ps4

Sijun Zhang 89934761

2019/11/18

In Problem Set 4, all confidence intervals are calculated at the level 0.05 and the lwr indicates the 2.5% lower bound and the upr indicates the 97.5% upper bound for each confidence interval.

Question 1

Firstly, the csv data file generated by ps2_q3.R is imported to the .do script as the data table for manuplation. The confidence interval is calculated at level 0.05

```
import delimited df_ps4_q1.csv

replace auc = max(auc,1)

replace tot_dist = log(tot_dist)
replace max_abs_dev = log(max_abs_dev)
replace avg_abs_dev = log(avg_abs_dev)
replace auc = log(auc)

encode condition, generate(condition_i)
replace condition_i = (-1)*condition_i+2
encode exemplar, generate(exemplar_i)

mixed tot_dist condition_i ||_all:R.subject_nr ||_all:R.exemplar
matrix b1 = e(b)
matrix v1 = e(V)

mixed max_abs_dev condition_i ||_all:R.subject_nr ||_all:R.exemplar
matrix b2 = e(b)
matrix v2 = e(V)

mixed avg_abs_dev condition_i ||_all:R.subject_nr ||_all:R.exemplar
matrix b3 = e(b)
matrix v3 = e(V)

mixed auc condition_i ||_all:R.subject_nr ||_all:R.exemplar
matrix b4 = e(b)
matrix v4 = e(V)

mata
b1_mata = st_matrix("b1")
b2_mata = st_matrix("b2")
b3_mata = st_matrix("b3")
b4_mata = st_matrix("b4")

v1_mata = st_matrix("v1")
v2_mata = st_matrix("v2")
v3_mata = st_matrix("v3")
v4_mata = st_matrix("v4")

a1 = (exp(b1_mata[1,1]), exp(b1_mata[1,1]-sqrt(v1_mata[1,1])*1.96),  exp(b1_mata[1,1]+sqrt(v1_mata[1,1])*
1.96))
a2 = (exp(b2_mata[1,1]), exp(b2_mata[1,1]-sqrt(v2_mata[1,1])*1.96),  exp(b2_mata[1,1]+sqrt(v2_mata[1,1])*
1.96))
a3 = (exp(b3_mata[1,1]), exp(b3_mata[1,1]-sqrt(v3_mata[1,1])*1.96),  exp(b3_mata[1,1]+sqrt(v3_mata[1,1])*
1.96))
a4 = (exp(b4_mata[1,1]), exp(b4_mata[1,1]-sqrt(v4_mata[1,1])*1.96),  exp(b4_mata[1,1]+sqrt(v4_mata[1,1])*
1.96))

a = a1\a2\a3\a4

st_matrix("re", a)

end
matrix rownames re=Total_Distance Maximum_Absolute_Deviation Average_Absolute_Deviation AUC
matrix colnames re=Relative_effect_est Relative_effect_lwr Relative_effect_upr

putexcel set ps4_q1.xls, replace
putexcel A1 = matrix(re), names

* Script Cleanup: ------------------------------------------------------------
log close
exit
```

```
library(readxl)
ps4_q1 = read_excel("ps4_q1.xls",sheet=1,na="NA")
```

```
## New names:
## * `` -> ...1
```

```
knitr::kable(ps4_q1)
```

...1	Relative_effect_est	Relative_effect_lwr	Relative_effect_upr
Total_Distance	1.175112	1.092844	1.263572
Maximum_Absolute_Deviation	1.665213	1.334694	2.077579
Average_Absolute_Deviation	1.917491	1.486285	2.473799
AUC	1.418568	1.162286	1.731359

From the above table, we find the coefficient of Condition on “the average absolute deviation of the observed trajectory from the direct path” is the maximum. Moreover, the expectation of average absolute deviation is also the smallest, which means the unit change in Condition variable is more influential in changing the distribution of average absolute deviation. Considering the two facts in average absolute deviation, we can conclude that the Condition has largest effect on the average absolute deviation. Moreover, the 95% confidence intervals are obtained in the above table.

Question 2

In this question, we aims to find which census division has the largest disparity between urban and rural areas in terms of the proportion of homes with internet access. And the result is stored in a csv file.

```
clear                   // Start clean

* Data Prep: ----------------------------------------------------------------
// data prep
*? Maybe tidy up over-long line with locals?
local recs_file "recs2015_public_v4.csv"
local recs_url = "https://www.eia.gov/consumption/residential/data/2015/csv/"
capture confirm file `recs_file'
if _rc==0 {
  display "Loading local file"
  import delimited `recs_file', clear
}
else {
  display "Download file and create local copy"
  import delimited `recs_url'`recs_file', clear
  export delimited `recs_file'
}

encode uatyp10, generate(uatyp10_i)
replace uatyp10_i = 1 if uatyp10_i==3

label define division_i 1 "New England" 2 "Middle Atlantic" 3 "East North Central" 4 "West North Central"
5 "South Atlantic" 6 "East South Central" 7 "West South Central" 8 "Mountain North" 9 "Mountain South" 10
"Pacific"
label values division division_i

preserve

local vars = "uatyp10_i division internet"
keep doeid nweight `vars'
save recs2015_prop_est.dta, replace

// in order to merge the nweight and brrwt values
restore
keep doeid brrwt1-brrwt96 `vars'
reshape long brrwt, i(doeid) j(repl)
merge m:1 doeid using recs2015_prop_est.dta

// calc the proportion of homes with internet
generate brrwt_with_internet = internet*brrwt
generate nw_with_internet = internet*nweight
collapse (sum) brrwt brrwt_with_internet nweight nw_with_internet, by(division uatyp10_i repl)
generate prop_repl = brrwt_with_internet/brrwt
generate prop = nw_with_internet/nweight
drop brrwt brrwt_with_internet
drop nweight nw_with_internet

// calc the confidence interval using brrwt replications
reshape wide prop_repl prop,i(division repl) j(uatyp10_i)
generate diff_prop = prop1-prop2
generate diff_prop_repl = prop_repl1 - prop_repl2
generate rsq_prop1 = (prop1-prop_repl1)^2/(1-0.5)^2
generate rsq_prop2 = (prop2-prop_repl2)^2/(1-0.5)^2
generate rsq_prop_diff = (diff_prop-diff_prop_repl)^2/(1-0.5)^2
collapse (mean) prop1 prop2 diff_prop rsq_prop1 rsq_prop2 rsq_prop_diff , by(division)

generate diff_prop_lwr = diff_prop - 1.96*sqrt(rsq_prop_diff)
generate diff_prop_upr = diff_prop + 1.96*sqrt(rsq_prop_diff)

rename stderr_prop1 rsq_prop1
replace stderr_prop1 = sqrt(stderr_prop1)
rename stderr_prop2 rsq_prop2
replace stderr_prop2 = sqrt(stderr_prop2)
rename stderr_diff_prop rsq_prop_diff
replace stderr_diff_prop = sqrt(stderr_diff_prop)
export delimited recs_prop_with_internet.csv, replace
```

```
ps4_q2 = read.delim("recs_prop_with_internet.csv", sep = ",")
names(ps4_q2) = c("division", "prop_urb", "prop_rur", "diff_prop","SE_prop_urb", "SE_prop_rur", "SE_diff_
prop", "diff_prop_lwr", "diff_prop_upr")
options(digits = 4)
knitr::kable(ps4_q2)
```

division	prop_urb	prop_rur	diff_prop	SE_prop_urb	SE_prop_rur	SE_diff_prop	diff_prop_lwr	diff_prop_upr
New England	0.8757	0.8579	0.0178	0.0259	0.0175	0.0218	-0.0248	0.0604
Middle Atlantic	0.8934	0.9129	-0.0195	0.0280	0.0305	0.0364	-0.0909	0.0519
East North Central	0.8625	0.8621	0.0004	0.0127	0.0233	0.0273	-0.0530	0.0539
West North Central	0.8800	0.8033	0.0768	0.0172	0.0451	0.0517	-0.0245	0.1781
South Atlantic	0.8530	0.8204	0.0326	0.0136	0.0294	0.0347	-0.0354	0.1005
East South Central	0.7836	0.6903	0.0933	0.0399	0.0282	0.0547	-0.0140	0.2006
West South Central	0.8161	0.7650	0.0510	0.0265	0.0223	0.0377	-0.0230	0.1250
Mountain North	0.8742	0.8193	0.0550	0.0277	0.0414	0.0596	-0.0619	0.1719
Mountain South	0.8527	0.6675	0.1852	0.0201	0.0433	0.0578	0.0719	0.2984
Pacific	0.8871	0.8528	0.0343	0.0129	0.0400	0.0405	-0.0451	0.1137

```
ps4_q2[which( ps4_q2[,"diff_prop"] == max(ps4_q2[,"diff_prop"])),1]
```

```
## [1] Mountain South
## 10 Levels: East North Central East South Central ... West South Central
```

From the above table, prop_urb shows the proportion of home with internet in urban area, prop_rur shows the proportion of home with internet in rural area and diff_prop shows the disparity between the urban proportion and rural proportion. Each of them has been obtained 95% confidence inerval which is shown above. From the diff_prop columns, we can find the “Mountain South” census division has the largest disparity between urban and rural areas in terms of the proportion of homes with internet access.

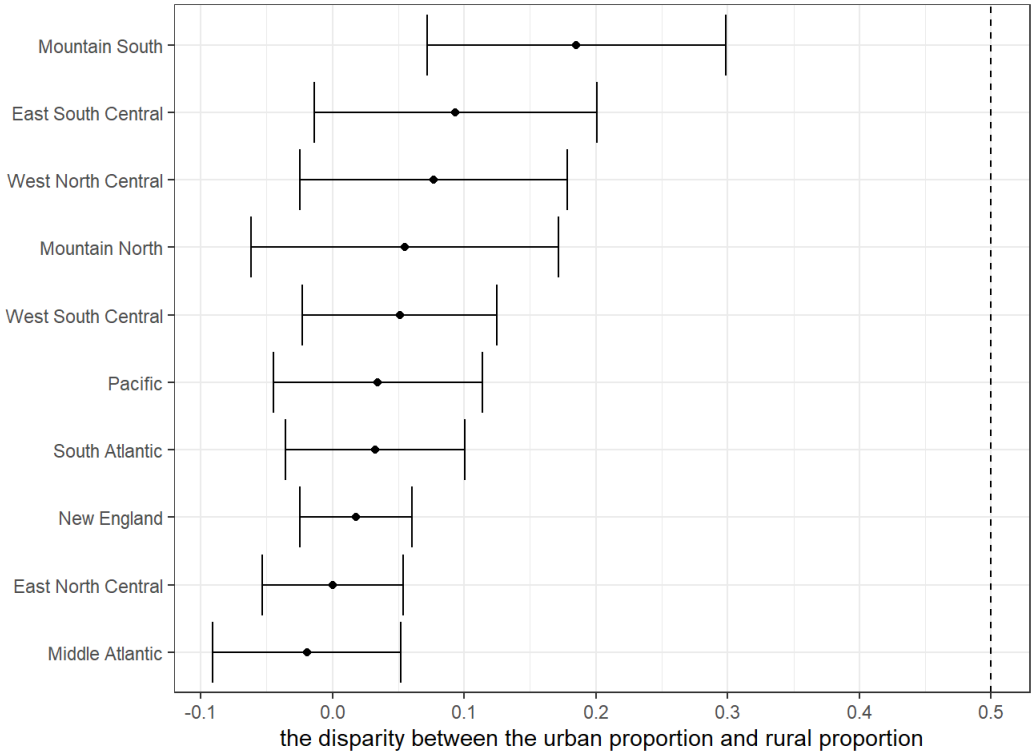
```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------
---------------------------------------- tidyverse 1.2.1 --
```

```
## √ ggplot2 3.2.1     √ purrr   0.3.2
## √ tibble  2.1.3     √ dplyr   0.8.3
## √ tidyr   1.0.0     √ stringr 1.4.0
## √ readr   1.3.1     √ forcats 0.4.0
```

```
## -- Conflicts ------------------------------------------------------------------------------------
------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
ps4_q2 %>%
  arrange( diff_prop ) %>%
  mutate( measure = factor(division, levels = unique(division) ) ) %>%
  ggplot( aes( y = measure, x = diff_prop) ) +
  geom_point() +
  geom_errorbarh( aes(xmin = diff_prop_lwr, xmax = diff_prop_upr) ) +
  geom_vline( xintercept = 0.5, lty = 'dashed') +
  xlab('the disparity between the urban proportion and rural proportion') +
  ylab('') +
  theme_bw()
```



Question 3

Part a

In this part, the binary indicators are all labelled to be factor variables.

```
import sasxport5 DEMO_D, clear
keep seqn riagendr ridageyr indfmpir ridexmon
// find if it in winter
generate if_winter = ridexmon
generate age = ridageyr
generate gender = riagendr
generate pir = indfmpir
drop riagendr ridageyr indfmpir ridexmon
save demo_d.dta, replace

import sasxport5 DR1TOT_D, clear

keep seqn dr1day dr1_320z
// find if it was weekday
generate in_weekday1 = 1 if dr1day <= 6 & dr1day >=2
replace in_weekday1 = 0 if dr1day == 7 | dr1day == 1
// find if the responent has drank water
generate drank_water1 = 0 if dr1_320z == 0
replace drank_water1 = 1 if dr1_320z > 0
replace drank_water1 = . if dr1_320z == .
drop dr1_320z
save dr1tot_d.dta, replace

import sasxport5 DR2TOT_D, clear
keep seqn dr2day dr2_320z
// find if it was weekday
generate in_weekday2 = 1 if dr2day <= 6 & dr2day >=2
replace in_weekday2 = 0 if dr2day == 7 | dr2day == 1
// find if the responent has drank water
generate drank_water2 = 0 if dr2_320z == 0
replace drank_water = 1 if dr2_320z > 0
replace drank_water2 = . if dr2_320z == .
drop dr2_320z
merge m:1 seqn using dr1tot_d.dta
drop _merge

generate drday1 = dr1day
generate drday2 = dr2day
drop dr1day dr2day
reshape long in_weekday drank_water drday, i(seqn) j(svy_day)

merge m:1 seqn using demo_d.dta
drop _merge

label define winter_i 1 "in winter" 2 "not in winter"
label define drank_water_i 1 "has drank water" 0 "not drank water"
label define week_i 1 "weekday" 0 "weekend"
label define gender_i 1 "Male" 2 "Female"

label values in_weekday week_i
label values drank_water drank_water_i
label values if_winter winter_i
label values gender gender_i

export delimited ps4_q3_a_df.csv, replace
```

```
ps4_q3_a = read.delim("ps4_q3_a_df.csv", sep = ",")
options(digits = 4)
knitr::kable(head(ps4_q3_a) )
```

seqn	svy_day	in_weekday	drank_water	drday	if_winter	age	gender	pir
31127	1	weekend	not drank water	7	not in winter	0	Male	0.75
31127	2	weekend	not drank water	1	not in winter	0	Male	0.75
31128	1	weekday	not drank water	6	in winter	11	Female	0.77
31128	2	weekday	has drank water	2	in winter	11	Female	0.77
31129	1	weekday	has drank water	2	not in winter	15	Male	2.71

seqn	svy_day	in_weekday	drank_water	drday	if_winter	age	gender	pir
31129	2	weekend	has drank water	1	not in winter	15	Male	2.71

Part b

```
generate missing = 0
replace missing = 1 if in_weekday == . | drank_water == . | age == . | gender == . | pir == . | if_winter
== .

generate fid = 0
save ps4_q3_df_missing.dta, replace

// calc the mean of pir and age
drop if missing == 1
collapse (mean) pir age
generate mean_pir = pir
generate mean_age = age
drop pir age
export delimited mean_pir_age.csv, replace

generate fid = 0
merge 1:m fid using ps4_q3_df_missing.dta

// Centering and decading to as it is continuous
replace age = (age - mean_age) / 10
replace pir = pir - mean_pir

drop mean_age mean_pir _merge

export delimited ps4_q3_b_df.csv, replace
```

```
ps4_q3_b = read.delim("ps4_q3_b_df.csv", sep = ",")
options(digits = 4)
knitr::kable(head(ps4_q3_b) )
```

fid	seqn	svy_day	in_weekday	drank_water	drday	if_winter	age	gender	pir	missing
0	31127	1	weekend	not drank water	7	not in winter	-2.798	Male	-1.6452	0
0	31127	2	weekend	not drank water	1	not in winter	-2.798	Male	-1.6452	0
0	31128	1	weekday	not drank water	6	in winter	-1.698	Female	-1.6252	0
0	31128	2	weekday	has drank water	2	in winter	-1.698	Female	-1.6252	0
0	31129	1	weekday	has drank water	2	not in winter	-1.298	Male	0.3148	0
0	31129	2	weekend	has drank water	1	not in winter	-1.298	Male	0.3148	0

In this problem, the unit of age is decades, but variable age is still continuous and has a mean of 0 among no missing value data.

```
ps4_q3_b_mean = read.delim("mean_pir_age.csv", sep = ",")
options(digits = 8)
knitr::kable(head(ps4_q3_b_mean) )
```

mean_pir	mean_age
2.3952346	27.975784

Part c

```
save ps4_q3_log_df.dta, replace
// only use day 1 data
drop if svy_day == 2 | svy_day == .
drop if missing == 1
logistic drank_water in_weekday if_winter c.age##c.age gender c.pir
matrix b1 = e(b)
matrix v1 = e(V)

margins, dydx(*)
matrix b2 = r(b)
matrix v2 = r(V)

mata
    b1_m = st_matrix("b1")
    b1_m = b1_m[1],b1_m[2],b1_m[3],b1_m[5],b1_m[6],b1_m[4]
    v1_m = st_matrix("v1")
    v1_m = diagonal(v1_m)'
    v1_m = v1_m[1],v1_m[2],v1_m[3],v1_m[5],v1_m[6],b1_m[4]
    b2_m = st_matrix("b2"),(.)
    v2_m = st_matrix("v2")
    v2_m = diagonal(v2_m)',(.)

    a1 = exp(b1_m - 1.96*sqrt(v1_m))
    b1 = exp(b1_m + 1.96*sqrt(v1_m))
    re1 = a1\b1

    a2 = (b2_m - 1.96*sqrt(v2_m))
    b2 = (b2_m + 1.96*sqrt(v2_m))
    re2 = a2\b2

    re = exp(b1_m)\re1\b2_m\re2

    st_matrix("re_c",re)

end

matrix rownames re_c=Odds_ratios Odds_ratios_lwr Odds_ratios_upr Marginal_effect Marginal_effect_lwr Marg
inal_effect_upr
matrix colnames re_c=in_weekday if_winter age gender pir age^2

putexcel set ps4_q3_c.xls, replace
putexcel A1 = matrix(re_c), names
```

```
ps4_q3_c = read_excel("ps4_q3_c.xls",sheet=1,na="NA")
```

```
## New names:
## * `` -> ...1
```

```
knitr::kable(ps4_q3_c)
```

...1	in_weekday	if_winter	age	gender	pir	age^2
Odds_ratios	1.13013637	1.10980804	1.15818199	1.33265955	1.09384890	0.97373228
Odds_ratios_lwr	1.02774566	1.00991703	1.12831459	1.21330492	1.06038751	0.34062927
Odds_ratios_upr	1.24272790	1.21957929	1.18884000	1.46375528	1.12836619	2.78353808
Marginal_effect	0.02400408	0.02044262	0.03165433	0.05634712	0.01760061	NA
Marginal_effect_lwr	0.00539069	0.00195126	0.02568416	0.03805279	0.01153741	NA
Marginal_effect_upr	0.04261748	0.03893397	0.03762451	0.07464146	0.02366380	NA

The odds ratio and average marginal effects for each variable are shown above, which also includes the 95% confidence interval of odds ratio and average marginal effect.

Part d

```
use ps4_q3_log_df, clear
drop if missing == 1
meglm drank_water in_weekday if_winter c.age##c.age gender c.pir ||seqn:, family(binomial) link(logit)
matrix b1 = e(b)
matrix v1 = e(V)

margins, dydx(*)
matrix b2 = r(b)
matrix v2 = r(V)

mata
    b1_m = st_matrix("b1")
    b1_m = b1_m[1],b1_m[2],b1_m[3],b1_m[5],b1_m[6],b1_m[4]
    v1_m = st_matrix("v1")
    v1_m = diagonal(v1_m)'
    v1_m = v1_m[1],v1_m[2],v1_m[3],v1_m[5],v1_m[6],v1_m[4]
    b2_m = st_matrix("b2"),(.)
    v2_m = st_matrix("v2")
    v2_m = diagonal(v2_m)',(.)

    a1 = (b1_m - 1.96*sqrt(v1_m))
    b1 = (b1_m + 1.96*sqrt(v1_m))
    re1 = exp(a1)\exp(b1)
    re3 = a1\b1

    a2 = (b2_m - 1.96*sqrt(v2_m))
    b2 = (b2_m + 1.96*sqrt(v2_m))
    re2 = a2\b2

    re = exp(b1_m)\re1\b2_m\re2\b1_m\re3

    st_matrix("re_d",re)

end

matrix rownames re_d=Odds_ratios Odds_ratios_lwr Odds_ratios_upr Marginal_effect Marginal_effect_lwr Marg
inal_effect_upr Coefficients Coefficients_lwr Coefficients_lwr
matrix colnames re_d=in_weekday if_winter age gender pir age^2

putexcel set ps4_q3_d.xls, replace
putexcel A1 = matrix(re_d), names
```

```
ps4_q3_d = read_excel("ps4_q3_d.xls",sheet=1,na="NA")
```

```
## New names:
## * `` -> ...1
```

```
knitr::kable(ps4_q3_d)
```

...1	in_weekday	if_winter	age	gender	pir	age^2
Odds_ratios	1.23997947	1.07781793	1.36247825	1.75791484	1.17901763	0.95507806
Odds_ratios_lwr	1.10901761	0.93689837	1.30828144	1.52708791	1.12590614	0.94204654
Odds_ratios_upr	1.38640637	1.23993330	1.41892022	2.02363240	1.23463450	0.96828985
Marginal_effect	0.02304188	0.00802774	0.03590087	0.06043184	0.01764139	NA
Marginal_effect_lwr	0.01111321	-0.00697934	0.03093848	0.04554587	0.01275459	NA
Marginal_effect_upr	0.03497054	0.02303482	0.04086326	0.07531782	0.02252820	NA
Coefficients	0.21509482	0.07493856	0.30930529	0.56412835	0.16468157	-0.04596220
Coefficients_lwr	0.10347459	-0.06518046	0.26871440	0.42336260	0.11858817	-0.05970060
Coefficients_lwr	0.32671506	0.21505759	0.34989617	0.70489411	0.21077498	-0.03222381

From the above two logistic models, we can find when the second day data is added in the model, the variable if_winter becomes insignificant and the marginal effect of if_winter turned from positive to negative in the second mixed model. The odds ratio and marginal

effect of variable is_weekday, age, gender and pir become more significant when the second day data is added. The marginal effects in these two models are very close to each other. And the corresponding confidence intervals also overlap.

To answer the overall question: Yes, in the US, people are more likely to drink water on a weekday than on a weekend day. The odds ratios for that are 1.13 and 1.24 respectively in the above models.