Augmented Lagrange for constrained optimizations in empirical likelihood estimations

By

Andrew Locke

A thesis submitted to Macquarie University

for the degree of Master of Research

Department of Statistics

October 2015

MACQUARIE
University
SYDNEY · AUSTRALIA

Examiner's Copy

Typeset in LaTeX 2_ε.

Except where acknowledged in the customary manner, the material presented in this thesis is, to the best of my knowledge, original and has not been submitted in whole or part for a degree in any university.

Andrew Locke

Abstract

Empirical Likelihood is a useful tool for parameter estimation and inference as it does not require knowledge about where the data comes from. A large strength is its applicability with different methods, it can be extended in many ways including regression or adding constraints using estimating equations. The positivity constraint of p_i has often been overlooked or ignored but this means existing methods may experience difficulties for some problems. This thesis looks at enforcing this constraint by applying the Karush–Kuhn–Tucker conditions together with a multiplicative iterative optimization method of updating parameters which ensures movement towards the constrained maximum. For other equality constraints, we apply Augmented Lagrange to the Empirical Likelihood maximisation. We demonstrate our method using simulation examples in linear regression and estimating equations on raw moments.

Keywords: Empirical Likelihood, Augmented Lagrange, Multiplicative Iterative Algorithm

Contents

1

Introduction

Empirical Likelihood (EL) has been applied to numerous problems and areas since gaining popularity. It is particularly useful since it is a nonparametric method, meaning we do not need to know or assume the data comes from a known distribution, which can then be used to perform analysis such as hypothesis testing and construction of confidence intervals. Data may be multivariate, come from multiple distributions or be censored, all of which are able to be handled by EL.

The basis of empirical likelihood as a tool for inference was established by Owen (1988). Many others have provided significant contributions allowing empirical likelihood to handle regression, estimating equations and smoothing problems by incorporating them in the form of constraints on the likelihood. A major review by Hall & La Scala (1990) summarizes key properties of Empirical Likelihood up to the time while the Owen (2001) book provides a broad overview of the subject.

Empirical likelihood's ability of combining with existing methods and incorporating information as constraints in the likelihood allows EL to be flexible in terms of handling problems which can take advantage of the asymptotic properties of EL to perform inference. Constraints containing information about parameters are usually applied to EL in the form of estimating equations to link the parameters with the maximization problem. A Lagrangian multiplier approach can be used to find the optimal solution for the EL ratio function with theses constraints.

Parametric likelihood methods are very popular for inference on data but require the knowledge of the kind of distribution the data comes from. For example, we have observed data X, known to come from a normal distribution. We know the probability density function $f(x)$ is of the form $\frac{1}{\sqrt{2\pi\sigma^2}}\exp^{-}\frac{(x-\mu)^2}{2\sigma^2}$, where we have parameters μ and σ^2. For a known σ^2 and unknown μ we are able to conduct hypothesis tests on values of μ and to construct confidence intervals. However, we may encounter a set of data which we do not know the distribution of, particularly in a parametric form. If an incorrect distribution is chosen for the data, this misspecification may lead to inefficient likelihood based estimates and incorrect tests and confidence intervals. Empirical likelihood does not suffer from this problem and does not require the assumption of data following a known distribution. Other nonparametric methods for hypothesis testing or inference include the jacknife and types of bootstrap but are less flexible when compared to empirical likelihood.

1.1 Parametric and Nonparametric models

Owen (2001) uses the example of earthworm somite data from Pearl & Fuller (1905) to demonstrate the advantages of empirical likelihood. The dataset contains the No. of somites on each of 487 worms gathered near Ann Arbor in 1902. A histogram gives an idea of the shape of distribution skewness. Here the data is left skewed. Naturally the next step is to quantify this in some way. We have general terms ‘mildly’ and ‘extremely’ to describe the amount of skewness but a numerical quantity may provide a better description. For a random variable X the coefficient of skewness is

$$\gamma = \frac{E((X - E(X))^3)}{E((X - E(X))^2)^{3/2}}$$

The coefficient of skewness is 0 for symmetric data whenever $E(|X|^3)$ exists such as a normal distribution.

Kurtosis, given in the formula below describes the weight of the distribution’s tails compared to a normal distribution (0 kurtosis). Positive kurtosis indicates heavier (fatter) tails than normal while negative indicates lighter tails (thinner) than normal.

$$\kappa = \frac{E((X - E(X))^4)}{E((X - E(X))^2)^2} - 3$$

Using these formulae we obtain the skewness and kurtosis of this sample. Confidence regions for the true γ and κ can be constructed using empirical likelihood.

Parametric likelihood methods must make an assumption about the distribution to be able to construct confidence regions for γ and κ. Most common methods involve a Normal distribution, which will be quite reliable for inferences on the mean μ as asymptotically the sample mean will tend towards a normal distribution due to the central limit theorem, for finite variance. However, for other statistics of parameters, a normal distribution may not be suitable and we may be unable to find a parametric family distribution which reliably fits the data. Rather than having to assign a distribution which we know does not fit the data, nonparametric methods are able to account for larger generalization of distributions but usually at a loss of power. Power is a useful tool for comparing competing tests on the same hypothesis. Owen (2001) argues the loss of power in empirical likelihood tests is insignificant.

It is logical to compare empirical likelihood to other nonparametric methods. Bootstrapping is an alternative nonparametric option which may also be able to handle data for which we do not have the distribution in a parametric form. Owen (2001) considers bootstrap analysis on the earthworm somite data stating it may be more reliable than parametric options. Bootstrapping is achieved by resampling the data. The results can be plotted and confidence regions can be created by looking at the central 100(1-α)% points. However, this would still require us to make an assumption on the shape and orientation of this region. Attempts have been made to solve this problem but have not proven successful, such as that by Owen (1990) for constructing polygonal regions.

The likelihood nature of Empirical likelihood is the cause of its advantages and disadvantages when compared with bootstrapping methods. In addition to creating data-determined confidence regions EL is able to take into account constraints on parameters and handle biased or incomplete data as well combine data from different sources. DiCiccio et al. (1991) show EL has improved inference accuracy over bootstrap via Bartlett correction. However, it can be difficult and computationally challenging to optimize likelihood functions over some nuisance parameters, with other parameters fixed at test values. Estimating equations are used to reduce the optimization problem to a convex problem, the solution of which can be found using iterated least squares.

It is also possible for EL to be used in conjunction with bootstrap. For example, EL can be used to determine a nested family of confidence regions, bootstrap can then be used to select the region for a given confidence level. Another way is to resample from a distribution which maximizes the empirical likelihood subject to some constraints.

1.2 Aim

While much has been written on the various methods and applications empirical likelihood can handle as well as comparisons with alternative methods. This dissertation draws attention to the positivity constraint of p_i's which has often been overlooked or ignored by existing methods giving way for difficulties to arise for certain problems. Chen et al. (2008) provide an adjustment to the EL function to handle this problem. We develop an alternative method of updating the p_is which will maintain the positivity constraint and incorporate constraints using an augmented Lagrange method. MATLAB code has been used to run simulations for specific examples to demonstrate how our model works.

2

Empirical Likelihood Methods

2.1 Empirical likelihood

Empirical likelihood is formed using a nonparametric likelihood function. Let $X = (X_1, \ldots, X_n)^T$ be a random sample from an unknown distribution f.

$$L(F) = \prod_{i=1}^{n} f(x_i)$$

Note: Here we have not assumed any parametric distribution for f. In practice we observe the data $X_i = x_i$, for $i = 1, \ldots, n$.. By using the notation $p_i = f(x_i)\Delta x_i$ which is the probability for $x_i \leq X_i \leq x_i + \Delta x_i$, we have constraints p_i such that $p_i \geq 0$ and $\sum_{i=1}^{n} p_i = 1$. Since X_i's are independent the likelihood for observations $x_1, \ldots, x_n$ is:

$$L(p_1, \ldots, p_n; \mathbf{X}) = \prod_{i=1}^{n} p_i$$

This is called the empirical likelihood. Note that here we have the same number of parameters as observations n.

Maximum empirical likelihood estimation can now be directly applied allowing us to estimate the parameters p_i , which can be shown to be $\frac{1}{n}$. That is, an equal probability mass for each of the n observed values $x_1, \ldots, x_n$. Kiefer & Wolfowitz (1956) first showed this result. Owen (2001) has shown that the X_i's do not need to be distinct, that is when there are ties in the data (e.g. $X_i = X_j$ for $i \neq j$) we get that same likelihood function. Empirical likelihood can also handle multivariate data, that is when X_i is a random vector. The EL is now defined on $\mathbb{R}^d$ rather than $\mathbb{R}$. Transformations can be applied to parameters or data for easier interpretation or making the data easier to visualize. EL maintains transformation invariance in the same way parametric likelihood are invariant under one to one transformations. This means the empirical likelihood ratio is the same for transformations of parameters.

As with parametric likelihood, empirical likelihood ratios form a basis for hypothesis tests and confidence intervals. The use of empirical likelihood ratios for hypothesis testing was demonstrated by Thomas & Grunkemeier (1975) for use in survival probabilities estimated by the Kaplan-Meier Curve. EL allowed hypothesis testing to be conducted for particular values of survival probabilities at the α level of significance for censored data. This required the use of Wilks's theorem (Wilks (1938)), which states under mild regularity conditions, a hypothesis test testing a nested model, the likelihood ratio test statistic $-2L(\theta_0)/L(\hat{\theta}))$ tends to a χ^2_q distribution as n $\rightarrow \infty$. Where q, the degrees of freedom, is the difference in the dimensionality of Θ under the null compared to alternative hypothesis. This also allows for confidence intervals to be constructed.

Following the definition that empirical likelihood is maximized by $p_i = \frac{1}{n}$. We can obtain a likelihood ratio function $R(F)$ which can be used to test hypotheses and construct confidence intervals:

$$R(F) = \frac{L(F)}{L(F_n)} = \prod_{i=1}^{n} np_i \tag{2.1}$$

For univariate data, such as the censored survival data, -2log$R$ follows a χ^2_1 distribution.

Owen (1988) studied an example of an empirical likelihood ratio for the univariate mean. Following the definition of the likelihood ratio R(F), the EL ratio function for the univariate mean is obtained as follows.

$$\mathcal{R}(\mu) = \max\left\{\prod_{i=1}^{n} np_i \mid \sum_{i=1}^{n} p_i x_i = \mu, p_i \geq 0, \sum_{i=1}^{n} p_i = 1\right\}$$

The method of Lagrange multipliers can be used to solve this optimization problem. See section 2.4.1 for a brief description of Lagrange multipliers. A hypothesis test for $\mu = \mu_0$ can then be constructed using the property that empirical likelihood admits a nonparametric version of Wilks's theorem. For $X_1, \ldots, X_n$, i.i.d. and $\mu = E[X_1]$, the EL ratio test statistic is:

$$T = \frac{\max L(p_1, \ldots, p_n)}{\max_{H_0} L(p_1, \ldots, p_n)} = \frac{(1/n)^n}{L(\mu_0)} = \prod_{i=1}^{n} \frac{1}{np_i(\mu_0)} = \prod_{1}^{n} (1 - \frac{\gamma}{n}(X_i - \mu_0))$$

where γ is the Lagrange multiplier for the constraint on the mean $\sum_{i=1}^{n} p_i x_i = \mu$ If we let $E[X_1] < \infty$ then under H_0 as $n \to \infty$

$$-2\log T = 2\sum_{i=1}^{n} \log(1 - \frac{\gamma}{n}(X_i - \mu_0)) \to \chi_1^2$$

and $(1 - \alpha)\%$ Confidence Interval is:

$$\left\{\mu | -2\log\{L(\mu)n^n\} < \chi^2_{1,1-\alpha}\right\} = \left\{\mu | \sum_{i=1}^{n} \log\{np_i(\mu)\} > -0.5\chi^2_{1,1-\alpha}\right\}$$

For multivariate X, -2log$T$ tends to a chi-square distribution with degrees of freedom equal to the dimension of X.

Owen (2001) noted the critical value for $-2\log\mathcal{R}$ should accordingly be $\chi_d^{2,1-\alpha}$, but Bartlett correction has been shown to reduce coverage error compared to using χ^2 or F calibrations. For some small data sets χ^2 calibration will not be reasonable and Bartlett correction will give little improvement and bootstrap calibration may obtain better results.

2.2 Estimating Equations

Estimating equations describe how parameters are related to corresponding statistics. They can be used to tell a model how to estimate the parameters use the sample data. This allows prior information about parameters to be added to the model. As shown by Qin & Lawless (1994), estimating equations are easily applied to Empirical Likelihood in the form of constraints on the likelihood. We define our estimating function $h(X, \theta))$ for i.i.d. random variables $X_1, \ldots, X_n$ be $\in \mathbb{R}^d$, a parameter $\theta \in \mathbb{R}^p$ and our vector-valued function

$h(X,\theta) \in \mathbb{R}^s$. Suppose that

$$E(h(X_1,\theta)) = 0$$

then θ can be estimated by:

$$\frac{1}{n}\sum_{i=1}^{n} h(X_i,\hat{\theta}) = 0 \tag{2.2}$$

and is known as the estimating equation. For example an estimating equation for the mean will be written as: $h(X,\theta) = X - \theta$, then equation (2.2) gives $\theta = \bar{X}$. Other examples include (i) the k-th moment $\theta = E(X_1^k)$ where $h(x,\theta) = x^k - \theta$, (ii) indicator function, $\theta = P(X_1 \in A)$ where $h(x,\theta) = I(x \in A) - \theta$. Note θ is the α–quantile if $h(x,\theta) = I(x \leq \theta) - \alpha$.

The estimating equations are unbiased if

$$E[\frac{1}{n}\sum_{i=1}^{n} h(X_i,\hat{\theta})] = 0$$

In the case F is a parametric family and h is the score function, $\hat{\theta}$ is the ordinary maximum likelihood estimator. When $s = p$, this is called the determined case, $\hat{\theta}$ can be uniquely determined by the estimating equation. For the underdetermined case, where $s < p$, solutions may form a $(p - s)$ dimensional set. However in some cases where $h(x,\theta)$ is a poor choice or we have an unfortunate distribution of F, θ may not be estimable. The overdetermined case, $s > p$, may not have a solution for θ from $E[(h(X,\theta)] = 0$ in some cases. The generalized method of moments which is very popular in econometrics looks for an approximate θ .

The empirical likelihood ratio with estimating equations $h(X,\theta)$ takes the form:

$$\mathcal{R}(\theta) = \max\left\{\prod_{i=1}^{n} np_i \mid \sum_{i=1}^{n} p_i h(X_i,\theta) = 0, p_i \geq 0, \sum_{i=1}^{n} p_i = 1\right\}$$

By using estimating equations this way, empirical likelihood can handle quantiles, likelihood-based estimating equations and robust estimators such as M-estimates. Empirical likelihood with estimating equations as constraints greatly increases the amount of problems empirical likelihood can handle.

2.3 Regression

Regression is one of the most widely used statistical methods. It is very useful as a tool for inference to describe relationships between variables. Though it is usually performed using parametric models it can also be applied to semi-parametric and nonparametric models. In addition to complete data, regression is able to handle censored or missing data. Empirical Likelihood can be used in conjunction with regression. Chen & Van Keilegom (2009) provide a review of empirical likelihood regression methods.

The simplest case of regression is simple linear regression, where data has two variables: y the dependent variable and x the independent variable. There are two parameters β_0 relating to y-intercept on an $x-y$ plane and β_1 relating to the slope. For a set of data of size n. The model can be written as follows:

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i, \quad i = 1, 2, ..., n.$$

where ϵ_i is the error term, the amount the i-th observation differs from its expected value. The population parameters of the model are estimated using the sample data. The fitted model is:

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$$

The estimates $\hat{y}_i$, $\hat{\beta}_0$ and $\hat{\beta}_1$ may be obtained using estimation techniques such as ordinary least squares. The residual $e_i = y_i - \hat{y}_i$ is the difference between the dependent variable predicted by the model with the true value. Similarly, for multiple regression, we have observations $(Y_i, \mathbf{X_i}), \mathrm{i} = 1, \ldots, n$, where $\mathbf{X_i} = (X_{i1}, \ldots X_{id})^T$, and covariate matrix $\mathbf{X} = (X_1, \ldots, X_d)^T$, with corresponding vector of coefficients $\beta = (\beta_1, \ldots \beta_d)^T \in \mathbb{R}^p$. We have the model:

$$Y = X_1^T \beta_1 + \ldots X_d^T \beta_d + \epsilon$$

and the fitted model:

$$\hat{Y} = \mathbf{X}\hat{\beta}$$

where $\hat{\beta}$ can be obtained by least squares:

$$\min \sum_{i=1}^{n} (Y_i - \mathbf{X_i}\beta)^2$$

The solution is:

$$\beta_{LS} = E(X^T X)^{-1} E(X^T Y)$$

The sample lease squares estimate for β_{LS} is

$$\hat{\beta_{LS}} = (\frac{1}{n} \sum_{i=1}^{n} X_i X_i^T)^{-1} (\frac{1}{n} \sum_{i=1}^{n} X_i Y_i)$$

This definition for β_{LS} is equivalent to

$$E(X^T (Y - X\beta_{LS})) = 0$$

and $\hat{\beta_{LS}}$ is equivalent to

$$\frac{1}{n} \sum_{i=1}^{n} X_i (Y_i - X_i^T \hat{\beta_{LS}}) = 0$$

The empirical likelihood ratio function for β becomes

$$\mathcal{R}(\beta) = \max \left\{ \prod_{i=1}^{n} np_i \mid \sum_{i=1}^{n} p_i h(X_i, \theta) = 0, p_i \geq 0, \sum_{i=1}^{n} p_i = 1 \right\}$$

where $h_i = X_i (Y_i - X_i^T \beta)$

Often our predictors are known, not random, so we are modelling the mean of Y given $X = x_i$, E$[Y|\mathbf{X_i} = x_i]$. The model requires several assumptions: The mean of the response variable E$[Y]$is a linear combination of the parameters β and the predictor variables X. Errors ϵ must be independent, have constant variance and follow a normal distribution.

Alterations to linear regression can allow it to handle more complex problems. Predictor variables can be transformed to have a linear relationship with the dependent variable. For example, a log or square root transformation when the relationship is non-linear or perhaps polynomial terms may be added. The effect of interactions between predictor variables may be added to the model for interpretation.

2.3.1 Generalized Linear Models

Generalized Linear models first developed by McCullagh (1984) is a popular extension which allows the response variables to have non-Normal distributions. They may be continuous, discrete or categorical. They require the data to come from an exponential family distribution. Following the notation of McCullagh & Nelder (1989), the distribution belongs to an exponential family if the probability density function is of the form:

$$f(y;\theta,\phi) = \exp\left\{\frac{y\theta - b(\theta)}{a(\phi)} + c(y,\phi)\right\}$$

for particular functions $a(.)$, $b(.)$ and $c(.)$. θ is canonical the parameter and ϕ is a nuisance parameter or dispersion parameter. Some examples are the Poisson distribution, binomial distribution and gamma distribution. The canonical GLM takes the form:

$$\theta = X^T\beta$$

where $\theta = g(\mu)$ (with $\mu = E[Y]$) and $g(.)$ is a known monotone function called the link function. Model parameters are updated using the iteratively reweighted least squares algorithm for maximum likelihood using the Newton-Raphson method or Fisher's scoring method.

Empirical likelihood handles regression by adding estimating equations in the form of constraints to the empirical likelihood ratio function. The empirical likelihood ratio function for β becomes

$$\mathcal{R}(\beta) = \max\left\{\prod_{i=1}^{n} np_i \mid \sum_{i=1}^{n} p_i h(X_i,\theta) = 0, p_i \geq 0, \sum_{i=1}^{n} p_i = 1\right\}$$

where $h_i(\beta) = x_i(Y_i - g(x_i^T\beta))$ and independent predictor variables are known $(X_i = x_i)$.

2.3.2 Density estimation and nonparametric regression

Consider the nonparametric regression model;

$$Y_i = m(X_i) + \epsilon_i$$

where $m(.)$ is a smooth function.

$m(X)$ may be estimated using nonparametric methods such as splines and kernel smoothing. Kernel methods are a popular choice with empirical likelihood since they can be written in the form of estimating equations and incorporated accordingly. Wand & Jones (1994) and Fan & Gijbels (1996) provide thorough overviews of kernel regression. Kernel regression uses local averaging to form a smooth function based on noisy observations. The Nadaraya–Watson estimator is the simplest kernel regression estimator for $m(X)$ defined by:

$$\hat{m}(x) = \frac{\sum_{i=1}^{n} K_h(x - X_i)Y_i}{\sum_{i=1}^{n} K_h(x - X_i)}$$

where $K_h(.)$ is a kernel function with bandwidth h. Commonly used kernel functions are the Gaussian kernel, Epanechnikov kernel and the Tri-cube kernel.

The above kernel estimator can be obtained by minimizing the following locally weighted sum of least squares:

$$\sum_{i=1}^{n} K_h(x - X_i)(Y_i - m(x))^2$$

with respect to m(x). This can be rewritten in the form of the estimating equation:

$$\sum_{min}^{max} K_h(x - X_i)(Y_i - m(x)) = 0$$

allowing it to be incorporated in empirical likelihood.

Semi-parametric regression is a mixture of parametric and nonparametric regression. So it may look like

$$Y_i = X_i^T \beta + g(Z_i) + \epsilon_i$$

for $i = 1, \dots, n$. Where we have the parametric part $X_i^T \beta$ and nonparametric part $g(Z_i)$ and $g(Z_i)$ is a smoothing function. This type of regression can also be incorporated into empirical likelihood as a constraint in the form of estimating equitations. Wang & Jing (2003) show that asymptotically, the empirical likelihood ratio tends to a χ_p^2 distribution where p is the number of parameters β. This is the same as the parametric case, showing the unknown nonparametric function has no effect on the asymptotic limit. This result allows us to obtain empirical likelihood confidence regions for β without estimating any variance.

EL can be combined with regression in many different ways allowing it to handle a wide range of cases. Improvements and alterations can be made to handle specific problems. Zhang & Gijbels (2003) explore an alteration called sieve empirical likelihood for cases where empirical likelihood cannot be used directly when an infinite dimensional parameter of interest is involved.

2.4 Constrained Optimizations

2.4.1 Equality constraints by Lagrange Multipliers

The method of Lagrange multipliers is very useful for finding the local maxima and minima of a function subject to equality constraints. Such as those problems we found in empirical likelihood. The method allows maximization problems to be rewritten in an unconstrained format which can then be solved. Consider the following optimization problem: Maximize $g(x)$ subject to $f(x) = 0$. Provided g and f are both continuously differentiable, the optimization problem can be rewritten in the form:

$$\mathcal{L}(x, \lambda) = g(x) + \lambda . f(x)$$

where λ is a Lagrange multiplier and $\mathcal{L}$ called the auxiliary function.

The method can be visualized by thinking in terms of contours. Here we have one contour when our constraint $f(x) = 0$ is satisfied, call this space C. We then consider contours of g, $g(x) = d$, for various values of d. If we move along the contour line $f = 0$, we look for points where g is at a maxima. At the maximum, g will be reduced if we continue to move along $f = 0$. This means the contour for g, at this point must be parallel to f.Since the gradient of a function is perpendicular to the contour lines, the contour lines of g and f are parallel if and only if the gradients of g and f are parallel. Therefore we require:

$$\nabla g = -\lambda \nabla f$$

for some λ, where λ is a constant which determines the magnitude in which the gradients are parallel. Note: This formula may also be written without the negative sign. Also, this is only a necessary condition for constrained optimization. The second case is the special case when $\lambda = 0$. As if g is level, then its gradient is zero, and setting $\lambda = 0$ is a solution regardless of g.

The interpretation of the Lagrange multiplier λ can be learned by noting the $\frac{\partial \mathcal{L}}{\partial f} = \lambda$. λ can be thought as the rate of change of $\mathcal{L}$ with respect to f.The minimum of a function can be found in the same way as maximizing $g(x)$ is equivalent to minimizing $-g(x)$.

The method of Lagrange multipliers can handle multiple constraints. This takes the form:

$f_j(x) = 0$ for $j = 1, \ldots, k$. The necessary condition is

$$\nabla f(x) = -\sum_{j=1}^{k} \lambda_j \nabla f_j(x)$$

where we have k Lagrange multipliers λ_j.

2.4.2 Inequality constraints by KKT condistions

The Karush–Kuhn–Tucker (KKT) conditions generalize the method of Lagrange multipliers, allowing it to be extended to inequality constraints (Kuhn & Tucker (1951)). This is a requirement of the constraints seen in empirical likelihood $p_i \geq 0$. KKT lists first order necessary conditions for a solution in nonlinear programming to be optimal, provided that some regularity conditions are satisfied. These conditions are listed below.

For a function we wish to maximize $g(x) = \sum c_i x_i$ constrained by m linear inequalities,

$$f_h(x) \geq 0, \quad h = 1, \ldots, m$$

Using the method of Lagrange multipliers we can write the constrained maximization problem in the form:

$$\phi(x, \lambda) = g(x) + \sum_{h=1}^{m} \lambda_h f_h(x)$$

where λ (vector of λ_h) is a set of m non-negative Lagrange multipliers. Denote partial derivatives at a particular point, (x^0, λ^0)

$$\phi_x^0 = \frac{\partial \phi(x^0, \lambda)}{\partial x_i}, \qquad \phi_\lambda^0 = \frac{\partial \phi(x, \lambda^0)}{\partial \lambda_h}$$

Here ϕ_x^0 is an n-vector and ϕ_λ^0 an m-vector.

Then, a particular vector x^0 maximizes $g(x)$ subject to the m constraints if, and only if, there is some vector λ^0 with nonnegative components such that the KKT conditions:

$$\phi_x^0 \leq 0, \quad \phi_x^{0\prime} x^0 = 0, \quad x^0 \geq 0$$

$$\phi_\lambda^0 \geq 0, \quad \phi_\lambda^{0\prime} \lambda^0 = 0, \quad \lambda^0 \geq 0$$

Note these are only necessary conditions, we also need the Second Order Sufficient

Conditions for assuring a solution is optimal.

Owen (1988) makes use of the method of Lagrange multipliers to maximize the empirical likelihood ratio function. For the example for the univariate mean:

$$\mathcal{R}(\mu) = \max\left\{\prod_{i=1}^{n} np_i \mid \sum_{i=1}^{n} p_i X_i = \mu, p_i \geq 0, \sum_{i=1}^{n} p_i = 1\right\}$$

$$L = \sum_{i=1}^{n} \log(np_i) - n\gamma \sum_{i=1}^{n} p_i(X_i - \mu) + \lambda(\sum_{i=1}^{n} p_i - 1))$$

Set partial derivatives to 0 and solve:

$$\frac{\partial L}{\partial p_i} = \frac{1}{p_i} - n\gamma(X_i - \mu) + \lambda = 0$$

By applying the KKT conditions,

$$0 = \sum_{i=1}^{n} p_i \frac{\partial L}{\partial p_i} = n + \lambda$$

so $\lambda = -n$

$$p_i = \frac{1}{n} \frac{1}{1 + \gamma(X_i - \mu)} \tag{2.3}$$

where γ can then be found by numerical search. However, this does not guarantee $p_i \geq 0$. Existing methods simply assume $p_i > 0$. This problem will occur when 0 is not in the convex hull of the estimating equation function $h(X_i, \theta)$. In the case of the mean $h(X_i, \theta) = X_i - \mu$. This positivity constraint problem for p_i's leads to a need for a new method which can guarantee $p_i \geq 0$. Chen et al. (2008) notices this problem and suggests a solution by adjusting the empirical likelihood function while maintaining the asymptotic properties of EL. We develop a method in section 3 using a Multiplicative Iterative (MI) algorithm (see section 2.5.3) for details, which can also guarantee $p_i \geq 0$.

To solve γ, Owen considers the estimating equation:

$$\sum_{i=1}^{n} p_i(X_i - \mu) = 0$$

By assuming $p_i > 0$ and so every $p_i < 1$, and by substituting (2.3) for p_i in the estimating equation we get

$$\frac{1}{n}\sum_{i=1}^{n}\frac{X_i-\mu}{1+\gamma(X_i-\mu)}$$

Now by noticing this function is monotonic in γ, a bracketing interval known to contain $\gamma(\mu)$ can be found and search conducted:

$$\frac{1-n^{-1}}{\mu-X_{(n)}}<\gamma(\mu)<\frac{1-n^{-1}}{\mu-X_{(1)}}$$

The algorithm then refines the interval until endpoints agree to a high degree e.g. 10^{-6}. The monotonic nature of γ suggests a bisection method may not be feasible. Safeguarded search methods such as Brent's method or a type of Newton method may be preferred.

The method of Lagrange multipliers can be combined with other methods to handle more complex problems which neither method can solve alone. Bellman (1956) demonstrates this by combining the method of Lagrange multipliers with the theory dynamic programming.

The optimization problem can be difficult to be solved by hand. Computers use algorithms to solve this problems. Strong computational power allows the use of large amounts of data and numerical solutions to optimization problems can be found within a reasonable amount of time.

2.4.3 Augmented Lagrange Methods

Augmented Lagrange methods were first discussed by Hestenes (1969) and Powell (1969) to suggest a method which could find the minimum of a function $f(x)$ subject to some constraints $g(x) = 0$. It differs from the method of Lagrange multipliers by adding a penalty term designed to enforce the constraint more strongly. Rockafellar (1973) and Powell (1973) extended this method to include inequality constraints.

For a constrained minimization problem:

$$\min f(x)$$

subject to

$$g_i(x)=0,\qquad i=1,\dots,m,$$

where f and all $g_i, i = 1,\dots,m$, are continuous functions, $X \in \mathbb{R}^n$.

The augmented Lagrangian method uses the following unconstrained objective:

$$\min \mathcal{L}_k(x) = f(x) - \sum_{i=1}^{m} \lambda_i g_i(x) + \frac{\alpha_k}{2} \sum_{i=1}^{m} g_i^2(x)$$

At each iteration of solving this problem α and λ are updated and used to re-solve the problem. λ is updated by the rule below.

$$\lambda_i \leftarrow \lambda_i - \alpha_k g_i(x_k)$$

where x_k is the solution to the unconstrained problem at the k-th step, i.e. $x_k = argmin \mathcal{L}_k(x)$.

The penalty coefficient α_k is increased in each iteration by some constant factor. A similar method called the penalty method requires less computational cost but requires the condition $\alpha \to \infty$, hence the augmented Lagrange method is preferred. Nocedal & Wright (2006) has been proven, when exact Lagrange multiplier vector λ^* is known, the solution of x is a strict minimizer of $\mathcal{L}_k(x, \lambda^*, \alpha)$ for all α sufficiently large. This suggests minimizing $\mathcal{L}_k(x, \lambda, \alpha)$ will give a good estimate of x even when α is not particularly close to infinity, even when we do not know λ^* provided that λ is a reasonable of λ^*.

2.5 Algorithms, nonlinear programming

Algorithms are used to solve optimization problems such as those posed by empirical likelihood. These types of problems can be complicated to calculate as they can include statistics defined through estimating equations, nuisance parameters and side information. There are many different methods for computing these problems with advantages and disadvantages for each. Some methods may be better suited for certain situations. There is often a trade-off between speed and reliability. Luenberger & Ye (2008) provides a wide overview covering the concepts of optimization techniques for linear and nonlinear problems. We consider Newtons' method, the Levenberg-Marquardt algorithm and a Multiplicative Iterative Algorithm.

For function $\mathcal{L}(\theta)$ we wish to maximize, an iterative algorithm has the form:

$$\theta^{(k+1)} = \theta^{(k)} + \tau^{(k)} \tag{2.4}$$

where $\theta^{(k)}$ is the estimate for θ at the k-th iteration and $\tau^{(k)}$ is the increment, the amount which θ changes at the k-th iteration.

Iterative methods require an initial starting value $\theta^{(0)}$ to be chosen for the iterative formula to start to update from. The increment is determined with a formula defined by the algorithm method. This equation is continually updated until convergence which can be defined by some convergence criterion such as absolute difference $|\theta^{(k+1)} - \theta^{(k)}| < \epsilon$. When the initial value $\theta^{(0)}$ is poorly selected, this algorithm may not converge. A relaxation parameter $\omega \in (0,1)$ called the step-size can alter the increment to prevent this. This parameter can be determined by a line search method.

2.5.1 Newton's Method

Newton's method is developed from the Taylor series expansion of $\mathcal{L}(\theta)$ at $\theta^{(k)}$:

$$\mathcal{L}(\theta) = \mathcal{L}(\theta^{(k)}) + (\theta - \theta^{(k)})^T \frac{\partial \mathcal{L}(\theta^{(k)})}{\partial \theta} + \frac{1}{2}(\theta - \theta^{(k)})^T \frac{\partial^2 \mathcal{L}(\theta^{(k)})}{\partial \theta \partial \theta^T}(\theta - \theta^{(k)}) + \ldots$$

where derivatives are evaluated at $\theta^{(k)}$, $\frac{\partial \mathcal{L}(\theta^{(k)})}{\partial \theta} = \frac{\partial \mathcal{L}(\theta)}{\partial \theta}|_{\theta=\theta^(k)}$, $\frac{\partial^2 \mathcal{L}(\theta^{(k)})}{\partial \theta \partial \theta^T} = \frac{\partial^2 \mathcal{L}(\theta)}{\partial \theta \partial \theta^T}|_{\theta=\theta^(k)}$. When $\theta^{(k)}$ is close to the maximum likelihood estimator $\hat{\theta}$, higher order terms ≈ 0. Thus

$$\frac{\partial \mathcal{L}(\hat{\theta})}{\partial \theta} \approx \frac{\partial \mathcal{L}(\theta^{(k)})}{\partial \theta} + \frac{\partial^2 \mathcal{L}(\theta^{(k)})}{\partial \theta \partial \theta^T}(\hat{\theta} - \theta^{(k)})$$

As we are solving $\frac{\partial \mathcal{L}(\hat{\theta})}{\partial \theta} = 0$, the above equation becomes

$$\hat{\theta} \approx \theta^{(k)} - \left[\frac{\partial^2 \mathcal{L}(\theta^{(k)})}{\partial \theta \partial \theta^T}\right]^{-1} \frac{\partial \mathcal{L}(\theta^{(k)})}{\partial \theta}$$

This is in the form of equation (2.4).

2.5.2 Levenberg-Marquardt Algorithm

The Levenberg-Marquardt algorithm (Marquardt (1963)) also known as the damped least squares method, interpolates between the Taylor series method and the gradient descent methods to overcome problems suffered by each individually. The updating algorithm has the form

$$\theta^{(k+1)} = \theta^{(k)} + \mathbb{A}^{-1\,(k)} \frac{\partial \mathcal{L}(\theta^{(k)})}{\partial \theta}$$

where

$$\mathbb{A} = \sum_{i=1}^{n} (B_i B_i^T + \delta \text{diag}(B_i B_i^T))$$

with B_i defined from

$$\frac{\partial \mathcal{L}(\theta)}{\partial \theta} = \sum_{i=1}^{n} \frac{\partial \mathcal{L}_i(\theta)}{\partial \theta} = \sum_{i=1}^{n} B_i$$

where δ is a damping factor which is adjusted at each iteration. If the change in $\mathcal{L}(\theta)$ is large, δ can be reduced, bringing the algorithm closer to a Guass-Newton algorithm. If the change is small δ can be increased giving a larger step alike the gradient descent method.

2.5.3 Multiplicative Iterative Algorithm

Multiplicative iterative algorithms (Ma (2006)) are an updating method which are particularly useful for a parameter which is strictly positive. The multiplicative iterative algorithm has the form

$$\theta_j^{(k+1/2)} = \theta_j^{(k)} \frac{[\frac{\partial \mathcal{L}(\theta^{(k)})}{\partial \theta_j}]^+}{[\frac{\partial \mathcal{L}(\theta^{(k)})}{\partial \theta_j}]^-}$$

where $[a]^+ = \max(a, 0)$ and $[a]^- = \min(a, 0)$ such that $a = [a]^+ + [a]^-$ This can be rearranged to the common format seen earlier.

$$\theta^{(k+1)} = \theta^{(k)} + \omega^{(k)} \text{diag}\left(\frac{\theta_j}{[\frac{\partial \mathcal{L}(\theta^{(k)})}{\theta_j}]^-} \right) \frac{\partial \mathcal{L}(\theta^{(k)})}{\partial \theta}$$

where $\omega^{(k)} \in (0, 1)$ is the line search step size and can be found using Armijo rule.

2.5.4 Line search: Armijo Rule

Line search methods work to determine a step-size which can increase reliability in finding convergence from algorithms but sacrifices computation time. The Armijo rule (Armijo et al. (1966)) is an inexact line search method which is quick and has sufficient accuracy. For the function we wish to maximize $L(\theta)$ we have:

$$\theta^{(k+1)} = \theta^{(k)} + \omega^{(k)} \tau^{(k)}$$

This is updated as us usual for steps $\mathcal{L}(\theta^{(k+1)}) > \mathcal{L}(\theta^{(k)})$. When $\mathcal{L}(\theta^{(k+1)}) < \mathcal{L}(\theta^{(k)})$ the step size ω is decreased by some factor $\sigma > 0$ to become $\frac{1}{\sigma}\omega$ and $\mathcal{L}(\theta^{(k+1)}) > \mathcal{L}(\theta^{(k)})$ is rechecked. This repeats until we reach convergence.

3

Method

3.1 Augmented Lagrange on Empirical Likelihood

This method stems from the idea of enforcing the p_i positivity constraint by updating p_i's using a Multiplicative Iterative algorithm. Using an augmented Lagrange method should improve convergence times by penalizing ill fitting solutions. Estimating equations are incoporated into empirical likelihood in the form of constraints as follows.

When we have k estimating equations

$$\mathbf{h}(X_i, \theta) = \begin{bmatrix} h_1(X_i, \theta) \\ \vdots \\ h_k(X_i, \theta) \end{bmatrix}$$

the empirical likelihood ratio has the form:

$$\mathcal{R}(\theta) = \max_{p_i, \theta} \left\{ \prod_{i=1}^{n} p_i \mid \sum_{i=1}^{n} p_i \mathbf{h}(X_i, \theta) = 0, p_i \geq 0, \sum_{i=1}^{n} p_i = 1 \right\} \tag{3.1}$$

To solve this we take the log-likelihood and apply an augmented Lagrange method.

$$\mathcal{L}_\alpha = \sum_{i=1}^{n} \log p_i - \lambda(1 - \sum_{i=1}^{n} p_i) - \gamma^T \sum_{i=1}^{n} p_i \mathbf{h}(X_i, \theta) - \frac{\alpha}{2} || \sum_{i=1}^{n} p_i \mathbf{h}(X_i, \theta)||^2 \tag{3.2}$$

where $\gamma^T = \left[\gamma_1, \ldots, \gamma_k\right]$, $||\mathbf{a}||$ is the Euclidean norm of $\mathbf{a} = \sqrt{a_1^2 + \ldots + a_n^2}$ and $\alpha > 0$.

To solve this we differentiate first with respect to p_i. Then use the KKT conditions to solve λ. The solution can be directly substituted back into our optimization equation. We can then update our p_i estimates using a multiplicative iterative algorithm.

$$\frac{\partial \mathcal{L}_\alpha}{\partial p_i} = 1/p_i + \lambda - \gamma^T \mathbf{h}(X_i, \theta) - \alpha \{ \sum_{j=1}^{k} (\sum_{t=1}^{n} p_t h_j(X_t, \theta) h_j(X_i, \theta))\}$$

For optimal λ set

$$\frac{\partial \mathcal{L}_\alpha}{\partial \lambda} = 0$$

we get:

$$\sum_{i=1}^{n} p_i = 1$$

Karush–Kuhn–Tucker conditions state: For $p_i \geq 0$

$$\frac{\partial \mathcal{L}_\alpha}{\partial p_i} = 0 \quad \text{if} \quad p_i > 0 \qquad \text{and} \qquad \frac{\partial \mathcal{L}_\alpha}{\partial p_i} < 0 \quad \text{if} \quad p_i = 0$$

Thus $p_i \frac{\partial \mathcal{L}_\alpha}{\partial p_i} = 0$ follows for all $p_i \geq 0$ and so the sum will also be 0:

$$\sum_{i=1}^{n} p_i \frac{\partial \mathcal{L}_\alpha}{\partial p_i} = 0$$

We can solve this for λ as

$$\sum_{i=1}^{n} p_i \frac{\partial \mathcal{L}_\alpha}{\partial p_i} = \sum_{i=1}^{n} 1 + \sum_{i=1}^{n} \lambda p_i - \gamma^T \sum_{i=1}^{n} p_i \mathbf{h}(X_i, \theta) - \alpha \{ \sum_{j=1}^{k} (\sum_{t=1}^{n} p_t h_j(X_t, \theta) \sum_{i=1}^{n} p_i h_j(X_i, \theta))\}.$$

By substituting $\sum_{i=1}^{n} p_i = 1$ we get

$$\lambda = -n + \gamma^T \sum_{i=1}^{n} p_i \mathbf{h}(X_i, \theta) + \alpha \{ \sum_{j=1}^{k} (\sum_{t=1}^{n} p_t h_j(X_t, \theta) \sum_{i=1}^{n} p_i h_j(X_i, \theta))\}$$

and noting the last term can be simplified to

$$\alpha\{\sum_{j=1}^{k}(\sum_{t=1}^{n}p_t h_j(X_t,\theta)\sum_{i=1}^{n}p_i h_j(X_i,\theta))\} = \alpha\{\sum_{j=1}^{k}(\sum_{t=1}^{n}p_t h_j(X_t,\theta))^2\} = \alpha||\sum_{t=1}^{n}p_t\mathbf{h}(X_t,\theta)||^2$$

so we get

$$\lambda = -n + \gamma^T\sum_{i=1}^{n}p_i\mathbf{h}(X_i,\theta) + \alpha||\sum_{t=1}^{n}p_t\mathbf{h}(X_t,\theta)||^2 \tag{3.3}$$

By substituting this λ we get:

$$\frac{\partial\mathcal{L}_\alpha}{\partial p_i} = 1/p_i - n + \gamma^T\sum_{i=1}^{n}p_i\mathbf{h}(X_i,\theta) + \alpha||\sum_{t=1}^{n}p_t\mathbf{h}(X_t,\theta)||^2 - \gamma^T\mathbf{h}(X_i,\theta) - \alpha\{\sum_{j=1}^{k}(\sum_{t=1}^{n}p_t h_j(X_t,\theta)h_j(X_i,\theta))\}$$

p_i can now be updated using the MI algorithm,

$$p_i^{(k+1/2)} = p_i^{(k)}\frac{[\frac{\partial\mathcal{L}_\alpha^{(k)}}{\partial p_i}]^+}{-[\frac{\partial\mathcal{L}_\alpha^{(k)}}{\partial p_i}]^-} \tag{3.4}$$

where $[a]^+ = \max(a,0)$ and $[a]^- = \min(a,0)$ such that $a = [a]^+ + [a]^-$, $p_i^{(k)}$ denotes the solution for p_i on the k-th iteration and $\frac{\partial\mathcal{L}_\alpha^{(k)}}{\partial p_i} = \frac{\partial\mathcal{L}_\alpha}{\partial p_i}|_{p_i=p_i^{(k)}}$ For this we separate $\frac{\partial\mathcal{L}_\alpha}{\partial p_i}$ into positive and negative parts:

$$\begin{aligned}[\frac{\partial\mathcal{L}_\alpha}{\partial p_i}]^+ = 1/p_i + [\gamma^T\sum_{i=1}^{n}p_i\mathbf{h}(X_i,\theta)]^+ + \alpha||\sum_{t=1}^{n}p_t\mathbf{h}(X_t,\theta)||^2 - [\gamma^T\mathbf{h}(X_i,\theta)]^- \\ - \alpha[\{\sum_{j=1}^{k}(\sum_{t=1}^{n}p_t h_j(X_t,\theta)h_j(X_i,\theta))\}]^-\end{aligned}$$

and

$$[\frac{\partial\mathcal{L}_\alpha}{\partial p_i}]^- = -n + [\gamma'\sum_{i=1}^{n}p_i\mathbf{h}(X_i,\theta)]^- - [\gamma'\mathbf{h}(X_i,\theta)]^+ - \alpha[\{\sum_{j=1}^{k}(\sum_{t=1}^{n}p_t h_j(X_t,\theta)h_j(X_i,\theta))\}]^+$$

This forms our MI algorithm updating method for p_i given in equation (3.4).

One full iteration of the MI algorithm follows the formula:

$$p_i^{(k+1)} = p_i^{(k)} + w^{(k)}(p_i^{(k+1/2)} - p_i^{(k)}) \tag{3.5}$$

where $\omega^{(k)} \in (0,1)$ is the line search step size and can be found using Armijo rule. This guarantees $\mathcal{L}_\alpha(p_i^{(k+1)}) \geq \mathcal{L}_\alpha(p_i^{(k)})$.

Then we can update our θ using a number of options. We can first try to solve $\frac{\partial \mathcal{L}_\alpha}{\partial \theta} = 0$ and update accordingly. If the solution to this is hard to obtain we can use a Newton or Quasi-Newton method. A general formula for $\frac{\partial \mathcal{L}_\alpha}{\partial \theta}$ is provided below.

$$\begin{aligned}\frac{\partial \mathcal{L}_\alpha}{\partial \theta} &= \gamma^T \sum_{i=1}^{n} p_i \frac{\partial \mathbf{h}(X_i, \theta)}{\partial \theta} - \alpha(\sum_{i=1}^{n} p_i \mathbf{h}(X_i, \theta))^T \sum_{i=1}^{n} \frac{\partial \mathbf{h}(X_i, \theta)}{\partial \theta} \\ &= (-\sum_{i=1}^{n} p_i \frac{\partial \mathbf{h}(X_i, \theta)}{\partial \theta})^T (\gamma + \alpha \sum_{i=1}^{n} p_i \mathbf{h}(X_i, \theta)) \qquad (3.6)\end{aligned}$$

Some methods require the second derivative:

$$\begin{aligned}\frac{\partial^2 \mathcal{L}_\alpha}{\partial \theta \partial \theta^T} &= -\gamma^T \sum_{i=1}^{n} p_i \frac{\partial^2 h(X_i, \theta)}{\partial \theta \partial \theta^T} - \alpha(\sum_{i=1}^{n} p_i \frac{\partial h(X_i, \theta)}{\partial \theta})^T \sum_{i=1}^{n} p_i \frac{\partial h(X_i, \theta)}{\partial \theta} \\ &- \alpha(\sum_{i=1}^{n} p_i h(X_i, \theta))^T \sum_{i=1}^{n} p_i \frac{\partial^2 h(X_i, \theta)}{\partial \theta \partial \theta^T} \qquad (3.7)\end{aligned}$$

The second derivative may be difficult to calculate. To avoid this difficultly, we may use the Levenberg-Marquardt algorithm to update θ which only requires $\frac{\partial \mathcal{L}_\alpha}{\partial \theta}$

$$\theta^{(k+1)} = \theta^{(k)} + \omega_2^{(k)} \mathbb{A}^{-1\,(k)} \frac{\partial \mathcal{L}_\alpha(p_i^{(k+1)}, \theta^{(k)})}{\partial \theta} \qquad (3.8)$$

where

$$\mathbb{A} = \sum_{i=1}^{n} (B_i B_i^T + \delta \text{diag}(B_i B_i^T))$$

with

$$B_i = (p_i \frac{\partial \mathbf{h}(X_i, \theta)}{\partial \theta})^T (\gamma + \alpha \sum_{i=1}^{n} p_i \mathbf{h}(X_i, \theta))$$

We also need to obtain and update an estimate for γ. γ can be updated by a number of ways, we use the standard method to update after we have found p_i and θ

$$\gamma^{(k+1)} = \gamma^{(k)} + \alpha \sum_{i=1}^{n} p_i^{(k+1)} \mathbf{h}(X_i, \theta^{(k+1)}) \qquad (3.9)$$

3.2 Asymptotic Properties

The asymptotic properties of Empirical Likelihood has been stated by Owen (2001). Our method does not alter this Empirical Likelihood function, therefore the asymptotic properties of EL remain intact. Of particular importance are the results of Qin & Lawless (1994) which proved the asymptotic properties for EL with estimating equation constraints.

Theorem 1 *The empirical likelihood ratio statistic for testing $H_0 : \theta = \theta_0$ is*

$$-2\mathcal{L}(\theta_0) - 2\mathcal{L}(\hat{\theta}) \qquad \sim \chi^2_p \qquad as \;\; n \to \infty$$

when H_0 is true, where p is the dimension of θ and $\mathcal{L}(\theta)$ is the log-likelihood.

Corollary 2

$$-2\mathcal{L}(\theta_0) - 2\mathcal{L}(\hat{\theta}) \qquad \sim \chi^2_{(r-p)} \qquad as \;\; n \to \infty$$

for r estimating equations, if $E[\mathbf{h}(X_i, \theta)] = 0$.

Corollary 3 *Let $\theta^T = (\theta_1, \theta_2)^T$, where $\theta_1 and \theta_2$ are $q \times 1$ and $(p-q) \times 1$ vectors, respectively. For $H_0 : \theta = \theta_1^0$,*

$$-2\mathcal{L}(\theta_1^0, \hat{\theta_2^0}) - 2\mathcal{L}(\hat{\theta_1}, \hat{\theta_2}) \qquad \sim \chi^2_q \qquad as \;\; n \to \infty]$$

under H_0, where $\hat{\theta_2^0}$ minimizes $log\mathcal{L}(\theta_1^0, \theta_2)$ with respect to θ_2.

See Qin & Lawless (1994) for full details.

3.3 Examples

This section demonstrates how to apply our method to specific situations for the overdetermined case. We show how to include these constraints in the empirical likelihood maximization function in the form of estimating equations. The first case is the scenario where we wish to apply empirical likelihood with linear regression to obtain parameter estimates for a set of data. In our example we wish to test the hypothesis of a particular covariate coefficient in a linear regression, this alters the problem to an overdetermined case when we substitute in the parameter under the null hypothesis. The second example shows how to apply estimating equations of moments for parameters we wish to estimate. The simplest scenario for the overdetermined case is 2 estimating equations with one parameter. We use the example where the mean is equal to the variance such as for a Poisson distribution.

3.3.1 Example: Linear regression with test for a particular covariate coefficient

For linear regression we have: $Y = X\beta + \epsilon$ where

$$Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

$$X = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix} = \begin{bmatrix} x_{11} \dots x_{1p} \\ \vdots \ddots \vdots \\ x_{n1} \dots x_{np} \end{bmatrix}$$

$$\beta = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_p \end{bmatrix}, \epsilon = \begin{bmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

We can write the Empirical Likelihood maximization function subject to constraints.

$$\mathcal{R}(\theta) = \max_{p_i,\theta} \left\{ \prod_{i=1}^{n} p_i \mid \sum_{i=1}^{n} p_i, h_i(X_i, y_i, \beta) = 0, p_i \geq 0, \sum_{i=1}^{n} p_i = 1 \right\}$$

where $h_i(X_i, y_i, \beta) = x_i(y_i - x_i^T \beta)$ This is in the same form as (3.1) and so we can rewrite this as a an augmented Lagrange as in (3.2).

$$\mathcal{L}_\alpha = \sum_{i=1}^{n} \log p_i - \lambda(1 - \sum_{i=1}^{n} p_i) - \underline{\gamma^T} \sum_{i=1}^{n} p_i h_i(X_i, Y_i, \beta) - \frac{\alpha}{2} || \sum_{i=1}^{n} p_i h_i(X_i, y_i \beta)||^2$$

We may be interested in testing a hypothesis about particular values of β. For example we wish to test: $H_0 : \beta_1 = \beta_{10}$ We can substitute this value of β_{10} into our estimating equation matrix.

To incorporate the assumption H_0, $\beta_1 = \beta_{10}$ into our estimating equation matrix we define: X_r as the X covariates related to coefficient β we are not testing for. e.g. All x_i other than x_1. X_{nr} as the X variables related to coefficient we are testing for: e.g. β_1 and x_1. Now

$$h_i(X_i, y_i, \theta) = x_i(y_i - x_{nr,i}^T\beta_{10} - x_{r,i}\beta_*)$$

where β_* is a vector of β without β_1. We update p_i using the MI Algorithm equations (3.4) and (3.5).

To estimate β_*, we need $\frac{\partial \mathcal{L}_\alpha}{\partial \beta_*}$. We know the first derivative of $\mathcal{L}$ from equation (3.6). Note: We are testing for β_{10} under H_0 so do not need to estimate β_1, we substitute this value into our equation. Here we have

$$\frac{\partial \mathcal{L}_\alpha}{\partial \beta_*} = (-\sum_{i=1}^{n} p_i \frac{\partial h_i(X_i, y_i, \beta)}{\partial \beta_*})^T(\gamma + \alpha\sum_{i=1}^{n} p_i h_i(X_i, y_i, \beta_*))$$

We can simply solve $\frac{\partial \mathcal{L}_\alpha}{\partial \beta_*} = 0$ to update β_1 as it is linear in terms of β_1. We have:

$$\frac{\partial h_i(X_i, y_i, \beta)}{\partial \beta_*} = -X_i^T X_{r,i}$$

By substitution:

$$\begin{aligned}\frac{\partial \mathcal{L}_\alpha}{\partial \beta} &= (\sum_{i=1}^{n} p_i X_i^T X_{r,i})^T(\gamma + \alpha\sum_{i=1}^{n} p_i Xi^T(y_i - X_{r,i}\beta_*)) \\ &= (\sum_{i=1}^{n} p_i X_i^T X_{r,i})^T(\gamma + \alpha\sum_{i=1}^{n} p_i Xi^T y_i - \alpha\sum_{i=1}^{n} p_i X_i^T X_{r,i}\beta_*) = 0\end{aligned}$$

Therefore

$$\alpha(\sum_{i=1}^{n} p_i X_i^T X_{r,i})^T(\sum_{i=1}^{n} p_i X_i^T X_{r,i}\beta_*) = (\sum_{i=1}^{n} p_i X_i^T X_{r,i})^T(\gamma + \alpha\sum_{i=1}^{n} p_i Xi^T y_i)$$

$$\beta_* = [\alpha(\sum_{i=1}^{n} p_i X_i^T X_{r,i})^T(\sum_{i=1}^{n} p_i X_i^T X_{r,i})]^{-1}(\sum_{i=1}^{n} p_i X_i^T X_{r,i})^T(\gamma + \alpha\sum_{i=1}^{n} p_i Xi^T y_i)$$

The multipliers vector γ is updated simply by equation (3.9)

$$\gamma^{(k+1)} = \gamma^{(k)} + \alpha\sum_{i=1}^{n} p_i^{(k+1)} h_i(X_i, y_i, \beta_1^{(k+1)})$$

3.3.2 Example: Estimating equations on Poisson mean

For a set of data we suspect belongs to a Poisson distribution, we can use our knowledge of μ and σ^2 to form estimating equations to improve our estimates of p_i and θ. We have the empirical likelihood

$$\mathcal{R}(\theta) = \max_{p_i,\theta} \left\{ \prod_{i=1}^{n} p_i \mid \sum_{i=1}^{n} p_i, h_i(X_i, \theta) = 0, p_i \geq 0, \sum_{i=1}^{n} p_i = 1 \right\}$$

where we have estimating equations $h_1(X_i, \theta) = X_i - \mu$ and $h_2(X_i, \theta) = X_i^2 - \mu - \sigma^2$ where $\mu = \theta$ and $\sigma^2 = \theta$. Thus

$$\mathbf{h}(X_i, \theta) = \begin{bmatrix} h_1(X_i, \theta) \\ h_2(X_i, \theta) \end{bmatrix} = \begin{bmatrix} X_i - \theta \\ X_i - \theta - \theta^2 \end{bmatrix}$$

This is in the general form, so p_i can be updated using MI Algorithm equations (3.4) and (3.5) In order to update θ we try to solve $\frac{\partial \mathcal{L}_\alpha}{\partial \theta} = 0$ from equation (3.6)

$$\frac{\partial \mathcal{L}_\alpha}{\partial \theta} = (-\sum_{i=1}^{n} p_i \frac{\partial \mathbf{h}(X_i, \theta)}{\partial \theta})^T (\gamma + \alpha \sum_{i=1}^{n} p_i \mathbf{h}(X_i, \theta))$$

where

$$\frac{\partial \mathbf{h}(X_i, \theta)}{\partial \theta} = \begin{bmatrix} \frac{\partial h_1}{\partial \theta} \\ \frac{\partial h_2}{\partial \theta} \end{bmatrix} = \begin{bmatrix} -1 \\ -1 - 2\theta \end{bmatrix}$$

which means $\frac{\partial \mathcal{L}_\alpha}{\partial \theta}$ is a cubic in terms of θ which can be solved. Alternatively, we can use equation (3.8), the Levenberg-Marquardt algorithm to update θ. This algorithm is chosen since it does not require calculation of the second derivative.

$$\theta^{(k+1)} = \theta^{(k)} + \omega_2^{(k)} \mathbb{A}^{-1\,(k)} \frac{\partial \mathcal{L}_\alpha(p_i^{(k+1)}, \theta^{(k)})}{\partial \theta}$$

where

$$\mathbb{A} = \sum_{i=1}^{n} (B_i B_i^T + \delta \text{diag}(B_i B_i^T))$$

with

$$B_i = (p_i \begin{bmatrix} -1 \\ -1 - 2\theta \end{bmatrix})^T (\gamma + \alpha \sum_{i=1}^{n} p_i \begin{bmatrix} -1 \\ -1 - 2\theta \end{bmatrix})$$

The multipliers vector γ is updated simply by equation (3.9)

$$\gamma^{(k+1)} = \gamma^{(k)} + \alpha \sum_{i=1}^{n} p_i^{(k+1)} \begin{bmatrix} X_i - \theta^{(k+1)} \\ X_i - \theta^{(k+1)} - (\theta^{(k+1)})^2 \end{bmatrix}$$

4

Simulation

Simulations for the examples in section 3.3 have been conducted to demonstrate our method works. MATLAB program was used, the relevant codes are attached in the appendix. In this section we explain the type of data used, the choices of initial starting values of parameters we wish to estimate and discuss the results and accuracy of our simulation. We obtain estimates for p_i and parameter θ for each simulation as well convergence information and likelihood values. In the simulation, 500 repetitions have been run with a single sample size $n = 30$ for each example.

4.1 Linear regression with test for a particular covariate coefficient

The linear regression example from section 3.3.1 was designed to show how our method can handle regression being incorporated into the empirical likelihood while also testing a hypothesis. Data was generated so that

$$y_i = x_i b + \epsilon_i \qquad\qquad i = 1, \ldots, 30$$

for known true β, $b = [2, 3, 5]^T$, $\epsilon \sim 10N(0, 1)$, $x_i = [x_{i0}^T, x_{i1}^T, x_{i2}^T]$, where x_0 is a vector of 1's relating to the intercept coefficient. x_1 was generated from a Binomial distribution with 30 trials and a 0.2 probability of success and x_2 was generated from a uniform$(-5, 5)$ distribution.

The method requires initial starting values for p_i, β and γ to be chosen. $p_i^{(0)} = 1/n$ is chosen as it is the maximum likelihood estimate of the empirical likelihood estimator. It puts equal probability mass $1/n$ on the n observed values $y_1, y_2, \ldots y_n$. We used the least squares solution as a starting point for β. With the hypothesis test $\beta_{10} = 2$ this becomes

$$\beta^{(0)} = (X^T X_r)^{-1} X^T Y - 2X_{nr}$$

where X_r is $[X_2, X_3]$ the X covariates for coefficients we are not testing, β_2 and β_3. X_{nr} is $[X_1]$, the covariate for which we are testing the coefficient of β_1. Initial value for γ was chosen to be 0 i.e. $\gamma^{(0)} = 0$

Box plots have been used to check if our constraints are satisfied. As shown in figure 4.1 $\sum_{i=1}^n p_i \approx 1$ in most cases with some repetitions having a $\sum_{i=1}^n p_i >> 1$. Note: we have suggested an adjustment to our method to more strongly account for this constraint but due to time limitations it has yet to be implemented. Our constraint on the estimating equations $\sum_{i=1}^n p_i h_i = 0$ has been much better maintained, though there are still some repetitions where the constraint is not satisfied.

We also wish to compare our β estimates $\hat{\beta}_2$ and $\hat{\beta}_3$ with the true values $b_2 = 3$ and $b_3 = 5$. Figures 4.3, 4.4 , 4.5,4.6, 4.7 and 4.8 show the distribution of our estimates. We notice both $\hat{\beta}_2$ $\hat{\beta}_3$ have underestimated the true parameter value, with $\hat{\beta}_2$ being close to 2.4 and $\hat{\beta}_3$ close to 4. $\hat{\beta}_2$ appears slightly right skewed while $\hat{\beta}_3$ appears quite symmetric.

The likelihood plot in figure 4.9 shows how the method with the likelihood converging towards a maximum value.

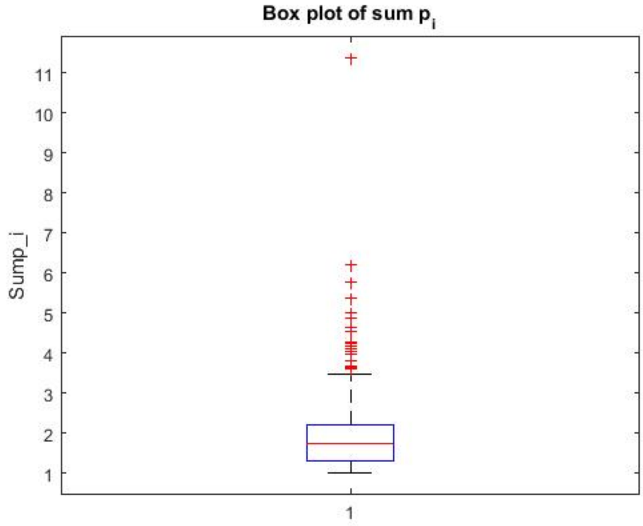

Figure 4.1: Box plot of $\sum_{i=1}^{n} p_i$. The median is close to 2, with some very large outliers.

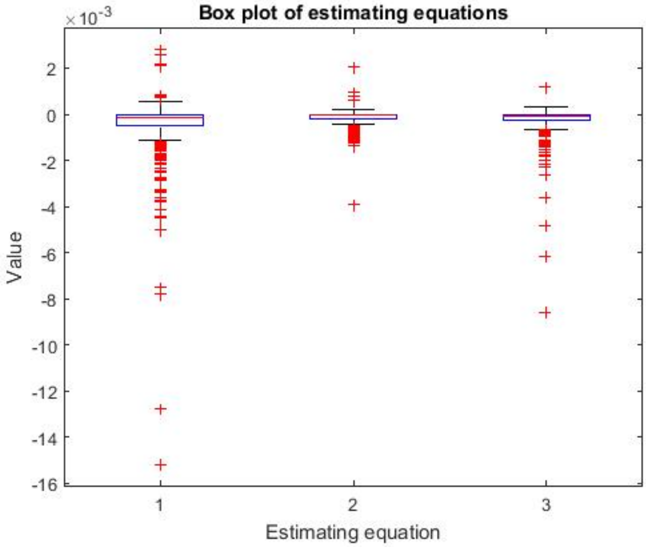

Figure 4.2: Box plot of the estimating equations, $\sum_{i=1}^{n} p_i h_i$. The medians are very close to 0 with low variance.

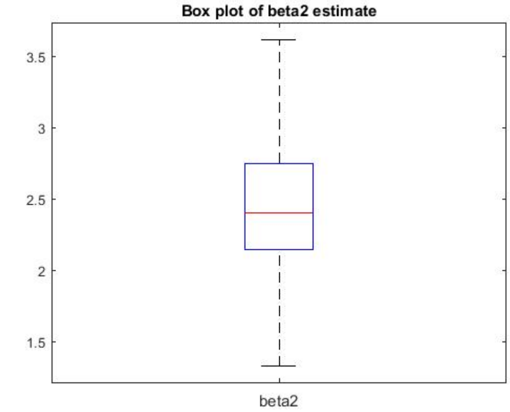

Figure 4.3: Box plot of $\hat{\beta}_2$. The median is slightly lower than 2.4.

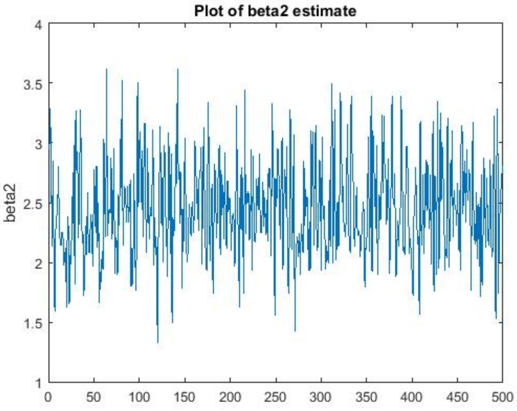

Figure 4.4: Plot for β_2 estimate versus repetition.

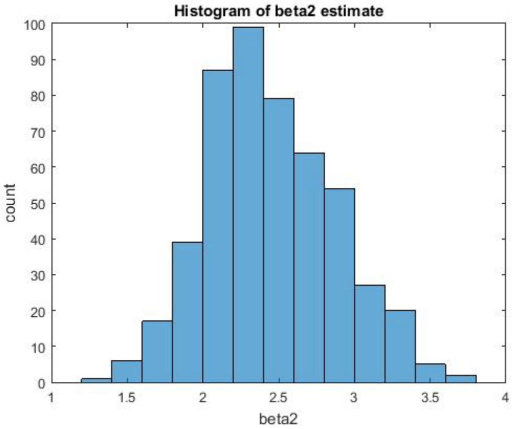

Figure 4.5: Histogram of the $\hat{\beta}_2$. The distribution is slightly right skewed.

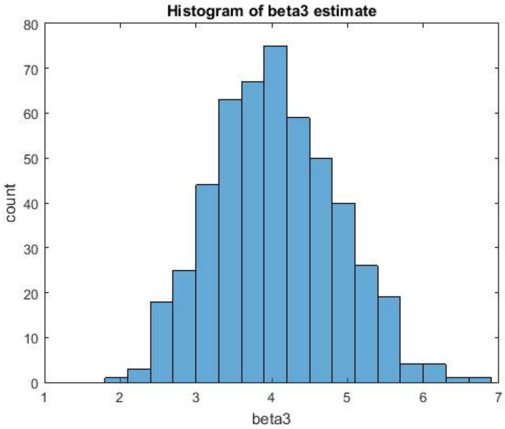

Figure 4.6: Histogram of the $\hat{\beta}_3$. The distribution is roughly symmetric.

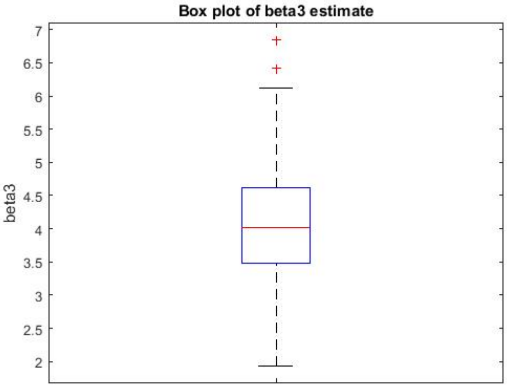

Figure 4.7: Box plot of the $\hat{\beta}_3$. Median is close to 4.

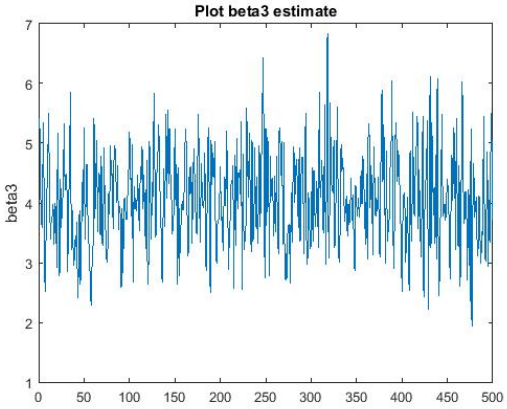

Figure 4.8: Plot of the $\hat{\beta}_3$ versus repetition.

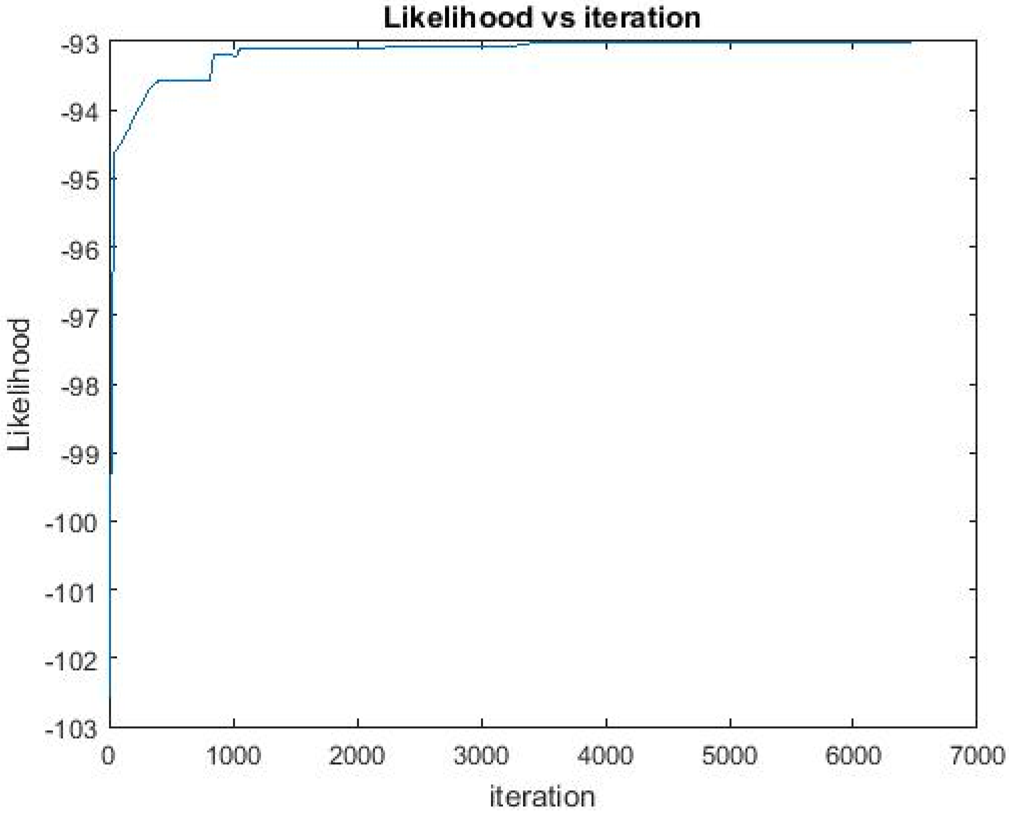

Figure 4.9: Likelihood until convergence for the first repetition.

4.2 Estimating Equations on Poisson mean

The example from section 3.3.2 was designed to show how our method incorporates estimating equations into the empirical likelihood in an overdetermined case. The simplest case is that of two estimating equations for one parameter. Data from a Poisson distribution would be suitable here as we have one parameter for both mean and variance. Accordingly data was generated from a Poisson distribution with rate $\lambda = 3$ i.e. $X \sim \text{Poisson}(3)$. Initial starting values for p_i and γ are the same as in the previous example for the same reasons. The sample mean was used as a the initial starting value for θ i.e. $\theta^{(0)} = \text{mean}(X)$.

Box plots 4.10 and 4.11 evaluate the success of our constraints. We can see these constraints are more strongly satisfied here than in the previous example but we still have some problem points. From figure 4.10 we can see there are some of repetitions with $\sum_{i=1}^{n} p_i > 1$. Figure 4.11 shows most repetitions have the estimating equations constraints satisfied and there are also some outliers with constraints up to -0.1.

Figures 4.14, 4.12, 4.13 show our estimates for θ are reasonable, with a mean close to the true parameter 3. Variance in our estimates is also not too large. The histogram figure 4.13 shows our estimates are slightly right skewed.

The likelihood plot figure 4.15 shows the likelihood converges properly to a maximum point.

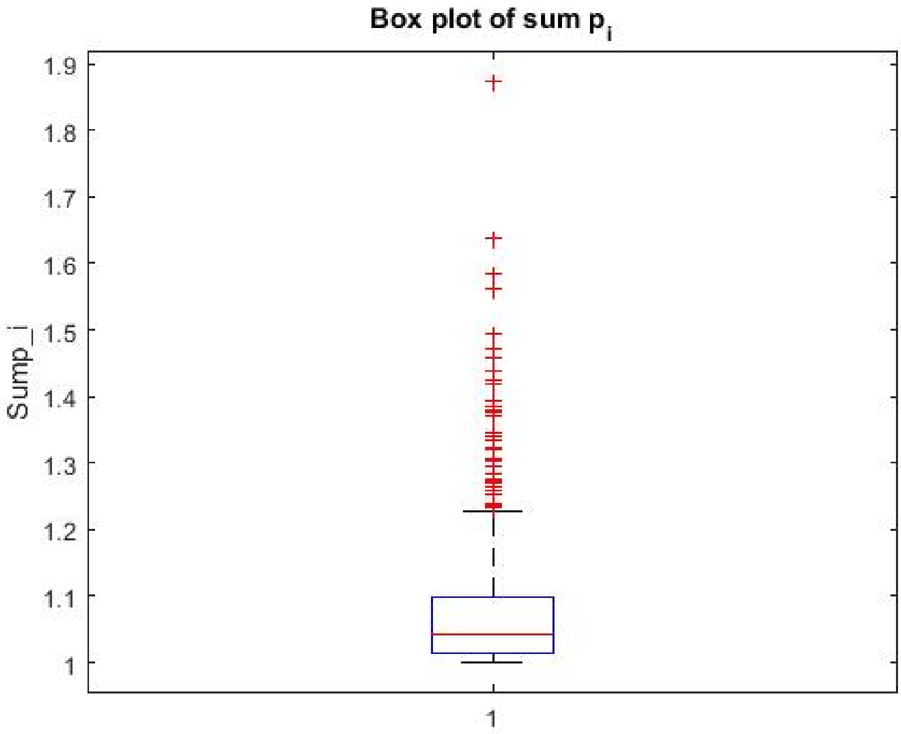

Figure 4.10: Box plot of $\sum_{i=1}^{n} p_i$. The median is slightly larger than 1 with some outliers larger than 1.3.

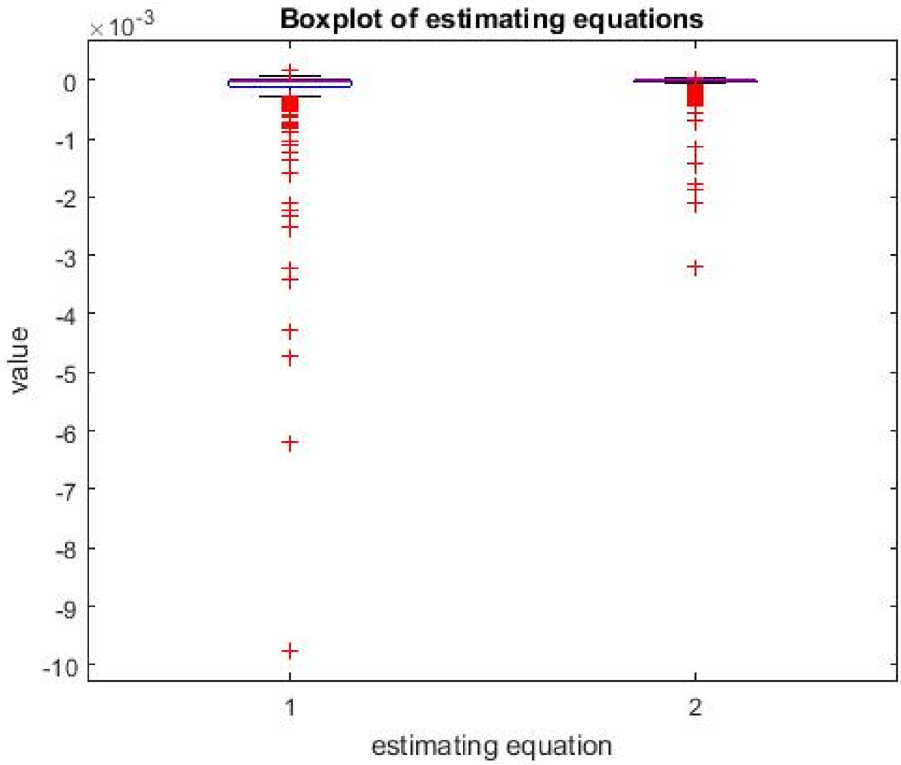

Figure 4.11: Box plot of the estimating equations, $\sum_{i=1}^{n} p_i h_i$. The figure shows the constraint has been fairly well satisfied.

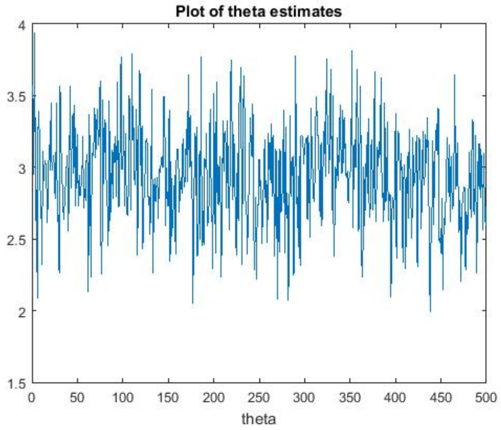

Figure 4.12: Plot of $\hat{\theta}$ versus repetition.

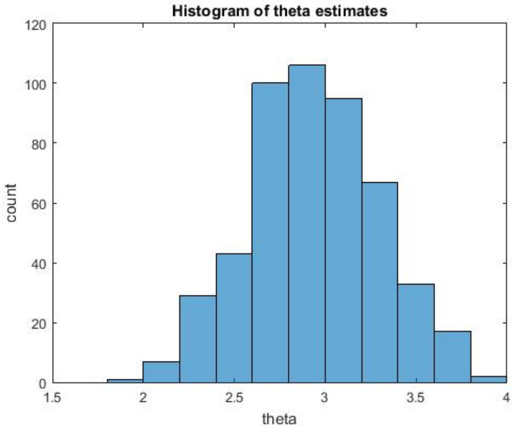

Figure 4.13: Histogram of $\hat{\theta}$. Slight right skewness is seen in the distribution.

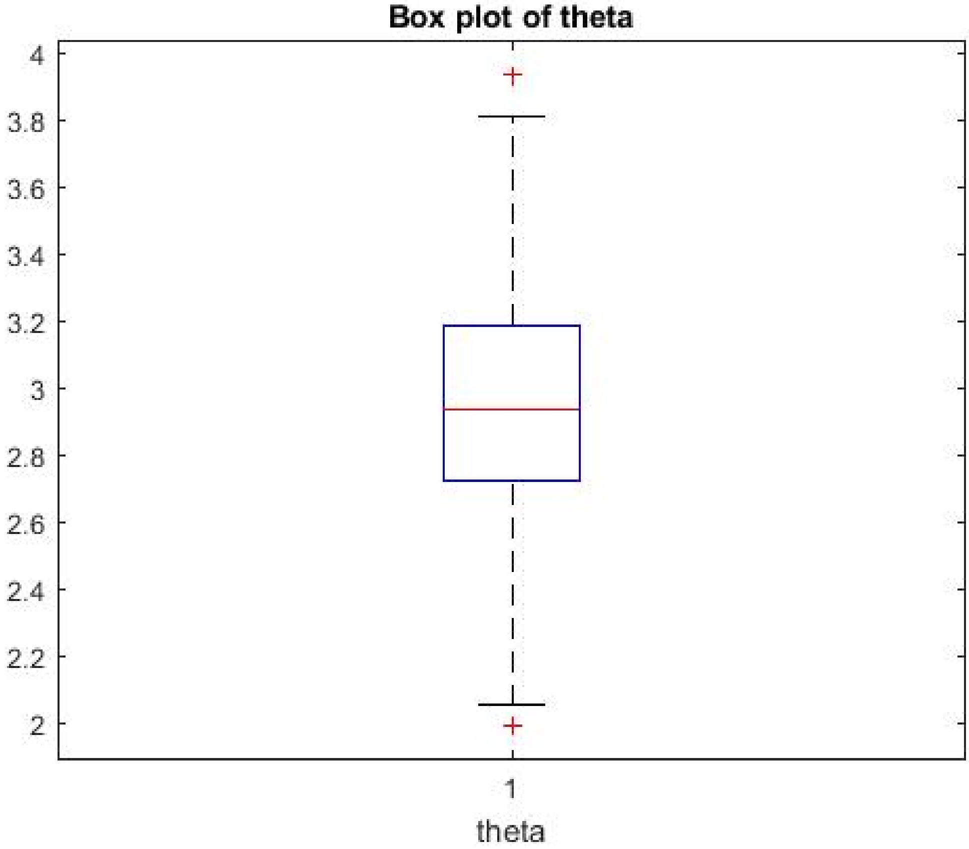

Figure 4.14: Box plot of $\hat{\theta}$. The median is quite close to the true parameter 3. Variance is not too large.

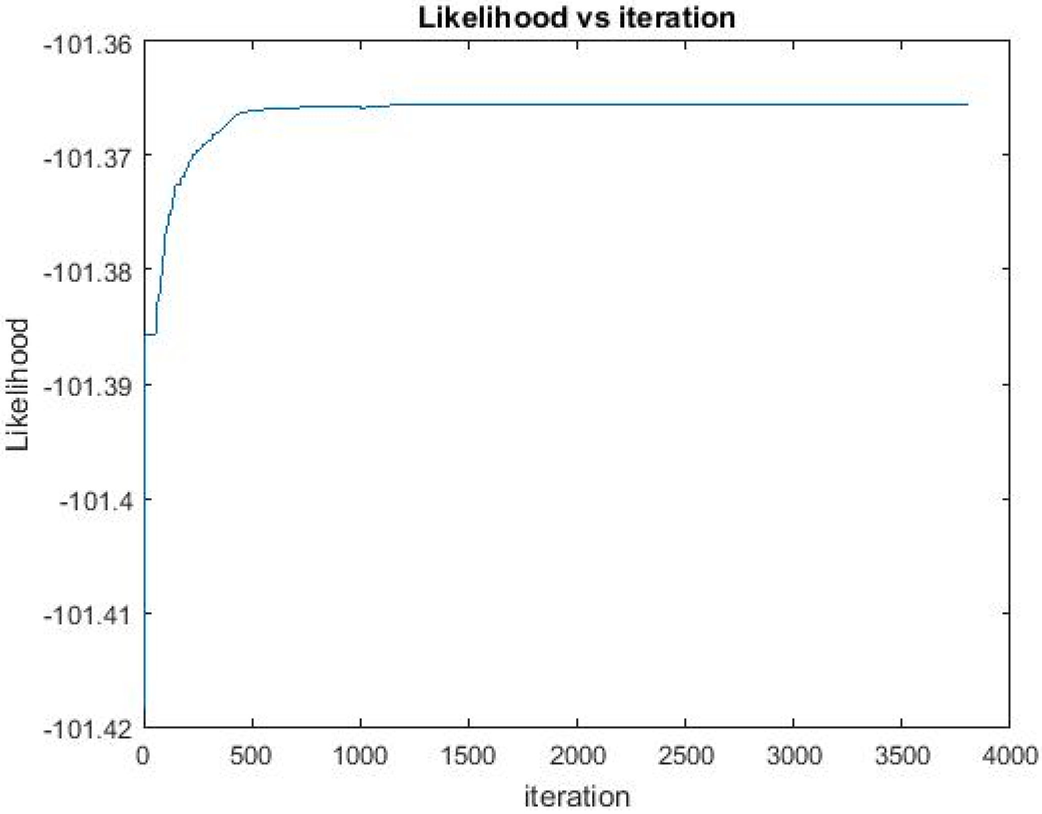

Figure 4.15: Likelihood until convergence of the first repetition.

5

Conclusion

This thesis has outlined the key areas of the extensive theory of empirical likelihood. From the development of the empirical likelihood function, likelihood ratio for asymptotic properties allowing confidence regions to be constructed as well as how to apply estimating equations and incorporate regression into empirical likelihood. Existing methods of solving the maximization empirical likelihood function along with optimization algorithms were compared. The problem with existing methods occurring from the assumption $p_i \geq 0$ was discussed and a working method which ensures this constraint is satisfied has been demonstrated. Simulation results from MATLAB code demonstrate how the method is implemented and obtains its estimates with working examples.

Simulations show our method struggled to enforce the constraint $\sum_{i=1}^{n} p_i = 1$ through the use of Lagrange multipliers. This problem calls for alterations to be made to improve our method's accuracy of the constraint. One method may be to transform the p_i such that our variable p_i guarantees the sum is 1 rather than enforcing the constraint through Lagrange.

This could be done by rescaling p_i. We can transform p_i by

$$p_i = \frac{\xi_i}{\sum_{t=1}^{n} \xi_t}$$

to incorporate the constraint $\sum_{i=1}^{n} p_i = 1$. Using this transformation will guarantee $\sum_{i=1}^{n} p_i = 1$ in the multiplicative iterative algorithm, which we allow our method to maintain this result.

It would interesting to see how the method performs with different sample sizes n and to consider how the likelihood ratio compares to the asymptotic result for different values of n. Exploration of how the method compares against other methods such as Owen's method (Owen (1988)) and Chen's method (Chen et al. (2008)) particularly with specific examples when Owen's method fails. Research for how our method can be altered to handle problems which are determined or undetermined is required. We have shown this method is useful for problems which are overdetermined, where we have more estimating equations than the number of parameters we wish to estimate. If the number of estimating equations is equal to the number of parameters we wish to estimate, this method will return the same solution as the method of moments. This can be seen from the MI algorithm equation (3.4), given initial starting values for $p_i = \frac{1}{n}$, $\gamma = 0$ and θ as the method of moments estimate, the numerator and denominator are equal and the likelihood is already at a saddle point. Further work could involve writing an R package to run the code of this method for greater accessibility.

Empirical likelihood is an exceptional tool for inference due to the nonparametric nature of the empirical likelihood ratio asymptotic properties. This is source of its strengths and weaknesses compared with other methods. The ability to be adjusted to handle a wide variety of problems in conjunction with existing methods such as estimating equations is the greatest strength of the method and have allowed empirical likelihood to have far reaching applications.

A
Appendix

A.1 MATLAB Code

A.1.1 Example: Linear regression with test for a particular covariate coefficient

Name

`ELAugregt`

Description

The program calculates estimates for p_i and β given a set of data with response variable Y, covariates $X = [X_1, \ldots, X_n]$ and a hypotheses on the true level of β. It uses a multiplicative iterative algorithm to update p_i and an augmented Lagrange method to include the estimating equation regression constraints.

Code

```
function [p_ih, beta, other] = ELAugregt(X, Y, tIdxVal,
   varargin)  % give theta (mu, sigma)
%
%ELAugregt      Hypothesis test on beta for empirical
   likelihood regression using Augmented Lagrange
%
%Usage: [p_i, beta, other] = ELAugregt(y, X, tIdxVal,
   varargin)
%
%Inputs:
%  X = covariates obs vector
%  Y =  response obs vector
% tIdxVal = 2-element vec for test parameter index and its
   test value under H0
%Default values in "varargin":
```

```
%  'maxiter' = maximum iteration number
%  'p_i0' = initial p_i value
%  'beta0' = initial beta value
%  'lam0' = initial lam2 (I call gamma)
%  'allrho' = vector of alpha values (usually in 1 - 150)
%   'hmat' = estimating equation matrix
%
%Outputs:
%  p_ih = p_i estimate
%  beta = beta estimate

%  other = struct array for other estimates, including
%  other.cvg = list of [outiter, inneriter, penalized
   likelihood]
%  other.score = score function values at convergence
%  other.lam = Lagrange mutiplier gamma for score function
   constraints

p = size(X,2);
tIdx = tIdxVal(1); tVal = tIdxVal(2);
rX = X(:,1:end~=tIdx);% remove covariate related to
   coefficient beta we are testing e.g. beta_1 and x_1
nrX = X(:, tIdx);
rp = size(rX,2);%reduced num of beta's
%default values; can be changed in inputs
maxiter = 1000;
allrho = 2.^(1:10); %alpha values
%rho1 = 100;
n=length(Y); %n1=(n-1)/n;
p_i0 = ones(n, 1)/n; %nonrescaled pi, initial =1/n
%p_i0=rand(n,1); %inital p_i is rand
%p_i0=ones(n,1); %inital p_i is 1/n
%p_i0=p_i0/sum(p_i0); %rescaled pi
lam0 = zeros(p, 1);
%lam0=[1;2];
%lam10 = 0;
%damp = 5e-1;

XtY = zeros(p,1,n);
for i = 1:n
    XtY(:,:,i) = X(i,:)'*Y(i);
end

beta0 = X'*rX\X'*(Y-tVal*nrX);

hmat=zeros(n,p);
resi = Y-nrX*tVal-rX*beta0;
for i=1:n
```

```
    hmat(i,:)=resi(i)'*X(i,:);
end


varglen = length(varargin);
if varglen ~= 0
    numvarg = varglen/2;
    t = 1;
    for k = 1:numvarg
        eval([varargin{t} '=varargin{t+1};'])
        t = t+2;
    end
end
cvg = [];

%rename variables
oldp_i = p_i0;
lam = lam0; %lam1 = lam10;
oldbeta=beta0;

%%%%%%%%%%%%%
% main part %
%%%%%%%%%%%%%

for outiter = 1:length(allrho)
    rho = allrho(outiter);
    pthmat = repmat(oldp_i,1,p).*hmat;
    sph = sum(pthmat);
   %alval0 = sum(log(max(eps, oldp_i)))-0.5*rho*sum(sph.^2)
      -0.5*rho*(1-sum(oldp_i))^2;
    alval0 = sum(log(max(eps, oldp_i)))-0.5*rho*sum(sph.^2);
       %-lam1*(1-sum(oldpi))-sph*lam

    %estimate all the parameters for this given rho (alpha)
    for iter = 1:maxiter
        %lam1 = -n+sph*lam+rho*sum(sph.^2);
        %%update pi
        f1 = hmat*lam;
        f2 = hmat*sph';
        num = 1./oldp_i- min(0,f1) - rho*min(0,f2)+max(0,sph*
           lam)+rho*sum(sph.^2); %-lam
        den = n +max(0,f1) + rho*max(0,f2)-min(0,sph*lam);
        gradp_i = num-den;
        ss = oldp_i./den;
        incp_i = ss.*gradp_i;
        newp_i = oldp_i+incp_i;
        pthmat = repmat(newp_i,1,p).*hmat;
        sph = sum(pthmat);
```

```
%alvalp = sum(log(max(eps, newp_i)))-0.5*rho*sum(sph
   .^2)-0.5*rho*(1-sum(newp_i))^2;
alvalp = sum(log(max(eps, newp_i)))-0.5*rho*sum(sph
   .^2); %-lam1*(1-sum(newp_i))-sph*lam
ome = 0.6;
%Armijo line search
while alvalp < alval0 %this occurs when step size was
    too large (jumped too far, in that case try step
   size ome)
    newp_i = oldp_i+ome*incp_i;
    pthmat = repmat(newp_i,1,p).*hmat;
    sph = sum(pthmat);
    %alvalp = sum(log(max(eps, newp_i)))-0.5*rho*sum(
       sph.^2)-0.5*rho*(1-sum(newp_i))^2;
    alvalp = sum(log(max(eps, newp_i)))-0.5*rho*sum(
       sph.^2); %-lam1*(1-sum(newpi))-sph*lam
    if ome >= 1e-2
        ome = ome*0.6;
    elseif ome < 1e-2 && ome >= 1e-5
        ome = ome*5e-2;
    elseif ome < 1e-5 && ome >= 1e-20
        ome = ome*1e-5;
    else
        break;
    end
end

   %%update regression coef beta using quasi-Newton
%first compute derivative of h matrix with respect to
    beta
%resi = Y-nrX*tVal-rX*oldbeta;
dhmat = zeros(p,rp,n); %3d matrix: pxrpxn
ptdhmat = dhmat;
   %computes terms for updating beta
   for i = 1:n
       dhmat(:,:,i) = -X(i,:)'*rX(i,:);
       ptdhmat(:,:,i)=newp_i(i)*dhmat(:,:,i);
       ptXtY = newp_i(i)*XtY(:,:,i);
   end
   spdh = sum(ptdhmat, 3);
   spXtY = sum(ptXtY, 3);
 newbeta = rho*(spdh)'*spdh\spdh'*(lam+rho*spXtY);
 incb = newbeta-oldbeta;
 resi = Y-nrX*tVal-rX*newbeta;
    for i=1:n
        hmat(i,:)=resi(i)'*X(i,:);
    end
```

```
pthmat = repmat(newp_i,1,p).*hmat;
sph = sum(pthmat);
%alvalt = sum(log(max(eps, newp_i)))-0.5*rho*sum(sph
   .^2) -0.5*rho*(1-sum(newp_i))^2;
 alvalb = sum(log(max(eps, newp_i)))-0.5*rho*sum(sph
    .^2); %-lam1*(1-sum(newpi))-sph*lam
ome = 0.6;
%Armijo line search
while alvalb < alvalp
    newbeta = oldbeta+ome*incb;
    resi = Y-nrX*tVal-rX*newbeta;
    for i=1:n
        hmat(i,:)=resi(i)'*X(i,:);
    end
pthmat = repmat(newp_i,1,p).*hmat;
sph = sum(pthmat);
% alvalt = sum(log(max(eps, newp_i)))-0.5*rho*sum(sph
   .^2)-0.5*rho*(1-sum(newp_i))^2;
    alvalb = sum(log(max(eps, newp_i)))-0.5*rho*sum(
       sph.^2); %-lam1*(1-sum(newpi))-sph*lam
    if ome >= 1e-2
        ome = ome*0.6;
    elseif ome < 1e-2 && ome >= 1e-5
        ome = ome*5e-2;
    elseif ome < 1e-5 && ome >= 1e-20
        ome = ome*1e-5;
    else
        break;
    end
end

%%update Lagrange multpliers vector gamma
lam = lam+rho*sph';
%lam1 = lam1+1e-3*rho*(1-sum(newp_i));

if all(abs(newp_i-oldp_i)<1e-5)&&all(abs(newbeta-
   oldbeta)<1e-5)&&all(rho*sph'<1e-3);
    %all(abs([newp_i;newbeta]-[oldp_i;oldbeta])<1e-5)
       &&all(rho*sph'<1e-3);
    break
else
    oldp_i = newp_i;
    oldbeta = newbeta;
    alval0 = alvalb;
    %lam = newlam;
end
cvg = [cvg; [outiter, iter, alval0, sum(log(max(eps,
   newp_i))), sum(newp_i)]]; %#ok<*AGROW>
```

```
    end
    if all(abs(sph)<1e-6)
        break;
    end
end
p_ih = newp_i;
beta = newbeta;
other.cvg = cvg;
other.score = sph;
other.lam = lam;
```

A.1.2 Example: Estimating Equations on Poisson first and second moments

Name

ELAug2eeLM

Description

The program calculates estimates for p_i and θ given a set of data points X. It uses a multiplicative iterative algorithm to update p_i and an augmented Lagrange method to include the estimating equation constraints on the first and second moments.

Code

```
function [p_ih, theta, other] = ELAug2eeLM(X, varargin)  %
   gives theta
%
%sp_augl       Saddle point test for GEE linear model using
   Augmented Lagrange
%
%Usage: [p_i, theta, lam, other] = ELAug2eeLM(X, varargin)
%Inputs:
%  X = response obs vector
%
%Default values in "varargin":
%  'maxiter' = maximum iteration number
%  'p_i0' = initial p_i value
%  'theta0' = initial theta value
%  'lam0' = initial gamma
%  'allrho' = vector of alpha values (usually in 1 - 150)
%  'hmat' = estimating equation matrix
%
%Outputs:
%  p_ih = p_i estimate
%  theta = theta estimate

%  other = struct array for other estimates, including
%   other.cvg = list of [outiter, inneriter, penalized
   likelihood]
%   other.score = score function values at convergence
%   other.lam = Lagrange mutiplier gamma for score function
   constraints

%default values; can be changed in inputs
maxiter = 1000;
allrho = 2.^(1:10); %alpha values
n=length(X);
p_i0 = ones(n, 1)/n; %nonrescaled pi, initial =1/n
%p_i0=rand(n,1); p_i0=p_i0/sum(p_i0); %rescaled pi
%p is number of rows on h matrix
%lam0 = zeros(p, 1);
lam0 = zeros(2, 1);
%lam0=[1;2];
%lam10 = 0;
damp = 5e-1;

theta0=mean(X); %or Var(X)
h1 = X - theta0;
h2 = X.^2-theta0^2-theta0;
hmat=[h1,h2];

varglen = length(varargin);
if varglen ~= 0
    numvarg = varglen/2;
    t = 1;
    for k = 1:numvarg
        eval([varargin{t} '=varargin{t+1};'])
        t = t+2;
    end
end
cvg = [];

%rename variables
oldp_i = p_i0; oldtheta = theta0;
lam = lam0; %lam1 = lam10;


%%%%%%%%%%%%%
% main part %
%%%%%%%%%%%%%

for outiter = 1:length(allrho)
    rho = allrho(outiter);
    pthmat = repmat(oldp_i,1,2).*hmat;
    % make variable cols in p-mat since must be equal to rows
        in hmat i.e. =d1=p
    sph = sum(pthmat);
   %alval0 = sum(log(max(eps, oldp_i)))-0.5*rho*sum(sph.^2)
      -0.5*rho*(1-sum(oldp_i))^2;
    alval0 = sum(log(max(eps, oldp_i)))-0.5*rho*sum(sph.^2);
       %-lam1*(1-sum(oldpi))-sph*lam

    %estimate all the parameters for this given rho
    for iter = 1:maxiter
        %lam1 = -n+sph*lam+rho*sum(sph.^2);
        %%update pi
        f1 = hmat*lam;
        f2 = hmat*sph';
        num = 1./oldp_i- min(0,f1) - rho*min(0,f2)+max(0,sph*
           lam)+rho*sum(sph.^2); %-lam
        den = n +max(0,f1) + rho*max(0,f2)-min(0,sph*lam);
        gradp_i = num-den;
        ss = oldp_i./den;
        incp_i = ss.*gradp_i;
        newp_i = oldp_i+incp_i;
        pthmat = repmat(newp_i,1,2).*hmat;
        sph = sum(pthmat);

        %alvalp = sum(log(max(eps, newp_i)))-0.5*rho*sum(sph
           .^2)-0.5*rho*(1-sum(newp_i))^2;
        alvalp = sum(log(max(eps, newp_i)))-0.5*rho*sum(sph
           .^2); %-lam1*(1-sum(newp_i))-sph*lam
        ome = 0.6;
        %Armijo line search
        while alvalp < alval0
            newp_i = oldp_i+ome*incp_i;
            pthmat = repmat(newp_i,1,2).*hmat;
            sph = sum(pthmat);
            %alvalp = sum(log(max(eps, newp_i)))-0.5*rho*sum(
               sph.^2)-0.5*rho*(1-sum(newp_i))^2;
            alvalp = sum(log(max(eps, newp_i)))-0.5*rho*sum(
               sph.^2); %-lam1*(1-sum(newpi))-sph*lam
            if ome >= 1e-2
                ome = ome*0.6;
            elseif ome < 1e-2 && ome >= 1e-5
                ome = ome*5e-2;
            elseif ome < 1e-5 && ome >= 1e-20
                ome = ome*1e-5;
            else
                break;
            end
        end
```

```
%update theta by  using Levenberg-Marquardt
 %first compute derivative of h matrix with respect to
     theta;
    dhmat = zeros(2, 1, n); %3d matrix: 2x1xn
    ptdhmat = dhmat;
    %computes hessian matrix Sb
    for i = 1:n
        dhmat(:,:,i) = [-1; -1-2*oldtheta];
        ptdhmat(:,:,i)=newp_i(i)*dhmat(:,:,i);
       lmai = ptdhmat(:,:,i)'*(lam+rho*sph');
        tmps = lmai*lmai';
        %damp = 1./min(diag(tmps)+1e-1);
        Sbi(:,:,i) = tmps+damp*diag(diag(tmps));
    end
    spdh = sum(ptdhmat, 3);
    gradt = -spdh'*(lam+rho*sph');
    Sb = sum(Sbi, 3);
    inct = Sb\gradt;

  newtheta= oldtheta + inct;
     h1= X-newtheta;
     h2= X.^2-newtheta^2-newtheta;

 hmat=[h1 ,h2 ];
 pthmat = repmat(newp_i,1,2).*hmat;
 sph = sum(pthmat);
 %alvalt = sum(log(max(eps, newp_i)))-0.5*rho*sum(sph
    .^2) -0.5*rho*(1-sum(newp_i))^2;
  alvalt = sum(log(max(eps, newp_i)))-0.5*rho*sum(sph
     .^2); %-lam1*(1-sum(newpi))-sph*lam
 ome = 0.6;
 %Armijo line search
 while alvalt < alvalp
     newtheta = oldtheta+ome*inct;
     h1= X-newtheta;
     h2= X.^2-newtheta^2-newtheta;
 hmat=[h1 ,h2 ];
 pthmat = repmat(newp_i,1,2).*hmat;
 sph = sum(pthmat);
 % alvalt = sum(log(max(eps, newp_i)))-0.5*rho*sum(sph
    .^2)-0.5*rho*(1-sum(newp_i))^2;
     alvalt = sum(log(max(eps, newp_i)))-0.5*rho*sum(
        sph.^2); %-lam1*(1-sum(newpi))-sph*lam
     if ome >= 1e-2
         ome = ome*0.6;
     elseif ome < 1e-2 && ome >= 1e-5
         ome = ome*5e-2;
```

```
            elseif ome < 1e-5 && ome >= 1e-20
                ome = ome*1e-5;
            else
                break;
            end
        end
        lam = lam+rho*sph';
        %lam1 = lam1+1e-3*rho*(1-sum(newp_i));

        if all(abs([newp_i;newtheta]-[oldp_i;oldtheta])<1e-5)
           &&all(rho*sph'<1e-3);
            %all(abs(newp_i-oldp_i)<1e-7)&&all(rho*sph'<1e-3)
            break
        else
            oldp_i = newp_i;
            oldtheta = newtheta;
            alval0 = alvalt;
            %lam = newlam;
        end
        cvg = [cvg; [outiter, iter, alval0, sum(log(max(eps,
           newp_i))), sum(newp_i)]]; %#ok<*AGROW>
    end
    if all(abs(sph)<1e-6)
        break;
    end
end
p_ih = newp_i;
theta = newtheta;
other.cvg = cvg;
other.score = sph;
other.lam = lam;
```

References

Armijo, L. et al. (1966), ‘Minimization of functions having lipschitz continuous first partial derivatives’, *Pacific Journal of mathematics* **16**(1), 1–3.

Bellman, R. (1956), ‘Dynamic programming and lagrange multipliers’, *Proceedings of the National Academy of Sciences of the United States of America* **42**(10), 767.

Chen, J., Variyath, A. M. & Abraham, B. (2008), ‘Adjusted empirical likelihood and its properties’, *Journal of Computational and Graphical Statistics* **17**(2), 426–443.

Chen, S. X. & Van Keilegom, I. (2009), ‘A review on empirical likelihood methods for regression’, *Test* **18**(3), 415–447.

DiCiccio, T., Hall, P. & Romano, J. (1991), ‘Empirical likelihood is bartlett-correctable’, *The Annals of Statistics* pp. 1053–1061.

Fan, J. & Gijbels, I. (1996), *Local polynomial modelling and its applications: monographs on statistics and applied probability 66*, Vol. 66, CRC Press.

Hall, P. & La Scala, B. (1990), ‘Methodology and algorithms of empirical likelihood’, *International Statistical Review/Revue Internationale de Statistique* pp. 109–127.

Hestenes, M. R. (1969), ‘Multiplier and gradient methods’, *Journal of optimization theory and applications* **4**(5), 303–320.

Kiefer, J. & Wolfowitz, J. (1956), ‘Consistency of the maximum likelihood estimator in the presence of infinitely many incidental parameters’, *The Annals of Mathematical Statistics* pp. 887–906.

Kuhn, H. W. & Tucker, A. W. (1951), Nonlinear programming, *in* ‘Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability’, University of California Press, Berkeley, Calif., pp. 481–492.
URL: *http://projecteuclid.org/euclid.bsmsp/1200500249*

Luenberger, D. G. & Ye, Y. (2008), *Linear and nonlinear programming*, Vol. 116, Springer Science & Business Media.

Ma, J. (2006), ‘Multiplicative algorithms for maximum penalized likelihood inversion with non negative constraints and generalized error distributions’, *Communications in Statistics-Theory and Methods* **35**(5), 831–848.

Marquardt, D. W. (1963), ‘An algorithm for least-squares estimation of nonlinear parameters’, *Journal of the Society for Industrial & Applied Mathematics* **11**(2), 431–441.

McCullagh, P. (1984), ‘Generalized linear models’, *European Journal of Operational Research* **16**(3), 285–292.

McCullagh, P. & Nelder, J. A. (1989), *Generalized linear models*, Vol. 37, CRC press.

Nocedal, J. & Wright, S. (2006), *Numerical optimization*, Springer Science & Business Media.

Owen, A. (1990), ‘Empirical likelihood ratio confidence regions’, *The Annals of Statistics* pp. 90–120.

Owen, A. (1991), ‘Empirical likelihood for linear models’, *The Annals of Statistics* pp. 1725–1747.

Owen, A. B. (1988), ‘Empirical likelihood ratio confidence intervals for a single functional’, *Biometrika* **75**(2), 237–249.

Owen, A. B. (2001), *Empirical likelihood*, CRC press.

Pearl, R. & Fuller, W. N. (1905), ‘Variation and correlation in the earthworm’, *Biometrika* pp. 213–229.

Powell, M. J. D. (1969), ‘A method for nonlinear constraints in minimization problems, optimization’, pp. 283–298.

Powell, M. J. D. (1973), ‘On search directions for minimization algorithms’, *Mathematical Programming* **4**(3), 193–201.

Qin, J. & Lawless, J. (1994), ‘Empirical likelihood and general estimating equations’, *The Annals of Statistics* pp. 300–325.

Rockafellar, R. T. (1973), ‘The multiplier method of hestenes and powell applied to convex programming’, *Journal of Optimization Theory and applications* **12**(6), 555–562.

Thomas, D. R. & Grunkemeier, G. L. (1975), ‘Confidence interval estimation of survival probabilities for censored data’, *Journal of the American Statistical Association* **70**(352), 865–871.

Wand, M. P. & Jones, M. C. (1994), *Kernel smoothing*, Crc Press.

Wang, Q.-H. & Jing, B.-Y. (2003), ‘Empirical likelihood for partial linear models’, *Annals of the Institute of Statistical Mathematics* **55**(3), 585–595.

Wilks, S. S. (1938), ‘The large-sample distribution of the likelihood ratio for testing composite hypotheses’, *The Annals of Mathematical Statistics* **9**(1), 60–62.

Zhang, J. & Gijbels, I. (2003), ‘Sieve empirical likelihood and extensions of the generalized least squares’, *Scandinavian Journal of Statistics* **30**(1), 1–24.