b=1}^B \lambda(\theta_m^b, m^b, \mathbf{y}^b, \mathbf{d}), \tag{4}$$

where $\{\theta_m^b, m^b, \mathbf{y}^b\}_{b=1}^B$ is a sample generated from the joint distribution θ_m, m and \mathbf{y} . The stochastic nature of Monte Carlo approximation means that $\hat{L}(\mathbf{d})$ is not a smooth function and makes application of standard optimisation methods (including heuristic methods) difficult. Instead, Müller and Parmigiani (1995) proposed an approach whereby $\hat{L}(\mathbf{d})$ is evaluated at a series of designs. A statistical model (or smoother) is then fitted which builds a relationship between design and expected loss allowing prediction of the expected loss for any design. This prediction is minimised in place of the true expected loss to give an approximation to the Bayesian optimal design. Similar to the simulation-based Müller algorithm, the scalability of this approach to higher dimensional design spaces remains an issue. The chosen smoother needs to balance the increased flexibility required for adequate predictive accuracy with the computational effort of the increased number of evaluations of $\hat{L}(\mathbf{d})$ required. Müller and Parmigiani (1995) employed local regression models and considered design spaces up to $W = 2$. More recently, Weaver et al. (2016) used a Gaussian process model (e.g. Santner et al. 2003) and considered an application with $W = 3$ and Jones et al. (2016) used Bayes linear analysis (Goldstein and Wooff 2007) and considered up to $W = 9$. To increase the applicability to higher dimensional design spaces, Overstall and Woods (2017) proposed the approximate coordinate exchange (ACE) algorithm whereby a cyclic descent algorithm (commonly called coordinate exchange in the design of experiments literature, Meyer and Nachtsheim 1995) is used to minimise the expected loss. Very briefly, a Gaussian process prediction of the expected loss is sequentially minimised over each one-dimensional element of the design space. This can be seen as a generalisation of the approaches of Müller and Parmigiani (1995) and Weaver et al. (2016) to higher dimensional design spaces via the use of coordinate exchange, and thus allowed consideration of examples with design spaces of dimensionality nearly two orders of magnitude greater than previously addressed in the literature.

Both the Müller algorithm and the smoothing-based approaches require evaluations of the loss function $\lambda(\theta_m, m, \mathbf{y}, \mathbf{d})$ either in the evaluation of $h(\theta_m, m, \mathbf{y}, \mathbf{d})$ or $\hat{L}(\mathbf{d})$, respectively. In both cases, a large number of evaluations of the loss function will be required to find a design. Since the loss function depends on \mathbf{y} through the posterior distribution of θ_m and m , and given that this distribution will typically be analytically intractable, a further approximation is required.

The most obvious approach to this problem is to use an additional Monte Carlo approximation, the exact nature of which depends on the chosen loss function. In the case of the Monte Carlo approximation to the expected loss, $\hat{L}(\mathbf{d})$, given by (4), this will result in a nested or double loop Monte Carlo (DLMC) approximation to the expected loss where the inner loop refers to the approximation to the loss function and the outer loop to the approximation to the expected loss. Let \tilde{B} denote the Monte Carlo sample size in the inner loop with B being the corresponding value in the outer loop. For typical loss functions, DLMC can induce a bias of order \tilde{B}^{-1} in the approximation (e.g. Ryan 2003; Rainforth et al. 2016) to the expected loss. Moreover, the computational complexity of this approach is typically in the order of $B \times \tilde{B}$ evaluations of $\pi(\mathbf{y}|\theta_m, m, \mathbf{d})$ for each $m \in \mathcal{M}$. Since B and \tilde{B} will typically be $\mathcal{O}(10^3)$ or higher, this will be a computationally expensive approach. The result of which is that, even by using the ACE algorithm, finding Bayesian optimal designs with the DLMC approximation to the expected loss has been confined to simple problems where the number of models under consideration is $|\mathcal{M}| = 1$ and inference has been focused on parameter estimation.

In this paper, we consider using normal-based approximations to the posterior distribution of θ_m , centred around the posterior mode of θ_m , and how these can be used to approximate, in principle, any loss function. The normal-based Laplace approximation has previously been used to approximate the commonly used self-information loss function (see Sect. 3.1) for a nonlinear model by Long et al. (2013) for a low-dimensional design space and under no model uncertainty. However, application to other loss functions has not previously been considered. We apply the new methodology to illustrative examples which are challenging in the context of Bayesian optimal design. In cases where the computationally more expensive DLMC approach is feasible, we show, empirically, that the difference in performance (measured in terms of expected loss) between designs found under the two different approximations is negligible. We also apply the proposed approach to problems where use of the DLMC approximation to find a design would be computationally infeasible.

2 Methodology

2.1 Normal-based approximations to posterior quantities

As discussed in Sect. 1, a typical loss function compares a summary of the joint posterior distribution of θ_m and m to the “true” values of θ_m and m in a way that is relevant to the experimental aim. However, the joint posterior distribution is usually not available in closed form. The methodology

in this paper is based on forming an approximation to this joint distribution using normal-based approximations and therefore providing an alternative approximation to the loss function than the Monte Carlo approximation. The normal-based approximation to the loss function, denoted by $\tilde{\lambda}(\theta_m, m, \mathbf{y}, \mathbf{d})$, can then be substituted into the Monte Carlo approximation to the expected loss, given by (4), and we refer to such an approximation as normal-based Monte Carlo (NBMC). We can then use the ACE algorithm or one of the other smoothing-based approaches discussed in Sect. 1. Conversely, we may also substitute $\tilde{\lambda}(\theta_m, m, \mathbf{y}, \mathbf{d})$ into the density of the joint distribution over θ_m, m, \mathbf{y} and \mathbf{d} , given by (3), and apply the Müller algorithm.

First, note that the joint posterior distribution of θ_m and m , given by (1), can be decomposed as follows (e.g. O’Hagan and Forster 2004, page 169)

$$\pi(\theta_m, m|\mathbf{y}, \mathbf{d}) = \pi(\theta_m|\mathbf{y}, m, \mathbf{d})\pi(m|\mathbf{y}, \mathbf{d}),$$

where the posterior model probability of model $m \in \mathcal{M}$ is given by

$$\pi(m|\mathbf{y}, \mathbf{d}) = \frac{\pi(\mathbf{y}|m, \mathbf{d})\pi(m)}{\sum_{m \in \mathcal{M}} \pi(\mathbf{y}|m, \mathbf{d})\pi(m)}, \tag{5}$$

and the posterior distribution of θ_m (conditional on model $m \in \mathcal{M}$) is given by

$$\pi(\theta_m|\mathbf{y}, m, \mathbf{d}) = \frac{\pi(\mathbf{y}|\theta_m, m, \mathbf{d})\pi(\theta_m|m)}{\pi(\mathbf{y}|m, \mathbf{d})}, \tag{6}$$

with

$$\pi(\mathbf{y}|m, \mathbf{d}) = \int_{\Theta_m} \pi(\mathbf{y}|\theta_m, m, \mathbf{d})\pi(\theta_m|m) d\theta_m, \tag{7}$$

usually called the marginal likelihood or evidence (Friel and Wyse 2012) for model $m \in \mathcal{M}$.

The posterior model probabilities are completely determined by the marginal likelihoods. Therefore, for cases where there is model uncertainty, i.e. $|\mathcal{M}| > 1$, it will be necessary to approximate $\pi(\mathbf{y}|m, \mathbf{d})$ for all $m \in \mathcal{M}$. First, let the posterior mode for model $m \in \mathcal{M}$ be denoted by $\hat{\theta}_m(\mathbf{y}) = \arg \max_{\theta_m \in \Theta_m} \pi(\mathbf{y}|\theta_m, m, \mathbf{d})\pi(\theta_m|m)$. Now let

$$\hat{\Sigma}_m(\mathbf{y}) = \mathbf{H}(\hat{\theta}_m(\mathbf{y}); m)^{-1},$$

where

$$\begin{aligned} \mathbf{H}(\theta_m; m) &= E_{\mathbf{y}|\theta_m, m} \left[\frac{\partial \log \pi(\mathbf{y}|\theta_m, m, \mathbf{d})}{\partial \theta_m} \frac{\partial \log \pi(\mathbf{y}|\theta_m, m, \mathbf{d})}{\partial \theta_m^T} \right] \\ &\quad - \frac{\partial^2 \log \pi(\theta_m|m)}{\partial \theta_m \partial \theta_m^T}, \end{aligned}$$

i.e. the Fisher information matrix minus the second derivative of the log prior density. A second-order Taylor series expansion of the log integrand in (7) about the posterior mode yields the following so-called Laplace approximation (e.g. Gelman et al. 2014; Long et al. 2013) to the marginal likelihood for model $m \in \mathcal{M}$

$$\begin{aligned} \tilde{\pi}(\mathbf{y}|m, \mathbf{d}) &= (2\pi)^{\frac{p_m}{2}} |\hat{\Sigma}_m(\mathbf{y})|^{-\frac{1}{2}} \pi(\mathbf{y}|\hat{\boldsymbol{\theta}}_m(\mathbf{y}), m, \mathbf{d}) \pi(\hat{\boldsymbol{\theta}}_m(\mathbf{y})|m, \mathbf{d}). \end{aligned} \quad (8)$$

The posterior model probabilities are now approximated via

$$\tilde{\pi}(m|\mathbf{y}, \mathbf{d}) = \frac{\tilde{\pi}(\mathbf{y}|m, \mathbf{d})\pi(m)}{\sum_{m \in \mathcal{M}} \tilde{\pi}(\mathbf{y}|m, \mathbf{d})\pi(m)}. \quad (9)$$

Furthermore, we approximate the posterior distribution of $\boldsymbol{\theta}_m$ (conditional on m), by a normal distribution with mean $\hat{\boldsymbol{\theta}}_m(\mathbf{y})$ and variance $\hat{\Sigma}_m(\mathbf{y})$, i.e.

$$N\left(\hat{\boldsymbol{\theta}}_m(\mathbf{y}), \hat{\Sigma}_m(\mathbf{y})\right). \quad (10)$$

The tractability of the normal distribution means there are now many direct approximations to posterior summaries of interest (e.g. O'Hagan and Forster 2004, page 237). For example, trivially, the posterior median of $\boldsymbol{\theta}_m$ conditional on m is approximated by the posterior mode. These approximations to posterior summaries of interest can then be used to approximate many loss functions of practical interest (see Sect. 3.1 for examples).

Since the posterior distribution of $\boldsymbol{\theta}_m$ converges to a normal distribution as $n \rightarrow \infty$ (e.g. Gelman et al. 2014, pp. 585–588) the approximation will be more accurate for large n . The approximation can be expected to be poor when the true posterior distribution of $\boldsymbol{\theta}_m$ is multi-modal or has significant skewness. In the latter case, the approximation could be improved by a reparameterisation (e.g. Achcar and Smith 1990).

2.2 Finding the posterior mode and Fisher information

The approximations above are reliant on finding the posterior mode, $\hat{\boldsymbol{\theta}}_m(\mathbf{y})$, for each $m \in \mathcal{M}$. This will need to be accomplished for each \mathbf{y}^b in the sample $\left\{\boldsymbol{\theta}_{m^b}^b, m^b, \mathbf{y}^b\right\}_{b=1}^B$, from the joint distribution of $\boldsymbol{\theta}_m, m$ and \mathbf{y} , to evaluate the NBMC approximation to the expected loss. To do this, we use a scoring algorithm (e.g. Lange 2013, pgs 254–257), i.e. Newton's method where evaluation of the Hessian matrix of the log posterior density is replaced by evaluation of $-\mathbf{H}(\boldsymbol{\theta}_m; m)$. Specifically, let

$$\mathbf{f}(\boldsymbol{\theta}_m; m) = \frac{\partial \log \pi(\mathbf{y}|\boldsymbol{\theta}_m, m, \mathbf{d})}{\partial \boldsymbol{\theta}_m} + \frac{\partial \log \pi(\boldsymbol{\theta}_m|m)}{\partial \boldsymbol{\theta}_m},$$

be the gradient of the log posterior density with respect to $\boldsymbol{\theta}_m$.

For $r = 0, 1, 2, \dots$, the scoring algorithm iterates through the following steps

$$\boldsymbol{\theta}_m^{(r+1)} = \boldsymbol{\theta}_m^{(r)} + \kappa \mathbf{H}(\boldsymbol{\theta}_m^{(r)}; m)^{-1} \mathbf{f}(\boldsymbol{\theta}_m^{(r)}; m),$$

for some $0 < \kappa \leq 1$, until convergence. In the examples in this paper, we use $\kappa = \frac{1}{4}$ and a starting value of $\boldsymbol{\theta}_m^{(0)} = E(\boldsymbol{\theta}_m|m)$, i.e. the prior mean. Convergence is deemed to have occurred when $\left(\boldsymbol{\theta}_m^{(r+1)} - \boldsymbol{\theta}_m^{(r)}\right)^T \left(\boldsymbol{\theta}_m^{(r+1)} - \boldsymbol{\theta}_m^{(r)}\right) < \epsilon$, where $\epsilon = 10^{-4}$.

The scoring algorithm requires the repeated inversion of the $p_m \times p_m$ matrix $\mathbf{H}(\boldsymbol{\theta}_m; m)$. Often experiments are conducted in blocks where a block consists of homogenous experimental units. A suitable model (e.g. Pawitan 2013, Chapter 17) in this case is a hierarchical (or mixed model) where the effect of the block is accounted for by using block-specific parameters (sometimes referred to as random effects) for each block, which are assumed to be independent having a common prior distribution. A consequence is that the number of parameters, p_m , for these types of models is proportional to the number of blocks and therefore can be large. However, due to the conditional independence structure exhibited by hierarchical models, $\mathbf{H}(\boldsymbol{\theta}_m; m)$ will be sparse leading to computationally efficient methods for finding the inverse.

3 Examples

In this section, we begin by discussing a range of exemplar loss functions and how they can be approximated using the approach described in Sect. 2. This selection of loss functions is not exhaustive but instead demonstrate how typical loss functions may be approximated by using the approximations outlined in Sect. 2. We then apply the proposed methodology to find designs for experiments involving standard and hierarchical logistic regression (Sect. 3.2) and a nonlinear model (Sect. 3.3), for experimental aims of parameter estimation and model discrimination. In the examples, we use the ACE algorithm (briefly described in Sect. 1) to find the optimal design since this is the only method in the literature suitable for finding Bayesian designs for the dimensionality of design space considered. However, the approximations to the loss function could be applied with any method such as the Müller algorithm or another smoothing-based method. A more detailed description of the ACE algorithm is provided in "Appendix 1" with a description of the choice of tuning parameters.

All designs found in Sects. 3.2 and 3.3 can be reproduced via the R (R Core Team 2016) package `NBdesigns` (Overstall et al. 2017a). This is available as a Supplementary Material to this paper. This package allows users to compare future computational methodology to the approaches described in this paper via the benchmark examples considered.

3.1 Loss functions

The loss functions considered in this section can be categorised into those for (a) parameter estimation (Sect. 3.1.1); and (b) model discrimination (Sect. 3.1.2).

3.1.1 Parameter estimation

Suppose interest lies in the $q \times 1$ vector of transformed parameters $\phi = g_m(\theta_m)$, where $q \leq \min_{m \in \mathcal{M}} p_m$, the g_m are a set of one-to-one and invertible functions, and ϕ has a consistent interpretation for all $m \in \mathcal{M}$. Inference about ϕ is based on the model-averaged posterior distribution (O’Hagan and Forster 2004, pp. 171–174) given by

$$\pi_\phi(\phi | \mathbf{y}, \mathbf{d}) = \sum_{m \in \mathcal{M}} \pi_\phi(\phi | m, \mathbf{y}, \mathbf{d}) \pi(m | \mathbf{y}, \mathbf{d}),$$

where we have introduced a subscript of ϕ on the density function to indicate it refers to the transformed parameters. Consider the following loss functions representing the experimental aim of estimating ϕ . In each case, a special case occurs when there is no model uncertainty, i.e. $|\mathcal{M}| = 1$, in which case we are interested in parameter estimation under a single model.

Self-information loss The self-information (SI) loss is

$$\lambda_{SI}(\phi, \mathbf{y}, \mathbf{d}) = \log \pi_\phi(\phi) - \log \pi_\phi(\phi | \mathbf{y}, \mathbf{d}). \tag{11}$$

Minimising the expected SI loss is equivalent to maximising the expected Shannon information gain (Lindley 1956) and expected Kullback-Liebler divergence between prior and posterior distributions. Note that the expected SI loss is non-positive. Typically, the density of the posterior distribution of ϕ , given by $\pi_\phi(\phi | \mathbf{y}, \mathbf{d})$, in the SI loss will be analytically intractable. However, we can approximate it by first approximating the posterior distribution of θ_m by (10) and then deriving the approximate distribution of $\phi = g_m(\theta_m)$ (conditional on m) after taking a first-order Taylor series expansion of $\phi = g_m(\theta_m)$ about $\hat{\theta}_m(\mathbf{y})$ (e.g. Khuri 2003, Chapter 4). This leads to the following approximation to $\pi_\phi(\phi | \mathbf{y}, \mathbf{d})$

$$\tilde{\pi}_\phi(\phi | \mathbf{y}, \mathbf{d}) = \sum_{m \in \mathcal{M}} \tilde{\pi}_\phi(\phi | m, \mathbf{y}, \mathbf{d}) \tilde{\pi}(m | \mathbf{y}, \mathbf{d}), \tag{12}$$

where $\tilde{\pi}(m | \mathbf{y}, \mathbf{d})$ is given by (9) and $\tilde{\pi}_\phi(\phi | m, \mathbf{y}, \mathbf{d})$ is the density of

$$N \left(g_m(\hat{\theta}_m(\mathbf{y})), \frac{\partial g_m(\theta_m)}{\partial \theta_m} \Big|_{\theta_m = \hat{\theta}_m(\mathbf{y})} \hat{\Sigma}_m \frac{\partial g_m(\theta_m)}{\partial \theta_m} \Big|_{\theta_m = \hat{\theta}_m(\mathbf{y})} \right). \tag{13}$$

The result is that the approximate model-averaged posterior distribution of ϕ given by (12) is a mixture of normal distributions where each component is given by (13) and weighted by $\tilde{\pi}(m | \mathbf{y}, \mathbf{d})$.

It may not always be possible to find a closed form for the density of the prior distribution of ϕ , evaluation of which is necessary for the calculation of the SI loss given by (11). In these cases, we suggest approximating the prior distribution of θ_m for each $m \in \mathcal{M}$ by a normal distribution (with mean $\mu_m = E(\theta_m | m)$ and variance $\Psi_m = \text{var}(\theta_m | m)$). Now, the prior density $\pi_\phi(\phi)$ can be approximated via

$$\tilde{\pi}_\phi(\phi) = \sum_{m \in \mathcal{M}} \tilde{\pi}_\phi(\phi | m) \pi(m),$$

where $\tilde{\pi}_\phi(\phi | m)$ is the density of $N(\mu_m, \Psi_m)$. Similar to the posterior distribution, the model-averaged prior distribution of ϕ is approximated by a mixture of normal distributions but where the weights are the true prior model probabilities.

It is well known (e.g. Chaloner and Verdinelli 1995) that in cases where there is no model uncertainty, $|\mathcal{M}| = 1$, $g(\theta) = \theta$, and under a normal approximation to the posterior distribution the expected SI loss is equal to the objective function that defines pseudo-Bayesian D-optimality (a commonly used criterion in classical optimal design of experiments). In ‘‘Appendix 2’’, we discuss the relationship between the normal-based approximation to the SI loss above and the pseudo-Bayesian D-optimal approximation. It is shown that the objective function for pseudo-Bayesian D-optimality is itself an approximation to the expectation of the normal-based approximation to the SI loss. This places the normal-based approximations as being a compromise between the computationally expensive DLMC approximation and the computationally cheap pseudo-Bayesian D-optimal approximation. Furthermore, in Sect. 3.2, we empirically compare the two approximations.

Absolute error loss The absolute error (AE) loss (e.g. Robert 2007, pp. 79–80) is given by

$$\lambda_{AE}(\phi, \mathbf{y}, \mathbf{d}) = \sum_{j=1}^q |\phi_j - Q(\phi_j | \mathbf{y}, \mathbf{d})|,$$

where $Q(\phi_j | \mathbf{y}, \mathbf{d})$ is the model-averaged posterior marginal median of ϕ_j . If there is no model uncertainty, i.e. $|\mathcal{M}| = 1$, then we can approximate the posterior median by $\tilde{Q}(\phi_j | \mathbf{y}, \mathbf{d}) = g_m(\hat{\theta}_m(\mathbf{y}))_j$. However, if $|\mathcal{M}| > 1$, then there is no closed

form for the median (or any quantile) of a mixture of normal distributions. To overcome this problem, we use simulation. We generate a sample $\{\phi^c\}_{c=1}^C$ from the approximate model-average posterior distribution of ϕ . We then approximate $Q(\phi_j|\mathbf{y}, \mathbf{d})$ by the corresponding sample median. The sample $\{\phi^c\}_{c=1}^C$ can be generated as follows

1. Generate $\{m^c\}_{c=1}^C$ where model m is chosen with probability $\tilde{\pi}(m|\mathbf{y}, \mathbf{d})$.
2. For $c = 1, \dots, C$, complete the following steps
 - (a) Generate θ_{m^c} from $N(\hat{\theta}_m^c(\mathbf{y}), \hat{\Sigma}_{m^c})$.
 - (b) Set $\phi^c = g_{m^c}(\theta_{m^c})$.

Squared error loss The squared error loss (e.g. Robert 2007, pp. 77–79) is given by

$$\lambda_{SE}(\phi, \mathbf{y}, \mathbf{d}) = (\phi - E_{\phi|\mathbf{y}, \mathbf{d}}[\phi])^T (\phi - E_{\phi|\mathbf{y}, \mathbf{d}}[\phi]),$$

where

$$E_{\phi|\mathbf{y}, \mathbf{d}}[\phi] = \sum_{m \in \mathcal{M}} E_{\theta_m|\mathbf{y}, m, \mathbf{d}}[g_m(\theta_m)] \pi(m|\mathbf{y}, \mathbf{d}). \tag{14}$$

Typically, unless the g_m are linear functions, the posterior mean $E_{\theta_m|\mathbf{y}, m, \mathbf{d}}[g_m(\theta_m)]$ will not be available in closed form. To approximate this posterior mean, we use the simulation approach described above for the case of approximating the posterior median, only replacing the sample median by sample mean.

3.1.2 Model discrimination

Now, suppose the experimental aim is model discrimination, and thus inference is based on the posterior model probabilities. Consider the following loss functions. In both cases, we can derive approximations by replacing the marginal likelihood, $\pi(\mathbf{y}|m, \mathbf{d})$, or posterior model probability, $\pi(m|\mathbf{y}, \mathbf{d})$ by the corresponding approximations, $\tilde{\pi}(\mathbf{y}|m, \mathbf{d})$ or $\tilde{\pi}(m|\mathbf{y}, \mathbf{d})$, given by (8) and (9), respectively.

0–1 loss The 0–1 loss (e.g. Robert 2007, pp. 80–81) is given by

$$\lambda_{01}(m, \mathbf{y}, \mathbf{d}) = 1 - I(m = M(m|\mathbf{y}, \mathbf{d})),$$

where $I(A)$ denotes the indicator of event A and $M(m|\mathbf{y}, \mathbf{d}) = \arg \max_{m \in \mathcal{M}} \pi(\mathbf{y}|m, \mathbf{d})\pi(m)$ is the posterior modal model. The design that minimises the expected 0–1 loss equivalently maximises the expected posterior model probability of the modal model.

Model self-information loss The model self-information loss (e.g. McGree 2017) is derived by extending the self-information loss for parameters to the posterior model probabilities. It is given by

$$\lambda_{MSI}(m, \mathbf{y}, \mathbf{d}) = \log \pi(m) - \log \pi(m|\mathbf{y}, \mathbf{d}).$$

3.2 Logistic regression

The setup for the following example is adapted from Overstall and Woods (2017) who correspondingly adapted it from a simpler problem studied by Woods et al. (2006) and Gotwalt et al. (2009). It concerns a first-order logistic regression model in $k = 4$ factors and n runs. Although from a Bayesian inference perspective, logistic regression is a relatively simple model, it (or more generally some type of binary response model) is frequently used to benchmark new computational approaches in statistics (e.g. Minka 2001; Girolami and Calderhead 2011; Hoffman and Gelman 2014). Moreover, until Overstall and Woods (2017), fully Bayesian design for such a model had not previously been attempted in the literature indicating that in the context of fully Bayesian design logistic regression remains a non-trivial problem.

A binary response is measured for G blocks each of $n_G = 6$ runs, i.e. the total number of runs is $n = Gn_G$. Let y_{ij} and x_{tij} denote the experimental response and value of the t th factor for the j th run from the i th block ($i = 1, \dots, G$; $j = 1, \dots, n_G$; $t = 1, \dots, 4$), respectively. It is assumed that $y_{ij} \sim \text{Bernoulli}(\rho_{ij})$, independently, where

$$\begin{aligned} \log\left(\frac{\rho_{ij}}{1 - \rho_{ij}}\right) &= \beta_0 + \gamma_{0i} + \sum_{t=1}^4 v_t (\beta_t + \gamma_{ti}) x_{tij}, \\ &= \mathbf{x}_{ij}^T (\mathbf{v} \circ (\boldsymbol{\beta} + \boldsymbol{\gamma}_i)), \end{aligned}$$

where $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2, \beta_3, \beta_4)$ are the regression parameters, $\boldsymbol{\gamma} = (\boldsymbol{\gamma}_1, \dots, \boldsymbol{\gamma}_G)$ are the block-specific parameters, $\mathbf{v} = (v_0, \dots, v_4)$ is a binary vector with $v_t = 1$ ($v_t = 0$) indicating whether the t th factor is active (inactive), and \circ denotes element-wise multiplication. The complete $p_m \times 1$ vector of parameters is $\boldsymbol{\theta}_m = (\boldsymbol{\beta}_m, \boldsymbol{\gamma}_m)$. Each model $m \in \mathcal{M}$ is determined by the $|\mathcal{M}| = 2^4 = 16$ combinations of \mathbf{v} .

Let $\mathbf{X} = (\mathbf{X}_1^T, \dots, \mathbf{X}_G^T)^T$ be the $n \times 5$ model matrix where \mathbf{X}_i is the $n_G \times 5$ model matrix for the i th group with j th row given by \mathbf{x}_{ij}^T . The design is given by $\mathbf{d} = \text{vec}(\mathbf{D}^T)$ where \mathbf{D} is the $n \times 4$ matrix given by \mathbf{X} with the first column of ones (corresponding to the intercept) removed. The design space \mathcal{D} has dimensionality $W = 4n$ and is such that each element of \mathbf{d} lies in the interval $[-1, 1]$.

From Woods et al. (2006) and Overstall and Woods (2017), we assume independent prior distributions for each

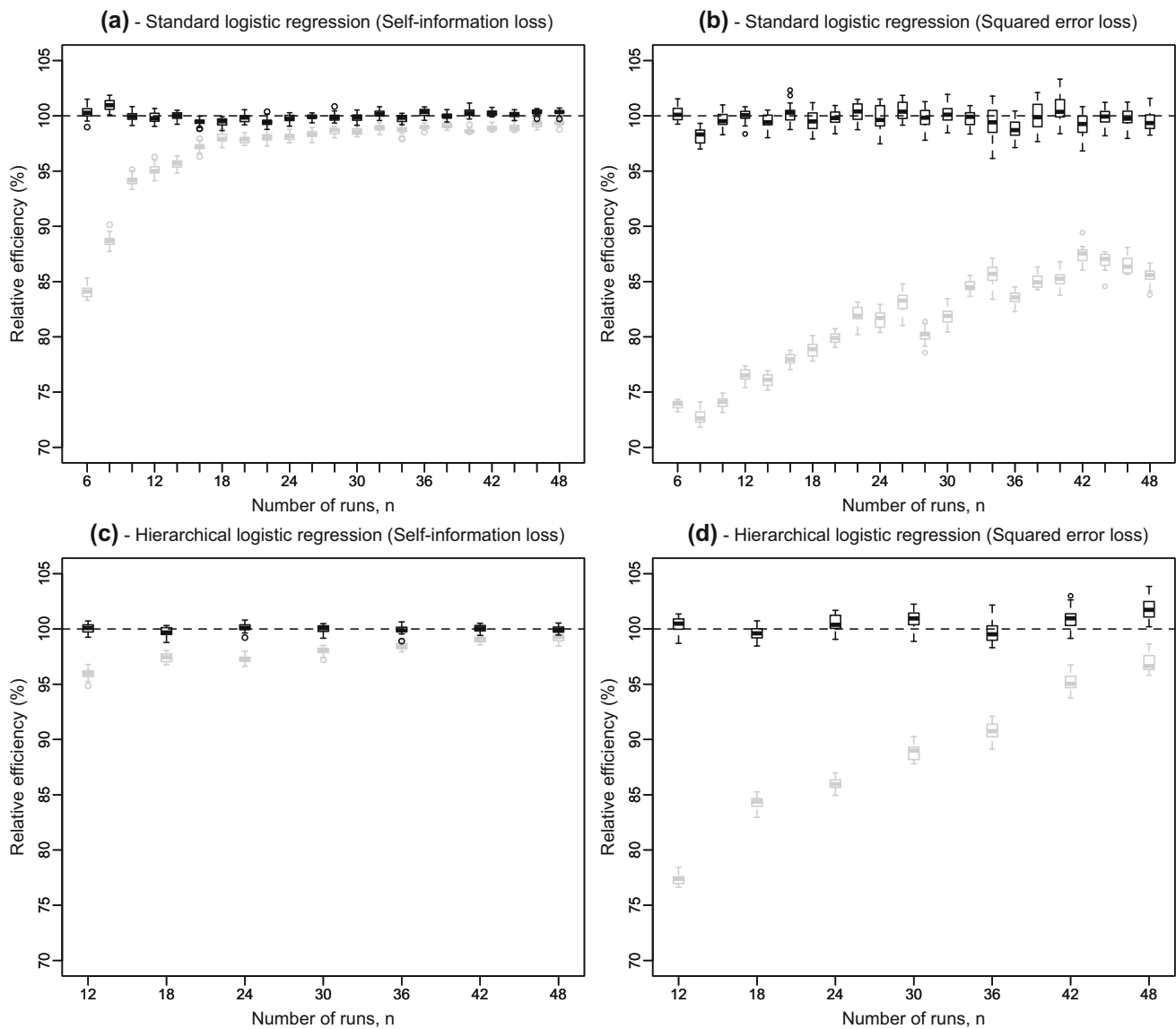


Fig. 1 Boxplots of twenty DLMC approximations ($B = 20,000$) to the relative self-information (a, c) and squared error (b, d) efficiencies plotted against $n \in N_S = \{6, 8, \dots, 48\}$ for standard logistic regression (a, b) and against $n \in N_H = \{12, 18, \dots, 48\}$ for hierarchical logistic regression (c, d)

element of β with lower and upper limits given by $\mathbf{L} = (-3, 4, 5, -6, -2.5)$ and $\mathbf{U} = (3, 10, 11, 0, 3.5)$. Following Overstall and Woods (2017), we assume two different prior distributions for each γ_i ($i = 1, \dots, G$).

- (i) A prior point mass at $\gamma_i = \mathbf{0}$ for all i , resulting in standard logistic regression with $p_m = 1 + \sum_{t=1}^4 v_t$.
- (ii) A hierarchical prior distribution in which elements of γ_i are independent and identically distributed as $\gamma_{it} \sim U[-\zeta_t, \zeta_t]$, for $t = 0, \dots, 4$ with $\zeta_t \in (0, Z_t)$ unknown and having triangular prior density $\pi(\zeta_t) = 2(Z_t - \zeta_t)/Z_t^2$ with $(Z_0, Z_1, \dots, Z_4) = (3, 3, 3, 1, 1)$. The numbers of parameters is $p_m = 1 + G + (1 + G) \sum_{t=1}^4 v_t$.

Under each of these two prior distributions, we find designs for the experimental aims of parameter estimation and model discrimination.

3.2.1 Parameter estimation

We set $v_t = 1$ for all t so that there is no model uncertainty and consider generating designs under the aim of estimating $\phi = \beta$. This means $g_m = g$ is a linear function given by $g(\theta) = A\theta$, where for (1) standard logistic regression, $A = I_5$; and for (2) hierarchical logistic regression, $A = (I_5, R)$ is a $5 \times 5(1 + G)$ matrix where R is a $5 \times 5G$ matrix of zeros.

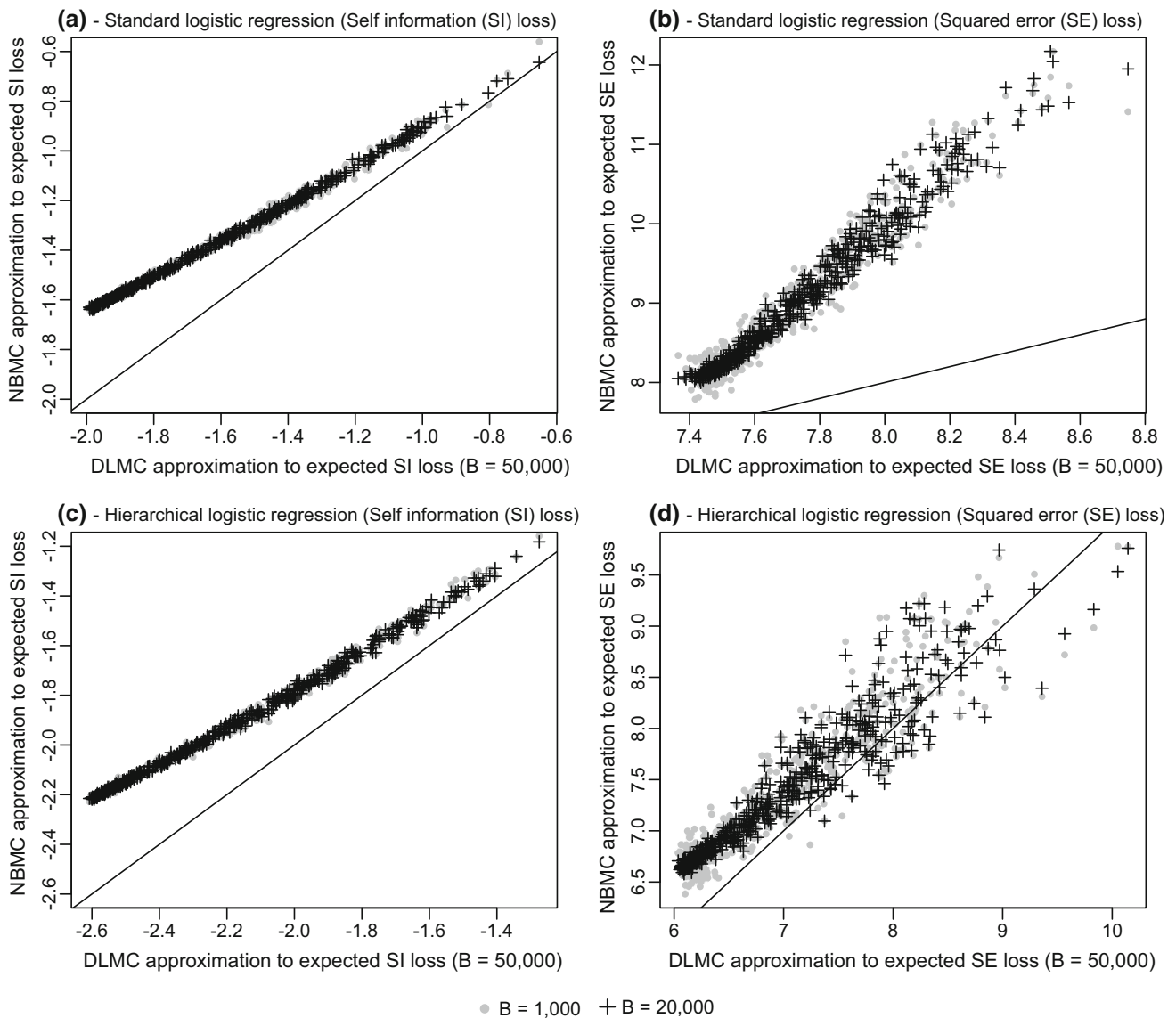


Fig. 2 Plots of NBMC approximations (with $B=1000$ and $B = 20,000$) to the expected loss plotted against the DLMC approximation ($B = 50,000$) to the expected loss for standard (a, b for $n = 6$) and hierar-

chical (c, d) for $n = 12$) logistic regression under the self-information (a, c) and squared error (b, d) loss function. A line through the origin with slope one has been added to aid in the comparison

We consider the self-information and squared error loss functions and compare designs found under the DLMC approximation (as found by Overstall and Woods 2017 and referred henceforth as DLMC designs) against designs found under the NBMC approximations proposed in this paper (henceforth referred to as NBMC designs). To make comparisons valid, we found the NBMC designs under exactly the same implementation of ACE as those used to find the DLMC designs (see “Appendix 1” for details). Additionally, we also compare against pseudo-Bayesian D- and A-optimal designs (also as found by Overstall and Woods 2017).

We compare designs using relative efficiency. Let \mathbf{d}_{SI}^* and \mathbf{d}_{SE}^* be the DLMC designs under the SI and SE loss func-

tions, respectively, found by Overstall and Woods (2017). The relative SI and SE efficiencies of a design \mathbf{d} are defined as

$$R_{SI}(\mathbf{d}) = \frac{L_{SI}(\mathbf{d})}{L_{SI}(\mathbf{d}_{SI}^*)} \times 100\%, \tag{15}$$

$$R_{SE}(\mathbf{d}) = \frac{L_{SE}(\mathbf{d}_{SE}^*)}{L_{SE}(\mathbf{d})} \times 100\%, \tag{16}$$

respectively, where L_{SI} and L_{SE} refer to the expected SI and SE loss functions, respectively. Note that the definition of relative SI efficiency, given by (15), follows from how the expected SI loss is non-positive. The relative efficiencies are

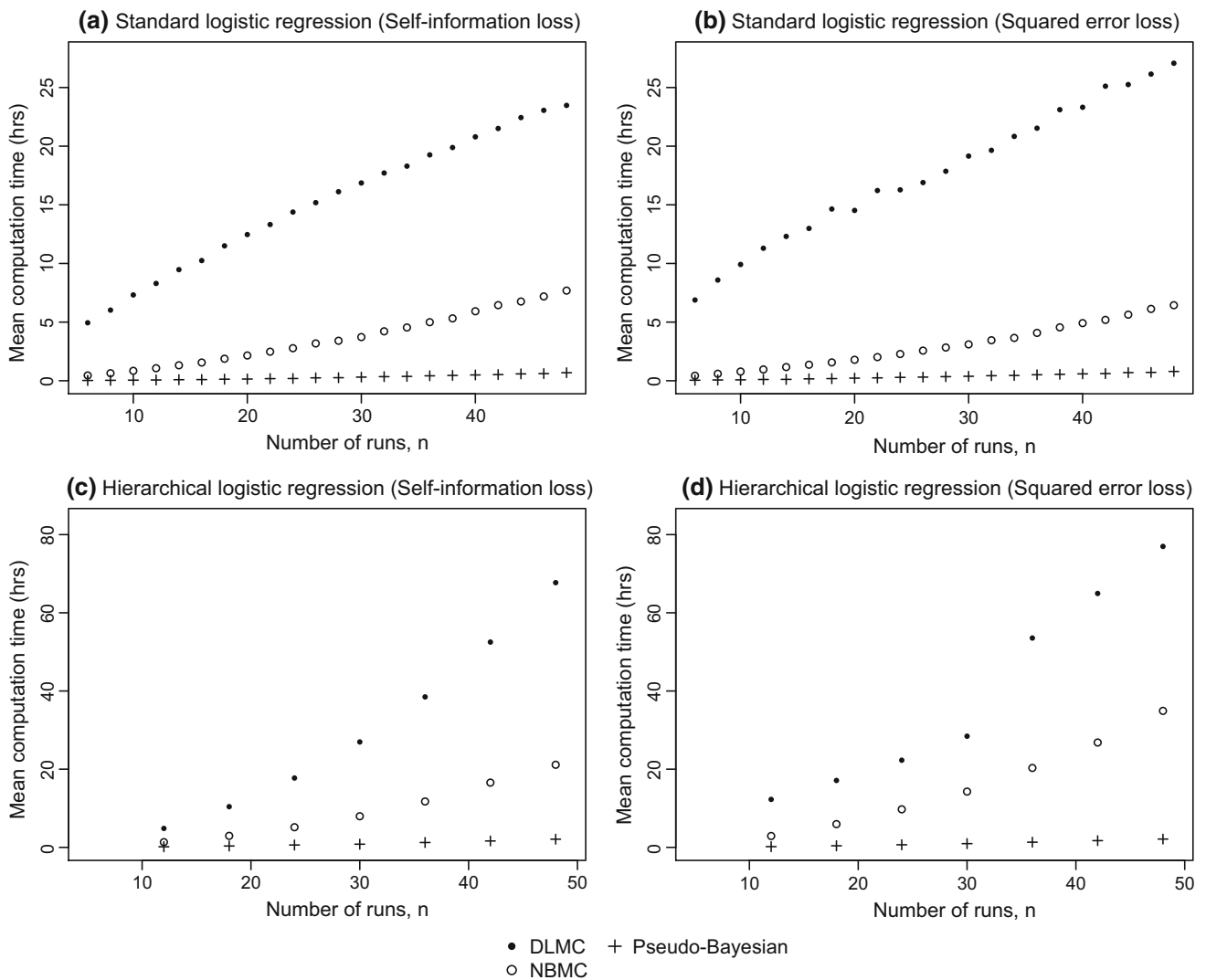


Fig. 3 Plots of mean computer time required to find the NBMC, DLMC and pseudo-Bayesian designs for standard (a, b) and hierarchical (c, d) logistic regression under the SI (a, c) and SE (b, d) loss function

approximated by approximating the expected losses in (15) and (16) by using the DLMC approximation.

The top row of Fig. 1 shows boxplots of twenty DLMC approximations to the relative efficiency for SI (left) and SE (right) for the NBMC and pseudo-Bayesian designs plotted against $n \in N_S = \{6, 8, \dots, 48\}$ (meaning W ranges from 24 to 192) for standard logistic regression. The bottom row shows the same boxplots for hierarchical logistic regression plotted against $n \in N_H = \{12, 18, \dots, 48\}$ (meaning W ranges from 48 to 192). In both cases, the relative efficiencies of the NBMC designs are clearly very close to one for all values of n indicating that the performance of these designs (in terms of expected loss) is very close to the performance of the DLMC designs. However, the relative efficiency of the pseudo-Bayesian designs only become close to one as n gets larger. In the case of the SI loss, it appears from Fig. 1 that

we could obtain a negligible difference in expected SI loss for values of n over approximately forty by using the pseudo-Bayesian D-optimal design. This design is computationally more efficient to find than a fully Bayesian design, however, knowing that it had nearly equivalent performance to the fully Bayesian design would be hard without first finding the fully Bayesian design.

We now investigate the accuracy of the NBMC approximation to the expected loss. For the SI and SE loss, we randomly generate T designs and for each one calculate three different approximations to the expected loss: a DLMC approximation with $B = 50,000$ (which we consider near exact evaluation of the expected loss), and NBMC approximations with $B = 1000$ and $B = 20,000$. The t th design for $t = 1, \dots, T$, is generated by perturbing the DLMC designs under each of the loss functions as follows

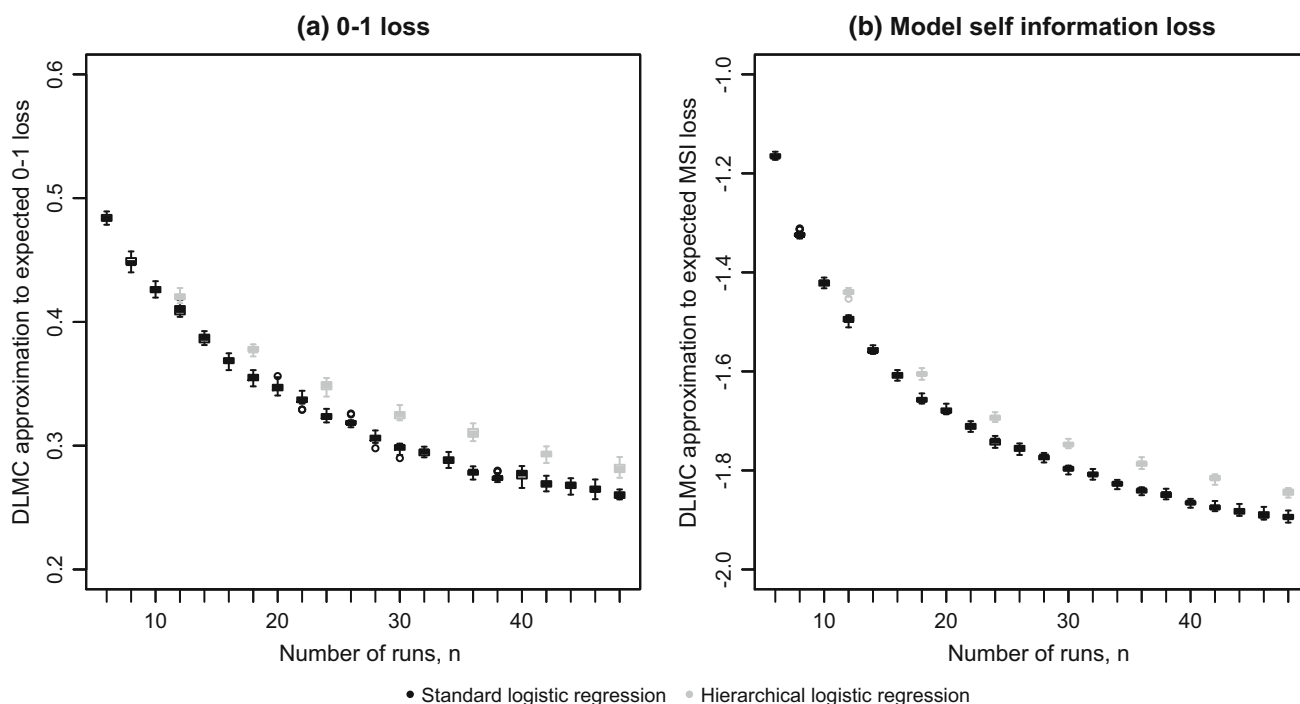


Fig. 4 Plots of the DLMC approximation ($B = 20,000$) to the expected 0–1 (a) and model self-information (b) loss function for standard and hierarchical logistic regression against n

$$\mathbf{d}_{SI}^{(t)} = (1 - u_t)\mathbf{d}_{SI}^* + u_t\mathbf{d}^{(t)},$$

$$\mathbf{d}_{SE}^{(t)} = (1 - u_t)\mathbf{d}_{SE}^* + u_t\mathbf{d}^{(t)}$$

where, at each iteration, $u_t \sim U(0, \frac{1}{2})$, and $\mathbf{d}^{(t)}$ is a design in which each element is generated from $U(-1, 1)$. The top row of Fig. 2 shows plots of the two NBMC approximations to the expected loss plotted against the DLMC approximation to the expected loss for standard logistic regression under the SI (left) and SE (right) loss function for $n = 6$ (i.e. the smallest number of runs considered). The bottom row of Fig. 2 shows the same plots for hierarchical logistic regression with $n = 12$ (again the smallest number of runs considered). In all four cases, although the NBMC approximation to the expected loss can be inaccurate, especially for designs close to the minimum, the ordering of designs in terms of expected loss is the same as for the DLMC approximation. This means the NBMC approximation to the expected loss is minimised for design close to the design that minimises the DLMC approximation.

Figure 3 shows plots of the mean computer time required to compute the three types of design (NBMC, DLMC and pseudo-Bayesian) for the models and loss functions considered. Note that the algorithm was run on the IRIDIS 4 supercomputer facility at the University of Southampton which has 2.6Ghz processors with 4Gb memory. Note that the MBMC designs are typically found in a third of the computing time required to find the DLMC designs.

The pseudo-Bayesian designs require the smallest amount of computing time but this should be judged in parallel with the lack of efficiency these designs exhibit when compared to the corresponding fully Bayesian design (see Fig. 1).

3.2.2 Model discrimination

Now consider the aim of model discrimination. Suppose that $v_0 = 1$ (corresponding to the intercept) but that v_t is unknown for $t = 1, \dots, 4$. Each unique combination of (v_1, \dots, v_4) corresponds to a unique model m . The total number of models is $|\mathcal{M}| = 2^4 = 16$ ranging from $(v_1, \dots, v_4) = (0, 0, 0, 0)$ (intercept only; all factors inactive) to $(v_1, \dots, v_4) = (1, 1, 1, 1)$ (full model; all factors active). The prior model probabilities are such that $P(v_t = 1) \sim U[0, 1]$ independently, for each $t = 1, \dots, 4$. For model m , let (v_{m1}, \dots, v_{m4}) denote the values of v_t where the number of regression parameters is given by $b_m = 1 + \sum_{t=1}^4 v_{mt}$. The marginal prior model probabilities are then given by

$$\pi(m) = \frac{1}{5 \binom{4}{b_m - 1}},$$

which corrects for Bayesian multiplicity (Scott and Berger 2010).

We find designs for both standard and hierarchical logistic regression for each value of n (N_S for standard, and N_H for

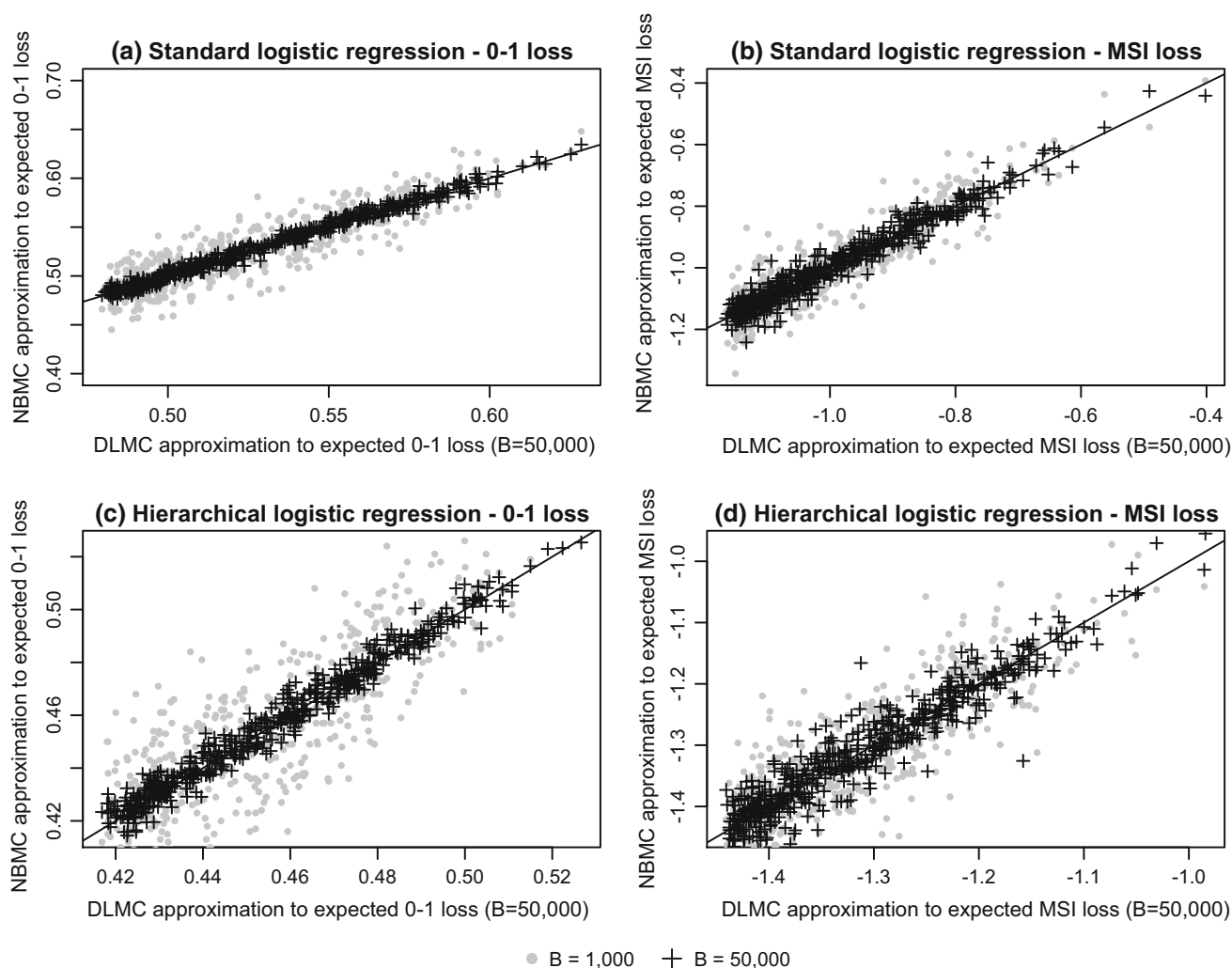


Fig. 5 Plots of NBMC approximation (with $B = 1000$ and $B = 20,000$) to the expected 0–1 (a, c) and model self-information (b, d) loss plotted against the corresponding DLMC approximation ($B = 50,000$) for standard (a, b for $n = 6$) and hierarchical (c, d for $n = 12$) logistic regression

hierarchical), under both the 0–1 and model self-information loss functions, respectively. Bayesian optimal design on such a scale for the experimental aim of model discrimination has not been addressed previously in the literature. It would be infeasible to find DLMC designs in this situation since it would require $|\mathcal{M}| = 16$ Monte Carlo approximations to the marginal likelihood for every $b = 1, \dots, B$. Conversely, the normal-based approximations will require $|\mathcal{M}| = 16$ Laplace approximations which will be computationally less intensive.

However, although it is infeasible to use the DLMC approximation to find designs, we can use it to assess the NBMC designs. Figure 4 shows boxplots of twenty DLMC approximations to the expected 0–1 and model self-information loss functions for both standard and hierarchical logistic regression against n for the NBMC designs. Note that the expected loss for the hierarchical model is always greater than for the standard model and that this difference

increases as n increases. This is due to the extra uncertainty introduced by the blocks and their associated block-specific parameters (whose number is proportional to n).

Similar to Sect. 3.2.1, we check the validity of the approximation by plotting NBMC approximations (with $B = 1000$ and $B = 20,000$) against the DLMC approximation to the expected loss with $B = 50,000$. Figure 5 shows the resulting plots for $n = 6$ (standard logistic regression) and $n = 12$ (hierarchical logistic regression). In this case, we can see that the NBMC approximations to the expected loss appear very accurate for both loss functions.

3.3 Mechanistic modelling of chemical reactions

In this section, we consider the famous example from [Box and Hill \(1967\)](#) concerning discriminating between nonlinear models for describing chemical reactions. Suppose the i th run consists of specifying temperature $x_{i1} \in (0, 150)$ and

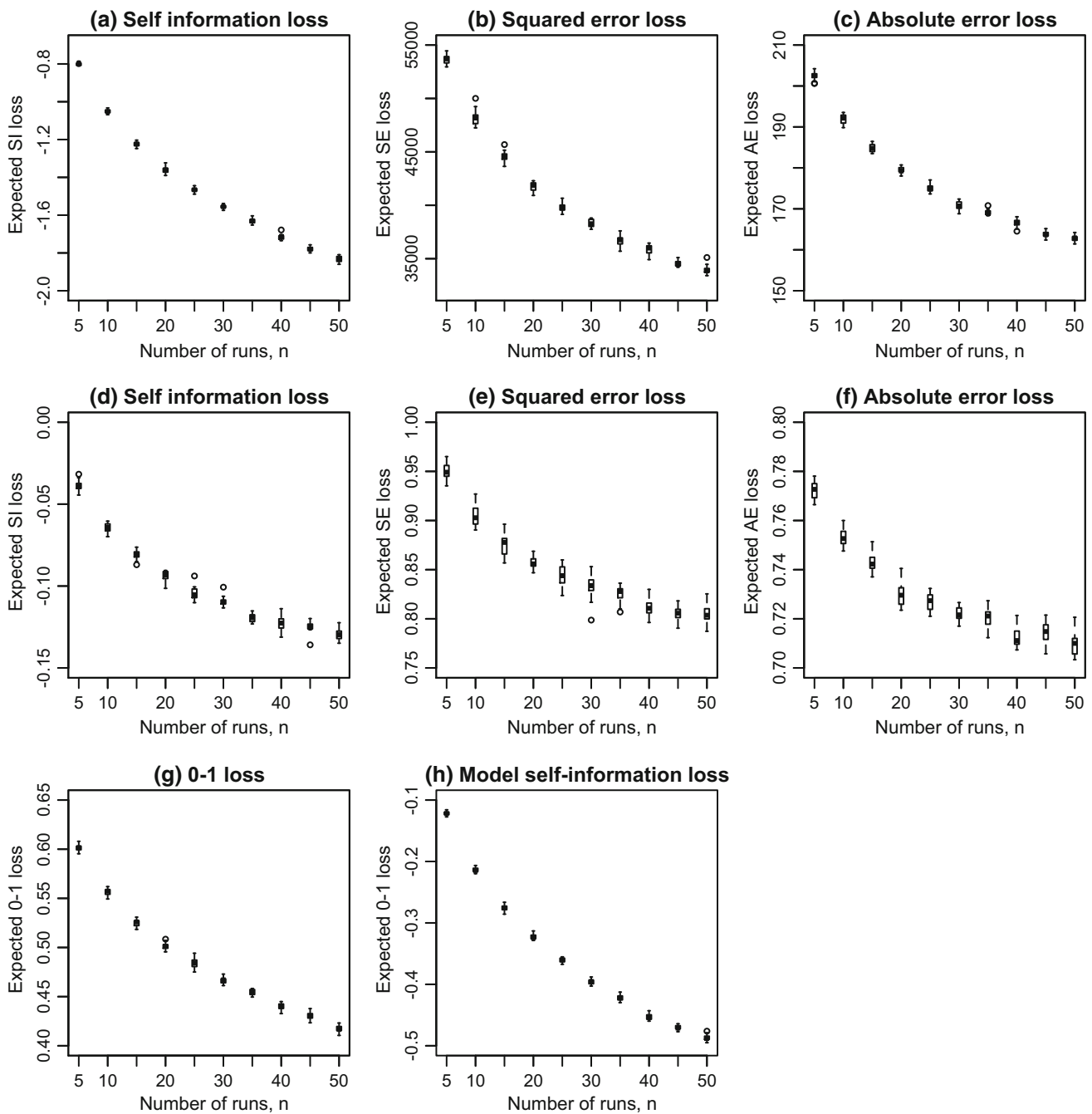


Fig. 6 Boxplots of twenty NBMC approximations ($B = 20,000$) to the expected loss for the NBMC designs plotted against n under the each of the loss functions for the nonlinear model example

reaction time $x_{i2} \in (450, 600)$, i.e. $\mathbf{d}_i = (x_{i1}, x_{i2})$, and measuring the yield y_i from a chemical reaction, for $i = 1, \dots, n$. It is assumed that

$$y_i \sim N(\eta_m(\boldsymbol{\theta}; \mathbf{d}_i), \sigma^2),$$

independently for $i = 1, \dots, n$ where $m \in \mathcal{M}$. Consider the set $\mathcal{M} = \{1, 2, 3, 4\}$ of $|\mathcal{M}| = 4$ competing models for $\eta_m(\boldsymbol{\theta}; \mathbf{d}_i)$ given as follows

$$m = 1 : \eta_1(\boldsymbol{\theta}; \mathbf{d}_i) = \exp\left(-\theta_{11}x_{i1} \exp\left(-\frac{\theta_{12}}{x_{i2}}\right)\right);$$

$$m = 2 : \eta_2(\boldsymbol{\theta}; \mathbf{d}_i) = 1/\left(1 + \theta_{21}x_{i1} \exp\left(-\frac{\theta_{22}}{x_{i2}}\right)\right);$$

$$m = 3 : \eta_3(\boldsymbol{\theta}; \mathbf{d}_i) = 1/\left(1 + \theta_{31}x_{i1} \exp\left(-\frac{\theta_{32}}{x_{i2}}\right)\right)^{\frac{1}{2}};$$

$$m = 4 : \eta_4(\boldsymbol{\theta}; \mathbf{d}_i) = 1/\left(1 + \theta_{41}x_{i1} \exp\left(-\frac{\theta_{42}}{x_{i2}}\right)\right)^{\frac{1}{3}};$$

where the unknown parameters $\theta_m = (\theta_{m1}, \theta_{m2})$ have the same interpretation under each model. Following Box and Hill (1967), a-priori we assume that $\theta_{m1} \sim N(400, 25^2)$ and $\theta_{m2} \sim N(5000, 250^2)$, independently, and that $\pi(m) = 1/4$. Box and Hill (1967) assumed that the response variance was fixed as $\sigma^2 = 0.1^2$. However, we let σ^2 be unknown and assume that $\sigma^2 \sim U[0, 1]$.

We consider $n = \{5, 10, 15, \dots, 50\}$ meaning that W ranges from 10 to 100. To demonstrate the versatility of the normal-based approximations presented in this paper, for each value of n , we find NBMC designs under a total of eight different loss functions. We consider the three exemplar loss functions (Sect. 3.1.1) for parameter estimation for (a) $\phi = g_m(\theta_m) = \theta_m$; and (b) $\phi = g_m(\theta_m) = \theta_{m2}/\theta_{m1}$, where in the latter case we are interested in the ratio of the two unknown parameters. In both cases, we are taking account of model uncertainty by considering the model-averaged posterior distribution of ϕ . We also consider the two exemplar loss functions for model discrimination (Sect. 3.1.2).

It is not clear how to find DLMC approximations to the expected loss for every loss function considered in this section. For example, a DLMC approximation to the model-averaged posterior median required for the AE loss would require samples from the posterior distribution of θ_m for each $m \in \mathcal{M}$ and \mathbf{y}^b , for $b = 1, \dots, B$. Therefore, in this section, we rely on the NBMC approximations to assess the performance of designs. Figure 6 shows twenty boxplots of the NBMC approximation ($B = 20,000$) to the expected loss for the NBMC designs plotted against n for each of the loss functions. Note how the expected loss functions for $\phi = \theta_m$ have a faster relative decrease in expected loss with increasing n than the expected loss for $\phi = \theta_{m2}/\theta_{m1}$. This is easiest to see for the SE in loss in Fig. 6b, where the expected SE loss for $\phi = \theta_m$ has a relative decrease of approximately 35% when n increase from 5 to 50. The corresponding (see Fig. 6e) relative decrease in expected SE loss for $\phi = \theta_{m2}/\theta_{m1}$ is approximately 15%. This is due to the form of the parameterisation of interest $g_m(\theta_m)$. When $\phi = \theta_m$, the trace of the prior variance of ϕ is 63,125 which gives an upper bound on the expected loss. The corresponding value for $\phi = \theta_{m2}/\theta_{m1}$ is approximately 1. Under the latter parameterisation, it appears the choice of design does not have as great an impact on the expected SE loss than the former.

4 Conclusion

In this paper, we have proposed the use of normal-based approximations to posterior summaries to aid in the approximation of the loss function. The resulting approximate loss can be used in conjunction with any algorithm suitable for finding the design that minimises the expected loss function. The methodology was used in conjunction with the ACE

algorithm to find designs which have similar performance (in terms of expected loss) to designs generated by the original ACE algorithm with the DLMC approximation to the expected loss, but using a fraction of the computing time. The methodology was also able to find designs for problems, under model uncertainty, where use of the DLMC approach would be infeasible and, as such, have not previously been addressed in the literature.

The normal-based approximations utilised in this paper are formed by using the location of, and curvature around, the posterior mode. Taking advantage of alternative normal-based or other deterministic approximations may be of future interest. In particular, one could consider using the expectation propagation (EP) algorithm of Minka (2001) to form efficient approximations to the posterior distribution of the parameters. Similar to mode-/curvature-based approximations used in this paper, EP also provides an efficient approximate of the marginal likelihood. EP could potentially be useful in a wider range of problems as any distribution in the exponential family could be considered as the parametric form for approximating the posterior distribution.

Acknowledgements The authors would like to thank the two anonymous referees for their insightful comments which have significantly improved the paper. The authors would also like to thank Prof David Woods (University of Southampton) for helpful suggestions on a draft of this paper. This work arose from discussions and collaborations initiated at the “Bayesian Optimal Design of Experiments” workshop (Queensland University of Technology, Brisbane, Australia, December 2015; <http://www.bode2015.wordpress.com>). A.M. Overstall was supported by a Research Incentive Grant (70424) from the Carnegie Trust for the Universities of Scotland. C.C. Drovandi was supported by an Australian Research Council’s Discovery Early Career Researcher Award funding scheme (DE160100741).

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix 1: Details on the approximate coordinate exchange algorithm

ACE algorithm

1. Choose an initial design $\mathbf{d}^0 = (d_1^0, \dots, d_W^0)$ and set the current design to be $\mathbf{d}^C = (d_1^C, \dots, d_W^C) = \mathbf{d}^0$.
2. For $i = 1, \dots, W$ complete the following steps
 - (a) Let $L^i(d) = L(d_1^C, \dots, d_{i-1}^C, d, d_{i+1}^C, \dots, d_W^C)$ be the function given by the expected loss function which only varies over the design space, \mathcal{D}_i , for the i th element.

- (b) For $j = 1, \dots, Q$, evaluate the MC approximation to the expected loss given by

$$z_j = \hat{L}^i(d_j),$$

for $\{d_1, \dots, d_Q\} \in \mathcal{D}_i$. Fit a GP emulator to $\{z_j, d_j\}_{j=1}^Q$ and set $\tilde{L}^i(d)$ to be the resulting predictive mean.

- (c) Find

$$d_i^* = \operatorname{argmin}_{d \in \mathcal{D}_i} \tilde{L}^i(d),$$

and let $\mathbf{d}^* = (d_1^C, \dots, d_{i-1}^C, d^*, d_{i+1}^C, \dots, d_W^C)$ be the proposed design.

- (d) Set $\mathbf{d}^C = \mathbf{d}^*$ with probability p^* .

3. Return to step 2.

Comparison procedure

In step 2d, we accept the proposed design, \mathbf{d}^* with probability p^* . The proposed design originates from the Gaussian process emulator. Similar to all statistical models, Gaussian process emulators, can fit inadequately. To mitigate the effects of an inadequate emulator, [Overstall and Woods \(2017\)](#) proposed a comparison between the proposed design \mathbf{d}^* and the current design \mathbf{d}^C . Note that the proposed design \mathbf{d}^* should be accepted if

$$E_{\theta_m, m, \mathbf{y} | \mathbf{d}^*} [\lambda(\theta_m, m, \mathbf{y}, \mathbf{d}^*)] < E_{\theta_m, m, \mathbf{y} | \mathbf{d}^C} [\lambda(\theta_m, m, \mathbf{y}, \mathbf{d}^C)] \tag{17}$$

For $b = 1, \dots, B$ we generate samples $\{\lambda_*^b\}_{b=1}^B$ and $\{\lambda_C^b\}_{b=1}^B$ as follows

$$\begin{aligned} \lambda_*^b &= \lambda(\theta_m^{*b}, m^{*b}, \mathbf{y}^{*b}, \mathbf{d}^*), \\ \lambda_C^b &= \lambda(\theta_m^b, m^b, \mathbf{y}^b, \mathbf{d}^C), \end{aligned}$$

where $\{\theta_m^{*b}, m^{*b}, \mathbf{y}^{*b}\}_{b=1}^B$ and $\{\theta_m^b, m^b, \mathbf{y}^b\}_{b=1}^B$ are samples from the joint distribution of θ_m, m and \mathbf{y} conditional on \mathbf{d}^* and \mathbf{d}^C , respectively. We use these samples to perform a Bayesian hypothesis test of (17). The form of the Bayesian hypothesis test, as described in [Overstall and Woods \(2017\)](#), assumes that the λ_*^b 's and λ_C^b 's are continuous and their distribution reasonably assumed normal. In this case, the probability of accepting the proposed design is

$$p^* = 1 - F\left(\frac{\sum_{b=1}^B \lambda_C^b - \sum_{b=1}^B \lambda_*^b}{\sqrt{2B\hat{v}}}\right),$$

where $F(\cdot)$ is the distribution function of the t -distribution with $2B - 2$ degrees of freedom,

$$\hat{v} = \frac{\sum_{b=1}^B (\lambda_C^b - \bar{\lambda}_C)^2 + \sum_{b=1}^B (\lambda_*^b - \bar{\lambda}_*)^2}{2B - 2},$$

and $\bar{\lambda}_C$ and $\bar{\lambda}_*$ are the sample means of the λ_C^b 's and λ_*^b 's, respectively.

The assumption of normality will clearly be violated for the 0–1 loss function for model discrimination, described in Sect. 3.1, where the λ_*^b 's and λ_C^b 's will be binary in the set $\{0, 1\}$. For such loss functions, we introduce the following modification. Assume that

$$\begin{aligned} \lambda_C^b &\stackrel{\text{iid}}{\sim} \text{Bernoulli}(\rho_C), \\ \lambda_*^b &\stackrel{\text{iid}}{\sim} \text{Bernoulli}(\rho_*), \end{aligned}$$

for $b = 1, \dots, B$. We also assume the following independent prior distributions: $\rho_C \sim \text{U}[0, 1]$ and $\rho_* \sim \text{U}[0, 1]$. The resulting posterior distributions are

$$\begin{aligned} \rho_C | \lambda_C^1, \dots, \lambda_C^B &\sim \text{Beta}(1 + B\bar{\lambda}_C, 1 + B - B\bar{\lambda}_C), \\ \rho_* | \lambda_*^1, \dots, \lambda_*^B &\sim \text{Beta}(1 + B\bar{\lambda}_*, 1 + B - B\bar{\lambda}_*). \end{aligned}$$

The probability of accepting the new design is then given by

$$p^* = \text{P}\left(\rho_* < \rho_C | \lambda_C^1, \dots, \lambda_C^B, \lambda_*^1, \dots, \lambda_*^B\right)$$

which is evaluated via simulation as follows

$$p^* \approx \frac{1}{B} \sum_{b=1}^B F\left(\rho_C^b; 1 + B\bar{\lambda}_*, 1 + B - B\bar{\lambda}_*\right),$$

where $F(\cdot; a, b)$ denotes the distribution function of the Beta(a, b) and $\{\rho_C^b\}_{b=1}^B$ is a sample from Beta($1 + B\bar{\lambda}_C, 1 + B - B\bar{\lambda}_C$).

Implementation details

To reduce the likelihood of the ACE algorithm converging to local optima, it is restarted from E different starting designs. These E repetitions of the ACE algorithm are run in an embarrassingly parallel fashion. Note that there is further scope for parallelising the ACE algorithm. The Q evaluations of the Monte Carlo approximation to the expected loss in step 2b could be parallelised. Furthermore, the calculation of the posterior mode for $b = 1, \dots, B$ could also be parallelised. These have not been pursued here through. Even by repeating the ACE algorithm E times does not guarantee that it will converge to the true optimal design. Following [Overstall and Woods \(2017\)](#) we set $E = 20$, $Q = 20$, and $B = 1000$,

except for in the comparison procedure where $B = 20,000$. These values were found by Overstall and Woods (2017) to perform well for a variety of examples. Additionally, Müller and Parmigiani (1995) also found that B could be similarly small when using their smoothing-based approach. Note that in their use of DLMC approximation to the expected loss, Overstall and Woods (2017) used $\tilde{B} = B$ for the Monte Carlo sample size in the inner loop of the DLMC approximation. Again following Overstall and Woods (2017), we fit the Gaussian process model using a squared exponential correlation function.

The ACE algorithm is implemented in the R package *acebayes* (Overstall et al. 2017b) available from the Comprehensive R Archive Network.

Appendix 2: Relationship between proposed approximations and pseudo-Bayesian design

Consider the case where $|M| = 1$ and $g(\boldsymbol{\theta}) = \boldsymbol{\theta}$. The approximated SI loss is then

$$\tilde{\lambda}_{SI}(\boldsymbol{\theta}, \mathbf{y}, \mathbf{d}) = \log \pi(\boldsymbol{\theta}) + \frac{p}{2} \log(2\pi) + \frac{1}{2} \log |\hat{\Sigma}_{\theta}| + \frac{1}{2} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}(\mathbf{y}))^T \hat{\Sigma}_{\theta}^{-1} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}(\mathbf{y})).$$

The corresponding expected approximate SI loss is given by

$$\tilde{L}_{SI}(\mathbf{d}) = E_{\theta} \left[E_{\mathbf{y}|\theta, \mathbf{d}} \left[\tilde{\lambda}_{SI}(\boldsymbol{\theta}, \mathbf{y}, \mathbf{d}) \right] \right]. \quad (18)$$

Assume that the prior distribution for $\boldsymbol{\theta}$ is sufficiently diffuse so that $\hat{\Sigma}_{\theta} = \mathcal{I}(\boldsymbol{\theta}; \mathbf{d})^{-1}$ and the posterior mode is approximately equal to the maximum likelihood estimator (MLE). Using the delta-method and the approximate distribution of the MLE, the following approximation for $\tilde{L}_{SI}(\mathbf{d})$ can be derived

$$\tilde{L}_{SI}(\mathbf{d}) = E_{\theta} \left[\frac{p}{2} \log(2\pi) + \frac{p}{2} + \log \pi(\boldsymbol{\theta}) - \frac{1}{2} \log |\mathcal{I}(\boldsymbol{\theta}; \mathbf{d})| \right].$$

This is proportional to the objective function for pseudo-Bayesian D-optimality. Similarly, an approximation to the expected squared error loss is given by

$$\tilde{L}_{SI}(\mathbf{d}) = E_{\theta} \left[\text{tr} \left\{ \mathcal{I}(\boldsymbol{\theta}; \mathbf{d})^{-1} \right\} \right],$$

the objective function for pseudo-Bayesian A-optimality.

References

Achcar, J., Smith, A.: Aspects of reparameterization in approximate Bayesian inference. In: Hodges, J. (ed.) *Essays in Honour of G. A. Barnard*. North-Holland, Amsterdam (1990)

- Amzal, B., Bois, F.Y., Parent, E., Robert, C.P.: Bayesian-optimal design via interacting particle systems. *J. Am. Stat. Assoc.* **101**, 773–785 (2006)
- Box, G.E.P., Hill, W.J.: Discrimination among mechanistic models. *Technometrics* **9**, 57–71 (1967)
- Chaloner, K., Verdinelli, I.: Bayesian experimental design: a review. *Stat. Sci.* **10**, 273–304 (1995)
- Friel, N., Wyse, J.: Estimating the evidence—a review. *Stat. Neerlandica* **66**, 288–308 (2012)
- Gelman, A., Carlin, J., Stern, H., Dunson, D., Vehtari, A., Rubin, D.: *Bayesian Data Analysis*, 3rd edn. Chapman and Hall, Boca Raton (2014)
- Girolami, M., Calderhead, B.: Riemann manifold langevin and hamiltonian monte carlo methods. *J. R. Stat. Soc. Ser. B* **73**, 123–214 (2011)
- Goldstein, M., Wooff, D.: *Bayes Linear Statistics: Theory and Methods*. Wiley, Chichester (2007)
- Gotwalt, C.M., Jones, B.A., Steinberg, D.M.: Fast computation of designs robust to parameter uncertainty for nonlinear settings. *Technometrics* **51**, 88–95 (2009)
- Hamada, M., Martz, H., Reese, C., Wilson, A.: Finding near-optimal Bayesian experimental designs via generic algorithms. *Am. Stat.* **55**, 175–181 (2001)
- Hoffman, M., Gelman, A.: The no-u-turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *J. Mach. Learn. Res.* **15**, 1593–1623 (2014)
- Jones, M., Goldstein, M., Jonathan, P., Randell, D.: Bayes linear analysis for Bayesian optimal experimental design. *J. Stat. Plan. Inference* **171**, 115–129 (2016)
- Khuri, A.: *Advanced Calculus with Applications in Statistics*, 2nd edn. Wiley, Hoboken (2003)
- Lange, K.: *Optimization*, 2nd edn. Springer, New York (2013)
- Lindley, D.: On a measure of the information provided by an experiment. *Ann. Math. Stat.* **27**, 986–1005 (1956)
- Long, Q., Scavino, M., Tempone, R., Wang, S.: Fast estimation of expected information gains for Bayesian experimental designs based on Laplace approximations. *Comput. Methods Appl. Mech. Eng.* **259**, 24–39 (2013)
- McGree, J.M.: Developments of the total entropy utility function for the dual purpose of model discrimination and parameter estimation in Bayesian design. *Comput. Stat. Data Anal.* To appear (2017)
- Meyer, R., Nachtsheim, C.: The coordinate-exchange algorithm for constructing exact optimal experimental designs. *Technometrics* **37**, 60–69 (1995)
- Minka, T.: Expectation propagation for approximate Bayesian inference. In: Breese, J., Koller, D. (eds.) *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence* (2001)
- Müller, P.: Simulation-based optimal design. *Bayesian. Statistics* **6**, 459–474 (1999)
- Müller, P., Parmigiani, G.: Optimal design via curve fitting of monte carlo experiments. *J. Am. Stat. Assoc.* **90**(432), 1322–1330 (1995)
- Müller, P., Sanso, B., De Iorio, M.: Optimal Bayesian design by inhomogeneous Markov chain simulation. *J. Am. Stat. Assoc.* **99**, 788–798 (2004)
- O’Hagan, A., Forster, J.J.: *Kendall’s Advanced Theory of Statistics: Volume 2B Bayesian Inference*, 2nd edn. Wiley, Chichester (2004)
- Overstall, A.M., McGree, J.M., Drovandi, C.C.: *Normal Based Designs*. R package Version 1.0 (2017a)
- Overstall, A.M., Woods, D.C., Adamou, M.: *acebayes: Optimal Bayesian experimental design using the ACE algorithm*. R package version 1.3.4 (2017b). <https://CRAN.R-project.org/package=acebayes>
- Overstall, A.M., Woods, D.C.: Bayesian design of experiments using approximate coordinate exchange. *Technometrics*, (2017). doi:10.1080/00401706.2016.1251495

- Pawitan, Y.: In All Likelihood: Statistical Modelling and Inference using Likelihood. Oxford University Press, Oxford (2013)
- R Core Team R: A language and environment for statistical computing. R Foundation for Statistical Computing Vienna, Austria (2016) <https://www.R-project.org/>
- Rainforth, T., Cornish, R., Yang, H., Wood, F.: On the pitfalls of nested Monte Carlo. Tech. rep. University of Oxford (2016) [ArXiv:1612.00951](https://arxiv.org/abs/1612.00951)
- Robert, C.: The Bayesian Choice, 2nd edn. Springer, New York (2007)
- Ryan, E.G., Drovandi, C.C., McGree, J.M., Pettitt, A.N.: A review of modern computational algorithms for Bayesian optimal design. *Int. Stat. Rev.* **84**, 128–154 (2016)
- Ryan, K.: Estimating expected information gains for experimental designs with application to the random fatigue-limit model. *J. Comput. Graph. Stat.* **12**, 585–603 (2003)
- Santner, T., Williams, B., Notz, W.: The Design and Analysis of Computer Experiments. Springer, New York (2003)
- Scott, J., Berger, J.: Bayes and empirical-Bayes multiplicity adjustment in the variable-selection problem. *Ann. Stat.* **38**, 2587–2619 (2010)
- Weaver, B., Williams, B., Anderson-Cook, C., Higdon, D.: Computational enhancements to bayesian design of experiments using gaussian computational enhancements to bayesian design of experiments using gaussian processes. *Bayesian Anal.* **11**, 191–213 (2016)
- Woods, D., Overstall, A., Adamou, M., Waite, T.: Bayesian design of experiments for generalised linear models and dimensional analysis with industrial and scientific application. *Qual. Eng.* **29**, 91–103 (2017)
- Woods, D.C., Lewis, S.M., Eccleston, J.A., Russell, K.G.: Designs for generalized linear models with several variables and model uncertainty. *Technometrics* **48**, 284–292 (2006)