melt: Multiple Empirical Likelihood Tests in R

Eunseop Kim
The Ohio State University

Steven N. MacEachern
The Ohio State University

Mario Peruggia
The Ohio State University

Abstract

The R package **melt** provides a unified framework for data analysis with empirical likelihood methods. ...

Keywords: empirical likelihood, generalized linear models, multiple testing, optimization, R.

1. Introduction

Likelihood is an essential component of statistical inference. In a nonparametric or semiparametric setting, where the quantity of interest is finite-dimensional, the maximum likelihood approach is not applicable since the underlying data-generating distribution is left unspecified. A popular approach in this context is the method of moments or the two-step generalized method of moments (GMM) (Hansen 1982) where only partial information is specified by moment conditions. Various one-step alternatives to GMM have been proposed over the last decades in statistics and econometrics literature (see, e.g., Efron 1981; Imbens 1997; Newey and Smith 2004).

One such alternative is empirical likelihood (EL) (Owen 1988, 1990; Qin and Lawless 1994). EL defines a likelihood function by profiling a nonparametric likelihood subject to the moment restrictions. While it is nonparametric in nature, some desirable properties of parametric likelihood apply to EL. Most notably, the EL ratio functions have limiting chi-square distributions under certain conditions. Without explicit studentization, confidence regions for the parameters can be constructed in much the same way as using parametric likelihood. As the name suggests, however, the empirical distribution of the data determines the shape of the confidence regions. Also, coverage accuracy of the confidence regions can further be improved in principle since EL is Bartlett-correctable (DiCiccio, Hall, and Romano 1991). In terms of estimation, the standard expansion argument (e.g., Yuan and Jennrich 1998; Jacod and Sørensen 2018) establishes the consistency and asymptotic normality of the maximum empirical likelihood estimator (MELE). Moreover, Newey and Smith (2004) showed that the MELE generally has a smaller bias than its competitors and achieves higher-order efficiency after bias correction. EL methods have been extended to other areas, including linear models (Owen 1991), generalized linear models (Kolaczyk 1994; Chen and Cui 2003), survival analysis (Li, Li, and Zhou 2005), time series models (Kitamura 1997; Nordman and Lahiri 2014), and high-dimensional data (Chen, Peng, and Qin 2009; Hjort, McKeague, and van Keilegom 2009). For an overview of EL and its applications, see Owen (2001) and Chen and van Keilegom (2009).

In R language (R Core Team 2022), some software packages implementing EL and its related methods are available from the Comprehensive R Archive Network (CRAN). The **emplik** package (Zhou 2020) provides a wide range of functions for analyzing censored and truncated data with EL. Confidence intervals for a one-dimensional parameter can also be constructed. Other examples and applications of the package can be found in Zhou (2015). The **emplik2** package (Barton 2022) is an extension for two samples. Both packages cover the methods for the mean with uncensored data, which is the simplest case in terms of computation. In addition, the **EL** package (Valeinis and Cers 2022) performs EL tests for the difference between two sample means and the difference between smoothed Huber estimators. The **eel** package (Wu and Zhang 2015) implements the extended empirical likelihood method (Tsao and Wu 2013) that expands the domain of EL to the full parameter space by applying a similarity transformation. It escapes the so-called "convex hull constraint" of EL that confines the domain to a bounded region. In fact, the gradient of log EL ratio functions diverges at the boundary. Using this property, the **elhmc** package (Kien, Chaudhuri, and Wei 2017) contains a single function `ELHMC` for Hamiltonian Monte Carlo sampling in Bayesian EL computation (Chaudhuri, Mondal, and Yin 2017). The **ELCIC** package (Shen and Wang 2022) develops an EL-based consistent information criterion in a model selection framework. The methods are relevant to longitudinal data. In a broader context of GMM and generalized empirical likelihood (Smith 1997), a few packages can be used for estimation with EL. The **gmm** package (Chaussé 2010) provides flexibility in specifying moment conditions. Other than GMM and EL, continuous updating (Hansen, Heaton, and Yaron 1996) and several estimation methods that belong to the family of generalized empirical likelihood are available. It has been superseded by the **momentfit** package (Chaussé 2020), which adds exponential tilting (Kitamura and Stutzer 1997) estimation and methods for constructing two-dimensional confidence regions.

In this paper, we present the R package **melt** (Kim 2022) that performs multiple empirical likelihood tests for regression analysis. The primary focus of the package is on linear and generalized linear models, perhaps most commonly used with `lm()` and `glm()` functions in R. The package only considers just-identified models where the number of moment conditions equals the number of parameters. Typical linear models specified by `formula` objects in R are just-identified. In this case, the MELE is identical to the maximum likelihood estimator, and the estimate is easily obtained using `lm.fit()` or `glm.fit()` in the **stats** package. Then the fitted model serves as a basis for testing hypotheses, which is a core component of the package. Standard tests performed by `summary.lm()` and `summary.glm()` methods are available, such as significance tests of the coefficients and overall F test or chi-square test. In line with `linearHypothesis()` in the **car** package (Fox and Weisberg 2019) or `glht()` in the **multcomp** package (Hothorn, Bretz, and Westfall 2008), the user can specify linear hypotheses to be tested. Multiple testing procedures are provided as well to control the family-wise error rate. Constructing confidence regions and detecting outliers on a fitted model can also be done, adding more options for data analysis.

Note that all the tests and methods are based on EL and its asymptotic properties. Although conceptually advantageous over parametric methods, this could lead to poor finite sample performance. Therefore, several calibration techniques are implemented in **melt** to mitigate the drawback of EL. Another feature that distinguishes the package from others is the absence of standard errors and `vcov()` methods due to the implicit studentization. Apart from computational difficulties, this fundamental difference makes it challenging for EL methods to be directly extended to other existing packages for parametric models. We aim to bridge the gap and provide an easy-to-use interface that enables applying the methods to tasks routinely made in R.

The rest of the paper is organized as follows. Section 2 describes EL methods and computational aspects of testing hypotheses with EL. Section 3 provides an overview of the **melt** package. Section 4 shows the basic usage of **melt** with implementation details. Examples are included to illustrate the applications of the package. We conclude with a summary and directions for future development in Section 5.

2. Background

2.1. Empirical likelihood framework

We describe a general framework for EL formulation. Suppose we observe independent and identically distributed (i.i.d.) p-dimensional random variables $X_1, \ldots, X_n$ from a distribution P. Consider a parameter $\theta \equiv \theta(P) \in \Theta$ and an estimating function $g(X_i, \theta)$ that take their values in $\mathbb{R}^p$ and satisfy the following moment condition:

$$\mathsf{E}[g(X_i, \theta)] = 0, \tag{1}$$

where the expectation is taken with respect to P. Without further information on P, we restrict our attention to a family of multinomial distributions supported on the data. The nonparametric likelihood is given by

$$L(P) = \prod_{i=1}^{n} P(\{X_i\}) = \prod_{i=1}^{n} p_i,$$

and the empirical distribution P_n maximizes L with $L(P_n) = n^{-n}$. Then the (profile) EL ratio function is defined as

$$R(\theta) = \max_{p_i} \left\{ \prod_{i=1}^{n} np_i : \sum_{i=1}^{n} p_i g(X_i, \theta) = 0,\ p_i \geq 0,\ \sum_{i=1}^{n} p_i = 1 \right\}, \tag{2}$$

with $L(\theta) = \prod_{i=1}^{n} p_i$ denoting the corresponding EL function. The profiling removes all the nuisance parameters, the p_is attached to the data, yielding a p-dimensional subfamily indexed by θ. Note that the data determine the multinomial distributions; thus, the reduction to a subfamily does not correspond to a parametric model. See DiCiccio and Romano (1990) for a detailed discussion of its connection to the notion of least favorable families (Stein 1956).

We maximize $\prod_{i=1}^{n} np_i$, or equivalently $\sum_{i=1}^{n} \log(np_i)$, subject to the constraints in Equation 2. The convex hull constraint refers to the condition that the convex hull of the points $\{g(X_i, \theta)\}_{i=1}^{n}$ contains the zero vector. If the constraint is not satisfied, the problem is infeasible as some p_is are forced to be negative. Otherwise, the problem admits a unique interior solution since the objective function is concave and the feasible set is convex. Using the method of Lagrange multipliers, we write

$$\mathcal{L}(p_1, \ldots, p_n, \lambda, \gamma) = \sum_{i=1}^{n} \log(np_i) - n\lambda^\top \sum_{i=1}^{n} p_i g(X_i, \theta) + \gamma \left(\sum_{i=1}^{n} p_i - 1 \right),$$

where λ and γ are the multipliers. Differentiating $\mathcal{L}$ with respect to p_is and setting the derivatives to zero gives $\gamma = -n$. Then the solution is given by

$$p_i = \frac{1}{n}\frac{1}{1 + \lambda^\top g(X_i, \theta)}, \tag{3}$$

where $\lambda \equiv \lambda(\theta)$ satisfies

$$\frac{1}{n}\sum_{i=1}^{n} \frac{g(X_i, \theta)}{1 + \lambda^\top g(X_i, \theta)} = 0. \tag{4}$$

Instead of solving the nonlinear Equation 4, we solve the dual problem with respect to λ. Substituting the expression for p_i in Equation 3 into Equation 2 gives

$$\log\left(R(\theta)\right) = -\sum_{i=1}^{n} \log\left(1 + \lambda^\top g(X_i, \theta)\right) =: r(\lambda). \tag{5}$$

Now consider minimizing $r(\lambda)$ subject to $1 + \lambda^\top g(X_i, \theta) \geq 1/n$ for $i = 1, \dots, n$. This is a convex optimization problem, where the constraints correspond to the condition that $0 \leq p_i \leq 1$ for all i. Next, we remove the constraints by employing a pseudo logarithm function

$$\log^\star(x) = \begin{cases} \log(x) & \text{if } x \geq 1/n \\ -n^2x^2/2 + 2nx - \log(n) - 3/2 & \text{if } x < 1/n. \end{cases} \tag{6}$$

Minimizing $r^\star(\lambda) = -\sum_{i=1}^{n} \log^\star(1 + \lambda^\top g(X_i, \theta))$ without the constraints does not affect the solution and the Newton-Raphson method can be applied to find it. If the convex hull constraint is violated, the algorithm does not converge with $\|\lambda\|$ increasing as the iteration proceeds. Hence, it can be computationally more efficient to minimize $r^\star(\lambda)$ first to get $\log(R(\theta))$ and indirectly check the convex hull constraint by observing λ and p_is. Note that EL is maximized when $\lambda = 0$ and $p_i = 1/n$ for all i. It follows from Equation 4 that $\hat{\theta}$, the MELE, is obtained by solving the estimating equations

$$\sum_{i=1}^{n} g(X_i, \theta) = 0.$$

The existence, uniqueness, and asymptotic properties of $\hat{\theta}$ are well established in the literature. Assume that there exists $\theta_0 \in \Theta$ that is the unique solution to the moment condition in Equation 1. Similar to the parametric likelihood method, define the minus twice the empirical log-likelihood ratio function as $l(\theta_0) = -2\log(R(\theta_0))$. Under regularity conditions, it is known that the following nonparametric version of Wilks' theorem holds:

$$l(\theta_0) \to_d \chi^2_p \text{ as } n \to \infty,$$

where χ^2_p is the chi-square distribution with p degrees of freedom. See, e.g., Qin and Lawless (1994) for proof and the treatment of more general cases, including the over-identified ones. For a level $\alpha \in (0, 1)$, let $\chi^2_{p,\alpha}$ be the $100(1-\alpha)$%th percentile of χ^2_p. Since $P(l(\theta_0) \leq \chi^2_{p,\alpha}) \to 1 - \alpha$, an asymptotic $100(1-\alpha)$% confidence region for θ can be obtained as

$$\left\{\theta : l(\theta) \leq \chi^2_{p,\alpha}\right\}. \tag{7}$$

Often the chi-square calibration is unsatisfactory due to the slow convergence, especially when n is small. We review other calibration methods that address this issue. First, consider EL for the mean with $g(X_i, \theta) = X_i - \theta$ and $\theta_0 = \mathsf{E}[X_i]$. Then we have

$$l(\theta_0) = n(\bar{X} - \theta_0)^\top V^{-1}(\bar{X} - \theta_0) + o_P(1),$$

where $V = n^{-1}\sum_{i=1}^n (X_i - \theta_0)(X_i - \theta_0)^\top$. Let $S = (n-1)^{-1}\sum_{i=1}^n (X_i - \bar{X})(X_i - \bar{X})^\top$ and define a Hotelling's T squared statistic as $T^2 = n(\bar{X} - \theta_0)^\top S^{-1}(\bar{X} - \theta_0)$. It can be shown that

$$n(\bar{X} - \theta_0)^\top V^{-1}(\bar{X} - \theta_0) = \frac{nT^2}{T^2 + n - 1} \to_d \frac{p(n-1)}{n-p} F_{p,n-p},$$

where $F_{p,n-p}$ is the F distribution with p and $n-p$ degrees of freedom. This suggests that we can use $p(n-1)F_{p,n-p,\alpha}/n-p$ in place of $\chi^2_{p,\alpha}$. The F calibration yields a larger critical value than the chi-square calibration, which leads to a better coverage probability of the confidence region in Equation 7. Next, a more generally applicable method is the Bartlett correction. Based on the Edgeworth expansion, it requires the Cramér's condition and finite higher moments of $g(X_i, \theta)$. The correction is given by a scale multiple of $\chi^2_{p,\alpha}$ as $(1 + a/n)\chi^2_{p,\alpha}$ with an unknown constant a. In practice, the Bartlett correction involves getting a consistent estimate $\hat{a}$ with plug-in sample moments. The coverage error of the Bartlett corrected confidence region is reduced from $O(n^{-2})$ to $O(n^{-1})$ (DiCiccio *et al.* 1991), which is unattainable by the F calibration. Another effective calibration method is the bootstrap. Let $\widetilde{\mathcal{X}}_n = \{\widetilde{X}_1, \ldots, \widetilde{X}_n\}$ denote the null-transformed data such that

$$\mathsf{E}_{P_n}[g(\widetilde{X}_i, \theta)] = \frac{1}{n}\sum_{i=1}^n g(\widetilde{X}_i, \theta) = 0,$$

so Equation 1 holds for $\widetilde{\mathcal{X}}_n$ with P_n. Define a bootstrap sample $\widetilde{\mathcal{X}}_n^* = \{\widetilde{X}_1^*, \ldots, \widetilde{X}_n^*\}$ as i.i.d. observations from $\widetilde{\mathcal{X}}_n$. We can compute the bootstrap EL ratio $l^*(\theta)$ with $\widetilde{\mathcal{X}}_n^*$ in the same way. The critical value, the $100(1-\alpha)\%$th percentile of the distribution of $l^*(\theta)$, can be approximated using a large number, say B, of bootstrap samples. As an example, we may set $\widetilde{X}_i = X_i - \bar{X} + \theta$ when $g(X_i, \theta) = X_i - \theta$. It is equivalent to computing $l^*(\bar{X})$ with a bootstrap sample from the observed data directly. The $O(n^{-2})$ coverage error can also be achieved by the bootstrap calibration (Hall and Scala 1990).

Although EL does require full model specification, it is not entirely free of the misspecification issue. Developing diagnostic measures for EL is still an open problem, and we briefly introduce the technique of empirical likelihood displacement (ELD) (Lazar 2005). Much like the concept of likelihood displacement (Cook 1986), ELD can be used to detect influential observations or outliers. With the MELE $\hat{\theta}$ from the complete data, consider reduced data with the ith observation deleted and the corresponding MELE estimate $\hat{\theta}_{(i)}$, Then ELD is defined as

$$\mathrm{ELD}_i = 2\left(L(\hat{\theta}) - L(\hat{\theta}_{(i)})\right), \tag{8}$$

where $\hat{\theta}_{(i)}$ is plugged into the original EL function $L(\theta)$. If ELD_i is large, the ith observation is an influential point and can be inspected as a possible outlier. See Zhu, Ibrahim, Tang, and Zhang (2008) for other diagnostic measures for EL.

2.2. Empirical likelihood for linear models

We now turn our attention to linear models, which are the main focus of **melt**. First, suppose we have independent observations $\{(Y_i, X_i)\}_{i=1}^n$, where Y_i is the univariate response and X_i is the p-dimensional covariate (including the intercept, if any). For illustration purposes, we consider X_i fixed and do not explicitly distinguish between random and fixed designs. Then the analysis can be performed conditional on X_i, and the EL methods need slight modification (Owen 1991). See Kitamura, Tripathi, and Ahn (2004) for formal methods for models with conditional moment restrictions. For standard linear regression models, assume that

$$\mathsf{E}[Y_i] = \mu_i,\ \mathsf{VAR}[Y_i] = \sigma_i^2,\ i = 1, \ldots, n,$$

where $\mu_i = X_i^\top \theta^*$ for some $\theta^* \in \mathbb{R}^p$. Since θ^* minimizes $\mathsf{E}[(Y_i - X_i^\top \theta)^2]$, we have the following moment conditions

$$\mathsf{E}[X_i(Y_i - X_i^\top \theta)] = 0,\ i = 1, \ldots, n,$$

and the estimating equations

$$\sum_{i=1}^n X_i(Y_i - X_i^\top \theta) = 0.$$

Let $Z_i = (Y_i, X_i)$ and $g(Z_i, \theta) = (Y_i - X_i^\top \theta)X_i$. Note that $g(Z_i, \theta)$s are independent with non-constant variances, regardless of whether σ_i^2s are constant. Following the steps in Section 2.1, we can compute the EL ratio function

$$R(\theta) = \max_{p_i} \left\{ \prod_{i=1}^n np_i : \sum_{i=1}^n p_i g(Z_i, \theta) = 0,\ p_i \geq 0,\ \sum_{i=1}^n p_i = 1 \right\}. \tag{9}$$

Under mild moment conditions it follows that $l(\theta^*) \to_d \chi_p^2$. Note also from Equation 9 that the least square estimator $\hat{\theta}$ is the MELE for θ, with $L(\hat{\theta}) = n^{-n}$ and $R(\hat{\theta}) = 0$.

Next, generalized linear models assume that

$$\mathsf{E}[Y_i] = \mu_i,\ G(\mu_i) = X_i^\top \theta,\ \mathsf{VAR}[Y_i] = \phi V(\mu_i),\ i = 1, \ldots, n,$$

where G and V are known link and variance functions, respectively, and $\phi > 0$ is an optional dispersion parameter. EL for generalized linear models builds upon quasi-likelihood methods (Wedderburn 1974). The log quasi-likelihood for Y_i is given by

$$Q(Y_i, \mu_i) = \int_{Y_i}^{\mu_i} \frac{Y_i - t}{\phi V(t)} dt.$$

Differentiating $Q(Y_i, \mu_i)$ with respect to θ yields the quasi-score

$$X_i \frac{H'(X_i^\top \theta) \left(Y_i - H(X_i^\top \theta) \right)}{\phi V \left(H(X_i^\top \theta) \right)} =: g_1(Z_i, \theta),$$

where H denotes the inverse link function. From $\mathsf{E}[g_1(Z_i, \theta^*)] = 0$ for $i = 1, \ldots, n$, we get the estimating equations

$$\sum_{i=1}^n g_1(Z_i, \theta) = 0.$$

Then the EL ratio function can be derived as in Equation 9 with the same asymptotic properties. It can be seen that the MELE for θ is the same as the quasi-maximum likelihood estimator. When overdispersion is present with unknown ϕ, we introduce another estimating function based on the squared residuals. Let $\eta = (\theta, \phi)$ and

$$g_2(Z_i, \eta) = \frac{\left(Y_i - H(X_i^\top \theta)\right)^2}{\phi^2 V\left(H(X_i^\top \theta)\right)} - \frac{1}{\phi},$$

where $\mathsf{E}[g_2(Z_i, \eta^*)] = 0$ with the true value η^*. We compute the EL ratio function with an additional constraint as

$$R(\eta) = \max_{p_i} \left\{ \prod_{i=1}^n np_i : \sum_{i=1}^n p_i g_1(Z_i, \eta) = 0,\ \sum_{i=1}^n p_i g_2(Z_i, \eta) = 0,\ p_i \geq 0,\ \sum_{i=1}^n p_i = 1 \right\}.$$

The computation is straightforward since the number of parameters equals the number of constraints. Confidence regions for θ can be constructed by applying a calibration method to $l(\theta)$. One advantage of using EL for linear models is that the confidence regions have data-driven shapes and orientations.

2.3. Hypothesis testing with empirical likelihood

As seen in Section 2.2, it is easy to compute the MELE and evaluate the EL ratio function at a given value for linear models. Conducting significance tests, or hypothesis testing in general, is often the main interest when using linear models. The EL methods can be naturally extended to testing hypotheses by imposing appropriate constraints on the parameter space Θ (Qin and Lawless 1995; Adimari and Guolo 2010). Consider a null hypothesis $\mathcal{H}$ corresponding to a nonempty subset of Θ through a smooth q-dimensional function h such that $\mathcal{H} = \{\theta \in \Theta : h(\theta) = 0\}$. With additional conditions on $\mathcal{H}$ and h, it can be shown that

$$\inf_{\theta : h(\theta) = 0} l(\theta) \to_d \chi^2_q. \tag{10}$$

In practice, computing the solution in Equation 10 is a nontrivial task. Recall that the convex hull constraint restricts the domain of $l(\theta)$ to

$$\Theta_n \coloneqq \{\theta \in \Theta : 0 \in \mathrm{Conv}_n(\theta)\},$$

where $\mathrm{Conv}_n(\theta)$ denotes the convex hull of $\{g(Z_i, \theta)\}_{i=1}^n$ with an estimating function g. Except for a few cases, both $l(\theta)$ and Θ_n are nonconvex in θ, and fully identifying Θ_n can be even more challenging than the constrained minimization problem itself. Given that the solution can only be obtained numerically by an iterative process, it is essential to monitor the entire solution path in $\mathrm{Conv}_n(\theta) \cap \mathcal{H}$. Another difficulty is in the nested optimization structure. The Lagrange multiplier λ needs to be updated for each update of θ, which amounts to solving an inner layer of optimization in Equation 5 at every step. It is clear that no single method can be applied to all estimating functions and hypotheses. Tang and Wu (2014) proposed a nested coordinate descent algorithm for general constrained EL problems, where the outer layer is optimized with respect to θ with λ fixed. After some algebra, we obtain for $\theta \in \Theta_n$ the gradient of EL ratio function

$$\nabla \log\left(R(\theta)\right) = -\frac{1}{n} \sum_{i=1}^n \frac{1}{1 + \lambda^\top g(Z_i, \theta)} \partial_\theta g(Z_i, \theta) \lambda, \tag{11}$$

where $\partial_\theta g(Z_i, \theta)$ represents the Jacobian matrix of $g(Z_i, \theta)$. Observe that the expression does not involve any derivative of λ. In order to reduce the computational complexity, we focus only on linear hypotheses of the form

$$\mathcal{H} = \{\theta \in \Theta : L\theta = r\}, \tag{12}$$

which works well with linear models. We use projected gradient descent instead of the coordinate descent approach to obtain a local minimum of $l(\theta)$ in Equation 10. The projected gradient descent can be computed efficiently with Equation 11. Then it would take a relatively small number of iterations for convergence, reducing the required number of inner layer updates of λ.

Controlling the type 1 error rate is necessary when testing multiple hypotheses simultaneously. Recently there has been interest in multiplicity-adjusted test procedures for Wald-type test statistics that asymptotically have a multivariate chi-square distribution under the global null hypothesis (Dickhaus and Royen 2015; Dickhaus and Sirotko-Sibirskaya 2019). Kim, MacEachern, and Peruggia (2021) proposed single-step multiple testing procedures for EL that asymptotically control the family-wise error rate with Monte Carlo simulations or bootstrap. Wang and Yang (2018) applied the F-calibrated EL statistics to the Benjamini-Hochberg procedure (Benjamini and Hochberg 1995) to control the false discovery rate.

3. Overview of melt

The latest stable release of **melt** is available from the CRAN at `https://CRAN.R-project.org/package=melt`, and the development version is on GitHub at `https://github.com/markean/melt`. Computational tasks are implemented in parallel using OpenMP (Dagum and Menon 1998) API in C++ with the **Rcpp** (Eddelbuettel and Balamuta 2018) and **RcppEigen** (Bates and Eddelbuettel 2013) packages to interface with R. Depending on the platform, the package can be compiled from source with support for OpenMP.

The overall design of **melt** adopts functional object-oriented programming approach (Chambers 2014) with S4 classes and methods. Every function of the package is either a wrapper that creates a single instance of an object or a method that can be applied to a class object. The workflow of the package consists of three steps: (1) fitting a model, (2) examining and diagnosing the fitted model, and (3) testing hypotheses with the model. Four functions are available to build a model object whose names start with the prefix `el_`, which stands for empirical likelihood. A summary of the functions is provided below.

- `el_mean()`: creates an ‘`EL`’ object for the mean.
- `el_sd()`: creates a ‘`SD`’ object for the standard deviation.
- `el_lm()`: creates an ‘`LM`’ object for the linear model.
- `el_glm()`: creates a ‘`GLM`’ object for the generalized linear model.

For univariate data, `el_mean()` corresponds to `t.test()` in the **stats** package. `el_lm()` and `el_glm()` correspond to `lm()` and `glm()` as well.

All model objects inherit from class ‘`EL`’, and a description of the slots in ‘`EL`’ is given in Table 1. Notably, the `optim` slot summarizes the optimization results with an accessor method `getOptim()`. It is a list with the following four components:

Slots	Class	Description
`optim`	`list`	Optimization results.
`logl`	`numeric(1)`	Empirical log-likelihood.
`loglr`	`numeric(1)`	Empirical log-likelihood ratio.
`statistic`	`numeric(1)`	Minus twice the empirical log-likelihood ratio.
`df`	`integer(1)`	Degrees of freedom associated with `statistic`.
`pval`	`numeric(1)`	p value of `statistic`.
`weights`	`numeric`	Re-scaled weights used for model fitting.
`coefficients`	`numeric`	MELE of the parameters.

Table 1: A description of some of the slots in an ‘`EL`’ object. `numeric(1)` and `integer(1)` refer a single numeric and integer, respectively. A full explanation of the class and slots can be found in the documentation of `EL-class` in the package.

- `par`: a numeric vector for the user-supplied parameter value θ where EL is evaluated.
- `lambda`: a numeric vector for the Lagrange multiplier λ.
- `iterations`: a single integer for the number of iterations performed.
- `convergence`: a single logical for the convergence status. It is either `TRUE` or `FALSE`.

Note that `par` is fixed in evaluating EL. The optimization is performed with respect to `lambda`, so `iterations` and `convergence` need to be understood in terms of `lambda`. Here we make a distinction between EL evaluation and EL optimization. The EL optimization refers to the constrained EL problem discussed in Section 2.3 and corresponds to another class ‘`CEL`’ that directly extends ‘`EL`’. The `optim` slot in a ‘`CEL`’ object has the same components. However, the optimization results are now interpreted in terms of `par`, the solution to the constrained problem. The ‘`LM`’ and ‘`GLM`’ classes contain ‘`CEL`’, meaning that a constrained optimization is performed initially when `el_lm()` or `el_glm()` is called. In order to avoid confusion, the ‘`CEL`’ class only distinguishes between EL optimization from EL evaluation, and the user does not directly interact with a ‘`CEL`’ object. Once `par` is obtained through evaluation or optimization, it uniquely determines `lambda` and, in turn, `logl` and `loglr`. Accessor methods, `logL()` and `logLR()`, are available to retrieve these slots. Then `statistic` is equivalent to `-2 * loglr` and has an asymptotic chi-square distribution under the null hypothesis, with the associated `df` and `pval`. All four model fitting functions above accept an optional argument `weights` for weighted data. A vector of weights is then re-scaled internally for numerical stability in the computation of weighted EL (Glenn and Zhao 2007). Although other accessor methods `weights()` and `coef()` can extract `weights` and `coefficients`, these slots are stored for subsequent analyses and methods.

In the next step, the following methods can be applied to an ‘`EL`’ object to evaluate the model fit or compute summary statistics:

- `conv()`: extracts convergence status from a model. The distinction between EL evaluation and EL optimization applies here as well. It can be used to check the convex hull constraint indirectly.
- `confint()`: computes confidence intervals for model parameters.

- `confreg()`: computes a two-dimensional confidence region for model parameters. It returns an object of class `ConfregEL` where a subsequent `plot()` method is applicable.

- `eld()`: computes empirical likelihood displacement in Equation 8 for model diagnostics and outlier detection. It returns an object of class `ELD` where a subsequent `plot()` method is applicable.

- `summary()`: summarizes the results of the overall model test and the significance tests for coefficients. Similar to `summary.lm()` and `summary.glm()`, it applies to a '`LM`' or '`GLM`' object and returns an object of class '`SummaryLM`'.

Lastly, we introduce the two main functions of **melt** that perform hypothesis testing. These generic methods that take an '`EL`' object with other arguments that specify the problem in Equation 10.

- `elt()`: tests a linear hypothesis with EL. It returns an object of class '`ELT`' that contains the test statistic, the critical value, and the level of the test. Several calibration options discussed in Section 2.2 are available, and the p value is determined by the calibration method chosen.

- `elmt()`: tests multiple linear hypotheses simultaneously with EL. Each test can be considered as one instance of `elt()`. It returns an object of class '`ELMT`' with slots similar to those in '`ELT`'.

An '`ELT`' object also has `optim` slot, which does not necessarily correspond to the EL optimization. The user can supply an arbitrary parameter value to test, reducing the problem to the EL evaluation. In terms of Equation 12, this is equivalent to testing $\theta = r$ by setting $L = I_p$, where I_p is the identity matrix of order p. `elmt()` applies the single-step multiple testing procedure of Kim *et al.* (2021). The multiplicity-adjusted critical value and p values are estimated by Monte Carlo simulation.

Note that every step of the workflow involves possibly multiple EL evaluations or optimizations. Hence it is necessary to flexibly control the details of the execution and computation at hand. All model fitting functions and most methods in accept an argument `control`, which allows the user to specify the control parameters. Only an object of class '`ControlEL`' can be supplied as `control` to ensure validity and avoid unexpected errors. Some of the slots in '`ControlEL`' are described in Table 2. Another wrapper, `el_control()`, is available to construct a '`ControlEL`' object and specify the parameters. The default values are shown below.

```
el_control(
  maxit = 200L, maxit_l = 25L, tol = 1e-06, tol_l = 1e-06, step = NULL,
  th = NULL, verbose = FALSE, keep_data = TRUE, nthreads,
  seed = sample.int(.Machine$integer.max, 1L), b = 10000L, m = 1000000L
)
```

An important feature is that '`ControlEL`' is independent of the other classes in the package, making it possible to apply different parameters for different tasks.

Slots	Class	Description
`maxit`	`integer(1)`	Maximum number of iterations for the EL optimization.
`maxit_l`	`integer(1)`	Maximum number of iterations for the EL evaluation.
`tol`	`numeric(1)`	Convergence tolerance for the EL optimization.
`tol_l`	`numeric(1)`	Convergence tolerance for the EL evaluation.
`step`	`numeric(1)`	Step size for projected gradient descent method in the EL optimization.
`th`	`numeric(1)`	Threshold for the negative empirical log-likelihood ratio. The iteration stops if the value exceeds the threshold.
`nthreads`	`integer(1)`	Number of threads for parallel computation.

Table 2: A description of some of the slots in an '`ControlEL`' object. A full explanation of the class and slots can be found in the documentation of `ControlEL-class` or `el_control()` in the package.

4. Usage

4.1. Model building

For a simple illustration of building a model, we apply `el_mean()` to the synthetic classification problem data `synth.tr` from the **MASS** package (Venables and Ripley 2002). The **tidyverse** package (Wickham, Averick, Bryan, Chang, McGowan, François, Grolemund, Hayes, Henry, Hester, Kuhn, Pedersen, Miller, Bache, Müller, Ooms, Robinson, Seidel, Spinu, Takahashi, Vaughan, Wilke, Woo, and Yutani 2019) is used to perform data manipulation and visualization. To ensure reproducibility, we set the seed for pseudo-random number generation.

```
R> library("melt")
R> library("MASS")
R> library("tidyverse")
R> set.seed(114230)
R> theme_set(theme_bw())
R> data("synth.tr", package = "MASS")
R> data <- dplyr::select(synth.tr, c(xs, ys))
R> summary(data)
```

```
       xs                 ys
 Min.   :-1.24652   Min.   :-0.1913
 1st Qu.:-0.50923   1st Qu.: 0.3234
 Median :-0.04183   Median : 0.4898
 Mean   :-0.07276   Mean   : 0.5044
 3rd Qu.: 0.36996   3rd Qu.: 0.7044
 Max.   : 0.86130   Max.   : 1.0932
```

With the focus on `xs` and `ys`, the x and y coordinates, we first visualize the domain of EL function with the convex hull constraint in Figure 1. Any parameter value inside the convex

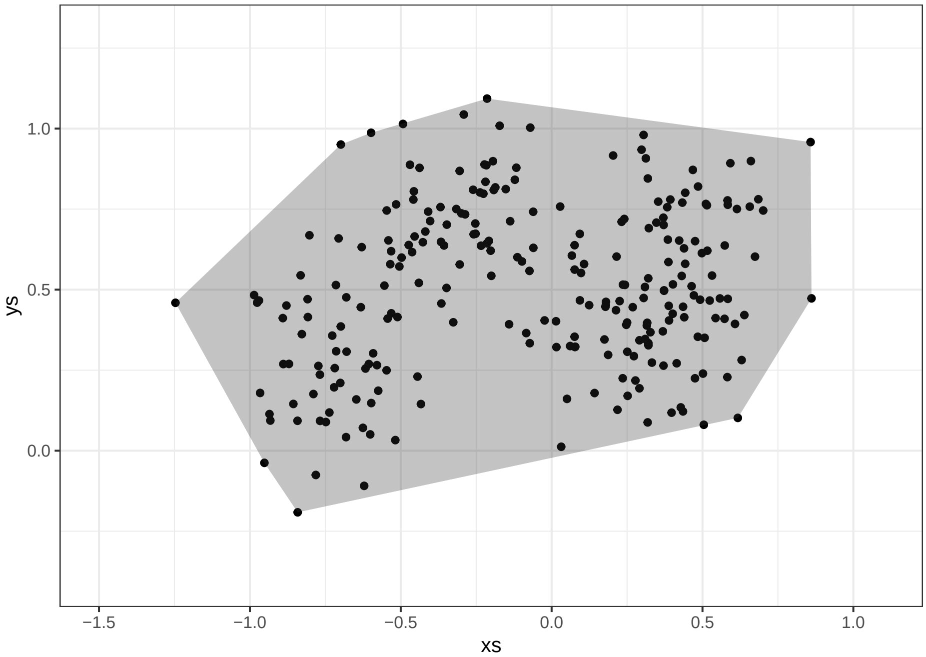

Figure 1: Scatter plot of `ys` versus `xs` in the `synth.tr` data with 250 observation. The convex hull of the observations is shaded in grey.

hull leads to proper EL evaluation. We specify `c(0, 0.5)` as `par` in `el_mean()` and build an '`EL`' object with the '`data.frame`' `data`.

```
R> fit_mean <- el_mean(data, par = c(0, 0.5))
```

`data` is implicitly coerced into a '`matrix`' since `el_mean()` takes a numeric '`matrix`' as an input for the data. Basic `print` and `show` methods display relevant information about an '`EL`' object.

```
R> fit_mean

        Empirical Likelihood: mean

Maximum EL estimates:
      xs       ys
-0.07276  0.50436

Chisq: 6.158, df: 2, Pr(>Chisq): 0.04601

EL evaluation: converged
```

The asymptotic chi-square statistic is displayed, along with the associated degrees of freedom and the p value. The MELE is just the average of the observations, and the empirical log-likelihood ratio is minimized at the MELE. We note that the MELE is independent of the par specified, which makes it convenient to build a model when the user is more interested in a subsequent analysis with an 'EL' object.

```
R> fit2_mean <- el_mean(data, par = c(100, 100))
R> all.equal(coef(fit2_mean), colMeans(data))

[1] TRUE

R> fit3_mean <- el_mean(data, par = coef(fit2_mean))
R> all.equal(logLR(fit3_mean), 0)

[1] TRUE
```

As an illustration of weighted EL, we specify an arbitrary weight in el_mean() for weighted EL evaluation. The MELE is the weighted average of the observations in this case. The re-scaled weights returned by weights() add up to the total number of observations.

```
R> weights <- rep(c(1, 2), each = 125)
R> (wfit_mean <- el_mean(data, par = c(0, 0.5), weights = weights))

        Weighted Empirical Likelihood: mean

Maximum EL estimates:
      xs       ys
-0.02319  0.56390

Chisq: 18.33, df: 2, Pr(>Chisq): 0.0001047

EL evaluation: converged

R> all.equal(sum(weights(wfit_mean)), nobs(wfit_mean))

[1] TRUE
```

Next, we consider an infeasible parameter value c(1, 0,5) outside the convex hull to show that how el_control() interacts with the model fitting functions through control argument. By employing the pseudo logarithm function in Equation 6, the evaluation algorithm continues until the iteration reaches maxit_l or the negative empirical log-likelihood ratio exceeds th. Setting a large th for the infeasible value, we observe that the algorithm hits the maxit with each element of lambda diverging quickly.

```
R> ctrl <- el_control(maxit_l = 50, th = 10000)
R> fit4_mean <- el_mean(data, par = c(1, 0.5), control = ctrl)
R> logL(fit4_mean)

[1] -10001.14
```

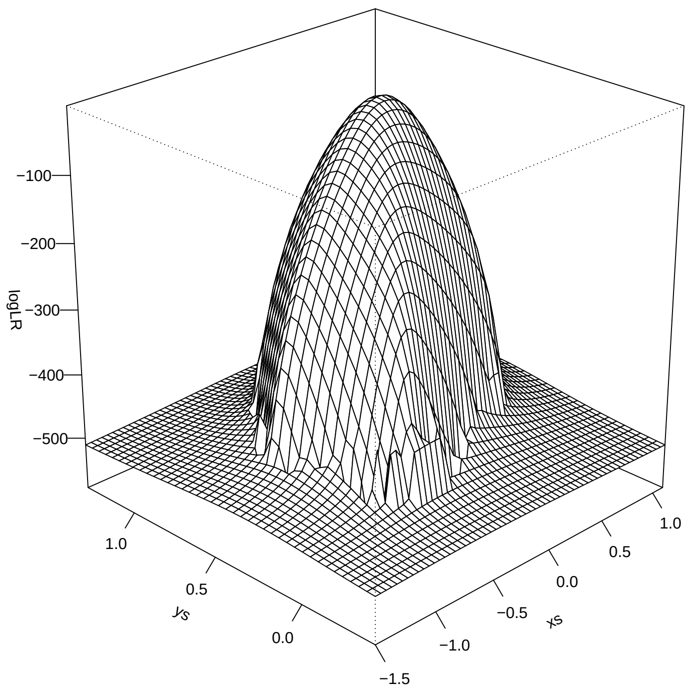

Figure 2: Surface plot of empirical log-likelihood ratio obtained from `synth.tr` with `el_mean()`. `th` is set to 400.

```
R> logLR(fit4_mean)

[1] -8620.776

R> getOptim(fit4_mean)

$par
 xs  ys
1.0 0.5

$lambda
[1] -9.908531e+14  2.757135e+14

$iterations
[1] 50

$convergence
[1] FALSE
```

Figure 2 shows a surface plot of the empirical log-likelihood ratio on the grid of Figure 1. The boundary of the convex hull separates the feasible region from the infeasible region.

A similar process applies to the other model fitting functions, except that `el_lm()` and `el_glm()` require a '`formula`' object for model specification. In addition, **melt** contains another function `el_eval()` to perform the EL evaluation for other general estimating functions. For example, consider the mean and standard deviation denoted by $\theta = (\mu, \sigma)$. For a given value of θ, we evaluate the estimating function $g(X_i, \theta) = (X_i - \mu, (X_i - \mu)^2 - \sigma^2)$ with the available data $X_1, \ldots, X_n$. `el_eval()` takes a '`matrix`' argument `g`, where each row corresponds to $g(X_i, \theta)$.

```
R> mu <- 0
R> sigma <- 1
R> x <- rnorm(100)
R> g <- matrix(c(x - mu, (x - mu)^2 - sigma^2), ncol = 2)
R> fit_eval <- el_eval(g)
R> fit_eval$pval
```

```
[1] 0.1509259
```

Although the user can supply a custom `g`, `el_eval()` is not the main function of the package. `el_eval()` returns a '`list`' with the same components as in an '`EL`' object, but no other methods are applicable further. The scope is also limited to just-identified estimating functions. For more flexible and over-identified estimating functions, it is recommended to use other packages, e.g., **gmm** (Chaussé 2010) or **momentfit** (Chaussé 2020).

4.2. Linear regression analysis

We illustrate the use of `el_lm()` for regression analysis with the Boston housing price data `Boston` available in **MASS** (Venables and Ripley 2002). We first update the control parameters for significance tests of the coefficients.

```
R> data("Boston", package = "MASS")
R> ctrl <- el_control(maxit = 500, tol = 1e-04, th = 10000)
R> (fit_lm <- el_lm(medv ~ crim + indus + chas + nox + age + lstat,
+    data = Boston,
+    control = ctrl
+ ))
```

```
        Empirical Likelihood: lm

Maximum EL estimates:
(Intercept)        crim       indus        chas         nox 
   32.76605    -0.05674    -0.17924     4.68855    -0.02926 
        age       lstat 
    0.04812    -0.91991 

Chisq: 623.3, df: 6, Pr(>Chisq): < 2.2e-16

Constrained EL: converged
```

The `print()` method also applied and shows the MELE, the overall model test result, and the convergence status. The estimates are obtained from `lm.fit()`. The hypothesis for the overall test is that all the parameters except the intercept are 0. The convergence status shows that a constrained optimization is performed in testing the hypothesis. The EL evaluation applies to the test and the convergence status if the model does not include an intercept. `conv()` can be used to extract the convergence status.

```
R> conv(fit_lm)

[1] TRUE
```

It is designed to return a single logical, which can be helpful in a control flow where the convergence status decides the course of action. The large chi-square value implies that the data do not support the hypothesis, regardless of the convergence. Note that failure to converge does not necessarily indicate unreliable test results. Most commonly, the algorithm fails to converge if the additional constraint imposed by a hypothesis is incompatible with the convex hull constraint. The control parameters affect the test results as well. The `summary()` method reports the results of significance tests, where each test involves solving a constrained EL problem.

```
R> summary(fit_lm)

Call:
el_lm(formula = medv ~ crim + indus + chas + nox + age + lstat,
    data = Boston, control = ctrl)

Coefficients:
            Estimate   Chisq Pr(>Chisq)    
(Intercept) 32.76605 385.205    < 2e-16 ***
crim        -0.05674   3.301   0.069223 .  
indus       -0.17924  12.859   0.000336 ***
chas         4.68855  16.904   3.93e-05 ***
nox         -0.02926   0.001   0.973702    
age          0.04812   9.936   0.001621 ** 
lstat       -0.91991 279.671    < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Chisq: 623.3, df: 6, Pr(>Chisq): < 2.2e-16

Constrained EL: converged
```

These tests are all asymptotically pivotal without explicit studentization. As a result, the output does not have standard errors. `getSigTests()` returns the details of the tests.

```
R> getSigTests(fit_lm)$convergence

(Intercept)        crim       indus        chas         nox 
      FALSE        TRUE        TRUE        TRUE        TRUE 
        age       lstat 
       TRUE        TRUE 
```

DDD

```
R> # confint(fit_lm)
R> # plot(confreg(fit_lm, parm = c("crim", "lstat"), npoints = 30))
```

- `confint()`:
- `confreg()`:
- `eld()`:

4.3. Hypothesis testing

4.4. Multiple testing

count data or grouped binary data

5. Conclusion

Acknowledgments

This work was supported by the U.S. National Science Foundation under Grants No. SES-1921523 and DMS-2015552.

References

Adimari G, Guolo A (2010). "A Note on the Asymptotic Behaviour of Empirical Likelihood Statistics." *Statistical Methods & Applications*, **19**(4), 463–476. doi:10.1007/s10260-010-0137-9.

Barton WH (2022). ***emplik2**: Empirical Likelihood Ratio Test for Two Samples with Censored Data*. R package version 1.32, URL https://CRAN.R-project.org/package=emplik2.

Bates D, Eddelbuettel D (2013). "Fast and Elegant Numerical Linear Algebra Using the **RcppEigen** Package." *Journal of Statistical Software*, **52**(5), 1–24. doi:10.18637/jss.v052.i05.

Benjamini Y, Hochberg Y (1995). "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing." *Journal of the Royal Statistical Society B*, **57**(1), 289–300. doi:10.1111/j.2517-6161.1995.tb02031.x.

Chambers JM (2014). "Object-Oriented Programming, Functional Programming and R." *Statistical Science*, **29**(2), 167–180. doi:10.1214/13-STS452.

Chaudhuri S, Mondal D, Yin T (2017). "Hamiltonian Monte Carlo Sampling in Bayesian Empirical Likelihood Computation." *Journal of the Royal Statistical Society B*, **79**(1), 293–320. doi:https://doi.org/10.1111/rssb.12164.

Chaussé P (2010). "Computing Generalized Method of Moments and Generalized Empirical Likelihood with R." *Journal of Statistical Software*, **34**(11), 1–35. doi:10.18637/jss.v034.i11.

Chaussé P (2020). ***momentfit**: Methods of Moments*. R package version 0.2, URL https://CRAN.R-project.org/package=momentfit.

Chen SX, Cui H (2003). "An Extended Empirical Likelihood for Generalized Linear Models." *Statistica Sinica*, **13**(1), 69–81.

Chen SX, van Keilegom I (2009). "A Review on Empirical Likelihood Methods for Regression." *Test*, **18**(3), 415–447. doi:10.1007/s11749-009-0159-5.

Chen SX, Peng L, Qin YL (2009). "Effects of Data Dimension on Empirical Likelihood." *Biometrika*, **96**(3), 711–722. doi:10.1093/biomet/asp037.

Cook RD (1986). "Assessment of Local Influence." *Journal of the Royal Statistical Society B*, **48**(2), 133–155. doi:10.1111/j.2517-6161.1986.tb01398.x.

Dagum L, Menon R (1998). "OpenMP: An Industry Standard API for Shared-Memory Programming." *IEEE Computational Science and Engineering*, **5**(1), 46–55. doi:10.1109/99.660313.

DiCiccio T, Hall P, Romano J (1991). "Empirical Likelihood is Bartlett-Correctable." *The Annals of Statistics*, **19**(2), 1053–1061. doi:10.1214/aos/1176348137.

DiCiccio TJ, Romano JP (1990). "Nonparametric Confidence Limits by Resampling Methods and Least Favorable Families." *International Statistical Review / Revue Internationale de Statistique*, **58**(1), 59–76. doi:10.2307/1403474.

Dickhaus T, Royen T (2015). "A Survey on Multivariate Chi-Square Distributions and Their Applications in Testing Multiple Hypotheses." *Statistics*, **49**(2), 427–454. doi:10.1080/02331888.2014.993639.

Dickhaus T, Sirotko-Sibirskaya N (2019). "Simultaneous Statistical Inference in Dynamic Factor Models: Chi-Square Approximation and Model-Based Bootstrap." *Computational Statistics & Data Analysis*, **129**, 30–46. doi:10.1016/j.csda.2018.08.012.

Eddelbuettel D, Balamuta JJ (2018). "Extending R with C++: A Brief Introduction to **Rcpp**." *The American Statistician*, **72**(1), 28–36. doi:10.1080/00031305.2017.1375990.

Efron B (1981). "Nonparametric Standard Errors and Confidence Intervals." *Canadian Journal of Statistics*, **9**(2), 139–158. doi:10.2307/3314608.

Fox J, Weisberg S (2019). *An R Companion to Applied Regression*. Third edition. Sage, Thousand Oaks CA. URL https://socialsciences.mcmaster.ca/jfox/Books/Companion/.

Glenn N, Zhao Y (2007). "Weighted Empirical Likelihood Estimates and Their Robustness Properties." *Computational Statistics & Data Analysis*, **51**(10), 5130–5141. doi:10.1016/j.csda.2006.07.032.

Hall P, Scala BL (1990). "Methodology and Algorithms of Empirical Likelihood." *International Statistical Review / Revue Internationale de Statistique*, **58**(2), 109–127. doi:10.2307/1403462.

Hansen LP (1982). "Large Sample Properties of Generalized Method of Moments Estimators." *Econometrica*, **50**(4), 1029–1054. doi:10.2307/1912775.

Hansen LP, Heaton J, Yaron A (1996). "Finite-Sample Properties of Some Alternative GMM Estimators." *Journal of Business & Economic Statistics*, **14**(3), 262–280. doi:10.2307/1392442.

Hjort NL, McKeague IW, van Keilegom I (2009). "Extending the Scope of Empirical Likelihood." *The Annals of Statistics*, **37**(3), 1079–1111. doi:10.1214/07-AOS555.

Hothorn T, Bretz F, Westfall P (2008). "Simultaneous Inference in General Parametric Models." *Biometrical Journal*, **50**(3), 346–363. doi:10.1002/bimj.200810425.

Imbens GW (1997). "One-Step Estimators for Over-Identified Generalized Method of Moments Models." *The Review of Economic Studies*, **64**(3), 359–383. doi:10.2307/2971718.

Jacod J, Sørensen M (2018). "A Review of Asymptotic Theory of Estimating Functions." *Statistical Inference for Stochastic Processes*, **21**(2), 415–434. doi:10.1007/s11203-018-9178-8.

Kien DT, Chaudhuri S, Wei NH (2017). ***elhmc**: Sampling from a Empirical Likelihood Bayesian Posterior of Parameters Using Hamiltonian Monte Carlo.* R package version 1.1.0, URL https://CRAN.R-project.org/package=elhmc.

Kim E (2022). ***melt**: Multiple Empirical Likelihood Tests.* R package version 1.5.1, URL https://CRAN.R-project.org/package=melt.

Kim E, MacEachern S, Peruggia M (2021). "Empirical Likelihood for the Analysis of Experimental Designs." arXiv:2112.09206 [stat.ME]. URL https://arxiv.org/abs/2112.09206.

Kitamura Y (1997). "Empirical Likelihood Methods with Weakly Dependent Processes." *The Annals of Statistics*, **25**(5), 2084–2102. doi:10.1214/aos/1069362388.

Kitamura Y, Stutzer M (1997). "An Information-Theoretic Alternative to Generalized Method of Moments Estimation." *Econometrica*, **65**(4), 861–874. doi:10.2307/2171942.

Kitamura Y, Tripathi G, Ahn H (2004). "Empirical Likelihood-Based Inference in Conditional Moment Restriction Models." *Econometrica*, **72**(6), 1667–1714. doi:10.1111/j.1468-0262.2004.00550.x.

Kolaczyk ED (1994). "Empirical Likelihood for Generalized Linear Models." *Statistica Sinica*, **4**(1), 199–218.

Lazar NA (2005). "Assessing the Effect of Individual Data Points on Inference From Empirical Likelihood." *Journal of Computational and Graphical Statistics*, **14**(3), 626–642. doi:10.1198/106186005X59568.

Li G, Li R, Zhou M (2005). *Empirical Likelihood in Survival Analysis*, pp. 337–349. World Scientific. doi:10.1142/9789812567765_0020.

Newey WK, Smith RJ (2004). "Higher Order Properties of GMM and Generalized Empirical Likelihood Estimators." *Econometrica*, **72**(1), 219–255. doi:10.1111/j.1468-0262.2004.00482.x.

Nordman DJ, Lahiri SN (2014). "A Review of Empirical Likelihood Methods for Time Series." *Journal of Statistical Planning and Inference*, **155**, 1–18. doi:10.1016/j.jspi.2013.10.001.

Owen A (1988). "Empirical Likelihood Ratio Confidence Intervals for a Single Functional." *Biometrika*, **75**(2), 237–249. doi:10.1093/biomet/75.2.237.

Owen A (1990). "Empirical Likelihood Ratio Confidence Regions." *The Annals of Statistics*, **18**(1), 90–120. doi:10.1214/aos/1176347494.

Owen A (1991). "Empirical Likelihood for Linear Models." *The Annals of Statistics*, **19**(4), 1725–1747. doi:10.1214/aos/1176348368.

Owen A (2001). *Empirical Likelihood*. Chapman & Hall/CRC. doi:10.1201/9781420036152.

Qin J, Lawless J (1994). "Empirical Likelihood and General Estimating Equations." *The Annals of Statistics*, **22**(1), 300–325. doi:10.1214/aos/1176325370.

Qin J, Lawless J (1995). "Estimating Equations, Empirical Likelihood and Constraints on Parameters." *Canadian Journal of Statistics*, **23**(2), 145–159. doi:10.2307/3315441.

R Core Team (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Shen B, Wang M (2022). ***ELCIC**: The Empirical Likelihood-Based Consistent Information Criterion*. R package version 0.2.0, URL https://CRAN.R-project.org/package=ELCIC.

Smith RJ (1997). "Alternative Semi-parametric Likelihood Approaches to Generalised Method of Moments Estimation." *The Economic Journal*, **107**(441), 503–519. doi:10.1111/j.0013-0133.1997.174.x.

Stein C (1956). "Efficient Nonparametric Testing and Estimation." In *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pp. 187–195. doi:10.1525/9780520313880.

Tang CY, Wu TT (2014). "Nested Coordinate Descent Algorithms for Empirical Likelihood." *Journal of Statistical Computation and Simulation*, **84**(9), 1917–1930. doi:10.1080/00949655.2013.770514.

Tsao M, Wu F (2013). "Empirical Likelihood on the Full Parameter Space." *The Annals of Statistics*, **41**(4), 2176–2196. doi:10.1214/13-AOS1143.

Valeinis J, Cers E (2022). ***EL**: Two-Sample Empirical Likelihood*. R package version 1.1, URL https://CRAN.R-project.org/package=EL.

Venables WN, Ripley BD (2002). *Modern Applied Statistics with S*. Fourth edition. Springer, New York. ISBN 0-387-95457-0, URL https://www.stats.ox.ac.uk/pub/MASS4/.

Wang L, Yang D (2018). "F-Distribution Calibrated Empirical Likelihood Ratio Tests for Multiple Hypothesis Testing." *Journal of Nonparametric Statistics*, **30**(3), 662–679. doi:10.1080/10485252.2018.1461867.

Wedderburn RWM (1974). "Quasi-Likelihood Functions, Generalized Linear Models, and the Gauss—Newton Method." *Biometrika*, **61**(3), 439–447. doi:10.1093/biomet/61.3.439.

Wickham H, Averick M, Bryan J, Chang W, McGowan LD, François R, Grolemund G, Hayes A, Henry L, Hester J, Kuhn M, Pedersen TL, Miller E, Bache SM, Müller K, Ooms J, Robinson D, Seidel DP, Spinu V, Takahashi K, Vaughan D, Wilke C, Woo K, Yutani H (2019). "Welcome to the **tidyverse**." *Journal of Open Source Software*, **4**(43), 1686. doi:10.21105/joss.01686.

Wu F, Zhang Y (2015). ***eel**: Extended Empirical Likelihood*. R package version 1.1, URL https://CRAN.R-project.org/package=eel.

Yuan KH, Jennrich RI (1998). "Asymptotics of Estimating Equations under Natural Conditions." *Journal of Multivariate Analysis*, **65**(2), 245–260. doi:10.1006/jmva.1997.1731.

Zhou M (2015). *Empirical Likelihood Method in Survival Analysis*, volume 79. Chapman & Hall/CRC. doi:10.1201/b18598.

Zhou M (2020). ***emplik**: Empirical Likelihood Ratio for Censored/Truncated Data*. R package version 1.1-1, URL https://CRAN.R-project.org/package=emplik.

Zhu H, Ibrahim JG, Tang N, Zhang H (2008). "Diagnostic Measures for Empirical Likelihood of General Estimating Equations." *Biometrika*, **95**(2), 489–507. doi:10.1093/biomet/asm094.

Affiliation:

Eunseop Kim
Department of Statistics
The Ohio State University
1958 Neil Ave.
Columbus, OH 43210, United States of America
E-mail: kim.7302@osu.edu

Steven N. MacEachern
Department of Statistics
The Ohio State University
1958 Neil Ave.
Columbus, OH 43210, United States of America
E-mail: snm@stat.osu.edu

Mario Peruggia
Department of Statistics
The Ohio State University
1958 Neil Ave.
Columbus, OH 43210, United States of America
E-mail: peruggia@stat.osu.edu