Statistical Methods for Data Science
CS 6313.001: Mini Project #4

Due on Thursday March 30, 2017 at 4pm

Instructor: Pankaj Choudhary

Hanlin He / Lizhong Zhang (hxh160630 / lxz160730)

Contents

Contribution

Both team members made the same contribution in this project.

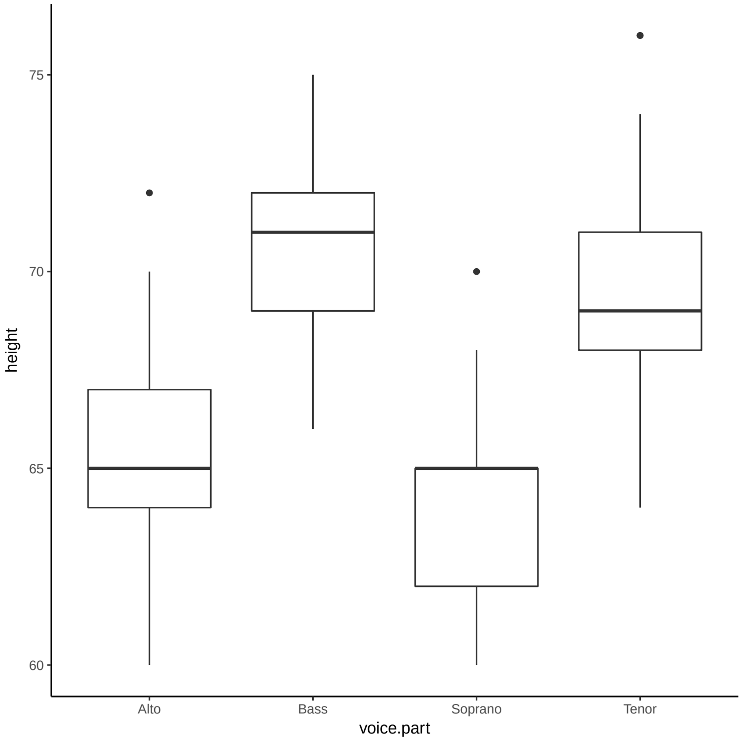
Section 1 Answers

Problem 1.1 Bass to Soprano

(a) Exploratory Analysis

The boxplot is shown in fig. 1.

Figure 1: Boxplot



We plotted the four groups side by side on a box plot. The box plots show that the means and medians of four groups are almost different, so there is no any two distributions seemed similar.

(b) Test The Hypotheses

Based on information given:

- The null hypothesis $H_0 : \mu_x = \mu_y$, Bass singers' heights is same to Tenor singers'.
- The alternative hypothesis $H_1 : \mu_x > \mu_y$, Bass singers are taller than Tenor singers.

Assume 5% level of significance,

$$t_{obs} = \frac{(mean_{Bass} - mean_{Tenor})}{\sqrt{\frac{sd^2_{Bass}}{nums_{bass}} + \frac{sd^2_{Tenor}}{nums_{Tenor}}}}$$

$$p - value = 1 - F(t_{obs}) = 0.001591409 < 0.05$$

Thus, reject H_0. On average Bass singers are taller than Tenor singers.

(c) Comparison

From (b), we know that Bass singers are taller than Tenor singers. In the meantime, by observing box plots in (a), we can see that the mean, median and the range of values is higher for Bass singer than Tenor singer, which makes me conclude that Bass singers are taller than Tenor singer on average.

Problem 1.2 Test Hypothesis

(a) Set up the null and alternative hypotheses

Assume the following:

$$\boxed{H_0 : \mu = 10 \quad \text{vs} \quad H_1 : \mu > 10}$$

(b) Which test would you use? What is the test statistic? What is the null distribution of the test statistic?

We would use the *t test*, because we do not know the standard deviation of the distribution.

The test statistic is

$$T = \frac{\overline{x} - \mu_0}{\sigma / \sqrt{n}} \sim t_{n-1}$$

(c) Compute the observed value of the test statistic.

$$t_{obs} = \frac{9.02 - 10}{2.22/\sqrt{20}} = -1.974186$$

(d) Compute the p-value of the test using the usual way.

$$p - value = 1 - pt(-1.974186,\ 20 - 1) = 0.9684606$$

(e) Estimate the p-value of the test using Monte Carlo simulation. How do your answers in (d) and (e) compare?

We made simulation using t-distribution 10000 times. The R code will be shown in section 2. And we get $p - value = 0.97$. Comparing the result in (d) and (e), we find two result is almost same.

(f) State your conclusion at 5% level of significance.

From (d) and (e) result, we know that $p - value > 0.05$, so accept H_0, the mean of the population is not greater than 10.

Problem 1.3 Credit Rating

(a) Construct an appropriate 95% confidence interval

To find the 95% confidence interval for the difference in mean credit limits of all credit issued in January 2011 and in May 2011, we know the large number of samples (400 and 500) should let us assume either a normal distribution or normal distribution for the difference between the credit cards.

An alpha value of 0.05 was used to find the quantile as shown in the R code below. The sample mean and standard deviation were used. The 95% confidence interval is: $CI = [201.1711, \ 302.8289]$, so based on the confidence interval, we know that the credit limits on newly issued credit cards increased between January 2011 and May 2011.

(b) Perform an appropriate 5% level test

Assume the following:

- $H_0 : \mu_x = \mu_y$: Not greater
- $H_1 : \mu_x < \mu_y$: The mean credit limit of all credit cards issued in May 2011 is greater than the same in January 2011.

Use large sample:

$$z = \frac{2887 - 2635}{\sqrt{\frac{365^2}{400} + \frac{412^2}{500}}} = 9.717132$$

$$p - value = 1 - pnorm(9.717132) = 0 < 0.05$$

So, reject H_0, the mean credit limit of all credit cards issued in May 2011 is greater than the same in January 2011.

Section 2 R Code

```
################################################################
# R code for exercise 1
################################################################

# Read singer.txt file.
mydata <- read.table("singer.txt", header = TRUE, sep = ",")

# Extract data by different voice part.
```

```
mydata_Bass = subset(mydata, voice.part == "Bass")
mydata_Tenor = subset(mydata, voice.part == "Tenor")
mydata_Alto = subset(mydata, voice.part == "Alto")
mydata_Soprano = subset(mydata, voice.part == "Soprano")

# Use ggplot to draw boxplots.
pdf("boxplot.pdf")
library(ggplot2)
ggplot(mydata, aes(x=voice.part, y=height)) + geom_boxplot() + theme_classic()
dev.off()
```

```
## pdf
##   2
```

```
# Calculate numbers, means and standard deviations of Bass singers
# and Tenor singers.
nums_Bass <- nrow(mydata_Bass)
mean_Bass <- mean(mydata_Bass[, 1])
sd_Bass <- sd(mydata_Bass[, 1])

nums_Tenor <- nrow(mydata_Tenor)
mean_Tenor <- mean(mydata_Tenor[, 1])
sd_Tenor <- sd(mydata_Tenor[, 1])

# Calculate nu.
i <- (sd_Bass^2) / nums_Bass + (sd_Tenor)^2 / nums_Tenor
j <- (sd_Bass^4) / ((nums_Bass^2)*(nums_Bass - 1))
k <- (sd_Tenor^4) / ((nums_Tenor^2)*(nums_Tenor - 1))
nu <- i / (j + k)

# Calculate t_obs.
t_obs <- (mean_Bass - mean_Tenor) /
    sqrt(sd_Bass^2 / nums_Bass + sd_Tenor^2 / nums_Tenor)

# Calculate p-value.
p_value <- 1 - pt(t_obs, nu)

##############################################################
# R code for exercise 2
##############################################################

# Calculate test statistic.
t_obs1 <- (9.02 - 10) / (2.22 / sqrt(20))
p1_value <- 1 - pt(t_obs1, 20 - 1)
```

```r
# Estimate the p-value of the test using Monte Carlo simulation
time <- 10000
est_value <- sum(rt(time, 20-1) > t_obs1) / time

##############################################################
# R code for exercise 3
##############################################################

# Calculate CI.
##CI <- mean(y) - mean(x) + c(-1, 1) * qt(1-(alpha/2), nu) *
##      sqrt(sd(x)^2 / n_1 + sd(y)^2 / n_2)
##CI <- mean(y) - mean(x) + c(-1, 1) * qnorm(1-(alpha/2)) *
##      sqrt(sd(x)^2 / n_1 + sd(y)^2 / n_2)
CI <- 2887 - 2635 + c(-1, 1) * qnorm(1 - (0.05/2)) *
    sqrt(365^2 / 400 + 412^2 / 500)

# Calculate z.
z_1 <- (2887 - 2635) /sqrt(365^2 / 400 + 412^2 / 500)
# Calculate p-value
p2_value <- 1 - pnorm(z_1)
```