# Point estimation (Chapter 9)

**Problem:** $X \sim f_\theta(x)$, where $\theta$ is an unknown parameter. This $\theta$ may be a vector.

**Data:** $X_1, \ldots, X_n$ — a random sample of $X$.

We have seen a number of descriptive statistics and what they estimate. But the choice of an $\hat{\theta}$ of $\theta$ may not be obvious.

**A general method of parameter estimation:** Method of maximum likelihood. It has generally good properties.

# Method of Maximum Likelihood

**Likelihood function of data**: Joint pdf or pmf of sample data considered as a function of $\theta$ with data held fixed at the observed values $X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n$.

$L(\theta) = L(\theta; x_1, x_2, , x_n) =$

- A function of $\theta$ — the data are held fixed.

**Maximum likelihood estimator (MLE) of $\theta$**: The value $\hat{\theta}$ of $\theta$ that maximizes the likelihood function as a function of $\theta$.

- Can think of MLE as the value of $\theta$ that is "most likely" to have led to the observed data.
- Essentially a calculus problem.

# How to find MLE?

**Direct approach:** Directly maximize the likelihood function.

**Ex:** Let $X_1, X_2, \ldots, X_n$ represent a random sample from a Uniform $(0, \theta)$ distribution where $\theta > 0$. Find the MLE of $\theta$.

**Differentiation technique:** Maximize the log-likelihood function $\log\{L(\theta)\}$ with respect to $\theta$ instead of $L(\theta)$ as the former tends to be easier. The value of $\theta$ that maximizes $L(\theta)$ also maximizes $\log\{L(\theta)\}$. (**Why?**)

Step 1: Set up the log-likelihood function.

Step 2: Find the *likelihood equation* by partially differentiating $\log\{L(\theta)\}$ with respect to $\theta$ and setting the derivative to equal to zero.

Step 3: Solve the likelihood equation for $\theta$. The solution is MLE if it is a point of maxima (no need to verify).

**Recall:** Some useful properties of natural log:

- $\log(ab) = \log(a) + \log(b)$
- $\log(a^b) = b\log(a)$
- $\log(e^a) = a$

**Ex:** Let $X_1, X_2, \ldots, X_n$ represent a random sample from an Exponential $(\lambda)$ distribution where $\lambda > 0$. Find the MLE of $\lambda$.

**Ex:** Suppose $X_1, X_2, \ldots, X_n$ denote a random sample from a Bernoulli $(p)$ distribution, where $p$ is unknown. Find its MLE.

# Finding standard error (SE) of $\hat{\theta}$

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

**Result:** Assume that $\{x : f_\theta(x) > 0\}$ is free of $\theta$. 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**: $\theta$ is scalar. Then, $\widehat{SE}(\hat{\theta}) \approx \sqrt{\hat{I}^{-1}}$

**Case 2:** $\theta$ is a vector, say, $\theta = (\theta_1, \ldots, \theta_d)$. In this case, $\hat{I}$ is a $d \times d$ matrix. Here $\widehat{SE}(\hat{\theta}_j) \approx$ ($j$-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

# Using `R` to get MLE

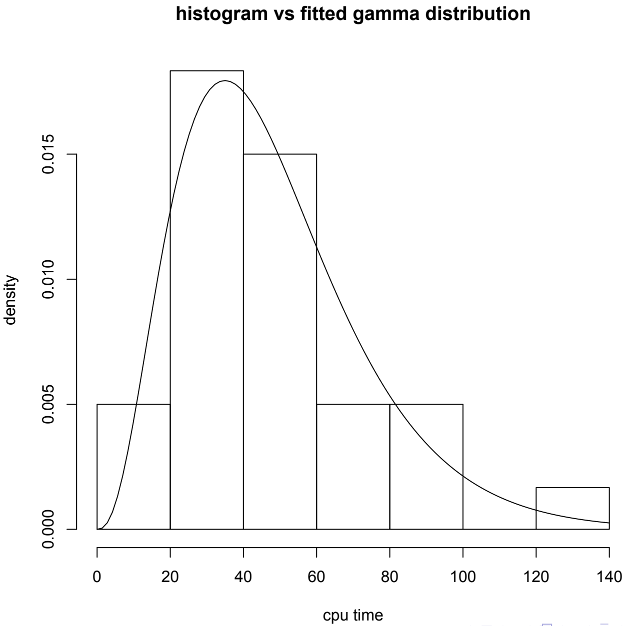
**Ex:** Recall the CPU data — CPU times for $n = 30$ randomly chosen jobs (in seconds): 70, 36, 43, 69, 82, 48, 34, 62, 35, 15, 59, 139, 46, 37, 42, 30, 55, 56, 36, 82, 38, 89, 54, 25, 35, 24, 22, 9, 56, 19. Graphics suggested that the distribution of these CPU times may be right-skewed. Suppose we ***assume***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

```
# 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,
```

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

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

# $value
# [1] 136.561

# $counts
# function gradient
      # 20        20

# $convergence
# [1] 0

# $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
```

```
# >

# 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)
```


histogram vs fitted gamma distribution
density
0.000
0.005
0.010
0.015
0
20
40
60
80
100
120
140
cpu time