

代表性与显著性

Representativeness and significance

Qingyao Zhang

2026-05-14

目录

1 总体	2
2 总体分布	3
3 取样过程	4
4 样本	4
5 基于样本估计总体均值与标准差	5
6 样本分布	5
6.1 样本分布与总体分布的关系	6
7 重复取样	6
8 1000 个样本的均值的均值	7
9 抽样分布	7
9.1 抽样分布与总体分布、样本分布的关系	8
10 代表性	9
11 显著性	9
12 差异显著的概率标准	9
13 sample1 均值与总体均值的直观比较	12
14 z 分数与单样本 z 检验	12

15 t 分数与单样本 t 检验	14
16 自由度	15
17 t 分布的自由度	15

```
1 library(Keng)
2 data("well")
3 library(ggplot2)
```

1 总体

我们捏造一个人数为 10000 人的总体，为 10000 名被试设定编号 ID，并捏造出其 IQ 数据，设定 IQ 的均值为 100，标准差为 15。

```
1 # Population, 总体
2 # 总体均值设置为 100
3 mu <- 100
4 # 总体标准差设置为 15
5 sigma <- 15
6 # 随机种子值设置为 2026
7 set.seed(2026)
8 # 将 ID 与 IQ 存入数据表中
9 population10000 <- data.frame(
10   # 生成 10000 个 ID (被试编号)
11   ID = seq(1:10000),
12   # 生成 10000 个均值为 mu、标注差为 delta 的 IQ 分数
13   IQ = round(rnorm(10000, mean = mu, sd = sigma), 0)
14 )
15 # 查看数据表
16 # View(population10000)
17 # 查看数据表前 6 行
18 range(population10000$IQ)
19 ## [1] 45 158
20 head(population10000)
21 ##   ID  IQ
22 ## 1  1 108
23 ## 2  2  84
24 ## 3  3 102
25 ## 4  4  99
26 ## 5  5  90
```

```

27 ## 6 6 62
28 # 总体的实际均值
29 mean(population10000$IQ)
30 ## [1] 100.0531
31 # 总体的实际标准差
32 sqrt(sum((population10000$IQ - mean(population10000$IQ))^2)/10000)
33 ## [1] 15.01792

```

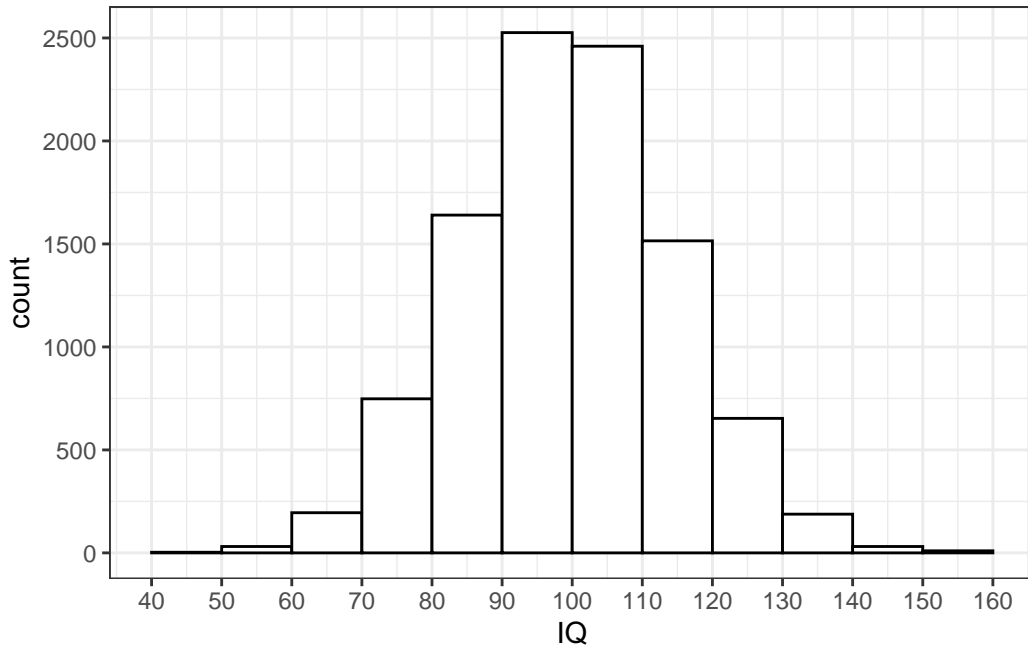
2 总体分布

总体分布 (population distribution) 是总体 (10000 名被试 IQ 得分) 的分布。注意该分布的横纵坐标的全距。

```

1 # 作总体的直方图
2 # 总体分布是 10000 个数据点的分布。
3 library(ggplot2)
4 ggplot(population10000, aes(x = IQ)) +
5   geom_histogram(breaks = seq(40, 160, 10),
6                 fill = "white",
7                 color = "black") +
8   scale_x_continuous(breaks = seq(40, 160, 10)) +
9   coord_cartesian(xlim = c(40, 160)) +
10  theme_bw()

```



3 取样过程

接下来，我们模拟取样的过程。我们从 $N = 10000$ 的总体中随机抽取一个样本量 $n = 100$ 的样本。我们称这个样本为 `sample1`。

```
1 # 从总体中随机抽取一个样本量为 100 的样本，抽取其 ID
2 sample1ID <- sample(population10000$ID, 100)
3 # 根据 ID 选出 sample1 的数据
4 sample1 <- population10000[sample1ID, ]
```

4 样本

查看 `sample1` 中的数据。注意：被试 ID 的顺序是乱的，这是由随机取样导致的。

```
1 # 查看样本
2 head(sample1)
3 ##      ID  IQ
4 ## 5784 5784 107
5 ## 4443 4443 103
6 ## 1968 1968  91
```

```
7 ## 1890 1890 85
8 ## 697 697 111
9 ## 9873 9873 101
```

5 基于样本估计总体均值与标准差

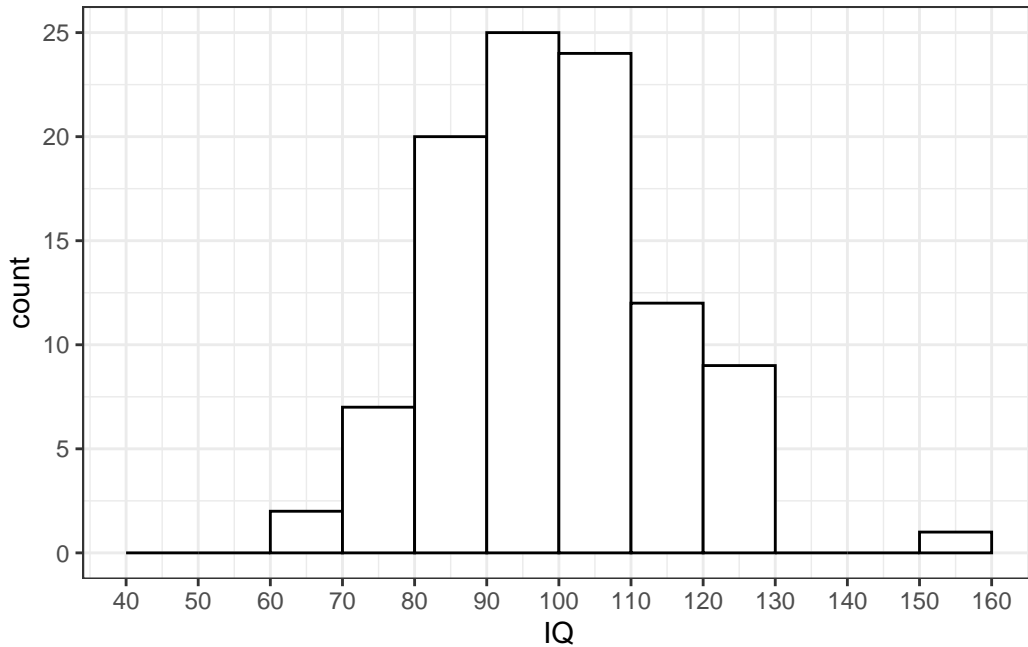
样本的均值是总体均值的无偏估计。样本的标准差总是小于总体的标准差，因此使用样本的无偏标准差作为总体标准差的无偏估计。尽管名为“无偏”，实际上还是有偏差的。

```
1 # 取值范围
2 range(sample1$IQ)
3 ## [1] 65 154
4 # 计算 sample1 的 IQ 的均值
5 sample1_mean <- mean(sample1$IQ)
6 sample1_mean
7 ## [1] 99.2
8 # 计算 sample1 的 IQ 的标准差
9 sample1_sd <- sqrt(sum((sample1$IQ - sample1_mean)^2)/100)
10 sample1_sd
11 ## [1] 15.12614
12 # sample1 的无偏标准差 (基于 sample1 估计总体的标准差)
13 sample1_sd_unbiased <- sqrt(sum((sample1$IQ - sample1_mean)^2)/(100 - 1))
14 sample1_sd_unbiased
15 ## [1] 15.20234
16 # 使用`sd()`函数计算 sample1 的无偏标准差 (总体标准差的估计值)
17 sd(sample1$IQ)
18 ## [1] 15.20234
```

6 样本分布

样本分布 (sample distribution) 是样本中 100 个被试 IQ 得分的分布。

```
1 ggplot(sample1, aes(IQ)) +
2   geom_histogram(breaks = seq(40, 160, 10),
3                 fill = "white",
4                 color = "black") +
5   scale_x_continuous(breaks = seq(40, 160, 10)) +
6   coord_cartesian(xlim = c(40, 160)) +
7   theme_bw()
```



6.1 样本分布与总体分布的关系

若总体的均值 μ 与标准差 σ 是未知的，我们可以用样本的均值与无偏标准差来估计总计的均值与标准差：

$$\mu = M_{sample} \sigma = s_{n-1} \quad (1)$$

在上式中， M_{sample} 是样本的均值， s_{n-1} 是样本的无偏标准差。

7 重复取样

使用一个样本计算得到的均值估计总体的均值产生的偏差可能较大，如果我们重复取样的过程，抽取多个样本，将多个样本的均值求均值，那么，我们将得到更为准确的估计值。下面，我们从人数 $N = 10000$ 的总体中取出 $m = 1000$ 个样本量 $n = 100$ 的样本。

```

1 # 从总数为 10000 的样本中累计抽取 1000 个样本量为 100 的样本，存入 samples
2 samples <- list()
3 for (index in 1:1000) samples[[index]] <- population10000[sample.int(10000, 100), ]
4 # 查看 samples

```

```

5 length(samples)
6 ## [1] 1000
7 head(samples[[1]])
8 ##      ID  IQ
9 ## 4095 4095  77
10 ## 7241 7241  76
11 ## 2867 2867  98
12 ## 6947 6947 114
13 ## 3949 3949 109
14 ## 1464 1464 133

```

8 1000 个样本的均值的均值

与单个样本（例如：sample1）相比，1000 个实际样本的均值的均值更加接近总体均值。

```

1 # 分别计算 1000 个样本的均值，一共得到 1000 个均值，存入 samples_means
2 samples_means <- sapply(samples, function(x) mean(x$IQ))
3 head(samples_means)
4 ## [1] 97.21 101.42 101.24 101.03 102.31 99.55
5 range(samples_means)
6 ## [1] 95.11 106.23
7 head(samples_means)
8 ## [1] 97.21 101.42 101.24 101.03 102.31 99.55
9
10 # 1000 个实际样本的均值的均值。
11 mean(samples_means)
12 ## [1] 100.0201
13 # 1000 个实际样本的均值的标准差。
14 sum((samples_means-mean(samples_means))^2)/1000
15 ## [1] 2.231855

```

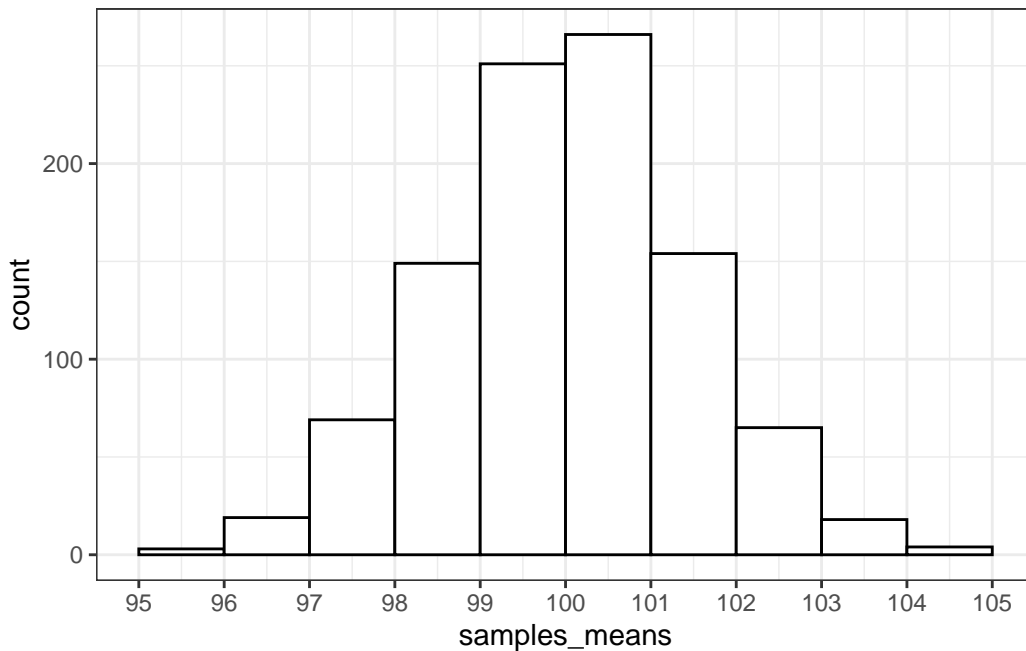
9 抽样分布

抽样分布是 1000 个样本的均值的分布。这个分布中的每一个数据点是一个样本中 100 名被试的 IQ 的均值，而不是一个被试的 IQ 得分，这个分布的中心是是 1000 个样本的均值的均值，这个分布的标准差是 1000 个样本的均值的标准差。

```

1 # 抽样分布。即，1000 个样本的均值的分布。
2 ggplot(data.frame(samples_means), aes(samples_means)) +
3   geom_histogram(breaks = seq(95, 105, 1),
4                 fill = "white",
5                 color = "black") +
6   scale_x_continuous(breaks = seq(95, 105, 1)) +
7   theme_bw()

```



```

1 # + coord_cartesian(xlim = c(40, 160))
2
3 # 抽样分布的标准差。
4 # 即，1000 个均值的标准差。
5 sqrt(sum((samples_means - mean(samples_means)) ^ 2) / 1000)
6 ## [1] 1.493939

```

9.1 抽样分布与总体分布、样本分布的关系

若总体的均值 μ 与标准差 σ 是已知的：

$$\begin{aligned}
 M_{\text{sampling}} &= \mu SD_{\text{sampling}} & &= \frac{\sigma}{\sqrt{n}} & (2)
 \end{aligned}$$

在上式中, n 是样本量。

若总体的均值 μ 与标准差 σ 是未知的, 我们可以用样本的均值与无偏标准差来估计总计的均值与标准差:

$$M_{sampling} = M_{sample} s_{sampling} = \frac{s_{n-1}}{\sqrt{n}} \quad (3)$$

在上式中, M_{sample} 是样本的均值, s_{n-1} 是样本的无偏标准差。

10 代表性

集中趋势的统计指标能够代表一组数据 (样本) 的一般水平。换句话说, 众数、中位数、均值对一组数据具有代表性 (Representativeness)。

11 显著性

我们计算得到 W 研究中 flourish1 的均值为 5.32, 我们可以查到编号为 1787 的被试的幸福感得分为 5.75。那么, 该学生的幸福感是否高于幸福感的一般水平? 尽管表面上该学生的幸福感得分高出样本均值 0.43 分, 但这一差异可能是测量误差导致的。例如, 在填写问卷时, 学生经常在两个相近的选项 (例如: 6 与 7) 上纠结, 一分之差就会造成其幸福感得分的差异, 但学生的幸福感水平实质上是相同的。概括地讲, 统计上关注的“差异”并非“两个数值表面上看起来有差异”, 因为数值上的差异可能是由测量误差与抽样随机性导致的, 这种表面上的差异不能说明两个数值存在实质上的差异。如果两个数值具有实质上的差异, 我们就称两个数值的差异显著 (significant), 或者说, 两个数值的差异具有统计学意义。那么, 判断该学生的幸福感水平与幸福感的一般水平差异显著的标准是什么呢? 是概率。

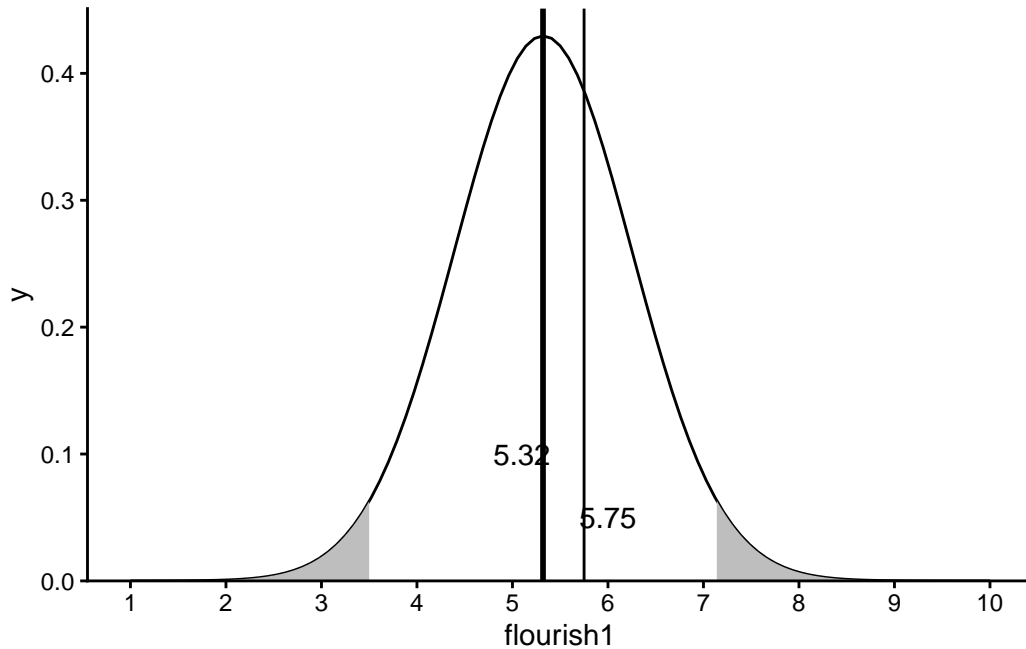
12 差异显著的概率标准

我们在前面的章节学习了频数分布的一般规律: 理想条件下, 数据分布服从正态分布, 左侧低分占少数, 右侧高分占少数, 中间中等分占绝大多数。换句话说, 分数的高低是与其概率相联系的, 极端低分和极端高分是小概率事件。统计上判断一个分数是否极端的标准为概率 $p < 0.05$, p 是 probability 的缩写。意即, 分布在两侧尾部的极端得分 (含左侧极低分与右侧极高分) 占总数的 5%。由于正态分布具有对称性, 因此左侧尾部 (左侧红线以左) 占 2.5%, 右侧尾部 (右侧红线以右) 占 2.5%。当得分落在两侧尾部 5% 的区域时, 我们就说该得分与分布的中心 (分布的均值) 具有显著差异。

```

1 dat_plot <- data.frame(flourish1 = 1:10)
2 dat_ltail <- data.frame(
3   x = seq(1, 3.50, 0.01),
4   y = dnorm(seq(1, 3.50, 0.01),
5             mean = mean(well$flourish1),
6             sd = sd(well$flourish1)))
7 dat_rtail <- data.frame(
8   x = seq(7.14, 10, 0.01),
9   y = dnorm(seq(7.14, 10, 0.01),
10            mean = mean(well$flourish1),
11            sd = sd(well$flourish1)))
12 ggplot(dat_plot, aes(flourish1)) +
13   stat_function(fun = dnorm,
14               args = list(mean = mean(well$flourish1),
15                             sd = sd(well$flourish1))) +
16   scale_x_continuous(breaks = 1:10) +
17   geom_area(data = dat_ltail,
18            mapping = aes(x, y),
19            fill = "grey") +
20   geom_area(data = dat_rtail,
21            mapping = aes(x, y),
22            fill = "grey") +
23   scale_y_continuous(expand = expansion(mult = c(0, 0.05))) +
24   geom_vline(xintercept = 5.32, linewidth = 1) +
25   geom_vline(xintercept = 5.75) +
26   annotate("text", x = 5.1, y = 0.1, label = "5.32") +
27   annotate("text", x = 6, y = 0.05, label = "5.75") +
28   theme_classic()

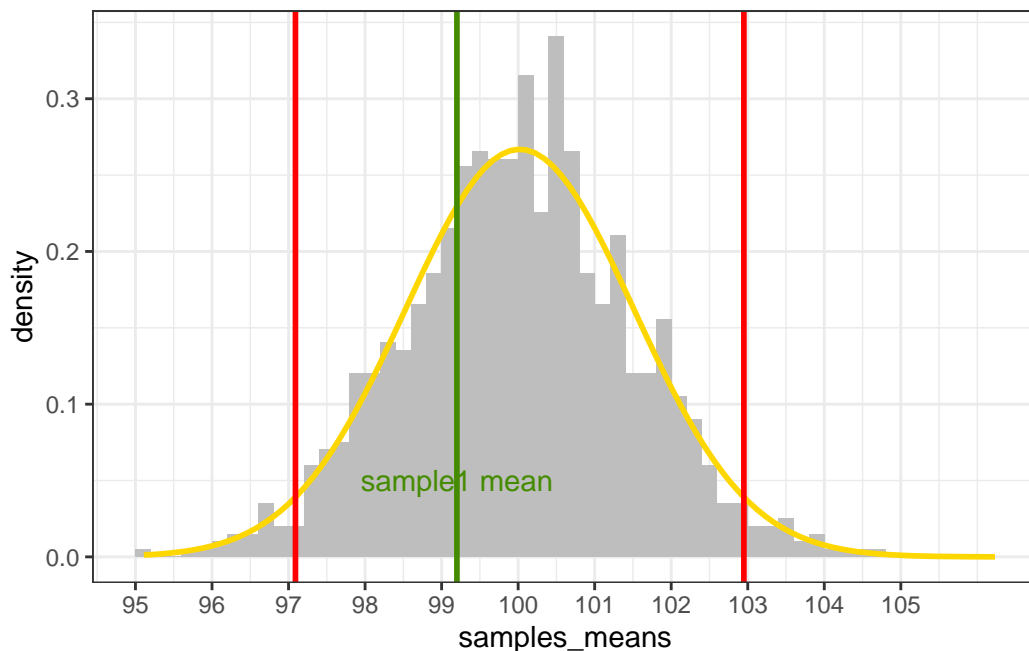
```



```

1 # 抽样分布。即，1000 个样本的均值的分布。
2 ggplot(data.frame(samples_means), aes(samples_means)) +
3   geom_histogram(aes(y = after_stat(density)),
4     breaks = seq(95, 105, 0.2),
5     fill = "grey") +
6   scale_x_continuous(breaks = seq(95, 105, 1)) +
7   stat_function(fun = dnorm,
8     args = list(mean = mean(samples_means), sd = sd(samples_means)),
9     linewidth = 1,
10    color = "gold") +
11   geom_vline(xintercept = mean(samples_means) - qnorm(0.975)*sd(samples_means),
12     color = "red", linewidth = 1) +
13   geom_vline(xintercept = mean(samples_means) + qnorm(0.975)*sd(samples_means),
14     color = "red", linewidth = 1) +
15   geom_vline(xintercept = sample1_mean,
16     color = "chartreuse4", linewidth = 1) +
17   annotate("text", x = sample1_mean, y = 0.05,
18     label = "sample1 mean", color = "chartreuse4") +
19   theme_bw()

```



13 sample1 均值与总体均值的直观比较

sample1 均值为 `sample1_mean`，比总体均值小。但是，这一差异是否显著？我们需要观察 `sample1` 的均值在抽样分布中的位置。上图蓝色线标出了 `sample1` 均值的位置。可见，`sample1` 均值并未落在双侧尾部，故 `sample1` 均值与总体均值没有显著差异。

14 z 分数与单样本 z 检验

理想情况下，抽样分布服从正态分布，正态分布的形状是固定的，横坐标与其概率具有严格的对应关系，只要知道了横坐标，我们就可以计算出这一横坐标所切割出来的尾部的精确概率。由于众多变量（例如：智商、年龄、身高）的单位不同，为了便于计算，我们可以将变量得分转化为 z 分数，将数据的分布转化为标准正态分布，从而计算 z 分数所对应的尾部的概率。 z 分数转换的公式如下，其中 M_x 为 x 的均值， SD_x 为 x 的标准差：

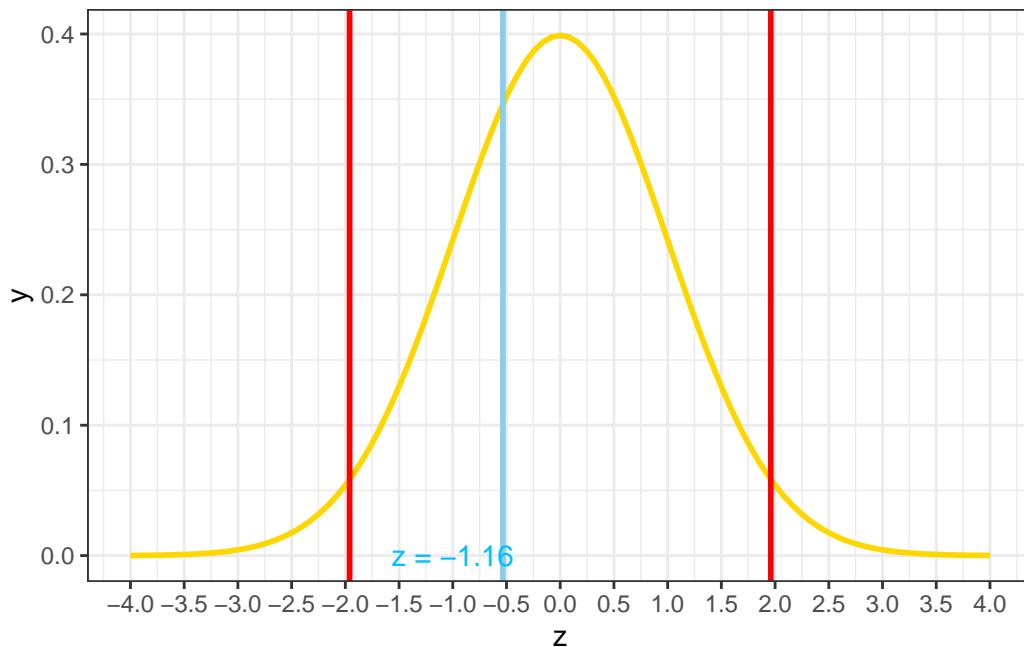
$$z = \frac{x - M_x}{SD_x}$$

转换为 z 分数后， z 分数的均值为 0，标准差为 1。 z 分数的分布如下。

```

1 ggplot(data.frame(z = seq(-4, 4, 0.1)), aes(z)) +
2   scale_x_continuous(breaks = seq(-4, 4, 0.5)) +
3   stat_function(fun = dnorm,
4                 linewidth = 1,
5                 color = "gold") +
6   geom_vline(xintercept = qnorm(0.025),
7              color = "red", linewidth = 1) +
8   geom_vline(xintercept = qnorm(0.975),
9              color = "red", linewidth = 1) +
10  geom_vline(xintercept = (sample1_mean - mu)/(sigma/sqrt(100)),
11             color = "skyblue", linewidth = 1) +
12  annotate("text", x = -1, y = 0,
13          label = "z = -1.16",
14          color = "deepskyblue") +
15  theme_bw()

```



z 分数的均值 (分布的中心) 为 0, z 的取值范围大致为 $[-4, 4]$ 。 z 反映了一个得分在分布中的位置。当 $z < 0$ 时, z 位于分布中心的左侧, 当 $z > 0$ 时, z 位于分布中心的右侧。已知 z , `pnorm()` 函数可以计算累积概率 p (即上图中金色正态曲线与垂直线 z 包围的面积)。 `pnorm()` 默认计算从左到右的累积概率, 即左侧尾部的概率。例如, 当 $z = -1$ 时, 左尾的概率为 0.1586553。另外, 已知 p , `qnorm()` 函数可以计算 z 。

因此, 我们可以将 `sample1` 均值转换为 z 分数, 然后计算 `sample1` 均值在抽样分布中的尾部概率。理想情况下 (例如: 样本量 n 更大, 抽取的样本的数量 m 更大), 抽样分布的均值为 $\mu = 100$, 抽样分布的

标准差为 $\frac{\sigma}{\sqrt{n}} = \frac{15}{\sqrt{100}} = 1.5$ 。抽样分布的标准差通常被称为标准误 (Standard Error), 统计符号为 SE 。样本均值的标准误反映了样本均值估计总体均值的标准误差。

```
1 # 使用总体标准差估计均值的标准误
2 sigma / sqrt(100)
3 ## [1] 1.5
4 # 基于总体的标准差, 对 sample1 的均值进行单样本 z 检验
5 z <- (sample1_mean - mu)/(sigma/sqrt(100))
6 z
7 ## [1] -0.5333333
8 # z 左尾概率
9 p_one_tailed <- pnorm(z)
10 p_one_tailed
11 ## [1] 0.2969014
12 # z 双尾 p 值
13 p_two_tailed <- p_one_tailed*2
14 p_two_tailed
15 ## [1] 0.5938029
```

可得, $z = -1.16$, $p = 0.25$, `sample1` 均值与总体均值 100 没有显著差异。

这里, 我们计算 z 分数, 继而计算 p 值的统计检验方法就是单样本 z 检验。单样本 z 检验的作用是在总体均值、标准差已知的情况下, 比较样本均值与总体均值的差异。

15 t 分数与单样本 t 检验

但是, 总体均值与标准差通常是未知的, 需要用样本的均值与无偏标准差进行估计。按照与单样本 z 检验相同的方法, 我们可以计算出“ z ”及其 p 值。由于样本对总体的估计有偏差, 由样本均值与样本无偏标准差估计得到的抽样分布与基于总体均值与总体标准差估计得到的抽样分布有偏差。因此, 此时得到的“ z ”不是 z , 而是 t 。这一检验就是单样本 t 检验。

```
1 # 使用样本 1 的标准差估计样本 1 均值的标准误
2 sample1_sd_unbiased / sqrt(100)
3 ## [1] 1.520234
4 # 基于 sample1 无偏标准差, 对 sample1 的均值进行单样本 t 检验
5 t <- (sample1_mean - mu)/(sample1_sd_unbiased/sqrt(100))
6 t
7 ## [1] -0.5262348
8 # t 的双尾 p 值
9 p_t <- pt(t, df = 100 - 1)*2
10 p_t
11 ## [1] 0.599902
```

可见，单样本 z 检验计算得到的 z 值、 p 值与单样本 t 检验计算得到的 z 值、 p 值是接近的，统计结论是相同的。

事实上，我们不必手写上述略微复杂的代码来进行单样本 t 检验，`t.test()` 函数可以帮我们做：

```
1 t.test.out <- t.test(sample1$IQ, alternative = "two.sided", mu = 100)
2 print(t.test.out)
3 ##
4 ## One Sample t-test
5 ##
6 ## data: sample1$IQ
7 ## t = -0.52623, df = 99, p-value = 0.5999
8 ## alternative hypothesis: true mean is not equal to 100
9 ## 95 percent confidence interval:
10 ## 96.18353 102.21647
11 ## sample estimates:
12 ## mean of x
13 ## 99.2
```

可见，`t.test()` 检验的结果与我们编辑公式代码计算的结果是一样的。

16 自由度

在上面的 t 检验中， p 值的计算会使用自由度 (degree of freedom, df) 这一统计量。什么是自由度？自由度是数据可以自由变化的程度。在前文中，我们对 `sample1` 均值进行单样本 t 检验，这一检验需要我们首先计算 `sample1` 的均值。当 `sample1` 的均值保持不变时，样本中只有 $n - 1 = 100 - 1 = 99$ 个人的 IQ 值可以自由变化，因此自由度 $df = 99$ 。我们以三个人的数据为例：假设张三、李四、王二的 IQ 的均值为 100，我们在猜测张三、李四的 IQ 得分时可以任意猜测，张三、李四的 IQ 得分可以自由变化，假定张三、李四的 IQ 得分分别为 140、90。那么接下来，王二的 IQ 得分不能任意猜测，王二的 IQ 得分不再自由，因为张三、李四、王二的 IQ 的均值为 100，所以王二的 IQ 得分必须是： $100 * 3 - 140 - 90 = 70$ 。

17 t 分布的自由度

在单样本 t 检验中， $df = n - 1$ ，其中 n 为样本量。样本的样本量可能不同，由样本均值与样本无偏标准差估计得到的抽样分布会随着样本量发生变化，样本量越大，由样本均值与样本无偏标准差估计得到的抽样分布会越接近基于总体均值与总体标准差估计得到的抽样分布。

在下图中，金色曲线为 z 分布，蓝色线由深变浅为自由度为 1、4、9 时的 t 分布。可见，与 z 分布相比，样本量越小 (df 较小)， t 分布的尾部会越厚，这就造成了 z 检验与 t 检验的 p 值的差异。

```
1 ggplot(data.frame(x = seq(-4, 4, 0.1)), aes(x)) +  
2   stat_function(fun = dnorm, color = "gold", linewidth = 2) +  
3   stat_function(fun = dt, args = list(df = 1),  
4               color = "slateblue4", linewidth = 1) +  
5   stat_function(fun = dt, args = list(df = 4),  
6               color = "slateblue3", linewidth = 1) +  
7   stat_function(fun = dt, args = list(df = 9),  
8               color = "slateblue2", linewidth = 1) +  
9   theme_bw()
```

