COMPARING TWO ESTIMATORS

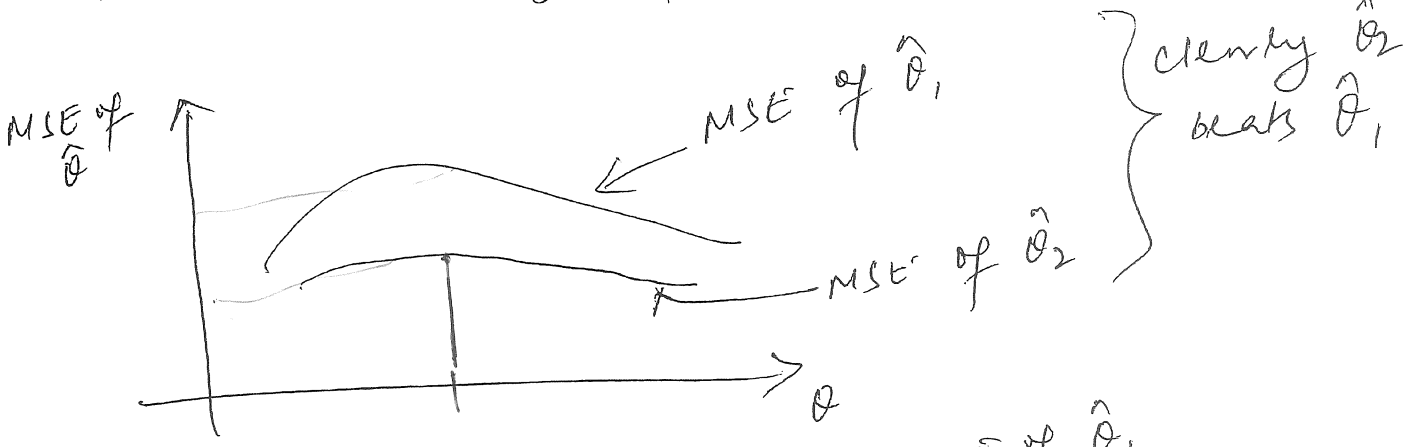
Mean squared error (MSE) of $\hat{\theta}$ of θ:

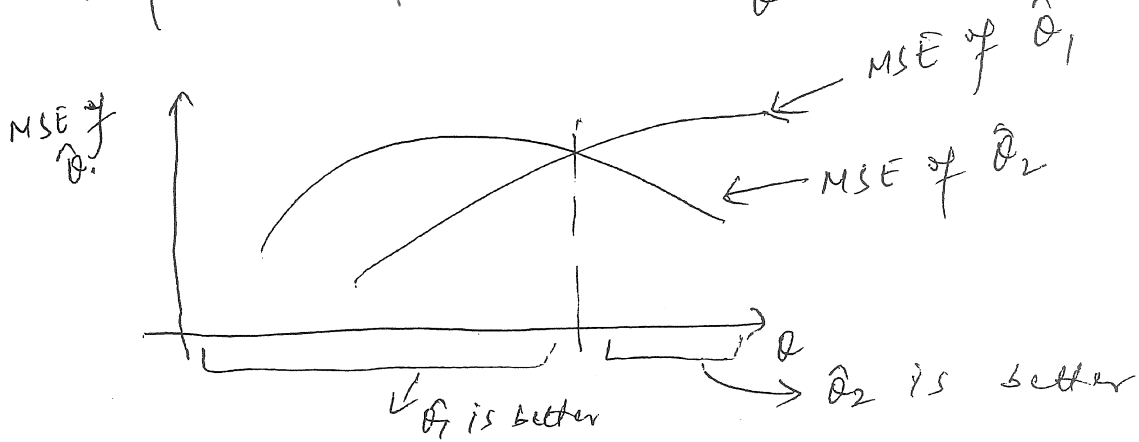
$$E[(\hat{\theta}-\theta)^2] \rightarrow \text{depends on } \theta.$$

Average squared-distance b/w $\hat{\theta}$ and θ.

- Note:

$$MSE_\theta(\hat{\theta}) = \underset{\theta}{Var}[\hat{\theta}] + [\underset{\theta}{Bias}(\hat{\theta})]^2.$$





Note:

$$MSE_\theta(\hat{\theta}) = E[(\hat{\theta}-\theta)^2]$$

$$= E[(\hat{\theta} - E(\hat{\theta}) + E(\hat{\theta}) - \theta)^2]$$

$$= Var_\theta(\hat{\theta}) + [Bias_\theta(\hat{\theta})]^2.$$

where $Bias_\theta = E[\hat{\theta}] - \theta$

- If $\hat{\theta}$ is unbiased, $MSE_\theta(\hat{\theta}) = Var_\theta(\hat{\theta})$.

Large sample properties of MLE $\hat{\theta}$ of θ

Result: Assume that $\{x : f_\theta(x) > 0\}$ is free of θ. Then, under certain conditions when n is large,

$$\hat{\theta} \approx N(\theta, \hat{I}^{-1}), \text{ where } \hat{I} = \left. -\frac{\partial^2 \log\{L(\theta)\}}{\partial \theta^2} \right|_{\theta=\hat{\theta}}.$$

Case 1: θ is scalar. Then, $\widehat{SE}(\hat{\theta}) \approx \sqrt{\hat{I}^{-1}}$

Case 2: θ is a vector, say, $\theta = (\theta_1, \ldots, \theta_d)$. In this case, $\hat{I}$ is a $d \times d$ matrix. Here $\widehat{SE}(\hat{\theta}_j) \approx (j\text{-th diagonal element of } \hat{I}^{-1})^{1/2}$.

Properties of MLE:

- Consistent; asymptotically unbiased
- Asymptotically normal
- Optimal if the assumed model holds
- Not a good choice if the assumed model does not hold

Recall: $SE(\bar{X}) = \frac{\sigma}{\sqrt{n}}$

$\widehat{SE}(\bar{X}) = \frac{s}{\sqrt{n}}$

$SE(\hat{\theta}) \approx \sqrt{1/\hat{I}}$

n is absorbed in $\hat{I}$

Using R to get MLE

Ex: Recall the CPU data — CPU times for $n = 30$ randomly chosen jobs (in seconds): 70, 36, 43, 69, 82, 48, 34, 62, 35, 15, 59, 139, 46, 37, 42, 30, 55, 56, 36, 82, 38, 89, 54, 25, 35, 24, 22, 9, 56, 19. Graphics suggested that the distribution of these CPU times may be right-skewed. Suppose we ***assume*** that the parent distribution is Gamma (α, λ), with both parameters unknown. What are MLE's of these parameters?

$X \sim \text{Gamma}(\alpha, \lambda)$

$$f_{\alpha,\lambda}(x) = \begin{cases} \frac{\lambda^\alpha}{\Gamma\alpha} x^{\alpha-1} e^{-\lambda x}, & x > 0, \ (\alpha > 0, \lambda > 0) \\ 0, & \text{o/w} \end{cases}$$

constraint on α, λ that optim needs to know

$$\log\left[L(\alpha,\lambda)\right] = \log\left[\prod_{i=1}^{n} f_{\alpha,\lambda}(x_i)\right]$$

— cannot maximize this explicitly.

maximize this numerically.

```
# We will continue working with the CPU data
# that we saw earlier

cpu <- scan(file="cputime.txt")

# Negative of log-likelihood function assuming gamma
# parent distribution

neg.loglik.fun <- function(par, dat)
{
result <- sum(dgamma(dat, shape=par[1], rate=par[2],
log=TRUE))
return(-result)
}

# Minimize -log (L), i.e., maximize log (L)

ml.est <- optim(par=c(3, 0.1), fn=neg.loglik.fun,
```

Optimization routines by default do minimization.

(α, λ)

vector

log-likelihood fn.

approximately the method of moments estimates

(read book)

```
method = "L-BFGS-B", lower=rep(0,2), hessian=TRUE,
dat=cpu)
```

allows taking into account constraints.

$c(0,0)$ $\Rightarrow \alpha > 0, \lambda > 0$.

```
# > ml.est
# $par
# [1] 3.63149628 0.07529459
```

$\rightarrow$ MLE's

$\hat{\alpha}$ $\hat{\lambda}$

```
# $value
# [1] 136.561
```

} min of $-\log L(\hat{\alpha}, \hat{\lambda}) \Rightarrow$ -136.561 is max. value of $\log L(\alpha, \lambda)$, i.e., the value at $(\alpha, \lambda) = (\hat{\alpha}, \hat{\lambda})$

```
# $counts
# function gradient
      # 20       20

# $convergence
# [1] 0
```

$\leftarrow$ "good"

```
# $message
```

```
# [1] "CONVERGENCE: REL_REDUCTION_OF_F <= FACTR*EPSMCH"

# $hessian
#              # [,1]        [,2]
# [1,]     9.501374  -398.4584
# [2,] -398.458449 19223.5065
# >

# MLE

# > ml.est$par
# [1] 3.63149628 0.07529459
# >

# their standard errors

# > sqrt(diag(solve(ml.est$hessian)))
# [1] 0.89720941 0.01994668
```

$\hat{I}$

$\text{Var}\left[\begin{pmatrix}\hat{\alpha}\\ \hat{\lambda}\end{pmatrix}\right] = \hat{I}^{-1}$

In R, to find A^{-1}, solve(A)

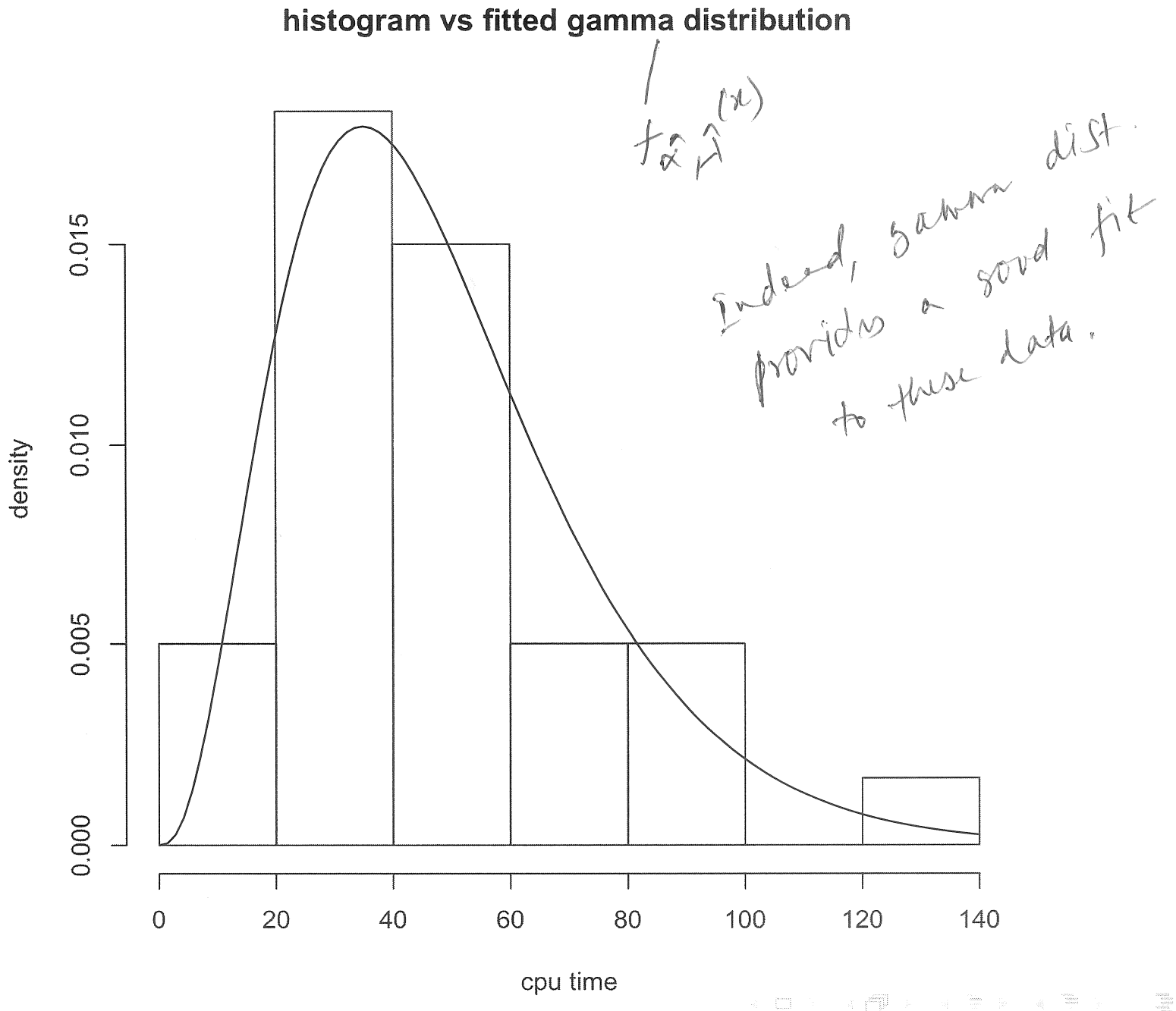
$\hat{\alpha}, \hat{\lambda}$.

$SE(\hat{\alpha})$ $SE(\hat{\lambda})$.

```
# >
```

— Do a Q-Q plot with gamma as reference dist.?
— will see later on.

```
# How well the fitted model represents the data?

# density histogram

hist(cpu, freq=FALSE, xlab="cpu time",
ylab="density",
main="histogram vs fitted gamma distribution")

# superimpose the fitted density

gamma.pdf <- function(x, shape=ml.est$par[1],
rate=ml.est$par[2])
{ dgamma(x, shape=shape, rate=rate) }

curve(gamma.pdf, from=0, to=140, add=TRUE)
```



$f_{\hat{\alpha}, \hat{\mu}}(x)$

Indeed, gamma dist. provides a good fit to these data.

Pitfalls of numerical techniques

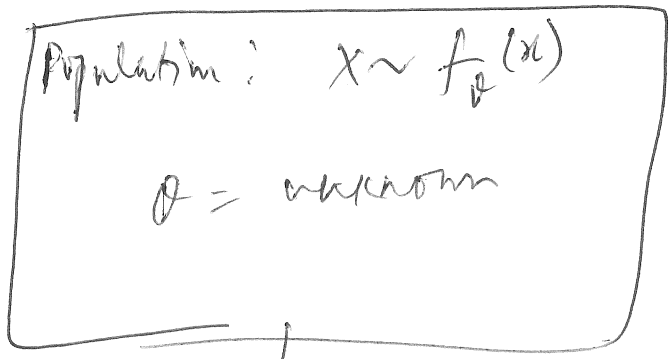
- Function needs to be well-behaved
- Multiple roots, roots need to be captured in an interval
- Convergence to local maxima
- flat likelihood in region of maxima
- Negative estimate for variance.

and others.

So far: Estimating θ by $\hat{\theta}$ — point estimation

Confidence intervals (Section 9.2)

Set up: Same as before, i.e.,



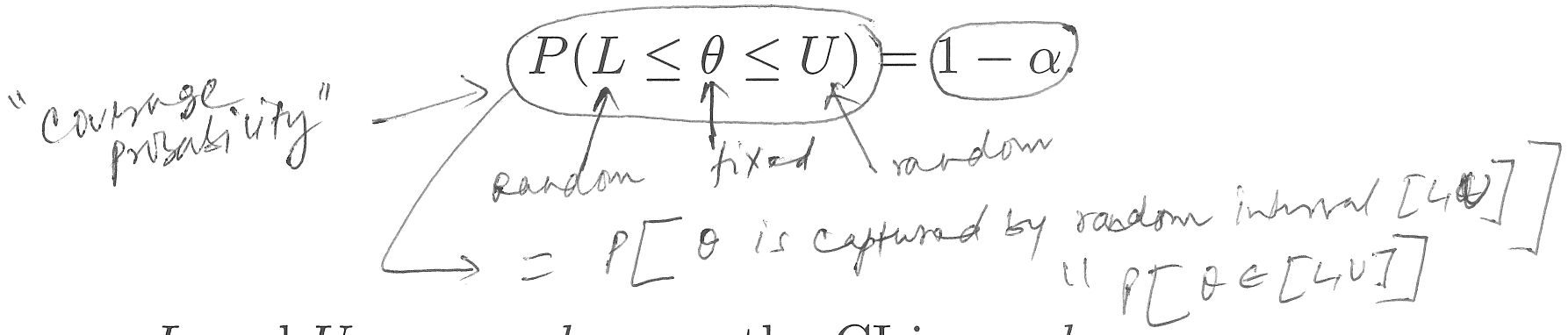
$X_1, X_2, \ldots, X_n$ (random sample).

$? = P[\hat{\theta} = \theta] = 0$.

cont. rv, which is typically the case

Motivation: Estimator $\hat{\theta}$ is a single number that gives a plausible value of the unknown θ. But rarely the two will be equal. So, often it is preferable to give an interval of plausible values — a **confidence interval** (CI), which contains the unknown θ with a specified high probability.

Definition: An interval $[L, U]$ is a $100(1-\alpha)\%$ CI for θ if $L = L(X_1, \ldots, X_n)$ and $U = U(X_1, \ldots, X_n)$ are such that



- L and U are *random*, so the CI is *random*.
- Parameter θ is not random — it is unknown but fixed.
- $(1-\alpha)$ = *confidence coefficient* or *confidence level*.
- In practice, $(1-\alpha) = 0.90$ or 0.95 (most common) or 0.99.

A general method for constructing CI for θ

Step 1: Find an estimator $\hat{\theta}$ of θ that has a normal distribution with *known* variance, i.e., $\hat{\theta} \sim N(\theta, \text{var}(\hat{\theta}))$.

Step 2: Standardize $\hat{\theta}$ to get Z, where

"pivot" $\rightarrow Z = \dfrac{\hat{\theta} - \theta}{SE(\hat{\theta})} \sim N(0,1)$

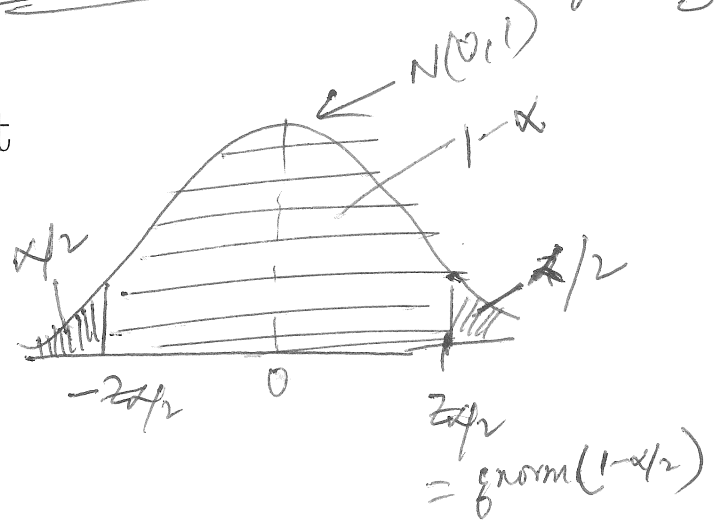
$SD(\hat{\theta}) = SE(\hat{\theta})$

A function of $\hat{\theta}$ and θ whose dist. is completely known

Step 3: Find a *critical point* $z_{\alpha/2}$ such that

$1 - \alpha = P(-z_{\alpha/2} \leq Z \leq z_{\alpha/2})$

$\downarrow N(0,1)$



Thus, the $100(1-\alpha)\%$ CI is:

see next page

Have $z_{\alpha/2}$ such that

$$1-\alpha = P\left[-z_{\alpha/2} \le Z \le z_{\alpha/2}\right]$$

$$= P\left[-z_{\alpha/2} \le \frac{\hat{\theta}-\theta}{SE(\hat{\theta})} \le z_{\alpha/2}\right] =$$

Plug in def. of Z

$$P\left[-z_{\alpha/2} SE(\hat{\theta}) - \hat{\theta} \le -\theta \le z_{\alpha/2} SE(\hat{\theta}) - \hat{\theta}\right]$$

Multiply by $SE(\hat{\theta})$ and subtract $\hat{\theta}$

$$= P\left[\underbrace{\hat{\theta} - z_{\alpha/2} SE(\hat{\theta})}_{L} \le \theta \le \underbrace{\hat{\theta} + z_{\alpha/2} SE[\hat{\theta}]}_{U}\right]$$

multiply by -1

$\Rightarrow$ From def., $100(1-\alpha)\%$ CI for θ: $\hat{\theta} \pm z_{\alpha/2} SE(\hat{\theta})$

$= [L, U]$.

A general method, continued

Note 1: If the distribution of $\hat{\theta}$ is approximately normal, then the CI is also approximate.

Note 2: In case of MLE, if n is large, then often $\hat{\theta}$ approximately follows a $N(\theta, \hat{I}^{-1})$ distribution. In this case, an approximate $100(1-\alpha)\%$ CI for θ is: $\hat{\theta} \pm z_{\alpha/2}\sqrt{1/\hat{I}}$

$\approx SE(\hat{\theta})$.

Note 3: If the distribution is not normal or n is not large, we can use the method of bootstrap to construct a CI (later).