PLSC 503 – Spring 2017
Optimization

April 4, 2017

Stuff We Won't Cover Today

- Grid search / "hill climbing"
- Genetic algorithms
- Annealing methods
- Local search methods (tabu, etc.)
- many others...

The Basic Problem

Find

$$\max_{\hat{\boldsymbol{\beta}} \in \mathbb{R}^k} \ln L(\hat{\boldsymbol{\beta}} | Y, \mathbf{X})$$

Unconstrained optimization problem...

The Intuition

- Start with $\hat{\boldsymbol{\beta}}_\mathbf{0}$
- Adjust:

$$\hat{\boldsymbol{\beta}}_\mathbf{1} = \hat{\boldsymbol{\beta}}_\mathbf{0} + \mathbf{A_0}$$

- Repeat.

More Specifically...

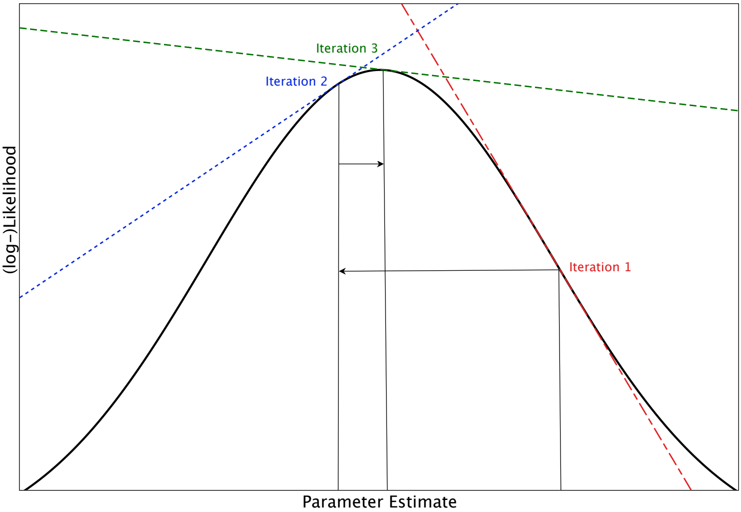
$$\hat{\boldsymbol{\beta}}_\ell = \hat{\boldsymbol{\beta}}_{\ell-\mathbf{1}} + \mathbf{A}_{\ell-\mathbf{1}}$$

$$\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}}_\ell \ni \hat{\boldsymbol{\beta}}_\ell - \hat{\boldsymbol{\beta}}_{\ell-1} (\equiv \mathbf{A}_\ell) < \tau$$

What's **A**?

$$\mathbf{A} = f[\mathbf{g}(\hat{\beta})]$$

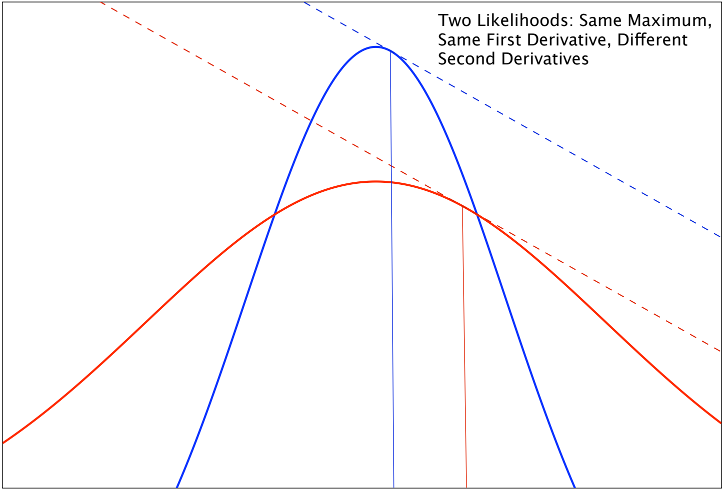
- $\mathbf{g}(\hat{\beta}) =$ "directionality" of change
  - $\mathbf{g}(\hat{\beta}_k) < 0 \rightarrow A_k < 0$
  - $\mathbf{g}(\hat{\beta}_k) > 0 \rightarrow A_k > 0$


Iteration 3
Iteration 2
Iteration 1
(log−)Likelihood
Parameter Estimate

“Steepest Ascent”

$$\mathbf{A}_\ell = \frac{\partial \ln L}{\partial \hat{\boldsymbol{\beta}}_\ell}$$

$$\hat{\boldsymbol{\beta}}_\ell = \hat{\boldsymbol{\beta}}_{\ell-1} + \frac{\partial \ln L}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}}$$


Two Likelihoods: Same Maximum,
Same First Derivative, Different
Second Derivatives

“Step Size”

$$\hat{\boldsymbol{\beta}}_\ell = \hat{\boldsymbol{\beta}}_{\ell-\mathbf{1}} + \lambda_{\ell-\mathbf{1}} \boldsymbol{\Delta}_{\ell-\mathbf{1}}$$

- $\boldsymbol{\Delta} \rightarrow$ *direction*
- $\lambda \rightarrow$ *amount* (“step size”)

Key: Hessian

$$\mathbf{H}(\hat{\boldsymbol{\beta}}) = \frac{\partial^2 \ln L}{\partial \hat{\boldsymbol{\beta}}^2}$$

How?

Newton-Raphson

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_\ell &= \hat{\boldsymbol{\beta}}_{\ell-1} - \left(\frac{\partial^2 \ln L}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}^2} \right)^{-1} \frac{\partial \ln L}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}} \\
&= \hat{\boldsymbol{\beta}}_{\ell-1} - \mathbf{H}(\hat{\boldsymbol{\beta}}_{\ell-1})^{-1} \mathbf{g}(\hat{\boldsymbol{\beta}}_{\ell-1})
\end{aligned} \tag{1}$$

Sidebar: Newton-Raphson, re-revealed

Taylor series, anyone?

$$f(X) \approx f(a) + f'(a)(x - a)$$

Here,

$$\frac{\partial \ln L}{\partial \hat{\boldsymbol{\beta}}_{\ell}} \approx \frac{\partial \ln L}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}} + \frac{\partial^2 \ln L}{\partial \hat{\boldsymbol{\beta}}^2_{\ell-1}}(\hat{\boldsymbol{\beta}}_{\ell} - \hat{\boldsymbol{\beta}}_{\ell-1})$$

What we really want...

$$\frac{\partial \ln L}{\partial \hat{\boldsymbol{\beta}}_\ell} = \mathbf{0}$$

Punch Line

$$\mathbf{0} \approx \frac{\partial \ln L}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}} + \frac{\partial^2 \ln L}{\partial \hat{\boldsymbol{\beta}}^2_{\ell-1}}(\hat{\boldsymbol{\beta}}_{\ell} - \hat{\boldsymbol{\beta}}_{\ell-1})$$

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{\ell} &\approx \hat{\boldsymbol{\beta}}_{\ell-1} - \left(\frac{\partial^2 \ln L}{\partial \hat{\boldsymbol{\beta}}^2_{\ell-1}}\right)^{-1} \frac{\partial \ln L}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}} \\ &\approx \hat{\boldsymbol{\beta}}_{\ell-1} - \mathbf{H}(\hat{\boldsymbol{\beta}}_{\ell-1})^{-1}\mathbf{g}(\hat{\boldsymbol{\beta}}_{\ell-1})\end{aligned}$$

Newton-Raphson

- Uses $\mathbf{H}(\hat{\boldsymbol{\beta}})^{-1}$ so
- *Calculates* $\mathbf{H}(\hat{\boldsymbol{\beta}})^{-1}$.

Modified Marquardt

- Used when $\mathbf{H}(\hat{\beta})$ isn't invertable
- Adds a constant $\mathbf{C}$ to $\text{diag}[\mathbf{H}(\hat{\beta})]$
- Variants: Add $\mathbf{C}(h_k)$

"Method of Scoring"

Uses:

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{\ell} &= \hat{\boldsymbol{\beta}}_{\ell-1} - \left[\mathsf{E}\left(\frac{\partial^2 \ln L}{\partial \hat{\boldsymbol{\beta}}^2_{\ell-1}}\right)^{-1}\right] \frac{\partial \ln L}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}} \\ &= \hat{\boldsymbol{\beta}}_{\ell-1} - \{\mathsf{E}[\mathbf{H}(\hat{\boldsymbol{\beta}}_{\ell-1})]\}^{-1}\mathbf{g}(\hat{\boldsymbol{\beta}}_{\ell-1})\end{aligned} \tag{2}$$

- Due to Fisher
- Advantages:
  - $\approx$ Newton-Raphson
  - <u>Can</u> be faster/simpler

Berndt, Hall^2, and Hausman ("BHHH")

Uses:

$$\hat{\boldsymbol{\beta}}_\ell = \hat{\boldsymbol{\beta}}_{\ell-1} - \left(\sum_{i=1}^{N} \frac{\partial \ln L}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}} \frac{\partial \ln L}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}}' \right)^{-1} \frac{\partial \ln L}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}}$$

Advantages:

- (Relatively) very easy to compute
- Reasonably accurate...

Other "Newton Jr.s"

- Davidson-Fletcher-Powell ("DFP")
- Broyden et al. ("BFGS")
- They are:
  - Very fast/efficient
  - Pretty bad at getting $-\left(\mathbf{H}(\hat{\boldsymbol{\beta}})\right)^{-1}$

Summary

Method	"Step size" (∂^2) matrix	Variance-Covariance Estimate
Newton	Inverse of the observed second derivative (Hessian)	Inverse of the negative Hessian
Scoring	Inverse of the expected value of the Hessian (information matrix)	Inverse of the negative information matrix
BHHH	Outer product approximation of the information matrix	Inverse of the outer product approximation

Software Issues: R

Lots of optimizers:

- `maxLik` package: options for Newton-Raphson, BHHH, BFGS, others
- `optim` (in `stats`) – quasi-Newton, plus others
- `nlm` (in `stats`) – nonlinear minimization "using a Newton-type algorithm"
- `newton` (in `Bhat`) – Newton-Raphson solver
- `solveLP` (in `linprog`) – linear programming optimizer

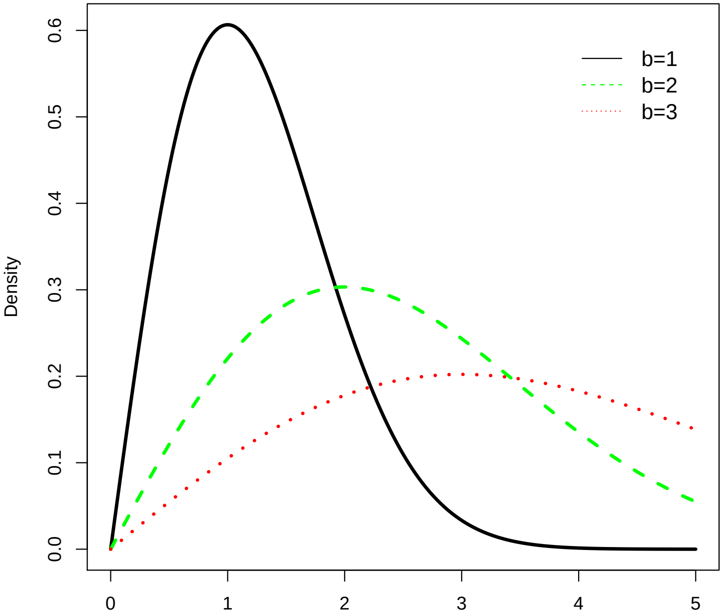
R : Using `maxLik`

- *Must* provide log-likelihood function
- Can provide $\mathbf{H}(\hat{\beta})$, $\mathbf{g}(\hat{\beta})$, both, or neither
- Choose optimizer (Newton, BHHH, BFGS, etc.)
- Returns an object of class `maxLik`

R : Examples

Rayleigh distribution:

$$\Pr(X) = \frac{x}{b^2} \exp\left[\frac{-x^2}{2b^2}\right]$$



b=1
b=2
b=3
Density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0
1
2
3
4
5

R : What We Like To See

```
> library(maxLik,distr)
> set.seed(7222009)
> U<-runif(100)
> rayleigh<-3*sqrt(-2*log(1-U))
> loglike <- function(param) {
+     b <- param[1]
+     ll <- (log(x)-log(b^2)) + ((-x^2)/(2*b^2))
+     ll
+  }
```

R : What We Like To See

```
> x<-rayleigh
> hats <- maxLik(loglike, start=c(1))
> summary(hats)
--------------------------------------------
Maximum Likelihood estimation
Newton-Raphson maximisation, 8 iterations
Return code 2: successive function values within tolerance limit
Log-Likelihood: -195.7921
1  free parameters
Estimates:
     Estimate Std. error t value Pr(> t)
[1,]   2.9168     0.1459      20  <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
--------------------------------------------
```

R : What We *Don't* Like To See

```
> Y<-c(0,0,0,0,0,1,1,1,1,1)
> X<-c(0,1,0,1,0,1,1,1,1,1)
> logL <- function(param) {
+   b0<-param[1]
+   b1<-param[2]
+   ll<-Y*log(exp(b0+b1*X)/(1+exp(b0+b1*X))) +
+       (1-Y)*log(1-(exp(b0+b1*X)/(1+exp(b0+b1*X))))
+   ll
+ }
```

R : What We *Don't* Like To See

```
> Bhat<-maxLik(logL,start=c(0,0))
> summary.maxLik(Bhat)
--------------------------------------------
Maximum Likelihood estimation
Newton-Raphson maximisation, 9 iterations
Return code 1: gradient close to zero
Log-Likelihood: -4.187887
2  free parameters
Estimates:
     Estimate Std. error t value Pr(> t)
[1,]   -104.3        Inf       0       1
[2,]    105.2        Inf       0       1
--------------------------------------------
```

Practical Optimization...

- Potential Problems
- Likely Causes
- Tips

Problems

Enemy # 1: Noninvertable $\mathbf{H}(\hat{\boldsymbol{\beta}})$

- "Non-concavity," "non-invertability," etc.
- (Some part of) the likelihood is "flat"
- Why? (Bob Dole...)

Problems

Identification

- Possible due to functional form alone...
- "Fragile"
- Manifestation: parameter instability

Poor Conditioning

- Numerical issues
- Potentially:
  - Collinearity
  - Other weirdnesses (nonlinearities)

Potential Causes

- Bad specification!
- Missing data
- Variable scaling
- Typical $\Pr(Y)$

Hints

- T-h-i-n-k!
- Know thy data
- Keep an eye on your iteration logs...
- Don't overreach