Likelihood / Optimization

Objectives

- likelihood principle
- likelihood connection to probability function
- optimization / parameter estimation

The others side of the coin

Statistics: Interested in estimating population-level characteristics; i.e., the parameters

$$y \rightarrow f(y|\theta)$$

Estimation

- likelihood
- Bayesian

Likelihood

Likelihood principle

All the evidence/information in a sample ($\mathbf{y}$, i.e., data) relevant to making inference on model parameters (θ) is contained in the *likelihood function*.

The pieces

- The sample data, $\mathbf{y}$
- A probability function for $\mathbf{y}$:
  - $f(\mathbf{y}|\theta)$ or $f(\mathbf{y};\theta)$ or $[\mathbf{y}|\theta]$ or $P(\mathbf{y}|\theta)$
  - the unknown parameter(s) (θ) of the probability function

The Likelihood Function

$$\mathcal{L}(\theta|y) = f(y|\theta)$$

. . .

The likelihood ($\mathcal{L}$) of the unknown parameters, given our data, can be calculated using our probability function.

The Likelihood Function

For example, for $y_1 \sim \text{Normal}(\mu, \sigma)$

CODE:

```
# A data point
  y = c(10)

# The likelihood the mean is 8, given our data
  dnorm(y, mean = 8)
```

```
[1] 0.05399097
```

. . .

If we knew the mean is truly 8, it would also be the probability density of the observation y = 10. But, we don't know what the mean truly is.

. . .

The **key** is to understand that the likelihood values are relative, which means we need many guesses.

. . .

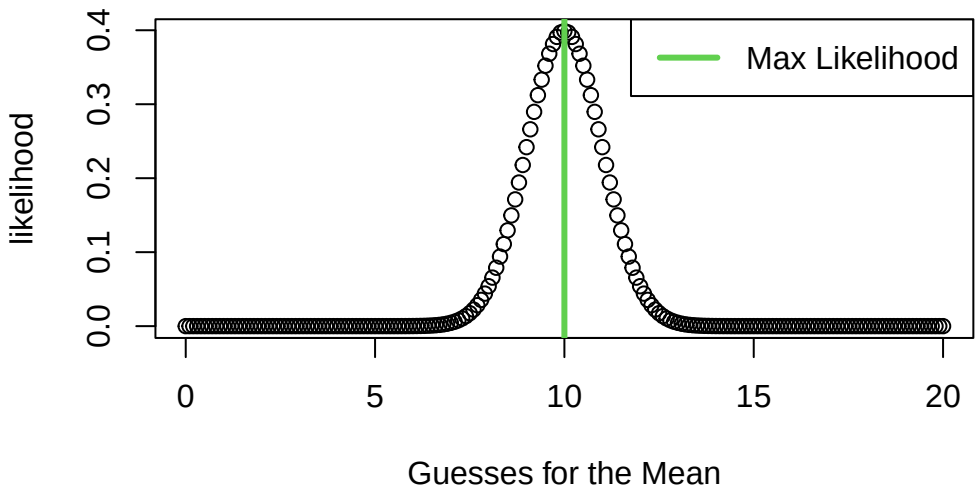
```r
# The likelihood the mean is 9, given our data
  dnorm(y, mean = 9)
```

```
[1] 0.2419707
```

Optimization

Grid Search

```r
  means = seq(0, 20, by = 0.1) # many guesses of the mean
  likelihood = dnorm(y, mean = means, sd = 1) # likelihood of each guess of the mean
```



Maximum Likelihood Properties

- Central Tenet: evidence is relative.
- Parameters are not RVs. They are not defined by a PDF/PMF.
- MLEs are consistent. As sample size increases, they will converge towards (get closer) the true parameter value.

- MLEs are asymptotically unbiased. The $E[\hat{\theta}]$ converges to θ as the sample size gets larger.
- No guarantee that MLE is unbiased at small sample size. Can be tested!
- MLEs will have the minimum variance among all estimators, as the sample size gets larger.

MLE with n $>$ 1

What is the mean height of King Penguins?

MLE with n $>$ 1

We go and collect data,

$y = [4.34 \quad 3.53 \quad 3.75]$

. . .

Let's decide to use the Normal Distribution as our PDF.

$$f(y_1 = 4.34|\mu, \sigma) = \frac{1}{\sigma\sqrt{(2\pi)}} e^{-\frac{1}{2}(\frac{y_1-\mu}{\sigma})^2}$$

AND

$$f(y_2 = 3.53|\mu, \sigma) = \frac{1}{\sigma\sqrt{(2\pi)}} e^{-\frac{1}{2}(\frac{y_2-\mu}{\sigma})^2}$$

AND

$$f(y_3 = 3.75|\mu, \sigma) = \frac{1}{\sigma\sqrt{(2\pi)}} e^{-\frac{1}{2}(\frac{y_3-\mu}{\sigma})^2}$$

Need to connect data together

Or simply,

$$\mathbf{y} \overset{iid}{\sim} \text{Normal}(\mu, \sigma)$$

iid = independent and identically distributed

Need to connect data together

The joint probability of our data with shared parameters μ and σ,

$$P(Y_1 = y_1, Y_2 = y_2, Y_3 = y_3|\mu, \sigma)$$

$$P(Y_1 = 4.34, Y_2 = 3.53, Y_3 = 3.75|\mu, \sigma)$$
$$= \mathcal{L}(\mu, \sigma|\mathbf{y})$$

Need to connect data together

IF each y_i is **independent**, the *joint probability* of our data are simply the multiplication of all three probability densities,

$$= f(4.34|\mu, \sigma) \times f(3.53|\mu, \sigma) \times f(3.75|\mu, \sigma)$$

$$= \prod_{i=1}^{3} f(y_i|\mu, \sigma)$$
$$= \mathcal{L}(\mu, \sigma|y_1, y_2, y_3)$$

Conditional Independence Assumption

We can do this because we are assuming knowing one observation does not tell us any new information about another observation. Such that,

$P(y_2|y_1) = P(y_2)$

Code

Translate the math to code…

```
# penguin height data
  y = c(4.34, 3.53, 3.75)

# Joint likelihood of mu=3, sigma =1, given our data
  prod(dnorm(y, mean = 3,sd = 1))
```

```
[1] 0.01696987
```

Optimization Code (Grid Search)

Calculate likelihood of many guesses of μ and σ simultaneously,

```r
# The Guesses
  mu = seq(0,6,0.05)
  sigma = seq(0.01,2,0.05)
  try = expand.grid(mu,sigma)
  colnames(try) = c("mu","sigma")

# function
fun = function(a,b){
          prod(dnorm(y,mean = a, sd = b))
      }

# mapply the function with the inputs
  likelihood = mapply(a = try$mu, b = try$sigma, FUN=fun)
```

MLE Code

```r
# maximum likelihood of parameters
  try[which.max(likelihood),]
```

```
      mu sigma
925 3.85  0.36
```

. . .

Lets compare to,

```r
sum(y)/length(y)
```

```
[1] 3.873333
```

Likelihood plot (3D)

Sample Size

What happens to the likelihood if we increase the sample size to N=100?

. . .

Relativeness

log-likelihood

```
fun.log = function(a,b){
              sum(dnorm(y,mean = a, sd = b, log=TRUE))
          }

log.likelihood = mapply(a = try$mu, b = try$sigma, FUN=fun.log)

# maximum log-likelihood of parameters
  try[which.max(log.likelihood),]
```

```
       mu sigma
55683 3.9  0.93
```

Optimization Code (Numerical)

Let's let the computer do some smarter guessing, i.e., optimization.

```
# Note: optim function uses minimization, not maximization.
# WE want to find the minimum negative log-likelihood
# THUS, need to put negative in our function

neg.log.likelihood=function(par){
  -sum(dnorm(y,mean=par[1],sd=par[2],log=TRUE))
  }

#find the values that minimizes the function
#c(1,1) are the initial values for mu and sigma
fit <- optim(par=c(1,1), fn=neg.log.likelihood,
             method="L-BFGS-B",
             lower=c(0,0),upper=c(10,1)
             )

#Maximum likelihood estimates for mu and sigma
fit$par
```

```
[1] 3.901219 0.927352
```

Linear Regression

King Penguin Height Data (N=100)

```
out = lm(y~1)
summary(out)
```

```
Call:
lm(formula = y ~ 1)

Residuals:
     Min       1Q   Median       3Q      Max
-3.00136 -0.61581  0.01208  0.67407  2.58369

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3.9012     0.0932   41.86   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.932 on 99 degrees of freedom
```