Data Cleaning

Part 2

```
library(tidyverse)
```

```
── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0
──
✔ dplyr     1.1.4     ✔ readr     2.1.5
✔ forcats   1.0.1     ✔ stringr   1.5.2
✔ ggplot2   4.0.0     ✔ tibble    3.3.0
✔ lubridate 1.9.4     ✔ tidyr     1.3.1
✔ purrr     1.1.0
── Conflicts ────────────────────────────────────────── tidyverse_conflicts()
──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
conflicts to become errors
```

```
library(readr)

raw <- read_csv("data/g24_sov_by_g24_svprec.csv")
```

```
Rows: 51123 Columns: 76
── Column specification
────────────────────────────────────────────────────────
Delimiter: ","
chr (49): FIPS, SVPREC, SVPREC_KEY, ELECTION, GEO_TYPE, ASSAIP01,
ASSDEM01, ...
dbl (27): COUNTY, ADDIST, CDDIST, SDDIST, BEDIST, TOTREG, DEMREG, REPREG,
AI...

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

```
glimpse(raw)
```

```
Rows: 51,123
Columns: 76
$ COUNTY     <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1,…
$ FIPS       <chr> "06001", "06001", "06001", "06001", "06001", "06001",
"0600…
$ SVPREC     <chr> "200100", "200100A", "200200", "200200A", "201400",
"201400…
$ ADDIST     <dbl> 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14,
14,…
$ SVPREC_KEY <chr> "06001200100", "06001200100A", "06001200200",
"06001200200A…
$ ELECTION   <chr> "g24", "g24", "g24", "g24", "g24", "g24", "g24", "g24",
"g2…
$ GEO_TYPE   <chr> "svprec", "svprec", "svprec", "svprec", "svprec",
"svprec",…
$ CDDIST     <dbl> 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12,
12,…
$ SDDIST     <dbl> 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7,
7,…
$ BEDIST     <dbl> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2,…
$ TOTREG     <dbl> 3535, 0, 2442, 0, 3773, 0, 541, 0, 1105, 0, 948, 0, 2721,
0…
$ DEMREG     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ REPREG     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ AIPREG     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ GRNREG     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ LIBREG     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ NLPREG     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ REFREG     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ DCLREG     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ MSCREG     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ TOTVOTE    <dbl> 256, 2804, 262, 1816, 283, 2782, 89, 343, 394, 297, 837,
29…
$ DEMVOTE    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ REPVOTE    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
```

```
$ AIPVOTE    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ GRNVOTE    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ LIBVOTE    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ NLPVOTE    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ REFVOTE    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ DCLVOTE    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ MSCVOTE    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ PRCVOTE    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ ABSVOTE    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0,…
$ ASSAIP01   <chr> "0", "0", "0", "0", "0", "0", "0", "0", "0", "0", "0",
"0",…
$ ASSDEM01   <chr> "94", "444", "117", "348", "107", "588", "45", "105",
"181"…
$ ASSDEM02   <chr> "110", "2023", "91", "1243", "128", "1841", "24", "172",
"1…
$ ASSREP01   <chr> "0", "0", "0", "0", "0", "0", "0", "0", "0", "0", "0",
"0",…
$ ASSREP02   <chr> "0", "0", "0", "0", "0", "0", "0", "0", "0", "0", "0",
"0",…
$ CNGDEM01   <chr> "102", "1668", "108", "1063", "139", "1688", "35", "192",
"…
$ CNGDEM02   <chr> "102", "771", "99", "513", "98", "739", "38", "93", "143",
…
$ CNGIND01   <chr> "0", "0", "0", "0", "0", "0", "0", "0", "0", "0", "0",
"0",…
$ CNGREP01   <chr> "0", "0", "0", "0", "0", "0", "0", "0", "0", "0", "0",
"0",…
$ CNGREP02   <chr> "0", "0", "0", "0", "0", "0", "0", "0", "0", "0", "0",
"0",…
$ PRSAIP01   <chr> "3", "10", "1", "6", "3", "13", "2", "2", "3", "4", "5",
"2…
$ PRSDEM01   <chr> "181", "2562", "207", "1647", "231", "2522", "73", "297",
"…
$ PRSGRN01   <chr> "9", "48", "13", "41", "9", "52", "0", "13", "6", "13",
"23…
$ PRSLIB01   <chr> "1", "10", "2", "3", "4", "13", "0", "0", "3", "2", "3",
"0…
$ PRSPAF01   <chr> "5", "17", "7", "12", "8", "32", "2", "4", "3", "7", "16",
…
```

```
$ PRSREP01   <chr> "51", "108", "26", "83", "17", "111", "11", "23", "55",
"24…
$ PR_2_N     <chr> "58", "493", "45", "342", "39", "399", "17", "45", "52",
"3…
$ PR_2_Y     <chr> "169", "2156", "196", "1385", "226", "2231", "66", "278",
"…
$ PR_32_N    <chr> "78", "636", "55", "439", "74", "536", "14", "60", "81",
"6…
$ PR_32_Y    <chr> "148", "1966", "187", "1261", "190", "2070", "68", "255",
"…
$ PR_33_N    <chr> "136", "1774", "105", "1092", "124", "1509", "30", "127",
"…
$ PR_33_Y    <chr> "86", "784", "126", "584", "133", "1053", "49", "177",
"231…
$ PR_34_N    <chr> "123", "1485", "121", "1027", "144", "1515", "46", "186",
"…
$ PR_34_Y    <chr> "98", "980", "105", "601", "96", "941", "33", "107",
"174",…
$ PR_35_N    <chr> "54", "581", "45", "419", "58", "563", "20", "57", "61",
"5…
$ PR_35_Y    <chr> "171", "2003", "188", "1261", "196", "1988", "58", "248",
"…
$ PR_36_N    <chr> "106", "1356", "142", "888", "146", "1487", "49", "197",
"2…
$ PR_36_Y    <chr> "118", "1223", "99", "786", "119", "1084", "31", "112",
"14…
$ PR_3_N     <chr> "51", "133", "25", "116", "33", "152", "10", "26", "38",
"2…
$ PR_3_Y     <chr> "183", "2553", "220", "1646", "240", "2508", "74", "295",
"…
$ PR_4_N     <chr> "52", "381", "37", "271", "37", "330", "14", "41", "40",
"2…
$ PR_4_Y     <chr> "181", "2294", "209", "1472", "231", "2316", "68", "279",
"…
$ PR_5_N     <chr> "94", "961", "66", "605", "63", "742", "19", "76", "72",
"6…
$ PR_5_Y     <chr> "132", "1660", "168", "1096", "197", "1862", "61", "240",
"…
$ PR_6_N     <chr> "75", "607", "59", "407", "57", "532", "17", "53", "85",
"5…
$ PR_6_Y     <chr> "143", "1958", "180", "1274", "196", "2029", "62", "257",
"…
$ SENDEM01   <chr> "107", "1719", "101", "1102", "136", "1578", "37", "153",
"…
$ SENDEM02   <chr> "103", "809", "114", "516", "105", "908", "34", "133",
"174…
$ SENREP01   <chr> "0", "0", "0", "0", "0", "0", "0", "0", "0", "0", "0",
"0",…
```

```
$ SENREP02   <chr> "0", "0", "0", "0", "0", "0", "0", "0", "0", "0", "0",
"0",…
$ USPDEM01   <chr> "172", "2461", "199", "1572", "217", "2444", "67", "285",
"…
$ USPREP01   <chr> "53", "155", "34", "111", "32", "153", "11", "23", "53",
"3…
$ USSDEM01   <chr> "173", "2487", "207", "1593", "222", "2478", "67", "288",
"…
$ USSREP01   <chr> "55", "155", "29", "109", "33", "151", "12", "23", "51",
"2…
```

```
meta_cols <- c(
  "COUNTY", "FIPS", "SVPREC", "SVPREC_KEY",
  "ELECTION", "GEO_TYPE"
)

raw2 <- raw |>
  mutate(across(all_of(meta_cols), as.character))

clean <- raw2 |>
  mutate(
    across(
      .cols = setdiff(names(raw2), meta_cols),
      .fns = ~ if (is.character(.x) && all(str_detect(na.omit(.x), "^[0-9]+
$"))) {
                 as.numeric(.x)
               } else {
                 .x
               }
    )
  )

summary(clean$TOTVOTE)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
      0       1      91    2202     446 3793980 
```

```
summary(clean$CNGDEM01)
```

```
   Length     Class      Mode 
    51123 character character 
```

```
summary(clean$CNGREP01)
```

```
   Length     Class      Mode 
    51123 character character 
```

```r
colSums(is.na(clean))
```

```
    COUNTY       FIPS     SVPREC     ADDIST SVPREC_KEY   ELECTION   GEO_TYPE 
         0          0          0          0          0          0          0 
    CDDIST     SDDIST     BEDIST     TOTREG     DEMREG     REPREG     AIPREG 
         0          0          0          0          0          0          0 
    GRNREG     LIBREG     NLPREG     REFREG     DCLREG     MSCREG    TOTVOTE 
         0          0          0          0          0          0          0 
   DEMVOTE    REPVOTE    AIPVOTE    GRNVOTE    LIBVOTE    NLPVOTE    REFVOTE 
         0          0          0          0          0          0          0 
   DCLVOTE    MSCVOTE    PRCVOTE    ABSVOTE   ASSAIP01   ASSDEM01   ASSDEM02 
         0          0          0          0          0          0          0 
  ASSREP01   ASSREP02   CNGDEM01   CNGDEM02   CNGIND01   CNGREP01   CNGREP02 
         0          0          0          0          0          0          0 
  PRSAIP01   PRSDEM01   PRSGRN01   PRSLIB01   PRSPAF01   PRSREP01     PR_2_N 
         0          0          0          0          0          0          0 
    PR_2_Y    PR_32_N    PR_32_Y    PR_33_N    PR_33_Y    PR_34_N    PR_34_Y 
         0          0          0          0          0          0          0 
   PR_35_N    PR_35_Y    PR_36_N    PR_36_Y     PR_3_N     PR_3_Y     PR_4_N 
         0          0          0          0          0          0          0 
    PR_4_Y     PR_5_N     PR_5_Y     PR_6_N     PR_6_Y   SENDEM01   SENDEM02 
         0          0          0          0          0          0          0 
  SENREP01   SENREP02   USPDEM01   USPREP01   USSDEM01   USSREP01 
         0          0          0          0          0          0 
```

```r
clean <- clean |>
mutate(across(where(is.numeric), ~ replace_na(.x, 0)))

clean |>
count(SVPREC) |>
filter(n > 1)
```

```
# A tibble: 1,923 × 2
   SVPREC       n
   <chr>    <int>
 1 0000001      2
 2 0000001A     2
 3 0000002      2
 4 0000002A     2
 5 0000003      2
 6 0000003A     2
 7 0000004      2
 8 0000004A     2
 9 0000005      2
10 0000005A     2
# ℹ 1,913 more rows
```

```r
write_csv(clean, "data/g24_sov_by_g24_svprec_clean.csv")
```

Part 5

```r
library(tidyverse)
library(sf)
```

```
Linking to GEOS 3.13.0, GDAL 3.8.5, PROJ 9.5.1; sf_use_s2() is TRUE
```

```r
library(readr)

sr_votes_raw <- read_csv("data/g24_sov_by_g24_srprec.csv")
```

```
Rows: 25245 Columns: 76
```

```
── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
chr (49): FIPS, SRPREC, ELECTION, SRPREC_KEY, GEO_TYPE, ASSAIP01,
ASSDEM01, ...
dbl (27): COUNTY, ADDIST, CDDIST, SDDIST, BEDIST, TOTREG, DEMREG, REPREG,
AI...

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

```r
sr_votes <- sr_votes_raw |>
  mutate(
    SRPREC = as.character(SRPREC),

    dem1 = replace_na(as.numeric(CNGDEM01), 0),
    dem2 = replace_na(as.numeric(CNGDEM02), 0),
    rep1 = replace_na(as.numeric(CNGREP01), 0),
    rep2 = replace_na(as.numeric(CNGREP02), 0),

    dem_votes = dem1 + dem2,
    rep_votes = rep1 + rep2
  ) |>
  select(SRPREC, dem_votes, rep_votes)
```

```
Warning: There were 4 warnings in `mutate()`.
The first warning was:
ℹ In argument: `dem1 = replace_na(as.numeric(CNGDEM01), 0)`.
Caused by warning in `replace_na()`:
! NAs introduced by coercion
ℹ Run `dplyr::last_dplyr_warnings()` to see the 3 remaining warnings.
```

```r
sr_shp <- st_read("data/srprec_state_g24_v01_shp/
srprec_state_g24_v01_shp.shp")
```

```
Reading layer `srprec_state_g24_v01_shp' from data source
  `/Users/joannazhu77/Desktop/STAT133/gerrymandering-joannazhu77/data/
srprec_state_g24_v01_shp/srprec_state_g24_v01_shp.shp'
  using driver `ESRI Shapefile'
```

```
Warning in CPL_read_ogr(dsn, layer, query, as.character(options), quiet, :
GDAL
Message 1:
/Users/joannazhu77/Desktop/STAT133/gerrymandering-joannazhu77/data/
srprec_state_g24_v01_shp/srprec_state_g24_v01_shp.shp
contains polygon(s) with rings with invalid winding order. Autocorrecting
them,
but that shapefile should be corrected using ogr2ogr for example.
```

```
Simple feature collection with 24224 features and 6 fields
Geometry type: MULTIPOLYGON
Dimension:     XY
Bounding box:  xmin: -124.482 ymin: 32.52883 xmax: -114.1312 ymax: 42.0095
Geodetic CRS:  NAD83
```

```r
sr_geo <- sr_shp |>
  mutate(SRPREC = as.character(SRPREC)) |>
  st_transform(3310) |>
  st_set_precision(1) |>
  st_make_valid() |>
  st_collection_extract("POLYGON") |>
  mutate(sr_area = as.numeric(st_area(geometry)))
```

```r
sr_joined <- sr_geo |>
  left_join(sr_votes, by = "SRPREC")
```

```
Warning in sf_column %in% names(g): Detected an unexpected many-to-many
relationship between `x` and `y`.
i Row 11 of `x` matches multiple rows in `y`.
i Row 376 of `y` matches multiple rows in `x`.
i If a many-to-many relationship is expected, set `relationship =
  "many-to-many"` to silence this warning.
```

```r
ab604 <- st_read("data/AB604/AB604.shp") |>
  st_transform(3310)
```

```
Reading layer `AB604' from data source
  `/Users/joannazhu77/Desktop/STAT133/gerrymandering-joannazhu77/data/AB604/
AB604.shp'
  using driver `ESRI Shapefile'
Simple feature collection with 52 features and 15 fields
Geometry type: MULTIPOLYGON
Dimension:     XY
Bounding box:  xmin: -13857270 ymin: 3832931 xmax: -12705030 ymax: 5162404
Projected CRS: WGS 84 / Pseudo-Mercator
```

```r
names(ab604)
```

```
 [1] "DISTRICT"   "A_POP"      "DEVIATION"  "CVAP"       "HSP_CVAP"
 [6] "IND_CVAP"   "BLK_CVAP"   "ASN_CVAP"   "WHT_CVAP"   "CVAP_PCT"
[11] "HSP_CVAP_P" "IND_CVAP_P" "BLK_CVAP_P" "ASN_CVAP_P" "WHT_CVAP_P"
[16] "geometry"
```

```r
sr_for_int <- sr_joined |>
  select(SRPREC, sr_area, dem_votes, rep_votes)

cd_for_int <- ab604 |>
  select(DISTRICT)

sr_cd_intersect <- st_intersection(sr_for_int, cd_for_int) |>
  mutate(
    piece_area = as.numeric(st_area(geometry)),
    weight = piece_area / sr_area,
    dem_weighted = dem_votes * weight,
    rep_weighted = rep_votes * weight
  )
```

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

```
district_votes_ab604 <- sr_cd_intersect |>
  st_drop_geometry() |>
  group_by(DISTRICT) |>
  summarise(
    dem_votes = sum(dem_weighted, na.rm = TRUE),
    rep_votes = sum(rep_weighted, na.rm = TRUE),
    total_two_party = dem_votes + rep_votes,
    dem_share = dem_votes / total_two_party,
    .groups = "drop"
  )

district_votes_ab604
```

```
# A tibble: 52 × 5
   DISTRICT dem_votes rep_votes total_two_party dem_share
   <chr>        <dbl>     <dbl>           <dbl>     <dbl>
 1 01         252630.   230082.         482713.     0.523
 2 02         327415.   303132.         630547.     0.519
 3 03         206640.   195614.         402254.     0.514
 4 04         214196.   163606.         377802.     0.567
 5 05         235364.   330737.         566100.     0.416
 6 06         177363.   154598.         331961.     0.534
 7 07         194160.   152272.         346432.     0.560
 8 08         230286.   119332.         349618.     0.659
 9 09         169808.   119462.         289270.     0.587
10 10         252544.   131217.         383761.     0.658
# ℹ 42 more rows
```

```
write_csv(district_votes_ab604,
          "data/district_votes_2024_under_ab604.csv")
```