Pembangkitan Bilangan Acak dan Resampling

Materi 5 - STK571 Komputasi Statistik

Pembangkitan Bilangan Acak

Des, 2017

Pendahuluan

- Pembangkitan bil. acak merupakan alat yang diperlukan dalam komputasi statistik → umumnya untuk simulasi
- Bilangan acak yang dibangkitkan merupakan pseudorandom (acak yang semu)
- Bilangan acak yg dibangkitkan diharapkan memenuhi sebaran statistik tertentu (pdf/pmf, cdf)
- Semua metode pembangkitkan bil. acak tergantung dari pembangkitan bil. acak uniform

Membangkitkan Bilangan Acak

- R telah menyiapkan banyak fungsi untuk membangkitkan data berdasarkan sebaran
- Fungsi umumnya dimulai dengan huruf r diikuti dengan nama sebaran

  Contoh:

  - Data sebaran pseudo normal : rnorm
  - Data sebaran pseudo seragam : runif
- Pembangkit bilangan menggunakan *seed* yang umumnya mengambil waktu di komputer

Peluang Sebaran

- Fungsi density/mass (pdf/pmf) dimulai huruf d diikuti dengan nama sebarannya
  - dnorm
  - dunif
- Fungsi kumulatif (cdf) dimulai huruf p diikuti dengan nama sebarannya
  - pnorm
  - punif
- Fungsi quantile/invers cdf dimulai huruf q diikuti dengan nama sebarannya
  - qnorm
  - qunif

Tabel Fungsi sebaran peluang dalam R

Distribution	Name	Parameters
Beta	`beta`	`shape1`
		`shape2`
Binomial	`binom`	`size`
		`prob`
Cauchy	`cauchy`	`location`
		`scale`
χ^2	`chisq`	`df`
Exponential	`exp`	`rate`
F	`f`	`df1`
		`df2`
Gamma	`gamma`	`shape`
Geometric	`geom`	`shape`
Lognormal	`lnorm`	`meanlog`
		`sdlog`
Logistic	`logis`	`location`
		`scale`
Negative Binomial	`nbinom`	`size`
		`prob`
Normal (Gaussian)	`norm`	`mean`
		`sd`
Poisson	`pois`	`lambda`
Student's t	`t`	`df`
Uniform	`unif`	`min`
		`max`
Weibull	`weibull`	`shape`
Empirical cdf	`ecdf`	
Box–percentile plot	`bpplot`	list of vectors

Bagaimana jika belum ada fungsi pembangkit bil. acak?

Teknik Pembangkitan Bil. Acak

- Teknik umum dalam pembangkitan bil. acak
  - Inverse-transform method
  - Acceptance-rejection method
  - Other Special techniques: Direct Transformation

Inverse Transform Method

- Berdasarkan teori Probability Integral Transformation: Jika X adalah peubah acak kontinu dengan cdf F(x), maka U = F(X) ~ Uniform(0,1).
- Menerapkan transformasi integral peluang. Didefiniskan transformasi invers:

  $F^{-1}(u) = \inf\{x : F(x) = u\}, 0 < u < 1$

  Jika U ~ uniform(0,1), maka untuk semua x anggota R

  - $P(F^{-1}{}_X(u) \leq x) = P(\inf\{t : F_X(t) = U\} \leq x) = P(U \leq F_X(x)) = F_U(F_X(x)) = F_X(x)$
  - Akhirnya $F^{-1}{}_X(u)$ memiliki sebaran yang sama dengan X

Inverse Transform Method

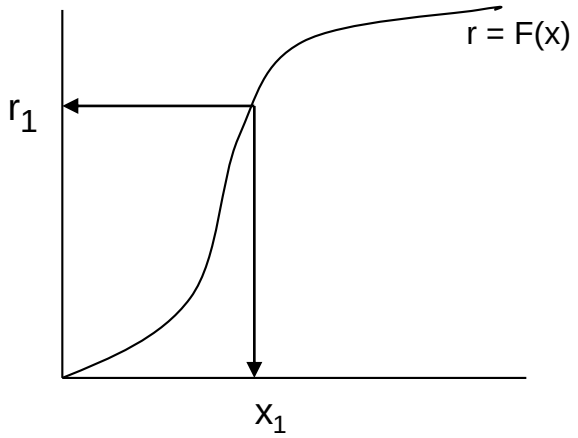
- Konsep:

  Untuk fungsi cdf : r = F(x)

  Bangkitkan r dari uniform (0,1)

  Maka x:

$$x = F^{-1}(r)$$



Inverse Transform Method

- Ilustrasi:

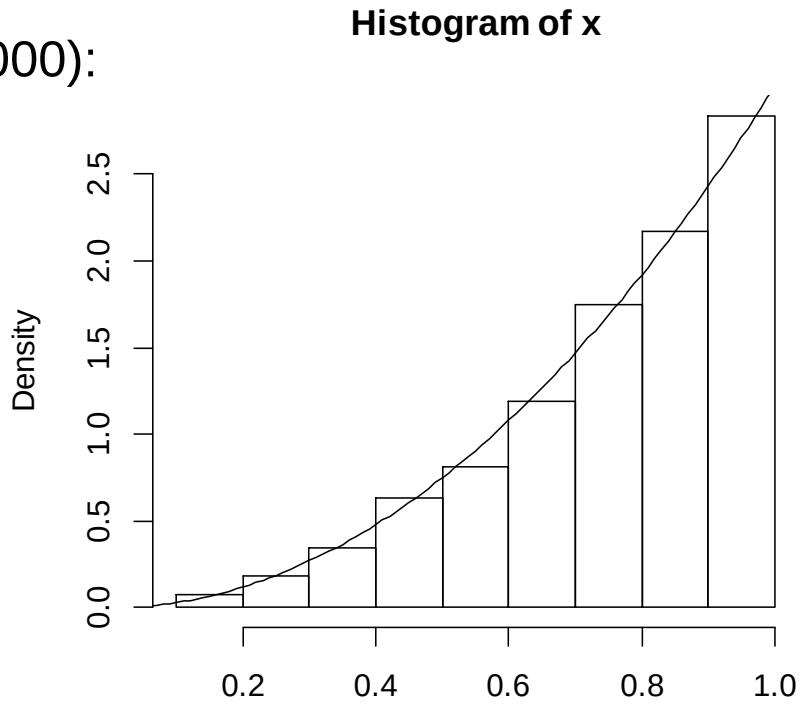
  Diketahui pdf : $f(x) = 3x^2,\ 0 < x < 1$

  $F_X(x) = x^3$, $0 < x < 1$

  $F^{-1}_X(u) = u^{1/3}$,

  Dalam R (misal membangkitkan 1000):

```
n <- 1000
u <- runif(n)
x <- u^(1/3)
#cek
hist(x, prob=TRUE)
y <- seq(0,1,.01)
lines(y, 3*y^2)
```



Inverse Transform Method

- Latihan:

  X dari sebaran eksponensial dengan mean 1/λ

  Jika X ~ Exp(λ), maka untuk x > 0 cdf dari X adalah

  $$F_X(x) = 1-e^{-\lambda x}$$

  Bangkitkan X ~ Exp (λ) sebanyak 1000

ITM: Sebaran Diskret

- Jika X ~ p.a. diskret dan $\ldots < x_{i-1} < x_i < x_{i+1} < \ldots$ adalah titik tidak kontinu dari $F_X(x)$, maka transformasi inversnya adalah $F^{-1}{}_X(u)=x_i$ dimana $F_X(x_{i-1}) < u < F_X(x_i)$.
- Langkah:

  Bangkitkan uniform (0,1)

  Tentukan x_i dimana $F_X(x_{i-1}) < u < F_X(x_i)$

ITM: Sebaran Diskret

- Ilustrasi:

  Membangkitkan bil. acak ~ Bernoulli (0.4)

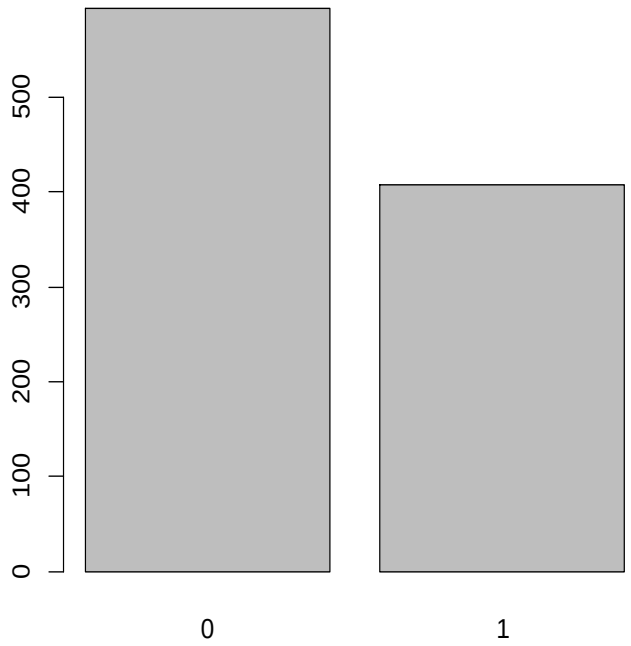
  $F_X(0) = f_x(0) = 1\text{-}p$ dan $F_X(1) = 1$.

  $F^{-1}_X(u) = 1$ jika $u > 0.6$

  $F^{-1}_X(u) = 0$ jika $u <= 0.6$

  Dalam R

```
n <- 1000
p <- 0.4
u <- runif(n)
x <- as.integer(u>0.6)
barplot(table(x))
```



ITM: Kasus Sebaran Diskret

Ilustrasi: Misal banyaknya pengiriman, x, dari suatu perusahaan adalah 0, 1, atau 2 kali

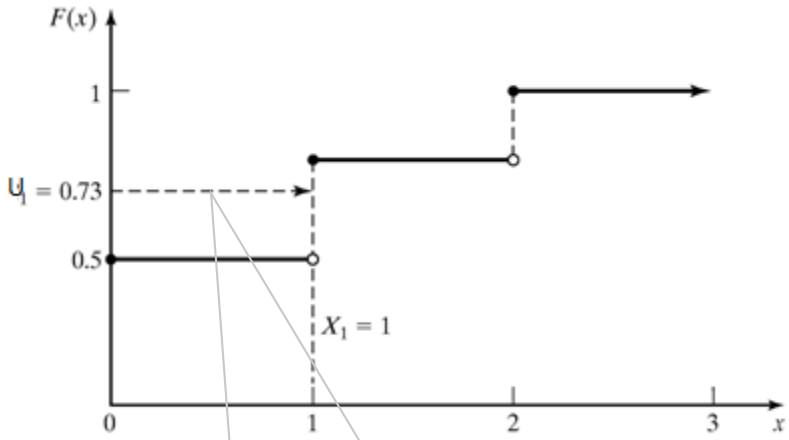
Data – Sebaran Peluang:

x	*p(x)*	*F(x)*
0	0.50	0.50
1	0.30	0.80
2	0.20	1.00

Metode– **Diberikan U,**

Skema pembangkit:

$$x = \begin{cases} 0, & U \le 0.5 \\ 1, & 0.5 < U \le 0.8 \\ 2, & 0.8 < U \le 1.0 \end{cases}$$



Perhatikan $U_1 = 0.73$:
$F(x_{i-1}) < U <= F(x_i)$
$F(x_0) < 0.73 <= F(x_1)$
Maka, $x_1 = 1$

Bagaimana jika sulit untuk mendapatkan cdf?

Acceptance-Rejection method

- Misalkan X dan Y adalah peubah acak dengan pdf/pmf *f* dan *g* dan terdapat konstanta *c* sehingga

  $f(t) / g(t) \le c$. Untuk semua t: $f(t) > 0$

- Teknik:

  1. Tetapkan peubah acak Y dengan density g yg memenuhi f(t)/g(t) ≤ c. Untuk semua *t: f*(t) > *0.*
  2. Untuk setiap satu bil. acak:
     a. Bangkitkan y acak dari sebaran dengan density g
     b. Bangkitkan u acak dari sebaran Uniform(0,1).
     c. Jika u < f(y)/(c g(y)) terima y dan x=y; selainnya tolak y dan ulangi langkah 2(a)

Acceptance-Rejection method

- Ilustrasi:

  Membangkitkan bil. acak sebaran beta (shape1=2, shape2=2)

  Pdf dari beta(2,2) : $f(x) = 6x(1-x)$, $0 < x < 1$.

$$\frac{x^{\alpha-1}(1-x)^{\beta-1}}{\mathrm{B}(\alpha,\beta)}$$

$$\mathrm{B}(x,y) = \int_0^1 t^{x-1}(1-t)^{y-1}\,dt$$

  Tahap:

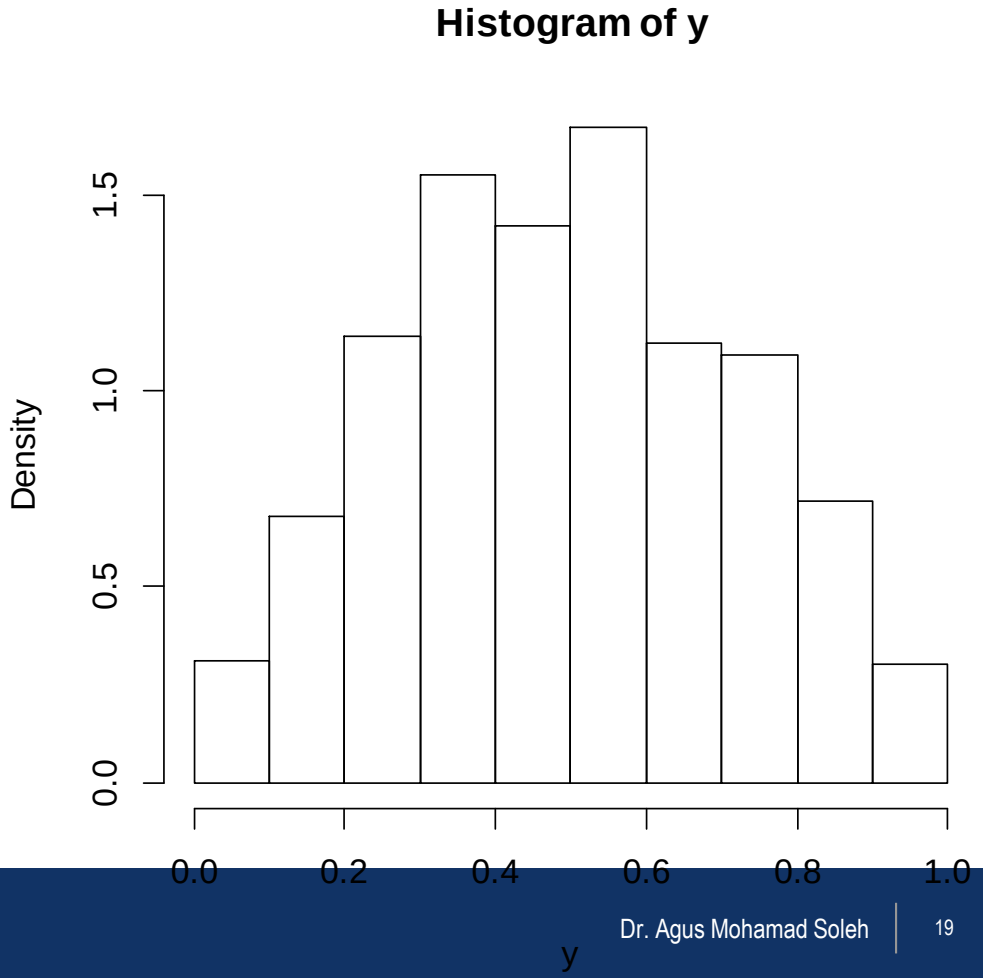
  1. Ambil $g(x)$ dari sebaran Uniform(0,1)
  2. Maka $f(x)/g(x) \leq 6$ untuk $0 < x < 1$.
  3. Sebuah x acak dari $g(x)$ diterima jika

     $f(x)/ [c\ g(x)] = 6x(1-x) / [6(1)] = x(1-x) > u$

Acceptance-Rejection method

- Dalam R:

```
n <- 1000
k <- 0
j <- 0
y <- numeric(n)
while (k < n) {
  u <- runif(1)
  j <- j+1
  x <- runif(1)
  if (x*(1-x) > u) {
    k <- k+1  # terima x
    y[k] <- x
  }
}
```



Metode Lain: Direct Transformation

- Beberapa transformasi dari tranformasi invers sebaran dapat digunakan untuk membangkitkan bil. acak:

Jika $Z \sim N(0,1)$, maka $V = Z^2 \sim \chi^2(1)$

Jika $U \sim \chi^2(m)$, $V \sim \chi^2(n)$, U dan V bebas, maka

$F = (U/m) / (V/n) \sim F(m,n)$

Jika $Z \sim N(0,1)$ $V \sim \chi^2(n)$ dan U dan V bebas, maka

$T = Z / \text{sqrt}(V/n) \sim$ t-student (n)

dst

Resampling

Des, 2017

Resampling

- Bootstrap
- Jacknife
- Cross Validation

Why resampling?

- Fewer assumptions

  Ex: resampling methods do not require that distributions be Normal or that sample sizes be large

- Greater accuracy: Permutation tests and come bootstrap methods are more accurate in practice than classical methods

- Generality: Resampling methods are remarkably similar for a wide range of statistics and do not require new formulas for every statististic.

- Promote understanding: Boostrap procedures build intuition by providing concrete analogies to theoretical concepts.

Use Bootstraps and Jackknives when:

- Deriving statistical model

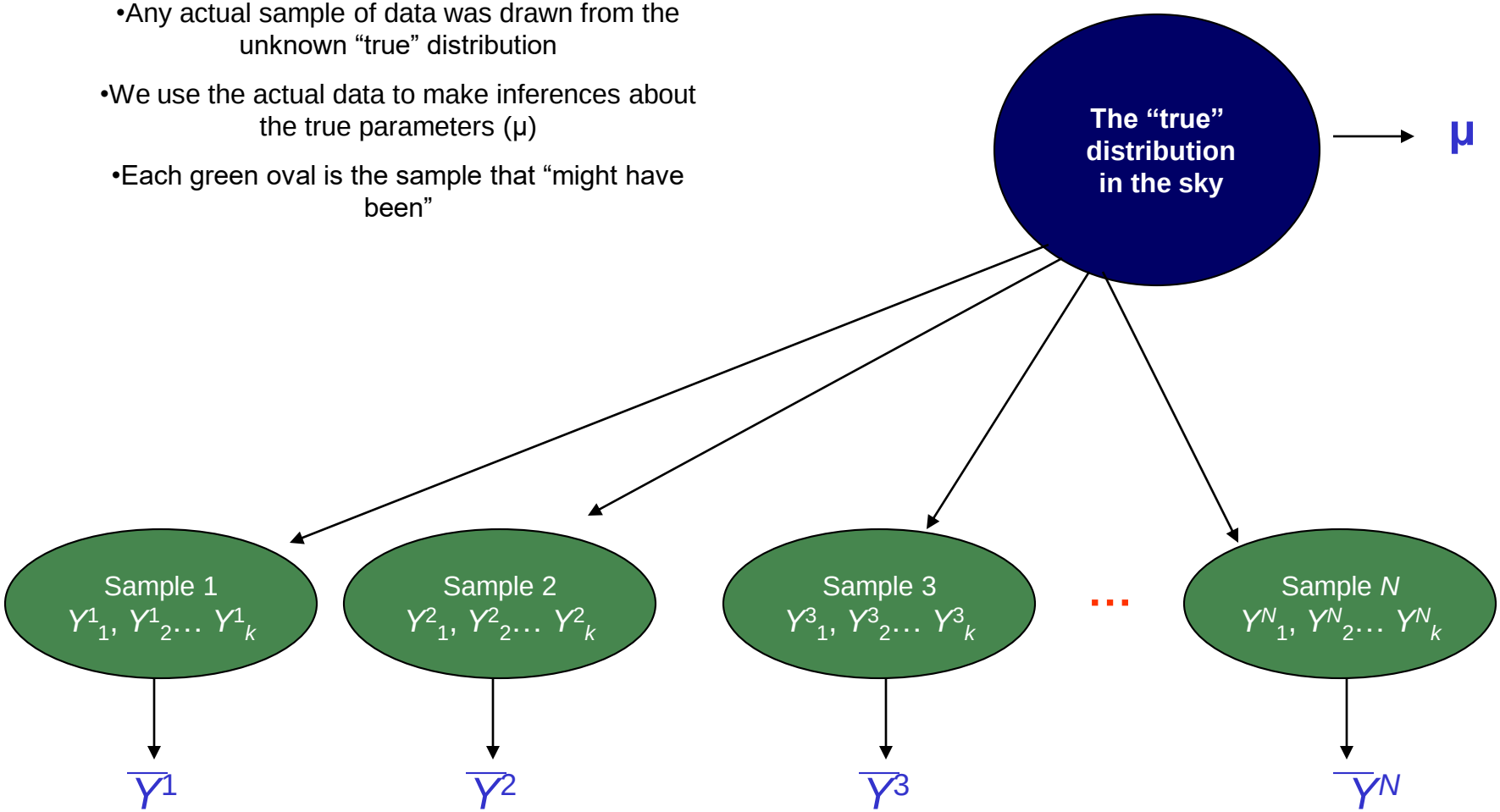
  very difficult, impossible, or tedious
- Statistical model too complicated to be useful
- Model may not be quite valid
- Accurate measure of precision under statistical model only possible with large *n*

Bootstrap

The Basic Idea

Theoretical Picture

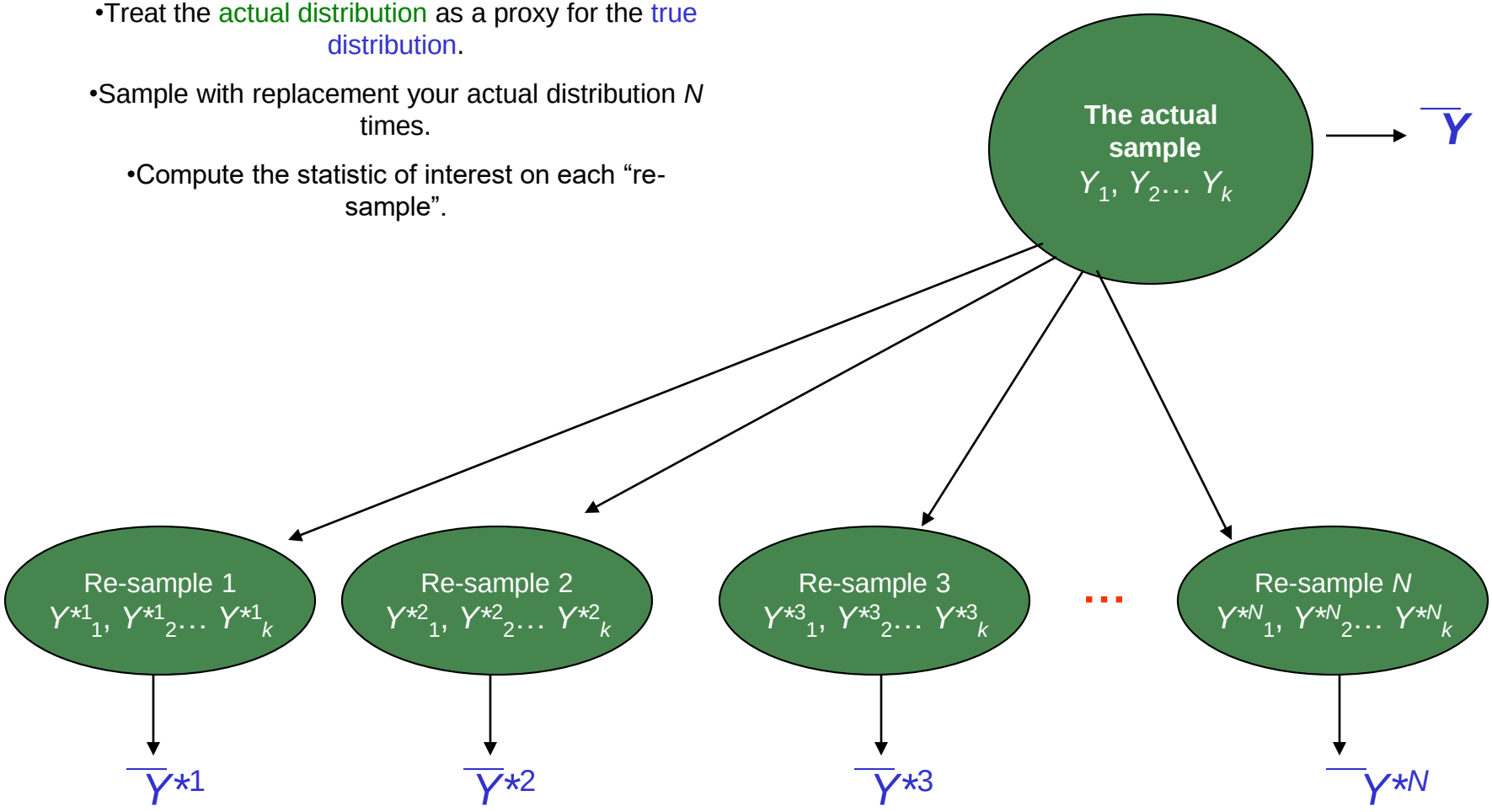
- Any actual sample of data was drawn from the unknown "true" distribution
- We use the actual data to make inferences about the true parameters (μ)
- Each green oval is the sample that "might have been"



- The distribution of our estimator $(\overline{Y})$ depends on both the true distribution *and* the size (k) of our sample

The Basic Idea

- Treat the actual distribution as a proxy for the true distribution.
- Sample with replacement your actual distribution *N* times.
- Compute the statistic of interest on each "re-sample".



- $\{\overline{Y}^*\}$ constitutes an estimate of the *distribution* of $\overline{Y}$.

Procedure for bootstrapping

Let the original sample be $x=(x_1,x_2,\ldots,x_n)$

Repeat B times

Generate a sample x* of size n from x by sampling with replacement.

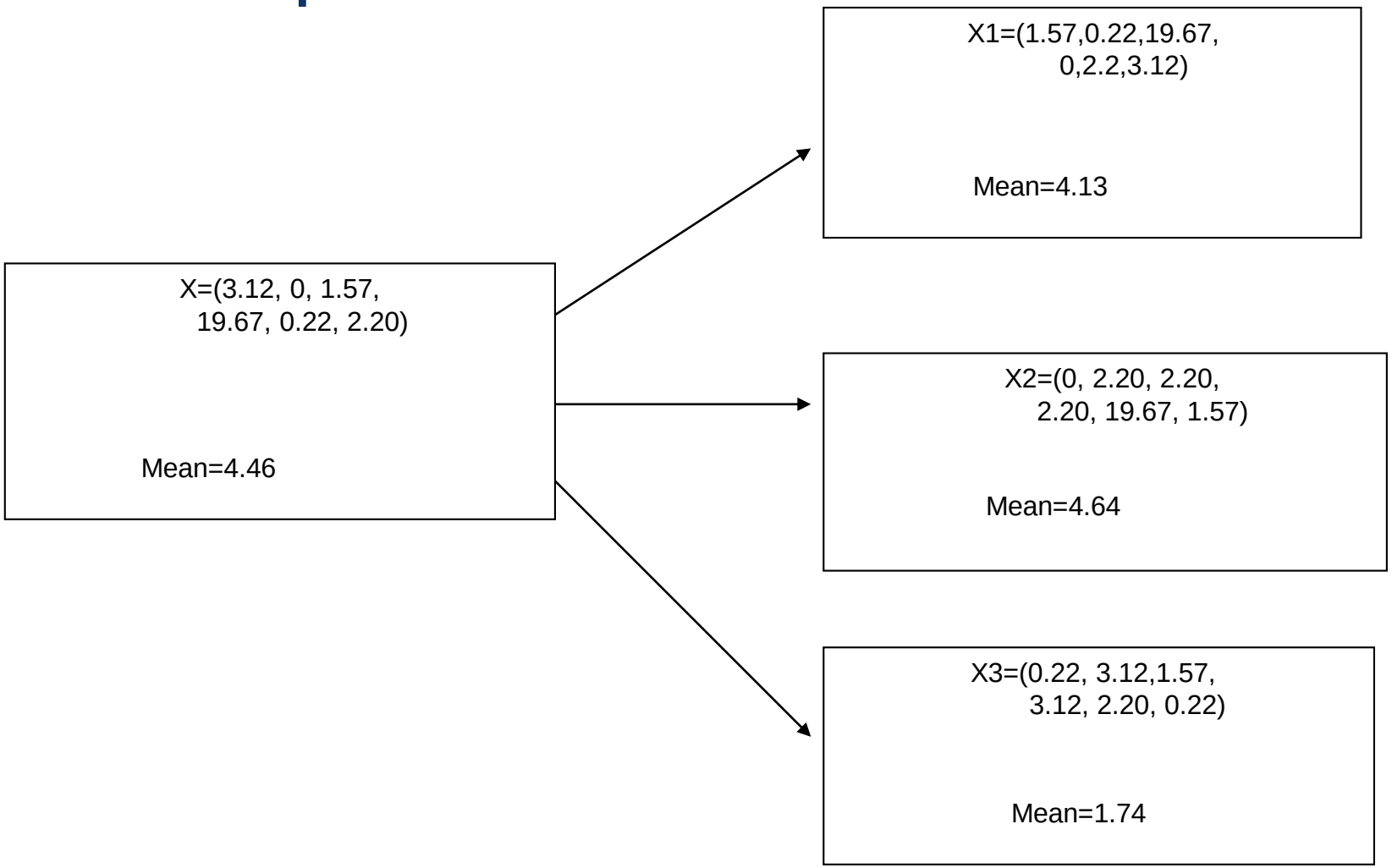
Compute $\hat{\theta}*$ for x*.

➔ ***Now we end up with bootstrap values***

$$\hat{\theta}* = (\hat{\theta}_1^*,\ldots,\hat{\theta}_B^*)$$

Use these values for calculating alll the quantities of interest (e.g., standard deviation, confidence intervals)

An example


X=(3.12, 0, 1.57,
19.67, 0.22, 2.20)
Mean=4.46
X1=(1.57,0.22,19.67,
0,2.2,3.12)
Mean=4.13
X2=(0, 2.20, 2.20,
2.20, 19.67, 1.57)
Mean=4.64
X3=(0.22, 3.12,1.57,
3.12, 2.20, 0.22)
Mean=1.74

Cases where bootstrap does not apply

- Small data sets: the original sample is not a good approximation of the population

- Dirty data: outliers add variability in our estimates.

- Dependence structures (e.g., time series, spatial problems): Bootstrap is based on the assumption of independence.

How many bootstrap samples are needed?

Choice of B depends on

- Computer availability
- Type of the problem: standard errors, confidence intervals, …
- Complexity of the problem

The JACKNIFE

The Jackknife

- Data **D** = {X_1, X_2, X_3, ... ,X_n} => statistic s
- *Jackknife replicates* miss out units (or groups of units) in turn:

  J1 = X_2, X_3, ... ,X_n => statistic s_{-1} (missing unit 1)

  J2 = X_1, X_3, ... ,X_n => statistic s_{-2} (missing unit 2)

  etc.

- Convert into pseudovalues:

  $\varphi_1 = s_{-1}$

  $\varphi_2 = s_{-2}$

  Etc.

The Jackknife

- The Jackknifed Estimate of s is then:

  $s_J = \text{mean}(\varphi_1,...,\varphi_n)$

- Estimate of Bias:

  $(n-1)\,(s_J - s\,)$

- $SE(s) = SE(\varphi_1,...,\varphi_n)$

- Another methods:
- Convert into pseudovalues:

  $\varphi_1 = n \cdot s - (n-1)s_{-1}$

  $\varphi_2 = n \cdot s - (n-1)s_{-2}$

  Etc.

The Jackknife

- Jackknifed Estimate removes bias
- Jackknife SE “rough and ready”

  usually “conservative” (overestimates SE)
- Jackknife on blocks of units, if data not independent
- Assumes normality for confidence intervals
- Fails when s is not “smooth”

Bootstraps

"Better" estimate of confidence

Variable n

Self-comparisons a problem

e.g. Mean of associations

Gives SE's, confidence intervals and profile of confidence

Jackknives

"Worse" estimate of confidence

Usually conservative

- underestimates precision

Fixed n

Self-comparisons not a problem

Reduces Bias

Only directly gives SE

Confidence intervals need assumption of normality

Bootstraps and Jackknives

- Give estimates of confidence (and bias) when:

  distributions unknown, approximate, or intractable

- Non-parametric bootstrap

  widely applicable (except self-referencing situations)

  few assumptions

- Jackknife

  approximate

  only standard error given directly

  useful when bootstrap not applicable

Cross validation

- Data partitioning method
- The jackknife could be considered a special case
- Another version: “k-fold” cross validation

Prosedur

- Split data menjadi k bagian

  Untuk bagian 1 – (k-1) gunakan untuk menduga model

  Bagian ke-k digunakan untuk menduga respons

  Hitung galat prediksi: cv(-k) = 1/nk sqrt((sum(yhat-y)))

  Ulangi sampai setiap bagian pernah digunakan untuk menduga galat prediksi

  Hitung RMSEP = mean(cv(-1),cv(-2), …, cv(-k))

Selesai...