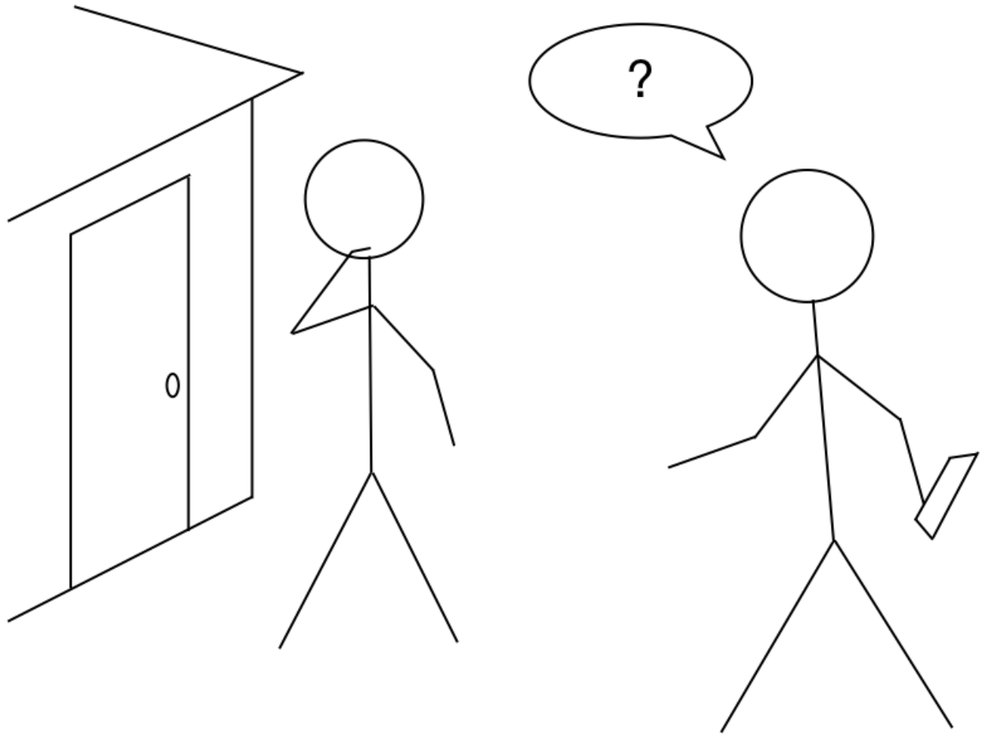
The General Social Survey

INFERENCE FOR CATEGORICAL DATA IN R

Andrew Bray

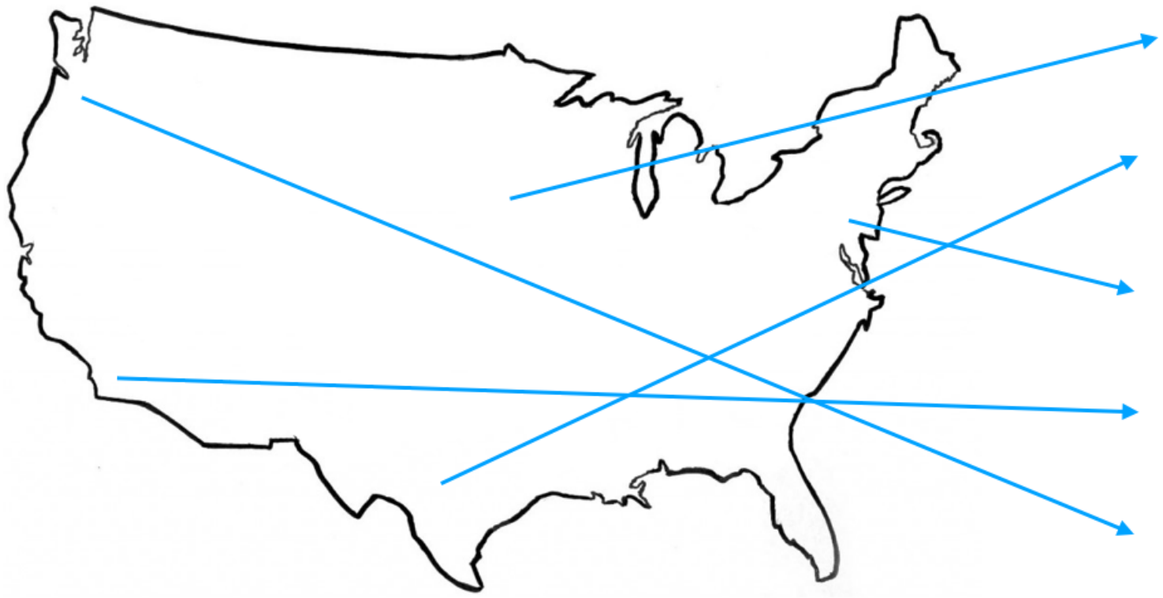
Assistant Professor of Statistics at Reed College

GSS Data Explorer

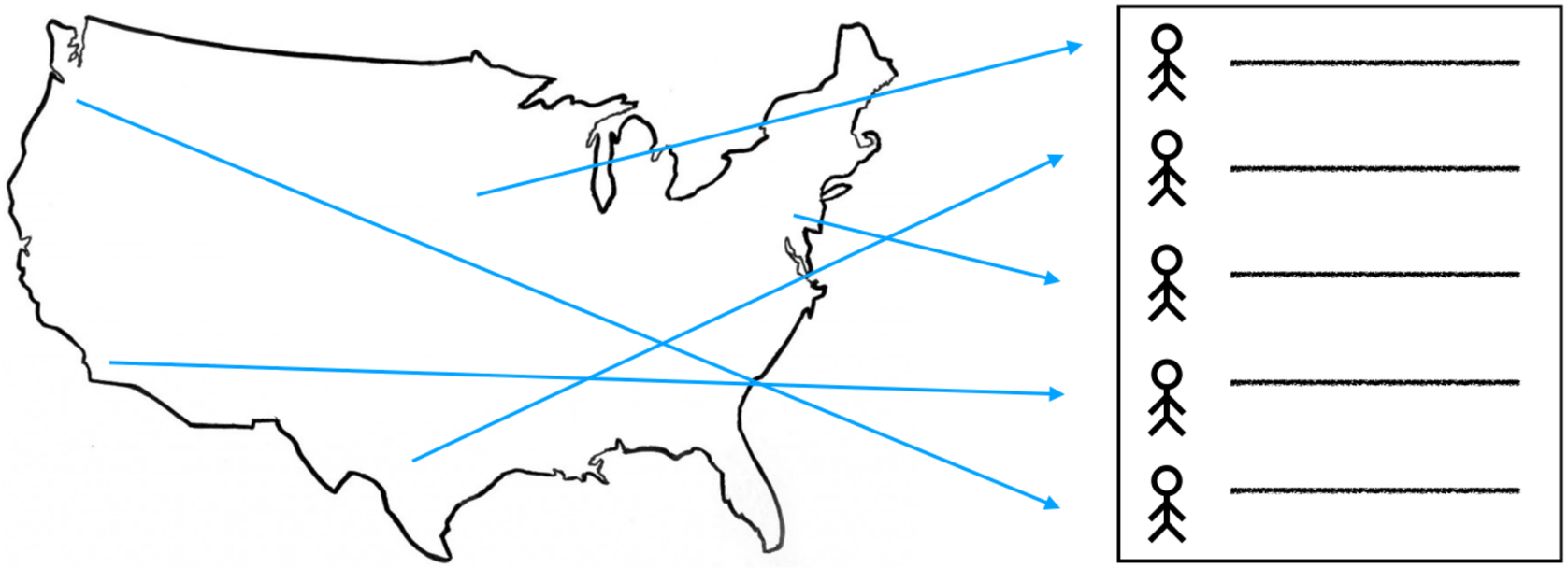


GSS Data Explorer

GSS Data Explorer



GSS Data Explorer



GSS Data Explorer

Exploring GSS

```
library(dplyr)
glimpse(gss)
```

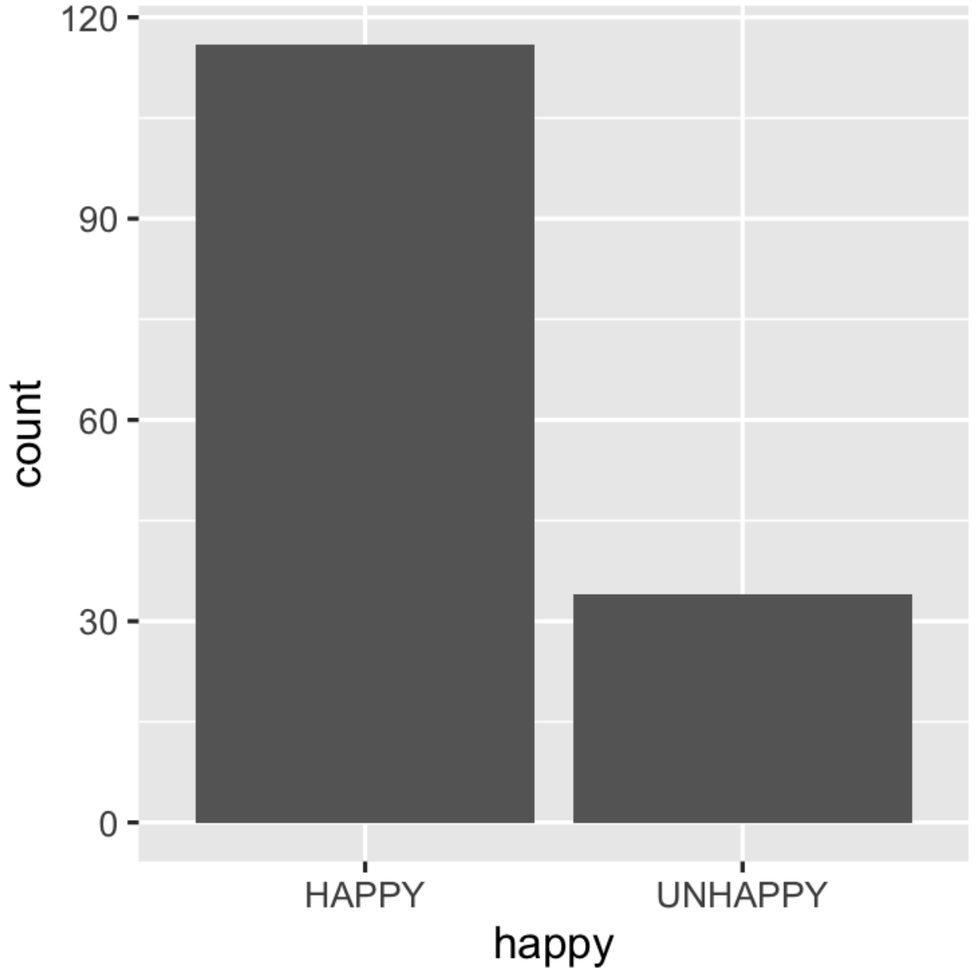
```
Observations: 3,300
Variables: 25
$ id       <dbl> 518, 1092, 2094, 229, 979, 554, 491, 319, 3143, 1...
$ year     <dbl> 1982, 1982, 1982, 1982, 1982, 1982, 1982, 1982, 1...
$ age      <fct> 49, 22, 26, 75, 71, 33, 56, 33, 69, 40, 44, 42, 5...
$ class    <fct> WORKING CLASS, WORKING CLASS, WORKING CLASS, LOWE...
$ degree   <fct> HIGH SCHOOL, HIGH SCHOOL, HIGH SCHOOL, LT HIGH SC...
$ sex      <fct> MALE, MALE, MALE, MALE, FEMALE, FEMALE, MALE, FEM...
```

```
$ happy    <fct> HAPPY, HAPPY, HAPPY, HAPPY, HAPPY, HAPPY, HAPPY, ...
```

Exploring GSS

```
gss2016 <- filter(gss, year == 2016)
```

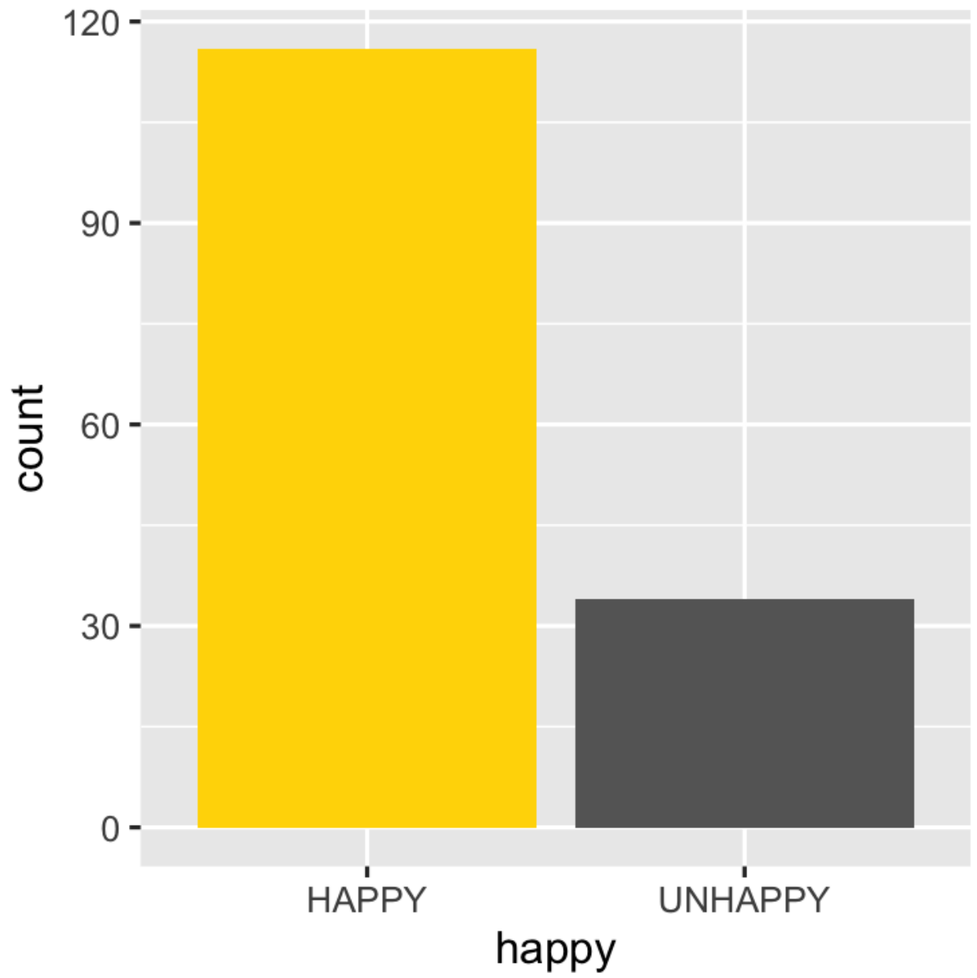
```
ggplot(gss2016, aes(x = happy)) +
  geom_bar()
```



Exploring GSS

```
gss2016 <- filter(gss, year == 2016)
```

```
ggplot(gss2016, aes(x = happy)) +
  geom_bar()
```



Exploring GSS

```
p_hat <- gss2016 %>%
  summarize(prop_happy = mean(happy == "HAPPY")) %>%
  pull()
```

```
p_hat
```

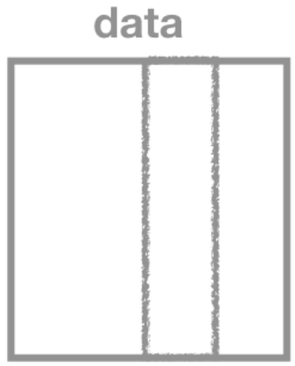
```
0.7733333
```

General 95% confidence interval

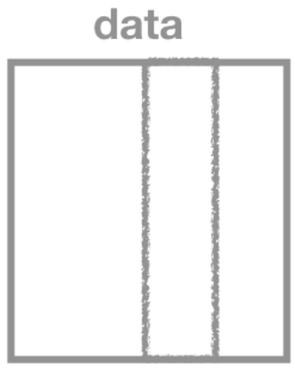
$$(\hat{p} - 2 \times SE, \hat{p} + 2 \times SE)$$

Sample proportion plus or minus two standard errors

Bootstrap

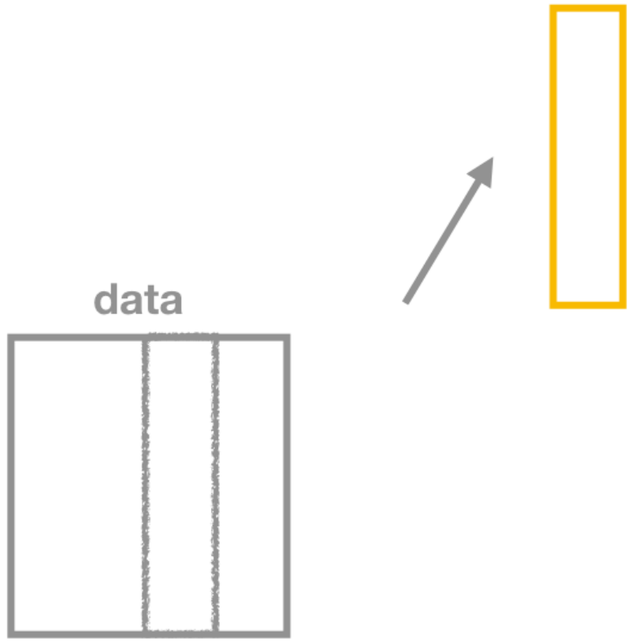


Bootstrap



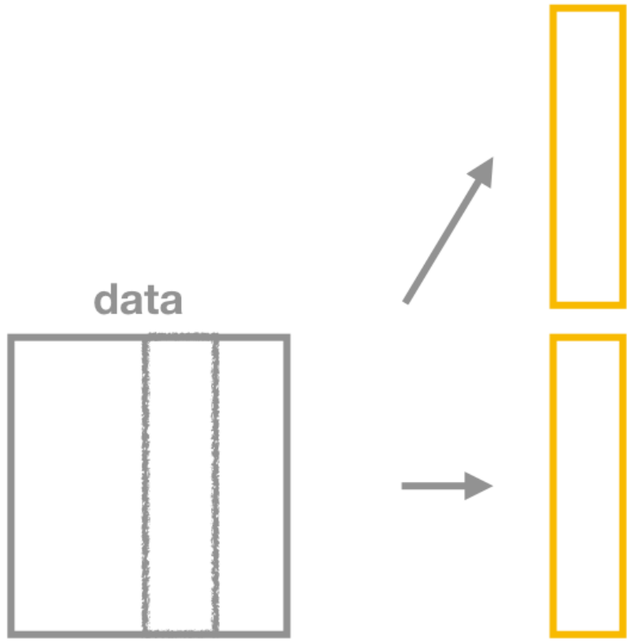
`specify()`

Bootstrap



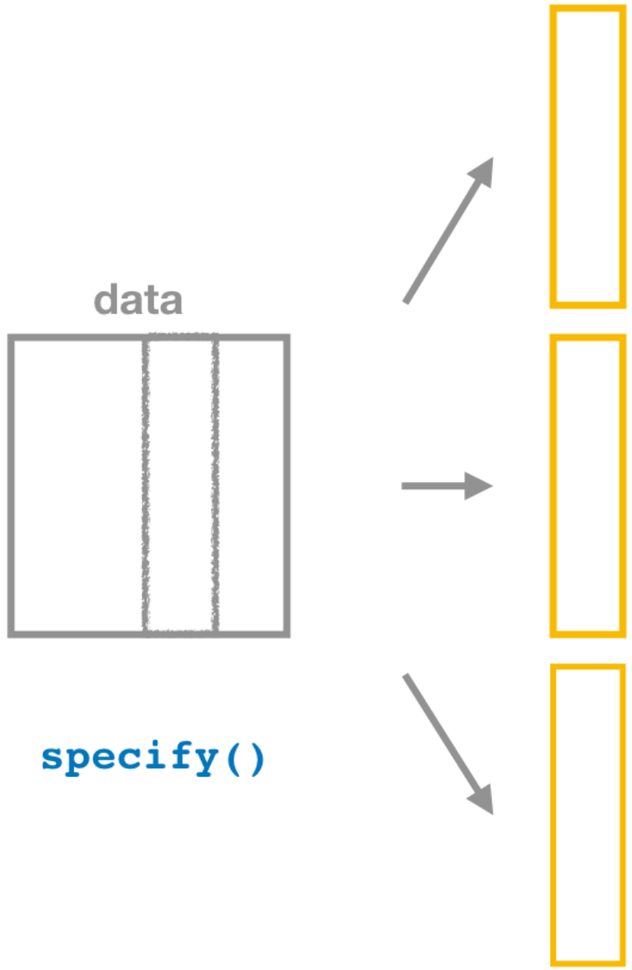
`specify()`

Bootstrap

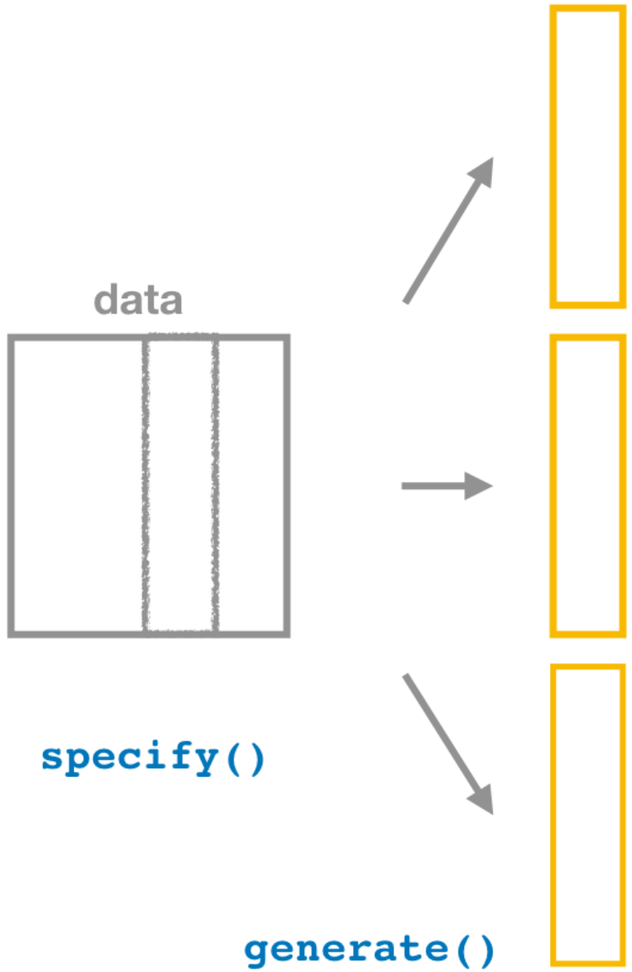


`specify()`

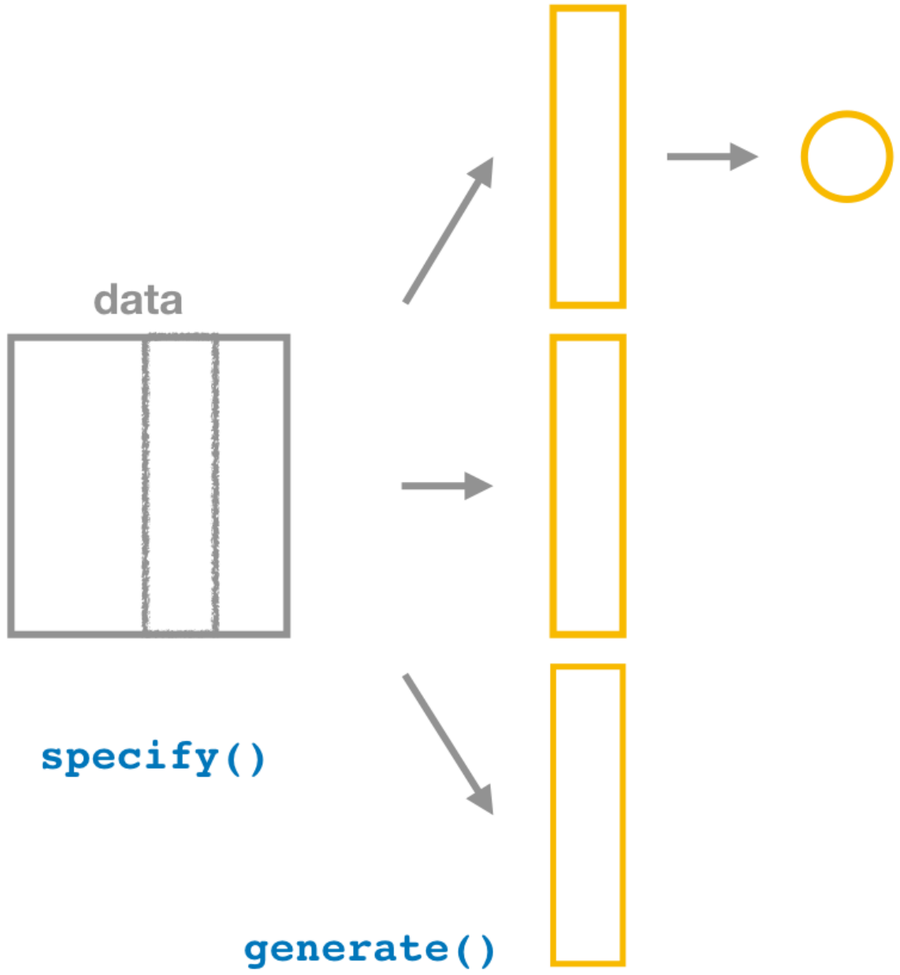
Bootstrap



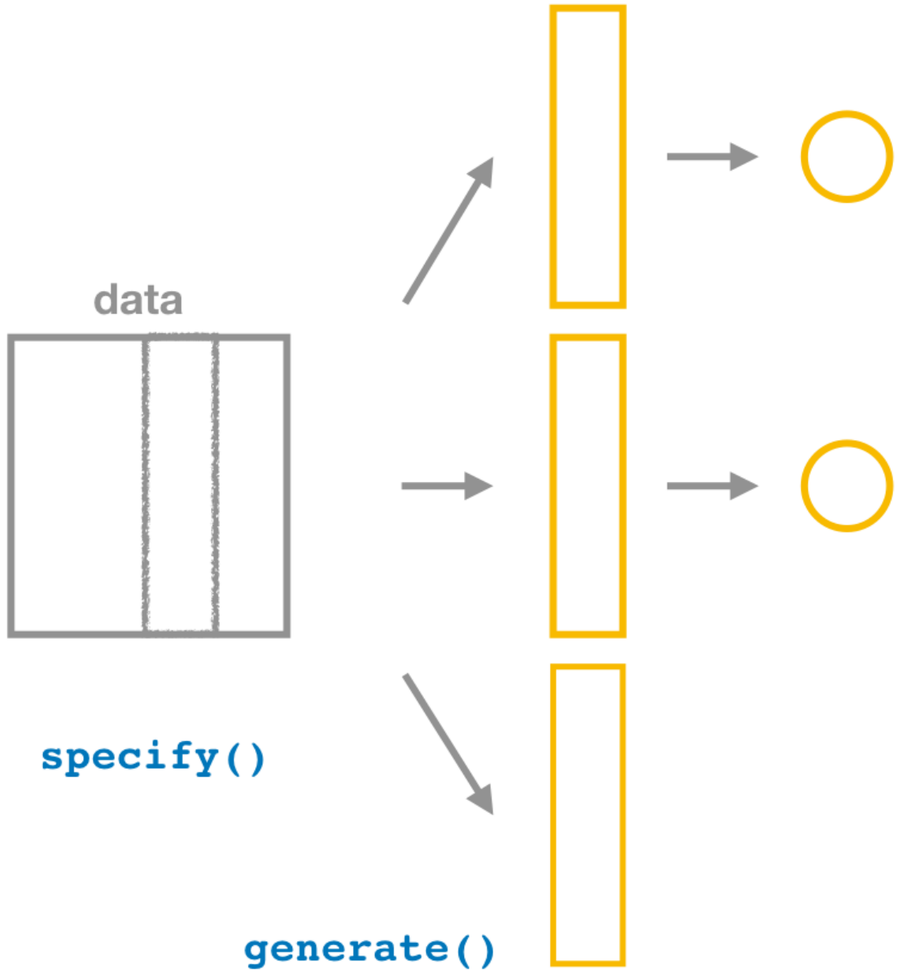
Bootstrap



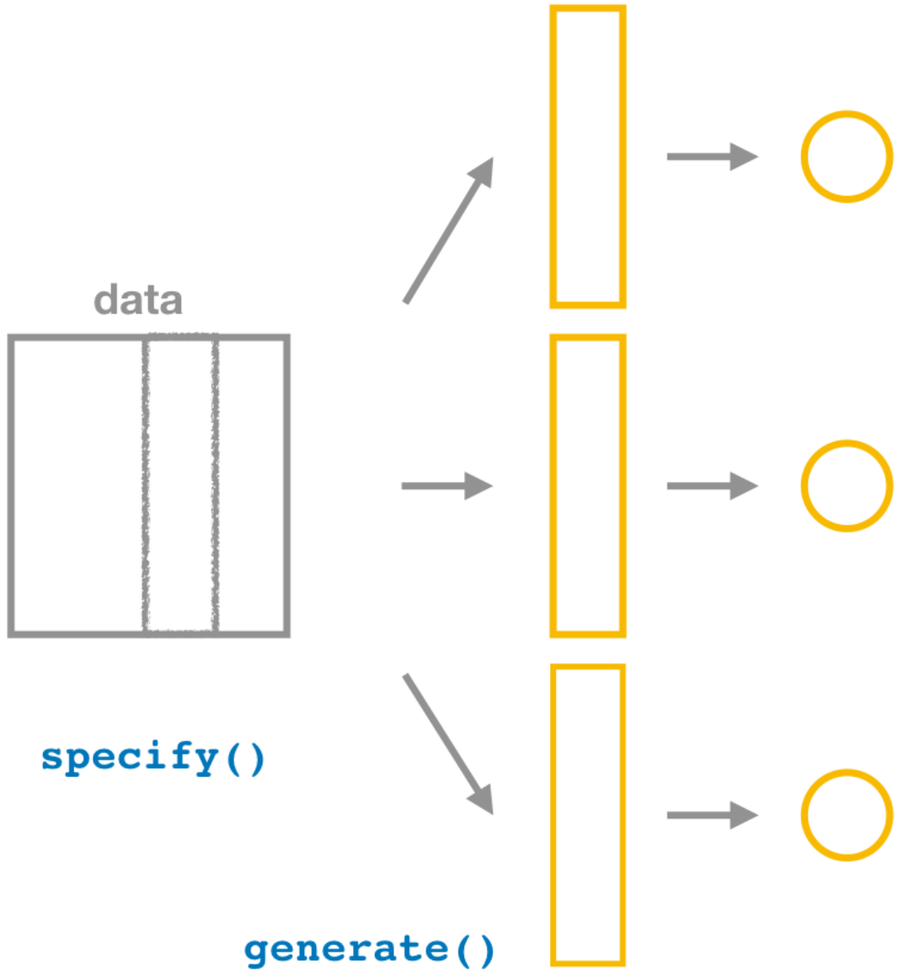
Bootstrap



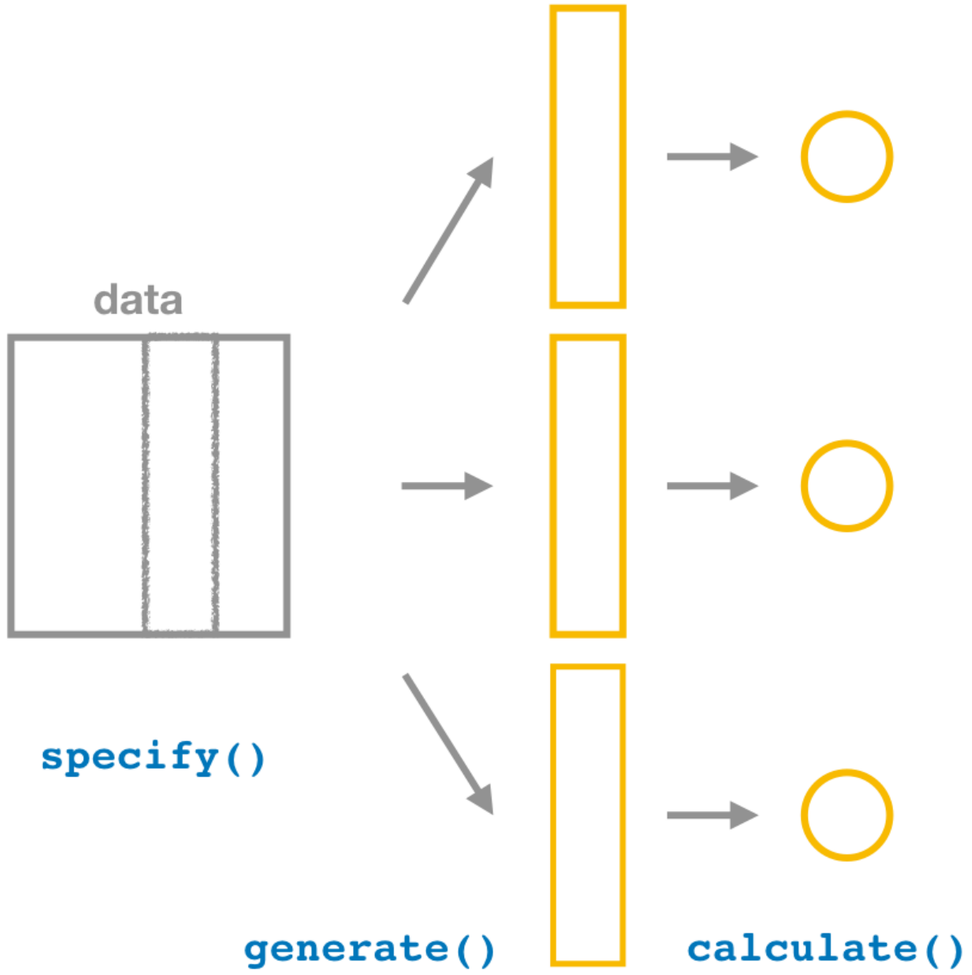
Bootstrap



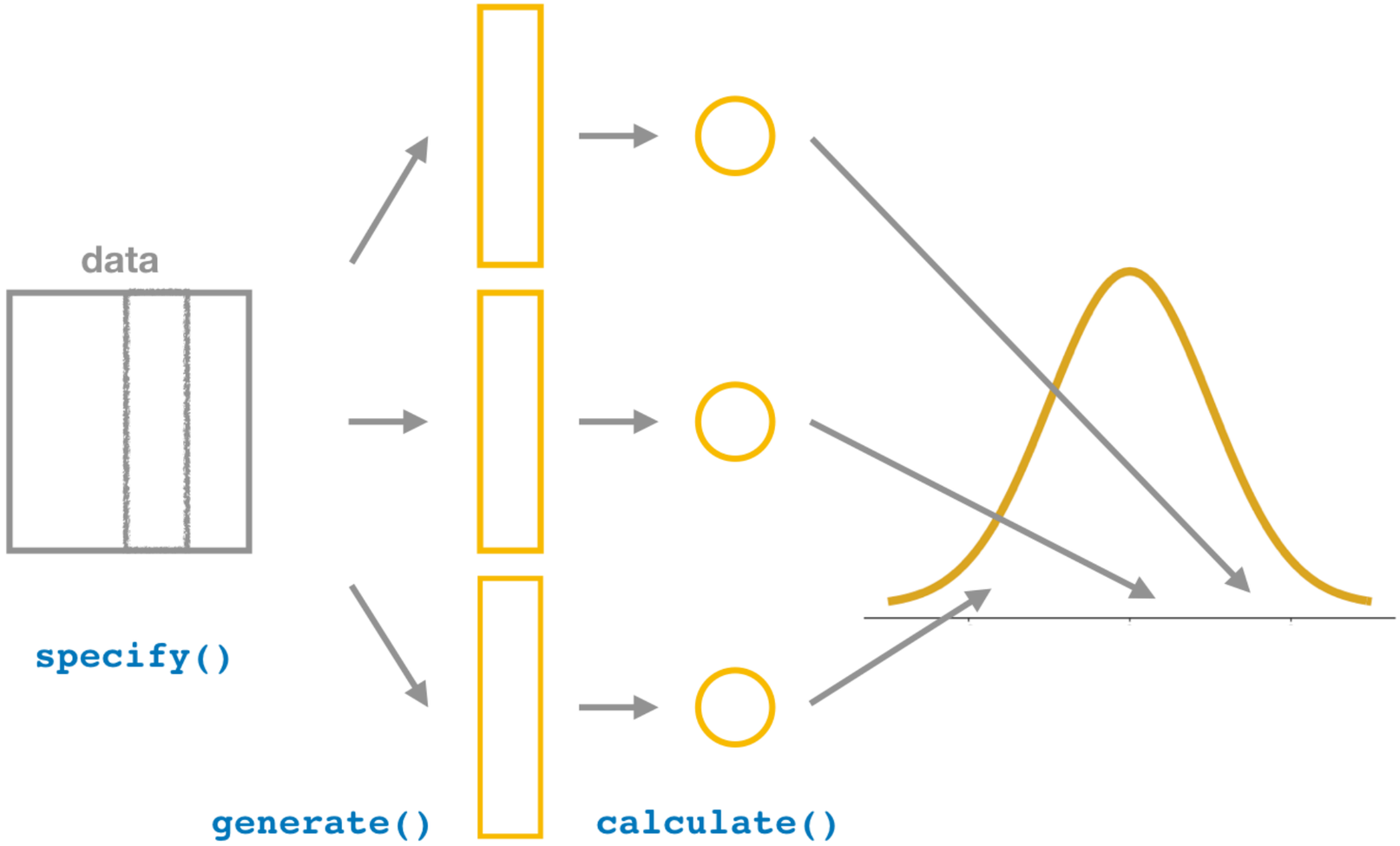
Bootstrap



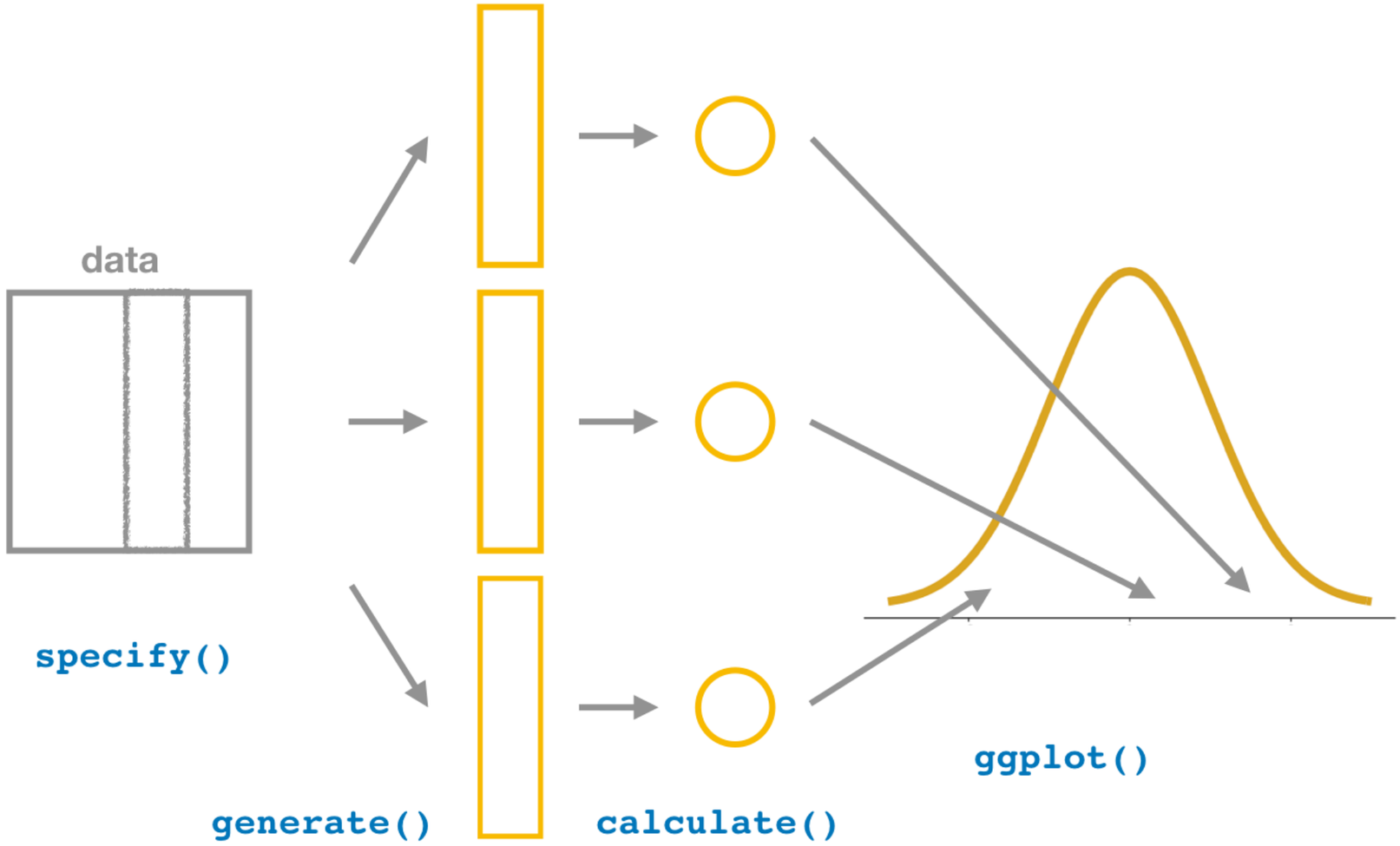
Bootstrap



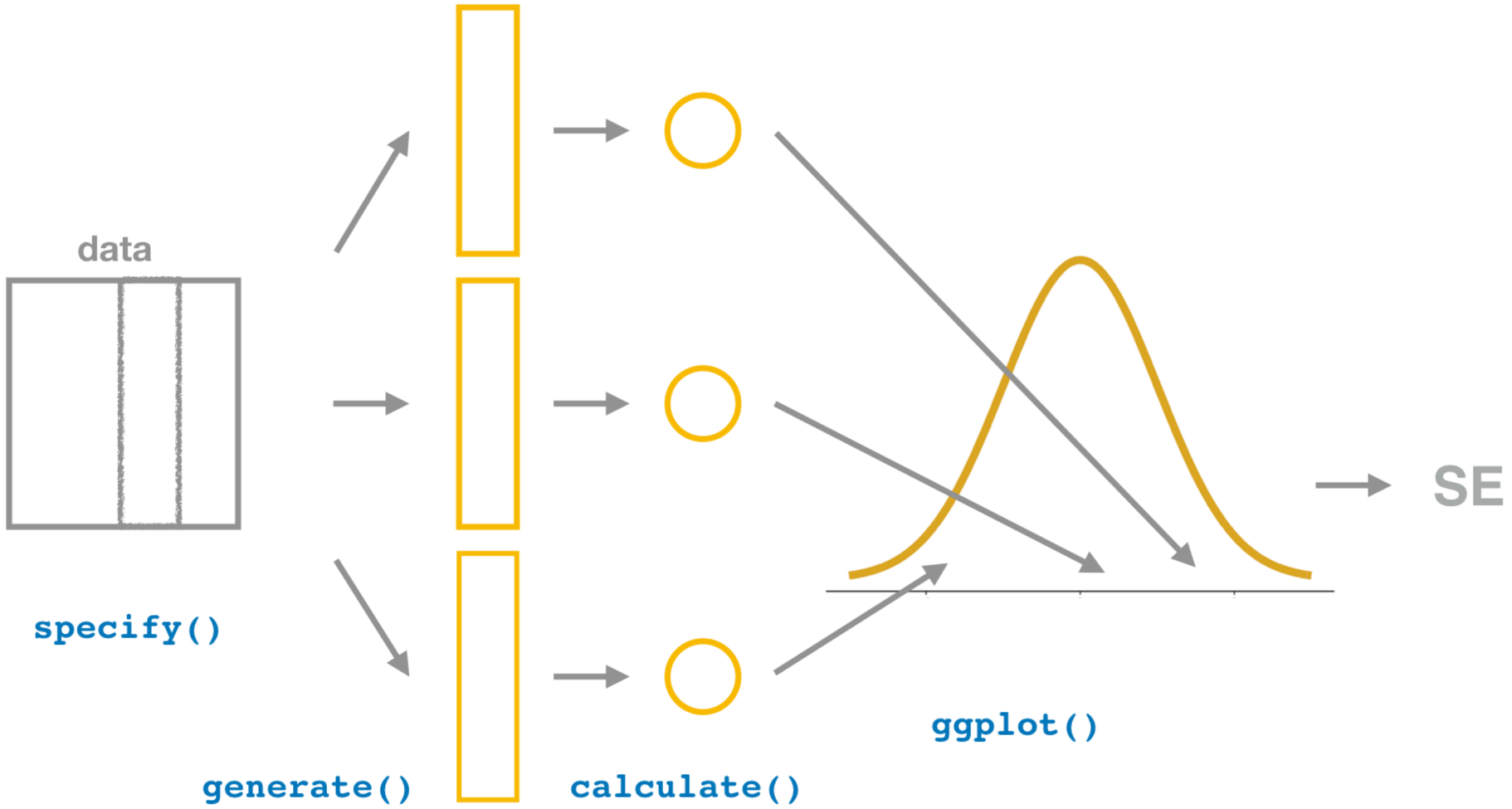
Bootstrap



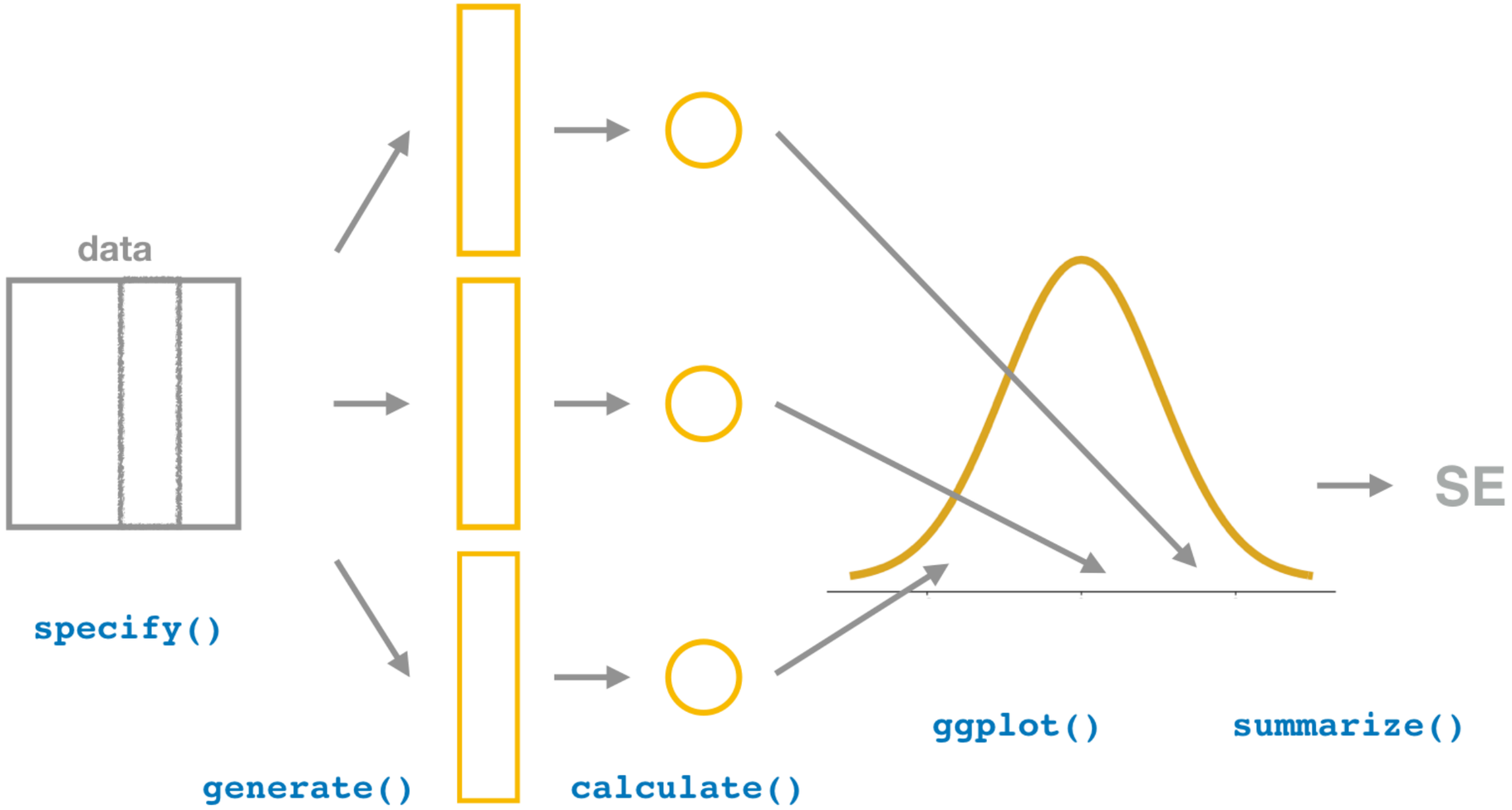
Bootstrap



Bootstrap



Bootstrap



Bootstrap Confidence Interval

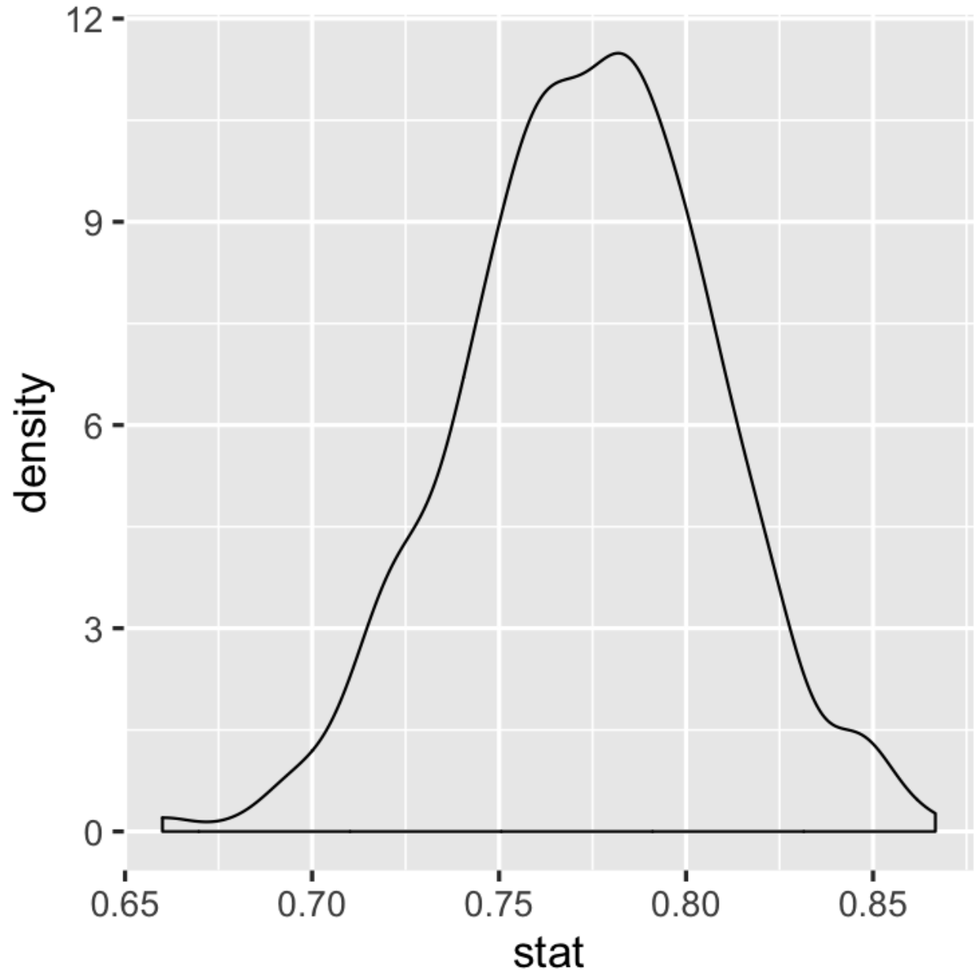
```
library(infer)
boot <- gss2016 %>%
  specify(response = happy,
          success = “HAPPY”) %>%
  generate(reps = 500,
           type = "bootstrap") %>%
  calculate(stat = "prop")
```

```
boot
```

```
Response: happy (factor)
# A tibble: 500 x 2
   replicate  stat
       <int> <dbl>
 1         1 0.827
 2         2 0.740
 3         3 0.780
 4         4 0.773
 5         5 0.747
 6         6 0.753
```

Bootstrap Confidence Interval

```
ggplot(boot, aes(x = stat)) +
  geom_density()
```



Bootstrap Confidence Interval

```
SE <- boot %>%
  summarize(sd(stat)) %>%
  pull()
```

```
SE
```

```
0.03482251
```

$$(\hat{p} - 2 \times SE, \hat{p} + 2 \times SE)$$

```
c(p_hat - 2 * SE, p_hat + 2 * SE)
```

```
0.7051883 0.8412784
```

Let's practice!

INFERENCE FOR CATEGORICAL DATA IN R

Interpreting a Confidence Interval

INFERENCE FOR CATEGORICAL DATA IN R

Andrew Bray

Assistant Professor of Statistics at Reed College

Confidence intervals

Conclusion: the true proportion of Americans that are happy is between 0.705 and 0.841.

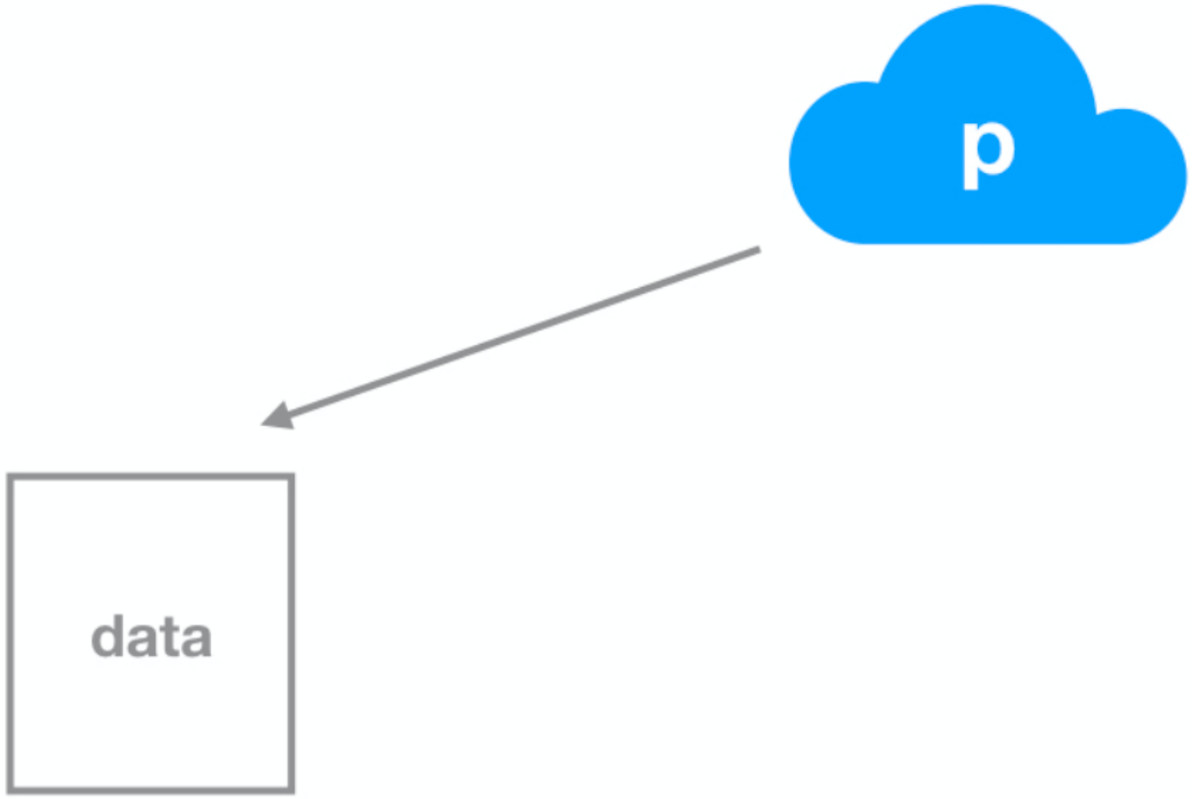
What do we mean by *confident*?

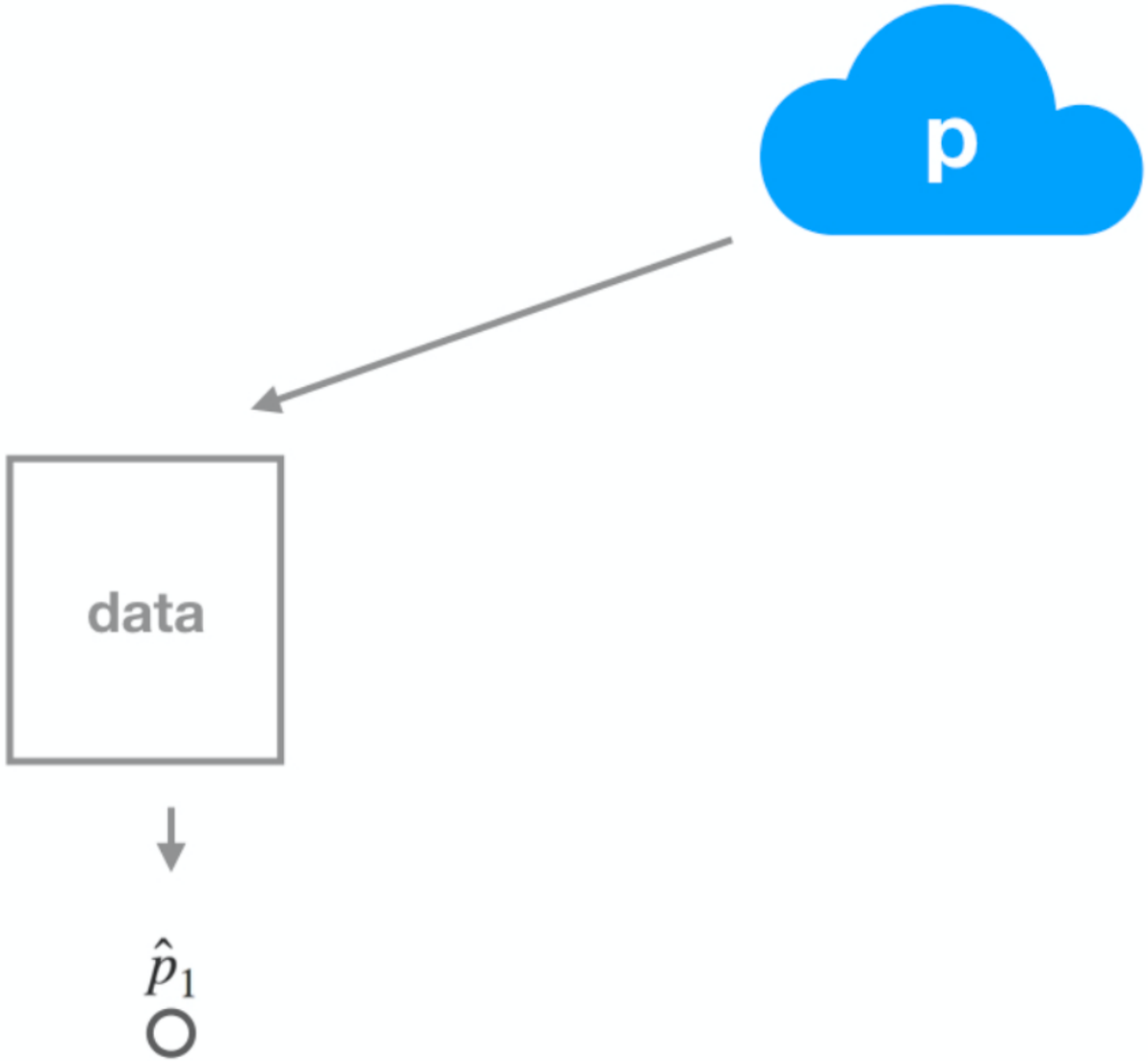
Dataset 1

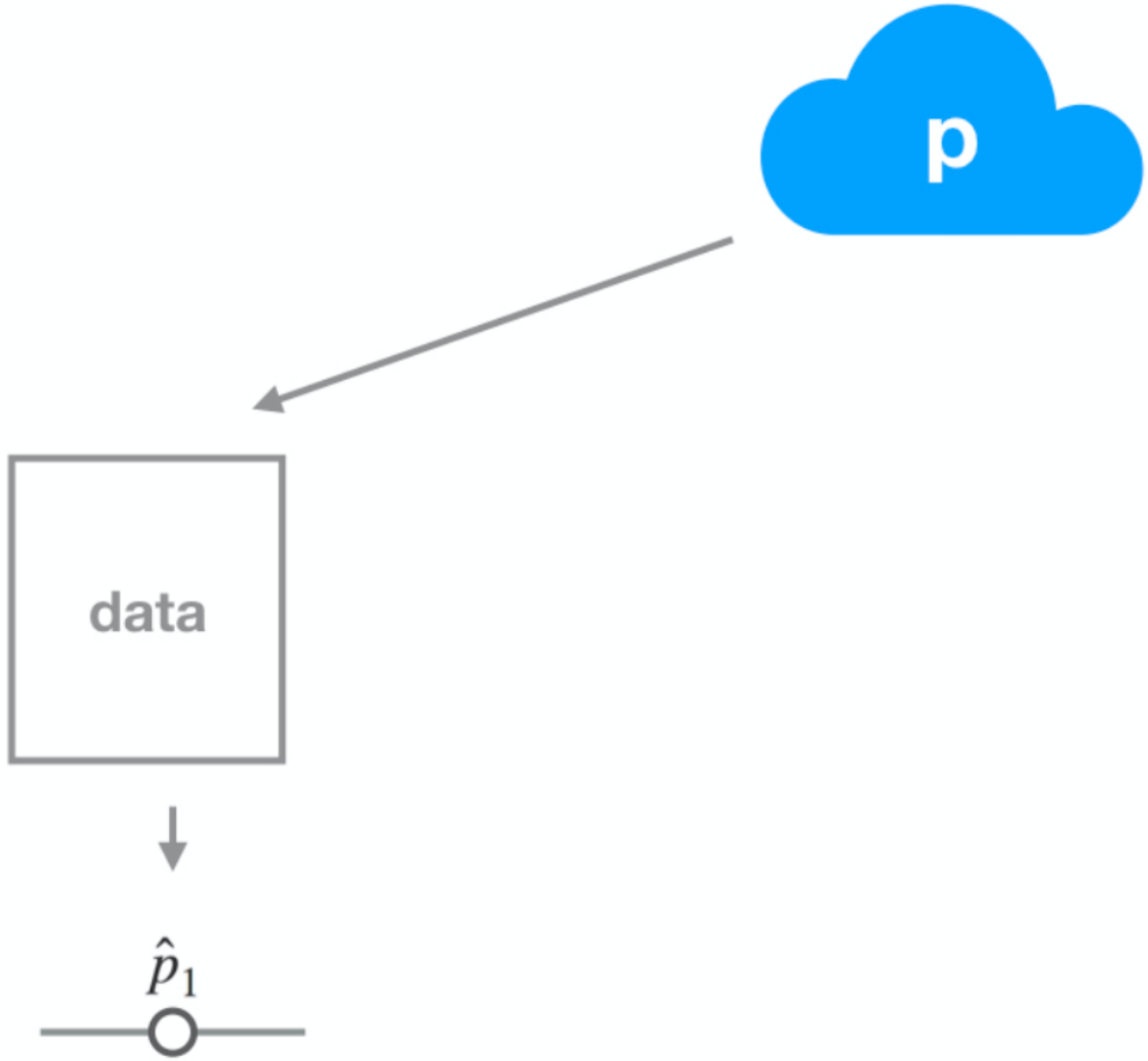
```
ds1 <- filter(gss, year == 2016)
p_hat <- ds1 %>%
  summarize(mean(happy == "HAPPY")) %>%
  pull()
SE <- ds1 %>%
  specify(response = happy,
          success = "HAPPY") %>%
  generate(reps = 500,
           type = "bootstrap") %>%
  calculate(stat = "prop") %>%
  summarize(sd(stat)) %>%
  pull()
c(p_hat - 2 * SE, p_hat + 2 * SE)
```

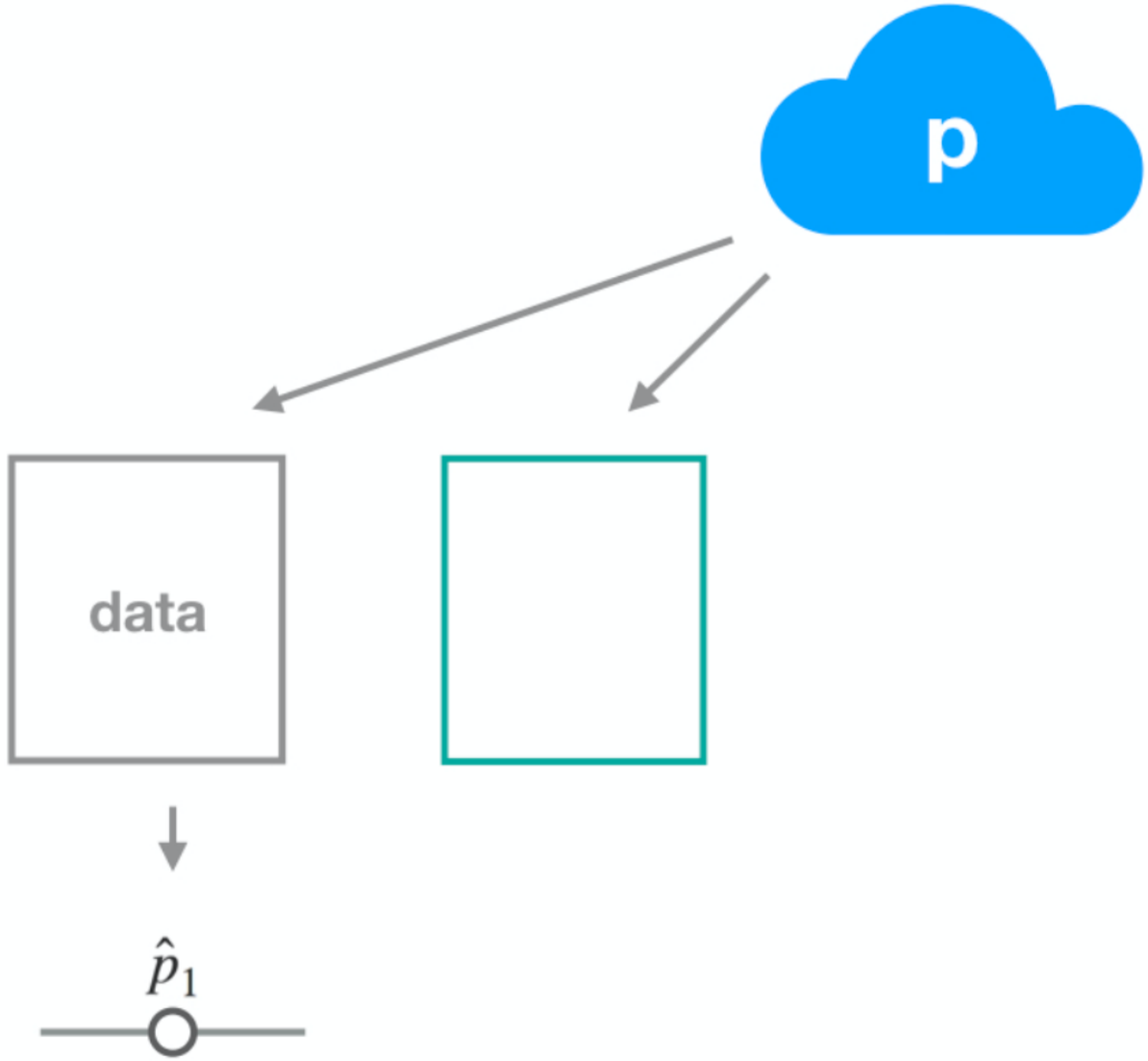
```
0.7073114 0.8393553
```

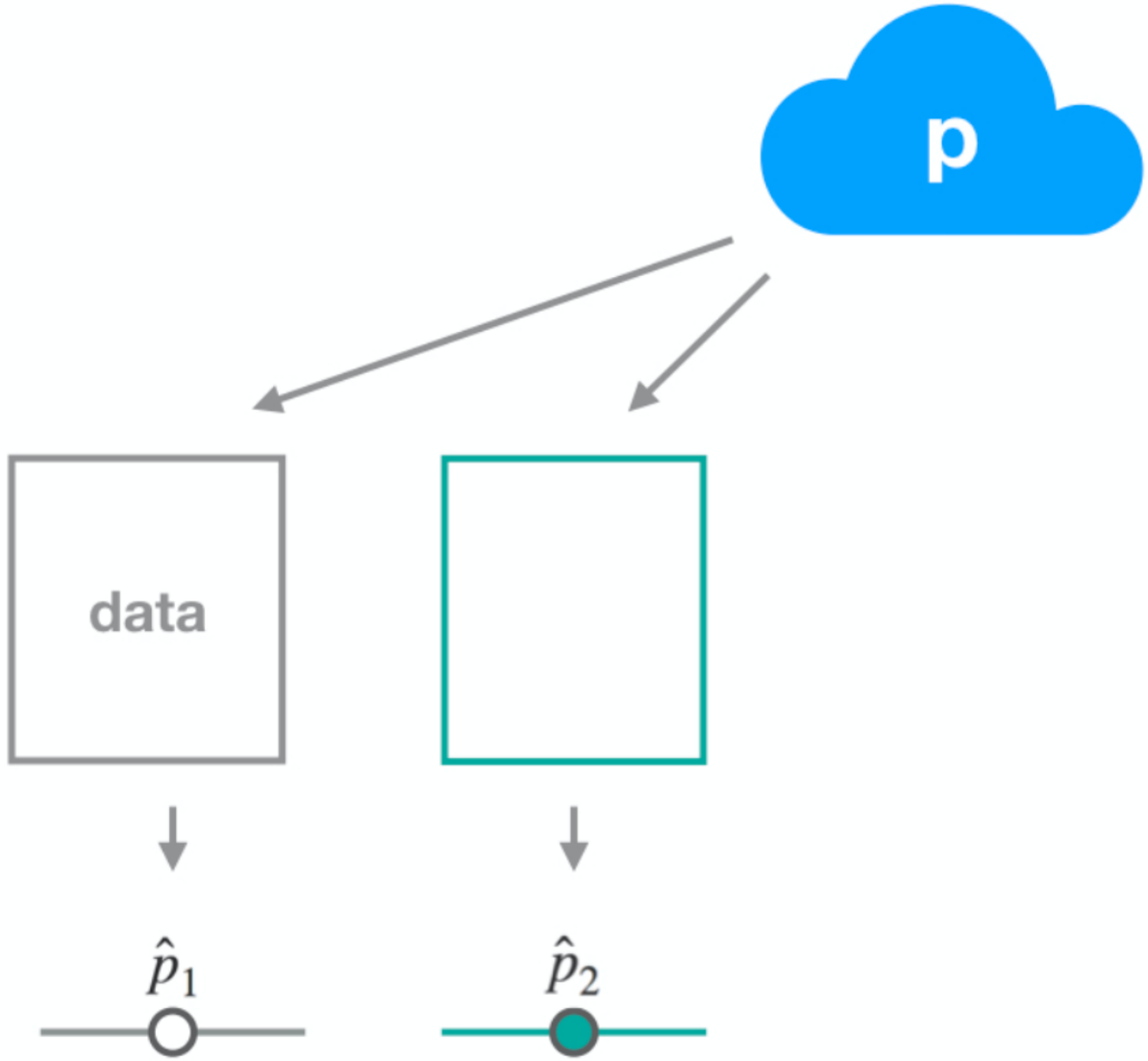

p

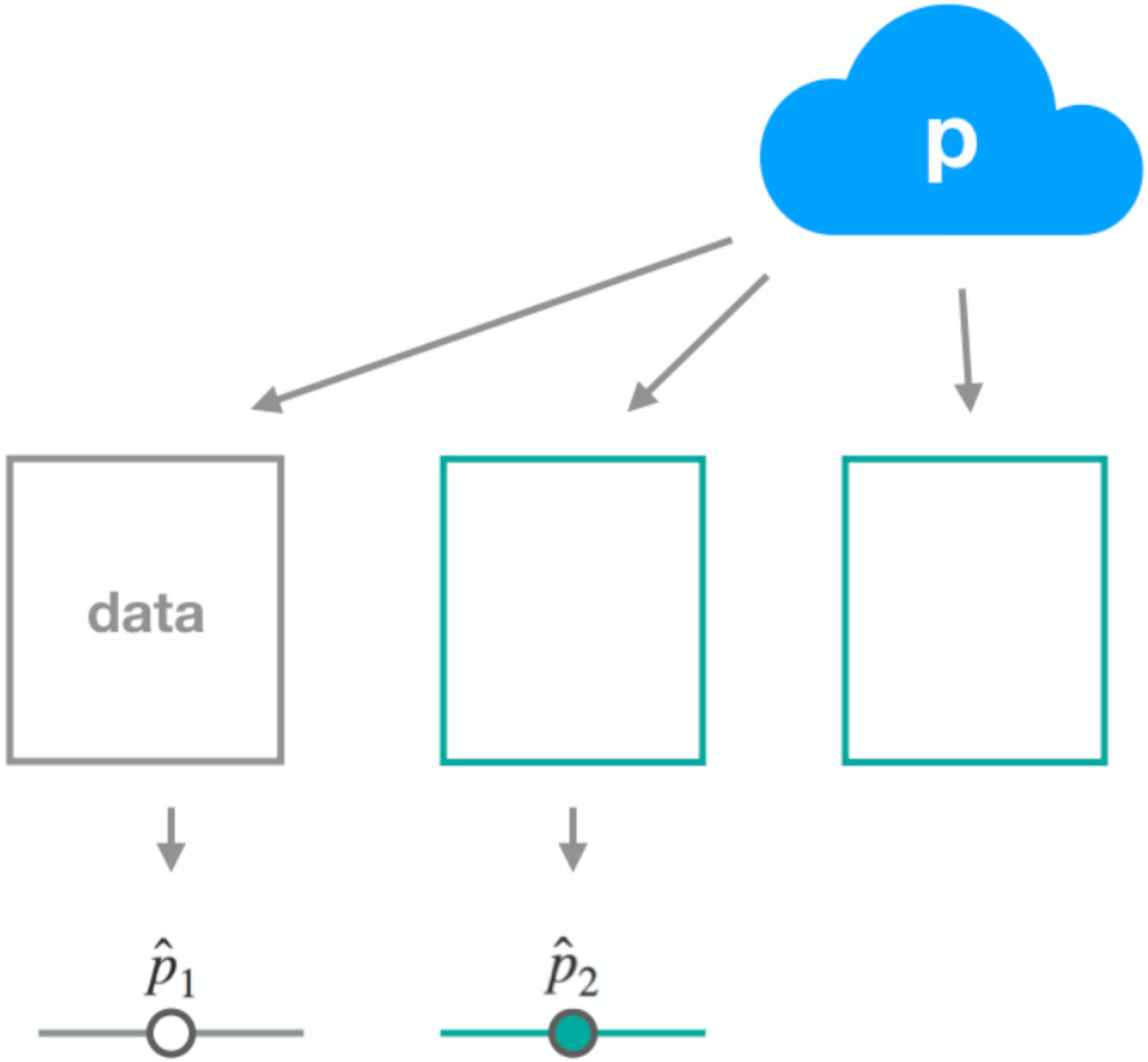

p
data

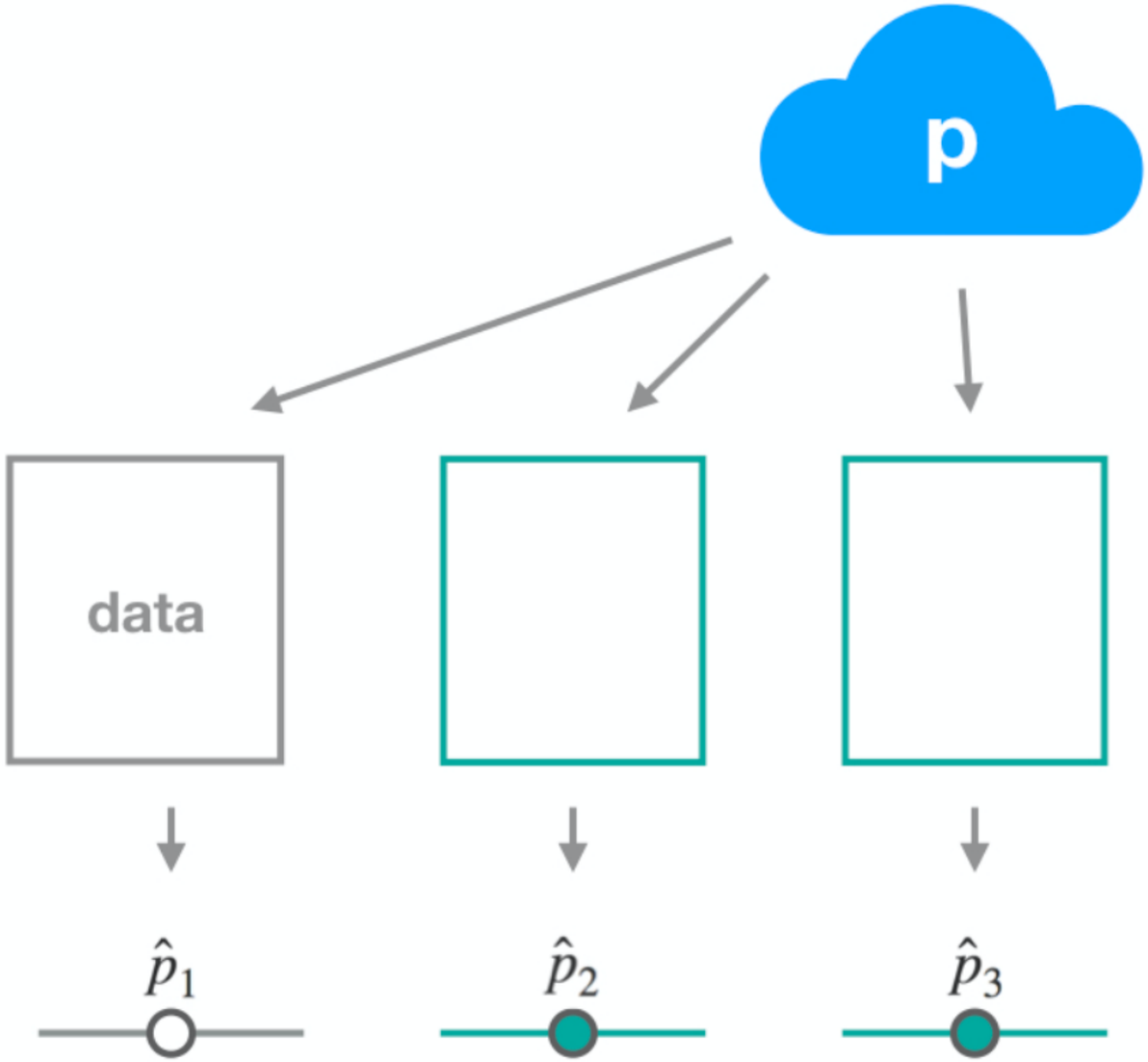

p
data
$\hat{p}_1$

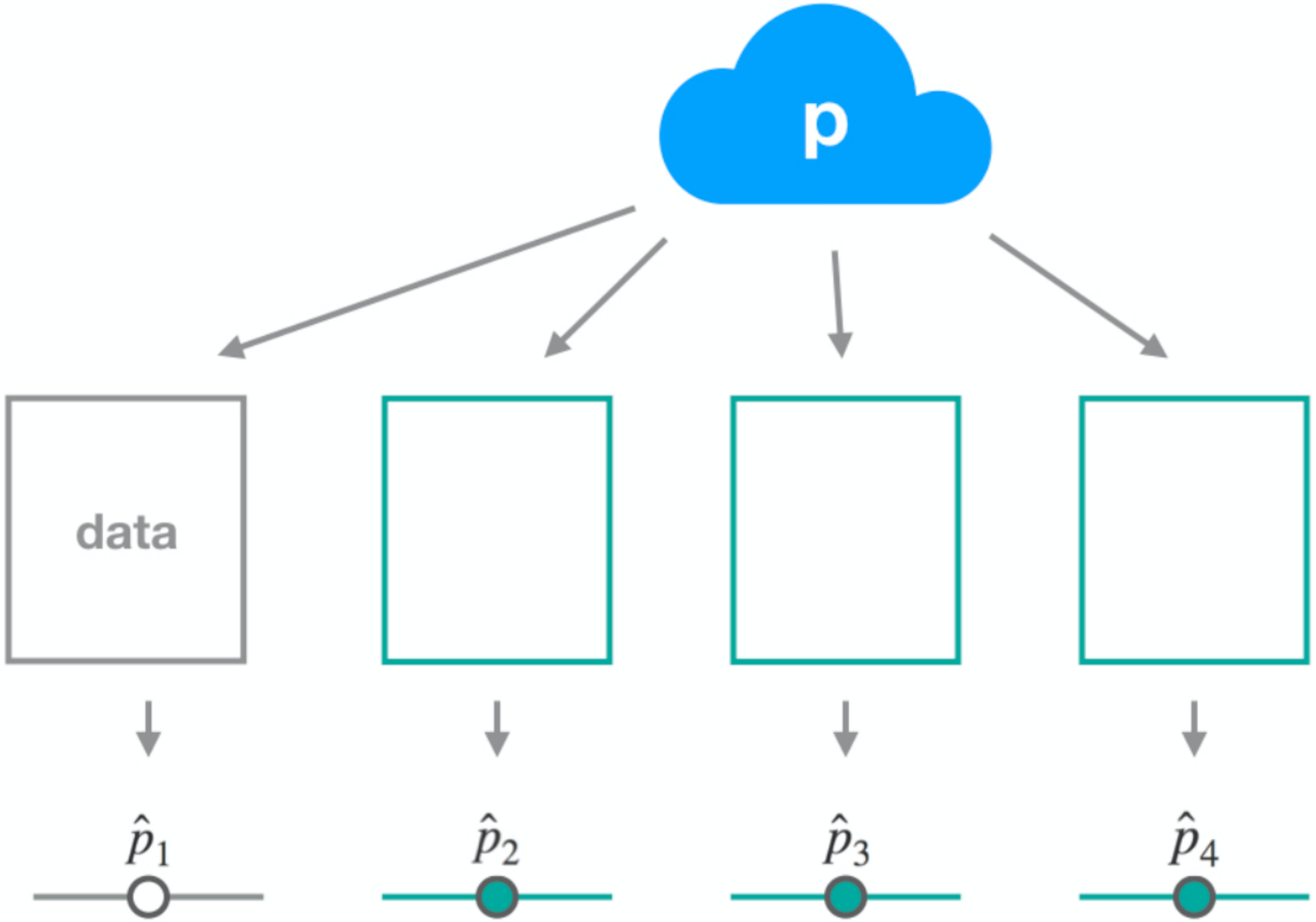

p
data
$\hat{p}_1$

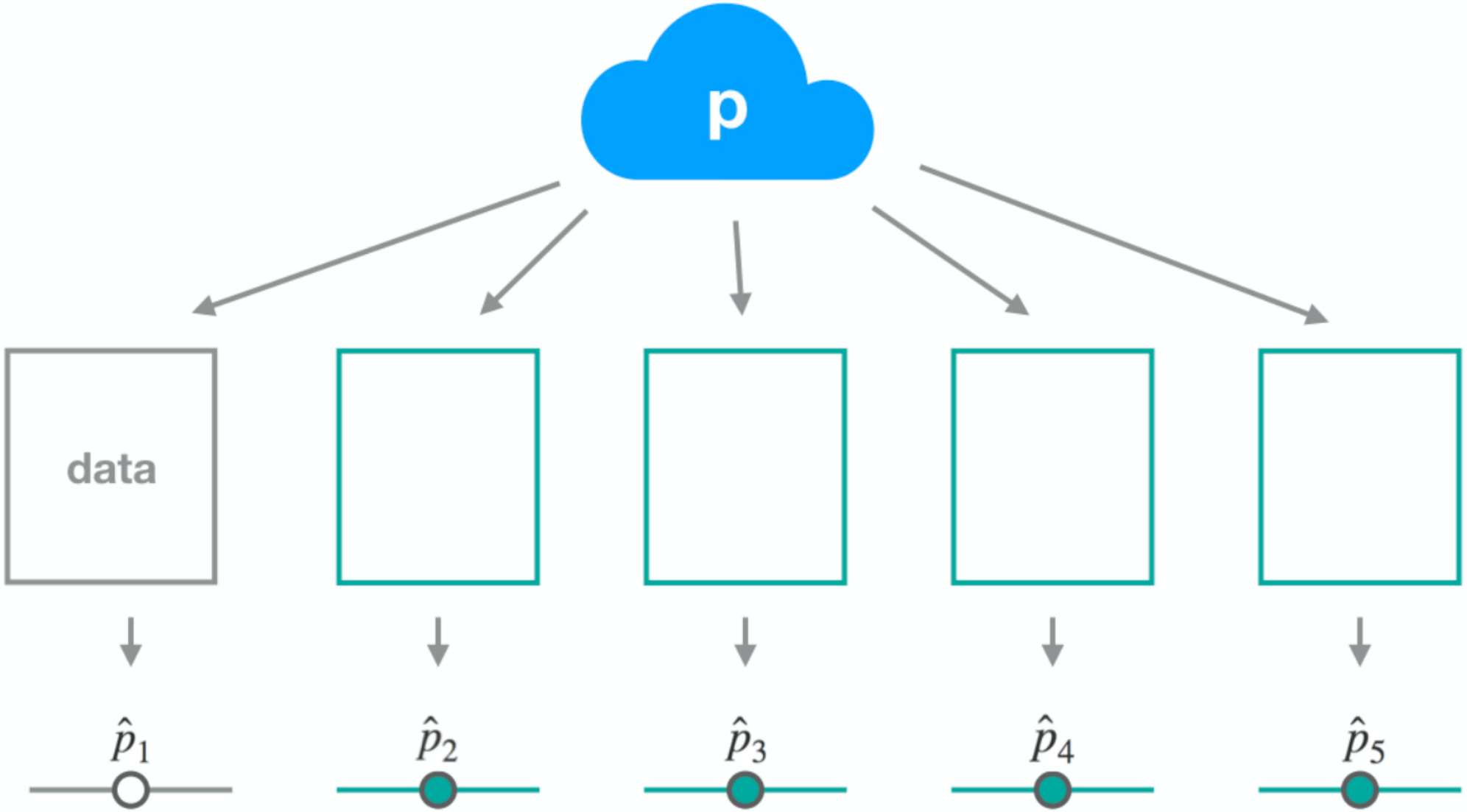

p
data
$\hat{p}_1$


p
data
$\hat{p}_1$
$\hat{p}_2$


p
data
$\hat{p}_1$
$\hat{p}_2$


p
data
$\hat{p}_1$
$\hat{p}_2$
$\hat{p}_3$

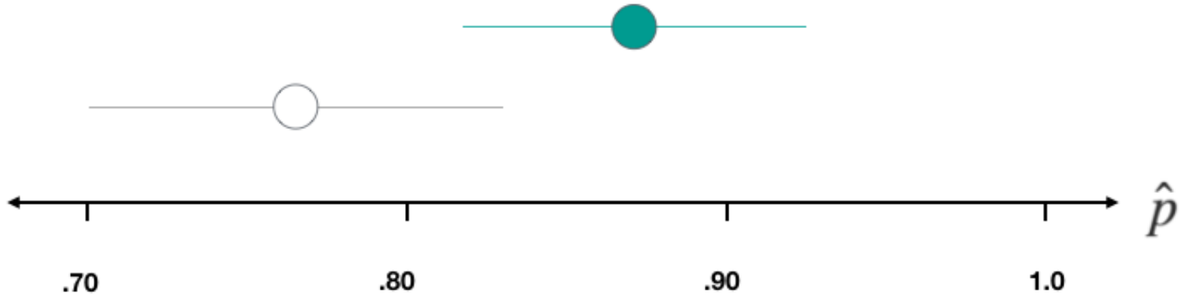

p
data
$\hat{p}_1$
$\hat{p}_2$
$\hat{p}_3$
$\hat{p}_4$


p
data
$\hat{p}_1$
$\hat{p}_2$
$\hat{p}_3$
$\hat{p}_4$
$\hat{p}_5$

Dataset 2

```
ds2 <- filter(gss, year == 2014)
p_hat <- ds1 %>%
  summarize(mean(happy == "HAPPY")) %>%
  pull()
SE <- ds1 %>%
  specify(response = happy,
          success = "HAPPY") %>%
  generate(reps = 500,
           type = "bootstrap") %>%
  calculate(stat = "prop") %>%
  summarize(sd(stat)) %>%
  pull()
c(p_hat - 2 * SE, p_hat + 2 * SE)
```

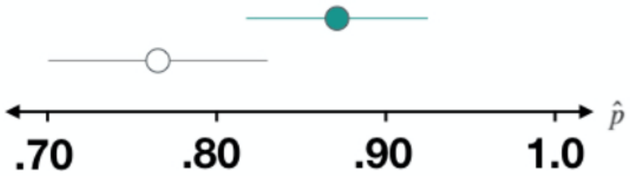
```
0.8348831 0.9384503
```



Dataset 3

```
ds3 <- filter(gss, year == 2012)
p_hat <- ds1 %>%
  summarize(mean(happy == "HAPPY")) %>%
  pull()
SE <- ds1 %>%
  specify(response = happy,
          success = "HAPPY") %>%
  generate(reps = 500,
           type = "bootstrap") %>%
  calculate(stat = "prop") %>%
  summarize(sd(stat)) %>%
  pull()
c(p_hat - 2 * SE, p_hat + 2 * SE)
```

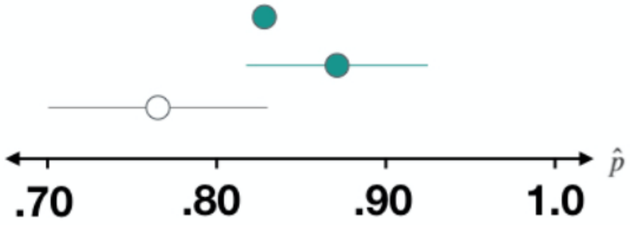
```
0.7626359 0.8906974
```



Dataset 3

```
ds3 <- filter(gss, year == 2012)
p_hat <- ds3 %>%
  summarize(mean(happy == "HAPPY")) %>%
  pull()
SE <- ds3 %>%
  specify(response = happy,
          success = "HAPPY") %>%
  generate(reps = 500,
           type = "bootstrap") %>%
  calculate(stat = "prop") %>%
  summarize(sd(stat)) %>%
  pull()
c(p_hat - 2 * SE, p_hat + 2 * SE)
```

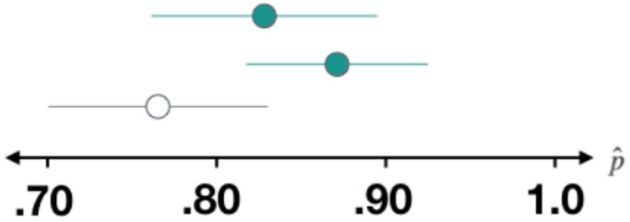
```
0.7626359 0.8906974
```



Dataset 3

```
ds3 <- filter(gss, year == 2012)
p_hat <- ds3 %>%
  summarize(mean(happy == "HAPPY")) %>%
  pull()
SE <- ds3 %>%
  specify(response = happy,
          success = "HAPPY") %>%
  generate(reps = 500,
           type = "bootstrap") %>%
  calculate(stat = "prop") %>%
  summarize(sd(stat)) %>%
  pull()
c(p_hat - 2 * SE, p_hat + 2 * SE)
```

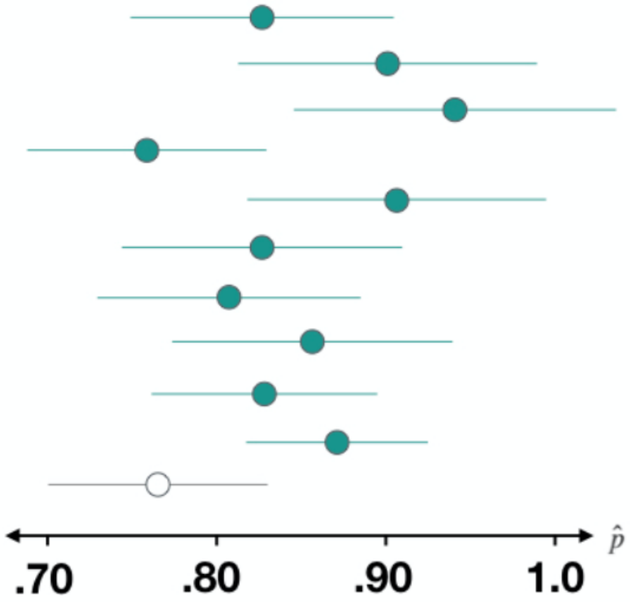
```
0.7626359 0.8906974
```



Dataset 3

```
ds3 <- filter(gss, year == 2012)
p_hat <- ds3 %>%
  summarize(mean(happy == "HAPPY")) %>%
  pull()
SE <- ds3 %>%
  specify(response = happy,
          success = "HAPPY") %>%
  generate(reps = 500,
           type = "bootstrap") %>%
  calculate(stat = "prop") %>%
  summarize(sd(stat)) %>%
  pull()
c(p_hat - 2 * SE, p_hat + 2 * SE)
```

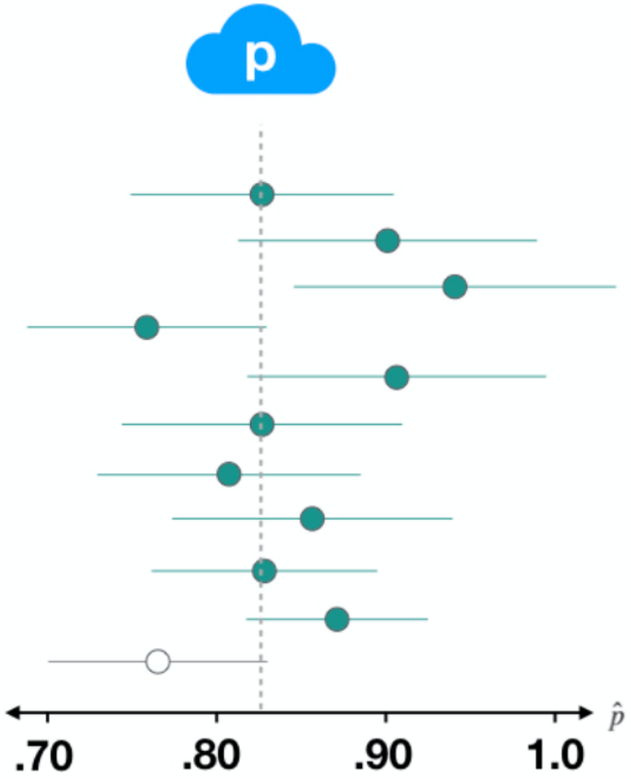
```
0.7626359 0.8906974
```



Dataset 3

```
ds3 <- filter(gss, year == 2012)
p_hat <- ds3 %>%
  summarize(mean(happy == "HAPPY")) %>%
  pull()
SE <- ds3 %>%
  specify(response = happy,
          success = "HAPPY") %>%
  generate(reps = 500,
           type = "bootstrap") %>%
  calculate(stat = "prop") %>%
  summarize(sd(stat)) %>%
  pull()
c(p_hat - 2 * SE, p_hat + 2 * SE)
```

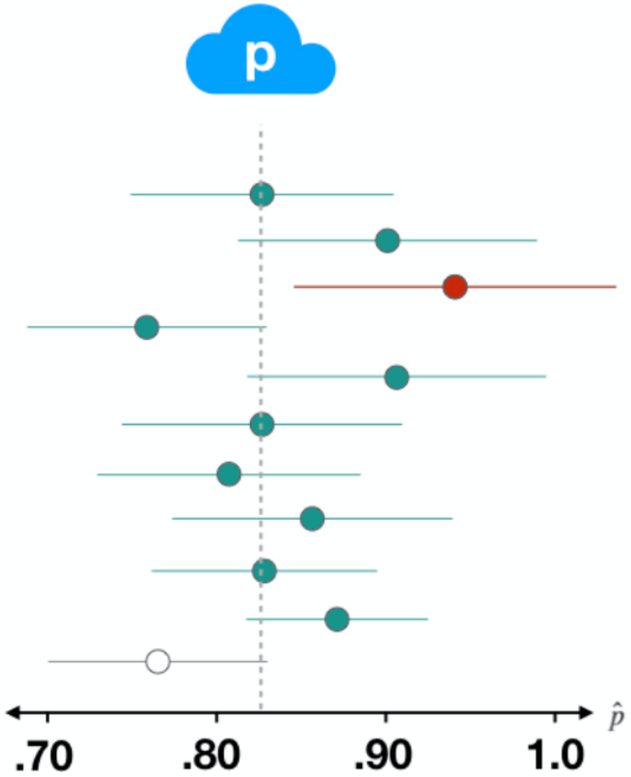
```
0.7626359 0.8906974
```



Dataset 3

```
ds3 <- filter(gss, year == 2012)
p_hat <- ds3 %>%
  summarize(mean(happy == "HAPPY")) %>%
  pull()
SE <- ds3 %>%
  specify(response = happy,
          success = "HAPPY") %>%
  generate(reps = 500,
           type = "bootstrap") %>%
  calculate(stat = "prop") %>%
  summarize(sd(stat)) %>%
  pull()
c(p_hat - 2 * SE, p_hat + 2 * SE)
```

```
0.7626359 0.8906974
```



Confidence Intervals

Interpretation: “We’re 95% confident that the true proportion of Americans that are happy is between 0.705 and 0.841.”

Width of the interval affected by

- `n`
- confidence level
- `p`

Let's practice!

INFERENCE FOR CATEGORICAL DATA IN R

The approximation shortcut

INFERENCE FOR CATEGORICAL DATA IN R

Andrew Bray

Assistant Professor of Statistics at Reed College

Confidence Intervals

```
SE
```

```
0.009998905
```

```
SE_small_n
```

```
0.03809731
```

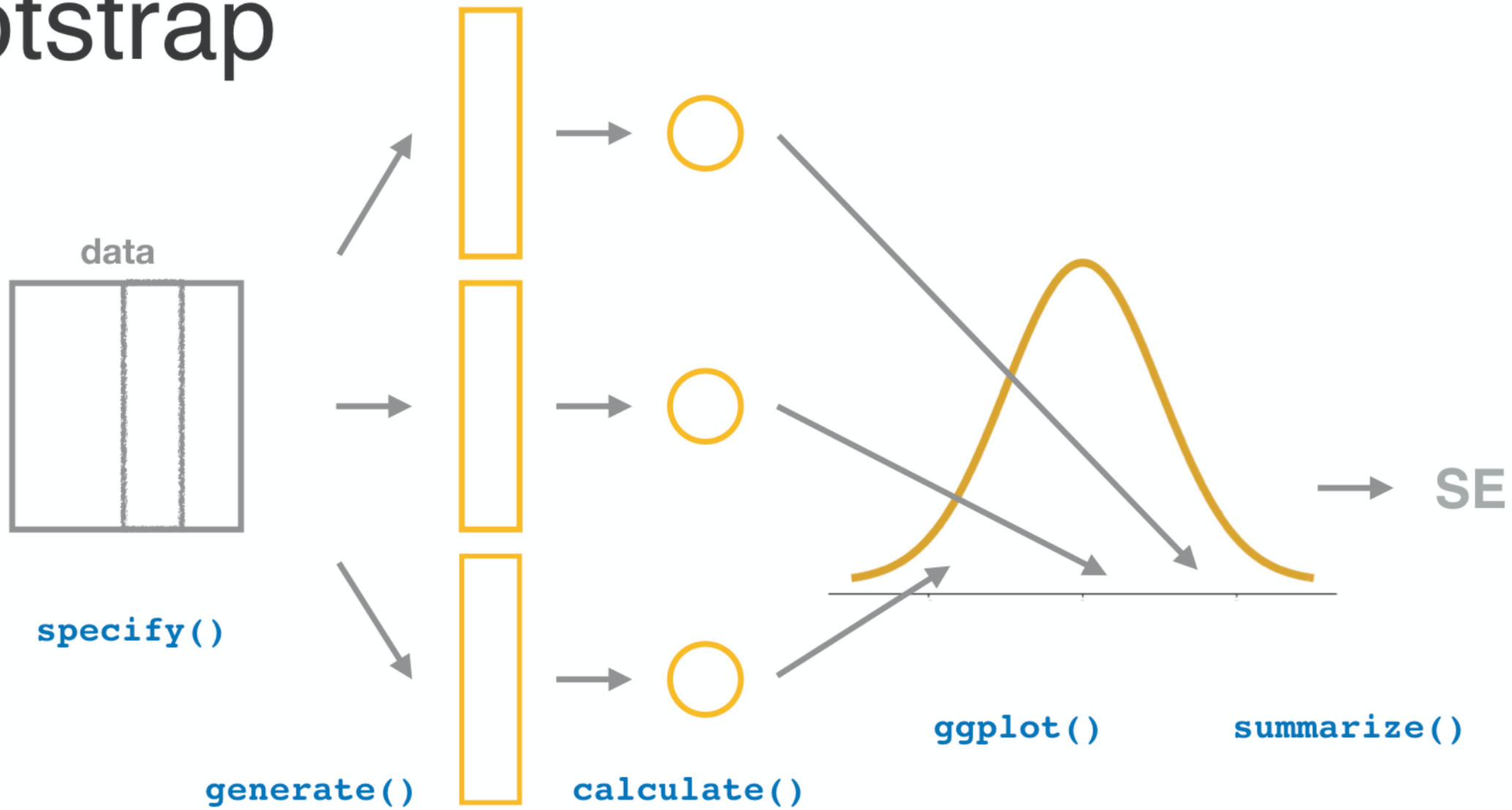
```
SE_low_p
```

```
0.00547912
```

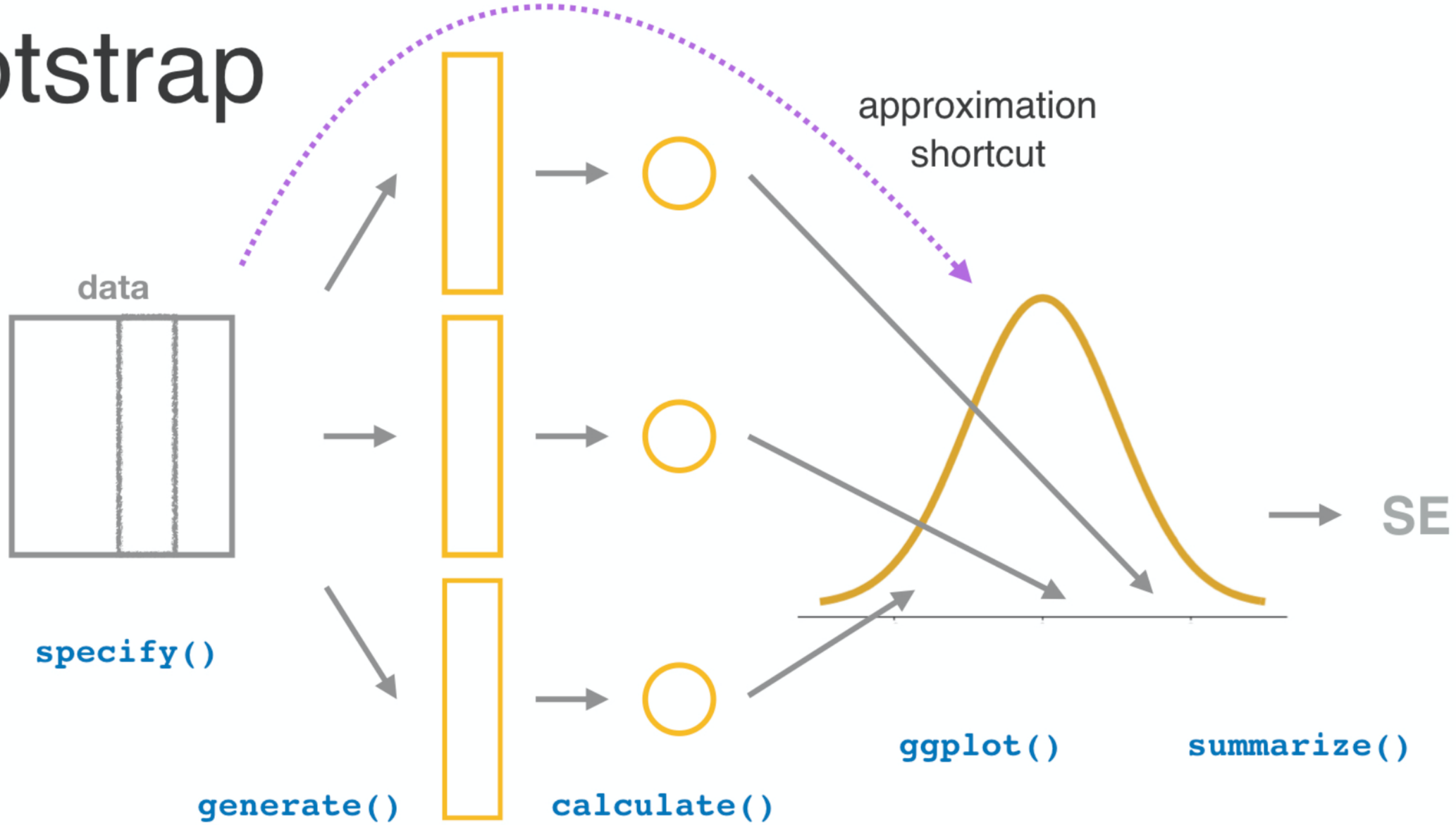
Standard errors increase when

- `n` is small
- `p` is close to 0.5

Bootstrap



Bootstrap


approximation
shortcut
data
specify()
generate()
calculate()
ggplot()
summarize()
SE

The normal distribution

A.K.A the "bell curve".

If

- observations are independent
- `n` is large

Then

- $\hat{p}$ follows a normal distribution

Standard deviation

$$\sqrt{\frac{\hat{p} \times (1 - \hat{p})}{n}}$$

Assessing model assumptions

How do I check "observations are independent"?

- This depends upon the data collection method.

What does "n is large" mean?

- $n \times \hat{p} > 10$
- $n \times (1 - \hat{p}) > 10$

Calculating standard error: approximation

```
p_hat <- gss2016 %>%
  summarize(mean(happy == "HAPPY")) %>%
  pull()
n <- nrow(gss2016)
```

```
c(n * p_hat, n * (1 - p_hat))
```

```
116  35
```

```
SE_approx <- sqrt(p_hat * (1 - p_hat) / n)
SE_approx
```

```
0.03418468
```

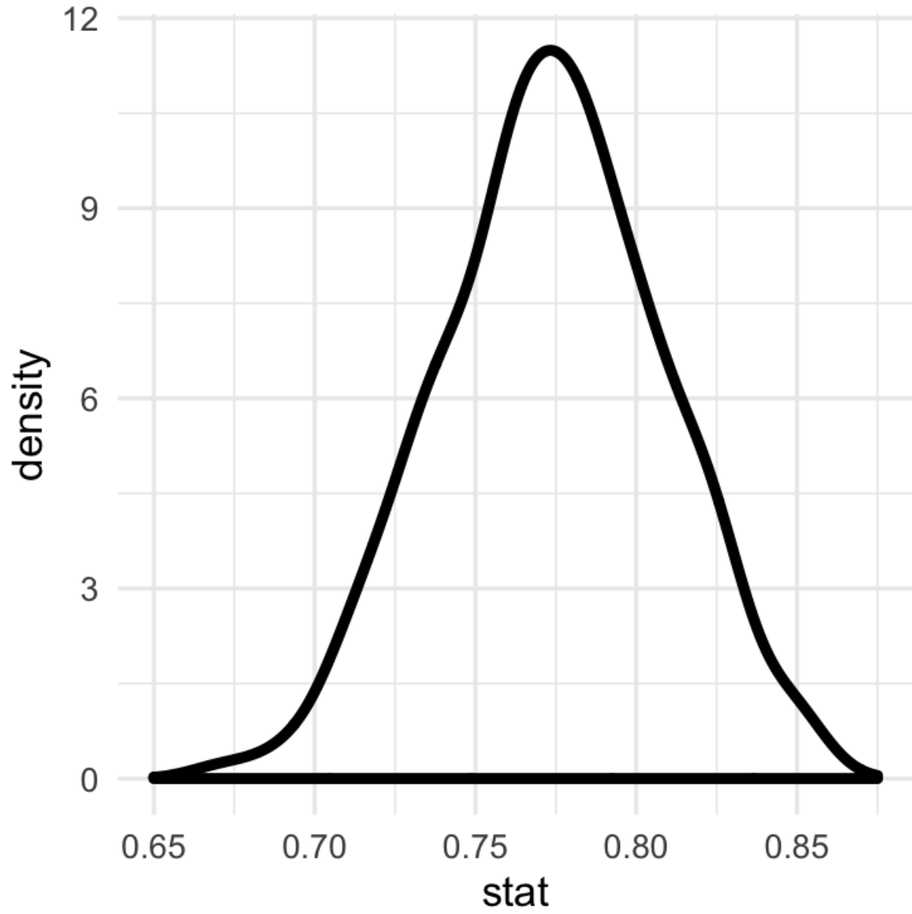
Calculating standard error: computation

```
boot <- gss2016 %>%
  specify(response = happy, success = "HAPPY") %>%
  generate(reps = 500, type = "bootstrap") %>%
  calculate(stat = "prop")
SE_boot <- boot %>%
  summarize(sd(stat)) %>%
  pull()
SE_boot
```

```
0.03176741
```

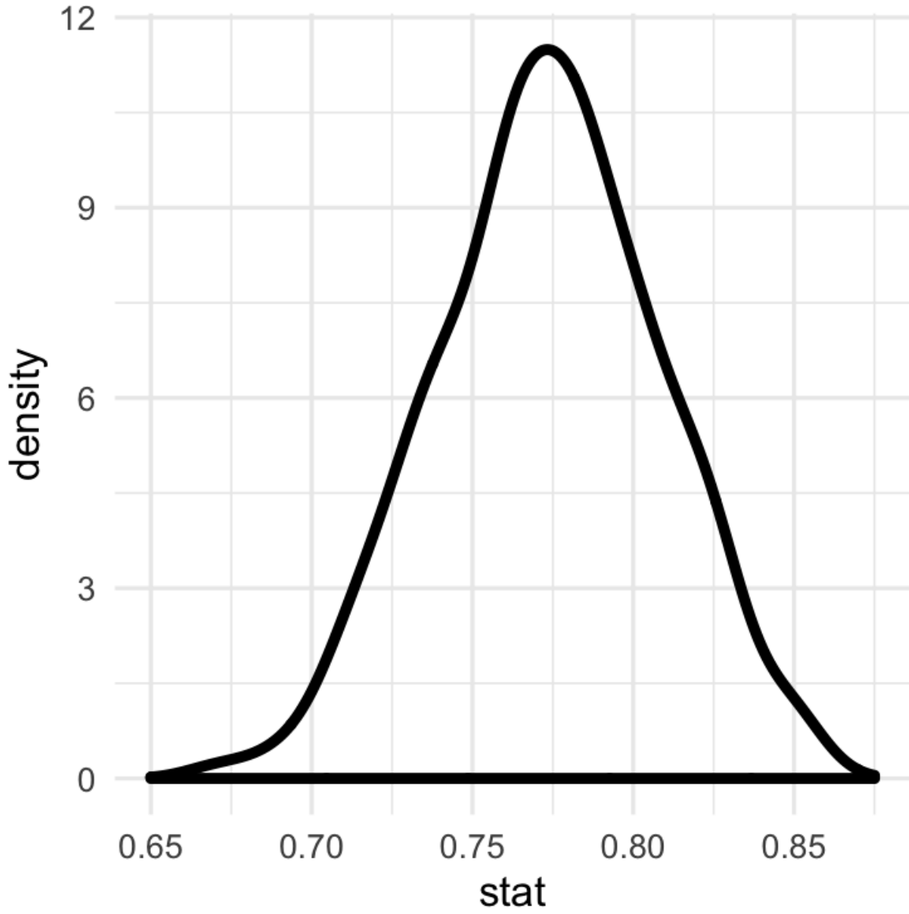
Sampling distributions

```
ggplot(boot, aes(x = stat)) +
  geom_density()
```



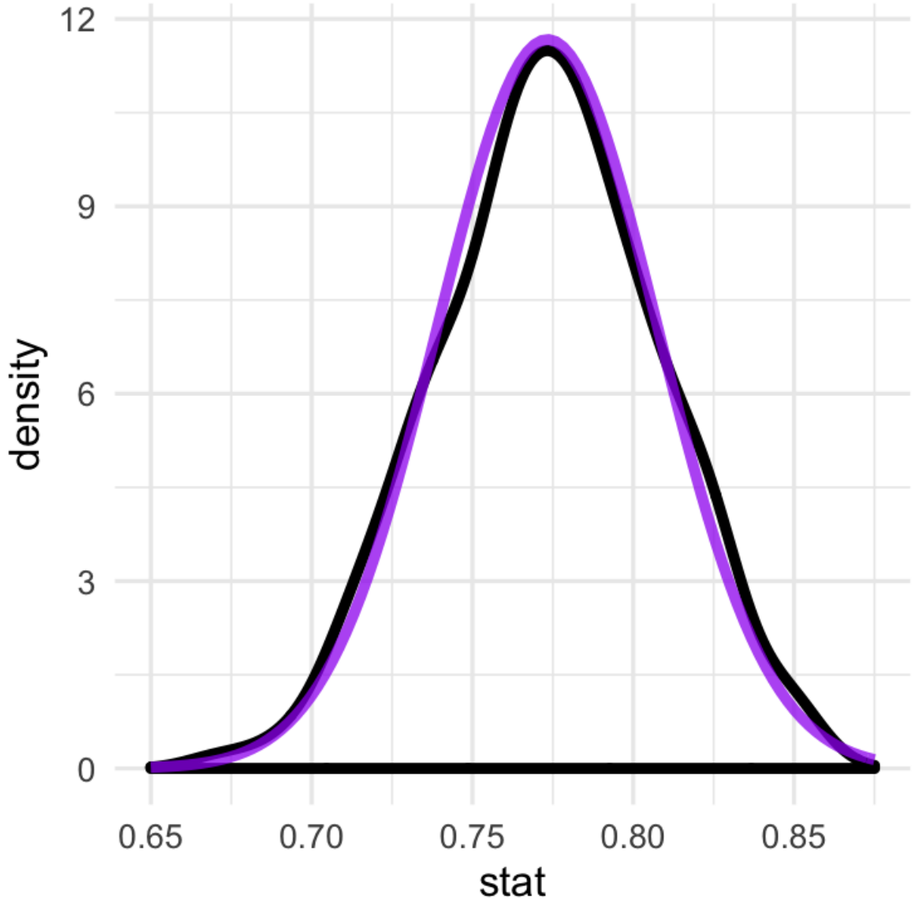
Sampling distributions

```
ggplot(boot, aes(x = stat)) +
  geom_density() +
  stat_function(fun = dnorm,
                color = "purple",
                args =
                  list(mean = p_hat,
                       sd = SE_approx))
```



Sampling distributions

```
ggplot(boot, aes(x = stat)) +
  geom_density() +
  stat_function(fun = dnorm,
                color = "purple",
                args =
                  list(mean = p_hat,
                       sd = SE_approx))
```



Let's practice!

INFERENCE FOR CATEGORICAL DATA IN R