Rによるトマトのシミュレーション

超絶イケメンの藤田

2024-02-01

目次

はじめに

この pdf は 3 年生の授業において標本の大きさをかえたときに推定の精度がどのように変化するかを教科書のデータをもとにシミュレーションしたものである。R を使用しており、再現性のある内容となっている（これは R の利点である）。しかし、皆さんにとって重要なことは**コードの内容よりも結果である**（とくに図を大事にしてもらいたい）。なので、コードは表示しているが読み飛ばし、結果のみを見て、ああこんなもんか、と感じてもらいたい。内容自体は大したことは言っていないように感じるかもしれないが、高校数学における統計的な推測という範囲につながる非常に重要な内容が含まれている。さらにいえば、高校数学でこの領域が重要視された原因はデータ分析が社会的に重要であるにも関わらず今回の内容にかかわる領域を正しく理解している人が少ないことが問題視されたことが背景にあるとされることがある。高校の数学の先生でも新しい内容であるため正しく理解していない人は決して少なくないと言われている。もし将来 R に出会うことがあれば記憶に出てきてくれれば少し嬉しいかもしれない。なお、R のバージョンが古いため現時点でも古い記述方法を行っている箇所がある（例:%>% は古い記法で現在は|>が主流）。また、毎度のことだが**思いつきで作成している**節があるので、日本語の体裁を整える気が一切ない。そのため段落などがぐちゃぐちゃになっているが気にしないでいただきたい。

初期設定

```
pacman::p_load(tidyverse,
               broom,
               extraDistr,
               patchwork)

if (.Platform$OS.type == "windows") {
  # Window
  if (require(fontregisterer)) {
    my_font <- "Yu Gothic"
  } else {
    my_font <- "Japan1"
  }
} else if (capabilities("aqua")) {
  # macOS
  my_font <- "HiraginoSans-W3"
```

```
} else {
  # Unix/Linux
  my_font <- "IPAexGothic"
}

theme_set(theme_gray(base_size    = 9,
                     base_family = my_font))
set.seed(1230524)
```

データクレンジング

```
dat <- read_csv("data/tt.csv")
dat1 <- c()
for (i in seq(2, 11, 1)) {
dat1 <- c(dat1, unlist(dat[ ,i]))
}
dat1 <- dat1 %>%
  as_tibble() %>%
  filter(value != "NA")

myd <- tibble(id = seq(1, 300, 1),
              suger = dat1$value)
```

無作為抽出と平均値

母集団の平均値を求める。これが真の値である。

```
t_mean <- mean(myd$suger)
```

母集団の平均値は 7.035 であることが分かった。

サンプルサイズ 5 のとき

サンプルサイズを 5 にして平均値を計算する。

```
N_5 <- sample(myd$suger, 5)
```

このとき、抽出されたデータは

```
N_5
```

```
## [1] 8.4 7.3 7.7 6.8 7.0
```

である。平均値を計算する。

```
t5_mean <- mean(N_5)
```

平均値は 7.44 である。この作業を 20 回繰り返し、箱ひげ図をかく。

※この作業は標本の大きさ 5 の標本を 20 個取り出すことを意味する。つまり、標本の大きさが 5 であり、標本の数が 20 である。サンプルサイズとサンプル数が違うということに注意されたい。

```
m_5 <- c()
for (i in 1:20) {
  N_5 <- sample(myd$suger, 5)
  m_5 <- c(m_5, mean(N_5))
}
```

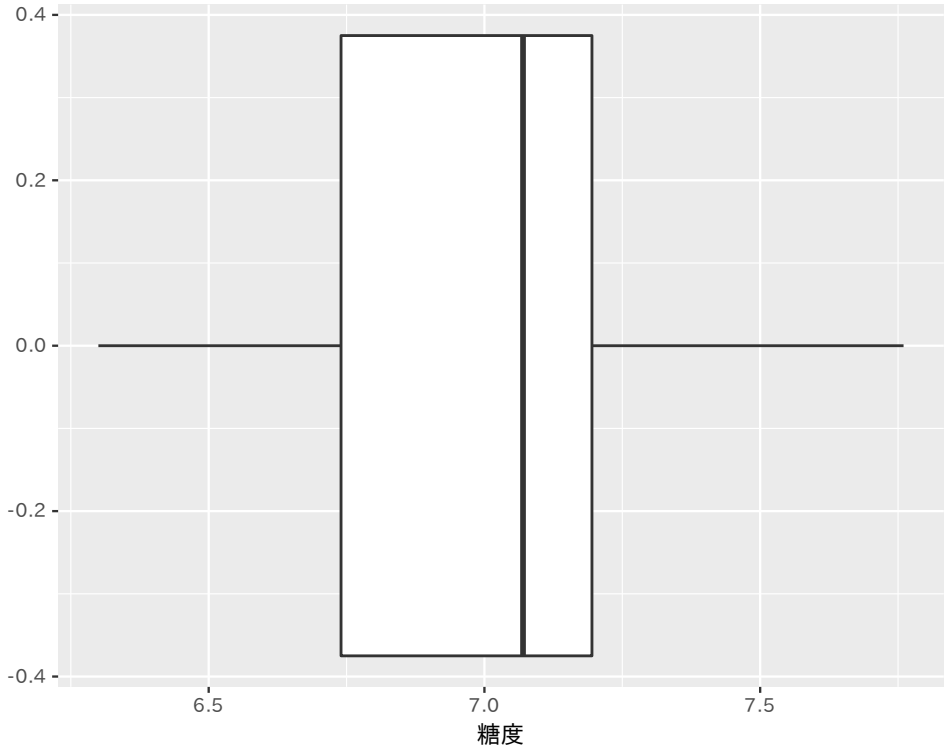
抽出したデータ各々の平均値は

```
m_5
```

```
##  [1] 6.86 7.56 7.14 7.14 7.08 7.46 7.42 6.60 7.36 6.30 6.90 7.06 6.38 7.08 6.74
## [16] 6.88 6.74 7.76 7.12 6.60
```

となっている。このデータを用いて、箱ひげ図をかく。

```
plt1 <- m_5 %>%
  as_tibble() %>%
  ggplot(aes(x = value)) +
  geom_boxplot() +
  labs(x = "糖度")
plot(plt1)
```



上の図の解釈は後に任せる。

サンプルサイズ 10 のとき

同様の操作を行う。

```
N_10 <- sample(myd$suger, 10)
```

抽出したデータは

```
N_10
```

```
##  [1] 7.3 7.8 7.2 6.9 7.0 7.4 5.7 8.1 6.9 6.8
```

である。平均値を計算する。

```
t10_mean <- mean(N_10)
```

平均値は 7.11 であることが分かった。この操作を 20 回繰り返す。

```
m_10 <- c()
for (i in 1:20) {
  N_10 <- sample(myd$suger, 10)
```

```
  m_10 <- c(m_10, mean(N_10))
}
```

このデータの中身は

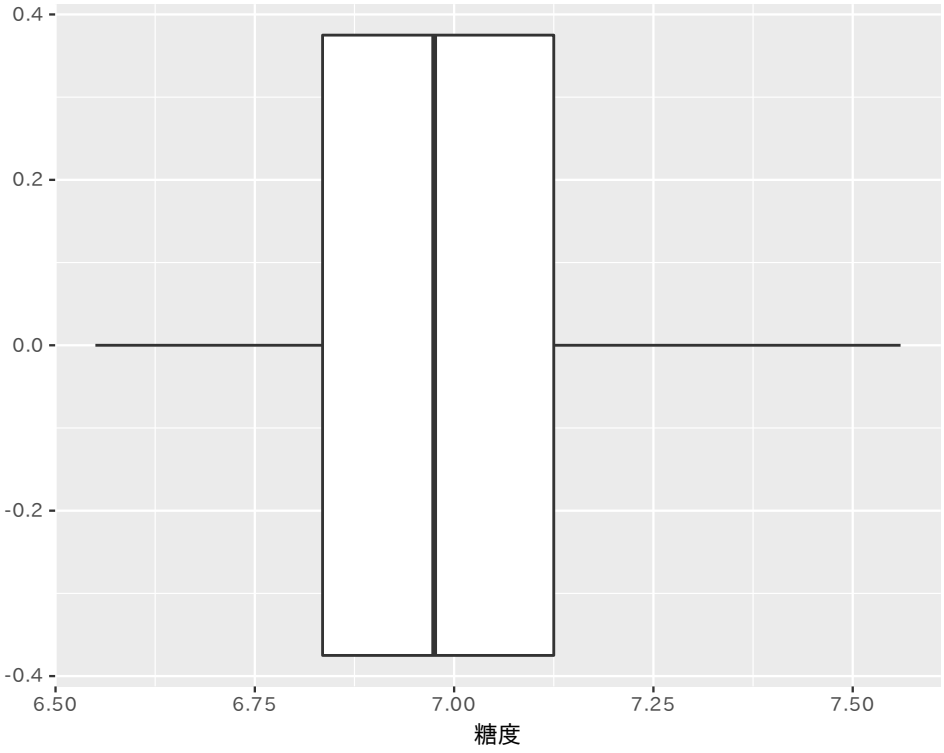
```
m_10
```

```
##  [1] 7.41 6.73 6.95 6.90 7.32 6.79 7.56 6.55 6.79 7.03 7.01 6.97 6.96 6.85 7.35
## [16] 6.98 6.98 7.06 7.43 6.74
```

となっている。

箱ひげ図をかく。

```
plt2 <- m_10 %>%
  as_tibble() %>%
  ggplot(aes(x = value)) +
  geom_boxplot() +
  labs(x= "糖度")
plot(plt2)
```



サンプルサイズ 50 のとき

標本 50 個を取り出す。

```
N_50 <- sample(myd$suger, 50)
```

取り出されたデータは

```
N_50
```

```
##  [1] 8.4 5.9 6.5 6.6 6.4 9.2 5.3 7.7 8.1 6.6 6.7 7.0 6.6 8.1 7.9 6.9 8.2 6.5 7.4
## [20] 7.7 6.4 6.9 7.4 7.8 8.7 6.4 7.9 8.0 7.6 6.8 7.4 7.3 6.5 7.1 6.1 6.1 6.9 8.2
## [39] 6.7 6.6 7.5 6.4 8.1 7.9 6.7 6.3 6.9 5.9 7.0 8.2
```

である。このとき、平均値は

```
t50_mean <- mean(N_50)
```

平均値は 7.148 であることがわかった。

この操作を 20 回繰り返す。

```
m_50 <- c()
for (i in 1:20) {
  N_50 <- sample(myd$suger, 50)
  m_50 <- c(m_50, mean(N_50))
}
```

20 回の平均値のデータは

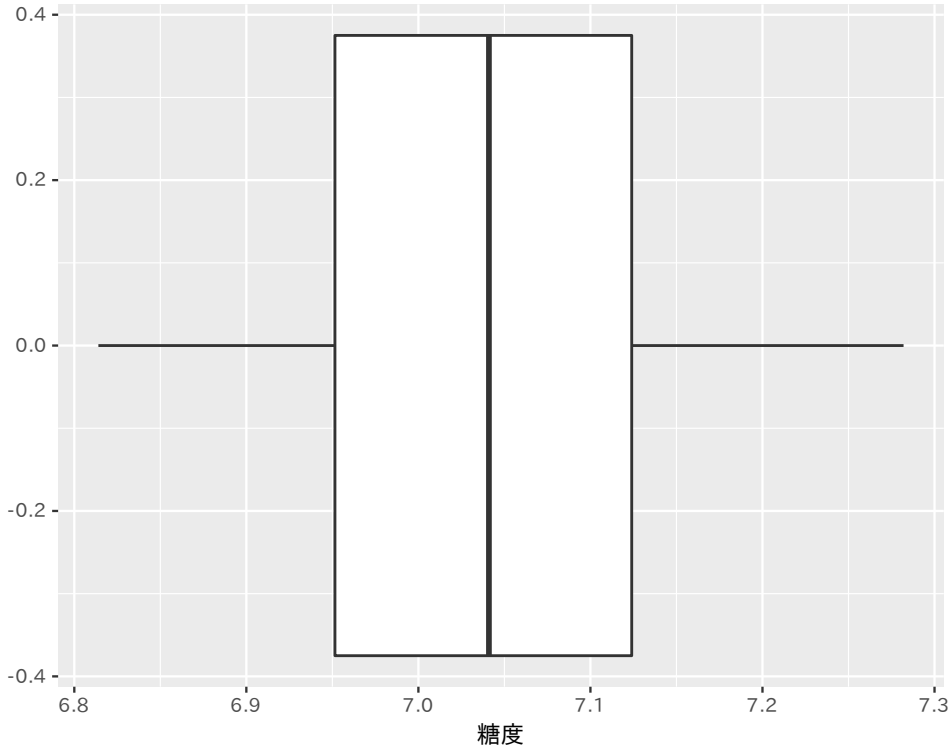
```
m_50
```

```
##  [1] 6.814 7.282 7.136 7.194 7.092 6.958 7.060 6.996 7.216 6.968 7.118 7.152
## [13] 7.022 6.902 6.932 6.828 7.074 7.010 6.904 7.120
```

となっている。このデータを使って箱ひげ図をかいてみる。

```
plt3 <- m_50 %>%
  as_tibble() %>%
  ggplot(aes(x = value)) +
  geom_boxplot() +
  labs(x = "糖度")
```

```
plot(plt3)
```

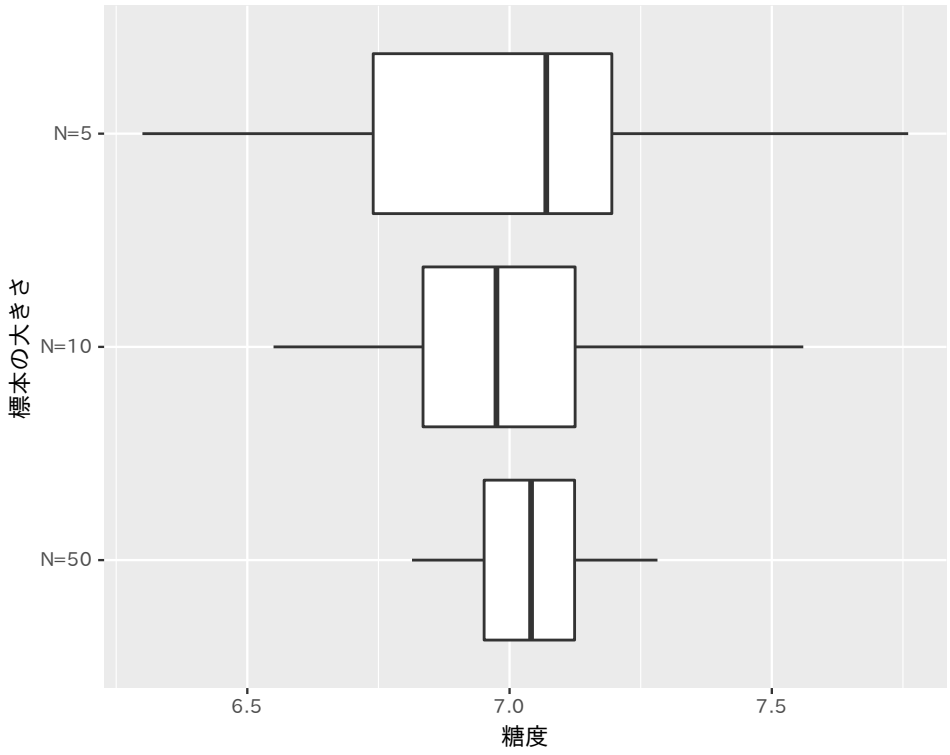


箱ひげ図を 3 つ並べる

箱ひげ図を 3 つ合体させて標本の大きさをかえることでどのような結果になるかを確認してみる。

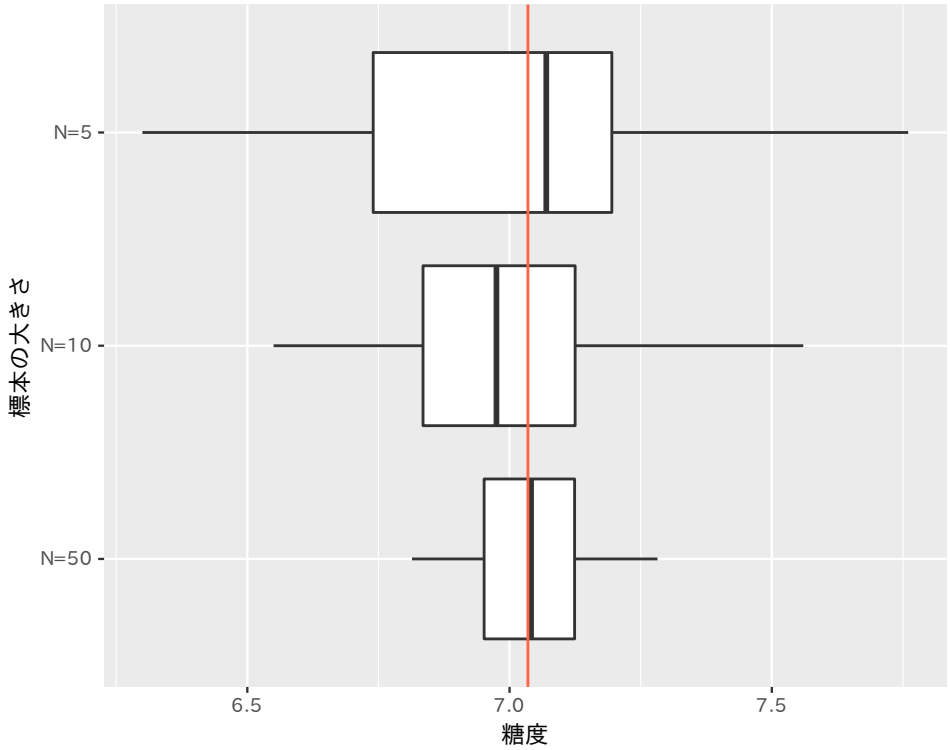
```
d_box <- tibble("N=5" = m_5,
                "N=10" = m_10,
                "N=50" = m_50) %>%
  pivot_longer(cols = everything(),
               names_to = "N",
               values_to = "value")
ord_N <- c("N=5", "N=10", "N=50")
d_box <- d_box %>%
  mutate(f_N = factor(N, levels = ord_N))

plt <- d_box %>%
  ggplot(aes(x = value, y = fct_rev(f_N))) +
```

```
  geom_boxplot() +
  labs(x = "糖度",
       y = "標本の大きさ")
plot(plt)
```



上の箱ひげ図から、標本の大きさが大きくなるにつれ、平均値の散らばりが小さくなっていることがわかる。この図に最初に計算した母集団の平均値を表す線を追加してみる。

```
plt <- plt +
  geom_vline(xintercept = t_mean, color = "tomato")
plot(plt)
```



上の図で赤い線が母集団の平均値を表している。標本の大きさが大きくなるとちらばりが小さくなっていることから、真の値（付近）をとらえた結果を手に入れられる確率が高くなっていることがわかる。

このままだと教科書通りなので、標本の大きさをいろいろ変えてシミュレーションしてみる

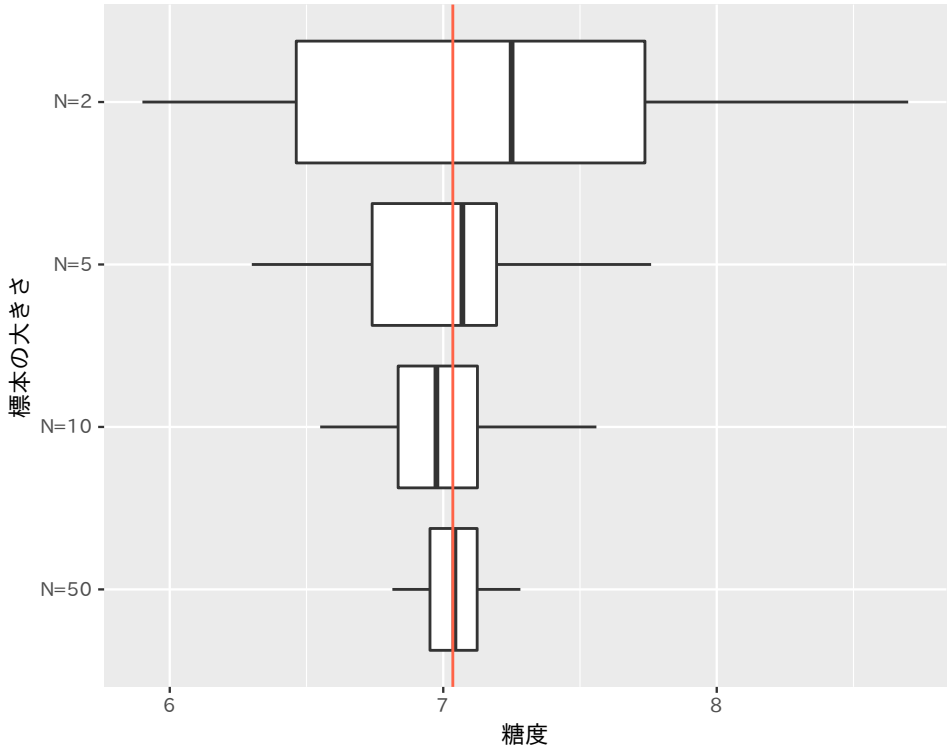
この作業を何度も繰り返すのは非常に面倒なので、関数を作成する。

```
tomato_sim <- function(N) {
  m_N <- c()
  for (i in 1:20) {
    N_n <- sample(myd$suger, N)
    m_N <- c(m_N, mean(N_n))
  }
  m_5 <- tibble(value = m_5,
                N = "N=5")
  m_10 <- tibble(value = m_10,
                 N = "N=10")
  m_50 <- tibble(value = m_50,
```

```
                  N = "N=50")
  names_N <- paste0("N=", as.character(N))
  m_N <- tibble(value = m_N,
                N = names_N)
  d_box <- bind_rows(m_5, m_10) %>%
    bind_rows(m_50) %>%
    bind_rows(m_N)
   ord_N <- c("N=5", names_N, "N=10", "N=50")
  if(5<N & N<10) {
     ord_N <- c("N=5", names_N, "N=10", "N=50")
  } else {
    if(10<N & N<50) {
       ord_N <- c("N=5", "N=10", names_N, "N=50")
    } else {
      if(50 < N) {
         ord_N <- c("N=5", "N=10", "N=50", names_N)
      } else {
         ord_N <- c(names_N, "N=5", "N=10", "N=50")
      }
    }
  }
   d_box <- d_box %>%
     mutate(N = factor(N, levels = ord_N))
   plt <- d_box %>%
     ggplot(aes(x = value, y = fct_rev(N))) +
     geom_boxplot() +
     geom_vline(xintercept = t_mean, color = "tomato") +
     labs(x = "糖度",
          y = "標本の大きさ")
  plot(plt)
}
```

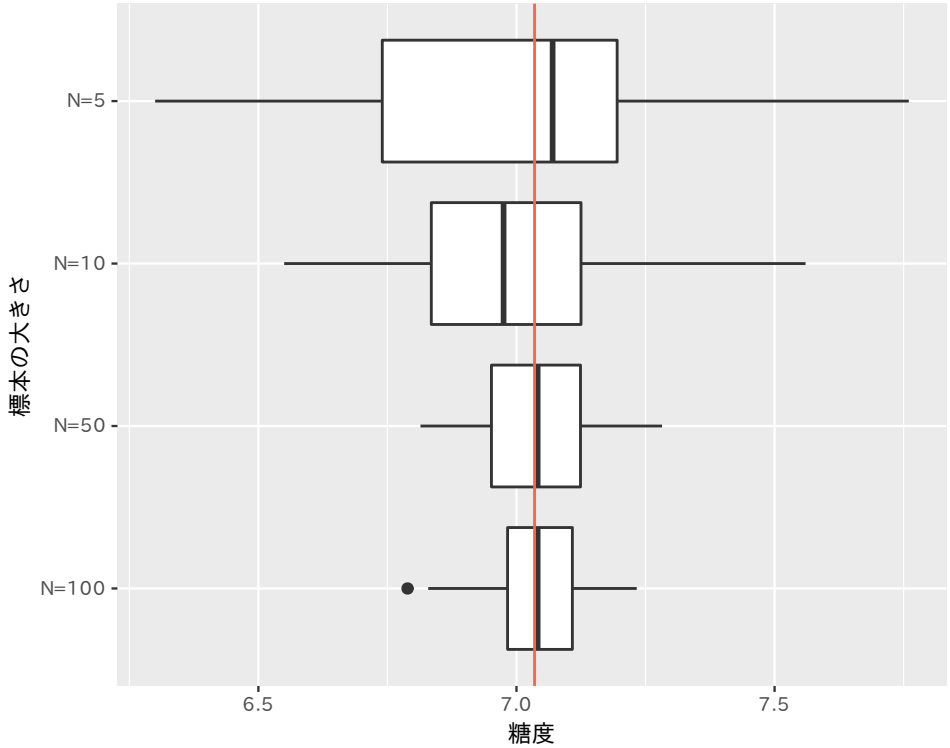
N = 2 のとき

```
tomato_sim(2)
```



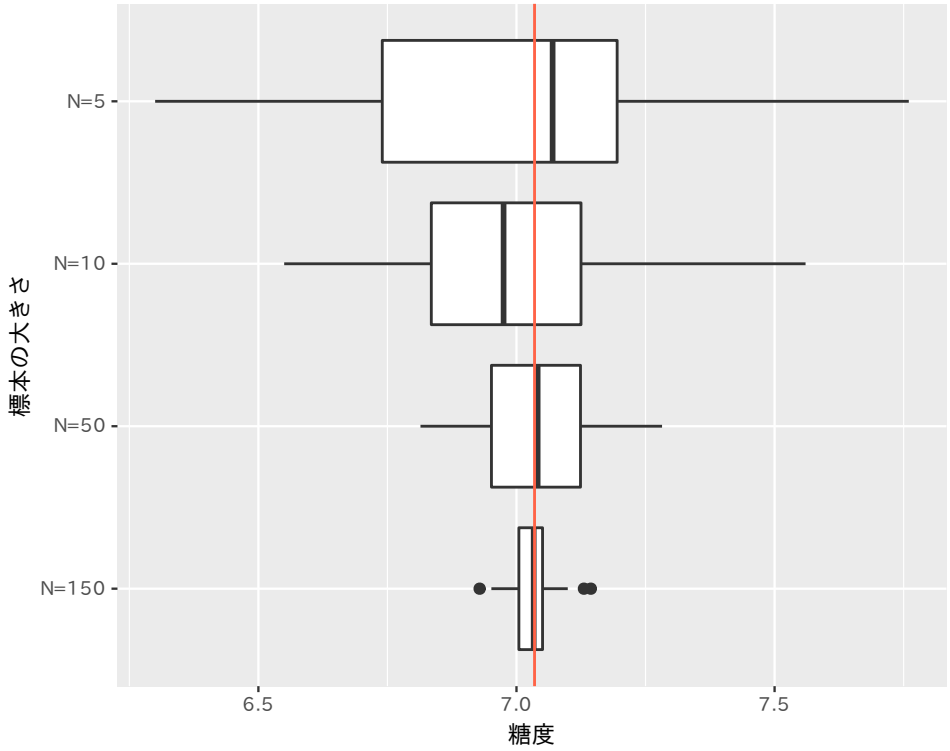
N = 100 のとき

```
tomato_sim(100)
```



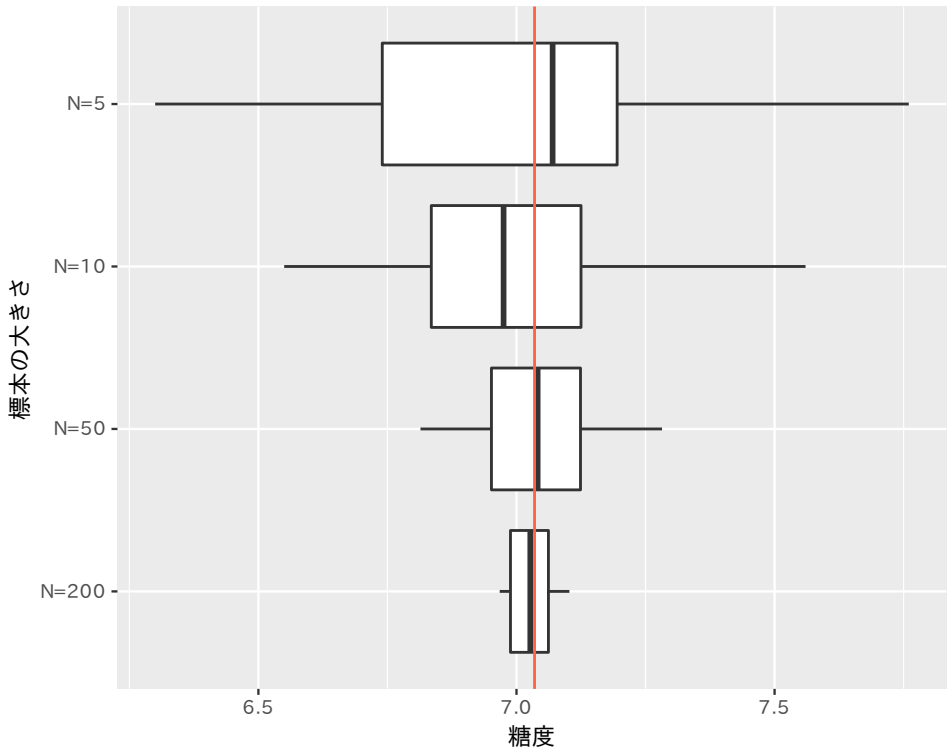
N = 150 のとき

```
tomato_sim(150)
```



N = 200 のとき

```
tomato_sim(200)
```


N=5
N=10
N=50
N=200
標本の大きさ
6.5
7.0
7.5
糖度