SML 201 Project 2 Solutions

Tyler Campbell

April 18, 2018

Project 2 is due by 11:59p.m. on Wednesday April 18 on Blackboard. Please submit both a .Rmd and a .pdf file on Blackboard. If the due date falls on a date that has a lecture on the next day please bring a hard copy of the pdf file to the first lecture after the due date; otherwise, please drop off the pdf copy at 26 Prospect Avenue (see the *Submitting Problem Sets and Projects* section under *Problem Sets and Projects* on the Syllabus for detailed instructions) by 5pm on the next day of the due date. Please do not modify your .pdf hard copy after your BB submission; otherwise, you might get points deducted.

Late **projects** will be penalized at intervals rounded up to multiples of 24 hours. For example, if you are 3 hours late, 10% off or if you are 30 hours late, 20% off.

Make sure that you have all your digital signatures along with the honor pledge in each of these documents (there should be more than one signature if you work in groups).

This project can be completed in groups of up to 3 students. It is okay to work by yourself, if this is preferable. **You may not work with a given student on more than one project.** In other words, if you work with Student_1 and Student_2 on Project 1, then you cannot work with Student_1 or Student_2 on any other projects. You must form completely new groups for every project.

When working in a group it is your responsibility to make sure that you are satisfied with all parts of the report and the submission is on time (e.g., we will not entertain arguments that deficiencies are the responsibility of other group members). We expect that the work on any given problem set or project contains approximately equal contributions from all members of the group; we expect that you each work independently first and then compare your answers with each other once you all finish or you all work together. Failing to make contributions and then putting your name on a project will be considered a violation of the honor code. Also, please do not divide work among your group mates.

In general you are not allowed to get help on projects from other people except from partners in your group. Clarification questions are always welcome. Please treat projects as take-home exams.

For all parts of this problem set, you **MUST** use R commands to print the output as part of your R Markdown file. You are not permitted to find the answer in the R console and then copy paste the answer into this document.

If you are completing this problem set in a group, please have only **one** person in your group turn in the .Rmd and .pdf files; **other people in your group should turn in the list of the people in your group in the *Text Submission* field on the submission page**.

Please type your name(s) after "Digitally signed:" below the honor pledge to serve as digital signature(s). Put the pledge and your signature(s) at the beginning of each document that you turn in.

I pledge my honor that I have not violated the honor code when completing this assignment.

Digitally signed: Tyler Campbell

In order to receive full credits, please have sensible titles and axis labels for all your graphs and adjust values for all the relevant graphical parameters so that your plots are informative. Also, all answers must be written in complete sentences.

Just a friendly reminder: Please remember to annotate your code and have answers in the write up section, not in code chunks.

Objective of this project

The outcome of the 2016 presidential election was a big surprise to many of us; all 6 probabilistic and 3 non-probabilistic models published on New York Times missed the election result (click here for reference). Even on November 8, 2016 (the election day) Clinton was estimated to have a winning chance of 71.4% (compared to 28.6% for Trump) by Nate Silver, a statistician who had a track record of predicting the presidential election outcomes correctly until 2016 (click here for reference).

Many believe the main factor responsible for the failure of these models was the bias in the polling data (click here for reference). In this project we will explore the potential problems in the polling data that could lead to prediction failure.

Background info and datasets used in this project

We will use three datasets: `oct_poll.Rdata`, `2016_election_result.csv` and `county_facts.csv` in this project.

The dataset `oct_poll.Rdata` is a subset of the dataset from this website (https://www.r-bloggers.com/fivethirtyeights-polling-data-for-the-us-presidential-election/). `oct_poll.Rdata` contains poll data collected from polls that closed between Oct. 1, 2016 and Oct. 27, 2016. The original dataset was first posted on fivethirtyeight.com, a website created by Nate Silver. (Note: Even though Silver's prediction missed the truth Nate Silver's model was the most robust one among all the models published on *New York Times* for predicting the 2016 presidential election result.)

How did all 6 probabilistic models published on New York Times missed the election result?

In this section we will study the patterns of the errors in the poll data. We would like to answer these two questions:

- Do the errors seem random? Are they distributed symmetrically around zero?
- Among the states with CI estimates that do not cover zero (i.e., among the states that we were confident about who the winning candidates for the states would be), how many of their CIs predicted the correct state winner?

Question 1

In 2016 the election day was Nov. 8. `oct_poll.Rdata` contains poll data collected from polls that ended between Oct. 1, 2016 and Oct. 27, 2016; the dataset has 752 rows and 6 columns. If you forget how to read in this dataset try running `?load` to get help. The descriptions of the column names are:

- **state**: The state which was polled;
- **grade**: The rating of the company taking the poll;
- **samplesize**: Sample size of the poll;
- **rawpoll_clinton**: The percentage of people favoring Clinton in the sample;
- **rawpoll_trump**: The percentage of people favoring Trump in the sample;

- **end.date**: The closing date of the poll.

Note that there are multiple entries for each state since there are multiple poll agencies for each state. Also note that on each row the percentages for Clinton and Trump do not add up to 100% since there are other candidates.

Make a data frame `poll.oct2016`; this data frame should have 4 columns: `state`, `samplesize`, `rawpoll_clinton`, and `rawpoll_trump`. `poll.oct2016` should aggregate the data in `oct_poll` to the state level and express `rawpoll_clinton` and `rawpoll_trump` as proportions instead of percentages. You should keep `District of Columbia` even though technically it is not a state. `poll.oct2016` should have one row for each state. You need to be careful about how to aggregate the percentages; e.g., suppose state 1 has two polls: poll 1 has a sample of 100 and 30% of the people in the sample voted for Clinton and poll 2 has a sample of 400 and 60% of the people in the sample voted for Clinton; then, the aggregated proportion of people voted for Clinton in the polls for state1 is (100 x .3 + 400x.6)/(100+400) = .54.

Hint: Remember that outputs of `tapply()` are arrays and that `as.vector()` removes element names of the input variable by default; thus, you will need to assign the array element names back to the vector elements after you convert an array to a vector; see Week5b_notes.

```
load("oct_poll.Rdata")
library(dplyr)

Attaching package: 'dplyr'
The following objects are masked from 'package:stats':

    filter, lag
The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
# aggregate the data in oct_poll to the state level to
# poll.oct2016
poll.oct2016 = oct_poll %>% group_by(state) %>% mutate(A = rawpoll_trump *
    samplesize/100) %>% mutate(B = rawpoll_clinton * samplesize/100) %>%
    summarise(samplesize = sum(samplesize), rawpoll_trump = sum(A)/samplesize,
        rawpoll_clinton = sum(B)/samplesize)
```

Question 2

'2016_election_result.csv' contains the presidential election results for 2016. Read in the dataset `2016_election_result.csv` and name this dataset `true.perc`. Use the `merge()` function in R to merge `true.perc` with `poll.oct2016` by using the state names. Also, add the following columns to the data frame:

- **poll.diff** = rawpoll_trump - rawpoll_clinton
- **vote.diff** = trump.vote - clinton.vote
- **deviation** = poll.diff - vote.diff

Name the resulting data frame `election`. `election` should have 51 rows and 9 columns.

```
# merge presidential election results for 2016 with poll.oct2016
# by state
true.perc = read.csv("2016_election_result.csv")
election = merge(true.perc, poll.oct2016, by = "state")
# add 3 columns (diff and devation)
election = mutate(election, poll.diff = rawpoll_trump - rawpoll_clinton,
    vote.diff = trump.vote - clinton.vote, deviation = poll.diff -
        vote.diff)
```

```
dim(election)
[1] 51  9
```

Question 3

Part a

Create the following vectors and add them as the columns to `election`:

- **std.unit**: For the i^{th} state `std.unit[i]` is `poll.diff[i]` in standard units assuming `vote.diff[i]` is the mean for `poll.diff[i]`; e.g., if `poll.diff[1]` is 3 SE(poll.diff[1]) above `vote.diff[i]`, `std.unit[i]` should be 3; if `poll.diff[1]` is 2.5 SE(poll.diff[1]) below `vote.diff[1]`, `std.unit[1]` should be -2.5.
- **same.sign**: A logical vector whose i^{th} element should be `TRUE` if both `poll.diff[i]` and `vote.diff[i]` have the same sign (i.e., either both positive or both negative), and `FALSE` otherwise.
- **significant**: A logical vector that shows `TRUE` if the 95% CI for estimating (proportion of Trump supporters - proportion of Clinton supporters) does not cover 0 and `FALSE` otherwise.

```
std.unit = rep(0, 51)
significant = rep(T, 51)
# loop for each state
for (x in 1:51) {
    n = election$samplesize[x]
    t = election$rawpoll_trump[x]
    c = election$rawpoll_clinton[x]
    # recreate sample
    i = rep(1, round(t * n))
    i = append(i, rep(-1, round(c * n)))
    i = append(i, rep(0, round((1 - t - c) * n)))
    SE = sd(i)/sqrt(n)
    # calculate std.unit and if 95% CI covers 0
    std.unit[x] = (election$poll.diff[x] - election$vote.diff[x])/SE
    significant[x] = !between(0, t.test(i)$conf.int[1], t.test(i)$conf.int[2])
}
# poll results and actual results match
same.sign = sign(election$poll.diff) == sign(election$vote.diff)
# add new vectors to election data frame
election = mutate(election, std.unit = std.unit, same.sign = same.sign,
    significant = significant)
```

Part b

Use the information that you calculated in 3.a to answer the following questions:

i. How many of the 95% CIs for estimating (proportion of Trump supporters - proportion of Clinton supporters) mentioned in Part (a) are away from 0?

48 of the 95% CIs are away from 0 ii. Among the CIs that are away from 0 how many of them predict the winning candidate of the state correctly?

Of the 48 95% CIs that are away from 0, 43 of the them predict the winning candidate of the state correctly

```
# number of states 95% CI that don't cover 0
length(significant[significant == T])
[1] 48
# of those states the number that correctly predicted winning
# candidate of the state
a = (significant == rep(T, 51) & same.sign == rep(T, 51))
length(a[a == T])
[1] 43
```

iii. How many of the 95% CIs cover 0?
3 of the 95% CIs cover 0

iv. Among the CIs that cover 0 how many of them predict the winning candidate of the state correctly?
Of the 3 95% CIs that cover 0, 2 of the them predict the winning candidate of the state correctly.

```
# number of states 95% CI that cover 0
length(significant[significant == F])
[1] 3
# of those states the number that correctly predicted winning
# candidate of the state
a = (significant == rep(F, 51) & same.sign == rep(T, 51))
length(a[a == T])
[1] 2
```

Part c

The error for a prediction is defined as

Prediction error := Predicted value - Actual value

The errors in the polls are actually not much bigger than that in previous elections (See NY Times article here). However, the problem is that the polls that over estimated Clinton's votes are polls for the battleground states. According to your answers in part b, how many of the CIs with either strictly-positive or strictly-negative interval estimates predict the winner of the state incorrectly? List the names of the states that associate with these CIs.

5 of the CIs with either strictly-positive or strictly-negative interval estimates predict the winner of the state incorrectly. These states are Florida, Michigan, North Carolina, Pennsylvania, and Wisconsin

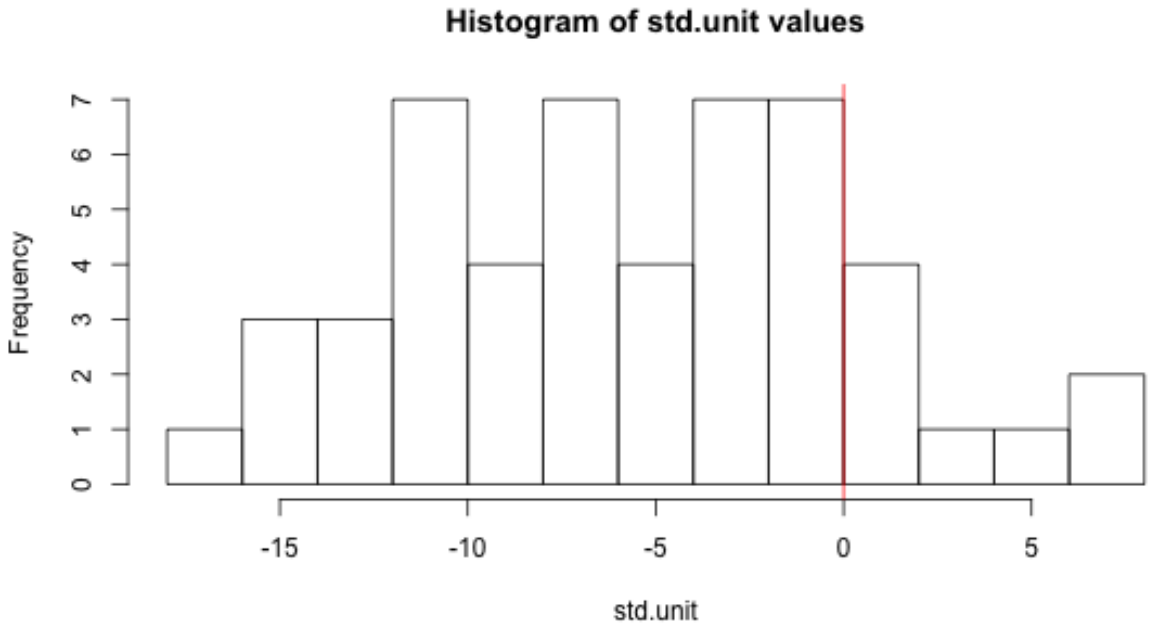
```
# from b) 48 total and 43 correctly predicted
48 - 43
[1] 5
# state names that incorrectly predicted with 95% CI that don't
# cover 0
a = (significant == rep(T, 51) & same.sign == rep(F, 51))
election[a == T, 1]
[1] Florida        Michigan       North Carolina Pennsylvania
[5] Wisconsin
51 Levels: Alabama Alaska Arizona Arkansas ... Wyoming
```

Part d

Make a histogram for the values of `std.unit` and mark the location of `x = 0` with a red vertical line. Do the errors in the estimates look like they have mean zero?

The errors in the estimates do no have a mean zero as there is a greater distribution to the left of the redline. Mean is closer to -5.5

```
# histogram of of std.unit with red line at x = 0
hist(election$std.unit, breaks = 10, main = "Histogram of std.unit values",
    xlab = "std.unit")
abline(v = 0, col = "red")
```



Part e

How many states have poll differences (i.e., proportion of Trump supporters - proportion of Clinton supporters) that are smaller than the actual election difference for the state?
There are 43 states that have poll differences that are smaller than the actual election difference for the state Under the following set of conditions:

- the errors for all 50 states plus D.C. are independent
- there is equally likely chance for a combined poll difference to be greater or less than the actual election difference for the state

what is the chance that you would observe this many or more states with poll differences smaller than the actual election difference? (Hint: under the set of conditions above what is the distribution of the number of locations with poll difference less than the actual election difference?)

Based on your data is it reasonable to assume that the two conditions mentioned above are met? Explain why.

There is a 3.43e-05 % chance that 43 or more states have poll differences smaller than the actual election difference. Based on the data it is not reasonable to think the two conditions mentioned are met because when the conditions are true the probability that 43 or more states have poll differences smaller than the actual election difference is very small(very close to 0%), thus conditions aren't met.

```
# number of states that poll differences are smaller than vote
# differences
```

```
num = length(election$deviation[election$deviation < 0])
num
[1] 43
# probability assuming condtions are true
sum(dbinom(x = num:51, size = 51, prob = 0.5))
[1] 3.433559e-07
```

How did the polls miss the results?

Question 4

When being asked how the polls missed the election result two writers on fivethirtyeight.com mentioned that

> While the errors were nationwide, they were spread unevenly. The more whites without college degrees were in a state, the more Trump outperformed his FiveThirtyEight polls-only adjusted polling average, suggesting the polls underestimated his support with that group.

(click here for reference)

We would like to investigate the relationships of the prediction errors in standard units with two variables (the percentage of whites and the percentage of people without college degrees).

part a

Read in the dataset `county_facts.csv` and extract out the columns:

- **area_name** Name of region
- **RHI825214** White alone, not Hispanic or Latino, percent, 2014
- **EDU685213** Bachelor's degree or higher, percent of persons age 25+, 2009-2013

Call the resulting dataset `demographic`.

Use the `merge()` function to add the two columns (`RHI825214` and `EDU685213`) of `demographic` to `election` and match the rows by using the state names. Call the resulting data frame `election.dem`.

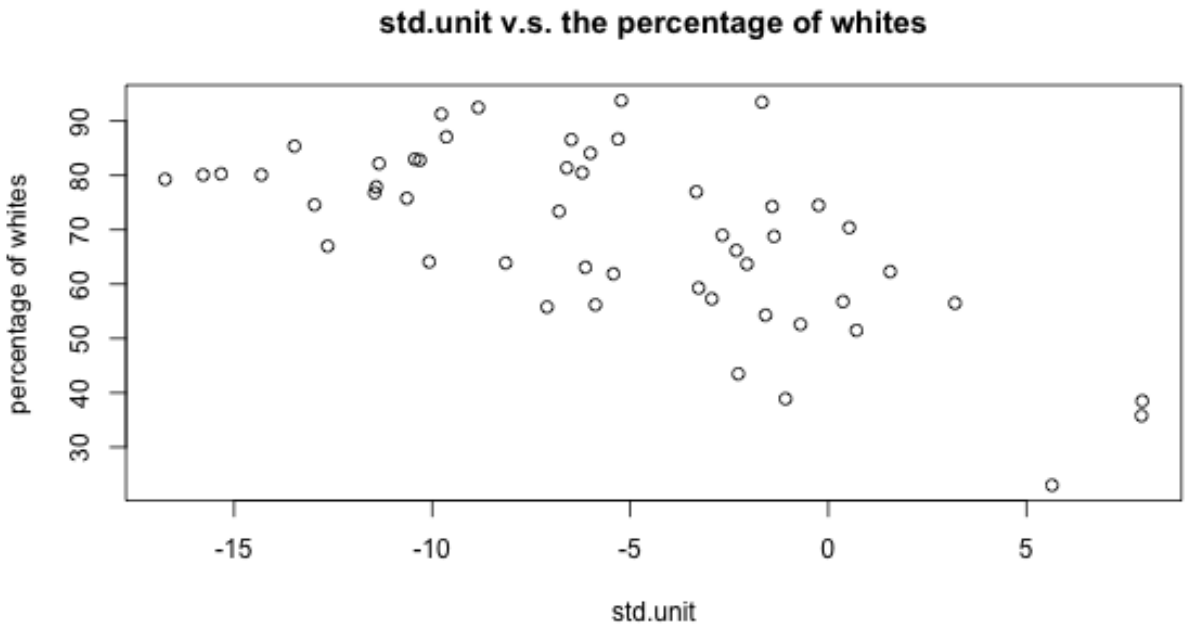
```
# read 3 colums from file and merge with election
dem = select(read.csv("county_facts.csv"), area_name, RHI825214,
    EDU685213)
election.dem = merge(election, dem, by.x = "state", by.y = "area_name")
```

part b

With `election.dem` make two scatterplots, one for the errors in standard units `std.unit` v.s. the percentage of whites, and another for `std.unit` v.s. the percentage of people without college degrees. For the second graph note that you will need to transform the variable `EDU685213`. Comment on whether you see a positive or negative trend on the scatterplots. For the areas with high percentages of non-hispanic white voters did the poll tend to over or under estimate how well Trump would do in the actual election? What about in the areas with high percentages of non-college-graduate voters?

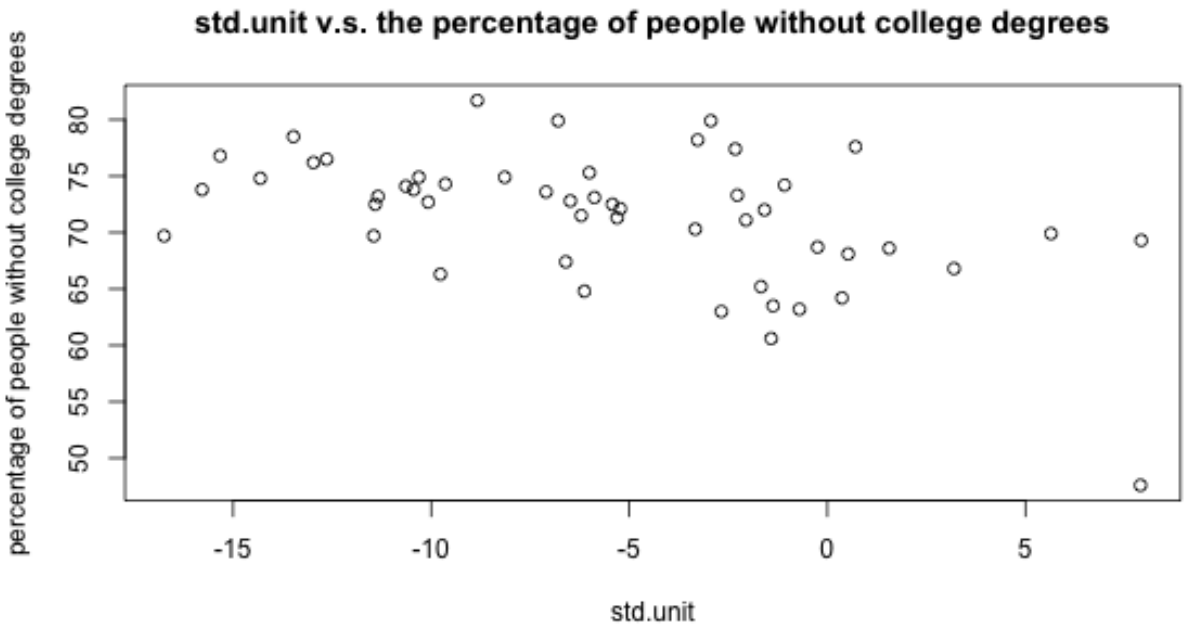
There is a negative trend on the scatterplots. For both areas with high percentages of non-hispanic white voters and areas with high percentages of non-college-graduate voters the polls tend to under estimate how well Trump would do in the actual election.

```r
# plot std.unit vs perc of whites
plot(election.dem$std.unit, election.dem$RHI825214, main = "std.unit v.s. the percentage of whites",
    xlab = "std.unit", ylab = "percentage of whites")
```




```r
# abline(lm(election.dem$RHI825214 ~ election.dem$std.unit))

# plot std.unit vs perc of people without degrees
plot(election.dem$std.unit, 100 - election.dem$EDU685213, main = "std.unit v.s. the percentage of peopl
    xlab = "std.unit", ylab = "percentage of people without college degrees")
```




```
# abline(lm(100 - election.dem$EDU685213 ~
# election.dem$std.unit))
```