Chapter 1

Derivative-Free Optimisation and the Paremeter Cascade

The Parameter Cascade requires the computation of derivatives for the various levels of the problem to perform optimisation. However, computing these can be a time-consuming and complex task. In some cases the derivative might not even exist. In this chapter, the use of derivative-free methods for optimisation. Derivative-free methods are used to tackle a series of increasingly complex problems, culminating in fitting one level of the Parameter Cascade without derivatives.

1.1 Overview of Quadratic Optimisation Methods

A large class of numerical optimisation methods rely on constructing a quadratic approximation to objective function $f(\theta)$. Given an iterate θ_n and possibly some associated data, a quadratic approximation $m_n(\theta)$ to the objective function is constructed. The next iterate θ_{n+1} is then found by minimising $m_n(\theta)$. Constructing the approximate quadratic and then minimising it tends to be straightforward. If the next iterate $\theta_{n+1}(\theta)$ is unsatisfactory, a new quadratic model function $m_{n+1}(\theta)$ is minimised, producing a new iterate θ_{n+2}. Ideally, the θ_n will approach the optimal point and the sequenece of quadratic models will become increasingly accurate approximations so that the process can be repeated until convergence. [16]

1.1.1 Newton's Method

The Newton-Raphson Method is a well-known member of this class. Newton's method constructs the approximation using a second-order Taylor expansion around θ_n :

$$f(\theta) \approx m_n(\theta) = f(\theta_n) + f'(\theta_n)(\theta - \theta_n) + \frac{1}{2}f''(\theta_n)(\theta - \theta_n)^2$$

It is not difficult to show that the critical point of $m_n(\theta)$ is given by $\theta_{n+1} = \theta_n - f'(\theta)/f''(\theta)$, which is the usual Newton formula [16, 9, 11, 12].

For a point close to θ_n, the difference between $f(\theta)$ and $m_n(\theta)$ is roughly equal to $[f'''(\theta_n)/3!](\theta - \theta_n)^3$ so long as $f(\theta)$ is sufficiently well behaved[9]. This formula suggests that if θ_n is close to the optimal point θ^* so that $|\theta^* - \theta_n|$ is sufficiently small, then $|\theta_n - \theta^*|^3$ will be very small indeed and so the quadratic model will be a very accurate approximation of $f(\theta)$ around θ^*. As a result, θ_{n+1} will be quite close to θ^*. The next model $m_{n+1}(\theta)$ will thus be substantially better than $m_n(\theta)$ at approximating $f(\theta)$ around θ^*, and so θ_{n+2} will be much closer to θ^* than θ_{n+1}. Newton's method converges very rapidly so long as one is sufficiently close to θ^* to start with. In fact, it can be shown that Newton's method exhibits *quadratic convergence* subject to technical conditions. This means that $|\theta_{n+1} - \theta^*| \approx C|\theta_n - \theta^*|^2$. For example, if the error in the first iterate is approximately 0.1, the next iterate will have error on the order of 10^{-2}, the next will have error on the order of 10^{-4}, and so on [16, 12].

Newton's method is a very effective estimation algorithm so long as the derivatives $f'(\theta)$ and $f''(\theta)$ can be computed, and so long as the initial starting value is not too far from the optimal value. Choosing a good initial value is thus very important. For maximum likelihood estimation for example, a method of moments estimator or the median could be used to provide an initial starting value.

1.1.2 Secant Method

If the second derivative is difficult to calculate, one can approximate it with a difference quotient instead [9, 16][1]:

$$f''(\theta) \approx \frac{f'(\theta_n) - f'(\theta_{n-1})}{\theta_n - \theta_{n-1}} \tag{1.1}$$

This leads to the quadratic approximation:

$$m_n(\theta) = f(\theta_n) + f'(\theta_n)(\theta - \theta_n) + \frac{1}{2}\left(\frac{f'(\theta_n) - f'(\theta_{n-1})}{\theta_n - \theta_{n-1}}\right)(\theta - \theta_n)^2$$

And the update formula:

$$\begin{aligned}\theta_{n+1} &= \theta_n - \left[\frac{f'(\theta_n) - f'(\theta_{n-1})}{\theta_n - \theta_{n-1}}\right]^{-1} f'(\theta_n) \\ &= \theta_n - \frac{f'(\theta_n)[\theta_n - \theta_{n-1}]}{f'(\theta_n) - f'(\theta_{n-1})} \\ &= \frac{\theta_{n-1}f'(\theta_n) - \theta_n f'(\theta_{n-1})}{f'(\theta_n) - f'(\theta_{n-1})}\end{aligned}$$

The Secant Method is straightforward to implement, and only requires first derivatives. Relying on on $f(\theta_n), f'(\theta_n)$ and $f'(\theta_{n-1})$ instead of $f(\theta), f'(\theta_n)$ and $f''(\theta_n)$ has a drawback however. The Secant Method's model is less accurate because θ_{n-1} tends to be further from θ^* than θ_n. More formally, the error for the model is roughly equal to $[f'''(\theta_n)/3!](\theta_n - \theta)^2(\theta_{n-1} - \theta)$. If the sequence is converging to θ^*, substituting in the $(\theta - \theta_{n-1})$ term inflates the error relative to Newton's Method and acts as a drag on convergence. It can be shown that the Secant Method only has a convergence rate of 1.618, but avoiding the cost of computing a second derivative on each step means that more iterations can be completed in a given period of time. The Secant Method is comparable with Newton's Method, and can be faster if computing the second derivative is difficult.

The Secant Method is a widely used method that provides a good trade-off between convergence speed and ease of implementation. R's `optim` routine uses a multivariate generalisation of the Secant Method for gradient-based optimisation [19]. Multivariate versions of the Secant Method for optimisation are usually referred to as *Quasi-Newton Methods* [16, 12, 11].

For multivariate problems, the second derivative is in the form of a matrix, so there is not enough information to construct a full approximation afresh on each iteration. Rather the approximate Hessian is partially updated using one of several approaches. No matter the exact approach, the approximate Hessians $\mathbf{B}_{n+1}$ are required to satisfy the multivariate secant condition[16, 12, 11]:

$$\nabla\mathbf{f}(\theta_{n+1}) - \nabla\mathbf{f}(\theta_n) = \mathbf{B}_{n+1}(\theta_{n+1} - \theta_n)$$

This is a generalisation of Equation 1.1. A further condition generally imposed in the multivariate case is that the approximate Hessians must be positive-definite. This ensures that the approximate

[1] The Secant Method is denoted as the *Method of False Position* in [9]

qudaratics don't have any saddle points or surfaces on which the second derivative is zero so that a well defined search direction is guaranteed [16, 11, 12].

R's `optim` routine uses the BFGS method to compute the next approximate Hessian[19]. BFGS uses the symmetric matrix $\mathbf{B}_{n+1}$ satisfying the secant condition such that the inverse $\mathbf{B}_{n+1}^{-1}$ minimises a weighted Frobenius distance o between itself and the previous inverse $\mathbf{B}_n^{-1}$. A low memory variant of BFGS known as L-BFGS is also available [19, 16].

1.1.3 Successive Parabolic Interpolation

Parabolic interpolation goes one step further than the Secant Method and dispenses with derivatives entirely. Instead, a model function is constructed by interpolation through the points $(\theta_n, f(\theta_n))$, $(\theta_{n-1}, f(\theta_{n-1}))$, and $(\theta_{n-2}, f(\theta_{n-2}))$[16, 23].

$$\begin{aligned} m_n(\theta) =& f(\theta_n)\frac{(\theta-\theta_{n-1})(\theta-\theta_{n-2})}{(\theta_n-\theta_{n-1})(\theta_n-\theta_{n-2})} \\ &+ f(\theta_{n-1})\frac{(\theta-\theta_n)(\theta-\theta_{n-2})}{(\theta_n-\theta_n)(\theta_n-\theta_{n-2})} \\ &+ f(\theta_{n-2})\frac{(\theta-\theta_{n-1})(\theta-\theta_n)}{(\theta_n-\theta_{n-1})(\theta_n-\theta_n)} \end{aligned}$$

This model has a approximate error of $[f'''(\theta_n)/3!](\theta-\theta_n)(\theta-\theta_{n-1})(\theta-\theta_{n-2})$. By relying on the past two iterates, the rate of convergence is slowed further. Parabolic interpolation has an order of convergence of 1.32.

An issue with parabolic interpolation is providing enough initial points to seed the method[16]. This is more acute for multivariate problems in particular. One approach is to provide enough points at the start and run the alogorithm from there. Alternatively, one can start off with just enough points needed to estimate an ascent or descent direction and construct a linear approximation, and then run the optimisation routine using a sequence of linear approximations until there enough points to construct a parabola. If one is using a linear approximation, one must impose a limit on the maximum distance that the routine can travel on each iteration since linear functions do not have a minimum or maximum and diverge off to infinity.

1.1.4 Discussion

All three approaches are governed by the same fundamental theory of approximating functions by polynomials. The only difference is the precise inputs used to construct an approximation. This means that if a problem is suitable for Newton's Method, the other two methods will very likely perform well. If one applies parabolic interpolation to a sufficiently smooth objective function, then one is in a sense automatically employing Newton's Method even if one made no effort to investigate the differentiabilty of the objective function.

On the other hand, the methods all share the same fundamental handicap as well, these methods are not guaranteed to converge unless the starting point is close to the optimal value. Local convergence doesn not necessarily imply global convergence. The error terms in the quadratric approximations are all something like $(\theta-\theta_n)^3$. If $|(\theta-\theta_n)|$ and any other error terms are small, the error in the approximation will be much smaller since it is proportional to the product of three such errors. If however the errors are large, their product might be so large that the method fails to converge. [16, 9, 12]

This is less academic than it might seem. Suppose one had a complicated likelihood function $L(\theta)$. Perhaps to evaluate the likelihood one must numerically integrate some kind of complex marginal distribution that depends on θ. Instead of attempting to find explicit formulae for the score and information functions, if one could produce a crude estimate $\hat{\theta}$ and crude estimate of the error $\hat{\sigma}_\theta$,

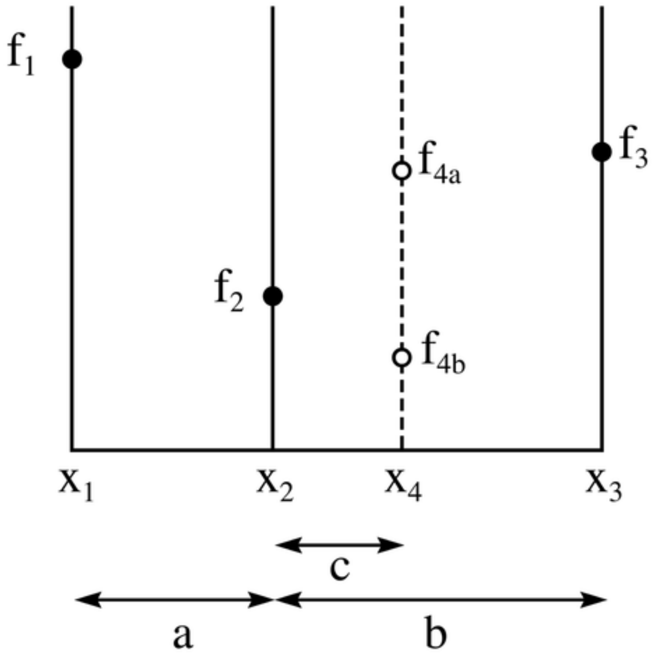

Figure 1.1: Diagram of golden-section search taken from Wikipedia. **TODO:** Find a better plot???

then one could use successive parabolic interpolation with $\{\hat{\theta}, \hat{\theta} - 2\hat{\sigma}_\theta, \hat{\theta} + 2\hat{\sigma}_\theta\}$ as a set of starting points. If $L(\theta)$ is in fact a well behaved smooth function, then parabolic interpolation will find the value of θ that maximises $L(\theta)$ fairly quickly. It is necessary to provide plausible starting values for θ because the quadratic model is only certain to be valid if one is already near the optimal value.

1.2 Bisection Methods

In contrast to the methods discussed above, bisection methods tend to be slow, but are guaranteed to ensure consistent and steady progress towards the optimal point provided the function is continuous and does not have more than one local minima or maxima and that the interval of interest is always shrunk by a given amount on each iteration. One starts with an interval $[a, b]$ and a third point c. such that $f(c) < f(a)$ and $f(c) < f(b)$. A fourth point d within the interval $[a, b]$ is selected, and $f(d)$ is computed. If d is between a and c, and $f(d) < f(a)$ and $f(d) < f(c)$, then $[a, c]$ becomes the new interval and d becomes the new provisional minimum. If $f(c) < f(d)$, then the new interval becomes $[d, b]$, - c remains the provisional minimum, but the interval has been narrowed. A similar approach applies if d is between c and b. The whole process is plotted in 1.1

The most common bisection method is known as Golden-Section Search, where the point d is chosen so that the width of the new interval is equal to that of the old one divided by 1.618.[12]

1.3 Brent's Method

Brent's Method is a hybrid of successive parabolic interpolation and golden-section search [2]. If parabolic interpolation is failing to provide a sufficiently rapid decrease in the objective function, a bisection step is performed. While the bisection steps might not produce as much progress as the parabolic steps, they are certain to produce a consistent rate of improvement no matter how close the algorithm is to the optimal point, while parabolic interpolation is only certain to work if one is already within a neighbourhood of the optimal point as noted in Section 1.1.4. Brent's method will also perform a bisection step if the interpolating parabola is ill-conditioned, or if a bisection step has not been performed recently.

The hybrid method is robust as a result of the golden section steps, and the parabolic steps ensure it performs well when applied to smooth functions along with a decent starting value.

1.4 Estimation Of Parameters For A Standard Cauchy Distribution Using Brent's Method

To illustrate how Brent's metthod is employed in practice it will be used on a straightforwards estimation problem first. Consider the quesion of fitting a Cauchy distribution to some data. Given n observations $x_1, \ldots, x_n$ from an unknown Cauchy distribution, the likelihood function is given by:

$$L(\mu, \sigma) = \prod_{i=1}^{n} \frac{1}{\pi\sigma\left[1 + \left(\frac{x-\mu}{\sigma}\right)^2\right]}$$

Attempting to maximise this likelihood by the usual method entails solving a fairly complex system of rational equations in μ and σ. Our purpose is to demonstrate that Brent's Method can tackle this problem without much difficulty.

Brent's Method can only optimise a function in one dimension at a time, so it is necessary to attempt to optimise for μ and σ separately. The profile log-likelihood of σ is computed:

$$\ell(\sigma) = \sup_{\mu} \log(L(\mu, \sigma))$$

R can evaluate $\ell(\sigma)$ straightfowardly by using Brent's method to optimise $L(\mu, \sigma)$ with respect to μ and holding σ constant. The function $\ell(\sigma)$ can then be in turn optimised with respect to σ to find the optimal value of σ. This procedure is illustrated in Figure 1.2.

One subtlety with optimising a Cauchy likelihood is that the likelihood function can have multiple local maxima since the likelihood function is the ratio of two multivariate polynomials in μ and σ. To ensure that the algorithm was sufficiently close to the MLE, the median was used as an initial estimate of μ, and half the interquartile range was used as an initial estimate for σ. Given these somewhat crude estimates $\tilde{\mu}$ and $\tilde{\sigma}$, the the standard error of the median $\sigma_{\tilde{\mu}}$ is approximately given by:

$$\hat{\sigma}_{\tilde{\mu}} \approx \frac{1}{2f(\tilde{\mu}; \tilde{\mu}, \tilde{\sigma})\sqrt{n}}$$

Where $f(x; \mu, \sigma)$ is the Cauchy density function with location parameter μ and scale parameter σ. The values $\tilde{\mu} \pm 2\hat{\sigma}_{\tilde{\mu}}$ are then used to provide the initial lower and upper bounds for the optimiser. The aim is to construct a confidence interval that is highly likely to contain the MLE for μ (rather than the actual true parameter), but isn't so wide that the interval is in danger of containing multiple local maxima for the likelihood.

Not only can the likelihood be maximised without derivitives, but asymptotic inference can be done without derivatives as well. Given the score function and the Fisher information at the maximum likelihood estimates, it is possible in principle to compute an approximate confidence interval for σ and μ[18]. Instead of analytic methods, one can use finite differences to approximately compute the necessary derivatives to the desired degree of accuracy[14, 6]. This was successful at producing a valid approximation for the profile likelihood, shown as a red dotted parabola in Figure 1.2.

It is thus possible to compute a confidence interval using the Score test. The test statistic $S(\sigma)^2/I(\sigma)$ could be accurately approximated using finite differences. One takes the value of σ for which the test statistic is less than or equal to the appropriate critical value from a chi-squared distribution. By inspecting the plot in Figure 1.3 and then solving for σ, an approximate confidence interval for σ can be computed. It is approximately the case that σ lies in $(0, 2.20)$ with 95 percent confidence.

An important assumption underpinning such asymptotic confidence intervals is that the two term quadratic Taylor expansion based on the score and information functions is valid over the range of interest. This is not the case here as can be seen in the spike in the score statistic on the left caused by the Fisher information changing sign at approximately $\sigma = 2.35$. This indicates that the confidence interval might be wider than the range of for which a quadratic approximation around the MLE is valid, and should perhaps be treated with some scepticism.

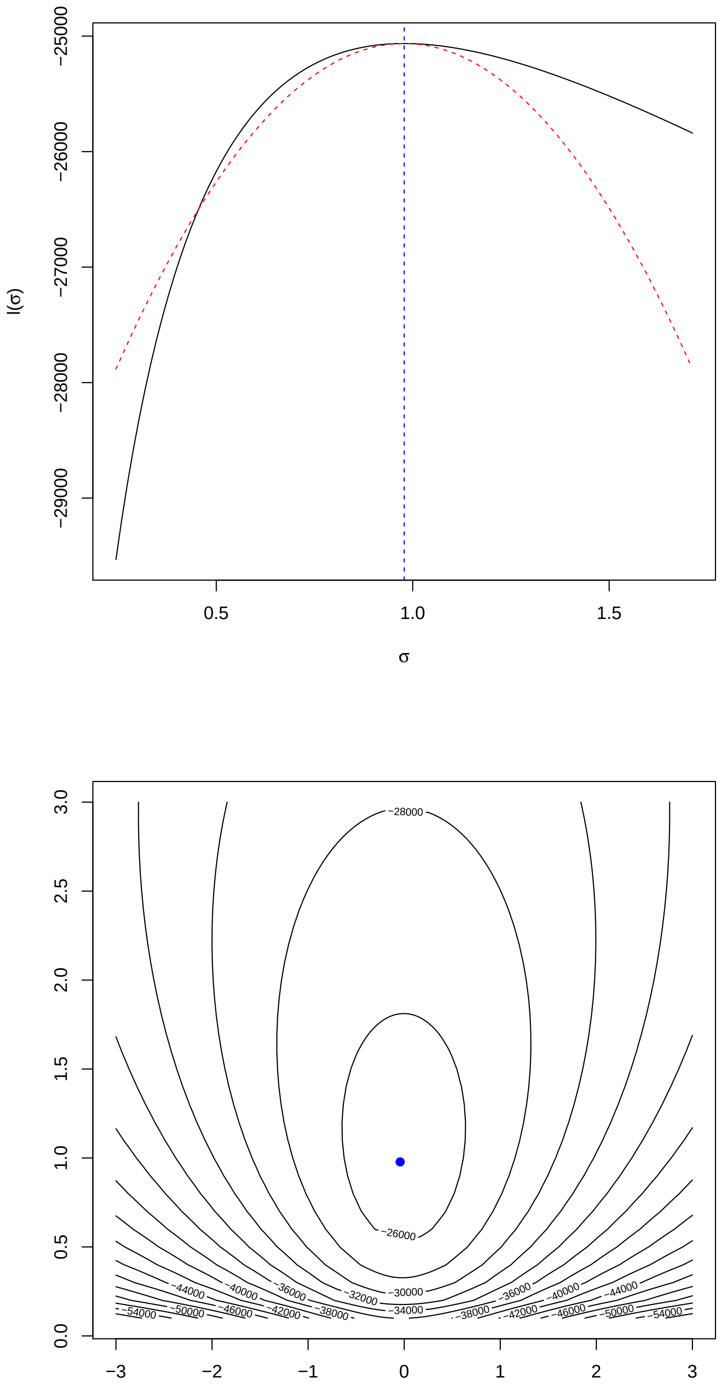

Figure 1.2: Profile log likelihood in σ, and contour plot of the joint log likelihood.

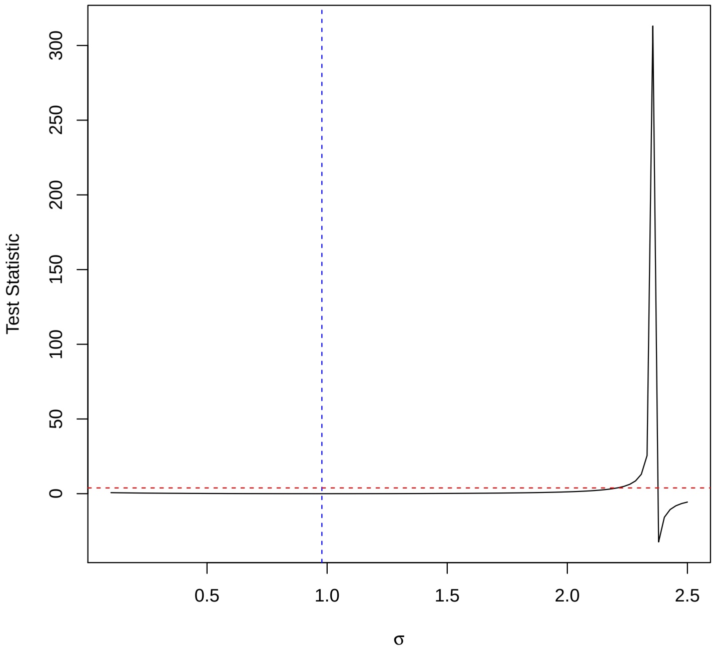

Figure 1.3: Plot of profile score statistic.

1.5 Robust ODE Parameter Estimation

If observations of values from an ODE are subject to heavy-tailed noise such as in the Cauchy case, least squares regression becomes unsuitable. An obvious candidate is L1 regression, which attempts to minimise the sum of the absolute values of the residuals instead of the sum of the squared residuals. An important property of L1 regression is that median is naturally associated with this approach; the sample median of a set of numbers is the constant value that minimises the L1 error just as the sample mean is the constant value that minimises the least squares error[21][2]. L1 regression can greatly complicate the process of estimation however, because the the function $|x|$ is not everywhere differentiable. This means that the usual gradient-based approaches to nonlinear regression such as gradient descent should not be applied. Even methods that attempt to numerically approximate the derivatives such as parabolic interpolation are either entirely unsuitable at worst, or not guaranteed to converge quickly at best.

Brent's Method can tackle such problems however, being robust against non differentiabilty. For nonlinear L1 regression, the objective function tends to be piecewise smooth - between the "kinks", the function is differentiable and amenable to parabolic interpolation. Once the bisection steps have reached a neighbourhood of the optimal value, parabolic interpolation will find it fairly quickly.

Consider for example, the following ODE with $\beta = -0.5$:

$$\begin{cases} y'' - \beta(1-y^2)y' + y = 0 \\ y(0) = 1 \\ y'(0) = 0 \end{cases} \tag{1.2}$$

This ODE describes a non-linear oscillator, and is representative of quasi-linear mathematical

[2] This is discussed in more detail in the next chapter

models that can't be tackled by the FDA package or Data2LD. Note that this ODE is of the form $y'' + \beta(y)y' + y = 0$ with $\beta(y) = -\beta(1 - y^2)$. By definition, the linear ODEs usually used in FDA cannot model systems where the $\beta(\cdot)$ terms have y as a dependent variable, they can only model situations where the parameters vary with time alone (and/or space in the case of a linear PDE).

We wish to investigate the problem of estimating β from noisy observations.

The `desolve` package [22] was used to numerically find the values of $y(t)$ at choosen time points $\{t_1, \dots, t_K\}$. The values of $y(t)$ at these points - corrupted by random Cauchy noise - were independently sampled N times. This produced a set of KN observations: $\{y_{11}, y_{12}, \dots, y_{1N}, \dots, y_{K1} \dots, y_{KN}\}$. Because the data is heavy-tailed, least squares regression is inappropriate. Instead, the goodness of fit associated with a given choice of β was measured by the sum of absolute errors associated with a given choice of β :

$$SAE(\beta) = \sum_{i=1}^{K}\sum_{j=1}^{N} |y(t_i;\beta) - y_{ij}|$$

Here $y(t;\beta)$ denotes the solution of Equation 1.2 for a given choice of β. To evaluate $SAE(\beta)$ at a given value of β, it is necessary to use `desolve` to numerically find the values of $y(t_i|\beta)$. Brent's method was used to find the number $\hat{\beta}$ that minimised $SAE(\beta)$. Figure 1.4 shows the original curve, the generated points, the realisation of $SAE(\beta)$, and the fitted curve generated by $\hat{\beta}$

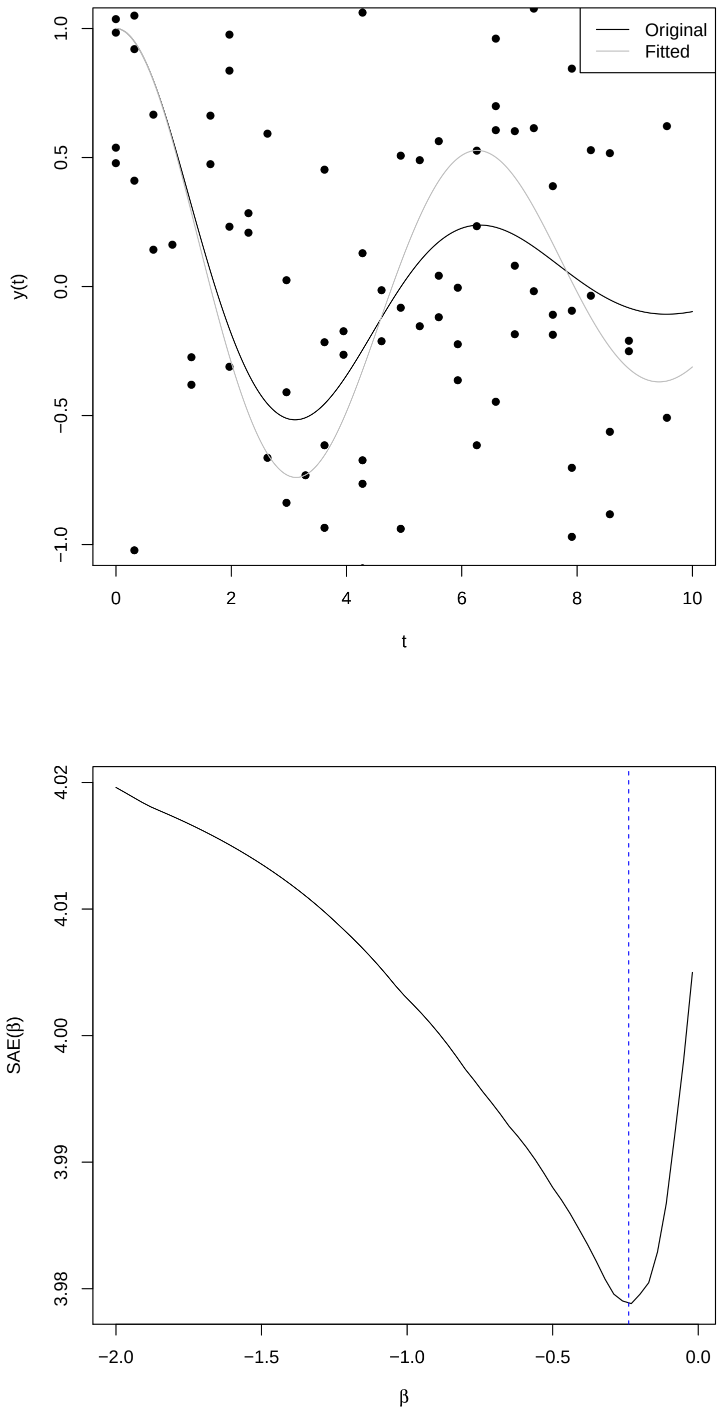

Figure 1.4: Original curve, fitted curve, and objective function.

1.6 The Parameter Cascade and Brent's Method

Recall that the Parameter Cascade has three levels.

First, the inner problem. There is a given functional $J(f;\theta,\lambda)$ that takes a function f and associated parameters θ and λ and returns a real number. Usually, the function f is represented by a vector of coefficients with a given associated basis. The function $\hat{f}(t|\omega,\lambda)$ that optimises $J(\cdot;\theta,\lambda)$ is then found. Outside of toy cases, this problem cannot be solved analytically. The problem is nearly always solved numerically by restricting the space of functions to the span of some set of choosen basis functions and optimising over that.

This in turn defines the middle problem, $H(\theta,\hat{f}(t|\omega,\lambda);\lambda) = H(\theta;\lambda)$, which is usually defined as the least squares error associated with the optimal f given θ and λ :

$$H(\theta;\lambda) = \sum[x_i - \hat{f}(t_i|\omega,\lambda)]^2$$

As suggested in the previous section on fitting an ODE with Cauchy noise, the middle error might be another loss function besides least squares error such as the sum of absloute errors. As before, value of θ that optimises $H(\cdot)$ holding λ constant, defined by $\hat{\theta}(\lambda)$, is computed.

And finally, the outer problem attempts to determine the value of λ that minimises the prediction error (generalisation error) by minimising another function $F(\lambda,\hat{\theta}(\lambda),\hat{f}(t|\omega,\lambda)) = F(\lambda)$. There are several plausible choices for $F(\cdot)$, one could use leave-one-out cross-validation, one could partition the data set into a training set and a validation one, and let $F(\lambda)$ be the associated error for the validation set, one could use Generalised Cross-Validation. This criterion is in turn optimised to find the optimal λ.

Note that the three levels are somewhat isolated from each other and only interact by exchanging parameters downwards and optimal values upwards. The middle function $H(\cdot)$ for example only requires the value of the optimal $f(\cdot)$ evaluated at the choosen points t_i, and does not care about how these values were found or how $f(\cdot)$ is represented.

The inner problem consists of finding a function that minimises a certain criterion for a given set of parameters. As previously discussed, the complexity of such problems can increase fairly rapidly and require a considerable degree of non-Statistical expert knowledge and often must be essentially developed from scratch if the differential penalty changes too much. It is thus desirable that the inner problem can be solved with already existing methods and tools such as the FDA package or Data2LD to avoid the effort of having to develop one's own. Ideally, it should be possible for one to plug in existing code that can compute $H(\cdot)$ and the optimal function as required.

There is thus a considerable degree of potential modularity present in the Parameter Cascade that is not fully investigated in Ramsay and Cao's paper [3], and research that inherits that framework. The Parameter Cascade can be adapted to heavy-tailed errors for example, by using appropriate loss functions for the various levels of the problem.

Not only is it good research practice to have mostly independent components that can be tackled and verified seperately before being combined, it is also good practice from a software engineering perspective because the potential for complex interactions between different parts of code is reduced. This tends to save on debugging and testing requirements, which can be quite high when implenenting codes for FDA.

The Data2LD package is fairly tightly coupled. Rather than use R's built-in routines to implement the Quasi-Newton alogrithm for example to optimise the associated middle problem, the authors wrote their own code. With Brent's method however, there is more seperation, which makes it very easy to build optimisation routines on top of other code. This substantially elides the cost and effort of tackling the inner problem and allows one to concentrate on the statistical questions such as fitting the model to data.

Melanoma Data

This derivative free optimisation strategy was applied to fitting the melanoma dataset with a parameterised linear differential operator:

$$L_\omega = D^2 - \omega^2 D^4. \tag{1.3}$$

The inner problem consists of finding the function $f(t)$ that minimises a penalised regression problem of the form:

$$PENSSE(f; \omega, \lambda) = \sum (x_i - f(t_i))^2 + \lambda \int |L_\omega f(t)|^2 dt$$

The penalty term measures the extent to which a function lies outside of the span of the functions $\{1, t, \cos(\omega t), \sin(\omega t)\}$.

The FDA package has routines that can do the numerical work of fitting the data with differential penalty given in (1.3) for given choices of λ and ω, and then report the associated mean square error.

Using Brent's method, the function $H(\omega; \mathbf{x}, \lambda)$ can be optimised with respect to ω for a given fixed λ. In turn, the outer objective function can be parameterised in terms of λ and the associated optimal choice of ω. This defines an objective function that can be again optimised to find the optimal choice of λ.

For ω, Figure 1.5 shows tht the error is not particularly sensitive to small deviations from the optimal value even for fairly high values of λ. This suggests that the fitted curve will be adjusted to ensure no substantial increase in the error so long as ω isn't altered too much from the optimal value.

Heuristcally speaking, a flat objective function in the neighbourhood of the optimal point as can be seen in Figure 1.5 increases the uncertainty in estimation because it is more difficult to argue that the optimal value is definitively better than adjacent ones. The loss function associated with a given fitting problem only approximates the 'true' loss function as the sample size goes to infinity.

If λ is set too low, the optimal value of ω is numerically indistinguishable from zero. This is the case when ω is optimised for the value of λ that minimises the GCV, Brent's method reports zero as the optimal value to within its default tolerance.

For λ, the curve has two critical points, with an asymptote as λ tends to infinity.

A huge advantage of this approach compared to Data2LD's use of quasi-Newton methods is that it allows for the use of more robust loss functions since no use at all is made of derivatives.

Suppose one wanted to choose ω to minimise the Median Absolute Deviation - median($|y_i - \hat{f}(t_i|\omega, \lambda)|$) - instead of the least squares error. This loss function is choosen instead of the usual L1 error for the sake of demonstration because the L1 error might sometimes be tackled using a generalised version of gradient descent known as the subgradient method, while getting any kind of a derivative for MAD is difficult. It is quite simple, one just replaces the code that computes the least squares error with a few lines of R code that computes the MAD and run the optimisation routine again. It can be seen in Figures 1.6 and 1.7 that the MAD gives similar results to the usual least squares criterion, which suggests that both estimators are mutually consistent with each other.

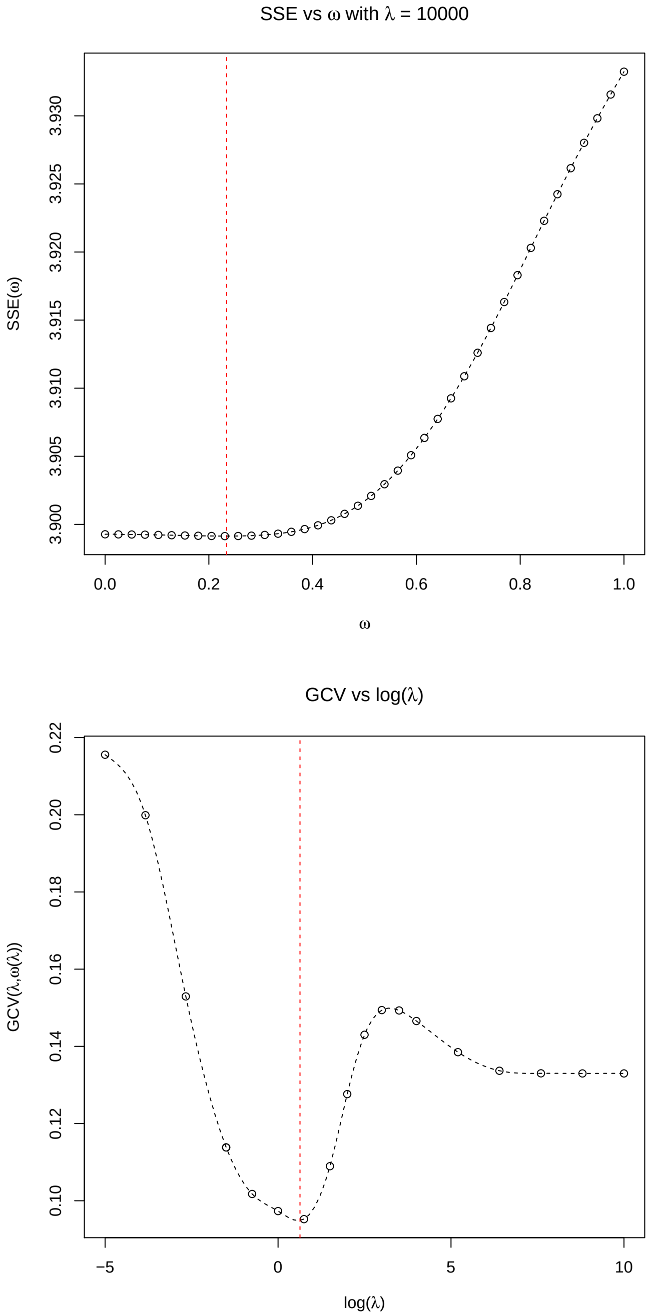

Figure 1.5: Plots of the middle and outer optimisation problems.

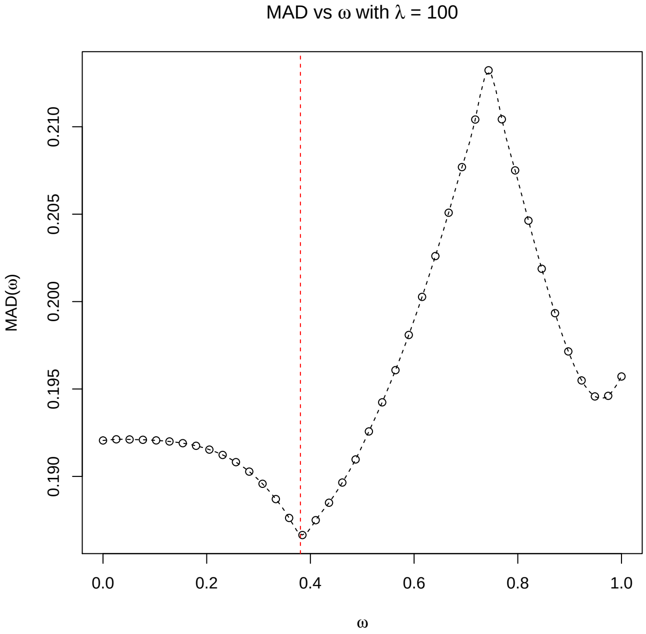

Figure 1.6: Plot of the middle optimisation problem with MAD used as a loss function

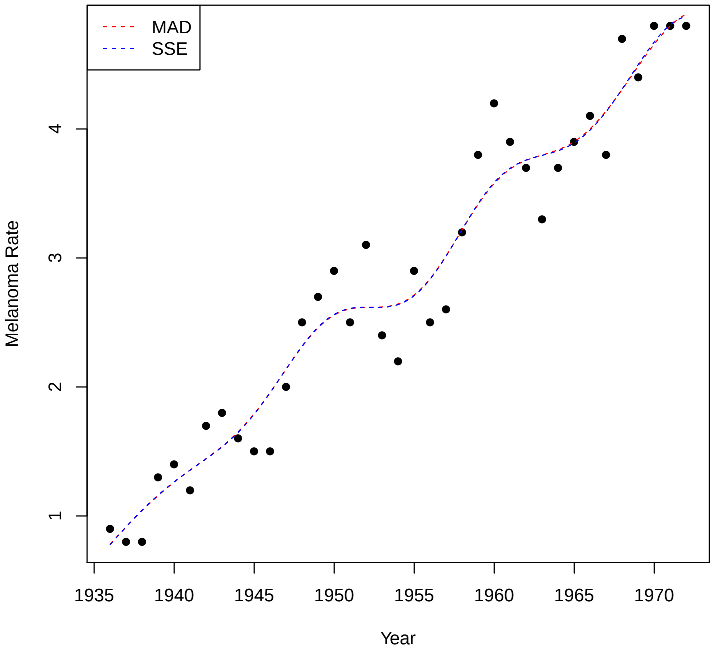

Figure 1.7: Comparison of fits for MAD and SSE criteria for middle problem

Bibliography

[1] Stephen Boyd and Lieven Vandenberghe. *Convex optimization.* Cambridge university press, 2004.

[2] Richard P Brent. *Algorithms for minimization without derivatives.* Courier Corporation, 2013.

[3] Jiguo Cao and James O Ramsay. Parameter cascades and profiling in functional data analysis. *Computational Statistics*, 22(3):335–351, 2007.

[4] Kwun Chuen Gary Chan. Acceleration of expectation-maximization algorithm for length-biased right-censored data. *Lifetime data analysis*, 23(1):102–112, 2017.

[5] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[6] Bengt Fornberg. Generation of finite difference formulas on arbitrarily spaced grids. *Mathematics of computation*, 51(184):699–706, 1988.

[7] PR Graves-Morris, DE Roberts, and A Salam. The epsilon algorithm and related topics. *Journal of Computational and Applied Mathematics*, 122(1-2):51–80, 2000.

[8] David R Hunter and Kenneth Lange. A tutorial on MM algorithms. *The American Statistician*, 58(1):30–37, 2004.

[9] Eugene Isaacson and Herbert Bishop Keller. *Analysis of numerical methods.* Courier Corporation, 2012.

[10] Jack Kiefer. Sequential minimax search for a maximum. *Proceedings of the American mathematical society*, 4(3):502–506, 1953.

[11] Kenneth Lange. *Optimization.* Springer, 2004.

[12] Kenneth Lange. *Numerical analysis for statisticians.* Springer Science & Business Media, 2010.

[13] Kenneth Lange. The MM algorithm. `https://www.stat.berkeley.edu/~aldous/Colloq/lange-talk.pdf`, April 2007. [Online, accessed 18-September-2019].

[14] Randall J LeVeque. *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems*, volume 98. Siam, 2007.

[15] Geoffrey McLachlan and Thriyambakam Krishnan. *The EM algorithm and extensions*, volume 382. John Wiley & Sons, 2007.

[16] J Nocedal and SJ Wright. *Numerical Optimisation.* Springer verlag, 1999.

[17] Naoki Osada. *Acceleration methods for slowly convergent sequences and their applications.* PhD thesis, PhD thesis, Nagoya University, 1993.

[18] Yudi Pawitan. *In all likelihood: statistical modelling and inference using likelihood.* Oxford University Press, 2001.

[19] R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2013. ISBN 3-900051-07-0.

[20] Walter Rudin et al. *Principles of mathematical analysis*, volume 3. McGraw-hill New York, 1964.

[21] Christopher G Small. A survey of multidimensional medians. *International Statistical Review/Revue Internationale de Statistique*, pages 263–277, 1990.

[22] Karline Soetaert, Thomas Petzoldt, and R. Woodrow Setzer. Solving differential equations in R: Package deSolve. *Journal of Statistical Software*, 33(9):1–25, 2010.

[23] Keller Vandebogart. Method of quadratic interpolation. `http://people.math.sc.edu/kellerlv/Quadratic_Interpolation.pdf`, September 2017. [Online; accessed 13-September-2019].

[24] Jet Wimp. *Sequence transformations and their applications.* Elsevier, 1981.

[25] Tong Tong Wu, Kenneth Lange, et al. The MM alternative to EM. *Statistical Science*, 25(4):492–505, 2010.