

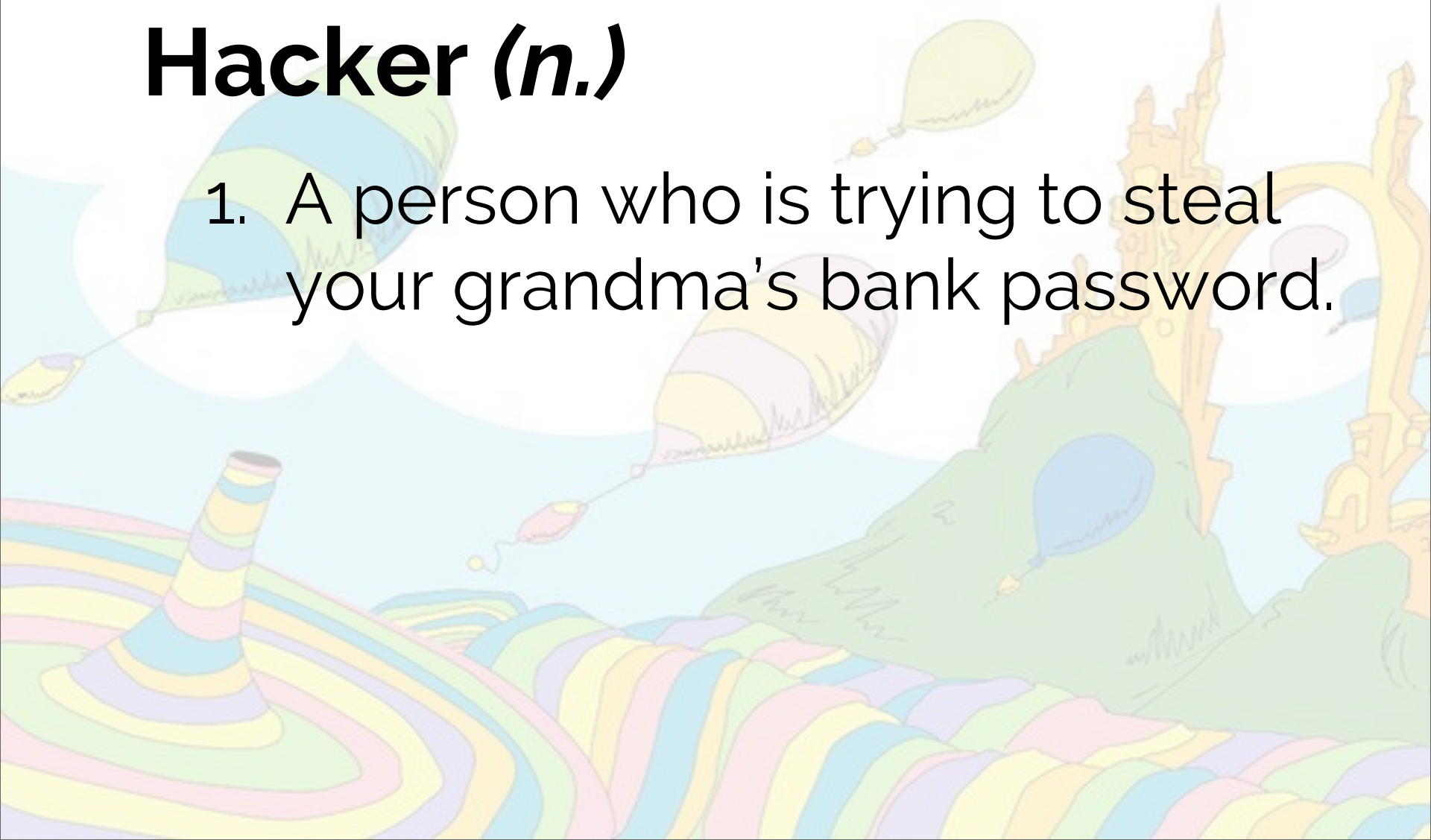
# Statistics for Hackers



# Statistics for Hackers

## Hacker (*n.*)

1. A person who is trying to steal your grandma's bank password.



# Statistics for Hackers

## Hacker (*n.*)

1. ~~A person who is trying to steal your grandma's bank password.~~
2. A person whose natural approach to problem-solving involves writing code.

**Statistics is Hard.**

**Statistics is Hard.**

**Using programming skills,  
it can be easy.**

*My thesis today:*

**If you can write a for-loop,  
you can do statistics**

*Statistics is fundamentally about*

**Asking the Right Question.**



Sometimes the  
questions are  
complicated  
and the  
answers are  
simple.



- Dr. Seuss (attr)



**Warm-up**

# Warm-up: Coin Toss

You toss a coin **30**  
times and see **22**  
heads. Is it a fair coin?



*A fair coin should show 15 heads in 30 tosses. This coin is biased.*

*Even a fair coin could show 22 heads in 30 tosses. It might be just chance.*



# Classic Method:

Assume the Skeptic is correct:  
test the *Null Hypothesis*.

What is the probability of a fair coin showing 22 heads simply by chance?



# Classic Method:

$$N_H = 22, N_T = 8$$

Start computing probabilities . . .

$$P(H) = \frac{1}{2}$$

$$P(HH) = \left(\frac{1}{2}\right)^2$$



# Classic Method:

$$N_H = 22, N_T = 8$$

$$P(HHT) = \left(\frac{1}{2}\right)^3$$

$$\begin{aligned} P(2H, 1T) &= P(HHT) \\ &\quad + P(HTH) \\ &\quad + P(THH) \\ &= \frac{3}{8} \end{aligned}$$



# Classic Method:

$$N_H = 22, N_T = 8$$

$$P(N_H, N_T) = \binom{N}{N_H} \left(\frac{1}{2}\right)^{N_H} \left(1 - \frac{1}{2}\right)^{N_T}$$

Number of  
arrangements  
(binomial  
coefficient)

Probability of  
 $N_H$  heads

Probability of  
 $N_T$  tails

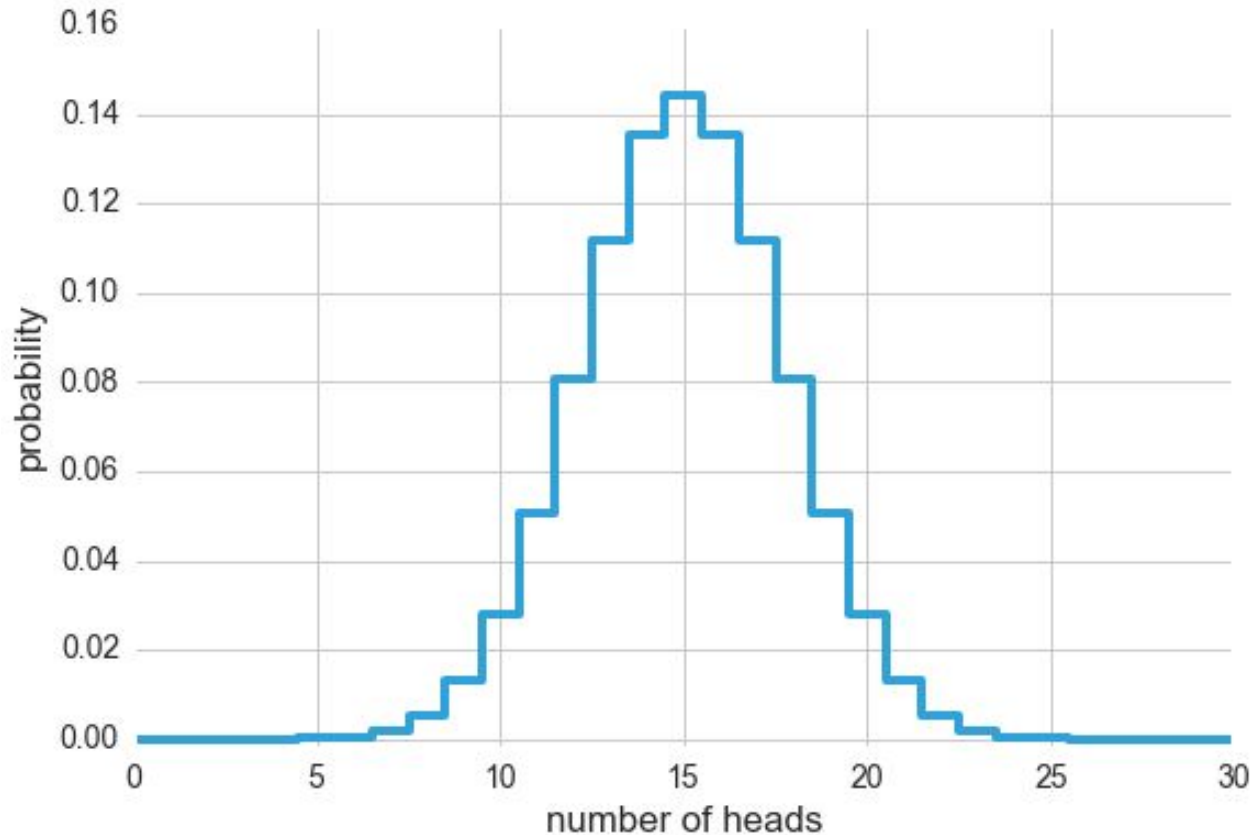




# Classic Method:

$$N_H = 22, N_T = 8$$

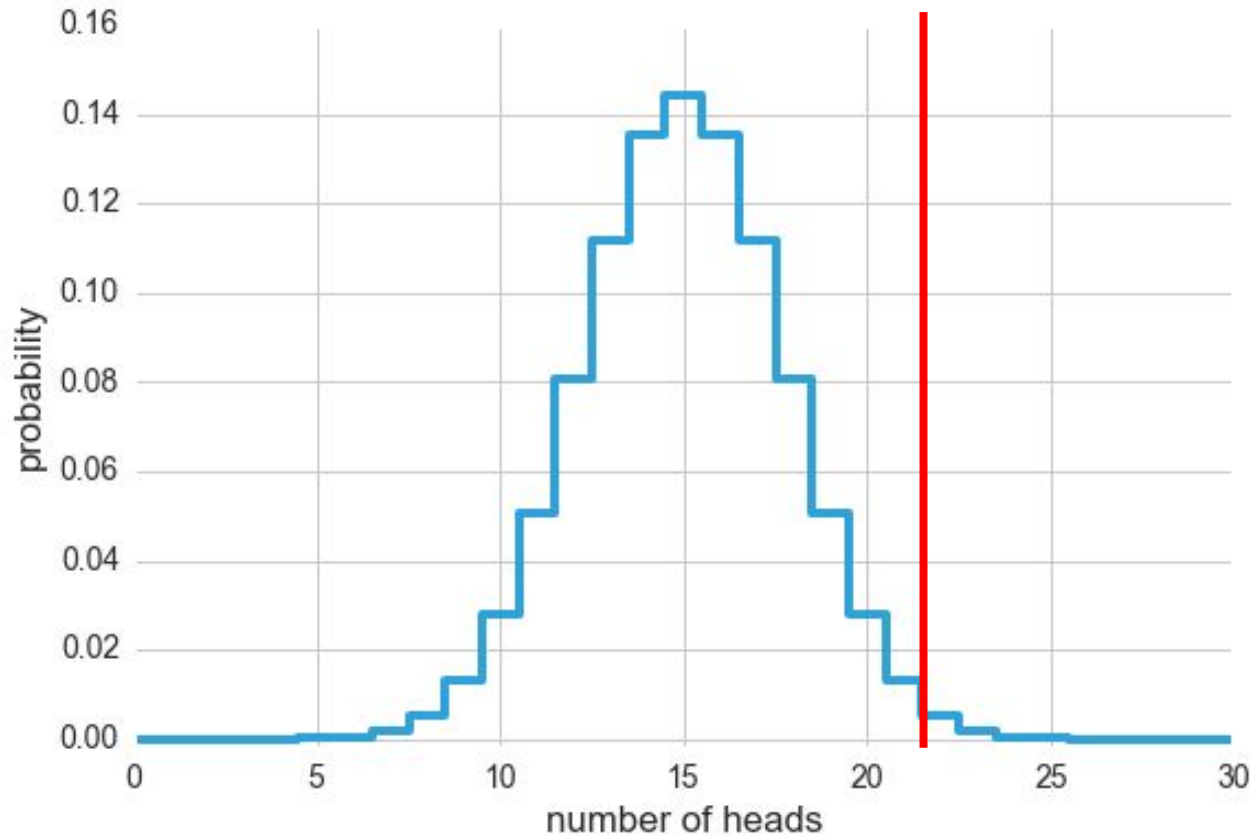
$$P(N_H, N_T) = \binom{N}{N_H} \left(\frac{1}{2}\right)^{N_H} \left(1 - \frac{1}{2}\right)^{N_T}$$



# Classic Method:

$$N_H = 22, N_T = 8$$

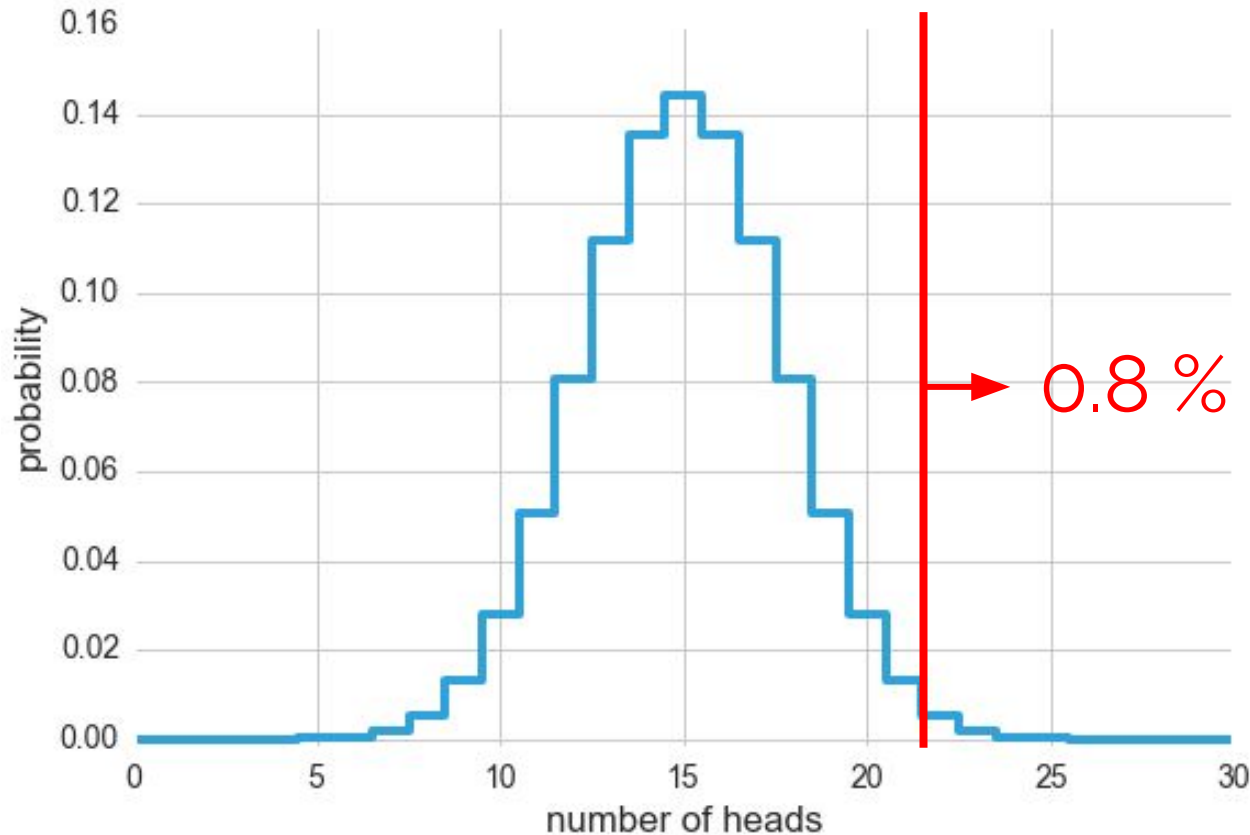
$$P(N_H, N_T) = \binom{N}{N_H} \left(\frac{1}{2}\right)^{N_H} \left(1 - \frac{1}{2}\right)^{N_T}$$



# Classic Method:

$$N_H = 22, N_T = 8$$

$$P(N_H, N_T) = \binom{N}{N_H} \left(\frac{1}{2}\right)^{N_H} \left(1 - \frac{1}{2}\right)^{N_T}$$

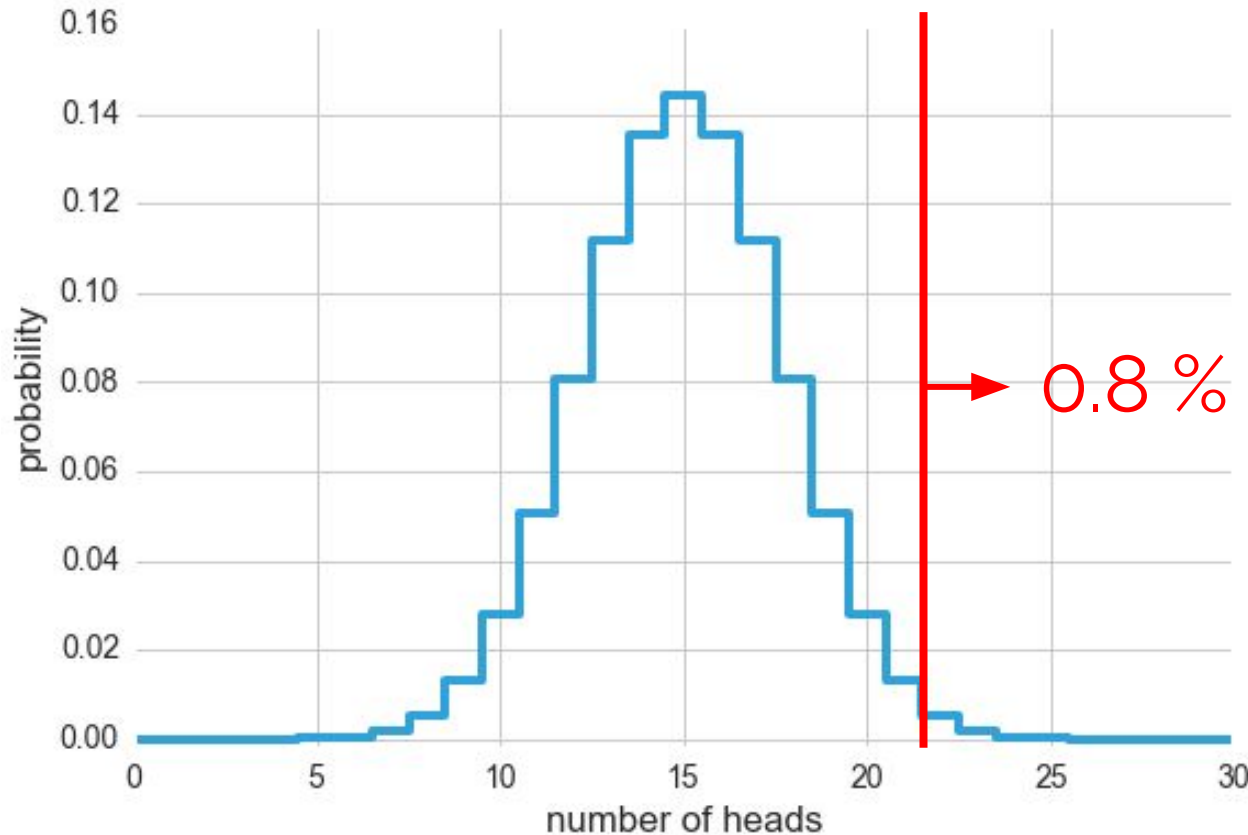


# Classic Method:

$$N_H = 22, N_T = 8$$

Probability of 0.8% (i.e.  $p = 0.008$ ) of observations given a fair coin.

→ **reject fair coin hypothesis at  $p < 0.05$**



**Could there be  
an easier way?**

# Easier Method:

Just simulate it!

```
M = 0
for i in range(10000):
    trials = randint(2, size=30)
    if (trials.sum() >= 22):
        M += 1
p = M / 10000 # 0.008149
```

→ reject fair coin at  $p = 0.008$



In general . . .

**Computing** the Sampling  
Distribution is **Hard**.



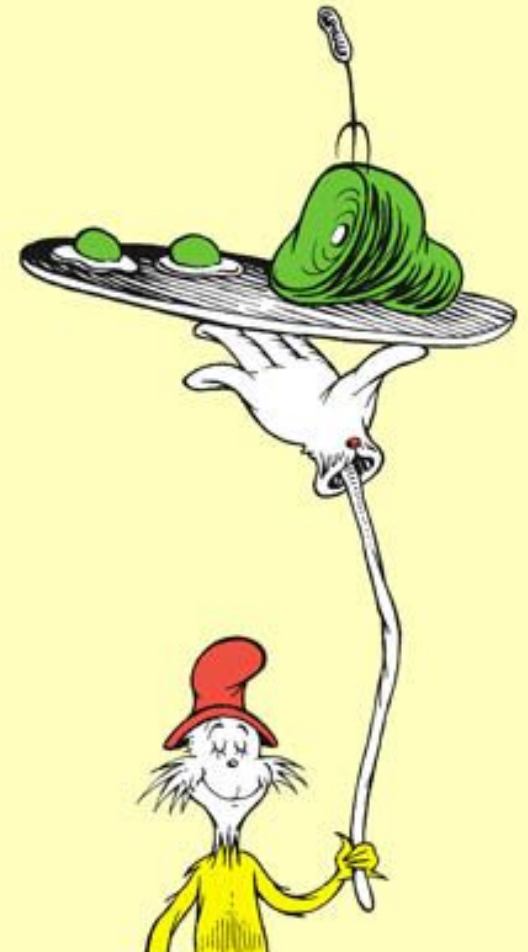
In general . . .

**Computing** the Sampling  
Distribution is **Hard**.

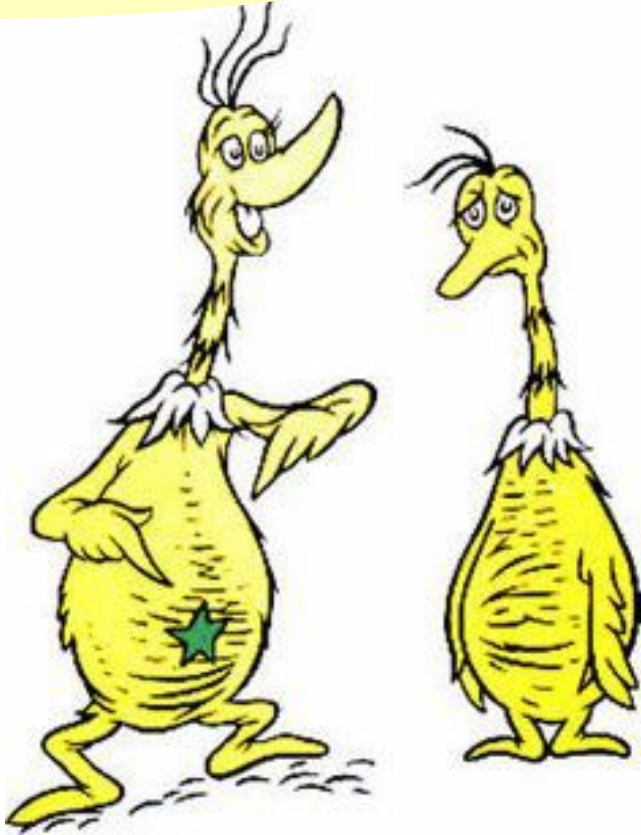
**Simulating** the Sampling  
Distribution is **Easy**.

# Four Recipes for Hacking Statistics:

1. Direct Simulation ✓
2. Shuffling
3. Bootstrapping
4. Cross Validation



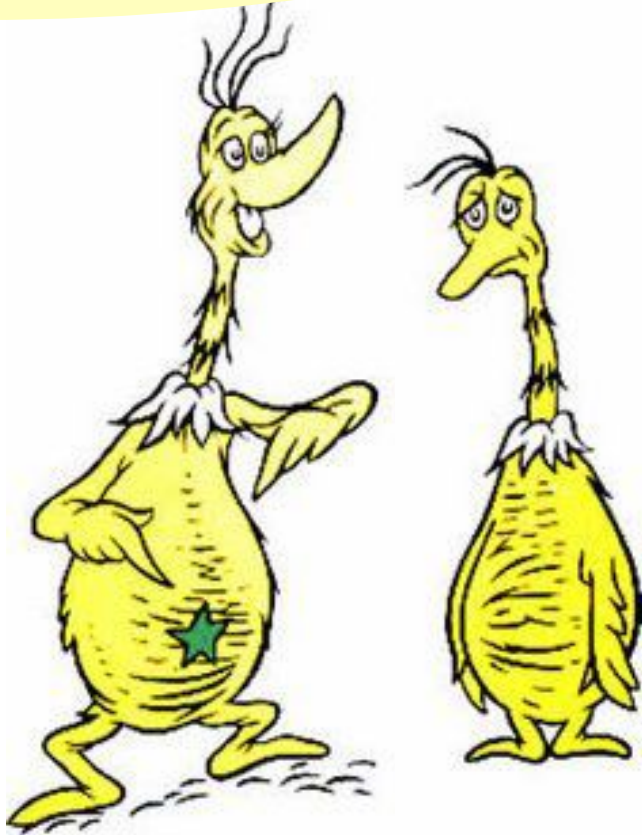
# Sneeches: Stars and Intelligence



*Now, the Star-Belly Sneetches  
had bellies with stars.  
The Plain-Belly Sneetches  
had none upon thars . . .*

*\*inspired by John Rauser's  
Statistics Without All The Agonizing Pain*

# Sneeches: Stars and Intelligence



Test Scores

★		×	
84	72	81	69
57	46	74	61
63	76	56	87
99	91	69	65
		66	44
		62	69

★ mean: 73.5  
× mean: 66.9  
difference: 6.6

# Is this difference of 6.6 statistically significant?

★ mean: 73.5  
✕ mean: 66.9  
difference: 6.6

# Classic Method

(Welch's t-test)

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$$

# Classic Method

(Welch's t-test)

$$t = \frac{73.5 - 66.9}{\sqrt{\frac{316.3}{8} + \frac{124.8}{12}}} = 0.932$$



# Classic Method

(Student's t distribution)

$$p(t; \nu) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{\nu\pi} \Gamma\left(\frac{\nu}{2}\right)} \left(1 + \frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}}$$

# Classic Method

(Student's t distribution)

$$p(t; \nu) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{\nu\pi} \Gamma\left(\frac{\nu}{2}\right)} \left(1 + \frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}}$$

*Degree of Freedom:* “The number of independent ways by which a dynamic system can move, without violating any constraint imposed on it.”

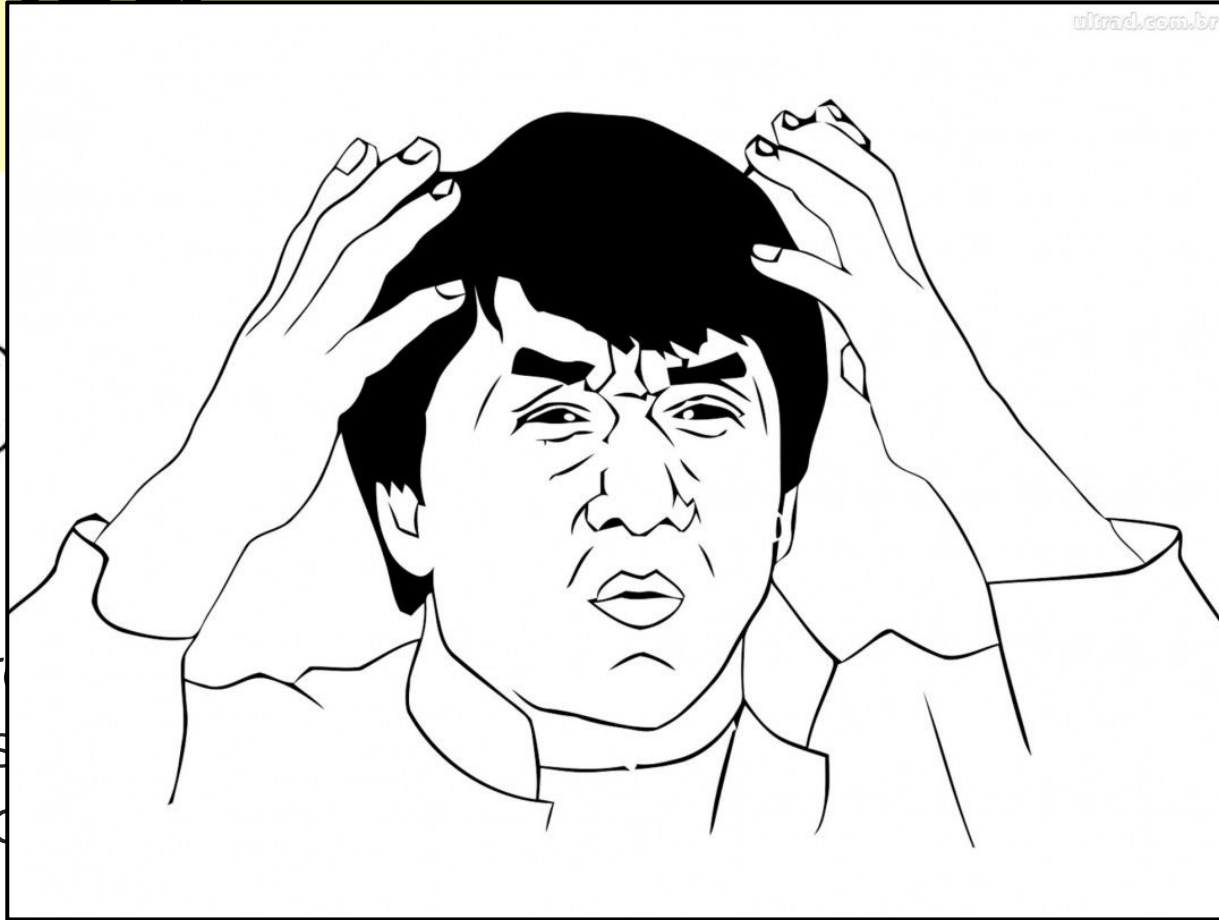
-Wikipedia

# Classic Method

(Student's t distribution)

$p(t; \nu)$

Degr  
ways  
witho



$$-\frac{\nu+1}{2}$$

ent

pedia

# Classic Method

( Welch–Satterthwaite  
equation)

$$\nu \approx \frac{\left( \frac{s_1^2}{N_1} + \frac{s_2^2}{N_2} \right)^2}{\frac{s_1^4}{N_1^2(N_1-1)} + \frac{s_2^4}{N_2^2(N_2-1)}}$$

# Classic Method

( Welch–Satterthwaite  
equation)

$$\nu \approx \frac{\left( \frac{316.3}{8} + \frac{124.8}{12} \right)^2}{\frac{316.3^2}{8^2(8-1)} + \frac{124.8^2}{12^2(12-1)}} = 10.7$$

# Classic Method

$\alpha$ (1 tail)	0.05	0.025	0.01	0.005	0.0025	0.001	0.0005
$\alpha$ (2 tail)	0.1	0.05	0.02	0.01	0.005	0.002	0.001
df							
1	6.3138	12.7065	31.8193	63.6551	127.3447	318.4930	636.0450
2	2.9200	4.3026	6.9646	9.9247	14.0887	22.3276	31.5989
3	2.3534	3.1824	4.5407	5.8408	7.4534	10.2145	12.9242
4	2.1319	2.7764	3.7470	4.6041	5.5976	7.1732	8.6103
5	2.0150	2.5706	3.3650	4.0322	4.7734	5.8934	6.8688
6	1.9432	2.4469	3.1426	3.7074	4.3168	5.2076	5.9589
7	1.8946	2.3646	2.9980	3.4995	4.0294	4.7852	5.4079
8	1.8595	2.3060	2.8965	3.3554	3.8325	4.5008	5.0414
9	1.8331	2.2621	2.8214	3.2498	3.6896	4.2969	4.7809
10	1.8124	2.2282	2.7638	3.1693	3.5814	4.1437	4.5869
11	1.7959	2.2010	2.7181	3.1058	3.4966	4.0247	4.4369
12	1.7823	2.1788	2.6810	3.0545	3.4284	3.9296	4.3178
13	1.7709	2.1604	2.6503	3.0123	3.3725	3.8520	4.2208
14	1.7613	2.1448	2.6245	2.9768	3.3257	3.7874	4.1404
15	1.7530	2.1314	2.6025	2.9467	3.2860	3.7328	4.0728
16	1.7459	2.1199	2.5835	2.9208	3.2520	3.6861	4.0150
17	1.7396	2.1098	2.5669	2.8983	3.2224	3.6458	3.9651
18	1.7341	2.1009	2.5524	2.8784	3.1966	3.6105	3.9216
19	1.7291	2.0930	2.5395	2.8609	3.1737	3.5794	3.8834
20	1.7247	2.0860	2.5280	2.8454	3.1534	3.5518	3.8495

# Classic Method

$\alpha$ (1 tail)	0.05	0.025	0.01	0.005	0.0025	0.001	0.0005
$\alpha$ (2 tail)	0.1	0.05	0.02	0.01	0.005	0.002	0.001
df							
1	6.3138	12.7065	31.8193	63.6551	127.3447	318.4930	636.0450
2	2.9200	4.3026	6.9646	9.9247	14.0887	22.3276	31.5989
3	2.3534	3.1824	4.5407	5.8408	7.4534	10.2145	12.9242
4	2.1319	2.7764	3.7470	4.6041	5.5976	7.1732	8.6103
5	2.0150	2.5706	3.3650	4.0322	4.7734	5.8934	6.8688
6	1.9432	2.4469	3.1426	3.7074	4.3168	5.2076	5.9589
7	1.8946	2.3646	2.9980	3.4995	4.0294	4.7852	5.4079
8	1.8595	2.3060	2.8965	3.3554	3.8325	4.5008	5.0414
9	1.8331	2.2621	2.8214	3.2498	3.6896	4.2969	4.7809
10	1.8124	2.2282	2.7638	3.1693	3.5814	4.1437	4.5869
11	1.7959	2.2010	2.7181	3.1058	3.4966	4.0247	4.4369
12	1.7823	2.1788	2.6810	3.0545	3.4284	3.9296	4.3178
13	1.7709	2.1604	2.6503	3.0123	3.3725	3.8520	4.2208
14	1.7613	2.1448	2.6245	2.9768	3.3257	3.7874	4.1404
15	1.7530	2.1314	2.6025	2.9467	3.2860	3.7328	4.0728
16	1.7459	2.1199	2.5835	2.9208	3.2520	3.6861	4.0150
17	1.7396	2.1098	2.5669	2.8983	3.2224	3.6458	3.9651
18	1.7341	2.1009	2.5524	2.8784	3.1966	3.6105	3.9216
19	1.7291	2.0930	2.5395	2.8609	3.1737	3.5794	3.8834
20	1.7247	2.0860	2.5280	2.8454	3.1534	3.5518	3.8495



# Classic Method

$\alpha$ (1 tail)	0.05	0.025	0.01	0.005	0.0025	0.001	0.0005
$\alpha$ (2 tail)	0.1	0.05	0.02	0.01	0.005	0.002	0.001
df							
1	6.3138	12.7065	31.8193	63.6551	127.3447	318.4930	636.0450
2	2.9200	4.3026	6.9646	9.9247	14.0887	22.3276	31.5989
3	2.3534	3.1824	4.5407	5.8408	7.4534	10.2145	12.9242
4	2.1319	2.7764	3.7470	4.6041	5.5976	7.1732	8.6103
5	2.0150	2.5706	3.3650	4.0322	4.7734	5.8934	6.8688
6	1.9432	2.4469	3.1426	3.7074	4.3168	5.2076	5.9589
7	1.8946	2.3646	2.9980	3.4995	4.0294	4.7852	5.4079
8	1.8595	2.3060	2.8965	3.3554	3.8325	4.5008	5.0414
9	1.8331	2.2621	2.8214	3.2498	3.6896	4.2969	4.7809
10	1.8181	2.2282	2.7638	3.1693	3.5814	4.1437	4.5869
11	1.8047	2.2010	2.7181	3.1058	3.4966	4.0247	4.4369
12	1.7929	2.1788	2.6810	3.0545	3.4284	3.9296	4.3178
13	1.7709	2.1604	2.6503	3.0123	3.3725	3.8520	4.2208
14	1.7613	2.1448	2.6245	2.9768	3.3257	3.7874	4.1404
15	1.7530	2.1314	2.6025	2.9467	3.2860	3.7328	4.0728
16	1.7459	2.1199	2.5835	2.9208	3.2520	3.6861	4.0150
17	1.7396	2.1098	2.5669	2.8983	3.2224	3.6458	3.9651
18	1.7341	2.1009	2.5524	2.8784	3.1966	3.6105	3.9216
19	1.7291	2.0930	2.5395	2.8609	3.1737	3.5794	3.8834
20	1.7247	2.0860	2.5280	2.8454	3.1534	3.5518	3.8495

# Classic Method

$$t > t_{crit}$$

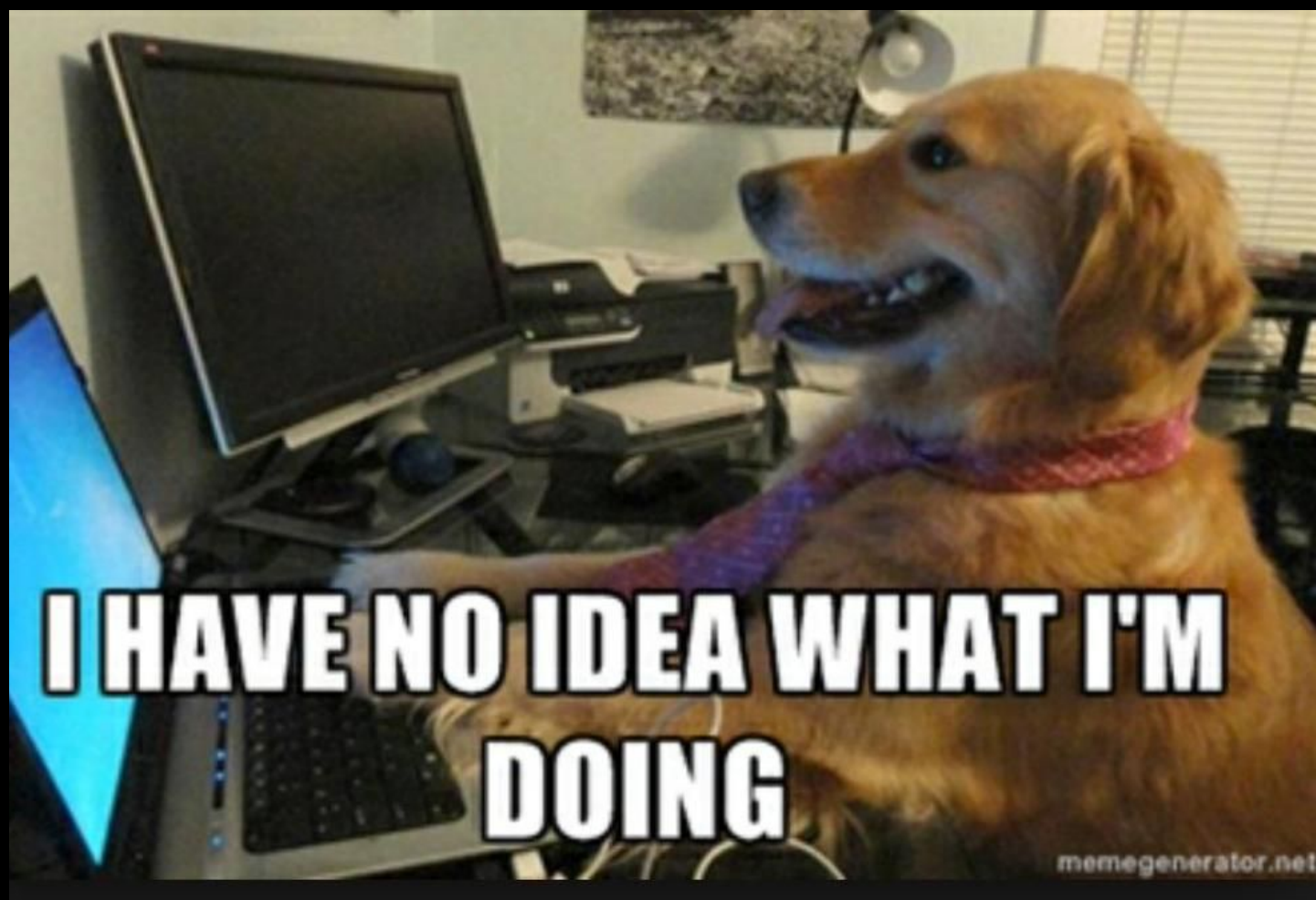
# Classic Method

$$0.932 > 1.796$$

# Classic Method

$$\del{0.932 > 1.796}$$

**“The difference of 6.6 is not significant at the  $p=0.05$  level”**



**I HAVE NO IDEA WHAT I'M  
DOING**

memegenerator.net

The biggest problem:

**We've entirely lost-track  
of what **question** we're  
answering!**

## < One popular alternative . . . >

# “Why don’t you just . . .”

```
from statsmodels.stats.weightstats import ttest_ind  
t, p, dof = ttest_ind(group1, group2,  
                        alternative='larger',  
                        usevar='unequal')  
  
print(p)    # 0.186
```



< One popular alternative . . . >

**“Why don’t you just . . .”**

```
from statsmodels.stats.weightstats import ttest_ind
t, p, dof = ttest_ind(group1, group2,
                      alternative='larger',
                      usevar='unequal')
print(p)    # 0.186
```

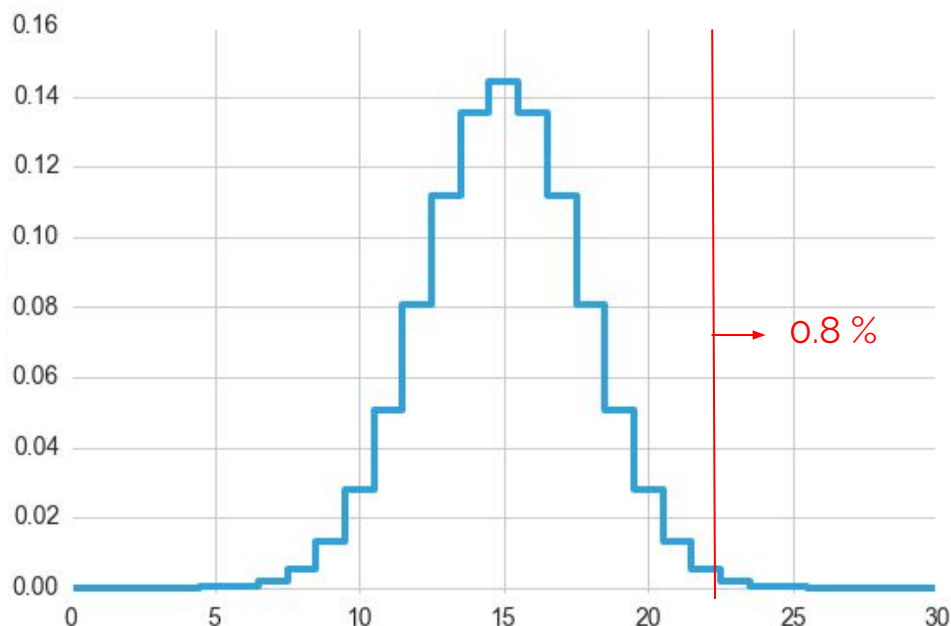
**. . . But what question is  
this answering?**

# Stepping Back...

The deep meaning lies in the *sampling distribution*:

$$p(t; \nu) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{\nu\pi} \Gamma\left(\frac{\nu}{2}\right)} \left(1 + \frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}}$$

Same principle as the coin example:



**Let's use a sampling  
method instead**

The Problem:

Unlike coin flipping, we *don't*  
have a **generative model** . . .

The Problem:

Unlike coin flipping, we *don't*  
have a **generative model** . . .

Solution:

**Shuffling**

★		×	
84	72	81	69
57	46	74	61
63	76	56	87
99	91	69	65
		66	44
		62	69

## Idea:

Simulate the distribution by *shuffling* the labels repeatedly and computing the desired statistic.

## Motivation:

if the labels really don't matter, then switching them shouldn't change the result!

★		×	
84	72	81	69
57	46	74	61
63	76	56	87
99	91	69	65
		66	44
		62	69

1. Shuffle Labels
2. Rearrange
3. Compute means

★		×	
84	72	81	69
57	46	74	61
63	76	56	87
99	91	69	65
		66	44
		62	69

1. **Shuffle Labels**
2. Rearrange
3. Compute means



★		×	
84	81	72	69
61	69	74	57
65	76	56	87
99	44	46	63
		66	91
		62	69

1. Shuffle Labels
- 2. Rearrange**
3. Compute means

★		×	
84	81	72	69
61	69	74	57
65	76	56	87
99	44	46	63
		66	91
		62	69

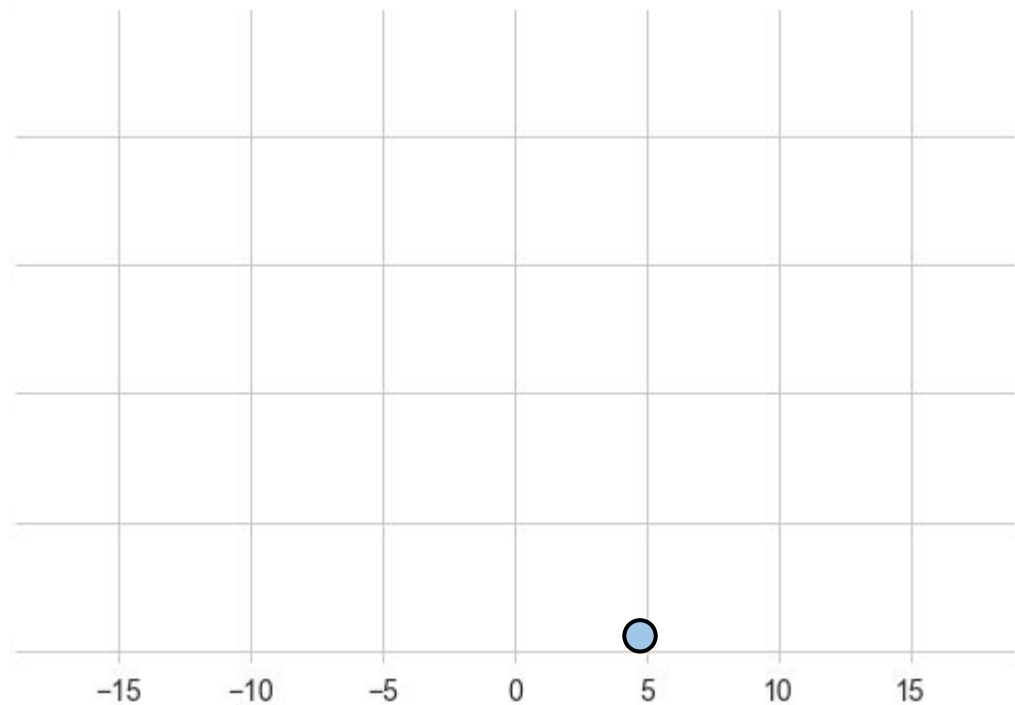
1. Shuffle Labels
2. Rearrange
- 3. Compute means**

★ mean: 72.4  
 × mean: 67.6  
 difference: 4.8

★		×	
84	81	72	69
61	69	74	57
65	76	56	87
99	44	46	63
		66	91
		62	69

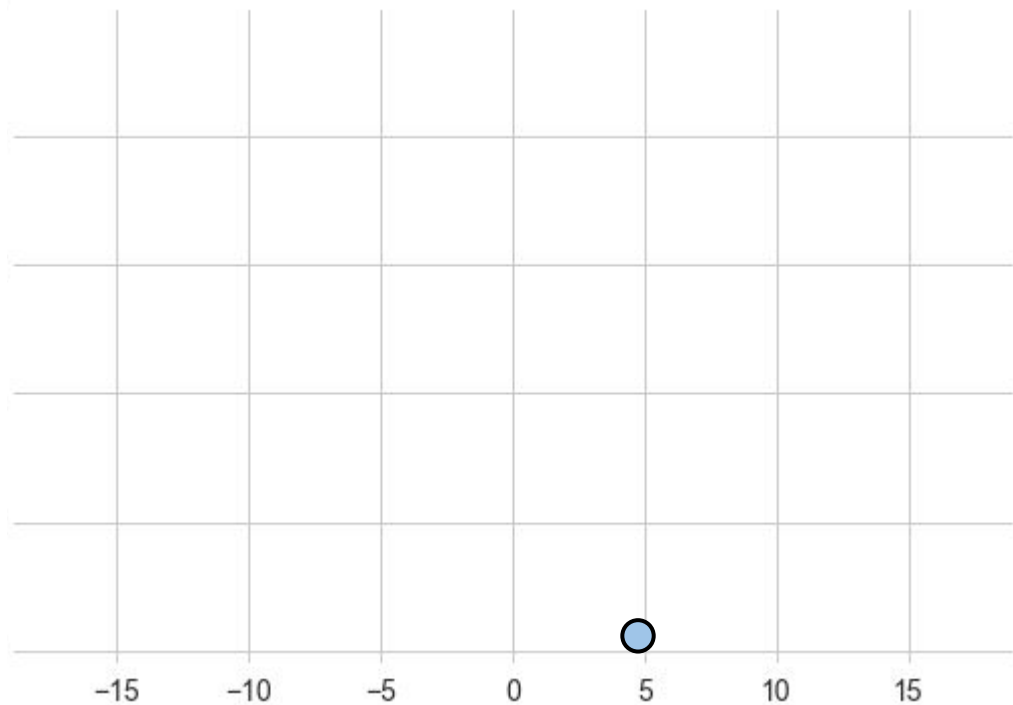
★ mean: 72.4  
 × mean: 67.6  
 difference: 4.8

1. Shuffle Labels
2. Rearrange
- 3. Compute means**



★		×	
84	81	72	69
61	69	74	57
65	76	56	87
99	44	46	63
		66	91
		62	69

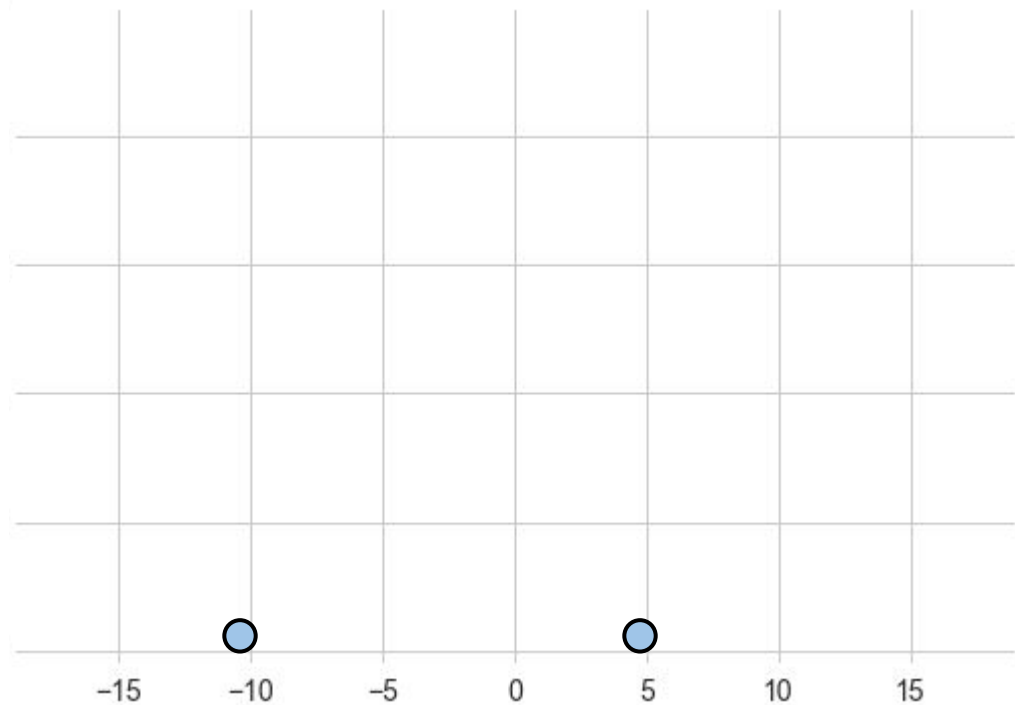
1. Shuffle Labels
2. Rearrange
3. Compute means



★		×	
84	56	72	69
61	63	74	57
65	66	81	87
62	44	46	69
		76	91
		99	69

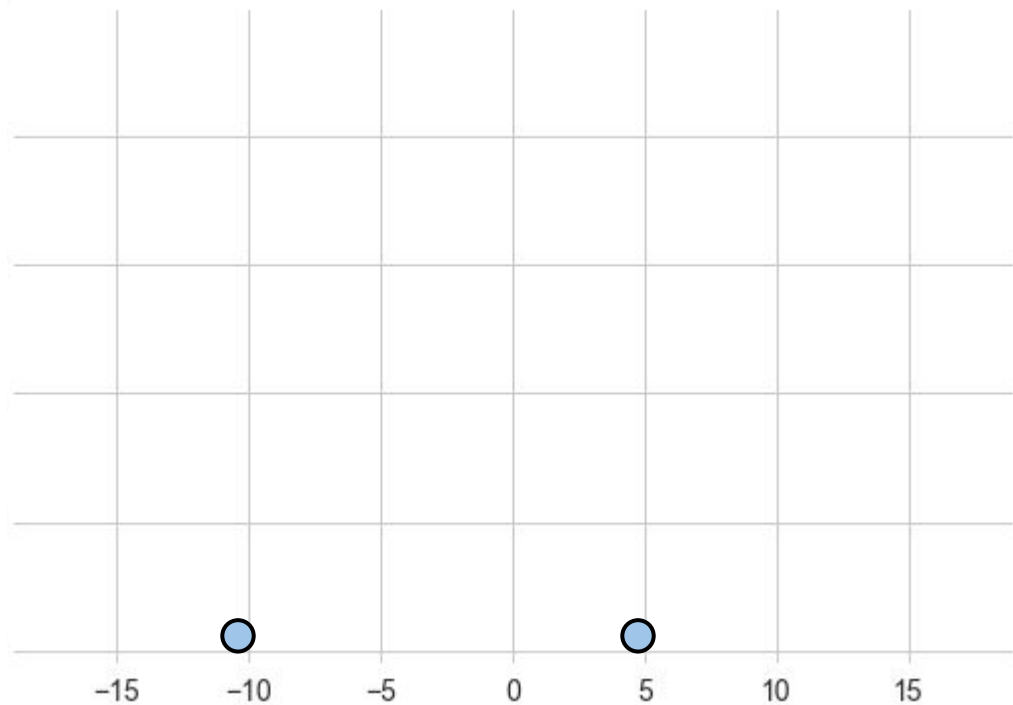
★ mean: 62.6  
 × mean: 74.1  
 difference: -11.6

1. Shuffle Labels
2. Rearrange
- 3. Compute means**



★		×	
84	56	72	69
61	63	74	57
65	66	81	87
62	44	46	69
		76	91
		99	69

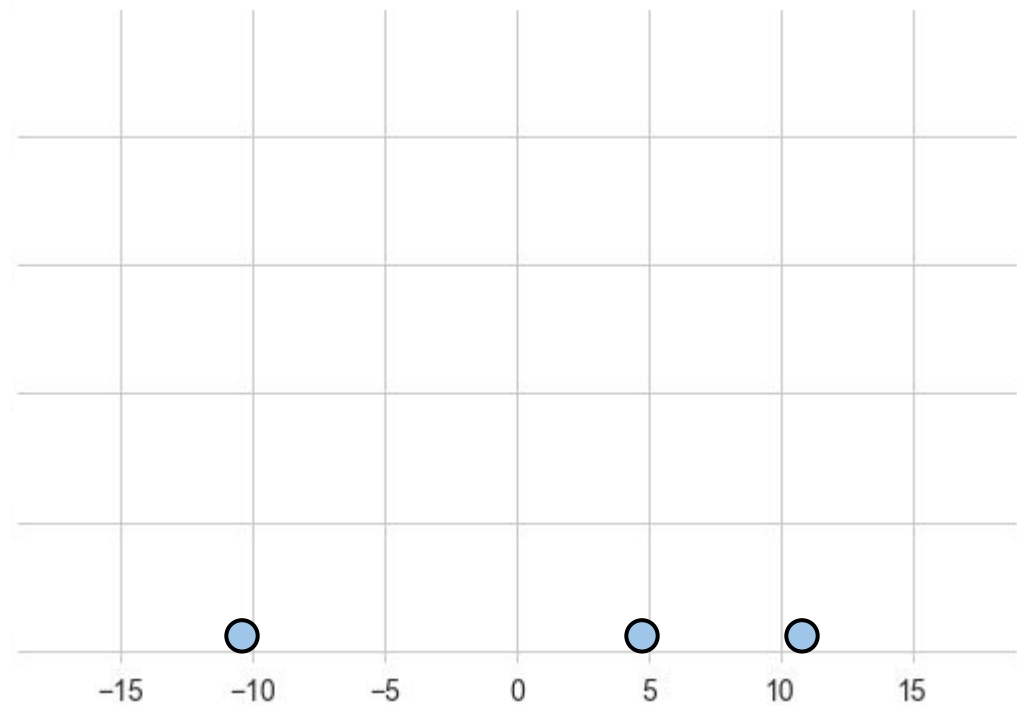
1. Shuffle Labels
2. Rearrange
3. Compute means



★		×	
74	56	72	69
61	63	84	57
87	76	81	65
91	99	46	69
		66	62
		44	69

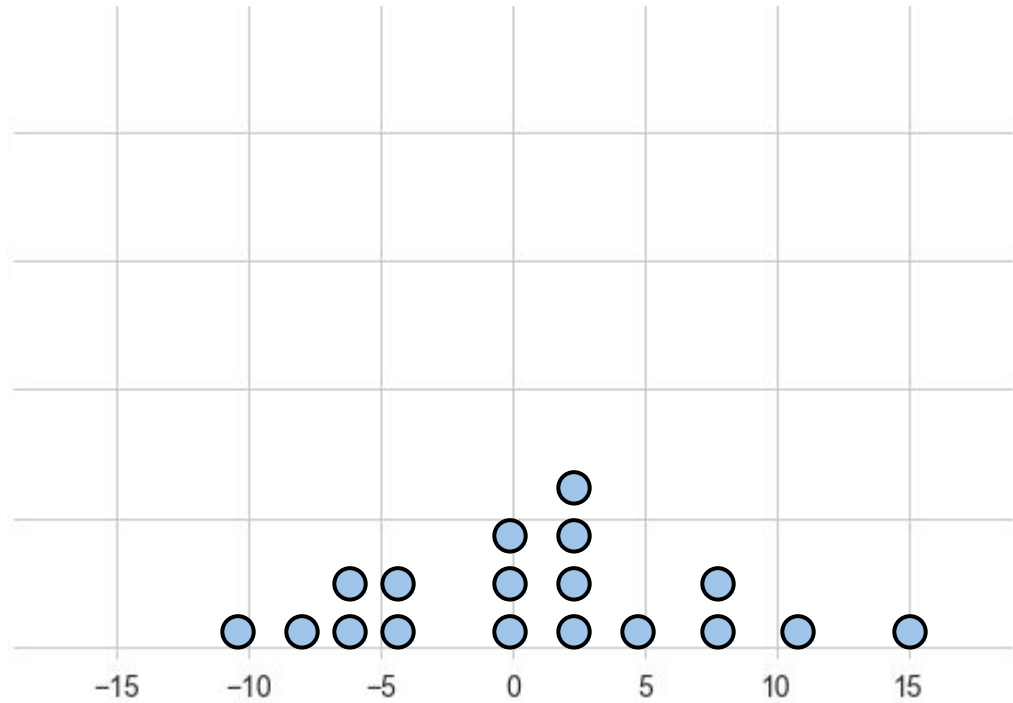
★ mean: 75.9  
 × mean: 65.3  
 difference: 10.6

1. Shuffle Labels
2. Rearrange
- 3. Compute means**



★		×	
84	56	72	69
61	63	74	57
65	66	81	87
62	44	46	69
		76	91
		99	69

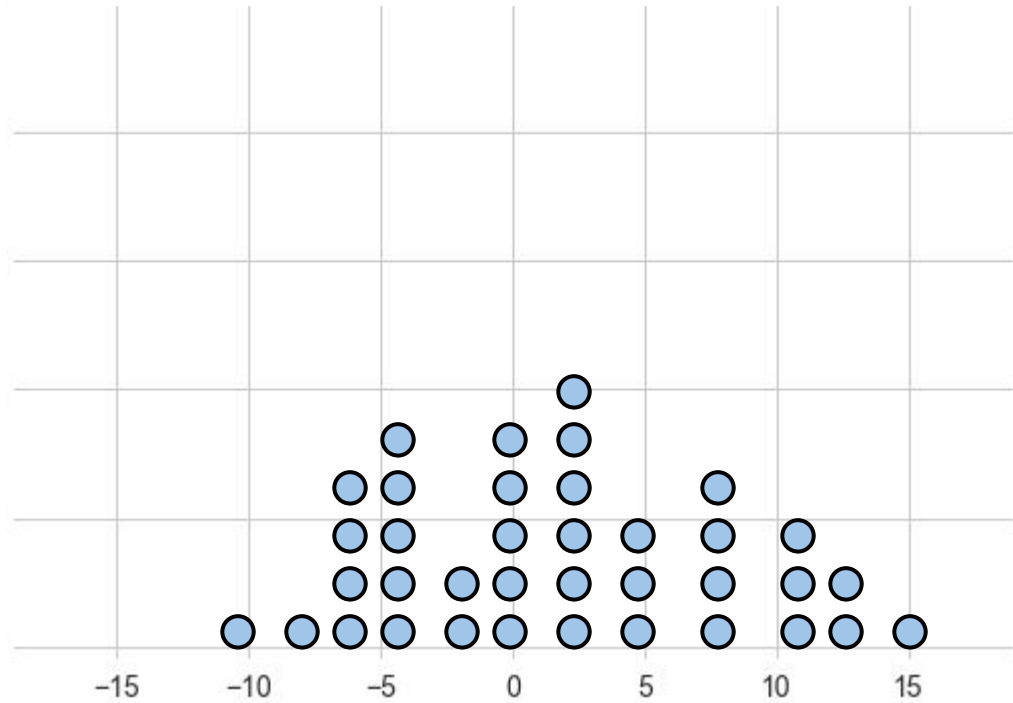
1. Shuffle Labels
2. Rearrange
3. Compute means





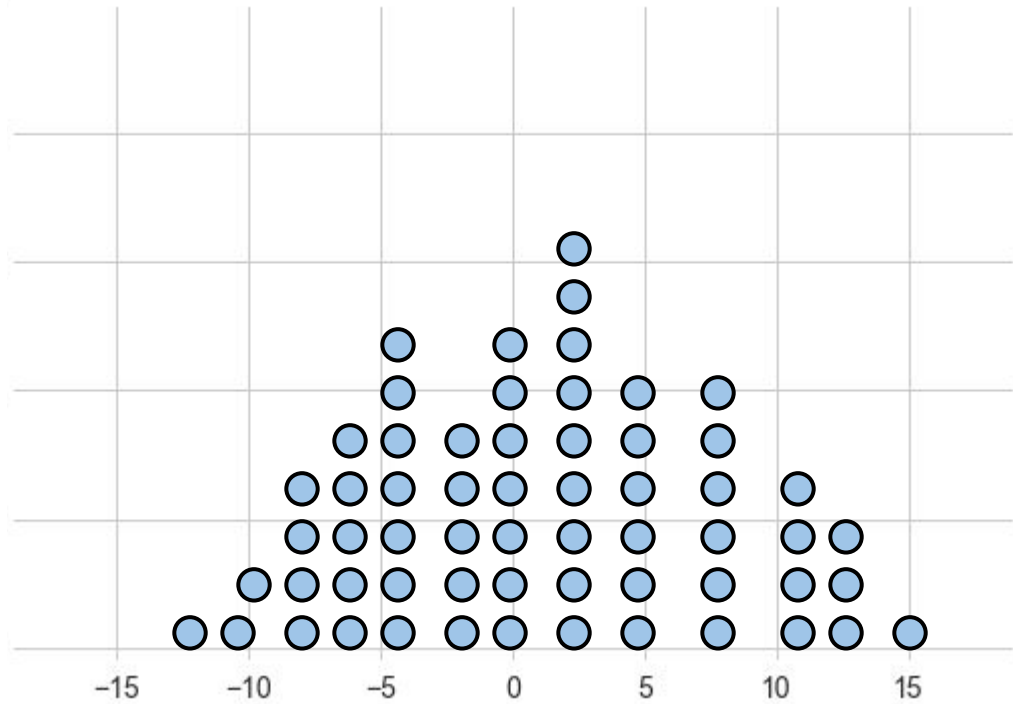
★		×	
84	81	69	69
61	69	87	74
65	76	56	57
99	44	46	63
		66	91
		62	72

1. Shuffle Labels
2. Rearrange
3. Compute means



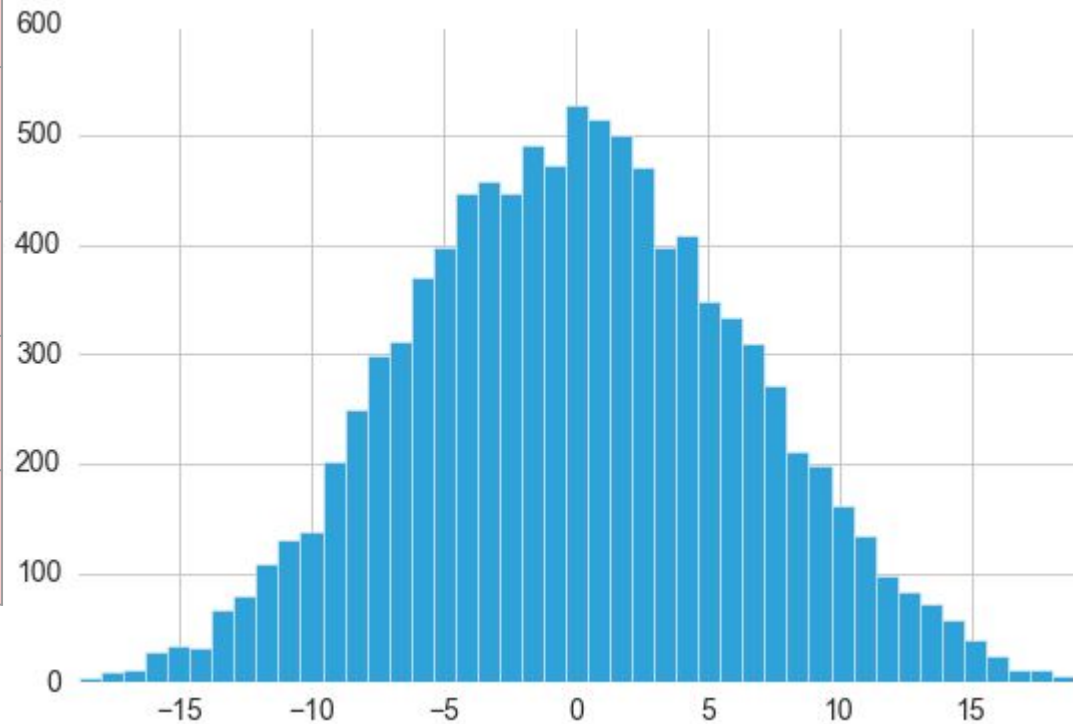
★		×	
74	62	72	57
61	63	84	69
87	81	76	65
91	99	46	69
		66	56
		44	69

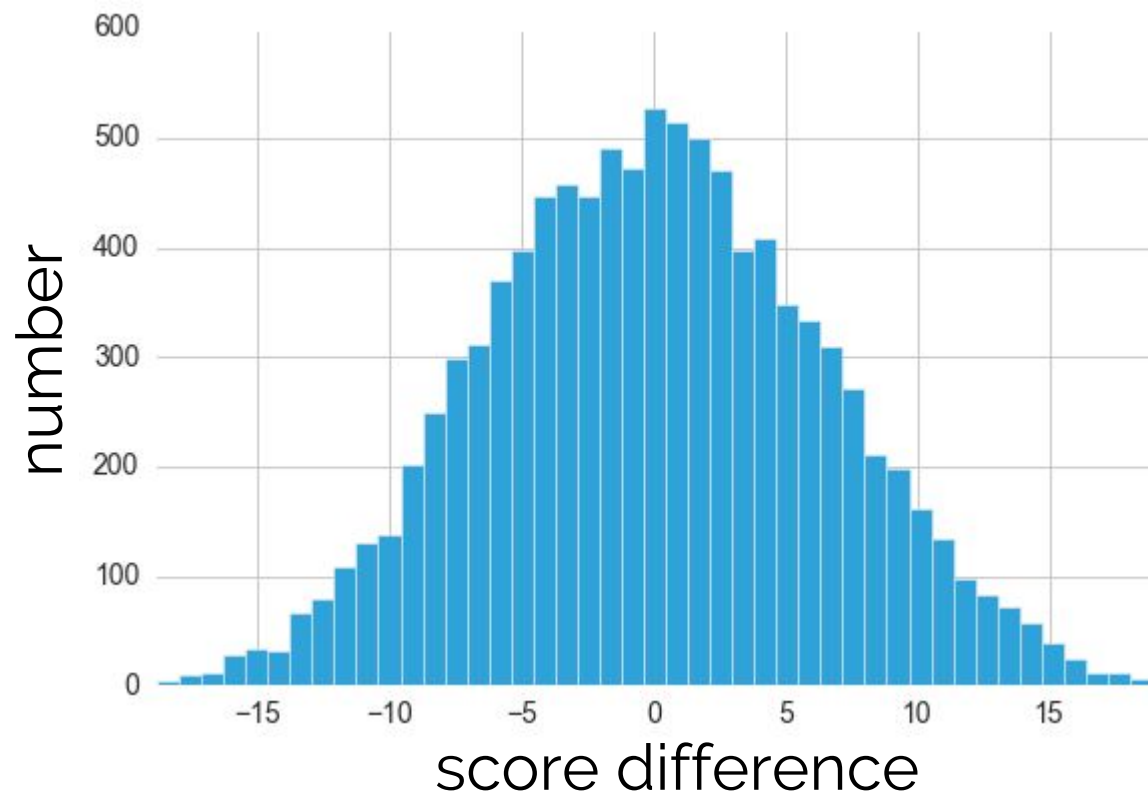
1. Shuffle Labels
2. Rearrange
3. Compute means

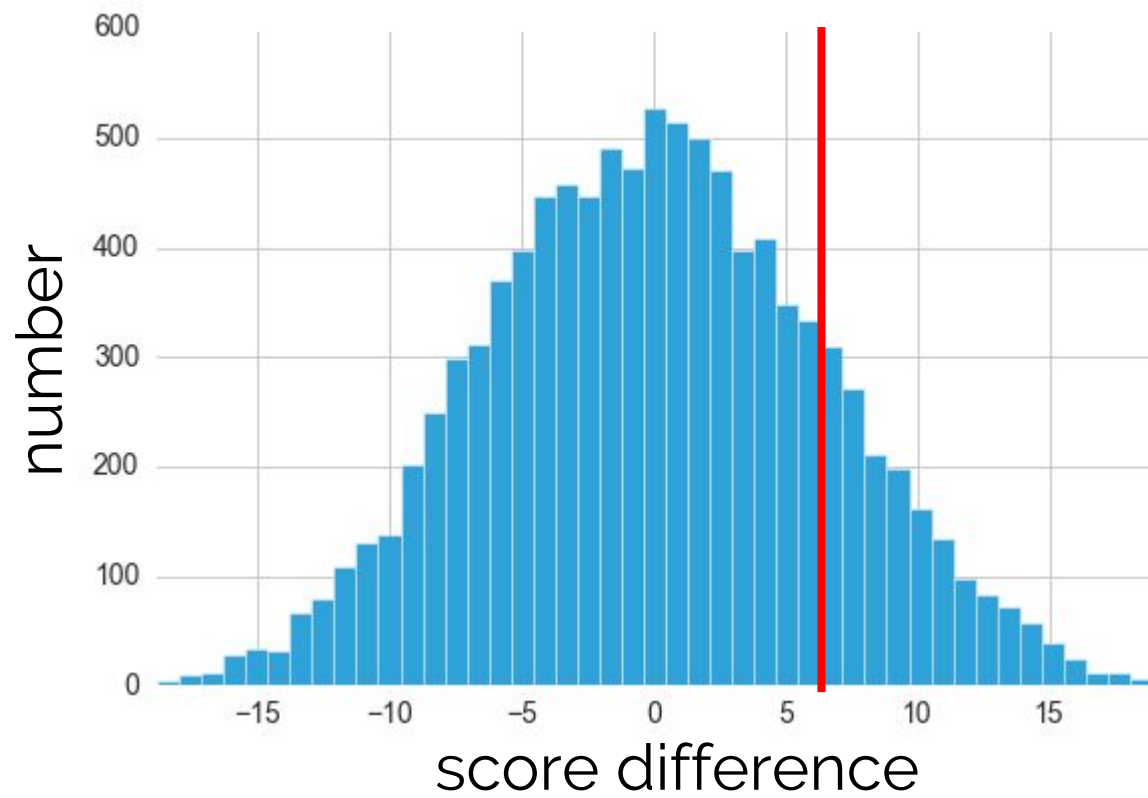


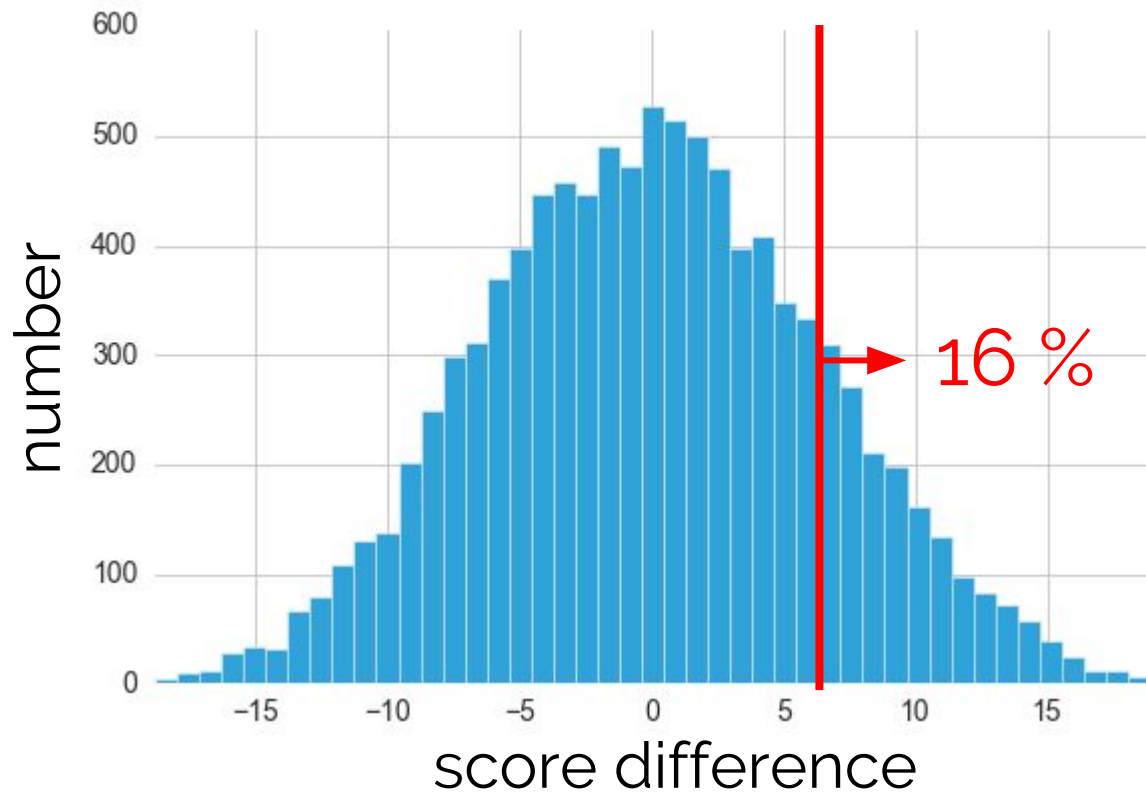
★		×	
84	81	72	69
61	69	74	57
65	76	56	87
99	44	46	63
		66	91
		62	69

1. Shuffle Labels
2. Rearrange
3. Compute means









$$\frac{N_{>6.6}}{N_{tot}} = \frac{1608}{10000} = 0.16$$

**“A difference of 6.6 is not  
significant at  $p = 0.05$ .”**



*That day, all the Sneetches  
forgot about stars  
And whether they had one,  
or not, upon thars.*

# Notes on Shuffling:

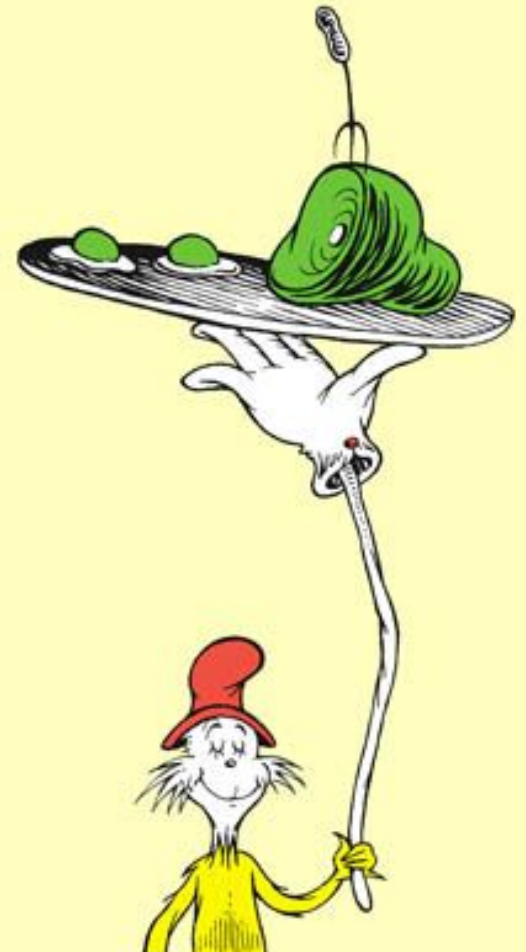
- Works when the *Null Hypothesis* assumes two groups are equivalent
- Like all methods, it will only work if your samples are representative – always be careful about selection biases!
- Needs care for non-independent trials.  
Good discussion in Simon's *Resampling: The New Statistics*





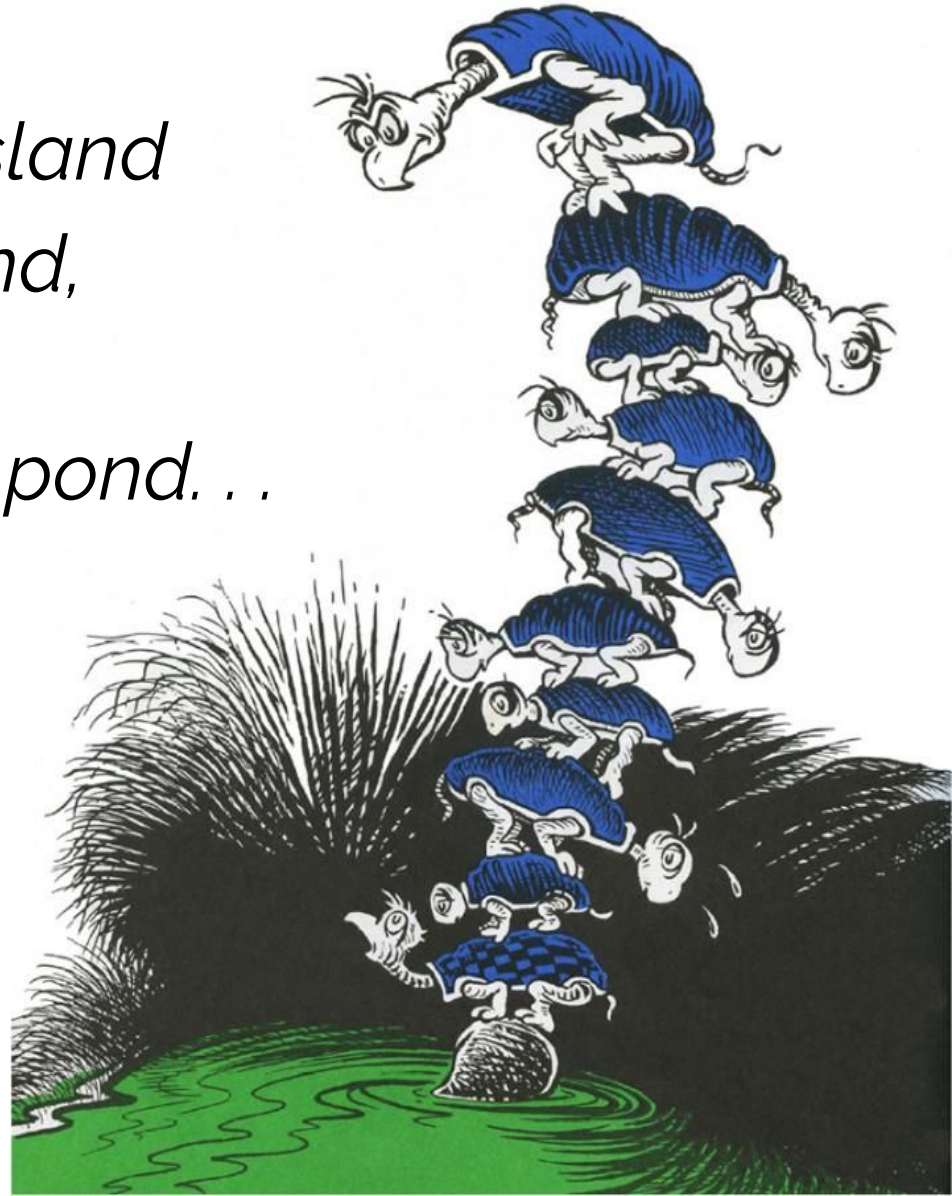
# Four Recipes for Hacking Statistics:

1. Direct Simulation ✓
2. Shuffling ✓
3. Bootstrapping
4. Cross Validation



# Yertle's Turtle Tower

*On the far-away island  
of Sala-ma-Sond,  
Yertle the Turtle  
was king of the pond. . .*



# How High can Yertle stack his turtles?

Observe 20 of Yertle's turtle towers . . .

# of turtles	48	24	32	61	51	12	32	18	19	24
	21	41	29	21	25	23	42	18	23	13

- What is the mean of the number of turtles in Yertle's stack?
- What is the uncertainty on this estimate?



# Classic Method:

Sample Mean:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i = 28.9$$

Standard Error of the Mean:

$$\sigma_{\bar{x}} = \frac{1}{\sqrt{N}} \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2} = 3.0$$

**What assumptions go into  
these formulae?**

**Can we use  
sampling instead?**

**Problem:**

**As before, we don't have a  
generating model . . .**

**Problem:**

**As before, we don't have a  
generating model . . .**

**Solution:**

**Bootstrap Resampling**

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

## Idea:

Simulate the distribution by *drawing samples with replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.



# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

# Idea:

Simulate the distribution  
by *drawing samples with  
replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

[illegible]

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

# Idea:

Simulate the distribution  
by *drawing samples with  
replacement*.

# Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

[illegible]

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

# Idea:

Simulate the distribution  
by *drawing samples with  
replacement*.

# Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

[illegible]

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

# Idea:

Simulate the distribution  
by *drawing samples with  
replacement*.

# Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

[illegible]

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

# Idea:

Simulate the distribution  
by *drawing samples with  
replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

[illegible]

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

# Idea:

Simulate the distribution  
by *drawing samples with  
replacement*.

# Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

[illegible]

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

# Idea:

Simulate the distribution  
by *drawing samples with  
replacement*.

# Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

[illegible]

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

# Idea:

Simulate the distribution  
by *drawing samples with  
replacement*.

# Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

[illegible]



# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

# Idea:

Simulate the distribution  
by *drawing samples with  
replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

[illegible]

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

# Idea:

Simulate the distribution  
by *drawing samples with  
replacement*.

# Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

[illegible]

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

# Idea:

Simulate the distribution  
by *drawing samples with  
replacement*.

# Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

[illegible]



# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

## Idea:

Simulate the distribution by *drawing samples with replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

21	19	25	24	23	19	41	23	41	18
61	12								

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

## Idea:

Simulate the distribution by *drawing samples with replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

21	19	25	24	23	19	41	23	41	18
61	12	42							

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

## Idea:

Simulate the distribution by *drawing samples with replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

21	19	25	24	23	19	41	23	41	18
61	12	42	42						

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

## Idea:

Simulate the distribution by *drawing samples with replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

21	19	25	24	23	19	41	23	41	18
61	12	42	42	42					



# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

## Idea:

Simulate the distribution by *drawing samples with replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

21	19	25	24	23	19	41	23	41	18
61	12	42	42	42	19				

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

## Idea:

Simulate the distribution by *drawing samples with replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

21	19	25	24	23	19	41	23	41	18
61	12	42	42	42	19	18			

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

## Idea:

Simulate the distribution by *drawing samples with replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

21	19	25	24	23	19	41	23	41	18
61	12	42	42	42	19	18	61		

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

## Idea:

Simulate the distribution by *drawing samples with replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

21	19	25	24	23	19	41	23	41	18
61	12	42	42	42	19	18	61	29	

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

## Idea:

Simulate the distribution by *drawing samples with replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

21	19	25	24	23	19	41	23	41	18
61	12	42	42	42	19	18	61	29	41

# Bootstrap Resampling:

48	24	51	12
21	41	25	23
32	61	19	24
29	21	23	13
32	18	42	18

## Idea:

Simulate the distribution by *drawing samples with replacement*.

## Motivation:

The data estimates its own distribution – we draw random samples from this distribution.

21	19	25	24	23	19	41	23	41	18
61	12	42	42	42	19	18	61	29	41

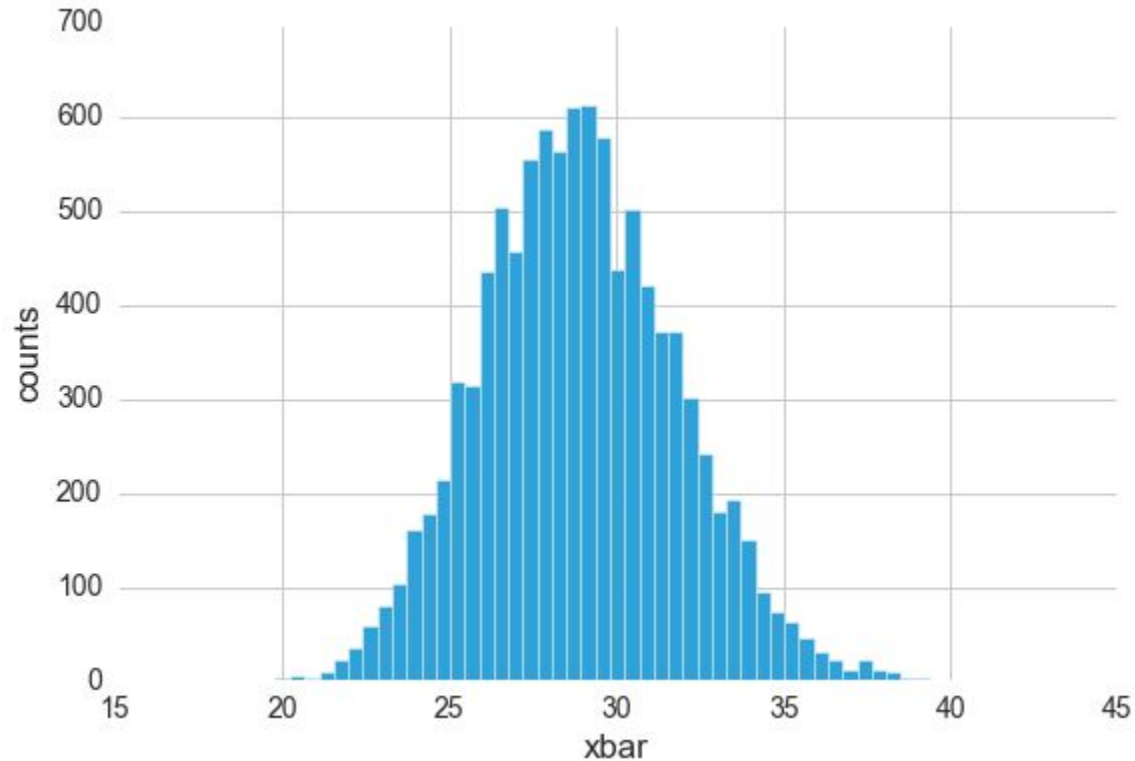
→ 31.05

**Repeat this  
several thousand times . . .**

# Recovers The Analytic Estimate!

```
for i in range(10000):  
    sample = N[randint(20, size=20)]  
    xbar[i] = mean(sample)  
mean(xbar), std(xbar)  
# (28.9, 2.9)
```

Height =  $29 \pm 3$  turtles

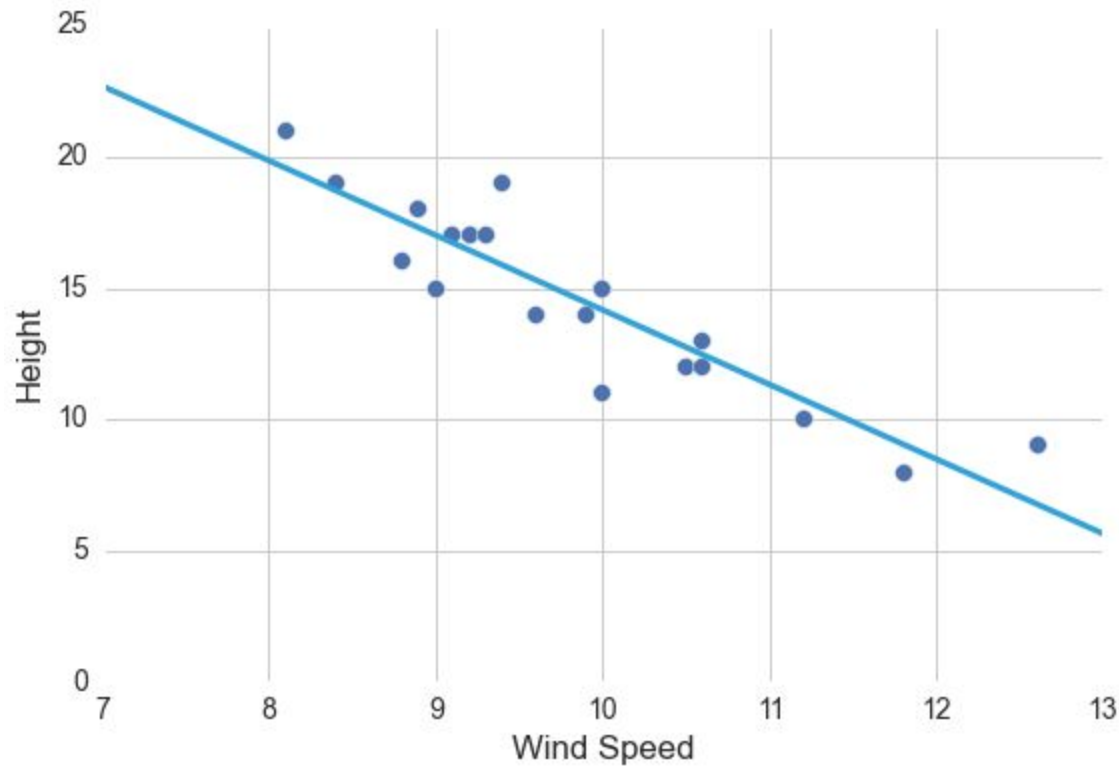




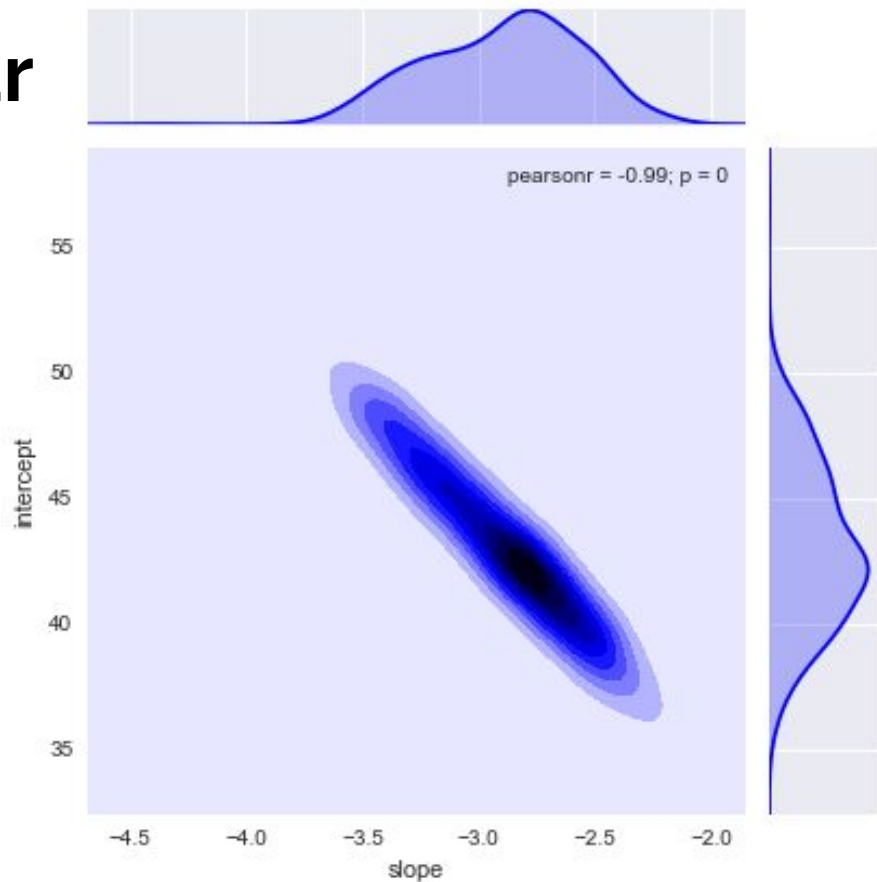
**Bootstrap sampling  
can be applied even to  
more involved statistics**

# Bootstrap on Linear Regression:

What is the relationship between speed of wind and the height of the Yertle's turtle tower?



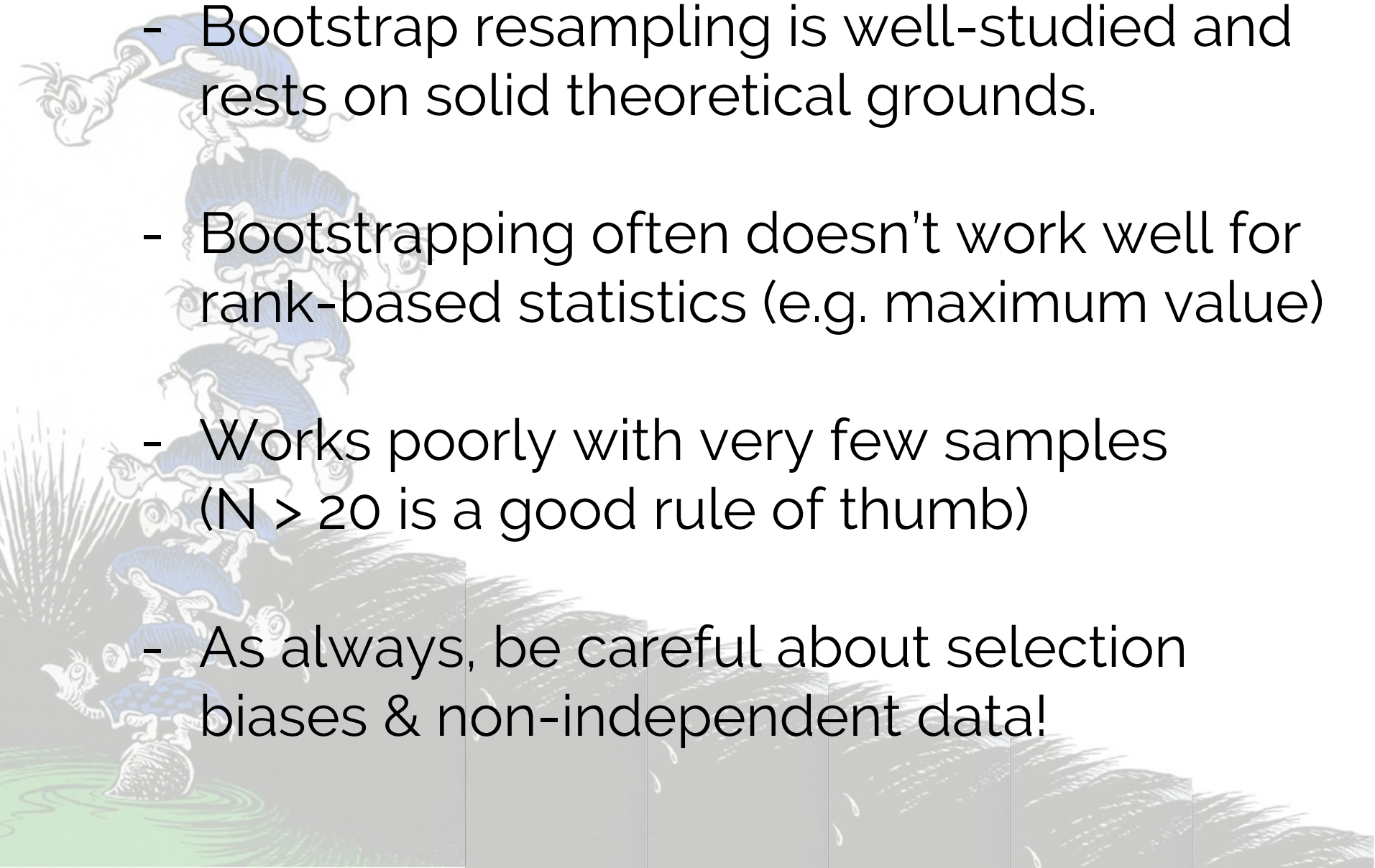
# Bootstrap on Linear Regression:



```
for i in range(10000):  
    i = randint(20, size=20)  
    slope, intercept = fit(x[i], y[i])  
    results[i] = (slope, intercept)
```

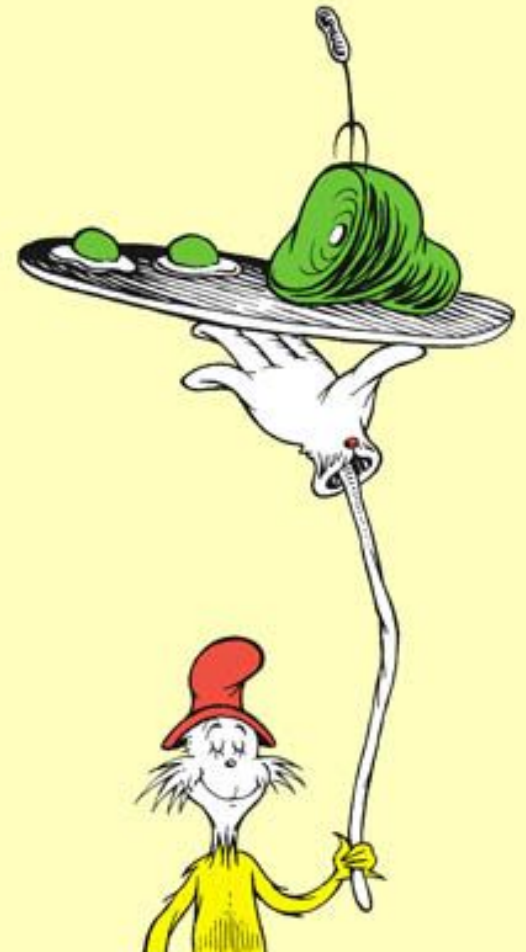
# Notes on Bootstrapping:

- Bootstrap resampling is well-studied and rests on solid theoretical grounds.
- Bootstrapping often doesn't work well for rank-based statistics (e.g. maximum value)
- Works poorly with very few samples ( $N > 20$  is a good rule of thumb)
- As always, be careful about selection biases & non-independent data!



# Four Recipes for Hacking Statistics:

1. Direct Simulation ✓
2. Shuffling ✓
3. Bootstrapping ✓
4. Cross Validation

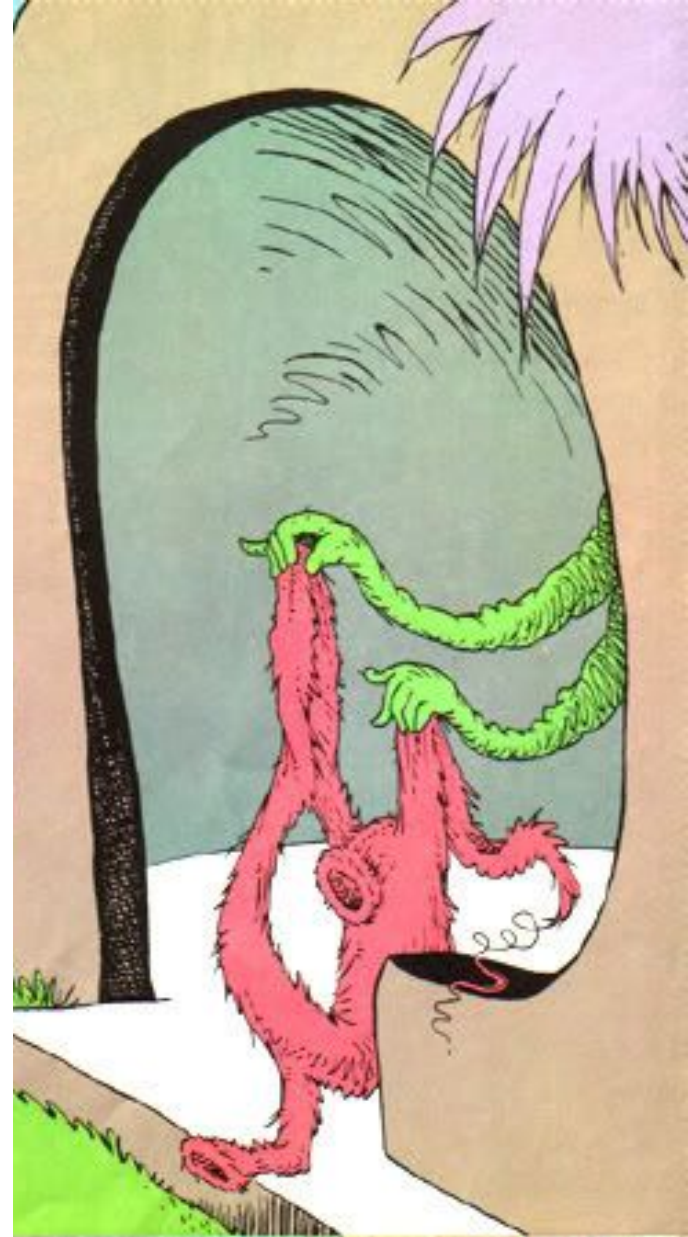


# Onceler Industries: Sales of Thneeds

*I'm being quite useful!*

*This thing is a Thneed.*

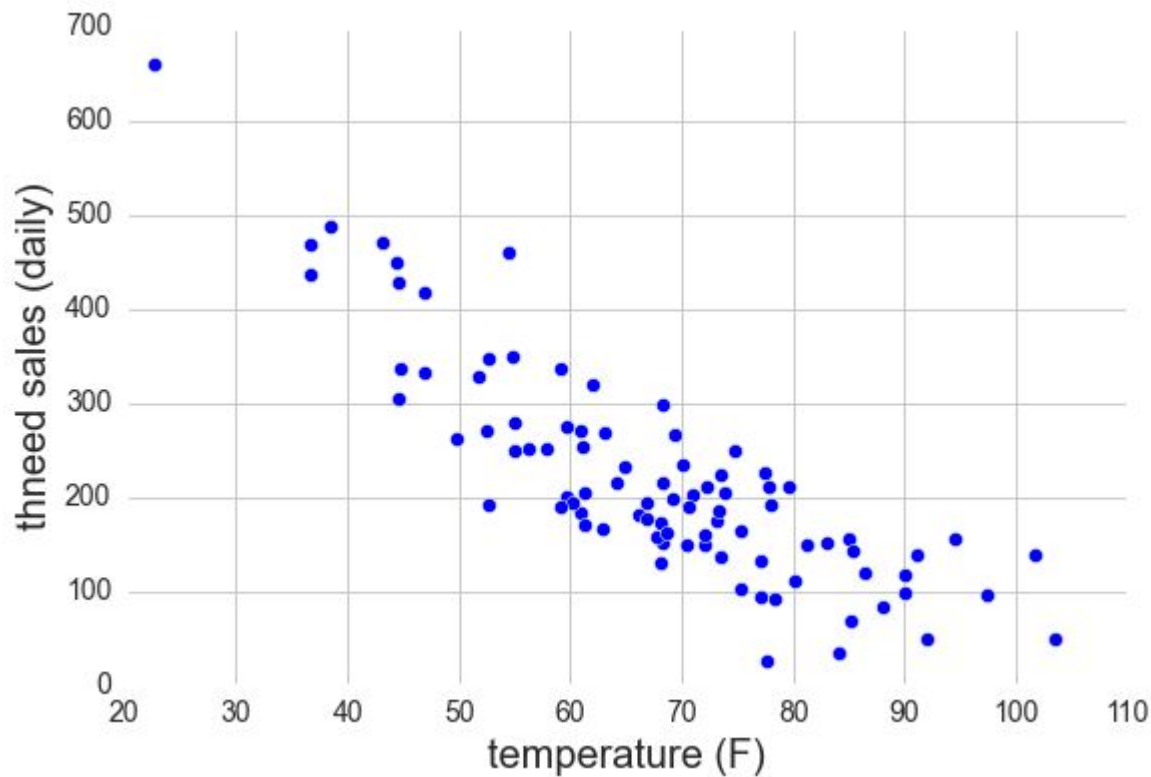
*A Thneed's a Fine-Something-  
That-All-People-Need!*





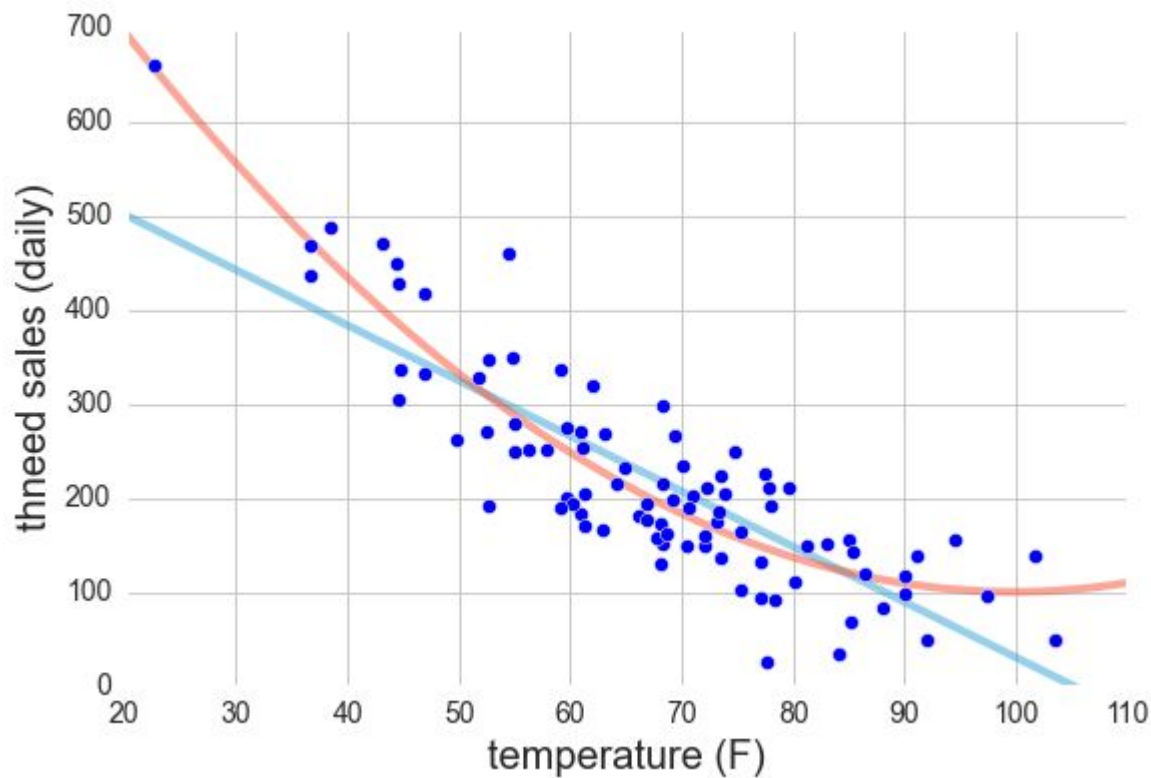


Thneed sales seem to show a trend with temperature . . .





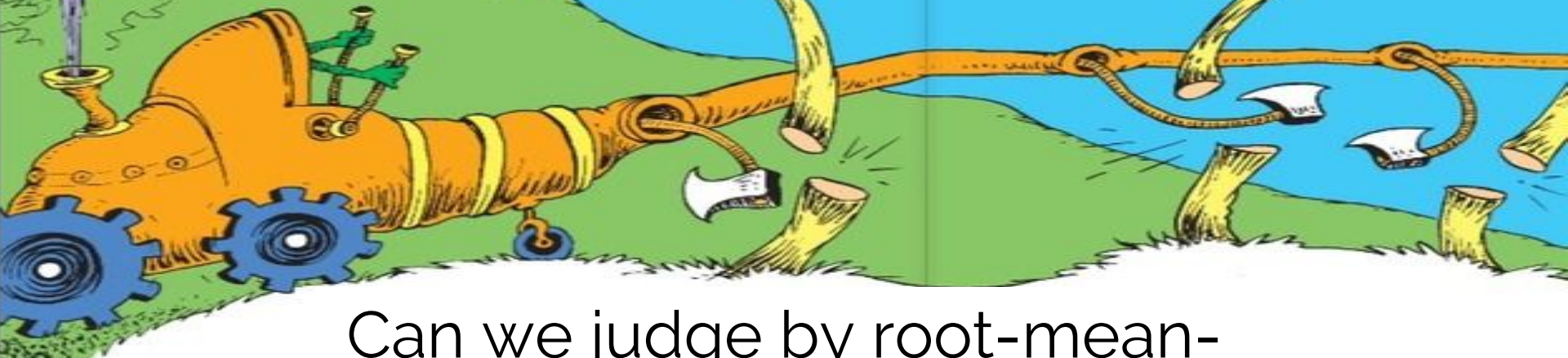
But which model is a better fit?



$$y = a + bx$$

$$y = a + bx + cx^2$$

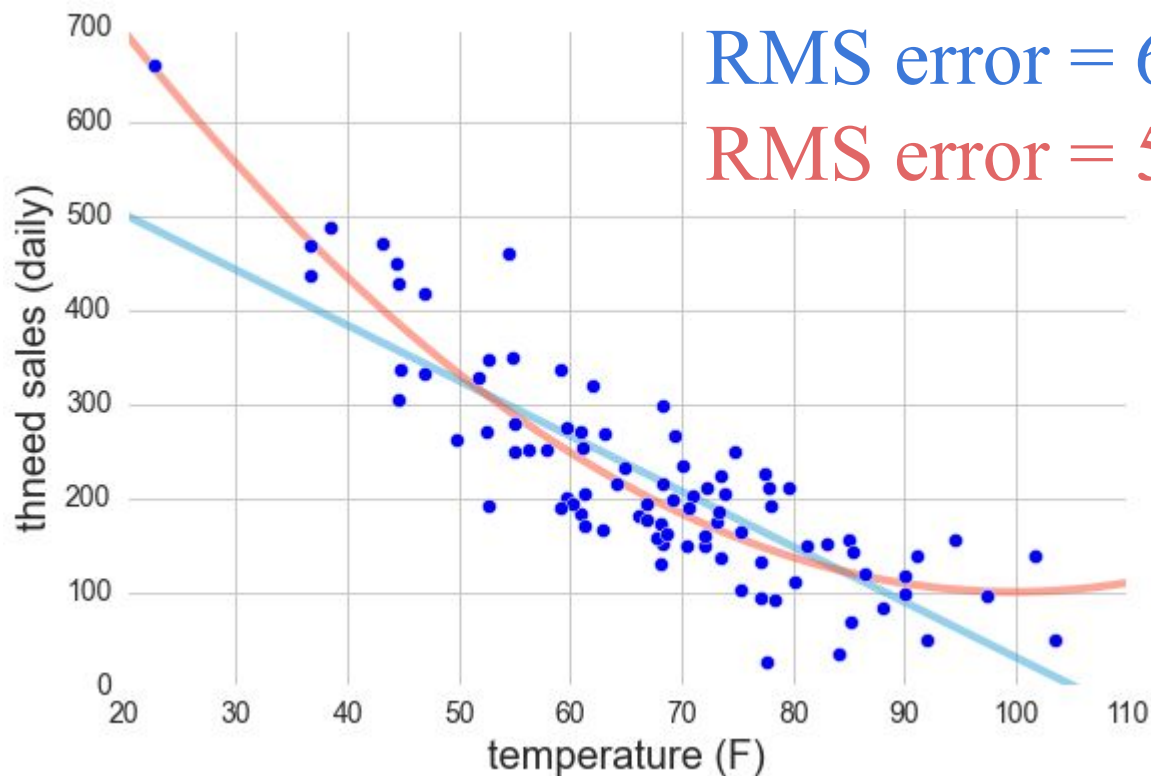




Can we judge by root-mean-square error?

RMS error = 63.0

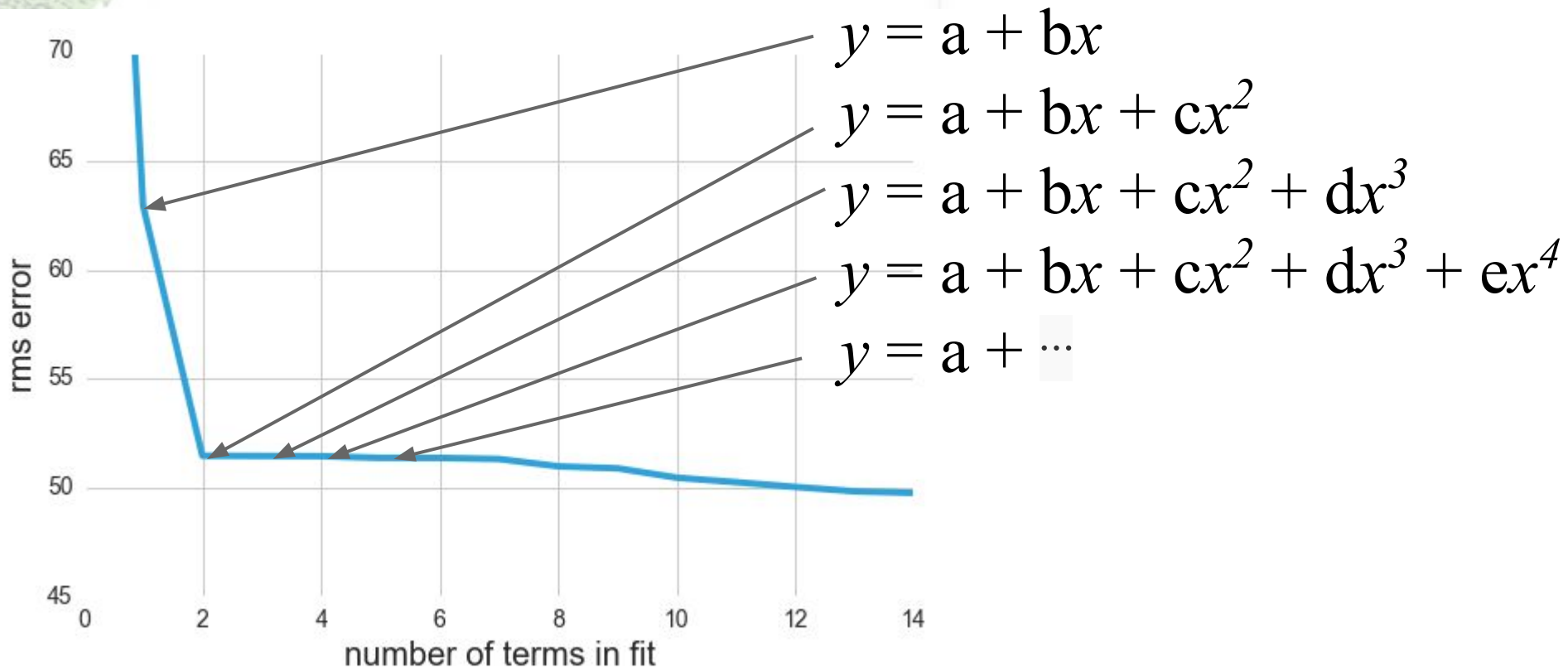
RMS error = 51.5



$$y = a + bx$$

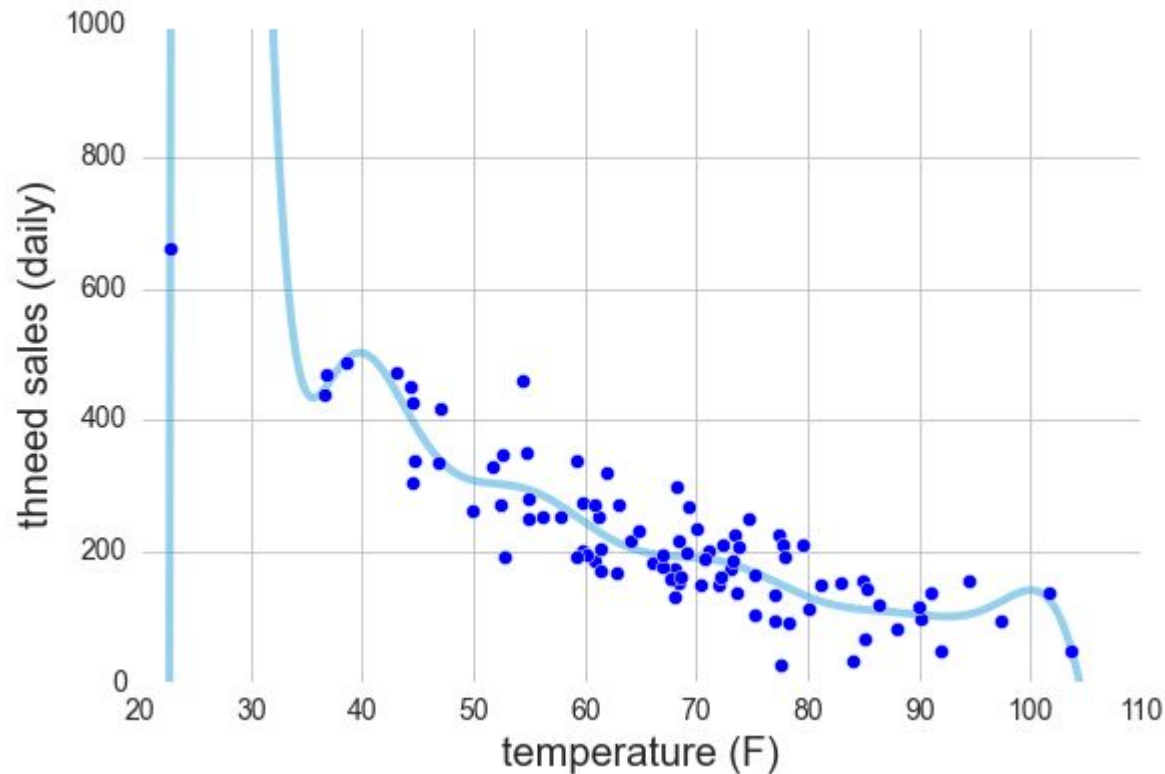
$$y = a + bx + cx^2$$

In general, more flexible models will *always* have a lower RMS error.



# RMS error does not tell the whole story.

$$y = a + bx + cx^2 + dx^3 + ex^4 + fx^5 + \dots + nx^{14}$$





**Not to worry:  
Statistics has figured this out.**



# Classic Method

A whimsical illustration at the top of the slide depicts a mechanical device on the left with large blue gears and a yellow cylindrical body. To its right, a hammer with a yellow handle and a silver head is shown in the process of chopping a tree stump. Several other tree stumps are scattered across a green grassy field under a light blue sky.

Difference in Mean  
Squared Error follows  
chi-square distribution:

$$p(x; \nu) = \frac{1}{2^{\nu/2} \Gamma\left(\frac{\nu}{2}\right)} x^{\frac{\nu}{2}-1} e^{-\frac{x}{2}}$$



# Classic Method

Difference in Mean  
Squared Error follows  
chi-square distribution:

$$p(x; \nu) = \frac{1}{2^{\nu/2} \Gamma\left(\frac{\nu}{2}\right)} x^{\frac{\nu}{2}-1} e^{-\frac{x}{2}}$$

Can estimate degrees of  
freedom easily because  
the models are *nested* ...

$$\nu \approx \nu_2 - \nu_1$$

$$\nu_2 \approx (N - d_2)$$

$$\nu_1 \approx (N - d_1)$$

# Classic Method

A whimsical illustration at the top of the slide depicts a mechanical device on the left, featuring large blue gears and a yellow cylindrical component. To its right, a hammer with a long wooden handle and a metal head is shown in the process of chopping through a tree trunk. Several other tree stumps are visible in the background, suggesting a forest being cleared. The scene is set against a light blue sky and green grass.

Difference in Mean  
Squared Error follows  
chi-square distribution:

$$p(x; \nu) = \frac{1}{2^{\nu/2} \Gamma\left(\frac{\nu}{2}\right)} x^{\frac{\nu}{2}-1} e^{-\frac{x}{2}}$$

Can estimate degrees of  
freedom easily because  
the models are *nested* . . .

$$\nu \approx \nu_2 - \nu_1$$

$$\nu_2 \approx (N - d_2)$$

$$\nu_1 \approx (N - d_1)$$

Plug in our numbers . . .



# Classic Method

**Wait... what question  
were we trying to  
answer again?**

Difference in Mean  
Squared Error follows  
chi-square distribution

Can estimate degrees of  
freedom easily because  
the models are *nested* ...

Plug in our numbers ...

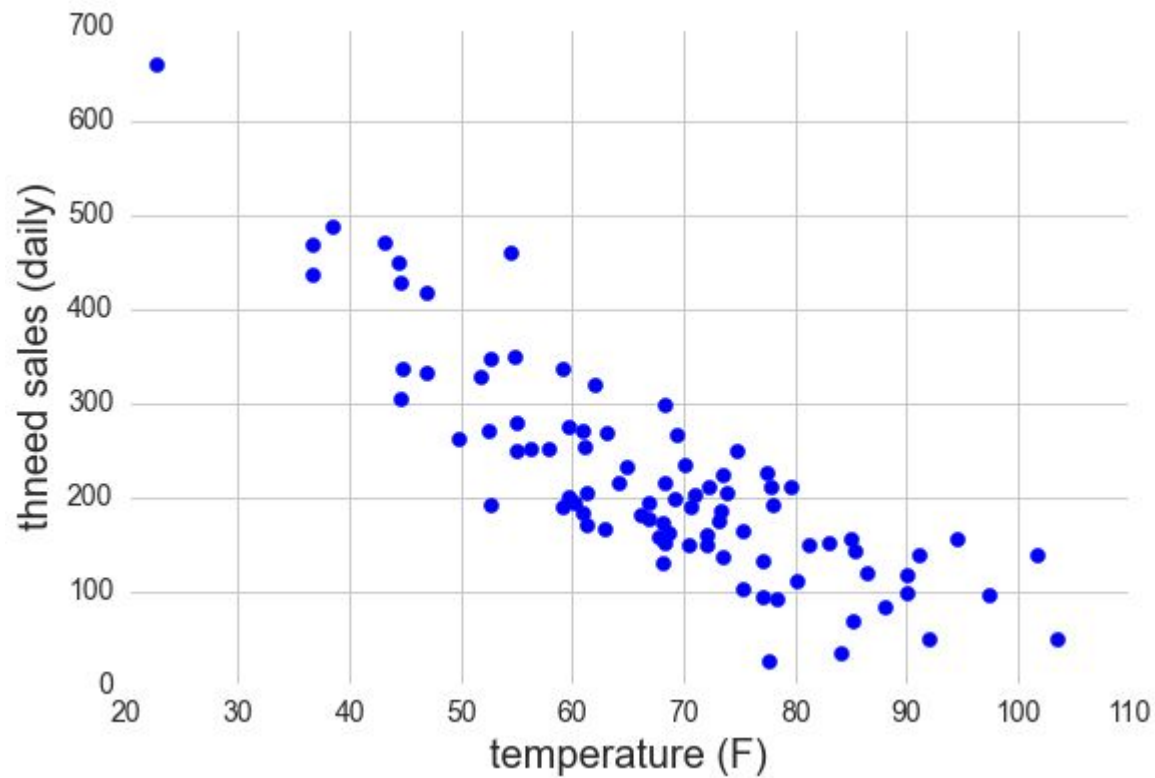
$$p(x; \nu) = \frac{1}{\Gamma(\frac{\nu}{2}) 2^{\frac{\nu}{2}}} x^{\frac{\nu}{2}-1} e^{-\frac{x}{2}}$$

$$\begin{aligned} N &\approx \nu_2 - \nu_1 \\ \nu_2 &\approx (N - d_2) \\ \nu_1 &\approx (N - d_1) \end{aligned}$$



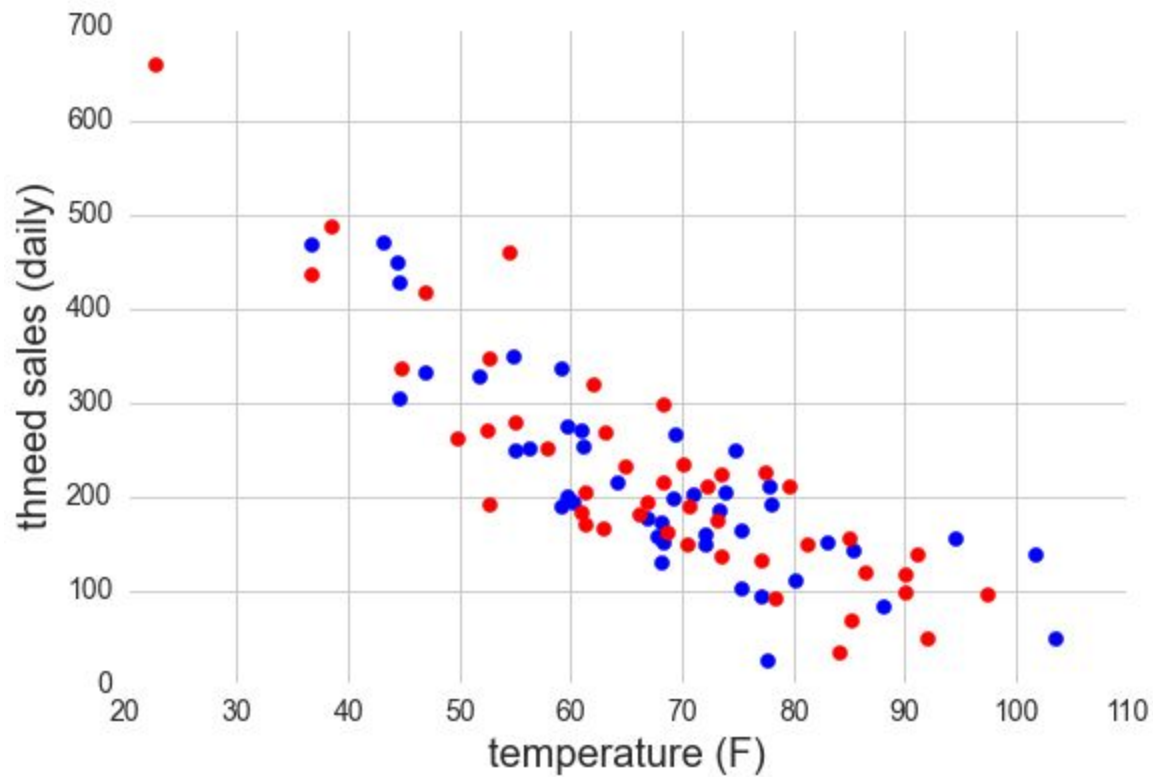
# **Another Approach: Cross Validation**

# Cross-Validation



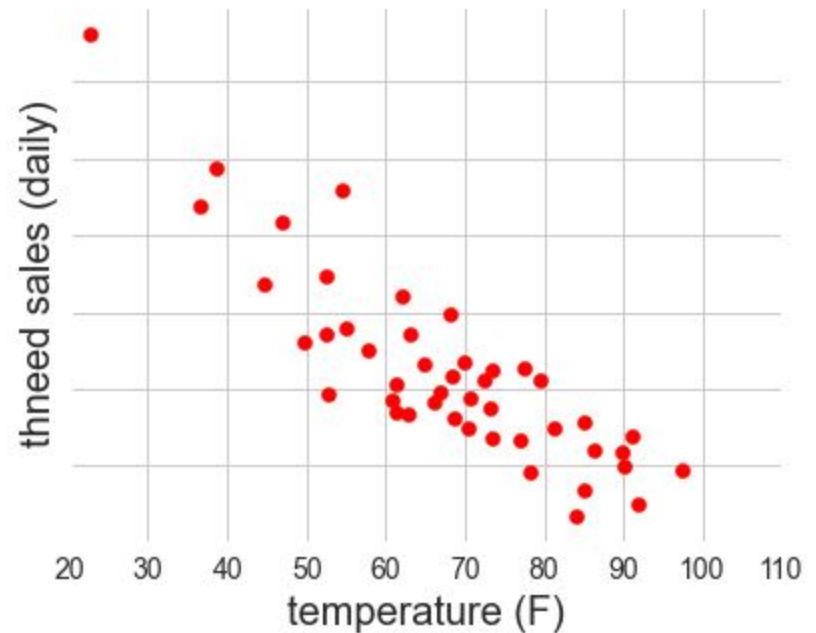
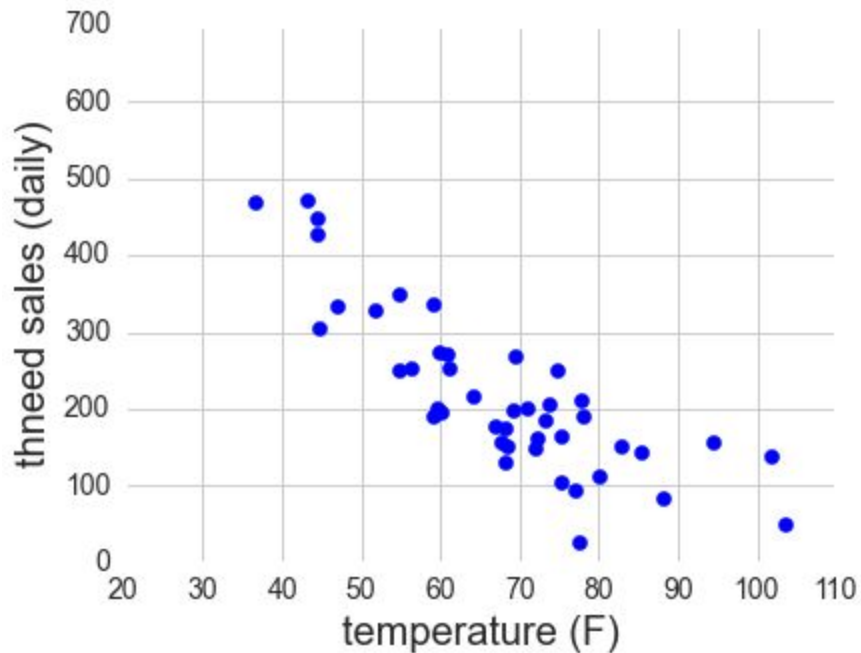
# Cross-Validation

## 1. Randomly Split data



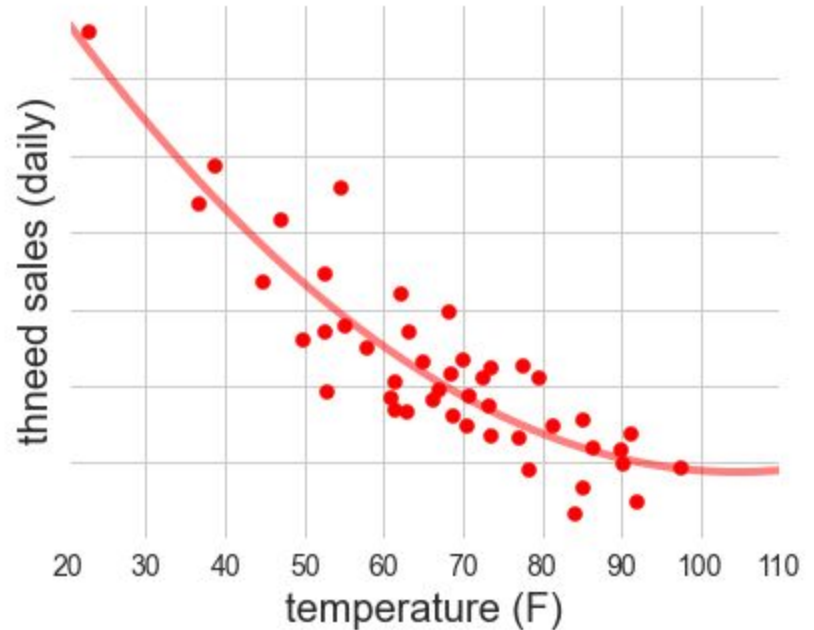
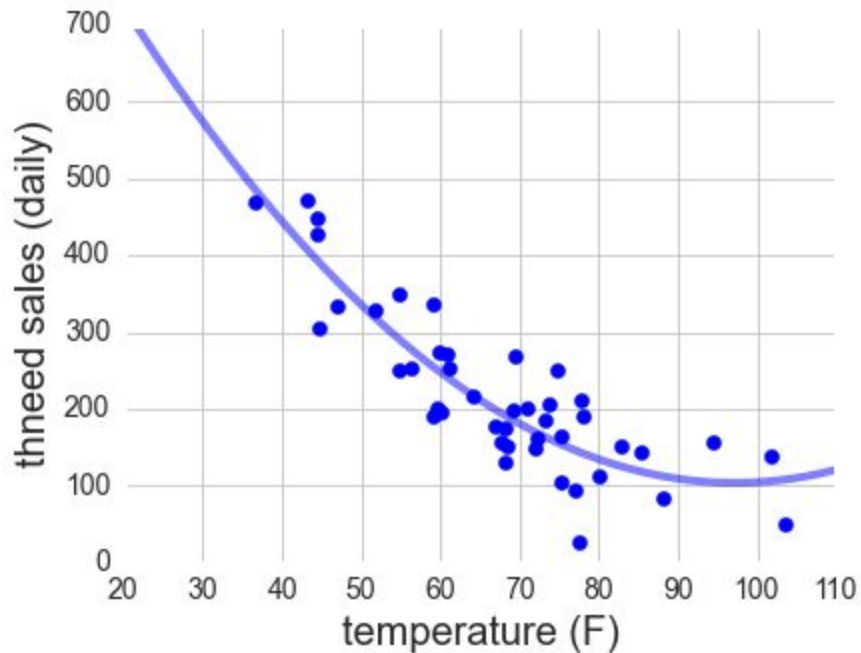
# Cross-Validation

## 1. Randomly Split data



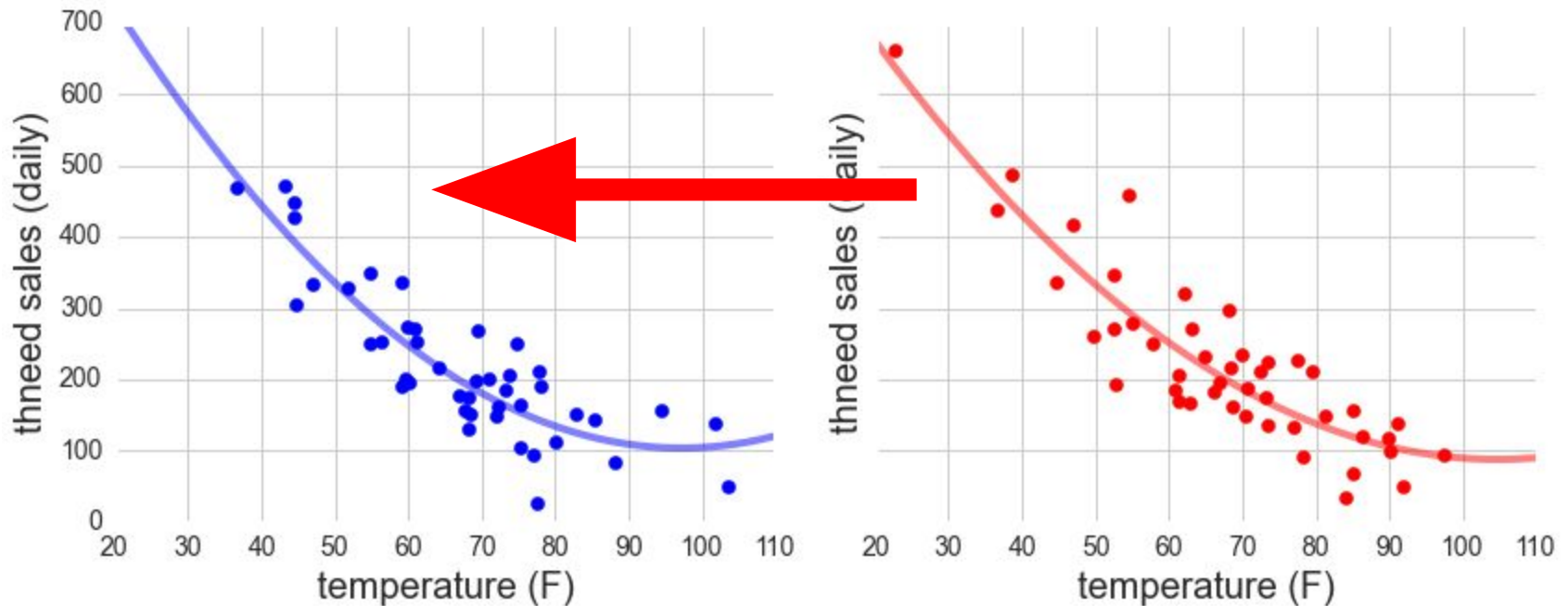
# Cross-Validation

2. Find the best model for each subset



# Cross-Validation

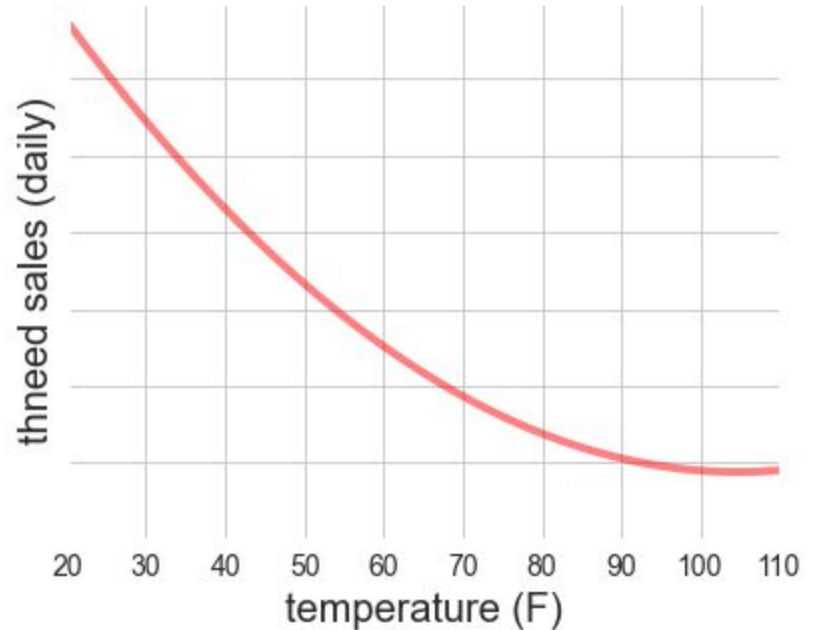
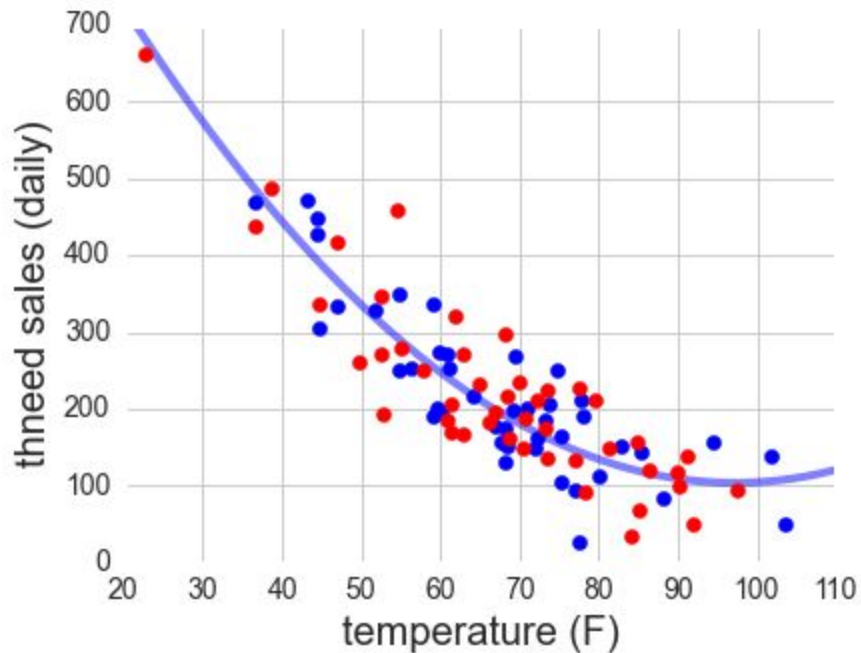
## 3. Compare models across subsets





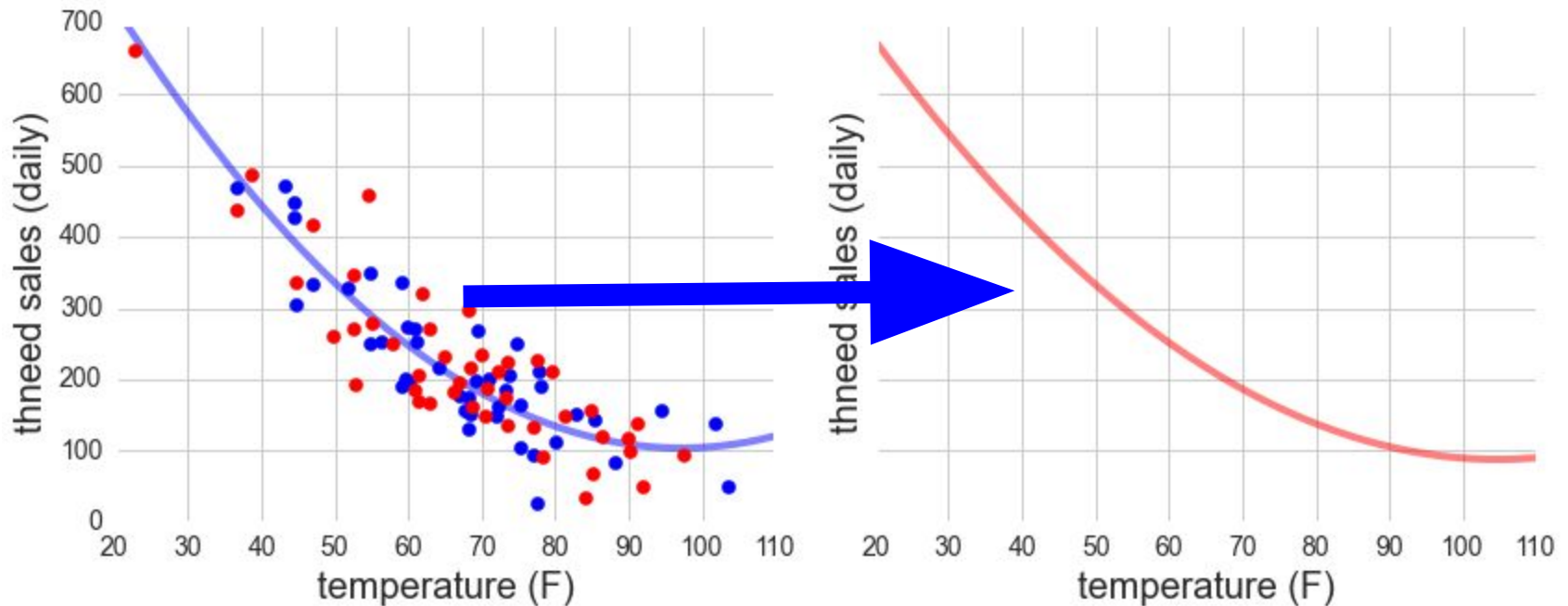
# Cross-Validation

## 3. Compare models across subsets



# Cross-Validation

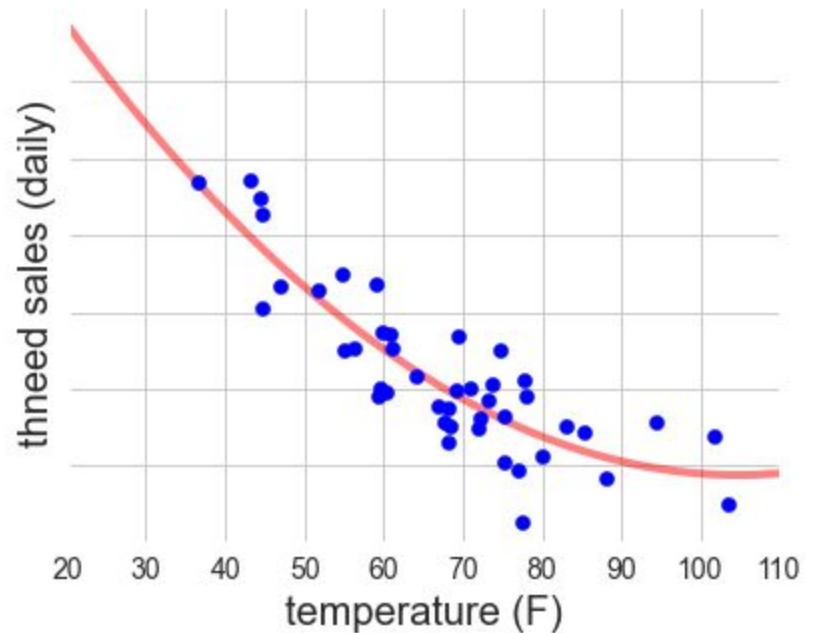
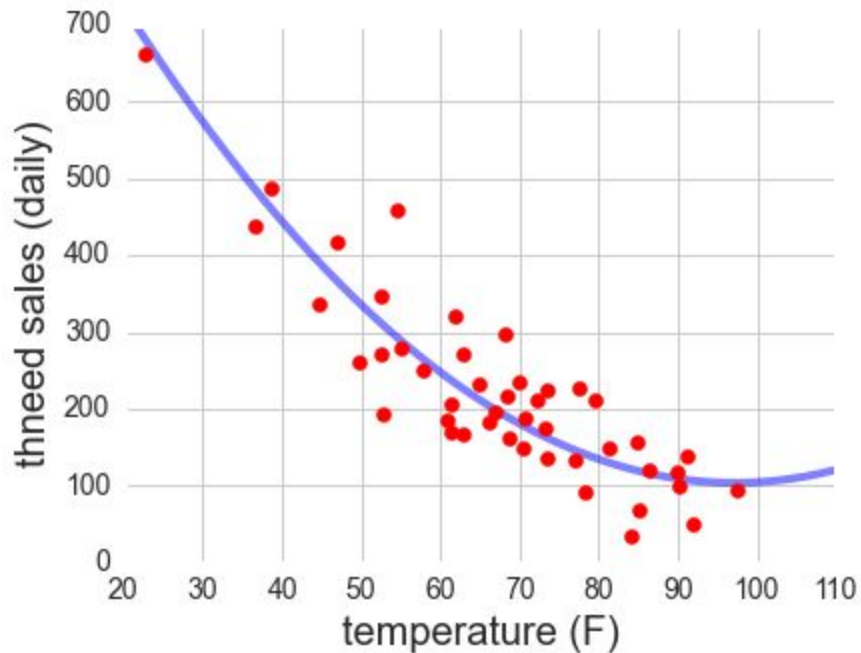
## 3. Compare models across subsets





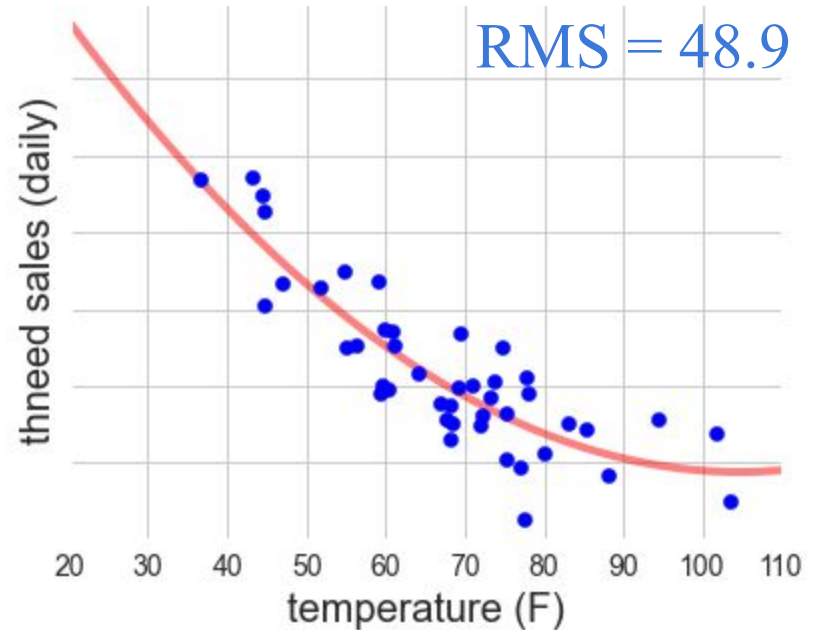
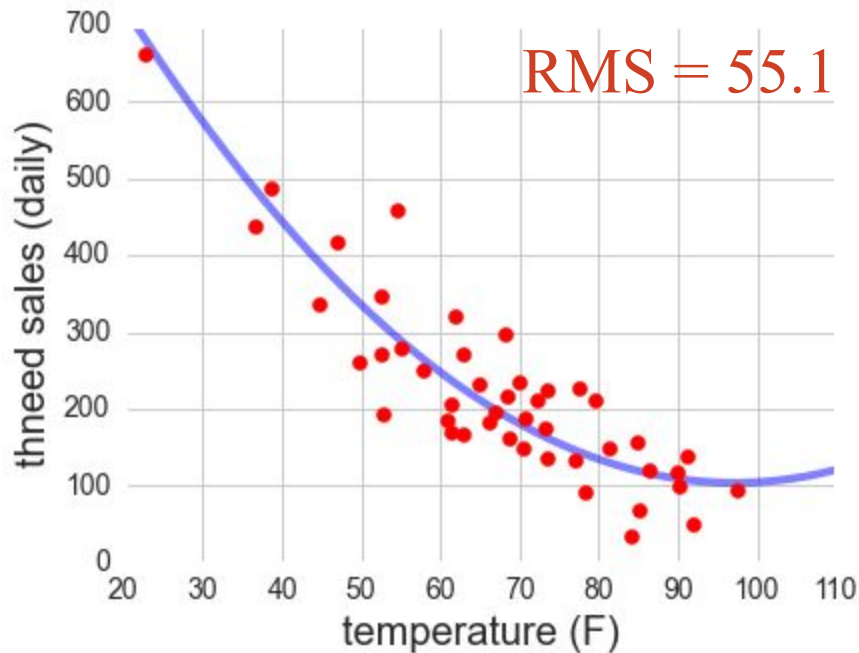
# Cross-Validation

## 3. Compare models across subsets



# Cross-Validation

4. Compute RMS error for each



RMS estimate = 52.1

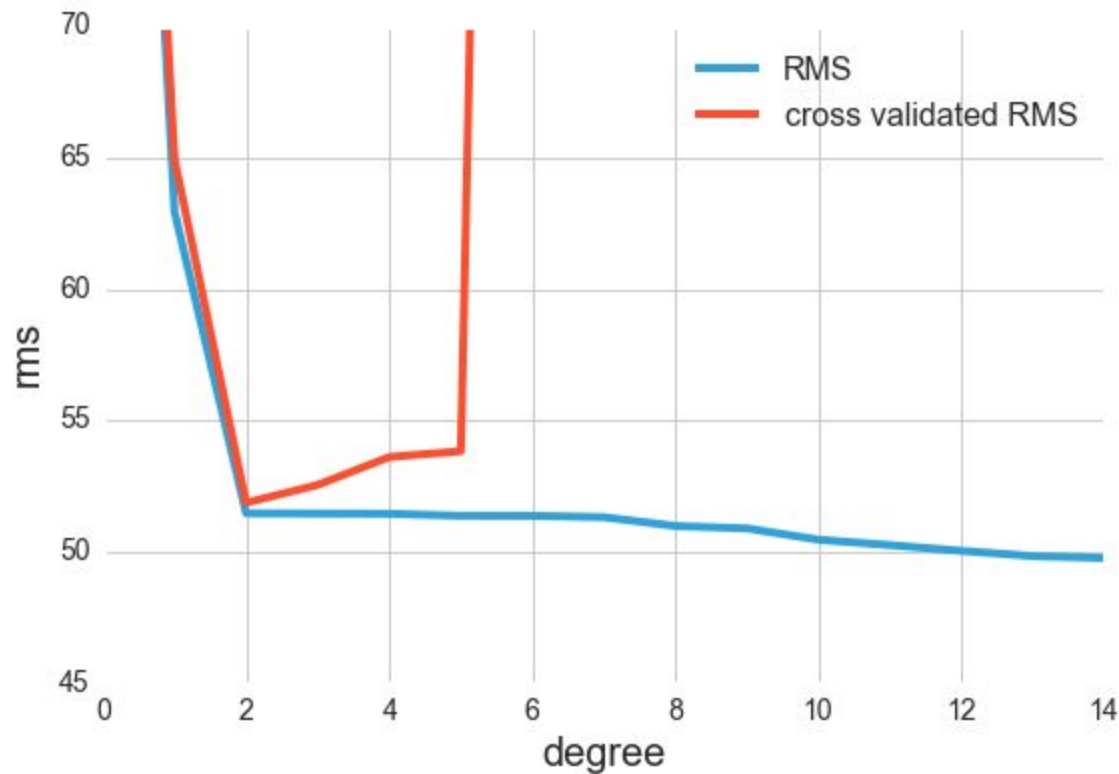


# Cross-Validation

**Repeat for as long as  
you have patience . . .**

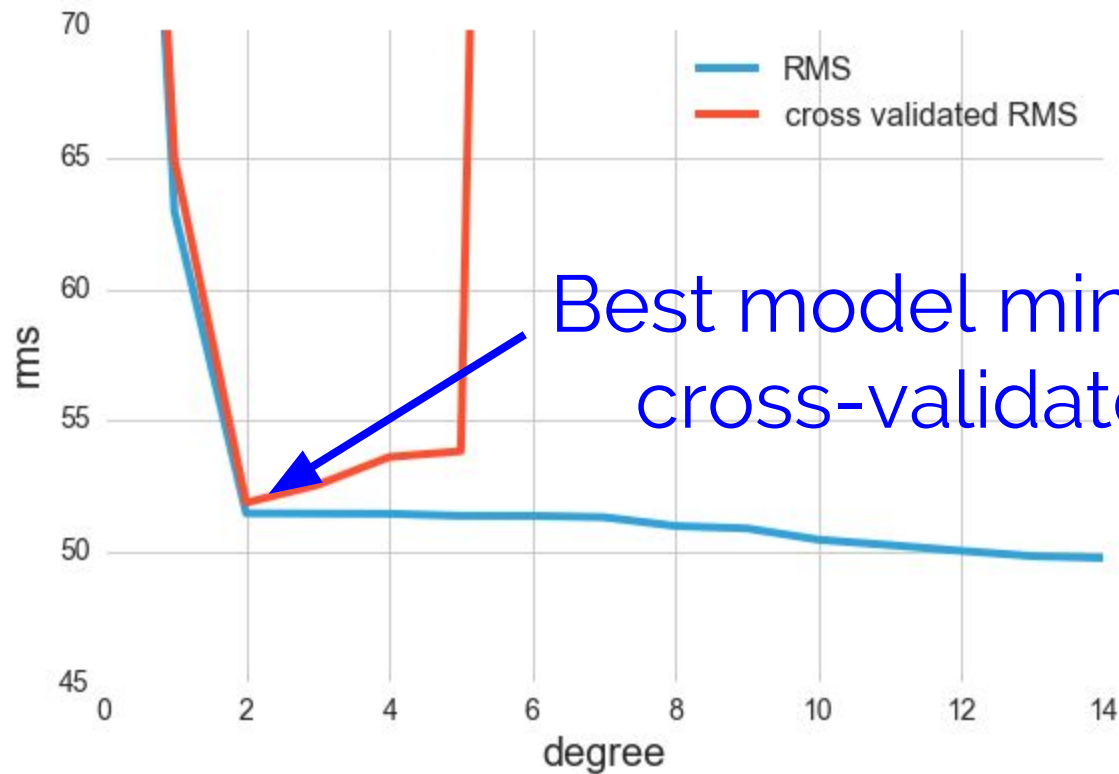
# Cross-Validation

5. Compare cross-validated RMS for models:



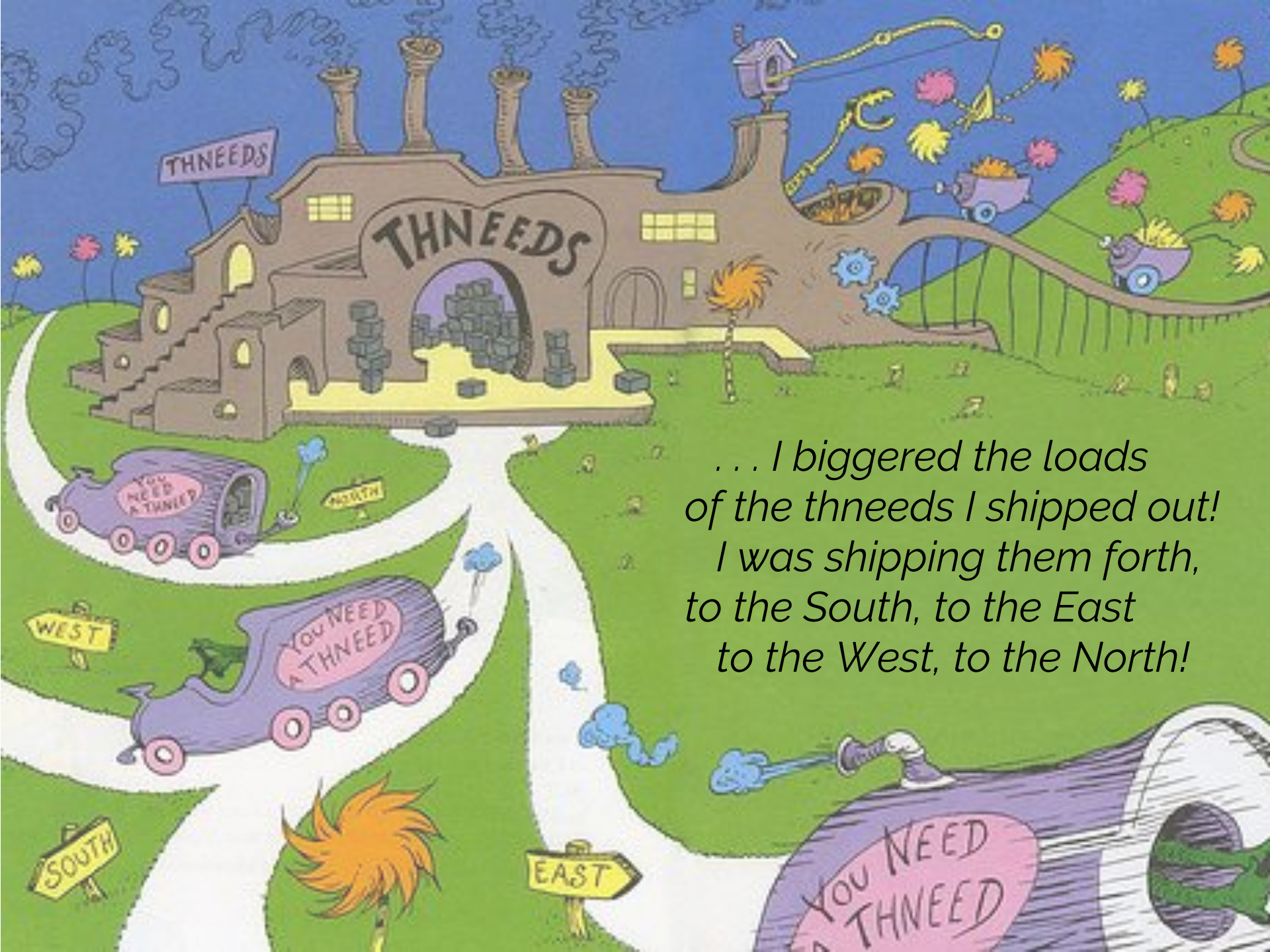
# Cross-Validation

5. Compare cross-validated RMS for models:



Best model minimizes the cross-validated error.





*... I biggered the loads  
of the thneeds I shipped out!  
I was shipping them forth,  
to the South, to the East  
to the West, to the North!*

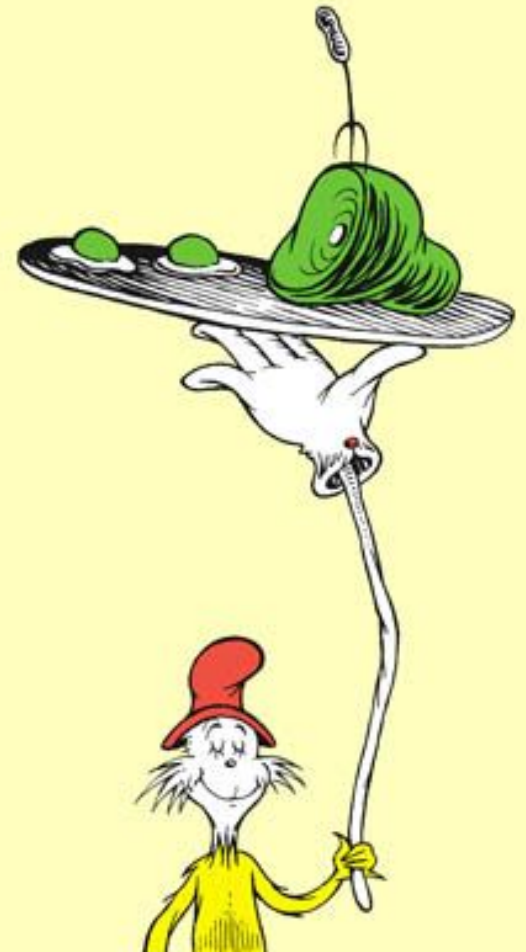


# Notes on Cross-Validation:

- This was “**2-fold**” **cross-validation**; other CV schemes exist & may perform better for your data (see e.g. scikit-learn docs)
- Cross-validation is the go-to method for model evaluation in **machine learning**, as statistics of the models are often not known in the classical sense.
- Again: caveats about selection bias and independence in data.

# Four Recipes for Hacking Statistics:

1. Direct Simulation ✓
2. Shuffling ✓
3. Bootstrapping ✓
4. Cross Validation ✓





# Sampling Methods

allow you to use intuitive **computational** approaches in place of often non-intuitive statistical rules.

If you can write a for-loop  
you can do statistical analysis.

# Things I didn't have time for:

- **Bayesian Methods:** very intuitive & powerful approaches to more sophisticated modeling.  
(see e.g. *Bayesian Methods for Hackers* by Cam Davidson-Pilon)
- **Selection Bias:** if you get data selection wrong, you'll have a bad time.  
(See Chris Fonnesbeck's Scipy 2015 talk, *Statistical Thinking for Data Science*)
- **Detailed considerations** on use of sampling, shuffling, and bootstrapping.  
(I recommend *Statistics Is Easy* by Shasha & Wilson  
And *Resampling: The New Statistics* by Julian Simon)

Sometimes the  
questions are  
complicated  
and the  
answers are  
simple.



- Dr. Seuss (attr)