STA 522 Project 1 (Survey of Textbooks)

Arpita Mandan, Daniel Truver

3/5/2018

(1) Sampling Design

We first used stratified sampling, where the strata are the departments that we considered to be natural science departments. These included: Biochem, Biology, CMB, Chem, CompSci, Evanth, Math, MGM, Neurobio, Neurosci, Physics, Psy, and Stat. We used stratified sampling because we wanted to sample courses from every department, so that our estimator has a smaller variance than if we did a simple random sample, and is therefore more precise.

Then using simple random sampling, we sampled courses from each stratum. We used the optimum allocation formula from class to determine the sample size within each stratum for all but four departments. For Biochem, CMB, MGM, and Neurobio that have fewer than ten courses, we enumerated all of them.

Our sampling frame has 358 courses, with 63 courses in the final sample.

(2) Data Collection

From the Duke University natural sciences website, https://trinity.duke.edu/natural-sciences, we obtained which courses are considered natural sciences. We obtained the required course numbers via web scraping of the Duke University bookstore. See web scraping Appendix. We then entered values for the sampled courses by hand from the bookstore website. As requested, we did not use courses numbered 700 and above.

The basic test for whether or not a course made the frame was whether its course name included "biology," "chemistry," "computer science," "evolutionary antrhopoly," "mathematics," "physics," "psychology," "neuroscience," or "statistical science." For example, we did not include "info science", but did include "biology" and "neurobiology." Some course names, such as "biostatistics," have no courses numbered below 700; we excluded them.

There are several types course we did not deem to be natural sciences. We did not include any type of engineering. They are not listed on the natural sciences webpage, and they just hit things with wrenches mostly. The Nicholas School of the Environment also does not appear; their federal grants are probably getting cut anyway. Sociology is absent, and we hold the reasons for this to be self-evident. Despite the opinions of a vocal minority, "old testament studies" also do not appear in our sample.

We acknowledge these human judgments are a potential source of selection bias, but we took care to include all relevant courses.

(3) Survey Values

```r
load("surveyData.Rdata")
load("classes.Rdata")
library(survey)
```

```r
fpcvar = c() #bad practice, but data small enough to allow for it
wtvar = c()

N_h = c() #construct number of classes per stratum
for (school in names(classes.df)){
  N_h[school] = sum(classes.df[,school] != "")
}
n_h = c()
```

```
for (school in names(classes.df)){
  n_h[school] = sum(surveyData$department == school)
}

for (dep in names(n_h)){
  fpcvar = c(fpcvar, rep(N_h[dep], n_h[dep])) # grow vector of fpc's
  wtvar = c(wtvar, rep(N_h[dep]/n_h[dep], n_h[dep]))
}

des = svydesign(~1, strata = ~department, weights = wtvar, fpc = fpcvar, data = surveyData)
```

```
sum_new = svytotal(~newprice, des)
confint_sum_new = confint(sum_new)
sum_used = svytotal(~usedprice, des)
confint_sum_used = confint(sum_used)
```

Table 1: Total Cost of Textbooks by Class

	Mean	2.5%	97.5%
New Price	16412.75	9788.58	23036.92
Used Price	12322.22	7350.10	17294.33

```
mean_books = svymean(~numtexts, des)
confint_mean_books = confint(mean_books)
```

Table 2: Average number of Textbooks

	Mean	2.5%	97.5%
Books	0.34	0.22	0.45

```
mean_new = svymean(~newprice, des)
confint_new = confint(mean_new)
mean_used = svymean(~usedprice, des)
confint_used = confint(mean_used)
```

Table 3: Mean Cost of Textbooks by Class

	Mean	2.5%	97.5%
New Price	45.85	27.34	64.35
Used Price	34.42	20.53	48.31

Appendix (web scraping)

The Appendix is best viewed in the electronic document as many url strings will leave the page.

All files can be found at https://github.com/truverdj/SurveyDesign_Project1

```
library(readr)
library(rvest)
library(dplyr)
library(stringr)

load("classes.Rdata")


N_h = c()
for (school in names(classes.df)){
  N_h[school] = sum(classes.df[,school] != "")
}
n = 50
N = sum(N_h)
n_h = round(n* (N_h/N) )
n_h[n_h == 0] = 1
n = sum(n_h)

load("sample.Rdata")

make_store_url = function(department, number){
  url_part1 = "http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&this_category=1&term=SP18
  url_part2 = department
  url_part3 = "&course="
  url_part4 = as.character(number)
  url_part5 = "&colspan=3&cellspacing=1&cellpadding=0&campus=MAIN&border=0&bgcolor=%23cccccc&action=list
  store_url = paste(url_part1, url_part2, url_part3, url_part4, url_part5, sep = "")
  return(store_url)
}

x = rep(NA, n)
for (i in seq_along(x)){
  x[i] = make_store_url(sample.df[i,"department"], sample.df[i, "course"])
}
# now begins the suffering, scraping too dificult here
num_book = c()
new_cost = c()
use_cost = c()

if (!file.exists("sample.Rdata")){
  department = c()
  for (i in seq_along(n_h)) {
    department = c(department, rep(names(n_h[i]), n_h[i]))
  }
  course = c()
  set.seed(2018)
  for (school in names(classes.df)){
    course = c(course, sample(x = as.character(classes.df[,school] %>% .[.!=""]), n_h[school]))
  }
  sample.df = data.frame(department, course)
  save(sample.df, file = "sample.Rdata")
}


if (!file.exists("classes.Rdata")){
  make_store_url = function(department, number){
  url_part1 = "http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&this_category=1&term=SP1
  url_part2 = department
  url_part3 = "&course="
  url_part4 = as.character(number)
  url_part5 = "&colspan=3&cellspacing=1&cellpadding=0&campus=MAIN&border=0&bgcolor=%23cccccc&action=lis
  store_url = paste(url_part1, url_part2, url_part3, url_part4, url_part5, sep = "")
  return(store_url)
}
biol_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&this_category=1&
biol_courses = rep(NA, 200)
for (i in 2:200){
  biol_courses[i-1] = biol_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
}
biol_courses[is.na(biol_courses)] = ""

chem_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&term=SP18&store=
chem_courses = rep(NA, 200)
for (i in 2:200){
  chem_courses[i-1] = chem_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
}
chem_courses[is.na(chem_courses)] = ""

biochem_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&term=SP18&sto
biochem_courses = rep(NA, 200)
for (i in 2:200){
  biochem_courses[i-1] = biochem_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
}
biochem_courses[is.na(biochem_courses)] = ""

biostat_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&term=SP18&sto
biostat_courses = rep(NA, 200)
for (i in 2:200){
  biostat_courses[i-1] = biostat_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
}
biostat_courses[is.na(biostat_courses)] = ""

cell.molec.bio_courses = c("640", rep("", 199))
```

```
cellbio_courses = c("493", "503", "668", rep("", 197))

compsci_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&term=SP18&sto
compsci_course = rep(NA, 200)
for (i in 2:200){
  compsci_courses[i-1] = compsci_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
}
compsci_courses[is.na(compsci_courses)] = ""

evanth_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&term=SP18&store
evanth_courses = rep(NA, 200)
for (i in 2:200){
  evanth_courses[i-1] = evanth_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
}
evanth_courses[is.na(evanth_courses)] = ""

math_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&term=SP18&store=
math_courses = rep(NA, 200)
for (i in 2:200){
  math_courses[i-1] = math_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
}
math_courses[is.na(math_courses)] = ""

neurobio_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&term=SP18&st
neurobio_courses = rep(NA, 200)
for (i in 2:200){
  neurobio_courses[i-1] = neurobio_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
}
neurobio_courses[is.na(neurobio_courses)] = ""

neurosci_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&term=SP18&st
neurosci_courses = rep(NA, 200)
for (i in 2:200){
  neurosci_courses[i-1] = neurosci_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
}
neurosci_courses[is.na(neurosci_courses)] = ""

physics_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&term=SP18&sto
physics_courses = rep(NA, 200)
for (i in 2:200){
  physics_courses[i-1] = physics_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
```

```
}
physics_courses[is.na(physics_courses)] = ""

psy_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&term=SP18&store=3
psy_courses = rep(NA, 200)
for (i in 2:200){
  psy_courses[i-1] = psy_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
}
psy_courses[is.na(psy_courses)] = ""

sta_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&term=SP18&store=3
sta_courses = rep(NA, 200)
for (i in 2:200){
  sta_courses[i-1] = sta_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
}
sta_courses[is.na(sta_courses)] = ""

mgm_page = read_html("http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&term=SP18&store=3
mgm_courses = rep(NA, 200)
for (i in 2:200){
  mgm_courses[i-1] = mgm_page %>%
    html_node(paste0("#course > option:nth-child(",i,")")) %>%
    html_attr("value")
}
mgm_courses[is.na(mgm_courses)] = ""

classes.df = data.frame(biochem_courses,
                        biol_courses,
                        biostat_courses,
                        cell.molec.bio_courses,
                        chem_courses,
                        compsci_courses,
                        evanth_courses,
                        math_courses,
                        mgm_courses,
                        neurobio_courses,
                        neurosci_courses,
                        physics_courses,
                        psy_courses,
                        sta_courses)
colnames(classes.df) = c("BIOCHEM", "BIOLOGY", "BIOSTAT", "CMB",
                         "CHEM", "COMPSCI", "EVANTH", "MATH",
                         "MGM", "NEUROBIO", "NEUROSCI", "PHYSICS",
                         "PSY", "STA")
for (school in names(classes.df)){
  numbers = str_extract(classes.df[,school], "\\d\\d\\d")
  numbers[is.na(numbers)] = 999
  in.range = numbers <= 699
  classes.df[,school][!in.range] = ""
}

classes.df = classes.df %>%
  select(-BIOSTAT)
classes.df$BIOCHEM[3] = ""
classes.df$BIOLOGY[c(29:32,50,51,65:68)] = ""
classes.df$CHEM[c(10,15:16,20:21)] = ""
classes.df$COMPSCI[c(19:20)] = ""
classes.df$EVANTH[c(16:17)] = ""
classes.df$MATH[c(6,21:24,33:36)] = ""
classes.df$MGM[c(1,2,6)] = ""
classes.df$NEUROBIO[c(2)] = ""
classes.df$NEUROSCI[c(22,24:27)] = ""
classes.df$PHYSICS[c(32:34)] = ""
classes.df$PSY[c(45,53)] = ""
classes.df$STA[c(13,17)] = ""
save(classes.df, file = "classes.Rdata")
}
# page_biol = read_html("http://biology.duke.edu/courses")
# table_biol = page_biol %>%
#   html_node("#block-views-courses-block > div > div > div > table") %>%
#   html_table() %>%
#   filter(str_detect(.[,"Course Notes"], "offered Spring 2018"))
```

Appendix (data entry)

```
library(dplyr)
load("classes.Rdata")
load("sample.Rdata")
sample.df$numtexts = rep(NA, nrow(sample.df))
sample.df$newprice = rep(NA, nrow(sample.df))
sample.df$usedprice = rep(NA, nrow(sample.df))
n = nrow(sample.df)
##############################################
# This function does not always return a valid url
# Some classes may be listed as section 001 reather than 01
##############################################
make_store_url = function(department, number){
  url_part1 = "http://dukebooks.collegestoreonline.com/ePOS?wpd=1&width=100%25&this_category=1&term=SP1
  url_part2 = department
  url_part3 = "&course="
  url_part4 = as.character(number)
  url_part5 = "&colspan=3&cellspacing=1&cellpadding=0&campus=MAIN&border=0&bgcolor=%23cccccc&action=lis
  store_url = paste(url_part1, url_part2, url_part3, url_part4, url_part5, sep = "")
  return(store_url)
}

x = rep(NA, n)
for (i in seq_along(x)){
  x[i] = make_store_url(sample.df[i,"department"], sample.df[i, "course"])
}

sample.df$numtexts[1:(n/2)] = c(0,1,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,2,
                                2,1,1,0,1,1,0)

sample.df$newprice[1:(n/2)] = c(0,181.50,186.75,0,0,0,0,0,0,181.50,0,0,0,0,0,0,0,0,0,118.74,
                                204,181.50,118,0,142,69.50,0)

sample.df$usedprice[1:(n/2)] = c(0,136.25,140.25,0,0,0,0,0,0,136.25,0,0,0,0,0,0,0,0,0,89.25,
                                 153.25,136.25,88.50,0,106.50,52.25,0)

sample.df$numtexts[(n/2 + 1):n] = c(0,1,0,0,0,1,0,0,0,0,1,0,0,0,0,0,1,0,1,
                                    0,1,0,0,0,1,0,0)
sample.df$newprice[(n/2 + 1):n] = c(0,253.50,0,0,0,117.50,0,0,0,0,52.50,0,0,0,0,0,48.00,0,251.75,
                                    0,92.75,0,0,0,200.00,0,0)
sample.df$usedprice[(n/2 +1):n] = c(0,190.25,0,0,0,88.25,0,0,0,0,39.50,0,0,0,0,0,36.00,0,189.00,
                                    0,69.75,0,0,0,150.00,0,0)

surveyData = sample.df

newBIOCHEMrows = matrix(NA, ncol = ncol(surveyData), nrow = 7)
colnames(newBIOCHEMrows) = names(surveyData)

surveyData = rbind(newBIOCHEMrows, surveyData)
surveyData[1:7,1] = "BIOCHEM"
surveyData[1:7,"course"] = classes.df["BIOCHEM"] %>% .[.!= "" & .!= "536"]

newMGMrows = matrix(NA, ncol = ncol(surveyData), nrow = 2)
colnames(newMGMrows) = names(surveyData)

surveyData = rbind(surveyData[1:37,], newMGMrows, surveyData[38:nrow(surveyData),])
surveyData[38:39,1] = "MGM"
surveyData[38:39,"course"] = c("552", "582")

####################################
# add new MGM and BIOCHEM data below here
####################################

surveyData[1:5, "numtexts"] = c(0,0,0,0,0)
surveyData[1:5, "newprice"] = c(0,0,0,0,0)
surveyData[1:5, "usedprice"] = c(0,0,0,0,0)

surveyData[6:7, "numtexts"] = c(0,0)
surveyData[6:7, "newprice"] = c(0,0)
surveyData[6:7,"usedprice"] = c(0,0)

surveyData[38:39, "numtexts"] = c(0,0)
surveyData[38:39, "newprice"] = c(0,0)
surveyData[38:39,"usedprice"] = c(0,0)

surveyData[,"rowindex"] = c(1:nrow(surveyData)) #right rows

save(surveyData, file = "surveyData.Rdata")

write.csv(surveyData, file = "full_data.csv", row.names = FALSE)
```