Reduction of Small-Sample Bias of GLM Parameter Estimates

Annie Yao (ID: 301327984)
Ravleen Bajaj (ID: 301617069)
Samir Arora (ID: 301355302)

April 22nd, 2025

1 Introduction

Maximum Likelihood Estimation is the predominant method for estimating parameters in Generalized Linear Models (GLMs) and in general. One key reason for its widespread use is that, under *usual regularity conditions*, the Maximum Likelihood Estimator (MLE) enjoys several optimal asymptotic properties. These regularity conditions ensure that the statistical model behaves well as the sample size increases. Informally, these conditions include:

1. The parametric model $f(y;\boldsymbol{\theta})$ is identifiable, meaning $f(y;\boldsymbol{\theta}) \neq f(y;\boldsymbol{\theta'})$ for any $\boldsymbol{\theta} \neq \boldsymbol{\theta'}$, except from sets of probability zero.

2. The parameter $\boldsymbol{\theta} \in \mathbb{R}^p$, where p is fixed and finite.

3. The parameter space $\boldsymbol{\Theta}$ does not depend on sample size n.

4. A sufficient number of derivatives of the log-likelihood function and their expectations exist under the model $f(y;\boldsymbol{\theta})$.

5. The true parameter value $\boldsymbol{\theta}$ lies in the interior of the parameter space $\boldsymbol{\Theta} \subset \mathbb{R}^p$.

Under these conditions, the MLE $\hat{\boldsymbol{\theta}}$ possesses the following desirable properties as $\boldsymbol{n \to \infty}$:

1. **Consistency**: $\hat{\boldsymbol{\theta}}$ converges in probability to the true parameter $\boldsymbol{\theta}$.

2. **Asymptotic unbiasedness**: The bias of $\hat{\boldsymbol{\theta}}$ is of order $\mathcal{O}(n^{-1})$, vanishing asymptotically.

3. **Asymptotic normality**: The distribution of $\hat{\boldsymbol{\theta}}$ converges to a multivariate Normal distribution with mean $\boldsymbol{\theta}$ and variance-covariance matrix $F(\boldsymbol{\theta})^{-1}$, where $F(\boldsymbol{\theta})$ is the Fisher Information matrix.

While MLEs are often favored due to their appealing theoretical properties, it is important to remember that these properties rely not only on the aforementioned regularity conditions, but also on the assumption of large sample sizes. Since these are asymptotic results, their performance can deteriorate in finite samples, and particularly in small-sample settings, leading to biased estimates and unreliable inference.

This report focuses on the issue of small-sample bias in MLEs for GLM coefficients. We explore both the theoretical insights and practical implementations in R for commonly-used bias-reduction techniques. Section 2 introduces asymptotic bias correction, a straightforward approach that adjusts MLE based on the estimated asymptotic bias. Section 3 presents Firth's method, which modifies the score function to reduce bias while preserving invariance properties of MLEs. Section 4 builds on this by introducing the log-F prior method, a Bayesian-inspired technique that simplifies Firth's adjustment and allows the incorporation of prior information. Finally, Section 5 provides a simulation study comparing the performance of all three approaches under various small-sample settings, followed by concluding remarks.

2 Asymptotic Bias Correction

Consider an arbitrary estimator $\hat{\boldsymbol{\theta}}$, not necessarily based on maximum likelihood, taking values in a parameter space $\boldsymbol{\Theta} \subset \mathbb{R}^p$. Then, bias reduction can be framed as the problem of constructing a new estimator $\tilde{\boldsymbol{\theta}}$ that satisfies

$$\hat{\boldsymbol{\theta}} - \tilde{\boldsymbol{\theta}} = B(\boldsymbol{\theta}), \tag{1}$$

where $B(\boldsymbol{\theta}) = E_{\boldsymbol{\theta}}[\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}]$ is the bias of estimator $\hat{\boldsymbol{\theta}}$. If both $B(\boldsymbol{\theta})$ and $\boldsymbol{\theta}$ were known, one could directly compute the bias-corrected estimator $\tilde{\boldsymbol{\theta}}$. The formalization in Equation (1) may seems trivial, but Kosmidis (2014) argue that all known methods to reduce bias can be though of as attempts to approximate its solution [7]. In addition, he stated that a bias-corrected estimator $\tilde{\boldsymbol{\theta}}$ derived in this way can offer the following desirable properties:

1. **Unbiasedness**: $\tilde{\boldsymbol{\theta}}$ has zero bias by construction and therefore, a smaller mean squared error compared to the original estimator $\hat{\boldsymbol{\theta}}$.

2. **Consistency**: If the $\hat{\boldsymbol{\theta}}$ has a variance-covariance matrix that vanishes as $n \to \infty$, then $\tilde{\boldsymbol{\theta}}$ is consistent, even if $\hat{\boldsymbol{\theta}}$ is not.

However, since the true parameter value $\boldsymbol{\theta}$ is unknown and the bias function $B(\boldsymbol{\theta})$ typically lacks a closed-form expression, direct bias-correction as in Equation (1) is generally infeasible. For many common estimators, including the MLE, the bias function can be expanded in decreasing powers of n, as shown below:

$$B(\boldsymbol{\theta}) = \frac{b_1(\boldsymbol{\theta})}{n} + \frac{b_2(\boldsymbol{\theta})}{n^2} + \frac{b_3(\boldsymbol{\theta})}{n^3} + \mathcal{O}(n^{-4}). \tag{2}$$

This expansion motivates the method of asymptotic bias correction, which seeks to reduce bias by approximating $B(\boldsymbol{\theta})$ by $b_1(\hat{\boldsymbol{\theta}})/n$, which is the first-term in the right hand side of the Equation (2) evaluated at the original estimator $\hat{\boldsymbol{\theta}}$. Therefore, the new bias-adjusted estimator under this method is:

$$\tilde{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}} - b_1(\hat{\boldsymbol{\theta}})/n. \tag{3}$$

For MLE, Cox and Snell (1968) derived a closed-form expression for $b_1(\boldsymbol{\theta})/n$ [2], given by:

$$E[\hat{\boldsymbol{\theta}}_s - \boldsymbol{\theta}_s] = \frac{1}{2}\boldsymbol{F}^{rs}\boldsymbol{F}^{tu}(K_{rtu} + 2K_{t,ru}) + \mathcal{O}(n^{-2}), \tag{4}$$

where $\boldsymbol{F}^{rs}$ is the (r, s) entry in inverse of the Fisher information matrix, $K_{rtu} = E\left[\frac{\partial^3 \log p(\boldsymbol{Y}|\boldsymbol{\theta})}{\partial\theta_r \partial\theta_t \partial\theta_u}\right]$ and $K_{t,ru} = E\left[\sum_j \frac{\partial}{\partial\boldsymbol{\theta}_t} \log p(Y_j;\boldsymbol{\theta}) \frac{\partial^2}{\partial\boldsymbol{\theta}_r \partial\boldsymbol{\theta}_u} \log p(Y_j;\boldsymbol{\theta})\right]$. Note that Cox and Snell (1968) used a special summation convention where whenever an index appears twice in a single term, it implies a summation over that index. The idea behind Equation (4) is discussed in detail in the following subsection.

Using some identities for null cumulants of log-likelihood derivatives, Kosmidis and Firth (2010) expressed the expression for the first order bias term in a convenient matrix form,

which is given below [8]:

$$\frac{b_1(\boldsymbol{\theta})}{n} = -\{F(\boldsymbol{\theta})\}^{-1}A(\boldsymbol{\theta}), \tag{5}$$

where, $A(\boldsymbol{\theta})$ is a p-dimensional vector with components

$$A_s(\boldsymbol{\theta}) = \frac{1}{2}tr[\{F(\boldsymbol{\theta})\}^{-1}\{P_s(\boldsymbol{\theta}) + Q_s(\boldsymbol{\theta})\}],$$

$$P_s(\boldsymbol{\theta}) = E_{\boldsymbol{\theta}}[U(\boldsymbol{\theta})U^T(\boldsymbol{\theta})U_s(\boldsymbol{\theta})],$$
$$Q_s(\boldsymbol{\theta}) = -E_{\boldsymbol{\theta}}[I(\boldsymbol{\theta})U_s(\boldsymbol{\theta})], \quad (s = 1, \cdots p).$$

where, $I(\boldsymbol{\theta})$ is the observed information matrix and $U(\boldsymbol{\theta})$ is the score function.

2.1 Idea Behind the Expression for $b_1(\boldsymbol{\theta})/n$

The following argument is in line with what is presented by Cox and Snell (1968) [2]. Also, the summation convention used by Cox and Snell (1968) is used in this derivation too. Let $\boldsymbol{\theta} \in \mathbb{R}^p$ is a parameter vector. Assume $\hat{\boldsymbol{\theta}}$ is an MLE near the true value $\boldsymbol{\theta}_0$. Then, by first order Taylor expansion,

$$\boldsymbol{U}_r(\hat{\boldsymbol{\theta}}) = \boldsymbol{U}_r(\boldsymbol{\theta}_0) + (\hat{\boldsymbol{\theta}}_s - \boldsymbol{\theta}_{0s})\frac{\partial}{\partial \boldsymbol{\theta}_s}\boldsymbol{U}_r(\boldsymbol{\theta}_0),$$

where, $\boldsymbol{U}_r(\boldsymbol{\theta}) = \frac{\partial}{\partial \boldsymbol{\theta}_r}\log p(\boldsymbol{Y};\boldsymbol{\theta})$ and $\boldsymbol{Y} \in \mathbb{R}^n$ is a random vector of observations. Since the score function evaluated at MLE is 0,

$$0 = \boldsymbol{U}_r(\boldsymbol{\theta}_0) + (\hat{\boldsymbol{\theta}}_s - \boldsymbol{\theta}_{0s})\frac{\partial}{\partial \boldsymbol{\theta}_s}\boldsymbol{U}_r(\boldsymbol{\theta}_0)$$

$$-\frac{\partial}{\partial \boldsymbol{\theta}_s}\boldsymbol{U}_r(\boldsymbol{\theta_0})(\hat{\boldsymbol{\theta}}_s - \boldsymbol{\theta}_{0s}) = \boldsymbol{U}_r(\boldsymbol{\theta}_0).$$

Let $\boldsymbol{I}_{rs} = -\frac{\partial}{\partial \boldsymbol{\theta}_s}\boldsymbol{U}_r(\boldsymbol{\theta}_0)$ and $\boldsymbol{F}_{rs} = E[\boldsymbol{I}_{rs}]$. Therefore, by replacing $\boldsymbol{I}_{rs}$ by its expectation in the above equation, we get

$$\boldsymbol{F}_{rs}(\hat{\boldsymbol{\theta}}_s - \boldsymbol{\theta}_{0s}) = \boldsymbol{U}_r(\boldsymbol{\theta}_0)$$
$$\hat{\boldsymbol{\theta}}_s - \boldsymbol{\theta}_{0s} = \boldsymbol{F}^{rs}\boldsymbol{U}_r(\boldsymbol{\theta}_0), \tag{6}$$

where $\boldsymbol{F}^{rs}$ is the (r, s) entry in inverse of the Fisher information matrix. To obtain a more refined answer, let's look at second-order Taylor Expansion.

$$\boldsymbol{U}_r(\hat{\boldsymbol{\theta}}) \approx \boldsymbol{U}_r(\boldsymbol{\theta}_0) + (\hat{\boldsymbol{\theta}}_s - \boldsymbol{\theta}_{0s})\frac{\partial}{\partial \boldsymbol{\theta}_s}\boldsymbol{U}_r(\boldsymbol{\theta}_0) + \frac{1}{2}(\hat{\boldsymbol{\theta}}_t - \boldsymbol{\theta}_{0t})(\hat{\boldsymbol{\theta}}_u - \boldsymbol{\theta}_{0u})\frac{\partial^2}{\partial \boldsymbol{\theta}_t \partial \boldsymbol{\theta}_u}$$

Again, the score function evaluated at MLE is zero. Taking expectation on both sides gives

$$
\begin{aligned}
0 \approx 0 &+ E\left[\hat{\boldsymbol{\theta}}_s - \boldsymbol{\theta}_{0s}\right] E\left[\frac{\partial}{\partial \boldsymbol{\theta}_s}\boldsymbol{U}_r(\boldsymbol{\theta}_0)\right] + \operatorname{Cov}\left[\hat{\boldsymbol{\theta}}_s - \boldsymbol{\theta}_{0s}, \frac{\partial}{\partial \boldsymbol{\theta}_s}\boldsymbol{U}_r(\boldsymbol{\theta}_0)\right] \\
&+ \frac{1}{2}\Bigg(E\left[\hat{\boldsymbol{\theta}}_t - \boldsymbol{\theta}_{0t}\right] E\left[\hat{\boldsymbol{\theta}}_u - \boldsymbol{\theta}_{0u}\right] E\left[\frac{\partial^2}{\partial \boldsymbol{\theta}_t \partial \boldsymbol{\theta}_u}\boldsymbol{U}_r(\boldsymbol{\theta}_0)\right] \\
&\quad + \operatorname{Cov}\left[\hat{\boldsymbol{\theta}}_u - \boldsymbol{\theta}_{0u}, \hat{\boldsymbol{\theta}}_t - \boldsymbol{\theta}_{0t}\right] E\left[\frac{\partial^2}{\partial \boldsymbol{\theta}_t \partial \boldsymbol{\theta}_u}\boldsymbol{U}_r(\boldsymbol{\theta}_0)\right] \\
&\quad + \operatorname{Cov}\left[\left(\hat{\boldsymbol{\theta}}_t - \boldsymbol{\theta}_{0t}\right)\left(\hat{\boldsymbol{\theta}}_u - \boldsymbol{\theta}_{0u}\right), \frac{\partial^2}{\partial \boldsymbol{\theta}_t \partial \boldsymbol{\theta}_u}\boldsymbol{U}_r(\boldsymbol{\theta}_0)\right]\Bigg).
\end{aligned}
$$

Using the fact that $\operatorname{Cov}[\hat{\boldsymbol{\theta}}_r, \hat{\boldsymbol{\theta}}_s] = \boldsymbol{F}^{rs}$ and the result given in Equation (6), we can simplify the above equation as

$$
E\left[\hat{\boldsymbol{\theta}}_s - \boldsymbol{\theta}_{0s}\right]\boldsymbol{F}_{rs} \approx \frac{1}{2}\boldsymbol{F}^{tu}\left(E\left[\frac{\partial^3}{\partial \boldsymbol{\theta}_u \partial \boldsymbol{\theta}_t \partial \boldsymbol{\theta}_r}\log p(\boldsymbol{Y};\boldsymbol{\theta})\right] + 2E\left[\sum_j \frac{\partial}{\partial \boldsymbol{\theta}_t}\log p(Y_j;\boldsymbol{\theta})\frac{\partial^2}{\partial \boldsymbol{\theta}_r \partial \boldsymbol{\theta}_u}\log p(Y_j;\boldsymbol{\theta})\right]\right)
$$

$$
E\left[\hat{\boldsymbol{\theta}}_s - \boldsymbol{\theta}_{0s}\right] \approx \boldsymbol{F}^{rs}\frac{1}{2}\boldsymbol{F}^{tu}\left(E\left[\frac{\partial^3}{\partial \boldsymbol{\theta}_u \partial \boldsymbol{\theta}_t \partial \boldsymbol{\theta}_r}\log p(\boldsymbol{Y};\boldsymbol{\theta})\right] + 2E\left[\sum_j \frac{\partial}{\partial \boldsymbol{\theta}_t}\log p(Y_j;\boldsymbol{\theta})\frac{\partial^2}{\partial \boldsymbol{\theta}_r \partial \boldsymbol{\theta}_u}\log p(Y_j;\boldsymbol{\theta})\right]\right) \tag{7}
$$

2.2 Asymptotic Bias Correction in GLMs

Building on the general expression for the first-order bias term given in Equation (4), Cordeiro and McCullagh (1991) derived a specific form for the first-order bias of the coefficients of a linear predictor in the GLM setting [1]. Assuming p coefficients, the expression—written without the summation convention of Cox and Snell (1968)—is given by:

$$
E[\hat{\boldsymbol{\beta}}_a - \boldsymbol{\beta}_a] = \sum_{r,t,u \in S} \boldsymbol{F}^{ra}\boldsymbol{F}^{tu}\left(\frac{1}{2}K_{rtu} + K_{t,ru}\right) + \mathcal{O}(n^{-2}), \tag{8}
$$

where $S = \{1, ..., p, \phi\}$, and F^{ra}, K_{rtu}, and $K_{t,ru}$ are as in Equation (4).

In matrix notation, a corresponding expression for the first-order bias term is given by:

$$
b_1(\boldsymbol{\beta})/n = -(2\phi)^{-1}(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Z}_d\mathbf{F}\mathbf{1} \tag{9}
$$

where,

$$
\mathbf{Z} = \{z_{ij}\} = \mathbf{X}(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T, \quad \mathbf{Z}_d = diag\{z_{11}, \cdots, z_{nn}\},
$$
$$
F = diag\{f_{11}, \cdots f_{nn}\}, \quad f = V^{-1}(d\mu/d\eta)(d^2\mu/d\eta^2),
$$

and $\mathbf{1}$ is an $n \times 1$ vector of ones. The similarity to Equation (5) can be observed here.

2.3 Properties of Asymptotic Bias Corrected Estimators

Given the expression for the first-order bias term $b_1(\boldsymbol{\beta})/n$ in the GLM setting (see Equation (9)), a bias-corrected estimator can be defined as:

$$\tilde{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}} - \frac{b_1(\hat{\boldsymbol{\beta}})}{n}. \tag{10}$$

In addition to being straightforward to implement, this estimator enjoys two key theoretical advantages, as demonstrated by Efron (1975) [3]:

1. **Reduced Bias:** The estimator $\tilde{\boldsymbol{\beta}}$ has bias of order $o(n^{-1})$, which is asymptotically smaller than the $\mathcal{O}(n^{-1})$ bias of the MLE.
2. **Second-Order Efficiency:** Among all estimators with bias of order $\mathcal{O}(n^{-2})$, $\tilde{\boldsymbol{\beta}}$ achieves the smallest possible asymptotic variance.

Despite these appealing properties, the method has certain limitations. First, it relies on the availability of a closed-form expression for $b_1(\boldsymbol{\beta})/n$, which is not always attainable in practice. Second, bias correction generally compromises the invariance of the MLE under reparameterization. Specifically, while $\tilde{\boldsymbol{\beta}}$ may be a less biased estimator of $\boldsymbol{\beta}$, the transformed estimator $g(\tilde{\boldsymbol{\beta}})$ is not guaranteed to be a less biased estimator of $g(\boldsymbol{\beta})$, where g is some one-to-one function. In fact, the bias of $g(\tilde{\boldsymbol{\beta}})$ as an estimator of $g(\boldsymbol{\beta})$ may not only fail to improve but can actually become substantially worse [7]. Finally, the corrected estimator inherits any instabilities present in the original MLE. For example, if the MLE is undefined, asymptotic bias correction cannot be applied.

2.4 Usage in R

Asymptotic bias correction in GLMs can be conveniently implemented in `R` using the `brglm2` package. Once the package is installed and loaded, users can fit a GLM using the familiar `glm()` function, with two additional arguments: `method` and `type`. The argument `method = brglm_fit` specifies the use of the bias reduction method and `type = "correction"` requests asymptotic bias correction.

The following example demonstrates how to fit a binomial GLM with asymptotic bias correction:

```r
# Installing the required libraries
install.packages("brglm2")

# Loading the required packages
library(brglm2)

# Fitting the asymptotic bias corrected binomial glm
fit_binomial_bias_corrected <- glm(cbind(y, n-y) ~ Grade + Sex + Participate,
                                   binomial(logit), data = school_binomial,
                                   method = brglm_fit, type = "correction")
```

```
# Getting the summary output
summary(fit_binomial_bias_corrected)
```

The summary output after applying asymptotic bias correction is returned in the same format as for standard GLMs. The parameter estimates are shifted according to Equation (10). Importantly, the standard errors correspond to the square roots of the diagonal elements of the inverse Fisher information matrix, evaluated at the bias-corrected estimates. This follows the approach described by Cox and Snell (1968), as implemented in the `coxsnell.bc()` function from the `mle.tools` package [9].

3 Bias-Reduced Maximum Likelihood Estimation (Firth, 1993)

3.1 Motivation

In a regular model with a p-dimensional parameter θ, the asymptotic bias of the maximum likelihood estimate $\hat{\theta}$ may be written as

$$b(\theta) = \frac{b_1(\theta)}{n} + \frac{b_2(\theta)}{n^2} + \ldots, \tag{11}$$

where n is usually interpreted as the number of observations but may be some other measure of the rate at which information is processed. The focus of this method is on a *general* method for reducing bias, with a specific aim being the removal of the term $O(n^{-1})$.

Two previously studied standard approaches, which have been extensively studied in the literature, are:

1. **Jackknife Method (Quenouille, 1949, 1956 [12, 13]):** It is very general and does not require calculation of $b_1(\theta)$ for its implementation.
2. The other standard approach simply substitutes $\hat{\theta}$ for the unknown θ in $b_1(\theta)/n$; the bias-corrected estimate is then calculated as

$$\hat{\theta}_{BC} = \hat{\theta} - \frac{b_1(\hat{\theta})}{n}. \tag{12}$$

Both of these methods succeed in removing the term $b_1(\theta)/n$ from the asymptotic bias. The jackknife has the advantage of requiring no theoretical calculation, but this is typically offset by a loss of precision. The estimator $\hat{\theta}_{BC}$ of (12) is generally second-order efficient.

A common feature of the two standard approaches is that they are 'corrective', rather than 'preventive' in character. The maximum likelihood estimate $\hat{\theta}$ is first calculated, then corrected. This is different from any philosophical considerations or matters of principle that might pertain here, a practical requirement for the application of either

method to a finite sample is the existence of $\hat{\theta}$ for that sample, and in the case of the jackknife for certain sub-samples also. In practice, particularly with small or medium-sized sets of data, it is not uncommon that $\hat{\theta}$ is infinite in some samples;

It is not uncommon that $\hat{\theta}$ is infinite in some samples; logistic models for a binary response, for example, are prone to such behaviour, In such cases, the jackknife and $\hat{\theta}_{BC}$ estimators are bias-reducing only in an asymptotic sense.

Motivated partly by this, **we explore an approach to bias reduction which does not depend on the finiteness of $\hat{\theta}$. A systematic correction will be developed for the mechanism that produces the maximum likelihood estimate, namely the score equation, rather than for the estimate itself.**

Firth's method is a procedure for reducing the leading-order (i.e., $\mathcal{O}(1/n)$) bias of maximum likelihood estimators. Rather than computing the ordinary MLE and then adjusting it *after* the fact, Firth's approach *modifies* the score equations themselves (or equivalently, penalizes the log-likelihood). In canonical exponential families, this penalty corresponds exactly to adding the Jeffreys invariant prior $\pi(\boldsymbol{\theta}) \propto \sqrt{\det\{I(\boldsymbol{\theta})\}}$ to the likelihood, where $I(\boldsymbol{\theta})$ is the Fisher information matrix [4].

A general modification of the score function is of the form:

$$U^*(\theta) = U(\theta) + A(\theta)$$

3.2 Method

Bias in $\hat{\boldsymbol{\theta}}$ arises from a combination of:

1. Unbiasedness of the score function at the true value of $\boldsymbol{\theta}$, i.e., $E\{U(\boldsymbol{\theta})\} = 0$.
2. Curvature of score function, i.e., $U''(\boldsymbol{\theta}) \neq 0$

So, if $U(\boldsymbol{\theta})$ is linear in $\boldsymbol{\theta}$ then $E(\hat{\boldsymbol{\theta}}) = \boldsymbol{\theta}$, but the positive curvature, as shown in Figure 1 for example combines with the unbiasedness of the score function to induce a bias in $\hat{\boldsymbol{\theta}}$ in this case in the positive direction.

Thus, we make a modification to $U(\boldsymbol{\theta})$:

$$U^*(\boldsymbol{\theta}) = U(\boldsymbol{\theta}) - I(\boldsymbol{\theta})\frac{B_1(\boldsymbol{\theta})}{n} \tag{13}$$

where $I(\boldsymbol{\theta})$ denotes Fisher's information of the sample, defined as the negative expected value of the first derivative of $U(\theta)$. The modified score function $U^*(\boldsymbol{\theta})$ originates from the simple triangle geometry shown in Figure 1 adapted from Firth (1993). If $\hat{\boldsymbol{\theta}}$ is subject to a positive bias $B_1(\boldsymbol{\theta})/n$, the score function is shifted downward at each point $\boldsymbol{\theta}$ by an amount $I(\boldsymbol{\theta})\boldsymbol{\theta})/n$, where $-I(\boldsymbol{\theta}) = U'(\boldsymbol{\theta})$ is the local gradient;

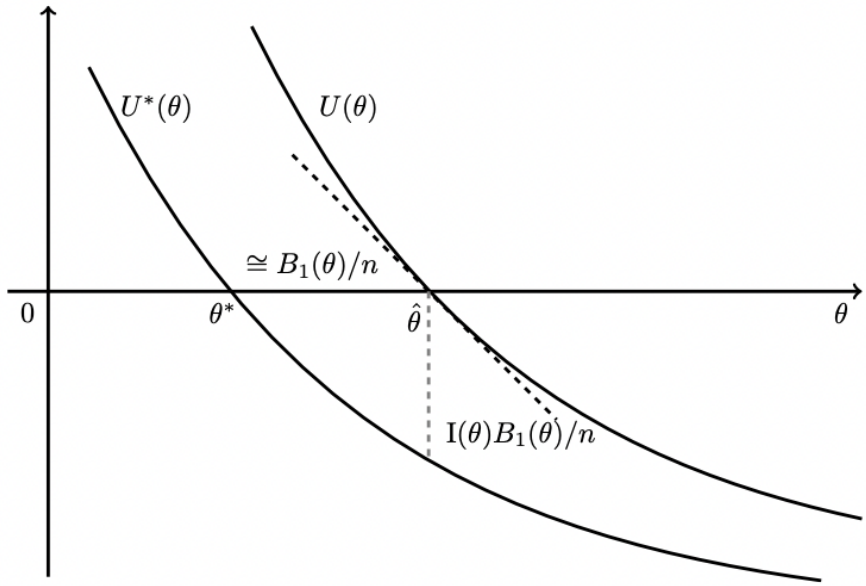

Figure 1: Modification of Score Function

And hence a modified estimate $\boldsymbol{\theta^*}$, given as a solution to $\boldsymbol{U^*(\theta)} = 0$. In the case of a vector parameter, (13) should be read as a vector equation, in which $I(\boldsymbol{\theta})$ is the Fisher information matrix.

In other words, if the MLE $\hat{\theta}$ has a positive first-order bias of $B_1(\theta)/n$, it can be removed by shifting the score function downward by $I(\theta)B_1(\theta)/n$, where the gradient of $U(\theta)$ is given by

$$\frac{\partial U(\theta)}{\partial \theta} = -I(\theta).$$

The corresponding estimate θ^* can then be calculated by setting the modified score function to 0, i.e.

$$U^*(\theta) = 0.$$

To formalize the heuristic argument above and extend it to problems other than canonical exponential families, it will be convenient to employ the notation and methods of McCullagh (1987, § 7.3) for log likelihood derivatives and their null cumulants [10]. The derivatives are denoted by:

$$U_r(\boldsymbol{\theta}) = \partial l/\partial \boldsymbol{\theta}^r, \quad U_{rs}(\boldsymbol{\theta}) = \partial^2 l/\partial \boldsymbol{\theta}^r \partial \boldsymbol{\theta}^s,$$

and so on, where $\boldsymbol{\theta} = (\boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^p)'$ is the parameter vector. The joint null cumulants are

$$\kappa_{r,s} = n^{-1}E\{U_r U_s\}, \quad \kappa_{r,s,t} = n^{-1}E\{U_r U_s U_t\}, \quad \kappa_{r,st} = n^{-1}E\{U_r U_{st}\},$$

and so on. We note here the well-known relationships

$$\kappa_{rs} + \kappa_{r,s} = 0, \quad \kappa_{rst} + \kappa_{r,s,t} + \kappa_{rs,t} + \kappa_{r,st} = 0. \tag{14}$$

Consider now a fairly general modification of the score function, of the form

$$U_r^*(\boldsymbol{\theta}) = U_r(\boldsymbol{\theta}) + A_r(\boldsymbol{\theta}),$$

in which A_r is allowed to depend on the data and is $O_p(1)$ as $n \to \infty$. Suppose that $\hat{\boldsymbol{\theta}}$ and $\boldsymbol{\theta}^*$ satisfy $U(\hat{\boldsymbol{\theta}}) = 0$ and $U^*(\boldsymbol{\theta}^*) = 0$, and write $\hat{\gamma} = n^{1/2}(\boldsymbol{\theta}^* - \theta)$. Then, by an argument closely following that of McCullagh (1987) [10], based on an expansion of $U_r^*(\boldsymbol{\theta}^*)$ about the true value $\boldsymbol{\theta}$, the bias of $\boldsymbol{\theta}^*$ is

$$E(n^{-1/2}\hat{\gamma}^r) = n^{-1}\kappa^{rs}\left\{-\kappa^{tu}(\kappa_{s,t,u} + \kappa_{s,tu})/2 + \alpha_s\right\} + O(n^{-3/2}),$$

where κ^{rs} denotes the inverse of the Fisher information matrix $\kappa_{r,s}$, α_s denotes the null expectation of A_s, and the summation convention applies. The term

$$-n^{-1}\kappa^{rs}\kappa^{tu}(\kappa_{s,t,u} + \kappa_{s,tu})/2 = n^{-1}B_1^r(\theta)$$

is the first-order bias of $\hat{\theta}$, for example Cox & Snell (1968) [**?**]. The modification A_r therefore removes the first-order term if it satisfies

$$\kappa^{rs}\alpha_s = -B_1^r + O(n^{-1}),$$

the solution to which is

$$\alpha_r = -\kappa_{rs}B_1^s + O(n^{-1}).$$

where $B_1 = B_1(\theta)$. In matrix notation, then, the vector α should be such that

$$E(A) = -I(\boldsymbol{\theta})B_1(\boldsymbol{\theta})/n + O(n^{-1/2}).$$

The obvious candidates for a bias-reducing choice of A are therefore

$$A^{(E)} = -I(\boldsymbol{\theta})B_1(\boldsymbol{\theta})/n \quad \text{and} \quad A^{(O)} = -I(\boldsymbol{\theta})B_1(\boldsymbol{\theta})/n,$$

using expected and observed information, respectively. In the case of an exponential family in canonical parameterization the observed information $I(\boldsymbol{\theta})$ does not involve the data, so $A^{(O)}$ and $A^{(E)}$ coincide. More generally, either of these modifications removes the $O(n^{-1})$ bias term.

3.3 Jeffreys Prior as Bias-Reducing Penalty Function

Firth's approach can also be described as a penalized likelihood method. The usual likelihood function $\mathcal{L}(\theta)$ is penalized by a factor equal to the square root of the determinant of the information matrix $|\mathbf{I}(\theta)|$. Firth (1993) also showed that if the target parameter is the canonical parameter of an exponential family, his correction scheme is equivalent to penalizing the likelihood by the Jeffery's invariant prior, which is essentially the square root of the

log determinant of the Fisher information matrix of the parameters [6].

If θ is the canonical parameter of an exponential family model, $\kappa_{r,st} = 0$ for all r, s and t. Therefore the rth element of $A^{(E)}(\theta)$, or equivalently of $A^{(O)}(\theta)$, is given by

$$a_r = -n\kappa_{r,s}b_1^s/n = \kappa_{r,s}\kappa^{s,t}\kappa^{u,v}\kappa_{t,u,v}/2 = \kappa^{u,v}\kappa_{r,u,v}/2 = -\kappa^{u,v}\kappa_{ruv}/2,$$

using the identities (14). In matrix notation, this may be written as

$$a_r = \frac{1}{2}\,\mathrm{tr}\left\{I^{-1}\left(\frac{\partial I}{\partial\theta_r}\right)\right\} = \frac{\partial}{\partial\theta_r}\left\{\frac{1}{2}\log|I(\theta)|\right\}.$$

Solution of $U_r^* \equiv U_r + a_r = 0$ therefore locates a stationary point of

$$l^*(\theta) = l(\theta) + \frac{1}{2}\log|I(\theta)|$$

or, equivalently, of the penalized likelihood function

$$L^*(\theta) = L(\theta)|I(\theta)|^{1/2}.$$

The penalty function $|I(\theta)|^{1/2}$ here is the Jeffreys (1946) invariant prior for the problem. Thus, for the canonical parameter of an exponential family model, the $O(n^{-1})$ bias is removed by calculation of the posterior mode based on this prior.

Hence, the penalized likelihood function for Firth's model is thus

$$\mathcal{L}^*(\theta) = \mathcal{L}(\theta)\cdot|\mathbf{I}(\theta)|^{\frac{1}{2}}. \tag{15}$$

Taking the natural logarithm of equation (15) yields the corresponding penalized log likelihood function:

$$\ell^*(\theta) = \ell(\theta) + \frac{1}{2}\log|\mathbf{I}(\theta)|.$$

3.4 Modified Score Function in a more Generalized Setting

Now, we discuss the modification of the score function in a more general setting which includes exponential family models in non-canonical parametrization as well as non-exponential models.

The modified score function in this general setting has the form $U_r^* = U_r + A_r$, where $A_r(\theta)$ is based either on the expected information,

$$A_r = A_r^{(E)} = n\kappa_{r,s}\kappa^{s,t}\kappa^{u,v}(\kappa_{t,u,v} + \kappa_{t,uv})/(2n) \tag{16}$$
$$= \kappa^{u,v}(\kappa_{r,u,v} + \kappa_{r,uv})/2, \tag{17}$$

or on the observed information,

$$A_r = A_r^{(O)} = -U_{rs}\kappa^{s,t}\kappa^{u,v}(\kappa_{t,u,v} + \kappa_{t,uv})/(2n).$$

Intuitively, we can say that estimates derived using $A_r^{(O)}$ may be preferable in terms of efficiency. To explore this further, consider an expansion of $U_r^*(\theta^*)$ about $\hat{\theta}$. By definition,

$$0 = U_r^*(\theta^*) = U_r(\theta^*) + A_r(\theta^*).$$

If $A_r(\theta) = A_r^{(O)}(\theta) = U_{rs}(\theta)B_1^s(\theta)/n$, we have that

$$(\theta^* - \hat{\theta})^r = -B_1^r(\hat{\theta})/n + O_p(n^{-2}), \tag{18}$$

while, if $A_r(\theta) = A_r^{(E)}(\theta) = -I_{rs}(\theta)B_1^s(\theta)/n$,

$$(\theta^* - \hat{\theta})^r = -B_1^r(\hat{\theta})/n - I^{rs}(\hat{\theta})\{U_{st}(\hat{\theta}) + I_{st}(\hat{\theta})\}B_1^t(\hat{\theta})/n + O_p(n^{-2}). \tag{19}$$

The difference $U_{st}(\hat{\theta}) + I_{st}(\hat{\theta})$ between expected and observed information at the maximum likelihood estimate is $O_p(n^{-1/2})$ in general, e.g., Pierce (1975), so that the extra term in (8) is $O_p(n^{-3/2})$. In the special case of a full exponential family model, with any parameterization, this term vanishes [11].

From (17) it may be concluded that if U^* is calculated using the observed information function, θ^* agrees with $\hat{\theta}_{BC}$ to second order. This is not the case if expected information is used, unless the model is a full exponential family. Thus both forms of U^* yield estimators that are first-order efficient, and the results of Efron (1975) show that both forms are second-order efficient in full exponential family models. In curved exponential families and more generally, use of the modification $A^{(E)}$ involves a second-order loss of precision relative to use of $A^{(O)}$ [**?**].

3.5 Example: The Logistic Regression Model

For *ith* subject, the response y_i is Binomial distributed with probability of success π_i where $x_i = (x_1 \cdots x_p)$

$$Y_i \sim Bin(n_i, \pi_i)$$

$$log\Big(\frac{\pi_i}{1-\pi_i}\Big) = \beta_0 + \sum_{i=1}^{p} \beta_i x_i$$

and

$$\pi_i = \frac{e^{\beta_0 + \mathbf{x_i^T}\beta}}{1 + e^{\beta_0 + \mathbf{x_i^T}\beta}}$$

where β_0 is an intercept term and $\beta = (\beta_1 \cdots \beta_p)^T$

Let $\theta = (\beta_0, \beta)^T$

The likelihood is:

$$\mathcal{L}(\boldsymbol{\theta}) = \prod_{i=1}^{n} \pi_i^{y_i}(1-\pi_i)^{1-y_i}$$

The log-likelihood is:

$$\ell(\theta) = log\mathcal{L} = \sum_{i=1}^{n} \left\{ y_i \log(\pi_i) + (1-y_i)\log(1-\pi_i) \right\}. \tag{20}$$

One of the most popular methods to estimate the unknown coefficients $\boldsymbol{\theta}$ is maximum likelihood (ML) estimation. In order to find the value that maximizes $\log \mathcal{L}(\boldsymbol{\theta})$, partial derivatives of the log-likelihood function with respect to $\boldsymbol{\theta}$ are calculated as follows:

$$\frac{\partial \log \mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = U(\boldsymbol{\theta}) = \sum_{i=1}^{n} (y_i - \pi_i)\boldsymbol{x}_i. \tag{21}$$

The second derivative with respect to $\boldsymbol{\theta}$ of the log likelihood function, or the Hessian matrix, can be expressed as

$$\mathbf{H}(\boldsymbol{\theta}) = \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}'} = -\sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^T \pi_i(1-\pi_i).$$

The solution to the score equation $U(\boldsymbol{\theta}) = 0$ gives the ML estimate of $\boldsymbol{\theta}$, $\hat{\boldsymbol{\theta}}$. There is no analytical solution to the score equations; therefore, numerical methods (e.g. Newton-Raphson or Fisher Scoring) are used to find $\hat{\boldsymbol{\theta}}$. With the starting value $\boldsymbol{\theta}^{(1)}$, $\hat{\boldsymbol{\theta}}$ is obtained iteratively until the convergence of parameter estimates. The iterative Newton-Raphson algorithm is defined as:

$$\boldsymbol{\theta}^{(r+1)} = \boldsymbol{\theta}^{(r)} + \mathbf{I}^{-1}(\boldsymbol{\theta}^{(r)}) U(\boldsymbol{\theta}^{(r)}). \tag{22}$$

where the superscript (r) denotes the number of the iteration, and $\mathbf{I}(\boldsymbol{\theta})$ denotes the Fisher information matrix, i.e., the expected value of minus the second derivative of the log likelihood, evaluated at $\boldsymbol{\theta}$. In the context of logistic regression,

$$\mathbf{I}(\boldsymbol{\theta}) = -\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}'} = \mathbf{X}^\top \mathbf{W} \mathbf{X}, \tag{23}$$

where $\mathbf{X}$ is an $n \times (p+1)$ design matrix with elements in the first column being 1, and $\mathbf{W}$ is an $n \times n$ diagonal matrix with general element $\pi_i(1-\pi_i)$.

Asymptotically, the MLEs $\hat{\boldsymbol{\theta}}$ are normally distributed around the true parameter $\boldsymbol{\theta}$, and the estimated variance-covariance matrix, $\text{Var}(\hat{\boldsymbol{\theta}})$, is obtained by evaluating the inverse of the Fisher information matrix $\mathbf{I}^{-1}$ at the MLEs, with the standard errors of single parameters corresponding to the diagonal elements of the matrix.

We make a modification to $U(\boldsymbol{\theta})$ such that:

$$U^*(\boldsymbol{\theta}) = U(\boldsymbol{\theta}) - I(\boldsymbol{\theta}) \frac{B_1(\boldsymbol{\theta})}{n}$$

and hence a modified estimate θ^* is a given as a solution $U^*(\theta) = 0$.

Taking the derivative of equation (20) with respect to $\boldsymbol{\theta}$, the modified score function has the following form:

$$\frac{\partial \ell^*(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = U^*(\boldsymbol{\theta}) = \sum_{i=1}^{n} \left[y_i - \pi_i + h_i \left(\frac{1}{2} - \pi_i \right) \right] \boldsymbol{x}_i,$$

where h_i is the ith diagonal element of the penalized version of the hat matrix:

$$\mathbf{H} = \mathbf{W}^{1/2}(\mathbf{X}^\top \mathbf{W}\mathbf{X})^{-1}\mathbf{X}^\top \mathbf{W}^{1/2}.$$

Penalized maximum likelihood estimates can be obtained by applying the standard method described in (22) with the $U(\boldsymbol{\theta}^{(r)})$ term replaced by $U^*(\boldsymbol{\theta}^{(r)})$. By imposing the penalty term at each step in the iteration process, this modified score function prevents the estimates from going off to infinity and failing to converge and ensures finite ML estimates when there is separation in the data. Similarly, the standard error can be estimated based on the roots of the diagonal elements of $\mathbf{I}^{-1}(\hat{\boldsymbol{\theta}})$, the standard information matrix from the unpenalized log-likelihood evaluated at $\hat{\boldsymbol{\theta}}$.

3.6 R Tutorial

Generating pseudo-data

```
install.packages("brglm2")
library(brglm2)

set.seed(2025)
generate_data <- function(n, beta_0, beta_1){
  X1 <- rnorm(n, mean = 0, sd = 2)

  X <- cbind(1, X1)
  beta_true <- c(beta_0, beta_1)

  eta <- X %*% beta_true
  p <- 1 / (1 + exp(-eta))
  y <- rbinom(n, size = 1, prob = p)
  df <- data.frame(
    y  = y,
    X1 = X1
  )
  return(df)
}

data <- generate_data(50, 0.5, -0.3)
```

```r
head(data)
```

Model Fit Using MLE

```r
fit_MLE <- glm(y ~ X1, family=binomial(link="logit"), data=data)

summary(fit_MLE)
```

Model Fit Using the Firth Method

```r
fit_Firth <- glm(y ~ X1, family=binomial(link="logit"), data=data,
                 method=brglmFit, type="AS_mean")

summary(fit_Firth)
```

Interpreting the outputs:

- The **coefficient for** `X1` is slightly less negative under Firth correction (-0.3862 vs -0.4180) — still indicating a negative relationship.
- The **standard error** is slightly lower for Firth, which leads to:
  - A slightly **higher p-value**, though still significant at the 5% level.
- **Firth correction reduces bias**, especially useful in small samples or when there is separation. While the difference is not drastic here, Firth's method tends to **stabilize estimates**.
- **AIC** is slightly higher under Firth, indicating a marginally worse fit — this is expected, as bias reduction introduces a penalty.
- Firth uses **one fewer iteration** to converge — possibly reflecting a more stable estimation path.

4 Penalization and Bias Reduction in Logistic Regression (Log–F Priors)

In the previous section, we briefly introduced the equivalency of likelihood penalization with the application of prior distribution under Firth's method, using Jeffrey's Prior. In this section, we will expand on this idea by exploring the usage of a log-F prior under the logistic regression case. Unlike Jeffrey's Prior, which is non-adjustable and uninformative, and is

used strictly for bias-removal of order $O(\frac{1}{n})$, the log-F prior allows for adjustable integration of prior knowledge and is therefore a stronger prior from a Bayesian perspective. This report summarizes the technique defined by Greenland & Mansournia [5].

4.1 Single-parameter Logistic Regression Example

As an example, consider a single-parameter logistic regression model

$$\log\Big(\frac{\pi}{1-\pi}\Big) = \beta_0$$

Here, the likelihood function for β_0 is

$$L(\beta_0) \;=\; \pi^y\,(1-\pi)^{n-y},$$

Under Firth's bias reduction method, this likelihood would be penalized by Jeffrey's Prior:

$$L^*(\beta_0) \propto\; L(\beta_0) * \Big[\frac{n\,e^{\beta_0}}{\big(1+e^{\beta_0}\big)^2}\Big]^{1/2}$$

Since the Fisher's Information under this setting is:

$$F(\beta_0) \;=\; \mathbb{E}\Big[-\frac{d^2}{d\beta_0^2}\,\ell(\beta_0)\Big] = \frac{n\,e^{\beta_0}}{\big(1+e^{\beta_0}\big)^2}$$

However, we can consider a more adaptive prior, which we call the log-F prior:

$$L^*(\beta_0) \propto\; L(\beta_0) * \Big[\frac{e^{\beta_0}}{\big(1+e^{\beta_0}\big)^2}\Big]^{m/2}$$

Note that while the two priors seem quite similar in form, with the exception that the log-F prior is raised to the power of $m/2$ as opposed to $1/2$, this does not remain the case as we move into the multiple-parameter logistic regression setting. The similarity between the two priors under this special case allows for a clearer comparison in the following section. [5]

4.1.1 Dispersion of Priors

Although both priors are centered around 0, a useful way to compare Jeffreys penalization and the log-F method is by comparing the dispersion of each prior. The shape of Jeffreys' prior depends solely on Fisher's Information, while the log-F family introduces the tuning parameter m, whose magnitude directly scales the dispersion. A less dispersed prior allocates relatively more probability mass to the mode of the prior, and thereby exerts a stronger pull on the likelihood and shifts the posterior mode closer towards 0 and further from the MLE. Increasing m in the log-F setting allows us to increase this shrinkage effect.

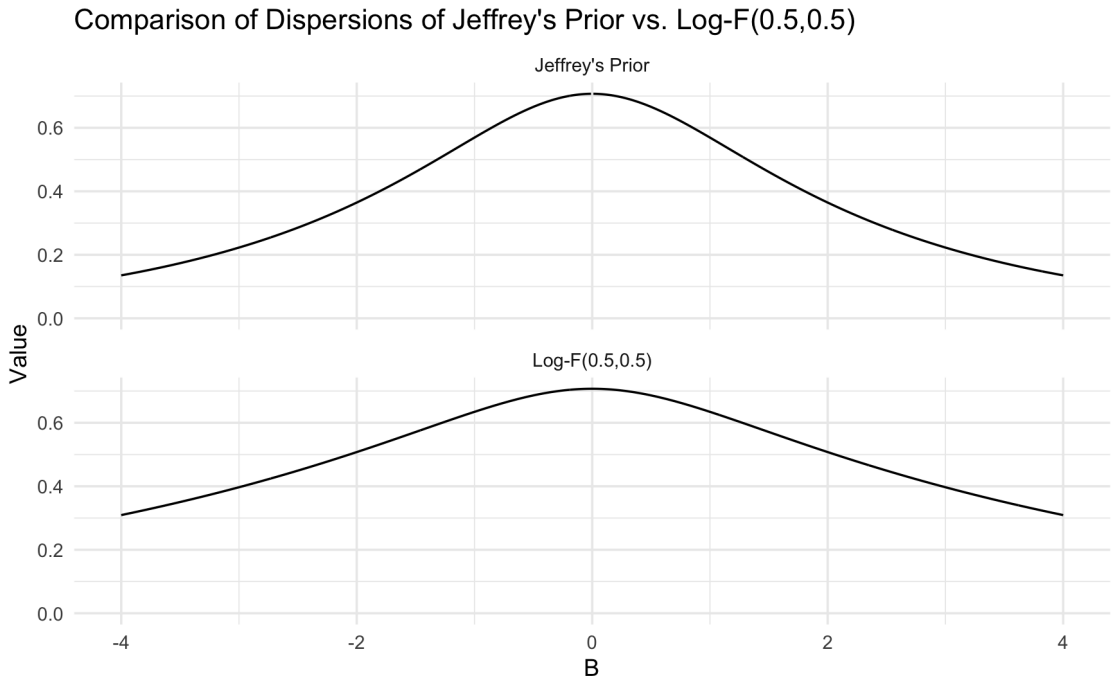

Figure 2: Jeffrey's Prior vs log-F(0.5, 0.5)

Figure 2 compares the dispersion of Jeffrey's Prior, against the log-F prior with $m = 0.5$. The proportionality have been adjusted such that the two distributions have the same maximum value, for better comparison of dispersion. We can see that in this case, the $Log - F(0.5, 0.5)$ prior has less dispersion, which will result in a weaker pull towards 0, and therefore a $\hat{\beta}_0$ which is closer to the MLE and further from 0 compared to the estimate under the Firth method.

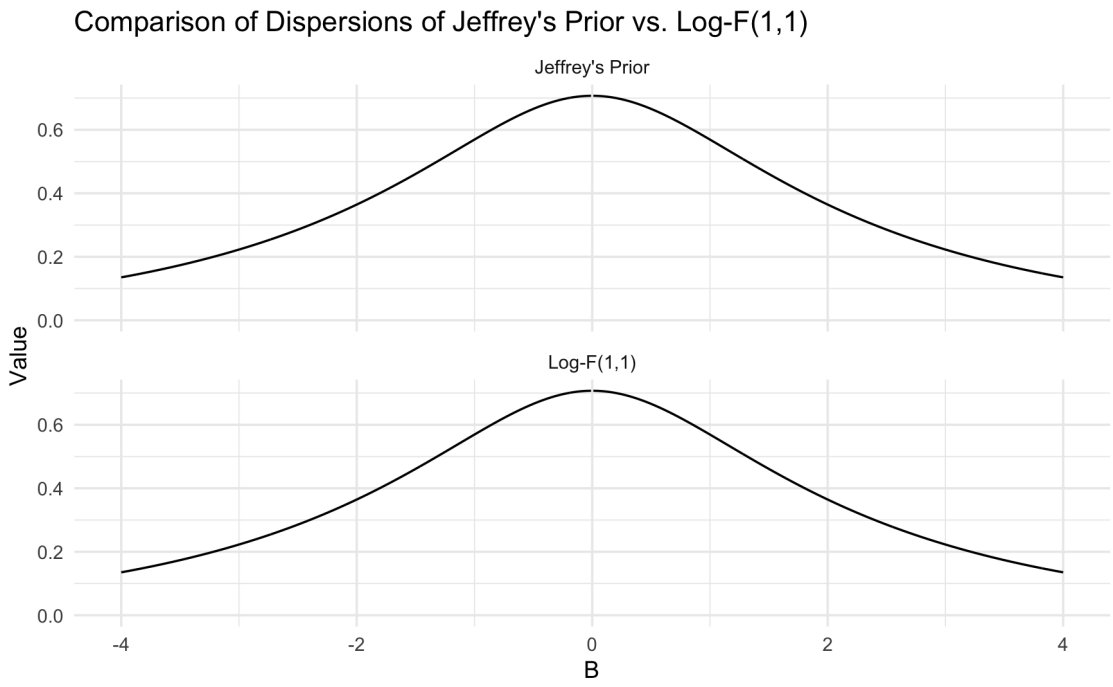

Figure 3: Jeffrey's Prior vs log-F(1,1)

Figure 3 shows that at $m = 1$, the two priors become equivalent in dispersion. This similarity is not maintained under the multi-parameter case.

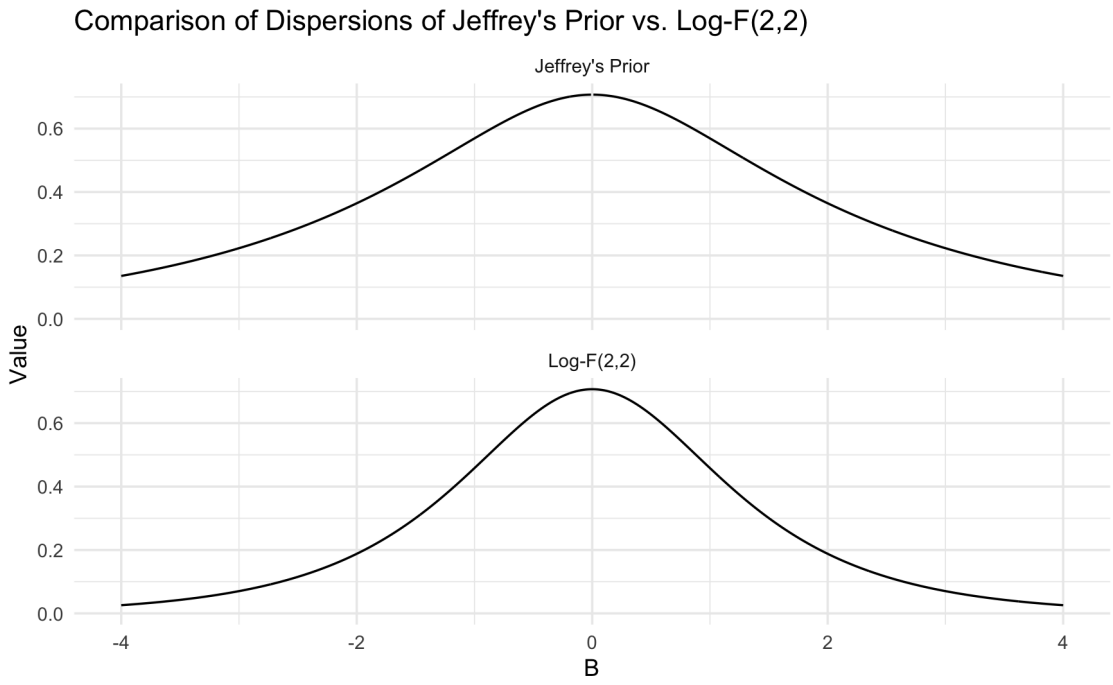

Figure 4: Jeffrey's Prior vs log-F(2, 2)

Figure 4 shows that with a larger m value, the log-F prior now has a stronger concentration around 0, and will result in stronger shrinkage of $\hat{\beta}_0$. In general, we would want to impose a smaller m when we believe the true β to be large, and a larger m for βs believed to be closer to 0.

4.2 Multiple-parameter Logistic Regression

For a multi-parameter logistic regression model:

$$\log\Big(\tfrac{\pi_i}{1-\pi_i}\Big) \;=\; \mathbf{x}_i^{\mathsf{T}}\,\boldsymbol{\beta},$$

Firth's Adjusted Likelihood is:

$$L^*(\boldsymbol{\beta}) = L(\boldsymbol{\beta}) * |F(\boldsymbol{\beta})|^{1/2}$$

Where Fisher's Information is a function of the design matrix, and is centred around $\boldsymbol{\beta} = \mathbf{0}$. This shifts all coefficients $\boldsymbol{\beta}$ towards 0. In comparison, the log-F prior allows for a separate, independent penalty to be applied to each coefficient as fit, where the strength of each penalty can be adjusted accordingly. As an example, if we wish to penalize β_j of $\boldsymbol{\beta}$ with a penalty strength of m_j, our posterior likelihood would be proportional to

$$L^*(\boldsymbol{\beta}) \propto\; L(\boldsymbol{\beta}) * \Big[\frac{e^{\beta_j}}{\big(1+e^{\beta_j}\big)^2}\Big]^{m_j/2}$$

To penalize additional coefficients, we simply continue multiplying the likelihood by the additional priors of the same form, each with its own adjustable value of m. [5]

4.3 Choice of m

The suggestion for the choice of m is to base the decision on prior knowledge of the corresponding β coefficient. If we believe that e^{β} rarely exceeds a value U, we set U as the upper-bound of a 95% confidence interval, and find an F-distribution such that $P(F > U) \approx 0.025$. [5]

The inspiration for this rule comes from the following: If we were to assume that e^{β} has an F-distribution with degrees of freedom m, m, then the probability density function is as:

$$f_{e^{\beta}}(x) \;=\; \frac{1}{B\big(\frac{m}{2},\,\frac{m}{2}\big)}\, x^{\frac{m}{2}-1}\,(1+x)^{-m}, \qquad x > 0.$$

Via transformation, β then has a distribution of:

$$f_{\beta}(\beta) \;=\; \frac{1}{B\big(\frac{m}{2},\,\frac{m}{2}\big)}\,\frac{e^{\frac{m}{2}\beta}}{\big(1+e^{\beta}\big)^{m}}, \qquad -\infty < \beta < \infty.$$

Where B is the beta function, which is constant for β. This distribution is then proportional to our log-F prior, and is also the inspiration for its name. However, we don't truly assume that $e^{\beta} \sim F(m, m)$, we treat m purely as a continuous hyper-parameter rather than as literal "degrees of freedom," so m is free to take non-integer values.

4.4 Implementation Example

The above penalized likelihood is mathematically equivalent to adding $m_j/2$ successes and $m_j/2$ failures where the covariate corresponding to our coefficient, x_j, is set to 1, and all other covariates, including the intercept constant, is set to 0. [5] This makes implementation very straight forward. Below shows a simple example with `X1` and `X2` as predictor variables, and we impose a penalty of $m = 3$ on $\hat{\beta}_1$ which corresponds to `X1` by adding pseudo-data. Existing data is in the dataframe `df` with columns `y`, `x1`, `X2`. In the case that m is not an even integer, you may receve a warning from R of non-integer success counts, which can be ignored as it does not impact our results.

```
#add a manual intercept, which takes a value of 0 for existing data
df$intercept = 1

#we will fit the glm using the weight method, assign a weight of 1 to existing data
#(or as appropriate based on existing grouping)
df$weight = 1

#select our penalty parameter m
m=3

#add_row function is from tibble/tidyverse
df <- add_row(df,
              y=0.5,   #y represents the proportion of successes out of weight trials
              X1 = 1,
              X2 = 0,
              intercept = 0,
              weight = m)

#fit a glm function with -1, which removes the automatic intercept fit by R
model <- glm(y ~ -1 + intercept + X1 + X2,
              family=binomial,
              data=df,
              weight = weight)
```

4.5 Arguments for Log-F Prior over Jeffrey's Prior

There are several arguments against using Jeffrey's Prior for penalization. Firstly, it is data-dependent: the prior is defined through the observed design matrix and Fisher information, not through an external belief. This means that Jeffrey's Prior is not a true prior from a Bayesian perspective. Secondly, in very low samples, coefficients that are inherently far from 0 tend to be over-shrunk towards the neighbourhood of 0 by Jeffrey's Prior, and there is no adjustable parameter to alleviate the strength of this shrinkage. Finally, Jeffrey's Prior penalizes all coefficients, which is generally not ideal as we often do not wish to impose a penalty on the intercept coefficient to avoid shifting base odds. [5]

In comparison, the log-F prior allows for independent, adjustable priors, imposed on coefficients of choice. It allows for the strength of the penalty to be dependent on prior information of the coefficient, which makes it a much stronger Bayesian prior compared to

Jeffrey's Prior. Coefficient values which are known to be large in value may have a softer penalty imposed to avoid over-shrinkage, and we can avoid penalizing the intercept term altogether. [5]

4.6 Covariate Coding Suggestions

Greenland & Mansournia [5] provide practical suggestions for dealing with different formats of data when applying this technique. The key suggestions are:

- **Use meaningful units.** Because an estimated odds ratio e^{β} corresponds to a *unit* change in its covariate, the choice of unit has a direct impact. Greenland & Mansournia suggest scaling continuous variables in contextually sensible units. Using simple SI units rather than study–specific measurements makes prior and posterior odds-ratio limits easy to compare across studies.

- **Centre at an interpretable zero.** Re-express quantitative covariates so that 0 falls inside the data range and represents a meaningful reference point. This yields an intercept that is the logit of the outcome risk when all covariates are at sensible baseline values.

- **Intercept considerations.** The intercept is determined almost entirely by coding choices and sampling design. It may shift when covariates are rescaled or centred. Penalizing the intercept is generally discouraged.

- **Main effects versus interactions.** Good centring also reduces collinearity between main-effect and product terms, allowing us to apply weak default priors on main effects while applying stronger shrinkage to second-order terms that are inherently less certain.

- **Categorical Levels.** When two regression coefficients, β_1 and β_2, are two levels of a categorical variable, the natural null hypothesis is $\beta_1 = \beta_2$, so treating them as independent is inappropriate. We can instead re-parameterize the model into a shared baseline effect β_1 and a contrast term $\beta_2^* = \beta_2 - \beta_1$, for which an independence prior makes sense.

4.7 Data Example

4.7.1 Bias Comparison with Strong Prior Knowledge

We generate some existing data with known coefficient values to compare the performance of Jeffrey's Prior against the log-F prior under different values of m. The generated data uses true values of $\beta_0 = -3$ and $\beta_1 = -4$, both of which are relatively far from zero.

First, We avoid penalizing β_0 as recommended, and assume we have prior knowledge that β_1 is, at most, roughly 8. This gives us a penalty of $m_1 = 0.8$

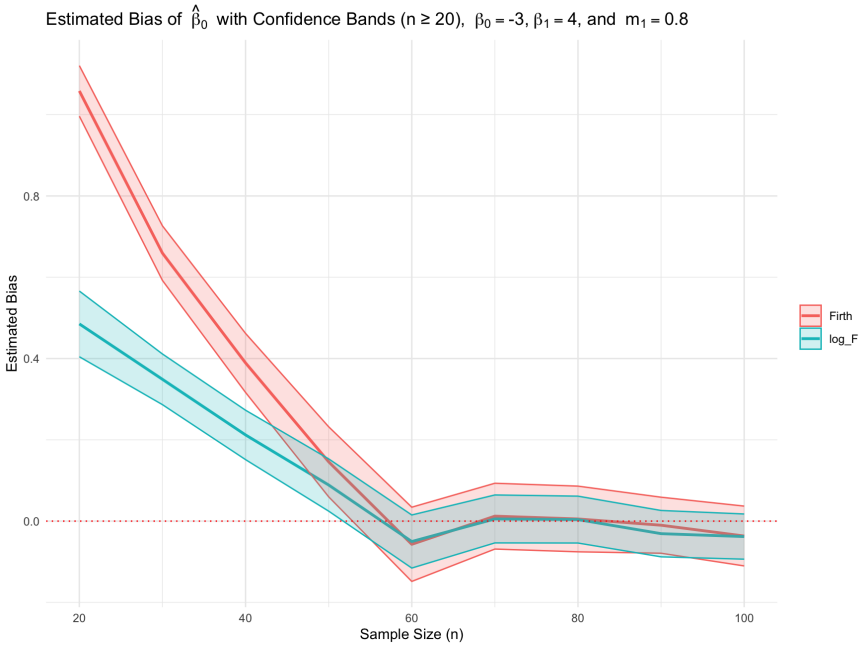

Figure 5: Comparison of Estimated Bias of Estimated β_0 using log-F VS Jeffrey's Prior, $m_1 = 0.8$

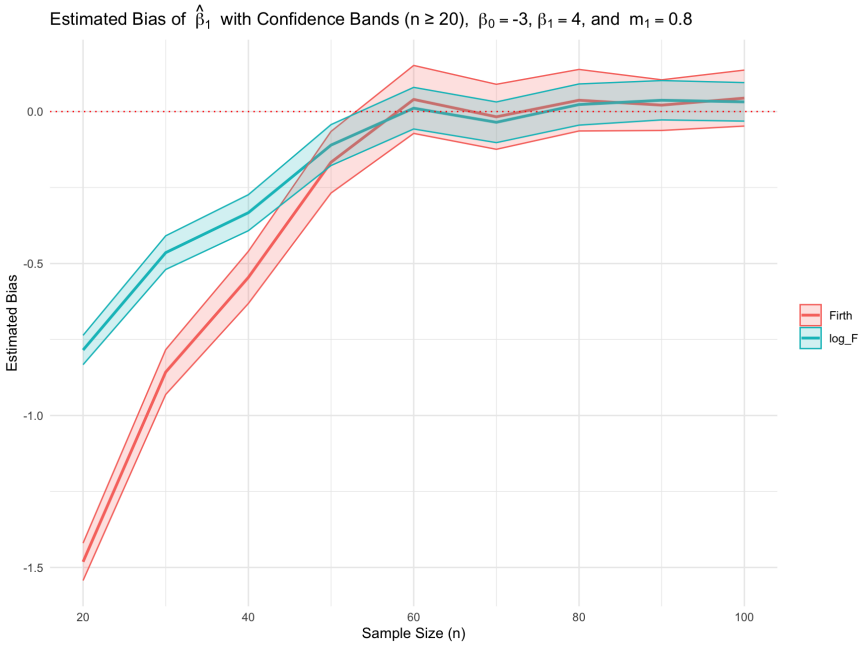

Figure 6: Comparison of Estimated Bias of Estimated β_1 using log-F VS Jeffrey's Prior, $m_1 = 0.8$

Figure 5 and Figure 6 show that both $\hat{\beta}_0$ and $\hat{\beta}_1$ have less bias in smaller sample sizes compared to estimates obtained with Jeffrey's Prior. Specifically, Jeffrey's Prior has added

bias in the direction of over-shrinkage towards 0. By avoiding penalties on the intercept $\hat{\beta}_0$ and by imposing a smaller value of m_1 on $\hat{\beta}_1$ due to prior knowledge that its true value is large, we avoid this issue under the log-F prior.

4.7.2 Bias Under Poorly Informed m

In a second case, we consider that our prior knowledge of β_1 leads us to believe it rarely exceeds a value of 5. This is quite unlikely for a true value of $\hat{\beta}_1 = 4$. The corresponding m value is then calculated as approximately $m_1 = 1.35$.

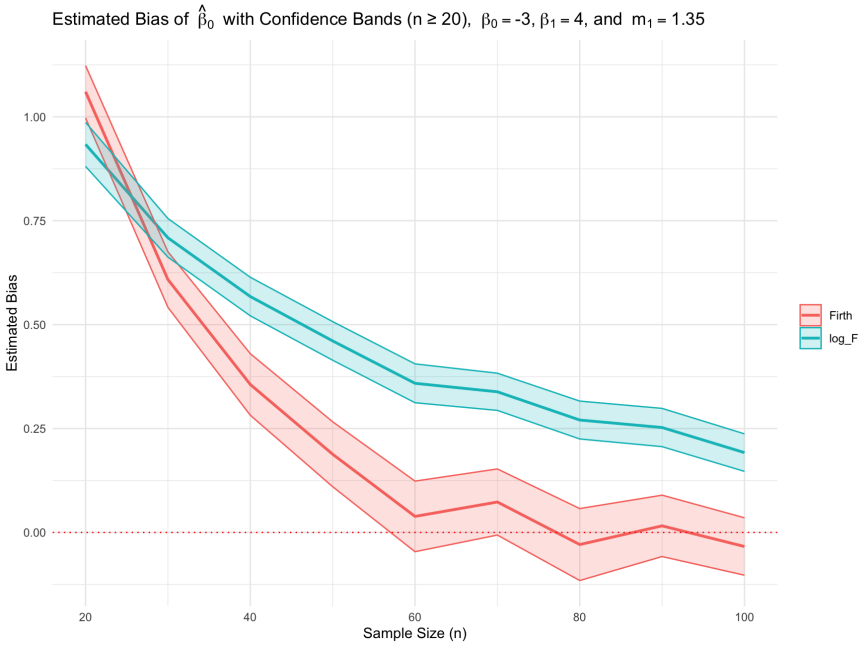

Figure 7: Comparison of Estimated Bias of Estimated β_0 using log-F VS Jeffrey's Prior, $m_1 = 1.35$

Figure 7 and Figure 8 show that both coefficients are now over-shrunk under this poorly chosen m_1. Interestingly, even though a penalty is only applied to $\hat{\beta}_1$, $\hat{\beta}_0$ is also over-shrunk as an indirect result of the added pseudo-data.

We can see that a poorly-informed choice of m can lead to additional bias added to our estimates. As a result, it's recommended to only adjust m when you have strong prior knowledge of your coefficients. As a default prior, however, Greenland & Mansournia [5] recommend imposing a penalty of $m = 1$ to all coefficients with the exception of the intercept. This is argued to be a better default prior than Jeffrey's Prior, even without prior knowledge, due to the lack of penalty on the intercept coefficient. The performance of these two default priors will be compared in the following section.

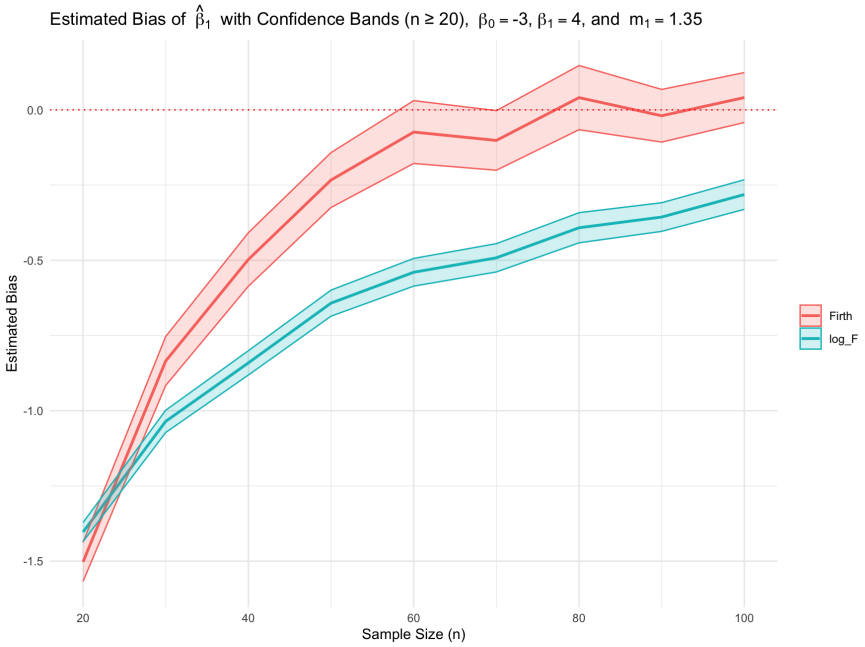

Figure 8: Comparison of Estimated Bias of Estimated β_1 using log-F VS Jeffrey's Prior, $m_1 = 1.35$

5 Simulation Study

To compare the three methodologies, we conducted a simulation study across a range of scenarios. Specifically, we employed a binomial GLM to assess the performance of the methods. We assumed that the observations Y_i are independent, with each Y_i following a binomial distribution: $Y_i \sim \text{Binomial}(\pi_i)$. The model is specified as follows:

$$logit(\pi_i) = \beta_0 + \beta_1 X_{1i}, \tag{24}$$

where X_{1i} is a continuous random variable drawn from a standard normal distribution. For the purpose of our study, all confidence intervals (CI) used were 95% Wald-type CIs. Also, for each true parameter vector (β_0, β_1), we generated 1000 datasets. Further details on the implementation can be found in the code provided in the Appendix.

The simulation study is divided into three distinct scenarios - easy, moderate, and extreme - based on the true values of the π_i for the majority of the observations. This is achieved by considering the fact that $X_{1i} \sim N(0, 1)$.

1. **Easy case**: In this scenario, we set the true parameters (β_0, β_1) such that most of the observations Y_i have π_i near 0.5.

2. **Moderate case**: In this scenario, we set the true parameters (β_0, β_1) such that most observations Y_i have π_i values either between 0.2 and 0.4, or between 0.6 and 0.8.

3. **Extreme case**: In this scenario, we set the true parameters (β_0, β_1) such that most observations Y_i have π_i values either between 0.01 and 0.1, or between 0.9 and 0.999.

5.1 Easy Case

Figure 9 presents estimated bias of $\hat{\beta}_1$, along with 95% confidence intervals, under the easy scenario. For sample sizes $n \geq 20$, the asymptotic bias correction method exhibits the lowest estimated bias, closely followed by Firth's approach. The log-F approach and standard maximum likelihood estimation (MLE) show slightly higher and similar levels of estimated bias in this range. However, when the sample size is very small, i.e., $n = 10$, the limitations of MLE and the asymptotic bias correction become apparent—their estimated bias increase dramatically. In contrast, Firth's method and the log-F approach remain stable and continue to have small values of estimated bias, highlighting their robustness in small-sample settings.

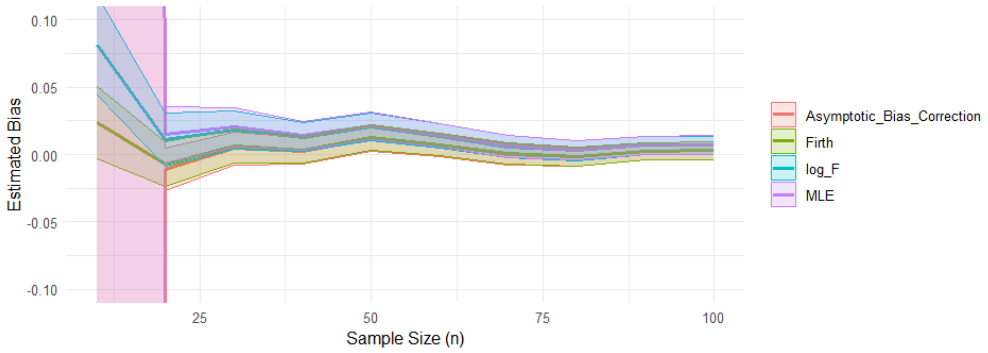

Figure 9: Estimated bias of $\hat{\beta}_1$ with 95% confidence intervals in easy case ($\beta_0 = 0, \beta_1 = 0.1, m_1 = 1$).

Figure 10 displays the estimated coverage probabilities of the 95% Wald CI for β_1 under the easy scenario. Across most sample sizes, all methods produce intervals with estimated coverage probabilities above the nominal 95% level. The asymptotic bias correction and Firth methods yield similar estimated coverage probabilities, except at $n = 10$. The estimated coverage probabilities for the CIs based on log-F and MLE methods are nearly identical and generally lower than those of the asymptotic bias correction and Firth based CIs.

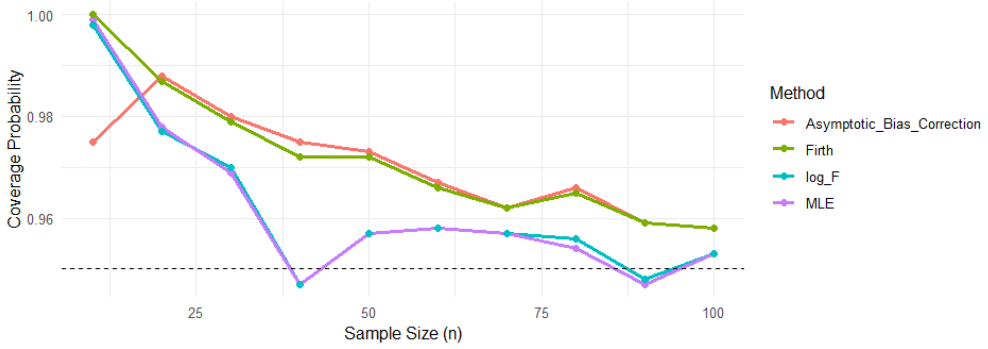

Figure 10: Estimated coverage probability of 95% Wald CI for β_1 in easy case ($\beta_0 = 0, \beta_1 = 0.1, m_1 = 1$).

5.2 Moderate Case

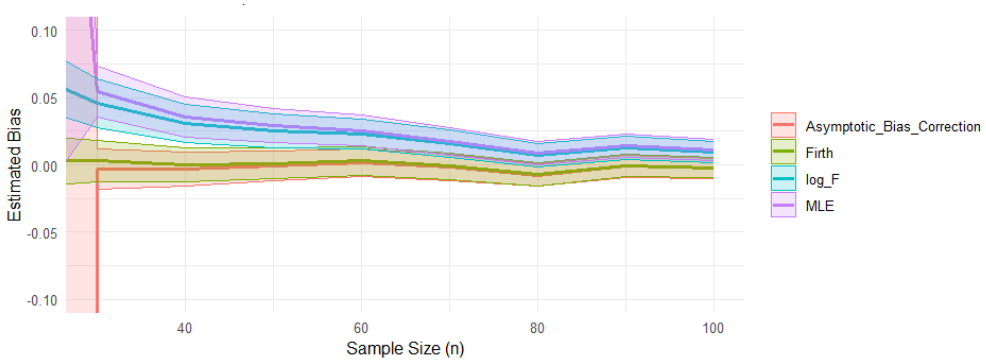

Figure 11: Estimated bias of $\hat{\beta}_1$ with 95% confidence intervals in moderate case ($\beta_0 = 1, \beta_1 = 0.3, m_1 = 1$).

Figure 11 shows the estimated bias of $\hat{\beta}_1$, along with 95% Wald confidence intervals, under the moderate scenario. For sample sizes $n \geq 30$, we observe a similar pattern to that seen in the easy case starting at $n = 20$: the asymptotic bias correction method yields the smallest estimated bias, followed closely by Firth's approach, with log-F and MLE exhibiting higher bias, and MLE consistently performing the worst. However, for smaller sample sizes ($n = 10$ and $n = 20$), both the MLE and asymptotic bias correction estimators show a substantial increase in estimated bias, with estimates deteriorating rapidly. In contrast, the log-F and Firth estimators remain stable and maintain low bias across these smaller sample sizes. Notably, while this sharp increase in estimated bias for MLE and asymptotic bias correction appeared only at $n = 10$ in the easy scenario, it emerges at both $n = 10$ and $n = 20$ in the moderate case, indicating greater sensitivity to small sample sizes under this setting.

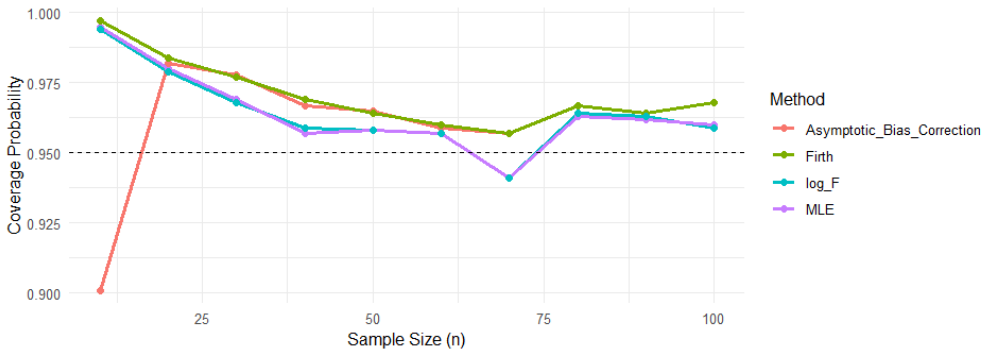

Figure 12: Estimated coverage probability of 95% Wald CI for β_1 in moderate case ($\beta_0 = 1, \beta_1 = 0.3, m_1 = 1$).

Figure 12 displays the estimated coverage probabilities of the 95% Wald CI under the moderate scenario. For sample size $n \geq 20$, the pattern is identical to that of the easy case: all methods generally achieve coverage above the nominal 95% level, with the asymptotic bias correction and Firth-based intervals showing similarly high coverage. The log-F and

MLE intervals yield slightly lower but comparable coverage probabilities. However, unlike the easy scenario, the moderate case exhibits a notable drop in performance of asymptotic bias correction at $n = 10$, where the estimated coverage probability falls to approximately 90%.

5.3 Extreme Case

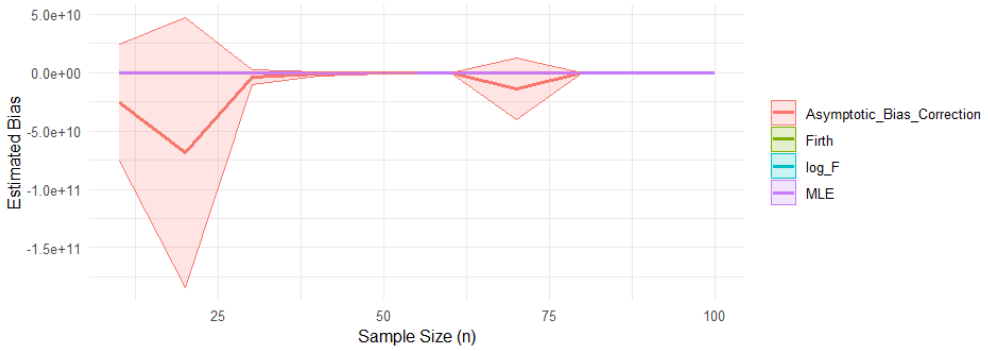

Figure 13: Estimated bias of $\hat{\beta}_1$ with 95% confidence intervals in extreme case ($\beta_0 = -3, \beta_1 = 1, m_1 = 1$).

Figure 13 presents the estimated bias of $\hat{\beta}_1$ under the extreme scenario. For the asymptotic bias correction method, the estimated bias appears to approach zero for sample sizes above $n = 30$. However, an unexpected spike in estimated bias occurs at $n = 70$, before it decreases again for larger sample sizes. Figure 14 suggests that this anomaly originates from the MLE itself, which also exhibits a sharp increase in estimated bias at $n = 70$, indicating that the asymptotic bias correction method inherits this instability. To better assess the performance of the remaining methods, Figure 15 excludes MLE and the asymptotic bias correction from the plot. This comparison reveals that Firth's method generally provides estimated bias closer to zero across most sample sizes. However, for very small sample sizes ($n = 10$), the log-F method outperforms Firth by exhibiting smaller estimated bias.

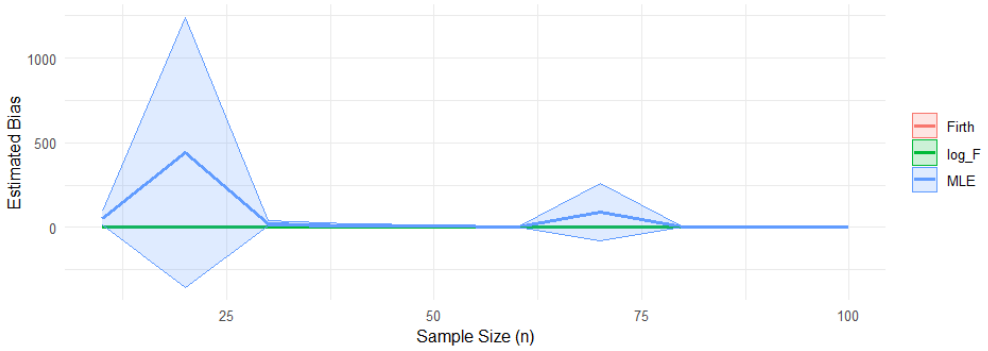

Figure 14: Estimated bias of $\hat{\beta}_1$ with 95% confidence intervals in extreme case ($\beta_0 = -3, \beta_1 = 1, m_1 = 1$) for all methods other than asymptotic bias correction.

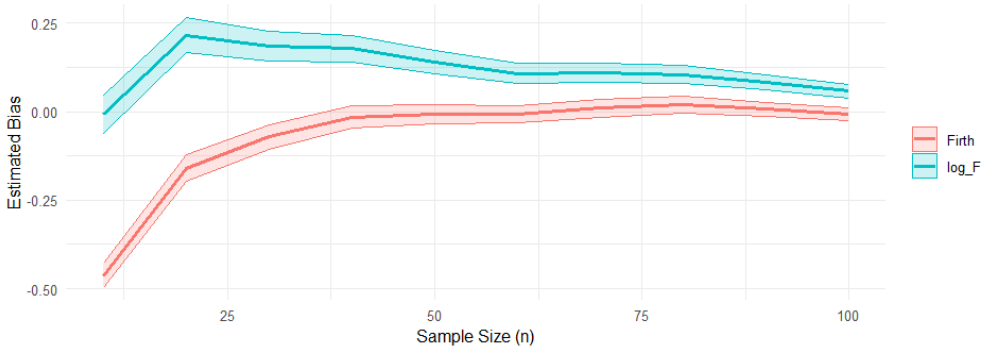

Figure 15: Estimated bias of $\hat{\beta}_1$ with 95% confidence intervals in extreme case ($\beta_0 = -3, \beta_1 = 1, m_1 = 1$) for log-F and Firth approach.

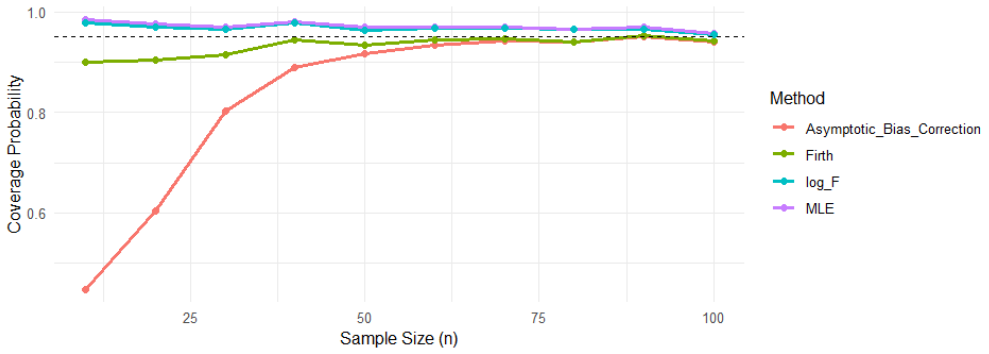

Figure 16: Estimated coverage probability of 95% Wald CI for β_1 in extreme case ($\beta_0 = -3, \beta_1 = 1, m_1 = 1$).

Figure 16 displays the estimated coverage probabilities of 95% Wald CIs based on different methods under the extreme scenario. The log-F approach, MLE, and Firth's method all maintain estimated coverage probabilities close to the nominal 95% level. Specifically, Firth's method tends to have slightly lower estimated coverage probability than 95% for smaller sample sizes ($n < 40$), while log-F and MLE consistently show slightly higher estimated coverage probability than 95% across all sample sizes. In contrast, the asymptotic bias correction method performs poorly for small samples, with estimated coverage probability dropping to as low as 50% at $n = 10$. Its coverage gradually improves with increasing sample size, approaching the nominal level around $n = 60$.

6 Conclusion

In this report, we examined three widely used methods aimed at reducing bias in parameter estimation within the framework of generalized linear models (GLMs): asymptotic bias correction, Firth's approach, and the log-F method. We evaluated their performance through a simulation study, comparing both the estimated bias of $\hat{\beta}_1$ and the estimated coverage probabilities of 95% Wald CIs across a range of scenarios.

Our simulation results show that asymptotic bias correction performs well in moderate-to-large sample sizes ($n \geq 30$), particularly when the true probabilities lie mostly between 0.2 and 0.8. However, in small samples, its performance deteriorates rapidly, with estimated bias increasing and estimated coverage probability falling substantially below the nominal level. Moreover, in extreme scenarios where the true probabilities are concentrated near 0 or 1, this method inherits the instability of the maximum likelihood estimator (MLE), sometimes exhibiting erratic spikes in estimated bias at specific sample sizes.

In contrast, the penalized likelihood methods—Firth's approach and the log-F method—demonstrate greater stability across different scenarios. Among these, Firth's method generally outperforms log-F when no prior information is available, specifically offering lower bias. However, when sample sizes are very small and strong prior knowledge is available, log-F method may be preferable.

Contributions

Task	**Contributor(s)**
Introduction	Samir
Asymptotic Bias Correction	Samir
Bias-Reduced Maximum Likelihood Estimation (Firth, 1993)	Ravleen
Penalization and Bias Reduction in Logistic Regression (Log–F Priors)	Annie
Simulation Study (section in the report)	Samir
Simulation Study (code)	Samir, Annie, and Ravleen
Conclusion	Samir, Annie, and Ravleen

References

[1] Gauss M. Cordeiro and Peter McCullagh. Bias correction in generalized linear models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 53(3):629–643, 1991.

[2] D. R. Cox and E. J. Snell. A general definition of residuals. *Journal of the Royal Statistical Society. Series B (Methodological)*, 30(2):248–275, 1968.

[3] Bradley Efron. Defining the curvature of a statistical problem (with applications to second order efficiency). *The Annals of Statistics*, 3(6):1189–1242, 1975.

[4] David Firth. Bias reduction of maximum likelihood estimates. *Biometrika*, 80(1):27–38, 1993.

[5] Sander Greenland and Mohammad Ali Mansournia. Penalization, bias reduction, and default priors in logistic and related categorical and survival regressions. *Statistics in Medicine*, 34(23):3133–3143, 2015.

[6] H. Jeffreys. An invariant form for the prior probability in estimation problems. *Proc. R. Soc. A*, 186:453–461, 1946.

[7] Ioannis Kosmidis. Bias in parametric estimation: reduction and useful side-effects. *WIREs Computational Statistics*, 6(3):185–196, 2014.

[8] Ioannis Kosmidis and David Firth. A generic algorithm for reducing bias in parametric estimation. *Electronic Journal of Statistics*, 4:1097–1112, 2010.

[9] Josmar Mazucheli. Expected/observed fisher information and bias-corrected maximum likelihood estimate(s). `CRAN`, 2022. R package version 1.0.0. Accessed 2025-04-21.

[10] P. McCullagh. *Tensor Methods in Statistics*. Chapman and Hall, London, 1987.

[11] D. A. Pierce. Discussion of paper by b. efron. *Ann. Statist.*, 3:1219–1221, 1975.

[12] M. H. Quenouille. Approximate tests of correlation in time-series. *J. R. Statist. Soc. B*, 11:68–84, 1949.

[13] M. H. Quenouille. Notes on bias in estimation. *Biometrika*, 43:353–360, 1956.

Appendix

Code for Simulation Study

```r
# Loading the required libraries
library(brglm2)
library(tidyverse)

# A function to simulate a dataset with one categorical variable with four categories
generate_data <- function(n, beta_0, beta_1){
  X1 <- rnorm(n, mean = 0, sd = 2)

  X <- cbind(1, X1)
  beta_true <- c(beta_0, beta_1)

  eta <- X %*% beta_true
  p <- 1 / (1 + exp(-eta))
  y <- rbinom(n, size = 1, prob = p)

  df <- data.frame(
    y  = y,
    X1 = X1
  )

  return(df)
}

# Function to check coverage for a model
check_coverage <- function(fit, true_betas) {
  ci <- confint.default(fit)  # Wald CI
  covers <- (true_betas >= ci[,1]) & (true_betas <= ci[,2])
  return(as.numeric(covers))
}

# A function to run simulation
run_simulation <- function(N=1000, n=100, beta_0=-1, beta_1=0.5,
                           m0 = 0, m1 = 0){
  # Initialize results data frames
  MLE_results <- data.frame(
    beta_0 = numeric(N), beta_1 = numeric(N),
    cover_beta_0 = numeric(N), cover_beta_1 = numeric(N)
  )

  MC_results <- data.frame(
    beta_0 = numeric(N), beta_1 = numeric(N),
    cover_beta_0 = numeric(N), cover_beta_1 = numeric(N)
  )

  Firth_results <- data.frame(
    beta_0 = numeric(N), beta_1 = numeric(N),
    cover_beta_0 = numeric(N), cover_beta_1 = numeric(N)
  )
```

```r
log_F_results <- data.frame(
  beta_0 = numeric(N), beta_1 = numeric(N),
  cover_beta_0 = numeric(N), cover_beta_1 = numeric(N)
)

for (i in 1:N){
  # Generate data
  df <- generate_data(n=n, beta_0=beta_0, beta_1=beta_1)

  # Fit models
  fit_MLE <- glm(y ~ X1, family=binomial(link="logit"), data=df)
  fit_MC <- glm(y ~ X1, family=binomial(link="logit"), data=df,
                method=brglmFit, type="correction")
  fit_Firth <- glm(y ~ X1, family=binomial(link="logit"), data=df,
                   method=brglmFit, type="AS_mean")

  #log-F
  df$intercept = 1
  df$weight = 1

  curr_coef = 0
  for (m in c(m0, m1)){
    df <- add_row(df,
                  y = 0.5,
                  intercept = ifelse(curr_coef==0,1,0),
                  X1 = ifelse(curr_coef==1,1,0),
                  weight = m)

    curr_coef = curr_coef + 1
  }

  fit_log_F <- glm(y ~ -1 + intercept + X1,
                   family=binomial(link="logit"),
                   data=df,
                   weight = weight)


  # Store estimates
  MLE_results[i,1:2] <- coef(fit_MLE)
  MC_results[i,1:2] <- coef(fit_MC)
  Firth_results[i,1:2] <- coef(fit_Firth)
  log_F_results[i,1:2] <- coef(fit_log_F)

  # Compute Wald CIs and check coverage
  true_betas <- c(beta_0, beta_1)

  MLE_results[i,3:4] <- check_coverage(fit_MLE, true_betas)
  MC_results[i,3:4] <- check_coverage(fit_MC, true_betas)
  Firth_results[i,3:4] <- check_coverage(fit_Firth, true_betas)
  log_F_results[i,3:4] <- check_coverage(fit_log_F, true_betas)
}

# Return results
```

```
  return(list(
    MLE = MLE_results,
    Asymptotic_Bias_Correction = MC_results,
    Firth = Firth_results,
    log_F = log_F_results
  ))
}

# A function to compute estimated bias, its CI, and coverage probability
show_results <- function(results, true_betas){

  results_MLE <- list(estimated_bias = vector(mode="numeric",
                      length=length(true_betas)),
                      bias_se = vector(mode="numeric",
                      length = length(true_betas)),
                      bias_CI = list(beta_0 = NA, beta_1 = NA),
                      coverage_prob = vector(mode="numeric",
                      length = length(true_betas)))
  results_Asymptotic_Bias_Correction <- list(estimated_bias = vector(mode="numeric",
                                    length=length(true_betas)),
                                     bias_se = vector(mode="numeric",
                                     length = length(true_betas)),
                                     bias_CI = list(beta_0 = NA, beta_1 = NA),
                                     coverage_prob = vector(mode="numeric",
                                     length = length(true_betas)))
  results_Firth <- list(estimated_bias = vector(mode="numeric",
                        length=length(true_betas)),
                        bias_se = vector(mode="numeric",
                        length = length(true_betas)),
                        bias_CI = list(beta_0 = NA, beta_1 = NA),
                        coverage_prob = vector(mode="numeric",
                        length = length(true_betas)))
  results_log_F <- list(estimated_bias = vector(mode="numeric",
                        length=length(true_betas)),
                        bias_se = vector(mode="numeric",
                        length = length(true_betas)),
                        bias_CI = list(beta_0 = NA, beta_1 = NA),
                        coverage_prob = vector(mode="numeric",
                        length = length(true_betas)))


  N = length(results$MLE[,1])


  # Computing estimated bias for each method
  for (i in 1:length(true_betas)){
    results_MLE$estimated_bias[i] = mean(results$MLE[,i] - true_betas[i])
    results_MLE$bias_se[i] = sd(results$MLE[,i] - true_betas[i])/sqrt(N)
    results_MLE$bias_CI[[i]] = c(results_MLE$estimated_bias[i] -
                                 1.96*results_MLE$bias_se[i],
                                  results_MLE$estimated_bias[i] +
                                  1.96*results_MLE$bias_se[i])
    results_MLE$coverage_prob[i] = sum(results$MLE[,2+i])/N
```

```
    results_Asymptotic_Bias_Correction$estimated_bias[i] =
                                            mean(results$Asymptotic_Bias_Correction[,i] -
                                            true_betas[i])
    results_Asymptotic_Bias_Correction$bias_se[i] =
    sd(results$Asymptotic_Bias_Correction[,i] -
    true_betas[i])/sqrt(N)
    results_Asymptotic_Bias_Correction$bias_CI[[i]] =
    c(results_Asymptotic_Bias_Correction$estimated_bias[i] -
    1.96*results_Asymptotic_Bias_Correction$bias_se[i],
                        results_Asymptotic_Bias_Correction$estimated_bias[i] +
                            1.96*results_Asymptotic_Bias_Correction$bias_se[i])
    results_Asymptotic_Bias_Correction$coverage_prob[i] =
    sum(results$Asymptotic_Bias_Correction[,2+i])/N

    results_Firth$estimated_bias[i] = mean(results$Firth[,i] - true_betas[i])
    results_Firth$bias_se[i] = sd(results$Firth[,i] - true_betas[i])/sqrt(N)
    results_Firth$bias_CI[[i]] = c(results_Firth$estimated_bias[i] -
    1.96*results_Firth$bias_se[i],
                                   results_Firth$estimated_bias[i] +
                                   1.96*results_Firth$bias_se[i])
    results_Firth$coverage_prob[i] = sum(results$Firth[,2+i])/N

    results_log_F$estimated_bias[i] = mean(results$log_F[,i] - true_betas[i])
    results_log_F$bias_se[i] = sd(results$log_F[,i] - true_betas[i])/sqrt(N)
    results_log_F$bias_CI[[i]] = c(results_log_F$estimated_bias[i] -
    1.96*results_log_F$bias_se[i],
                                   results_log_F$estimated_bias[i] +
                                   1.96*results_log_F$bias_se[i])
    results_log_F$coverage_prob[i] = sum(results$log_F[,2+i])/N
  }


  return(list(
    MLE= results_MLE,
    Asymptotic_Bias_Correction = results_Asymptotic_Bias_Correction,
    Firth = results_Firth,
    log_F = results_log_F
  ))
}


# A function to run experiments
run_exp <- function(beta_0 = 1, beta_1, m1, n_min, n_max){
  n = seq(from=n_min, to=n_max, by=10)
  true_betas = c(beta_0, beta_1)
  output = list()
  for (i in 1:length(n)){
    cat("n = ", n[i], "\n")
    results = run_simulation(N=1000, n=n[i], beta_0=beta_0, beta_1=beta_1,
                             m0 = 0, m1 = m1)
    output[[i]] = show_results(results, true_betas)
  }
```

```r
  return(output)
}


# Running experiments
data <- run_exp(beta_0 = -1, beta_1 = 0.5, m1 = 1, n_min = 20, n_max = 100)

# Extract results for beta_1
extract_results <- function(data, n_values) {
  methods <- c("MLE", "Asymptotic_Bias_Correction", "Firth", "log_F")
  results_list <- list()

  for (i in seq_along(n_values)) {
    for (method in methods) {
      results_list <- append(results_list, list(
        data.frame(
          n = n_values[i],
          method = method,
          bias = data[[i]][[method]]$estimated_bias[2],
          lower = data[[i]][[method]]$bias_CI[[2]][1],
          upper = data[[i]][[method]]$bias_CI[[2]][2]
        )
      ))
    }
  }

  return(do.call(rbind, results_list))
}


# Generate the data for plotting
n_values <- seq(from = 20, to = 100, by = 10)
plot_data <- extract_results(data, n_values)
plot_data_log <- plot_data %>%
  mutate(log_bias = sign(bias) * log1p(abs(bias)),
         log_lower = sign(lower) * log1p(abs(lower)),
         log_upper = sign(upper) * log1p(abs(upper)))

# plot with confidence bands
ggplot(plot_data, aes(x = n, y = bias, color = method, fill = method)) +
  geom_line(size = 1) +  # Line for estimated bias
  geom_ribbon(aes(ymin = lower, ymax = upper), alpha = 0.2) +
  labs(
    title = expression("Estimated bias of " * hat(beta)[1] *
                         "Confidence Bands with n >= 30, " * beta[0] *
                         " = -1, " * beta[1] * " = 0.5 and m1 = 1"),
    x = "Sample Size (n)",
    y = "Estimated Bias"
  ) +
  theme_minimal() +
  theme(legend.title = element_blank())
```

```r
# Plot with confidence bands (log scale)
ggplot(plot_data_log, aes(x = n, y = log_bias, color = method, fill = method)) +
  geom_line(size = 1) +  # Line for estimated bias
  geom_ribbon(aes(ymin = log_lower, ymax = log_upper), alpha = 0.2) +
  labs(
    title = expression("Log Transformed Estimated Bias of " * hat(beta)[1] *
                         " with Confidence Bands with n >= 30, " * beta[0] *
                         " = -1, " * beta[1] *
                         " = 0.5, and m1 = 1"),
    x = "Sample Size (n)",
    y = "Log Transformed Estimated Bias"
  ) +
  theme_minimal() +
  theme(legend.title = element_blank())


# Function to extract coverage probability for beta_1
extract_coverage <- function(data, n_values) {
  methods <- c("MLE", "Asymptotic_Bias_Correction", "Firth", "log_F")
  results_list <- list()

  for (i in seq_along(n_values)) {
    for (method in methods) {
      results_list <- append(results_list, list(
        data.frame(
          n = n_values[i],
          method = method,
          coverage = data[[i]][[method]]$coverage_prob[2]
        )
      ))
    }
  }

  return(do.call(rbind, results_list))
}

# Generate the data for plotting
plot_data <- extract_coverage(data, n_values)

# Plot coverage probability for beta_1
ggplot(plot_data, aes(x = n, y = coverage, color = method)) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  labs(
    title = expression("Coverage Probability of " * hat(beta)[1] *
                         " Across Sample Sizes when " * beta[0] *
                         " = -1, " * beta[1] *
                         " = 0.5 and m1 = 1"),
    x = "Sample Size (n)",
    y = "Coverage Probability",
    color = "Method"
  ) +
  geom_hline(yintercept = 0.95, linetype = "dashed", color = "black") +
```

```
  theme_minimal()
```