محاسبات آماری پیشرفته

ترم اول سال تحصیلی ۹۳

جلسه نهم: شبیه‌سازی و استنباط آماری

حسین باغیشنی

دانشگاه شاهرود

۲۳ آبان ۱۳۹۳

چرا شبیه‌سازی؟

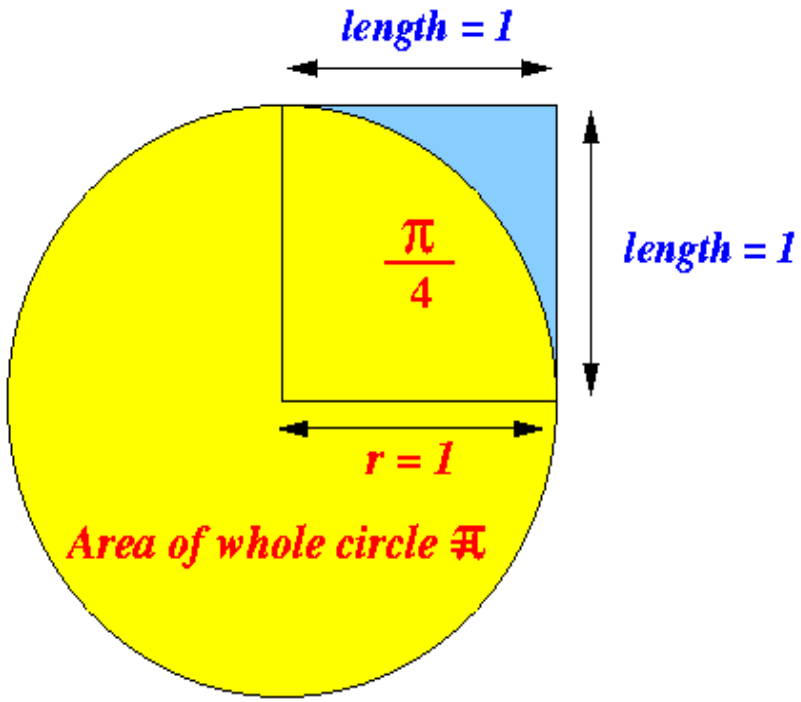
نیاز به تولید شانس در یک کامپیوتر برای:

- ارزیابی رفتار یک سیستم پیچیده (مانند یک شبکه، عملیات اقتصادی، بازی‌های کامپیوتری، خط تولید یک کارخانه و غیره)
- شناسایی ویژگی‌های احتمالاتی یک روش آماری جدید
- شناسایی ویژگی‌های یک روش آماری تحت توزیع نامعلوم (بوت‌استرپ)
- ارزیابی درستی یک مدل آماری
- محاسبه یک انتگرال
- ...

در این جلسه به قسمتی از کاربردهای روش‌های شبیه‌سازی مونت کارلو در استنباط آماری می‌پردازیم

تقریب π: طراحی شبیه‌سازی

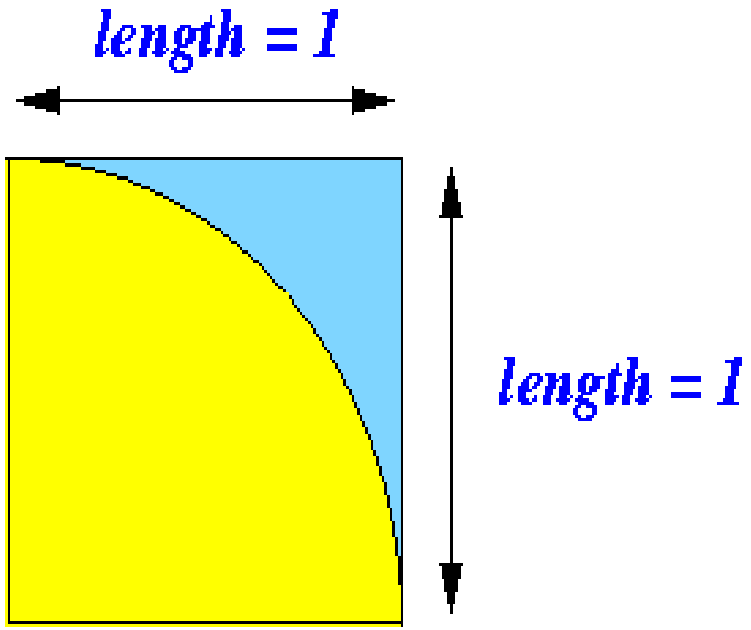
فرض کنید در مورد مقدار $\pi = ۳/۱۴۱۵۹۲۶۵۳۵...$ اطلاعی نداشته باشیم. یک روش برای تقریب این عدد، استفاده از مساحت دایره است:



تقریب π: طراحی شبیه‌سازی

اگر ربع دایره را در نظر بگیریم، نقطه‌ای مانند (x, y) درون آن در نامساوی زیر صدق می‌کند:

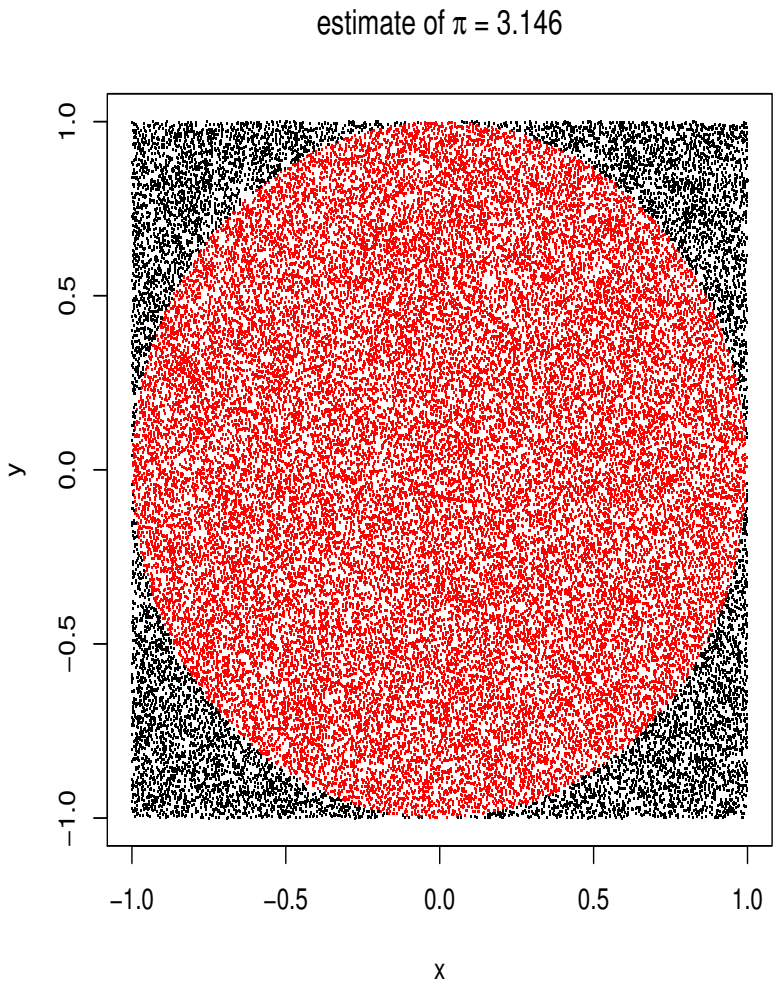
$$x^۲ + y^۲ \leq ۱.$$



تقریب π

```
n <- 50000
 x <- runif(n, -1, 1)
 y <- runif(n, -1, 1)
 isInCircle <- (x^2 + y^2 <= 1)
 est <- 4 * sum(isInCircle) / n

plot(x, y, type = "n")
 points(x[isInCircle], y[isInCircle],
 col = "red", pch = ".")
 points(x[!isInCircle], y[!isInCircle],
 col = "black", pch = ".")
 title(substitute(paste("estimate of ", pi,
 " = ", x), list(x = est)))
```



احتمال بلد نیستی؟ شبیه‌سازی چطور؟

مساله انطباق (روز تولد): احتمال آن‌که روز تولد حداقل دو دانشجوی کلاسی با n نفر یکی باشد، چقدر است؟

احتمال بلد نیستی؟ شبیه‌سازی چطور؟

مساله انطباق (روز تولد): احتمال آن‌که روز تولد حداقل دو دانشجوی کلاسی با n نفر یکی باشد، چقدر است؟

$$\begin{aligned} P(\text{حداقل تولد یکسان برای دو نفر}) &= 1 - P(\text{روز تولد هیچکس یکسان نباشد}) \\ &= 1 - \frac{365 \times 364 \times \cdots \times [365 - (n-1)]}{365^n}. \end{aligned}$$

مثلا برای $n = ۶$،

$$۱ - ((۳۶۵ \times ۳۶۴ \times ۳۶۳ \times ۳۶۲ \times ۳۶۱ \times ۳۶۰)/(۳۶۵^۶)) = ۰/۰۴۰۴۶$$

احتمال بلد نیستی؟ شبیه‌سازی چطور؟

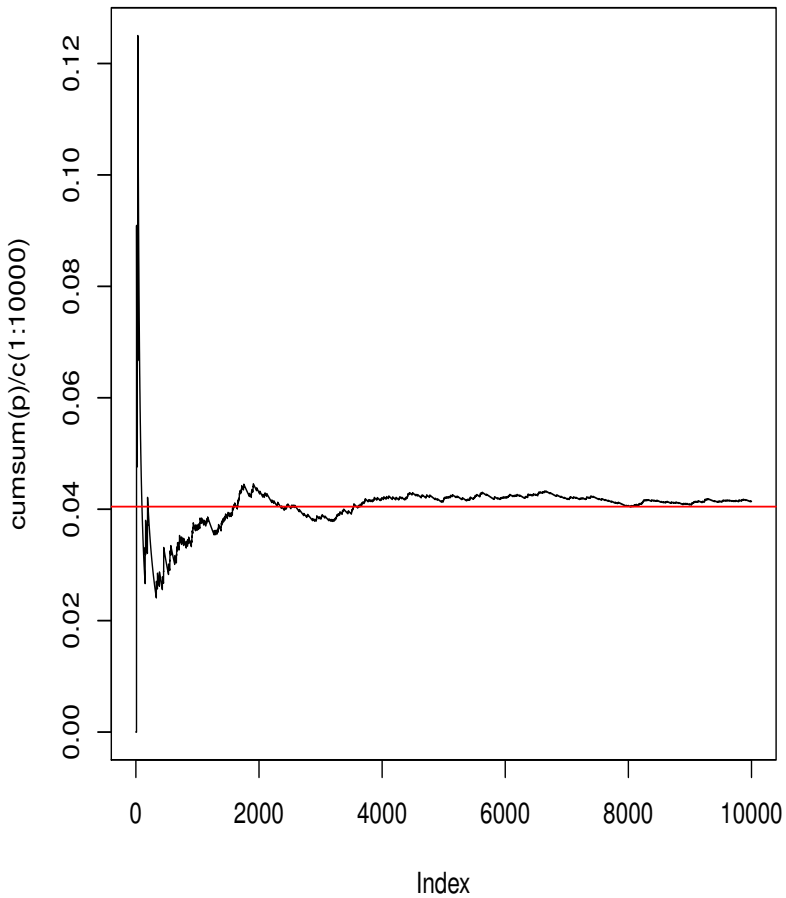
فرض کنید راه حل پاسخ را ندانید !

با یک شبیه‌سازی ساده می‌توان به یک جواب تقریبی (نزدیک) دست یافت:

```
count = 0
for (i in 1:10^6) {
    bdays = sample(1:365, 6, replace = T)
    if (length(unique(bdays)) < 6)
        count = count + 1
}
> count
[1] 40754
> p = count/(10^6)
> p
[1] 0.040754
```

نمایش همگرایی به مقدار واقعی

```
p = rep(0, 10000)
for (i in 1:10000) {
    bdays = sample(1:365, 6, replace = T)
    if (length(unique(bdays)) < 6)
        p[i] = 1
}
#
plot(cumsum(p)/c(1:10000), type = "l")
abline(h = 0.04046248, col = "red")
```



شانس داشتن دقیقا دو نفر از بین ۸ نفر با روز تولد یکسان؟

```
count2 = 0
for (i in 1:10^6) {
    bdays2 = sample(1:365, 8, replace = T)
    if (length(unique(bdays2)) == 7)
        count2 = count2 + 1
}

p2 = count2/(10^6)
> p2
[1] 0.072501
```

تمرین. مقدار نظری این احتمال را محاسبه کنید.

شانس داشتن دقیقا دو نفر از بین ۸ نفر با روز تولد یکسان؟

```
count2 = 0
for (i in 1:10^6) {
    bdays2 = sample(1:365, 8, replace = T)
    if (length(unique(bdays2)) == 7)
        count2 = count2 + 1
}

p2 = count2/(10^6)
> p2
[1] 0.072501
```

تمرین. مقدار نظری این احتمال را محاسبه کنید.

تمرین. برای $n = ۸$ احتمال این‌که حداقل ۲ نفر روزهای تولدی با یک روز اختلاف داشته باشند؟

جواب تمرین دوم اسلاید قبلی

```
count3 = 0
for (i in 1:10^6) {
  bdays3 = sample(1:365, 8, replace = T)
  bdays3 = sort(bdays3)
  if (min(bdays3[2:8] - bdays3[1:7]) == 1)
       count3 = count3 + 1
}
count3
[1] 134238
p3 = count3/(10^6)
p3
[1] 0.134238
```

مسایل ساده تا پیچیده

شبیه‌سازی، می‌تواند ابزاری برای استفاده در مسایل مختلف با درجه‌های پیچیدگی متفاوت باشد

استنباط آماری و شبیه‌سازی

مجموعه روش‌های استنباطی مورد نظر، عبارتند از:

- برآورد پارامترهای توزیع نمونه‌ای یک آماره، MSE، چندک‌ها و غیره
- محاسبه احتمال پوشش یک فاصله اطمینان به منظور مقایسه با سطح اطمینان اسمی
- محاسبه نرخ خطای نوع اول یک آزمون آماری
- برآورد توان یک آزمون
- مقایسه عملکرد روش‌های مختلف برای یک مساله مفروض

عدم قطعیت

برآوردها همیشه همراه با عدم قطعیت هستند.

علاقه به محاسبه توزیع‌های نمونه‌ای برآوردگرها، ناشی از تحقیق در مورد این عدم قطعیت است

اگر بتوان فرآیند تصادفی (مکانیسم مرجع) که داده‌ها از آن تولید شده‌اند را شبیه‌سازی کرد، می‌توان با تولید مکرر نمونه از آن به مطالعه رفتار فرآیند و جستجوی کمیت‌های مورد نظر پرداخت.

تولید نمونه از یک مدل آماری (فرآیند تصادفی) مشخص را بوت‌استرپ پارامتری نیز می‌گویند.

برآوردیابی مونت کارلویی

فرض کنید $X_1, \ldots, X_n$ یک نمونه تصادفی از توزیع متغیر تصادفی X باشد.

$$\hat{\theta} = \hat{\theta}(X_1, \ldots, X_n),$$

به عنوان برآوردی از θ تابعی از نمونه تصادفی است.

اگر نمونه‌های مختلفی از X داشته باشیم، دنباله‌ای از برآوردگرها نیز خواهیم داشت که بر اساس آن می‌توان توزیع برآوردگر را با محاسبه توزیع تجربی، برآورد کرد.

مثلا فرض کنید $X = (X_1, \ldots, X_n)$، همچنین فرض کنید $X^{(1)}, X^{(2)}, \ldots$ دنباله‌ای از نمونه‌های تصادفی تولیدشده از توزیع X باشند. بنابراین

$$\hat{\theta}^{(j)} = \hat{\theta}(X_1^{(j)}, \ldots, X_n^{(j)}), \; j = 1, 2, \ldots,$$

یک دنباله از برآوردگرهای θ می‌باشند. با این دنباله (مثلا m تایی) می‌توان کمیت‌های مورد نظر (توزیع نمونه‌ای $\hat{\theta}$) را برآورد کرد.

مثال

$$X_۱, X_۲ \sim N(۰, ۱).$$

$$\mathbb{E}|X_۱ - X_۲| = \theta = ?$$

با تولید m نمونه $x^{(j)} = (x_۱^{(j)}, x_۲^{(j)}),\ j = ۱, \ldots, m$ می‌توان m برآورد θ را محاسبه کرد:

$$\hat{\theta}^{(j)} = |x_۱^{(j)} - x_۲^{(j)}|,\ j = ۱, \ldots, m,$$

اکنون می‌توان از کمیت‌های نمونه‌ای برای برآورد کمیت‌های دلخواه استفاده کرد:

$$\hat{\theta} = \frac{۱}{m}\sum_{j=۱}^{m}\hat{\theta}^{(j)} = \frac{۱}{m}\sum_{j=۱}^{m}|x_۱^{(j)} - x_۲^{(j)}|.$$

$$\hat{se}(\hat{\theta}) = \frac{۱}{\sqrt{m}}\left\{\frac{۱}{m-۱}\sum_{j=۱}^{m}(\hat{\theta}^{(j)} - \hat{\theta})^۲\right\}^{۱/۲}$$

```
m <- 1000
g <- numeric(m)
for (i in 1:m) {
      x <- rnorm(2)
      g[i] <- abs(x[1] - x[2])
}
est <- mean(g)
> est
[1] 1.130063
> sqrt(sum((g-mean(g))^2))/m
[1] 0.02741058
```

تمرین نشان دهید در این مثال $\mathbb{E}|X_1 - X_2| = \frac{2}{\sqrt{\pi}} \approx 1/12837$ و

$$Var(|X_1 - X_2|) = 2 - \frac{4}{\pi}$$

$$se(\hat{\theta}) = \sqrt{\frac{2 - 4/\pi}{m}} \approx 0/02696$$

برآورد MSE

$$MSE(\hat{\theta}) = \mathbb{E}\left[(\hat{\theta} - \theta)^{۲}\right].$$

$$\hat{MSE} = \frac{۱}{m}\sum_{j=۱}^{m}(\hat{\theta}^{(j)} - \theta)^{۲}$$

تذکر: برای داده‌های واقعی که پارامتر نامعلوم است، محاسبه این کمیت ناممکن است!!

مثال: میانگین پیراسته

میانگین پیراسته در مواردی که داده‌های پرت وجود دارند، به عنوان برآوردگری تنومند مورد استفاده قرار می‌گیرد.
برای یک نمونه n تایی، فرض کنید $X_{(1)}, \ldots, X_{(n)}$ آماره‌های مرتب نمونه باشند. میانگین پیراسته نمونه، میانگین تمام داده‌ها به جز کوچکترین و بزرگترین مشاهدات است.

$$\bar{X}_{[-k]} = \frac{1}{n-2k} \sum_{j=k+1}^{n-k} X_{(j)}.$$

در این مثال برای یک توزیع نرمال استاندارد، به دنبال برآورد MSE میانگین پیراسته سطح اول ($k = 1$) هستیم.

در مثال قبلی (برای توزیع نرمال) پارامتر مورد نظر عبارتست از $\theta = \mathbb{E}(\bar{X}_{[-1]}) = 0$.

مثال: ادامه

اگر میانگین پیراسته نمونه را با T نشان دهیم، برای برآورد $MSE(T)$ مراحل زیر را باید اجرا کنیم:

۱. تولید m برآورد $T^{(j)},\ j = ۱, \ldots, m$:
   - تولید $x_۱^{(j)}, \ldots, x_n^{(j)}$ از توزیع X.
   - مرتب کردن نمونه‌ها و به دست آوردن $x_{(۱)}^{(j)} \leq \ldots x_{(n)}^{(j)}$
   - محاسبه $T^{(j)} = \frac{۱}{n-۲} \sum_{i=۲}^{n-۲} x_{(i)}^{(j)}$

۲. محاسبه $\hat{MSE}(T) = \frac{۱}{m} \sum_{j=۱}^{m} (T^{(j)} - \theta)^۲ = \frac{۱}{m} \sum_{j=۱}^{m} (T^{(j)})^۲$.

دقت کنید که مثل قبل، $T^{(۱)}, \ldots, T^{(m)}$ نمونه‌ای از توزیع نمونه‌ای میانگین پیراسته سطح اول برای یک توزیع نرمال است و می‌توان کمیت‌های مورد علاقه را بر اساس آن‌ها برآورد کرد.

```
n <- 20
m <- 1000
tmean <- numeric(m)
for (i in 1:m) {
     x <- sort(rnorm(n))
     tmean[i] <- sum(x[2:(n-1)]) / (n-2)
}
mse <- mean(tmean^2)
> mse
[1] 0.05571728
> sqrt(sum((tmean - mean(tmean))^2)) / m    #se
[1] 0.00745855
```

میانه

برای میانه هم می‌توان به همین روش عمل کرد:

```
n <- 20
m <- 1000
tmean <- numeric(m)
for (i in 1:m) {
      x <- sort(rnorm(n))
      tmean[i] <- median(x)
}
mse <- mean(tmean^2)
> mse
[1] 0.07431353
> sqrt(sum((tmean - mean(tmean))^2)) / m    #se
[1] 0.008613994
```

دقت کنید که میانه هم در واقع یک میانگین پیراسته است به طوری که همه داده‌ها به جز یک یا دو تا از آن‌ها را (وسط داده‌ها) را پیرایش می‌کند

نرمال آلوده

در این مثال می‌خواهیم MSE را برای میانگین پیراسته سطح k در دو خانواده نرمال و نرمال آلوده، مقایسه کنیم.

یک نرمال آلوده در واقع یک آمیخته از دو توزیع نرمال است. مثلاً:

$$pN(۰, \sigma^۲ = ۱) + (۱ - p)N(۰, \sigma^۲ = ۱۰۰)$$

در این جا پارامتر مورد نظر $\theta = ۰$ است.

دقت کنید که برای تولید از توزیع نرمال آلوده، مثل مبحث توزیع‌های آمیخته، باید ابتدا بر حسب توزیع احتمال $P(\sigma = ۱) = p;\ P(\sigma = ۱۰) = ۱ - p$ مقدار ۱ یا ۱۰ برای σ انتخاب شود و سپس از توزیع نرمال متناظر با آن σ نمونه تولید شود.

```
set.seed(522)
n <- 20
K <- n/2 - 1
m <- 1000
mse <- matrix(0, n/2, 6)
trimmed.mse <- function(n, m, k, p) {
# MC est of mse for k-level trimmed mean of contaminated normal pN(0,1) + (1-p)N(0,100)
tmean <- numeric(m)
for (i in 1:m) {
        sigma <- sample(c(1, 10), size = n,
        replace = TRUE, prob = c(p, 1-p))
        x <- sort(rnorm(n, 0, sigma))
        tmean[i] <- sum(x[(k+1):(n-k)]) / (n-2*k)
     }
    mse.est <- mean(tmean^2)
    se.mse <- sqrt(mean((tmean-mean(tmean))^2)) / sqrt(m)
    return(c(mse.est, se.mse))
}
for (k in 0:K) {
     mse[k+1, 1:2] <- trimmed.mse(n=n, m=m, k=k, p=1.0)
     mse[k+1, 3:4] <- trimmed.mse(n=n, m=m, k=k, p=.95)
     mse[k+1, 5:6] <- trimmed.mse(n=n, m=m, k=k, p=.9)
}
> n*mse
           [,1]      [,2]     [,3]      [,4]      [,5]      [,6]
 [1,] 0.9758575 0.1396680 6.229283 0.3528182 11.484562 0.4792567
 [2,] 1.0194569 0.1425744 1.954048 0.1976301  4.125681 0.2871660
 [3,] 1.0091575 0.1420550 1.304154 0.1614779  1.956211 0.1975056
 [4,] 1.0806823 0.1470113 1.168253 0.1528290  1.577817 0.1776404
 [5,] 1.0478085 0.1446125 1.279664 0.1599728  1.452725 0.1700661
 [6,] 1.1028720 0.1485146 1.395142 0.1670407  1.422924 0.1686735
 [7,] 1.3157601 0.1621938 1.348891 0.1642456  1.574427 0.1773341
 [8,] 1.3767837 0.1659384 1.503163 0.1733524  1.734379 0.1862460
 [9,] 1.3815624 0.1662126 1.525009 0.1746251  1.693839 0.1840468
[10,] 1.4907734 0.1720630 1.646402 0.1814599  1.843068 0.1916661
```

برآورد سطح اطمینان

فاصله‌های اطمینان بر حسب آماره‌هایی به دست می‌آیند که معمولاً توزیع نمونه‌ای آن‌ها نامعلوم هستند یا به دست آوردن آن‌ها کار بسیار دشواری است.

به عنوان مثال بسیاری از روش‌های برآورد معمول، بر اساس پذیره نرمال بودن توزیع جامعه ساخته می‌شوند، در حالی که در عمل موارد متعددی پیش می‌آیند که در آن‌ها توزیع جامعه غیرنرمال باشد.

در چنین مواردی ممکن است یافتن توزیع نمونه‌ای برآوردگرها ناممکن باشد.

در نتیجه فواصل اطمینانی که بر حسب توزیع نرمال جامعه ساخته می‌شوند، ممکن است سطح اطمینان تجربی آن‌ها با سطح اطمینان اسمی یکی نباشد.

با روش‌های مونت کارلو می‌توان سطح اطمینان این فواصل را برآورد کرد.

سطح اطمینان

فرض کنید (U, V) یک برآورد فاصله‌ای برای θ باشد. U و V آماره‌هایی هستند که توزیع نمونه‌ای آن‌ها به توزیع جامعه F_X وابسته است.

سطح اطمینان، احتمالی است که فاصله تصادفی (U, V) مقدار واقعی θ را در بر می‌گیرد. بنابراین محاسبه سطح اطمینان یک فاصله اطمینان، یک مساله انتگرال‌گیری است.

فرض کنید بخواهیم یک برآورد فاصله‌ای برای واریانس به دست آوریم.

مشهور است که چنین برآوردی به پذیره توزیع نرمال برای جامعه خیلی حساس است. یعنی تخطی از توزیع نرمال می‌تواند نتایج به شدت گمراه‌کننده‌ای در بر داشته باشد.

از نمونه‌گیری مونت کارلو می‌توان برای برآورد سطح اطمینان واقعی یک فاصله اطمینان با فرض توزیع نرمال جامعه وقتی که توزیع جامعه واقعا نرمال نیست، استفاده کرد.

فاصله اطمینان برای واریانس

ابتدا بر اساس توزیع واقعی نرمال مساله را بررسی می‌کنیم.

$$X_1, \ldots, X_n \sim N(\mu, \sigma^2)$$

$$V = \frac{(n-1)S^2}{\sigma^2} \sim \chi^2(n-1).$$

یک فاصله اطمینان یک طرفه در سطح $100(1-\alpha)\%$ عبارتست از $(0, \frac{(n-1)S^2}{\chi^2_\alpha})$ که در آن χ^2_α چندک αام توزیع $\chi^2(n-1)$ است.

اگر توزیع جامعه نرمال با واریانس σ^2 باشد، آنگاه احتمال آنکه فاصله مقدار واقعی σ^2 را پوشش دهد برابر $1-\alpha$ است.

فاصله اطمینان برای واریانس: ادامه

برآورد کران بالای فاصله برای $N(۰, \sigma^۲ = ۴)$ به صورت زیر قابل انجام است:

```
n <- 20
alpha <- .05
x <- rnorm(n, mean=0, sd=2)
UCL <- (n-1) * var(x) / qchisq(alpha, df=n-1)
```

در این مثال، وقتی که توزیع جامعه واقعا نرمال است، سطح اطمینان به طور دقیق قابل اندازه‌گیری است:

$$P(\frac{۱۹S^۲}{\chi^۲_{۰/۰۵}(۱۹)} > ۴) = P(\frac{(n-۱)S^۲}{\sigma^۲} > \chi^۲_{۰/۰۵}(n-۱)) = ۰/۹۵.$$

اگر تولید نمونه و محاسبه این فاصله به دفعات زیادی تکرار شود، تقریباً ٪۹۵ فاصله‌های عددی ساخته شده باید مقدار واقعی $\sigma^۲ = ۴$ را در بر داشته باشند.

برآورد سطح اطمینان تجربی

فرض کنید $X \sim F_X$ و θ پارامتر مورد نظر باشد. برای برآورد سطح اطمینان باید مراحل زیر اجرا شوند:

۱. برای هر تکرار مونت کارلو، $j = ۱, \ldots, m$:
   - نمونه $X_۱^{(j)}, \ldots, X_n^{(j)}$ را تولید کن
   - فاصله اطمینان C_j را محاسبه کن
   - متغیر نشانگر $y_j = I(\theta \in C_j)$ را محاسبه کن

۲. سطح اطمینان تجربی با کمیت $\bar{y} = \frac{۱}{m} \sum_{j=۱}^{m} y_j$ محاسبه می‌شود

برآوردگر $\bar{y}$ نسبت نمونه‌ای برای برآورد سطح اطمینان واقعی $۱ - \alpha^*$ است. بنابراین چون y_j ها متغیرهای برنولی هستند، $Var(\bar{y}) = \frac{(۱-\alpha^*)\alpha^*}{m}$. در نتیجه برآوردی برای خطای معیار برآورد نیز عبارتست از

$$\hat{se}(\bar{y}) = \sqrt{\frac{(۱ - \bar{y})\bar{y}}{m}}.$$

برآورد مونت کارلو در مثال واریانس توزیع نرمال

```
n <- 20
alpha <- .05
UCL <- replicate(1000, expr = {
       x <- rnorm(n, mean = 0, sd = 2)
       (n-1) * var(x) / qchisq(alpha, df = n-1)
       } )
# compute the mean to get the confidence level
> mean(UCL > 4)
[1] 0.95
```

تمرین. در مورد دستور *replicate* جزییات را همراه با مثال تشریح کنید

تخطی از پذیره نرمال

برآورد فاصله‌ای واریانس نسبت به پذیره نرمال بودن توزیع جامعه حساس است. به این معنی که اگر توزیع جامعه نرمال نباشد، سطح اطمینان واقعی با مقدار اسمی آن اختلاف دارد.

سطح اطمینان واقعی به توزیع آماره S^2 بستگی دارد.

$$P(\frac{(n-1)S^2}{\chi^2_\alpha} > \sigma^2) = P(S^2 > \frac{\sigma^2\chi^2_\alpha}{n-1}) = 1 - G(\frac{\sigma^2\chi^2_\alpha}{n-1}),$$

که در آن G تابع توزیع آماره S^2 است.
اگر توزیع واقعی جامعه نرمال نباشد، مساله برآورد سطح اطمینان به یک مساله انتگرال‌گیری منتهی می‌شود:

$$G(t) = P(S^2 \leq c_\alpha) = \int_0^{c_\alpha} g(x)dx,$$

به طوری که $g(\cdot)$ تابع چگالی (نامعلوم) S^2 و $c_\alpha = \frac{\sigma^2\chi^2_\alpha}{n-1}$.

سطح اطمینان تجربی

توجه داشته باشید که در این جا برای محاسبه مونت کارلو انتگرال بالا، شرط معلوم بودن ضابطه $g(\cdot)$ لازم نیست و تنها کافی است بتوان از این توزیع نمونه‌هایی تولید کرد

به عنوان مثال فرض کنید توزیع واقعی جامعه $\chi^2(2)$ باشد (واریانس جامعه ۴ است اما توزیع نرمال نیست).

```
n <- 20
alpha <- .05
UCL <- replicate(1000, expr = {
       x <- rchisq(n, df = 2)
       (n-1) * var(x) / qchisq(alpha, df = n-1)
       } )
> mean(UCL > 4)
[1] 0.777
```

آزمون فرضیه‌ها

$$H_0 : \theta \in \Theta_0 \quad vs \quad H_1 : \theta \in \Theta_1$$

$$\Theta_0 \cup \Theta_1 = \Theta, \quad \Theta_0 \cap \Theta_1 = \emptyset$$

سطح معنی‌داری یک آزمون را با α نشان می‌دهند که کران بالا برای احتمال رخداد خطای نوع اول است.

تابع توان یک آزمون، $\pi(\theta)$، احتمال رد فرضیه H_0 است. بنابراین

$$\alpha = \sup_{\theta \in \Theta_0} \pi(\theta).$$

اگر T آماره آزمون و T^* مقدار مشاهده شده این آماره باشد، آنگاه T^* معنی‌دار گفته می‌شود اگر تصمیم مبتنی بر آن منجر به رد فرضیه H_0 شود.

احتمال معنی‌داری یا همان pـ مقدار، کوچکترین مقدار ممکن α است به طوری که آماره آزمون مشاهده شده، معنی‌دار باشد.

استفاده از pـ مقدار باید با ملاحظه صورت گیرد!!!

نرخ خطای نوع اول تجربی

احتمال خطای نوع اول، احتمال شرطی رد فرضیه H_0 است زمانی که این فرضیه درست باشد.

بنابراین اگر آزمون تحت شرایط فرضیه صفر، H_0، به دفعات تکرار شود، نرخ خطای نوع اول مشاهده شده باید تقریباً حداکثر α باشد.

نرخ خطای نوع اول تجربی، در یک شبیه‌سازی مونت کارلو، نسبت آماره‌های آزمون معنی‌دار در نمونه‌های مونت کارلوی تولید شده است.

مراحل برآورد نرخ خطای نوع اول تجربی به صورت زیر است:

۱. برای هر تکرار مونت کارلو $j = 1, \ldots, m$:
   - نمونه $x_1^{(j)}, \ldots, x_n^{(j)}$ را تحت فرضیه صفر تولید کن
   - آماره آزمون T_j را محاسبه کن
   - اگر فرضیه H_0 رد می‌شود، قرار بده $I_j = 1$ در غیر این صورت قرار بده $I_j = 0$

۲. نسبت آزمون‌های معنی‌دار را با کمیت $\frac{1}{m}\sum_{j=1}^{m} I_j$ محاسبه کن

خطای استاندارد برآورد

پارامتر برآورد شده، یعنی نرخ خطای نوع اول تجربی، احتمالی است که با نسبت نمونه، $\hat{p}$، برآورد می‌شود. بنابراین برآوردی برای خطای استاندارد عبارتست از

$$\hat{se}(\hat{p}) = \sqrt{\frac{\hat{p}(۱ - \hat{p})}{m}} \leq \frac{۰/۵}{\sqrt{m}}.$$

مثال: آزمون t

$$X_۱, \ldots, X_{۲۰} \sim N(\mu, \sigma^۲),$$

$$H_۰ : \mu = ۵۰۰ \quad vs \quad H_۱ : \mu > ۵۰۰ \quad \& \quad \alpha = ۰/۰۵$$

برای آزمون این فرضیه‌ها از آماره t استفاده می‌شود:

$$T^* = \frac{\bar{X} - ۵۰۰}{S/\sqrt{۲۰}} \sim t(۱۹).$$

مقادیر بزرگ T^* از فرضیه جانشین $H_۱$ پشتیبانی می‌کنند.

در این مثال، از نمونه‌های مونت کارلو برای برآورد نرخ خطای نوع اول وقتی که $\sigma = ۱۰۰$ است، استفاده می‌کنیم.

برآورد خطای استاندارد برآورد نرخ خطای نوع اول باید نزدیک به $\sqrt{\frac{۰/۰۵ \times ۰/۹۵}{m}} \approx ۰/۰۰۲۲$ باشد.

مثال: ادامه

```
n <- 20
alpha <- .05
mu0 <- 500
sigma <- 100
m <- 10000  # number of replicates
p <- numeric(m)  # storage for p-values
for (j in 1:m) {
     x <- rnorm(n, mu0, sigma)
     ttest <- t.test(x, alternative = "greater", mu = mu0)
     p[j] <- ttest$p.value
}
p.hat <- mean(p < alpha)
se.hat <- sqrt(p.hat * (1 - p.hat) / m)
> print(c(p.hat, se.hat))
[1] 0.055700000 0.002293415
```

مثال: آزمون نرمال بودن بر اساس چولگی

یکی از آزمون‌های بررسی نرمال بودن یک نمونه، مبتنی بر پارامتر چولگی است:

$$\gamma_1 = \frac{\mathbb{E}[(X-\mu_X)^3]}{\sigma_X^3},$$

یک توزیع، متقارن است اگر برای آن $\gamma_1 = 0$. ضریب چولگی نمونه به صورت زیر به دست می‌آید:

$$b_1 = \frac{\frac{1}{n}\sum_{i=1}^{n}(X_i-\bar{X})^3}{(\frac{1}{n}\sum_{i=1}^{n}(X_i-\bar{X})^2)^{3/2}}.$$

نشان داده شده است اگر توزیع X نرمال باشد، آنگاه توزیع مجانبی b_1 نیز نرمال با میانگین صفر و واریانس $\frac{6}{n}$ خواهد بود.

در آزمون فرضیه‌های

$$H_0 : \gamma_1 = 0 \quad vs \quad H_1 : \gamma_1 \neq 0,$$

به ازای مقادیر بزرگ $|b_1|$، فرضیه نرمال بودن رد می‌شود.

مثال: ادامه

در این آزمون، توزیع نمونه‌ای آماره چولگی بر اساس پذیره نرمال بودن جامعه به دست می‌آید.

این توزیع نمونه‌ای تقریبی، برای حجم نمونه‌های کم و متوسط خوب نیست و سرعت همگرایی توزیع چولگی به نرمال کند است.

در این مثال، نرخ خطای نوع اول آزمون مبتنی بر چولگی را در سطح $\alpha = ۰/۰۵$ و برای نمونه‌های $n = ۱۰, ۲۰, ۳۰, ۵۰, ۱۰۰, ۵۰۰$، بر اساس توزیع مجانبی آماره چولگی، ارزیابی می‌کنیم.

مقادیر بحرانی آزمون برای این حجم‌های نمونه به صورت زیر محاسبه می‌شوند:

```
n <- c(10, 20, 30, 50, 100, 500) #sample sizes
cv <- qnorm(.975, 0, sqrt(6/n))  #crit. values for each n
> n
[1]  10  20  30  50 100 500
> cv
[1] 1.5181816 1.0735165 0.8765225 0.6789514 0.4800912 0.2147033
```

مثال: ادامه

```
sk <- function(x) {
      # computes the sample skewness coeff.
      xbar <- mean(x)
      m3 <- mean((x - xbar)^3)
      m2 <- mean((x - xbar)^2)
      return( m3 / m2^1.5 )
}
####
p.reject <- numeric(length(n)) #to store sim. results
m <- 10000
for (i in 1:length(n)) {
     sktests <- numeric(m)       #test decisions
     for (j in 1:m) {
          x <- rnorm(n[i])
          # test decision is 1 (reject) or 0
          sktests[j] <- as.integer(abs(sk(x)) >= cv[i] )
          }
     p.reject[i] <- mean(sktests) # proportion rejected
}
> p.reject
[1] 0.0141 0.0287 0.0358 0.0382 0.0406 0.0477
```

نتایج نشان می‌دهد، استفاده از تقریب نرمال برای توزیع آماره چولگی برای نمونه‌های $n \leq ۵۰$ مناسب نیست و برای نمونه‌های بزرگ، حتی $n = ۵۰۰$، هم مورد شک و تردید است.

مثال: ادامه

برای نمونه‌های کوچک باید از واریانس دقیق آماره استفاده کرد:

$$Var(b_1) = \frac{6(n-2)}{(n+1)(n+3)}.$$

```
cv <- qnorm(.975, 0, sqrt(6*(n-2)/((n+1)*(n+3))))
> round(cv, 4)
[1] 1.1355 0.9268 0.7943 0.6398 0.4660 0.2134
> n
[1]  10  20  30  50 100 500
>     p.reject
[1] 0.0540 0.0548 0.0504 0.0495 0.0491 0.0535
```

این برآوردها نزدیکتر به سطح اسمی ۰/۰۵ هستند.

توان آزمون

توان یک آزمون در تصمیم‌گیری درست، با تابع توان بیان می‌شود که احتمال رد H_0 به درستی می‌باشد.

البته تابع توان برای کل فضای پارامتر تعریف می‌شود:

$$\pi : \Theta \to [0, 1], \quad \pi(\theta) = P_\theta(RH_0).$$

بنابراین برای هر $\theta_1 \in \Theta_1$ احتمال رخداد خطای نوع دوم $1 - \pi(\theta_1)$ است.

آزمونی ترجیح داده می‌شود که دارای احتمال خطای کمتری باشد. خطای نوع اول که در سطح α کنترل شده است. بنابراین مقایسه آزمون‌ها برای فرضیه‌های یکسان و در سطح برابر، در واقع مقایسه توان آن‌ها در زیرفضای Θ_1 است.

اگر توان یک آزمون را به طور تحلیلی نتوان محاسبه کرد (که خیلی هم معمول است)، می‌توان از روش مونت کارلو آن را تقریب زد.

برآورد مونت کارلوی توان آزمون

برای محاسبه توان آزمون در یک نقطه از فضای Θ_1 باید به صورت زیر عمل کرد:

۱. انتخاب یک مقدار مشخص از فضای پارامتر تحت فرضیه جانشین، $\theta_1 \in \Theta_1$

۲. برای هر تکرار مونت کارلو $j = 1, \ldots, m$:

الف) تولید نمونه $x_1^{(j)}, \ldots, x_n^{(j)}$ تحت فرضیه جانشین $\theta = \theta_1$

ب) محاسبه آماره آزمون T_j

ج) ثبت تصمیم آزمون، یعنی اگر H_0 رد شود قرار بده $I_j = 1$ و در غیر این صورت قرار بده $I_j = 0$

۳. محاسبه نسبت آزمون‌های معنی‌دار یعنی $\hat{\pi}(\theta_1) = \frac{1}{m} \sum_{j=1}^{m} I_j$

توان تجربی

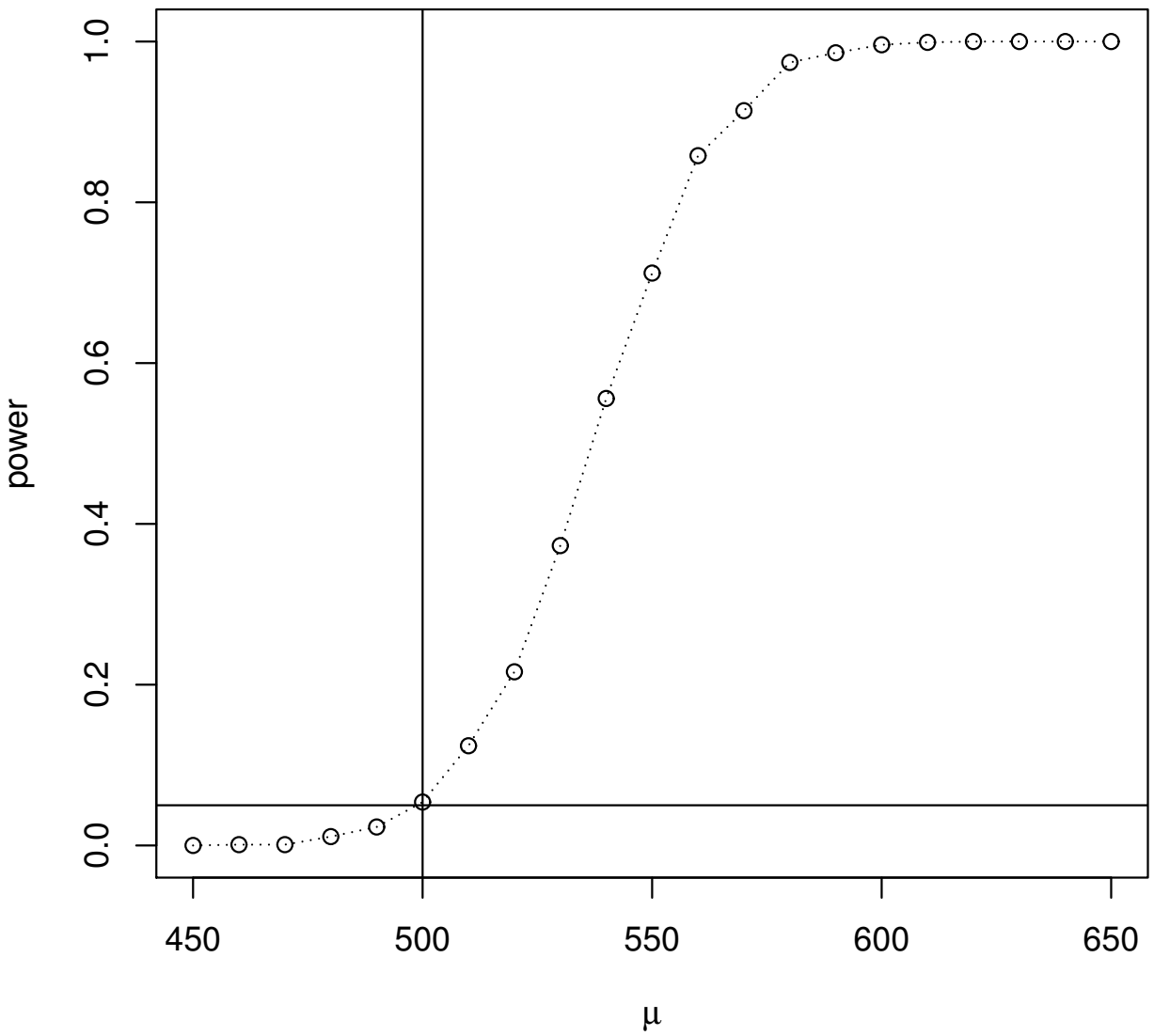
برای مثال آزمون t، توان آزمون را در چند نقطه از Θ_1، محاسبه می‌کنیم:

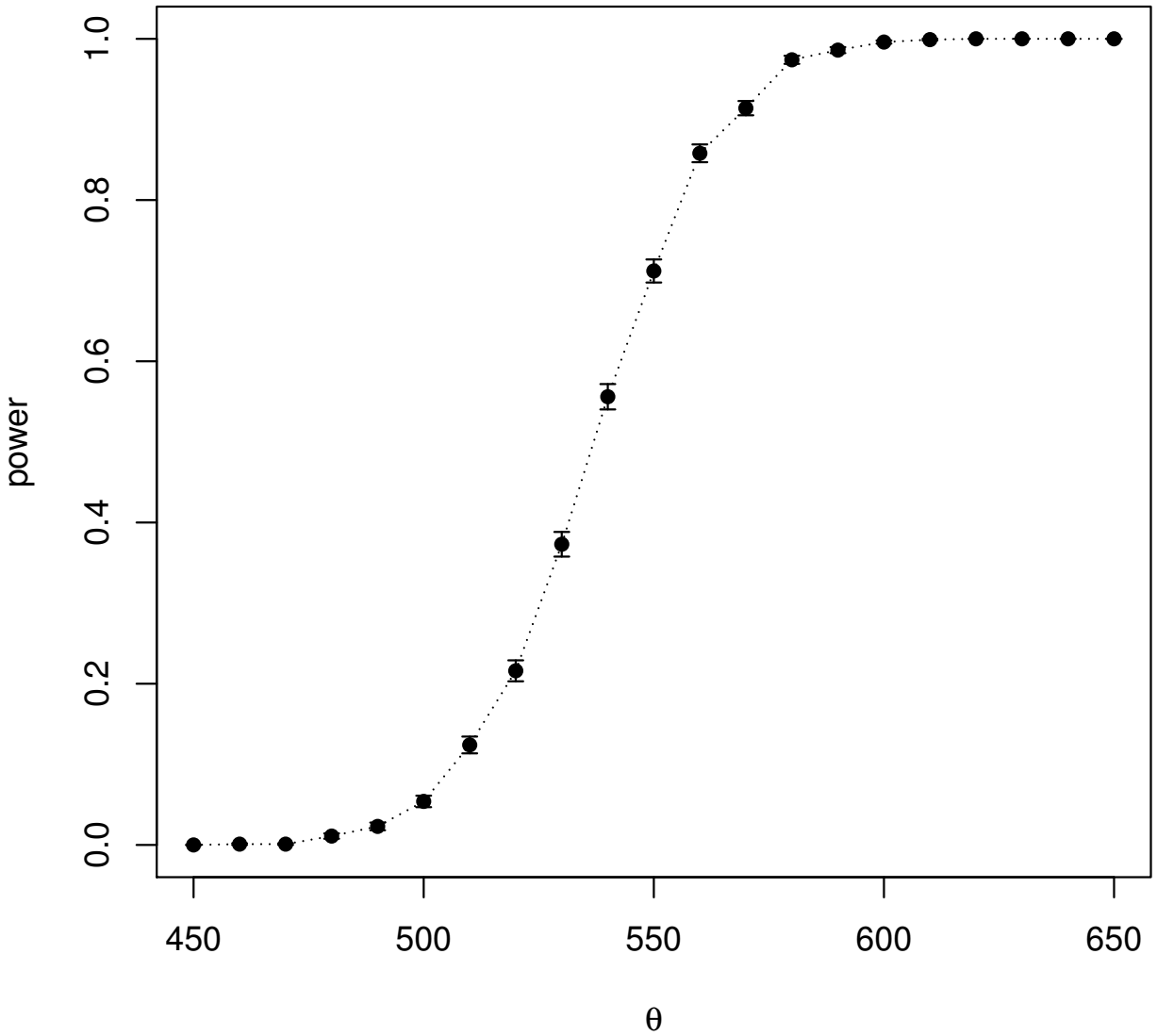
```
n <- 20
m <- 1000
mu0 <- 500
sigma <- 100
mu <- c(seq(450, 650, 10)) # alternatives
M <- length(mu)
power <- numeric(M)
for (i in 1:M) {
     mu1 <- mu[i]
     pvalues <- replicate(m, expr = {
             # simulate under alternative mu1
             x <- rnorm(n, mean = mu1, sd = sigma)
             ttest <- t.test(x,
             alternative = "greater", mu = mu0)
             ttest$p.value  } )
     power[i] <- mean(pvalues <= .05)
}
```

مثال: ادامه

برای رسم نمودار توان آزمون به همراه خطاهای متناظر در نقاطی که توان محاسبه شده است، یعنی $\hat{\pi}(\theta) \mp \hat{se}(\hat{\pi}(\theta))$، از دستور $errbar$ در بسته $Hmisc$ می‌توان استفاده کرد:

```
par(ask = TRUE)
library(Hmisc)  #for errbar
plot(mu, power, xlab = bquote(mu))
abline(v = mu0, lty = 1)
abline(h = .05, lty = 1)
lines(mu, power, lty=3)
# add standard errors
se <- sqrt(power * (1-power) / m)
errbar(mu, power, yplus = power+se, yminus = power-se,
xlab = bquote(theta))
lines(mu, power, lty=3)
detach(package:Hmisc)
par(ask = FALSE)
```


power
0.0
0.2
0.4
0.6
0.8
1.0
450
500
550
600
650
μ


power
0.0
0.2
0.4
0.6
0.8
1.0
450
500
550
600
650
θ

توان آزمون نرمال بودن بر اساس چولگی

برای توزیع نرمال آلوده

$$(۱ - p)N(۰, \sigma^۲ = ۱) + pN(۰, \sigma^۲ = ۱۰۰),$$

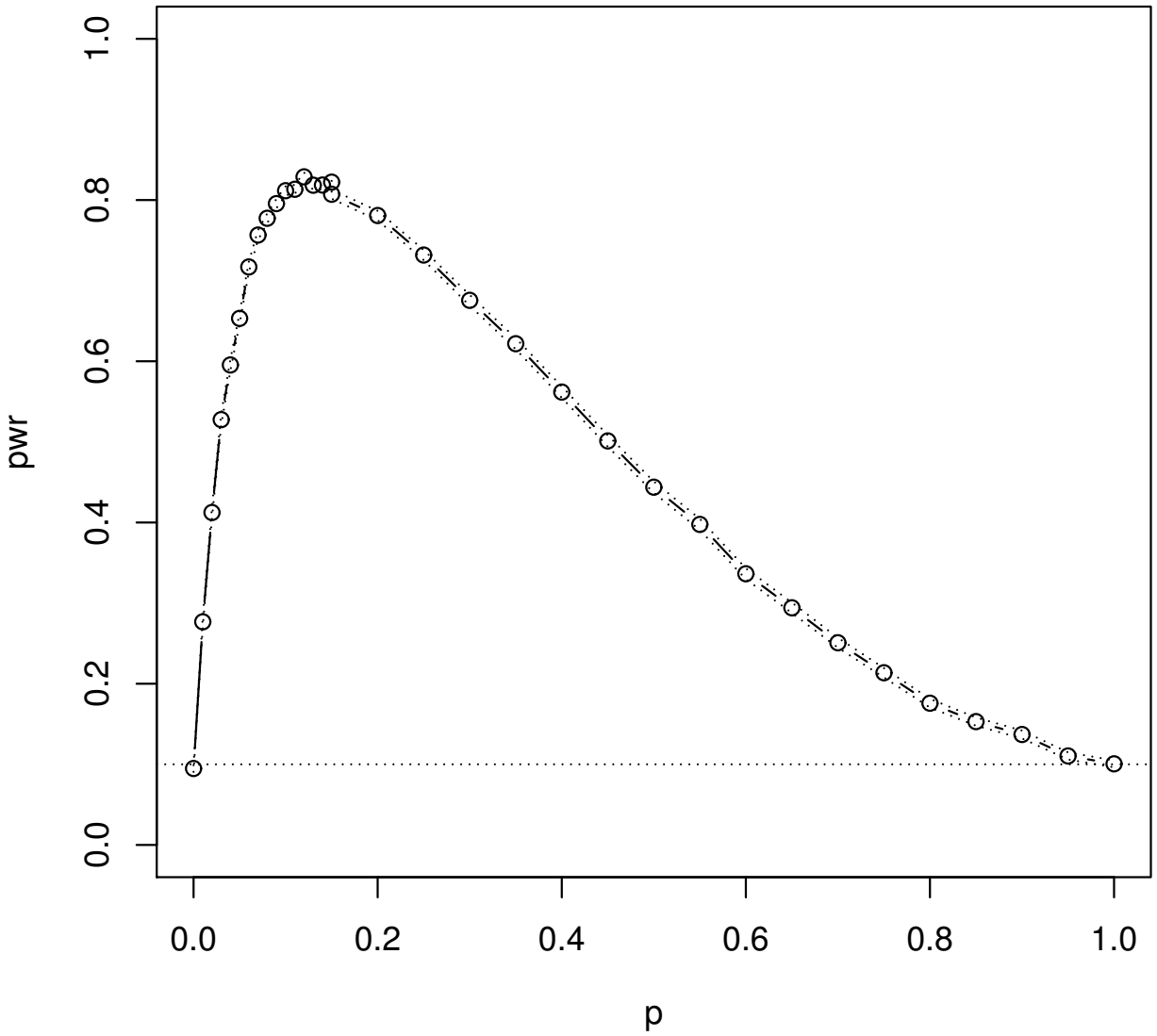
اگر $p = ۰$ یا $p = ۱$، توزیع نرمال است ولی برای هر $۰ < p < ۱$ توزیع نانرمال خواهد بود.

می‌توان توان آزمون سنجش نرمال بر اساس چولگی را برای دنباله‌ای از فرضیه‌های جانشین با تغییر p برآورد کرد و نمودار آن را رسم کرد.

فرض کنید $\alpha = ۰/۱$ و حجم نمونه $n = ۳۰$ انتخاب شود.

```
alpha <- .1
n <- 30
m <- 5000
p <- c(seq(0, .15, .01), seq(.15, 1, .05))
N <- length(p)
pwr <- numeric(N)
# critical value for the skewness test
cv <- qnorm(1-alpha/2, 0, sqrt(6*(n-2) / ((n+1)*(n+3))))
```

```
for (j in 1:N) {              # for each p
     e <- p[j]
     sktests <- numeric(m)
     for (i in 1:m) {          # for each replicate
          sigma <- sample(c(1, 10), replace = TRUE,
          size = n, prob = c(1-e, e))
          x <- rnorm(n, 0, sigma)
          sktests[i] <- as.integer(abs(sk(x)) >= cv)
          }
      pwr[j] <- mean(sktests)
}
# plot power vs p
plot(p, pwr, type = "b",
xlab = bquote(p), ylim = c(0,1))
abline(h = .1, lty = 3)
se <- sqrt(pwr * (1-pwr) / m)  # add standard errors
lines(p, pwr+se, lty = 3)
lines(p, pwr-se, lty = 3)
```


pwr
0.0
0.2
0.4
0.6
0.8
1.0
p
0.0
0.2
0.4
0.6
0.8
1.0

مقایسه توان آزمون‌ها

از شبیه‌سازی مونت کارلو، اغلب برای مقایسه عملکرد روش‌های مختلف استفاده می‌شود.

برای آزمون سنجش نرمال، عملکرد آزمون مبتنی بر چولگی را با دو آزمون معروف دیگر مقایسه می‌کنیم:

۱. آزمون شاپیرو ـ ویلک: این آزمون را در R می‌توان با تابع $shapiro.test$ انجام داد

۲. آزمون انرژی: این آزمون برای سنجش نرمال چندمتغیره معرفی شده است و مبتنی بر یک معیار فاصله معروف به فاصله انرژی است. این آزمون برای یک متغیره خیلی شبیه به آزمون اندرسون ـ دارلینگ عمل می‌کند. این آزمون توسط دستور $mvnorm.etest$ که در بسته $energy$ موجود است، قابل اجراست.

برای اطلاعات بیشتر در مورد این آزمون‌ها به کتاب مراجعه کنید.

برای این مثال، $\alpha = ۰/۱$ و فرضیه جانشین بر اساس توزیع نرمال آلوده تنظیم می‌شود.

```
library(energy)
alpha <- .1
n <- 30; m <- 500  # try small m for a trial run
test1 <- test2 <- test3 <- numeric(m)
# critical value for the skewness test
cv <- qnorm(1-alpha/2, 0, sqrt(6*(n-2) / ((n+1)*(n+3))))
sim <- matrix(0, 11, 4)
# estimate power
for (i in 0:10) {
     p <- i * .1
     for (j in 1:m) {
          e <- p
          sigma <- sample(c(1, 10), replace = TRUE,
          size = n, prob = c(1-e, e))
          x <- rnorm(n, 0, sigma)
          test1[j] <- as.integer(abs(sk(x)) >= cv)
          test2[j] <- as.integer(
                      shapiro.test(x)$p.value <= alpha)
          test3[j] <- as.integer(
                      mvnorm.etest(x, R=200)$p.value <= alpha)
          }
     print(c(p, mean(test1), mean(test2), mean(test3)))
     sim[i+1, ] <- c(p, mean(test1), mean(test2), mean(test3))
}
[1] 0.000 0.080 0.108 0.112
[1] 0.100 0.838 0.892 0.874
[1] 0.200 0.784 0.994 0.994
[1] 0.300 0.668 0.992 0.996
[1] 0.400 0.592 0.982 0.998
[1] 0.500 0.424 0.916 0.984
[1] 0.600 0.338 0.750 0.888
[1] 0.700 0.252 0.510 0.684
[1] 0.800 0.202 0.266 0.384
[1] 0.900 0.130 0.134 0.162
[1] 1.000 0.084 0.102 0.098
```

```
# plot the empirical estimates of power
plot(sim[,1], sim[,2], ylim = c(0, 1), type = "l",
xlab = bquote(p), ylab = "power")
lines(sim[,1], sim[,3], lty = 2)
lines(sim[,1], sim[,4], lty = 4)
abline(h = alpha, lty = 3)
legend("topright", 1, c("skewness", "S-W", "energy"),
lty = c(1,2,4), inset = .02)
```