

19: Sampling and the Bootstrap

Lisa Yan

May 18, 2020

Quick slide reference

3	Sampling definitions	19a_intro
11	Unbiased estimators	19b_sample_stats
23	Reporting estimation error	19c_statistical_error
29	Bootstrap: Sample mean	19d_bootstrap_mean
40	Bootstrap: Sample variance	LIVE
*	Bootstrap: Hypothesis testing	LIVE

Sampling definitions

Motivating example

You want to know the true mean and variance of happiness in Bhutan.

- But you can't ask everyone.
- You poll 200 random people.
- Your data looks like this:

Happiness = {72, 85, 79, 91, 68, ..., 71}

- The mean of all these numbers is 83.

Is this the **true mean happiness** of Bhutanese people?



Population



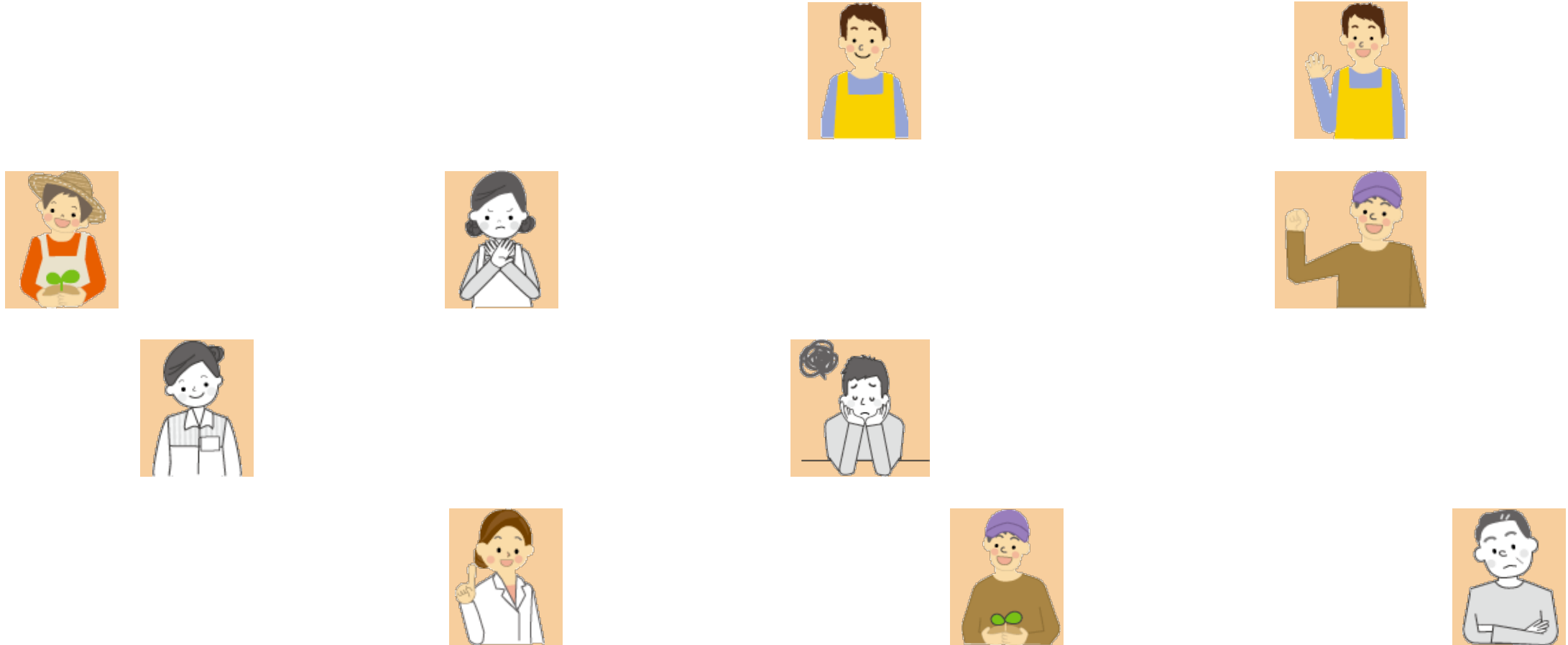
This is a **population**.

Sample



A **sample** is selected from a population.

Sample



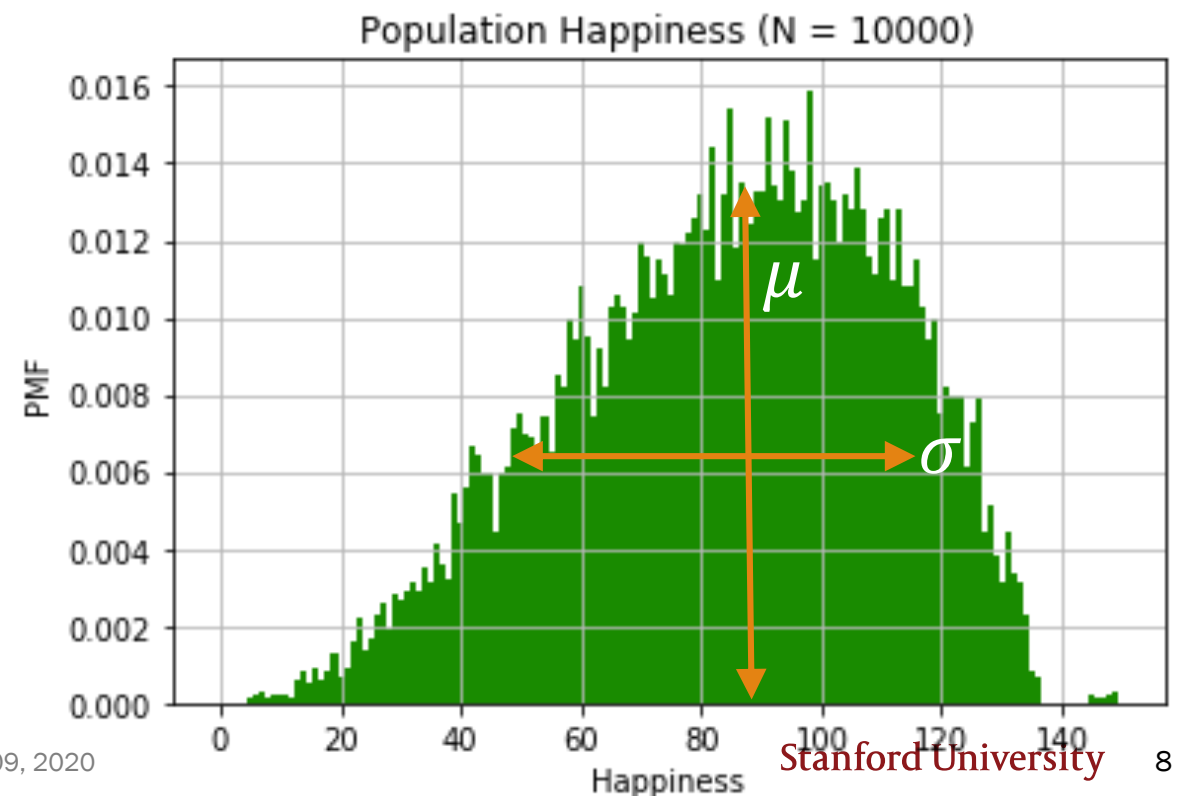
A **sample** is selected from a population.

A sample, mathematically

Consider n random variables X_1, X_2, \dots, X_n .

The sequence X_1, X_2, \dots, X_n is a **sample** from distribution F if:

- X_i are all independent and identically distributed (i.i.d.)
- X_i all have same distribution function F (the **underlying distribution**), where $E[X_i] = \mu$, $\text{Var}(X_i) = \sigma^2$



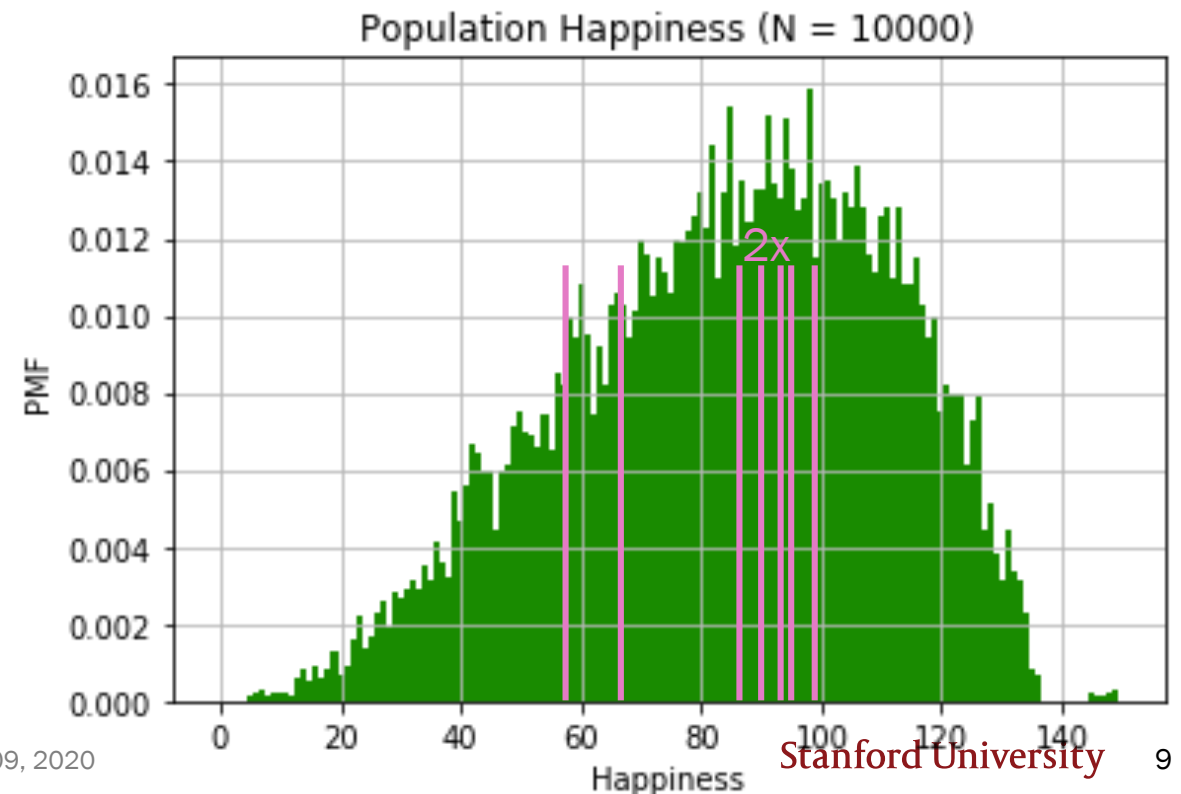
A sample, mathematically

A sample of **sample size 8**:

$(X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8)$

A **realization** of a sample of size 8:

$(59, 87, 94, 99, 87, 78, 69, 91)$



A single sample

If we had a distribution F of our entire population, we could compute exact statistics about about happiness.



A happy
Bhutanese person

But we only have 200 people (a sample).

Today: If we only have a single sample,

- How do we report *estimated* statistics?
- How do we report estimated error of these estimates?
- How do we perform hypothesis testing?

Unbiased estimators

A single sample

If we had a distribution F of our entire population, we could compute exact statistics about about happiness.



A happy
Bhutanese person

But we only have 200 people (a sample).

So these population statistics are unknown:

- μ , the **population mean**
- σ^2 , the **population variance**

A single sample

If we had a distribution F of our entire population, we could compute exact statistics about about happiness.



A happy
Bhutanese person

But we only have 200 people (a sample).

- From these 200 people, what is our best estimate of **population mean** and **population variance**?
- How do we define best estimate?

Estimating the population mean



1. What is our best estimate of μ , the **mean happiness** of Bhutanese people?

If we only have a sample, (X_1, X_2, \dots, X_n) :

The best estimate of μ is the **sample mean**:

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$$

\bar{X} is an unbiased estimator of the population mean μ . $E[\bar{X}] = \mu$

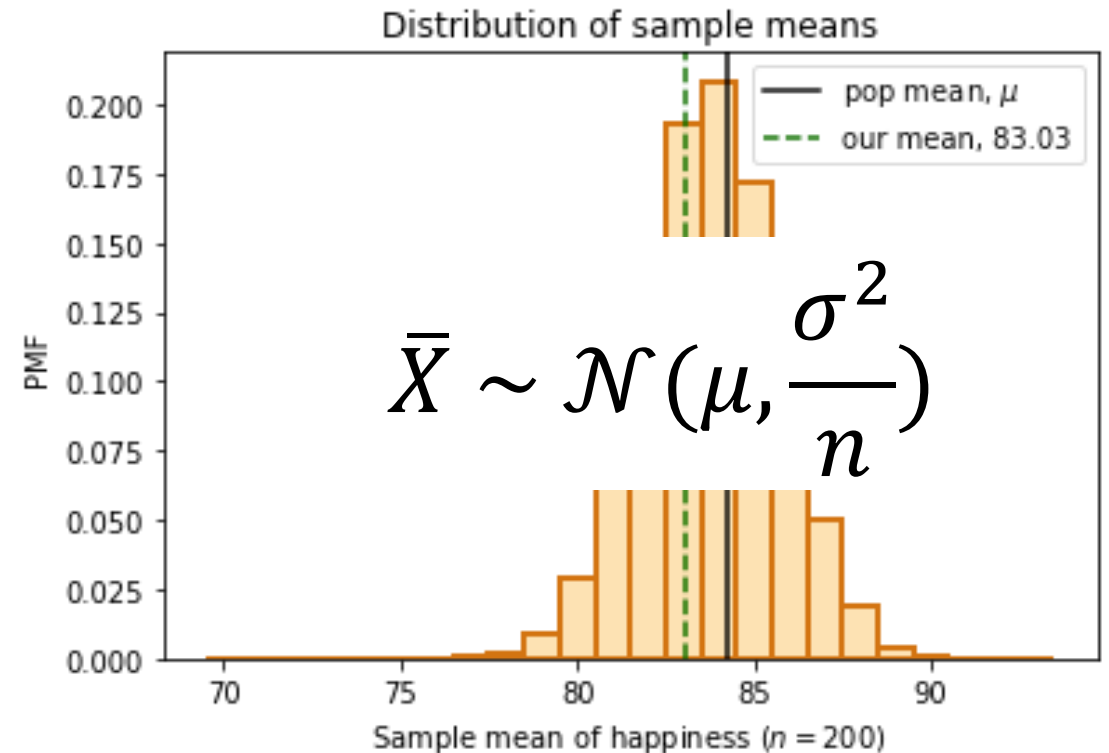
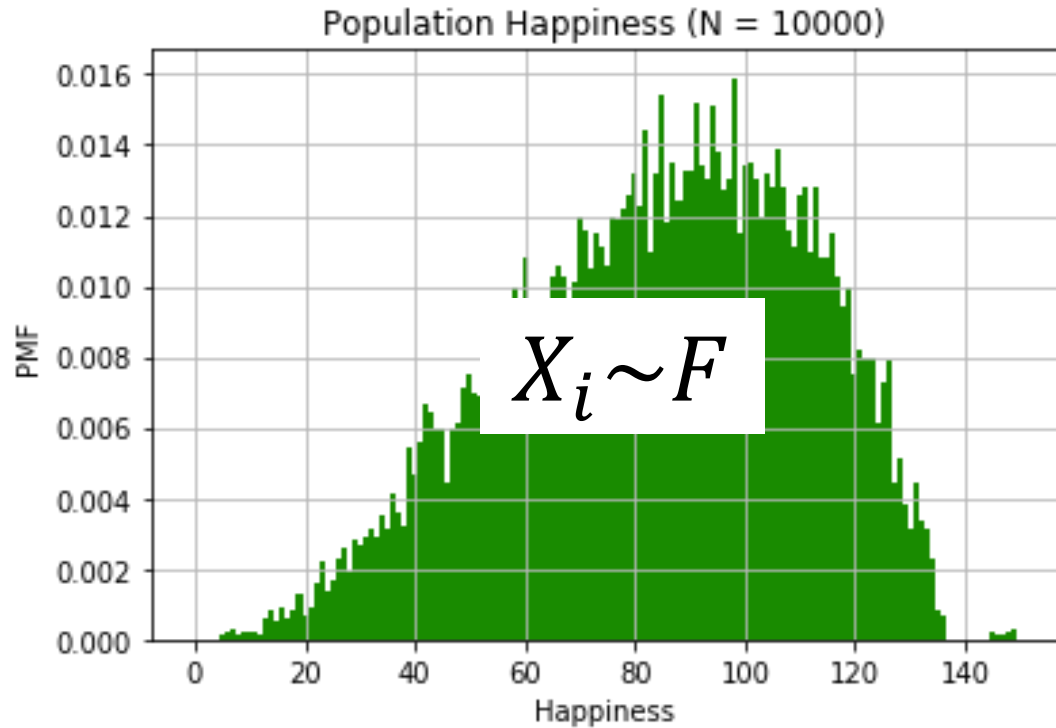
Intuition: By the CLT, $\bar{X} \sim \mathcal{N}\left(\mu, \frac{\sigma^2}{n}\right)$



If we could take *multiple* samples of size n :

1. For each sample, compute sample mean
2. On average, we would get the population mean

Sample mean



Even if we can't report μ , we can report our sample mean 83.03, which is an unbiased estimate of μ .

Estimating the population variance



2. What is σ^2 , the **variance of happiness** of Bhutanese people?

If we knew the entire population (x_1, x_2, \dots, x_N) :

population variance

$$\sigma^2 = E[(X - \mu)^2] = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2$$

population mean

If we only have a sample, (X_1, X_2, \dots, X_n) :

sample variance

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2$$

sample mean

Intuition about the sample variance, S^2

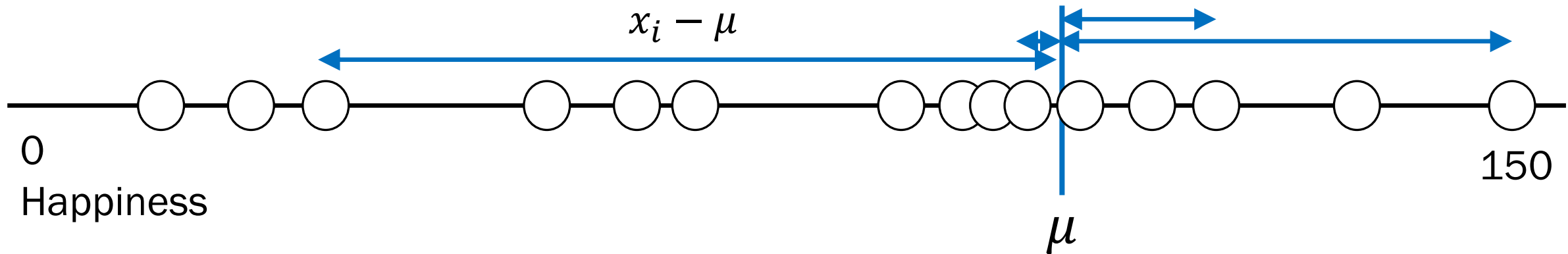
Actual, σ^2

population
variance

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2$$

population mean

$x_i - \mu$



Population size, N

Calculating population statistics exactly requires us knowing all N datapoints.

Intuition about the sample variance, S^2

Actual, σ^2

Estimate, S^2

population
variance

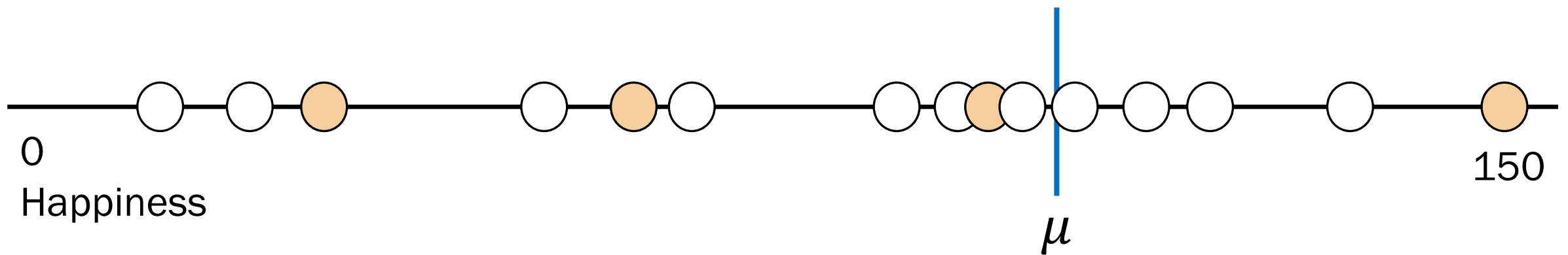
$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2$$

population mean

sample
variance

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2$$

sample mean



Population size, N

Intuition about the sample variance, S^2

Actual, σ^2

Estimate, S^2

population variance

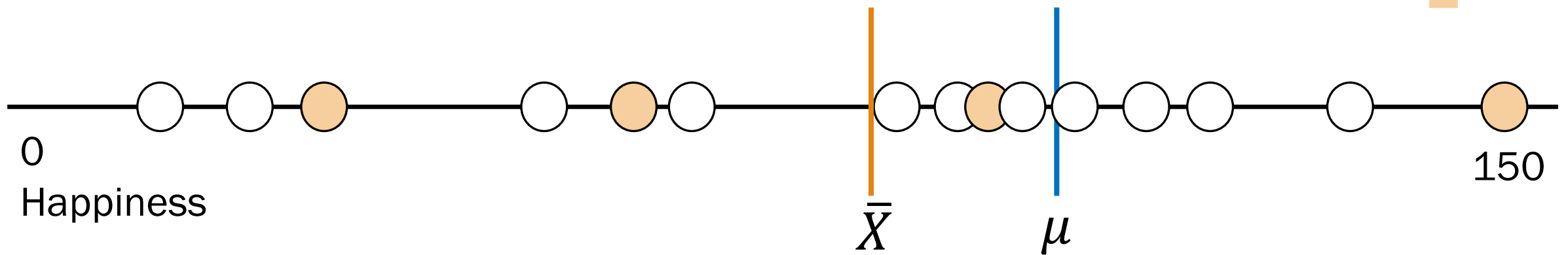
population mean

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2$$

sample variance

sample mean

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2$$



Population size, N

Intuition about the sample variance, S^2

Actual, σ^2

Estimate, S^2

population variance

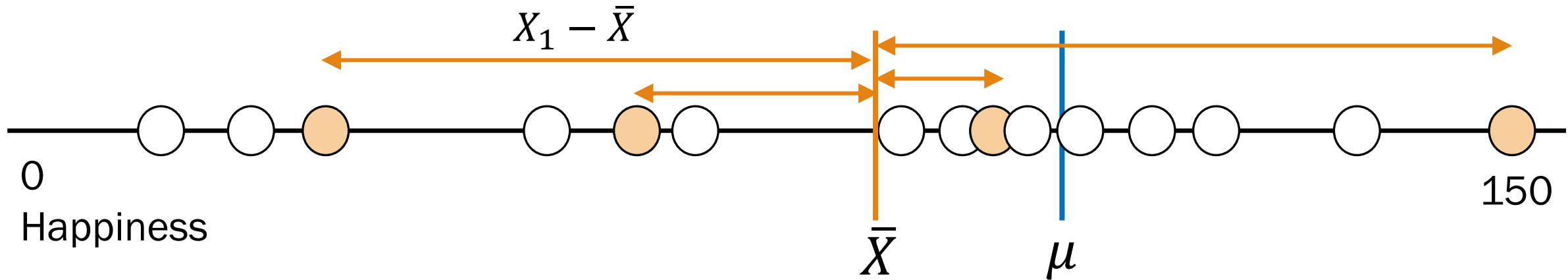
population mean
↓

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2$$

sample variance

sample mean
↓

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n \underbrace{(X_i - \bar{X})^2}$$



Population size, N

Sample variance is an estimate using an estimate, so it needs additional scaling.

Estimating the population variance



2. What is σ^2 , the **variance of happiness** of Bhutanese people?

If we only have a sample, (X_1, X_2, \dots, X_n) :

The best estimate of σ^2 is the **sample variance**:

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2$$

S^2 is an unbiased estimator of the population variance, σ^2 . $E[S^2] = \sigma^2$

Proof that S^2 is unbiased (just for reference)

$$E[S^2] = \sigma^2$$

$$E[S^2] = E\left[\frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2\right] \Rightarrow (n-1)E[S^2] = E\left[\sum_{i=1}^n (X_i - \bar{X})^2\right]$$

$$(n-1)E[S^2] = E\left[\sum_{i=1}^n ((X_i - \mu) + (\mu - \bar{X}))^2\right] \quad (\text{introduce } \mu - \mu)$$

$$= E\left[\sum_{i=1}^n (X_i - \mu)^2 + \sum_{i=1}^n (\mu - \bar{X})^2 + 2 \sum_{i=1}^n (X_i - \mu)(\mu - \bar{X})\right]$$

$$= E\left[\sum_{i=1}^n (X_i - \mu)^2 + n(\mu - \bar{X})^2 - 2n(\mu - \bar{X})^2\right]$$

$$= E\left[\sum_{i=1}^n (X_i - \mu)^2 - n(\mu - \bar{X})^2\right] = \sum_{i=1}^n E[(X_i - \mu)^2] - nE[(\bar{X} - \mu)^2]$$

$$= n\sigma^2 - n\text{Var}(\bar{X}) = n\sigma^2 - n\frac{\sigma^2}{n} = n\sigma^2 - \sigma^2 = (n-1)\sigma^2 \quad \text{Therefore } E[S^2] = \sigma^2$$

Standard error

Estimating population statistics

A particular outcome

1. Collect a sample, X_1, X_2, \dots, X_n .

(72, 85, 79, 79, 91, 68, ..., 71)
 $n = 200$

2. Compute **sample mean**, $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$.

$\bar{X} = 83$

3. Compute sample deviation, $X_i - \bar{X}$.

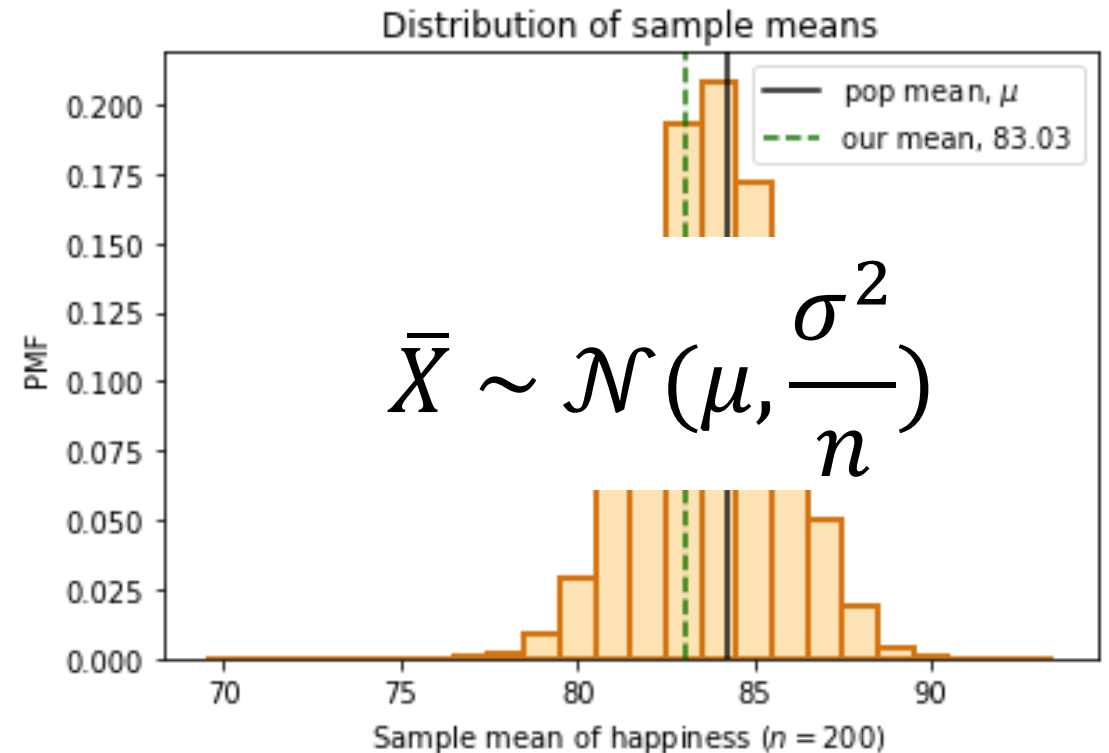
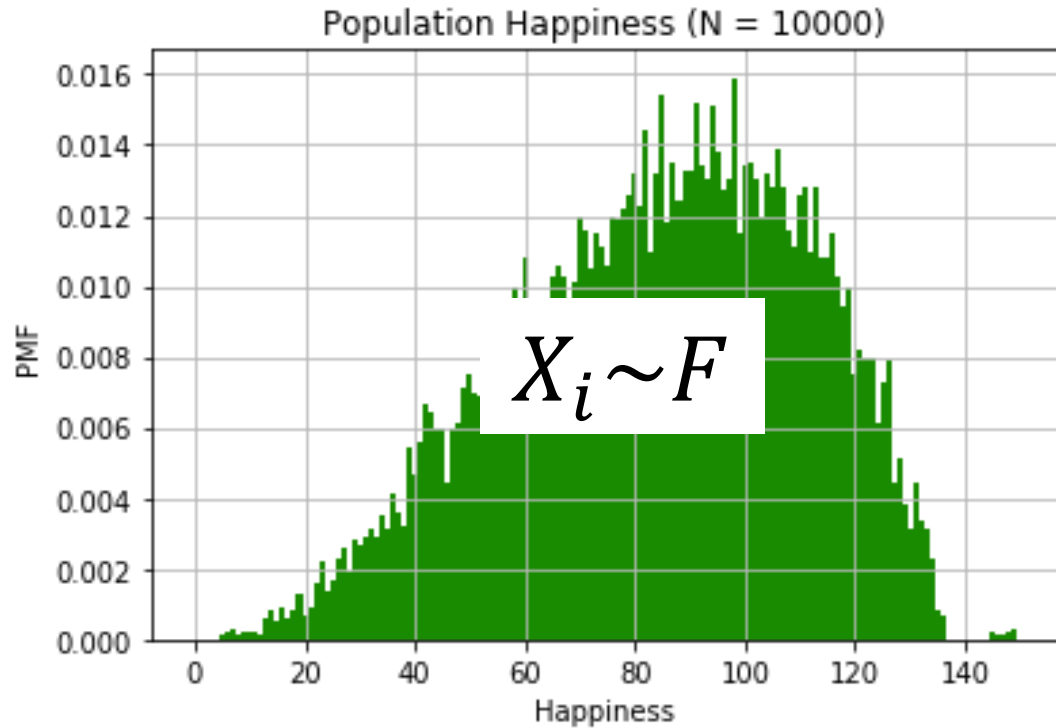
(-11, 2, -4, -4, 8, -15, ..., -12)

4. Compute **sample variance**, $S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2$.

$S^2 = 793$

How “close” are our estimates \bar{X} and S^2 ?

Sample mean



- $\text{Var}(\bar{X})$ is a measure of how “close” \bar{X} is to μ .
- How do we estimate $\text{Var}(\bar{X})$?

How “close” is our estimate \bar{X} to μ ?

$$E[\bar{X}] = \mu$$

$$\text{Var}(\bar{X}) = \frac{\sigma^2}{n}$$

We want to estimate this

def The **standard error** of the mean is an estimate of the standard deviation of \bar{X} .

$$SE = \sqrt{\frac{S^2}{n}}$$

Intuition:

- S^2 is an unbiased estimate of σ^2
- S^2/n is an unbiased estimate of $\sigma^2/n = \text{Var}(\bar{X})$
- $\sqrt{S^2/n}$ can estimate $\sqrt{\text{Var}(\bar{X})}$

[Errata \(5/18\)](#)

More info on bias of standard error: [wikipedia](#)

Standard error of the mean

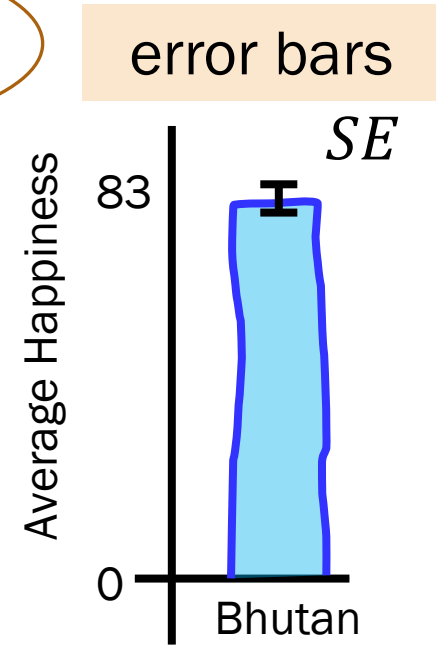
1. Mean happiness:

Claim: The average happiness of Bhutan is 83, with a standard error of 1.99.

Closed form: $SE = \sqrt{\frac{S^2}{n}}$

this is our estimate of how "close" we are

this is our best estimate of μ



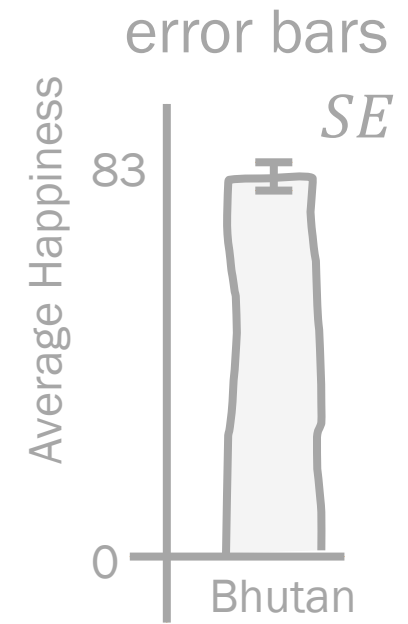
These 2 statistics give a sense of how the sample mean random variable \bar{X} behaves.

Standard error of variance?

1. Mean happiness:

Claim: The average happiness of Bhutan is 83, with a standard error of 1.99.

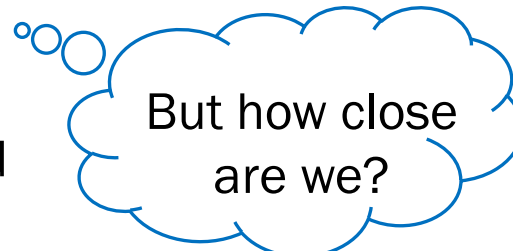
Closed form: $SE = \sqrt{\frac{S^2}{n}}$



2. Variance of happiness:

Claim: The variance of happiness of Bhutan is 793.

Closed form: Not covered in CS109



this is our best estimate of σ^2

Up next: Compute Statistics with code!

Bootstrap: Sample mean

Bootstrap

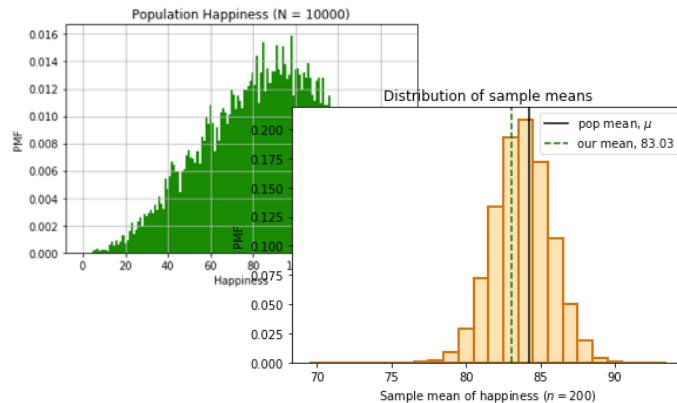
The Bootstrap:

Probability for Computer Scientists

Computing statistic of sample mean

What is the standard deviation of the sample mean \bar{X} ? (sample size $n = 200$)

Population distribution
(we don't have this)



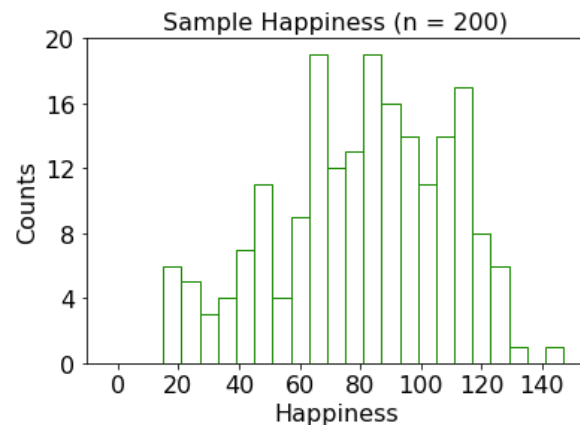
$$\frac{\sigma}{\sqrt{n}} = 1.886$$

Exact statistic
(we don't have this)

1.869

Simulated statistic
(we don't have this)

Sample distribution
(we do have this)



$$SE = \frac{S}{\sqrt{n}} = 1.992$$

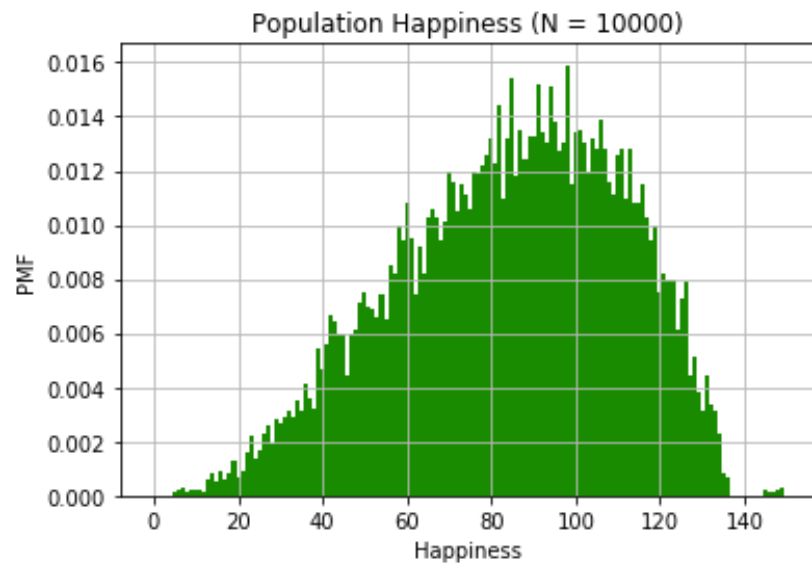
Estimated statistic,
by formula,
standard error

???

Simulated
estimated statistic

Note: We don't have access to the population.
But Lisa is sharing the exact statistic with you. Lisa Yan, CS109, 2020

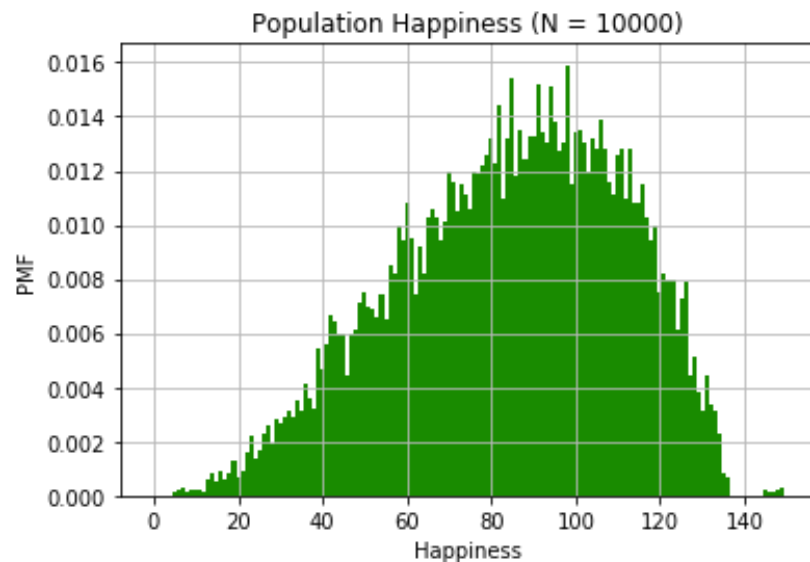
Bootstrap insight 1: Estimate the true distribution



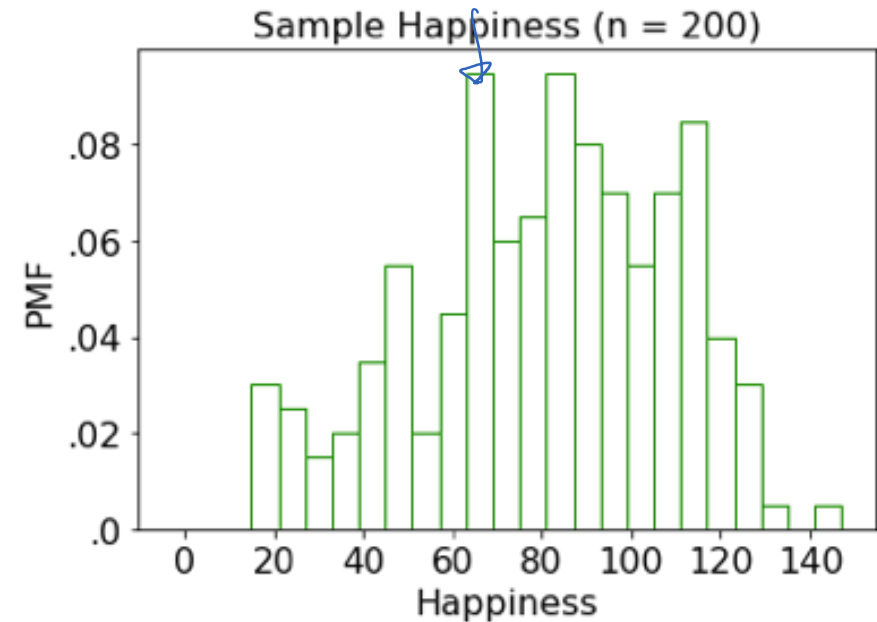
≈

Bootstrap insight 1: Estimate the true distribution

You can estimate the PMF of the underlying distribution, using your sample.*



\approx



The underlying
distribution



$$F \approx \hat{F}$$



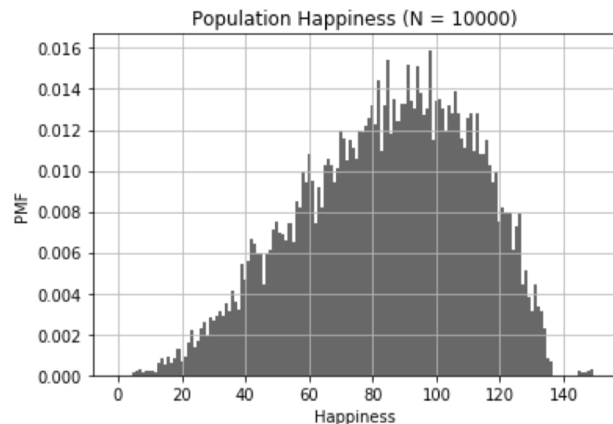
the sample distribution
(aka the histogram of
your data)

*This is just a histogram of your data!

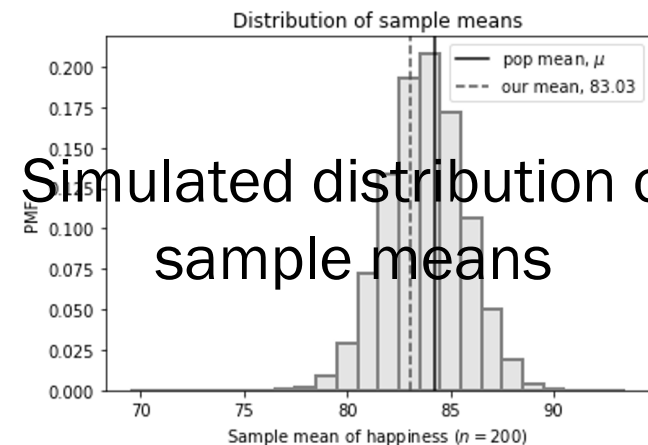
Bootstrap insight 2: Simulate a distribution

Approximate the procedure of simulating a distribution of a statistic, e.g., \bar{X} .

Population distribution
(we don't have this)

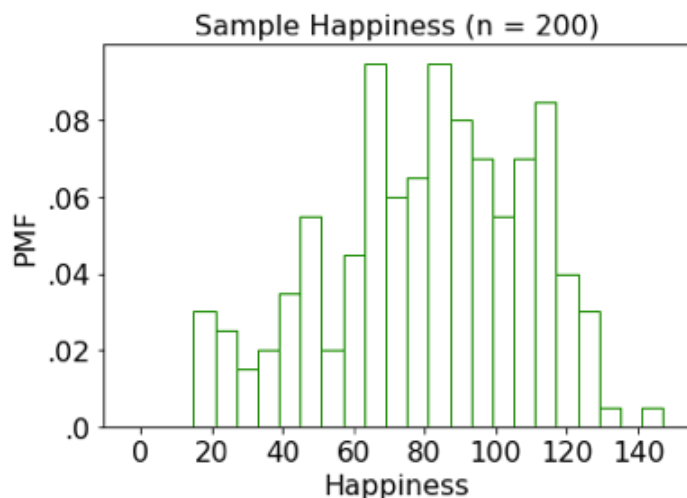


Distribution of \bar{X}

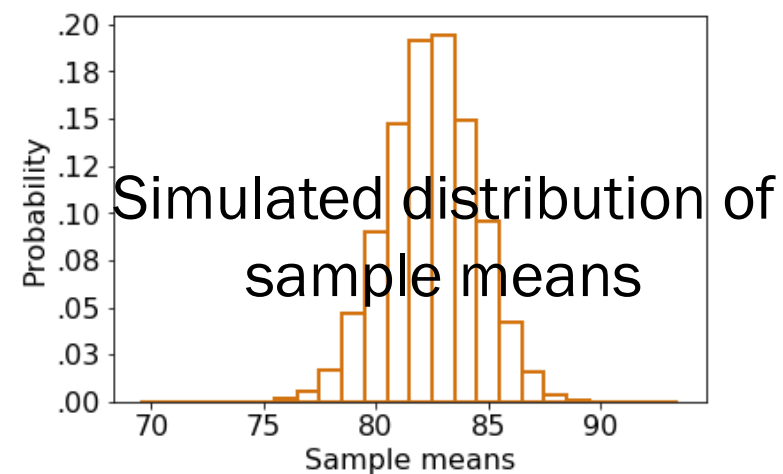


Simulated distribution of sample means

Sample distribution
(we do have this)



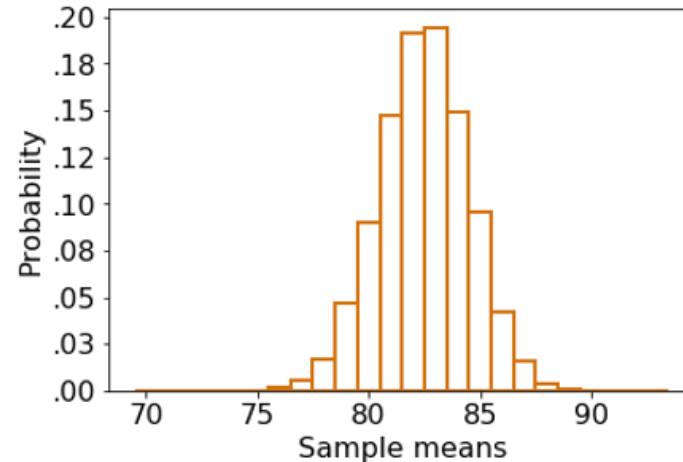
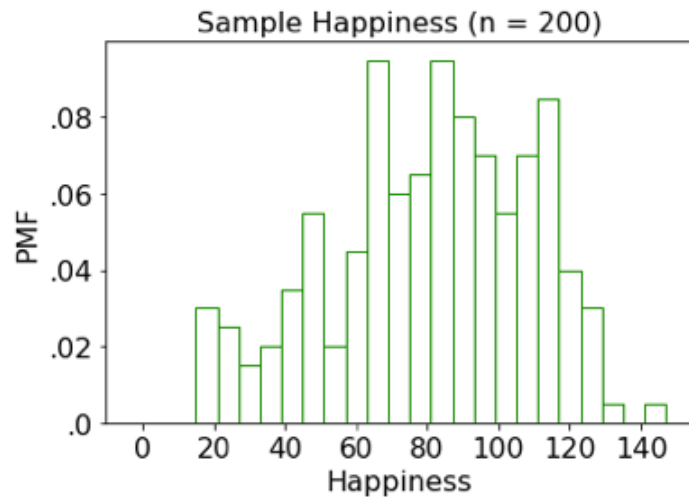
Bootstrap means



Simulated distribution of sample means

Bootstrapped sample means

`means = [84.7,
83.9, 80.6, 79.8,
90.3, ..., 85.2]`



`np.std(means)`
2.003

Estimate the true PMF
using our “PMF” (histogram)
of our sample.

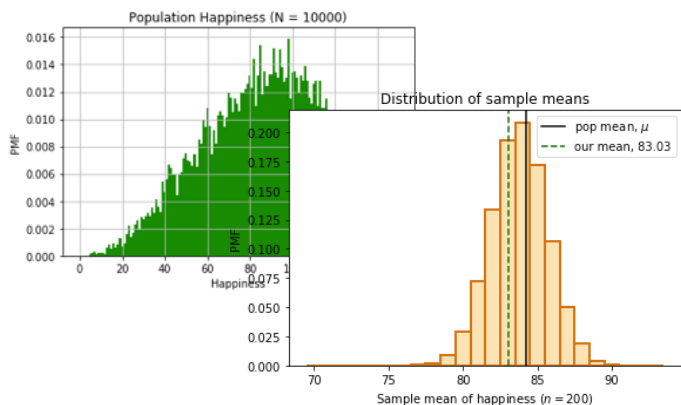
...generate a whole
bunch of sample means
of this estimated distribution...

...and compute the
standard deviation
of this distribution.

Computing statistic of sample mean

What is the standard deviation of the sample mean \bar{X} ? (sample size $n = 200$)

Population distribution
(we don't have this)



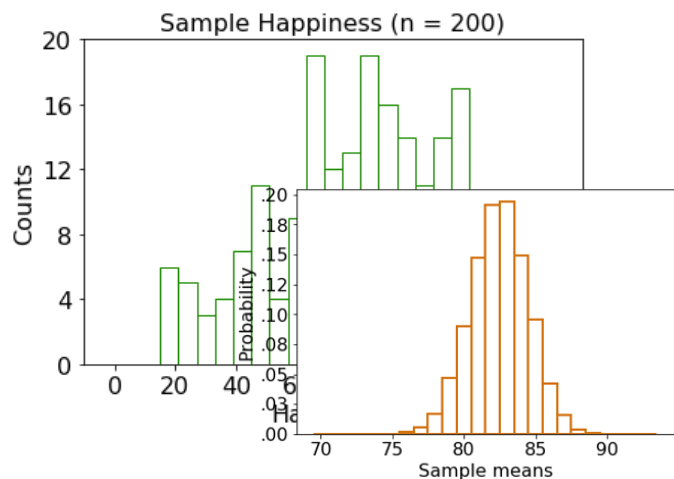
$$\frac{\sigma}{\sqrt{n}} = 1.886$$

Exact statistic
(we don't have this)

$$1.869$$

Simulated statistic
(we don't have this)

Sample distribution
(we do have this)



$$SE = \frac{S}{\sqrt{n}} = 1.992$$

Estimated statistic,
by formula,
standard error

$$2.003$$

Simulated estimated
statistic, **bootstrapped
standard error**

Bootstrap algorithm

Bootstrap Algorithm (sample):

1. Estimate the **PMF** using the sample
2. Repeat **10,000** times:
 - a. Resample **sample.size()** from PMF
 - b. Recalculate the **sample mean** on the resample
3. You now have a **distribution of your sample mean**

What is the distribution of your **sample mean**?

We'll talk about this algorithm in detail during live lecture!

Bootstrap algorithm

Bootstrap Algorithm (sample):

1. Estimate the **PMF** using the sample
2. Repeat **10,000** times:
 - a. Resample **sample.size()** from PMF
 - b. Recalculate the **statistic** on the resample
3. You now have a **distribution of your statistic**

What is the distribution of your **statistic**?

Bootstrapped sample variance

Bootstrap Algorithm (sample):

1. Estimate the **PMF** using the sample
2. Repeat **10,000** times:
 - a. Resample **sample.size()** from PMF
 - b. Recalculate the **sample variance** on the resample
3. You now have a **distribution of your sample variance**

What is the distribution of your **sample variance**?

Even if we don't have a closed form equation,
we estimate statistics of sample variance with bootstrapping!

19: Sampling and the Bootstrap (live)

Lisa Yan

May 18, 2020

Today's Ed Link

<https://us.edstem.org/courses/667/discussion/96703>

Quick check

1. μ , the population mean
2. $(X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8)$, a sample
3. σ^2 , the population variance
4. \bar{X} , the sample mean
5. $\bar{X} = 83$
6. $(X_1 = 59, X_2 = 87, X_3 = 94, X_4 = 99,$
 $X_5 = 87, X_6 = 78, X_7 = 69, X_8 = 91)$

- A. Random variable(s)
- B. Value
- C. Event

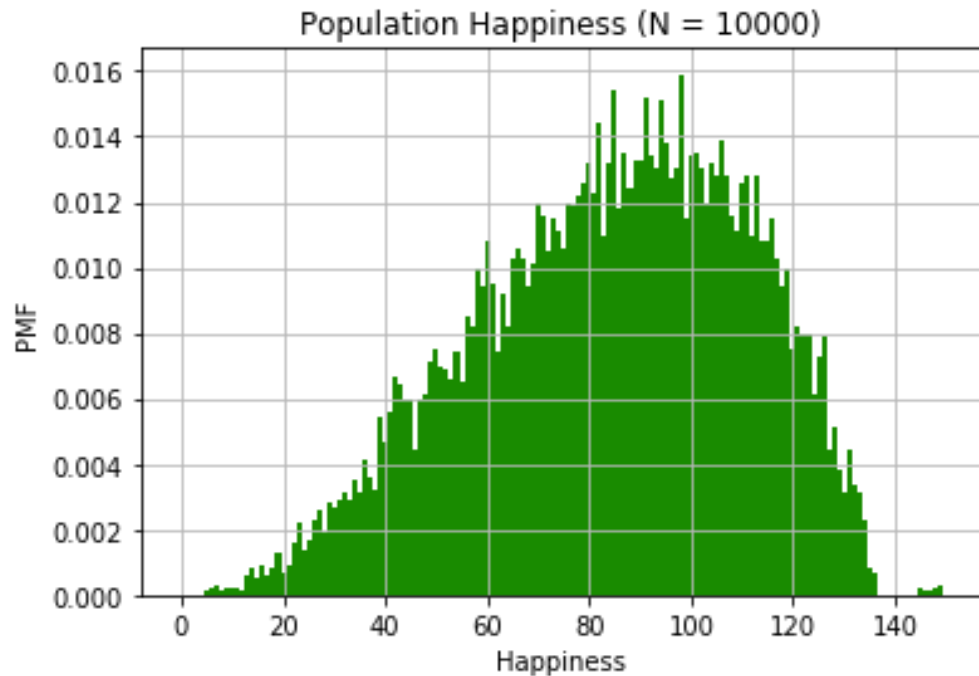


Quick check

1. μ , the population mean
2. $(X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8)$, a sample
3. σ^2 , the population variance
4. \bar{X} , the sample mean
5. $\bar{X} = 83$
6. $(X_1 = 59, X_2 = 87, X_3 = 94, X_4 = 99,$
 $X_5 = 87, X_6 = 78, X_7 = 69, X_8 = 91)$

- A. Random variable(s)
- B. Value
- C. Event

These are outcomes from your collected data.



Hypothetical questions:

- What is the probability that a Bhutanese peep is just straight up loving life?
- What is the probability that the mean of a subsample of 200 people is within the range 81 to 85?
- What is the variance of the sample variance of subsamples of 200 people?

The Bootstrap:

Probability for Computer Scientists

Allows you to do the following:

- Calculate distributions over statistics
- Calculate p values

How “close” is our estimate \bar{X} to μ ?

$$E[\bar{X}] = \mu$$

$$\text{Var}(\bar{X}) = \frac{\sigma^2}{n}$$

We want to estimate this

def The **standard error** of the mean is an estimate of the standard deviation of \bar{X} .

$$SE = \sqrt{\frac{S^2}{n}}$$

Intuition:

- S^2 is an unbiased estimate of σ^2
- S^2/n is an unbiased estimate of $\sigma^2/n = \text{Var}(\bar{X})$
- $\sqrt{S^2/n}$ is an unbiased estimate of $\sqrt{\text{Var}(\bar{X})}$

Errata (5/18)

More info on bias of standard error: [wikipedia](#)

1. Mean happiness:

Claim: The average happiness of Bhutan is 83, with a standard error of 1.99.

Closed form:

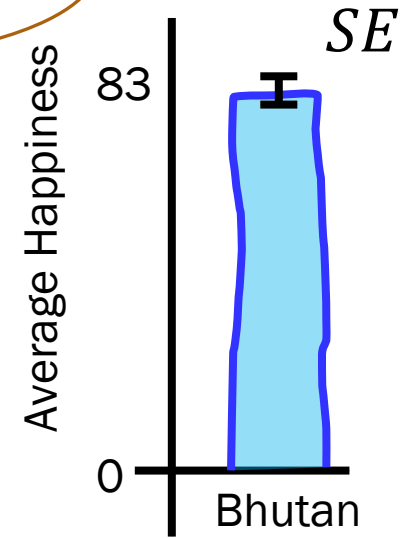
$$SE = \sqrt{\frac{S^2}{n}}$$

this is how close we are



Verified via bootstrap:
`np.std(means)`
`= 2.003`

this is our best estimate of μ



1. Mean happiness:

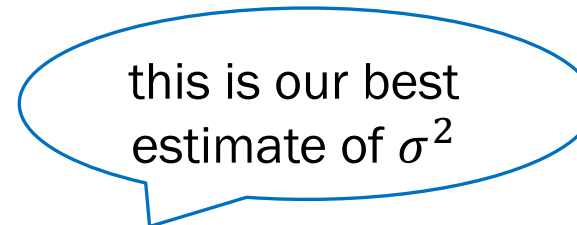
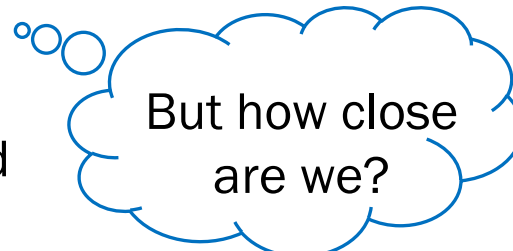
Claim: The average happiness of Bhutan is 83, with a standard error of 1.99.

Closed form: $SE = \sqrt{\frac{S^2}{n}}$

2. Variance of happiness:

Claim: The variance of happiness of Bhutan is 793.

Closed form: Not covered in CS109

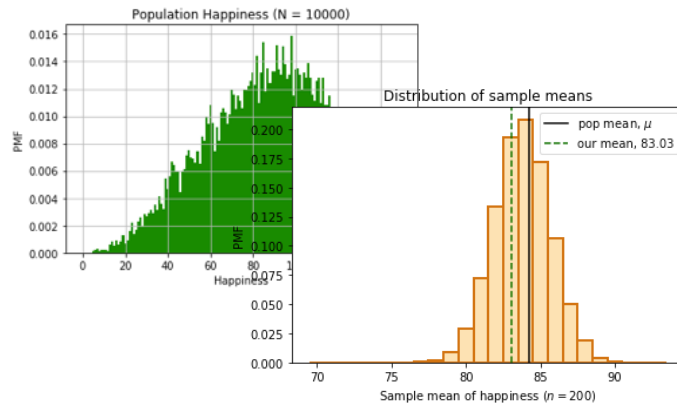


We can bootstrap for standard error of sample variance—a statistic of a statistic.

Computing statistic of sample mean

What is the standard deviation of the sample mean \bar{X} ? (sample size $n = 200$)

Population distribution
(we don't have this)



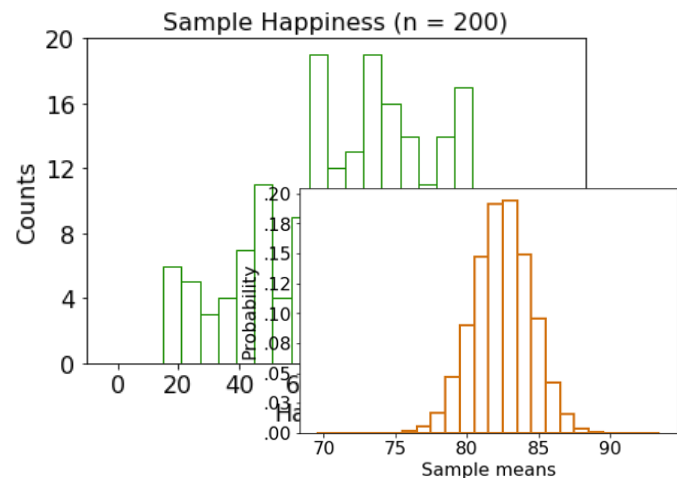
$$\frac{\sigma}{\sqrt{n}} = 1.886$$

Exact statistic
(we don't have this)

$$1.869$$

Simulated statistic
(we don't have this)

Sample distribution
(we do have this)



$$SE = \frac{S}{\sqrt{n}} = 1.992$$

Estimated statistic,
by formula,
standard error

$$2.003$$

Simulated estimated
statistic, **bootstrapped
standard error**

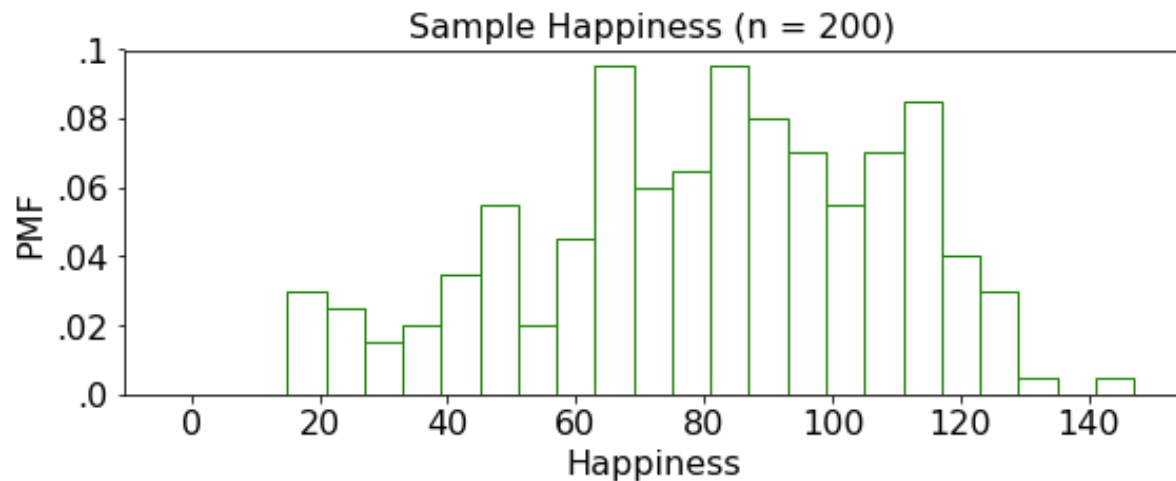
Bootstrapped sample variance

Bootstrap Algorithm (sample):

1. Estimate the **PMF** using the sample
2. Repeat **10,000** times:
 - a. Resample **sample.size()** from PMF
 - b. Recalculate the **sample variance** on the resample
3. You now have a **distribution of your sample variance**

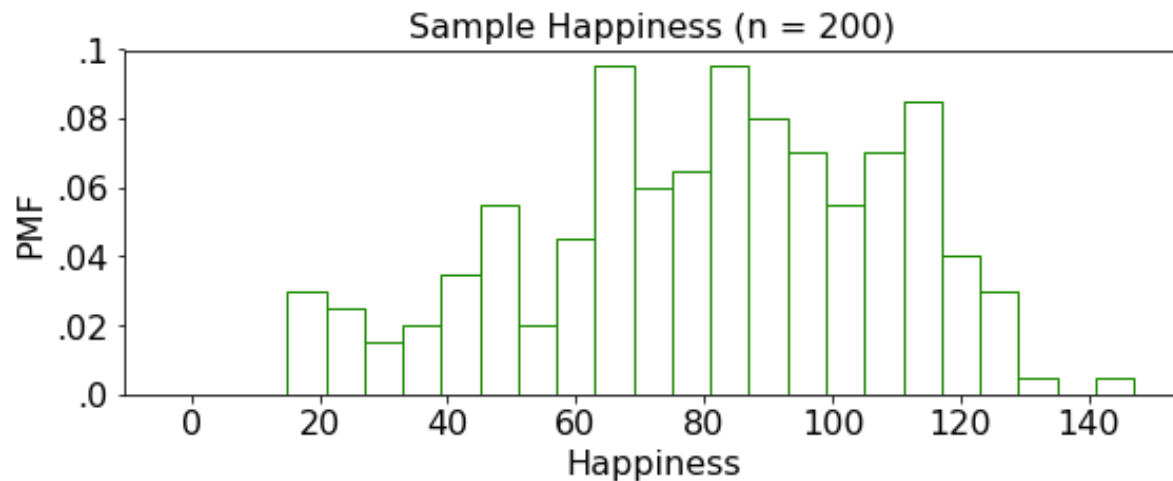
What is the distribution of your **sample variance**?

Bootstrapped variance



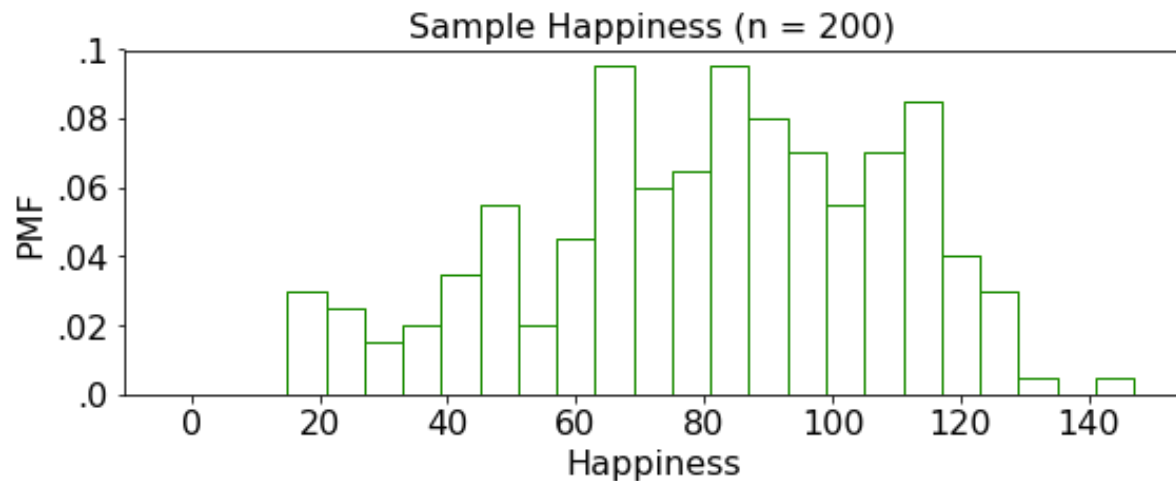
1. Estimate the **PMF** using the sample
2. Repeat **10,000** times:
 - a. Resample `sample.size()` from PMF
 - b. Recalculate the **sample variance** on the resample
3. You now have a **distribution of your sample variance**

Bootstrapped variance

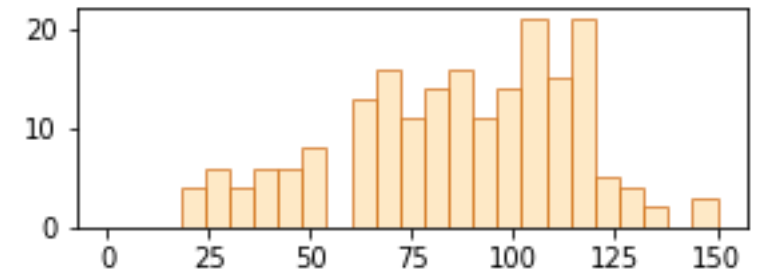


1. Estimate the **PMF** using the sample
- ➔ 2. Repeat **10,000** times:
 - a. Resample **sample.size()** from PMF
 - b. Recalculate the **sample variance** on the resample
3. You now have a **distribution of your sample variance**

Bootstrapped variance



[52, 38, 98, 107, ..., 94]



1. Estimate the **PMF** using the sample

2. Repeat **10,000** times:



a. Resample **sample.size()** from PMF

b. Recalculate the **sample variance** on the resample

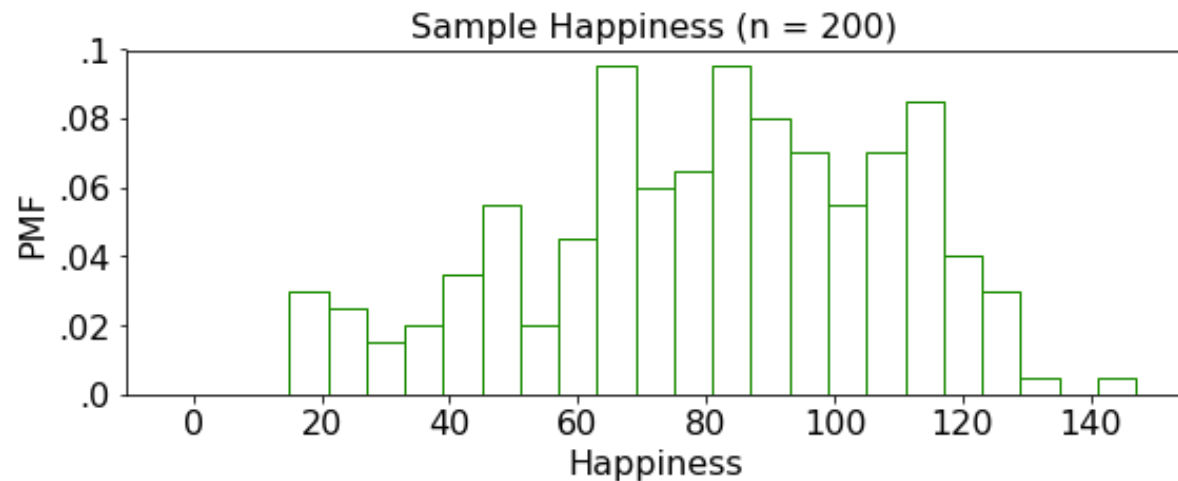
3. You now have a **distribution of your**



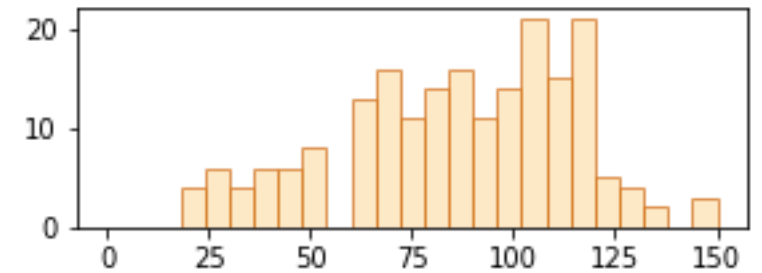
Why are these samples different? 🤔


This resampled sample is generated with replacement.

Bootstrapped variance



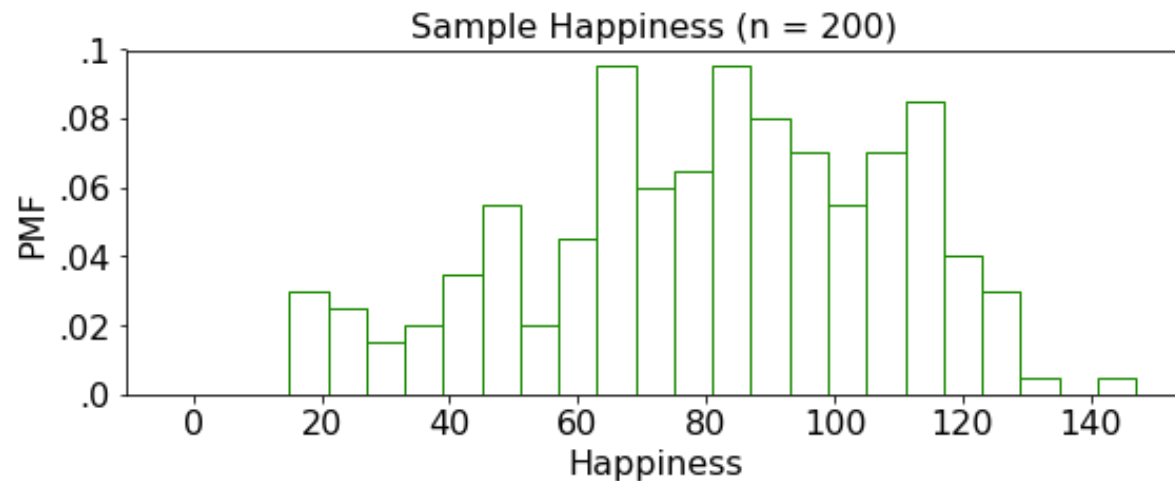
[52, 38, 98, 107, ..., 94]



1. Estimate the **PMF** using the sample
2. Repeat **10,000** times:
 - a. Resample `sample.size()` from PMF
 -  b. Recalculate the **sample variance** on the resample
3. You now have a **distribution of your sample variance**

variances = [827.4]

Bootstrapped variance



1. Estimate the **PMF** using the sample

➔ 2. Repeat **10,000** times:

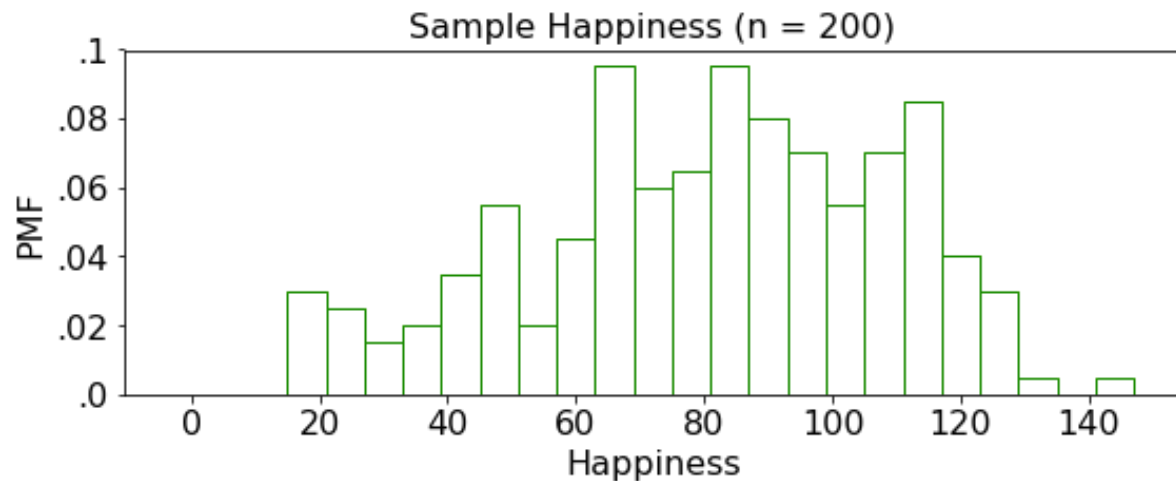
a. Resample **sample.size()** from PMF

b. Recalculate the **sample variance** on the resample

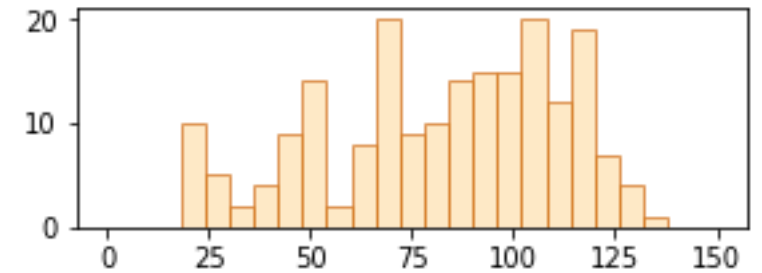
3. You now have a **distribution of your sample variance**

variances = [827.4]

Bootstrapped variance



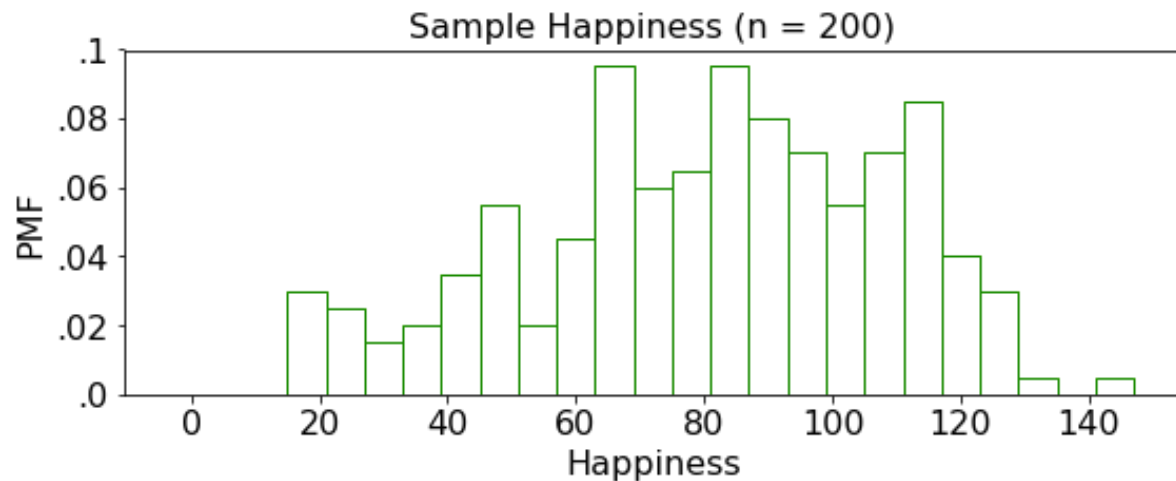
[116, 76, 132, 85, ..., 78]



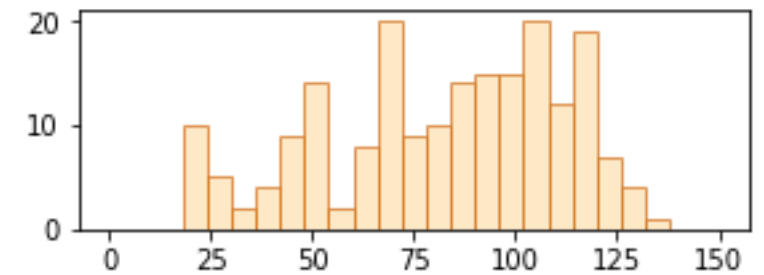
1. Estimate the **PMF** using the sample
2. Repeat **10,000** times:
 - a. Resample `sample.size()` from PMF
 - b. Recalculate the **sample variance** on the resample
3. You now have a **distribution of your sample variance**


variances = [827.4]

Bootstrapped variance



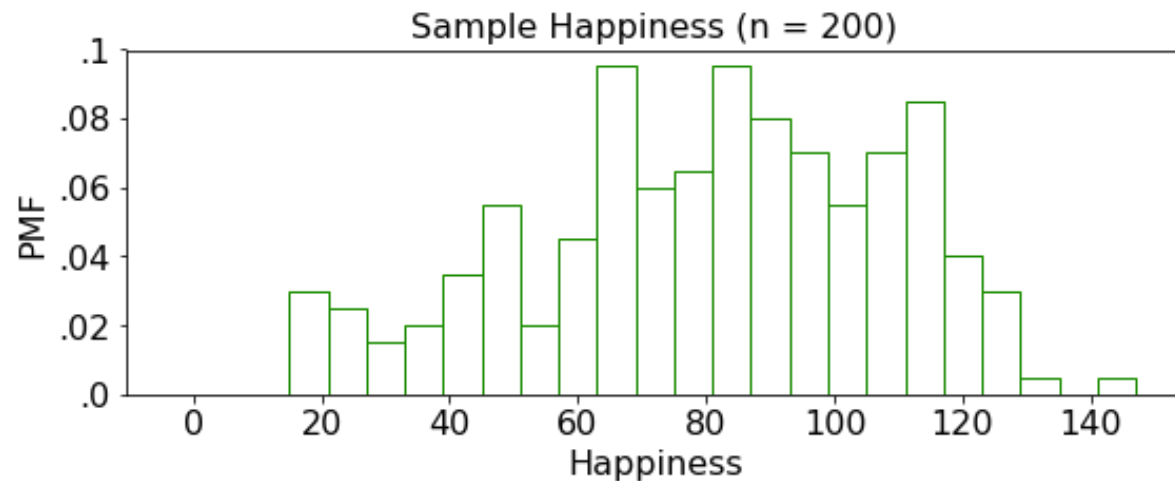
[116, 76, 132, 85, ..., 78]



1. Estimate the **PMF** using the sample
2. Repeat **10,000** times:
 - a. Resample **sample.size()** from PMF
 -  b. Recalculate the **sample variance** on the resample
3. You now have a **distribution of your sample variance**

variances = [827.4, 846.1]

Bootstrapped variance



1. Estimate the **PMF** using the sample

2. Repeat **10,000** times:

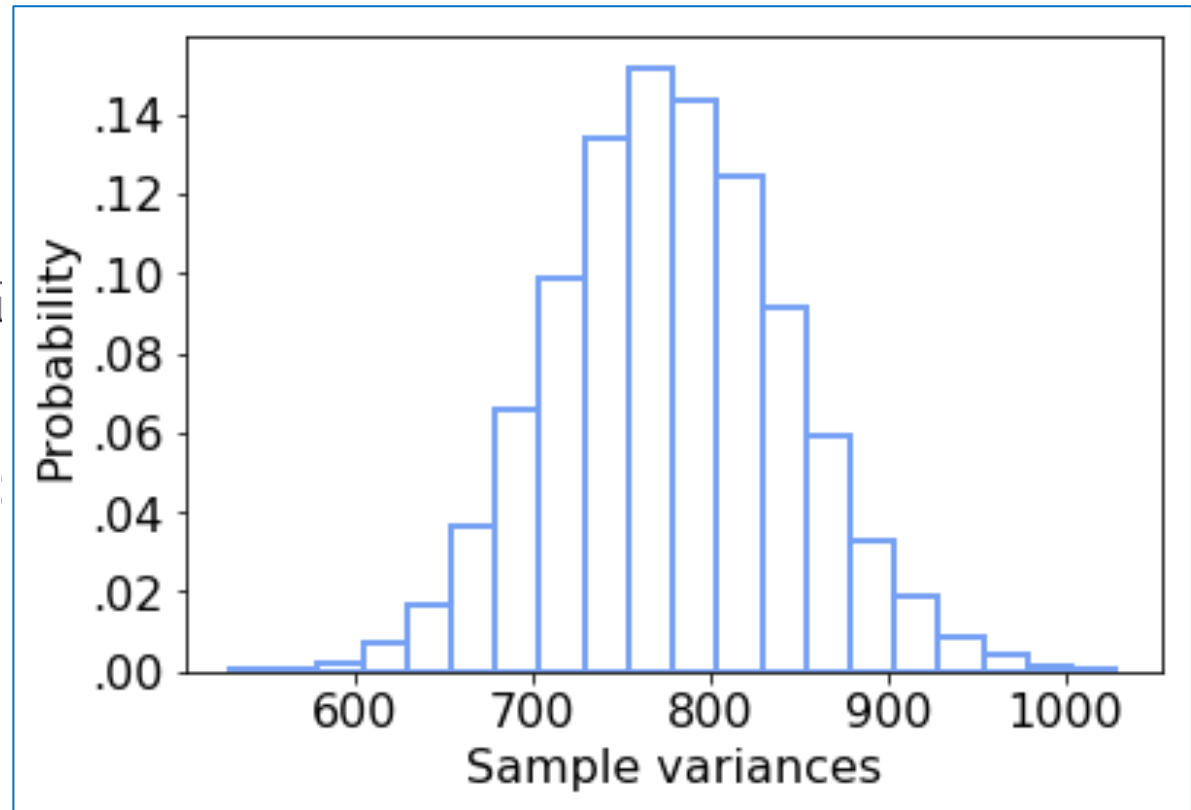
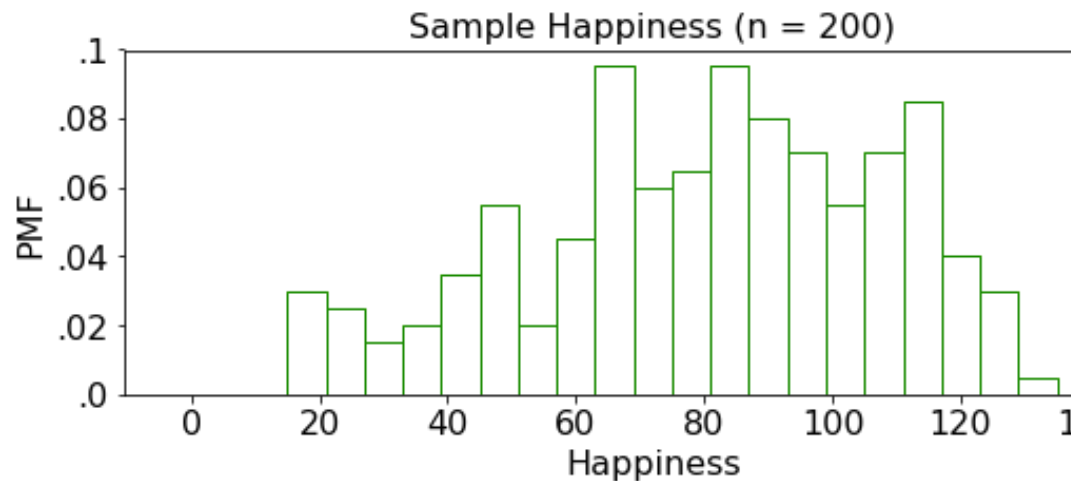
a. Resample `sample.size()` from PMF

b. Recalculate the **sample variance** on the resample

3. You now have a **distribution of your sample variance**

variances = [827.4, 846.1]

Bootstrapped variance



1. Estimate the **PMF** using the
2. Repeat **10,000** times:
 - a. Resample **sample.size()**
 - b. Recalculate the **sample**

→ 3. You now have a **distribution of your sample variance**

variances = [827.4, 846.1, 726.0, ..., 860.7]

Bootstrapped variance

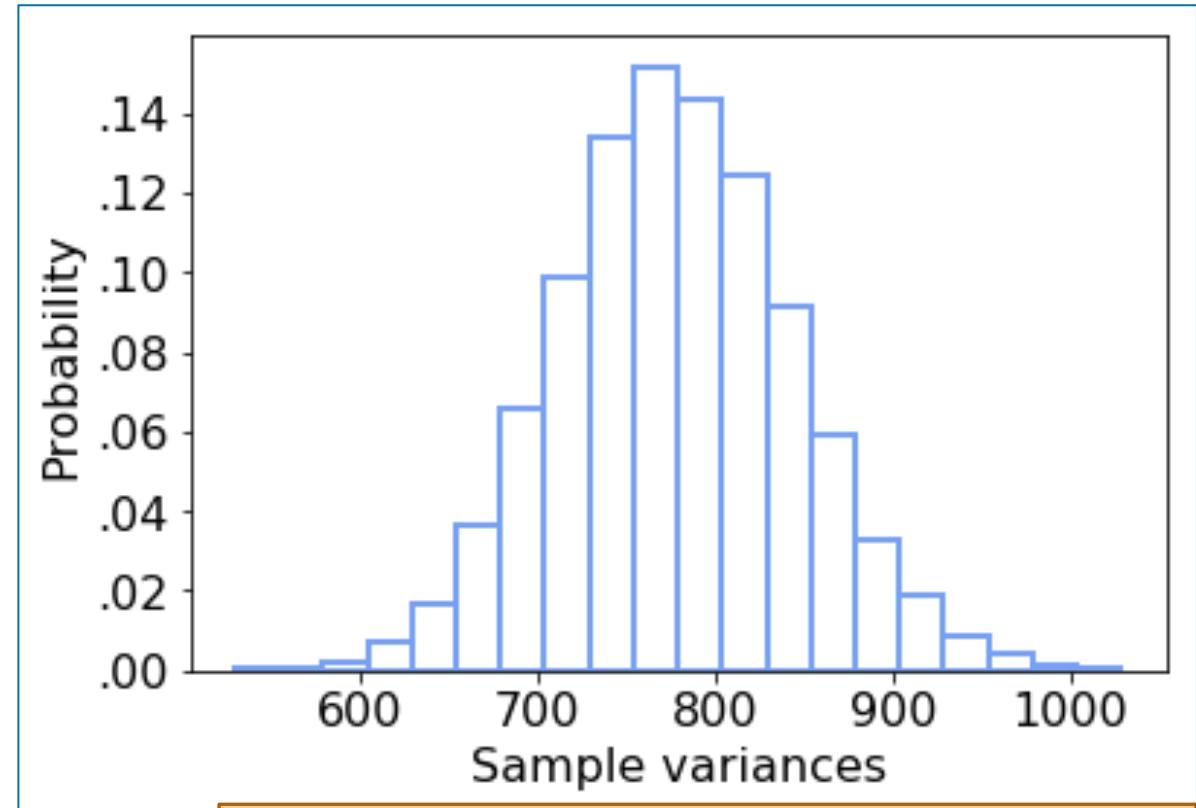
3. You now have a distribution of your **sample variance**

```
variances = [827.4,  
            846.1, 726.0, ...,  
            860.7]
```

What is the bootstrapped standard error?

```
np.std(variances)
```

Bootstrapped standard error: 66.16



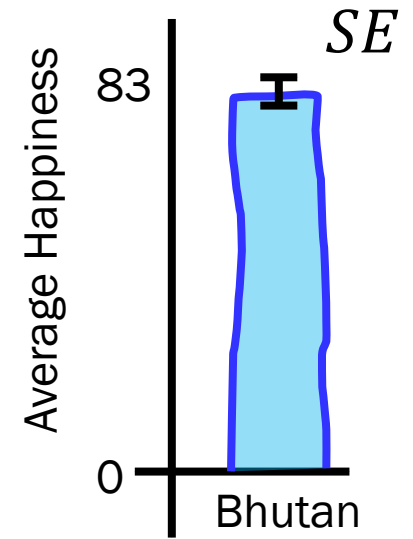
- Simulate a distribution of sample variances
- Compute standard deviation

Standard error

1. Mean happiness:

Claim: The average happiness of Bhutan is 83, with a standard error of 1.99.

Closed form: $SE = \sqrt{\frac{S^2}{n}}$



2. Variance of happiness:

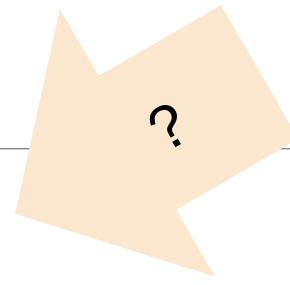
Claim: The variance of happiness of Bhutan is 793, with a **bootstrapped standard error of 66.16**.

S^2 is our best estimate of σ^2

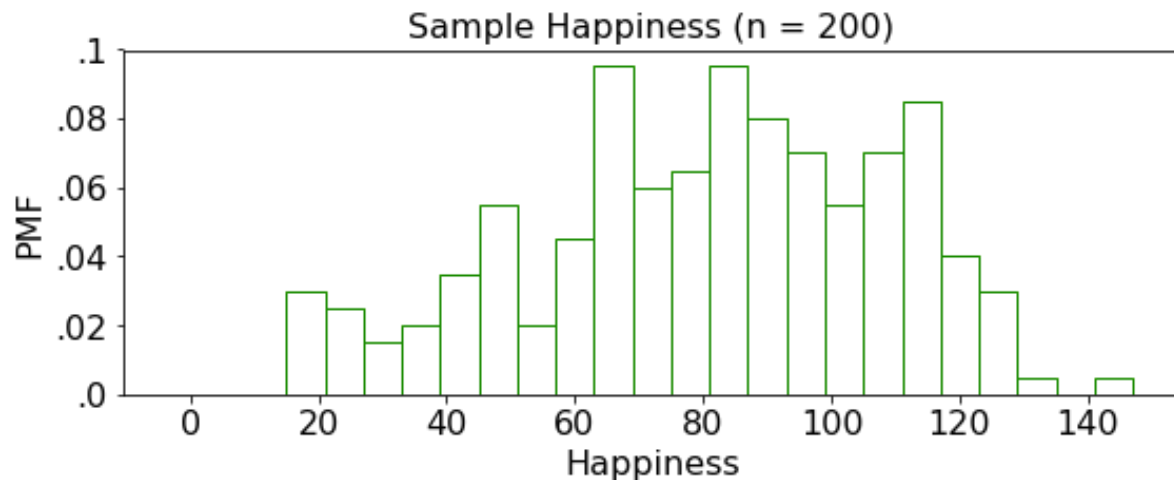
this is how close we are, calculated by bootstrapping



Algorithm in practice



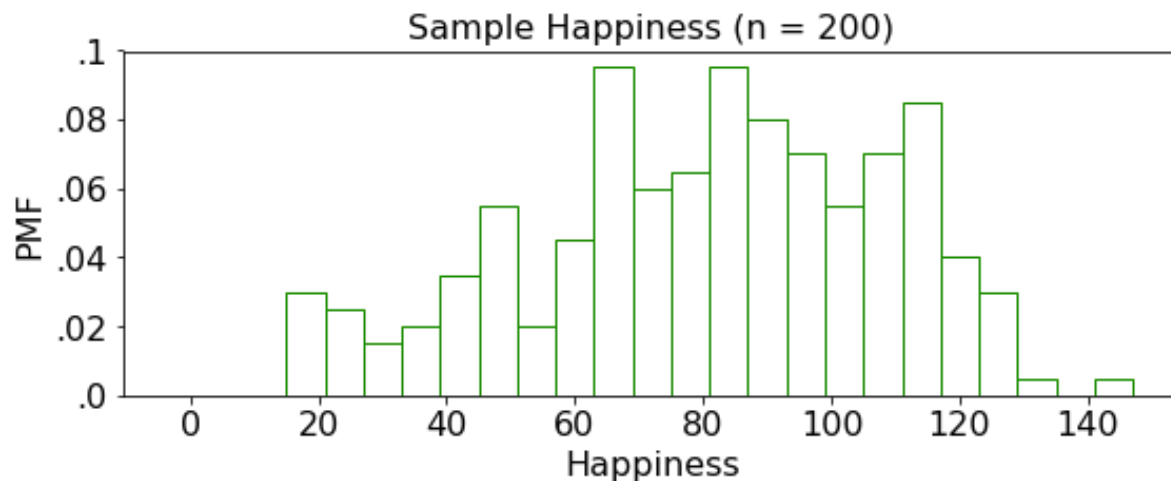
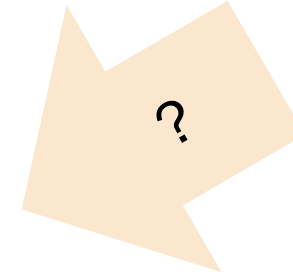
1. Estimate the **PMF** using the sample
2. Repeat **10,000** times:
 - a. Resample **sample.size()** from PMF
 - b. Recalculate the **statistic** on the resample
3. You now have a **distribution of your statistic**



$$P(X = k) = \frac{\text{\# values in sample equal to } k}{n}$$

Algorithm in practice

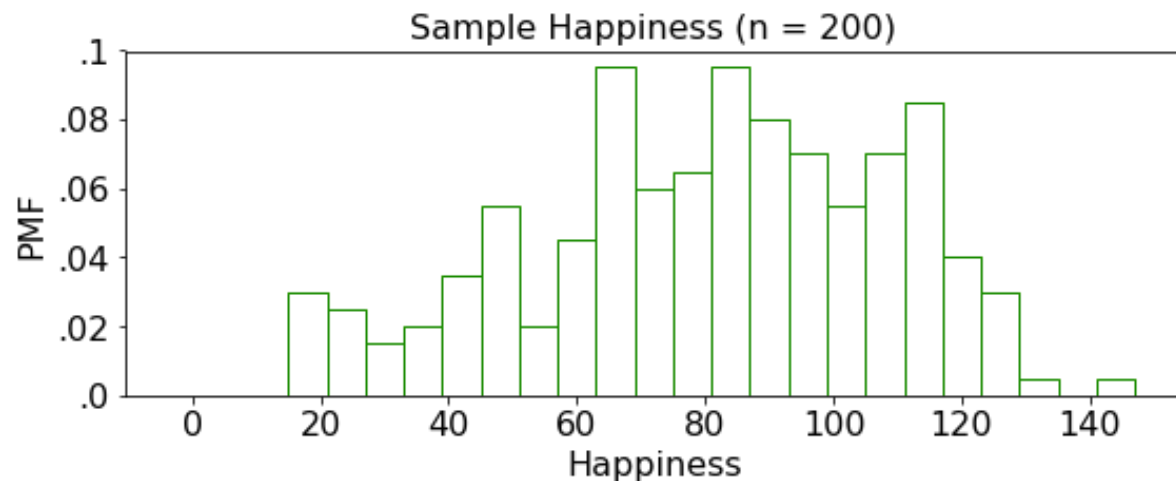
1. Estimate the **PMF** using the sample
2. Repeat **10,000** times:
 - a. Resample **sample.size()** from PMF
 - b. Recalculate the **statistic** on the resample
3. You now have a **distribution of your statistic**



$$P(X = k) = \frac{\text{\# values in sample equal to } k}{n}$$

Algorithm in practice

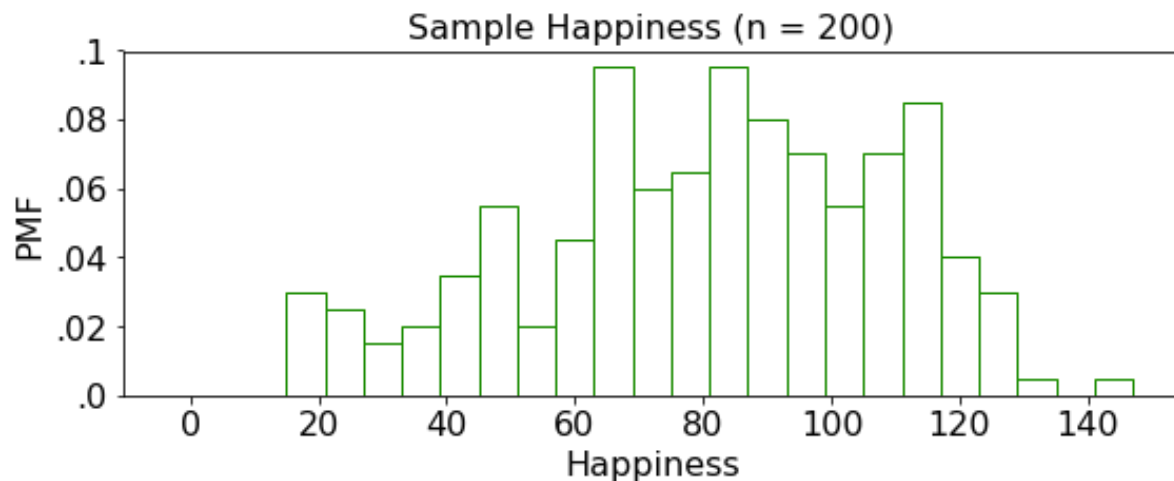
```
def resample(sample, n):  
    # estimate the PMF using the sample  
    # draw n new samples from the PMF  
    return np.random.choice(sample,  
                             n, replace=True)
```



$$P(X = k) = \frac{\text{\# values in sample equal to } k}{n}$$

Algorithm in practice

1. Estimate the **PMF** using the sample
2. Repeat **10,000** times:
 - a. Resample **sample.size()** from PMF, **with replacement**
 - b. Recalculate the **statistic** on the resample
3. You now have a **distribution of your statistic**



$$P(X = k) = \frac{\# \text{ values in sample equal to } k}{n}$$

Interlude for announcements

(The joke+news are at the end this time!)

Announcements

Final Logistics

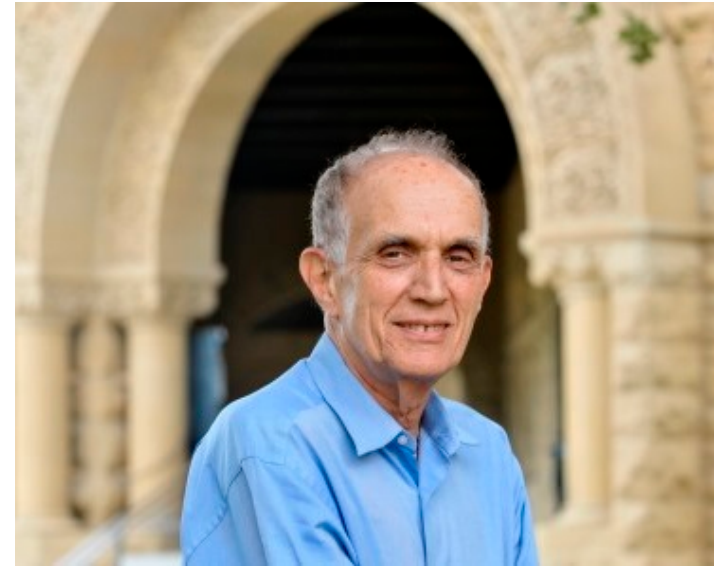
Out: Friday 8/14, 5:00pm
Due: Sunday 8/16, 5:00pm
Covers: all lectures, with a focus on later material
Further details here: <https://us.edstem.org/courses/667/discussion/96710>

To the code!

Bootstrap provides a way to calculate probabilities of statistics using code.

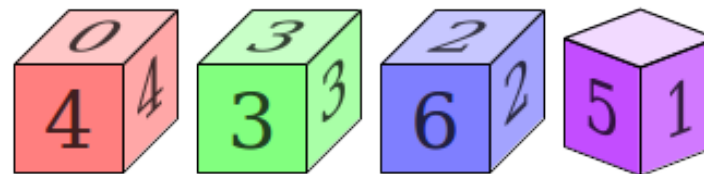
Bradley Efron

- Invented bootstrapping in 1979
- Still a professor at Stanford
- Won a National Science Medal



Efron's dice: 4 dice A, B, C, D such that

$$P(A > B) = P(B > C) = P(C > D) = P(D > A) = \frac{2}{3}$$



To the code!

Bootstrap provides a way to calculate probabilities of statistics using code.

Bootstrapping works for any statistic*

*as long as your sample is i.i.d. and the underlying distribution does not have a long tail

Bootstrap: p-value

Null hypothesis test

Sample 1
4.44
3.36
5.87
2.31
...
3.70

$$\bar{X}_1 = 3.1$$

Sample 2
4.44
3.36
5.87
2.31
...
3.70

$$\bar{X}_2 = 2.4$$

Claim: Sample 1 and Sample 2 have a 0.7 difference of means.

Null hypothesis test

Bhutan Happiness
4.44
3.36
5.87
2.31
...
3.70

$$\bar{X}_1 = 3.1$$

Nepal Happiness
4.44
3.36
5.87
2.31
...
3.70

$$\bar{X}_2 = 2.4$$

Claim: The difference in mean happiness between Nepal and Bhutan is 0.7 happiness points, and **this is significant.**

Null hypothesis test

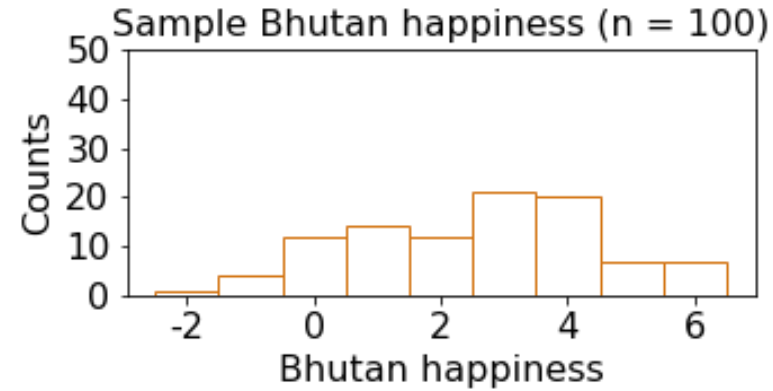
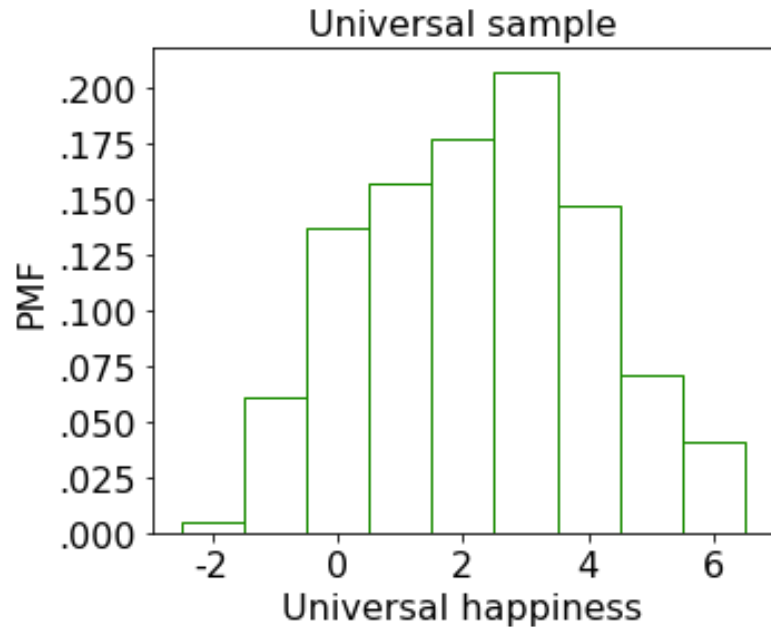
def **null hypothesis** – Even if there is no pattern (i.e., the two samples are from identical distributions), your claim might have arisen by chance.

def **p-value** – What is the probability that, under the null hypothesis, the observed difference occurs?

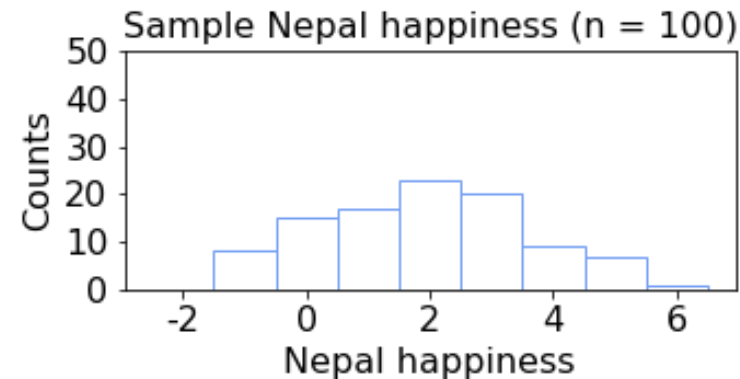
Claim: The difference in mean happiness between Nepal and Bhutan is 0.7 happiness points, and **this is significant.**

Universal sample

(this is what the null hypothesis assumes)



$$\bar{X}_1 = 3.1$$



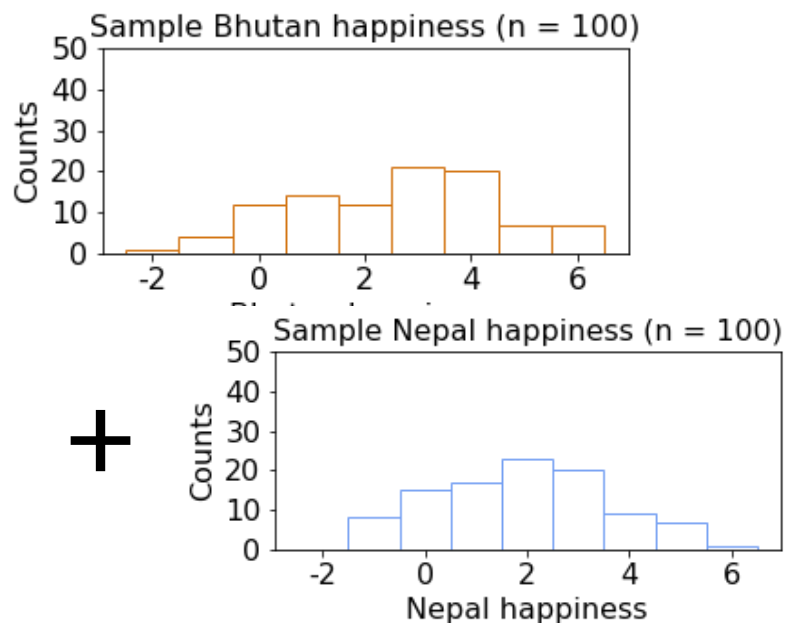
$$\bar{X}_2 = 2.4$$

Want **p-value**: probability $|\bar{X}_1 - \bar{X}_2| \geq$ happens under null hypothesis

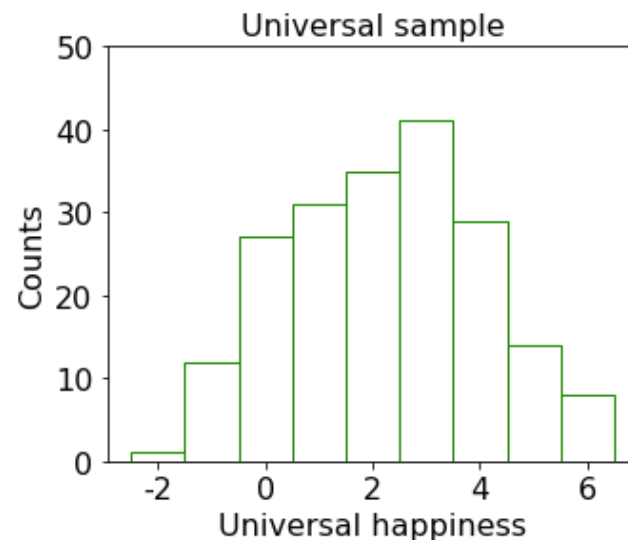
Bootstrap for p-values

1. Create a universal sample using your two samples

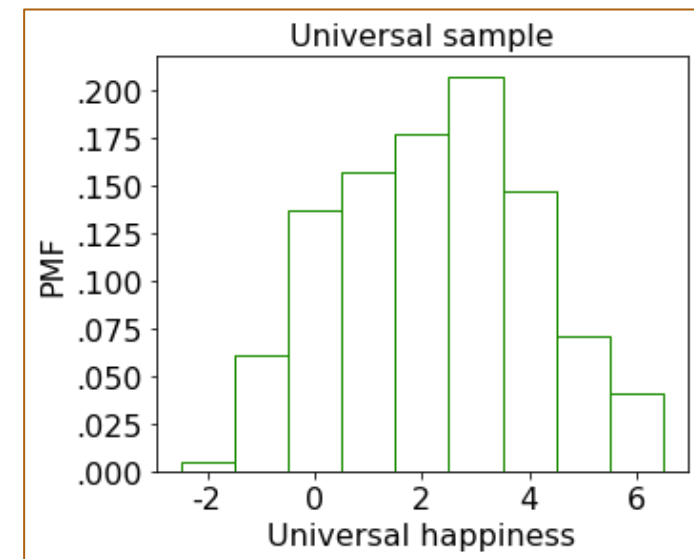
Recreate the null hypothesis



=

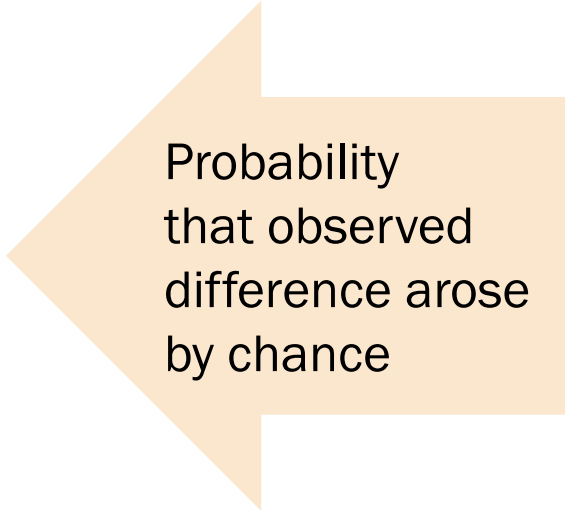


≈



Bootstrap for p-values

1. Create a universal sample using your two samples
2. Repeat **10,000** times:
 - a. Resample **both samples**
 - b. Recalculate the **mean difference** between the resamples
3. **p-value** =
$$\frac{\# (\text{mean diffs} \geq \text{observed diff})}{n}$$



Probability
that observed
difference arose
by chance

Bootstrap for p-values

```
def pvalue_boot(bhutan_sample, nepal_sample):  
    N = size of the bhutan_sample  
    M = size of the nepal_sample  
    observed_diff = |mean of bhutan_sample - mean of nepal_sample|  
  
    uni_sample = combine bhutan_sample and nepal_sample  
    count = 0  
  
    repeat 10,000 times:  
        bhutan_resample = draw N resamples from the uni_sample  
        nepal_resample = draw M resamples from the uni_sample  
        muBhutan = sample mean of the bhutan_resample  
        muNepal = sample mean of the nepal_resample  
        diff = |muNepal - muBhutan|  
        if diff >= observed_diff:  
            count += 1  
  
pValue = count / 10,000
```

Bootstrap for p-values

1. Create a universal sample using your two samples

```
def pvalue_boot(bhutan_sample, nepal_sample):
```

```
    N = size of the bhutan_sample
```

```
    M = size of the nepal_sample
```

```
    observed_diff = |mean of bhutan_sample - mean of nepal_sample|
```

```
    uni_sample = combine bhutan_sample and nepal_sample
```

```
    count = 0
```

```
    repeat 10,000 times:
```

```
        bhutan_resample = draw N resamples from the uni_sample
```

```
        nepal_resample = draw M resamples from the uni_sample
```

```
        muBhutan = sample mean of the bhutan_resample
```

```
        muNepal = sample mean of the nepal_resample
```

```
        diff = |muNepal - muBhutan|
```

```
        if diff >= observed_diff:
```

```
            count += 1
```

```
pValue = count / 10,000
```

Bootstrap for p-values

2. a. Resample both samples

```
def pvalue_boot(bhutan_sample, nepal_sample):  
    N = size of the bhutan_sample  
    M = size of the nepal_sample  
    observed_diff = |mean of bhutan_sample - mean of nepal_sample|  
  
    uni_sample = combine bhutan_sample and nepal_sample  
    count = 0  
  
    repeat 10,000 times:  
        bhutan_resample = draw N resamples from the uni_sample  
        nepal_resample = draw M resamples from the uni_sample  
        muBhutan = sample mean of the bhutan_resample  
        muNepal = sample mean of the nepal_resample  
        diff = |muNepal - muBhutan|  
        if diff >= observed_diff:  
            count += 1  
  
pValue = count / 10,000
```


Bootstrap for p-values

2. b. Recalculate the mean difference b/t resamples

```
def pvalue_boot(bhutan_sample, nepal_sample):  
    N = size of the bhutan_sample  
    M = size of the nepal_sample  
    observed_diff = |mean of bhutan_sample - mean of nepal_sample|
```

```
uni_sample = combine bhutan_sample and nepal_sample  
count = 0
```

```
repeat 10,000 times:
```

```
    bhutan_resample = draw N resamples from the uni_sample
```

```
    nepal_resample = draw M resamples from the uni_sample
```

```
    muBhutan = sample mean of the bhutan_resample
```

```
    muNepal = sample mean of the nepal_resample
```

```
    diff = |muNepal - muBhutan|
```

```
    if diff >= observed_diff:
```

```
        count += 1
```

```
pValue = count / 10,000
```

Bootstrap for p-values

$$3. \text{ p-value} = \frac{\# (\text{mean diffs} > \text{observed diff})}{n}$$

```
def pvalue_boot(bhutan_sample, nepal_sample):  
    N = size of the bhutan_sample  
    M = size of the nepal_sample  
    observed_diff = |mean of bhutan_sample - mean of nepal_sample|  
  
    uni_sample = combine bhutan_sample and nepal_sample  
    count = 0  
  
    repeat 10,000 times:  
        bhutan_resample = draw N resamples from the uni_sample  
        nepal_resample = draw M resamples from the uni_sample  
        muBhutan = sample mean of the bhutan_resample  
        muNepal = sample mean of the nepal_resample  
        diff = |muNepal - muBhutan|  
        if diff >= observed_diff:  
            count += 1
```

`pValue = count / 10,000`

Bootstrap for p-values

```
def pvalue_boot(bhutan_sample, nepal_sample):  
    N = size of the bhutan_sample  
    M = size of the nepal_sample  
    observed_diff = |mean of bhutan_sample - mean of nepal_sample|
```

```
    uni_sample = combine bhutan_sample and nepal_sample  
    count = 0
```

```
    repeat 10,000 times:
```

with replacement!

```
        bhutan_resample = draw N resamples from the uni_sample
```

```
        nepal_resample = draw M resamples from the uni_sample
```

```
        muBhutan = sample mean of the bhutan_resample
```

```
        muNepal = sample mean of the nepal_resample
```

```
        diff = |muNepal - muBhutan|
```

```
        if diff >= observed_diff:
```

```
            count += 1
```

```
pValue = count / 10,000
```

Bootstrap



Let's try it!

Null hypothesis test

Bhutan Happiness
4.44
3.36
5.87
2.31
...
3.70

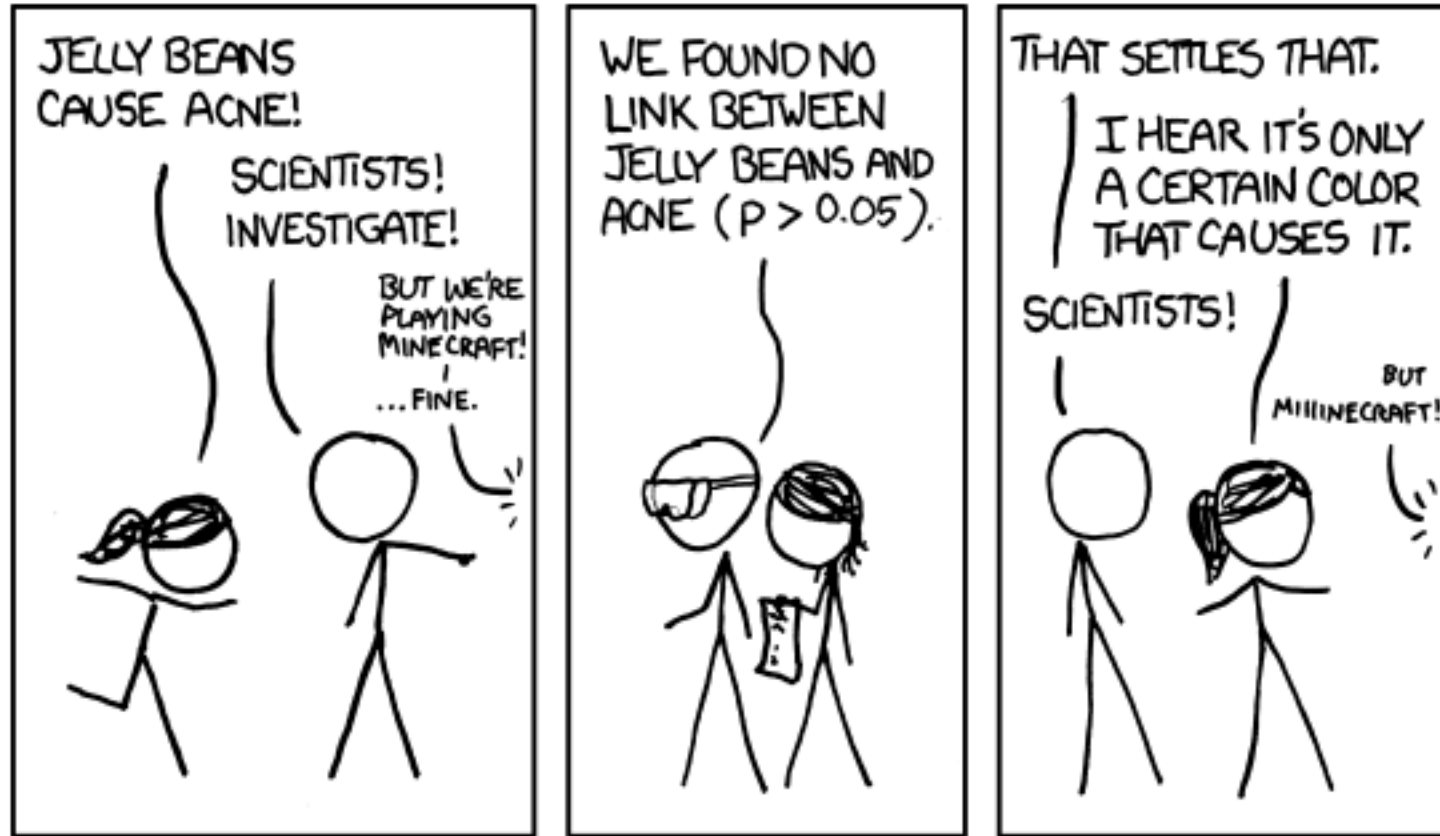
$$\bar{X}_1 = 3.1$$

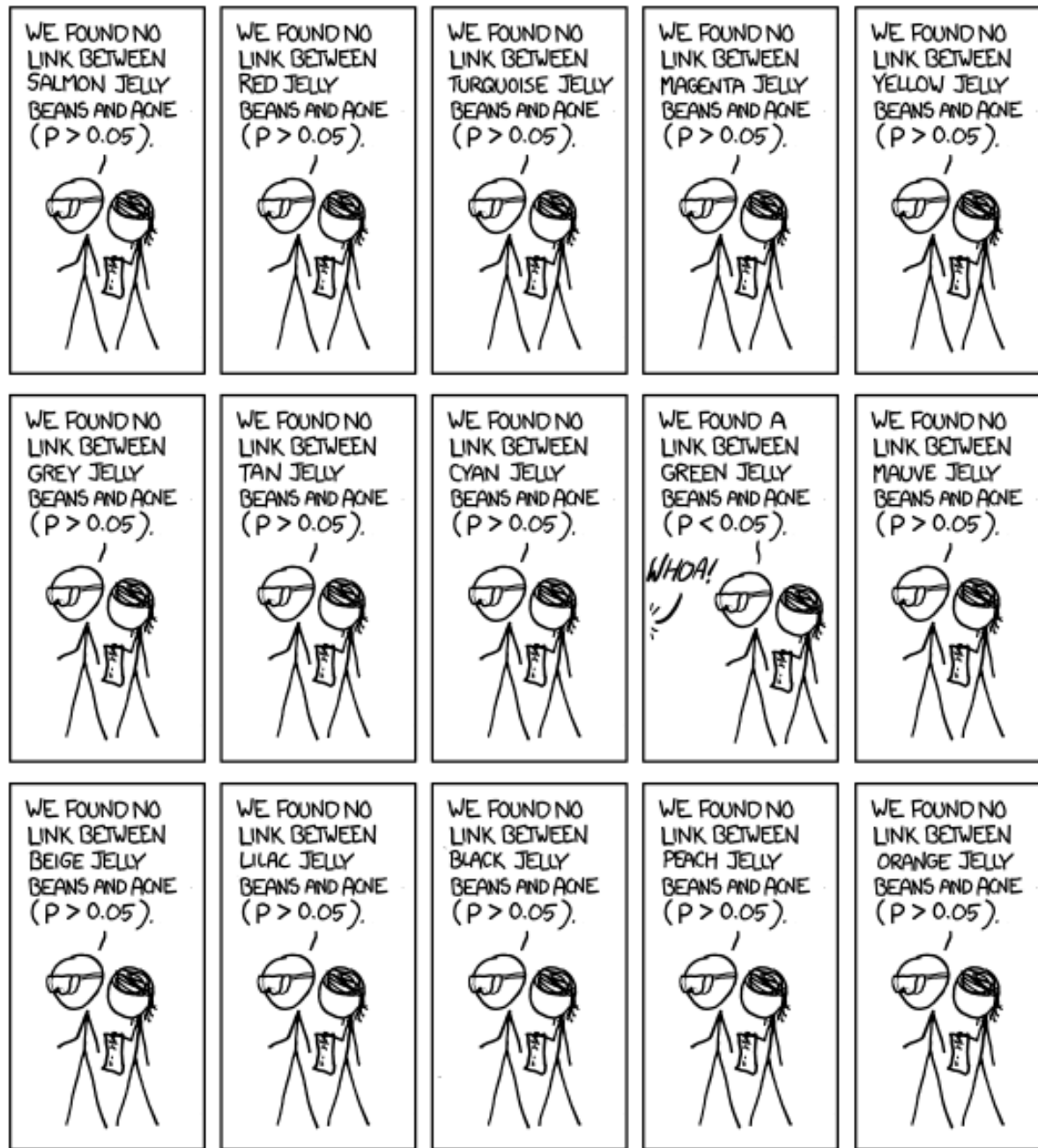
Nepal Happiness
4.44
3.36
5.87
2.31
...
3.70

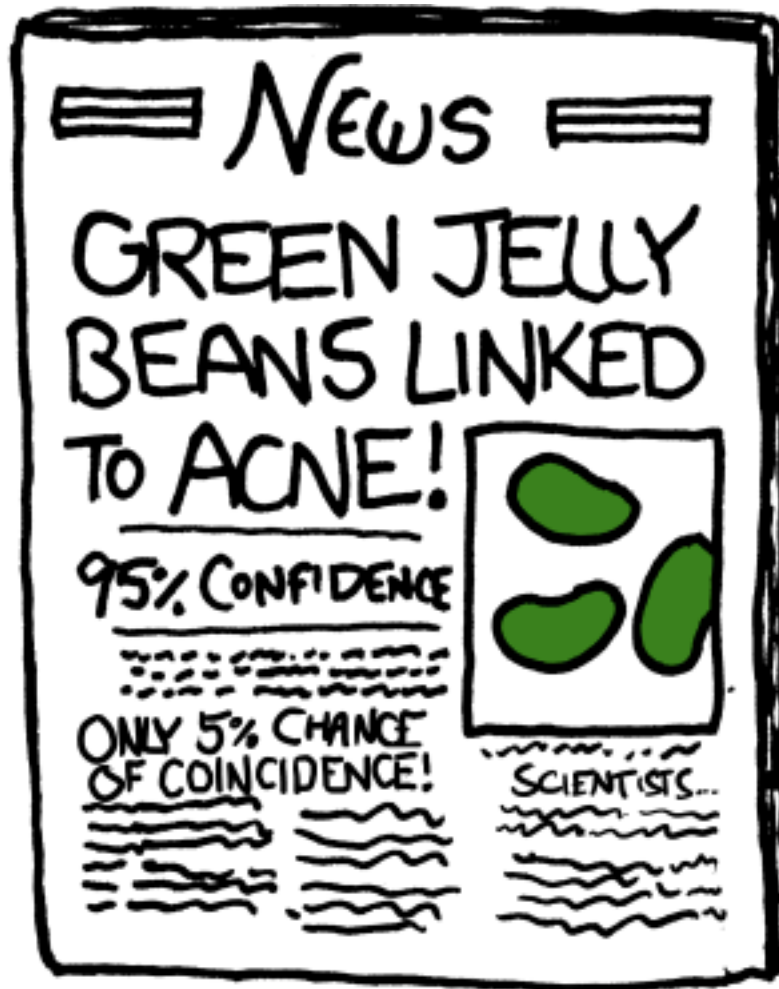
$$\bar{X}_2 = 2.4$$

Claim: The happiness of Nepal and Bhutan have a 0.7 difference of means, and this is significant ($p < 0.01$).

Joke (aka how not to do p-values, courtesy of XKCD)



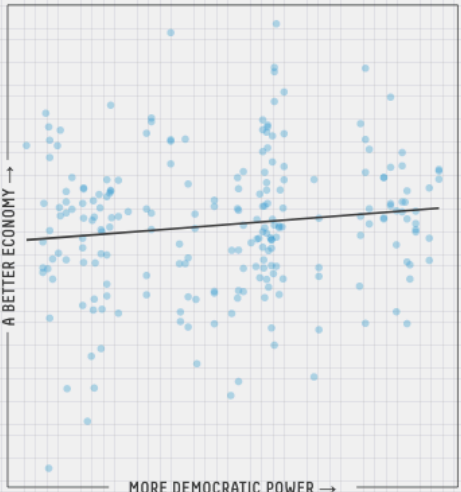





Ethics in Probability

Hack Your Way To Scientific Glory

You're a social scientist with a hunch: **The U.S. economy is affected by whether Republicans or Democrats are in office.** Try to show that a connection exists, using real data going back to 1948. For your results to be publishable in an academic journal, you'll need to prove that they are "statistically significant" by achieving a low enough p-value.

- 1 CHOOSE A POLITICAL PARTY**
Republicans Democrats
- 2 DEFINE TERMS**
Which politicians do you want to include?
 Presidents
 Governors
 Senators
 Representatives
How do you want to measure economic performance?
 Employment
 Inflation
 GDP
 Stock prices
Other options
 Factor in power
Weight more powerful positions more heavily
 Exclude recessions
Don't include economic recessions
- 3 IS THERE A RELATIONSHIP?**
Given how you've defined your terms, does the economy do better, worse or about the same when more Democrats are in power? Each dot below represents one month of data.

- 4 IS YOUR RESULT SIGNIFICANT?**
If there were no connection between the economy and politics, what is the probability that you'd get results at least as strong as yours? That probability is your p-value, and by convention, you need a p-value of 0.05 or less to get published.

Result: Almost
Your **0.06** p-value is close to the 0.05 threshold. Try tweaking your variables to see if you can push it over the line!

If you're interested in reading real (and more rigorous) studies on the connection between politics and the economy, see the work of Larry Bartels and Alan Blinder and Mark Watson.
Data from The @unitedstates Project, National Governors Association, Bureau of Labor Statistics, Federal Reserve Bank of St. Louis and Yahoo Finance.

“In February, Basic and Applied Social Psychology announced that it will no longer publish p-values. “We believe that the $p < .05$ bar is too easy to pass and sometimes serves as an excuse for lower quality research,” the editors wrote in their announcement... The important lesson here is that a single analysis is not sufficient to find a definitive answer.”

<https://fivethirtyeight.com/features/science-isnt-broken/#part1>