SE formulas

Chriss Jordan Oboa

4/19/2021

Packages

We use the `tidyverse` package, as always. We also use

- the `infer` package to generate sampling distributions.
- the `scales` package for a nicer axis with dollar values or percentages.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(infer)
library(ggdist)
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

Populations

We load the hypothetical population of $5 bills:

```
bills <- read_csv("data/bills_population.csv")
```

```
## Rows: 100000 Columns: 1
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (1): age
```

```
##
## 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.
```

We create a population to match the results of the 2020 election: 145,615 people cast a vote for President, and 77,675 of those people voted for Biden. The following chunk includes all the official results from the Frederick County Board of Elections.

```
votes <- tibble(vote = c(rep("Trump", 63682),
                         rep("Biden", 77675),
                         rep("Jorgensen", 2282),
                         rep("Hawkins", 686),
                         rep("Segal", 243),
                         rep("WriteIn", 1047)))
```

Sampling distributions

The following chunk takes 10,000 random samples of size 500 from the population of $5 bills and records the mean age in each sample. It may take half a minute to do its work.

```
sample_means <- bills %>%
  rep_slice_sample(n = 500, reps = 10000) %>%
  summarize(x_bar = mean(age))
```

The following chunk takes 10,000 random samples of size 500 from the population of voters and records the proportion of Biden votes in each sample. It may take half a minute to do its work.

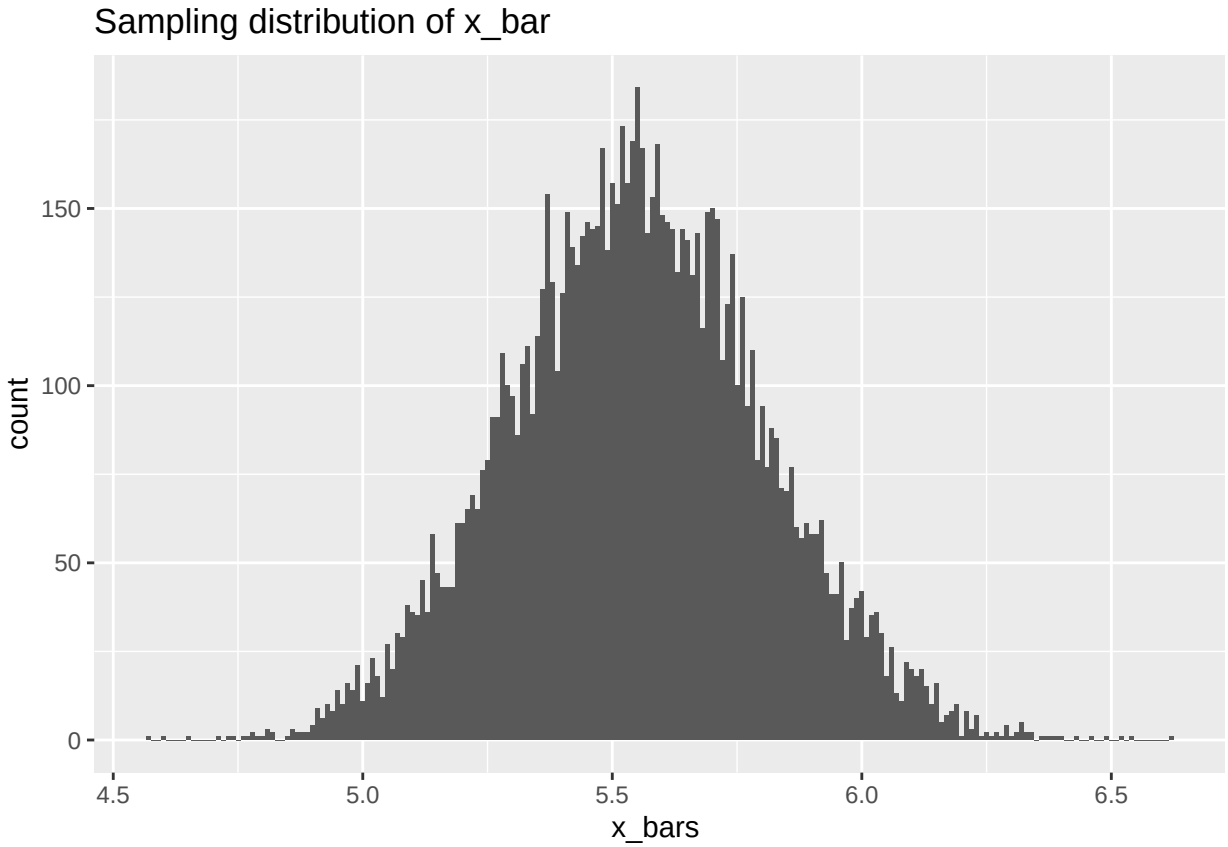
```
sample_props <- votes %>%
  rep_slice_sample(n = 500, reps = 10000) %>%
  count(vote) %>%
  mutate(p_hat = n /sum(n)) %>%
  filter(vote == "Biden") %>% ungroup
```

Checking your guesses

Sample means

Make a histogram of the sample means for samples of 500 $5 bills.

```
ggplot(data = sample_means, mapping = aes(x = x_bar))+
  geom_histogram(binwidth = 0.01)+
  labs(x = "x_bars ",
    title = "Sampling distribution of x_bar")
```



Calculate the center (mean) of the sampling distribution and the standard error.

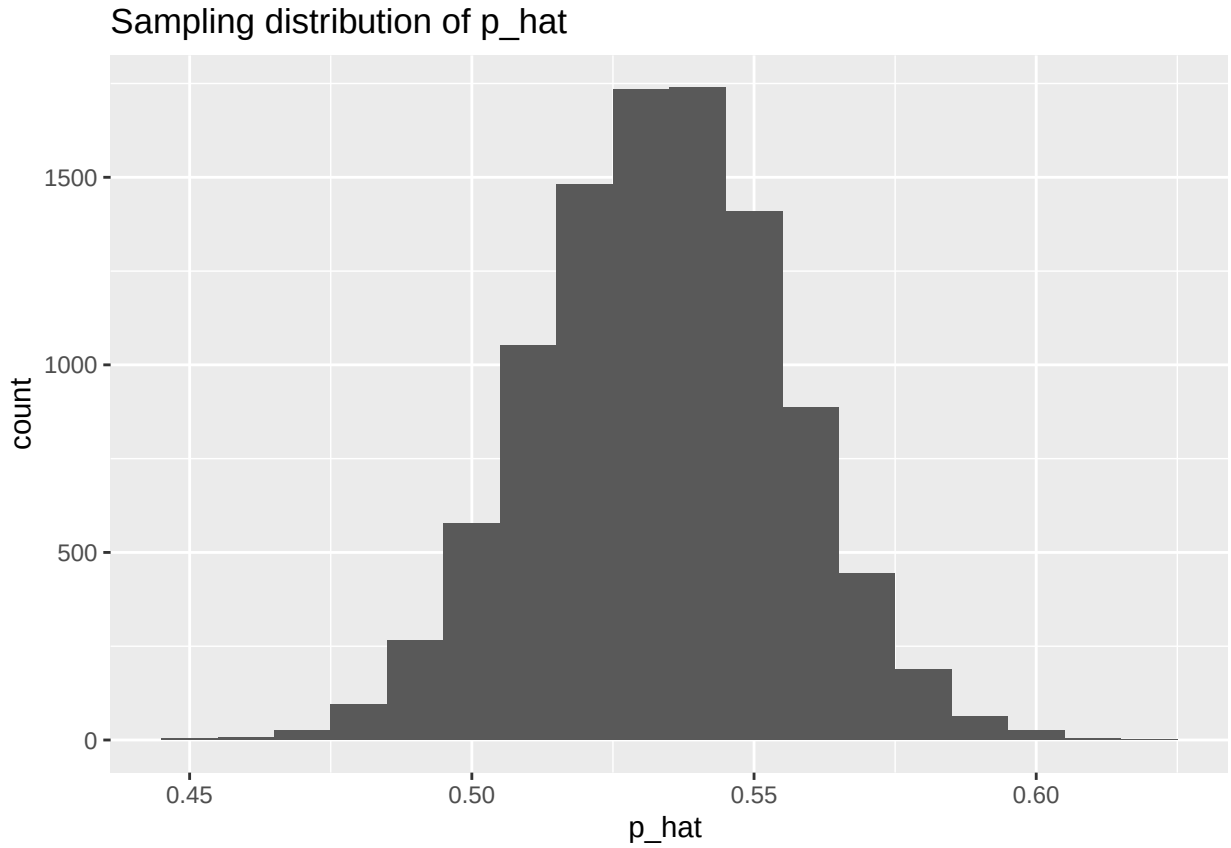
```
sample_means %>%
  summarize(x_bar_mean= mean (x_bar), std_x_bar= sd(x_bar))
```

```
## # A tibble: 1 x 2
##   x_bar_mean std_x_bar
##        <dbl>     <dbl>
## 1       5.55     0.255
```

Sample proportions

Make a historgram of teh sample proportions of Biden votes for samples of 500 voters.

```
ggplot(data = sample_props, mapping = aes(x = p_hat))+
  geom_histogram(binwidth = 0.01)+
  labs(x = "p_hat ",
    title = "Sampling distribution of p_hat")
```



Calculate the center (mean) of the sampling distribution and the standard error.

```
sample_props %>%
  summarize(p_hat_mean= mean (p_hat), std_p_hat= sd(p_hat))
```

```
## # A tibble: 1 x 2
##   p_hat_mean std_p_hat
##        <dbl>     <dbl>
## 1      0.533    0.0219
```