Case Study 2

Emily Gentles (Writer) Weiyi Liu (Coordinator and Checker)
Jack McCarthy (Presenter) Qinzhe Wang (Coder)

10/28/2021

Explication of abbreviations: https://s3.amazonaws.com/dl.ncsbe.gov/data/layout_ncvoter.txt

```
# contains info about the aggregate counts of voters who actually voted by demographic variables

votes <- read.table("history_stats_20201103.txt", header = TRUE, fill = TRUE, sep = '\t')

# contains info about the aggregate counts of registered voters by demographic variables
registers <- read.table("voter_stats_20201103.txt", header = TRUE, fill = TRUE, sep = '\t')
```

```
# set "" or " " to NA
registers[registers == ""] <- NA
registers[registers == " "] <- NA
votes[votes == " "] <- NA

# unique(registers$election_date) # "11/03/2020" NA
# unique(registers$stats_type) # "history" NA
# unique(registers$update_date) # "01/13/2021" NA

# remove above three columns

registers <- registers %>%
  select(-election_date, -stats_type, -update_date)
votes <- votes %>%
  select(-election_date, -stats_type, -update_date)

votes <- votes %>%
  mutate(total_voters = as.numeric(total_voters))

registers <- registers %>%
  mutate(total_voters2 = as.numeric(total_voters))
```

```
votes <- votes %>%
  group_by(county_desc, age, party_cd, race_code, ethnic_code, sex_code) %>%
  summarize(total_vot = sum(total_voters,na.rm =T))
```

```
## `summarise()` has grouped output by 'county_desc', 'age', 'party_cd', 'race_code', 'ethnic_code'. You
```

```
data <- registers %>%
  group_by(county_desc, party_cd, race_code, ethnic_code, sex_code, age) %>%
  summarize(total_reg = sum(total_voters2, na.rm= T)) %>%
  left_join(votes, by = c("county_desc", "age", "party_cd", "race_code",
                          "ethnic_code", "sex_code"))
```

```
## `summarise()` has grouped output by 'county_desc', 'party_cd', 'race_code', 'ethnic_code', 'sex_code
```

```
data <- data %>%
  mutate(total_vot = as.numeric(total_vot),
         total_reg = as.numeric(total_reg))
         # county_desc = as.factor(county_desc),
         # party_cd = as.factor(party_cd),
         # race_code = as.factor(race_code),
         # sex_code = as.factor(sex_code),
         # ethnic_code = as.factor(ethnic_code),
         # age = as.factor(age))
```

```
#data <- data %>%
#  filter( !is.na(total_reg))
data$total_vot <- ifelse(is.na(data$total_vot), 0, data$total_vot)

data$total_vot <- ifelse(data$total_vot <= data$total_reg, data$total_vot, data$total_reg)

data <- data %>%
  drop_na()

# 13066 rows
```

```
set.seed(10)
counties <- sample(unique(data$county_desc), 30)

data <- data %>%
  filter(county_desc %in% counties)
```

```
data$race_code <- recode_factor(data$race_code, M = "Multiracial", U = "Undesignated")
data2 <- data %>%
  drop_na() %>%
  mutate(total_not_vote = total_reg - total_vot) %>%
  pivot_longer(cols = c("total_vot", "total_not_vote"), names_to = "vote_or_not", values_to = "prob") %
  mutate(vote_or_not = ifelse(vote_or_not == "total_vot", 1, 0)) %>%
  mutate(row_expand = map(prob, ~rep_len(1, .x))) %>%
  unnest(cols = c(row_expand)) %>%
  select(-prob, -row_expand)
```

EDA

```
table(data2$vote_or_not)
```

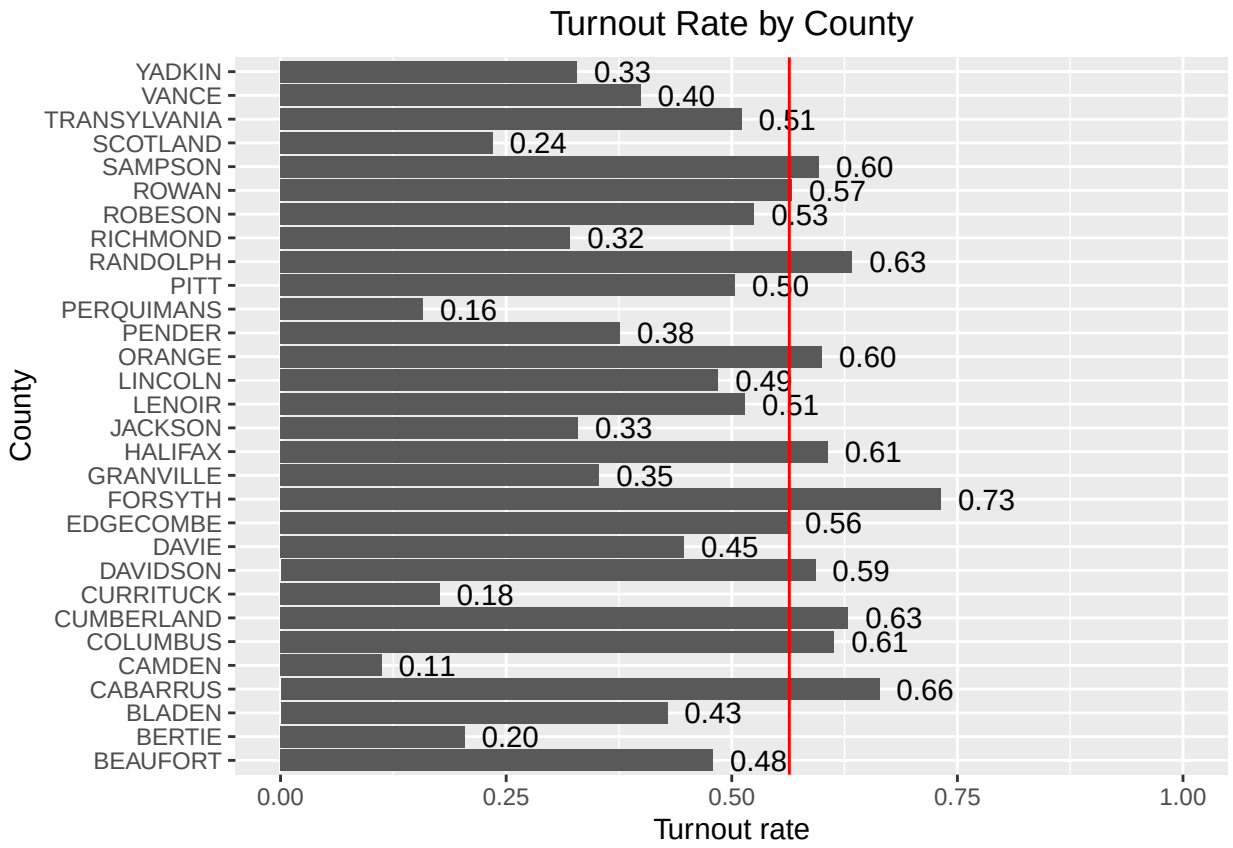
```
##
##     0     1
## 31888 41212
```

turnout rate

```
turnout_rate <- sum(data$total_vot) / sum(data$total_reg)
data.frame(group = "Total", turnout_rate = turnout_rate)
```

```
##   group turnout_rate
## 1 Total    0.5637756
```

```
data %>%
  group_by(county_desc) %>%
  summarise(total_reg = sum(total_reg),
            total_vot = sum(total_vot), .groups = "drop") %>%
  mutate(turnout_rate = total_vot / total_reg) %>%
  ggplot( aes(x = county_desc, y = turnout_rate)) +
  geom_bar(stat = "identity") +
  ylim(0,1) +
  xlab("County") + ylab("Turnout rate") +
  geom_text(aes(label=format(turnout_rate, digits = 2)), hjust = -0.3) +
  ggtitle("Turnout Rate by County") +
    theme(plot.title = element_text(hjust = 0.5)) +
  geom_hline(yintercept = turnout_rate, color = 'red') +
  coord_flip()
```



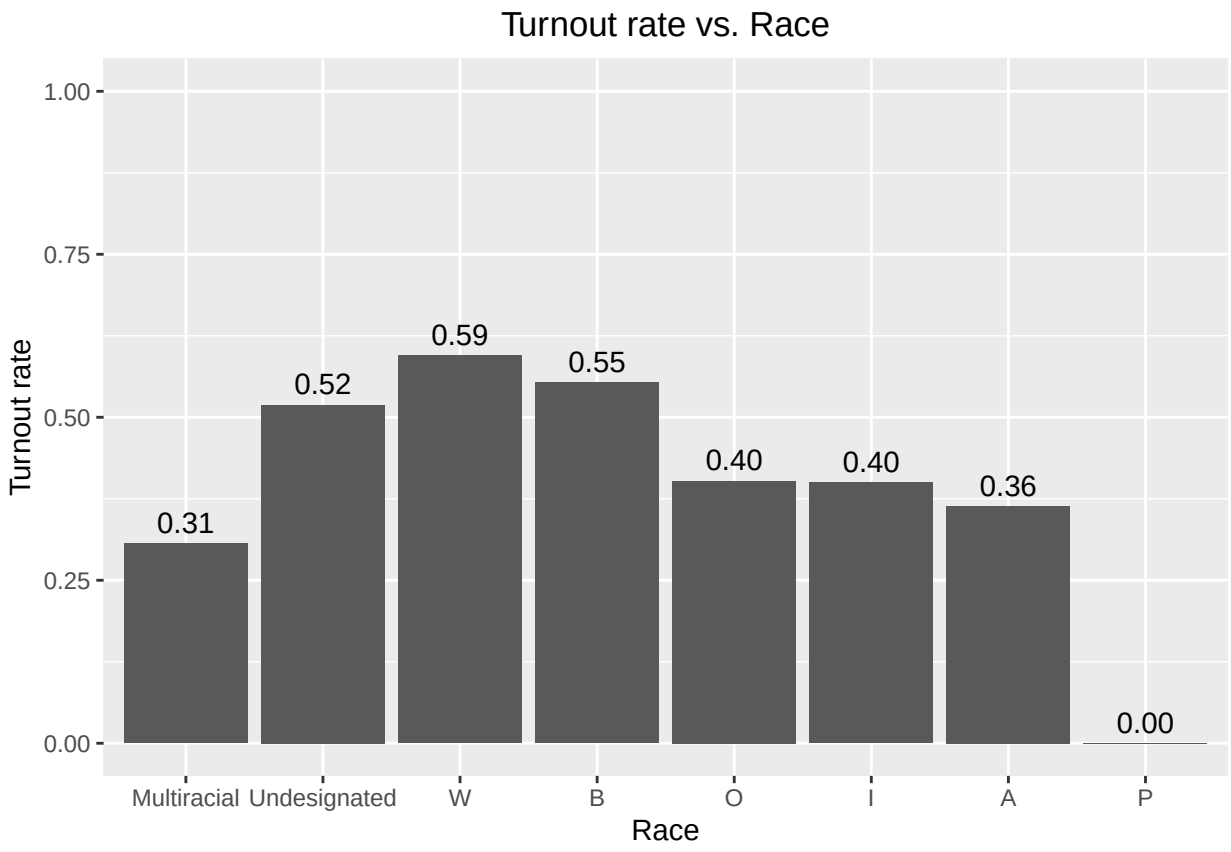

```
data %>%
  group_by(county_desc) %>%
  summarize(n = n())
```

```
## # A tibble: 30 x 2
##    county_desc     n
##    <chr>       <int>
##  1 BEAUFORT      105
##  2 BERTIE         44
##  3 BLADEN         74
##  4 CABARRUS      295
##  5 CAMDEN         23
##  6 COLUMBUS      103
##  7 CUMBERLAND    437
##  8 CURRITUCK      65
##  9 DAVIDSON      208
## 10 DAVIE          75
## # ... with 20 more rows
```

```
# no extremely small sample size
```

race

```
data %>%
  group_by(race_code) %>%
  summarise(total_reg = sum(total_reg),
            total_vot = sum(total_vot), .groups = "drop") %>%
  mutate(turnout_rate = total_vot / total_reg) %>%
  select(group = race_code, turnout_rate) %>%
  ggplot( aes(x = group, y = turnout_rate)) +
  geom_bar(stat = "identity") +
  ylim(0,1) +
  xlab("Race") + ylab("Turnout rate") +
  geom_text(aes(label=format(turnout_rate, digits = 2)), vjust = -0.5) +
  ggtitle("Turnout rate vs. Race") +
    theme(plot.title = element_text(hjust = 0.5))
```




```
data %>%
  group_by(race_code) %>%
  summarize(n = n())
```

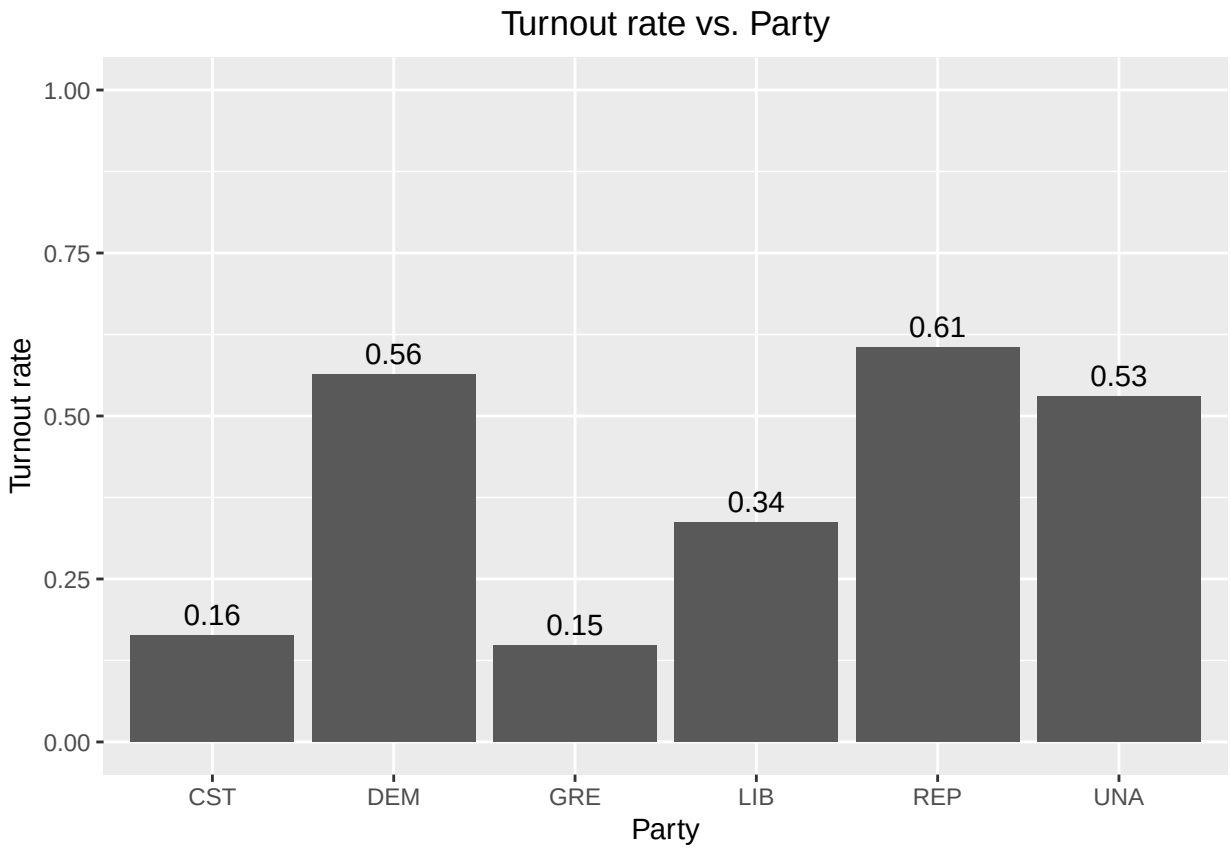
```
## # A tibble: 8 x 2
##   race_code        n
##   <fct>        <int>
## 1 Multiracial    275
## 2 Undesignated   859
## 3 W             1274
## 4 B              769
```

```
## 5 O              534
## 6 I              240
## 7 A              243
## 8 P                3
```

```
# race_code p has a small sample size, drop it

data$race_code <- droplevels(data$race_code, 'P')
data <- data %>%
  drop_na()
```

party

```
data %>%
  group_by(party_cd) %>%
  summarise(total_reg = sum(total_reg),
            total_vot = sum(total_vot), .groups = "drop") %>%
  mutate(turnout_rate = total_vot / total_reg) %>%
  select(group = party_cd, turnout_rate) %>%
  ggplot( aes(x = group, y = turnout_rate)) +
  geom_bar(stat = "identity") +
  ylim(0,1) +
  xlab("Party") + ylab("Turnout rate") +
  geom_text(aes(label=format(turnout_rate, digits = 2)), vjust = -0.5) +
  ggtitle("Turnout rate vs. Party") +
    theme(plot.title = element_text(hjust = 0.5))
```




```
data %>%
  group_by(party_cd) %>%
  summarize(n = n())
```

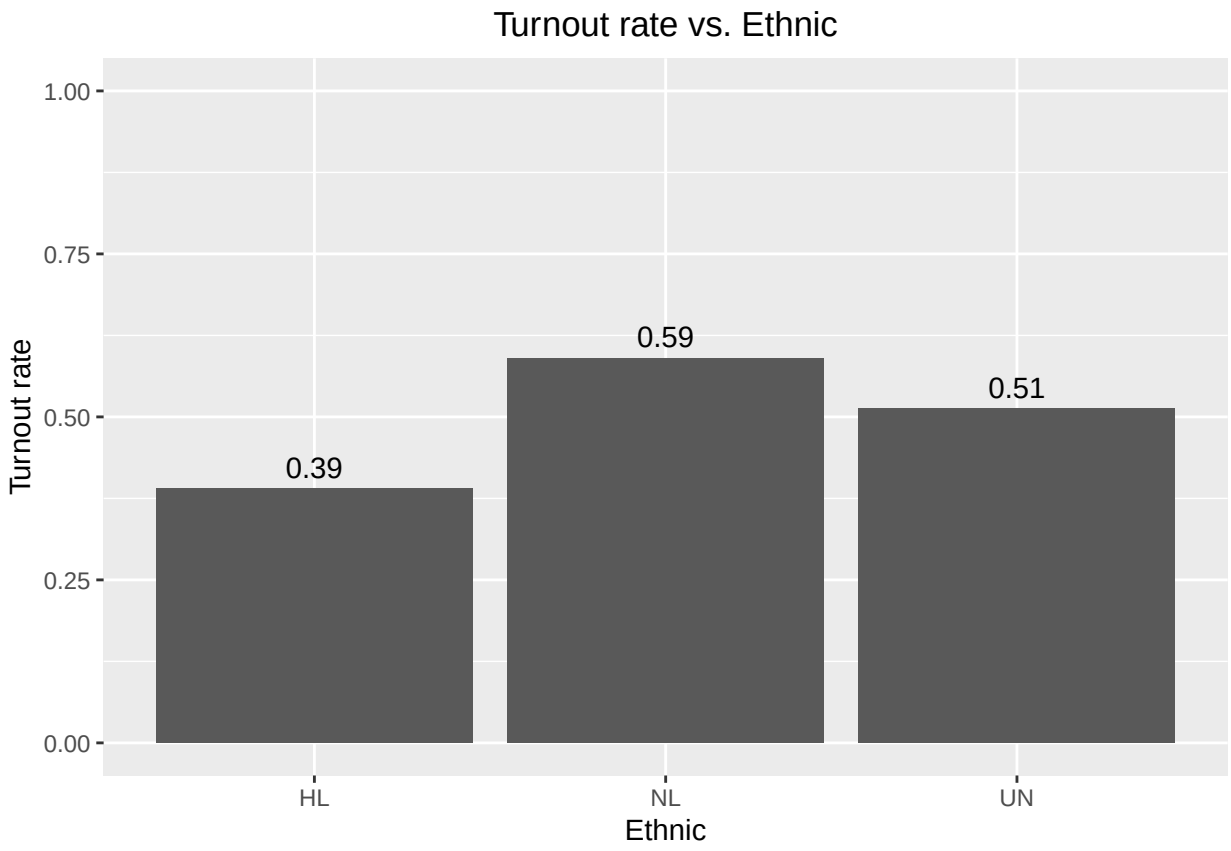
```
## # A tibble: 6 x 2
##   party_cd     n
##   <chr>    <int>
## 1 CST         50
## 2 DEM       1387
## 3 GRE         25
## 4 LIB        194
## 5 REP       1073
## 6 UNA       1465
```

```
# "CST","GRE" have small smaple size (50,25) and similar turnout rate (0.16,0.15)
data$party_cd <- ifelse(data$party_cd %in% c("CST","GRE"),"CST_GRE",data$party_cd)
```

ethnic groups

```
data %>%
  group_by(ethnic_code) %>%
  summarise(total_reg = sum(total_reg),
            total_vot = sum(total_vot), .groups = "drop") %>%
  mutate(turnout_rate = total_vot / total_reg) %>%
```

```
  select(group = ethnic_code, turnout_rate) %>%
  ggplot( aes(x = group, y = turnout_rate)) +
  geom_bar(stat = "identity") +
  ylim(0,1) +
  xlab("Ethnic") + ylab("Turnout rate") +
  geom_text(aes(label=format(turnout_rate, digits = 2)), vjust = -0.5) +
  ggtitle("Turnout rate vs. Ethnic") +
    theme(plot.title = element_text(hjust = 0.5))
```



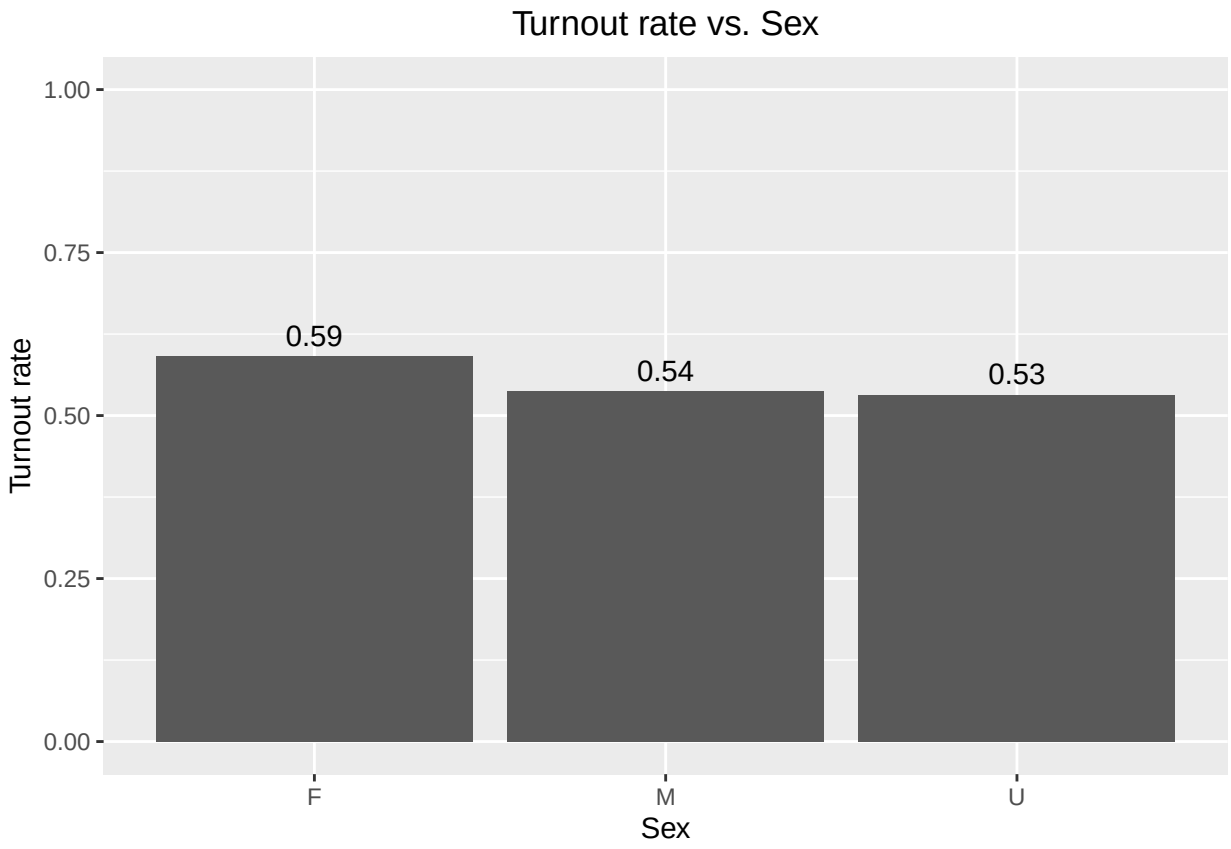

```
data %>%
  group_by(ethnic_code) %>%
  summarize(n = n())
```

```
## # A tibble: 3 x 2
##   ethnic_code     n
##   <chr>       <int>
## 1 HL            704
## 2 NL           1717
## 3 UN           1773
```

Sex

```
data %>%
  group_by(sex_code) %>%
```

```
  summarise(total_reg = sum(total_reg),
            total_vot = sum(total_vot), .groups = "drop") %>%
  mutate(turnout_rate = total_vot / total_reg) %>%
  select(group = sex_code, turnout_rate) %>%
  ggplot( aes(x = group, y = turnout_rate)) +
  geom_bar(stat = "identity") +
  ylim(0,1) +
  xlab("Sex") + ylab("Turnout rate") +
  geom_text(aes(label=format(turnout_rate, digits = 2)), vjust = -0.5) +
  ggtitle("Turnout rate vs. Sex") +
    theme(plot.title = element_text(hjust = 0.5))
```

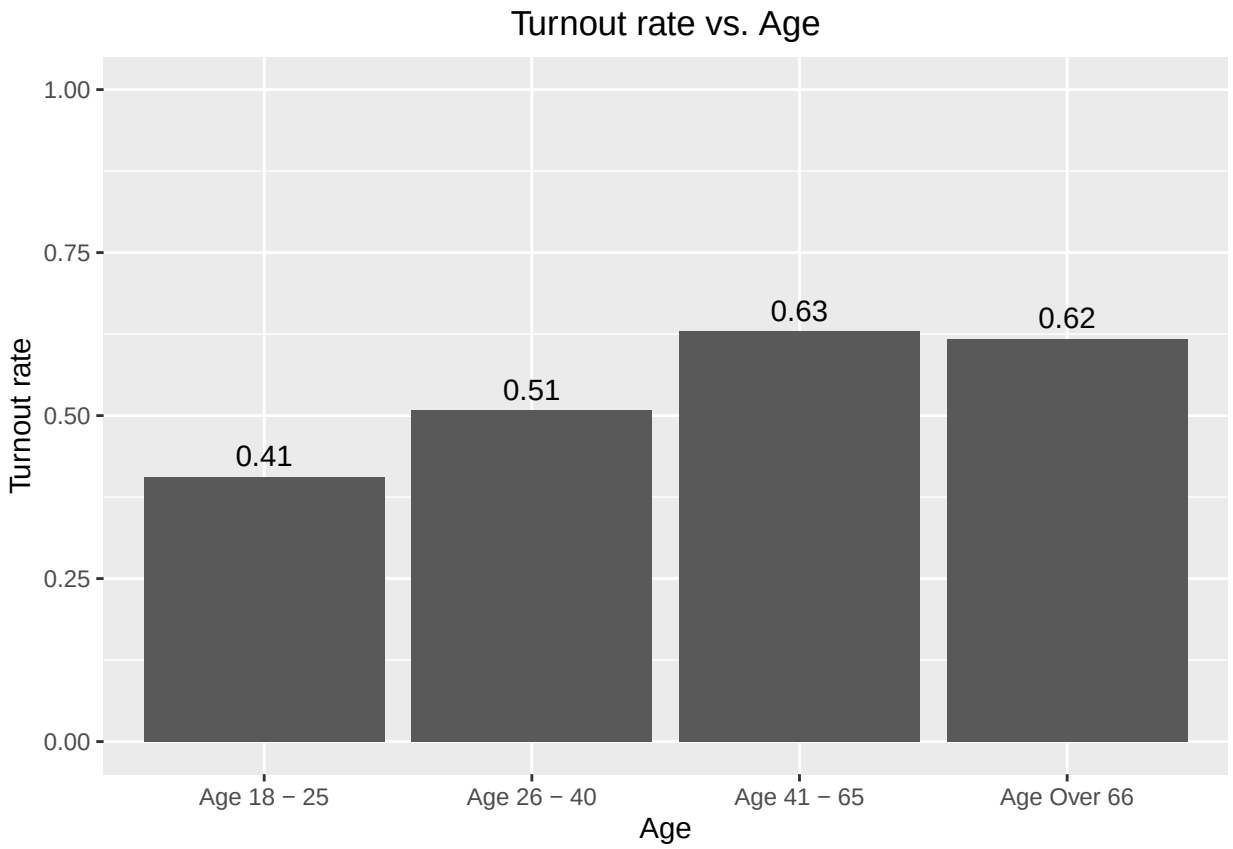



```
data %>%
  group_by(sex_code) %>%
  summarize(n = n())
```

```
## # A tibble: 3 x 2
##   sex_code     n
##   <chr>    <int>
## 1 F         1835
## 2 M         1780
## 3 U          579
```

age

```
data %>%
  group_by(age) %>%
  summarise(total_reg = sum(total_reg),
            total_vot = sum(total_vot), .groups = "drop") %>%
  mutate(turnout_rate = total_vot / total_reg) %>%
  select(group = age, turnout_rate) %>%
  ggplot( aes(x = group, y = turnout_rate)) +
  geom_bar(stat = "identity") +
  ylim(0,1) +
  xlab("Age") + ylab("Turnout rate") +
  geom_text(aes(label=format(turnout_rate, digits = 2)), vjust = -0.5) +
  ggtitle("Turnout rate vs. Age") +
    theme(plot.title = element_text(hjust = 0.5))
```



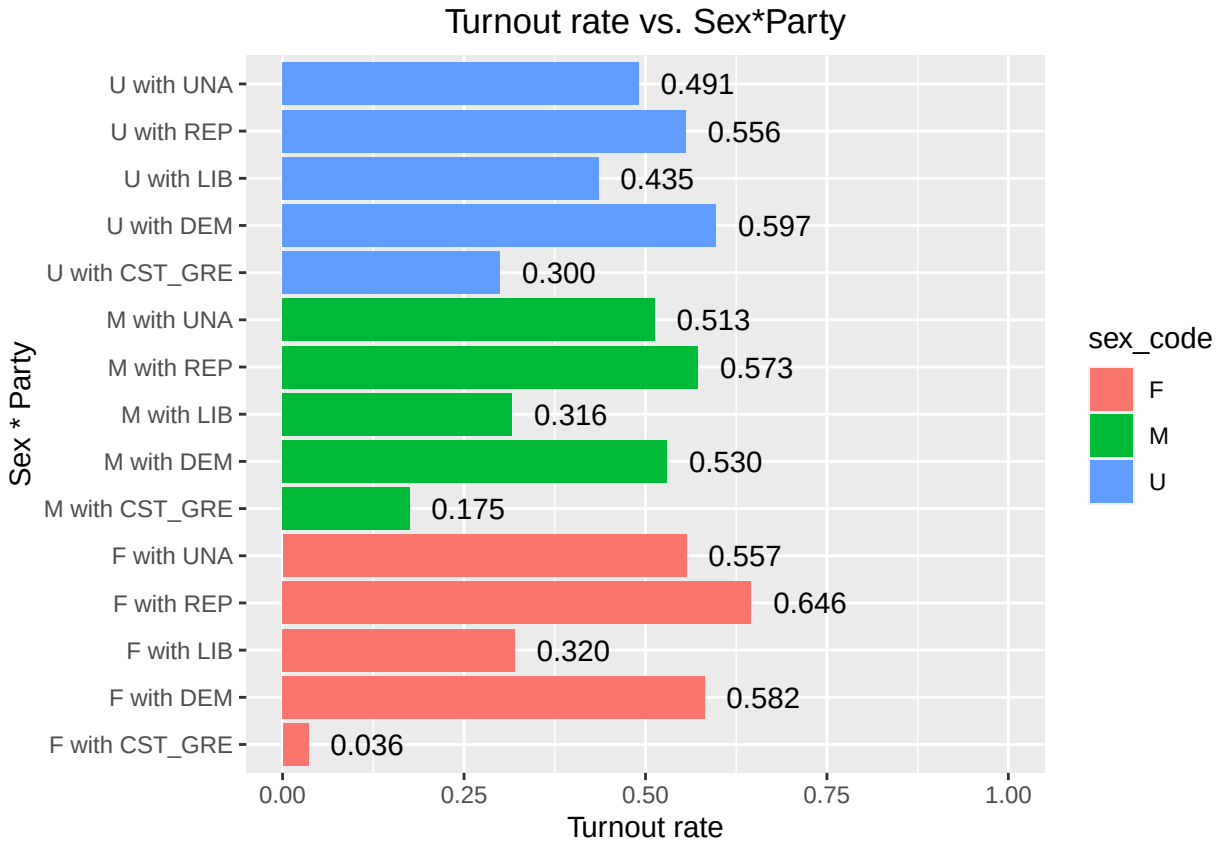

```
data %>%
  group_by(age) %>%
  summarize(n = n())
```

```
## # A tibble: 4 x 2
##   age             n
##   <chr>       <int>
## 1 Age 18 - 25  1032
## 2 Age 26 - 40  1171
```

```
## 3 Age 41 - 65  1248
## 4 Age Over 66   743
```

sex & party

```
data %>%
  group_by(sex_code, party_cd) %>%
  summarise(total_reg = sum(total_reg),
            total_vot = sum(total_vot), .groups = "drop") %>%
  mutate(turnout_rate = total_vot / total_reg,
         group = paste0(sex_code, " with ", party_cd)) %>%
  select(group = group, turnout_rate, sex_code, party_cd) %>%
  ggplot( aes(x = group, y = turnout_rate, fill = sex_code)) +
  geom_bar(stat = "identity") +
  ylim(0,1) +
  xlab("Sex * Party") + ylab("Turnout rate") +
  geom_text(aes(label=format(turnout_rate, digits = 2)), hjust = -0.3) +
  coord_flip() +
  ggtitle("Turnout rate vs. Sex*Party") +
    theme(plot.title = element_text(hjust = 0.5))
```



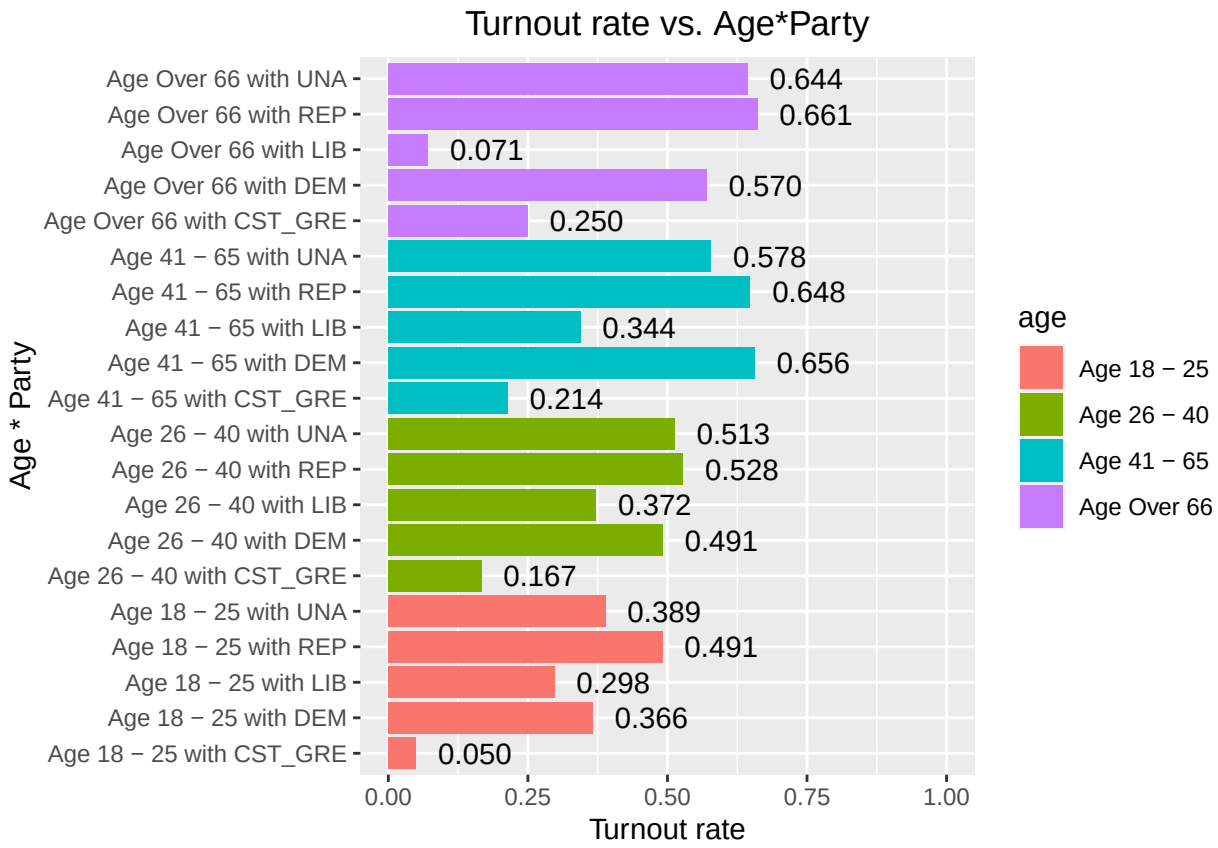

```
data %>%
  group_by(sex_code, party_cd) %>%
  summarize(n = n())
```

```
## `summarise()` has grouped output by 'sex_code'. You can override using the `.groups` argument.
```

```
## # A tibble: 15 x 3
## # Groups:   sex_code [3]
##    sex_code party_cd     n
##    <chr>    <chr>    <int>
##  1 F        CST_GRE     26
##  2 F        DEM        628
##  3 F        LIB         81
##  4 F        REP        450
##  5 F        UNA        650
##  6 M        CST_GRE     34
##  7 M        DEM        572
##  8 M        LIB         79
##  9 M        REP        480
## 10 M        UNA        615
## 11 U        CST_GRE     15
## 12 U        DEM        187
## 13 U        LIB         34
## 14 U        REP        143
## 15 U        UNA        200
```

age & party

```
data %>%
  group_by(age, party_cd) %>%
  summarise(total_reg = sum(total_reg),
            total_vot = sum(total_vot), .groups = "drop") %>%
  mutate(turnout_rate = total_vot / total_reg,
         group = paste0(age, " with ", party_cd)) %>%
  select(group = group, turnout_rate, age, party_cd) %>%
  ggplot( aes(x = group, y = turnout_rate, fill = age)) +
  geom_bar(stat = "identity") +
  ylim(0,1) +
  xlab("Age * Party") + ylab("Turnout rate") +
  geom_text(aes(label=format(turnout_rate, digits = 2)), hjust = -0.3) +
  coord_flip() +
  ggtitle("Turnout rate vs. Age*Party") +
    theme(plot.title = element_text(hjust = 0.5))
```




```
data %>%
  group_by(age, party_cd) %>%
  summarize(n = n())
```

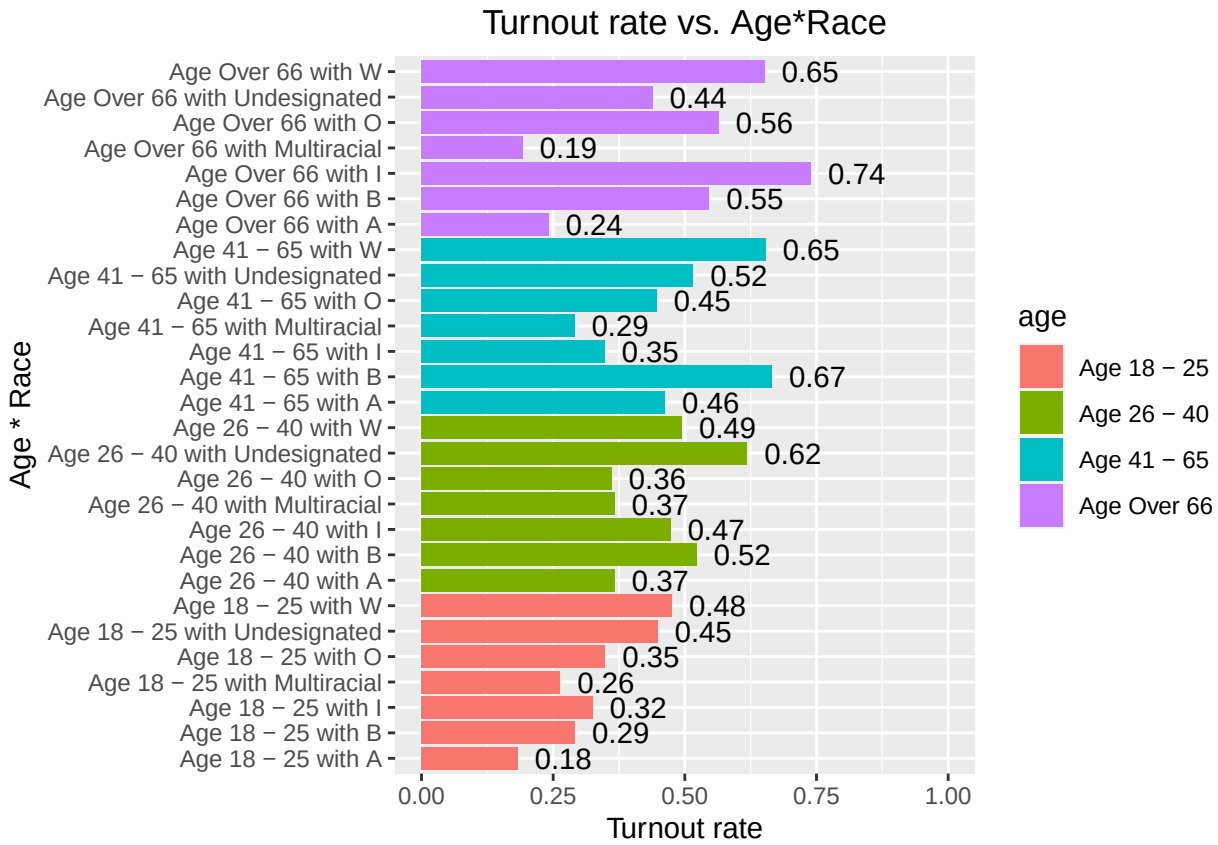
```
## `summarise()` has grouped output by 'age'. You can override using the `.groups` argument.
```

```
## # A tibble: 20 x 3
## # Groups:   age [4]
##    age         party_cd     n
##    <chr>       <chr>    <int>
##  1 Age 18 - 25 CST_GRE     18
##  2 Age 18 - 25 DEM        329
##  3 Age 18 - 25 LIB         60
##  4 Age 18 - 25 REP        237
##  5 Age 18 - 25 UNA        388
##  6 Age 26 - 40 CST_GRE     28
##  7 Age 26 - 40 DEM        383
##  8 Age 26 - 40 LIB         70
##  9 Age 26 - 40 REP        277
## 10 Age 26 - 40 UNA        413
## 11 Age 41 - 65 CST_GRE     25
## 12 Age 41 - 65 DEM        402
## 13 Age 41 - 65 LIB         53
## 14 Age 41 - 65 REP        348
## 15 Age 41 - 65 UNA        420
## 16 Age Over 66 CST_GRE      4
```

```
## 17 Age Over 66 DEM        273
## 18 Age Over 66 LIB         11
## 19 Age Over 66 REP        211
## 20 Age Over 66 UNA        244
```

age & race

```
data %>%
  group_by(age, race_code) %>%
  summarise(total_reg = sum(total_reg),
            total_vot = sum(total_vot), .groups = "drop") %>%
  mutate(turnout_rate = total_vot / total_reg,
         group = paste0(age, " with ", race_code)) %>%
  select(group = group, turnout_rate, age, race_code) %>%
  ggplot( aes(x = group, y = turnout_rate, fill = age)) +
  geom_bar(stat = "identity") +
  ylim(0,1) +
  xlab("Age * Race") + ylab("Turnout rate") +
  geom_text(aes(label=format(turnout_rate, digits = 2)), hjust = -0.3) +
  coord_flip() +
  ggtitle("Turnout rate vs. Age*Race") +
    theme(plot.title = element_text(hjust = 0.5))
```

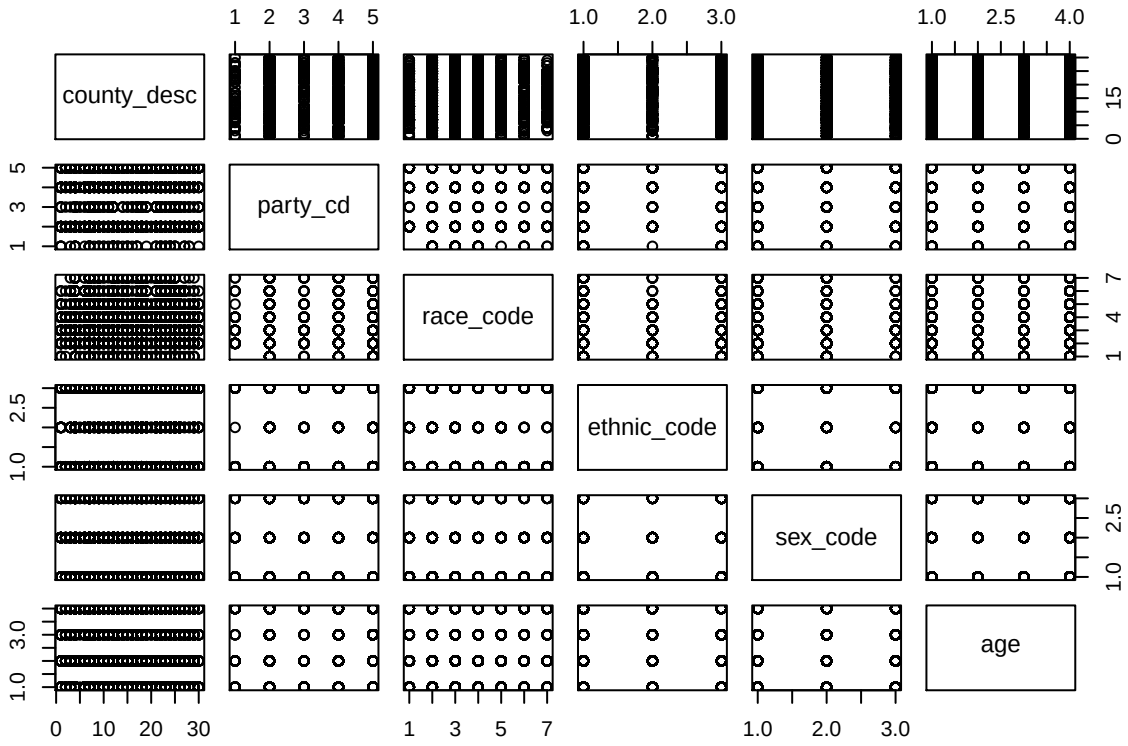


```
data %>%
  group_by(age, race_code) %>%
  summarize(n = n())
```

```
## `summarise()` has grouped output by 'age'. You can override using the `.groups` argument.
```

```
## # A tibble: 28 x 3
## # Groups:   age [4]
##    age         race_code        n
##    <chr>       <fct>        <int>
##  1 Age 18 - 25 Multiracial     84
##  2 Age 18 - 25 Undesignated   214
##  3 Age 18 - 25 W              308
##  4 Age 18 - 25 B              182
##  5 Age 18 - 25 O              149
##  6 Age 18 - 25 I               46
##  7 Age 18 - 25 A               49
##  8 Age 26 - 40 Multiracial     95
##  9 Age 26 - 40 Undesignated   235
## 10 Age 26 - 40 W              351
## # ... with 18 more rows
```

```
data <- data %>%
  mutate(total_vot = as.numeric(total_vot),
         total_reg = as.numeric(total_reg),
         county_desc = as.factor(county_desc),
         party_cd = as.factor(party_cd),
         race_code = as.factor(race_code),
         sex_code = as.factor(sex_code),
         ethnic_code = as.factor(ethnic_code),
         age = as.factor(age))


pairs(data[,1:6])
```



Model

```
# initial
mod1 <- glmer(cbind(total_vot, total_reg - total_vot) ~
                      party_cd + race_code + ethnic_code + sex_code + age + (1|county_desc),
                    data = data, family = binomial,
                control=glmerControl(optimizer = "bobyqa"))
BIC(mod1)
```

```
## [1] 40615.93
```

```
mod2 <- glmer(cbind(total_vot, total_reg - total_vot) ~
                      -1 + race_code + ethnic_code + sex_code + age + (1 | county_desc),
                    data = data, family = binomial,
                control=glmerControl(optimizer = "bobyqa", optCtrl=list(maxfun=2e5)))

BIC(mod2)
```

```
## [1] 40712.29
```

```
mod3 <- glmer(cbind(total_vot, total_reg - total_vot) ~
                   party_cd + race_code + ethnic_code + sex_code + age +
                   sex_code:party_cd + age:party_cd + (1|county_desc),
                 data = data, family = binomial,
                control=glmerControl(optimizer = "bobyqa", optCtrl=list(maxfun=2e5)))

BIC(mod3)
```

```
## [1] 40298.74
```

```
mod4 <- glmer(cbind(cbind(total_vot, total_reg - total_vot)) ~
                  party_cd + race_code + ethnic_code + sex_code + age +
                  sex_code:party_cd + age:party_cd + age:race_code + (1|county_desc),
                data = data, family = binomial,
      control=glmerControl(optimizer = "bobyqa",optCtrl=list(maxfun=2e5)))
BIC(mod4)
```

```
## [1] 39476.9
```

```
model <- c("Base model",
           "Without intercept",
           "Add the interaction of sex and party_cd, and age and party",
           "Add the interaction of sex and party_cd, age and party, and age and race")
LRT <- c("",
         round(anova(mod1,mod2)$`Pr(>Chisq)`[2],4),
         round(anova(mod1,mod3)$`Pr(>Chisq)`[2],4),
         round(anova(mod3,mod4)$`Pr(>Chisq)`[2],4))
BIC_score1 <- sapply(c(mod1, mod2, mod3, mod4), BIC)
data.frame("Model" = model, 'LRT p-value' = LRT, 'BIC' = BIC_score1) %>%
  kable(caption = "Forward model selection") %>%
    kable_styling(latex_options = c("HOLD_position","striped"))
```

Table 1: Forward model selection

Model	LRT.p.value	BIC
Base model		40615.93
Without intercept	0	40712.29
Add the interaction of sex and party_cd, and age and party	0	40298.74
Add the interaction of sex and party_cd, age and party, and age and race	0	39476.90

In order to answer the questions of interest, we require the following features in our model:

1. Fixed effects for various demographic subgroups (race, ethnicity, age, sex)
2. Random effects for county
3. Sex-Party interaction
4. Age-Party interaction

```
summary(mod4)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula:
## cbind(cbind(total_vot, total_reg - total_vot)) ~ party_cd + race_code +
##     ethnic_code + sex_code + age + sex_code:party_cd + age:party_cd +
##     age:race_code + (1 | county_desc)
##    Data: data
## Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+05))
```

```
##
##      AIC      BIC   logLik deviance df.resid
##  39115.4  39476.9 -19500.7  39001.4     4137
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -19.8597  -0.9768  -0.2823   1.4499  19.6901
##
## Random effects:
##  Groups      Name        Variance Std.Dev.
##  county_desc (Intercept) 0.7272   0.8528
## Number of obs: 4194, groups:  county_desc, 30
##
## Fixed effects:
##                                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)                           -4.10771    1.03678  -3.962 7.43e-05 ***
## party_cdDEM                            2.53214    1.01360   2.498 0.012484 *
## party_cdLIB                            1.84816    1.05097   1.759 0.078657 .
## party_cdREP                            2.62985    1.01358   2.595 0.009470 **
## party_cdUNA                            2.42949    1.01313   2.398 0.016485 *
## race_codeUndesignated                  0.75694    0.18951   3.994 6.49e-05 ***
## race_codeW                             0.92827    0.18421   5.039 4.67e-07 ***
## race_codeB                            -0.06776    0.18587  -0.365 0.715430
## race_codeO                             0.55664    0.20512   2.714 0.006655 **
## race_codeI                             0.19965    0.22622   0.883 0.377485
## race_codeA                            -0.77203    0.28168  -2.741 0.006129 **
## ethnic_codeHL                         -0.71634    0.05638 -12.706  < 2e-16 ***
## ethnic_codeUN                         -0.34210    0.02179 -15.698  < 2e-16 ***
## sex_codeF                             -1.57591    1.09263  -1.442 0.149213
## sex_codeU                              0.89680    0.73415   1.222 0.221876
## ageAge 41 - 65                         1.01423    1.10640   0.917 0.359306
## ageAge Over 66                         1.23936    1.60972   0.770 0.441348
## ageAge 26 - 40                         0.88208    1.06869   0.825 0.409156
## party_cdDEM:sex_codeF                  1.80057    1.09299   1.647 0.099480 .
## party_cdLIB:sex_codeF                  1.65679    1.12124   1.478 0.139505
## party_cdREP:sex_codeF                  1.76578    1.09303   1.615 0.106204
## party_cdUNA:sex_codeF                  1.70029    1.09305   1.556 0.119815
## party_cdDEM:sex_codeU                 -0.36358    0.73556  -0.494 0.621103
## party_cdLIB:sex_codeU                 -0.22507    0.80290  -0.280 0.779236
## party_cdREP:sex_codeU                 -0.65764    0.73488  -0.895 0.370841
## party_cdUNA:sex_codeU                 -0.88629    0.73420  -1.207 0.227375
## party_cdDEM:ageAge 41 - 65            -0.98003    1.07549  -0.911 0.362165
## party_cdLIB:ageAge 41 - 65            -1.75963    1.12155  -1.569 0.116665
## party_cdREP:ageAge 41 - 65            -1.27722    1.07549  -1.188 0.235001
## party_cdUNA:ageAge 41 - 65            -1.15928    1.07495  -1.078 0.280830
## party_cdDEM:ageAge Over 66            -2.02686    1.52278  -1.331 0.183179
## party_cdLIB:ageAge Over 66            -4.22539    1.88115  -2.246 0.024693 *
## party_cdREP:ageAge Over 66            -1.90879    1.52305  -1.253 0.210108
## party_cdUNA:ageAge Over 66            -1.31378    1.52280  -0.863 0.388281
## party_cdDEM:ageAge 26 - 40            -0.76260    1.04724  -0.728 0.466493
## party_cdLIB:ageAge 26 - 40            -0.51396    1.08314  -0.475 0.635136
## party_cdREP:ageAge 26 - 40            -0.44149    1.04735  -0.422 0.673365
## party_cdUNA:ageAge 26 - 40            -0.35145    1.04660  -0.336 0.737021
## race_codeUndesignated:ageAge 41 - 65   0.50540    0.28385   1.780 0.074994 .
```

```
## race_codeW:ageAge 41 - 65             0.93852    0.27987   3.353 0.000798 ***
## race_codeB:ageAge 41 - 65             1.84728    0.28132   6.567 5.15e-11 ***
## race_codeO:ageAge 41 - 65             0.39488    0.30552   1.292 0.196186
## race_codeI:ageAge 41 - 65             0.06011    0.31629   0.190 0.849274
## race_codeA:ageAge 41 - 65             1.52298    0.36870   4.131 3.62e-05 ***
## race_codeUndesignated:ageAge Over 66  0.67358    0.54479   1.236 0.216312
## race_codeW:ageAge Over 66             1.32806    0.53707   2.473 0.013407 *
## race_codeB:ageAge Over 66             2.01324    0.53894   3.736 0.000187 ***
## race_codeO:ageAge Over 66             1.19009    0.57100   2.084 0.037141 *
## race_codeI:ageAge Over 66             2.41249    0.59132   4.080 4.51e-05 ***
## race_codeA:ageAge Over 66             0.86591    0.65675   1.318 0.187340
## race_codeUndesignated:ageAge 26 - 40  0.58624    0.24576   2.385 0.017058 *
## race_codeW:ageAge 26 - 40            -0.39687    0.24006  -1.653 0.098289 .
## race_codeB:ageAge 26 - 40             0.78775    0.24271   3.246 0.001172 **
## race_codeO:ageAge 26 - 40            -0.33055    0.27118  -1.219 0.222859
## race_codeI:ageAge 26 - 40             0.33277    0.30020   1.109 0.267638
## race_codeA:ageAge 26 - 40             0.58063    0.35115   1.653 0.098230 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Correlation matrix not shown by default, as p = 56 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

```r
chart_fixef <- function(model) {
  summary(model)$coefficients %>%
    as.data.frame() %>%
    mutate(`95% CI` = paste0(
      '[',
      round(Estimate - 1.96 * `Std. Error`, 3), ', ',
      round(Estimate + 1.96 * `Std. Error`, 3), ']' )
    ) %>%
    select(Estimate, `95% CI`) %>%
    kable(digits=3) %>%
    kable_styling(full_width=FALSE) %>%
    kable_classic()
}
```

```r
# expand data from aggregated format
data_expand <- data %>%
  mutate(resp = map2(total_vot, total_reg, ~ c(
    rep(1, .x), rep(0, .y - .x)
  ))) %>%
  unnest(cols = c(resp)) %>%
  select(-c(total_reg, total_vot))

# split into train and test
sample <- sample(c(TRUE, FALSE), nrow(data_expand), replace=TRUE, prob=c(0.7,0.3))
train <- data_expand[sample,]
test <- data_expand[!sample,]

# fit model on train data
```

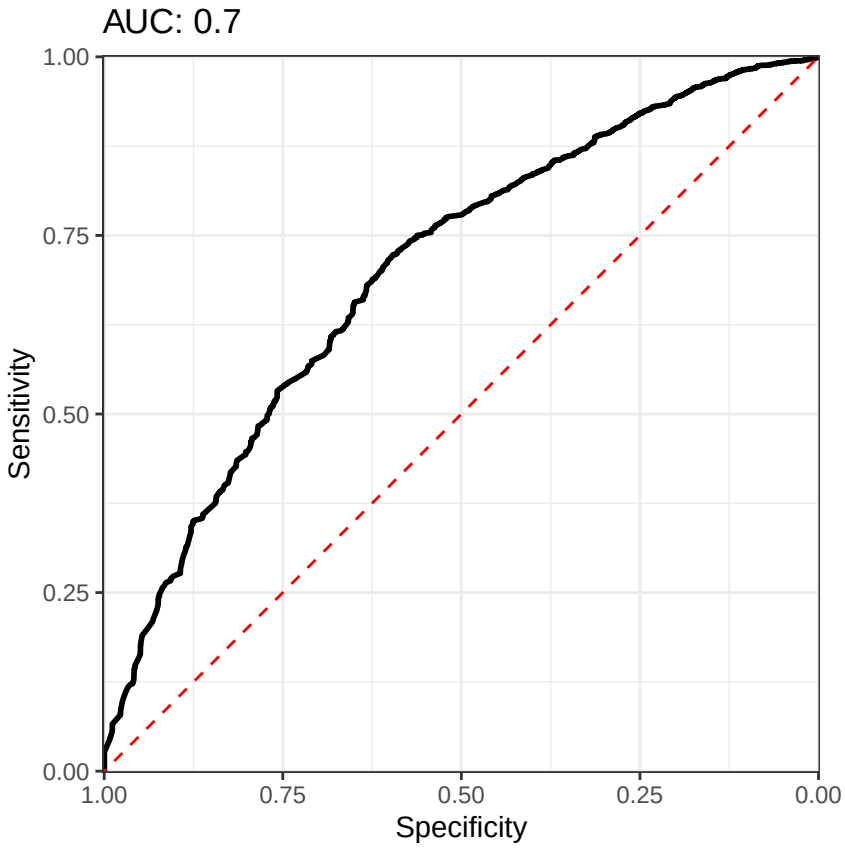
```
### fit any model you like on unaggregated train data
### will take a while...

# mod2 <- glmer(resp ~ 1 + party_cd + (1 | county_desc),
#               data = train, family = binomial)

# ROC + AUC
predicted <- predict(mod2, test, type="response")
rocobj <- roc(test$resp, predicted)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
ggroc(rocobj, size=1) +
  coord_fixed() +
  scale_x_reverse(
    name = "Specificity",
    limits = c(1,0),
    expand = c(0.001,0.001)
  ) +
  scale_y_continuous(
    name = "Sensitivity",
    limits = c(0,1),
    expand = c(0.001, 0.001)
  ) +
  geom_abline(intercept=1, slope=1, linetype="dashed", color="red") +
  labs(title=paste("AUC:", round(rocobj$auc, 3))) +
  theme_bw()
```

```
## Scale for 'x' is already present. Adding another scale for 'x', which will
## replace the existing scale.
```

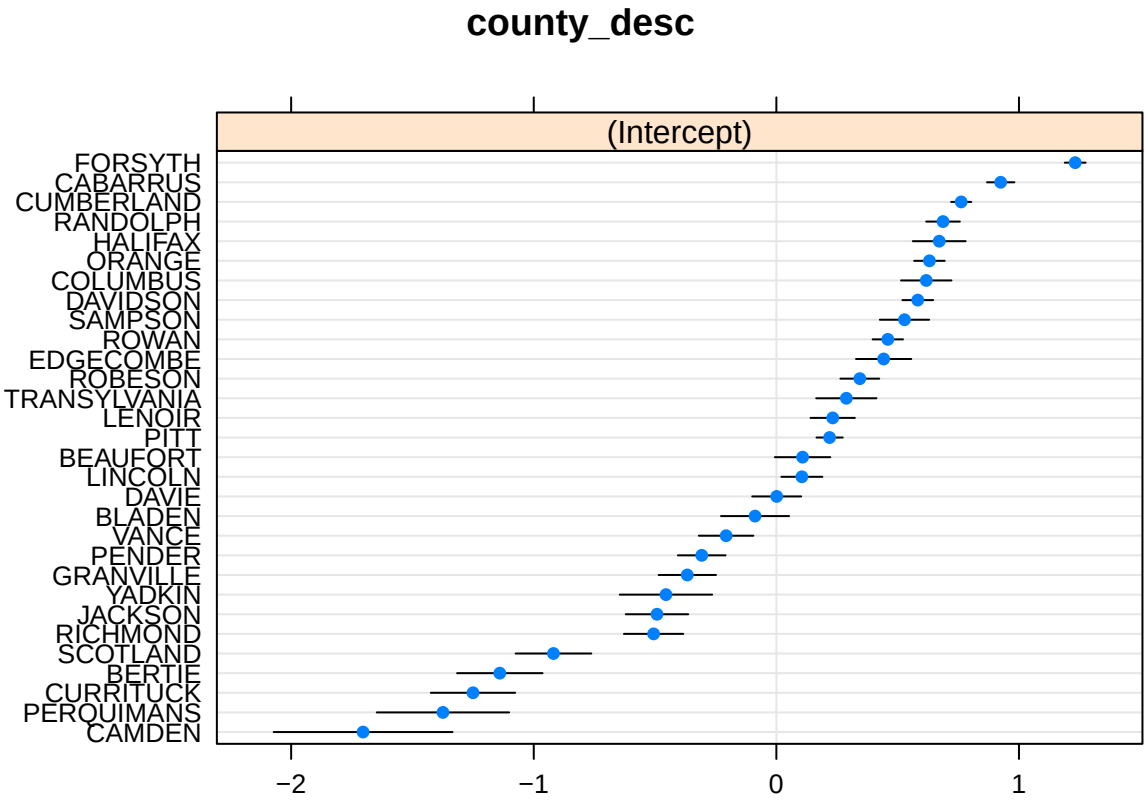



```
# fit model on full data

mod4 <- glmer(cbind(total_vot, total_reg - total_vot) ~
                1 + party_cd + sex_code:party_cd + (1|county_desc),
              data = data, family = binomial,
              control=glmerControl(optimizer = "bobyqa"))

# Fixed Effects
chart_fixef(mod4)
```

	Estimate	95% CI
(Intercept)	-1.988	[-2.859, -1.117]
party_cdDEM	1.633	[0.801, 2.466]
party_cdLIB	0.667	[-0.224, 1.559]
party_cdREP	1.821	[0.988, 2.653]
party_cdUNA	1.609	[0.777, 2.442]
party_cdCST_GRE:sex_codeF	-1.812	[-3.96, 0.336]
party_cdDEM:sex_codeF	0.238	[0.186, 0.291]
party_cdLIB:sex_codeF	0.043	[-0.433, 0.52]
party_cdREP:sex_codeF	0.243	[0.184, 0.302]
party_cdUNA:sex_codeF	0.179	[0.12, 0.237]
party_cdCST_GRE:sex_codeU	0.507	[-0.776, 1.79]
party_cdDEM:sex_codeU	0.178	[0.06, 0.297]
party_cdLIB:sex_codeU	0.616	[-0.001, 1.233]
party_cdREP:sex_codeU	-0.119	[-0.225, -0.014]
party_cdUNA:sex_codeU	-0.292	[-0.378, -0.206]

```
# Random Effects
dotplot(ranef(mod4))
```

```
## $county_desc
```




```
test <- binned_residuals(mod4)
```

```
## Warning: Probably bad model fit. Only about 9% of the residuals are inside the error bounds.
```

```
# mod <- brm(
#   data = data, family = binomial,
#   total_vot | trials(total_reg) ~ 1 + party_cd + (1 | county_desc),
#   prior = c(prior(normal(0, 10), class = Intercept),
#             prior(normal(0, 1), class = b)),
#   iter = 2500, warmup = 500, cores = 4, chains = 2,
#   seed = 10
# )
```

```
# fixef(mod) %>%
#   as.data.frame() %>%
#   mutate(`95% CI` = paste0(
#     "[", round(Q2.5, 3), ",", round(Q97.5, 3), "]"
#   )) %>%
#   select(-c(Est.Error, Q2.5, Q97.5)) %>%
#   kable(digits=3) %>%
#   kable_classic() %>%
#   kable_styling(full_width=FALSE)
#
# ranef(mod)$county_desc %>%
#   as.data.frame() %>%
#   tibble::rownames_to_column(var="County") %>%
#   ggplot(aes(y=reorder(County, Estimate.Intercept), x=Estimate.Intercept)) +
#     geom_point(color="blue", size=2) +
#     geom_linerange(aes(
#       xmin=Q2.5.Intercept, xmax=Q97.5.Intercept
#     )) +
#     labs(
#       x="Intercept",
#       y="County"
#     ) +
#     theme_bw()
```