JDSSV

Journal of Data Science, Statistics, and Visualisation

*January 2025, Volume V, Issue I.* *doi: 10.52933/jdssv.v5i1.123*

# Bootstrap Aggregated Designs for Generalized Linear Models

**Nicholas Rios**
George Mason University

**John Stufken**
George Mason University

**Abstract**

Many experiments require modeling a non-Normal response. In particular, count responses and binary responses are quite common. The relationship between predictors and the responses are typically modeled via a Generalized Linear Model (GLM). Finding D-optimal designs for GLMs, which reduce the generalized variance of the model coefficients, is desired. A common approach to finding optimal designs for GLMs is to use a locally optimal design, but these designs are vulnerable to parameter misspecification. The focus of this paper is to provide designs for GLMs that are robust to parameter misspecification. This is done by applying a bagging procedure to pilot data, where the results of many locally optimal designs are aggregated to produce an approximate design that reflects the uncertainty in the model coefficients. Results show that the proposed bagging procedure is robust to changes in the underlying model parameters. Furthermore, the proposed designs are shown to be preferable to traditional methods, which may be over-conservative.

*Keywords*: Design of experiments, generalized linear models, bootstrap aggregation.

## 1. Introduction

In many experiments, researchers are interested in modeling the effect of input factors on a response that does not follow the Normal distribution. Two common scenarios are binary responses and count responses. One timely example of a binary response is given

in Ouyang et al. (2016), where electronic medical records are examined to study the relationship between patient risk factors and disease status. Another modern example is using experimental designs to examine the probability of software failure (Salem et al. 2004). A good example of count data is given by Van Mullekom and Myers (2001) and Oris and Bailer (1993), who used Poisson regression to model how the number of fish eggs was impacted by concentrations of chemical toxicants in water.

While the bulk of traditional design methods in textbooks focuses on the linear model, much work has been done to find optimal designs for Generalized Linear Models (GLMs). The most common example of a GLM is the case of a binary response, which was one of the first cases to be studied. Sitter (1992) developed a minimax class of designs with the aim of minimizing the maximum loss over a range of plausible values for the parameters in a regression model with binary responses and two predictors. Chaloner and Larntz (1989) used a Bayesian framework to search for robust $D-$optimal designs for a logistic regression model. More recently, Denman et al. (2011) used Copulas to construct optimal designs for the case of two or more dependent binary responses.

Several authors have studied optimal designs for count data as well. In particular, the focus has been on $D-$optimal designs. Wang et al. (2006a) obtained locally $D-$optimal designs for Poisson regression models that included main effects and interactions for two toxicants. To address how local designs are sensitive to parameter misspecification, Wang et al. (2006b) studied sequential $D-$optimal designs for Poisson regression and demonstrated that it is difficult to analytically form the Fisher Information matrix in a sequential framework. Russell et al. (2009) found $D-$optimal designs for Poisson regression for any number of predictors in an additive model, and they proposed a clustering algorithm to create designs that were robust to parameter misspecification.

In this paper, we propose a Bootstrap Aggregated (BAG) approach for constructing designs for GLMs. This approach uses bootstrap resamples from prior experimental data to address the parameter dependence problem in finding optimal designs. Due to the nature of bagging, this is a flexible approach that can be applied to many GLMs. It will be shown that the BAG designs are more robust to parameter misspecification than local designs. This approach also allows experimenters to estimate the sensitivity of the design weights to changes in the model coefficients. Furthermore, this approach does not require experimenters to specify feasible regions or candidates for the unknown model parameters.

The rest of this paper is organized as follows. Section 2 reviews critical concepts needed to understand the problem and the proposed methodology. Section 3 describes the BAG design process in detail. Section 4 provides empirical evidence for the utility of the proposed methods. Section 5 concludes the paper.

## 2. Preliminaries

In order to discuss the proposed methodology, it is helpful to first review GLMs, their associated Fisher information matrices, and the criteria for optimal designs.

## 2.1. Generalized linear models

Suppose that we observe responses $y_i$ and predictors $\mathbf{x}_i$, where $i = 1, \ldots, N$ and each $\mathbf{x}_i$ is a $p \times 1$ vector. In the GLM framework, the traditional assumption that the response is Normally distributed is relaxed. The model is written as

$$E[y_i \mid \mathbf{x}_i] = \mu_i = g^{-1}(\mathbf{x}_i^T \boldsymbol{\beta}), \quad y_i \sim f, \tag{1}$$

where $\boldsymbol{\beta}$ is a $p \times 1$ vector of regression parameters to be estimated, $f$ is the probability density (or mass) function corresponding to $y_i$, $\mu_i$ is the conditional mean, and $g$ is a link function that specifies the relationship between the mean and the linear predictor $\eta_i = \mathbf{x}_i^T \boldsymbol{\beta}$.

We are particularly interested in GLMs for binary and count data. For binary responses, $y_i \sim \text{Bern}(\pi_i)$. It follows that $\mu_i = \pi_i$. A commonly used link function for binary data is the logit, i.e., $g(\pi_i) = \log(\pi_i/(1-\pi_i))$ which has the corresponding inverse $g^{-1}(\eta_i) = \exp(\eta_i)/(1+\exp(\eta_i))$. For count data, we assume that $y_i \sim \text{Poisson}(\lambda_i)$. It follows that $\mu_i = \lambda_i$. A commonly used link function for count data is the log-link $g(\lambda_i) = \log(\lambda_i)$, which has the corresponding inverse $g^{-1}(\eta_i) = \exp(\eta_i)$.

In the GLM framework, it is well-known that the maximum likelihood estimators (MLEs) for $\boldsymbol{\beta}$ have an asymptotic variance-covariance matrix that is the inverse of the Fisher information matrix. The Fisher information matrix, in general, may be written as

$$I(\boldsymbol{\beta}) = \mathbf{X}^T \mathbf{V} \mathbf{X}, \tag{2}$$

where $\mathbf{X}$ is an $N \times p$ matrix of covariates (often called the model matrix) and $\mathbf{V}$ is a diagonal matrix with entries

$$\mathbf{V}_{ii} = \frac{1}{\text{Var}[y_i \mid \mathbf{x}_i]} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2. \tag{3}$$

In particular, the diagonal entries for the two models assumed here are given below in Equations (4) and (5), respectively.

$$\mathbf{V}_{ii} = \frac{\exp(\mathbf{x}_i^T \boldsymbol{\beta})}{(1+\exp(\mathbf{x}_i^T \boldsymbol{\beta}))^2}, \tag{4}$$

$$\mathbf{V}_{ii} = \exp(\mathbf{x}_i^T \boldsymbol{\beta}). \tag{5}$$

In particular, note that the entries of $\mathbf{V}$ depend on $\boldsymbol{\beta}$, which is unknown a-priori.

## 2.2. Optimal designs

To fit a generalized linear model of the form (1), several data points need to be collected. In an experimental design framework, the experimenters have the ability to choose specific values of the $p$ covariates used to fit the generalized linear model. Let $\chi$ denote a $p-$dimensional experimental region, which is a subset of $\mathbb{R}^p$ of values for the covariates. Typically, $\chi$ is discretized into $N$ points, referred to as candidate points, that are feasible to run in real-world conditions. Since the experimenters have the ability to choose the points used to fit the model, they need to decide which candidate points to include in the model, and how many times each candidate point should be replicated. Suppose we

have a matrix $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_N]^T$ of dimension $N \times p$ whose rows are candidate points that lie in a $p-$dimensional experimental region $\chi$. An experimental design is a set of the form $\{(\mathbf{z}_j, w_j) \mid \sum_{j=1}^N w_j = 1, w_j \geq 0, j = 1, \ldots, N\}$. Here, $w_j$ is interpreted as the proportion of the $N$ runs that should be allocated to the candidate point $\mathbf{z}_j$. If each proportion $w_j$ is required to be an integer multiple of $1/N$, then the design is called an exact design; otherwise, it is an approximate design. Candidate points with weights greater than 0 are called support points. The focus of this work is on approximate designs, as they are easier to numerically and theoretically optimize. An example of an approximate design is given in Table 1. Here, there are 9 candidate points that lie in $\chi = [-1, 1]^2$ for $p = 2$ covariates. This design has four support points, which are $(-1, -1), (-1, 1), (1, -1)$, and $(1, 1)$.

Table 1: An example of an approximate design for $p = 2$ covariates.

| Candidate Point | (-1,-1) | (-1,0) | (-1,1) | (0,-1) | (0,0) | (0,1) | (1,-1) | (1,0) | (1,1) |
|---|---|---|---|---|---|---|---|---|---|
| Weight | 0.24 | 0 | 0.23 | 0 | 0 | 0 | 0.27 | 0 | 0.26 |

In the case of GLMs, an optimal approximate design is one that minimizes the asymptotic variance of $\hat{\boldsymbol{\beta}}$ in some sense. The asymptotic variance of $\hat{\boldsymbol{\beta}}$ depends on the information matrix of the design, which is denoted as

$$I(\mathbf{w}, \boldsymbol{\beta}) = \sum_{j=1}^{N} w_j \mathbf{z}_j \mathbf{V}_{jj} \mathbf{z}_j^T = \mathbf{Z}^T \mathbf{V} \mathbf{W} \mathbf{Z}, \tag{6}$$

where $\mathbf{W} = \text{diag}(w_1, \ldots, w_N)$. One of the most common approaches is to find weights that are $D-$optimal, i.e., minimize the generalized variance of $\hat{\boldsymbol{\beta}}$, which corresponds to maximizing the determinant of the Fisher information matrix (Sitter 1992; Wang et al. 2006a). These designs minimize the expected volume of a confidence ellipsoid for $\boldsymbol{\beta}$, and are desirable for inference (Russell et al. 2009). Let $\mathcal{S}^N = \{(w_1, \ldots, w_N) : \sum_{j=1}^N w_j = 1, w_j \geq 0\}$ be an $(N-1)$ dimensional simplex. In this case, the D-optimal design corresponds to the weights

$$\mathbf{w}^* = \underset{\mathbf{w} \in \mathcal{S}^N}{\arg\min} \log(|(\mathbf{Z}^T \mathbf{V} \mathbf{W} \mathbf{Z})^{-1}|). \tag{7}$$

A complication arises in the optimal design problem (7). Notably, the matrix $\mathbf{V}$ depends on the regression parameters $\boldsymbol{\beta}$, which are unknown at the design phase. As stated in Section 1, several attempts have been made to resolve this parameter dependence problem. One common approach is to find a locally D-optimal design; here, domain knowledge or prior data is used to construct an educated guess for $\boldsymbol{\beta}$, and then an optimal design is found (Wang et al. 2006a). Of course, the local method is sensitive to the assumed value of $\boldsymbol{\beta}$.

Much existing work has been done on finding locally optimal designs for GLMs under a variety of different criteria. Mathew and Sinha (2001) found locally $D-$ and $A-$optimal designs for two-parameter logistic regression models, but some of these designs needed to have symmetric design points. Later, Yang (2008) proposed an algebraic method for finding optimal designs for two-parameter GLMs under the $A-$optimality criterion.

Yang et al. (2011) identified general classes of optimal designs with $2^p$ support points under the $D-$, $A-$, and $E-$optimality criteria for the logistic and probit GLMs. Stufken and Yang (2012) discussed how locally optimal designs can be found by restricting the search to particular subclasses of designs known as complete classes. These complete classes were found for multi-parameter Poisson and logistic regression models, and they typically consist of designs with at least $p + 1$ support points. Recent developments have also been made for finding locally optimal designs for GLMs. Lukemire et al. (2019) proposed a modified quantum-behaved particle swarm optimization algorithm to find $D-$optimal approximate designs for binary response GLMs where the covariates are both categorical and continuous. Li and Deng (2021) proposed a hybrid algorithm that provides designs for GLMs that minimize the expected integrated prediction variance over a subregion of interest. Wang and Stufken (2022) found $D-$optimal designs for binary response GLMs that allowed for interactions between continuous covariates that satisfy the strong effect heredity principle, and they also used orthogonal arrays to construct $D-$optimal designs with fewer support points.

A promising approach for robust designs for GLMs is clustering. Dror and Steinberg (2006) examined designs that were found by clustering the support points of locally optimal designs for Poisson and binomial GLMs. The locally optimal designs were found using a greedy exchange algorithm. A similar approach was used in Russell et al. (2009), who considered clustering locally optimal designs for a Poisson regression model. Locally optimal designs were found for several user-specified values of $\boldsymbol{\beta}$, and the support points were clustered together. This was done in the context of equally weighted designs. A generalization of this procedure is summarized in Algorithm 1 below.

**Algorithm 1:** A Generalization of the Cluster Design from Russell et al. (2009).

**Inputs:** A grid of values $\mathcal{B}$ for $\boldsymbol{\beta}$, Candidate points $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_N]^T$, Number of support points $k$ for the final design.

**for** $\boldsymbol{\beta} \in \mathcal{B}$ **do**

1. Find a locally optimal design with weights $\mathbf{w}$ for the candidates $\mathbf{Z}$ given $\boldsymbol{\beta}$.
2. For $j = 1, \ldots, N$, replicate each candidate $\mathbf{z}_j$, $\lceil N w_j \rceil$ times and store these row-wise in a matrix $\mathbf{D}$, where $\lceil \cdot \rceil$ is the integer ceiling.

**end**

3. Use k-means (or another clustering algorithm) to cluster the rows of $\mathbf{D}$ into $k$ clusters.

**return** A design with the cluster means as equally weighted support points.

Algorithm 1 is implemented later in Section 4 for comparison with the proposed methods in Section 3. The original implementation used a theoretical result to generate optimal designs for each $\boldsymbol{\beta}$; the generalized version allows for any algorithm that produces optimal weights for a given set of candidates to be used. The generalized implementation uses k-means clustering, but any clustering method could be applied in practice.

Another approach is the optimal "on-the-average" design (Fedorov and Hackl 1997). In this framework, the experimenters specify a distribution $\pi(\beta)$, where $\pi$ is selected to emphasize importance of certain values of $\boldsymbol{\beta} \in \mathcal{B}$. Then, the following optimization

problem is solved to find the design weights:

$$\mathbf{w}^* = \arg\max_{\mathbf{w}\in\mathcal{S}^N} \int_{\boldsymbol{\beta}\in\mathcal{B}} \log(|\mathbf{Z}^T\mathbf{V}\mathbf{W}\mathbf{Z}|)\ d\pi(\boldsymbol{\beta}). \tag{8}$$

The optimal weights identified in (8) maximize the average $D-$optimality criterion with respect to the distribution $\pi$. A similar approach for considering model uncertainty was given by Woods et al. (2006), who proposed integrating the $D-$optimality criterion over a set of possible link functions, linear predictors, and parameter vectors.

Finally, a popular robust alternative is a minimax or maximin design (Sitter 1992; King and Wong 2000), which would focus on the following similar optimization problem:

$$\mathbf{w}^* = \arg\max_{\mathbf{w}\in\mathcal{S}^N} \left( \min_{\boldsymbol{\beta}\in\mathcal{B}} |\mathbf{Z}^T\mathbf{V}\mathbf{W}\mathbf{Z}| \right),$$

where $\mathcal{B}$ is a pre-specified set of plausible values for the unknown regression parameters. Welch (1983) and Chipman and Welch (1996) provide a method for finding approximate maximin designs over a grid of pre-specified values of $\boldsymbol{\beta}$. Suppose there are $n_\beta$ possible values of $\boldsymbol{\beta}$ in the grid. This method works as follows:

1. The locally $D-$optimal design weights are found for each of the $n_\beta$ values of $\boldsymbol{\beta}$; denote these as $\mathbf{w}_1, \ldots, \mathbf{w}_{n_\beta}$.

2. For each $\boldsymbol{\beta}_i, i = 1, \ldots, n_\beta$, the relative D-efficiency of the $j^{th}$ design to the $i^{th}$ design is

$$\text{relD}(j \mid i) = \frac{|\mathbf{Z}^T\mathbf{V}(\boldsymbol{\beta}_i)\text{diag}(\mathbf{w}_j)\mathbf{Z}|}{|\mathbf{Z}^T\mathbf{V}(\boldsymbol{\beta}_i)\text{diag}(\mathbf{w}_i)\mathbf{Z}|}$$

   for $j = 1, \ldots, n_\beta$. The relative $D-$efficiencies $\text{relD}_i^* = \min_{j=1,\ldots,n_\beta} \text{relD}(j \mid i)$ are stored, as they represent the worst case for that particular value of $\boldsymbol{\beta}_i$.

3. The maximin design corresponds to $\mathbf{w}_{a^*}$, where $a^* = \arg\max_{i=1,\ldots,n_\beta} \text{relD}_i^*$.

In Step 2, the term $\text{relD}(j \mid i)$ is the relative $D-$efficiency of the $j^{th}$ design to the $i^{th}$ design. If this relative efficiency is low for the $j^{th}$ design compared to that of other designs, then the $j^{th}$ design is a poor design for $\boldsymbol{\beta}_i$. The candidate parameter vector that minimizes this relative $D-$efficiency is then considered to be the worst-case possible value of $\boldsymbol{\beta}$ for a locally optimal design constructed using $\boldsymbol{\beta}_i$. In Step 3, the maximin design is approximated by choosing the weights that correspond to the design with the largest worst-case relative $D-$efficiency. The maximin design is appealing because it maximizes the Fisher information for the worst possible choice of $\boldsymbol{\beta}$; however, it is often more computationally intense than finding a locally optimal design.

Bagging has seen limited use before in optimal sampling frameworks. However, to the best of our knowledge, bootstrap methods have not been applied to finding optimal designs for GLMs. Recently, Rha et al. (2021) proposed the use of bagging for optimal sampling for functional linear models in functional data analysis. While conceptually similar to the proposed work here, the designs in Rha et al. (2021) are concerned with minimizing the mean squared prediction error; ours are focused on minimizing the generalized variance in a GLM.

# 3. Proposed Methods

In this section, a method for identifying robust weights and support points is proposed that leverages existing pilot data. This procedure is called a Bootstrap Aggregated Design (BAG design), and it is summarized below in Algorithm 2.

**Algorithm 2:** Find BAG Design Weights.

**Inputs:** Pilot Data $(\mathbf{X}, Y)$, number of bootstrap samples $B$, Candidate points $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_N]^T$, aggregation function $A : \mathcal{S}^{N \times B} \to \mathcal{S}^N$.

**for** $i = 1, \ldots, B$ **do**

1. Create a bootstrap sample $(\mathbf{X}^{(i)}, Y^{(i)})$ by sampling the rows of the data $(\mathbf{X}, Y)$.
2. Fit a GLM to the bootstrap sample in Step 1 to obtain an estimate of the coefficients $\hat{\boldsymbol{\beta}}^{(i)}$.
3. Given $\hat{\boldsymbol{\beta}}^{(i)}$, find and store the weights for the locally optimal design $\mathbf{w}^{(i)}$.

**end**

4. Let $\mathbf{W}_{all} = [\mathbf{w}^{(1)}, \ldots, \mathbf{w}^{(B)}]$ be an $N \times B$ matrix. Aggregate the weights:

$$\mathbf{w}_{BAG} = A(\mathbf{W}_{all}).$$

**return** $\boldsymbol{w}_{BAG}$

As inputs, Algorithm 2 takes an $n \times p$ matrix of predictors $\mathbf{X}$ from pilot data, an $n \times 1$ vector $Y$ of responses from pilot data, the number of bootstrap samples $B$, an $N \times p$ matrix of candidate points $\mathbf{Z}$, and an aggregation function $A$ that takes an $N \times B$ matrix of weights and returns a vector of $N$ weights. In Steps 1, 2 and 3 of Algorithm 2, $B$ bootstrap samples of size $n$ are taken from the pilot data, which also has a sample size of $n$. This is done by sampling rows from the pilot data with replacement. For each of the $B$ resamples, the coefficients $\boldsymbol{\beta}$ are estimated. Then, locally optimal weights for each of the $N$ support points are found for each $\hat{\boldsymbol{\beta}}$. In Step 4, these weights for the locally optimal design are aggregated using the aggregation function $A(\cdot)$. Two versions of the aggregation function are proposed. The first is the mean, i.e.,

$$\mathbf{w}_{BAG} = \frac{1}{B} \sum_{i=1}^{B} \mathbf{w}^{(i)}. \tag{9}$$

The second is the geometric median, i.e.,

$$\mathbf{w}_{BAG} = \underset{\mathbf{w} \in \mathcal{S}^N}{\arg\min} \sum_{i=1}^{B} ||\mathbf{w}^{(i)} - \mathbf{w}||_2, \tag{10}$$

which is the vector in $\mathcal{S}^N$ that minimizes the sum of the $L_2$ distances to all other bootstrapped weight vectors $\mathbf{w}^{(i)}, i = 1, \ldots, B$. Algorithm 2 outputs the weights $\mathbf{w}_{BAG}$ for a robust design on the candidate points. The Bootstrap Aggregated Design (BAG design) procedure is summarized in Figure 1 for the case when the mean is used for aggregation.

To find weights for the locally optimal design for each of the bootstrap samples, a fast algorithm is required. One could use a multiplicative algorithm (Yu 2011; Goos et al.

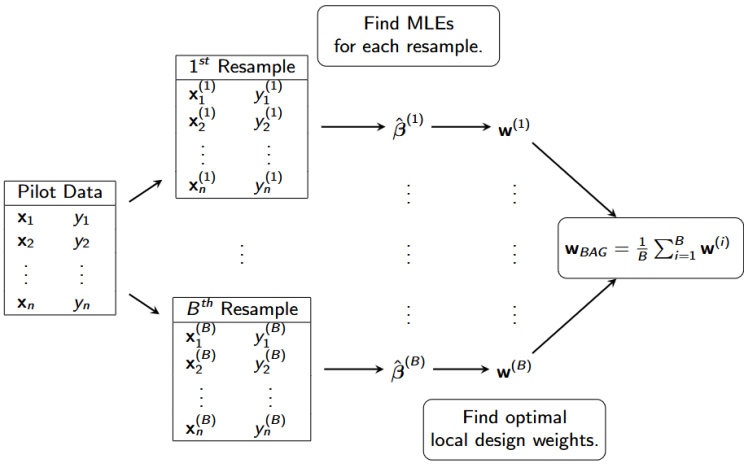


Figure 1: The BAG Design Process, with Mean Aggregation.

2016) to find optimal weights for all $N$ candidates, but this is not ideal because the multiplicative algorithm often takes too many iterations to identify the weights for a locally optimal design. In Algorithm 2, weights for locally optimal designs would need to be computed for each of the $B$ bootstrap resamples, so the multiplicative algorithm would be inefficient. It would be more efficient to use an algorithm that works on a subset of the $N$ candidates $\mathbf{Z}$, and updates both the weights and the subset with each iteration.

To this end, we implement the algorithm from Yang et al. (2013), which allows for fast identification of optimal designs for a given $\hat{\boldsymbol{\beta}}$. A high-level overview of this procedure is provided in Algorithm 3. Algorithm 3 uses a directional derivative $\partial\phi(I(\mathbf{w},\boldsymbol{\beta}))/\partial\mathbf{z}_j$ of the optimality criterion $\phi$. For the case of $D-$optimality, this is $\partial\phi(I(\mathbf{w},\boldsymbol{\beta}))/\partial\mathbf{z}_j = \mathbf{V}_{jj}\mathbf{z}_j^T I(\mathbf{w},\boldsymbol{\beta})^{-1}\mathbf{z}_j - p$. Note that in Algorithm 2, it is only required to find optimal weights for fixed values of $\boldsymbol{\beta}$, so the directional derivative can be clearly evaluated.

As inputs, Algorithm 3 takes the set of $N$ candidate points $\mathbf{Z}$, a vector of coefficients $\boldsymbol{\beta}$, a generalized linear model specified by a link function $g$ and a density $f$ of the response, and a small positive tolerance $\epsilon$. This algorithm returns a matrix of candidate points $\mathbf{Z}^{(t+1)}$ whose rows are a subset of the rows of $\mathbf{Z}$ and corresponding weights for each of the support points. During each iteration of the algorithm, a Newton-type algorithm is used to find the optimal weights for the current subset of support points (Step 5). More details behind the implementation of this algorithm can be found in Yang et al. (2013). It is important to note that when finding the optimal weights, this Newton algorithm should remove points that have zero weight. Then, the candidate that maximizes the directional derivative is identified (Step 6). If the directional derivative is below some small $\epsilon$, then the algorithm terminates, as the optimal design has been found (Step 7). Otherwise, the candidate is added to the set of support points, and the previous step repeats. Hence, at each iteration, at most one point is added. During Step 5, support points with weights equal to 0 will be removed. By running Algorithm 3 for each bootstrap resample in Algorithm 2, one can construct $\mathbf{w}^{(i)}$ by assigning the weights $\mathbf{w}_{local}^{(t+1)}$ to the entries of $\mathbf{w}^{(i)}$ that correspond to the elements of $\mathbf{Z}^{(t+1)}$, keeping all other weights zero for that iteration.

**Algorithm 3:** Find Locally $D-$Optimal Design (Yang et al. 2013).

**Input**: Candidate points $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_N]^T$, coefficients $\boldsymbol{\beta}$, a link function $g$ and density $f$ as specified in Model (1), and a tolerance $\epsilon > 0$.

1. Set $t = 0$ and select $p + 1$ candidate points $\mathbf{Z}^{(0)}$ from the rows of $\mathbf{Z}$ such that $\text{rank}(\mathbf{Z}^{(0)}) = p$.
2. Let

$$\mathbf{V}_{jj} = \left(\frac{\partial g^{-1}(\mathbf{z}_j^T\boldsymbol{\beta})}{\partial(\mathbf{z}_j^T\boldsymbol{\beta})}\right)^2 \frac{1}{\text{Var}_f[y_j \mid \mathbf{z}_j]}, \quad j = 1, \ldots, N.$$

**while** $\delta > \epsilon$ **do**

3. Let $q$ be the number of rows of $\mathbf{Z}^{(t)}$.
4. Let $\mathbf{V}^{(t)}$ be a $q \times q$ diagonal matrix with diagonal entries of the form

$$\left(\frac{\partial g^{-1}(\mathbf{z}^T\boldsymbol{\beta})}{\partial(\mathbf{z}^T\boldsymbol{\beta})}\right)^2 \frac{1}{\text{Var}_f[y \mid \mathbf{z}]}$$

for each row $\mathbf{z}$ of $\mathbf{Z}^{(t)}$.

5. Find $\mathbf{w}_{local}^{(t+1)} = \arg\min_{\mathbf{w}\in\mathcal{S}^q} \log(|[(\mathbf{Z}^{(t)})^T\mathbf{V}^{(t)}\text{diag}(\mathbf{w})\mathbf{Z}^{(t)}]^{-1}|)$ by using a Newtonian algorithm.
6. Let $j^* = \arg\max_{j=1,\ldots,N} \mathbf{V}_{jj}\mathbf{z}_j^T I(\mathbf{w}_{local}^{(t+1)}, \boldsymbol{\beta})^{-1}\mathbf{z}_j - p$.
7. Update $\delta = \mathbf{V}_{j^*j^*}\mathbf{z}_{j^*}^T I(\mathbf{w}_{local}^{(t+1)}, \boldsymbol{\beta})^{-1}\mathbf{z}_{j^*} - p$.
8. Append $\mathbf{z}_{j^*}$ row-wise to $\mathbf{Z}^{(t)}$ to form $\mathbf{Z}^{(t+1)}$. Add a weight of 0 to $\mathbf{w}_{local}^{(t+1)}$ for this point.
9. Update $t = t + 1$.

**end**

**return** $\boldsymbol{w}_{local}^{(t+1)}, \boldsymbol{Z}^{(t+1)}$

The BAG design procedure has many advantages. First, it is applicable to any GLM with an information matrix that has a closed form. It is particularly advantageous when locally optimal designs can be quickly identified. By aggregating the weights of $B$ bootstrap resamples, we can obtain weights for an optimal design that reflect the uncertainty associated with the original MLE for $\boldsymbol{\beta}$. Also, the BAG design can accommodate constraints in the experimental region. If the candidate points satisfy the experimental constraints, then the resulting BAG design is also feasible. Theorem 1 below provides an additional benefit for the BAG design, in the event that there is no model misspecification, and helps describe its asymptotic behavior as the pilot sample size increases.

**Theorem 1 (Asymptotic efficiency of the BAG design).** *Let the true parameter vector be $\boldsymbol{\beta} \in \mathbb{R}^p$. Suppose that there is no parameter misspecification, i.e., $E[y_i \mid \boldsymbol{x}_i] = g^{-1}(\boldsymbol{x}_i^T\boldsymbol{\beta})$ for $i = 1, \ldots, n$, where $n$ is the size of the pilot dataset. If Algorithm 2 is used with mean aggregation to find $\boldsymbol{w}_{BAG}$ for candidate points $\boldsymbol{z}_i, i = 1, \ldots, N$, it follows that, as $n \to \infty$,*

$$\log(|I(\boldsymbol{w}_{BAG}, \boldsymbol{\beta})|) \underset{p}{\to} \log(|I(\boldsymbol{w}^*, \boldsymbol{\beta})|).$$

Theorem 1 shows that as the size of the pilot dataset increases, the log $D-$efficiency of the BAG design evaluated at the true parameter vector $\boldsymbol{\beta}$ converges to the log $D-$efficiency of the optimal design in probability. Therefore, the chance of identifying an optimal design increases with the size of the pilot data.

Alternative robust designs presume knowledge of plausible values of $\boldsymbol{\beta}$, which may not always be available or may be incorrectly specified. A minimax or maximin optimal design often requires the user to specify either (1) feasible bounds for $\boldsymbol{\beta}$ or (2) a set of feasible candidate parameter vectors for $\boldsymbol{\beta}$. The BAG design does not require the user to pre-specify any bounds, nor does it require candidate parameter vectors. In a sense, the candidate parameter vectors are automatically generated via bootstrapping. The BAG design is also more intuitive. A Bayesian optimal design requires that a prior for $\boldsymbol{\beta}$ is specified, while this is not a requirement for the BAG design. In classical literature, a conjugate prior is often used for $\boldsymbol{\beta}$, but a conjugate prior may not always exist for a given link function $g$ (e.g., logistic regression), and even if it does exist, there is no guarantee that it is the correct prior distribution.

There are some concerns that must be addressed when dealing with the BAG design. While the BAG design requires more computation time than a single locally optimal design, this can be mitigated by finding optimal designs for different bootstrap samples in parallel. The other drawback of the BAG design is that it requires pilot data, which may not always be available at the design stage. However, the use of pilot data provides more realistic values of $\boldsymbol{\beta}$ to consider - a minimax or Bayesian approach might consider values of $\boldsymbol{\beta}$ that are too extreme and result in a design that is over-conservative.

## 3.1. Reducing the number of support points

As written, Algorithm 2 often produces approximate designs that have too many support points. This happens because some of the support points may only be selected by a small number of bootstrap resamples. For instance, if one candidate point is selected by only one of $B = 50$ bootstrap samples, then the weight of this candidate point in the final BAG design must be less than $1/B = 0.02$. In general, this can lead to designs with too many support points, each with a weight that may not be easy to convert to an exact design for a fixed sample size. To address this concern, we consider a modification of Algorithm 2 that filters out support points with low weights. Let $\mathbf{w}_{BAG}$ be the weights returned by Algorithm 2. The modified weights are given by

$$w_j^{**} = \frac{w_{BAG,j} I(w_{BAG,j} \geq \gamma)}{\sum_{\ell=1}^{N} w_{BAG,\ell} I(w_{BAG,\ell} \geq \gamma)}, \tag{11}$$

where $j = 1, 2, \ldots, N$ is an index over the candidate points, $w_{BAG,j}$ is the weight of the $j^{th}$ candidate point, $\gamma$ is a small tolerance level that satisfies $0 \leq \gamma \leq \max_{j=1,\ldots,N} w_{BAG,j}$, and $I(w_{BAG,j} \geq \gamma)$ is the indicator function that is equal to 1 if the weight of the $j^{th}$ candidate is greater than or equal to $\gamma$. The purpose of the tolerance parameter $\gamma$ is to remove support points with low average weights from the design. If $w_{BAG,j} < \gamma$, then $(1/B)\sum_{i=1}^{B} w_j^{(i)} < \gamma$. Therefore, if the average weight of candidate point $\mathbf{z}_j$ over the $B$ bootstrap resamples is less than $\gamma$, $w_j^{**} = 0$.

It is straightforward to show that increasing $\gamma$ will either maintain or decrease the number of support points in the design. In general, for any set of weights $\mathbf{w} \in \mathcal{S}^N$, let

$w_{\gamma,j} = w_j I(w_j \geq \gamma)/(\sum_{\ell=1}^{N} w_\ell I(w_\ell \geq \gamma))$. Suppose $0 \leq \gamma_1 \leq \gamma_2 < \max_{j=1,\ldots,N} w_j$. Let $S_1 = \{\mathbf{z}_j \mid w_{\gamma_1,j} > 0,\ j = 1,\ldots,N\}$ and $S_2 = \{\mathbf{z}_j \mid w_{\gamma_2,j} > 0,\ j = 1,\ldots,N\}$. $S_1$ and $S_2$ are the support points for designs with tolerance parameters $\gamma_1$ and $\gamma_2$, respectively. It follows that $S_2 \subset S_1$ and that $|S_2| \leq |S_1|$.

The effect of trimming the BAG weights on the log $D-$efficiency can also be shown. Consider arbitrary weights $\mathbf{w} \in \mathcal{S}^N$, and let $\mathbf{w}_\gamma$ be defined as above for $0 < \gamma < \max_{j=1,\ldots,N} w_j$. Let $c_\gamma = \sum_{\ell=1}^{N} w_\ell I(w_\ell \geq \gamma)$. Then, since $w_j \geq w_j I(w_j \geq \gamma)$, it follows that $w_j \geq c_\gamma w_{\gamma,j}$ for all $j = 1,\ldots,N$. Therefore,

$$\log(|I(\mathbf{w},\boldsymbol{\beta})|) = \log\left(\left|\sum_{j=1}^{N} w_j V_{jj}(\boldsymbol{\beta})\mathbf{z}_j\mathbf{z}_j^T\right|\right) \tag{12}$$

$$\geq \log\left(\left|c_\gamma \sum_{j=1}^{N} w_{\gamma,j} V_{jj}(\boldsymbol{\beta})\mathbf{z}_j\mathbf{z}_j^T\right|\right) = p\log(c_\gamma) + \log(|I(\mathbf{w}_\gamma,\boldsymbol{\beta})|). \tag{13}$$

This implies that $\log(|I(\mathbf{w},\boldsymbol{\beta})|) - \log(|I(\mathbf{w}_\gamma,\boldsymbol{\beta})|) \geq p\log(c_\gamma)$. Since $0 < c_\gamma \leq \sum_{\ell=1}^{N} w_\ell = 1$, it follows that $p\log(c_\gamma) \leq 0$, with equality at $\gamma = 0$. A tolerance of $\gamma = 0$ is identical to the regular BAG design. In Section 4, the impact of the modification to the weights in (11) on the relative $D-$efficiency and number of support points will be examined.

# 4. Results

Simulations were used to study the performance of the BAG designs. The aim of the simulations was to evaluate the robustness of the BAG designs when compared to the cluster and maximin designs. We also investigated the number of support points used by the various designs across several simulated scenarios. Finally, we examined the impact of trimming the weights as described in Equation (11) on the relative efficiency and number of support points.

## 4.1. Robustness of BAG design

Simulations were used to compare the relative $D-$efficiencies of several designs (BAG, cluster, and maximin) relative to the true local design for known values of $\boldsymbol{\beta}$. All of the designs used the same grid of parameter vector candidates that were generated by bootstrap resampling. Values of $\gamma = 0, 0.01, 0.02$ were considered for the BAG designs. In particular, $\gamma = 0.02$ was selected because each BAG design used $B = 50$ bootstrap resamples, so a value of $\gamma = 0.02$ would ensure that candidate points that only appeared in one of the 50 bootstrap resamples were excluded from the design. For each scenario, pilot data of size $n$ (for $n = 50, 200$) were simulated. The $p = 2$ covariates were drawn from independent $N(0,\sigma)$ distributions, with $\sigma = 0.5, 2$. For each iteration, the true value of $\boldsymbol{\beta}$ was drawn from a uniform distribution on $[0,1]^{p+1}$. The pilot responses were drawn from either the binomial distribution or the Poisson distribution. In the binomial case, $Y_i \sim \text{Bernoulli}(p_i = \exp(X_i^T\boldsymbol{\beta})/(1+\exp(X_i^T\boldsymbol{\beta})))$ for $i = 1,\ldots,n$. In the Poisson case, $Y_i \sim \text{Poisson}(\lambda_i = \exp(X_i^T\boldsymbol{\beta}))$ for $i = 1,\ldots,n$. The candidate points were taken from $\mathbf{Z} = \{-1, -0.9, \ldots, 0.9, 1\}^p$. To examine how robust the designs are to parameter misspecification, a test set of 100 parameter vectors $\tilde{\boldsymbol{\beta}}_s, s = 1,\ldots,100$ were

randomly sampled from independent multivariate normal distributions $N(\boldsymbol{\mu}, \tilde{\sigma}^2 I_{p+1})$, where $\boldsymbol{\mu} \in \mathbb{R}^{p+1}$ is a mean vector that is outside of $[0,1]^{p+1}$. Two mean vectors were considered, and they are denoted as $\boldsymbol{\mu}_1 = (-3,-4,-5)^T$ and $\boldsymbol{\mu}_2 = (1,3,1)^T$; $\boldsymbol{\mu}_1$ was placed far from $[0,1]^{p+1}$, while in $\boldsymbol{\mu}_2$, only the second coordinate is outside of $[0,1]$. The relative log $D-$efficiency of the BAG, cluster, and maximin designs to the local design dramatically increases as the distance between $\boldsymbol{\beta}$ and $\boldsymbol{\mu}$ increases. If the test parameter vector $\tilde{\boldsymbol{\beta}}$ is far from $\boldsymbol{\beta}$, then a locally optimal design based on a good estimate of $\boldsymbol{\beta}$ would have poor $D-$efficiency when evaluated at $\tilde{\boldsymbol{\beta}}$. Since $\boldsymbol{\beta}$ is uniformly generated on $[0,1]^{p+1}$, $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ were selected so that the test parameter vectors would be close enough to the true $\boldsymbol{\beta}$ to ensure that the relative efficiencies did not diverge to infinity, but far away enough that it would still make sense to use a robust design. The scenario where $\boldsymbol{\mu} = \boldsymbol{\mu}_1$ represents a more extreme misspecification case, as $\boldsymbol{\mu}_1$ is completely outside of $[0,1]^{p+1}$ and has misspecified signs. In the other case, where $\boldsymbol{\mu} = \boldsymbol{\mu}_2$, the signs are correctly specified, but the mean vector is closer to $[0,1]^{p+1}$. Looking at both cases provides us with a sense of how well the BAG designs perform under these different magnitudes of parameter misspecification. In this section, $\tilde{\sigma}^2 = 0.25$ is used. Additional simulation results for $\tilde{\sigma}^2 = 1$ are provided in Appendix B, which is included in the supplementary materials. The factors used to specify these simulated scenarios are summarized in Table 2. Overall, there are $2^4 = 16$ simulated scenarios.

Table 2: Summary of Simulation Factors.

| Simulation Factor | Levels |
|---|---|
| Pilot Data Size ($n$) | Small (50), Large (200) |
| Standard Deviation of Covariates ($\sigma$) | Low (0.5), High (2) |
| Distribution of Response | Binomial, Poisson |
| Parameter Vector Test Distribution Mean ($\boldsymbol{\mu}$) | $\boldsymbol{\mu}_1 = (-3,-4,-5)^T$, $\boldsymbol{\mu}_2 = (1,3,1)^T$ |

Each scenario was simulated using the following procedure:

1. Randomly sample $\tilde{\boldsymbol{\beta}}_s \sim N(\boldsymbol{\mu}, \tilde{\sigma}^2 I_{p+1}), s = 1, \ldots, 100$, where $\boldsymbol{\mu}$ is one of the two mean vectors specified in Table 2.

2. Next, a pilot dataset was generated for a combination of $n, \sigma$, and a response distribution given in Table 2, with the true $\boldsymbol{\beta}$ randomly drawn from $U([0,1]^{p+1})$.

3. Given the pilot dataset, the MLE for $\hat{\boldsymbol{\beta}}$ was found and used to find a locally optimal design. The weights selected by the locally optimal design were stored for use in Step 5.

4. $B = 50$ bootstrap resamples were taken from the pilot dataset, and used as parameter vector candidates to construct the cluster, maximin, and BAG (for $\gamma = 0, 0.01, 0.02$) designs. The cluster design used $k = 5$.

5. For each $\tilde{\boldsymbol{\beta}}_s, s = 1, \ldots, 100$, the relative $D-$efficiencies of the BAG, cluster, and maximin designs were computed with respect to the local designs identified in Step 3. The average of these 100 relative $D-$efficiencies was stored for each

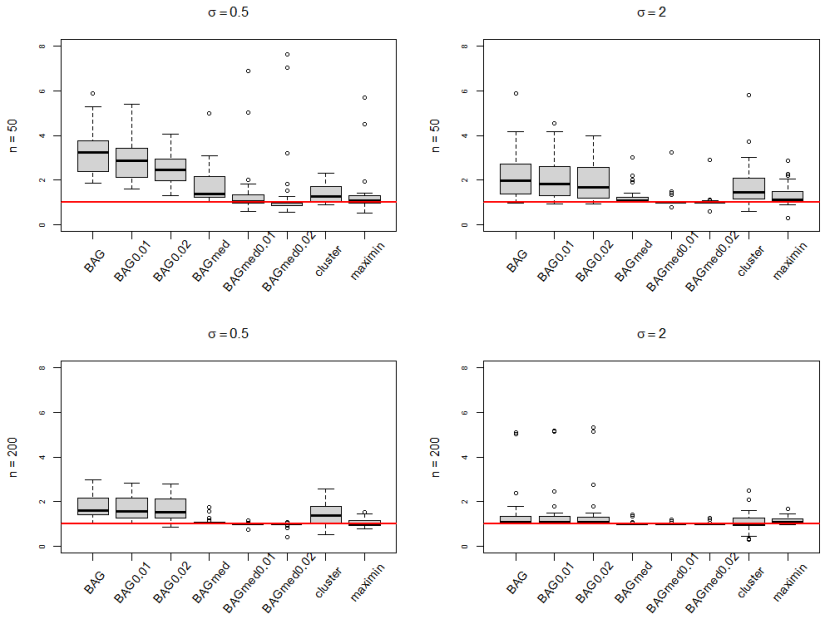


Figure 2: Average Relative $D-$efficiencies to Local Design, Binomial Response, $\boldsymbol{\mu}_1$

design construction method. This was done to see how robust the designs are under parameter misspecification.

6. Steps 2 to 5 were repeated 25 times. This was done to get a sense of how the BAG design (which is stochastic) performs on average for a given pilot dataset. The relative $D-$efficiencies were averaged across all 25 repetitions and stored.

7. Steps 2 to 6 were repeated 30 times. In total, 30 pilot datasets were created, and each design method had 30 average relative $D-$efficiencies, corresponding to the 30 pilot datasets. This was done to see how robust the designs were when the pilot dataset changed.

The relative $D-$efficiency of each of the designs was found with respect to the locally optimal design. The distributions of the average relative $D-$efficiencies for each of the 8 scenarios where $\boldsymbol{\mu} = \boldsymbol{\mu}_1 = (-3, -4, -5)^T$ are shown in Figures 2 and 3, for the Binomial and Poisson responses, respectively. In each figure, values above the red line indicate that the design has higher average $D-$efficiency than the locally optimal design. On the x-axis of the figures, "BAG 0.02" means that the BAG weights were adjusted as in (11) with $\gamma = 0.02$ with the mean aggregation function specified in (9); "BAG 0.01" has a similar meaning. "BAGmed" means that the BAG design was used with the geometric median aggregation function (10), and "BAGmed0.01" means that the BAG design was used with the geometric median aggregation function, and then the weights were trimmed with $\gamma = 0.01$.

In general, Figures 2 and 3 show that increasing the size of the pilot data reduces the variation in the relative $D-$efficiency of the BAG designs. In these figures, the horizontal red line is drawn at 1 on the y-axis, which corresponds with a relative $D-$efficiency of 1. In the Binomial case, when the standard deviation of the pilot covariates increases from $\sigma = 0.5$ to $\sigma = 2$, the median relative $D-$efficiency of the

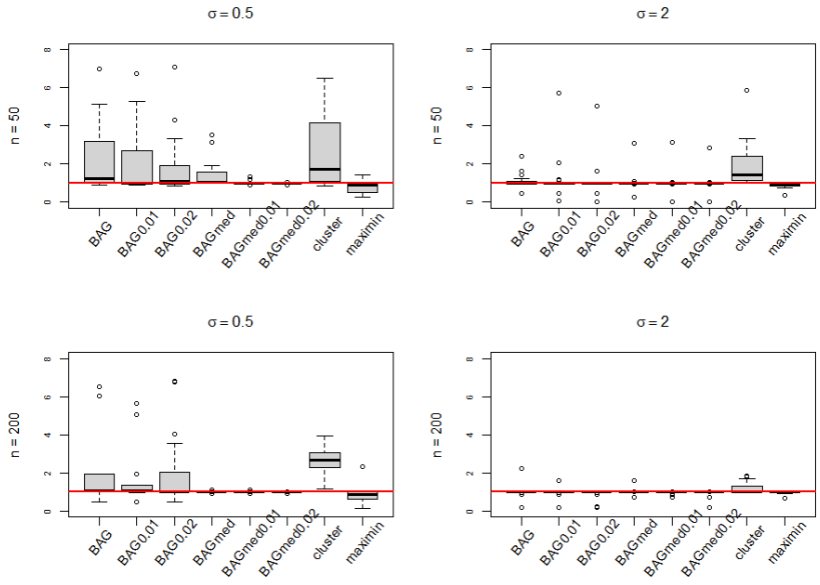


Figure 3: Average Relative $D-$efficiencies to Local Design, Poisson Response, $\boldsymbol{\mu}_1$.

BAG designs decreases. When the response was Binomial, the BAG designs that used the mean aggregation function had the highest median relative $D-$efficiencies. These are followed by the cluster designs, and then BAG designs with geometric median aggregation ($\gamma = 0$). This suggests that the BAG design compares well with the maximin and cluster designs in terms of robustness to parameter misspecification when $\boldsymbol{\mu} = \boldsymbol{\mu}_1$.

These simulations also allow us to examine the effect of $\gamma$, which was used to help filter out support points that had low weights. The plots reveal that, for the Binomial response, larger values of $\gamma$ led to designs with lower median relative $D-$efficiency to the local design. In most scenarios, increasing $\gamma$ also reduced the IQR of the relative $D-$efficiencies. These results make sense, since larger values of $\gamma$ lead to fewer support points in the BAG design. In the Poisson case, all of the BAG designs had similar median performance, especially in the case when $n = 200$.

Figures 4 and 5 show boxplots of the average relative $D-$efficiencies when $\boldsymbol{\mu} = \boldsymbol{\mu}_2$. In this scenario, only the second coordinate of $\boldsymbol{\mu}_2$ is far from the interval $[0, 1]$, so this scenario is less extreme than when $\boldsymbol{\mu} = \boldsymbol{\mu}_1$. This is reflected in Figures 4 and 5, since most of the relative $D-$efficiencies are close to 1; this indicates that the locally optimal design still performs quite well in this scenario. In Figure 4, when the response is Binomial, the BAG designs with geometric median aggregation performed the best in terms of relative $D-$efficiency. In the Poisson case, the performance of the BAG designs was similar regardless of aggregation function. In both the Poisson and Binomial cases, the BAG designs were preferred to cluster designs.

In terms of computation time, it is quite fast to find an individual BAG design. For example, when $\sigma = 0.5, n = 50$, and $\boldsymbol{\mu} = \boldsymbol{\mu}_1$ for the Poisson response, it takes an average of 0.752 minutes (45.12 seconds) to find the BAG design weights. In the same case, under the Binomial response, it takes an average of 1.312 minutes (78.72 seconds) to find the BAG design weights. Running the entire simulation for all 25 simulated

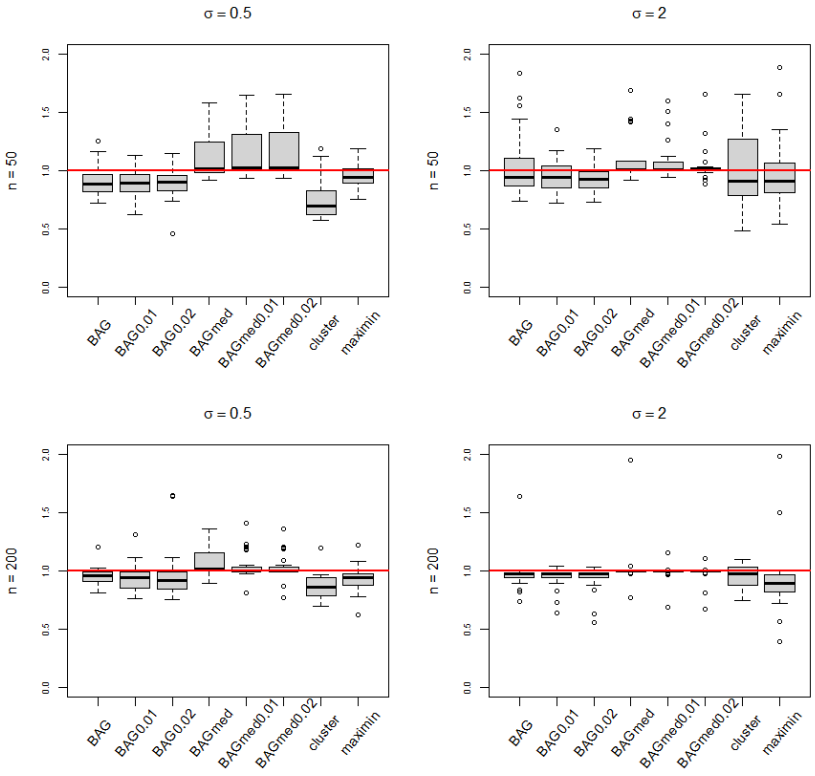


Figure 4: Average Relative $D-$efficiencies to Local Design, Binomial Response, $\boldsymbol{\mu}_2$.

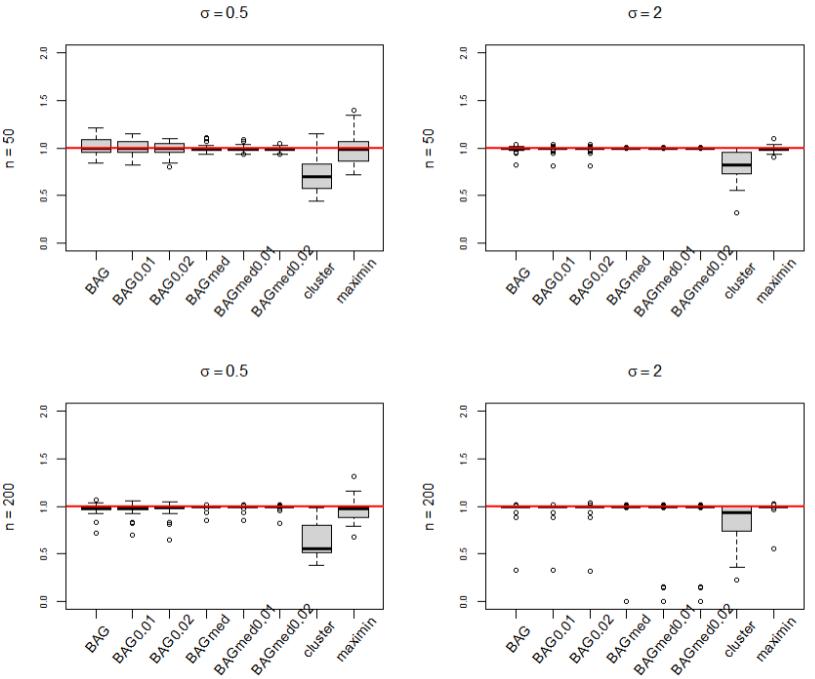


Figure 5: Average Relative $D-$efficiencies to Local Design, Poisson Response, $\boldsymbol{\mu}_2$

datasets is more efficient in parallel. For example, when $\sigma = 0.5, n = 50$, and $\boldsymbol{\mu} = \boldsymbol{\mu}_1$ for the Poisson response, the total computing time was 4 hours, 22 minutes, and 43 seconds, and the average computing time per simulated pilot dataset was 57.82 minutes.

Table 3: Average Support Size Comparison, Binomial Response, $\boldsymbol{\mu} = \boldsymbol{\mu}_1$.

| $n$ | $\sigma$ | Local | BAG (mean) | | | BAG (median) | | | Cluster | Maximin |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | $\gamma = 0$ | 0.01 | 0.02 | $\gamma = 0$ | 0.01 | 0.02 | | |
| 50 | 0.5 | 3.6 | 38.5 | 20.8 | 16.5 | 38.5 | 10.5 | 5.2 | 5.0 | 3.9 |
| 50 | 2 | 3.7 | 17.6 | 9.9 | 8.2 | 17.6 | 5.1 | 4.2 | 4.9 | 3.8 |
| 200 | 0.5 | 3.8 | 16.8 | 9.6 | 8.0 | 16.8 | 4.8 | 4.1 | 5.0 | 3.9 |
| 200 | 2 | 3.8 | 7.3 | 5.4 | 5.2 | 7.3 | 4.0 | 3.9 | 4.5 | 3.8 |

## 4.2. Support point comparison

It was also of interest to compare the average number of support points selected by the designs discussed in Section 3. For each of the 8 simulation scenarios in Section 4.1, the average number of support points was computed for the local, BAG (with $\gamma = 0, 0.01, 0.02$), cluster, and maximin designs. The average number of support points used for each method (over all simulated pilot datasets) are shown in Tables 3 and 4, for the binomial and Poisson responses, respectively, when $\boldsymbol{\mu} = \boldsymbol{\mu}_1$.

The cluster design used a target of $k = 5$ support points. However, in some cases, the number of distinct support points selected among the $B$ bootstrap resamples was actually less than 5. In this case, all of these points were selected by the cluster design, and the number of support points was less than 5. This explains why in many rows, the average number of support points for the cluster design is less than 5; this is especially true in the Poisson case.

Table 4: Average Support Size Comparison, Poisson Response, $\boldsymbol{\mu} = \boldsymbol{\mu}_1$.

| $n$ | $\sigma$ | Local | BAG (mean) | | | BAG (median) | | | Cluster | Maximin |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | $\gamma = 0$ | 0.01 | 0.02 | $\gamma = 0$ | 0.01 | 0.02 | | |
| 50 | 0.5 | 3.3 | 9.2 | 5.8 | 4.5 | 9.2 | 4.4 | 4.0 | 4.8 | 3.4 |
| 50 | 2 | 3.2 | 4.1 | 3.5 | 3.4 | 4.1 | 3.3 | 3.3 | 3.8 | 3.3 |
| 200 | 0.5 | 2.7 | 5.4 | 4.4 | 4.0 | 5.4 | 3.6 | 3.5 | 4.3 | 3.4 |
| 200 | 2 | 3.2 | 3.6 | 3.4 | 3.3 | 3.6 | 3.2 | 3.2 | 3.5 | 3.3 |

In Tables 3 and 4, the number of support points used in the BAG design decreases as $\gamma$ increases. This behavior is expected, as it is a direct consequence of Equation (11); larger values of $\gamma$ mean that more support points have the potential to be filtered out of the final design. In general, when $\gamma = 0.02$, the average number of support points used is slightly larger than the average number of support points used by the locally optimal design. The average number of support points used for the BAG designs in the binomial case is greater than that of the Poisson case; this is especially true when $\gamma = 0$. These tables also show that when $\gamma = 0.01$ and $\gamma = 0.02$, the BAG designs found using geometric median aggregation had fewer support points on average when compared to those found using mean aggregation. Tables 5 and 6 below show similar results for the case when $\boldsymbol{\mu} = \boldsymbol{\mu}_2$.

As an example, Figure 6 illustrates the average weights of candidate points (across all 25 simulated datasets and all 50 bootstrap resamples) that were selected by the BAG design in the case of a Poisson response for when $n = 50$ and $\sigma = 2$. Points

Table 5: Average Support Size Comparison, Binomial Response, $\boldsymbol{\mu} = \boldsymbol{\mu}_2$.

| $n$ | $\sigma$ | Local | BAG (mean) | | | BAG (median) | | | Cluster | Maximin |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | $\gamma = 0$ | 0.01 | 0.02 | $\gamma = 0$ | 0.01 | 0.02 | | |
| 50 | 0.5 | 3.3 | 37.3 | 20.2 | 16.5 | 37.3 | 8.8 | 4.9 | 5.0 | 3.9 |
| 50 | 2 | 3.5 | 23.9 | 13.4 | 10.6 | 23.9 | 6.1 | 4.3 | 5.0 | 3.7 |
| 200 | 0.5 | 3.6 | 20.6 | 12.0 | 10.4 | 20.6 | 5.3 | 4.2 | 5.0 | 3.8 |
| 200 | 2 | 3.6 | 9.6 | 7.0 | 6.4 | 9.6 | 4.2 | 4.0 | 4.6 | 3.9 |

Table 6: Average Support Size Comparison, Poisson Response, $\boldsymbol{\mu} = \boldsymbol{\mu}_2$.

| $n$ | $\sigma$ | Local | BAG (mean) | | | BAG (median) | | | Cluster | Maximin |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | $\gamma = 0$ | 0.01 | 0.02 | $\gamma = 0$ | 0.01 | 0.02 | | |
| 50 | 0.5 | 3.4 | 8.8 | 6.5 | 5.3 | 8.8 | 5.2 | 4.6 | 4.7 | 3.5 |
| 50 | 2 | 3.3 | 3.9 | 3.5 | 3.5 | 3.9 | 3.2 | 3.2 | 3.8 | 3.5 |
| 200 | 0.5 | 3.5 | 6.5 | 4.7 | 4.2 | 6.5 | 4.1 | 3.8 | 4.5 | 3.4 |
| 200 | 2 | 3.3 | 3.7 | 3.5 | 3.4 | 3.7 | 3.2 | 3.2 | 3.7 | 3.5 |

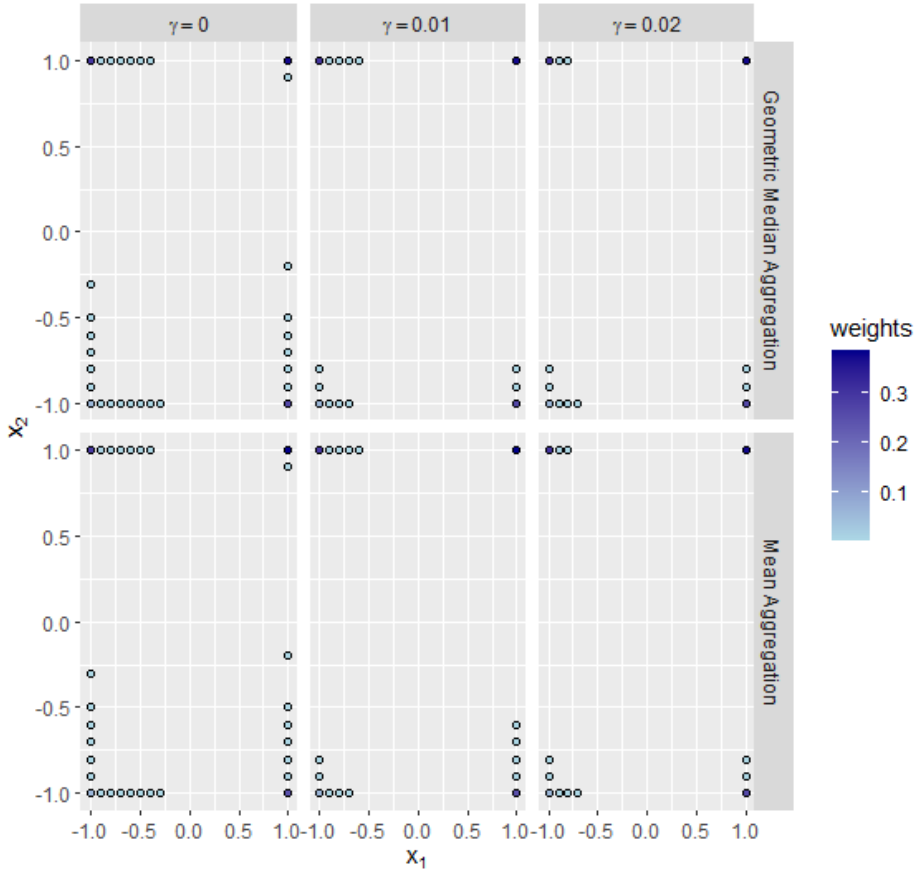


Figure 6: Support Points Selected by BAG Design for Poisson Response, $n = 50, \sigma = 2$.

that had an average weight of 0 were excluded. In all cases, most of the weight is concentrated on the extreme corner points $(-1,-1),(-1,1),(1,-1)$, and $(1,1)$. As $\gamma$ increases, the number of support points decreases. In this case, when $\gamma = 0.01$, the

geometric median aggregation method resulted in roughly two fewer support points than mean aggregation.

# 5. Conclusion

This paper proposed a novel algorithm for constructing experimental designs for generalized linear models. The proposed designs were shown to be robust for different parameters $\boldsymbol{\beta}$ for the linear predictor in a GLM. The empirical evidence showed that BAG designs had high $D-$efficiencies and are robust to parameter misspecification. These designs are competitive with existing approaches for robust designs, such as cluster or maximin designs. Moreover, the evidence showed that it is possible to reduce the number of support points in the BAG design without significantly lowering the relative efficiency of the BAG designs.

There are several open questions for future work in the development of BAG designs. It would be interesting to develop more theoretical results about the robustness and efficiency of BAG designs. Theorem 1 shows that as the pilot sample size increases to infinity, the BAG designs behave like locally optimal designs. In practice, the size of the pilot data might be small. Therefore, it would be useful to bound the log $D-$efficiency of the BAG designs for fixed pilot sample sizes when the parameter vector $\boldsymbol{\beta}$ is misspecified. Furthermore, it would be helpful to derive optimal values for the tolerance parameter $\gamma$ under various parameter misspecification scenarios. This paper focused entirely on the $D-$optimality criteria; future work could investigate the performance of the BAG design under different criteria, such as the $A-$ or $I-$ optimality criterion. Finally, although this paper considered both Poisson and Binomial data, there are other generalized linear models that should be investigated. For example, it would be useful to extend these results to GLMs with overdispersion parameters.

# Computational Details

The results in this paper were obtained using R 4.3.1. R itself and all packages used are available from the Comprehensive R Archive Network (CRAN) at https://CRAN.R-project.org/. All R code and simulation results can be found on the following public Github repository: https://github.com/nrios4/BAGdesign/.

# Acknowledgments

The work by John Stufken was supported by NSF grant DMS-2304767.

# References

Chaloner, K. and Larntz, K. (1989). Optimal Bayesian design applied to logistic regression experiments. *Journal of Statistical Planning and Inference*, 21(2):191–208, DOI: 10.1016/0378-3758(89)90004-9.

Chipman, H. A. and Welch, W. J. (1996). D-optimal design for generalized linear models. *Unpublished*.

Denman, N. G., McGree, J. M., Eccleston, J. A., and Duffull, S. B. (2011). Design of experiments for bivariate binary responses modelled by Copula functions. *Computational Statistics & Data Analysis*, 55(4):1509–1520, DOI: 10.1016/j.csda.2010.07.025.

Dror, H. A. and Steinberg, D. M. (2006). Robust experimental design for multivariate generalized linear models. *Technometrics*, 48(4):520–529, DOI: 10.1198/004017006000000318.

Fedorov, V. V. and Hackl, P. (1997). *Model-Oriented Design of Experiments*. Springer Science & Business Media. DOI: 10.1007/978-1-4612-0703-0.

Goos, P., Jones, B., and Syafitri, U. (2016). I-optimal design of mixture experiments. *Journal of the American Statistical Association*, 111(514):899–911, DOI: 10.1080/01621459.2015.1136632.

King, J. and Wong, W.-K. (2000). Minimax D-optimal designs for the logistic model. *Biometrics*, 56(4):1263–1267, DOI: 10.1111/j.0006-341X.2000.01263.x.

Li, Y. and Deng, X. (2021). An efficient algorithm for Elastic I-optimal design of generalized linear models. *Canadian Journal of Statistics*, 49(2):438–470, DOI: 10.1002/cjs.11571.

Lukemire, J., Mandal, A., and Wong, W. K. (2019). *d*-QPSO: A quantum-behaved particle swarm technique for finding D-optimal designs with discrete and continuous factors and a binary response. *Technometrics*, 61(1):77–87, DOI: 10.1080/00401706.2018.1439405.

Mathew, T. and Sinha, B. K. (2001). Optimal designs for binary data under logistic regression. *Journal of Statistical Planning and Inference*, 93(1-2):295–307, DOI: 10.1016/s0378-3758(00)00173-7.

Oris, J. T. and Bailer, A. J. (1993). Statistical analysis of the ceriodaphnia toxicity test: Sample size determination for reproductive effects. *Environmental Toxicology and Chemistry: An International Journal*, 12(1):85–90, DOI: 10.1002/etc.5620120110.

Ouyang, L., Apley, D. W., and Mehrotra, S. (2016). A design of experiments approach to validation sampling for logistic regression modeling with error-prone medical records. *Journal of the American Medical Informatics Association*, 23(e1):e71–e78, DOI: 10.1093/jamia/ocv132.

Rha, H., Kao, M.-H., and Pan, R. (2021). Bagging-enhanced sampling schedule for functional quadratic regression. *Journal of Statistical Theory and Practice*, 15:1–20, DOI: 10.1007/s42519-021-00223-x.

Russell, K. G., Woods, D. C., Lewis, S., and Eccleston, J. (2009). D-optimal designs for poisson regression models. *Statistica Sinica*, 19:721–730. https://www3.stat.sinica.edu.tw/sstest/j19n2/j19n217/j19n217.html.

Salem, A. M., Rekab, K., and Whittaker, J. A. (2004). Prediction of software failures through logistic regression. *Information and Software Technology*, 46(12):781–789, DOI: 10.1016/j.infsof.2003.10.008.

Sitter, R. R. (1992). Robust designs for binary data. *Biometrics*, 48:1145–1155, DOI: 10.2307/2532705.

Stufken, J. and Yang, M. (2012). Optimal designs for generalized linear models. *Design and Analysis of Experiments*, 3:137–164, DOI: 10.1002/9781118147634.ch4.

Van Mullekom, J. and Myers, R. (2001). Optimal experimental designs for Poisson impaired reproduction. Technical Report.

Wang, Y., Myers, R. H., Smith, E. P., and Ye, K. (2006a). D-optimal designs for Poisson regression models. *Journal of Statistical Planning and Inference*, 136(8):2831–2845, DOI: 10.1016/j.jspi.2004.10.017.

Wang, Y., Smith, E. P., and Ye, K. (2006b). Sequential designs for a Poisson regression model. *Journal of Statistical Planning and Inference*, 136(9):3187–3202, DOI: 10.1016/j.jspi.2004.12.007.

Wang, Z. and Stufken, J. (2022). Locally D-optimal designs for binary responses and multiple continuous design variables. *Journal of Quantitative Economics*, 20(Suppl 1):101–113, DOI: 10.1007/s40953-022-00304-z.

Welch, W. J. (1983). A mean squared error criterion for the design of experiments. *Biometrika*, 70(1):205–213, DOI: 10.2307/2335958.

Woods, D. C., Lewis, S. M., Eccleston, J. A., and Russell, K. (2006). Designs for generalized linear models with several variables and model uncertainty. *Technometrics*, 48(2):284–292, DOI: 10.1198/004017005000000571.

Yang, M. (2008). *A*-optimal designs for generalized linear models with two parameters. *Journal of Statistical Planning and Inference*, 138(3):624–641, DOI: 10.1016/j.jspi.2006.12.007.

Yang, M., Biedermann, S., and Tang, E. (2013). On optimal designs for nonlinear models: A general and efficient algorithm. *Journal of the American Statistical Association*, 108(504):1411–1420, DOI: 10.1080/01621459.2013.806268.

Yang, M., Zhang, B., and Huang, S. (2011). Optimal designs for generalized linear models with multiple design variables. *Statistica Sinica*, 21:1415–1430, DOI: 10.5705/ss.2009.115.

Yu, Y. (2011). D-optimal designs via a cocktail algorithm. *Statistics and Computing*, 21(4):475–481, DOI: 10.1007/s11222-010-9183-2.

# A. Proofs

**Proof of Theorem 1**. Let $n$ and $N$ denote the number of data points in the pilot sample and the number of candidate design points, respectively. For each $i = 1, \ldots, B$, $(\mathbf{X}^{(i)}, Y^{(i)})$ denote the $i^{th}$ bootstrap sample of the pilot data $(\mathbf{X}, Y)$, let $\boldsymbol{\beta}$ be the true vector of regression parameters in Model (1), and let $\hat{\boldsymbol{\beta}}^{(i)}$ be the MLE for $\boldsymbol{\beta}$ in the $i^{th}$ bootstrap sample. Let $\mathbf{z}_j$ be the $j^{th}$ candidate design point, where $j \in \{1, \ldots, N\}$. By Theorem 3 of Yang et al. (2013), it follows that $\mathbf{w}^{(i)} = \arg\max_{\mathbf{w} \in \mathcal{S}^N} \log(|I(\mathbf{w}, \hat{\boldsymbol{\beta}}^{(i)})|)$ for each $i = 1, \ldots, B$; i.e., each $\mathbf{w}^{(i)}$ is a locally $D-$optimal design for $\hat{\boldsymbol{\beta}}^{(i)}$. Let $\mathbf{w}_{BAG} = (1/B)\sum_{i=1}^{n} \mathbf{w}^{(i)}$, and let $\mathbf{w}^* = \arg\max_{\mathbf{w} \in \mathcal{S}^N} \log(|\mathbf{Z}^T\mathbf{V}(\boldsymbol{\beta})\text{diag}(\mathbf{w})\mathbf{Z}|)$. Let $w_{BAG,j}$, $w_j^{(i)}$, and $w_j^*$ denote the $j^{th}$ elements of $\mathbf{w}_{BAG}$, $\mathbf{w}^{(i)}$, and $\mathbf{w}^*$, respectively, for $j = 1, \ldots, N$. Notice that by definition of $\mathbf{w}^*$,

$$\log(|I(\mathbf{w}_{BAG}, \boldsymbol{\beta})|) = \log\left(\left|\sum_{j=1}^{N} \mathbf{V}_{jj}(\boldsymbol{\beta}) w_{BAG,j} \mathbf{z}_j \mathbf{z}_j^T\right|\right) \tag{14}$$

$$\leq \log\left(\left|\sum_{j=1}^{N} \mathbf{V}_{jj}(\boldsymbol{\beta}) w_j^* \mathbf{z}_j \mathbf{z}_j^T\right|\right) = \log(|I(\mathbf{w}^*, \boldsymbol{\beta})|). \tag{15}$$

Also notice that by concavity of the log determinant,

$$\log\left(\left|\sum_{j=1}^{N} \mathbf{V}_{jj}(\boldsymbol{\beta}) w_{BAG,j} \mathbf{z}_j \mathbf{z}_j^T\right|\right) = \log\left(\left|\frac{1}{B}\sum_{i=1}^{B}\sum_{j=1}^{N} \mathbf{V}_{jj}(\boldsymbol{\beta}) w_j^{(i)} \mathbf{z}_j \mathbf{z}_j^T\right|\right) \tag{16}$$

$$\geq \frac{1}{B}\sum_{i=1}^{B} \log\left(\left|\sum_{j=1}^{N} \mathbf{V}_{jj}(\boldsymbol{\beta}) w_j^{(i)} \mathbf{z}_j \mathbf{z}_j^T\right|\right). \tag{17}$$

Combining (15) and (17) and taking the limit in probability as $n \to \infty$ (with respect to the bootstrap resamples) yields

$$\operatorname*{plim}_{n\to\infty} \frac{1}{B}\sum_{i=1}^{B} \log(|I(\mathbf{w}^{(i)}, \boldsymbol{\beta})|) \leq \operatorname*{plim}_{n\to\infty} \log(|I(\mathbf{w}_{BAG}, \boldsymbol{\beta})|) \leq \operatorname*{plim}_{n\to\infty} \log(|I(\mathbf{w}^*, \boldsymbol{\beta})|) \tag{18}$$

$$\frac{1}{B}\sum_{i=1}^{B} \log\left(\left|I\left(\operatorname*{plim}_{n\to\infty} \mathbf{w}^{(i)}, \boldsymbol{\beta}\right)\right|\right) \leq \operatorname*{plim}_{n\to\infty} \log(|I(\mathbf{w}_{BAG}, \boldsymbol{\beta})|) \leq \log(|I(\mathbf{w}^*, \boldsymbol{\beta})|) \tag{19}$$

The limit in probability is moved inside of the log determinant in (19) by continuity of the log determinant. If $\text{plim}_{n\to\infty}\mathbf{w}^{(i)} = \mathbf{w}^*$ for every $i = 1, \ldots, B$, then the left hand side of (19) reduces to $\log(|I(\mathbf{w}^*, \boldsymbol{\beta})|)$. From the squeeze theorem, this would imply that $\text{plim}_{n\to\infty} \log(|I(\mathbf{w}_{BAG}, \boldsymbol{\beta})|) = \log(|I(\mathbf{w}^*, \boldsymbol{\beta})|)$, which would conclude the proof.

It suffices to prove that $\mathbf{w}^{(i)} \underset{p}{\to} \mathbf{w}^*$ for each $i = 1, \ldots, B$. For each $i = 1, \ldots, B$, $\hat{\boldsymbol{\beta}}^{(i)}$ is the MLE of $\boldsymbol{\beta}$, which is a consistent estimator of $\boldsymbol{\beta}$. Therefore, as $n \to \infty$ and for fixed $B$, it follows that $\forall i = 1, \ldots, B$, $\hat{\boldsymbol{\beta}}^{(i)} \underset{p}{\to} \boldsymbol{\beta}$. By continuity, it follows that as $n \to \infty$, $\mathbf{z}_j^T\hat{\boldsymbol{\beta}}^{(i)} \underset{p}{\to} \mathbf{z}_j^T\boldsymbol{\beta}$ for all $i = 1, \ldots, B$ and for all $j = 1, \ldots, N$. Let

$$\mathbf{V}_{jj}^{(i)} = \begin{cases} \dfrac{\exp(\mathbf{z}_j^T \hat{\boldsymbol{\beta}}^{(i)})}{(1+\exp(\mathbf{z}_j^T \hat{\boldsymbol{\beta}}^{(i)}))^2} & \text{for a Binomial response, and} \\ \exp(\mathbf{z}_j^T \hat{\boldsymbol{\beta}}^{(i)}) & \text{for a Poisson response,} \end{cases} \tag{20}$$

and let $\mathbf{V}_{jj}$ be similarly defined (i.e., replace $\hat{\boldsymbol{\beta}}^{(i)}$ with $\boldsymbol{\beta}$). In either case, $\mathbf{V}_{jj}^{(i)}$ is a continuous function of $\mathbf{z}_j^T \hat{\boldsymbol{\beta}}^{(i)}$, so it follows that as $n \to \infty$, $\mathbf{V}_{jj}^{(i)} \underset{p}{\to} \mathbf{V}_{jj}$ for all $i = 1, \ldots, B$ and for all $j = 1, \ldots, N$. Therefore, by continuity,

$$\mathbf{V}_{jj}^{(i)} \mathbf{z}_j^T I(\mathbf{w}^{(i)}, \hat{\boldsymbol{\beta}}^{(i)})^{(-1)} \mathbf{z}_j \to_p \mathbf{V}_{jj} \mathbf{z}_j^T I\left(\operatorname*{plim}_{n\to\infty} \mathbf{w}^{(i)}, \boldsymbol{\beta}\right)^{(-1)} \mathbf{z}_j. \tag{21}$$

Recall that $\mathbf{w}^{(i)}$ is a locally $D-$optimal design for $\hat{\boldsymbol{\beta}}^{(i)}$. By Theorem 4.1 of Stufken and Yang (2012), it follows that for all $i = 1, \ldots, B$ and $j = 1, \ldots, N$,

$$\mathbf{V}_{jj}^{(i)} \mathbf{z}_j^T I(\mathbf{w}^{(i)}, \hat{\boldsymbol{\beta}}^{(i)})^{(-1)} \mathbf{z}_j - p \leq 0 \tag{22}$$

$$\implies \operatorname*{plim}_{n\to\infty} \mathbf{V}_{jj}^{(i)} \mathbf{z}_j^T I(\mathbf{w}^{(i)}, \hat{\boldsymbol{\beta}}^{(i)})^{(-1)} \mathbf{z}_j - p \leq 0 \tag{23}$$

$$\implies \mathbf{V}_{jj} \mathbf{z}_j^T I\left(\operatorname*{plim}_{n\to\infty} \mathbf{w}^{(i)}, \boldsymbol{\beta}\right)^{(-1)} \mathbf{z}_j - p \leq 0. \tag{24}$$

By (24), it follows that $\operatorname{plim}_{n\to\infty} \mathbf{w}^{(i)} = \arg\max_{\mathbf{w}\in\mathcal{S}^N} \log(|I(\mathbf{w}, \boldsymbol{\beta})|) = \mathbf{w}^*$ for all $i = 1, \ldots, B$. This completes the proof.

□

# B. Additional Simulation Results

In this Appendix, additional simulation results are provided for the case when $\tilde{\sigma}^2 = 1$. The simulation scenarios are otherwise identical to those described in Section 4.1.

Figures 7 and 8 show the average relative $D-$efficiencies of the BAG, cluster, and maximin designs to the local designs for the case when $\boldsymbol{\mu} = \boldsymbol{\mu}_1$ and $\tilde{\sigma}^2 = 1$.

For the Binomial response, when $\sigma = 0.5$, the BAG designs with mean aggregation have higher median relative $D-$efficiencies than all other designs. When $n = 50$, increasing the tolerance parameter $\gamma$ slightly decreases the median relative $D-$efficiency, but this effect is negligible when $n = 200$. When $n = 50, \sigma = 2$, the cluster designs have the highest median relative $D-$efficiency for both the Binomial and Poisson responses.

Figures 9 and 10 show the average relative $D-$efficiencies of the BAG, cluster, and maximin designs to the local designs for the case when $\boldsymbol{\mu} = \boldsymbol{\mu}_2$ and $\tilde{\sigma}^2 = 1$. For the Poisson response, the BAG designs have higher median relative $D-$efficiency to the local design when $n = 50$. When $n = 200$ and $\sigma = 0.5$, the maximin designs perform slightly better than the BAG designs, but when $n = 200$ and $\sigma = 2$, all designs have relatively similar performance. For the Binomial response, when $\sigma = 0.5$, the BAG designs with geometric median aggregation and $\gamma = 0$ (untrimmed) have higher median relative $D-$efficiencies than all other designs. When $n = 50, \sigma = 2$, this is also

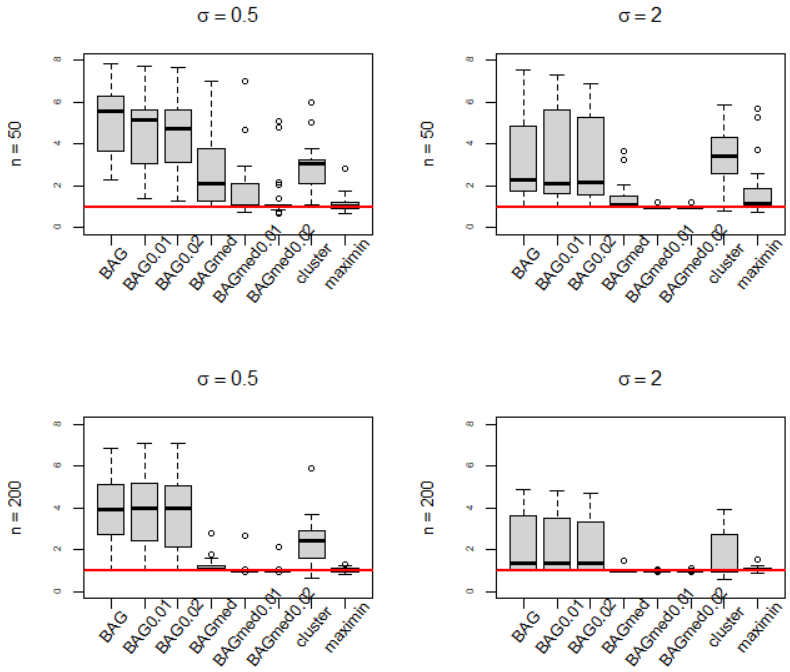


Figure 7: Relative $D-$efficiencies to Local Design, Binomial Response, $\boldsymbol{\mu}_1$, $\tilde{\sigma}^2 = 1$.

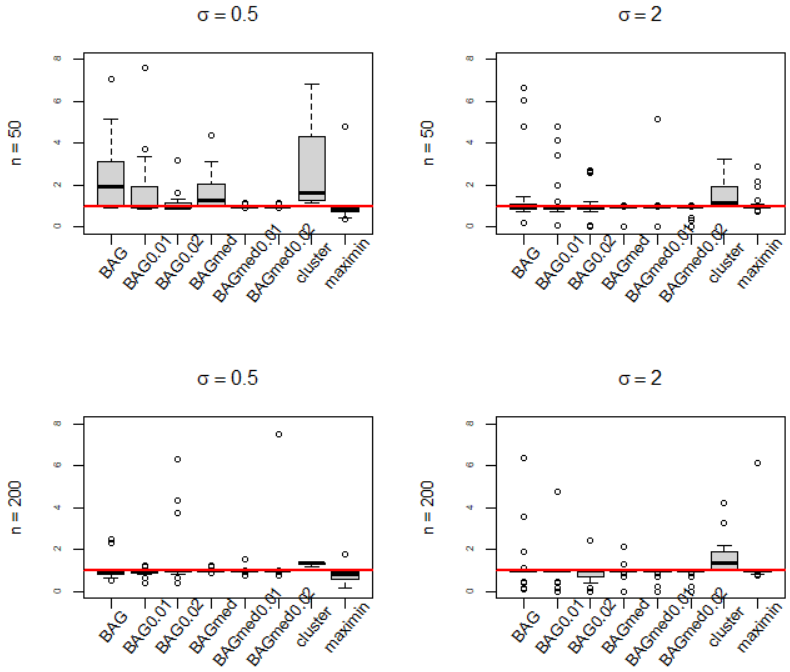


Figure 8: Relative $D-$efficiencies to Local Design, Poisson Response, $\boldsymbol{\mu}_1$, $\tilde{\sigma}^2 = 1$.

the case. When $n = 200, \sigma = 2$, all of the designs have similar performance apart from the cluster and maximin designs, which have lower median relative $D-$efficiency than all other considered designs.

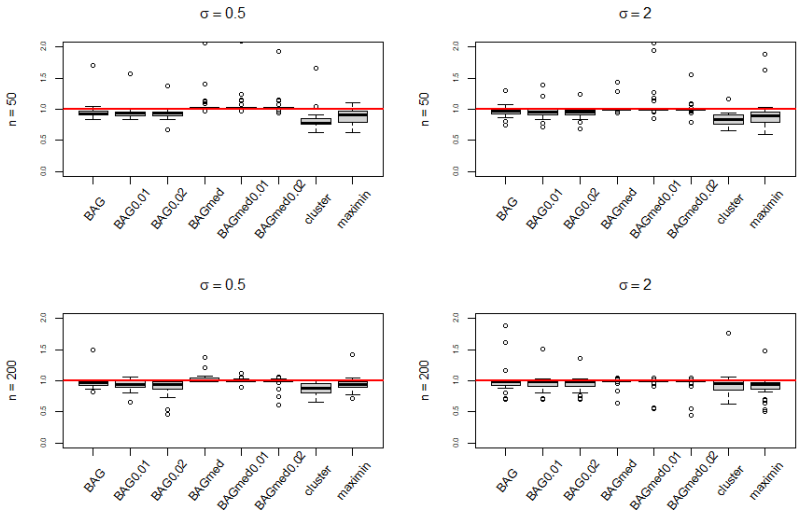


Figure 9: Relative $D-$efficiencies to Local Design, Binomial Response, $\boldsymbol{\mu}_2$, $\tilde{\sigma}^2 = 1$.

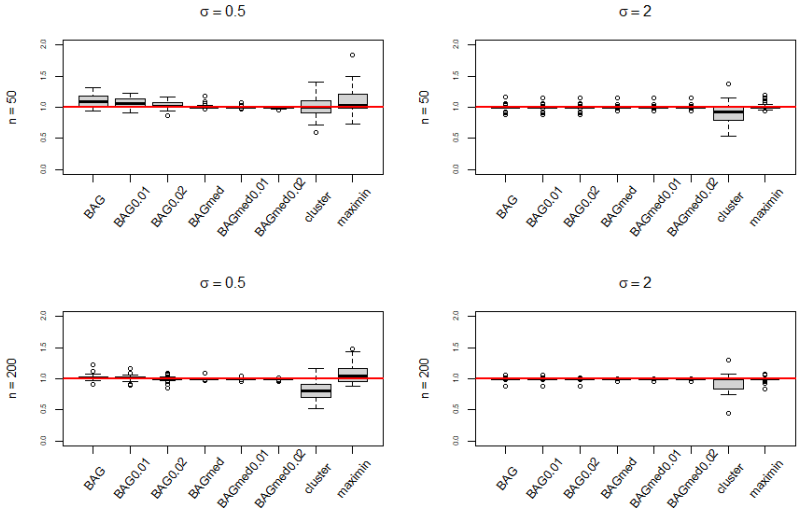


Figure 10: Relative $D-$efficiencies to Local Design, Poisson Response, $\boldsymbol{\mu}_2$, $\tilde{\sigma}^2 = 1$.

Overall, in all cases it is observed that the BAG design is a promising alternative for finding designs that are robust to parameter misspecification. For all considered scenarios with a Binomial response, one of the BAG designs was preferred to the cluster or maximin designs. Specifically, when $\boldsymbol{\mu} = \boldsymbol{\mu}_1$, mean aggregation had the best results. In this case, the test parameter vectors are centered on a point that is far away from the true $\boldsymbol{\beta}$. When $\boldsymbol{\mu} = \boldsymbol{\mu}_2$, which was a more mild misspecification scenario, geometric median aggregation was generally preferred. For the Poisson response, the cluster design was the most promising in the extreme case ($\boldsymbol{\mu} = \boldsymbol{\mu}_1$), but the cluster designs performed worse than BAG designs in the more mild misspecification scenario ($\boldsymbol{\mu} = \boldsymbol{\mu}_2$). Also, it was noted that in all scenarios, increasing $\gamma$ reduces the relative $D-$efficiency of the BAG designs to some degree for both mean aggregation and geometric median aggregation of the weights.

**Affiliation:**

Nicholas Rios
Department of Statistics
George Mason University
Nguyen Engineering Building, Fairfax, Virginia 22030
Suite 1705
E-mail: nrios4@gmu.edu
URL: https://statistics.gmu.edu/profiles/nrios4

John Stufken
Department of Statistics
George Mason University
Nguyen Engineering Building, Fairfax, Virginia 22030
Suite 1705
E-mail: jstufken@gmu.edu
URL: https://statistics.gmu.edu/profiles/jstufken

*Journal of Data Science, Statistics, and Visualisation* https://jdssv.org/
published by the International Association for Statistical Computing
http://iasc-isi.org/

ISSN 2773-0689
January 2025, Volume V, Issue I *Submitted:* 2024-10-23
doi:10.52933/jdssv.v5i1.123 *Accepted:* 2025-01-02