STAT3003: 统计算法基础

蒙特卡洛方法（2）

曾靖

中国科学技术大学管理学院

大纲

大纲

1. 参数估计中的蒙特卡罗方法
    - 均方误差（MSE）
    - 标准误差（SE）
2. 区间估计和假设检验中的蒙特卡罗方法
    - 置信区间的覆盖率
    - 假设检验的 I 型错误率和功效

本节内容来自于《统计计算使用 *R*》第 *6* 章

1 参数估计中的蒙特卡罗方法

1.1 均方误差

- 设 $x_1, \dots, x_n$ 是随机变量 X 的 i.i.d. 样本, 记参数 $\theta = \theta(X)$ 的估计量为:

$$\boxed{\hat{\theta}_n = \hat{\theta}_n(x_1, \dots, x_n).}$$

- 衡量估计量精度的一个常用指标是均方误差（MSE）:

$$\boxed{\mathrm{MSE}(\hat{\theta}_n) = E\left[(\hat{\theta}_n - \theta)^2\right].}$$

- 接下来介绍如何通过 Monte Carlo 重复抽样估计这一期望。
- 记号约定：n 表示每份数据样本的观测数；m 表示单次 Monte Carlo 估计的重复次数；m' 表示将整个估计过程独立重复的外层次数。

例 1. 估计切尾均值的均方误差

设 X 为非正态的连续对称随机变量，中心为 $\theta = E(X) = 0$。对样本 $x_1, \ldots, x_n$，记次序统计量为 $x_{(1)} \leq \cdots \leq x_{(n)}$。

对切尾层数 k，定义第 k 层切尾均值估计量

$$\hat{\theta}_{[-k]} = \frac{1}{n-2k} \sum_{i=k+1}^{n-k} x_{(i)}.$$

Question: 如何用 *Monte Carlo* 方法估计 $\hat{\theta}_{[-k]}$ 的 *MSE*?

- 切尾均值舍弃两端的极端观测，可以减弱异常值的影响。

- 均方误差 $\xi = E(\hat{\theta}^2_{[-k]})$ 的 Monte Carlo 估计步骤如下：对于 $j = 1, \ldots, m$,
  1. 生成服从 X 分布的独立样本 $x_1^{(j)}, \ldots, x_n^{(j)}$;
  2. 将 n 个独立样本按升序排列，得到有序样本 $x_{(1)}^{(j)}, \ldots, x_{(n)}^{(j)}$;
  3. 计算第 k 层切尾均值估计

$$\hat{\theta}^{(j)}_{[-k]} = \frac{1}{n-2k} \sum_{i=k+1}^{n-k} x^{(j)}_{(i)}$$

  4. 估计第 k 层切尾均值的均方误差:

$$\widehat{\mathrm{MSE}} = \frac{1}{m} \sum_{j=1}^{m} (\hat{\theta}^{(j)}_{[-k]})^2.$$

- 正态分布

```
n <- 20
m <- 1000
k <- 1  # trimmed number
tmean <- replicate(m,
                   mean(sort(rnorm(n))[(k+1):(n-k)]))
mean(tmean^2)  # MSE
```

```
## [1] 0.04763241
```

- t 分布

```
tmean <- nmean <- numeric(m)
for(i in 1:m){
  x <- sort(rt(n, 3))
  nmean[i] <- mean(x)
  tmean[i] <- mean(x[(k+1):(n-k)])
}
cat("Normal mean: ", mean(nmean^2),
    "Trimmed mean: ", mean(tmean^2))  # MSE
```

```
## Normal mean:  0.1411215 Trimmed mean:  0.09969
```

- “污染” 正态分布（Contaminated normal）

考虑 “污染” 正态分布如下（混合高斯模型 GMM 的特例）

$$X \sim pN(0,1) + (1-p)N(0,10^2)$$

```
rnorm_con <- function(n, p){
  sigma <- sample(c(1,10), size = n, replace = TRUE,
                  prob = c(p, 1-p))
  rnorm(n, 0, sigma)
}
trimmed.mse <- function(n, m, k, p){
  tmean <- replicate(m,
                     mean(sort(rnorm_con(n, p))
                          [(k+1):(n-k)]))
  mean(tmean^2)
}
```

考虑不同的 k 和 p 对切尾均值的 MSE 的影响。

```
n <- 20
K <- n/2 - 1
m <- 1000
mse <- matrix(0, n/2, 3)

for(k in 0:K){
  mse[k+1, 1] <- trimmed.mse(n,m,k, p=1.0)
  mse[k+1, 2] <- trimmed.mse(n,m,k, p=0.95)
  mse[k+1, 3] <- trimmed.mse(n,m,k, p=0.9)
}
```

```r
mse <- data.frame(mse)
colnames(mse) <- c("p=1", "p=0.95", "p=0.9")
rownames(mse) <- 0:9
round(mse,3)
```

```
##      p=1 p=0.95 p=0.9
## 0 0.050  0.266 0.568
## 1 0.052  0.099 0.183
## 2 0.052  0.062 0.093
## 3 0.053  0.059 0.075
## 4 0.057  0.069 0.074
## 5 0.055  0.068 0.075
## 6 0.058  0.068 0.076
## 7 0.062  0.076 0.083
## 8 0.070  0.075 0.084
## 9 0.069  0.080 0.087
```

1.3 标准误差

- 估计量 $\hat{\theta}$ 的标准误差为

$$\mathrm{se}(\hat{\theta}) = \sqrt{\mathrm{Var}(\hat{\theta})}.$$

（方法一）将整个 Monte Carlo 估计过程独立重复 m' 次，每次使用 m 次抽样。记第 j 次估计为 $\hat{\theta}^{(j)}$，$j = 1, \ldots, m'$，则标准误差的估计为

$$\widehat{\mathrm{se}}(\hat{\theta}) = \left\{ \frac{1}{m'-1} \sum_{j=1}^{m'} (\hat{\theta}^{(j)} - \bar{\hat{\theta}})^2 \right\}^{1/2}, \qquad \bar{\hat{\theta}} = \frac{1}{m'} \sum_{j=1}^{m'} \hat{\theta}^{(j)}.$$

（方法二）当 $\hat{\theta}$ 是均值估计量，则 $\hat{\theta}$ 的标准误差估计可以重复利用改 Monte Carlo 实验中的数据，减小计算量。

- 例如，考虑 X_1, X_2 独立且服从标准正态分布，目标参数为 $\theta = E|X_1 - X_2|$，其 Monte Carlo 估计为

$$\hat{\theta} = \frac{1}{m}\sum_{j=1}^{m} |x_1^{(j)} - x_2^{(j)}|.$$

- 则 $\hat{\theta}$ 的标准误差为

$$\mathrm{se}(\hat{\theta}) = \left(\frac{1}{m}\mathrm{Var}(|X_1 - X_2|)\right)^{1/2}.$$

- 该标准误差 $\mathrm{se}(\hat{\theta})$ 对应的估计为

$$\widehat{\mathrm{se}}(\hat{\theta}) = \left(\frac{1}{m}\widehat{\mathrm{Var}}(|X_1 - X_2|)\right)^{1/2}.$$

例 2. 绝对差估计的标准误差

- 理论标准误差为

$$\mathrm{se}(\hat{\theta}) = \sqrt{\frac{1}{m}\mathrm{Var}(|X_1 - X_2|)} = \sqrt{\frac{1}{m}(2 - \frac{4}{\pi})} \approx 0.02695850.$$

```r
# 方法一：每次估计使用 m 次抽样，外层独立重复 m_prime 次
m <- 1000
m_prime <- 1000
theta_hat_outer <- numeric(m_prime)
for(i in seq_len(m_prime)){
  g <- replicate(m, {
    x <- rnorm(2)
    abs(x[1] - x[2])
  })
  theta_hat_outer[i] <- mean(g)
}

sd(theta_hat_outer) # outer Monte Carlo estimate of s.e.
```

```
## [1] 0.02635541
```

```r
# 方法二：单次估计使用 m 次抽样
m <- 1000
g <- replicate(m, {
  x <- rnorm(2)
  abs(x[1] - x[2])
})
sqrt(mean((g-mean(g))^2)/m)
```

```
## [1] 0.02796293
```

思考：每次估计均使用 m 个 *Monte Carlo* 样本时，两种方法分别需要多少个样本?

2 区间估计和假设检验中的蒙特卡罗方法

2.1 置信水平估计

- 给定由样本构造的置信区间 $\widehat{C}$，其实际覆盖率为

$$\beta = \Pr_{\theta}(\theta \in \widehat{C}) = E_{\theta}\left[I(\theta \in \widehat{C})\right].$$

- 难点：复杂分布下，覆盖率通常难以直接计算。
- *Monte Carlo* 思路：重复生成样本并构造区间，以覆盖真参数的频率估计 β。

- 置信水平 $\beta = E_\theta[I(\theta \in \widehat{C})]$ 的 Monte Carlo 估计步骤如下：对第 j 次重复试验, $j = 1, \ldots, m$:
  1. 生成独立的随机样本 $x_1^{(j)}, \ldots, x_n^{(j)}$；
  2. 根据随机样本计算置信区间 $\widehat{C}^{(j)}$；
  3. 计算 $\beta_j = I(\theta \in \widehat{C}^{(j)})$；
  4. 计算经验置信水平 $\hat{\beta} = \frac{1}{m}\sum_{j=1}^{m} \beta_j$。

- 由于每次 Monte Carlo 试验得到一个服从 Bernoulli 分布的独立变量 $\beta_j = I(\theta \in \widehat{C}^{(j)}) \sim \text{Bernoulli}(\beta)$，则置信水平估计 $\hat{\beta}$ 的标准误差为

$$\text{se}(\hat{\beta}) = \sqrt{\frac{\beta(1-\beta)}{m}},$$

- 该标准误差的估计为

$$\widehat{\text{se}}(\hat{\beta}) = \sqrt{\frac{\hat{\beta}(1-\hat{\beta})}{m}},$$

思考：若每次估计使用 m 次 *Monte Carlo* 试验，并将整个过程独立重复 m' 次，总共需要多少次模拟?

例 3. 方差的置信水平

设 $X_1, \ldots, X_n$ 是服从正态分布 $N(\mu, \sigma^2)$ 的随机样本，样本方差 $S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2$。

则方差 σ^2 的 $100(1-\alpha)\%$ 的置信区间为

$$\left(0, \frac{(n-1)S^2}{\chi^2_{n-1}(\alpha)}\right),$$

其中 $\chi^2_{n-1}(\alpha)$ 是卡方分布 χ^2_{n-1} 的下 α 分位数。可以证明该区间包含 σ^2 的概率为 $1-\alpha$。

如果分布不是正态，则置信水平可能远小于 $1-\alpha$。

- 正态分布 $X \sim N(0, 4)$

```
n <- 20
m <- 1000
alpha <- 0.05

UCL <- replicate(m, expr = {
  x <- rnorm(n ,0, 2)
  (n-1) * var(x) / qchisq(alpha, df = n-1)
})

# 计算经验置信水平
mean(UCL > 4)
```

```
## [1] 0.941
```

- 卡方分布 $X \sim \chi^2_2$(满足 $\mathrm{Var}(X) = 4$)

```
UCL <- replicate(m, expr = {
  x <- rchisq(n ,df = 2) # Var(X) = 4
  (n-1) * var(x) / qchisq(alpha, df = n-1)
})

# 计算经验置信水平
mean(UCL > 4)
```

```
## [1] 0.769
```

2.2 I 型错误的估计

- 考虑关于参数 $\theta \in \Theta$ 的假设检验问题，Θ 为参数空间。该问题具有如下的零假设与对立假设:

$$H_0 : \{\theta \in \Theta_0\}, \quad H_1 : \{\theta \in \Theta_1\},$$

其中 $\Theta = \Theta_0 \bigcup \Theta_1$。

- 对于给定的检验方法，记 π_θ 为参数值 θ 下拒绝 H_0 的概率:

$$\pi_\theta = \Pr_\theta(\text{Reject } H_0) = E_\theta(I(\text{Reject } H_0)).$$

- 当 $\theta \in \Theta_0$ 时，π_θ 是点态 I 型错误概率；当 $\theta \in \Theta_1$ 时，它是检验功效。

- 计算难点： I 型错误率和检验功效取决于抽样分布；模型复杂或分布假设不适用时，通常难以直接计算。
- *Monte Carlo* 思路：在给定参数值 θ 下重复生成样本并进行检验，以拒绝 H_0 的频率估计 π_θ。

- Monte Carlo 估计步骤如下：选定零假设中一个特定的参数值 $\theta \in \Theta_0$。对第 j 次重复试验, $j = 1, \ldots, m$:
  1. 在参数为 θ 的情况下生成第 j 个随机样本集 $x_1^{(j)}, \ldots, x_n^{(j)}$;
  2. 对第 j 个随机样本计算检验统计量 T_j;
  3. 如果 H_0 在显著性水平 α 下被拒绝，那么令检验决策 $I_j = 1$，否则令 $I_j = 0$;
  4. 计算 I 型错误 $\hat{\pi}_\theta = \frac{1}{m} \sum_{j=1}^{m} I_j$。

- 每次 Monte Carlo 重复试验得到一个服从 Bernoulli 分布的独立变量 $I_j = I(\text{Reject } H_0) \sim \text{Bernoulli}(\pi_\theta)$.
- I 型错误估计 $\hat{\pi}_\theta$ 的标准误差为

$$\text{se}(\hat{\pi}_\theta) = \sqrt{\frac{\pi_\theta(1-\pi_\theta)}{m}},$$

- 则标准误差的估计为

$$\widehat{\text{se}}(\hat{\pi}_\theta) = \sqrt{\frac{\hat{\pi}_\theta(1-\hat{\pi}_\theta)}{m}}.$$

例 4. t 检验

设 $x_1, \ldots, x_n$, $n = 20$ 是服从 $N(\mu, \sigma^2)$ 分布的随机样本。在 $\alpha = 0.05$ 的显著水平下检验

$$H_0 : \mu = 500 \quad vs. \quad H_1 : \mu > 500.$$

- 记 $\bar{x}$ 和 S 为样本均值和样本方差，t_{19} 为自由度为 19 的 t 分布。在零假设下

$$T = \frac{\bar{x} - 500}{S/\sqrt{20}} \sim t_{19}.$$

- 构造拒绝域: $R_1 := \{T \geq t_{19}(0.95)\}$。
- 观测到的统计量取值为 t 时，右尾 p-value $= 1 - F_{t_{19}}(t)$。由于 $F_{t_{19}}$ 连续严格递增，p-value ≤ 0.05 等价于 $t \geq t_{19}(0.95)$，故 $R_1 = R_2$，这一等价关系不要求 H_0 为真。
- 若 H_0 成立，则 $T \sim t_{19}$，从而 p-value $\sim$ Unif$(0, 1)$。

```r
n <- 20
alpha <- 0.05
mu0 <- 500
sigma <- 100

m <- 1000
p <- numeric(m)
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] 0.056000000 0.007270763
```

2.4 检验功效的估计

- 给定 $\theta_1 \in \Theta_1$, II 型错误为

$$1 - \pi_{\theta_1} = \Pr(\text{Accept } \Theta_0 \mid \theta_1).$$

- 对应的功效为

$$\pi_{\theta_1} = \Pr_{\theta_1}(\text{Reject } \Theta_0).$$

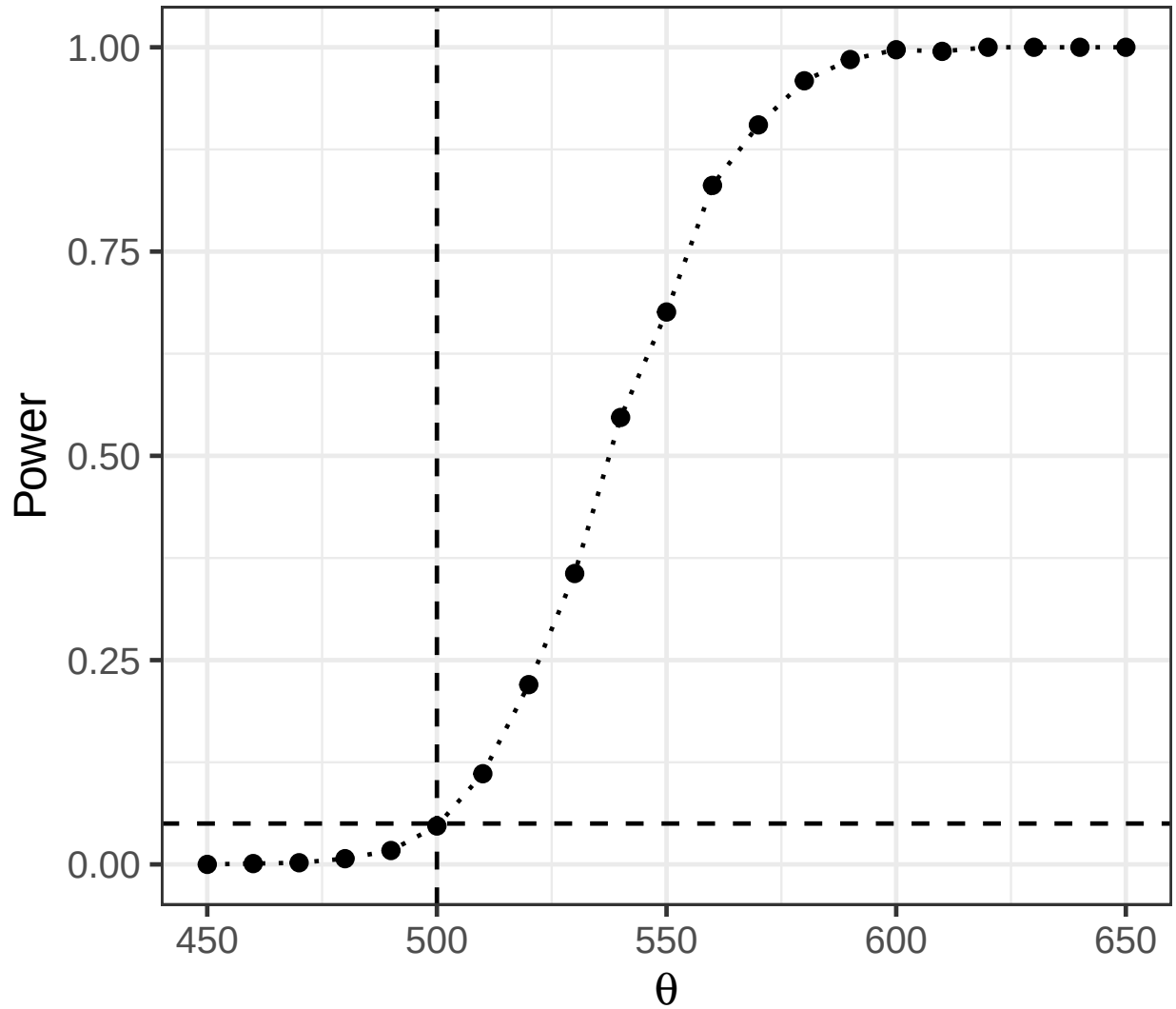
- 在假设检验中，我们一般控制 I 型错误，并且尽量使其功效较大。

- Monte Carlo 估计步骤如下：选定对立假设中一个特定的参数值 $\theta_1 \in \Theta_1$。对第 j 次重复试验，$j = 1, \ldots, m$:
  1. 在参数为 θ_1 的情况下生成第 j 个随机样本集 $x_1^{(j)}, \ldots, x_n^{(j)}$;
  2. 对第 j 个随机样本计算检验统计量 T_j;
  3. 如果 H_0 在显著性水平 α 下被拒绝，那么令检验决策 $I_j = 1$，否则令 $I_j = 0$;
  4. 计算功效 $\hat{\pi}_{\theta_1} = \frac{1}{m} \sum_{j=1}^{m} I_j$

例 5. t 检验的功效

```
m <- 1000 # 每个参数值下的 Monte Carlo 重复次数
alpha <- 0.05
mu <- seq(450, 650, 10)
power <- numeric(length(mu))
for (i in seq_along(mu)){
  mu1 <- mu[i]
  pvalues <- replicate(m, expr = {
      x <- rnorm(n, mu1, sigma)
      ttest <- t.test(x, alternative = "greater",
                      mu = mu0)
      ttest$p.value
  })
  power[i] <- mean(pvalues < alpha)
}
```

```r
power_data <- data.frame(mu = mu, power = power)
ggplot(power_data, aes(x = mu, y = power)) +
  geom_point() + geom_line(linetype = 3) +
  geom_vline(xintercept = mu0, linetype = 2) +
  geom_hline(yintercept = alpha, linetype = 2) +
  labs(x = expression(theta), y = "Power") +
  theme_bw()
```


1.00
0.75
0.50
0.25
0.00
Power
450
500
550
600
650
θ

例 6. “污染” 正态分布

考虑 “污染” 正态分布:

$$X \sim pN(\mu, 1) + (1-p)N(\mu, 10^2)$$

原假设和对立假设为

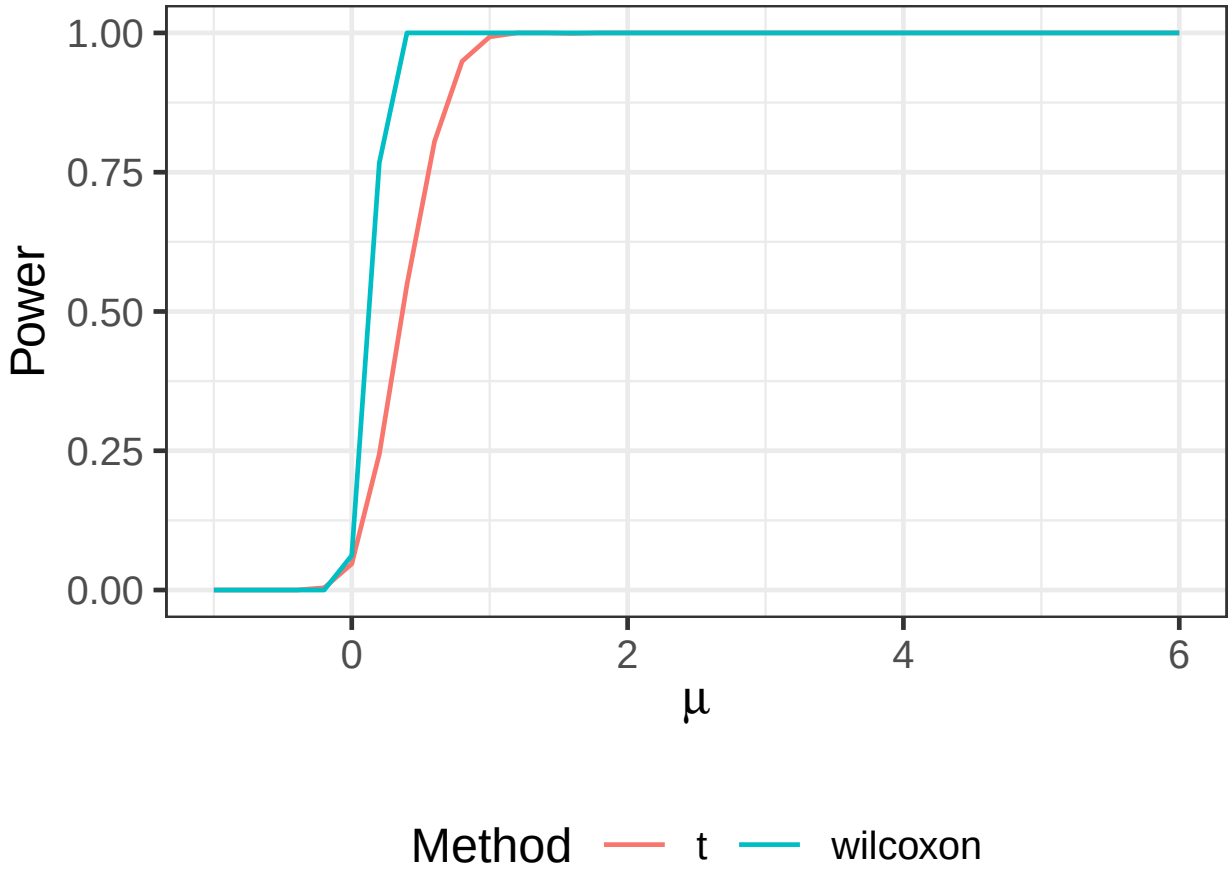
$$H_0 : \mu = 0 \quad vs. \quad H_1 : \mu > 0.$$

比较该假设检验问题下 t 检验和 *Wilcox* 检验的检验功效

```r
rnorm_con <- function(n, p, mu){
  sigma <- sample(c(1,10), size = n, replace = TRUE,
                  prob = c(p, 1-p))
  rnorm(n, mu, sigma)
}
```

```r
n <- 200 # 每份数据样本的观测数
m <- 1000 # Monte Carlo 重复次数
alpha <- 0.05
mu0 <- 0 # 原假设
mu <- seq(-1, 6, 0.2) # 不同的对立假设
power <- matrix(0, length(mu), 2)
for (i in seq_along(mu)){
  mu1 <- mu[i]
  pvalues <- replicate(m, expr = {
      x <- rnorm_con(n, 0.9, mu1)
      ttest <-
        t.test(x, alternative = "greater", mu = mu0)
      wtest <-
        wilcox.test(x, alternative = "greater", mu =
        mu0)
      c(ttest$p.value, wtest$p.value)
  })
  power[i,] <- apply(pvalues < alpha, 1, mean)
}
```

```r
library(reshape2)
data <- data.frame(x = mu, t = power[,1],
                   wilcoxon = power[,2])
data <- melt(data, id = "x", variable.name =
"Method")
  ggplot(data, aes(x = x, y = value, color =
  Method))+
    geom_line()+
    labs(x = TeX("$\\mu$"), y = "Power")+
    theme_bw()+
    theme(legend.position = "bottom")
```


1.00
0.75
0.50
0.25
0.00
Power
0
2
4
6
μ
Method
t
wilcoxon