

021 - Introduction to Regression

EPIB 607

Sahir Rai Bhatnagar

Department of Epidemiology, Biostatistics, and Occupational Health
McGill University

`sahir.bhatnagar@mcgill.ca`

slides compiled on November 1, 2021



Introduction to parameter-contrasts

- We started the course by talking about the case where there were no determinants, i.e., no subpopulations \rightarrow there was one global parameter (μ, π, λ) .

Introduction to parameter-contrasts

- We started the course by talking about the case where there were no determinants, i.e., no subpopulations \rightarrow there was one global parameter (μ, π, λ) .
- Now we concern ourselves with determinants of the global parameter.
For example:
 - ▶ μ_{north} vs. μ_{south}
 - ▶ π_{north} vs. π_{south}
 - ▶ λ_{north} vs. λ_{south}
- Today we introduce population parameter contrasts in a regression framework

Why regression for parameter-contrasts?

- Why do we start in a regression framework (as opposed to two-sample inference in DVB)?
- **Parameter contrasts are a special case of regression**

What is regression?

- How **parameters** relate to its determinants
- How to link the parameters between the different populations through generic equations, that looks like a regression equation.
- Then once you get data, you can actually fit or get your best estimates of those parameters

Linear regression: The Concept

- A regression model is said to be **linear** when it is of the form

$$\begin{aligned}\mu &= \mu_0 + \sum_{j=1}^p \beta_j X_j \\ &= \mu_0 + \beta_1 X_1 + \beta_1 X_1 + \cdots + \beta_p X_p\end{aligned}$$

- Which means that the value of the mean (μ) is viewed as a linear combination of the parameters $\mu_0, \beta_1, \beta_2, \dots, \beta_p$, the coefficients of the linear combination being the realizations for the X 's

Linear regression: Example

- Consider the depths of the ocean example
- Here, μ designates the true mean depth of the ocean
- For this parameter, one might consider the determinant
 - ▶ X which is an indicator variable defined by

$$X = \begin{cases} 1 & \text{if Southern hemisphere} \\ 0 & \text{if Northern hemisphere} \end{cases}$$

Linear regression: Example

- The model might be taken as

$$\mu_X = \mu_0 + \beta_1 \cdot X$$

and provides the mean depth of the ocean given X

- The subscript X indicates that μ depends on the value of x
- The mean depth of the ocean μ_X is a linear combination of μ_0 and β_1
- If we had an infinite amount of data, the mean depth of the ocean would be determined by hemisphere:

$$\mu_X = \begin{cases} \mu_0 + \beta_1 & \text{if Southern hemisphere} \\ \mu_0 & \text{if Northern Hemisphere} \end{cases}$$

Depths of the ocean: North vs. South Hemisphere

```
# load function to get depths
source("https://raw.githubusercontent.com/sahirbhatnagar/EPIB607/master/inst/labs/
003-ocean-depths/automate_water_task.R")

# get 1000 depths
set.seed(222333444)
depths <- automate_water_task(index = sample(1:50000, 1000),
student_id = 222333444, type = "depth")

# separate by north and south hemisphere
depths_north <- depths[which(depths$lat>0),]
depths_south <- depths[which(depths$lat<0),]

# restrict sample to 200 (at random)
depths_north <- depths_north[sample(1:nrow(depths_north), 200), ]
depths_south <- depths_south[sample(1:nrow(depths_south), 200), ]

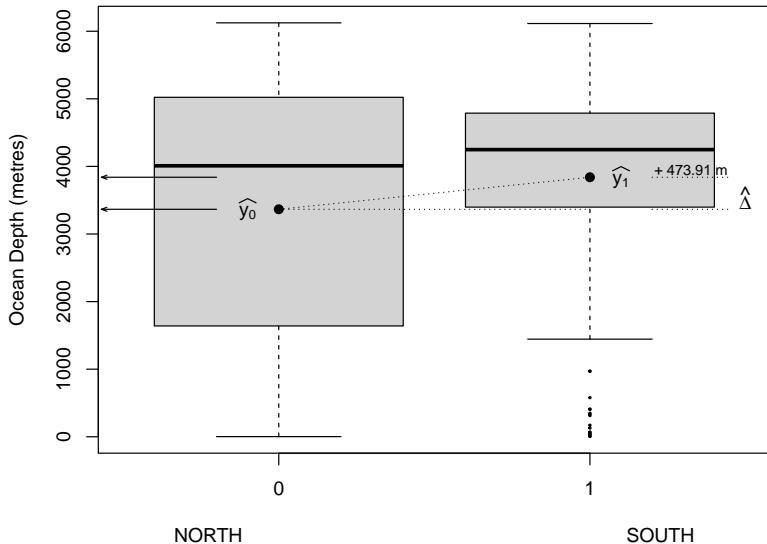
# add indicator variable
depths_north$South <- 0
depths_south$South <- 1

# combine data
depths <- rbind(depths_north, depths_south)
head(depths)

# calculate mean and sd by hemisphere
mean.sd <- depths %>% group_by(South) %>%
  summarise(means = mean(alt), sds = sd(alt))

means <- mean.sd$means
sds <- mean.sd$sds
```

Depths of the ocean: North vs. South Hemisphere



Standard error of the mean difference

To perform inference we first need to calculate the SE of the mean difference given by:

$$SE_{\bar{y}_1 - \bar{y}_0} = \sqrt{\frac{s_0^2}{n_0} + \frac{s_1^2}{n_1}} \quad (1)$$

Standard error of the mean difference

To perform inference we first need to calculate the SE of the mean difference given by:

$$SE_{\bar{y}_1 - \bar{y}_0} = \sqrt{\frac{s_0^2}{n_0} + \frac{s_1^2}{n_1}} \quad (1)$$

```
n0 <- nrow(depths_north)
n1 <- nrow(depths_south)

mean0 <- mean(depths_north$salt)
mean1 <- mean(depths_south$salt)

var0 <- var(depths_north$salt)
var1 <- var(depths_south$salt)

(SEM <- sqrt(var0/n0 + var1/n1))

## [1] 171.4861
```

95% Confidence Interval for the Mean Difference

We can then calculate a 95% CI for the mean difference given by:

$$(\bar{y}_1 - \bar{y}_0) \pm t_{(n_0+n_1-2)}^* \times SE_{\bar{y}_1 - \bar{y}_0} \quad (2)$$

95% Confidence Interval for the Mean Difference

We can then calculate a 95% CI for the mean difference given by:

$$(\bar{y}_1 - \bar{y}_0) \pm t_{(n_0 + n_1 - 2)}^* \times SE_{\bar{y}_1 - \bar{y}_0} \quad (2)$$

```
# assuming equal variances
(mean1 - mean0) + qt(c(0.025, 0.975), df = n0 + n1 - 2) * SEM

## [1] 136.7782 811.0418

# similar to z interval
qnorm(c(0.025, 0.975), mean = mean1 - mean0, sd = SEM)

## [1] 137.8034 810.0166
```


Parameter contrasts with regression

Using the `lm` function in R:

```
# regression. lm assumes equal variances
fit <- lm(alt ~ South, data = depths)
summary(fit)

## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3365.6      121.3   27.755 < 2e-16 ***
## South         473.9       171.5    2.764  0.00598 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1715 on 398 degrees of freedom
## Multiple R-squared:  0.01883, Adjusted R-squared:  0.01636
## F-statistic: 7.637 on 1 and 398 DF,  p-value: 0.005983
```

Confidence interval from regression fit

```
confint(fit)
```

```
##              2.5 %    97.5 %  
## (Intercept) 3127.2068 3603.9832  
## South       136.7782  811.0418
```

Unequal variances using `stats::t.test`

`stats::t.test` assumes unequal variances by default:

```
stats::t.test(alt ~ South, data = depths, var.equal = FALSE)

## Welch Two Sample t-test with alt by South
## t = -2.7635, df = 356.262, p-value = 0.006015
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -811.1623 -136.6577
## sample estimates:
## mean in group 0 mean in group 1
##      3365.595      3839.505

(mean0 - mean1) + qt(c(0.025, 0.975), df = 349.61783) * SEM

## [1] -811.1841 -136.6359
```

Equal variances using `stats::t.test`

We can specify equal variance assumption in `stats::t.test`:

```
stats::t.test(alt ~ South, data = depths, var.equal = TRUE)

## Two Sample t-test with alt by South
## t = -2.7635, df = 398, p-value = 0.005983
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal
## 95 percent confidence interval:
## -811.0418 -136.7782
## sample estimates:
## mean in group 0 mean in group 1
## 3365.595 3839.505

(mean0 - mean1) + qt(c(0.025, 0.975), df = n0 + n1 - 2) * SEM

## [1] -811.0418 -136.7782
```

Session Info

```
R version 4.1.1 (2021-08-10)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Pop!_OS 21.04

Matrix products: default
BLAS:   /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/libopenblaspr0.3.13.so

attached base packages:
[1] tools      stats      graphics  grDevices  utils      datasets  methods
[8] base

other attached packages:
[1] DT_0.16 mosaic_1.7.0 Matrix_1.3-2 mosaicData_0.20.1
[5] ggformula_0.9.4 ggstance_0.3.4 lattice_0.20-41 kableExtra_1.2.1
[9] socviz_1.2 gapminder_0.3.0 here_0.1 NCStats_0.4.7
[13] FSA_0.8.30 forcats_0.5.1 stringr_1.4.0 dplyr_1.0.7
[17] purrr_0.3.4 readr_1.4.0 tidyr_1.1.4 tibble_3.1.5
[21] ggplot2_3.3.5 tidyverse_1.3.0 knitr_1.36

loaded via a namespace (and not attached):
[1] fs_1.5.0 lubridate_1.7.9 webshot_0.5.2 httr_1.4.2
[5] rprojroot_2.0.2 backports_1.2.1 utf8_1.2.2 R6_2.5.1
[9] DBI_1.1.1 colorspace_2.0-2 withr_2.4.2 tidyselect_1.1.1
[13] gridExtra_2.3 leaflet_2.0.3 curl_4.3.2 compiler_4.1.1
[17] cli_3.0.1 rvest_1.0.0 pacman_0.5.1 xml2_1.3.2
[21] ggdendro_0.1.22 mosaicCore_0.8.0 scales_1.1.1 digest_0.6.28
[25] foreign_0.8-81 rmarkdown_2.11.3 rio_0.5.16 pkgconfig_2.0.3
[29] htmltools_0.5.2 highr_0.9 dbplyr_1.4.4 fastmap_1.1.0
[33] htmlwidgets_1.5.3 rlang_0.4.12 readxl_1.3.1 rstudioapi_0.13
[37] farver_2.1.0 generics_0.1.0 jsonlite_1.7.2 crosstalk_1.1.1
[41] zip_2.2.0 car_3.0-9 magrittr_2.0.1 Rcpp_1.0.7
[45] munsell_0.5.0 fansi_0.5.0 abind_1.4-5 lifecycle_1.0.1
[49] stringi_1.7.5 carData_3.0-4 MASS_7.3-53.1 plyr_1.8.6
[53] grid_4.1.1 blob_1.2.1 ggrepel_0.8.2 crayon_1.4.1
[57] cowplot_1.1.0 haven_2.3.1 splines_4.1.1 hms_1.1.1
[61] pillar_1.6.4 reprex_0.3.0 glue_1.4.2 evaluate_0.14
[65] data.table_1.14.2 modelr_0.1.8 vctrs_0.3.8 tweenr_1.0.1
[69] cellranger_1.1.0 gtable_0.3.0 polyclip_1.10-0 assertthat_0.2.1
[73] TeachingDemos_2.12 xfun_0.26 ggforce_0.3.2 openxlsx_4.1.5
[77] broom_0.7.9 tidyslice_0.4.0 ellipsis_0.3.2
```