Histograms

a problem

- You’re managing the HKN lounge for next semester
- How much coffee should you buy for each day?
    - Too much → waste money 😟
    - Too little → under-caffeinated students 😟
- What should you do?

collect data

- Count how many people get coffee in a day
    - Day 1: 37 people
- Should we just get enough coffee for 37 people?

(keep) collect(ing) data

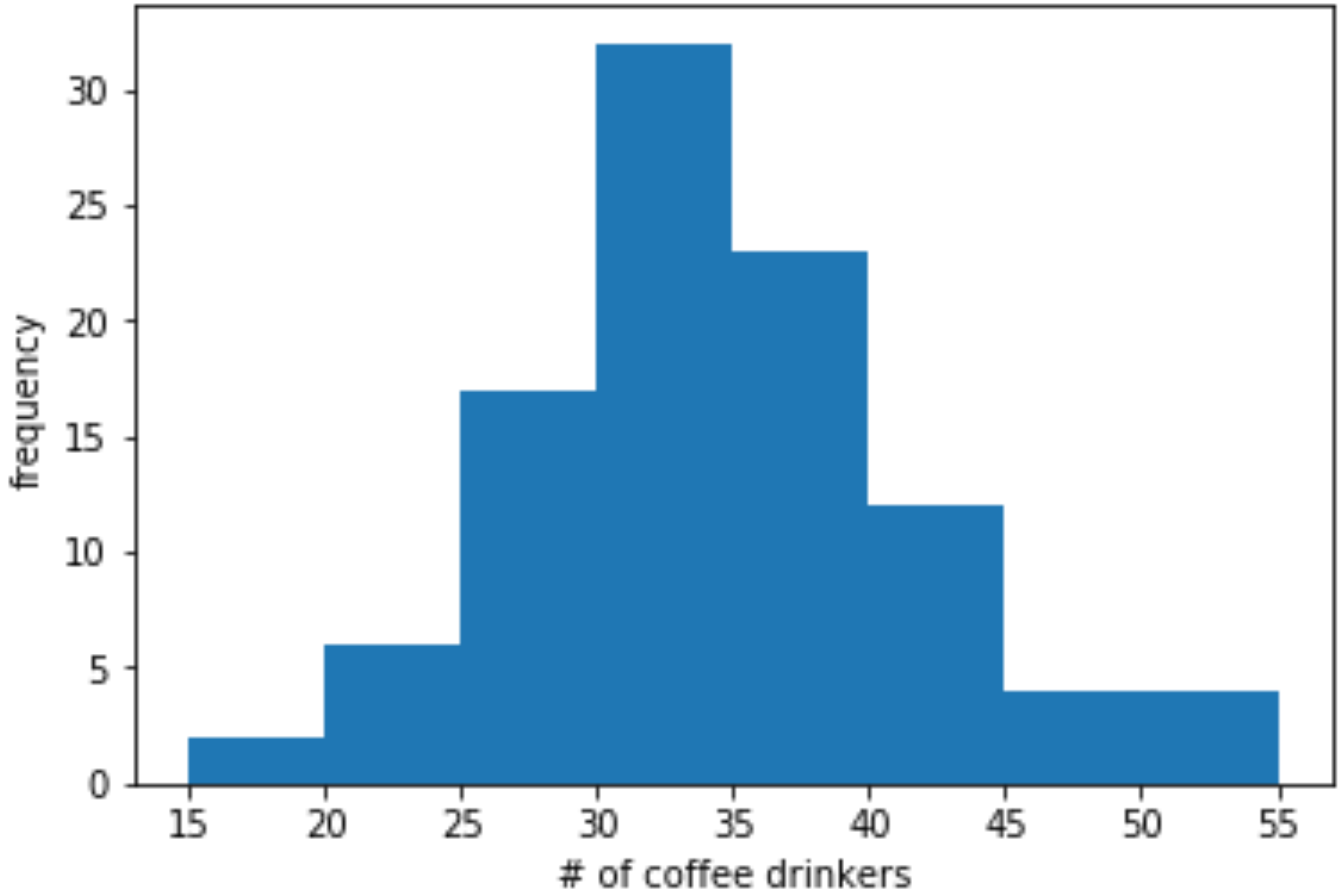
- Day 2: 43
- Day 3: 48
- Day 4: 41
- Day 5: 46
- Day 6: 19 (!)
- Day 7: 38
- …

100 days later ...

```
[37, 43, 48, 41, 46, 19, 28, 35, 34, 38,
31, 32, 32, 23, 23, 33, 35, 39, 34, 28,
39, 28, 29, 38, 28, 30, 25, 35, 39, 35,
31, 28, 25, 26, 15, 31, 28, 32, 40, 21,
34, 38, 30, 47, 34, 31, 51, 30, 41, 36,
33, 51, 22, 25, 29, 50, 32, 39, 25, 37,
54, 33, 36, 25, 30, 22, 41, 35, 31, 40,
30, 33, 27, 36, 27, 34, 24, 41, 37, 29,
48, 40, 31, 32, 33, 32, 40, 31, 32, 40,
31, 33, 32, 38, 37, 41, 37, 39, 38, 42]
```

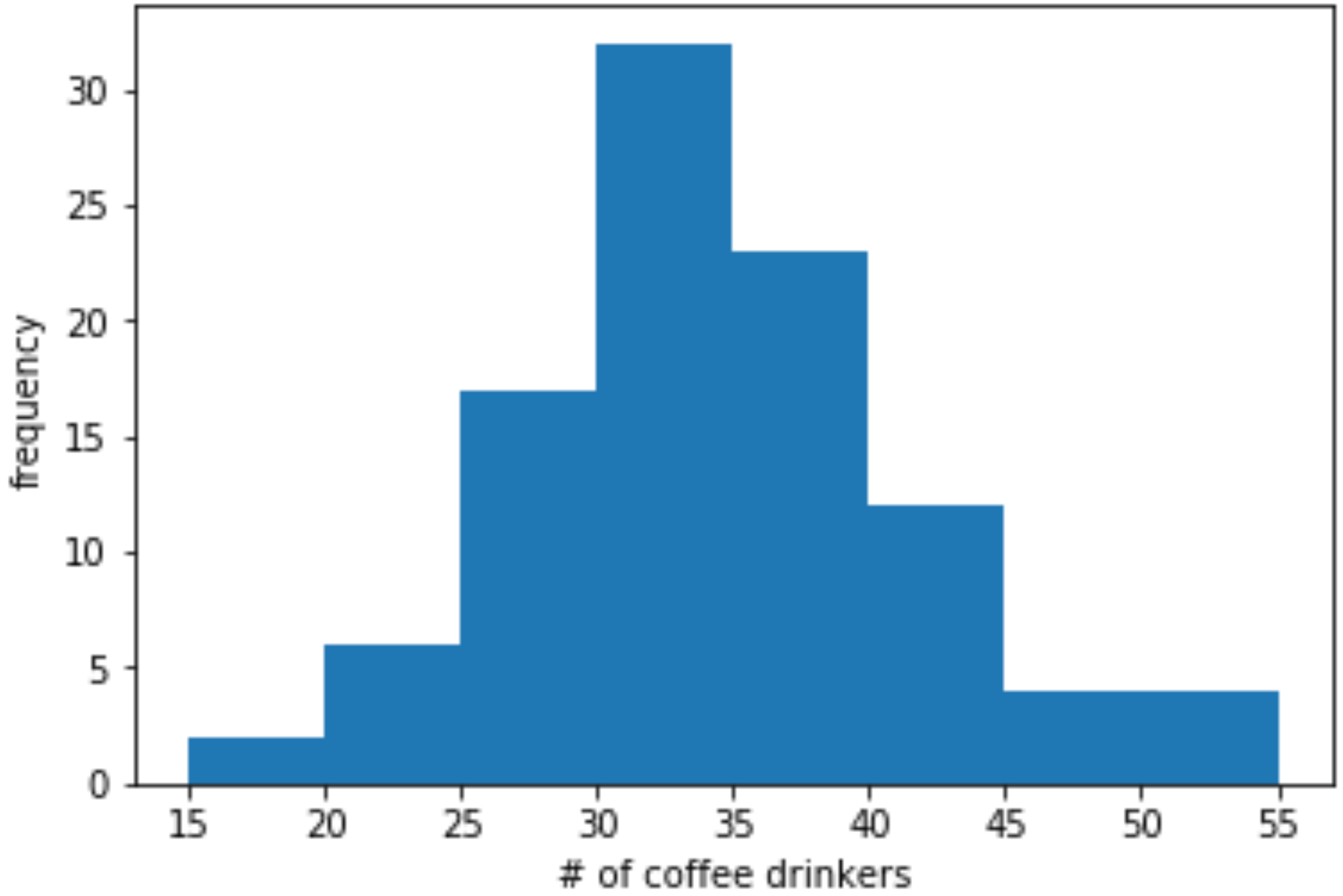
visualize the data

- Staring at a list of numbers is not very illuminating
- To identify patterns in data, we should *visualize* that data in a useful way
- Idea: a histogram!



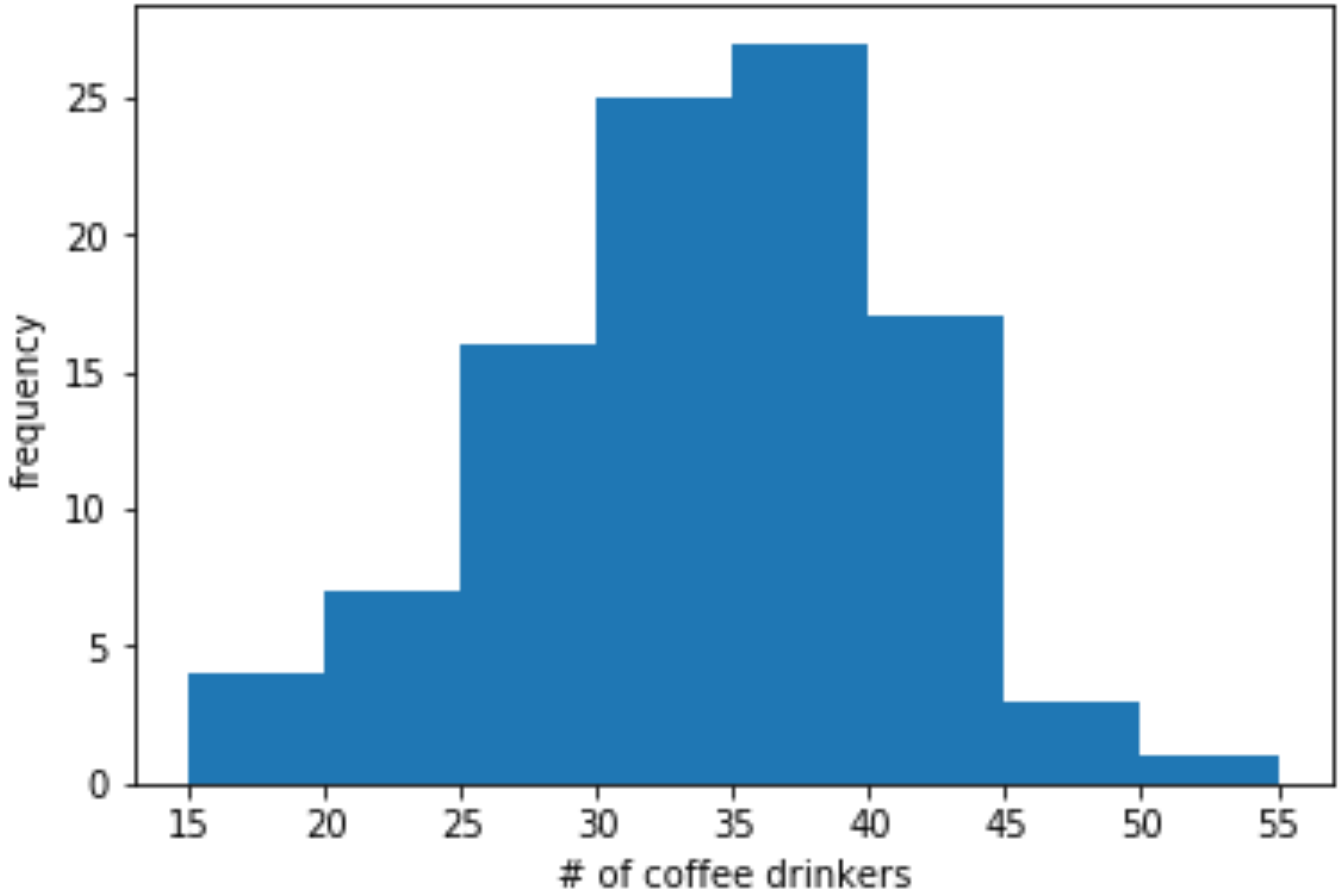
visualize the data

- Each bar in a histogram is a **bin**: $x_1, x_2, x_3, \ldots$
- Each observation goes into one bin: $x_1 : 15 \leq d < 20$
- The size of each bin is the number of observations in that bin
- If we divide each count by the number of observations, we get the empirical (measured) **frequency** of each bin: $\hat{p}_1, \hat{p}_2, \hat{p}_3, \ldots$
- Note: $\sum_k \hat{p}_k = 1$



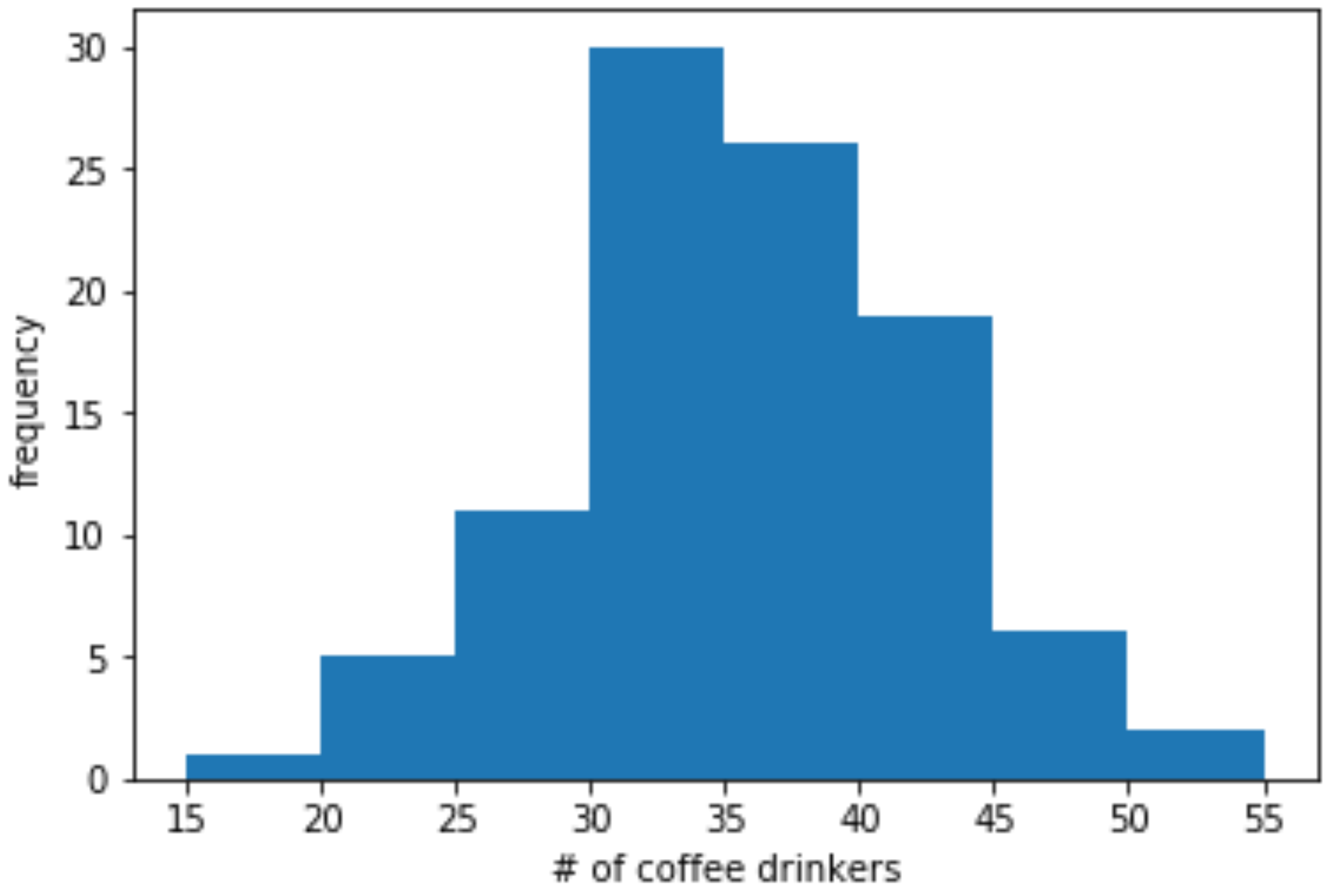
visualize the data

- Remember: this histogram comes from *observed* data
- If we repeat the experiment, we might get a different histogram!
- (This is because what we have is a *sample* of the data)



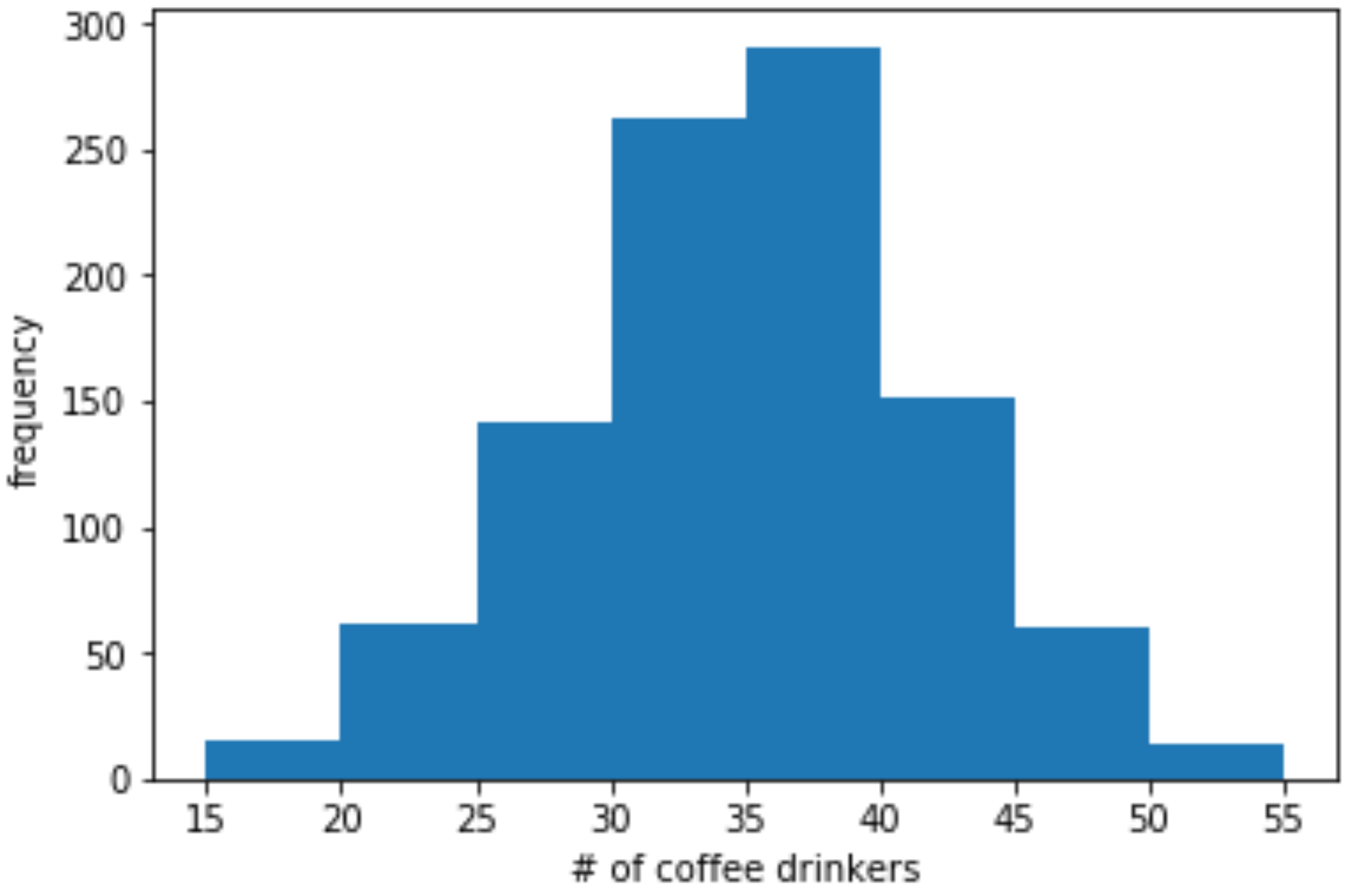
visualize the data

- Remember: this histogram comes from *observed* data
- If we repeat the experiment, we might get a different histogram!
- (This is because what we have is a *sample* of the data)



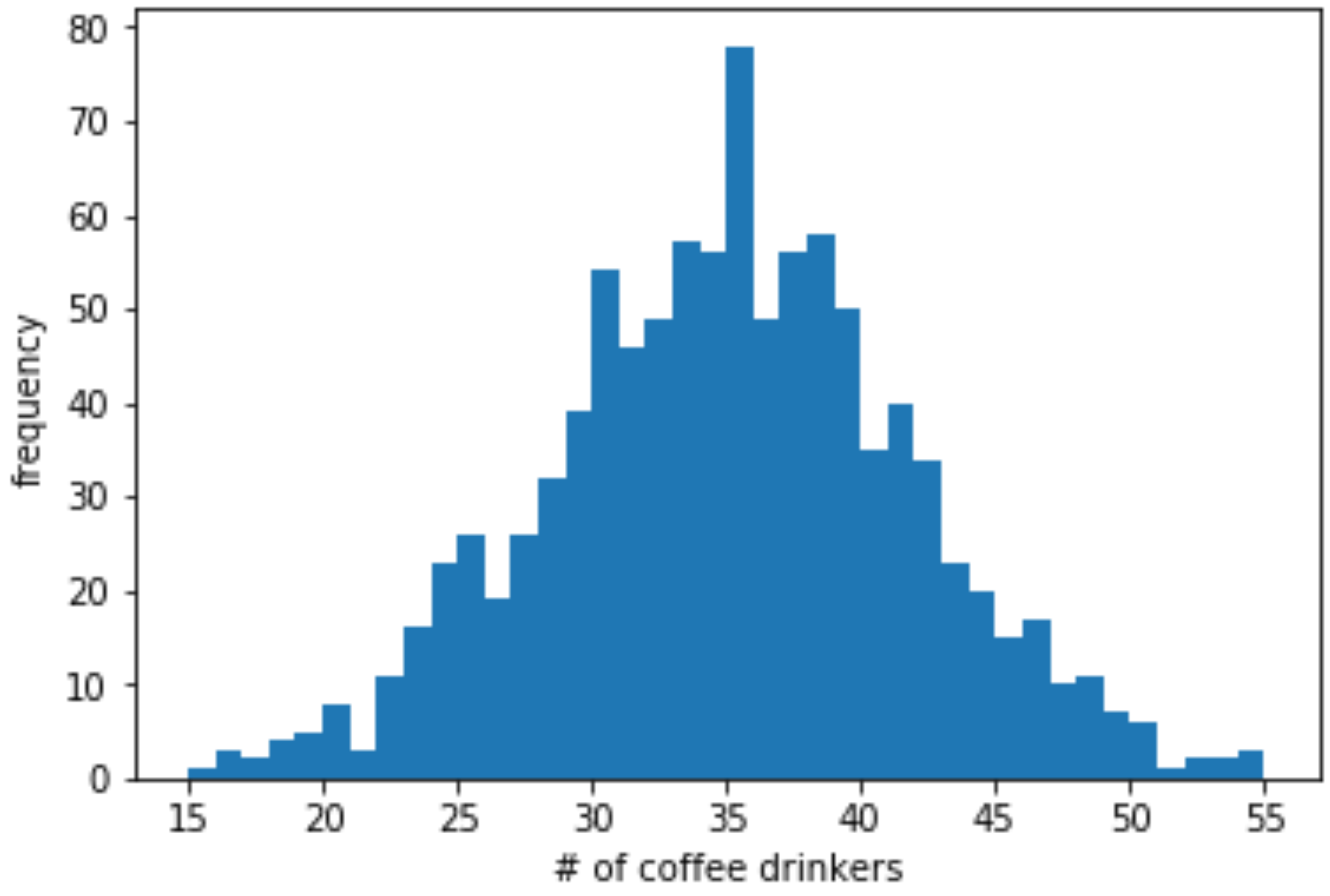
what if we collect more data?

- Hmm … this looks basically the same!
- Because we're using the same number of bins! Each bin has more observations in it, but *relative* to each other, each bin is basically the same
- The *frequencies* of the bins aren't changing much



so add bins!

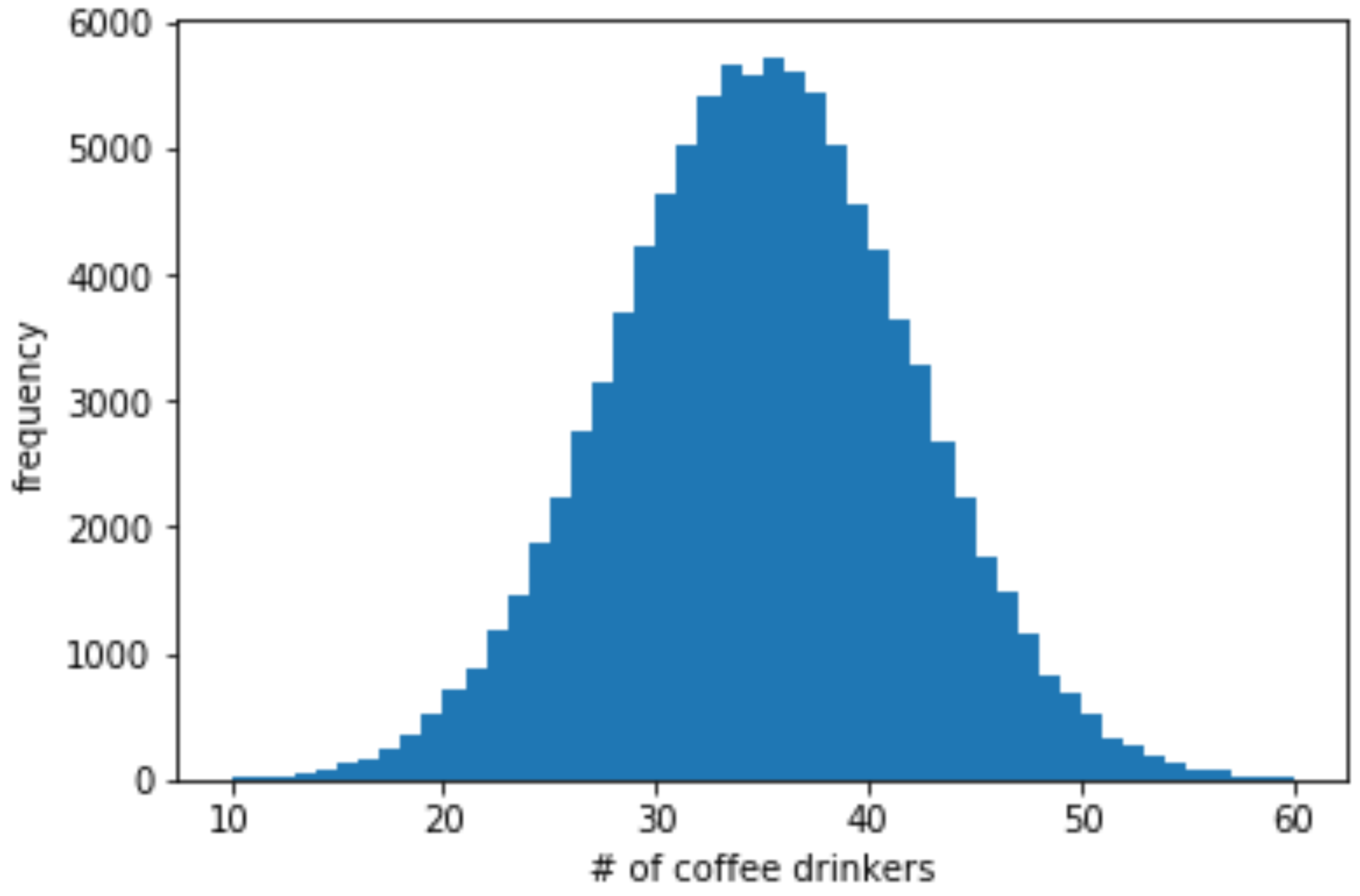
- This looks better!
- Gives us a good sense of what the data shape looks like



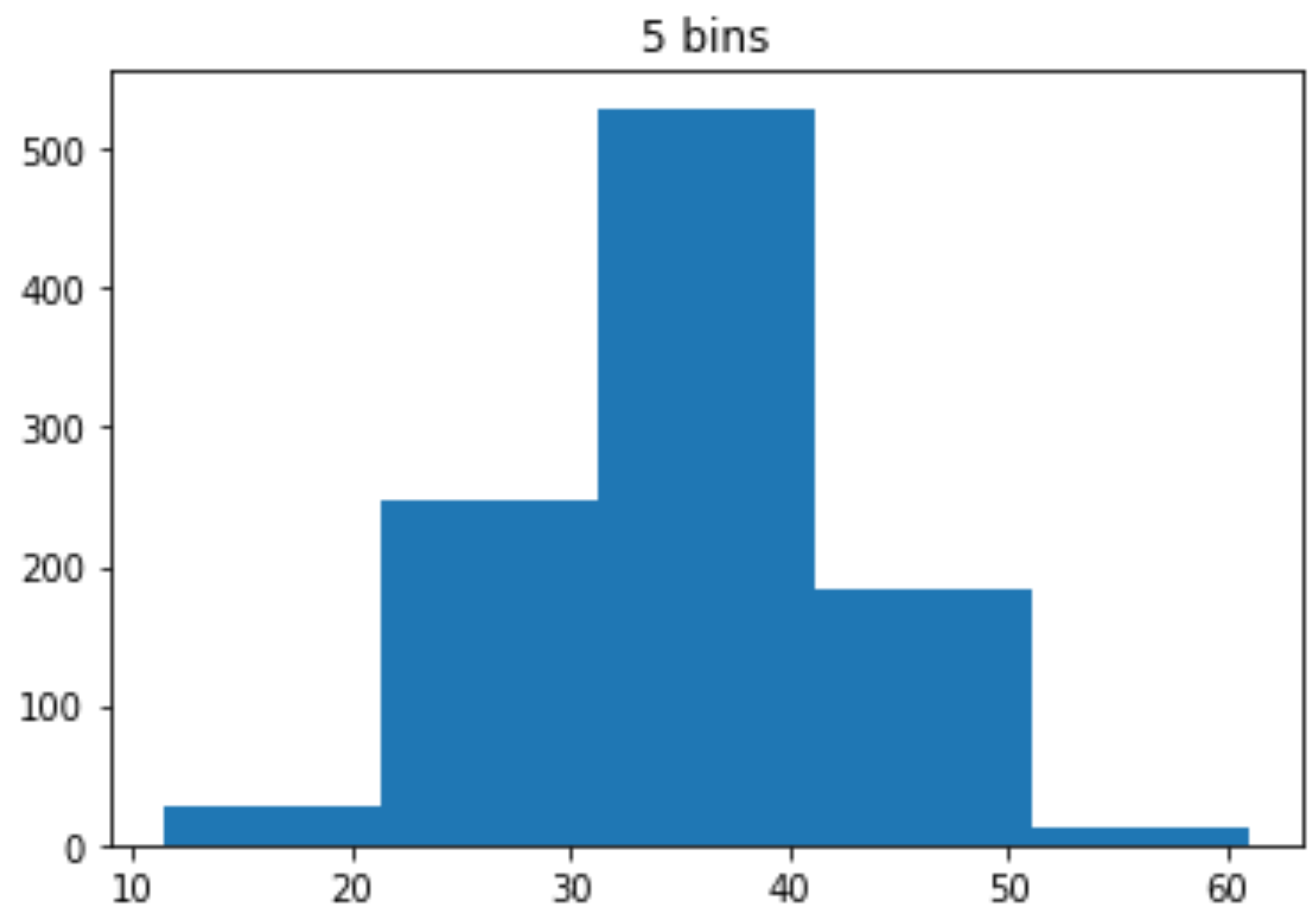
... and more data!

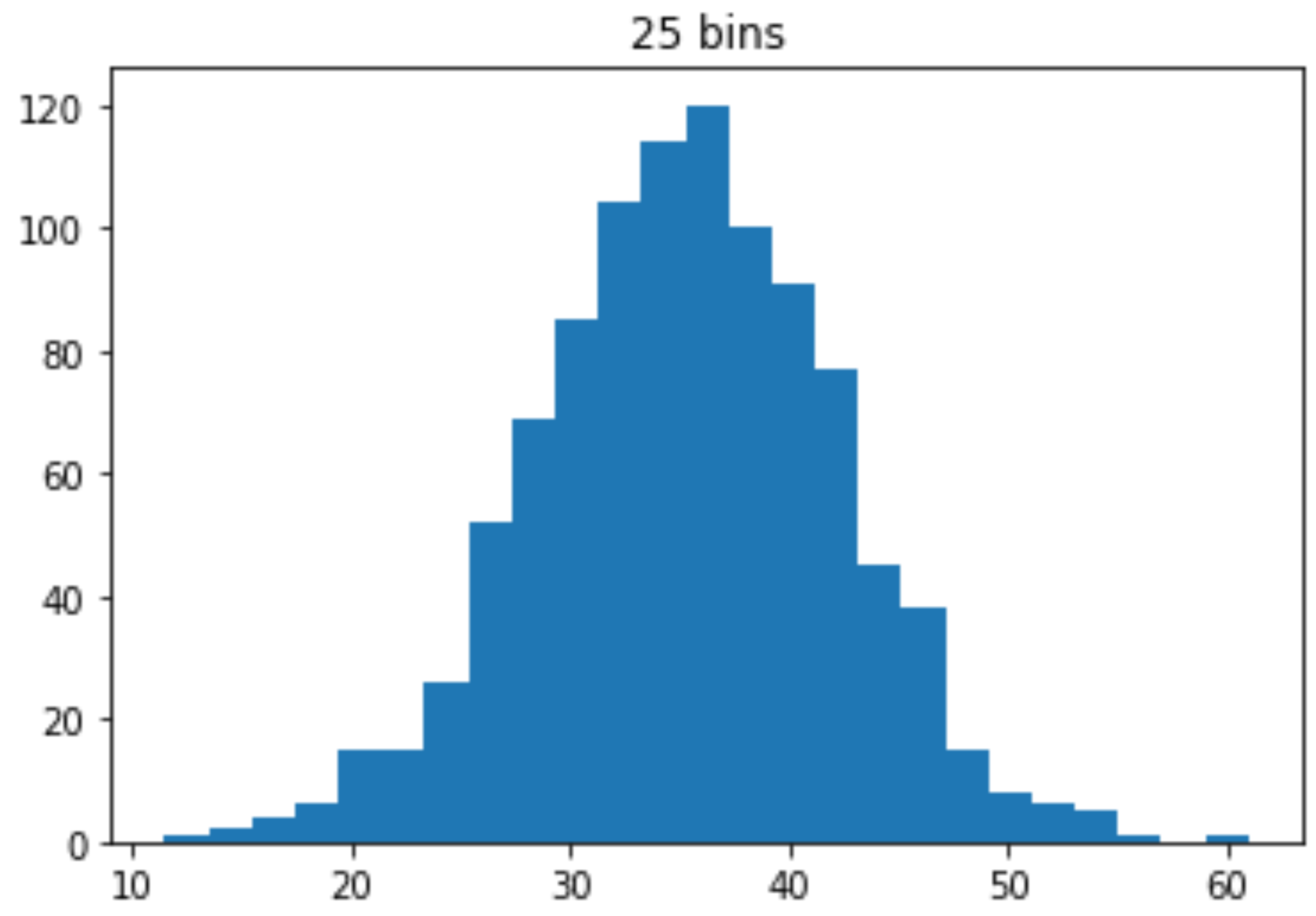
- This looks even better!

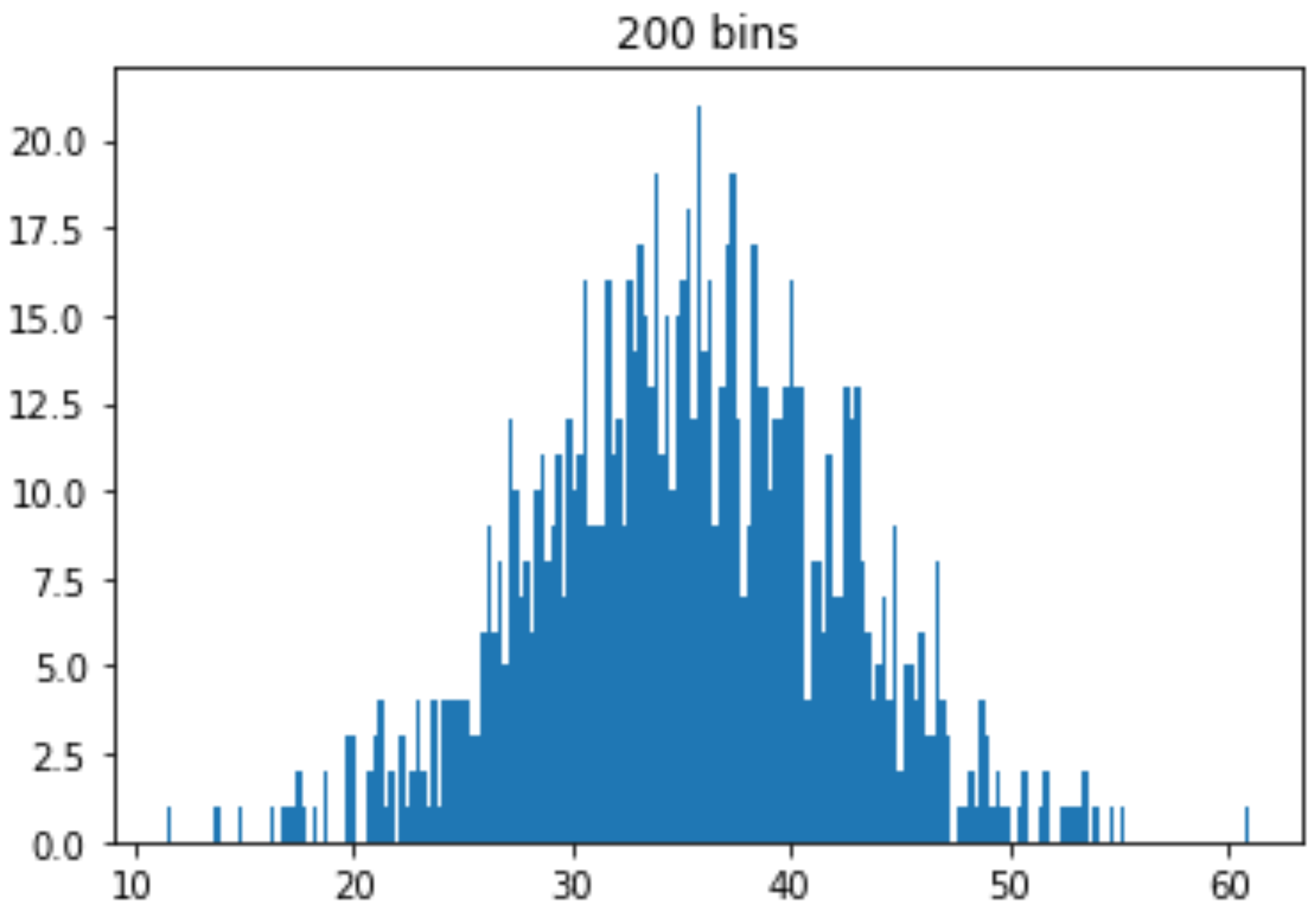
- As we add more data points, our histogram begins to look more and more like the “true” shape of the data (we’ll get in to what this means in a week or two)



how do we choose # bins?







how do we choose # bins?

k : **# bins**

h : **bin width**

$$k = \lceil \sqrt{n} \rceil$$

$$k = \lceil \log_2 n \rceil + 1$$

$$h = \frac{3.5\hat{\sigma}}{n^{1/3}}$$

$$k = \lceil 2n^{1/3} \rceil$$

bucket size intuition

- Intuition: the histogram *estimates* the "true" distribution of data using the *sample* of data you observe
- When given a new data point, can estimate how "likely" this data point is by looking at the frequency of the bucket the data point falls into
  - All data points that fall within the same bucket get the same estimate!
- Buckets too big: inaccurate because rare points and common points get put into the same bucket and get the same estimate
- Buckets too small: inaccurate because sample may not have an "accurate" picture of how common that bucket is (worst case: bucket may have size zero → will estimate that the data point *you just saw* has no chance of happening)
- Pick a bucket size that minimizes the error of estimating any point. But how do we do this if we don't know what the "true" data is?

cross validation

- Can use a *cross-validation* score.

$$J(h) = \frac{2}{(n-1)h} - \frac{n+1}{(n-1)h}(\hat{p}_1^2 + \hat{p}_1^2 + \ldots + \hat{p}_k^2)$$

cross validation

- For a given bucket width h, compute *J(h)*
- Find the *h* that minimizes *J(h)*