Week 7: t-test and Regression Basics

Wenhao Jiang

Department of Sociology
New York University

October 21, 2022

t-test

t-test

- A t-test is a statistical test that is used to compare the means of two groups.
- We use t-test in two scenarios:
  - We do not know the population standard deviation σ
  - We know the population σ, but the sample size is smaller than 30

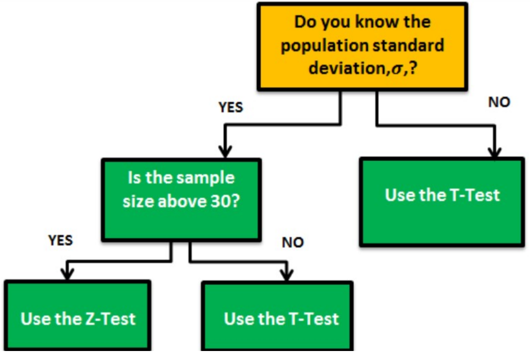


Figure 1: Selection of t-test and z-test

t-test

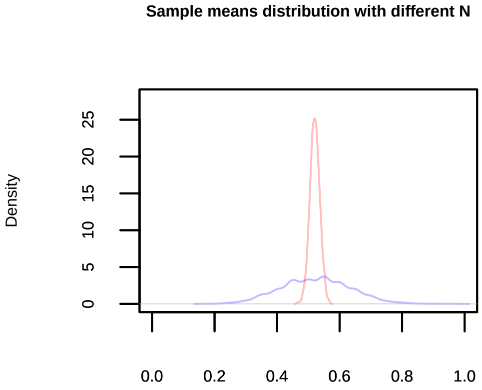
- Why do we care about unknown population σ?
  - The sample estimate of σ using s almost always underestimate σ.

t-test

- Why do we care about unknown population σ?
  - The sample estimate of σ using s almost always underestimate σ.
  - (not required) By construction, $\mathbb{E}[s^2] = \sigma^2$
  - The function $f(x) = \sqrt{x}$ is concave; according to jensen's inequality, $\mathbb{E}[\sqrt{s^2}] \leq \sqrt{\mathbb{E}[s^2]} = \sigma$
  - Therefore, when we calculate the deviation of the observed point estimate from the null hypothesis, we almost always overestimate it.

t-test

- Why do we care about small sample?
  - When the sample size is small (typically $n \leq 30$), the hypothesis normal distribution of sample means is flatter than the one based on larger samples
  - The smaller the sample size, the flatter the distribution of mean values is
  - Look at the $t-$score table



Read Data

- A **data frame** is the **most** common way that we store and interact with data

```
## set working directory
setwd("~/Dropbox/Teaching/SOCUA-302/Week 2")

## read the file
gss <- read.csv("GSS_SOCUA_W2.csv")
```

Subset Data

- We subset the data when we are interested in a smaller **portion** of the data
  - E.g., we are only interested in male/female sample
  - E.g., we are only interested in non-missing data

```
library(dplyr)

## only male sample
male <- gss[gss$sex==1,]

## or using dplyr
male <- gss %>% filter(sex==1)

## symmetrically for women sample
female <- gss[gss$sex==2,]

## or using dplyr
female <- gss %>% filter(sex==2)
```

Subset Data

- Suppose we are interested in testing whether men and women have different number of children on average (mean) in GSS, using t-test.

```r
## only male sample with non-missing
male <- male[male$childs>=0,]

## or using dplyr
male <- male %>% filter(childs>=0)

## symmetrically for women sample
female <- female[female$childs>=0,]

## or using dplyr
female <- female %>% filter(childs>=0)
```

t-test

- Suppose we are interested in testing whether men and women have different number of children on average (mean) in GSS, using t-test.

```
## t-test
t.test(male$childs,
       female$childs)
```

```
##
##  Welch Two Sample t-test
##
## data:  male$childs and female$childs
## t = -20.698, df = 65156, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.3061303 -0.2531672
## sample estimates:
## mean of x mean of y
##  1.769478  2.049126
```

Regression Basics

Basics

- The main aim of regression for now is to find the correlation (or relationship, or association) between two variables
- E.g., we are interested in the correlation between vaccinations per person and deaths per $100k$ across US states.

Basics

- Again, we never observe the true correlation at the population level; we can only estimate the correlation from the sample
- At the **population** level, there is a true **data-generating process** subject to
  - $Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$

Basics

- Again, we never observe the true correlation at the population level; we can only estimate the correlation from the sample
- At the **population** level, there is a true **data-generating process** subject to
  - $Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$
- We estimate the true **data-generating process** by the sample we draw
  - $y_i = \hat{\beta}_0 + \hat{\beta}_1 x_i + e_i$

Basics

- We estimate the true **data-generating process** by the sample we draw
  - $y_i = \hat{\beta}_0 + \hat{\beta}_1 x_i + e_i$
- We estimate $\hat{\beta}_0$ and $\hat{\beta}_1$ by minimizing the total sum of e_i^2, i.e., Ordinary Least Square (OLS)

Basics

- We estimate the true **data-generating process** by the sample we draw
  - $y_i = \hat{\beta}_0 + \hat{\beta}_1 x_i + e_i$
- We estimate $\hat{\beta}_0$ and $\hat{\beta}_1$ by minimizing the total sum of e_i^2, i.e., Ordinary Least Square (OLS)
- $S = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2$
- We use partial derivative to get $\hat{\beta}_0$ and $\hat{\beta}_1 x_i$ that minimizes S

Basics

- $S = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2$
- We use partial derivative to get $\hat{\beta}_0$ and $\hat{\beta}_1 x_i$ that minimizes S

$$
\begin{aligned}
\frac{\partial S}{\partial \hat{\beta}_0} &= \sum_{i=1}^{n} -2(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) \\
&= 0 \\
\sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) &= \sum_{i=1}^{n} y_i - n\hat{\beta}_0 - \sum_{i=1}^{n} \hat{\beta}_1 x_i \\
&= 0 \\
\hat{\beta}_0 &= \frac{\sum_{i=1}^{n} y_i - \sum_{i=1}^{n} \hat{\beta}_1 x_i}{n} \\
&= \bar{y} - \hat{\beta}_1 \bar{x}
\end{aligned}
$$

Basics

- We use partial derivative to get $\hat{\beta}_0$ and $\hat{\beta}_1 x_i$ that minimizes S

$$
\begin{aligned}
\frac{\partial S}{\partial \hat{\beta}_1} = \sum_{i=1}^{n} -2x_i(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) = 0 \\
\sum_{i=1}^{n} x_i(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) = 0 \\
\sum_{i=1}^{n} x_i(y_i - (\bar{y} - \hat{\beta}_1 \bar{x}) - \hat{\beta}_1 x_i) = 0 \\
\sum_{i=1}^{n} (x_i y_i - \bar{y} x_i - \hat{\beta}_1 \bar{x} x_i - \hat{\beta}_1 x_i^2) = 0 \\
\sum_{i=1}^{n} (x_i y_i - \bar{y} x_i) = \hat{\beta}_1 \sum_{i=1}^{n} (x_i^2 - \bar{x} x_i) \\
\hat{\beta}_1 = \frac{\sum_{i=1}^{n} (x_i y_i - \bar{y} x_i)}{\sum_{i=1}^{n} (x_i^2 - \bar{x} x_i)}
\end{aligned}
$$

Basics

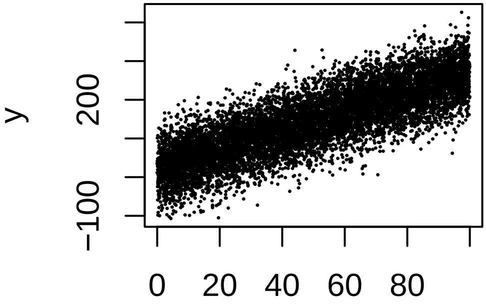
- With a little bit of algebra as I will show

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i y_i - \bar{y} x_i)}{\sum_{i=1}^{n}(x_i^2 - \bar{x} x_i)} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} = \frac{Cov(x_i, y_i)}{Var(x_i)}$$

R Operations

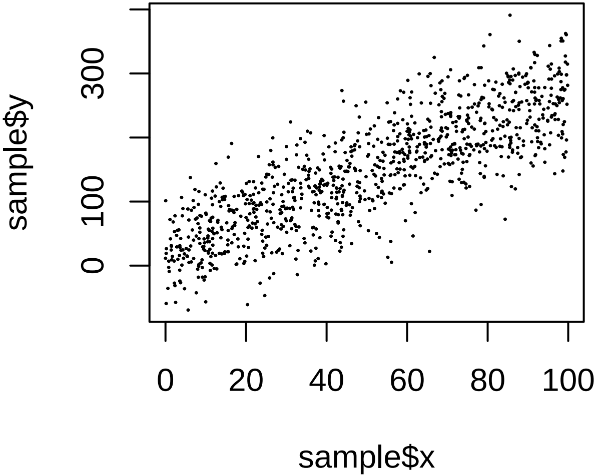
```
## create a population
x <- runif(10000, min=0, max=100)
beta1 <- 2.5
beta0 <- 20
epsilon <- rnorm(10000,mean=0,sd=50)
y <- beta0+beta1*x+epsilon

## plot the population-level correlation
plot(x,y,cex=0.1)
```



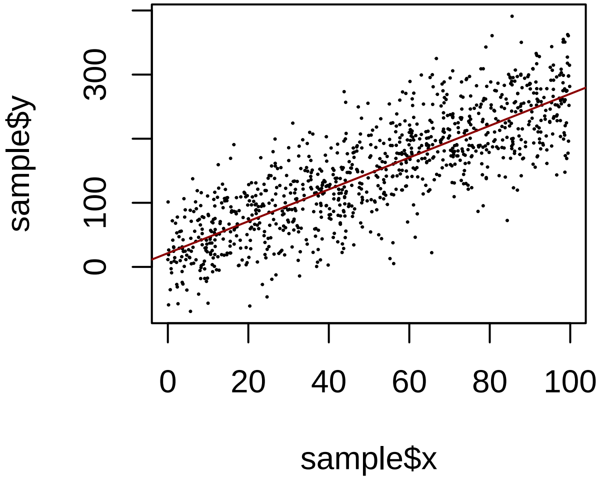
R Operations

- We draw a random sample ($n = 1000$) from the population we created



R Operations

- We fit a regression line to capture the correlation between x and y



R Operations

- How do we get the slope and the intercept from R?

```
model <- lm(y~x,data=sample)
summary(model)
```

```
##
## Call:
## lm(formula = y ~ x, data = sample)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -162.273  -32.980    0.621   33.269  156.925
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 21.46962    3.21311   6.682 3.92e-11 ***
## x            2.48552    0.05467  45.461  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 50.01 on 998 degrees of freedom
## Multiple R-squared:  0.6744, Adjusted R-squared:  0.674
## F-statistic:  2067 on 1 and 998 DF,  p-value: < 2.2e-16
```

R Operations

- Is this the same as our formula?

```
## slope
beta1 <- cov(sample$x,sample$y)/var(sample$x)
beta1
```

```
## [1] 2.485518
```

```
## intercept
mean(sample$y) - beta1*mean(sample$x)
```

```
## [1] 21.46962
```