Republican and Democrats Voting Difficulty

Oscar Casas

Oct 18 2021

Foundational Exercises – Applied Practice

A paired t-test specifically tests for the mean difference between two metric variables. Because the paired sample in this case is ordinal, calculating the distance/difference between values on a Likert Scale would result in a non-sensical value. Specifically, the interval between two adjacent points on the scale may not be equal to the interval between a different pair of adjacent points on the scale, therefore defining an ordinal rather than metric value. Taking a mean/stdev amongst these categorical differences means we will pass non-sensical values into our computed test statistic. This would render results ultimately uninterpretable.

Foundational Exercises – Test Assumptions

Proof Strategy Workshop: Expectation

1 sample answer.

two-sample t-test assumptions:

-independent samples -metric data -population is approximately normal, unless sample size is large such that -CLT applies -similar variances

##Statistical Analysis¶

```
temp <- tempfile()

download.file("https://electionstudies.org/anes_timeseries_2020_stata_20210719/",temp)

data.df <- read_dta(unz(temp, "anes_timeseries_2020_stata_20210719.dta"))

head(data.df)
```

```
## # A tibble: 6 x 1,771
##   version V200001 V160001_orig V200002 V200003 V200004 V200005  V200006  V200007
##   <chr>     <dbl>    <dbl+lbl> <dbl+l> <dbl+l> <dbl+l> <dbl+l> <dbl+lb> <dbl+lb>
## 1 ANES20~  200015       401318 3 [3. ~ 2 [2. ~ 3 [3. ~ 0 [0. ~ -2 [-2.~ -2 [-2.~
## 2 ANES20~  200022       300261 3 [3. ~ 2 [2. ~ 3 [3. ~ 0 [0. ~  4 [4. ~ -1 [-1.~
## 3 ANES20~  200039       400181 3 [3. ~ 2 [2. ~ 3 [3. ~ 0 [0. ~ -2 [-2.~ -2 [-2.~
## 4 ANES20~  200046       300171 3 [3. ~ 2 [2. ~ 3 [3. ~ 0 [0. ~ -2 [-2.~ -2 [-2.~
## 5 ANES20~  200053       405145 3 [3. ~ 2 [2. ~ 3 [3. ~ 1 [1. ~ -2 [-2.~ -2 [-2.~
## 6 ANES20~  200060       400374 3 [3. ~ 2 [2. ~ 3 [3. ~ 0 [0. ~ -2 [-2.~ -2 [-2.~
## # ... with 1,762 more variables: V200008 <dbl+lbl>, V200009 <dbl+lbl>,
## #   V200010a <dbl>, V200010b <dbl>, V200010c <dbl>, V200010d <dbl>,
## #   V200011a <dbl>, V200011b <dbl>, V200011c <dbl>, V200011d <dbl>,
## #   V200012a <dbl>, V200012b <dbl>, V200012c <dbl>, V200012d <dbl>,
## #   V200013a <dbl>, V200013b <dbl>, V200013c <dbl>, V200013d <dbl>,
## #   V200014a <dbl>, V200014b <dbl>, V200014c <dbl>, V200014d <dbl>,
```

```
## #   V200015a <dbl>, V200015b <dbl>, V200015c <dbl>, V200015d <dbl>, ...
```

Initial analysis of suitable variables for identifying R vs D, and target variables

- V201018: PARTY OF REGISTRATION: - 3197 R/D - 1029 Independent/None
- V201231x Party ID: - 8245 valid results (R/D/Ind) - So this is the variable I think would be good to use – I think I mixed this one up with party registration when we spoke earlier. This is also a Likert variable (strong D, not strong D, independent D and vice versa with republican plus a single independent option). I think we could make an argument that your self-identification is more applicable to voting behavior than party registration (not sure how hard of an argument that is to make... or if it is actually valid, would need to do some research). But that is an option.
- V201228 DOES R THINK OF SELF AS DEMOCRAT, REPUBLICAN, OR INDEPENDENT: - 5428 R/D - 2527 Independent
- V201230 NO PARTY IDENTIFICATION - CLOSER TO DEMOCRATIC PARTY OR REPUBLICAN PARTY: - 1855 R/D
- V202119: HOW DIFFICULT WAS IT FOR R TO VOTE: - Likert variable (not difficult = 1 -> extremely difficult = 5) - 6401 valid results - I think this is the "dependent" variable we should use. We have a high amount of responses so no issues with sample size
- V202443: WHICH PARTY DOES R FEEL CLOSEST TO: - 5810 valid results (R/D)

```
filt <- data.df[, c("V201231x","V201230","V201228","V202119")]

names(filt) = c('Party ID','Party Lean','Party Self ID','Voting Difficulty')

head(filt)
```

```
## # A tibble: 6 x 4
##          `Party ID`          `Party Lean`    `Party Self ID` `Voting Difficulty`
##           <dbl+lbl>             <dbl+lbl>          <dbl+lbl>           <dbl+lbl>
## 1 7 [7. Strong Rep~ -1 [-1. Inapplicable] 2 [2. Republican]  -1 [-1. Inapplicab~
## 2 4 [4. Independen~  2 [2. Neither {VOL ~ 5 [5. Other party~  1 [1. Not difficu~
## 3 3 [3. Independen~  3 [3. Closer to Dem~ 3 [3. Independent]  2 [2. A little di~
## 4 6 [6. Not very s~ -1 [-1. Inapplicable] 2 [2. Republican]   1 [1. Not difficu~
## 5 4 [4. Independen~  2 [2. Neither {VOL ~ 3 [3. Independent]  2 [2. A little di~
## 6 3 [3. Independen~  3 [3. Closer to Dem~ 3 [3. Independent]  2 [2. A little di~
```

The code below extracts true responses from voting difficulty data. If the response was scored on the 1-5 Likert Scale,then the data is unmodified, if it is not scored then we delete that row.

```
df2 <- filt[!(filt$`Voting Difficulty` < 1 | filt$`Voting Difficulty` > 5),]

df2$`Party ID` = ifelse(df2$`Party ID` == 1 | df2$`Party ID` == 2 | df2$`Party ID` == 3, "D", ifelse(df2

df2$`Party Lean` = ifelse(df2$`Party Lean` == 3, "D", ifelse(df2$`Party Lean` == 2, "Ind", ifelse(df2$`
df2$`Party Self ID` = ifelse(df2$`Party Self ID` == 1, "D", ifelse(df2$`Party Self ID` == 2, "R", ifels

head(df2)
```

```
## # A tibble: 6 x 4
##   `Party ID` `Party Lean` `Party Self ID`         `Voting Difficulty`
##   <chr>      <chr>        <chr>                           <dbl+lbl>
## 1 Ind        Ind          <NA>           1 [1. Not difficult at all]
## 2 D          D            Ind            2 [2. A little difficult]
## 3 R          <NA>         R              1 [1. Not difficult at all]
## 4 Ind        Ind          Ind            2 [2. A little difficult]
## 5 D          D            Ind            2 [2. A little difficult]
```

```
## 6 R          <NA>         R               1 [1. Not difficult at all]
```

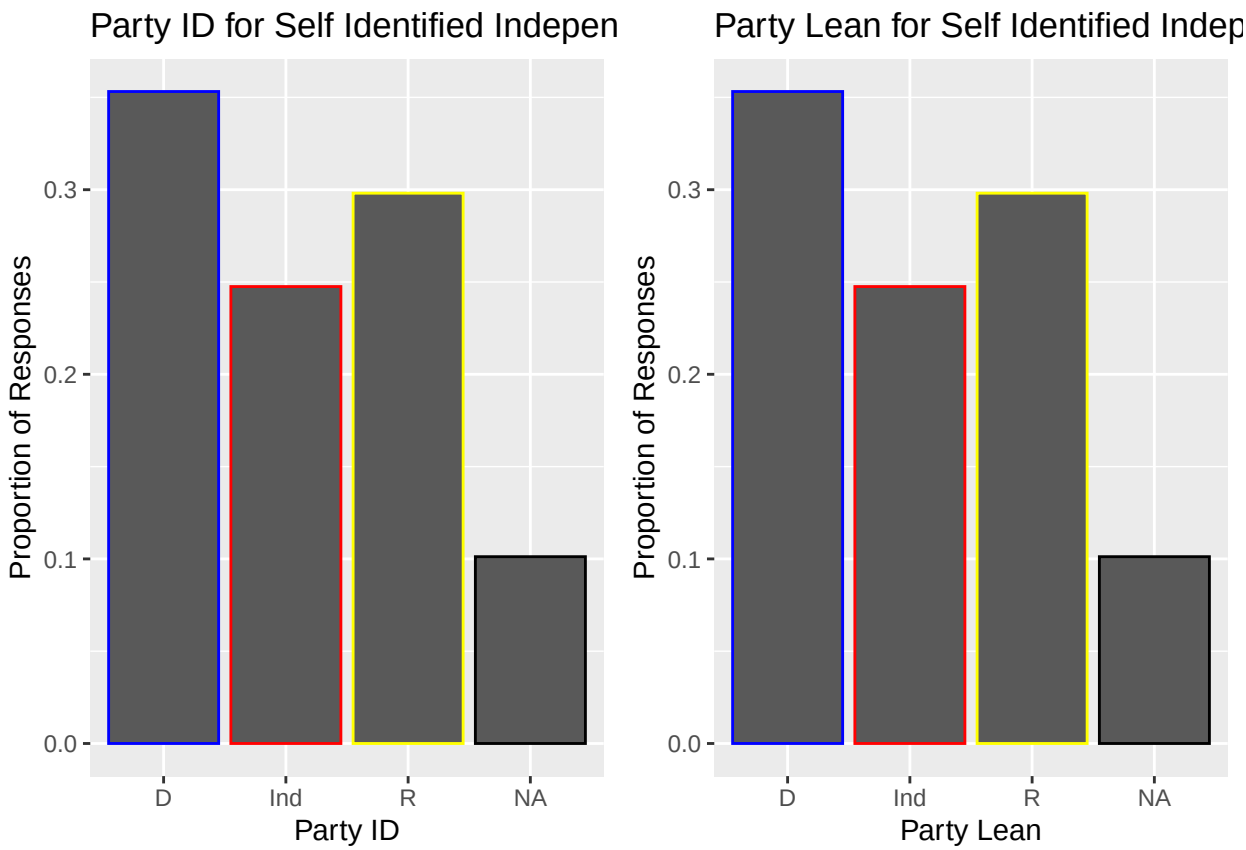
-The code above modifies the party ID column to be on a R/D/Ind/NA Scale where the specific scale can be given by 1) -9. Refused 2) -8. Don't know 3) 1. Strong Democrat 4) 2. Not very strong Democrat 5) 3. Independent-Democrat 6) 4. Independent 7) 5. Independent-Republican 8) 6. Not very strong Republican 9) 7. Strong Republican

-The code above modifies the party lean column to store R/D/Ind/NA if party lean is 1,2,3,other respectively where the specified scale can be given by 1) -9. Refused 2) -8. Don't know 3) -1. Inapplicable 4) 1. Closer to Republican 5) 2. Neither {VOL in video and phone} 6) 3. Closer to Democratic

-The code above specifies D/R/Ind/NA if party self id is 1,2,3,other respectively where the specified scale can be given by 1) -9. Refused 2) -8. Don't know 3) -4. Technical error 4) 0. No preference {VOL - video/phone only} 5) 1. Democrat 6) 2. Republican 7) 3. Independent 8) 5. Other party {SPECIFY}

```
ind_voters = df2[(df2$`Party Self ID` == "Ind"),]
```

```
require(gridExtra)
ind_voters_party_id <- ggplot(ind_voters, aes(x = `Party ID`)) +
  geom_bar(color=c('blue','red','yellow','black'),aes(y = (..count..)/sum(..count..))) + ggtitle("Party

ind_voters_party_lean <- ggplot(ind_voters, aes(x = `Party Lean`)) +
  geom_bar(color=c('blue','red','yellow','black'),aes(y = (..count..)/sum(..count..))) + ggtitle("Party
grid.arrange(ind_voters_party_id,ind_voters_party_lean,ncol = 2)
```




```
df2$`Party Classification` = ifelse(df2$`Party ID` == "D", "D", ifelse(df2$`Party ID` == "R", "R", ifels

table(df2$`Party Classification`)
```

```
##
```

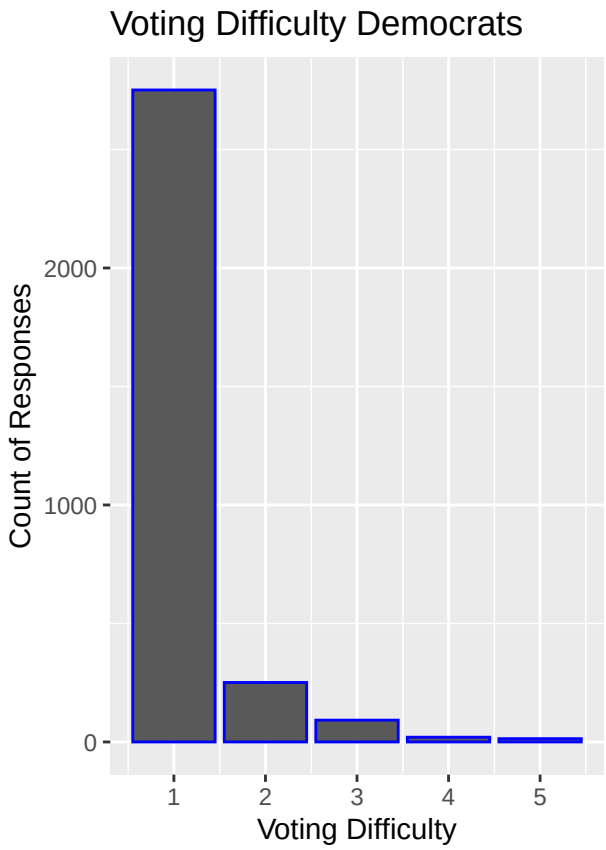
```
##    D  Ind    R
## 3128  499 2700
```

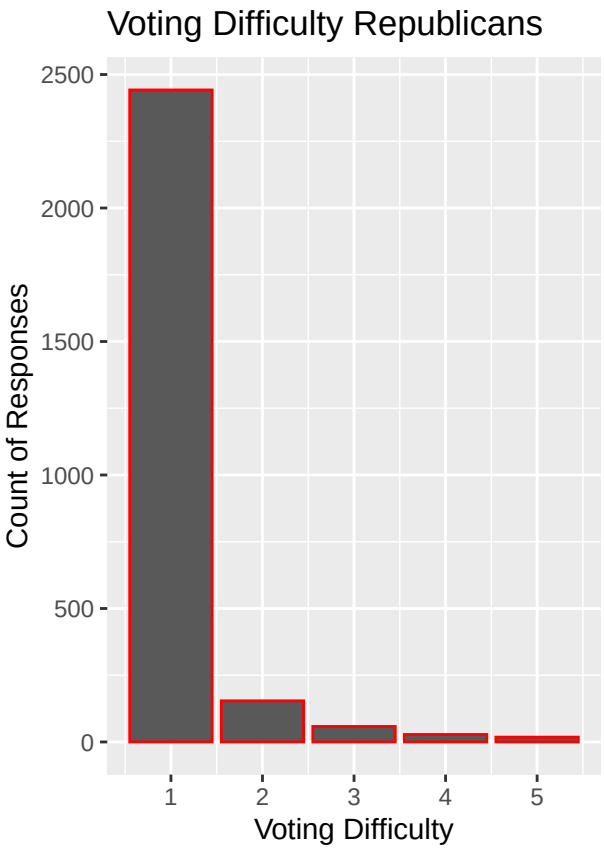
```
df2 = df2[(df2$`Party Classification` == "D" | df2$`Party Classification` == "R"),]

voter_data = na.omit(df2[,c(5,4)])

dem_data = voter_data[(voter_data$`Party Classification` == "D"),]
rep_data = voter_data[(voter_data$`Party Classification` == "R"),]


head(voter_data)
```

```
## # A tibble: 6 x 2
##   `Party Classification`         `Voting Difficulty`
##   <chr>                                    <dbl+lbl>
## 1 D                       2 [2. A little difficult]
## 2 R                       1 [1. Not difficult at all]
## 3 D                       2 [2. A little difficult]
## 4 R                       1 [1. Not difficult at all]
## 5 D                       2 [2. A little difficult]
## 6 D                       1 [1. Not difficult at all]
```

```
require(gridExtra)
plot1 <- ggplot(dem_data, aes(x = `Voting Difficulty`)) +
  geom_bar(color='blue') + ggtitle("Voting Difficulty Democrats") + ylab("Count of Responses")
plot2 <- ggplot(rep_data, aes(x = `Voting Difficulty`)) +
  geom_bar(color='red')+ ggtitle("Voting Difficulty Republicans") + ylab("Count of Responses")
grid.arrange(plot1, plot2, ncol=2)
```

```
## Don't know how to automatically pick scale for object of type haven_labelled/vctrs_vctr/double. Defau
## Don't know how to automatically pick scale for object of type haven_labelled/vctrs_vctr/double. Defau
```





Hypothesis Test Selection:

Our goal is to evaluate whether democratic or republican voters experience more voting difficulty. Upon analyzing the data, our reponse variable **Voting Difficulty** is an ordinal variable measured on a Likert Scale from 1-5 where 1 signifies less difficulty while 5 signifies more difficulty. Ordinal data results in the need for a **non-parametric test**. Additionally, we are comparing between two distinct groups without a natural pairing, **so a paired test is ruled out**. The samples here are independent as the perception of voting difficulty for one respondent does not inform on another's response. Finally, upon evaluating the response variable density across party lines, voting difficulty has a similar distribution in both cases. Therefore, the characteristics listed above meet the assumptions for a **Wilcoxon Rank Sum Test**.

```
res <- wilcox.test(dem_data$`Voting Difficulty`, rep_data$`Voting Difficulty`)
res
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  dem_data$`Voting Difficulty` and rep_data$`Voting Difficulty`
## W = 4323774, p-value = 0.003541
## alternative hypothesis: true location shift is not equal to 0
```

Wilcoxon Rank Sum Two Tailed Test. Null is voter difficulty is equal, alternative is voter difficulty is not equal

```
res_2 <- wilcox.test(dem_data$`Voting Difficulty`, rep_data$`Voting Difficulty`, alternative = "less")
res_2
```

```
##
##  Wilcoxon rank sum test with continuity correction
```

```
##
## data:  dem_data$`Voting Difficulty` and rep_data$`Voting Difficulty`
## W = 4323774, p-value = 0.9982
## alternative hypothesis: true location shift is less than 0
```

Wilcoxon Rank Sum One Tailed Test. Null is voter difficulty is equal, alternative is dem voter difficulty is greater

```
res_2 <- wilcox.test(dem_data$`Voting Difficulty`, rep_data$`Voting Difficulty`, alternative = "greater
res_2
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  dem_data$`Voting Difficulty` and rep_data$`Voting Difficulty`
## W = 4323774, p-value = 0.00177
## alternative hypothesis: true location shift is greater than 0
```

Wilcoxon Rank Sum One Tailed Test. Null is voter difficulty is equal, alternative is dem voter difficulty is less