Exercise: Analyzing Survey Experiments

Download the datasets from the website containing responses to survey experiments. Answer the following questions:

```
survey <- read_csv("Survey_data.csv")
```

```
## Parsed with column specification:
## cols(
##   mun = col_double(),
##   gender = col_character(),
##   age = col_double(),
##   education = col_character(),
##   random_order = col_character(),
##   cand_pref_cl_proga = col_character(),
##   cand_pref_cl_progb = col_character(),
##   random_list = col_character(),
##   list_treated = col_double(),
##   list_response = col_double(),
##   direct_clientelism = col_double(),
##   rnd_other = col_double(),
##   other_cards_A = col_character(),
##   Primed = col_double(),
##   Constitution_Reform_Support = col_double()
## )
```

```
#survey <- survey %>% filter(is.na(list_response)==F)
```

Priming Experiment

`Survey_data.csv`

1. Respondents were primed with one of two questions ('hope' and 'anger'). How does this prime affect subsequent answers to the question about the need for constitutional reform? Calculate the difference-in-means estimate of the average treatment effect. Interpret the result.

```
survey %>% group_by(Primed) %>%
  dplyr::summarize(mean=mean(Constitution_Reform_Support,na.rm=T)) %>%
  mutate(ATE=mean-lag(mean))
```

```
## # A tibble: 2 x 3
##   Primed  mean     ATE
##    <dbl> <dbl>   <dbl>
## 1      0 0.510 NA
## 2      1 0.594  0.0837
```

2. Perform the same analysis, but using a simple OLS regression.

```
survey %>% lm(Constitution_Reform_Support ~ Primed, data=.) %>% summary()
```

```
##
## Call:
## lm(formula = Constitution_Reform_Support ~ Primed, data = .)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -0.5940 -0.5103  0.4060  0.4897  0.4897
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.51028    0.01551  32.891  < 2e-16 ***
## Primed       0.08371    0.02188   3.825 0.000134 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4957 on 2051 degrees of freedom
## Multiple R-squared:  0.007085,   Adjusted R-squared:  0.0066
## F-statistic: 14.63 on 1 and 2051 DF,  p-value: 0.0001344
```

List Experiment

`Survey_data.csv`

The number of items the respondent states are contained in the variable `list_response`. If they were shown a control list (3 items) the variable `list_treated` is equal to zero, and if they were shown a treatmeng list (4 items) the variable `list_treated` is equal to one.

3. Calculate the average treatment effect by subtracting the mean number of responses between the treated and control lists. Interpret the results in terms of the proportion of respondents who have experienced the sensitive item.

```
survey %>% group_by(list_treated) %>%
  summarize(mean=mean(list_response,na.rm=T)) %>%
  mutate(ATE=mean-lag(mean))
```

```
## # A tibble: 2 x 3
##   list_treated  mean    ATE
##          <dbl> <dbl>  <dbl>
## 1            0  1.26 NA
## 2            1  1.47  0.209
```

4. Conduct the same analysis using a simple OLS regression. What is the 95% confidence interval of the proportion of respondents who received a clientelist offer?

```
survey %>% lm(list_response~list_treated, data=.) %>%
  tidy() %>%
  mutate(conf.lo=estimate-std.error*1.96,
         conf.hi=estimate+std.error*1.96) %>%
  filter(term=="list_treated") %>%
  select(conf.lo, conf.hi)
```

```
## # A tibble: 1 x 2
##   conf.lo conf.hi
##     <dbl>   <dbl>
## 1   0.144   0.274
```

5. Now let's check the assumptions of the list experiment. First, check if there is a design effect using the function `ict_test` in the `list` package for R. (Note you will need to remove missing values before running the test). Interpret the results.

```
survey_no_na <- survey %>% filter(is.na(list_response)==F)
```

```r
ict.test(survey_no_na$list_response, survey_no_na$list_treated, J=3)
```

```
##
## Test for List Experiment Design Effects
##
## Estimated population proportions
##                             est.   s.e.
## pi(Y_i(0) = 0, Z_i = 1) -0.0008 0.0117
## pi(Y_i(0) = 1, Z_i = 1)  0.1228 0.0211
## pi(Y_i(0) = 2, Z_i = 1)  0.0636 0.0111
## pi(Y_i(0) = 3, Z_i = 1)  0.0232 0.0047
## pi(Y_i(0) = 0, Z_i = 0)  0.0763 0.0082
## pi(Y_i(0) = 1, Z_i = 0)  0.4998 0.0175
## pi(Y_i(0) = 2, Z_i = 0)  0.2005 0.0172
## pi(Y_i(0) = 3, Z_i = 0)  0.0146 0.0076
##
##  Y_i(0) is the (latent) count of 'yes' responses to the control items. Z_i is the (latent) binary res
##
## Bonferroni-corrected p-value
## If this value is below alpha, you reject the null hypothesis of no design effect. If it is above alph
##
## Sensitive Item 1
##        0.9467432
```

6. Next, let's check for floor and ceiling effects. There is a complex statistical test for this in the `list` package: Try the code below and interpret the 'floor' and 'ceiling' parameters to see if they are statistically significant from zero. These are the estimates of whether anyone who should have answered '4' actually lied and answered '3', or who should have answered '1' actually lied and answered '0'.

```r
ictreg(list_response~1,
       data=survey %>% as.data.frame(),
       treat="list_treated",
       J=3,
       method="ml",
       floor=T,
       ceiling=T,
       floor.fit="bayesglm",
       ceiling.fit="bayesglm") %>%
  summary()
```

7. The survey also asked people directly, `direct_clientelism`. whether they had experienced the sensitive item (Has anyone ever offered you a gift, some food or money in exchange for your vote?). Compare the non-response rate (NA responses) to the direct and indirect questions.

```r
survey %>% summarize(na_list=sum(is.na(list_response))/dim(survey)[1]*100,
                     na_direct=sum(is.na(direct_clientelism))/dim(survey)[1]*100)
```

```
## # A tibble: 1 x 2
##   na_list na_direct
##     <dbl>     <dbl>
## 1   0.487      8.18
```

8. Compare the estimate of the incidence of clientelism from the direct responses to the indirect estimate from the list experiment. What does this suggest about the level of social desirability bias?

```r
survey %>% summarize(direct_mean=mean(direct_clientelism,na.rm=T))
```

```
## # A tibble: 1 x 1
##   direct_mean
##         <dbl>
## 1       0.177
```

9. Are men or women more likely to have engaged in the sensitive item (been offered a gift)? Run an OLS regression with an interaction between the list treatment and gender to find out. Interpret the results.

```r
survey %>% lm(list_response~list_treated*gender, data=.) %>%
  tidy()
```

```
## # A tibble: 5 x 5
##   term                   estimate std.error statistic   p.value
##   <chr>                     <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)              1.24      0.0288    43.1   2.67e-289
## 2 list_treated             0.164     0.0404     4.07  4.89e-  5
## 3 gendermale               0.0707    0.0500     1.41  1.57e-  1
## 4 genderother              0.595     0.747      0.796 4.26e-  1
## 5 list_treated:gendermale  0.132     0.0701     1.88  6.06e-  2
```

Conjoint Experiment

`Conjoint_data.csv`. The dataset is arranged with one row for every candidate that each respondent assessed (two candidates * two candidates * 4047 respondents). So there are four rows for every respondent - two experiments with two candidates in each. The first columns describe the attributes of each candidate profile. The variable `conjoint_choice` is a binary indicator of which candidate the respondent opted to vote for. There are also columns for the characteristics of the respondent (gender, age, and whether they are a co-ethnic of the candidate profile).

```r
conjoint <- read_csv("Conjoint_data.csv")
```

```
## Parsed with column specification:
## cols(
##   UID = col_character(),
##   Round = col_double(),
##   Choice = col_character(),
##   Profile_Gender = col_character(),
##   Profile_Caste = col_character(),
##   Profile_Party = col_character(),
##   Profile_PG = col_character(),
##   Profile_Promise = col_character(),
##   conjoint_choice = col_double(),
##   respondent_state = col_character(),
##   respondent_gender = col_character(),
##   respondent_age = col_double(),
##   respondent_co_ethnic = col_double()
## )
```

10. How many possible combinations of attributes for a single profile are there? Multiply the number of possible levels for every attribute.

```r
2*3*4*2*2
```

```
## [1] 96
```

11. Run a simple OLS regression to evaluate how the 'Public Goods (PG)' attribute of a Profile affects the respondent's choice of candidate.

```
conjoint %>% zelig(conjoint_choice~1+Profile_PG, data=.,model="ls")
```

```
## How to cite this model in Zelig:
##   R Core Team. 2007.
##   ls: Least Squares Regression for Continuous Dependent Variables
##   in Christine Choirat, Christopher Gandrud, James Honaker, Kosuke Imai, Gary King, and Olivia Lau,
##   "Zelig: Everyone's Statistical Software," http://zeligproject.org/

## Model:
##
## Call:
## z5$zelig(formula = formula, data = data, by = by)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5892 -0.3686 -0.3686  0.4108  0.6314
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)            0.368578   0.005428    67.9   <2e-16
## Profile_PGGood Roads 0.220598   0.007660    28.8   <2e-16
##
## Residual standard error: 0.4873 on 16186 degrees of freedom
## Multiple R-squared:  0.04875,    Adjusted R-squared:  0.04869
## F-statistic: 829.4 on 1 and 16186 DF,  p-value: < 2.2e-16
##
## Next step: Use 'setx' method
```

12. Since our outcome is a binary variable, run the same regression but with a logit model.

```
conjoint %>% zelig(conjoint_choice~1+Profile_PG, data=.,model="logit")
```

```
## How to cite this model in Zelig:
##   R Core Team. 2007.
##   logit: Logistic Regression for Dichotomous Dependent Variables
##   in Christine Choirat, Christopher Gandrud, James Honaker, Kosuke Imai, Gary King, and Olivia Lau,
##   "Zelig: Everyone's Statistical Software," http://zeligproject.org/

## Model:
##
## Call:
## z5$zelig(formula = formula, data = data, by = by)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3339  -0.9589  -0.9589   1.0286   1.4129
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -0.53832    0.02309  -23.31   <2e-16
## Profile_PGGood Roads  0.89888    0.03227   27.85   <2e-16
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 22414  on 16187  degrees of freedom
## Residual deviance: 21618  on 16186  degrees of freedom
## AIC: 21622
##
## Number of Fisher Scoring iterations: 4
##
## Next step: Use 'setx' method
```

13. Since each respondent participated in two experiments, their answers are likely to be highly correlated. So we have less 'N' than we think. Cluster the standard errors of your OLS regression according to the respondent identifier (UID).

```r
conjoint %>% lm_robust(conjoint_choice~1+Profile_PG, data=., clusters=UID)
```

```
##                          Estimate  Std. Error  t value      Pr(>|t|)
## (Intercept)             0.3685778 0.004343275 84.86172  0.000000e+00
## Profile_PGGood Roads 0.2205981 0.008140100 27.10017 2.203171e-147
##                          CI Lower  CI Upper       DF
## (Intercept)             0.3600615 0.3770941 2826.775
## Profile_PGGood Roads 0.2046384 0.2365577 3609.790
```

14. Assess the influence of all of the profile attributes at the same time in an OLS regression. Interpret the results.

```r
conjoint %>% zelig(conjoint_choice~1+Profile_Gender + Profile_Caste + Profile_Party + Profile_PG + Profi
```

```
## How to cite this model in Zelig:
##   R Core Team. 2007.
##   ls: Least Squares Regression for Continuous Dependent Variables
##   in Christine Choirat, Christopher Gandrud, James Honaker, Kosuke Imai, Gary King, and Olivia Lau,
##   "Zelig: Everyone's Statistical Software," http://zeligproject.org/
```

```
## Model:
##
## Call:
## z5$zelig(formula = formula, data = data, by = by)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.6270 -0.3954 -0.3275  0.4278  0.6776
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)               0.3606796  0.0108467  33.252  < 2e-16
## Profile_GenderMale        0.0018807  0.0076544   0.246 0.805921
## Profile_CasteOBC          0.0340117  0.0093917   3.621 0.000294
## Profile_CasteSC           0.0213807  0.0093738   2.281 0.022567
## Profile_PartyINC         -0.0332341  0.0115122  -2.887 0.003896
## Profile_PartyJDU          0.0089421  0.0113588   0.787 0.431153
## Profile_PartyJMM         -0.0108667  0.0115892  -0.938 0.348437
## Profile_PartyRJD         -0.0383008  0.0114731  -3.338 0.000845
## Profile_PGGood Roads      0.2207923  0.0076546  28.844  < 2e-16
## Profile_PromisePatronage  0.0007034  0.0076543   0.092 0.926778
##
## Residual standard error: 0.4869 on 16178 degrees of freedom
## Multiple R-squared:  0.05079,    Adjusted R-squared:  0.05026
## F-statistic: 96.18 on 9 and 16178 DF,  p-value: < 2.2e-16
```

```
##
## Next step: Use 'setx' method
```

15. We an also assess how respondents' characteristics affect their choice. Does the importance of the 'Promise' attribute vary by gender? Use an interaction term between the Promise attribute and respondent gender, and interpret the results.

```
conjoint %>% zelig(conjoint_choice~1+Profile_Promise*respondent_gender, data=.,model="logit")
```

```
## How to cite this model in Zelig:
##   R Core Team. 2007.
##   logit: Logistic Regression for Dichotomous Dependent Variables
##   in Christine Choirat, Christopher Gandrud, James Honaker, Kosuke Imai, Gary King, and Olivia Lau,
##   "Zelig: Everyone's Statistical Software," http://zeligproject.org/

## Model:
##
## Call:
## z5$zelig(formula = formula, data = data, by = by)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.162  -1.146  -1.110   1.209   1.247
##
## Coefficients:
##                                          Estimate Std. Error z value
## (Intercept)                              -0.16166    0.03562  -4.539
## Profile_PromisePatronage                  0.08780    0.05088   1.726
## respondent_gendermale                     0.12544    0.04554   2.755
## Profile_PromisePatronage:respondent_gendermale -0.13546    0.06476  -2.092
##                                          Pr(>|z|)
## (Intercept)                              5.65e-06
## Profile_PromisePatronage                  0.08441
## respondent_gendermale                     0.00587
## Profile_PromisePatronage:respondent_gendermale  0.03646
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 22414  on 16187  degrees of freedom
## Residual deviance: 22406  on 16184  degrees of freedom
## AIC: 22414
##
## Number of Fisher Scoring iterations: 3
##
## Next step: Use 'setx' method
```