Chapter 1

Hierarchial Estimationn and the Parameter Cascade

First, a hierarchail parameter estimation method for PDEs is introduced. The Parameter Cascade is then introduced. Finally, a discussion on quasi-linear problems.

1.1 Hierarchial fitting of a Partial Differential Equation

A linear PDE that would be analogous to the linear ODE used to model the Reflux data would be the Transport Equation:

$$\frac{\partial u(x,t)}{\partial t} + \beta \frac{\partial u(x,t)}{\partial x} = 0$$

A general solution to the Transport Equation is given by:

$$u(x,t) = f(x - \beta t)$$

The function $f(\cdot)$ is unspecified. The solution $u(x,t)$ is constant along the rays $x = \beta t + C$. The solution is an animation of the shape $f(x)$ moving to the right at fixed speed β.

The ODE $y'(t) + \beta y(t) = 0$ can be thought of as a simplifcation of the Transport Equation, where it is assumed that $u(x,t)$ only varies with time, and not with space. It is apparent that this PDE has a much richer solution structure than is the case for the ODE, which only has solutions of the form $Ae^{-\beta t}$. Statistically speaking, fitting the Transport Equation to observed data is a semi-parametric problem because one of the parameters to be estimated is a function. The problem of fitting the Transport Equation is also a transformation model such as that used for the Box-Cox transformation, since the plot of $u(x,t)$ with respect to x at a fixed time t is a transformed version of $f(x)$, the curve at $t = 0$.

If the parameter governing the transformation process - β - is known, $f(\cdot)$ is reasonably easy to estimate. Suppose there were n observed values y_i at time t_i and location x_i. It has already been established that the value observed at a point x at time t depends only on $x - \beta t$. The function $f(\cdot)$ could thus be estimated by non-parametrically regressing the observed values at y_i against $x_i - \beta t_i$

What if β were unknown? The above discussion suggests a hierarchial approach to estimation: for a given choice of β, to fit an associated function $f(\cdot|\beta)$ using an appropriate non-parametric estimation method, and compute the associated least squares error. Let $H(\beta)$ be the function that associates each β with its sum of squared error:[1]

$$H(\beta) = \sum_{i=1}^{n} [y_i - f(x_i - \beta t_i|\beta)]^2$$

The problem of minimising $H(\beta)$ is a non-linear least squares problem that is also a two level hierachial estimation problem. The inner level consists of non-parametrically fitting a function to the set of points $\{(y_i, x_i - \beta_i)\}$ given β. The associated sum of squared errors is then returned as $H(\beta)$. The outer level entails optimising the profiled objective function $H(\beta)$.

This is a broad fitting strategy where different statistical and optimisation approaches can be swapped in and out as needed. There are several ways to tackle the inner function - LOESS; Kernel Regression; Penalised Splines, etc. The least squares loss function could be replaced with another one as suits the problem. There are many methods for optimising $H(\beta)$ that might be attempted - subgradient methods if $H(\beta)$ is convex, gradient descent, Gauss-Newton Method, derivative-free methods and so on.

1.2 The Two-Stage Parameter Cascade

Consider the following penalised regression problem:

$$PENSSE(f, \theta) = \sum_{i=1}^{N} (y_i - f(t_i))^2 + \lambda \int_0^1 |T_\theta f|^2 dt.$$

Here T_θ is some differential operator, that is parameterised by an unknown θ that is to be estimated.

T_θ can be an ordinary differntial operator or a partial differential operator; linear, quasi-linear, or nonlinear.

There are two statistical objects to be estimated here: the parameter θ, and the function $f(t)$.

Ramsay and Cao propose the following hierarchial approach to estimation[4]: Given a fixed value of θ, let $f(t|\theta)$ denote the function that minimises $PENSSE(f, \theta)$ For a given value of θ, it's associated mean square error is then defined by:

[1] In case the left hand side might be slightly unclear - for the ith observation, the associated function $f(\cdot|\beta)$ is evaluated at $x_i - \beta t_i$.

$$SSE(\theta) = \sum_{i=1}^{N} [y_i - f(t_i|\theta)]^2$$

By making $f(t)$ dependent on θ, the fitting problem has been reduced to a non-linear least squares problem.

This leaves the issue of estimating the optimal value of θ - Ramsay and Cao propose the use of gradient descent.

For a given value of θ, $f(t|\theta)$ is found. These two values together are then used to compute $MSE(\theta)$ and $\nabla MSE(\theta)$. Finally, a new value of θ is computed by perturbing θ in the direction of the gradient.This scheme is sketched out in Figure 1.1.

It is assumed that $f(t)$ can be represented by a finite vector $\mathbf{c}$ associated with an appropriate basis. This leads to a pair of nested optimisation problems: the *Inner Optimisation* involves finding the value of $\mathbf{c}$ that minimises the penalised least squares criterion given θ, and the *Middle Optimisation* entails finding the of value of θ that minimises $MSE(\theta)$.

There is thus a 'cascade' of estimation problems, where the results of the lower level estimation problems feed back in to the higher level ones.

Note that every time a new value of θ is introduced, the associated function $f(t|\theta)$ must be computed from scratch. The middle optimisation can thus generate many inner optimisation subproblems as the parameter space is explored, and these in turn could require multiple iterations to complete if no explicit formula for $\mathbf{c}$ given θ is available.

Figure 1.1 is an idealised sketch of the Parameter Cascade as Ramsay and Cao would understand it. The main abstraction is that the step of computing $f(t|\theta)$ is presented as a single atomic and organic step, even though it could be a complex process in its own right. This risks masking some of the comptuational work that is happening. A more realistic description is provided in Figure 1.2. In this thesis, Parameter Cascade problems that cannot be differentiated easily or at all are considered.

1.3 Three Stage Parameter Cascade

Up to this point, the structural parameter λ has been treated as fixed. But it is possible to extend the Parameter Cascade to estimate λ.

It is necessary to introduce an *Outer Criterion* $F(\lambda)$ that determines how good a given choice of λ is.

A common choice of outer criterion is Generalised Cross Validation[4, 28].

Just as the problem of fitting a function $f(\cdot|\theta)$ can generate an optimisation subproblem, that of fitting a third level in the cascade can generate a series of subproblems to find the best parameter choice associated with a given value of λ, which in turn generates a series of subproblems to find the fitted function as the parameter space is explored.

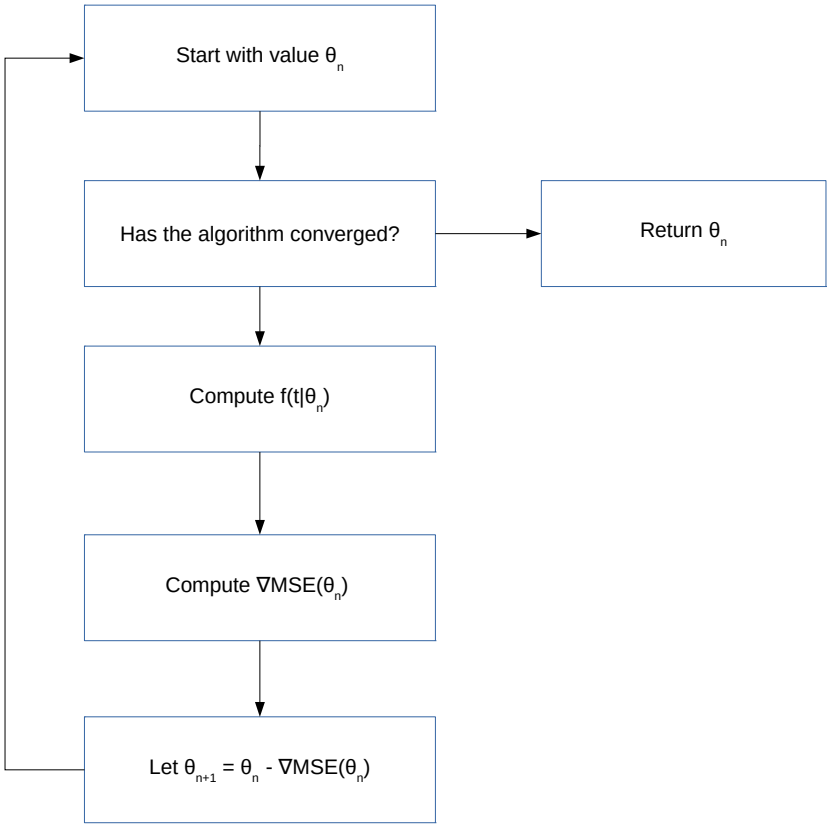

Figure 1.1: Two Stage Parameter Cascade (Simplified)

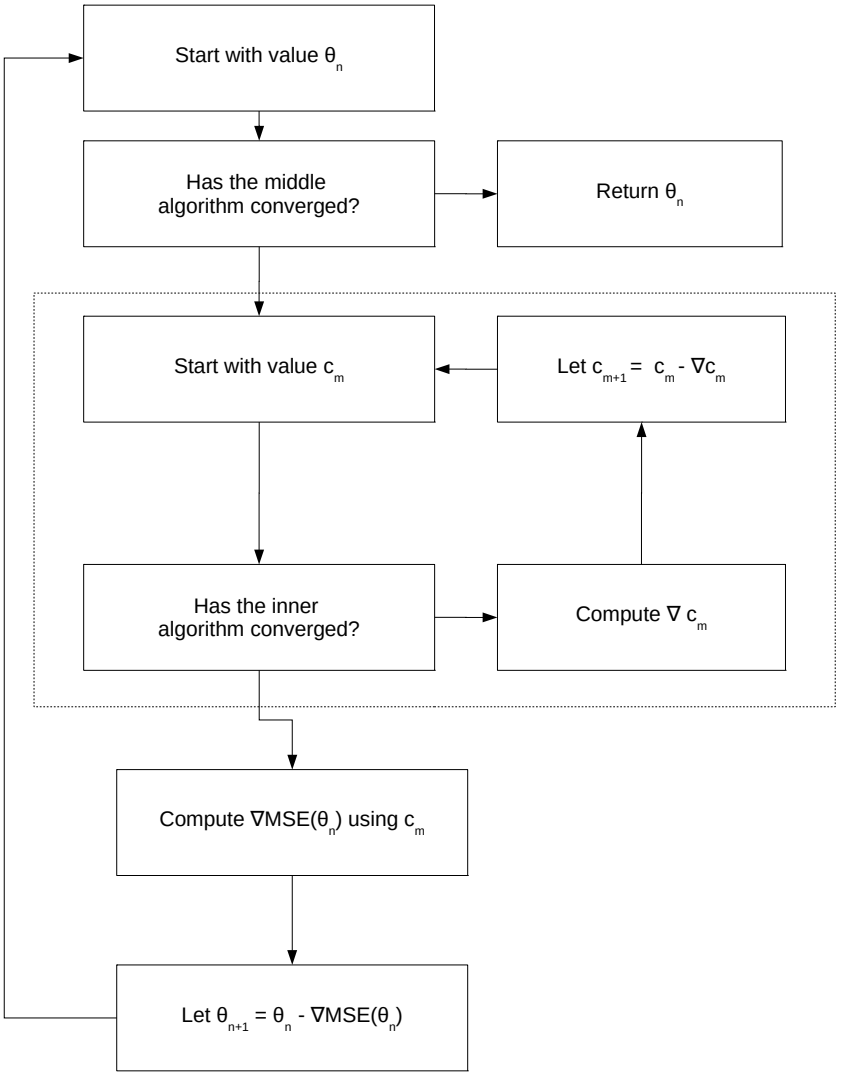

Figure 1.2: Schematic of the Two Stage Parameter Cascade With the Inner Optimisation Visible

As state in Chapter 1, neither the `FDA` nor `Data2LD` packages implement the three stage parameter cascade. They instead expect practioners to find the best choice of λ by cycling through a set of predetermined values or even just employing manual adjustment.

1.4 Dissecting the Data2LD Package

Data2LD uses a sophisticated two-level parameter cascade algorithm to fit parameters to the data, which is briefly described here.

The code for Data2LD is difficult to understand. For example, Data2LD hardcodes unnamed constants into the code. For example putting the number `3.14159` into code without context instead of `PI`. Allowing such 'Magic Numbers' is strongly discouraged because it makes code more error prone and difficult to understand [21].

1.4.1 How Data2LD Fits Parameters

The search directions used by Data2LD are the gradient descent direction:

$$\mathbf{p}_n = -\mathbf{g}_n \tag{S1}$$

and the Newton Direction:

$$\mathbf{p}_n = -\mathbf{H}_n^{-1}\mathbf{g}_n \tag{S2}$$

Data2LD uses four tests to determine how good a step is:[2]

- First Wolfe Condition:

$$f(\theta_n + \alpha_n \mathbf{p}_n) \leq f(\theta_n) + c_1 \alpha_n \mathbf{p}_n^\top \mathbf{g}_n \tag{T1}$$

- Second Wolfe Condition:

$$|\mathbf{p}_n^\top \nabla f(\theta_n + \alpha_n \mathbf{p}_n)| \leq c_2 |\mathbf{p}_n^\top \nabla f(\theta_n)| \tag{T2}$$

- Has the function even decreased compared to the previous iteration?

$$f(\theta_n + \alpha_n \mathbf{p}_n) \leq f(\theta_n) \tag{T3}$$

[2] `Data2LD` actually tests for the negation of **T3** and **T4**. For the sake of consistency the logical negations of the two tests used by `Data2LD` are presented here so that passing a test is consistently a good thing and failing consistently represents unsatisfactory or pathalogical behaviour.

- Has the slope along the search direction remained nonnegative?

$$\mathbf{p}_n^\top \nabla f(\theta_n + \alpha_n \mathbf{p}_n) \leq 0 \qquad (\mathbf{T4})$$

Written in terms of $\phi(\alpha) = f(\theta + \alpha \mathbf{p}_n)$ the tests are:

$$\phi(\alpha_n) \leq \phi(0) + c_1 \alpha_n \phi'(0) \qquad (\mathbf{T1'})$$
$$|\phi'(\alpha_n)| \leq c_2 |\phi'(0)| \qquad (\mathbf{T2'})$$
$$\phi(\alpha) \leq \phi(0) \qquad (\mathbf{T3'})$$
$$\phi'(\alpha) \leq 0 \qquad (\mathbf{T4'})$$

If **T1** and **T2** are satisfied, then the line search has converged completely. If **T3** has failed, this represents a total failure because it means the line search has failed to actually produce any improvement in the objective function. A failure in **T4** means the function has overshot a critical point.[3]

Depending on the outcome of the tests, Data2LD chooses the stepsize as follows:

- If **T1**, **T2**, and **T3** are passed, the algorithm terminates.
- If **T1** and **T1** are passed, or **T4** is passed; but **T3** is failed, it means that the slope is satisfactory, but the function has increased rather than decreased. Data2LD reduces the step size.
- If all four tests are failed, then the newest point is unsuitable entirely. Data2LD falls back on interpolation to try to find a critical point of $\phi(\alpha)$, falling back on quadratic interpolation methods if necessary.[4]

If the line search succeeds in reducing the objective function, Data2LD uses the Newton search direction for the next iteration. If the line search makes the objective function worse, the gradient descent direction is used. In the event of the line search making the objective function worse twice in a row, Data2LD returns an error.

Somewhat peculiarly, `Data2LD` does not make use of $\phi''(\alpha)$ despite being able to compute it easily.[5] One would think that the Newton-Raphson Method would be the

[3] If **T4** fails, this implies that $\mathbf{p}_n^\top \nabla f(\theta_n + \alpha_n \mathbf{p}_n)$ and $\mathbf{p}_n^\top \nabla f(\theta_n)$ are of opposite sign since $\mathbf{p}_n$ is choosen so that $\mathbf{p}_n^\top \mathbf{g}_n < 0$. The Intermediate Value Theorem means there is an $\bar{\alpha}$ between 0 and α_n such that $\mathbf{p}_n^\top \nabla f(\theta_n + \bar{\alpha} \mathbf{p}_n) = 0$, so that there is a critical point on the line segment between θ_n and $\theta_n + \alpha_n \mathbf{p}_n$.

[4] The line search code for the `Data2LD` is lightly commented and dense, all that one can be strictly certain of is that the method uses radicals to compute the next value of α, falling back on solving a linear equation if necessary. Getting the root of a quadratic is equivalent to finding a critical point of a cubic, and solving a linear equation is equivalent to finding the critical points of a quadratic.

[5] Differentiating the expression $\phi'(\alpha) = \mathbf{p}_n^\top \nabla f(\mathbf{x}_n + \alpha \mathbf{p}_n)$ with respect to α yields that $\phi''(\alpha) = \mathbf{p}_n^\top \mathbf{H}(\alpha) \mathbf{p}_n$, where $\mathbf{H}(\alpha)$ denotes the Hessian of f evaluated at $\mathbf{x}_n + \alpha \mathbf{p}_n$.

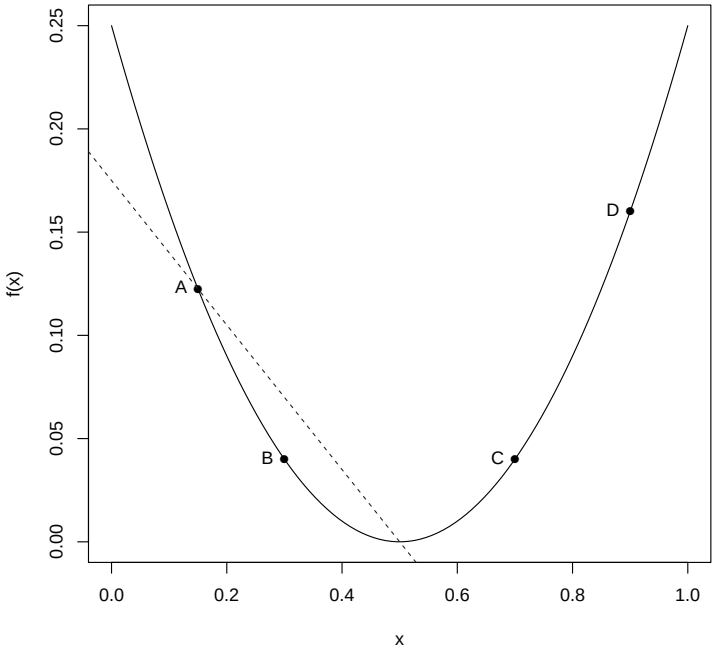

Figure 1.3: Point A is the initial point. Point B passes **T1** with $c_1 = 0.5$ and passes **T2** with $c_2 = 0.9$. Point C fails **T1** with $c_1 = 0.5$ and also fails **T3**, but passes **T4**, and passes **T2** with $c_2 = 0.9$. Point D fails all four tests.

first approach attempted to perform the line search before resorting to interpolation-based methods since it's both simpler to implement and faster to converge. The effort of computing $\phi''(\alpha)$ is mostly a sunk cost because of how the interface of `Data2LD` is defined.

Bibliography

[1] John P Boyd. *Chebyshev and Fourier spectral methods*. Courier Corporation, 2001.

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

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

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

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

[6] Edwin KP Chong and Stanislaw H Zak. *An introduction to optimization*, volume 76. John Wiley & Sons, 2013.

[7] 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.

[8] Ronald Aylmer Fisher. The wave of advance of advantageous genes. *Annals of eugenics*, 7(4):355–369, 1937.

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

[10] 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.

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

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

[13] Carl T Kelley. *Iterative methods for linear and nonlinear equations*, volume 16. Siam, 1995.

[14] Carl T Kelley. *Implicit filtering*, volume 23. SIAM, 2011.

[15] C.T. Kelley. A brief introduction to implicit filtering. `https://projects.ncsu.edu/crsc/reports/ftp/pdf/crsc-tr02-28.pdf`, 2002. [Online; accessed 12-October-2019].

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

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

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

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

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

[21] Steve McConnell. *Code complete*. Pearson Education, 2004.

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

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

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

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

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

[27] James Ramsay. Functional data analysis. *Encyclopedia of Statistics in Behavioral Science*, 2005.

[28] Jim O Ramsay, Giles Hooker, David Campbell, and Jiguo Cao. Parameter estimation for differential equations: a generalized smoothing approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(5):741–796, 2007.

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

[30] Larry Schumaker. *Spline functions: basic theory*. Cambridge University Press, 2007.

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

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

[33] Gerald Teschl. *Ordinary differential equations and dynamical systems*, volume 140. American Mathematical Soc., 2012.

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

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

[36] Stephen Wright. Optimization for data analysis. In Michael W. Mahoney, John C. Duchi, and John C. Duchi, editors, *The Mathematics of Data*, chapter 2, pages 49–98. American Mathematical Society and IAS/Park City Mathematics Institute and Society for Industrial and Applied Mathematics, 2018.

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