Gov 2001 Section 3: Optimization

Konstantin Kashin [1]

February 13, 2013

[1]Thank you to the previous TFs of Gov 2001, particularly Jen Pan and Brandon Stewart for contributing substantially to this material.

Outline

Outline

Bernoulli Example

Suppose we observe the following data from a dichotomous random variable Y: $\{1, 0, 0, 1, 1\}$.

Let's assume that Y is distributed Bernoulli with some constant probability of seeing a one across observations. We might also assume that observations are independent.

The model:

1. $Y_i \sim f_{\text{bern}}(y_i|\pi_i)$.
2. $\pi_i = \pi$.
3. Y_i and Y_j are independent for all $i \neq j$.

Bernoulli Example

1. What's the PMF for a Bernoulli again?

$$Y_i = \begin{cases} 1 & \text{with probability } \pi \\ 0 & \text{with probability } 1-\pi \end{cases}$$

2. So what's the joint distribution for our data conditional on π?

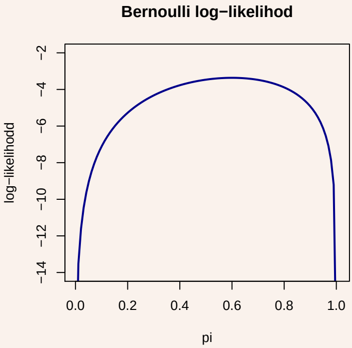
$$\begin{aligned} P(\boldsymbol{y}|\pi) &= P(Y_1 = 1, Y_2 = 0, \ldots, Y_5 = 1|\pi) \\ &= P(Y_1 = 1|\pi) \cdot P(Y_2 = 0|\pi) \cdots P(Y_5 = 1|\pi) \\ &= \pi \cdot (1-\pi) \cdot (1-\pi) \cdot \pi \cdot \pi \\ &= \pi^3 (1-\pi)^2 \end{aligned}$$

3. Recalling that $L(\theta|\boldsymbol{y}) \propto P(\boldsymbol{y}|\theta)$, let's find our likelihood function.

$$L(\pi|\boldsymbol{y}) \propto \pi^3(1-\pi)^2.$$

4. Let's take the natural logarithm, which is a relatively unimportant step now but will be essential for computational reasons later on.

$$\begin{aligned} \ln\left[\pi^3(1-\pi)^2\right] &= \ln(\pi^3) + \ln((1-\pi)^2) \\ &= 3\ln\pi + 2\ln(1-\pi) \end{aligned}$$



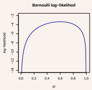
Where is the maximum, and how could we find it analytically using

$$\ln L(\pi|\boldsymbol{y}) = 3\ln\pi + 2\ln(1-\pi)?$$

5. Let's find the 1st derivative of $\ln L(\pi|\boldsymbol{y})$, which is a function which tells us the slope of a line tangent to our function at any point. We want to know at which value of π the slope of the tangent line is zero.

$$
\begin{aligned}
\frac{\partial \ln L(\pi|\boldsymbol{y})}{\partial \pi} &= \frac{\partial}{\partial \pi}\left[3 \ln \pi + 2 \ln(1-\pi)\right] \\
&= \frac{3}{\pi} + \frac{2}{1-\pi} \cdot (-1)
\end{aligned}
$$

When this function is equal to zero, $\pi = \frac{3}{5}$. Q: Assuming we hadn't seen



how would we know if $\pi = \frac{3}{5}$ was a maximum or a minimum?

6. Check the 'critical value' (proposed maximum or minimum) by examining whether the slope of the function is decreasing or increasing at the critical value.

$$\frac{\partial^2 \ln L(\pi^* | \boldsymbol{y})}{\partial \pi^2} = -20.83$$

This is **definitely negative**, so we have found the **maximum**.

First Derivative in R

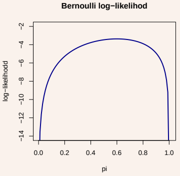
As shown above, the manual calculation is:

$$
\begin{aligned}
\frac{\partial \ln L(\pi|\boldsymbol{y})}{\partial \pi} &= \frac{\partial}{\partial \pi}\left[3 \ln \pi + 2 \ln(1-\pi)\right] \\
&= \frac{3}{\pi} + \frac{2}{1-\pi} \cdot (-1)
\end{aligned}
$$

<u>Note</u>: this can also be done in R using `deriv()`:

```
> deriv(f(x) ~ 3*log(x) + 2*log(1-x), "x")
expression({
    .expr3 <- 1 - x
    ...
    .grad[, "x"] <- 3 * (1/x) - 2 * (1/.expr3)
    ...
```

Second Derivative for Bernoulli Example in Detail



$$\begin{aligned}
\frac{\partial^2 \ln L(\pi|\boldsymbol{y})}{\partial \pi^2} &= \frac{\partial}{\partial \pi}\left[\frac{3}{\pi} + \frac{2}{1-\pi}\cdot(-1)\right] \\
&= -\frac{3}{\pi^2} - \frac{2}{(1-\pi)^2} \\
&= -\frac{3}{.6^2} - \frac{2}{(1-.6)^2} \\
&= -20.83.
\end{aligned}$$

This is **definitely negative**, so we have found the **maximum**.

Outline

Optimization

The previous example we just developed is an example of **optimization**: the process of minimizing or maximizing a function by systematically choosing the values of variables from within an allowed set. For example:

$$\min_{x \in [-\infty, \infty]} f(x) = -\frac{1}{2}(3-x)^2$$

- $f(x)$ is called the objective function
- x is the parameter (for us, β, π, σ, etc.)
- $x \in [-\infty, \infty]$ is the allowed set or the parameter space

Two ways to solve:

1. analytically (we just did this using derivatives)
2. numerically (we'll do this in a minute)

Analytical Optimization: Univariate Case

Step One: Take the first derivative of the function and identify the critical value(s).

- The derivative of a function at a value x_0, denoted by $f'(x_0)$ or $\frac{\partial f}{\partial x}(x_0)$, is the instantaneous rate of change in $f(\cdot)$ at x_0.
- It partially describes the behavior of a function on an interval [a,b]
  - If $f^{'}(x) > 0$ for all $x \in [a, b]$, then f is increasing on the interval $[a, b]$
  - If $f^{'}(x) < 0$ for all $x \in [a, b]$, then f is decreasing on the interval $[a, b]$
  - If $f^{'}(x) = 0$ at some $x \in [a, b]$ then we say x is a critical value of f. Critical values may be maxima, minima, or saddle points.

Analytical Optimization: Univariate Case

Step Two: Compute the second derivative of the function at the critical value(s) and evaluate.

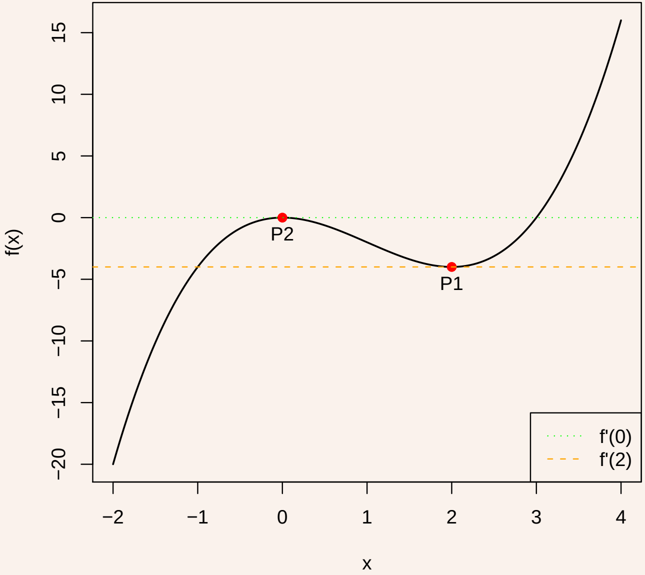
- The second derivative of a function $f''(x)$ or $\frac{\partial^2 f}{\partial x \partial x}(x)$ is the derivative of the derivative, or the rate of change of the rate of change.
- Use the following to evaluate your critical value(s):
  - If $f'(x_0) = 0$, and $f''(x_0) < 0$, then x_0 is a maximum
  - If $f'(x_0) = 0$ and $f''(x_0) > 0$, then x_0 is a minimum
  - If $f'(x_0) = 0$ and $f''(x_0) = 0$, then x_0 may be a minimum, a maximum, or neither.

Skip to numerical optimization

Analytical Optimization: Univariate Case

For example:

- Find extreme value of: $f(x) = x^3 - 3x^2$
- $f'(x) = 3x^2 - 6x = 3x(x-2)$ so $f'(x) = 0$ for $x = 0$ and $x = 2$.
- Critical values are:
  - $P_1 = (0, f(0)) = (0, 0)$
  - $P_2 = (2, f(2)) = (2, -4)$
- Second derivative test: $f''(x) = 6x - 6 = 6(x-1)$
- Evaluate second derivative at critical values:
  - $f''(2) = 6$ so P_1 will be a minimum
  - $f''(0) = -6$ so P_2 will be a maximum


f(x)
x
P1
P2
f'(0)
f'(2)
15
10
5
0
−5
−10
−15
−20
−2
−1
0
1
2
3
4

Numerical Optimization

For problems of only slightly more complexity, using derivatives and solving for parameters in order to maximize may be not only impractical but impossible.

There are a number of functions in R which can be used to optimize functions, but the one we will use most heavily is `optim()`.

Numerical Optimization Example

`optim()` takes a starting value (`par`) and a function (`fn`) as its main arguments.

```
ll <- function(pie) return(3*log(pie) + 2*log(1-pie))

optim(par = .3, fn = ll, control = list(fnscale = -1),
      method = "BFGS", hessian = TRUE)
```

What are these three extra arguments?

1. `fnscale` multiplies the function by the given constant. As a default `optim()` finds the minimum so multiplying our function by -1 fools `optim()` into finding the maximum.
2. `method` is the variety of algorithm used to find the maximum. More on this in a second.
3. `hessian = TRUE` requests that `optim` return a matrix of second derivatives which in the above case will be 1x1.

Numerical Optimization Example Cont.

```
$par
[1] 0.6000034

$value
[1] -3.365058
...
$hessian
          [,1]
[1,] -20.83365

$Warning messages:
1: In log(1 - x) : NaNs produced
2: In log(1 - x) : NaNs produced
```

What is `optim` doing?

It depends on your choice in the `method` argument.

- `Nelder-Mead`: this is the default; it is slow but somewhat robust to non-differentiable functions.
- `BFGS`: a quasi-Newton Method; it is fast but needs a well behaved objective function.
- `L-BFGS-B`: similar to BFGS but allows box-constraints (i.e. upper and lower bounds on variables).
- `CG`: conjugate gradient method, may work for really large problems (we won't really use this).
- `SANN`: uses simulated annealing – a stochastic global optimization method; it is very robust but *very* slow.

Newton's Method

Newton's method: a pretty good approach for a continuous and twice-differentiable function. We'll look at a univariate function here. Suppose we know our function $f(\cdot)$ and we have a starting value of x_0.

Our goal is to find move from x_0 to x_1 such that $f'(x_1) = 0$ (or is at least closer to 0 then than at x_0). This will be a sequential process of

approximation and eventually $f'(x_n)$ will be close enough to zero to let us declare that x_n a critical value.

Taylor Series Expansion: A Staple of Calculus

We can approximate function $f(\cdot)$ at point a using a Taylor series expansion:

$$\boxed{f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \ldots}$$

In this course, we'll work with the first- and second-order Taylor polynomials:

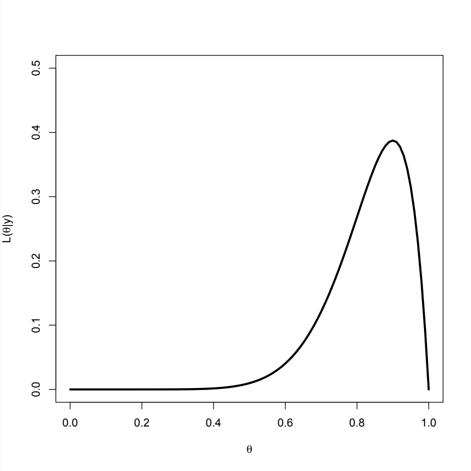
$$f(x) \approx f(a) + f'(a)(x-a)$$

$$f(x) \approx f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2$$

Skip to multivariate optimization

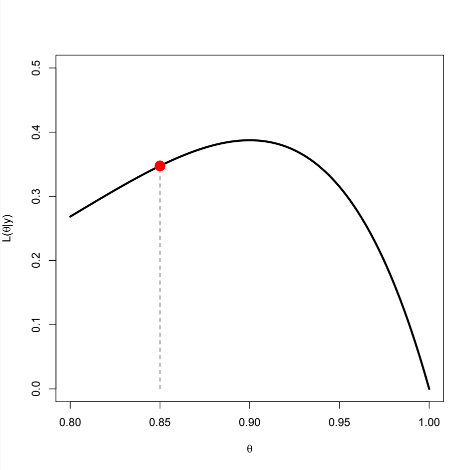
Likelihood of Binomial Distribution

This is the likelihood of a binomial distribution with 10 trials ($N = 10$) from which we drew one observation: $y = 9$.



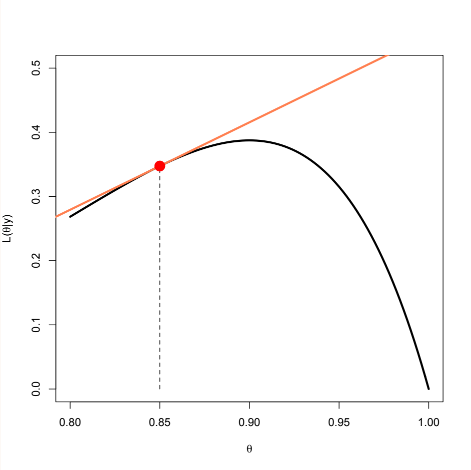
Likelihood of Binomial Distribution: Zoomed In

We want to approximate the likelihood curve around $\theta_0 = 0.85$.



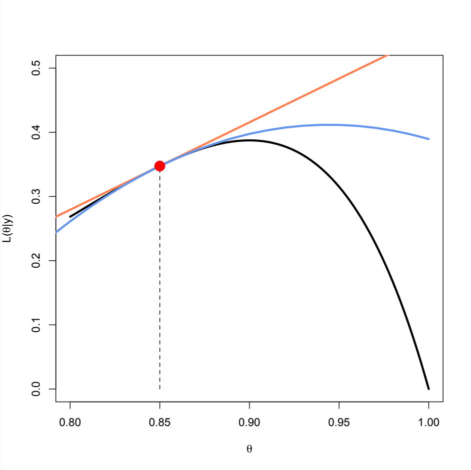
Likelihood of Binomial Distribution: Zoomed In

We can approximate the likelihood around $\theta_0 = 0.85$ using a first-order Taylor polynomial.



Likelihood of Binomial Distribution: Zoomed In

We can improve our approximation of the likelihood around $\theta_0 = 0.85$ by using a second-order Taylor polynomial.



Why Don't We Just Maximize the Second–Order Taylor Polynomial?

We can write a second-order Taylor expansion around θ_0 as:

$$f(\theta) \approx f(\theta_0) + f'(\theta_0)(\theta - \theta_0) + \frac{f''(\theta_0)}{2!}(\theta - \theta_0)^2$$

To maximize, take the derivative with respect to θ and set it equal to 0:

$$f'(\theta) = f'(\theta_0) + f''(\theta_0)(\theta - \theta_0) = 0$$

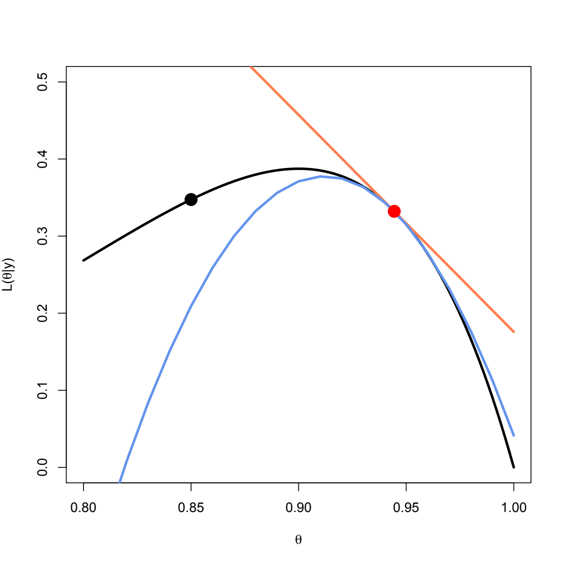
Rearranging:

$$\theta = \theta_0 - \frac{f'(\theta_0)}{f''(\theta_0)}$$

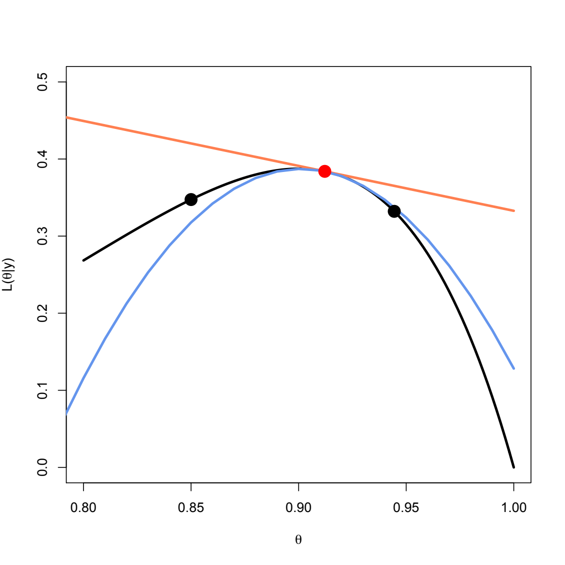
We've Just Derived the Update Step We Wanted!

$$\boxed{\theta_{n+1} = \theta_n - \frac{f'(\theta_n)}{f''(\theta_n)}}$$

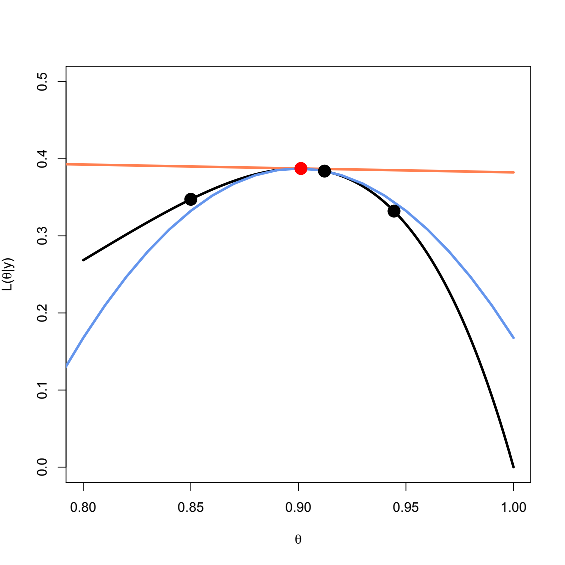
First Iteration



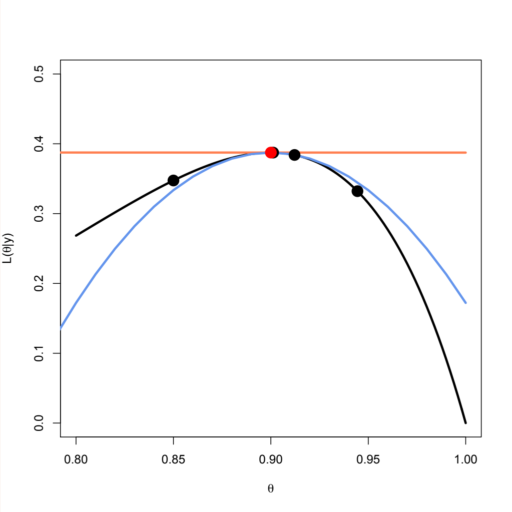
Second Iteration



Third Iteration



Fourth Iteration



Skip to Newton-Raphson properties
Skip to multivariate optimization

Another Example of Taylor Series Approximation

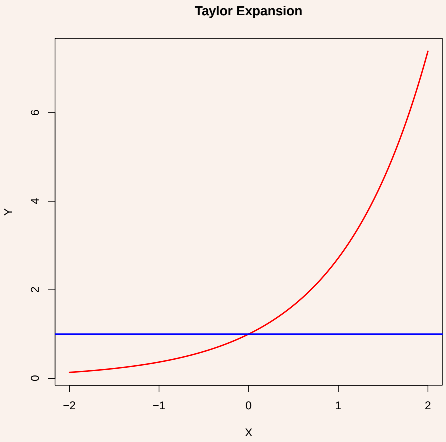


Figure : The exponential function, $g(x) = e^x$, and the Taylor Series approximation: $x_0 = 0$, $g_0(x_1) = 1$ (from Wikipedia)

Another Example of Taylor Series Approximation

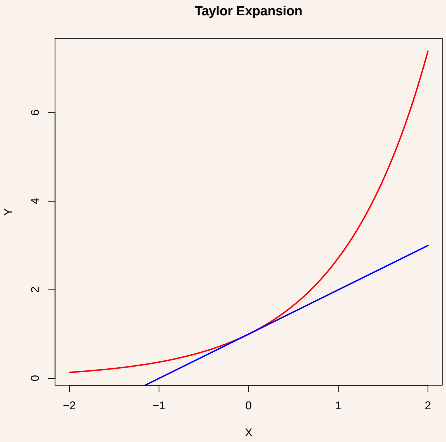


Figure : The exponential function, $g(x) = e^x$, and the Taylor Series approximation: $x_0 = 0$, $g_1(x_1) = 1 + x_1$ (from Wikipedia)

Another Example of Taylor Series Approximation

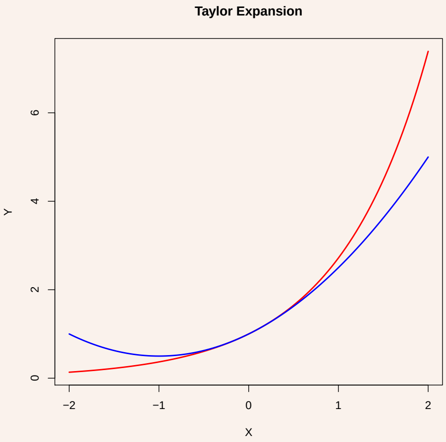


Figure : The exponential function, $g(x) = e^x$, and the Taylor Series approximation: $x_0 = 0$, $g_2(x_1) = 1 + x_1 + \frac{x^2}{2}$ (from Wikipedia)

Another Example of Taylor Series Approximation

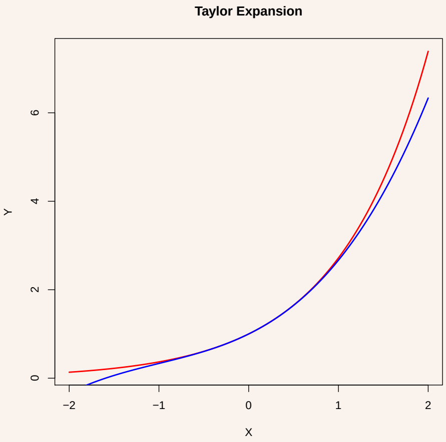


Figure : The exponential function, $g(x) = e^x$, and the Taylor Series approximation: $x_0 = 0$, $g_3(x_1) = 1 + x_1 + \frac{x^2}{2} + \frac{x^3}{6}$ (from Wikipedia)

Another Example of Taylor Series Approximation

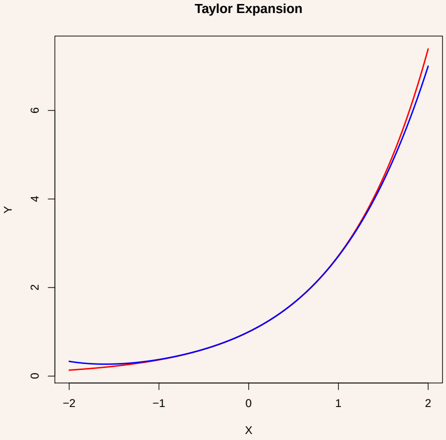


Figure : The exponential function, $g(x) = e^x$, and the Taylor Series approximation: $x_0 = 0$, $g_4(x_1) = 1 + x_1 + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24}$ (from Wikipedia)

Newton in Action: Bernoulli Example Revisited

Let's maximize our likelihood: $3 \ln \pi + 2 \ln(1 - \pi)$.

Recall that $L'(\pi) = \frac{3}{\pi} - \frac{2}{1-\pi}$ and $L''(\pi) = -\frac{3}{\pi^2} - \frac{2}{(1-\pi)^2}$.

Starting at $\pi_0 = .3$, we use our updating formula:

$$\pi_1 = \pi_0 - \frac{L'(\pi_0)}{L''(\pi_0)} = .3 - \frac{L'(.3)}{L''(.3)} = 0.4909.$$

Now use $\pi_1 = .4909$ as a starting value.

$$\pi_2 = \pi_1 - \frac{L'(\pi_1)}{L''(\pi_1)} = .4909 - \frac{L'(.4909)}{L''(.4909)} = 0.5991.$$

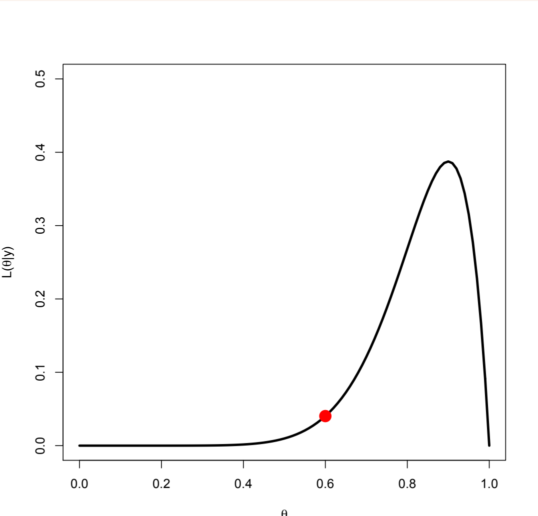
So we're already there!

Properties of Newton-Raphson

- Converges quickly
- Can get stuck in local minima/maxima
- Can have troubles with root jumping
- Won't walk at all on a flat space

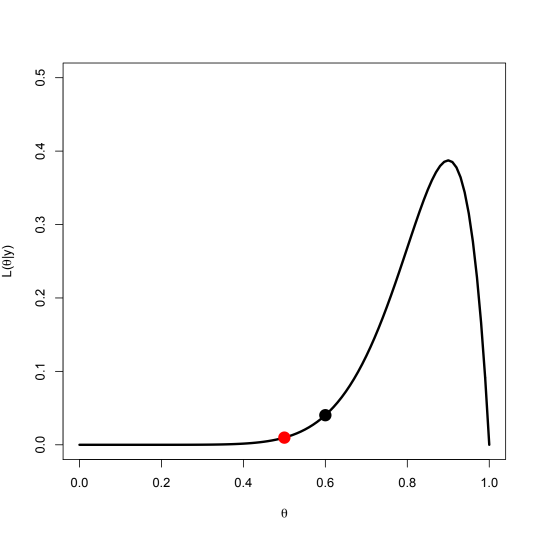
Newton-Raphson Gone Awry

What if we had taken $\theta_0 = 0.60$ to be the starting point for the Newton-Raphson maximization for the binomial likelihood?



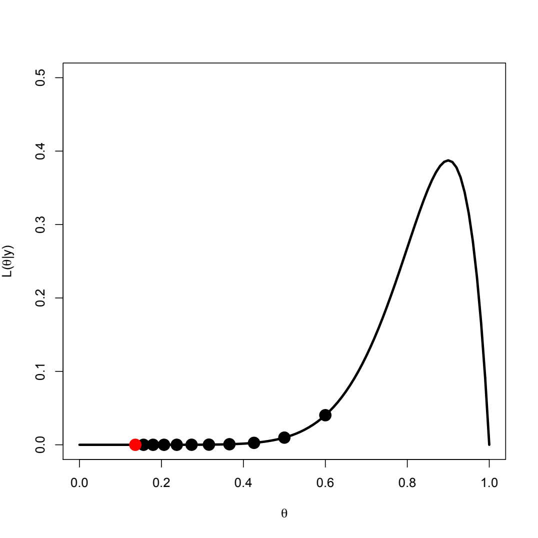
It Walks the Wrong Way...

The first iteration:



And Then Gets Stuck!

After 10 iterations:



Outline

The Multivariate Normal

Suppose I give you two observations (x=2, y=4) and tell you that they are from a bivariate normal distribution. It's known that $\sigma_x = \sigma_y = 1$ and x and y are uncorrelated. The two unknown parameters are μ_x and μ_y. The pdf is:

$$f(x, y|\mu_x, \mu_y) = \frac{1}{2\pi} e^{-\frac{1}{2}\left[(x-\mu_x)^2+(y-\mu_y)^2\right]}.$$

So what's the likelihood?

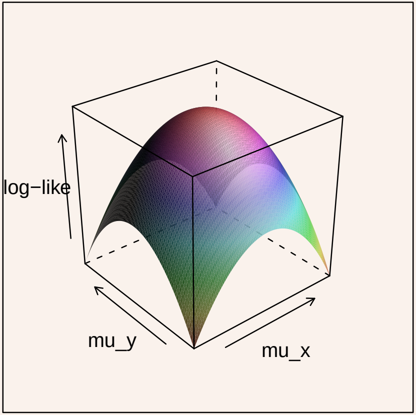
$$L(\mu_x, \mu_y|x, y) \propto \frac{1}{2\pi} e^{-\frac{1}{2}\left[(x-\mu_x)^2+(y-\mu_y)^2\right]}.$$

And the log-likelihood is especially nice:

$$\begin{aligned} \ln L(\mu_x, \mu_y|x, y) &\propto \ln \frac{1}{2\pi} - \frac{1}{2}\left[(x - \mu_x)^2 + (y - \mu_y)^2\right] \\ &\doteq -\frac{1}{2}\left[(x - \mu_x)^2 + (y - \mu_y)^2\right] \end{aligned}$$

Visualizing the Likelihood

We can plug in the datum and plot the likelihood:



Skip to numerical optimization of multivariate normal
Skip to full example of optimization

Multivariable Case: Analytically

Find extrema of:

$$\begin{aligned} \ell(\mu_x, \mu_y | x, y) &\propto -\frac{1}{2}\left[(x-\mu_x)^2 + (y-\mu_y)^2\right] \\ &= -\frac{1}{2}\left[(2-\mu_x)^2 + (4-\mu_y)^2\right] \\ &= -10 + 2\mu_x - \frac{\mu_x^2}{2} + 4\mu_y - \frac{\mu_y^2}{2}. \end{aligned}$$

Step One: Find critical value(s) using the the partial derivatives of $f(x, y)$ with respect to x and y.

- $\frac{\partial \ell}{\partial \mu_x} = 2 - \mu_x$ is 0 if $\mu_x = 2$.
- $\frac{\partial \ell}{\partial \mu_y} = 4 - \mu_y$ is 0 if $\mu_y = 4$.
- So we have a critical value at $(\mu_x = 2, \mu_y = 4)$.

Multivariable Case: Analytically

Step Two: Evaluate critical value using the second derivative, but we have a wrinkle here because we have two variables. We're going to end up with a 2x2 Hessian matrix.

$$H = \begin{pmatrix} \frac{\partial^2 l}{\partial \mu_x \partial \mu_x}(2,4) = -1 & \frac{\partial^2 l}{\partial \mu_x \partial \mu_y}(2,4) = \ \ 0 \\ \frac{\partial^2 l}{\partial \mu_y \partial \mu_x}(2,4) = \ \ 0 & \frac{\partial^2 l}{\partial \mu_y \partial \mu_y}(2,4) = -1 \end{pmatrix}$$

It turns out that the key issue here for determining whether we have a maximum settles on whether the Hessian matrix is negative or positive definite (or something else). Define $a = \binom{a_1}{a_2}$.

1. negative definite: $a'Ha < 0$ for all $a \neq 0 \rightarrow$ maximum.
2. positive definite: $a'Ha > 0$ for all $a \neq 0 \rightarrow$ minimum.

Our example is easy: $a'Ha = (-a_1 \ \ -a_2)\binom{a_1}{a_2} = -a_1^2 - a_2^2$. So we're **negative definite**.

Multivariate Case: Numerically

$$\ell(\mu_x, \mu_y | x, y) = -\frac{1}{2}\left[(2 - \mu_x)^2 + (4 - \mu_y)^2\right]$$

```
ll <- function(par){
  mux <- par[1]
  muy <- par[2]
  like <- -.5*((2-mux)^2+(4-muy)^2)
  return(like)
}

optim(par = c(0,0), fn = ll, method = "BFGS",
      control = list(fnscale = -1), hessian = TRUE)
```

```
$par
[1] 2 4

$value
[1] -4.930381e-31
...
$convergence
[1] 0
...
$hessian
              [,1]          [,2]
[1,] -1.000000e+00 -2.646978e-17
[2,] -2.646978e-17 -1.000000e+00
```

▸ Skip to full example of optimization

optim() Techniques in Practice

You can always test these techniques on some function.

```
fkt <- function(x){10*sin(0.3*x)*sin(1.3*x^2)
                   + 0.00001*x^4 + 0.2*x+80}
```

Figure : Plotting the Wild Function

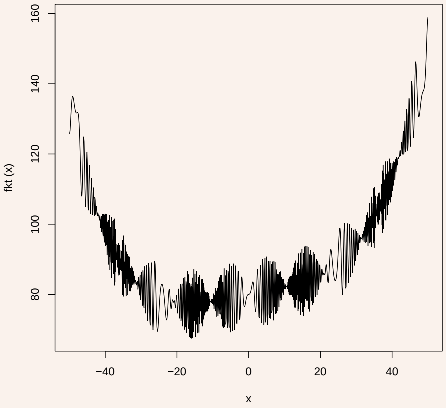
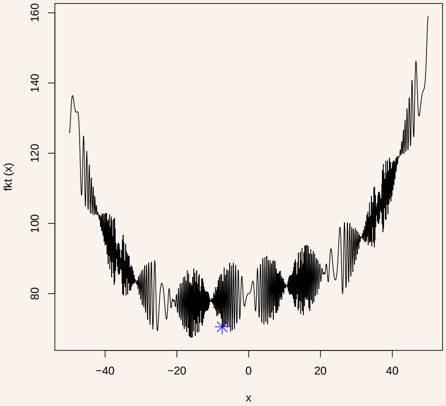
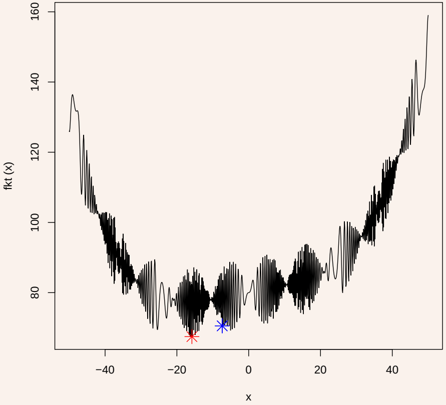


Figure : Using BFGS at x=50




```
out <- optim(par = 50, fkt, method="BFGS",
             control=list(maxit=20000, parscale=20))
points(out$par, out$val, pch = 8, col="blue", cex=2)
```

Figure : Using Simulated Annealing at x=50




```
out <- optim(par = 50, fkt, method="SANN",
             control=list(maxit=20000, temp=20, parscale=20))
points(out$par, out$val, pch = 8, col = "red", cex = 2)
```

Outline

Putting It All Together

Suppose we have some count data (number of coups in a year).

We can use a Poisson distribution to model the data (we will learn more about Poisson later).

$$Y_i \sim_{iid} \text{Poisson}(\lambda)$$

We want to find λ, which is the mean of the Poisson distribution.

The PMF (discrete) for the data is

$$p(\boldsymbol{y}|\lambda) = \prod_{i=1}^{n} \frac{\lambda^{y_i} e^{-\lambda}}{y_i!}$$

Since $L(\theta|\boldsymbol{y}) \propto p(\boldsymbol{y}|\theta)$, we have

$$L(\lambda|\boldsymbol{y}) \quad \propto \quad \prod_{i=1}^{n} \frac{\lambda^{y_i} e^{-\lambda}}{y_i!}$$

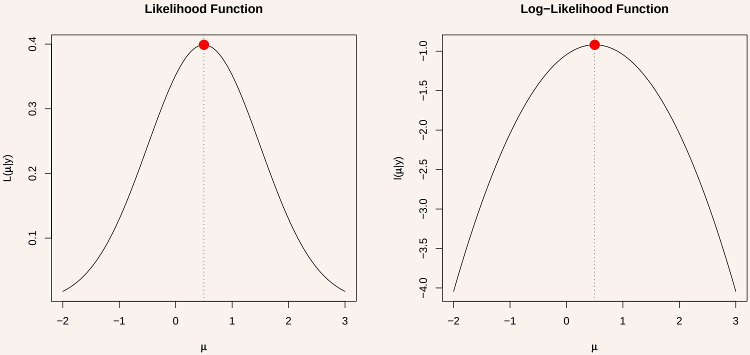
To make the math easier, we will take the log-likelihood.

$$\ell(\lambda|\boldsymbol{y}) \quad = \quad \sum_{i=1}^{n} (y_i \ln \lambda - \lambda - \ln y_i!)$$

We can drop all terms that don't depend on λ (because likelihood is a relative concept and is invariant to shifts).

$$\ell(\lambda|\boldsymbol{y}) \quad = \quad \sum_{i=1}^{n} (y_i \ln \lambda) - n\lambda$$

Why Can We Use the Log-likelihood?



Skip to finding MLE in R

Finding the Maximum Likelihood Estimate (MLE)

Remember that to find our MLE, we want to find the value of the parameter(s) that maximizes our log-likelihood.

$$\ell(\lambda|\boldsymbol{y}) = \sum_{i=1}^{n}(y_i \ln \lambda) - n\lambda$$

We need to set the derivative (known as the score function) to zero and solve for λ.

$$\begin{aligned}\frac{\partial \ell(\lambda|\boldsymbol{y})}{\partial \lambda} = S(\lambda) &= \frac{\sum_{i=1}^{n} y_i}{\lambda} - n \\ 0 &= \frac{\sum_{i=1}^{n} y_i}{\lambda} - n \\ \hat{\lambda} &= \frac{\sum_{i=1}^{n} y_i}{n}\end{aligned}$$

Maximum Likelihood In R

Write our log-likelihood function:

$$\ell(\lambda|\boldsymbol{y}) = \sum_{i=1}^{n}(y_i \ln\lambda) - n\lambda$$

```
> ll.poisson <- function(par, y) {
+     lambda <- exp(par)
+     out <- sum(y * log(lambda)) - length(y) * lambda
+     return(out)
+ }
```

Maximum Likelihood In R

Find the maximum (with sample data):

```
> data <- rpois(1000, 5)
> opt <- optim(par = 2, fn = ll.poisson, method = "
    BFGS", control = list(fnscale = -1),
+       y = data)$par
> mle <- exp(opt)
> mle
```

```
[1] 4.996001
```