Social Sciences Intro to Statistics

Week 5.2 Inferential Statistics, about a single variable

Week 5: Learning goal - Formulate hypothesis testing both by hand and with infer commands for a single population mean.

Introduction

Lecture overview:

- Hypothesis testing about single population mean

Load packages:

```
library(tidyverse)
library(ggplot2)
library(labelled)
library(patchwork)

# Load ipeds dataset from course website
load(url('https://raw.githubusercontent.com/bcl96/Social-Sciences-Stats/main/data/ipeds/outpu
```

```
#> Rows: 965
#> Columns: 38
#> $ instnm            <chr> "Alabama A & M University", "University of Alabama a~
#> $ unitid            <dbl> 100654, 100663, 100706, 100724, 100751, 100830, 1008~
#> $ opeid6            <chr> "001002", "001052", "001055", "001005", "001051", "0~
#> $ opeid             <chr> "00100200", "00105200", "00105500", "00100500", "001~
#> $ control           <dbl+lbl> 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 2, 1, 1, 2, ~
#> $ c15basic          <dbl+lbl> 18, 15, 16, 19, 16, 18, 16, 20, 18, 18, 19, 18, ~
#> $ stabbr            <chr+lbl> "AL", "AL", "AL", "AL", "AL", "AL", "AL", "AL", ~
#> $ city              <chr> "Normal", "Birmingham", "Huntsville", "Montgomery", ~
#> $ zip               <chr> "35762", "35294-0110", "35899", "36104-0271", "35487~
#> $ locale            <dbl+lbl> 12, 12, 12, 12, 13, 12, 13, 12, 23, 43, 21, 13, ~
#> $ region            <dbl+lbl> 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, ~
#> $ tuit_grad_res     <dbl> 10128, 8424, 10632, 7416, 11100, 7812, 10386, 15325,~
#> $ fee_grad_res      <dbl> 1414, 0, 1054, 2740, 690, 766, 1784, 900, 1000, 190,~
#> $ tuit_grad_nres    <dbl> 20160, 19962, 24430, 14832, 31460, 17550, 31158, 153~
#> $ fee_grad_nres     <dbl> 1414, 0, 1054, 2740, 690, 766, 1784, 900, 1000, 190,~
#> $ tuit_md_res       <dbl> NA, 31198, NA, NA, 31198, NA, NA, NA, NA, NA, NA, NA~
#> $ fee_md_res        <dbl> NA, 3464, NA, NA, 0, NA, NA, NA, NA, NA, NA, NA, NA,~
#> $ tuit_md_nres      <dbl> NA, 62714, NA, NA, 62714, NA, NA, NA, NA, NA, NA, NA~
#> $ fee_md_nres       <dbl> NA, 3464, NA, NA, 0, NA, NA, NA, NA, NA, NA, NA, NA,~
#> $ tuit_law_res      <dbl> NA, NA, NA, NA, 24080, NA, NA, 39000, NA, NA, NA, NA~
#> $ fee_law_res       <dbl> NA, NA, NA, NA, 300, NA, NA, 325, NA, NA, NA, NA, 65~
#> $ tuit_law_nres     <dbl> NA, NA, NA, NA, 44470, NA, NA, 39000, NA, NA, NA, NA~
#> $ fee_law_nres      <dbl> NA, NA, NA, NA, 300, NA, NA, 325, NA, NA, NA, NA, 65~
#> $ books_supplies    <dbl> 1600, 1200, 2416, 1600, 800, 1200, 1200, 1800, 998, ~
#> $ roomboard_off     <dbl> 9520, 14330, 11122, 7320, 14426, 10485, 14998, 8020,~
#> $ oth_expense_off   <dbl> 3090, 6007, 4462, 5130, 4858, 4030, 6028, 4600, 3318~
#> $ tuitfee_grad_res  <dbl> 11542, 8424, 11686, 10156, 11790, 8578, 12170, 16225~
#> $ tuitfee_grad_nres <dbl> 21574, 19962, 25484, 17572, 32150, 18316, 32942, 162~
#> $ tuitfee_md_res    <dbl> NA, 34662, NA, NA, 31198, NA, NA, NA, NA, NA, NA, NA~
#> $ tuitfee_md_nres   <dbl> NA, 66178, NA, NA, 62714, NA, NA, NA, NA, NA, NA, NA~
#> $ tuitfee_law_res   <dbl> NA, NA, NA, NA, 24380, NA, NA, 39325, NA, NA, NA, NA~
#> $ tuitfee_law_nres  <dbl> NA, NA, NA, NA, 44770, NA, NA, 39325, NA, NA, NA, NA~
#> $ coa_grad_res      <dbl> 25752, 29961, 29686, 24206, 31874, 24293, 34396, 306~
#> $ coa_grad_nres     <dbl> 35784, 41499, 43484, 31622, 52234, 34031, 55168, 306~
#> $ coa_md_res        <dbl> NA, 56199, NA, NA, 51282, NA, NA, NA, NA, NA, NA, NA~
#> $ coa_md_nres       <dbl> NA, 87715, NA, NA, 82798, NA, NA, NA, NA, NA, NA, NA~
#> $ coa_law_res       <dbl> NA, NA, NA, NA, 44464, NA, NA, 53745, NA, NA, NA, NA~
#> $ coa_law_nres      <dbl> NA, NA, NA, NA, 64854, NA, NA, 53745, NA, NA, NA, NA~
#> Rows: 200
#> Columns: 4
#> $ norm_dist    <dbl> 42.70513, 50.24400, 61.29008, 45.47494, 44.74406, 47.9912~
#> $ rskew_dist   <dbl> 0.34451771, 0.31359906, 0.09375337, 0.05581678, 0.0744584~
#> $ lskew_dist   <dbl> 0.6554823, 0.6864009, 0.9062466, 0.9441832, 0.9255415, 0.~
#> $ stdnorm_dist <dbl> -1.45897348, 0.04880097, 2.25801577, -0.90501164, -1.0511~
#> [1] 32528.35
#> [1] 31620.8
```

Assumptions

All statistical tests (based on some statistical analysis) depends on “assumptions”

- *if* the researcher is confident that the assumptions have been satisfied, then the researcher can make inferences about the population parameter by applying the relevant statistical analysis/test to sample data
- if we are concerned that one or more assumptions have not been satisfied, then the researcher should not make inferences about the population parameter

Assumptions necessary for testing hypotheses about a population mean

1. sample is a random sample from population
2. population distribution of variable is normal

"Robust"

- A statistical method is robust with respect to a particular assumption, when it performs adequately even when that assumption is violated

Hypothesis tests about a population mean is "robust" to the normal distribution assumption

- Statisticians have shown that hypothesis tests about population means are robust against violations of normal population assumption, especially when sample size > 30
- Why is hypothesis test about population means robust to normal population assumption? Because of central limit theorem

Central limit theorem:

- when sample size is large, the sampling distribution of the sample mean, , is approximately normal, even if the population distribution of the variable is not normal
- If population distribution is normal then sampling distribution is normal for any sample size
- If population distribution is not normal, sample size of about 30 is sufficient

Hypothesis test about population means is not robust to violations of random sampling

- i.e., if you take a non-random sample from the population, you cannot make good predictions about the population

Hypothesis test example, all steps using r

Research question:

- What is the population mean price of full-time nonresident graduate tuition + fees? [variable = `tuitfee_grad_nres`]

Let's imagine we want to test whether the population mean, $\mu_Y = \$17,000$, using a two-sided alternative hypothesis and an alpha level of .05

State null and alternative hypotheses

- Null hypothesis, H_0
  - $H_0 : \mu_Y = \mu_{Y0} = \$17,000$
  - H_0 : population mean price of full-time nonresident graduate tuition + fees is \$17,000
- Alternative hypothesis, H_a
  - $H_0 : \mu_Y \neq \$17,000$

Test statistic and p-value

- $t = \frac{\bar{Y} - \mu_{Y0}}{\hat{\sigma}_{\bar{Y}}}$
- where:
  - $\hat{\sigma}_Y$ refers to sample standard deviation of variable Y
  - n refers to sample size
  - sample standard error of the sample mean $= \hat{\sigma}_{\bar{Y}} = \frac{\hat{\sigma}_Y}{\sqrt{n}}$

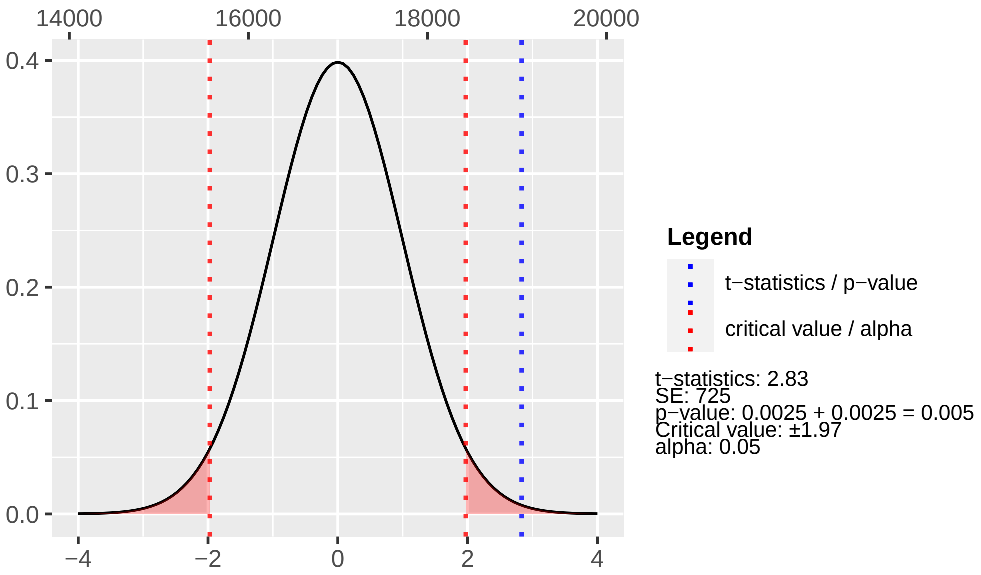
Components of t-test

- sample size, $n = 200$
- sample mean, $\bar{Y} = 1.9053445 \times 10^4$
- Population mean associated with H_0, $\mu_{Y0} = \$17,000$
- sample standard deviation, $\hat{\sigma}_Y = 1.0253114 \times 10^4$
- sample standard error of the sample mean, $\hat{\sigma}_{\bar{Y}} = 725.0046$

Calculating t-test p-value using `t.test()`

- $t = 2.83$
- p-value $= Pr(obs > t) + Pr(obs < -t) = 0.005$
  - $Pr(obs > t) = 0.0025$
  - $Pr(obs < -t) = 0.0025$
- below code chunk runs t-test and plots t-value against sampling distribution assuming H_0 is true, using alpha of .05

```
t.test(df_ipeds_sample$tuitfee_grad_nres, mu = 17000)
#> 
#>  One Sample t-test
#> 
#> data:  df_ipeds_sample$tuitfee_grad_nres
#> t = 2.8323, df = 199, p-value = 0.005096
#> alternative hypothesis: true mean is not equal to 17000
#> 95 percent confidence interval:
#>  17623.77 20483.12
#> sample estimates:
#> mean of x 
#>  19053.44
plot_t_distribution(df_ipeds_sample$tuitfee_grad_nres, mu = 17000, alpha = .05,
  shade_rejection = TRUE, shade_pval = FALSE)
```



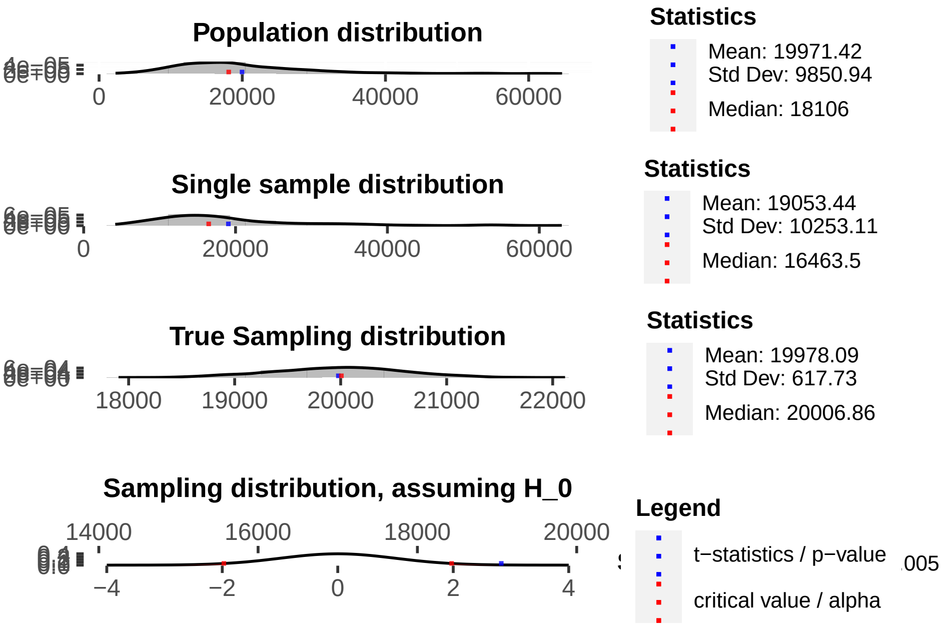
Conclusion

- Because the p-value of 0.005 is less than the alpha level of `.05`, we reject H_0
- we reject the null hypothesis H_0, population mean price of full-time nonresident graduate tuition + fees, μ_Y, is equal to $17,000$
- We can also say that μ_Y is greater than $17,000$

Finally, we usually don't have all data on the population. But since we do for IPEDS, we can plot:

- the population distribution (usually unknown)
- on top of the distribution from our single random sample
- on top of the sampling distribution (usually unknown)
- on top of the sampling distribution assuming H_0 is true

```
plot_distribution(df_ipeds_pop$tuitfee_grad_nres, plot_title = 'Population distribution') +
  plot_distribution(df_ipeds_sample$tuitfee_grad_nres, plot_title = 'Single sample distribut
  plot_distribution(get_sampling_distribution(df_ipeds_pop$tuitfee_grad_nres), plot_title =
  plot_t_distribution(df_ipeds_sample$tuitfee_grad_nres, mu = 17000, plot_title = 'Sampling
  plot_layout(ncol = 1)
```


Population distribution
Statistics
Mean: 19971.42
Std Dev: 9850.94
Median: 18106
Single sample distribution
Statistics
Mean: 19053.44
Std Dev: 10253.11
Median: 16463.5
True Sampling distribution
Statistics
Mean: 19978.09
Std Dev: 617.73
Median: 20006.86
Sampling distribution, assuming H_0
Legend
t−statistics / p−value
critical value / alpha