

Statistical Inference

Lecture 05b

ANU - RSFAS

Last Updated: Tue Mar 20 17:37:53 2018

Point Estimation

- What are some general approaches to determine a good guess?
 - method of moments
 - maximum likelihood
 - Bayesian estimation

Method of Moments

$$X_1, \dots, X_n \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(\mu, \sigma^2)$$

- One of the oldest approaches. We equate the moments of a distribution to the sample moments.

Consider the following distributional moments:

$$\begin{aligned} E(X) &= \mu \\ E(X^2) &= \sigma^2 + \mu^2 \\ \left(\text{Var} = E(X^2) - (E(X))^2 \right) \end{aligned}$$

$$\mu_k = E_\theta(X^k)$$

And sample moments:

$$\hat{\mu}_1 = \bar{X}$$

$$\hat{\mu}_k = \frac{1}{n} \sum_{i=1}^n x_i^k$$

$$\hat{\mu}_2 = \frac{1}{n} \sum_{i=1}^n x_i^2$$

for $k = 1, \dots, K$.

Method of Moments

- Typically the population moments are implicitly defined by parameters $\theta = (\theta_1, \dots, \theta_K)$.
- Equate the sample and population moments:

$$\begin{aligned}\mu_1(\theta_1, \dots, \theta_K) &= \hat{\mu}_1(x_1, \dots, x_n) \\ &\vdots \\ \mu_K(\theta_1, \dots, \theta_K) &= \hat{\mu}_K(x_1, \dots, x_n)\end{aligned}$$

- The estimator $T(\mathbf{X}) = \tilde{\theta}$ is the value for θ which solves the system of K equations.

Method of Moments

Eg. Poisson distribution

$$X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{Poisson}(\lambda)$$

- Set $E[X] = \bar{X}$.

$$\tilde{\lambda} = \bar{X}$$

Method of Moments

Eg. Normal distribution

Tutorial

$$X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{normal}(\mu, \sigma)$$

$$E(X) = \mu; \quad \text{Var}(X) = \sigma^2$$

$$\begin{aligned} E(X^2) &= V(X) + (E(X))^2 \\ &= \sigma^2 + \mu^2 \end{aligned}$$

$$\begin{aligned} \mu &= \bar{X} \\ \sigma^2 + \mu^2 &= \frac{1}{n} \sum x_i^2 \\ \text{solve for } \mu, \sigma^2 \end{aligned}$$

$$\begin{aligned} \hat{\mu} &= \bar{X} \\ \hat{\sigma}^2 &= \frac{1}{n} \sum (x_i - \bar{x})^2 \end{aligned}$$

Other Similar Approaches - Many!

We can replace the raw and sample moments with the so-called **central moments**:

$$\mu'_k = E_{\theta}(\{X - E_{\theta}(X)\}^k)$$

And sample moments:

$$\hat{\mu}'_k = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^k$$

for $k = 2, \dots, K$. Set $\mu_1 = \hat{\mu}_1 = \bar{x}$.

Do we actually need to pick the first k moments? How about any k moments?

Other Similar Approaches - Many!

- Can we generalize the functions? Yes. This leads to **generalized method of moments** based on some functions $\underline{g_1(), \dots, g_k()}$:

$$\begin{aligned} E_{\theta}(g_1(X)) &= \frac{1}{n} \sum_{i=1}^n g_1(x_i) \\ &\vdots \\ E_{\theta}(g_k(X)) &= \frac{1}{n} \sum_{i=1}^n g_k(x_i) \end{aligned}$$

Note: If we set $g_i(x) = x^i$ then we recover the **standard method of moments**.

Maximum Likelihood Estimation

This is the best known, most widely used, most intuitive, most important, ... of estimation procedures.

Definition 3.1: We find the estimator $\hat{\theta}$ which maximizes the likelihood function ~~$L(\theta|x)$~~ .

$$L(\theta; x)$$

$$L(\theta; x) = L(\theta; x_1, \dots, x_n) = f(x_1, \dots, x_n; \theta)$$

Before we move forward let's consider an example . . .

Maximum Likelihood Estimation

- Suppose that a particular population contains individuals of two types, A and B .
- Suppose that we are told that there are three times more of one type of individual than the other.
- We don't which is more prevalent A s or B s.
- We would like to know which is more prevalent. (our scientific question!)
- To try and answer this question we will sample 3 individuals ($n = 3$).
- Let X denote the number of A individuals in the sample (binomial distribution - what assumptions were made?).

$$P(X = x) = L(p; x) = \frac{3!}{x!(3-x)!} p^x (1-p)^{n-x}$$

Maximum Likelihood Estimation

| | Outcome of X | | | |
|---------------------------------------|-----------------|-----------------|-----------------|-----------------|
| Θ | 0 | 1 | 2 | 3 |
| <u>$p_1 = \frac{3}{4}$</u> | $\frac{1}{64}$ | $\frac{9}{64}$ | $\frac{27}{64}$ | $\frac{27}{64}$ |
| <u>$p_2 = \frac{1}{4}$</u> | $\frac{27}{64}$ | $\frac{27}{64}$ | $\frac{9}{64}$ | $\frac{1}{64}$ |

of As

- Based on this table of probabilities, we can now devise a reasonable estimator for the true population value of p , based on the notion of the “preponderance of evidence” or the likelihood.

$$\hat{p} = \operatorname{argmax}_{p \in \{1/4, 3/4\}} P(X = x) \begin{cases} 1/4 & \text{if } x = 0, 1 \\ 3/4 & \text{if } x = 2, 3 \end{cases}$$

Maximum Likelihood Estimation

Eg. continued: Suppose that we don't have a restricted parameter space $p \in \{1/4, 3/4\}$ but the full natural space $[0, 1]$:

$$P(X = x) = L(p; x) = \frac{3!}{x!(3-x)!} p^x (1-p)^{n-x}$$

$$\left. \frac{d}{dp} L(p; x) \right|_{p=\hat{p}} = \frac{3!}{x!(3-x)!} \left[x \hat{p}^{x-1} (1-\hat{p})^{3-x} - (3-x) \hat{p}^x (1-\hat{p})^{2-x} \right] = 0$$

$$\hat{p} = \frac{x}{n} \quad n=3$$

Maximum Likelihood Estimation

Definition 3.1: The MLE is the value $\hat{\theta}$ which maximizes $L(\theta; \mathbf{x})$.

- If the likelihood is differentiable in (θ_i) , **possible candidates** for the MLE are the values $(\theta_1, \dots, \theta_k)$ that solve

$$\frac{\partial}{\partial \theta_i} L(\theta; \mathbf{x}) = 0, \quad i = 1, \dots, k$$

- Possible: local vs. global maximum, extrema may occur on the boundary and thus the first derivative may not be 0, ...
- All the good points and bad points of optimizing a function are here!

Maximum Likelihood Estimation

Eg. Poisson: $X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{Poisson}(\lambda)$.

$$\begin{aligned}L(\lambda; \mathbf{x}) &= \prod_{i=1}^n \frac{e^{-\lambda} \lambda^{x_i}}{x_i!} \\&= \frac{e^{-n\lambda} \lambda^{\sum_{i=1}^n x_i}}{\prod_{i=1}^n x_i!} \\ \ell(\lambda; \mathbf{x}) &= -n\lambda + \sum_{i=1}^n x_i \log(\lambda) - \sum_{i=1}^n \log(x_i!) \\ \ell'(\lambda) &= -n + \frac{\sum_{i=1}^n x_i}{\lambda} = 0\end{aligned}$$

$$\hat{\lambda} = \bar{x} \text{ estimate}$$

$$\hat{\lambda} = \bar{X} \text{ estimator}$$

Maximum Likelihood Estimation

- Do we have a maximum? Yes.

$$\ell''(\lambda) = -\frac{\sum_{i=1}^n x_i}{\lambda^2} < 0$$

Maximum Likelihood Estimation

Eg. (Taken from Prof. Richard Lockhart): Consider $X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{Cauchy}(\theta)$.

$$f(x; \theta) = \frac{1}{\pi(1 + (x - \theta)^2)}$$

- The likelihood function is

$$L(\theta; \mathbf{x}) = L(\theta) = \prod_{i=1}^n \frac{1}{\pi(1 + (x_i - \theta)^2)}$$

- Here are some likelihood plots.

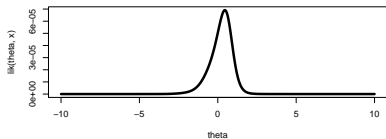

```
set.seed(2001)
n <- 5

lik <- function(theta,x){
  K <- length(theta)
  n <- length(x)
  out <- rep(0, K)
  for(k in 1:K){
    out[k] <- prod(dcauchy(x, theta[k], 1))
  }
  return(out)
}

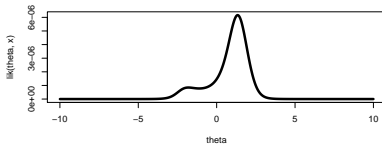
theta <- seq(-10, 10, by=0.01)
par(mfrow=c(3,2))
for(i in 1:6){
  x <- rcauchy(n, location=0, scale=1)
  plot(theta, lik(theta, x), type="l", lwd=3,
        main="Likelihood Function: Cauchy, n=5")
}
```

Likelihood $n = 5$

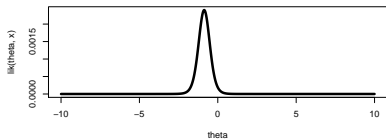
Likelihood Function: Cauchy, n=5



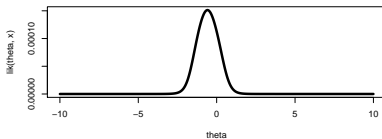
Likelihood Function: Cauchy, n=5



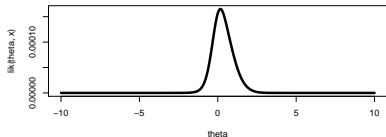
Likelihood Function: Cauchy, n=5



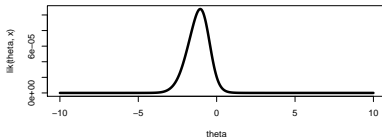
Likelihood Function: Cauchy, n=5



Likelihood Function: Cauchy, n=5



Likelihood Function: Cauchy, n=5

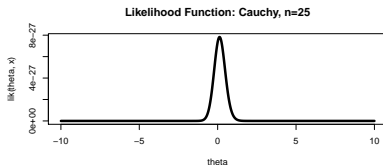
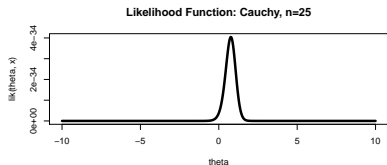
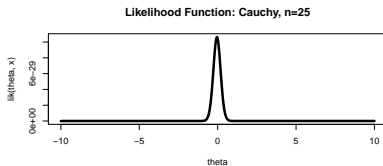
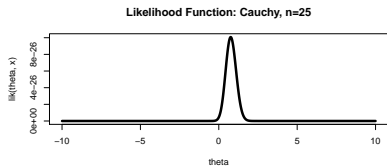
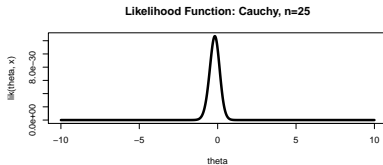
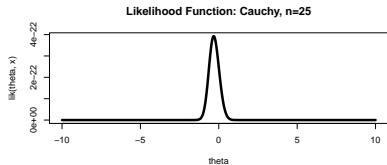


```
set.seed(2001)
n <- 25

lik <- function(theta,x){
  K <- length(theta)
  n <- length(x)
  out <- rep(0, K)
  for(k in 1:K){
    out[k] <- prod(dcauchy(x, theta[k], 1))
  }
  return(out)
}

theta <- seq(-10, 10, by=0.01)
par(mfrow=c(3,2))
for(i in 1:6){
  x <- rcauchy(n, location=0, scale=1)
  plot(theta, lik(theta, x), type="l", lwd=3,
        main="Likelihood Function: Cauchy, n=25")
}
```

Likelihood $n = 25$



Things to See in the Plots

- The likelihood functions have peaks near the true value of θ (which is 0 for the data sets I generated).
- The peaks are narrower for the larger sample size.
- The peaks have a more regular shape for the larger value of n .

collecting more data
 \Rightarrow more precise

Maximim Likelihood Estimation

- To maximize this likelihood: differentiate $L(\theta)$, set result equal to 0.
- Notice $L(\theta)$ is product of n terms

$$\begin{aligned} L(\theta) &= \prod_{i=1}^n \frac{1}{\pi(1 + (x_i - \theta)^2)} \\ &= \left(\frac{1}{\pi(1 + (x_1 - \theta)^2)} \right) \times \cdots \times \left(\frac{1}{\pi(1 + (x_n - \theta)^2)} \right) \end{aligned}$$

Maximim Likelihood Estimation

- The derivative is

$$\sum_{i=1}^n \prod_{j \neq i} \frac{1}{\pi(1 + (x_j - \theta)^2)} \frac{2(x_i - \theta)}{\pi(1 + (x_i - \theta)^2)^2}$$

Not fun!!

- Much easier to work with **logarithm of $L(\theta)$** : log of product is sum and logarithm is monotone increasing.
- The **log likelihood function** is

$$\ell(\theta) = \log[L(\theta)]$$

Maximim Likelihood Estimation

- For the Cauchy problem we have

$$L(\theta) = \prod_{i=1}^n \frac{1}{\pi(1 + (x_i - \theta)^2)}$$

$$\ell(\theta) = - \sum_{i=1}^n \log(1 + (x_i - \theta)^2) - n \log(\pi)$$

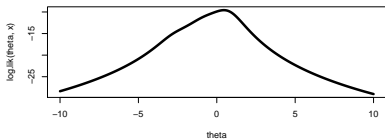

```
set.seed(2001)
n <- 5

log.lik <- function(theta,x){
  K <- length(theta)
  n <- length(x)
  out <- rep(0, K)
  for(k in 1:K){
    out[k] <- sum(dcauchy(x, theta[k], 1, log=TRUE))
  }
  return(out)
}

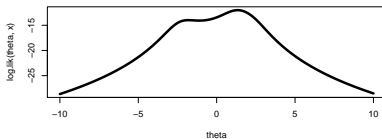
theta <- seq(-10, 10, by=0.01)
par(mfrow=c(3,2))
for(i in 1:6){
  x <- rcauchy(n, location=0, scale=1)
  plot(theta, log.lik(theta, x), type="l", lwd=3,
        main="Log Likelihood Function: Cauchy, n=5")
}
```

Cauchy log-likelihood $n = 5$

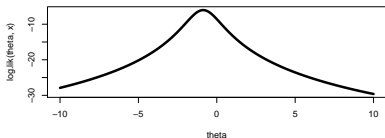
Log Likelihood Function: Cauchy, n=5



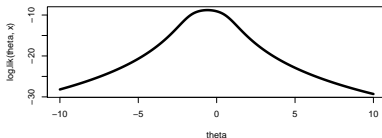
Log Likelihood Function: Cauchy, n=5



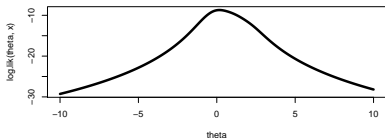
Log Likelihood Function: Cauchy, n=5



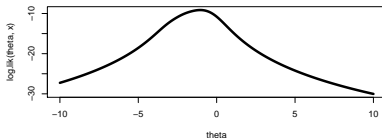
Log Likelihood Function: Cauchy, n=5



Log Likelihood Function: Cauchy, n=5



Log Likelihood Function: Cauchy, n=5



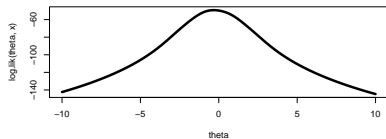
```
set.seed(2001)
n <- 25

log.lik <- function(theta,x){
  K <- length(theta)
  n <- length(x)
  out <- rep(0, K)
  for(k in 1:K){
    out[k] <- sum(dcauchy(x, theta[k], 1, log=TRUE))
  }
  return(out)
}

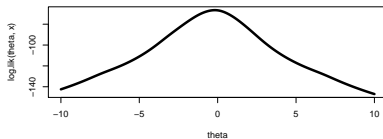
theta <- seq(-10, 10, by=0.01)
par(mfrow=c(3,2))
for(i in 1:6){
  x <- rcauchy(n, location=0, scale=1)
  plot(theta, log.lik(theta, x), type="l", lwd=3,
        main="Log Likelihood Function: Cauchy, n=25")
}
```

Cauchy log-likelihood $n = 25$

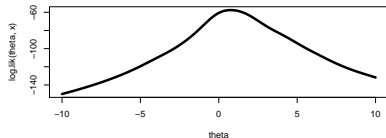
Log Likelihood Function: Cauchy, n=25



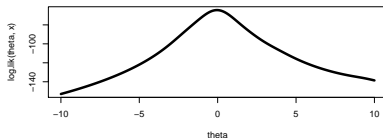
Log Likelihood Function: Cauchy, n=25



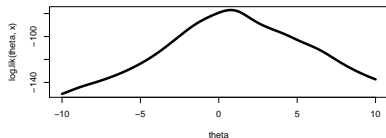
Log Likelihood Function: Cauchy, n=25



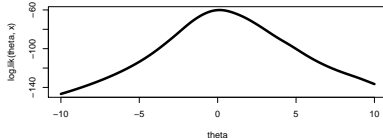
Log Likelihood Function: Cauchy, n=25



Log Likelihood Function: Cauchy, n=25



Log Likelihood Function: Cauchy, n=25



Things to Notice

- Plots of $\ell(\theta)$ for $n = 25$ quite smooth, rather parabolic.
- For $n = 5$ many local maxima and minima of ℓ .
- Likelihood tends to 0 as $|\theta| \rightarrow \infty$, so max of $\ell(\theta)$ occurs at a root of $\ell'(\theta)$.
- **Score Function** is the gradient of $\ell(\theta)$

$$U(\theta) = \frac{\partial \ell}{\partial \theta} = \ell'(\theta)$$

Maximum Likelihood Estimation

- As stated the MLE is usually a root of $U(\theta)$

$$U(\theta) = 0$$

- In our Cauchy example we find

$$U(\theta) = \sum_{i=1}^n \frac{2(x_i - \theta)}{1 + (x_i - \theta)^2} = 0$$

- Now let's examine plots of score function.
- Notice: often multiple roots.

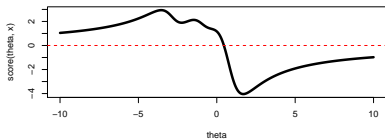
```
set.seed(2001)
n <- 5

score <- function(theta,x){
  K <- length(theta)
  n <- length(x)
  out <- rep(0, K)
  for(k in 1:K){
    out[k] <- sum( ( 2*(x - theta[k]))/( 1 + (x - theta[k])^2))
  }
  return(out)
}

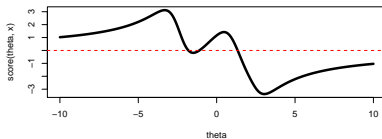
theta <- seq(-10, 10, by=0.01)
par(mfrow=c(3,2))
for(i in 1:6){
  x <- rcauchy(n, location=0, scale=1)
  plot(theta, score(theta, x), type="l", lwd=3,
        main="Score Function: Cauchy, n=5")
}
```

Cauchy Score Function $n = 5$

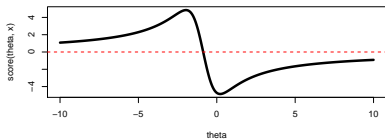
Score Function: Cauchy, $n=5$



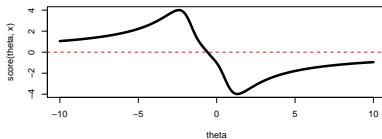
Score Function: Cauchy, $n=5$



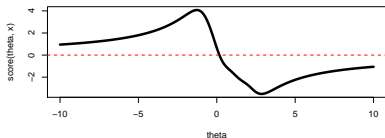
Score Function: Cauchy, $n=5$



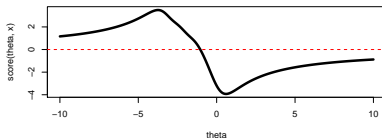
Score Function: Cauchy, $n=5$



Score Function: Cauchy, $n=5$



Score Function: Cauchy, $n=5$



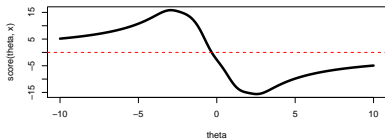

```
set.seed(2001)
n <- 25

score <- function(theta,x){
  K <- length(theta)
  n <- length(x)
  out <- rep(0, K)
  for(k in 1:K){
    out[k] <- sum( ( 2*(x - theta[k]))/( 1 + (x - theta[k])^2))
  }
  return(out)
}

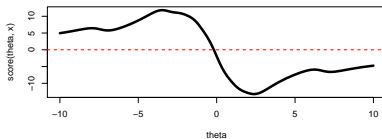
theta <- seq(-10, 10, by=0.01)
par(mfrow=c(3,2))
for(i in 1:6){
  x <- rcauchy(n, location=0, scale=1)
  plot(theta, score(theta, x), type="l", lwd=3,
        main="Score Function: Cauchy, n=25")
}
```

Cauchy Score Function $n = 25$

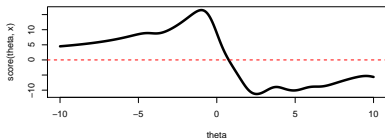
Score Function: Cauchy, $n=25$



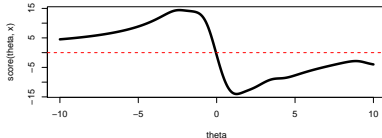
Score Function: Cauchy, $n=25$



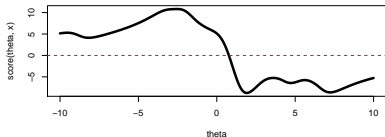
Score Function: Cauchy, $n=25$



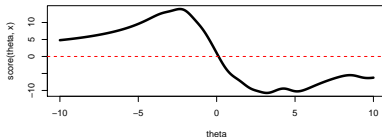
Score Function: Cauchy, $n=25$



Score Function: Cauchy, $n=25$



Score Function: Cauchy, $n=25$



Maximum Likelihood Estimation

Eg. Exponential distribution with ^{right}censoring. Consider:

$$X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{exponential}(\theta)$$

$$f_X(x; \theta) = \theta \exp(-\theta x)$$

$$X > 0, \quad \theta > 0$$

$$E(X) = 1/\theta; \quad \text{Var}(X) = 1/\theta^2$$

$$f(x; \theta) = \frac{1}{\theta} \exp\left(-\frac{x}{\theta}\right)$$

$$\downarrow \\ E(X) = \theta$$

- The experiment is run until time T .
- The values X_1, \dots, X_m are observed.
- The values X_{m+1}, \dots, X_n were not observed by time T . All we know is that they exceed time T (right censored).

Maximum Likelihood Estimation

- Based on this information we can derive the likelihood:

$$L(\theta) = \prod_{i=1}^m f_X(x_i; \theta) \prod_{i=(m+1)}^n (1 - F_X(T; \theta))$$

(1 - P(X < T)) = P(X > T)

$$F_X(T) = P(X \leq T) = \int_0^T \theta \exp(-\theta x) dx = 1 - \exp(-\theta T)$$

$$\begin{aligned} L(\theta) &= \prod_{i=1}^m \theta \exp(-\theta x_i) \prod_{i=(m+1)}^n \exp(-\theta T) \\ &= \theta^m \exp\left(-\theta \sum_{i=1}^m x_i\right) \exp(-\theta T)^{n-m} \end{aligned}$$

Maximum Likelihood Estimation

- So the log likelihood is:

$$l(\theta) = m \log(\theta) - \theta \sum_{i=1}^m x_i - (n - m)T\theta$$

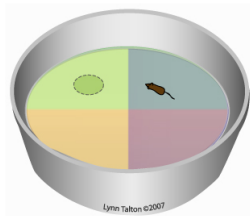
- From here we can get the score equation and solve for θ :

$$\underline{l'(\theta)} = \frac{m}{\theta} - \sum_{i=1}^m x_i - (n - m)T = \underline{0}$$

$$\hat{\theta} = \frac{m}{(n - m)T + \sum_{i=1}^m x_i}$$

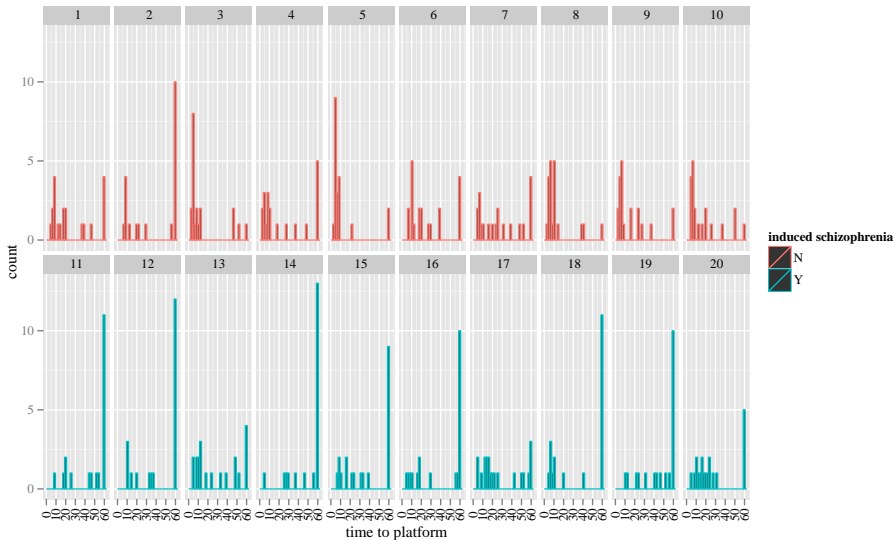
- Check the second derivative.

A Real Example - Chemically Induced Schizophrenia (Ketamine) in Rats



- Question: Do rats with "schizophrenia" find a hidden platform slower than those without?
- Each rat attempted to find the platform on 20 occasions over 5 days (4 times per day).

censored data: take out the rats if they
cannot find it in 60s.



Maximum Likelihood Estimation

Example: $X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{normal}(\mu, 1)$, where it is known that $\mu \geq 0$ must not be negative.

- Writing that in terms of an indicator function:

$$X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{normal}(\mu, 1) \mathbb{I}_{[0, \infty)}(\mu)$$

$$\begin{aligned} L(\mu; \mathbf{x}) &= \prod_{i=1}^n (2\pi)^{-1/2} \exp\left(-\frac{1}{2}(x_i - \mu)^2\right) \\ &= (2\pi)^{-n/2} \exp\left(-\frac{1}{2} \sum_{i=1}^n (x_i - \mu)^2\right) \\ \ell(\mu|\mathbf{x}) &= -\frac{n}{2} \log(2\pi) - \frac{1}{2} \sum_{i=1}^n (x_i - \mu)^2 \end{aligned}$$

Maximum Likelihood Estimation

$$\ell'(\mu; \mathbf{x}) = \sum_{i=1}^n (x_i - \mu) = 0$$

$$\hat{\mu} = \bar{x} \text{ if } \bar{x} \geq 0 \quad \text{and} \quad \hat{\mu} = 0 \text{ if } \bar{x} < 0$$

$$\ell''(\mu; \mathbf{x}) = -n < 0$$

Maximum Likelihood Estimation

Example: $X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{normal}(\mu, \sigma^2)$. No range restriction, but two parameters.

$$\begin{aligned} L(\mu, \sigma^2 | \mathbf{x}) &= \prod_{i=1}^n (2\pi\sigma^2)^{-1/2} \exp\left(-\frac{1}{2\sigma^2}(x_i - \mu)^2\right) \\ &= (2\pi\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2\right) \\ \ell(\mu, \sigma^2 | \mathbf{x}) &= -\frac{n}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2 \end{aligned}$$

Maximum Likelihood Estimation

- Taking the first partial derivatives $\theta = \{\mu, \sigma^2\}$, we get the vector of the score equations:

$$U = \begin{pmatrix} \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu) \\ \frac{1}{2\sigma^4} \sum_{i=1}^n (x_i - \mu)^2 - \frac{n}{2\sigma^2} \end{pmatrix}$$

- Setting $U = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ and solving for the parameters we get:

$$\hat{\mu} = \bar{x}, \quad \hat{\sigma}^2 = \sum_{i=1}^n (x_i - \bar{x})^2 / n$$

*biased in
small samples*

- Do we have a maximum?

Maximum Likelihood Estimation

- General approach for checking. See pp. Casella and Berger 322-323. For a function $H(\theta_1, \theta_2)$ to have a local maximum at $\hat{\theta}_1, \hat{\theta}_2$ the following three conditions must hold:
 1. First-order partial derivatives (score equations) at $\hat{\theta}_1, \hat{\theta}_2$ are zero:

$$\left. \frac{\partial}{\partial \theta_1} H(\theta_1, \theta_2) \right|_{\theta_1=\hat{\theta}_1, \theta_2=\hat{\theta}_2} = 0$$
$$\left. \frac{\partial}{\partial \theta_2} H(\theta_1, \theta_2) \right|_{\theta_1=\hat{\theta}_1, \theta_2=\hat{\theta}_2} = 0$$

2. At least one second-order partial derivative is negative:

$$\left. \frac{\partial}{\partial \theta_1^2} H(\theta_1, \theta_2) \right|_{\theta_1 = \hat{\theta}_1, \theta_2 = \hat{\theta}_2} < 0$$
$$\left. \frac{\partial}{\partial \theta_2^2} H(\theta_1, \theta_2) \right|_{\theta_1 = \hat{\theta}_1, \theta_2 = \hat{\theta}_2} < 0$$

$$\frac{\partial}{\partial \mu^2} \ell(\mu, \sigma^2) = \frac{-n}{\sigma^2} \Big|_{\sigma^2 = \hat{\sigma}^2} < 0$$

$$\frac{\partial}{\partial (\sigma^2)^2} \ell(\mu, \sigma^2) = \frac{n}{2\sigma^4} - \frac{\sum_{i=1}^n (x_i - \mu)^2}{\sigma^6} \Big|_{\mu = \hat{\mu}, \sigma^2 = \hat{\sigma}^2} \stackrel{?}{<} 0$$

$$\Rightarrow \frac{n}{2\hat{\sigma}^2{}^2} \stackrel{?}{<} \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{\hat{\sigma}^2{}^3}$$

$$\Rightarrow \frac{n}{2\hat{\sigma}^2{}^2} \stackrel{?}{<} \frac{n\hat{\sigma}^2}{\hat{\sigma}^2{}^3}$$

$$\Rightarrow \frac{n}{2\hat{\sigma}^2{}^2} \stackrel{?}{<} \frac{n}{\hat{\sigma}^2{}^2}$$

$$\Rightarrow \frac{1}{2} < 1$$

3. The determinant of the matrix of second order partial derivatives (the Hessian matrix) is positive.

$$\begin{vmatrix} \frac{\partial}{\partial \theta_1^2} & \frac{\partial}{\partial \theta_1 \partial \theta_2} \\ \frac{\partial}{\partial \theta_2 \partial \theta_1} & \frac{\partial}{\partial \theta_2^2} \end{vmatrix}_{\theta_1=\hat{\theta}_1, \theta_2=\hat{\theta}_2} > 0$$

- For our case we have:

$$\begin{vmatrix} \frac{-n}{\sigma^2} & -\frac{\sum_{i=1}^n (x_i - \mu)}{\sigma^4} \\ -\frac{\sum_{i=1}^n (x_i - \mu)}{\sigma^4} & \frac{n}{2\sigma^4} - \frac{\sum_{i=1}^n (x_i - \mu)^2}{\sigma^6} \end{vmatrix}_{\mu=\hat{\mu}, \sigma^2=\hat{\sigma}^2}$$

$$\begin{vmatrix} \frac{-n}{\hat{\sigma}^2} & -\frac{\sum_{i=1}^n (x_i - \bar{x})}{\hat{\sigma}^2{}^2} \\ -\frac{\sum_{i=1}^n (x_i - \bar{x})}{\hat{\sigma}^2{}^2} & \frac{n}{2\hat{\sigma}^2{}^2} - \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{\hat{\sigma}^2{}^3} \end{vmatrix}$$


$$\begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$$

$$\begin{vmatrix} \frac{-n}{\hat{\sigma}^2} & 0 \\ 0 & \frac{n}{2\hat{\sigma}^2} - \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{\hat{\sigma}^4} \end{vmatrix} = \frac{n^2}{2\hat{\sigma}^4} = \frac{n^2}{2\hat{\sigma}^6} > 0$$

- So the conditions are met. We should check boundaries as well.
- We could also just check that the eigenvalues of the Hessian matrix are negative. Here we can easily see them as we have a diagonal matrix.

$$\frac{-n}{\hat{\sigma}^2} \quad \text{and} \quad \frac{-n}{2\hat{\sigma}^4}$$

MLEs - Some Computation

- Newton-Raphson (N-R) Method:
- N-R is an extremely fast root finding approach, but is sensitive to starting values.
- Again let's consider a log likelihood function $\ell(\theta|\mathbf{x})$.
- Let $U(\theta) = \ell'(\theta|\mathbf{x})$ denote first derivative of $\ell(\theta)$.
- Let $H(\theta) = \ell''(\theta|\mathbf{x})$ denote the derivative of $\ell(\theta)$.
- Let θ_0 be an initial estimate of θ .  *second*
- Let $\hat{\theta}$ be the MLE.
- Let's do a Taylor series expansion of $U(\theta)$ around θ_0

$$U(\theta) = U(\theta_0) + (\theta - \theta_0)H(\theta_0) + \dots$$

- At $\theta = \hat{\theta}$ we know $U(\hat{\theta}) = 0$, so we have

$$0 = U(\theta_0) + (\hat{\theta} - \theta_0)H(\theta_0) + \dots$$

MLEs - Computation

- Let's say the one-step approximation is reasonable enough.

$$\hat{\theta} = \theta_0 - H^{-1}(\theta_0)U(\theta_0)$$

- This suggests that $\hat{\theta}$ is well approximated by θ_1 :

$$\theta_1 = \theta_0 - H^{-1}(\theta_0)U(\theta_0)$$

- We can then get an improved estimate:

$$\theta_2 = \theta_1 - U(\theta_1)H^{-1}(\theta_1)$$

- We can continue with $\theta_3, \theta_4, \dots$ until convergence is achieved.

$$|\theta_k - \theta_{k-1}| < \epsilon = 1e - 07$$

MLEs - Computation

Eg. Poisson: $X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{Poisson}(\lambda)$.

$$\ell(\lambda) = -n\lambda + \sum_{i=1}^n x_i \log(\lambda) - \sum_{i=1}^n \log(x_i!)$$

$$\ell'(\lambda) = -n + \frac{\sum_{i=1}^n x_i}{\lambda}$$

$$\ell''(\lambda) = -\frac{\sum_{i=1}^n x_i}{\lambda^2}$$

$$\lambda_{t+1} = \lambda_t - \left(-\frac{\sum_{i=1}^n x_i}{\lambda_t^2} \right)^{-1} \left(\frac{\sum_{i=1}^n x_i}{\lambda_t} - n \right)$$

```
set.seed(1001)
n <- 100
x <- rpois(n, 5)  $\lambda_{true} = 5$ 
```

```
## Let's find the MLEs using the Newton-Raphson Approach
## Starting values - typically we have to be a bit careful
lambda <- 10
```

```
## Write some functions for U and H
```

```
U <- function(lambda, x){
  n <- length(x)
  out <- -n + sum(x)/lambda
  return(out)
}
```

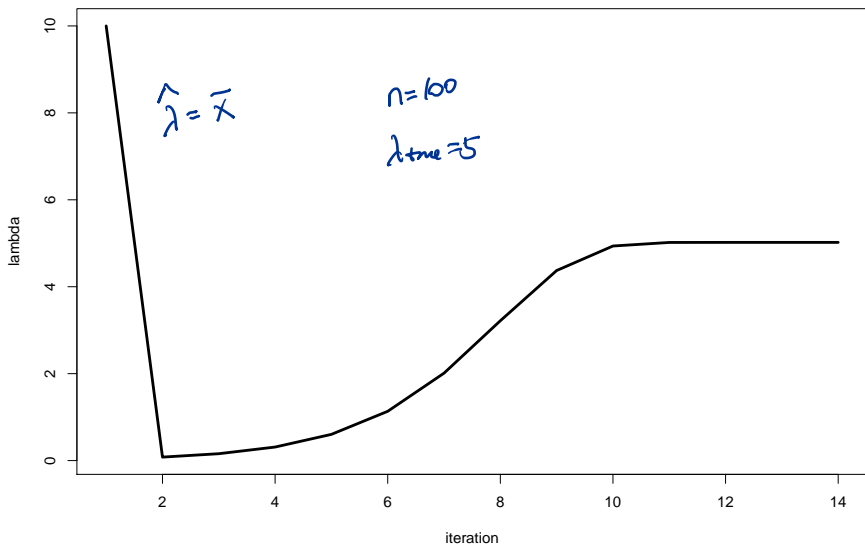
```
H <- function(lambda, x){
  n <- length(x)
  out <- -sum(x)/lambda^2
  return(out)
}
```

```
## set a stopping point
eps <- 1e-07
check <- 10

## We are only interested in the final results.
## Why not save them as we go along.
out <- lambda

## Run the algorithm
while(check > eps){
  lambda.new <- lambda - U(lambda, x)/H(lambda, x)
  check <- abs(lambda - lambda.new)
  lambda <- lambda.new
  out <- c(out, lambda)
}
```

```
plot(out, type="l", lwd=3, xlab="iteration", ylab="lambda")
```



```
lambda
```

```
## [1] 5.02
```

$$\hat{\lambda}_{NR}$$

*Newton-Raphson
works*

```
mean(x)
```

```
## [1] 5.02
```

$$= \hat{\lambda} = \bar{x}$$

analytical result

- We find that $\hat{\lambda} = \bar{x}$. What we want.

MLEs - Some Computation

- We can naturally extend the N-R approach to a multivariate setting.
- Let $U(\boldsymbol{\theta})$ denote the vector of first partial derivatives of $\ell(\boldsymbol{\theta})$.
- Let $H(\boldsymbol{\theta})$ denote the matrix of second partial derivatives of $\ell(\boldsymbol{\theta})$.

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - H^{-1}(\boldsymbol{\theta}_t)U(\boldsymbol{\theta}_t)$$

MLEs - Computation

Eg. Normal: $X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{normal}(\mu, \sigma^2)$.

- We saw U and H on previous slides.

```
[set.seed(1001)
 x <- rnorm(100, 2, 5)]
```

$n = 100$
 $\lambda_{true} = 2$

$x \leftarrow \text{rnorm}(n, \mu, \sigma)$

$\sigma_{true} = 5$

Let's find the MLEs using the Newton-Raphson Approach

Write some functions for U and H

```
U <- function(mu, sigma.sq, x){
```

```
  n <- length(x)
```

```
  out <- matrix( c( sum(x-mu)/sigma.sq, - n/(2*sigma.sq) +
                    (1/(2*sigma.sq^2))*sum((x-mu)^2) ), 2, 1)
```

```
  return(out)
```

```
}
```

row column

```
H <- function(mu, sigma.sq, x){
```

```
  n <- length(x)
```

```
  out <- matrix( c( -n/sigma.sq, -sum((x-mu))/sigma.sq^2,
                    -sum((x-mu))/sigma.sq^2, n/(2*sigma.sq^2) -
                    sum((x-mu)^2)/sigma.sq^3), 2, 2, byrow=TRUE)
```

```
  return(out)
```

```
}
```

r c

```
## Starting values - typically we have to be a bit careful.
mu <- 5
sigma.sq <- 1
theta <- c(mu, sigma.sq)

## set a stopping point
eps <- 1e-07
check <- 10
c <- 2

## Save the results.
out <- theta

## Run the algorithm
while(check > eps){
  theta.new <- theta - solve(H(mu, sigma.sq, x)) %*% U(mu, sigma.sq, x)
  check <- sum(abs(theta-theta.new))
  mu <- theta.new[1]
  sigma.sq <- theta.new[2]
  theta <- theta.new
  out <- rbind(out, t(theta))
  c <- c+1
}
```

Handwritten notes:

- H^T (with an arrow pointing to the `H(mu, sigma.sq, x)` term)
- matrix multiplication (with a squiggle pointing to the `%*%` operator)
- row bind (with an arrow pointing to the `t(theta)` term)

```
theta
```

```
##           [,1]  
## [1,]  1.995762  
## [2,] 32.418953
```

$\hat{\mu}_{NR}$
 $\hat{\sigma}_{NR}^2$

computational approach

```
mean(x)
```

```
## [1] 1.995762
```

$= \hat{\mu} = \bar{x}$

analytic method

```
n <- length(x)  
sum((x-mean(x))^2)/n
```

```
## [1] 32.41895
```

- Again $\hat{\mu} = \bar{x}$ and $\hat{\sigma}^2 = \sum_{i=1}^n (x_i - \bar{x})^2 / n$.

MLE Computation - Fisher Scoring

- The method of “scoring” is a simple modification of the Newton-Raphson method.
- The Hessian $H(\theta)$ is replaced by its expectation.

$$E[H(\theta)] = -I(\theta)$$

where $I(\theta)$ is Fisher's information matrix.

$$I(\theta) = E \left[\left(\frac{\partial \ell(\theta|\mathbf{x})}{\partial \theta_i} \right) \left(\frac{\partial \ell(\theta|\mathbf{x})}{\partial \theta_j} \right) \right] = -E \left[\frac{\partial^2 \ell(\theta|\mathbf{x})}{\partial \theta_i \partial \theta_j} \right]$$

$$\theta_{t+1} = \theta_t + I^{-1}(\theta_t)U(\theta_t)$$

- A great advantage is that $E[H(\theta)]$ is guaranteed to be positive definite, thus eliminating some possible convergence issue with the Newton-Raphson approach.

MLE Computation - Fisher Scoring

Eg. Normal: $X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{normal}(\mu, \sigma^2)$.

$$H = \begin{bmatrix} \frac{-n}{\sigma^2} & -\frac{\sum_{i=1}^n (x_i - \mu)}{\sigma^4} \\ -\frac{\sum_{i=1}^n (x_i - \mu)}{\sigma^4} & \frac{n}{2\sigma^4} - \frac{\sum_{i=1}^n (x_i - \mu)^2}{\sigma^6} \end{bmatrix}$$

$$l(\theta) = -E[H] = - \begin{bmatrix} \frac{-n}{\sigma^2} & -\frac{\sum_{i=1}^n (E[x_i] - \mu)}{\sigma^4} \\ -\frac{\sum_{i=1}^n (E[x_i] - \mu)}{\sigma^4} & \frac{n}{2\sigma^4} - \frac{\sum_{i=1}^n E[(x_i - \mu)^2]}{\sigma^6} \end{bmatrix}$$

$$I(\theta) = - \begin{bmatrix} \frac{-n}{\sigma^2} & 0 \\ 0 & \frac{n}{2\sigma^4} - \frac{n\sigma^2}{\sigma^6} \end{bmatrix}$$

$$I(\theta) = \begin{bmatrix} \frac{n}{\sigma^2} & 0 \\ 0 & \frac{n}{2\sigma^4} \end{bmatrix}$$

$$I(\theta)^{-1} = \begin{bmatrix} \frac{\sigma^2}{n} & 0 \\ 0 & \frac{2\sigma^4}{n} \end{bmatrix}$$

```

set.seed(1001)
x <- rnorm(100, 2, 5)

## Let's find the MLEs using the Fisher Scoring Approach
## Write some functions for U and H
U <- function(mu, sigma.sq, x){
  n <- length(x)
  out <- matrix( c( sum(x-mu)/sigma.sq, - n/(2*sigma.sq) +
                    (1/(2*sigma.sq^2))*sum((x-mu)^2) ), 2, 1)
  return(out)
}

I.fish <- function(mu, sigma.sq, x){
  n <- length(x)
  out <- matrix( c( n/sigma.sq, 0,
                    0, n/(2*sigma.sq^2)), 2,2, byrow=TRUE)
  return(out)
}

```



```
## Starting values - We don't need to be careful now!!!
mu <- -25
sigma.sq <- 50
theta <- c(mu, sigma.sq)

## set a stopping point
eps <- 1e-07
check <- 10
c <- 2

## Save the results.
out <- theta

## Run the algorithm
while(check > eps){
  theta.new <- theta + solve(I.fish(mu, sigma.sq, x)) %*% U(mu, sigma.sq, x)
  check <- sum(abs(theta-theta.new))
  mu <- theta.new[1]
  sigma.sq <- theta.new[2]
  theta <- theta.new
  out <- rbind(out, t(theta))
  c <- c+1
}
```

```
theta
```

```
##           [,1]  
## [1,]  1.995762  
## [2,] 32.418953
```

} Fisher
scoring

```
mean(x)
```

```
## [1] 1.995762
```

```
n <- length(x)  
sum((x-mean(x))^2)/n
```

```
## [1] 32.41895
```

- Again $\hat{\mu} = \bar{x}$ and $\hat{\sigma}^2 = \sum_{i=1}^n (x_i - \bar{x})^2 / n$.

MLE Computation - `optim()`

- R has a fairly robust optimizer (`optim()`). Additionally, R has several of the “new” genetic optimizer (we will not cover those).
- `optim()` has four different procedures built into it. We will use the “BFGS” (Broyden–Fletcher–Goldfarb–Shanno) method which is a quasi Newton-Raphson approach (just consider it a more robust version).
- `optim()` is actually a minimizer, so we have to tell it to maximize.

Eg. Normal: $X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{normal}(\mu, \sigma^2)$.

MLE Computation - optim()

```
set.seed(1001)
x <- rnorm(100, 2, 5)

## likelihood function
log.lik <- function(theta){
  mu <- theta[1]
  sigma.sq <- theta[2]
  out <- sum(dnorm(x, mu, sqrt(sigma.sq), log=TRUE))
  return(out)
}

## starting values
theta.start <- c(-25,50)

##
out <- optim(theta.start, log.lik, hessian = TRUE,
             control = list(fnscale=-1), method="BFGS")
```

$$L(\theta) = \prod_{i=1}^n f(x_i; \theta)$$

$$l(\theta) = \sum_{i=1}^n \log(f(x_i; \theta))$$

$$\max_{\theta} l(\theta) = \min_{\theta} (-l(\theta))$$

```
## Warning in sqrt(sigma.sq): NaNs produced
```

```
## Warning in sqrt(sigma.sq): NaNs produced
```

out

```
## $par
## [1] 1.995893 32.392865
##
## $value
## [1] -315.831
##
## $counts
## function gradient
##      30      23
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##           [,1]           [,2]
## [1,] -3.087100e+00 1.239897e-05
## [2,] 1.239897e-05 -4.772767e-02
```

$$(\hat{\mu}, \hat{\sigma}^2) = \hat{\theta}$$

$$l(\hat{\theta})$$

MLE Computation - `optim()`

- We did get some warnings. Likely the algorithm tried values where $\sigma^2 < 0$.
- However, the convergence value is 0, which means the algorithm met the convergence criterion (see the help for `optim()` for other codes).
- $\ell(\hat{\theta}) = -315.8$ and $\hat{\theta} = (\hat{\mu} = 1.996, \hat{\sigma}^2 = 32.393)$.
- There are a number of ways to tweak this function (more iterations, convergence criterion, provide the first and second derivative, . . .)

MLE Computation - optim()

- The estimated variances of the estimators are given by the inverse of the Fisher information matrix (we will get to this):

$$V(\hat{\theta}) = I^{-1}(\hat{\theta})$$

$$I(\hat{\theta}) = -H(\hat{\theta})$$

```
diag(solve(-out$hessian))
```

```
## [1] 0.3239287 20.9522078
```

$\hat{V}(\hat{\mu})$ estimated variance

$\hat{V}(\hat{\sigma}^2)$

- Quick check:

$$V(\hat{\mu}) = V(\bar{X}) = \sigma^2/n \Rightarrow \hat{\sigma}^2/n = 32.393/100 = 0.32393$$

- Wait. We didn't take expectations? That is ok. This is actually called the **Observed Fisher Information**.

3.1 (b)

$X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{Unif}(-\frac{\theta}{2}, \frac{\theta}{2})$

Find the MLE for θ

$$f(x, \theta) = \frac{1}{b-a} = \frac{1}{\frac{\theta}{2} - (-\frac{\theta}{2})} = \frac{1}{\theta}$$

$$L(\theta) = \prod_{i=1}^n \frac{1}{\theta} = \frac{1}{\theta^n} \Rightarrow \theta \uparrow \quad L(\theta) \downarrow$$

\Rightarrow want $\theta \downarrow \quad L(\theta) \uparrow$

make θ as small as possible

$$\underline{-\frac{\theta}{2}} < x_{(1)} < \dots < x_{(n)} < \underline{\frac{\theta}{2}}$$

$$\frac{\theta}{2} > \max(|x_{(1)}|, |x_{(n)}|)$$

$$\hat{\theta} > 2 \max(|x_{(1)}|, |x_{(n)}|)$$

$$\hat{\theta} = 2 \max(|x_{(1)}|, |x_{(n)}|)$$

next time