Graphing complex survey data

Week 10 (ch 7)

Stat 260, 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
- 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?

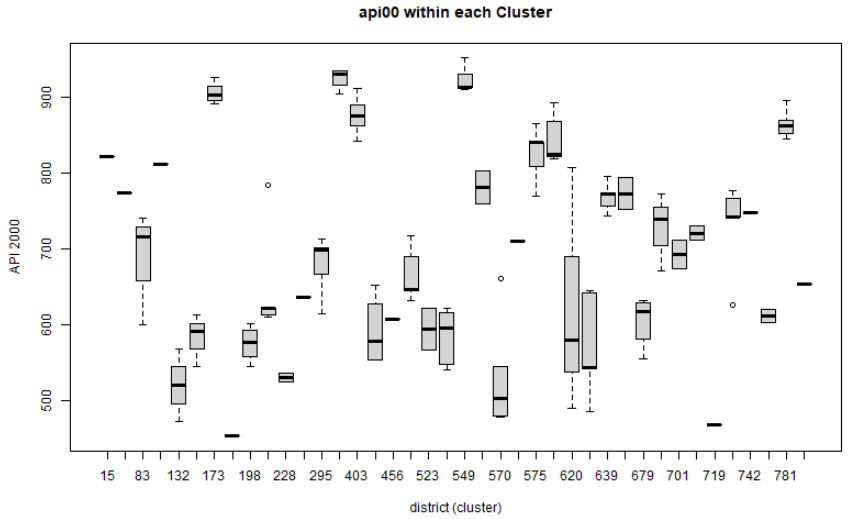
Two-stage cluster: California API scores

- (week 7) 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)

```
schools_design <- svydesign(id= ~dnum + snum,
                            fpc= ~N + district_size,
                            weights = ~wts,
                            data=schools)
```


API scores within districts:

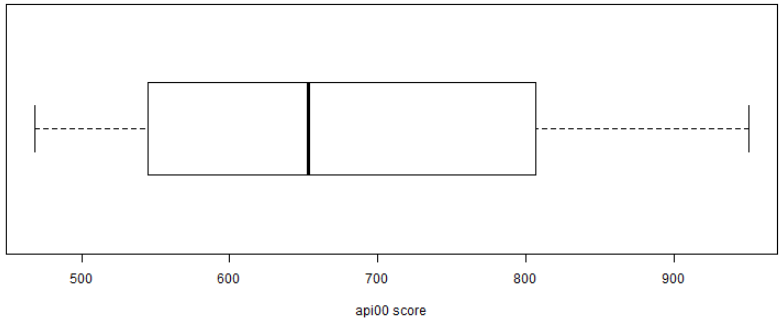
Represents `API00` scores within districts and variation between districts in the population



API scores in all of CA:

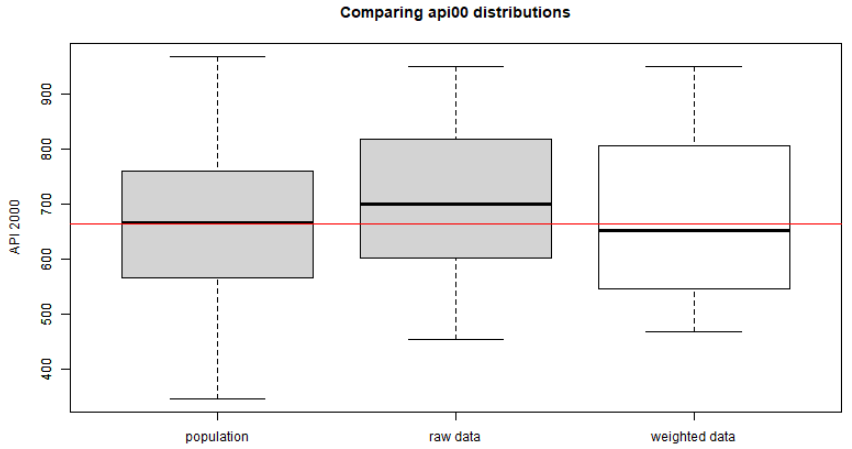
- What if we want a boxplot that represents API00 scores for all schools in CA, not just the schools in our sample?
  - Create the usual survey design object and use `svyboxplot`

```
svyboxplot(api00~1, schools_design,
           horizontal=TRUE, xlab="api00 score")
```



API scores in all of CA:

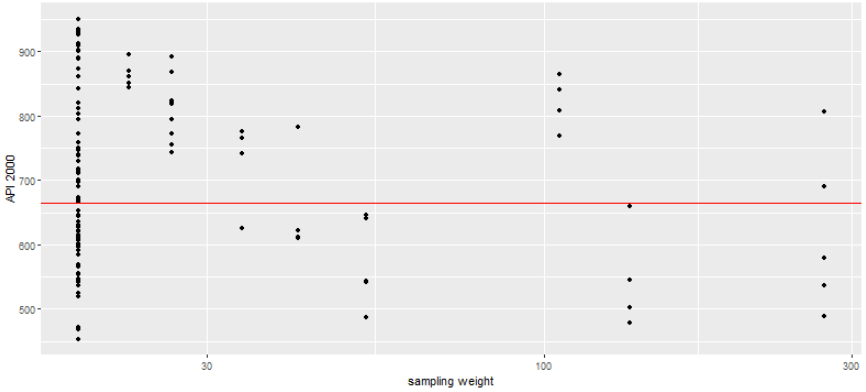
Why does the unweighted (raw) data misrepresent API00 scores across CA?



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)



Agpps: PPS ag survey data

- 15 Counties selected with probability proportional to `acres87`
- Goal: estimate average `acres92`
  - Sample mean is 849,371
  - HT estimate is 405,054
  - Population mean is 308,582
- 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):

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

  - The $\hat{\theta}_q$ quantile is the value where $F(\hat{\theta}_q) \geq q$.

Weighted Boxplots

```
ordered_agpps <- arrange(agpps, acres92) %>% select(acres92, Selectio
ordered_agpps %>% mutate(weight = 1/SelectionProb,
                         wt.ecdf = cumsum(weight)/sum(weight),
                         unwt.ecdf = 1:15/15)
## # A tibble: 15 x 5
##    acres92 SelectionProb weight wt.ecdf unwt.ecdf
##      <dbl>         <dbl>  <dbl>   <dbl>     <dbl>
##  1  161745       0.00286  349.    0.151    0.0667
##  2  174627       0.00297  337.    0.297    0.133
##  3  175847       0.00291  344.    0.446    0.2
##  4  194022       0.00312  320.    0.584    0.267
##  5  310184       0.00484  206.    0.673    0.333
##  6  332358       0.00525  191.    0.756    0.4
##  7  518907       0.00785  127.    0.811    0.467
##  8  545670       0.00818  122.    0.864    0.533
##  9  878447       0.0155    64.6   0.892    0.6
## 10 1152965       0.0182    54.9   0.916    0.667
## 11 1466580       0.0225    44.4   0.935    0.733
## 12 1484583       0.0240    41.7   0.953    0.8
## 13 1619482       0.0253    39.5   0.970    0.867
## 14 1639965       0.0250    40.0   0.987    0.933
## 15 2085181       0.0339    29.5   1        1
```


Weighted Boxplots

- Ignoring weights, the median `acres92` is 545,670

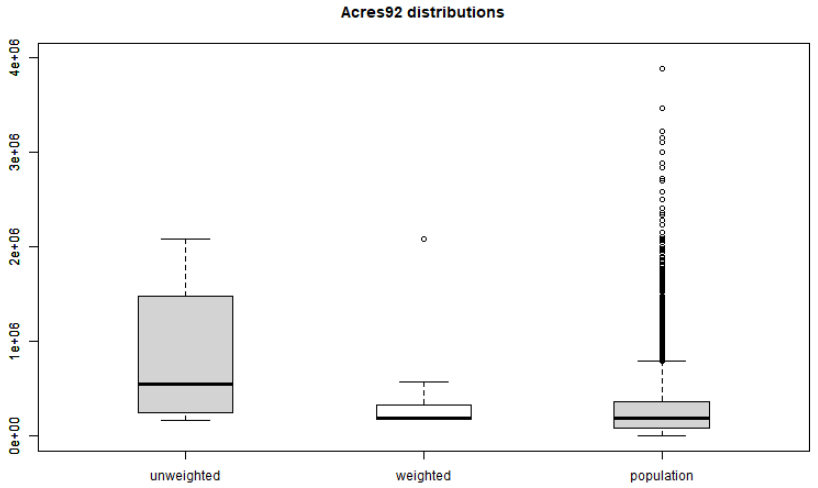
```
quantile(agpps$acres92,c(0,.25,.5,.75,1))
##      0%     25%     50%     75%    100%
##  161745  252103  545670 1475582 2085181
```

- Using weights, the estimated population median `acres92` is 194,022

```
svyquantile(~acres92,ag_pps, c(0,.25,.5,.75,1), ci=FALSE)
## $acres92
##           0   0.25    0.5   0.75       1
## [1,] 174627 174627 194022 332358 2085181
##
## attr(,"hasci")
## [1] FALSE
## attr(,"class")
## [1] "newsvyquantile"
```


Agpps: PPS ag survey data

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



Weighted Histograms

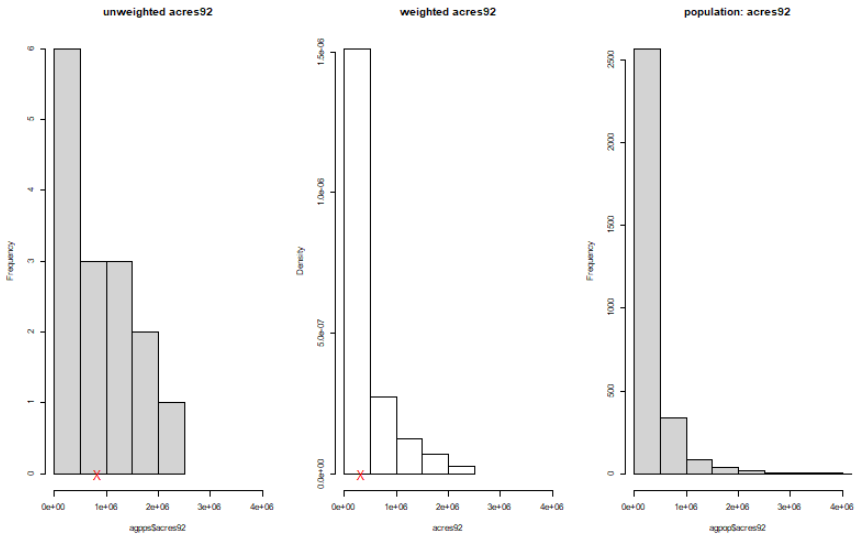
- Usual (density) histogram:
  - Divide data into equal width bins (b=width)
  - Count the number of data points in each bin
  - Height = (proportion of observations in bin)/b
  - Area of bar = proportion of observations

- 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

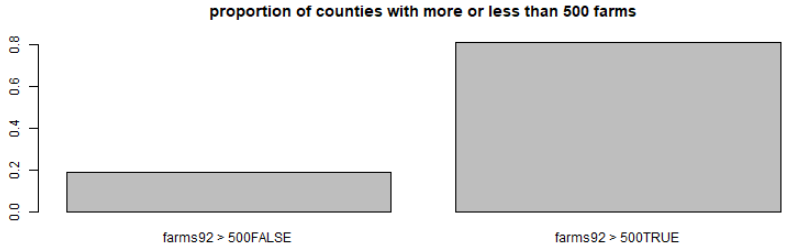
Use `svyhist(~acres92, ag_pps)` to generated a weighted histogram



Bar plots

Use `barplot` to plot proportions obtained from `svymean`

```
props <- svymean(~farms92 > 500, ag_pps)
props
##                         mean     SE
## farms92 > 500FALSE 0.18892 0.1038
## farms92 > 500TRUE  0.81108 0.1038
barplot(props, main = "proportion of counties with more or less than
```



Stacked Bar plots

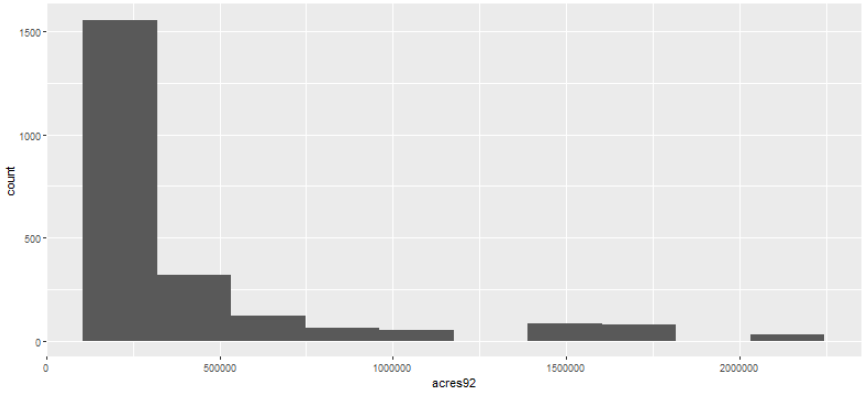
Proportion of counties with more or less than 500 farms by region (NC or not):

```
props <- svyby(~farms92 > 500, ~ region == "NC", ag_pps, svymean)
props
##       region == "NC" farms92 > 500FALSE farms92 > 500TRUE se.farms
## FALSE          FALSE          0.2555625         0.7444375
## TRUE            TRUE          0.1463512         0.8536488
##       se.farms92 > 500TRUE
## FALSE            0.1442014
## TRUE             0.1445314
barplot(props, beside = FALSE)
```

ggplot2 options

We can create a weighted histogram by adding a `weight` aesthetic:

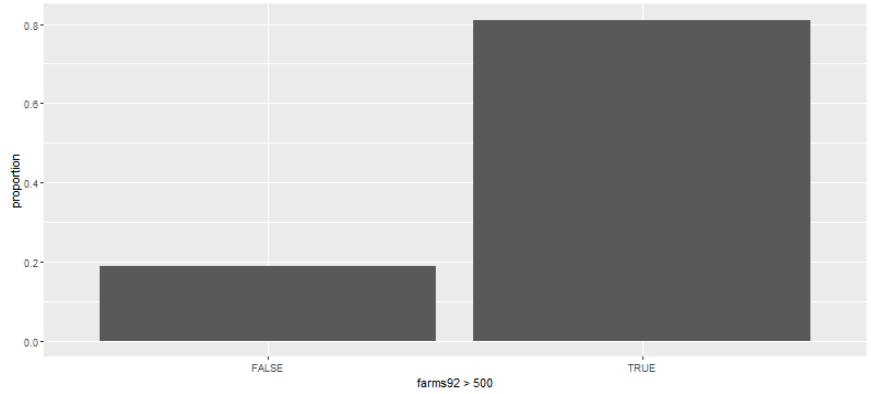
```
ggplot(agpps, aes(x = acres92, weight = SamplingWeight)) +
  geom_histogram(bins = 10)
```



ggplot2 options

Get proportions using `geom_bar` with `weight` equal to $w_i / \sum w_i$

```
ggplot(agpps, aes(x = farms92 > 500)) +
  geom_bar(aes(weight = SamplingWeight/sum(SamplingWeight))) +
  labs(y = "proportion")
```



ggplot2 options

Stacked bar graphs using `fill` position don't need

```
ggplot(agpps, aes(x = region == "NC",fill = farms92 > 500)) +
  geom_bar(aes(weight = SamplingWeight), position = "fill")+
  labs(y = "proportion")
```