Supplementary material document for Jeffreys-prior penalty, finiteness and shrinkage in binomial-response generalized linear models

Ioannis Kosmidis
ioannis.kosmidis@warwick.ac.uk
and
David Firth
d.firth@warwick.ac.uk

Department of Statistics, University of Warwick
Coventry CV4 7AL, UK
and
The Alan Turing Institute
British Library, London NW1 2DB, UK

March 23, 2020

S1 Supplementary material

All labels for the sections, equations, the table, the algorithm and the figure in the current document have been prefixed by "S" (e.g. Section S2, Table S1, Algorithm S1, etc).

The supplementary material for "Jeffreys-prior penalty, finiteness and shrinkage in binomial-response generalized linear models" provides i) the proofs of Theorem 1, Corollary 1, Theorem 2, Theorem 3 in the main text (see Section S2); ii) the pseudo-code for Algorithm `JeffreysMPL` (see Algorithm S1), which implements the repeated maximum-likelihood fits procedure of Section 4 in the main text to maximize the penalized log-likelihood $l^{\dagger}(\beta; a)$ in (4) for any supplied a and link function $G(\eta)$; iii) illustrations using an R implementation of Algorithm S1; and iv) R scripts to reproduce all numerical results and figures in the main text and the current document. The current supplementary material document and the R scripts are available for download at `http://www.ikosmidis.com/files/finiteness-jeffreys-supplementary-v1.3.zip`.

In particular, the script `nba-1415-case-study.R` reproduces the numerical results in Example 1, Figure 1, and Figure 3 in the manuscript, and in Table S1 in the current document; the files `nba-1415-functions.R` and `nba-1415-regular-season.csv` provide the R functions and the data, respectively, for the case study in the main text; the script `jeffreys-shrinkage.R` reproduces Figure 2 in the main text; the file `jeffreys-MPL.R` provides an R implementation of Algorithm `JeffreysMPL`, and the file `sur-candes-2019.R` computes the timings for Algorithm `JeffreysMPL` that are reported in Section S3.3, and reproduces Figure 2b on page 11 of the supplementary information appendix of Sur and Candès (2019) (Figure S1 here).

All code has been tested and run in R version 3.6.3 (R Core Team, 2020) using the R packages `brglm2` version 0.6.2 (Kosmidis, 2020a), `enrichwith` version 0.3.1 (Kosmidis, 2020b), `ggplot2` version 3.2.1 (Wickham, 2016), `qvcalc` version 1.0.2 (Firth, 2020), `rbenchmark` version 1.0.0 (Kusnierczyk, 2012).

S2 Proofs of Theorem 1, Corollary 1, Theorem 2, Theorem 3

S2.1 Proof of Theorem 1

Since X has full rank, $|X^\top W(\beta)X|$ is not trivially zero for all $\beta \in \Re^p$. Let $R = \{1, \ldots, n\}$ and

$$R_{(1)} = \{i : |\eta_i(\beta(r))| \to \infty \text{ as } r \to \infty; i \in R\}$$
$$R_{(2)} = \{i : \eta_i(\beta(r)) \to c_i, |c_i| < \infty \text{ as } r \to \infty; i \in R\} .$$

Then $R = R_{(1)} \cup R_{(2)}$ and $R_{(1)} \cap R_{(2)} = \emptyset$, where $\emptyset$ is the empty set.

We first consider the case where $R_{(1)}$ and $R_{(2)}$ are non-empty. Then, X and $W(\beta)$ can be partitioned as

$$X = \begin{bmatrix} X_{(1)} \\ X_{(2)} \end{bmatrix} \quad \text{and} \quad W(\beta) = \begin{bmatrix} W_{(1)}(\beta) & 0 \\ 0 & W_{(2)}(\beta) \end{bmatrix},$$

where $X_{(l)}$ has rows x_i with $i \in R_{(l)}$ and $W_{(l)}(\beta) = \text{diag}\left\{m_i\omega(\eta_i(\beta)), i \in R_{(l)}\right\}$ $(l = 1, 2)$. We can then write

$$X^\top W(\beta(r))X = X_{(1)}^\top W_{(1)}(\beta(r))X_{(1)} + X_{(2)}^\top W_{(2)}(\beta(r))X_{(2)} . \tag{S1}$$

The limit of the first term in the right hand side of (S1), as $r \to \infty$, is zero because the limit of $\omega(\eta) = e^\eta/(1+e^\eta)^2$, as η grows to infinity in absolute value, is zero. For the second term, $X_{(2)}$ is such that $X_{(2)}\beta(r) \to c$ as $r \to \infty$ where c has all of its components finite. There must exist vectors $a \in \Re^p$ and $b \in \Re^p$ with finite components such that $a + br \to \beta_0$. So, $X_{(2)}a + X_{(2)}br \to c$ as $r \to \infty$ which is possible if and only if $X_{(2)}b = 0$. Hence $X_{(2)}$ has rank smaller than p and $|X_{(2)}^\top W_{(2)}(\beta(r))X_{(2)}| = 0$ for all r. The result follows because $|X_{(2)}^\top W_{(2)}(\beta(r))X_{(2)}| = 0$ for all r and $|X_{(1)}^\top W_{(1)}(\beta(r))X_{(1)}| \to 0$ as $r \to \infty$.

Because X is of full rank and $X_{(2)}$ has rank smaller than p, $R_{(1)}$ cannot be empty. Hence, we only need to also examine the case that $R_{(2)}$ is empty and $R_{(1)}$ is not. In this case the same arguments as above give $|X^\top W(\beta(r))X| = |X_{(1)}^\top W_{(1)}(\beta(r))X_{(1)}| \to 0$ as $r \to \infty$.

S2.2 Proof of Corollary 1

The binomial log-likelihood $l(\beta)$ in (2) is bounded above by zero. Hence, according to Theorem 1 and expression (3), $\tilde{l}(\beta(r)) \to -\infty$ as $\beta(r) \to \beta_0$. Such a setting for β is always dominated, by a choice b with finite components for which $\tilde{l}(b)$ takes a finite value. Hence, the maximizer of $\tilde{l}(\beta)$ must have finite components.

S2.3 Proof of Theorem 2

The sum of m independent Bernoulli distributions with probability π is binomial with index m and probability π. For this reason and without any loss of generality, the proof proceeds with $m_i = 1$ so that $w_i(\beta) = \omega(\eta_i(\beta))$ $(i = 1, \ldots, n)$.

For proving i) decompose X as $X = QR$, where Q is a $n \times p$ matrix with orthonormal columns ($Q^\top Q = I_p$ where I_p is the $p \times p$ identity matrix) and R is a $p \times p$ non-singular matrix. This decomposition is always possible because X has full rank by assumption. Then $|X^\top W(\beta)X| = |Q^\top W(\beta)Q||R|^2$. The functions $|X^\top W(\beta)X|$ and $|Q^\top W(\beta)Q|$ will have stationary points of the same kind and at the same values of β, because $|R|^2 > 0$ does not depend on β.

Denote the ordered set of quadratic weights as $w_{(1)}(\beta), \ldots, w_{(n)}(\beta)$ with $w_{(1)}(\beta) \leq \ldots \leq w_{(n)}(\beta)$. The Poincaré separation theorem (see, for example Magnus and Neudecker, 1999, Chapter 11, Theorem 10 for statement) and the positive definiteness of $W(\beta)$ for $\beta \in \Re^p$ imply that

$$\prod_{t=1}^{p} w_{(t)}(\beta) \leq |Q^\top W(\beta)Q| \leq \prod_{t=1}^{p} w_{(n-p+t)}(\beta) . \tag{S2}$$

Note that $0 \leq \omega(\eta) \leq 1/4$, *with the upper bound achieved when* $\eta = 0$. *It follows that* $\prod_{t=1}^{p} w_{(t)}(\beta) \leq 1/4^p$ *and* $\prod_{t=1}^{p} w_{(n-t+t)}(\beta) \leq 1/4^p$ *and that, at* $\beta = 0$ *inequalities (S2) become* $1/4^p \leq |Q^\top W(0) Q| \leq 1/4^p$. *The proof of i) concludes by noting that* $|Q^\top W(0) Q| = 1/4^p$ *which is the maximum value that* $|Q^\top W(\beta) Q|$ *can take.*

For proving ii), note that $\bar{\omega}(z) = z(1-z)$ *is concave. Hence, for* $\theta \in (0,1)$, $\theta^* = 1 - \theta$ *and any pair of* n*-vectors of probabilities* π *and* ρ $\bar{\omega}(\theta\pi_i + \theta^*\rho_i) \geq \bar{\omega}(\theta\pi_i) + \bar{\omega}(\theta^*\rho_i)$ $(i = 1, \ldots, n)$. *Lemma 1 can then be used to show that*

$$\left| X^\top \bar{W}(\theta\pi + \theta^*\rho) X \right| \geq \left| \theta X^\top \bar{W}(\pi) X + \theta^* X^\top \bar{W}(\rho) X \right| ,$$

The result in ii) follows from Magnus and Neudecker (1999, Chapter 11,Theorem 25) which can be used to show that

$$\left| \theta X^\top \bar{W}(\pi) X + \theta^* X^\top \bar{W}(\rho) X \right| \geq \left| X^\top \bar{W}(\pi) X \right|^{\theta} \left| X^\top \bar{W}(\rho) X \right|^{\theta^*} .$$

Lemma 1. *If* A *and* B *are both diagonal* $n \times n$ *matrices with non-negative diagonal elements* $\{a_r\}$ *and* $\{b_r\}$, *respectively , and* $a_r \geq b_r$, *for every* $r \in \{1, \ldots, n\}$, *then, if* X *is a* $n \times p$ *matrix,* $|X^\top A X| \geq |X^\top B X|$.

Proof. Since $A \geq B$, elementwise, $A = B + C$, where C is a diagonal matrix with non-negative entries. Furthermore, $X^\top A X$, $X^\top B X$ and $X^\top C X$ are positive semidefinite, by the non-negativity of the diagonal elements of A, B and C, respectively. Hence, by Magnus and Neudecker (1999, Chapter 11, Theorem 9), $\lambda_t\left(X^\top A X\right) \geq \lambda_t\left(X^\top B X\right)$ $(t = 1, \ldots, p)$, where $\lambda_t(D)$ denotes the tth eigenvalue of the matrix D. Since the determinant of a matrix is the product of its eigenvalues the result follows. □

S2.4 Proof of Theorem 3

For c *as in Theorem 3, the adjusted responses and totals in (6) have the form*

$$\tilde{y} = y + 2ah\pi \left\{ 1 + (q - 1/2) \frac{1 - I(q \leq 1/2)}{\pi(1-\pi)} \right\} , \tag{S3}$$
$$\tilde{m} = m + 2ah \left\{ 1 + (q - 1/2) \frac{\pi - I(q \leq 1/2)}{\pi(1-\pi)} \right\} .$$

The result follows for any value of q, *because* $0 \leq h \leq 1$ *and* $0 \leq \pi \leq 1$.

S3 Algorithm `JeffreysMPL`

S3.1 Details

Algorithm `JeffreysMPL` (see Algorithm S1) implements the repeated maximum-likelihood fits procedure of Section 4 in the main text, to maximize the penalized log-likelihood $l^\dagger(\beta; a)$ in (4) for any supplied a and link function $G(\eta)$.

A satisfactory starting value b for `JeffreysMPL` is the maximum likelihood estimate of β, after adding a small positive constant and twice that constant to the actual binomial responses and totals, respectively.

Step 22 of `JeffreysMPL` can be carried out using readily available maximum-likelihood implementations for binomial-response generalized linear models, such as the `glm` function in R (R Core Team, 2020) and the various implementations in the Python modules `statsmodels` (Seabold and Perktold, 2010) and `scikit-learn` (Pedregosa et al., 2011).

Algorithm S1 Repeated maximum-likelihood fits for the maximization of $l^{\dagger}(\beta; a)$ in expression (4). The inputs `y`, `m`, `X`, `a`, `G`, `g`, `gdash` are $y = (y_1, \ldots, y_n)^{\top}$, $m = (m_1, \ldots, m_n)^{\top}$, X, a in expression (4) of the main text, $G(\eta)$, $g(\eta)$ and $g'(\eta)$, respectively, and $||.||$ is the L_2 norm. The starting vector for β is b and ϵ is a small positive constant. `ML` is any maximum likelihood procedure.

1: **procedure** JEFFREYSMPL(`y`, `m`, `X`, `a`, `G`, `g`, `gdash`, `ML`, `b`, ϵ)
2: `k` $\leftarrow$ `0`
3: `n` $\leftarrow$ `numberofrows(X)` ▷ Number of observations; must equal `length(y)` and `length(m)`
4: `p` $\leftarrow$ `numberofcolumns(X)` ▷ Number of parameters; must equal `length(b)`
5: `eta, pi, d, dd` $\leftarrow$ `vector(n)`
6: **for** `i` $\in \{1, 2, \ldots, n\}$ **do**
7: `xi` $\leftarrow$ `(X[i, 1], ..., X[i, p])`
8: `eta[i]` $\leftarrow$ `dotproduct(xi, b)` ▷ dot product of `xi` and `b`
9: `pi[i]` $\leftarrow$ `G(eta[i])`
10: `d[i]` $\leftarrow$ `g(eta[i])`
11: `dd[i]` $\leftarrow$ `g'(eta[i])`
12: **end for**
13: `w` $\leftarrow$ `d * d/pi/(1 - pi)` ▷ elementwise operations
14: `q` $\leftarrow$ `dd/w + pi` ▷ elementwise operations
15: `j` $\leftarrow$ `I(q` $\leq$ `1/2)` ▷ elementwise inequality and indicator function
16: `V` $\leftarrow$ diag$(\sqrt{\texttt{m[1]} * \texttt{w[1]}}, \ldots, \sqrt{\texttt{m[n]} * \texttt{w[n]}}) \cdot$ `X` ▷ $\cdot$ stands for matrix product
17: `Q` $\leftarrow$ orthogonal matrix from the QR decomposition of `V`
18: `h` $\leftarrow$ `rowsums(Q * Q)` ▷ sum the elements in each row of the elementwise product `Q * Q`
19: `y_adj` $\leftarrow$ `y + 2 * a * h * pi * (1 + (q - 1/2) * (1 - j)/pi/(1 - pi))` ▷ elementwise operations
20: `m_adj` $\leftarrow$ `m + 2 * a * h * (1 + (q - 1/2) * (pi - j)/pi/(1 - pi))` ▷ elementwise operations
21: `bp` $\leftarrow$ `b`
22: `b` $\leftarrow$ `ML(y_adj, m_adj, X, G)` ▷ Maximum likelihood fit on adjusted data with link `G`
23: **if** $||$`bp - b`$|| < \epsilon$ **then**
24: **return** b
25: **else**
26: `k` $\leftarrow$ `k + 1`
27: Go to 6
28: **end if**
29: **end procedure**

The estimated variance-covariance matrix of the penalized likelihood estimator can be obtained as $(R^{\top}R)^{-1}$, where R is the upper triangular matrix from the QR decomposition of $W(\beta)^{1/2}X$ at the final iteration of the procedure. That decomposition is a by-product of step 17 of `JeffreysMPL`.

S3.2 Illustration: evolution of adjusted responses and totals

Table S1 shows the values of the adjusted responses and totals for the first 6 games of Philadelphia 76ers in Example 1 of the main text, at the first 6 iterations of Algorithm `JeffreysMPL`, when computing the reduced-bias fit shown in Figure 1 (see script `nba-1415-case-study.R` for code to reproduce Table S1). The starting values (iteration 0) are the maximum likelihood estimates of the ability contrasts after adding 0.01 and 0.02 to the actual responses and totals, respectively.

Table S1: The adjusted responses (top) and totals (bottom) for the first 6 games of Philadelphia 76ers (P76) at the first 6 iterations of Algorithm S1, when computing the reduced-bias fit in Figure 1. Figures are shown in 3 decimal digits. The home team is mentioned first in column names. The actual response is 1 if the home team wins and 0 otherwise. The acronyms for the opponents are IP (Indiana Pacers), MB (Milwaukee Bucks), MH (Miami Heat), HR (Houston Rockets), OR (Orlando Magic) and CB (Chicago Bulls). The starting values are the maximum likelihood estimates of the ability contrasts after adding 0.01 and 0.02 to the actual responses and totals (iteration 0).

Iteration	P76 vs IP	P76 vs MB	MH vs P76	HR vs P76	OM vs P76	CB cs P76
Adjusted responses						
0	0.010	0.010	1.010	1.010	1.010	1.010
1	0.039	0.036	1.045	1.012	1.104	1.039
2	0.045	0.042	1.054	1.017	1.110	1.048
3	0.046	0.043	1.055	1.017	1.111	1.049
4	0.046	0.043	1.055	1.018	1.111	1.049
5	0.046	0.043	1.055	1.018	1.111	1.049
6	0.046	0.043	1.055	1.018	1.111	1.049
Adjusted totals						
0	1.020	1.020	1.020	1.020	1.020	1.020
1	1.114	1.105	1.067	1.018	1.158	1.059
2	1.128	1.120	1.081	1.025	1.170	1.073
3	1.130	1.122	1.083	1.026	1.171	1.075
4	1.131	1.122	1.084	1.026	1.172	1.075
5	1.131	1.122	1.084	1.026	1.172	1.075
6	1.131	1.122	1.084	1.026	1.172	1.075

S3.3 Illustration: Algorithm `JeffreysMPL` for the high-dimensional logistic regression setting in Sur and Candès (2019)

The R implementation of `JeffreysMPL` in the supplementary material (see script `Jeffreys-MPL.R` for code) is used here to compute the reduced-bias estimates for a logistic regression model with $n = 1000$ binary responses and $p = 200$ covariates, as considered in Figure 2(b) of the supplementary information appendix of Sur and Candès (2019) (see script `sur-candes-2019.R` for reproducible code).

In particular, we construct a 1000×200 model matrix X by simulating 200 000 independent random variables from a normal distribution with mean 0 and variance 10^{-3}. Then, we simulate 1000 Bernoulli random variables from a logistic regression model with linear predictors $X\beta$, where $\beta_1 = \ldots = \beta_{25} = 10$, $\beta_{26} = \ldots = \beta_{50} = -10$, and $\beta_{50} = \ldots = \beta_{200} = 0$.

The R implementation of `JeffreysMPL` relies on a full maximum-likelihood iteration for step 22 using the R function `glm.fit`, and it takes approximately 2.73 seconds to converge to the reduced-bias estimates of the 200 parameters in 4 decimal places on a MacBook Pro laptop with 3.5GHz processor and 16GB of memory. The resulting maximum penalized likelihood estimates are shown in Figure S1 along with the corresponding maximum likelihood estimates.

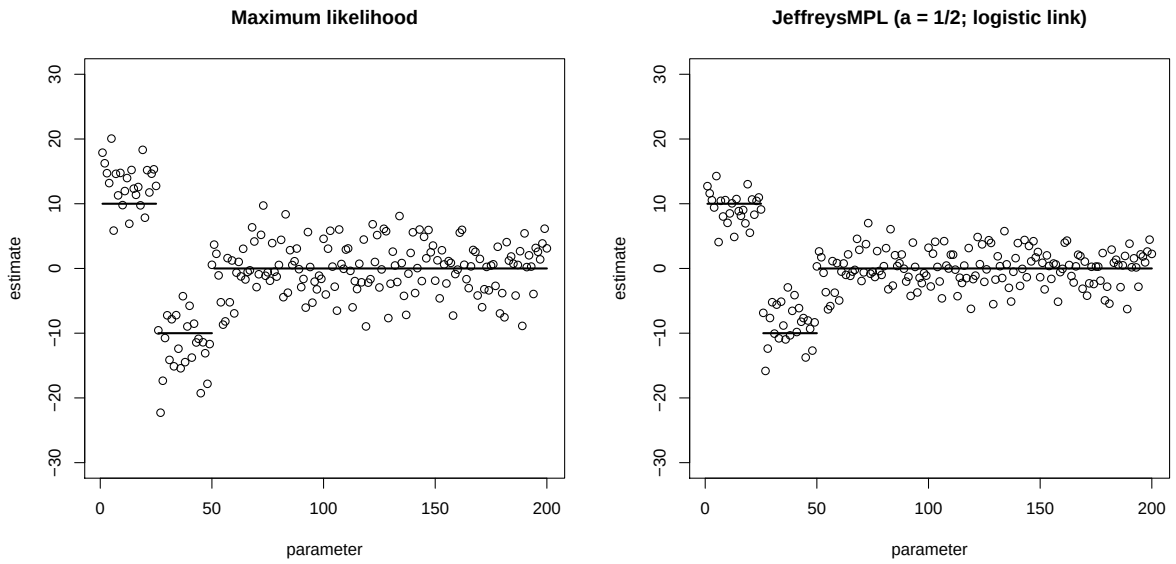

Figure S1: Maximum likelihood and reduced-bias estimates for the parameters of a logistic regression model on data simulated as in the supplementary information appendix of Sur and Candès (2019). In particular, we construct a 1000×200 model matrix X by simulating 200 000 independent random variables from a normal distribution with mean 0 and variance 10^{-3}. Then, we simulate 1000 Bernoulli random variables from a logistic regression model with linear predictors $X\beta$, where $\beta_1 = \ldots = \beta_{25} = 10$, $\beta_{26} = \ldots = \beta_{50} = -10$, and $\beta_{50} = \ldots = \beta_{200} = 0$. The maximum likelihood estimates are computed using the `glm` R function and the reduced-bias estimates are computed using Algorithm `JeffreysMPL`. The horizontal segments indicate the parameter values used for simulating the data.

References

Firth, D. (2020). *qvcalc: Quasi Variances for Factor Effects in Statistical Models.* R package version 1.0.2.

Kosmidis, I. (2020a). *brglm2: Bias Reduction in Generalized Linear Models.* R package version 0.6.2.

Kosmidis, I. (2020b). *enrichwith: Methods to enrich list-like R objects with extra components.* R package version 0.3.1.

Kusnierczyk, W. (2012). *rbenchmark: Benchmarking routine for R.* R package version 1.0.0.

Magnus, J. R. and H. Neudecker (1999). *Matrix Differential Calculus with Applications in Statistics and Econometrics.* Chichester: Wiley.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12*, 2825–2830.

R Core Team (2020). *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing.

Seabold, S. and J. Perktold (2010). Statsmodels: Econometric and statistical modeling with Python. In S. van der Walt and J. Millman (Eds.), *Proceedings of the 9th Python in Science Conference*, pp. 57–61.

Sur, P. and E. J. Candès (2019). A modern maximum-likelihood theory for high-dimensional logistic regression. *Proceedings of the National Academy of Sciences 116*(29), 14516–14525.

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.