Imputing turnout at ANC-level using precinct data

We'd like to get data on voter turnout to contextualize data on e.g. down-ballot roll-off for ANC elections. Roll-off can be calculated (post-2012) from ANC-level data on ballots cast and undervotes, but registrations are only available in the election data at the precinct level, so we need to think about how to aggregate from precinct to ANC.

Look at registration/ballot data

Registration / ballot data:

- Data at precinct level
- Broken out by ANC based on precinct/ANC overlaps in election data

```
## # A tibble: 816 x 6
##   precinct  ward anc    year registered_voters ballots
##      <int> <int> <fct> <int>             <int>   <int>
## 1        1     6 C      2012              5687    3063
## 2        1     6 C      2014              5562    1686
## 3        1     6 C      2016              6694    4137
## 4        1     6 C      2018              6736    2813
## 5        1     6 E      2012              5687    3063
## # ... with 811 more rows
```

How many precincts cross ANC's?

- About half of precincts fall in more than one ANC
- so we'll have to do some figuring to aggregate data up from precinct to ANC

```
##   ancs precincts
## 1    1        86
## 2    2        53
## 3    3         4
```

How many ANCs share precincts with neighbors?

- notate precincts which cross ANCs as 'duplicitous'

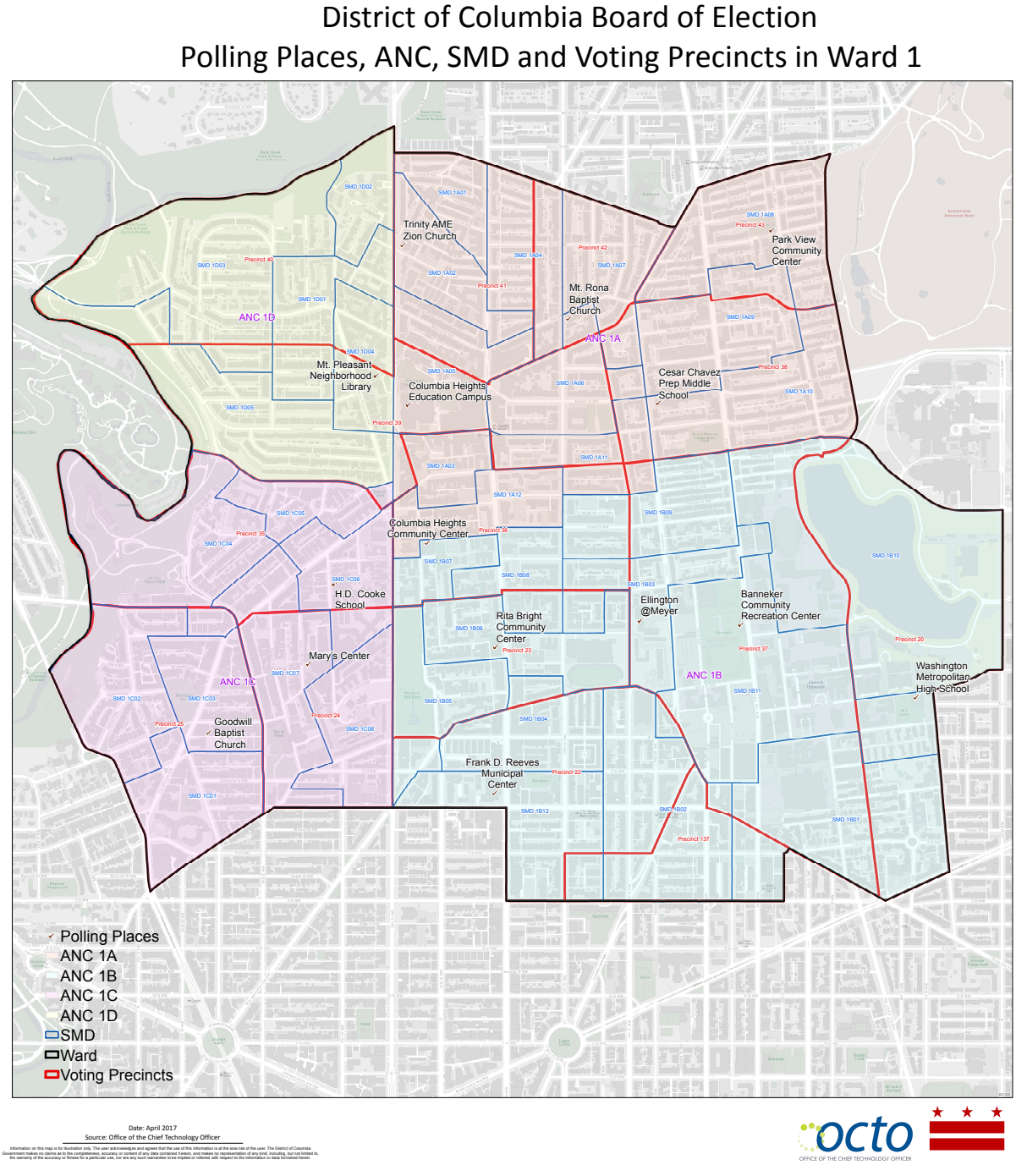
```
## # A tibble: 816 x 6
##   anc.full precinct  year duplicitous voters ballots
##   <chr>       <int> <int> <lgl>        <int>   <int>
## 1 6C              1  2012 TRUE          5687    3063
## 2 6C              1  2014 TRUE          5562    1686
## 3 6C              1  2016 TRUE          6694    4137
## 4 6C              1  2018 TRUE          6736    2813
## 5 6E              1  2012 TRUE          5687    3063
## # ... with 811 more rows
```

- count duplicitous precincts by ANC

```
## # A tibble: 38 x 3
##   anc.full count.dup count.tot
##   <chr>        <int>     <int>
## 1 1A               3         7
## 2 1B               2         6
## 3 1D               1         2
## 4 2A               2         4
## 5 2B               4         6
## # ... with 33 more rows
```

Quick validity check

Using a map of ward 1 ANCs/precincts



from ward 1 map, we know:

- pct 39 -> ANC 1A, 1D, 1C (trivially)
- 36 -> 1A 1B

- 37 -> 1B + 1 block of 1A

```
##   precinct  ancs
## 1       36 1A 1B
## 2       37 1A 1B
## 3       39 1A 1D
```

so at this point we could just toss precincts crossing ANCs (and lose around 40% of the data)
or we could do a naive average
or we could give them weighted averages based on GIS data....

GIS Data

Read in shapefiles

```
precinct.shapes <- st_read(paste(prefix, "raw_data/precinct_shapes_2012/Voting_Precinct__2012.shp", sep=
anc.shapes <- st_read(paste(prefix, "raw_data/anc_2013/Advisory_Neighborhood_Commissions_from_2013.shp"
```

Compute intersections between ANC & precinct shapes

```
overlap <- st_intersection(anc.shapes, precinct.shapes) %>%
        mutate(over.area = st_area(.) %>% as.numeric()) %>%
    rename(.anc = NAME, .precinct = NAME.1) %>%
    select(.anc, .precinct, over.area)
```

```
## although coordinates are longitude/latitude, st_intersection assumes that they are planar
```

```
## Warning: attribute variables are assumed to be spatially constant
## throughout all geometries
```

```
## Simple feature collection with 446 features and 3 fields
## geometry type:  GEOMETRY
## dimension:      XY
## bbox:           xmin: -77.11979 ymin: 38.79164 xmax: -76.90915 ymax: 38.99597
## epsg (SRID):    4326
## proj4string:    +proj=longlat +datum=WGS84 +no_defs
## First 5 features:
##   anc.full precinct    over.area                       geometry
## 1       7D      139 8.337090e+02 POLYGON ((-76.94353 38.9170...
## 2       5C      139 1.465533e+06 POLYGON ((-76.94252 38.9190...
## 3       2C      143 5.754348e+05 POLYGON ((-77.01408 38.8994...
## 4       6C      143 2.000379e+00 MULTIPOLYGON (((-77.01407 3...
## 5       1A       37 1.090658e+04 GEOMETRYCOLLECTION (LINESTR...
```

How many entries do we get in the overlap dataset?

```
## [1] 446
```

How many did we start with in the election data grouped by anc x pct?

```
## [1] 204
```

These aren't wildly off, so it's clearly only including shapes with intersections – it just might be counting some trivial ones

How many of the intersections in 'overlap' are nontrivial?

```
## [1] 298
```

what are the units? it's like 7-digit numbers... sq meters, feet, lat/lon minutes???

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0       1    4925  397921  548532 7807612
```

Compute relative areas of intersections w/r/t precincts

Get precinct total areas

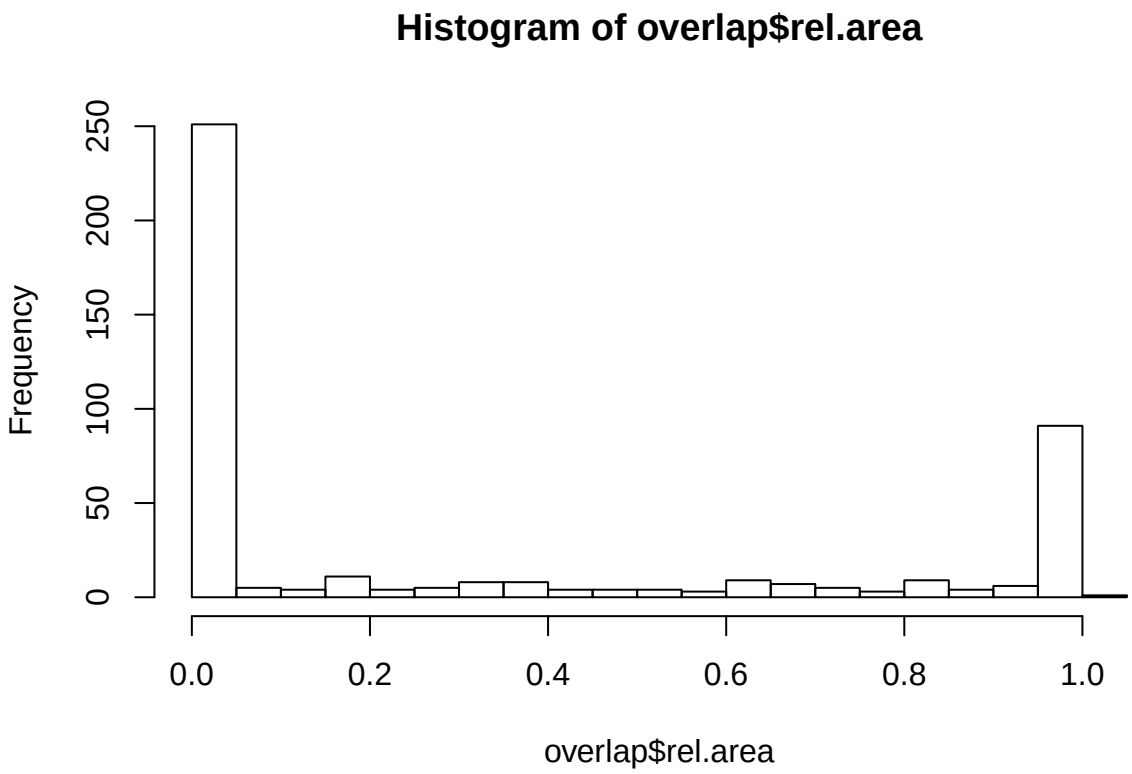
```
precinct.areas <- precinct.shapes %>%
            mutate(area = st_area(.) %>% as.numeric(),
                precinct = regmatches(NAME, regexpr("[[:digit:]]+", NAME)))
precinct.areas <- tibble(precinct=precinct.areas$precinct,
                            prec.area=precinct.areas$area)
```

Merge with overlap areas

```
overlap %<>% inner_join(precinct.areas, by=c("precinct"))
```

Compute relative area of ANC x pct as [area of overlap] / [precinct area]

```
overlap %<>% mutate(rel.area = over.area / prec.area)
```



Check against ward 1 map again

Expect:

- 36 -> 1A 1B
- 39 -> 1A 1D (1C)
- 37 -> 1A 1B

```
##   precinct  ancs  min.area
## 1       36 1A 1B 193701.88
## 2       37 1A 1B  10906.58
## 3       39 1D 1A 257982.56
```

Matches election data at 1% relative area cutoff (to toss noise)

Cross-reference with registration data

```
overlap %<>% mutate(precinct = as.integer(precinct))
crossref <- full_join(overlap, collapsed, by=c("anc.full", "precinct"))

crossref %<>% data.frame %>% select(-geometry)
crossref %>% as_tibble %>% print(n=5)
```

```
## # A tibble: 1,058 x 9
##   anc.full precinct over.area prec.area rel.area  year duplicitous voters
```

```
##   <chr>       <int>     <dbl>     <dbl>    <dbl> <int> <lgl>        <int>
## 1 7D            139      834.  1466371. 0.000569    NA NA             NA
## 2 5C            139  1465533.  1466371. 0.999     2012 FALSE         2590
## 3 5C            139  1465533.  1466371. 0.999     2014 FALSE         2411
## 4 5C            139  1465533.  1466371. 0.999     2016 FALSE         2669
## 5 5C            139  1465533.  1466371. 0.999     2018 FALSE         2989
## # ... with 1,053 more rows, and 1 more variable: ballots <int>
```

Any precinct-ANC combos from voting data missing from GIS data?

```
## [1] 0
```

```
## [1] anc.full    precinct    over.area   prec.area   rel.area    year
## [7] duplicitous voters      ballots
## <0 rows> (or 0-length row.names)
```

Are any precinct-ANC combos from GIS data missing from voting data?

- filtering at 5% relative area

```
## # A tibble: 3 x 9
##   anc.full precinct over.area prec.area rel.area  year duplicitous voters
##   <chr>       <int>     <dbl>     <dbl>    <dbl> <int> <lgl>        <int>
## 1 2A              6   633768.  3263321.    0.194    NA NA             NA
## 2 7B            107   325237.  1291722.    0.252    NA NA             NA
## 3 6D            129  4160730. 10102477.    0.412    NA NA             NA
## # ... with 1 more variable: ballots <int>
```

- We're matching up well!

Compute turnout

Weighting ambiguous precincts by goegraphic overlap

Drop GIS data which didn't match any election data

```
crossref %<>% filter(!is.na(duplicitous))
```

Fix hanging vote data

- First drop duplicitous precincts w/ missing GIS
- Then set rel.area to 1 if missing (having retained only non-duplicitous precincts)

```
crossref %<>% filter(!(is.na(rel.area) & duplicitous))
crossref %<>% mutate(rel.area = ifelse(is.na(rel.area), 1, rel.area))
```

- Recompute 'duplicitous' after dropping

```
crossref %<>% group_by(precinct, year) %>%
                mutate(duplicitous = length(unique(anc.full))>1)
```

- Make sure newly nonduplicitous obs have area 1

  - by renormalizing relative area w/r/t precinct

```
crossref %<>% group_by(precinct, year) %>%
                mutate(norm.area = over.area / sum(over.area))
```

- How much did this change relative areas?

```
##       Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
## -1.070e-06  1.400e-07  7.470e-06  5.545e-03  6.499e-04  2.518e-01
```

Aggregate with weighting

```
crossref %<>% mutate(voters = voters * norm.area, ballots = ballots * norm.area)

reg.fixed <- crossref %>% group_by(anc.full, year) %>%
               summarize(voters = round(sum(voters)),
               ballots = round(sum(ballots)),
           duplicitous = sum(duplicitous))

reg.fixed %<>% mutate(turnout = ballots / voters)
```

Recompute turnout by dropping ANC-crossing precincts

```
reg.fixed.drop <- collapsed %>% filter(!duplicitous) %>%
                    group_by(anc.full, year) %>%
             summarize(voters = round(sum(voters)),
                    ballots = round(sum(ballots)))

reg.fixed.drop %<>% mutate(turnout = ballots / voters)

#write.table(reg.fixed.drop, file=paste(prefix, "cleaned_data/2012_2018_imputedTurnoutDrop_anc.csv", se
```

Testing Imputed Turnout

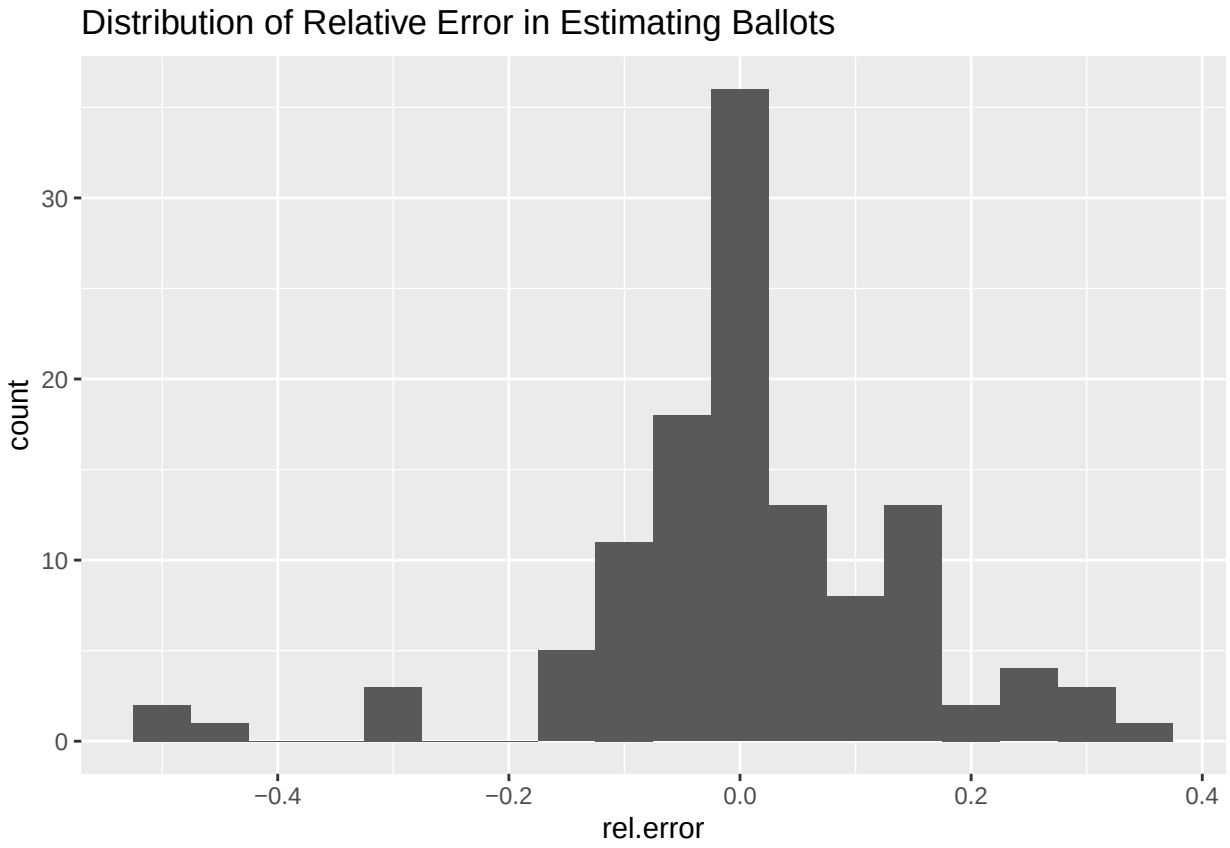
Now that we have turnout estimates, draw in the election data for some testing

Merge together dropped and geo-weighted turnout estimates with election data

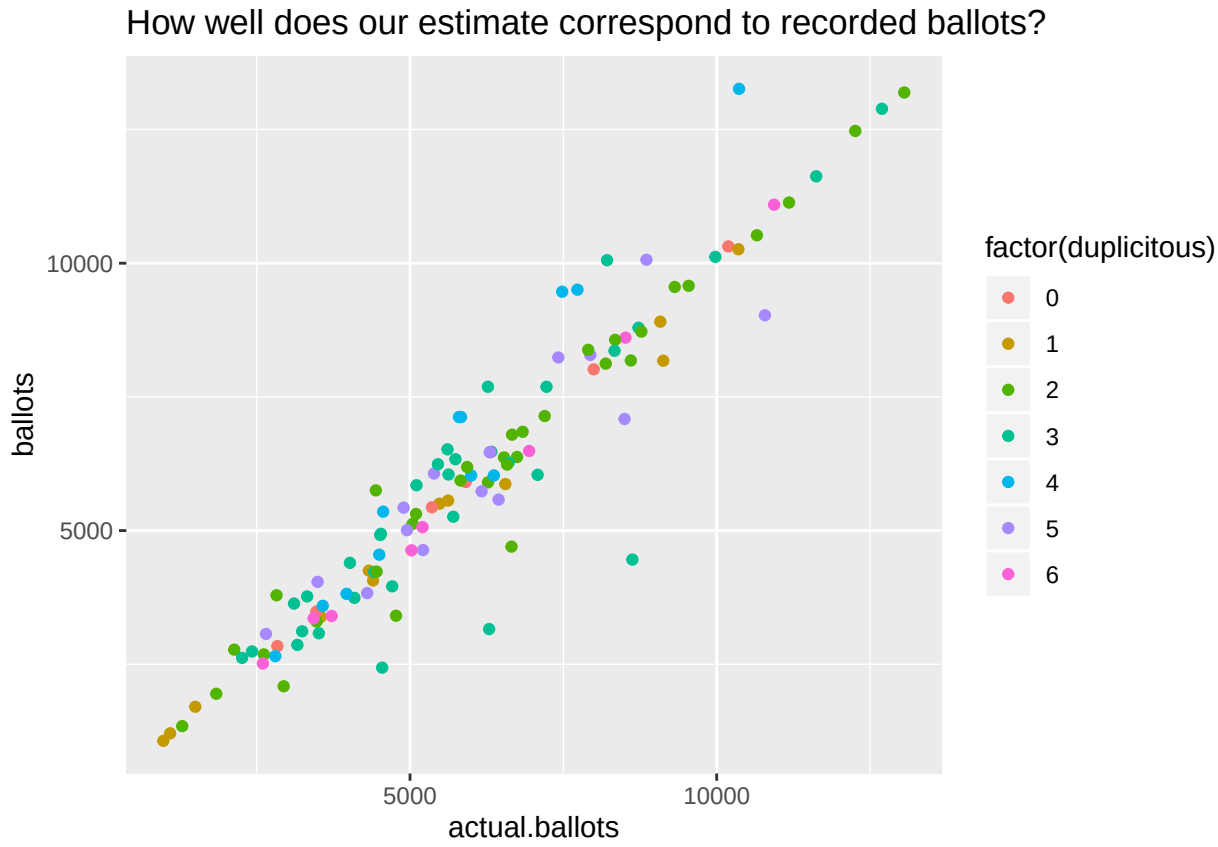
- losing some ANCs which had no fully contained precincts

```
## # A tibble: 12 x 7
## # Groups:   anc.full [3]
##   anc.full  year voters ballots duplicitous turnout turnout.drop
##   <chr>    <int>  <dbl>   <dbl>       <int>   <dbl>        <dbl>
## 1 2F        2012   6525    3923           3   0.601           NA
## 2 2F        2014   6182    2438           3   0.394           NA
## 3 2F        2016   6274    4458           3   0.711           NA
## 4 2F        2018   6405    3159           3   0.493           NA
## 5 3B        2012   7796    4883           3   0.626           NA
## # ... with 7 more rows
```

Compare estimated ballots with post-2012 actual ballots



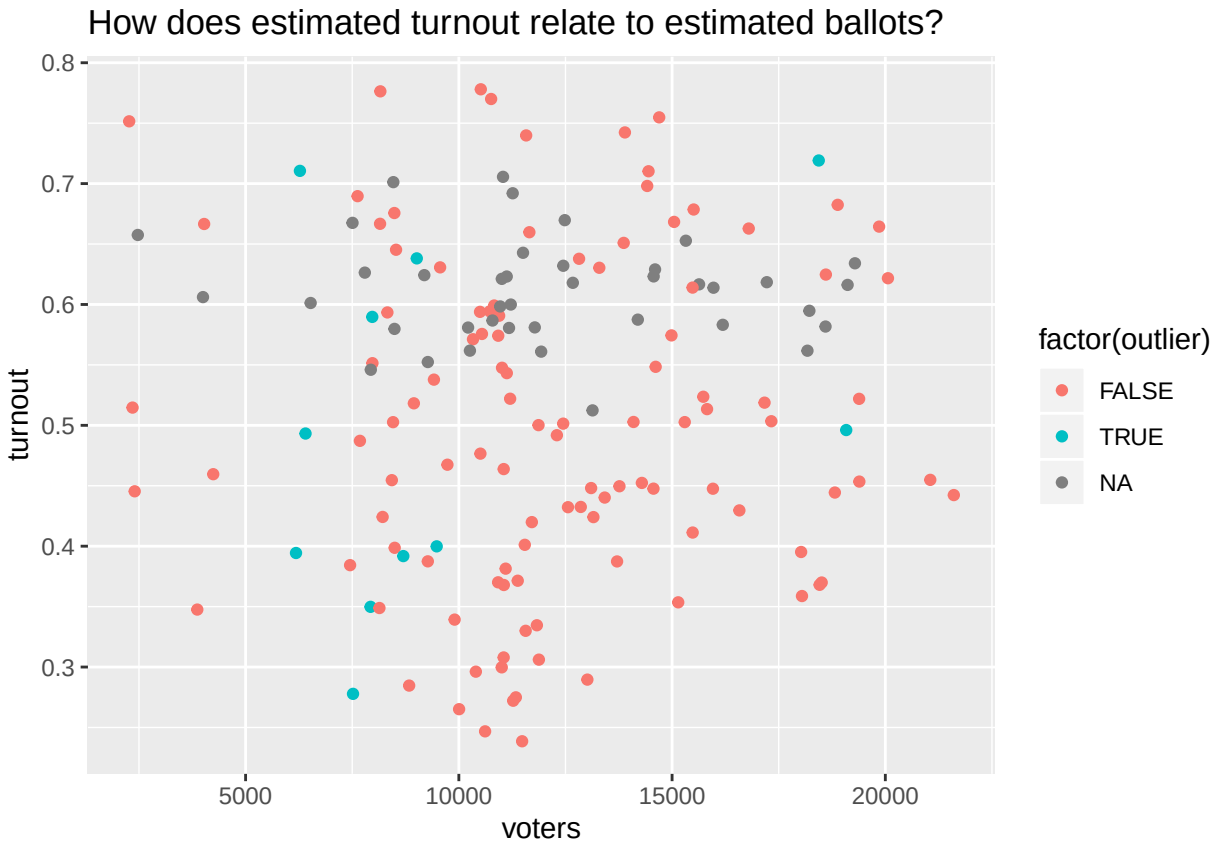
Make some plots



Which ANCs have particularly off estimates?

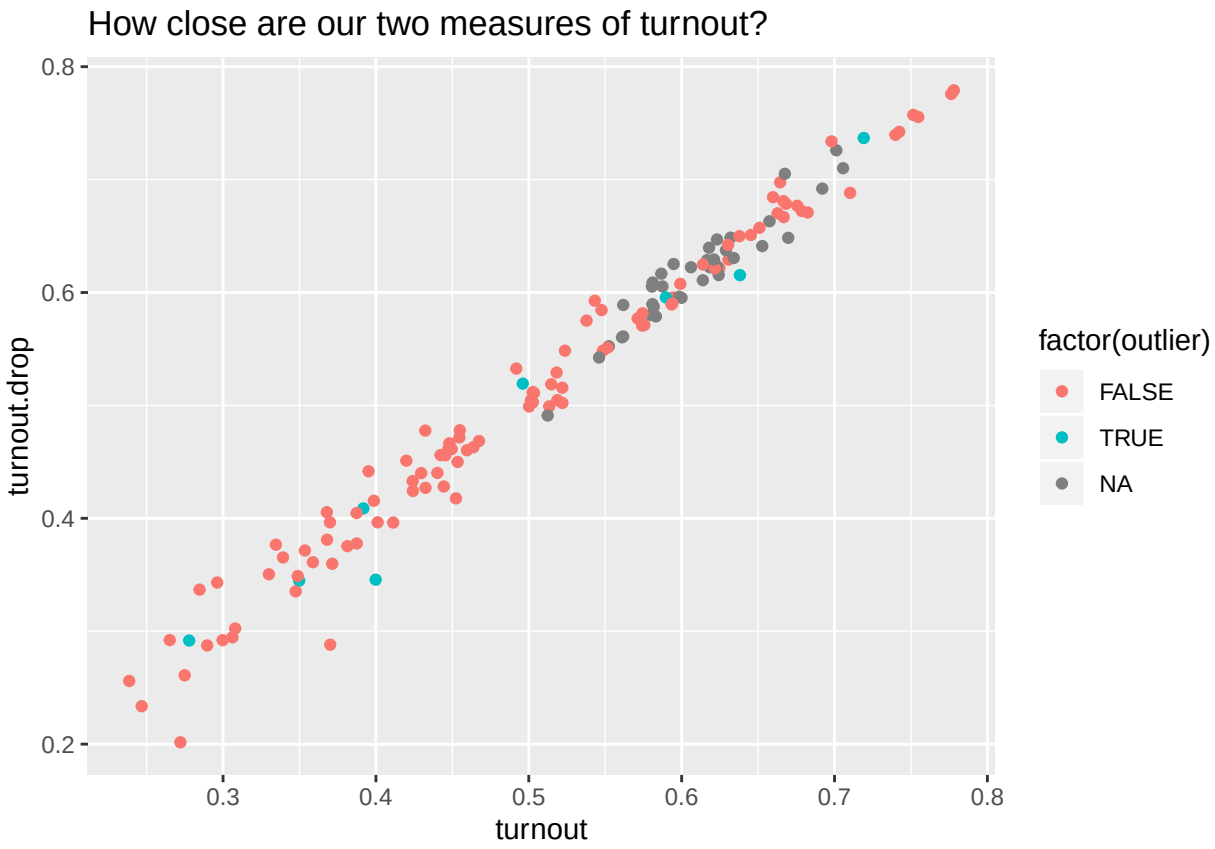
```
## # A tibble: 11 x 6
## # Groups:   anc.full [4]
##    anc.full  year duplicitous ballots actual.ballots rel.error
##    <chr>    <int>       <int>   <dbl>          <int>     <dbl>
##  1 2A        2014           2    2774           2135     0.299
##  2 2A        2016           2    5752           4443     0.295
##  3 2A        2018           2    3791           2825     0.342
##  4 2B        2016           4   13260          10364     0.279
##  5 2B        2018           4    9467           7481     0.265
##  6 2F        2014           3    2438           4547    -0.464
##  7 2F        2016           3    4458           8625    -0.483
##  8 2F        2018           3    3159           6290    -0.498
##  9 5D        2014           2    2089           2941    -0.290
## 10 5D        2016           2    4700           6655    -0.294
## 11 5D        2018           2    3408           4772    -0.286
```

How does est. precinct size (as registered voters) relate to est. turnout?



How closely do our two measure of turnout track?

```
## Warning: Removed 12 rows containing missing values (geom_point).
```



Which ANCs have the greatest discrepancy between turnout and turnout.drop?

```
##
## Which ANCs have the greatest discrepancy between turnout and turnout.drop?
## # A tibble: 160 x 12
## # Groups:   anc.full [40]
##    anc.full  year voters ballots duplicitous turnout turnout.drop
##    <chr>    <int>  <dbl>   <dbl>       <int>   <dbl>        <dbl>
##  1 8C        2014   8835    2515           6   0.285        0.337
##  2 8A        2016  11125    6043           3   0.543        0.593
##  3 8A        2014  10399    3080           3   0.296        0.343
##  4 2B        2014  18026    7124           4   0.395        0.442
##  5 6C        2014  12559    5429           5   0.432        0.478
##  6 8A        2018  11830    3958           3   0.335        0.377
##  7 5A        2018  12299    6049           3   0.492        0.533
##  8 2A        2012   7507    5011           2   0.668        0.705
##  9 6E        2014  11055    4067           1   0.368        0.405
## 10 8C        2016   9416    5064           6   0.538        0.575
## # ... with 150 more rows, and 5 more variables: actual.ballots <int>,
## #   error <dbl>, rel.error <dbl>, outlier <lgl>, turnout.diff <dbl>
```