Ch. 7 topics: Graphing complex survey data

Math 255, St. Clair

Goal:

Use survey data to create a visualization that represents the population.

- Self-weighting samples: (SRS)
  - Just use basic EDA tools: histogram, boxplot, scatterplots, bar graphs
- Stratified samples: self-weighting within strata
  - Basic EDA within each strata: side-by-side boxplots, faceted histograms/scatterplots, grouped bar graphs
- Clustered samples: self-weighting within clusters
  - Basic EDA within each cluster: side-by-side boxplots, faceted histograms/scatterplots, grouped bar graphs

Goal:

Use survey data to create a visualization that represents the population.

- But what if we want to visualize the distribution of a variable for the entire population, not just one strata/cluster?

- What if we have a more complex sampling design?

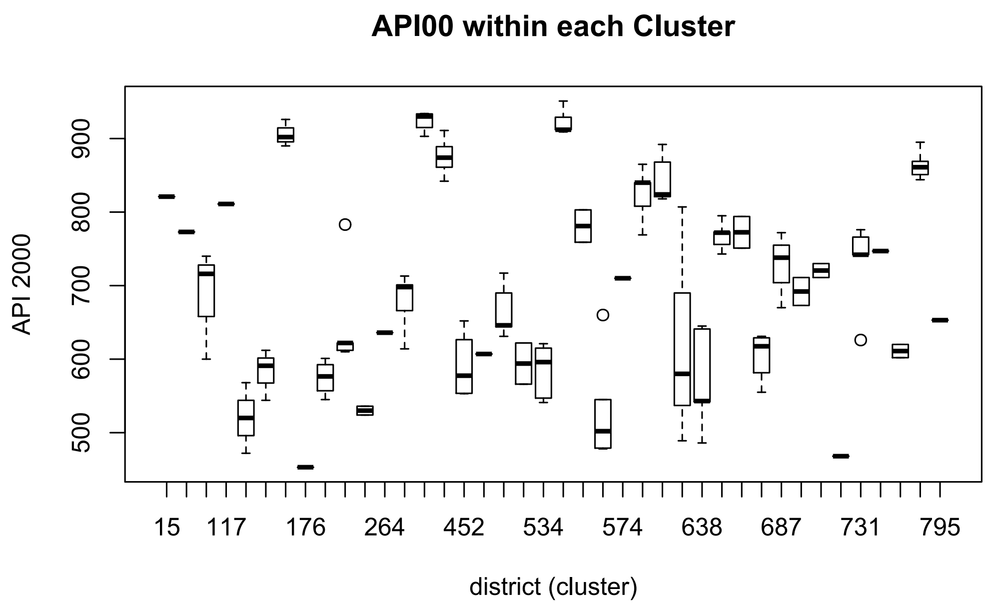
Two-stage cluster: CA API scores

- Recall the two-stage cluster sample of schools in CA
  - Cluster = district
  - Elements = schools
  - Unequal cluster and sample sizes so sampling weights vary across clusters

- Goal: understand API scores in 2000 (api00)

API scores within districts:

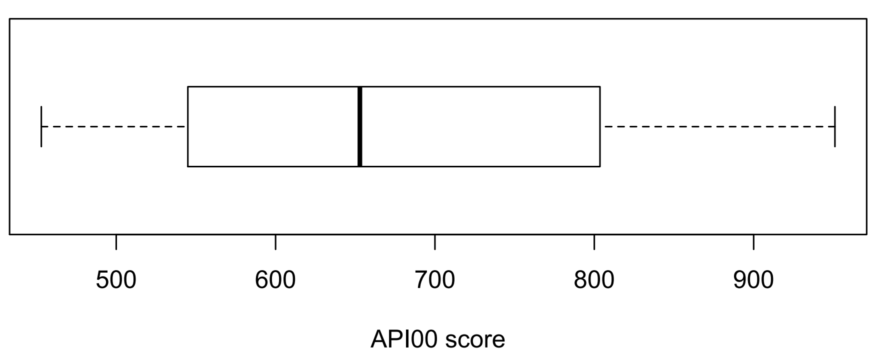
Represents `API00` scores within districts and variation between districts in the population



API scores in all of CA:

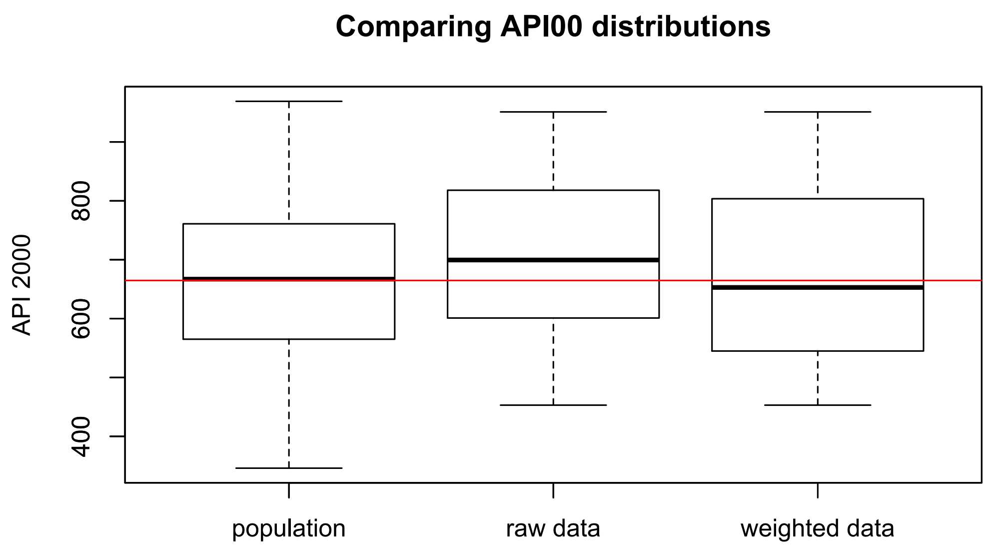
- What if we want a boxplot that represents API00 scores for all schools in CA, not just the schools in our sample?
  - Create the usual survey design object and use `svyboxplot`

```
> api.design<-svydesign(id=~dnum+snum, fpc=~fpc1+fpc2, weights = ~pw, data=
> svyboxplot(api00~1,api.design, horizontal=TRUE, xlab="API00 score")
```



API scores in all of CA:

Why does the unweighted (raw) data misrepresent API00 scores across CA?

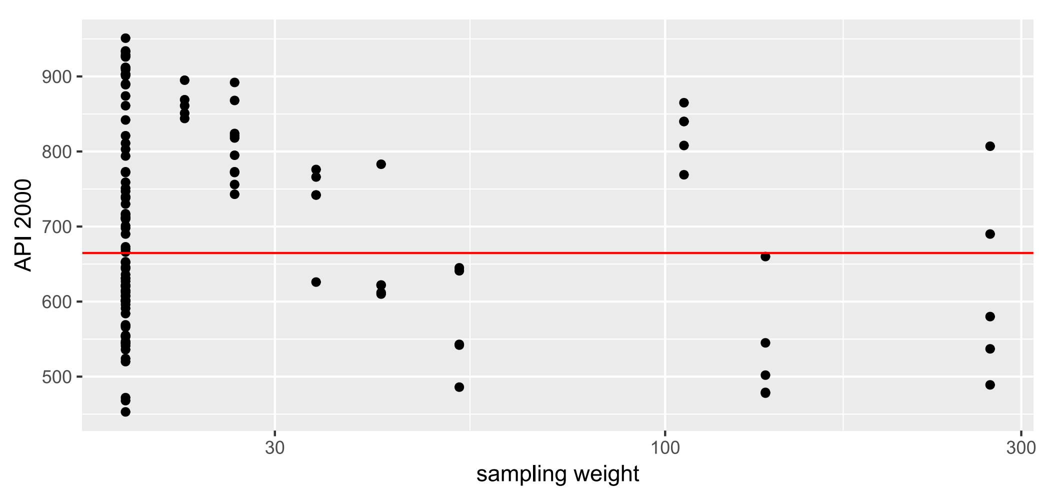


API scores in all of CA:

Answer is in the relationship between weights and response

- In the sample: schools with high score overrepresented (many schools with high scores but low sampling weights)

```
> ggplot(apiclus2, aes(x=pw, y=api00)) + geom_point() +
+   geom_hline(yintercept=mean(apipop$api00), color="red") +
+   scale_x_log10() + labs(x="sampling weight",y="API 2000")
```



Agpps: PPS ag survey data

- 15 Counties selected with probability proportional to `acres87`
- Goal: estimate average `acres92`
  - Sample mean is 614,764
  - HT estimate is 342,552
  - Population mean is 306,677
- Graphically:
  - (unweighted) sample of 15 counties is adequate for EDA for the sample, looking for outliers
  - (unweighted) sample will not reflect the population distribution of acres92
  - Solution: use the sampling weights when graphing

Weighted Boxplots

- Usual boxplot:
  - Find 5 number summary (min,Q1,median,Q3,max)
  - ID outliers using 1.5 IQR rule
  - Plot 5 number summary and outliers
- Weighted boxplot
  - Find Q1, median, Q3 using the weighted empirical cumulative distribution function (ecdf):

$$\hat{F}(a) = P(Y \leq a) = \frac{\sum_{\text{all } i \text{ where } y_i \leq a} w_i}{\sum_{i \in \mathcal{S}} w_i}$$

  - E.g. Q1 is the value $q_{.25}$ where $F(q_{.25}) \approx 0.25$

Weighted Boxplots

```
> ordered.agpps <- arrange(agpps, acres92) %>% select(acres92, pii)
> ordered.agpps %>% mutate(weight = 1/pii,
+                          wt.ecdf = cumsum(weight)/sum(weight),
+                          unwt.ecdf = 1:15/15)
   acres92         pii    weight   wt.ecdf  unwt.ecdf
1    70936 0.001137612 879.03433 0.3033396 0.06666667
2   204443 0.003234862 309.13220 0.4100157 0.13333333
3   297003 0.004906196 203.82390 0.4803518 0.20000000
4   300970 0.004586659 218.02362 0.5555880 0.26666667
5   353683 0.005341568 187.21095 0.6201913 0.33333333
6   370572 0.005135069 194.73935 0.6873925 0.40000000
7   395023 0.005715855 174.95195 0.7477654 0.46666667
8   397883 0.006072270 164.68306 0.8045947 0.53333333
9   551148 0.007990101 125.15486 0.8477834 0.60000000
10  596103 0.009043974 110.57086 0.8859395 0.66666667
11  678590 0.010440319  95.78251 0.9189924 0.73333333
12  879694 0.013558648  73.75367 0.9444435 0.80000000
13 1026353 0.015417708  64.86048 0.9668258 0.86666667
14 1117134 0.016036380  62.35821 0.9883445 0.93333333
15 1981938 0.029606891  33.77592 1.0000000 1.00000000
```

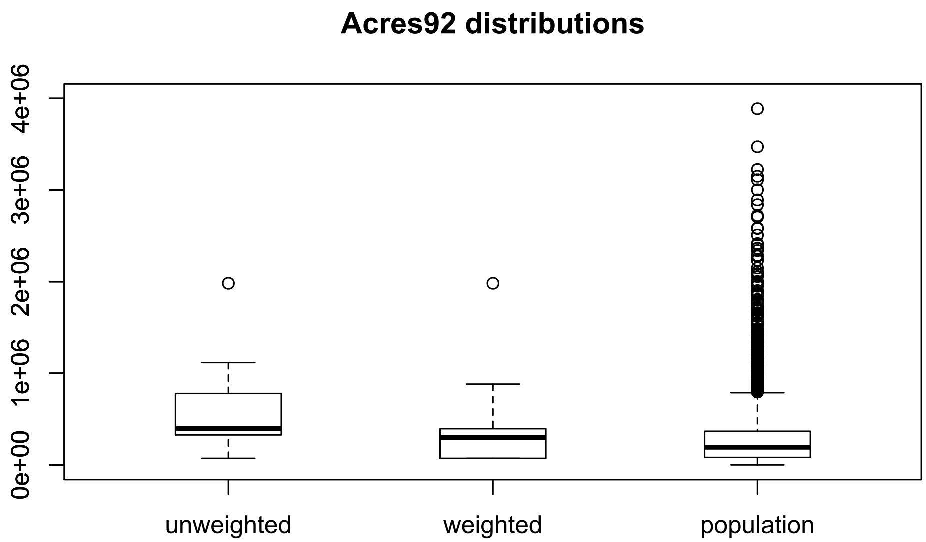

Weighted Boxplots

- In the sample, the median `acres92` is 397,883
- The estimated population median `acres92` is 298,039

```
> svyquantile(~acres92,ag.pps, c(0,.25,.5,.75,1))
            0  0.25    0.5     0.75       1
acres92 70936 70936 298039 395135.5 1981938
> quantile(agpps$acres92,c(0,.25,.5,.75,1))
       0%        25%        50%        75%       100%
  70936.0   327326.5   397883.0   779142.0 1981938.0
```

Agpps: PPS ag survey data

Counties with higher `acres92` have a higher inclusion probability: raw data favors large response values



Weighted Histograms

- Usual (density) histogram:
  - Divide data into equal width bins (b=width)
  - Count the number of data points in each bin
  - Height = (proportion of observations in bin)/b
  - Area of bar = proportion of observations

- Weighted (density) histogram
  - Height is weighted proportion in each bin:

$$\text{height of bin } j = \frac{\sum_{\text{all } y_i \text{ in bin } j} w_i}{b \sum_{i \in \mathcal{S}} w_i}$$

Agpps: PPS ag survey data

Use `svyhist(~acres92, ag.pps)` to generated a weighted histogram