Bootstrapping

Chris Piech
CS109, Stanford University

Improvement Between Midterm and Final

Bad midterm? The final can show me you have learned

Regrade Requests

Good use: TA missed my solution

Won't change the rubric

Before Monday Nov 11th 9a

Right to regrade your whole midterm

Today, we do science!

A real difference?

Learning in Context A	Learning in Context B
4.44	2.15
3.36	3.01
5.87	2.02
2.31	1.43
…	…
3.70	1.83

18 students

23 students

$\mu_1 = 3.1$

$\mu_2 = 2.4$

Claim: Group 1 and Group 2 are samples from different distributions with a 0.7 difference of means.

How confident are you in this claim?

The Classic Science Test

Group 1
4.44
3.36
5.87
2.31
…
3.70

Group 2
2.15
3.01
2.02
1.43
…
1.83

$\mu_1 = 3.1$

$\mu_2 = 2.4$

Claim: Group 1 and Group 2 are samples from **different distributions** with a 0.7 difference of means.

How confident are you in this claim?

GPT-4 Tells about the Importance of Bootstrapping

GPT-4

PI

Can you name 10 statistical tests that a datascientist should know?

PI

If you had enough samples, which tests could be replaced with bootstrapping?

GPT-4 Tells about the Importance of Bootstrapping

2 / 2

PI

Can you name 10 statistical tests that a datascientist should know?

Certainly! Here's a list of 10 statistical tests that are commonly used in data science, along with a brief description of when they might be used:

1. **Pearson's Chi-squared test**: Used for testing relationships between categorical variables.
2. **T-test (Independent and Paired)**: The Independent samples t-test compares the means of two independent groups in order to determine whether there is statistical evidence that the associated population means are significantly different. The Paired sample t-test is used to compare the means from the same group at different times.
3. **ANOVA (Analysis of Variance)**: Compares the means among three or more groups to understand if at least one group mean is statistically different from the others.
4. **Linear Regression**: Used for predicting the value of a variable based on the value of another variable. The variable you want to predict is called the dependent variable, and the one you are using to predict the other variable's value is called the independent variable.
5. **Logistic Regression**: Used when the dependent variable is categorical, to predict the outcome of a categorical dependent variable based on one or more predictor variables.
6. **Wilcoxon Signed-Rank Test**: A non-parametric test that compares two paired groups. It's used as an alternative to the paired t-test when the data does not meet the assumptions required by the t-test.
7. **Mann-Whitney U Test**: A non-parametric test for assessing whether two independent samples of observations come from the same distribution. It is an alternative to the independent sample t-test when the data does not satisfy the assumptions of the t-test.
8. **Kruskal-Wallis H Test**: A non-parametric version of ANOVA, which is used when the assumptions of ANOVA are not met. It is used to compare more than two groups.
9. **Fisher's Exact Test**: Used for testing the association between two categorical variables in small sample sizes. It's an alternative to the chi-squared test when the sample size is too small to meet the chi-squared test's assumptions.
10. **Time Series Analysis (e.g., ARIMA)**: Not a test per se, but a suite of methods for analyzing time series data, which can include tests for stationarity (ADF Test), seasonality, autocorrelation, and the ability to forecast future values from past data.

Regenerate

PI

If you had enough samples, which tests could be replaced with bootstrapping?

Bootstrapping is a powerful resampling method that can be used to estimate the distribution of a statistic (like the mean, median, or a regression coefficient) by sampling with replacement from the data. When you have a sufficiently large sample size, bootstrapping can be used to relax the assumptions of traditional parametric tests, like the normality assumption or the assumption of equal variances across groups.

In principle, any statistical test could be replaced with a bootstrap equivalent if you're interested in hypothesis testing or constructing confidence intervals, especially if the assumptions of the traditional test are questionable. However, it's particularly common to use bootstrapping for the following tests:

1. **T-test**: Both independent and paired t-tests can be replaced with bootstrap methods to estimate the difference in means without assuming normal distributions of the sample means.
2. **ANOVA**: Bootstrap methods can be used to assess the difference in means across groups without assuming normally distributed residuals or homogeneity of variance.
3. **Linear Regression**: Instead of assuming the normality of residuals, you can use bootstrapping to construct confidence intervals for the regression coefficients.
4. **Logistic Regression**: Similarly, bootstrapping can be used to obtain more robust confidence intervals for the coefficients in a logistic regression model.
5. **Wilcoxon Signed-Rank Test and Mann-Whitney U Test**: While these are non-parametric tests that do not assume normality, bootstrapping can still be used to obtain confidence intervals for the median difference or other statistics of interest.
6. **Kruskal-Wallis H Test**: Bootstrapping can be applied to obtain the distribution of the test statistic under the null hypothesis, and thus get a p-value without the assumption of identical distribution shapes across groups.
7. **Fisher's Exact Test**: When sample sizes are large, the exact calculations required for Fisher's Exact Test can become computationally intensive. Bootstrapping can be a practical alternative to obtain an approximation.

GPT-4 Tells about the Importance of Bootstrapping

2 / 2

PI Can you name 10 statistical tests that a datascientist should know?

Certainly! Here's a list of 10 statistical tests that are commonly used in data science, along with a brief description of when they might be used:

1. **Pearson's Chi-squared test**: Used for testing relationships between categorical variables.
2. **T-test (Independent and Paired)**: The Independent samples t-test compares the means of two independent groups in order to determine whether there is statistical evidence that the associated population means are significantly different. The Paired sample t-test is used to compare the means from the same group at different times.
3. **ANOVA (Analysis of Variance)**: Compares the means among three or more groups to understand if at least one group mean is statistically different from the others.
4. **Linear Regression**: Used for predicting the value of a variable based on the value of another variable. The variable you want to predict is called the dependent variable, and the one you

small sample sizes. It's an alternative to the chi-squared test when the sample size is too small to meet the chi-squared test's assumptions.

10. **Time Series Analysis (e.g., ARIMA)**: Not a test per se, but a suite of methods for analyzing time series data, which can include tests for stationarity (ADF Test), seasonality, autocorrelation, and the ability to forecast future values from past data.

Regenerate

PI If you had enough samples, which tests could be replaced with bootstrapping?

Bootstrapping is a powerful resampling method that can be used to estimate the distribution of a statistic (like the mean, median, or a regression coefficient) by sampling with replacement from the data. When you have a sufficiently large sample size, bootstrapping can be used to relax the assumptions of traditional parametric tests, like the normality assumption or the assumption of equal variances across groups.

In principle, any statistical test could be replaced with a bootstrap equivalent if you're interested in hypothesis testing or constructing confidence intervals, especially if the assumptions of the traditional test are questionable. However, it's particularly common to use bootstrapping for the following tests:

1. **T-test**: Both independent and paired t-tests can be replaced with bootstrap methods to

statistic under the null hypothesis, and thus get a p-value without the assumption of identical distribution shapes across groups.

7. **Fisher's Exact Test**: When sample sizes are large, the exact calculations required for Fisher's Exact Test can become computationally intensive. Bootstrapping can be a practical alternative to obtain an approximation.

In principle, any statistical test could be replaced with a bootstrap equivalent if you're interested in hypothesis testing or constructing confidence intervals, especially if the assumptions of the traditional test are questionable. However, it's particularly common to use bootstrapping for the following tests:

In other words.

Datascientists **Hate Him!**

He bongclouded from level 400 data scientist to a level 3000 in hours

using this

one ~~weird~~ trick!!!

(Computer science)

Where are we in CS109?

You are here

Counting Theory

Core Probability

Random Variables

Probabilistic Models

Uncertainty Theory

Machine Learning

Uncertainty Theory

Beta Distributions	Thompson Sampling	Adding Random Vars	Central Limit Theorem
Sampling	Bootstrapping	Algorithmic Analysis	Information Theory + Divergence

As requested by AI faculty

<review>

Central Limit Theorem (Summation)

Consider n independent and identically distributed **(i.i.d)** variables $X_1, X_2, \ldots, X_n$ with $E[X_i] = \mu$ and $\text{Var}(X_i) = \sigma^2$.

As $n \to \infty$

$$\sum_{i=1}^{n} X_i \sim \mathcal{N}(n\mu, n\sigma^2)$$

The **sum** of the variables is normally distributed

Central Limit Theorem (Average)

Consider n independent and identically distributed (i.i.d) variables $X_1, X_2, \dots, X_n$ with $E[X_i] = \mu$ and Var$(X_i) = \sigma^2$.

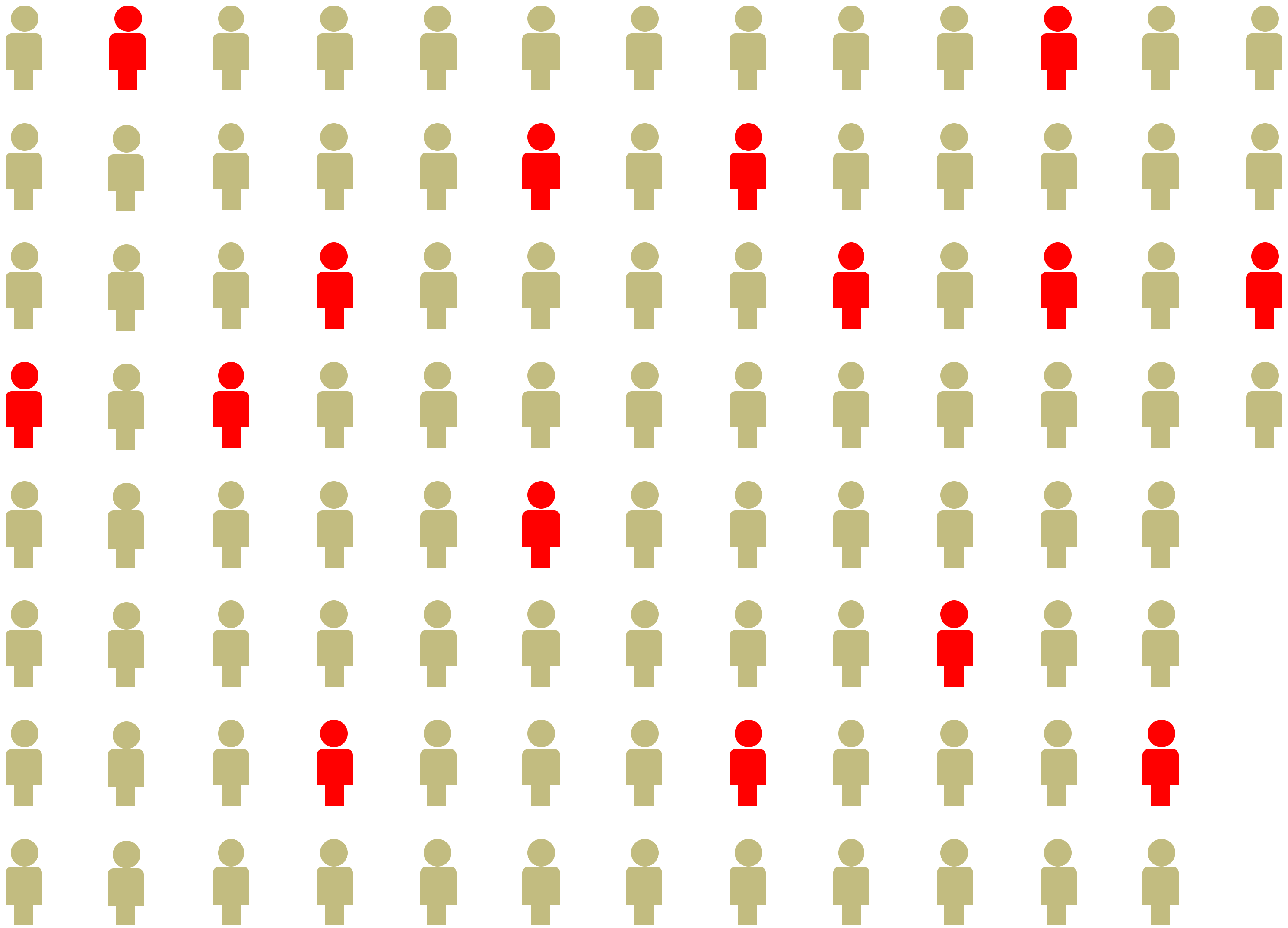
As $n \to \infty$

$$\frac{1}{n}\sum_{i=1}^{n} X_i \sim \mathcal{N}(\mu, \frac{\sigma^2}{n})$$

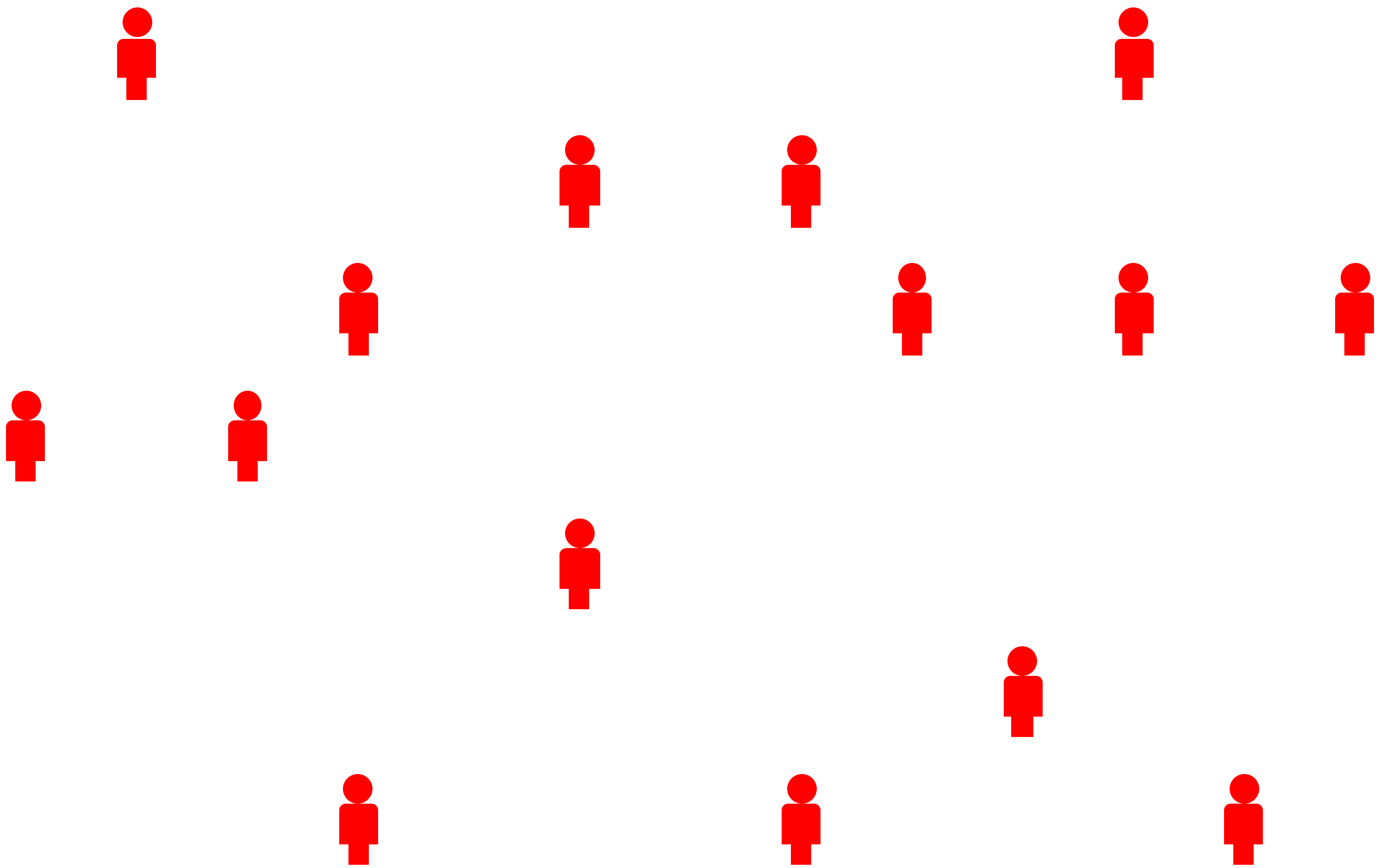
The **average** of the variables is normally distributed

Population

Sample



Sample



Collect one (or more) numbers from each person

Equations we used to get those values

sample mean estimate

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

Our best guess at the true mean

sample mean

sample variance estimate

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

Our best guess at the true variance

Std error of the mean estimate

$$\text{Std}(\bar{X}) = \sqrt{\frac{S^2}{n}}$$

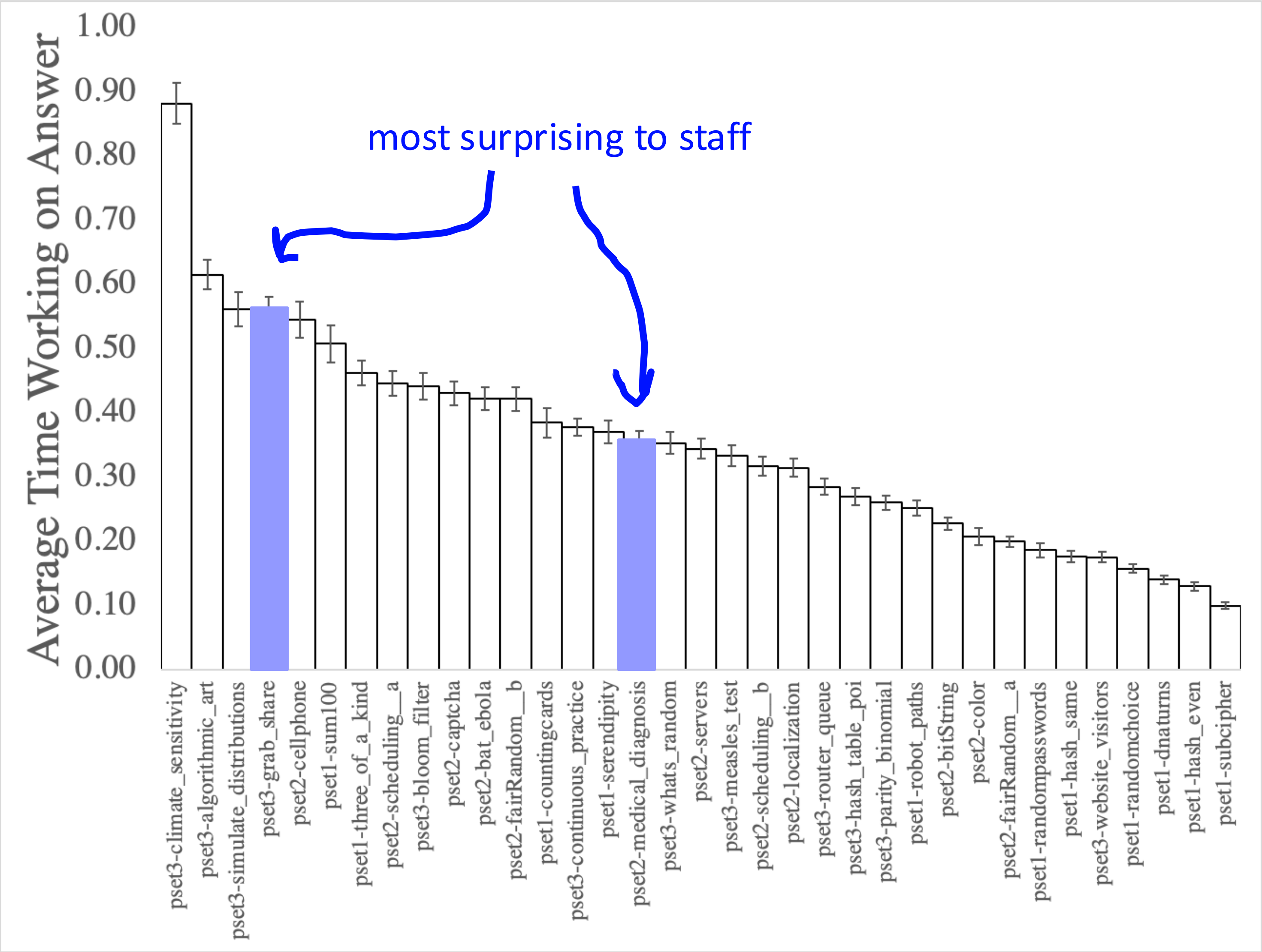
sample variance

How wrong do we think our mean estimate is?

Sample Mean and Standard Error for Song Averages

#	Song	Sample Mean PDF	Votes	NumVotes	SampleMean	SEOM	SongId	Pr(Top16)	Pr(Best)
1	Get Lucky - Daft Punk			45	3.82	0.18	117	0.851	0.500
2	Life is a Highway - Rascal Flatts			36	3.78	0.24	67	0.735	0.447
3	Let It Be - The Beatles			40	3.78	0.19	150	0.782	0.439
4	Upside Down - Jack Johnson			92	3.66	0.14	137	0.581	0.241
5	September - Earth, Wind & Fire			55	3.6	0.16	15	0.429	0.180
6	Time of Our Lives - Pitbull			24	3.58	0.26	39	0.424	0.224
7	Vienna - Billy Joel			24	3.58	0.28	140	0.427	0.235
8	Just the Two of Us (feat. Bill Withers) - Grover Washing			25	3.56	0.22	78	0.377	0.180
9	Voulez-Vous - ABBA			20	3.55	0.27	69	0.386	0.203
10	Let it Happen - Tame			22	3.55	0.29	73	0.392	0.214
11	Careless Whisper - George Michael			24	3.54	0.24	87	0.357	0.175
12	Take Five - Dave Brubeck			18	3.5	0.33	37	0.346	0.197
13	Clairo - Juna			18	3.5	0.28	57	0.324	0.168
14	We Are The Champions - Queen			22	3.5	0.24	77	0.294	0.143
15	All Star - Smash Mouth			17	3.41	0.37	0	0.273	0.160
16	Feel it Still - Portugal. The Man			17	3.41	0.32	68	0.247	0.132
17	Magnetic - ILLIT			15	3.4	0.38	16	0.272	0.159
18	The Dock of the Bay - Otis Redding			13	3.38	0.38	107	0.254	0.148

Sample Mean and Standard Error for Pset Time Working



Error bars are standard error of the mean

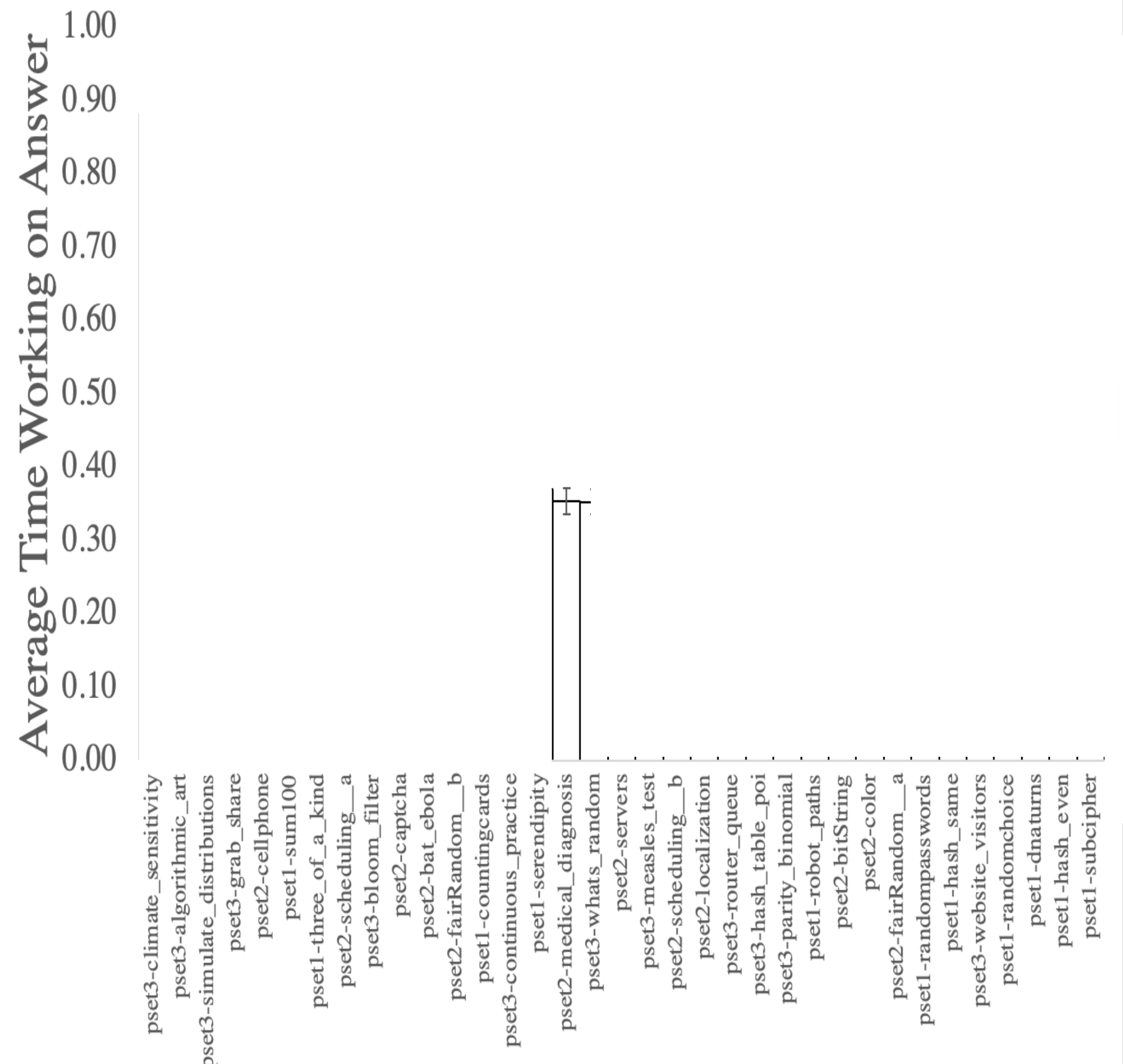
Expectation of the sum of problems is sum of expectations:

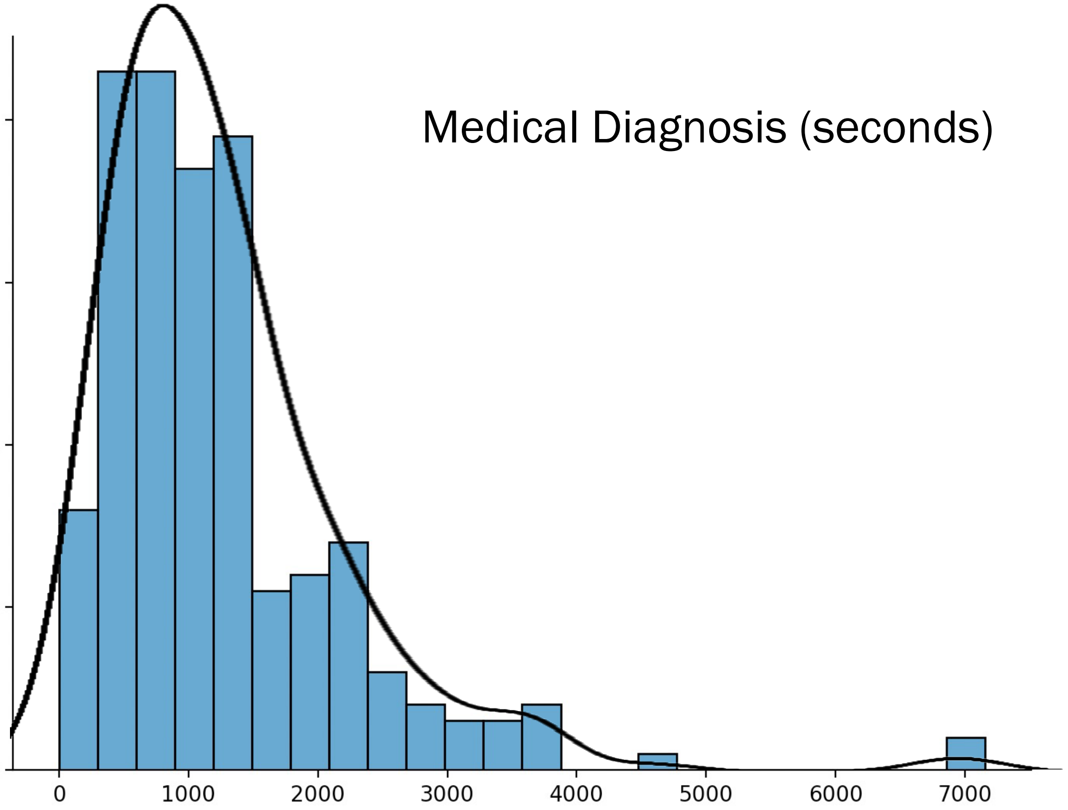
pset1: 2.87 hours on answers
pset2: 4.23 hours on answers
pset3: 5.11 hours on answers

Total: 12.1 hours on answers
Budget: 50 hours for psets

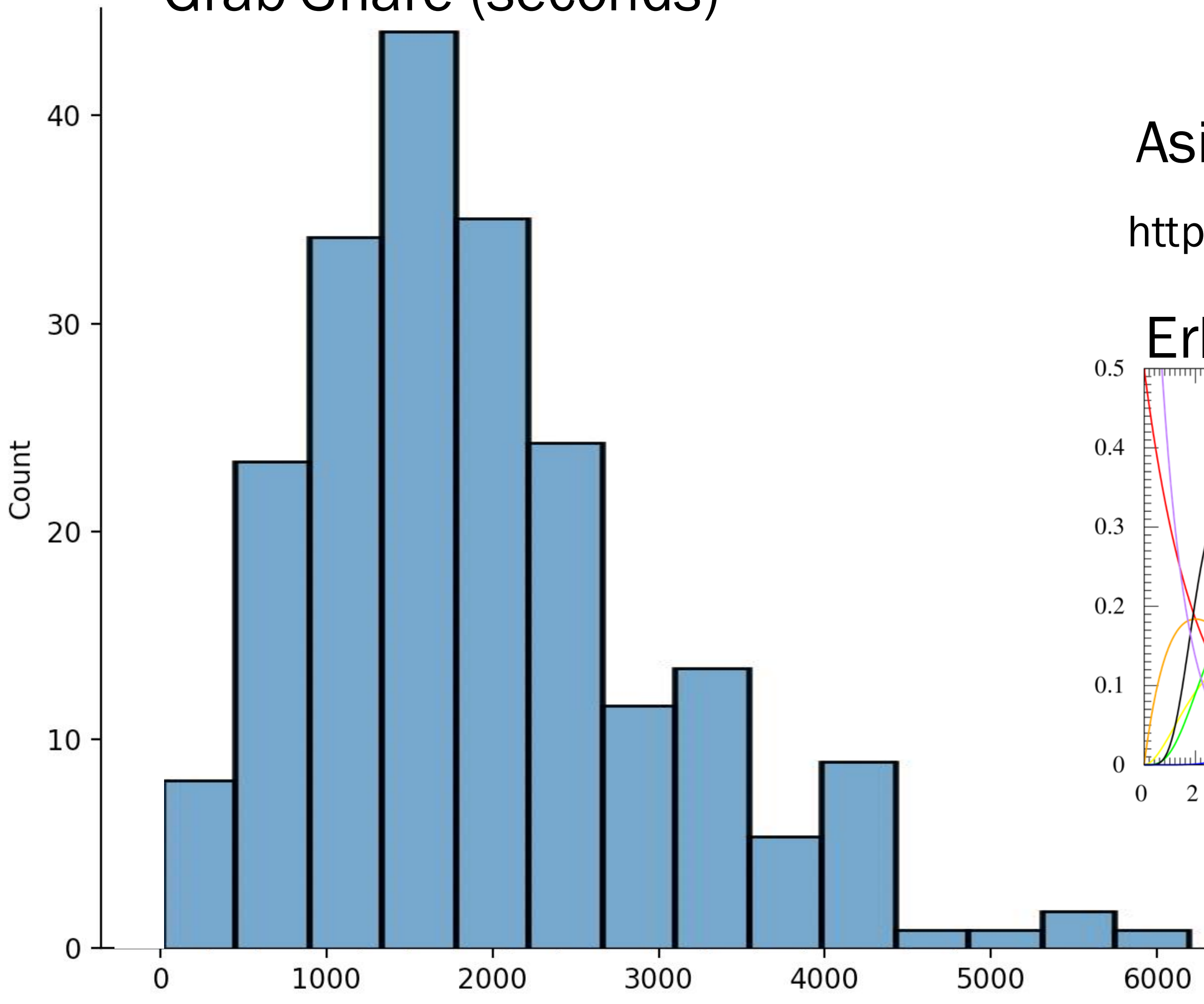
Statistics Vs Distribution

Sampling statistics vs Sampling distribution





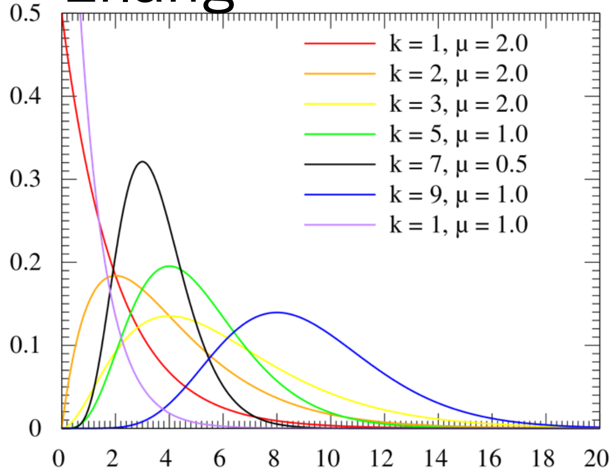
[Aside] Distribution of PSet Completion Times



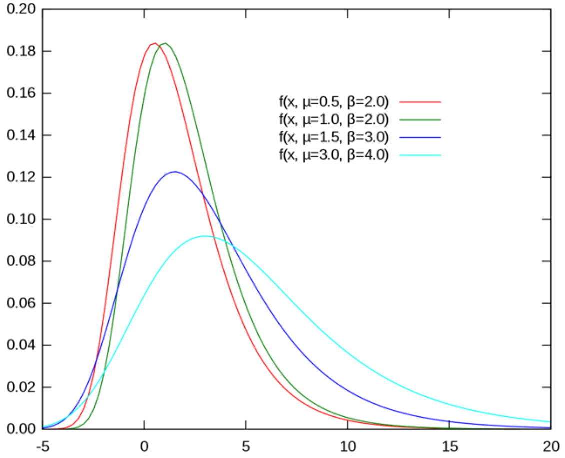
Aside: could be Erlang, could be Gumbel

https://en.wikipedia.org/wiki/Erlang_distribution

Erlang



Gumbel

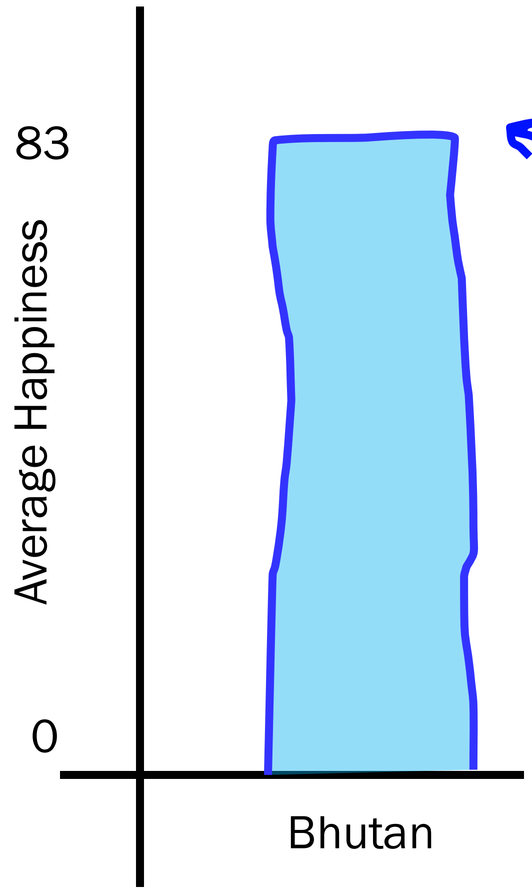


I don't think its exponential. Must not be a poisson process!

But what about Bhutan?

Our Report to Bhutan Government (after talking to 200 ppl)

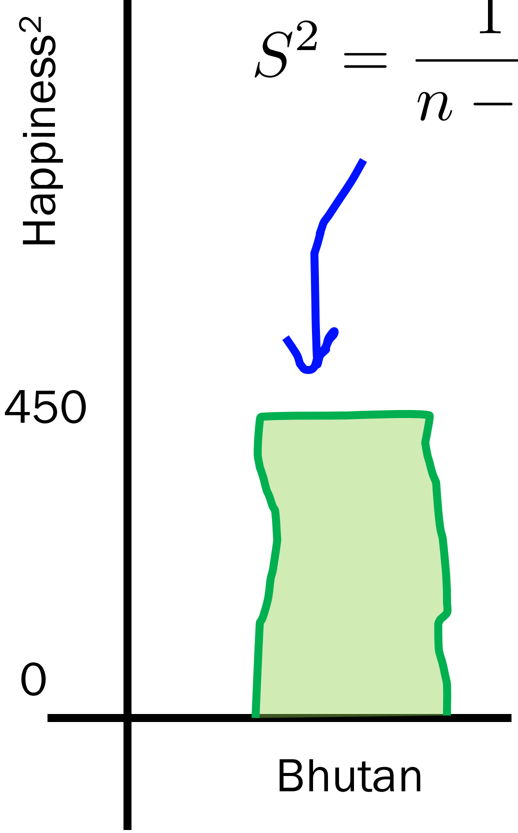
Average Happiness



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

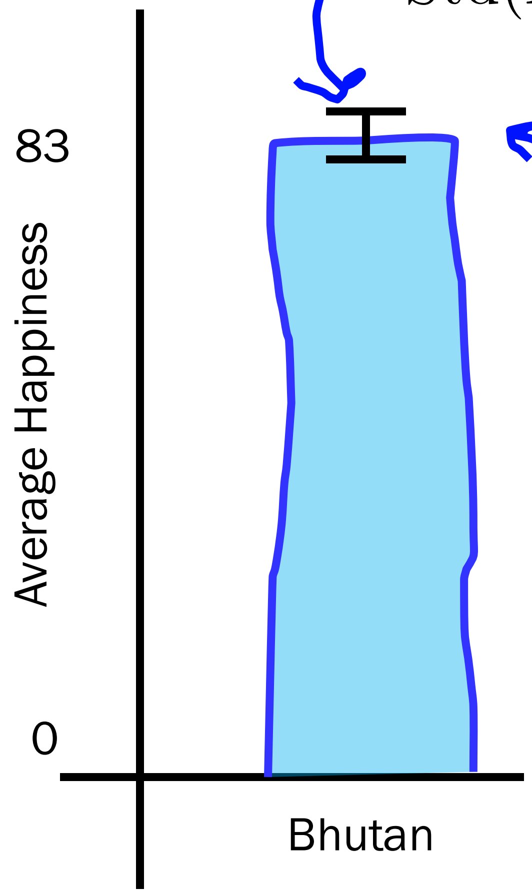
Variance of Happiness

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



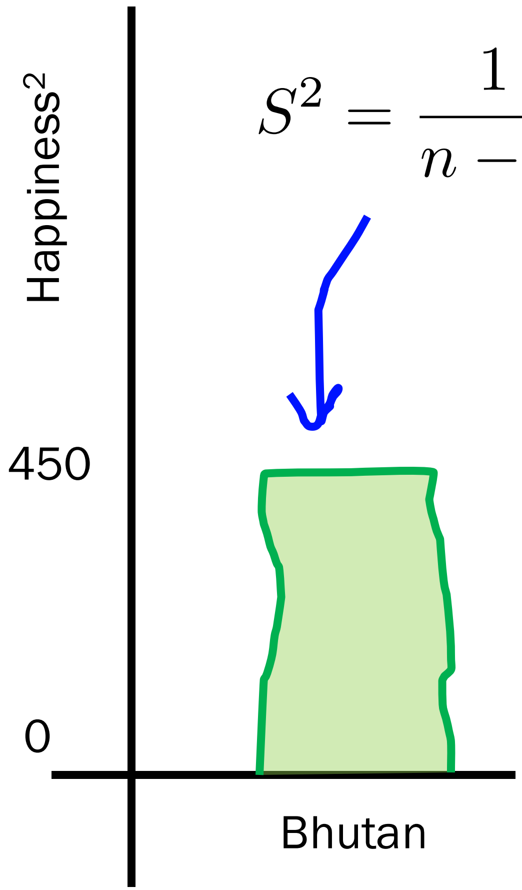
But what about **error bars**???

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



$$\text{Std}(\bar{X}) = \sqrt{\frac{S^2}{n}}$$

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



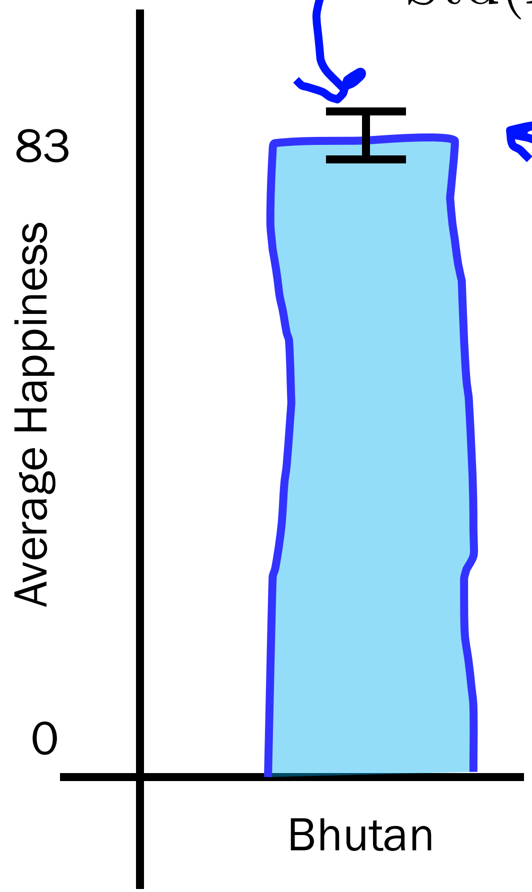
$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2$$

But what about **error bars**???

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

Average Happiness

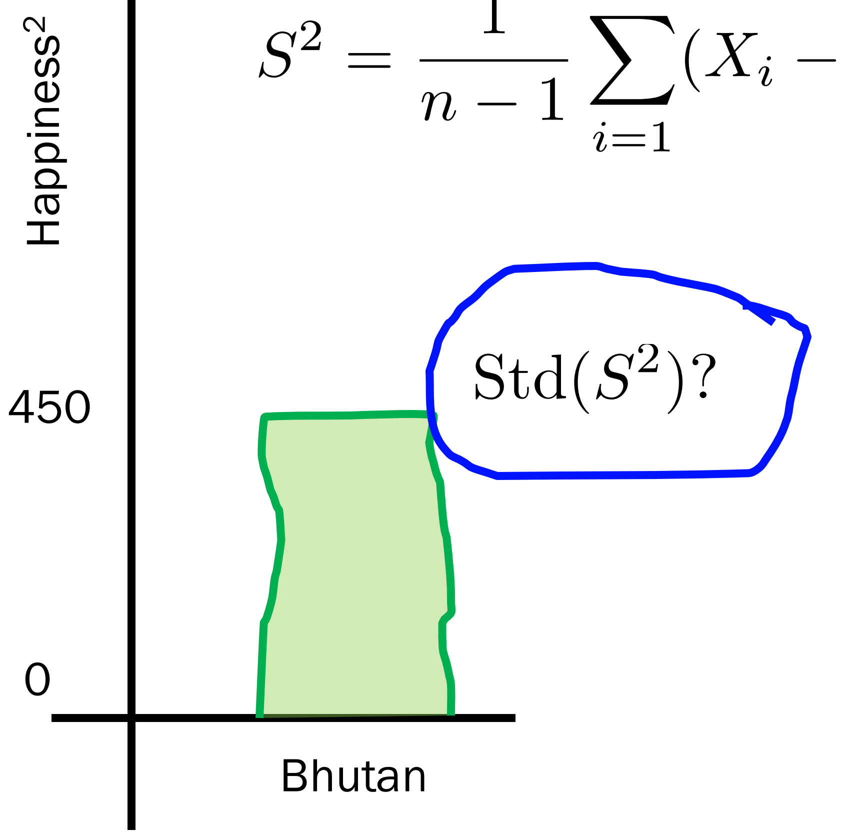
$$\text{Std}(\bar{X}) = \sqrt{\frac{S^2}{n}}$$



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

Variance of Happiness

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



Hypothetical – You have the underlying distribution!

How wrong is an estimate of **sample variance**, calculated from 200 people?

Plot twist: I give you the ***entire*** underlying distribution

Hypothetical – You have the underlying distribution!

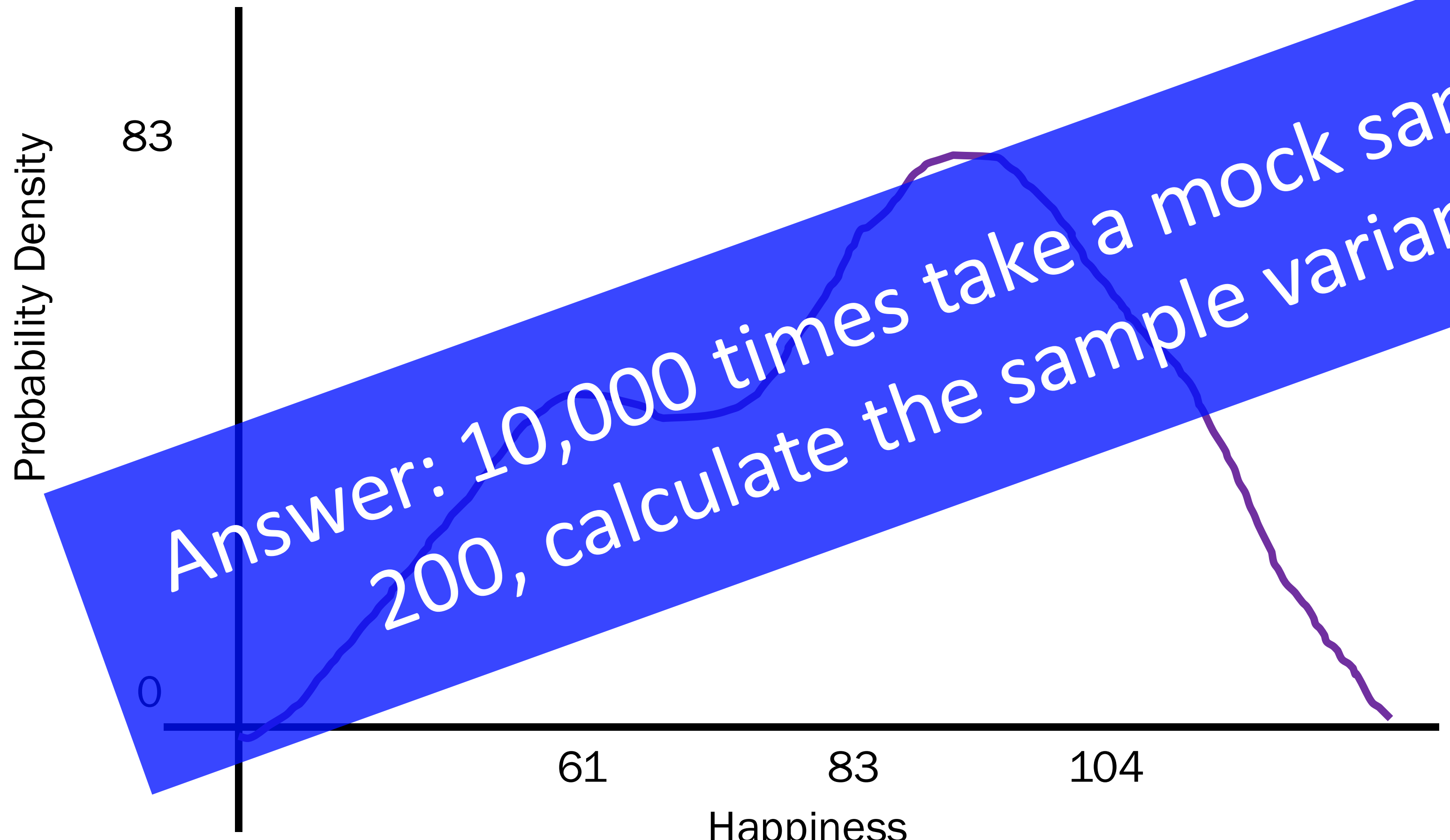
What is the **std** of the **sample variance**, calculated from 200 people?

Plot twist: I give you the ***entire*** underlying distribution

Hypothetical – You have the underlying distribution!

What is the **std** of the **sample variance**, calculated from 200 people?

Plot twist: I give you the ***entire*** underlying distribution



Answer: 10,000 times take a mock sample of 200, calculate the sample variance

Hypothetical – You have the underlying distribution!

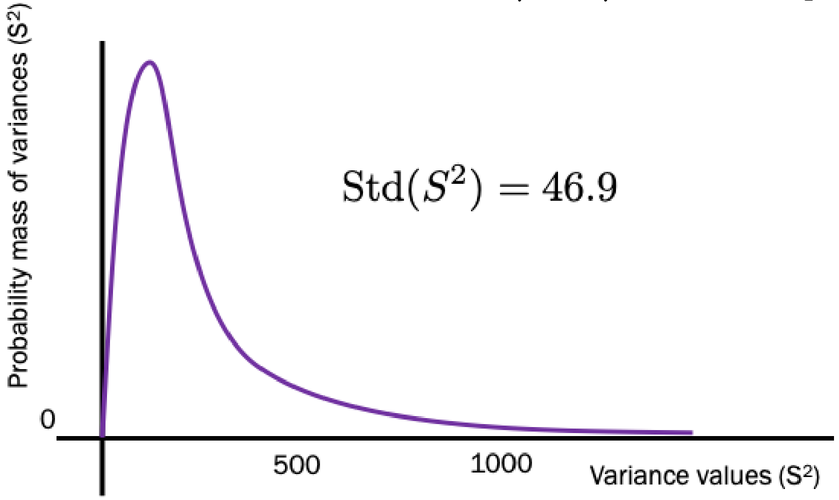
What is the **std** of the **sample variance**, calculated from 200 people?

```
brute_force_algorithm():
    # Estimate distribution of Sample Var with
    # infinite resources

    sample_vars = []
    Repeat 10,000 times:
        new_samples = collect_new_samples(n=200)
        sample_var = calculate_sample_var(new_samples)
        sample_vars.append(sample_var)

    # You now have a distribution of sample vars
```

```
sample_vars = [472.7, 478.4,
               469.2, …, 476.2]
```



<end review>

[suspense]

Bootstrap:
Probability for Computer Scientists

Bootstraping allows you to:

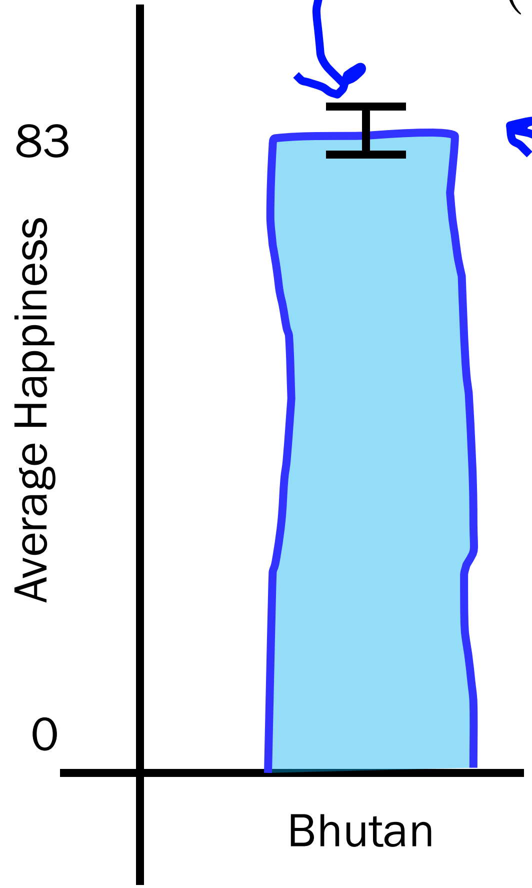
- Know the **distribution of *statistics***
- Calculate **p values**
- **Using computers**
- You totally **could have invented it**

But what about **error bars**???

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

Average Happiness

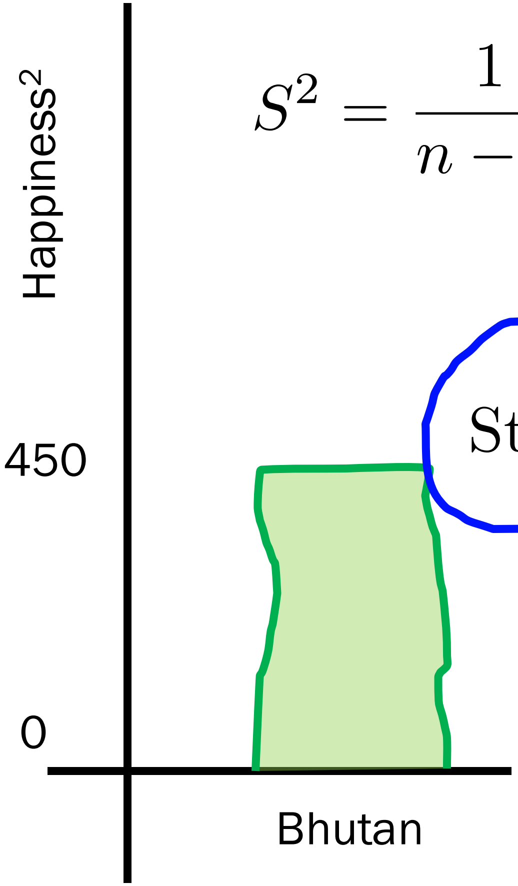
$$\text{Std}(\bar{X}) = \sqrt{\frac{S^2}{n}}$$



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

Variance of Happiness

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



$\text{Std}(S^2)$?

Hypothetical – You have the underlying distribution!

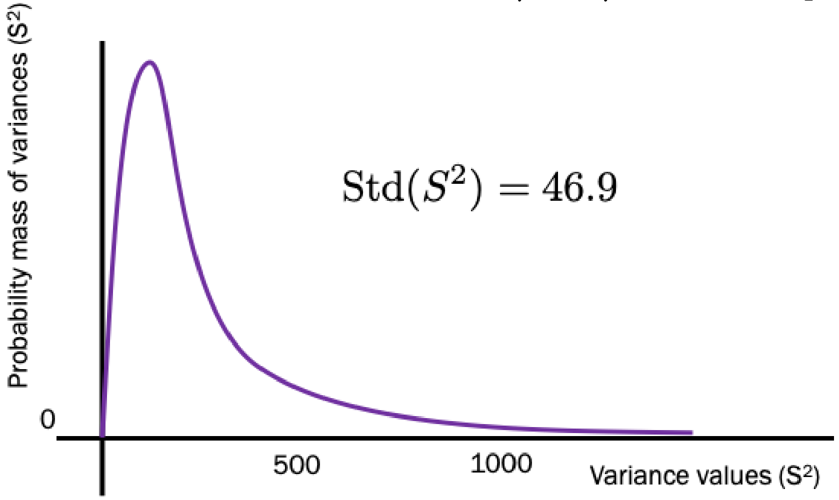
What is the **std** of the **sample variance**, calculated from 200 people?

```
brute_force_algorithm():
    # Estimate distribution of Sample Var with
    # infinite resources

    sample_vars = []
    Repeat 10,000 times:
        new_samples = collect_new_samples(n=200)
        sample_var = calculate_sample_var(new_samples)
        sample_vars.append(sample_var)

    # You now have a distribution of sample vars
```

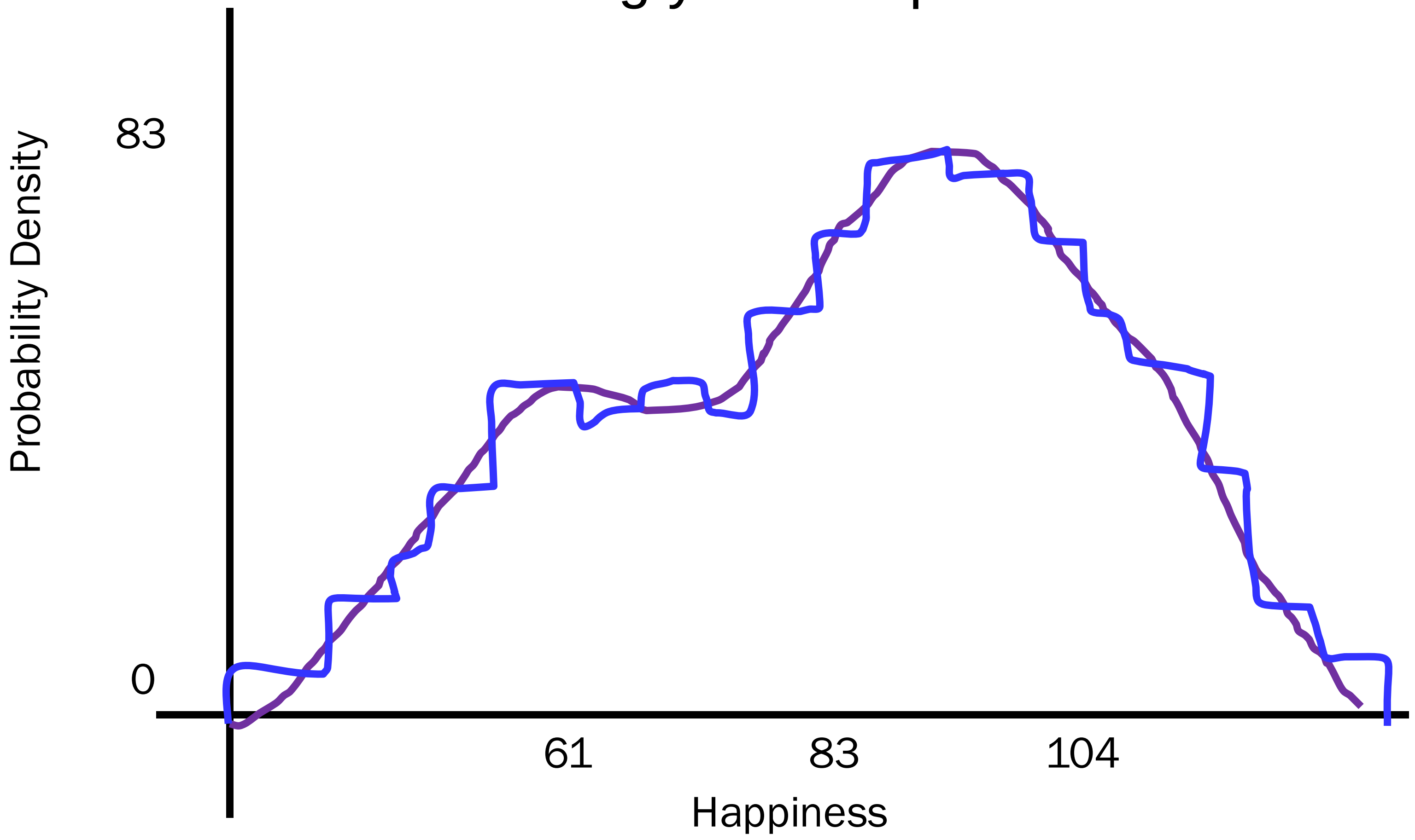
```
sample_vars = [472.7, 478.4,
               469.2, …, 476.2]
```



Here comes the award winning idea….

But Wait – What If You Actually Have a Good Estimate?

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



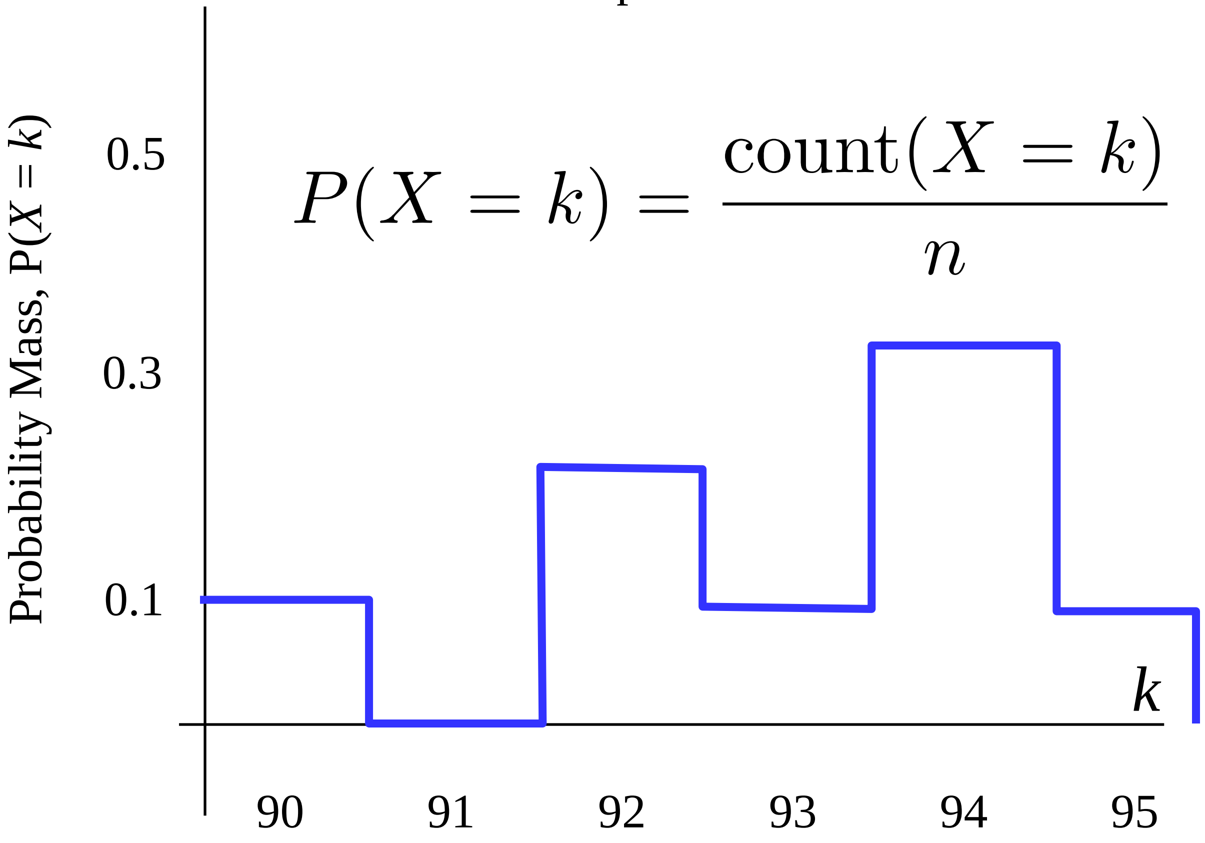
* This is just a histogram of your data!!

Key Insight

IID Samples

90,
92,
92,
93,
94,
94,
94,
95,

Sample Distribution



$$P(X = k) = \frac{\text{count}(X = k)}{n}$$

Bootstrapping Assumption

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

The underlying distribution

The sample distribution

(aka the histogram of your data)

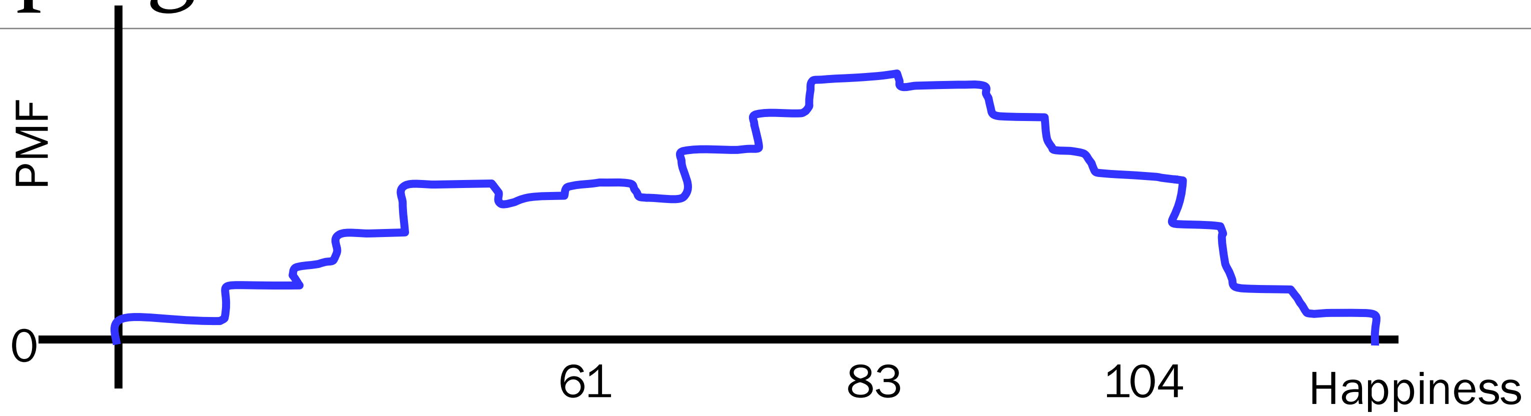
Algorithm

```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the stat on the resample
  You now have a distribution of your stat
```

Bootstrapping of Variance

```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the variance on the resample
  You now have a distribution of your variances
```

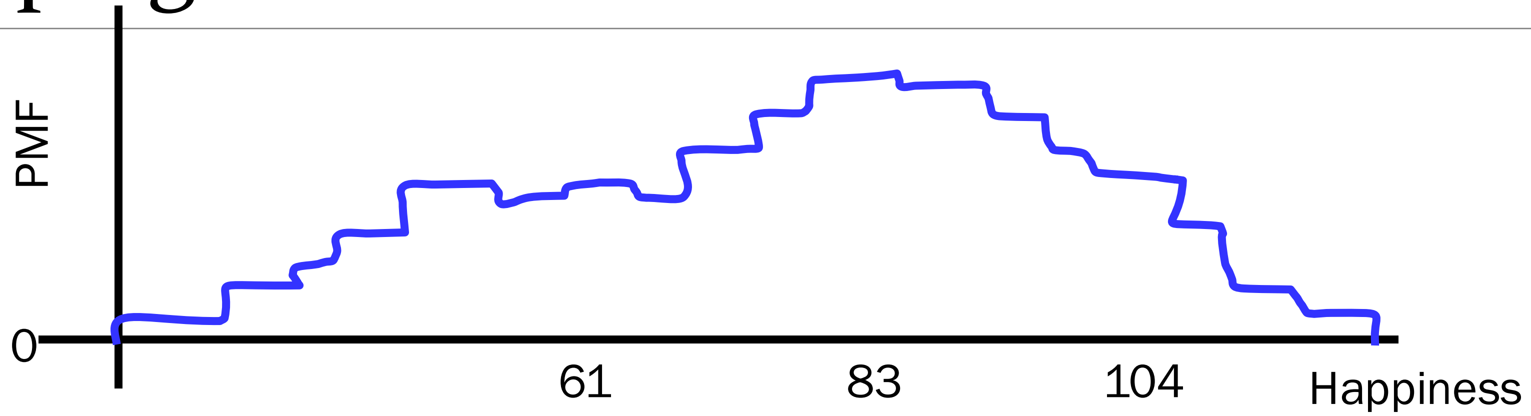
Bootstrapping of Variance



Bootstrap Algorithm (sample):

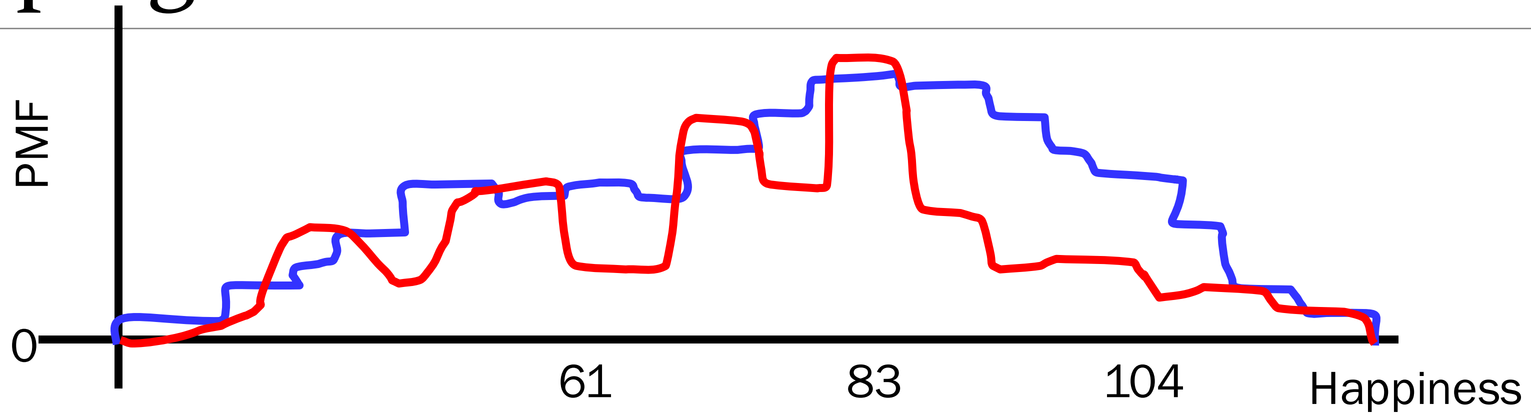
```
Estimate the PMF using the sample
Repeat 10,000 times:
  a. Draw len(sample) new samples from PMF
  b. Recalculate the var on the resample
You now have a distribution of your vars
```

Bootstrapping of Variance



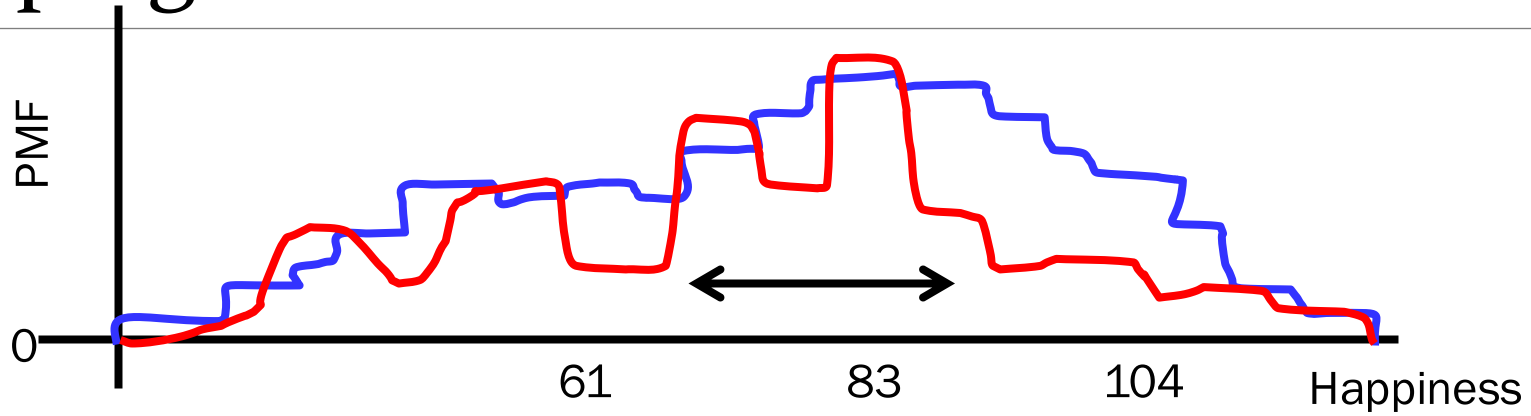

```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the var on the resample
  You now have a distribution of your vars
```

Bootstrapping of Variance




```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the var on the resample
  You now have a distribution of your vars
```

Bootstrapping of Variance

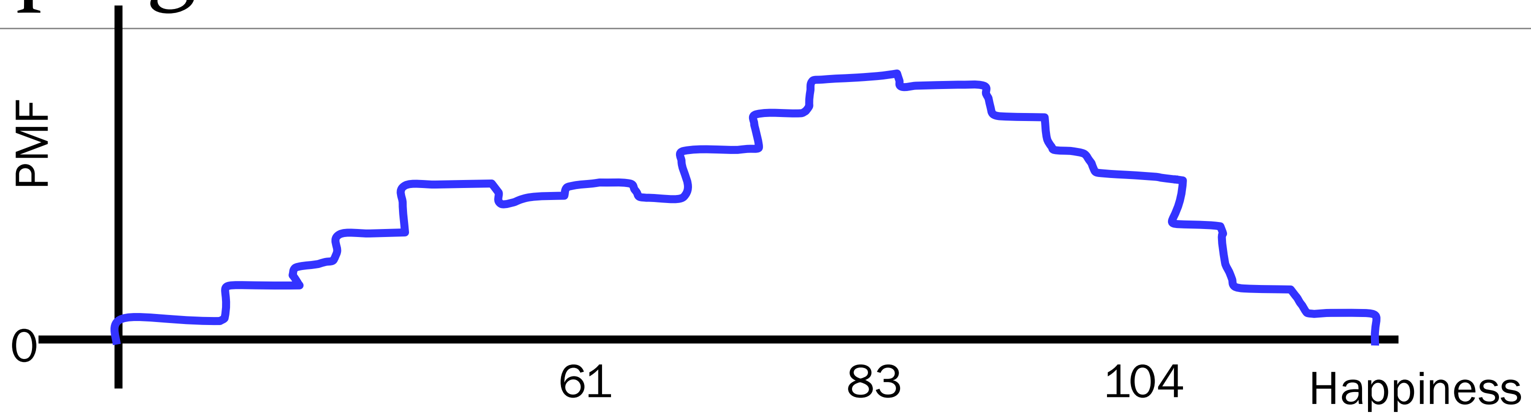


Bootstrap Algorithm (sample):

```
Estimate the PMF using the sample
Repeat 10,000 times:
  a. Draw len(sample) new samples from PMF
  b. Recalculate the vars on the resample
You now have a distribution of your vars
```

Vars = [472.7]

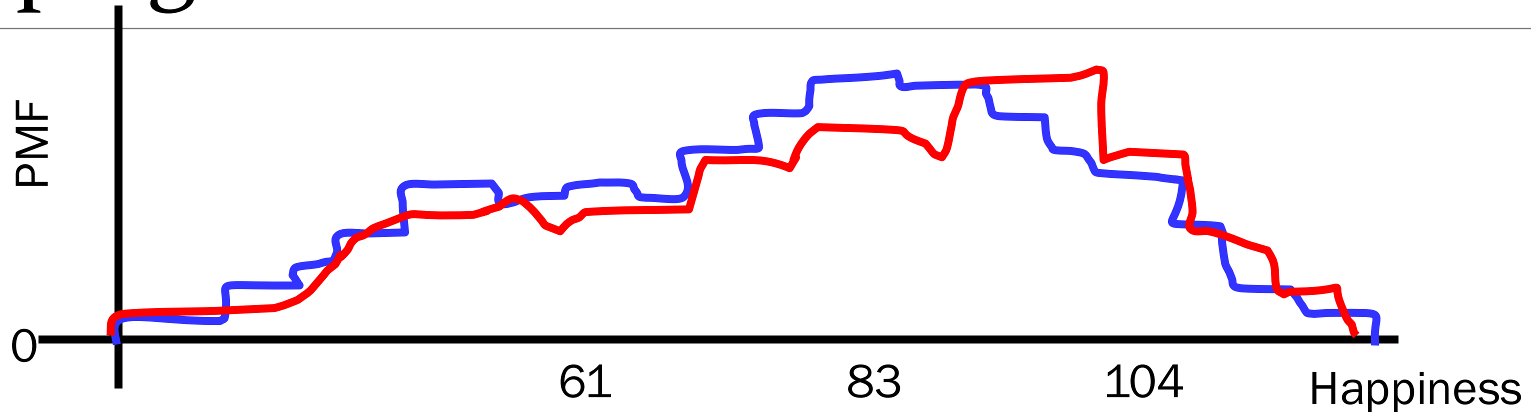
Bootstrapping of Variance




```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the var on the resample
  You now have a distribution of your vars
```

Vars = [472.7]

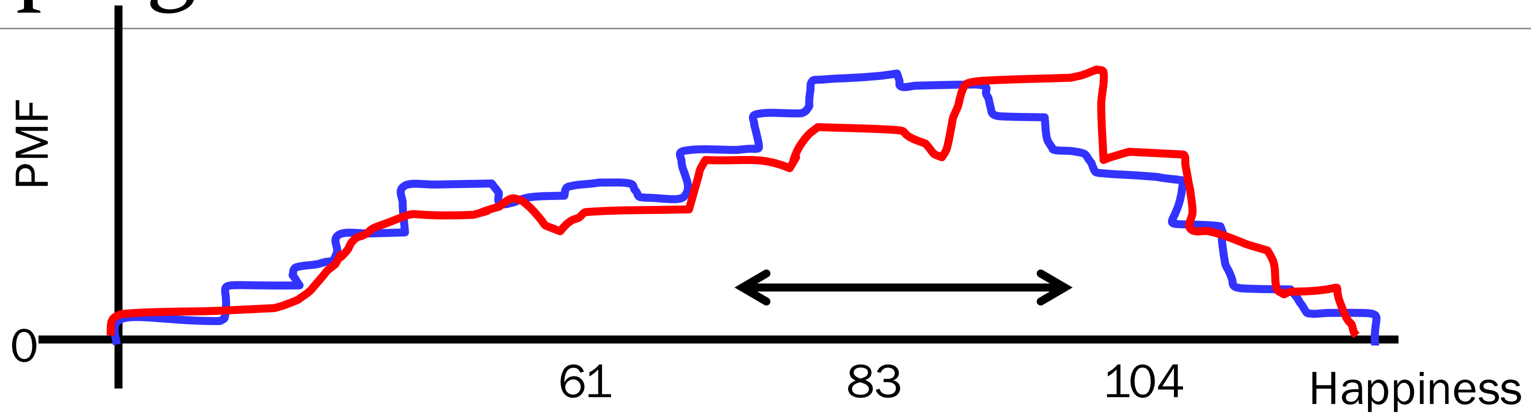
Bootstrapping of Variance




```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the var on the resample
  You now have a distribution of your vars
```

Vars = [472.7]

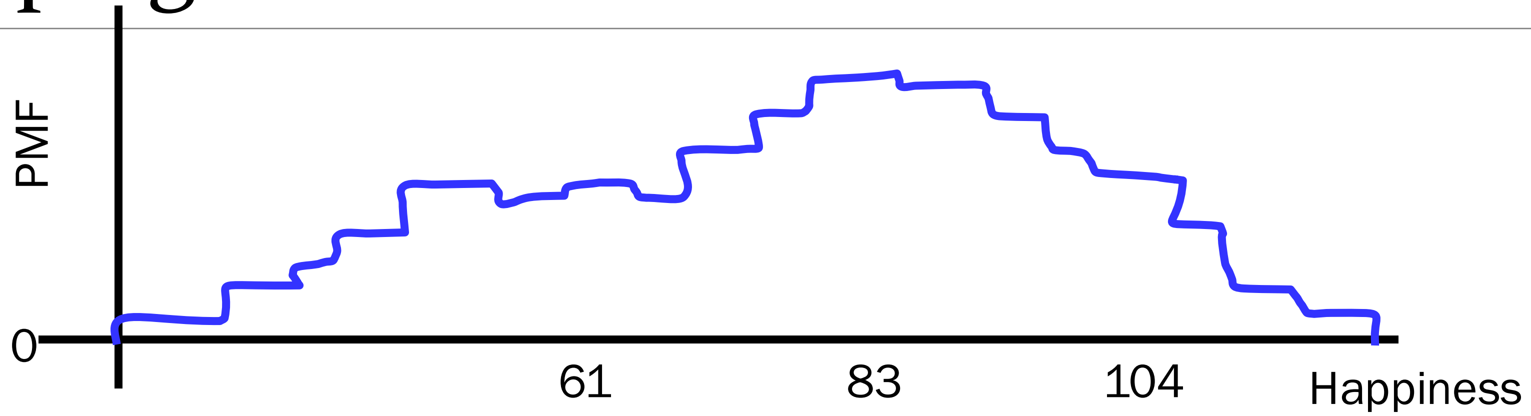
Bootstrapping of Variance




```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the var on the resample
  You now have a distribution of your vars
```

Vars = [472.7, 478.4]

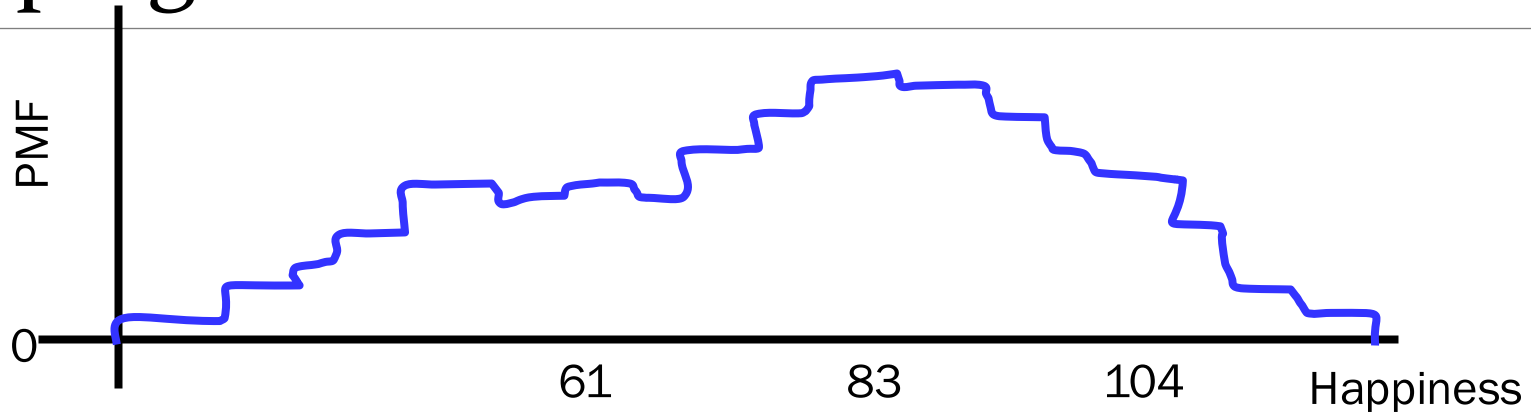
Bootstrapping of Variance




```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the var on the resample
  You now have a distribution of your vars
```

Vars = [472.7, 478.4]

Bootstrapping of Variance

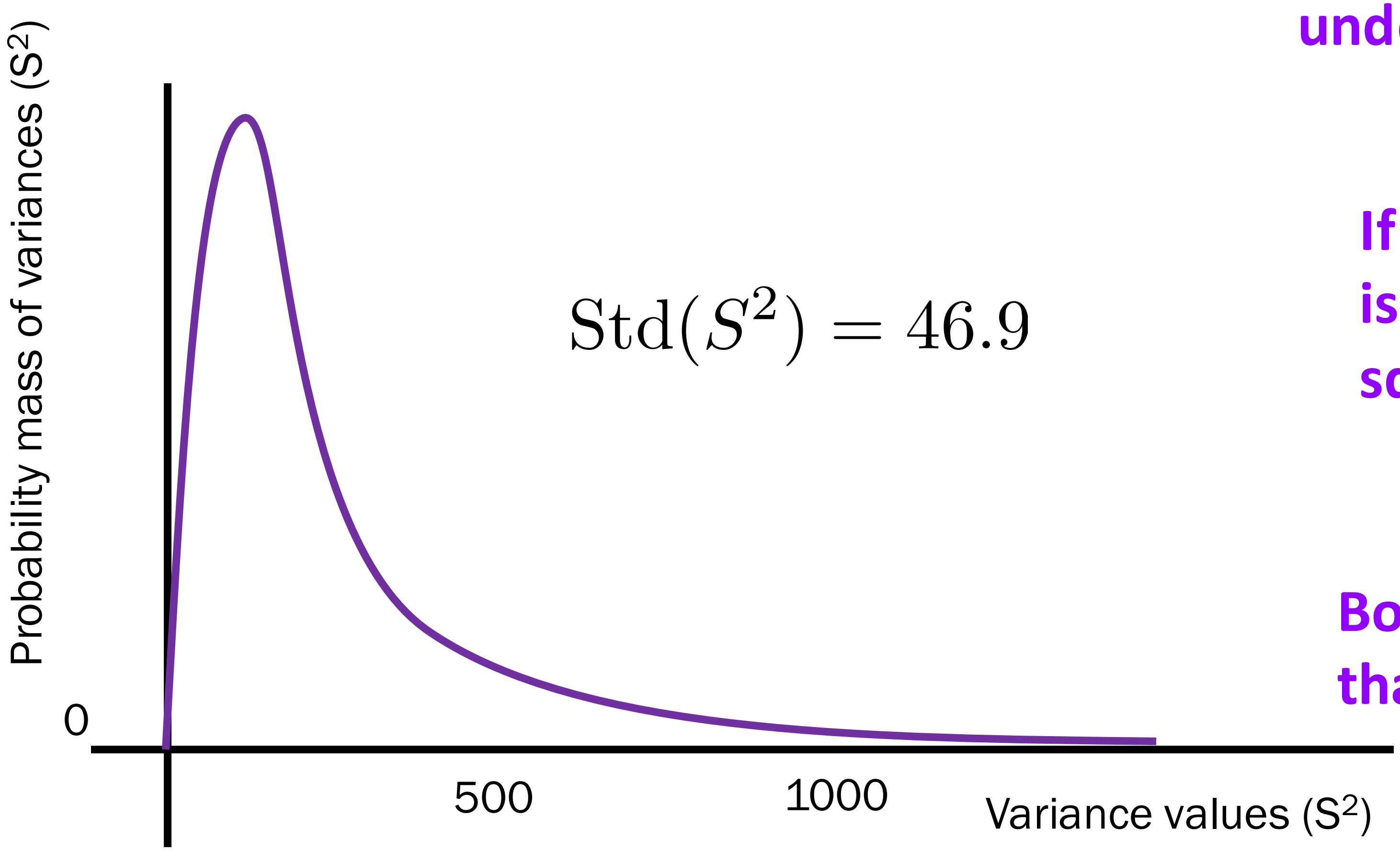



```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the var on the resample
  You now have a distribution of your vars
```

Vars = [472.7, 478.4, 469.2, ..., 476.2]

Bootstrapping of Variance

Sample Vars = [472.7, 478.4, 469.2, ..., 476.2]



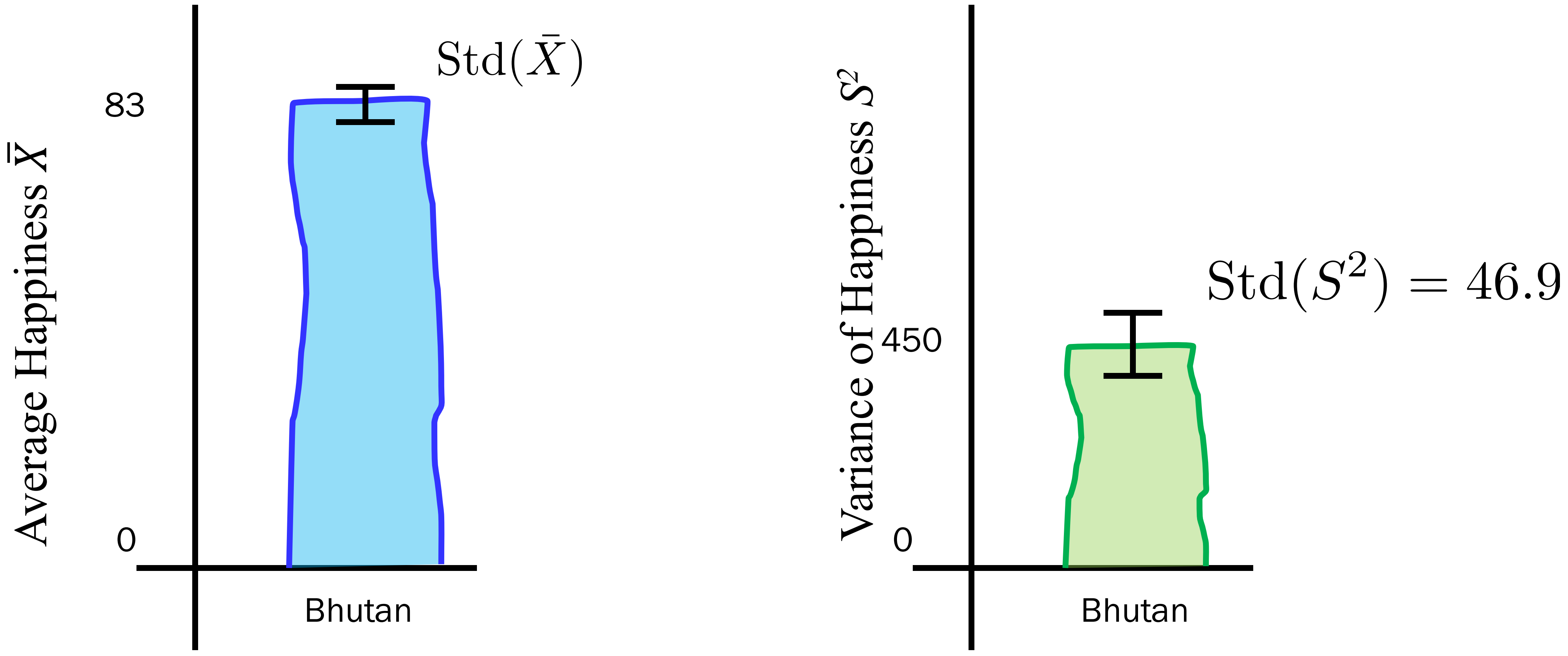
$$\text{Std}(S^2) = 46.9$$

Aside: the distribution of variance depends on the underlying distribution

If the underlying distribution is Gaussian, variance is "chi-squared"

Bootstrapping doesn't need to kno that…

Our Report to Bhutan Government



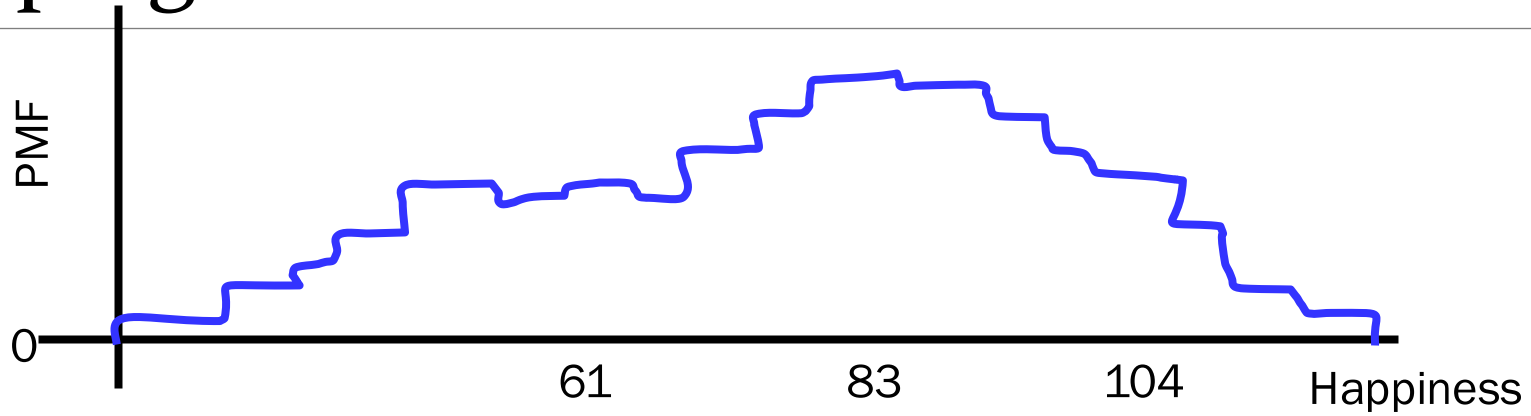
Claim: The average happiness of Bhutan is 83 ± 2

Validation with Sample Mean

Bootstrapping of Means (we could do this with CLT)

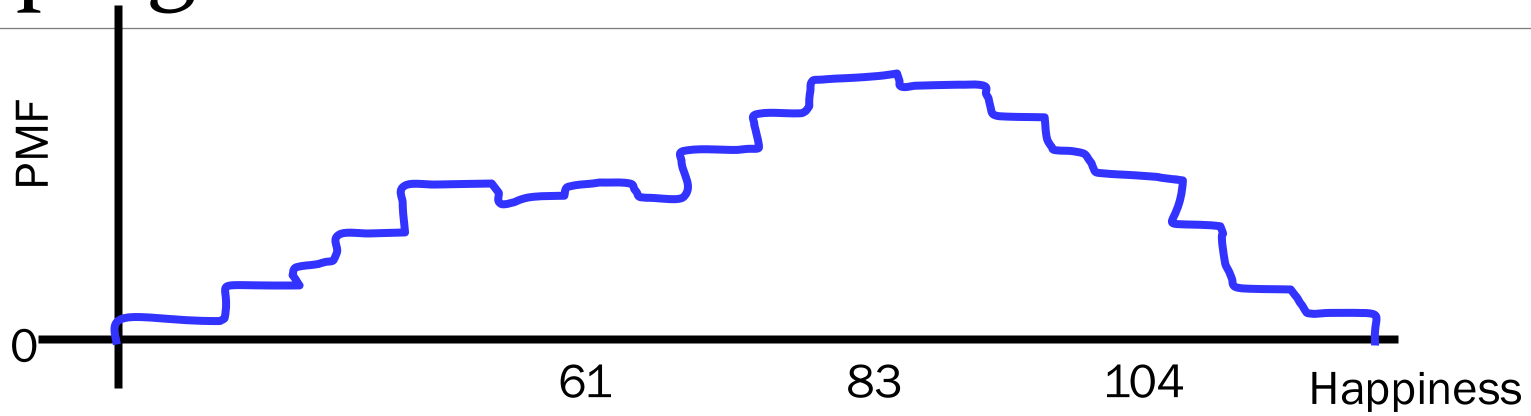
```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the mean on the resample
  You now have a distribution of your means
```

Bootstrapping of Means



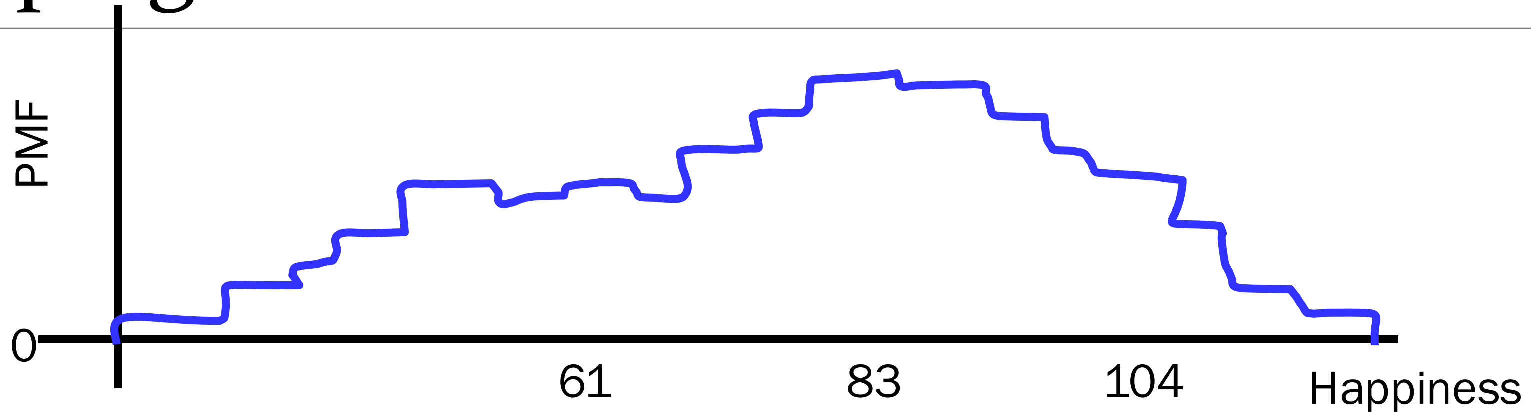

```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the mean on the resample
  You now have a distribution of your means
```

Bootstrapping of Means



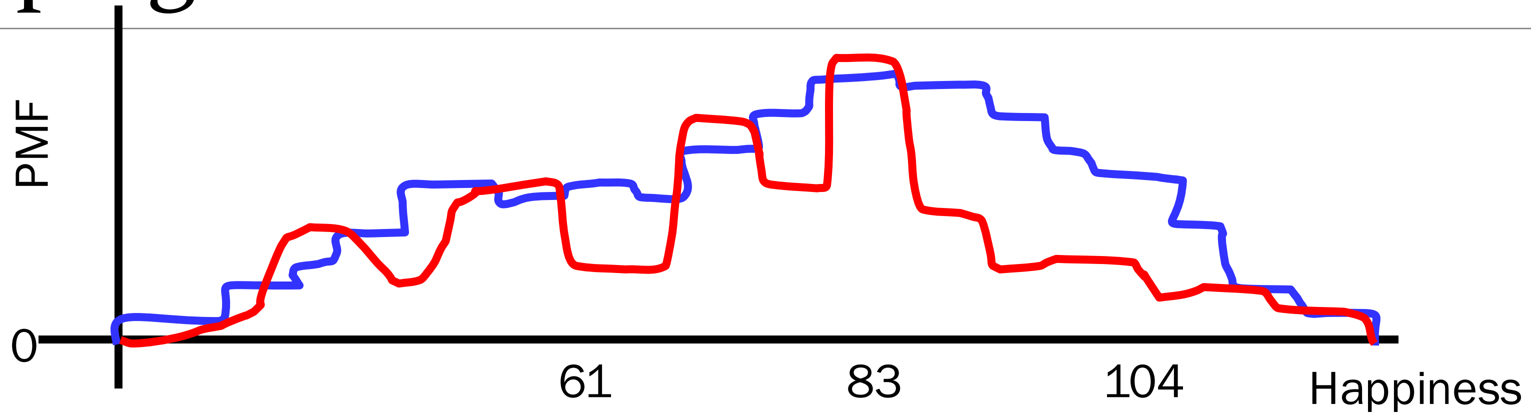

```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the mean on the resample
  You now have a distribution of your means
```

Bootstrapping of Means




```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the mean on the resample
  You now have a distribution of your means
```

Bootstrapping of Means



Bootstrap Algorithm (sample):

Estimate the **PMF** using the sample

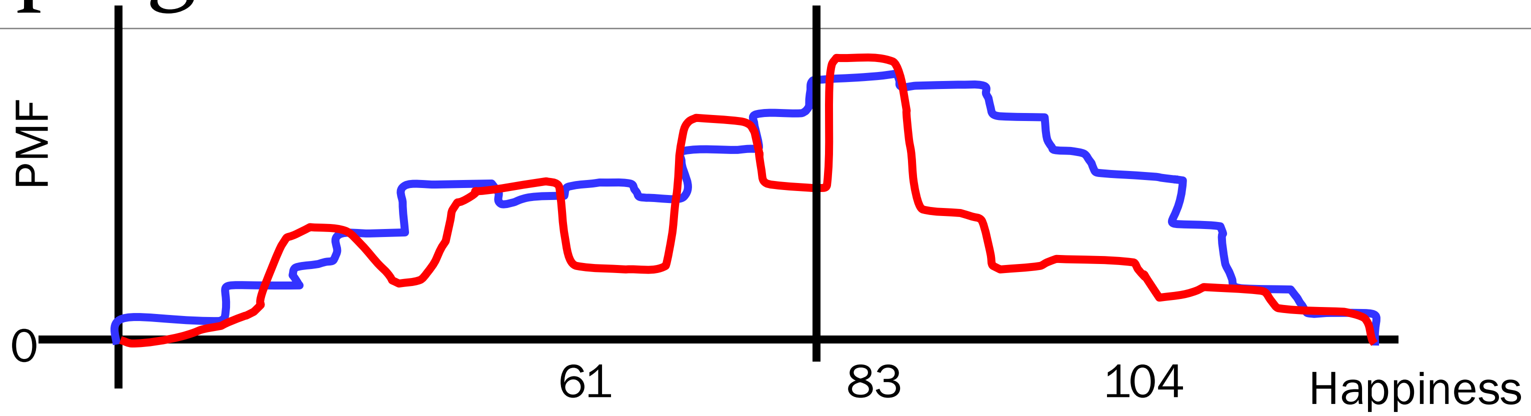
Repeat **10,000** times:

a. Draw **len(sample)** new samples from PMF

b. Recalculate the mean on the resample

You now have a **distribution of your means**

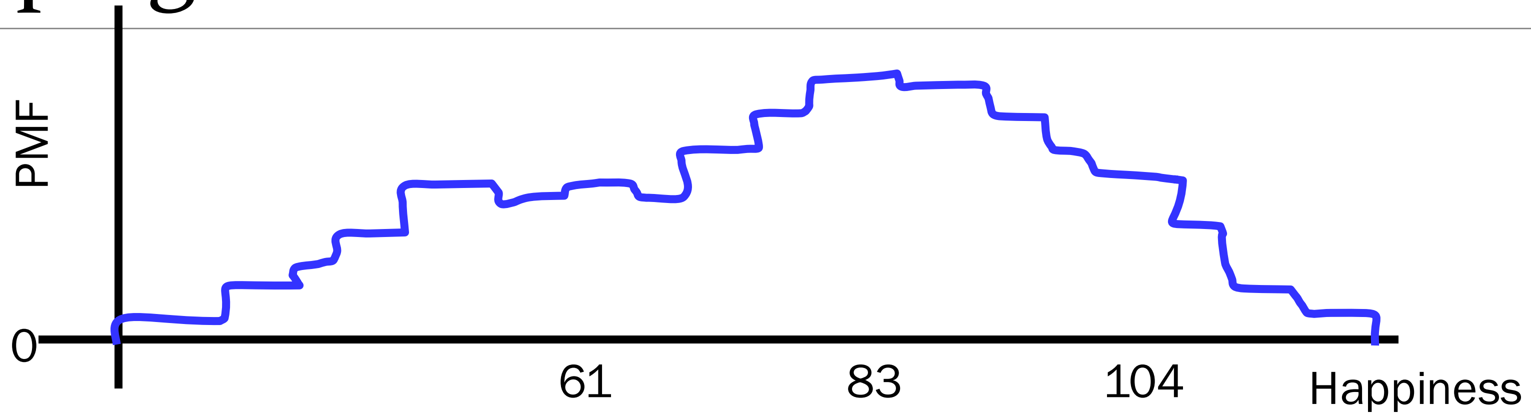
Bootstrapping of Means




```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the mean on the resample
  You now have a distribution of your means
```

Means = [82.7]

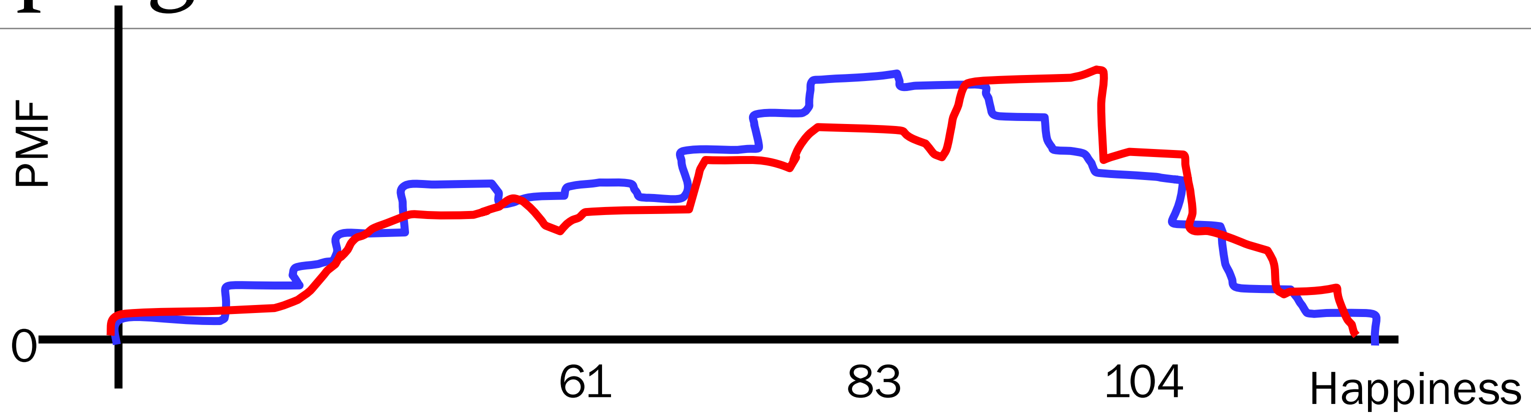
Bootstrapping of Means




```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the mean on the resample
  You now have a distribution of your means
```

Means = [82.7]

Bootstrapping of Means



Bootstrap Algorithm (sample):

Estimate the **PMF** using the sample

Repeat **10,000** times:

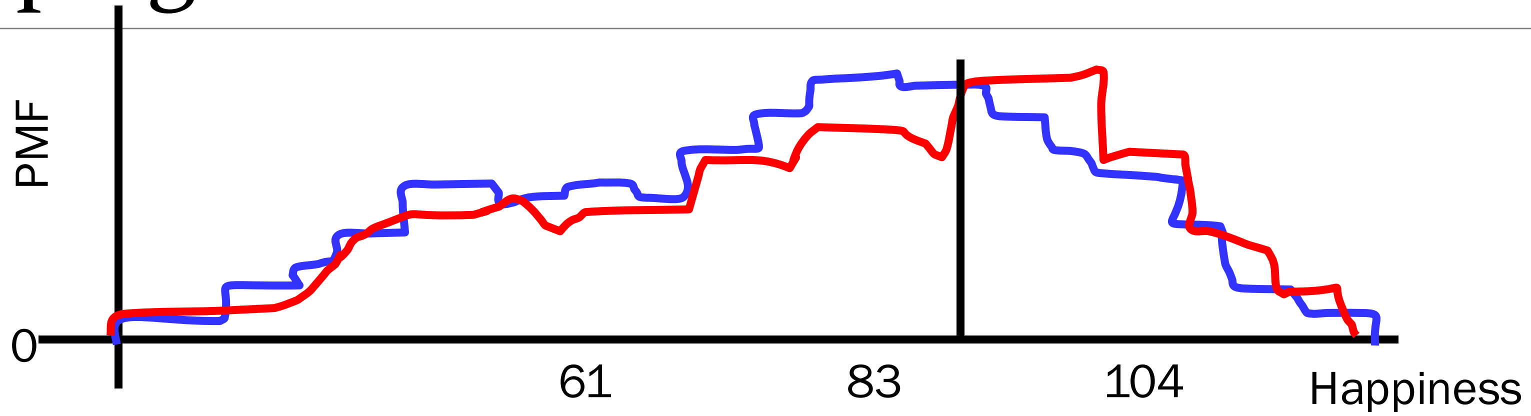
a. Draw **len(sample)** new samples from PMF

b. Recalculate the mean on the resample

You now have a **distribution of your means**

Means = [82.7]

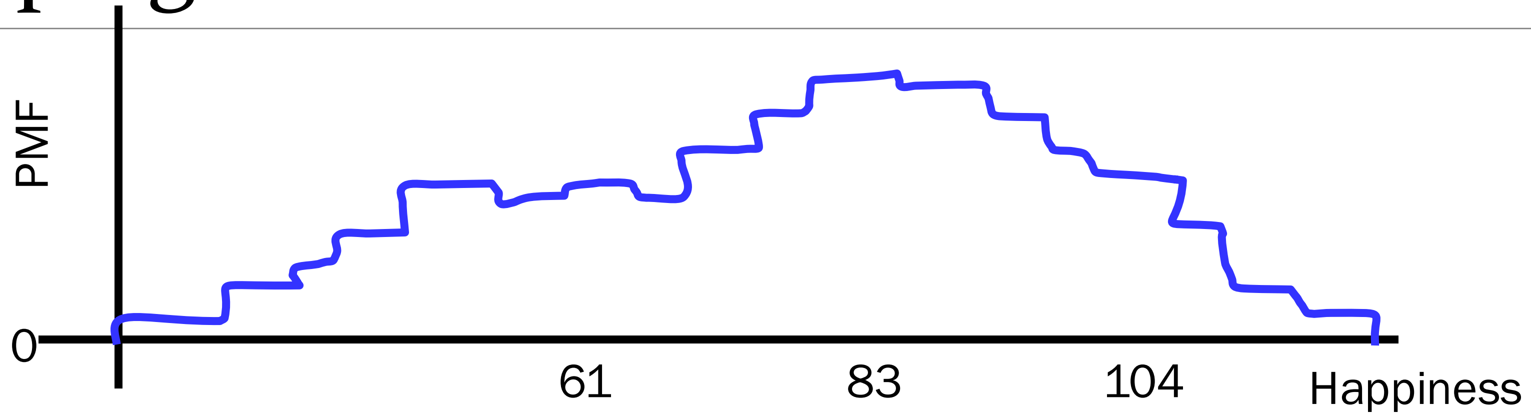
Bootstrapping of Means




```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the mean on the resample
  You now have a distribution of your means
```

Means = [82.7, 83.4]

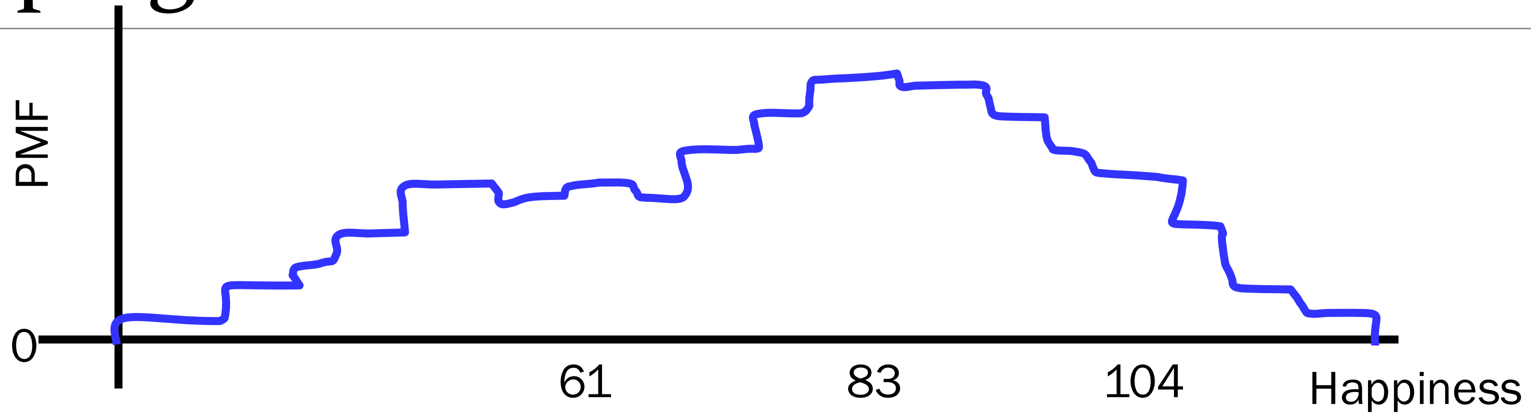
Bootstrapping of Means




```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the mean on the resample
  You now have a distribution of your means
```

Means = [82.7, 83.4]

Bootstrapping of Means

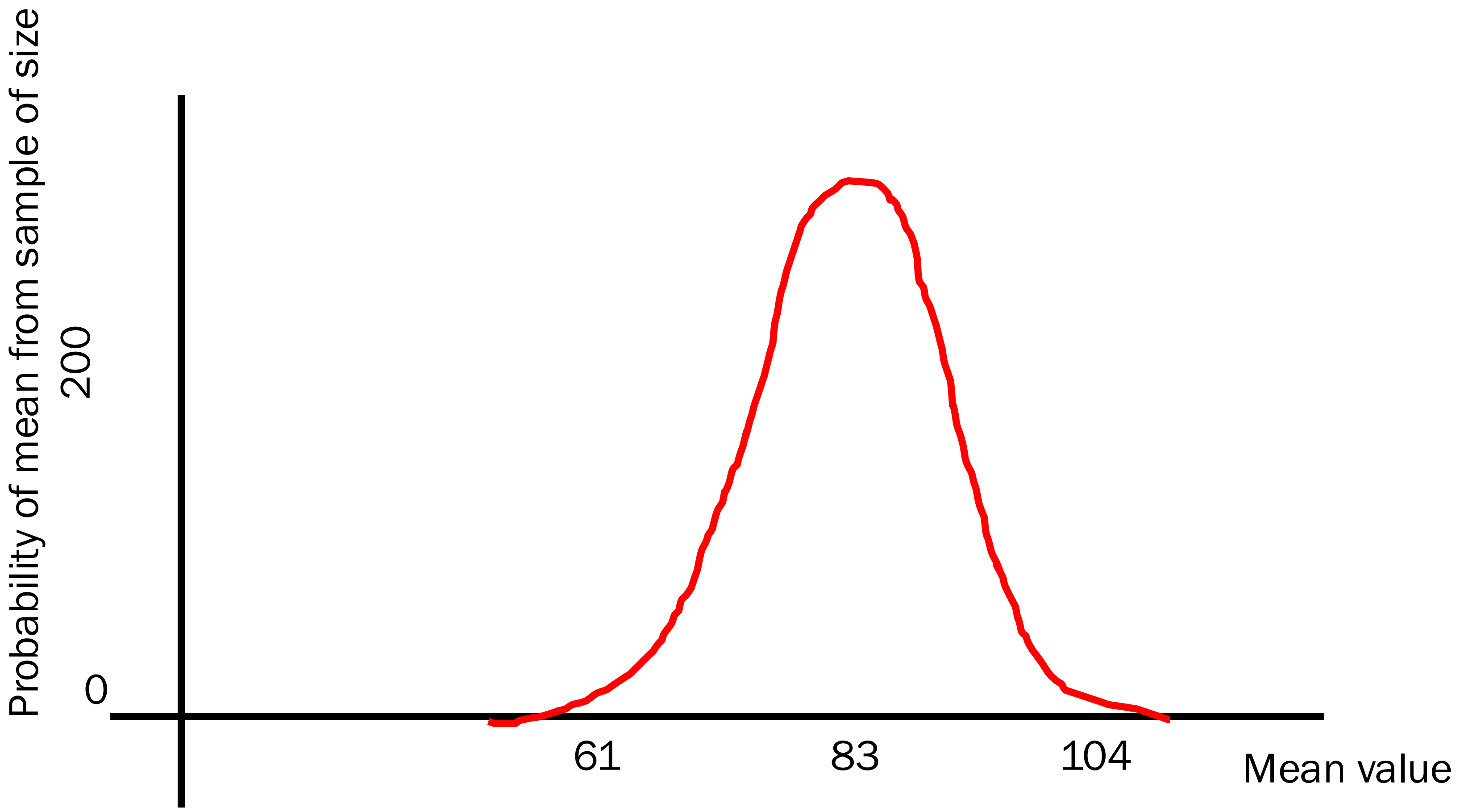



```
Bootstrap Algorithm (sample):
  Estimate the PMF using the sample
  Repeat 10,000 times:
    a. Draw len(sample) new samples from PMF
    b. Recalculate the mean on the resample
  You now have a distribution of your means
```

Means = [82.7, 83.4, 82.9, 91.4, 79.3, 82.1, ..., 81.7]

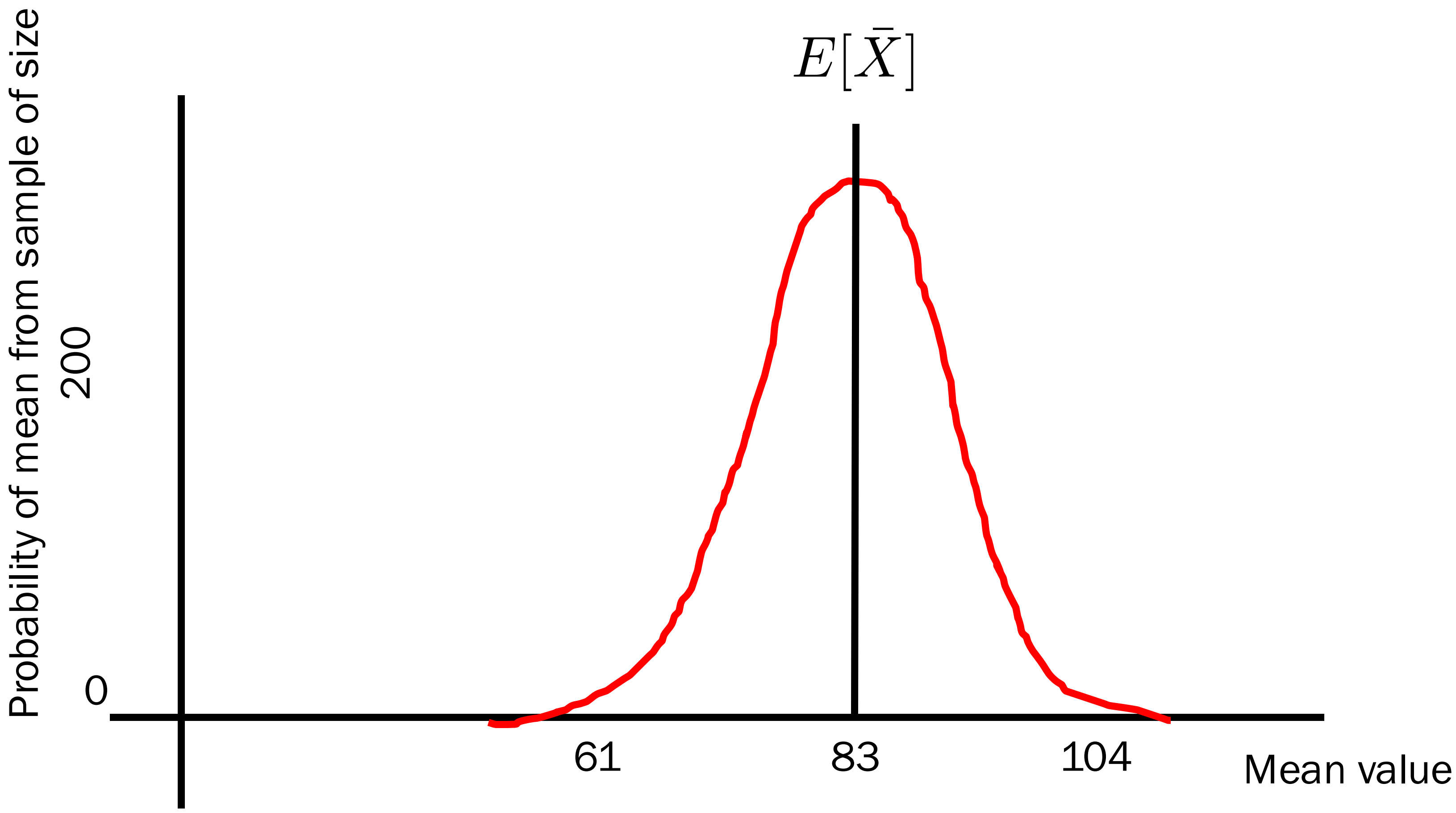
Bootstrapping of Means

Means = [82.7, 83.4, 82.9, 91.4, 79.3, 82.1, ..., 81.7]



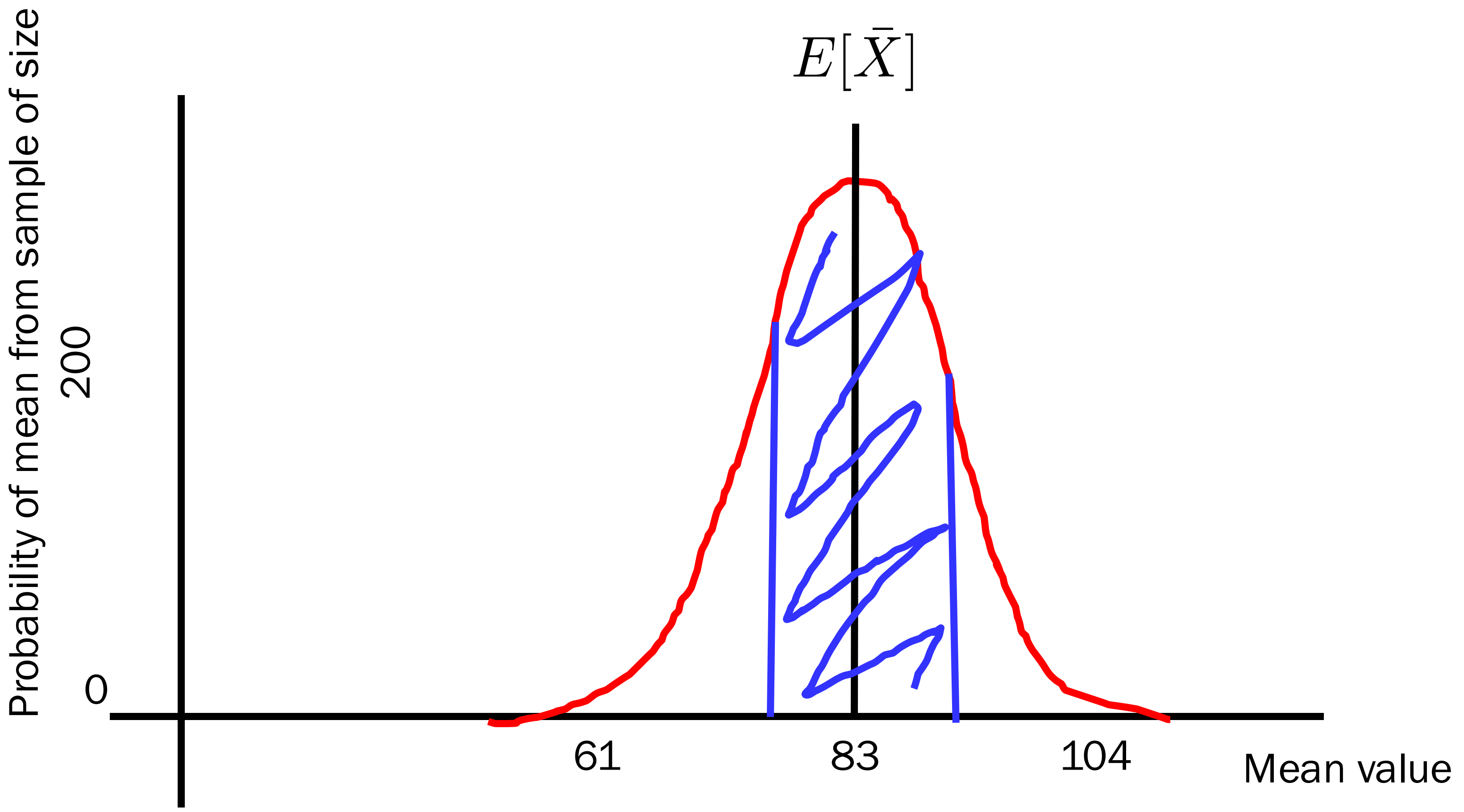
Bootstrapping of Means

Means = [82.7, 83.4, 82.9, 91.4, 79.3, 82.1, ..., 81.7]



Bootstrapping of Means

What is the probability that the mean is in the range 81 to 85?

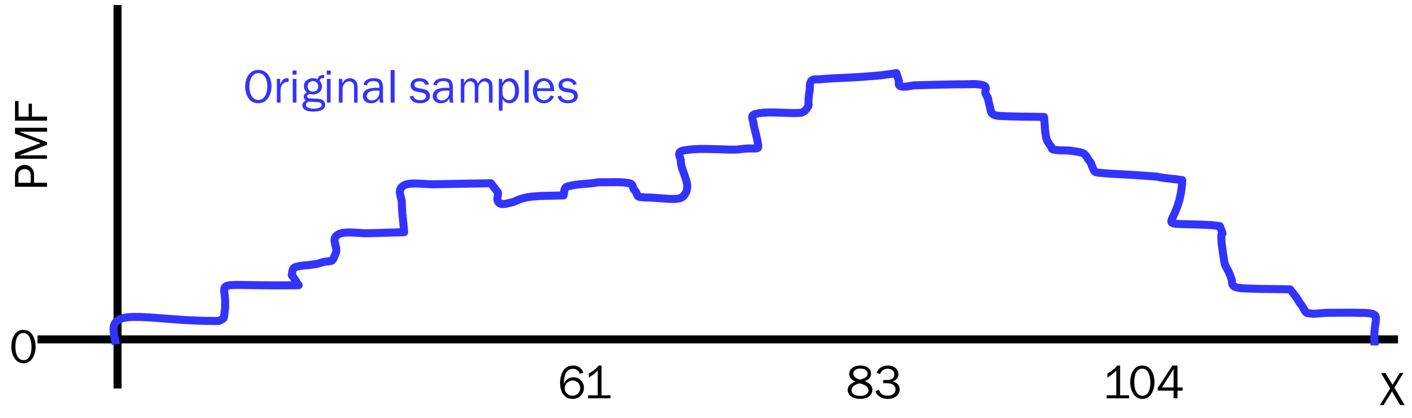


Contrast with Central Limit Theorem

Ok Good!

Bootstrapping in Practice

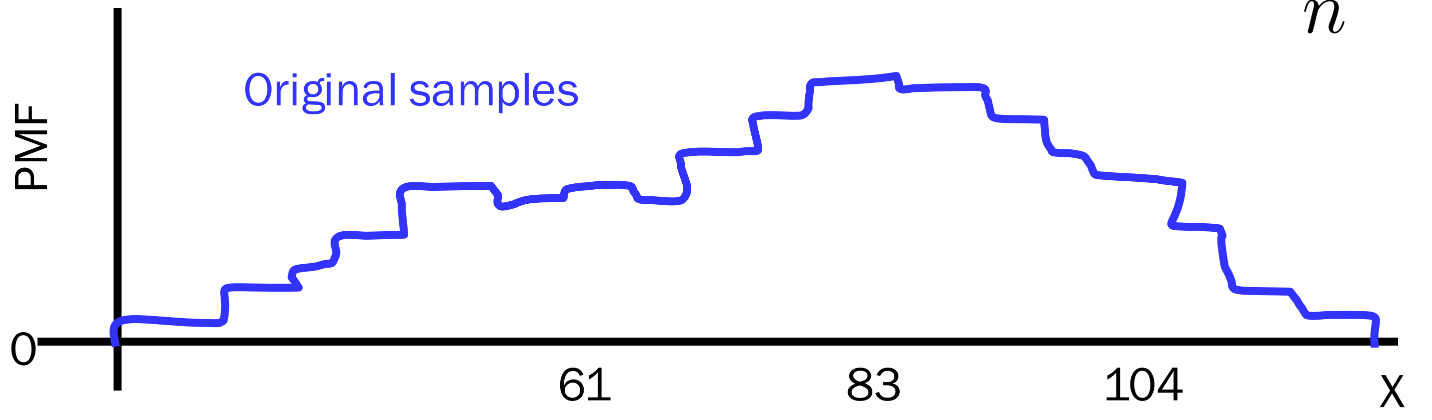
```
def resample(samples, K):
    # Estimate the PMF using the samples
    # Draw K new samples from the PMF
```



Bootstrapping in Practice

```
def resample(samples, K):
    # Estimate the PMF using the samples
    # Draw K new samples from the PMF
    return np.random.choice(samples, K,
                     replace = True)
```

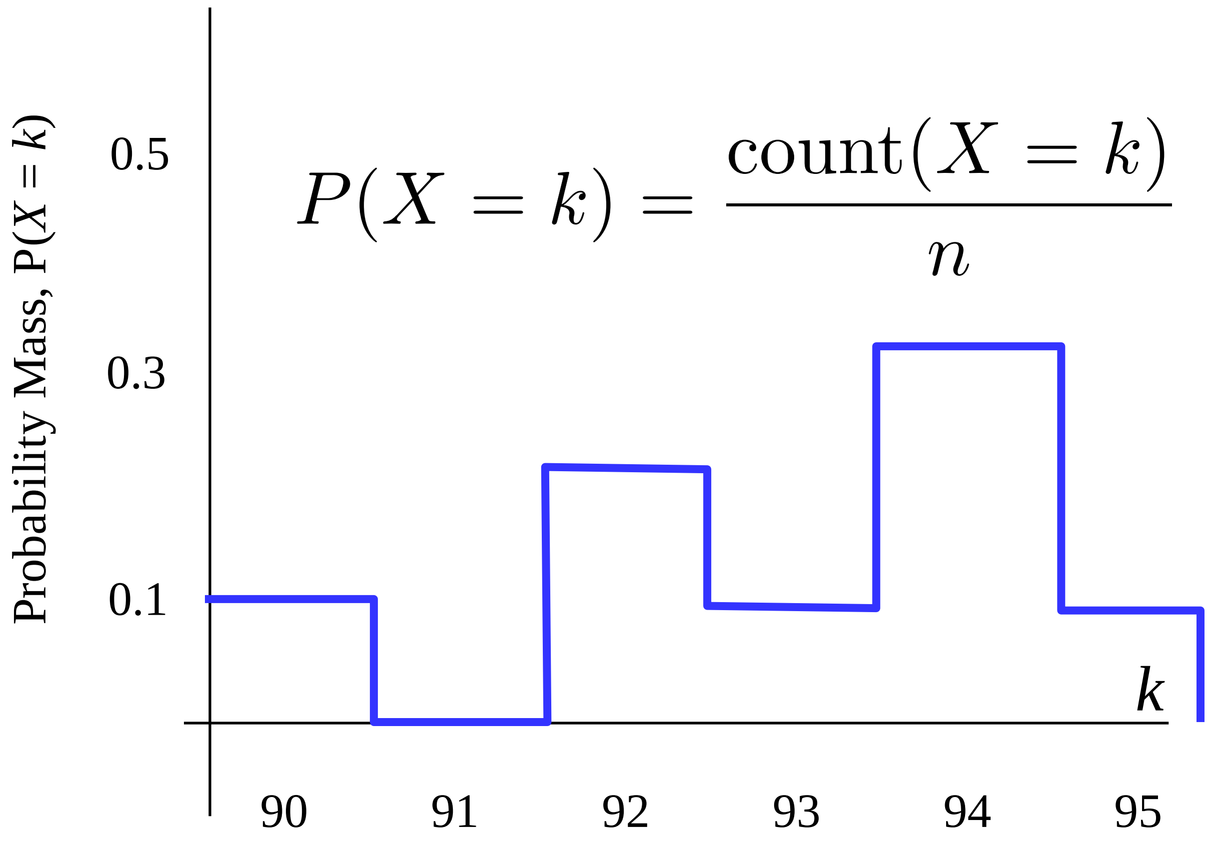
$$P(X = k) = \frac{\text{count}(X = k)}{n}$$



np.random.choice(samples, K, replace = True)

Original Samples: [90, 92, 92, 93, 94, 94, 94, 95]

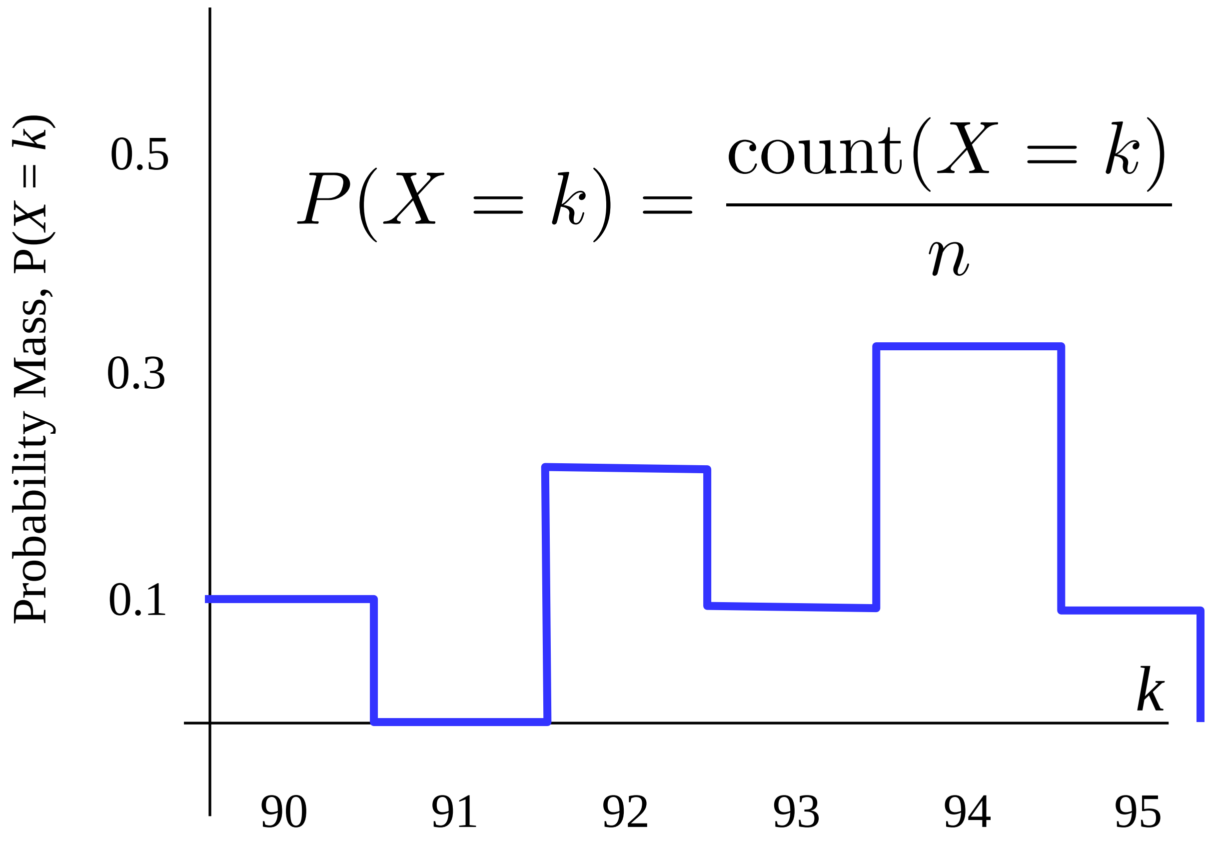
Resample:



`np.random.choice(samples, K, replace = True)`

Original Samples: [90, 92, 92, 93, **94**, 94, 94, 95]

Resample:

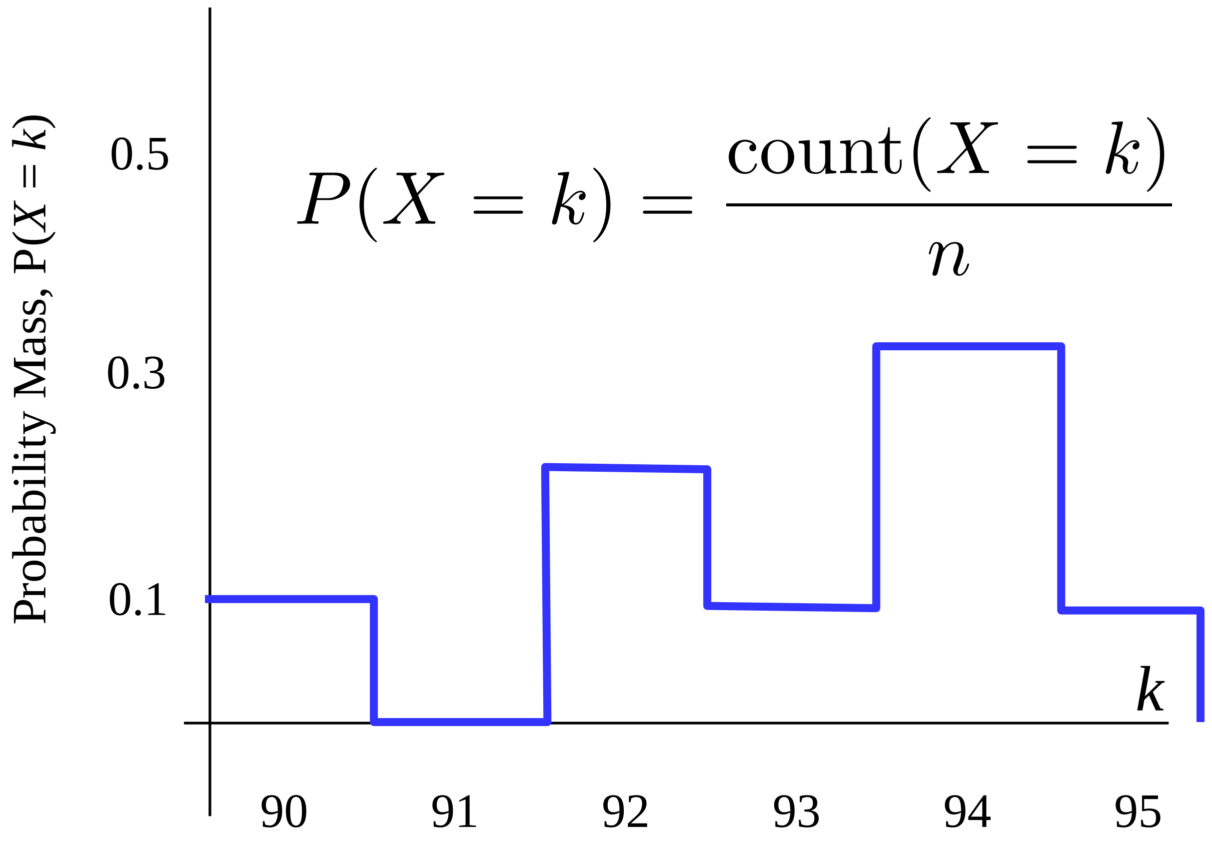


`np.random.choice(samples, K, replace = True)`

Original Samples: [90, 92, 92, 93, **94**, 94, 94, 95]

Resample:

[**94**]

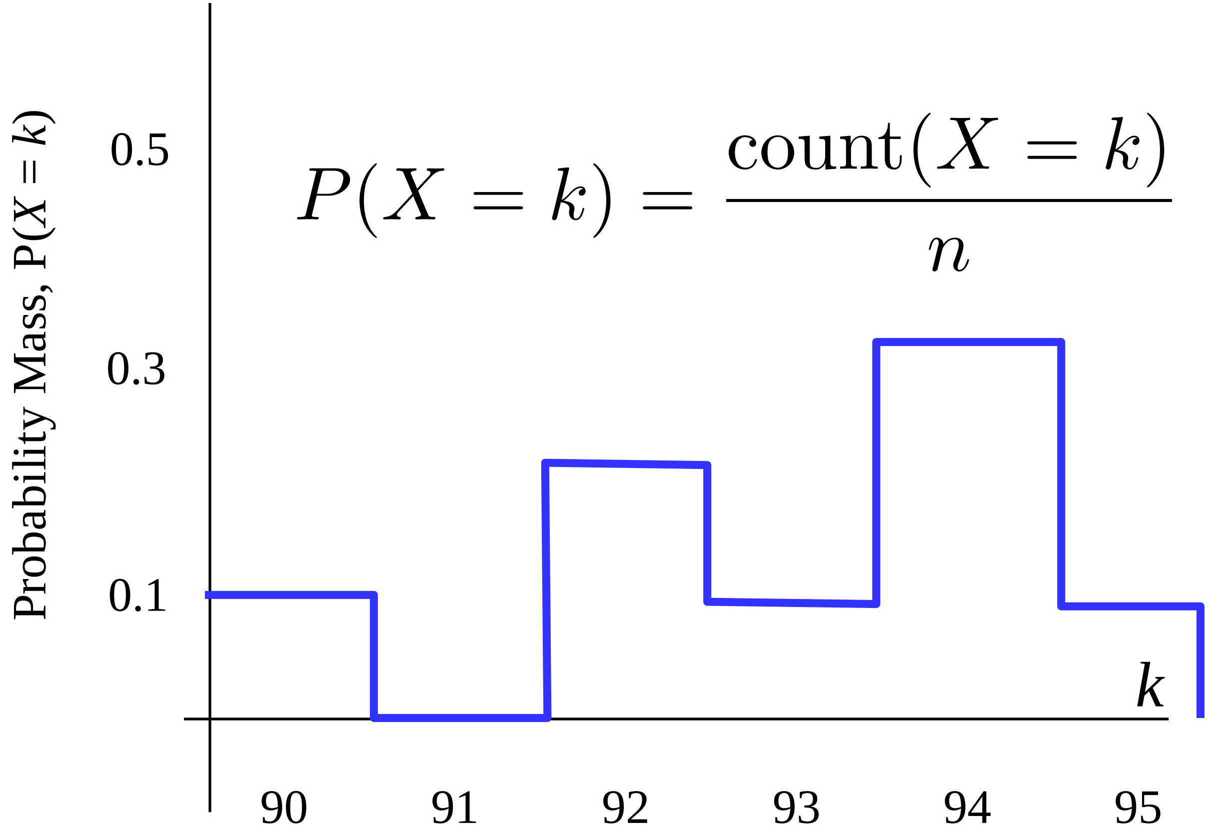


`np.random.choice(samples, K, replace = True)`

Original Samples: [90, 92, 92, 93, 94, 94, 94, 95]

Resample:

[94]



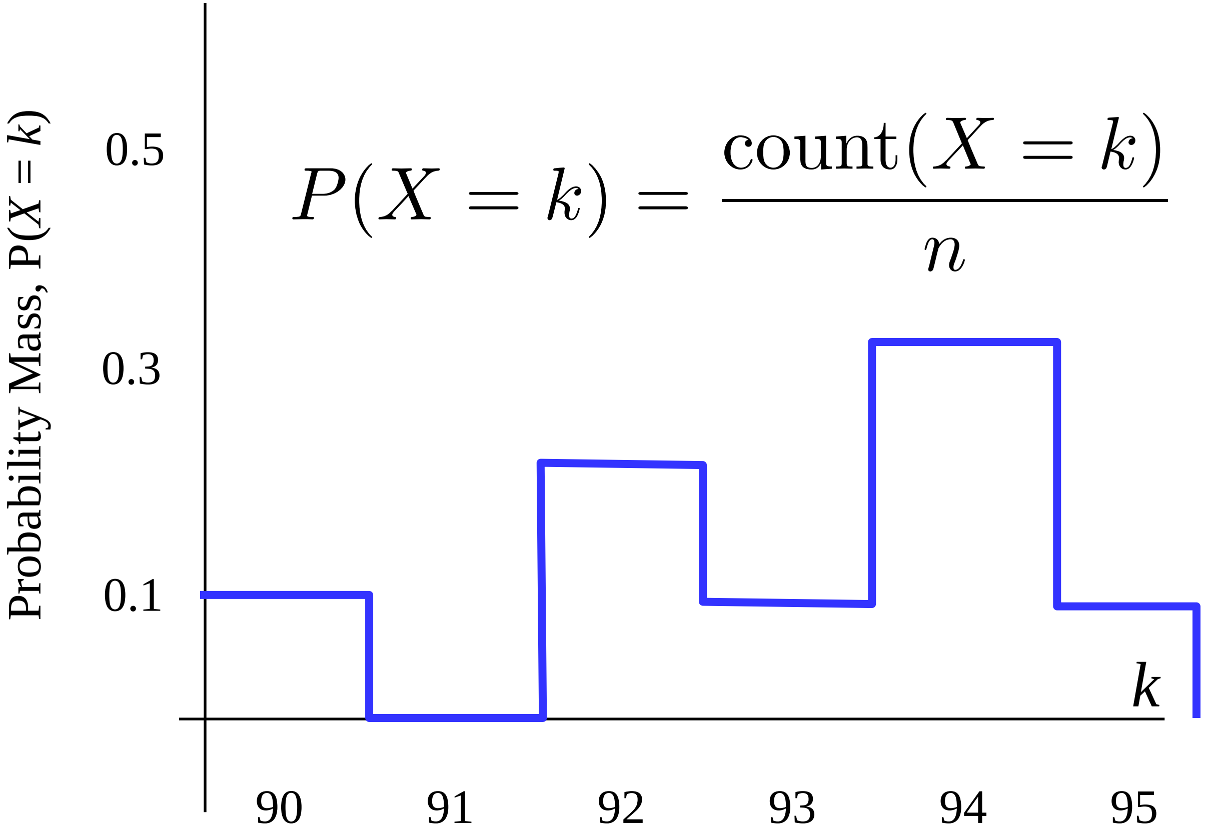
`np.random.choice(samples, K, replace = True)`

Original Samples: [**90**, 92, 92, 93, 94, 94, 94, 95]

Resample:

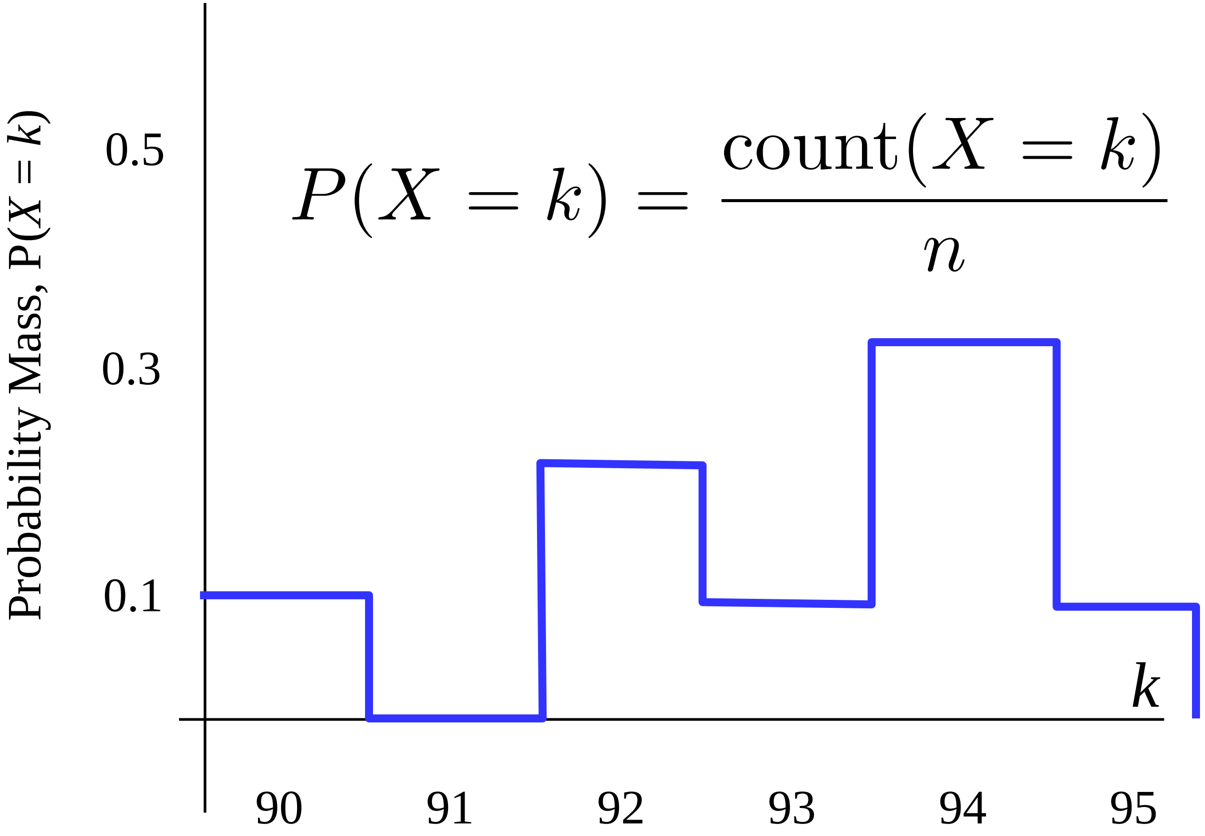
[94]

$$P(X = k) = \frac{\text{count}(X = k)}{n}$$



np.random.choice(samples, K, replace = True)

Original Samples: [**90**, 92, 92, 93, 94, 94, 94, 95]



$$P(X = k) = \frac{\text{count}(X = k)}{n}$$

Resample:

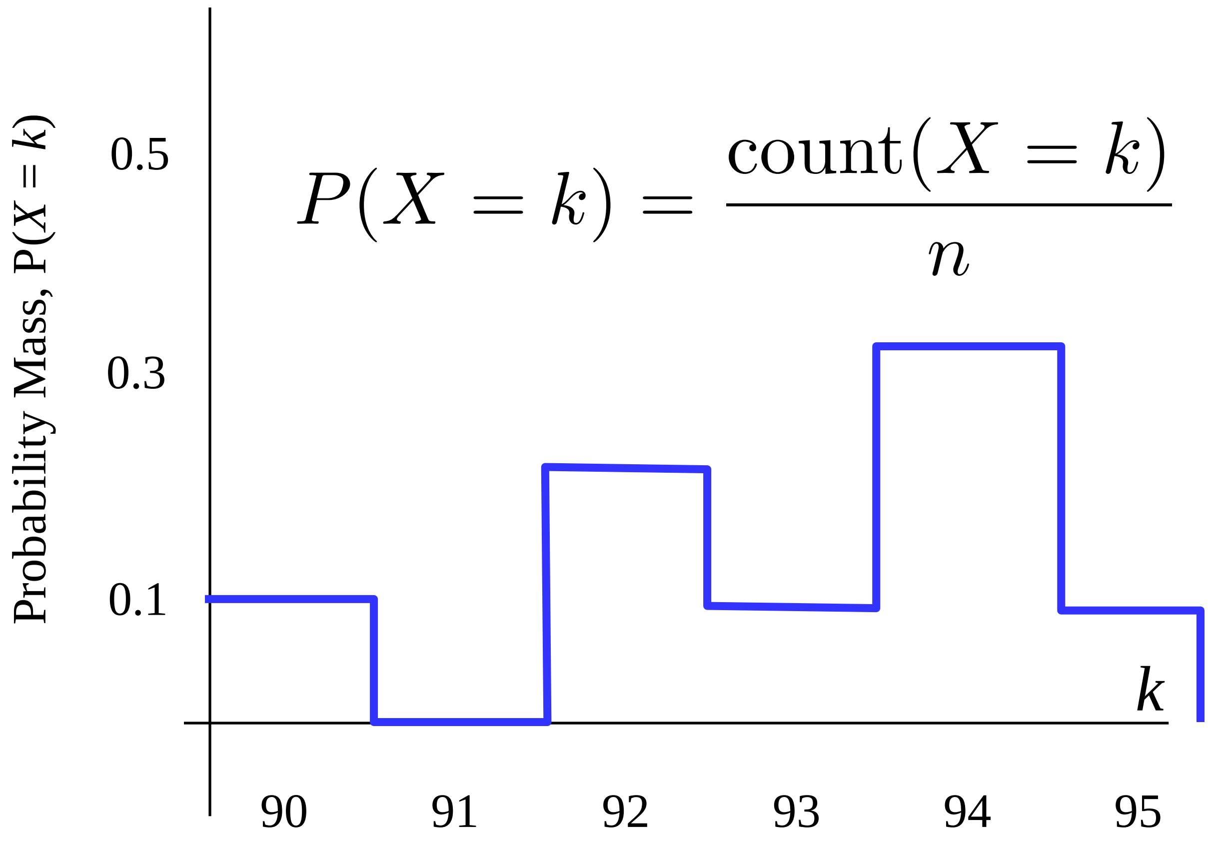
[94, **90**]

`np.random.choice(samples, K, replace = True)`

Original Samples: [90, 92, 92, 93, 94, 94, 94, 95]

Resample:

[94, 90]



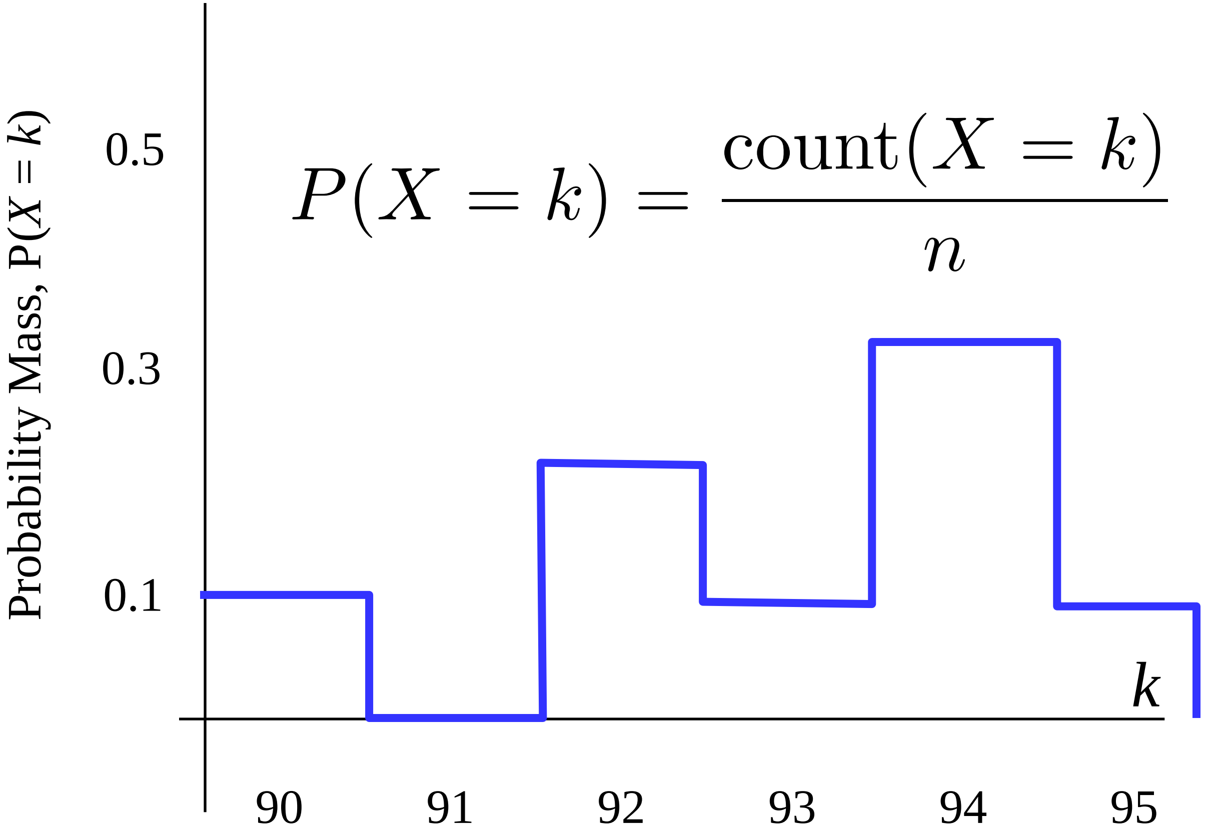
`np.random.choice(samples, K, replace = True)`

Original Samples: [**90**, 92, 92, 93, 94, 94, 94, 95]

Resample:

[94, 90]

$$P(X = k) = \frac{\text{count}(X = k)}{n}$$



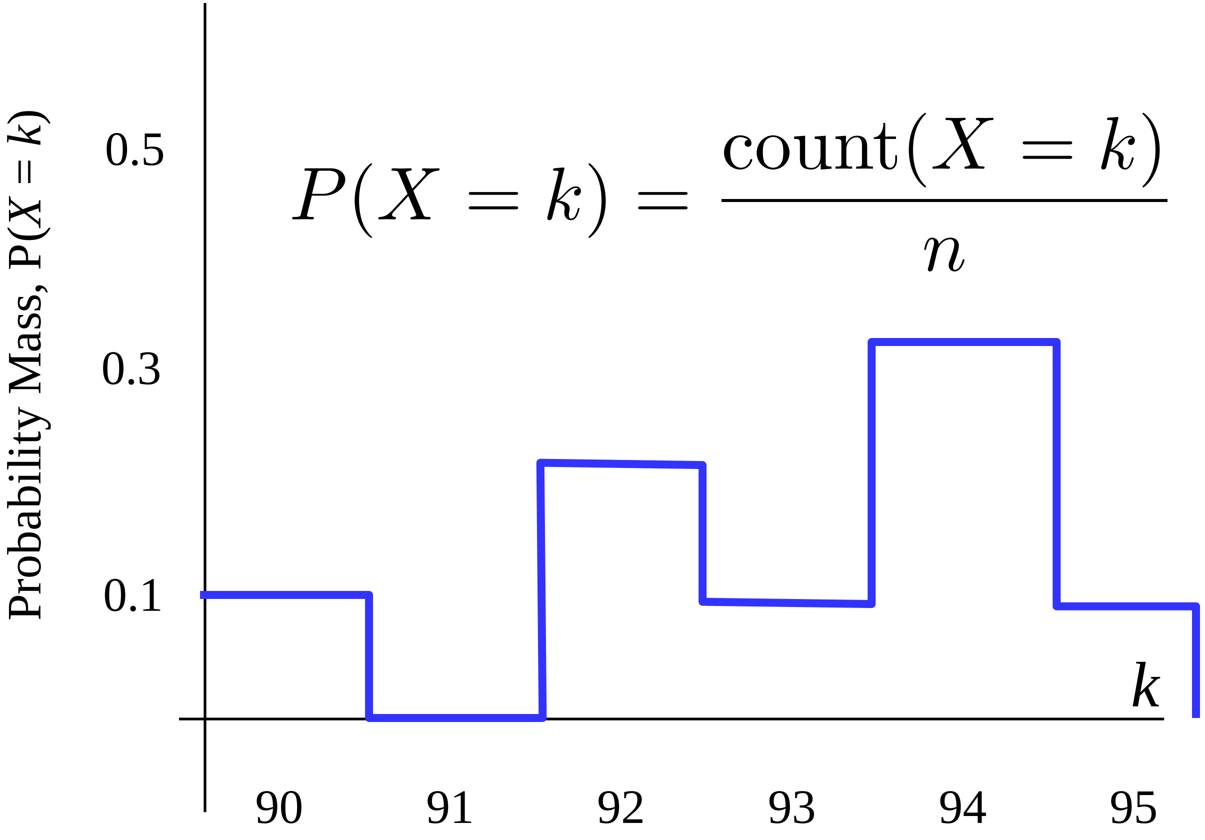
np.random.choice(samples, K, replace = True)

Original Samples: [**90**, 92, 92, 93, 94, 94, 94, 95]

Resample:

[94, 90, **90**]

$$P(X = k) = \frac{\text{count}(X = k)}{n}$$

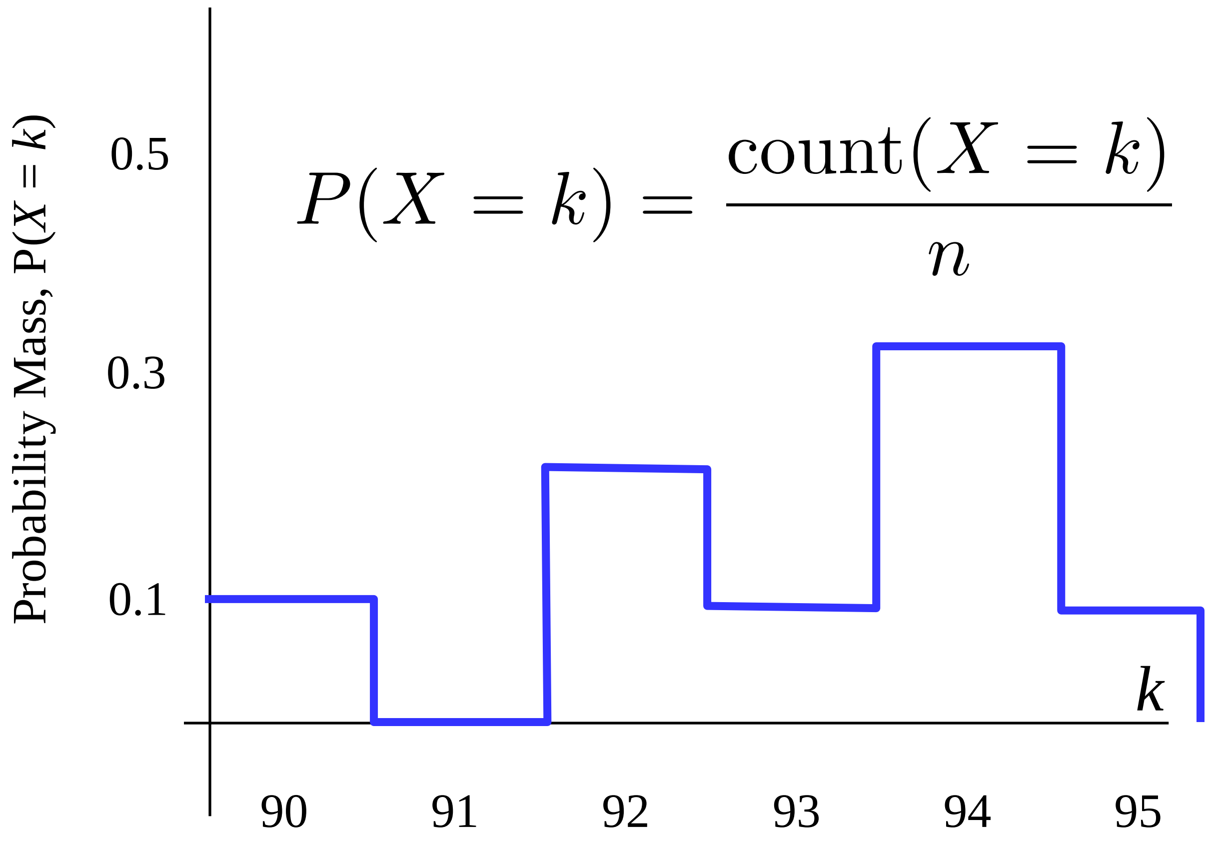


np.random.choice(samples, K, replace = True)

Original Samples: [90, 92, 92, 93, 94, 94, 94, 95]

Resample:

[94, 90, 90]



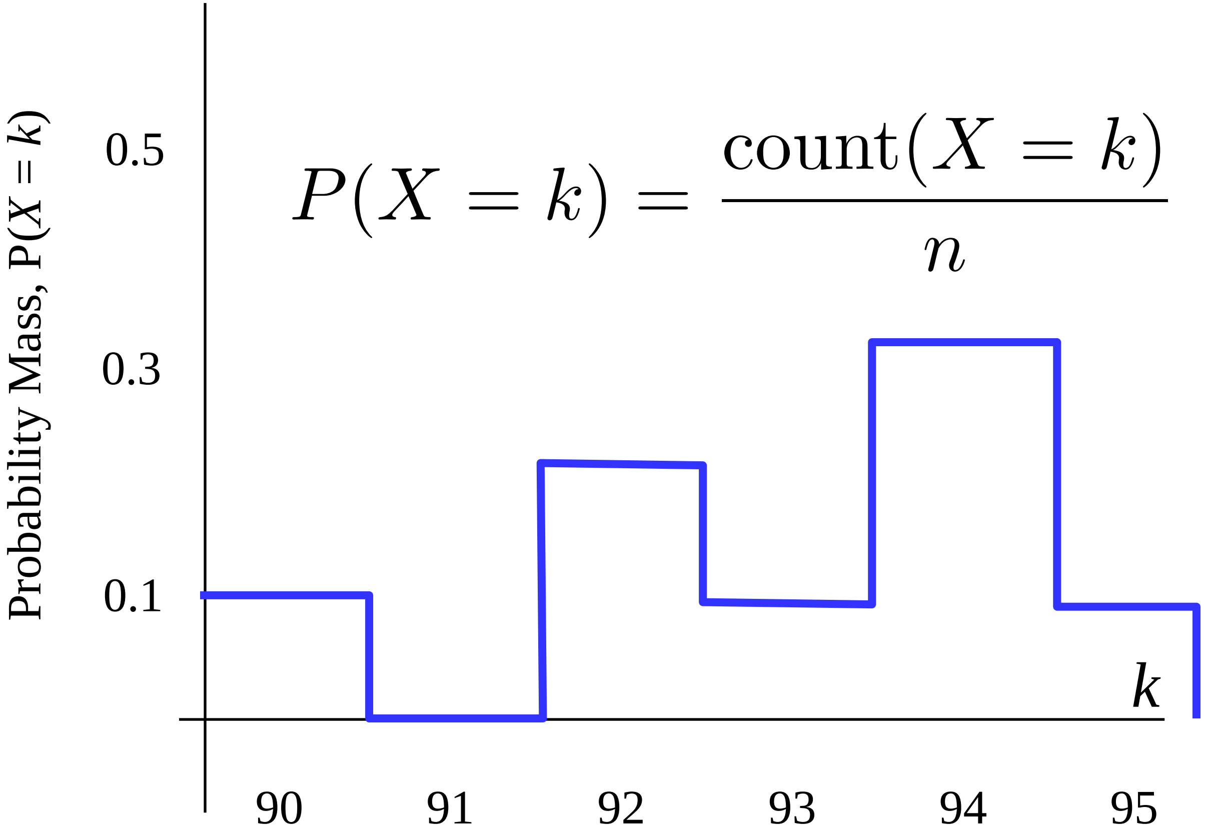
Now with replace = False

`np.random.choice(samples, K, replace = False)`

Original Samples: [90, 92, 92, 93, 94, 94, 94, 95]

Resample:

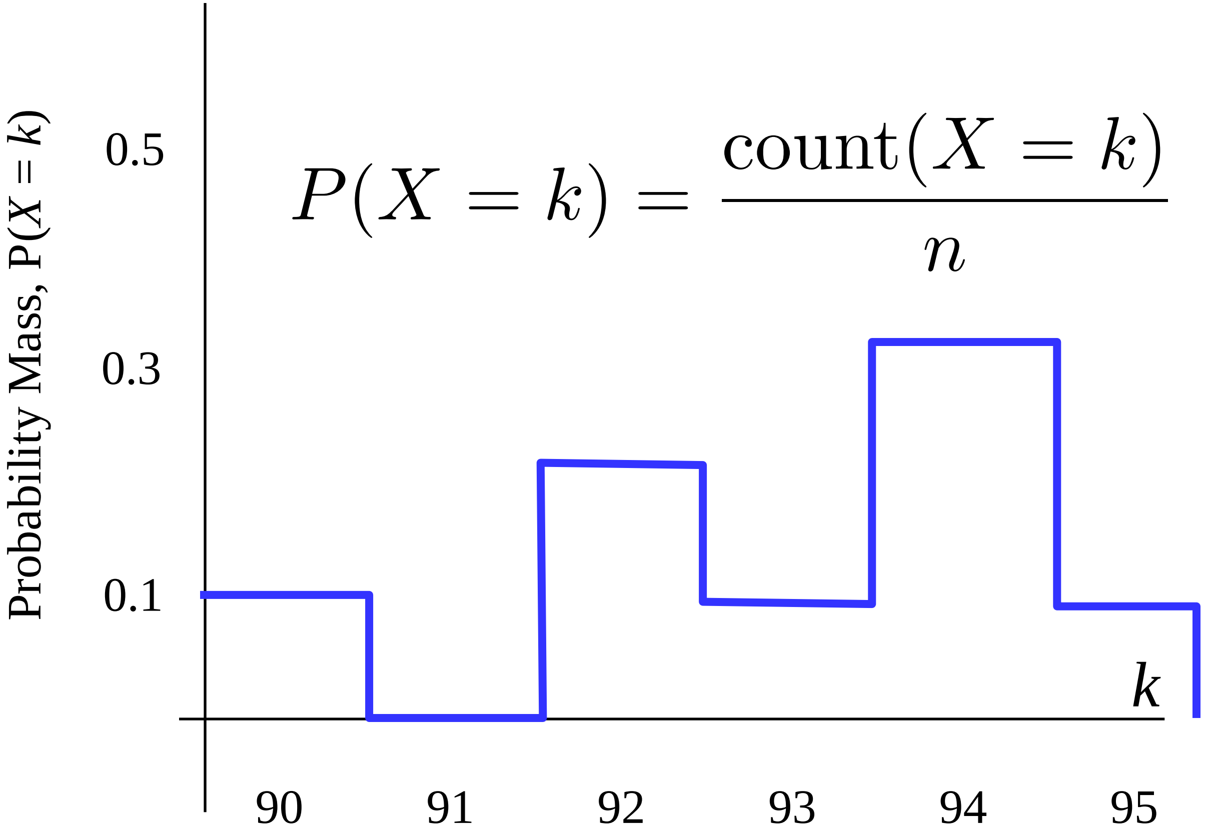
$$P(X = k) = \frac{\text{count}(X = k)}{n}$$



np.random.choice(samples, K, replace = False)

Original Samples: [90, 92, 92, 93, **94**, 94, 94, 95]

Resample:



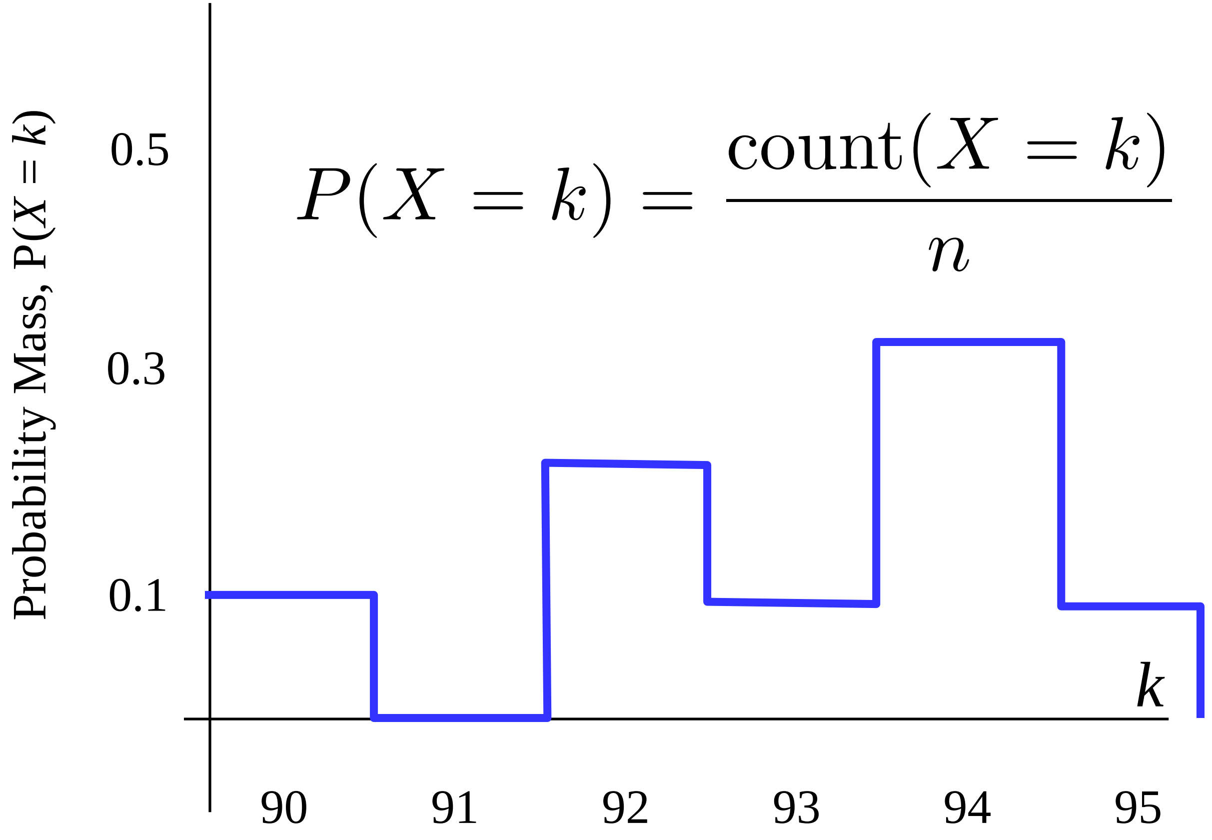
$$P(X = k) = \frac{\text{count}(X = k)}{n}$$

`np.random.choice(samples, K, replace = False)`

Original Samples: [90, 92, 92, 93, **94**, 94, 94, 95]

Resample:

[**94**]

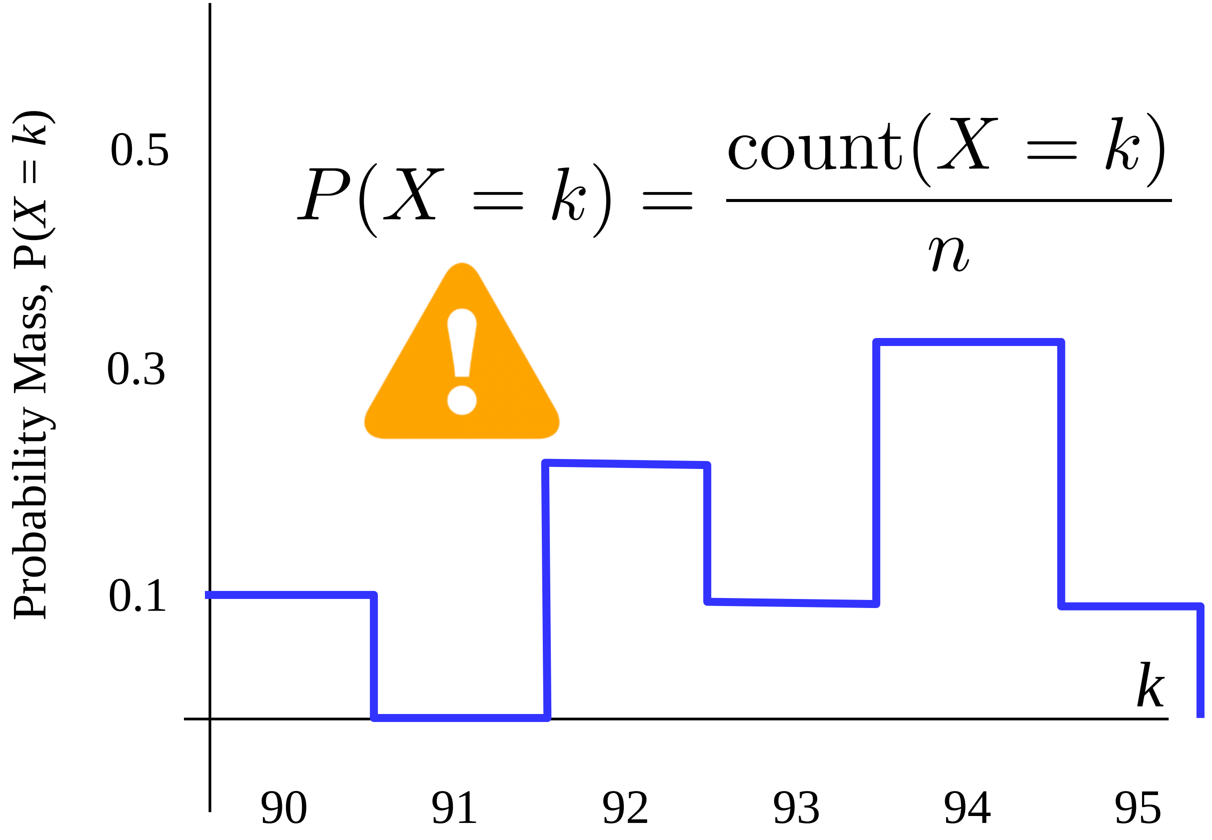


np.random.choice(samples, K, replace = False)

Original Samples: [90, 92, 92, 93, 94, 94, 95]

Removed 94

$$P(X = k) = \frac{\text{count}(X = k)}{n}$$

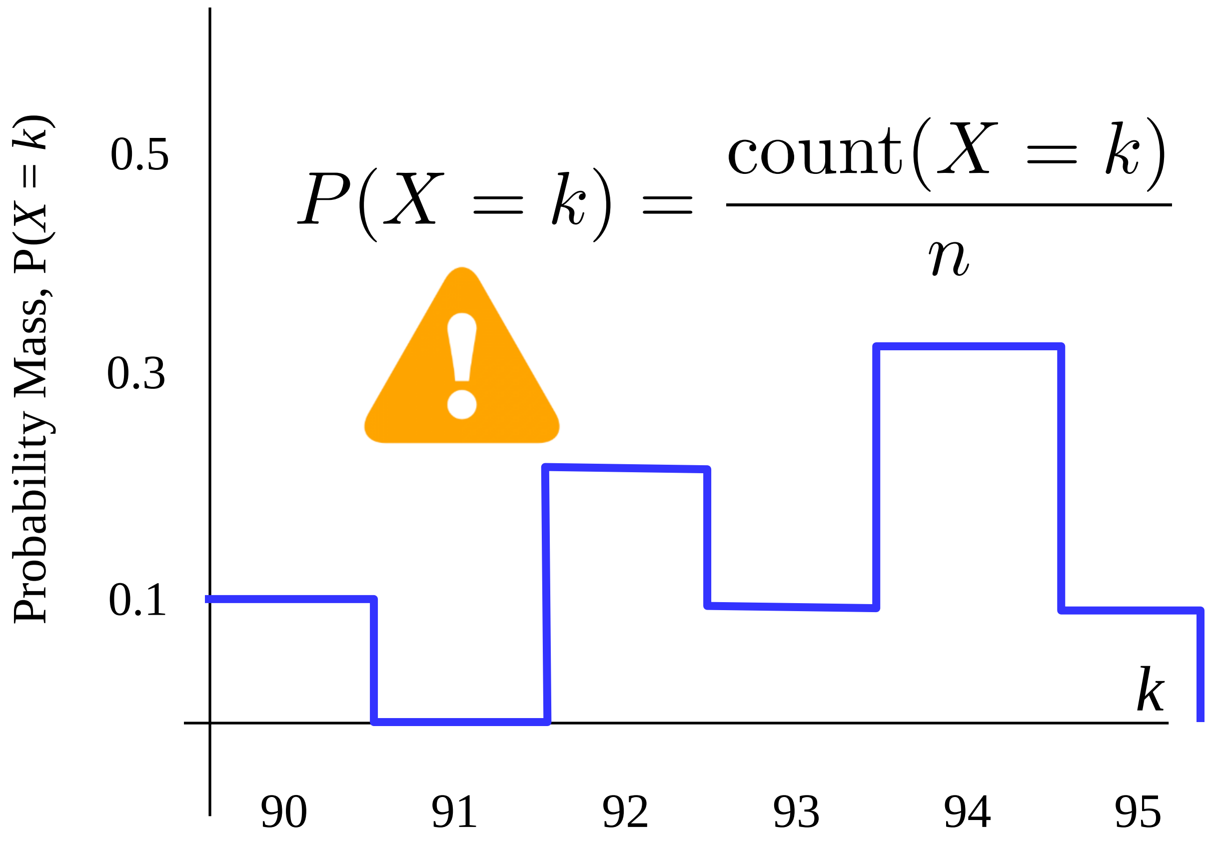


Resample:

[94]

np.random.choice(samples, K, replace = False)

Original Samples: [90, 92, 92, 93, 94, 94, 95]



Resample:

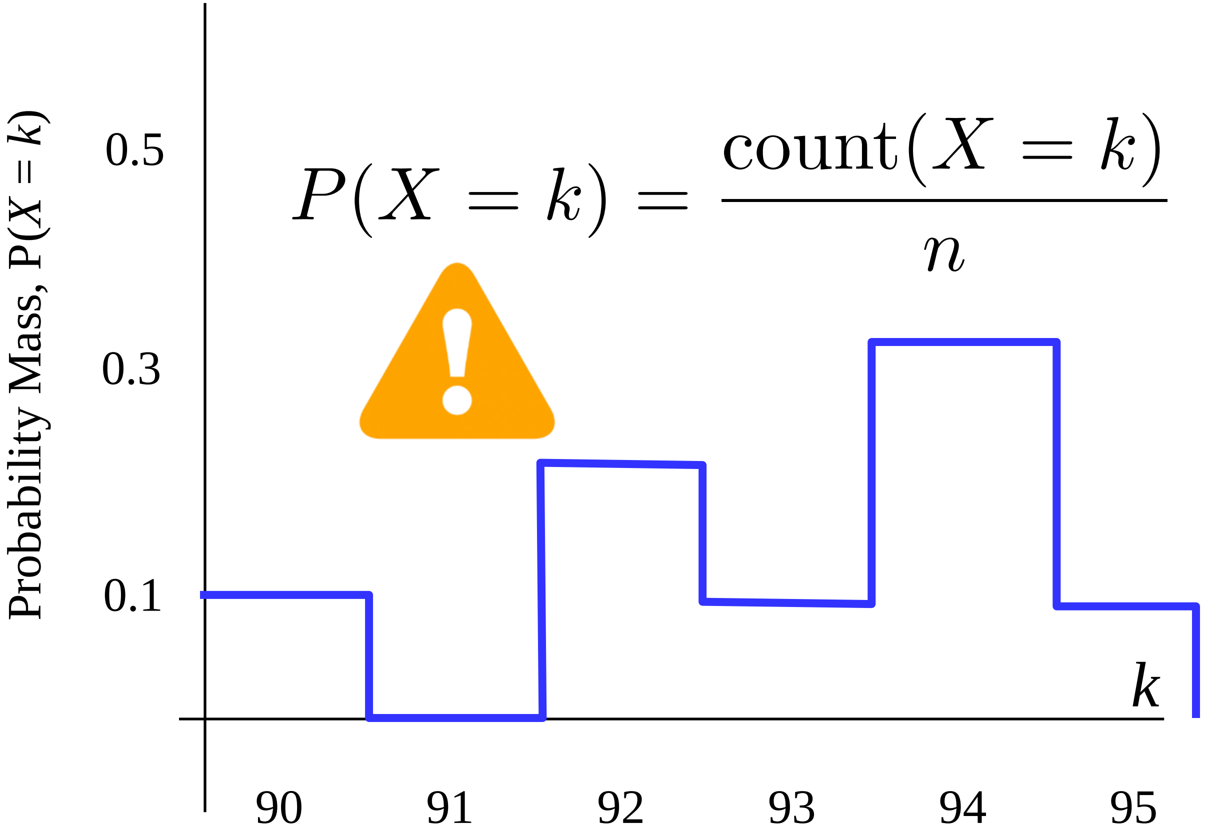
[94]

np.random.choice(samples, K, replace = False)

Original Samples: [90, 92, 92, 93, 94, 94, 95]

Resample:

[94]

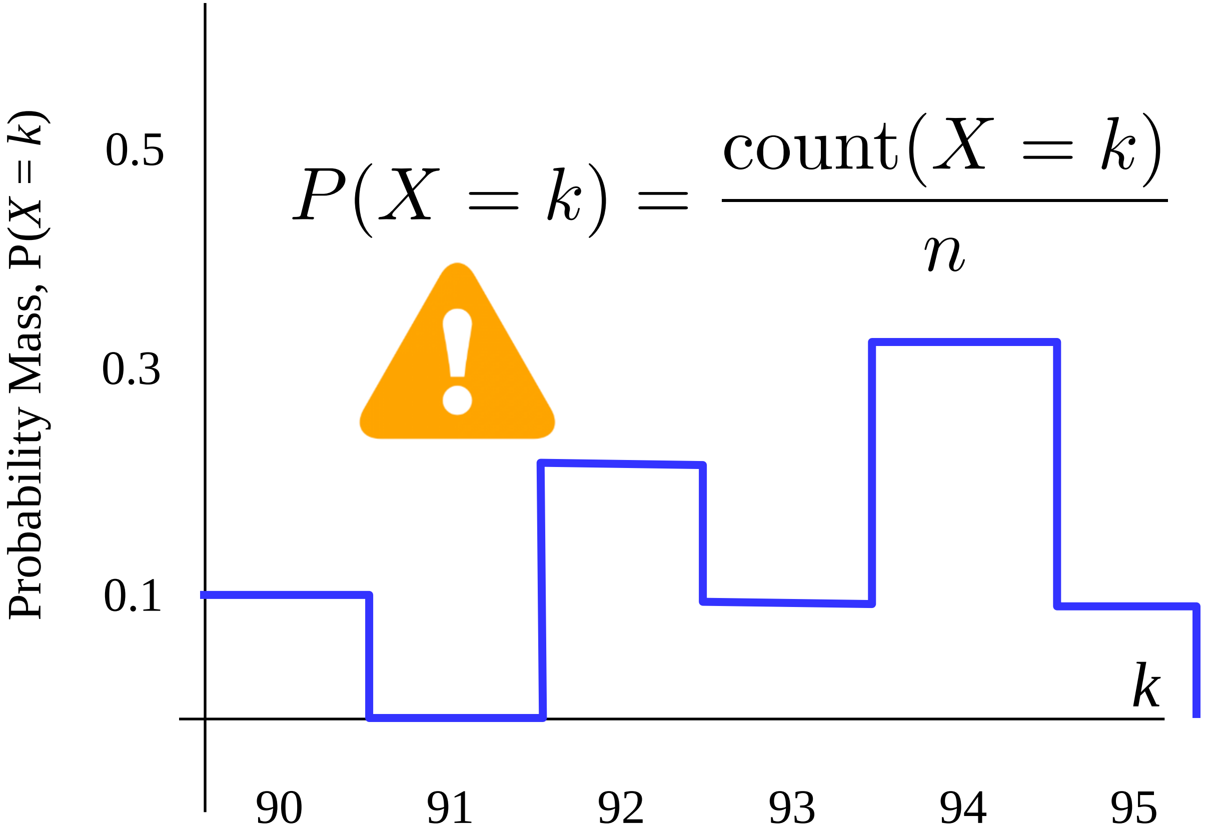


`np.random.choice(samples, K, replace = False)`

Original Samples: [**90**, 92, 92, 93, 94, 94, 95]

Resample:

[94, **90**]



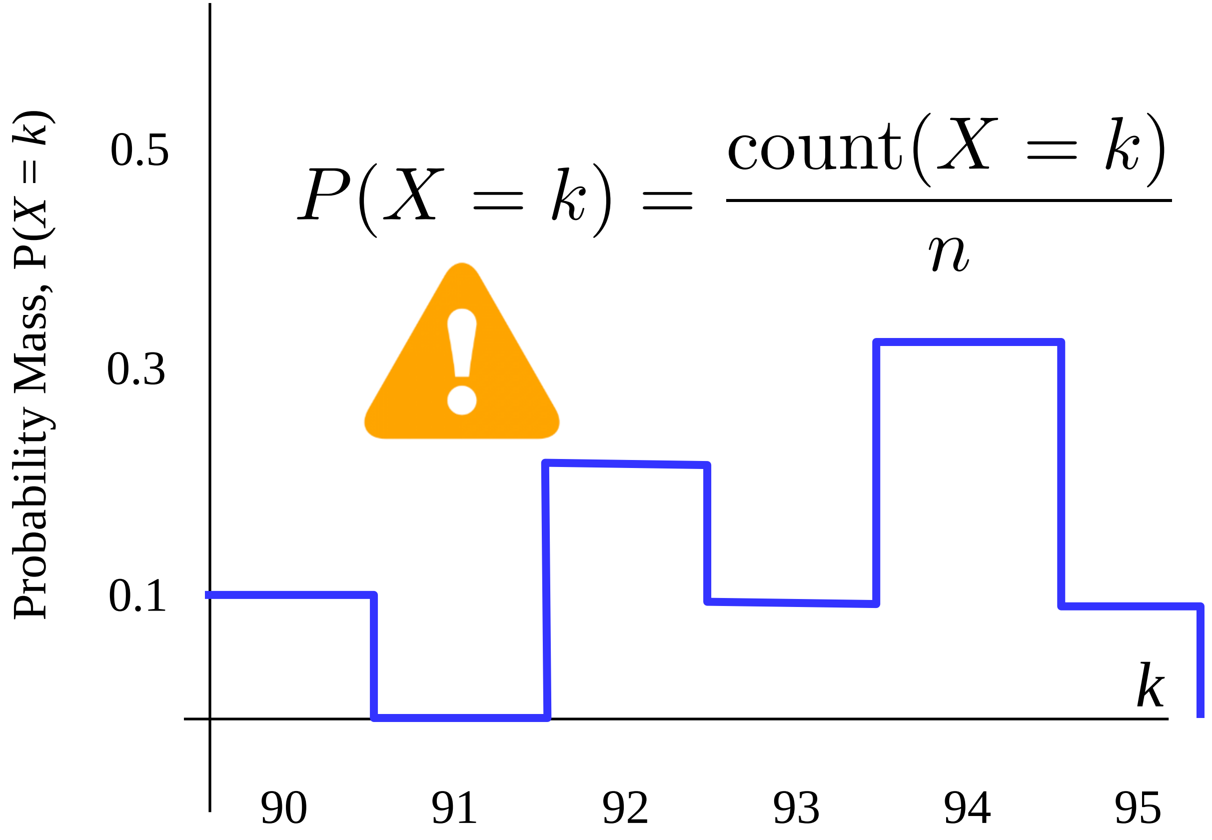
$$P(X = k) = \frac{\text{count}(X = k)}{n}$$

`np.random.choice(samples, K, replace = False)`

Original Samples: [92, 92, 93, 94, 94, 95]

Removed 90

$$P(X = k) = \frac{\text{count}(X = k)}{n}$$



Resample:

[94, 90]

`np.random.choice(samples, K, replace = False)`

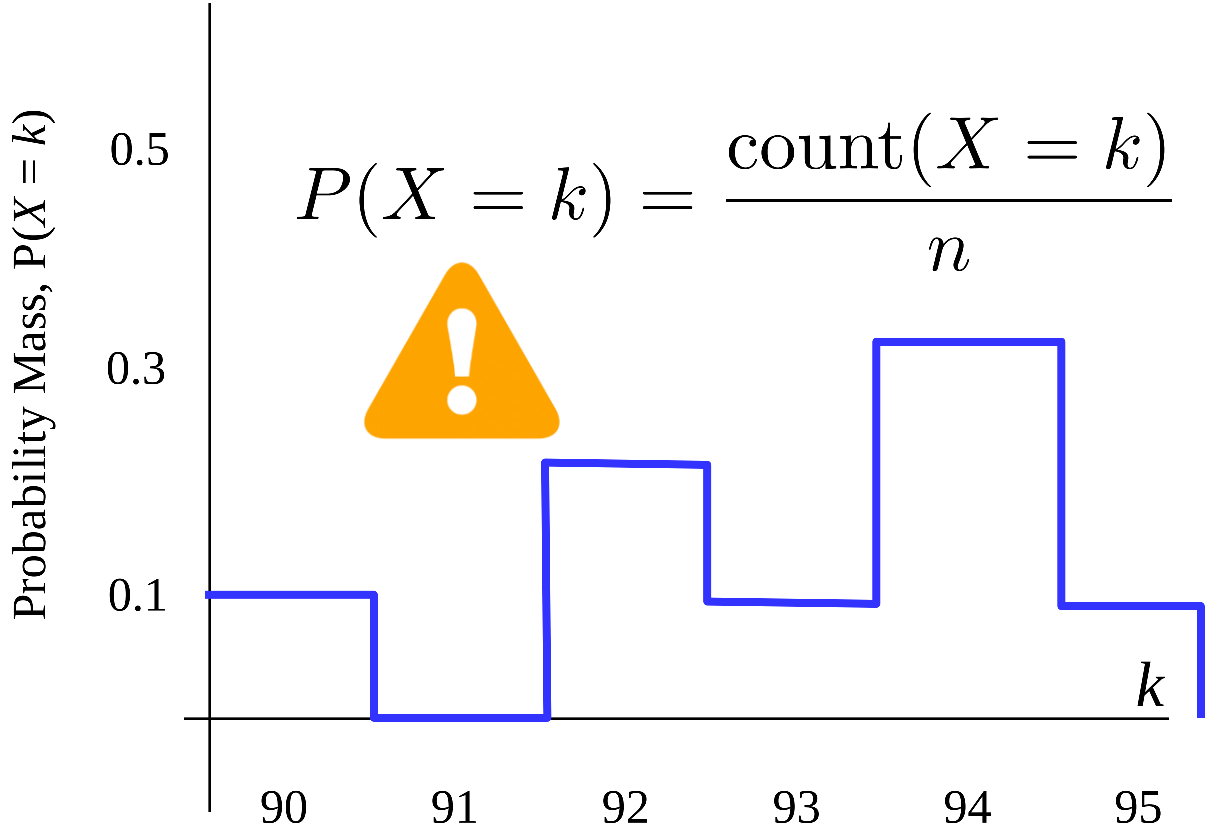
Original Samples: [92, 92, 93, 94, 94, 95]

Resample:

[94, 90]

Removed 90

$$P(X = k) = \frac{\text{count}(X = k)}{n}$$



The probability of sampling a 90 is no longer 0.1

The probability of sampling 94 is no longer 0.3

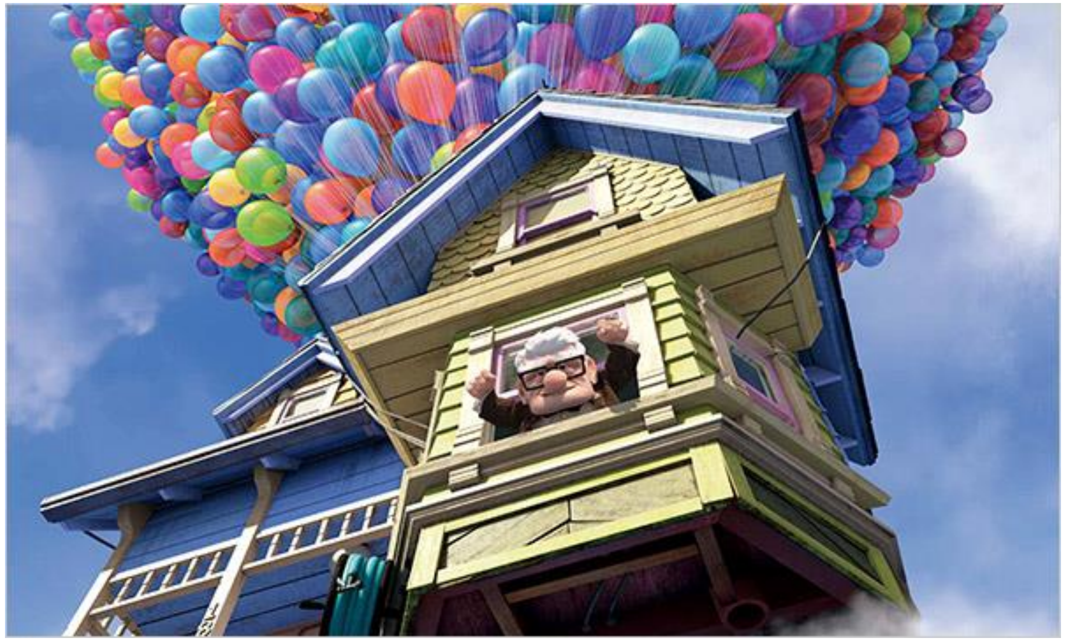
OG Bootstrapping

Bootstrap Algorithm (sample):

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

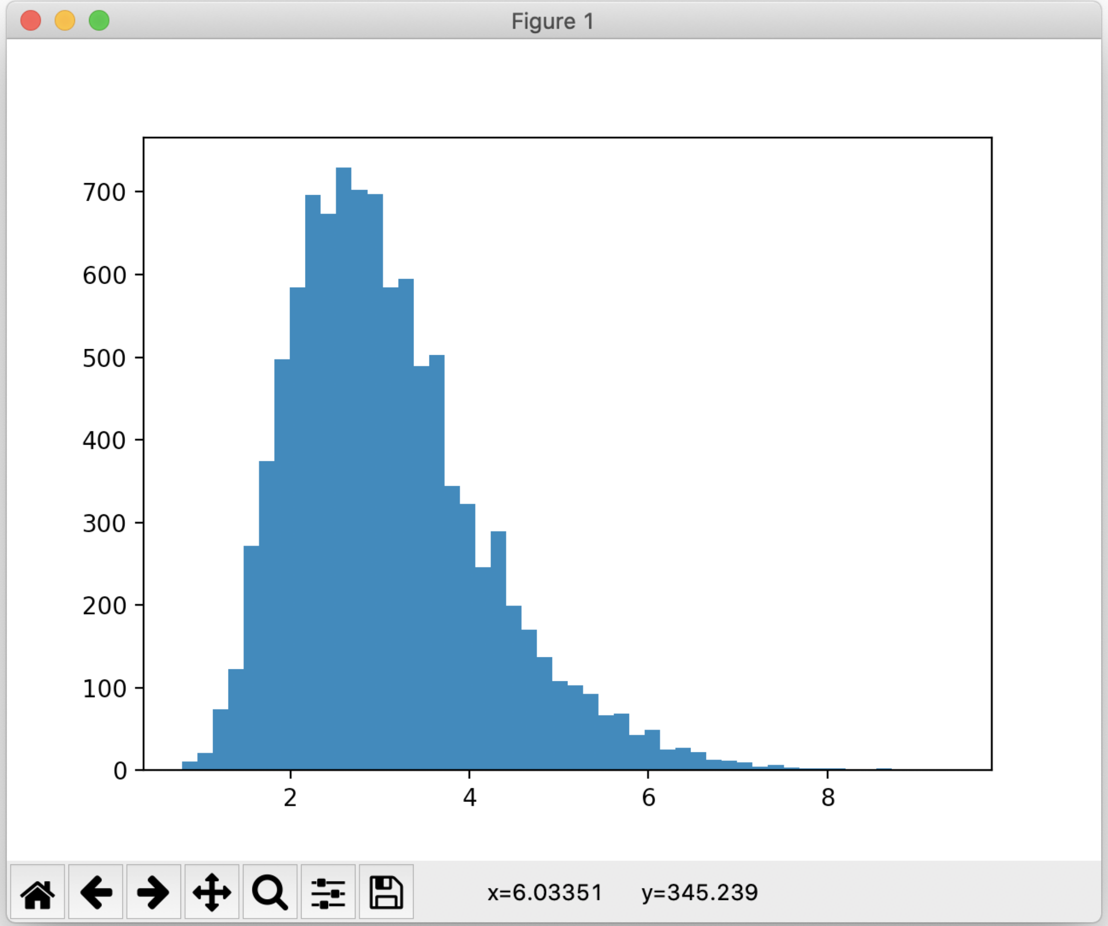
Bootstrapping in Practice

```
Bootstrap Algorithm (sample):
 1. Repeat 10,000 times:
   a. Choose len(sample) elems from sample,
      with replacement
   b. Recalculate the stat on the resample
 2. You now have a distribution of your stat
```



To the code!

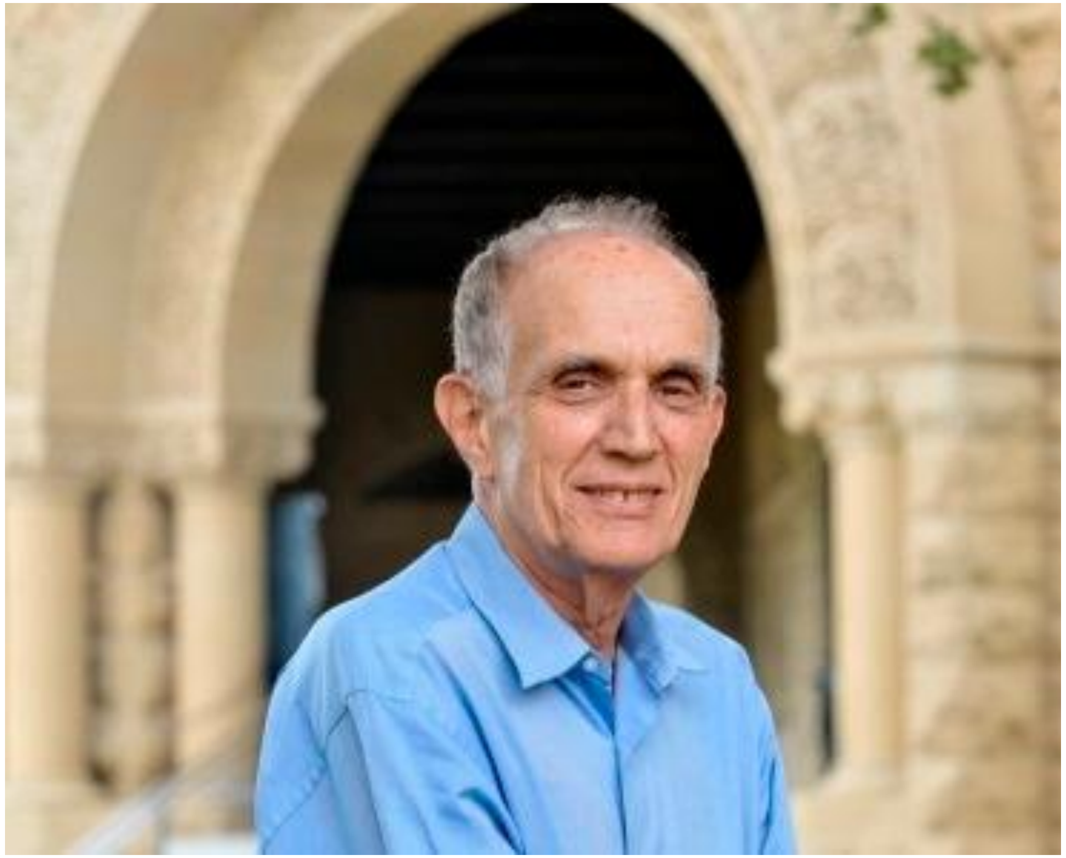
The Distribution of the Sampling Variance



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

Bootstrap

Bradley Efron



Invented bootstrapping in 1979

Still a professor at Stanford

Won a National Science Medal

According to starbyface.com:
Dolph Lundgren

Works for any statistic*

*as long as your samples are IID and the underlying distribution doesn’t have a long tail

The Classic Science Test

Group 1	Group 2
4.44	2.15
3.36	3.01
5.87	2.02
2.31	1.43
…	…
3.70	1.83

$\mu_1 = 3.1$ $\mu_2 = 2.4$

Claim: Group 1 and Group 2 are samples from **different distributions** with a 0.7 difference of means.

How confident are you in this claim?

A real difference?

18 students

Learning in Context A
4.44
3.36
5.87
2.31
…
3.70

$\mu_1 = 3.1$

23 students

Learning in Context B
2.15
3.01
2.02
1.43
…
1.83

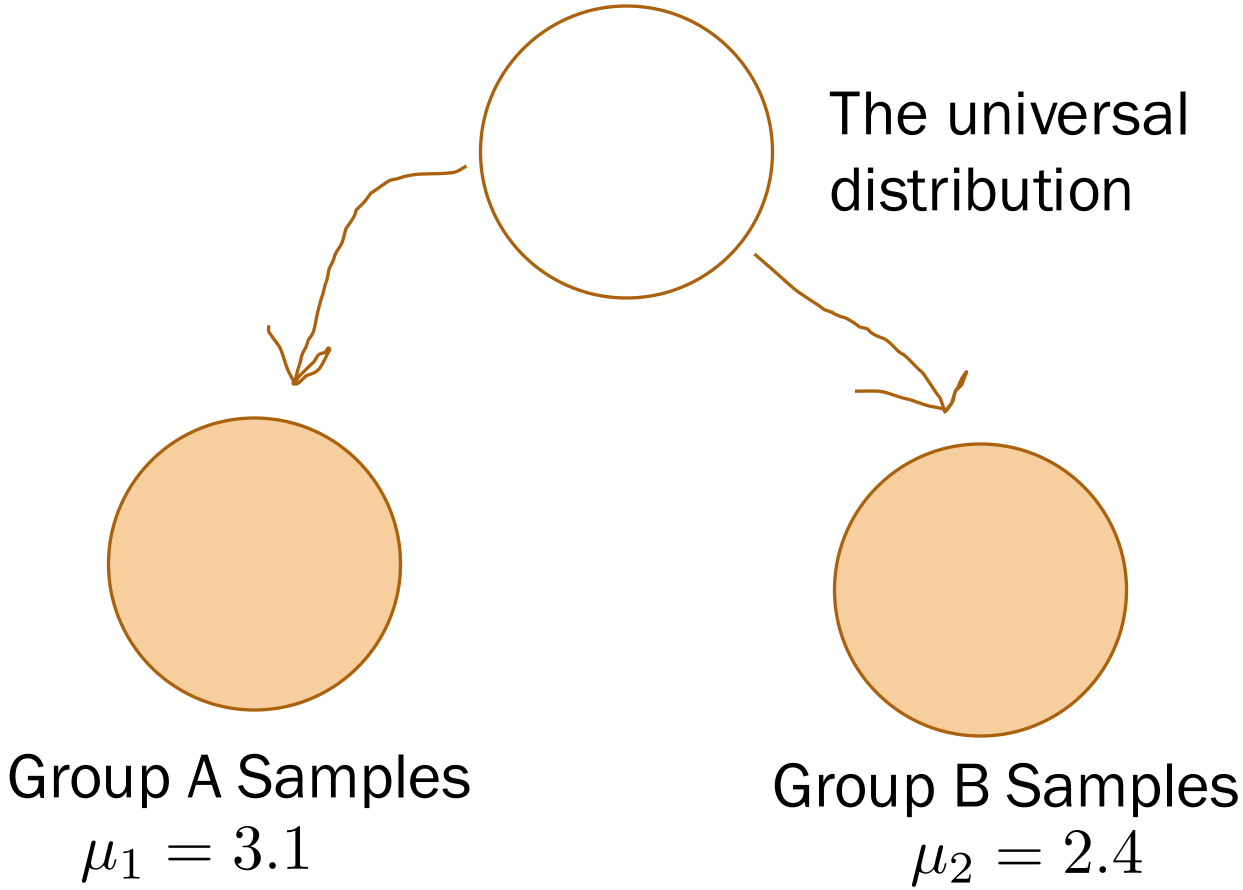
$\mu_2 = 2.4$

Claim: Group 1 and Group 2 are samples from **different distributions** with a 0.7 difference of means.

How confident are you in this claim?

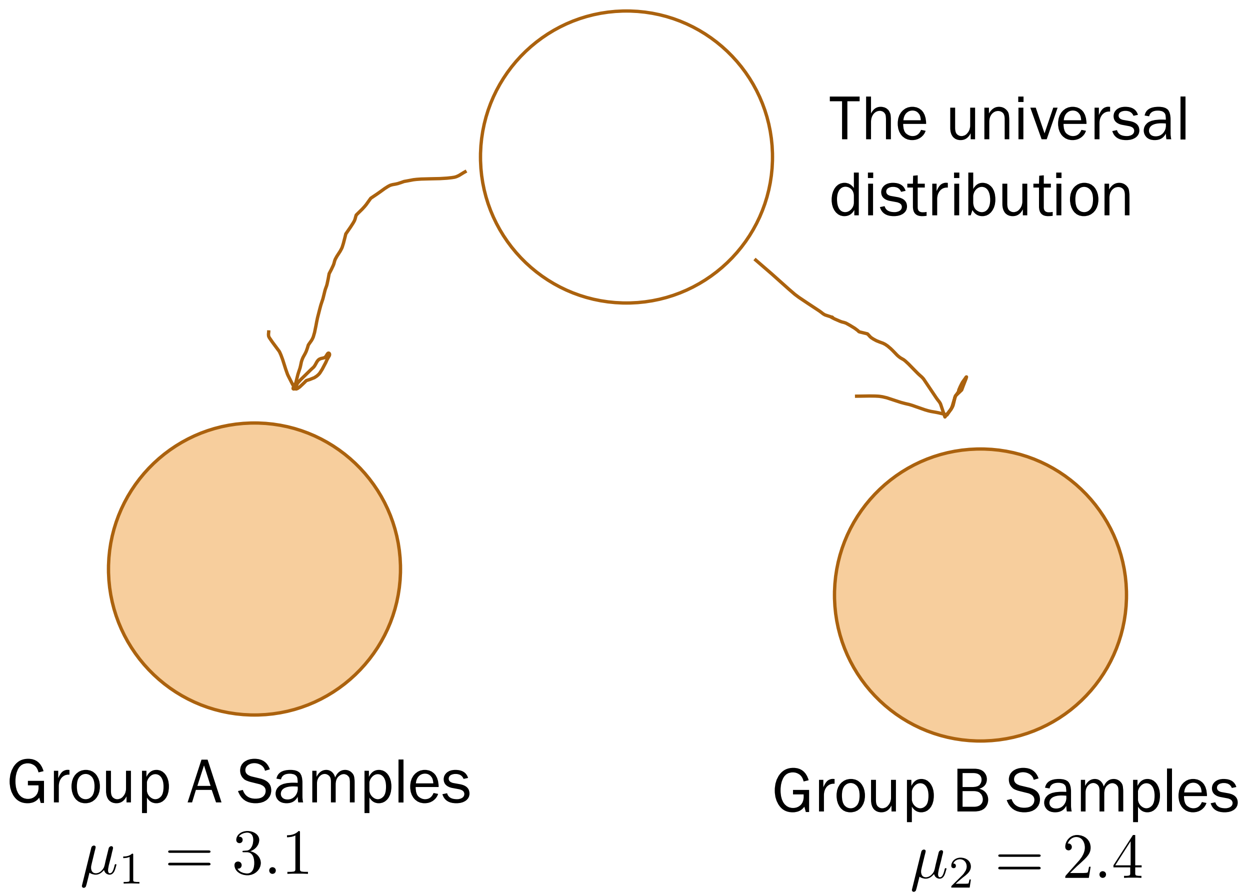
The Null Hypothesis

There is no difference between the two groups, so everyone is drawn from the same distribution. Any difference you observe is due to sampling error.



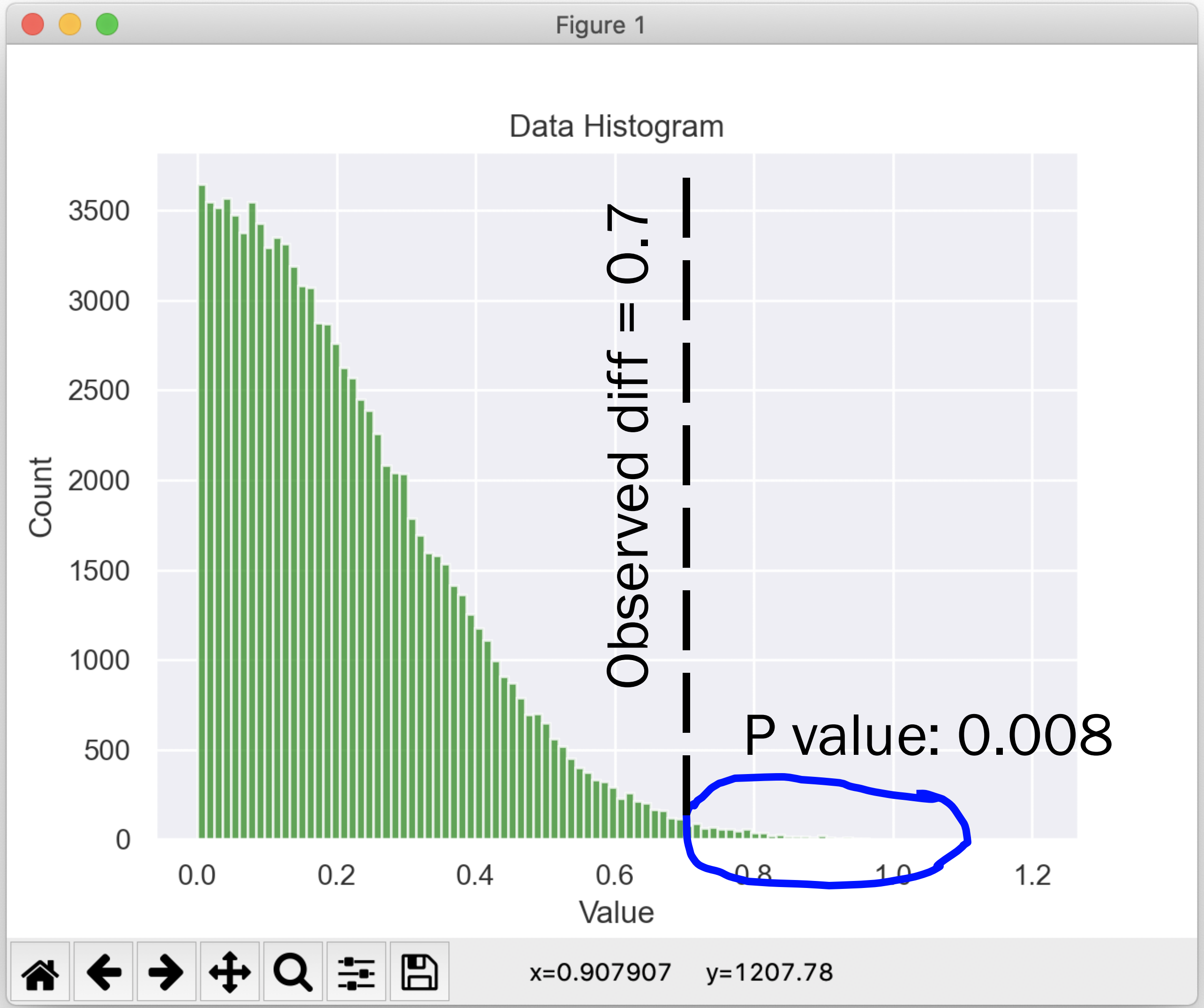
P-Value

The probability of obtaining test results **at least as extreme** as the result actually observed, if the null hypothesis is correct



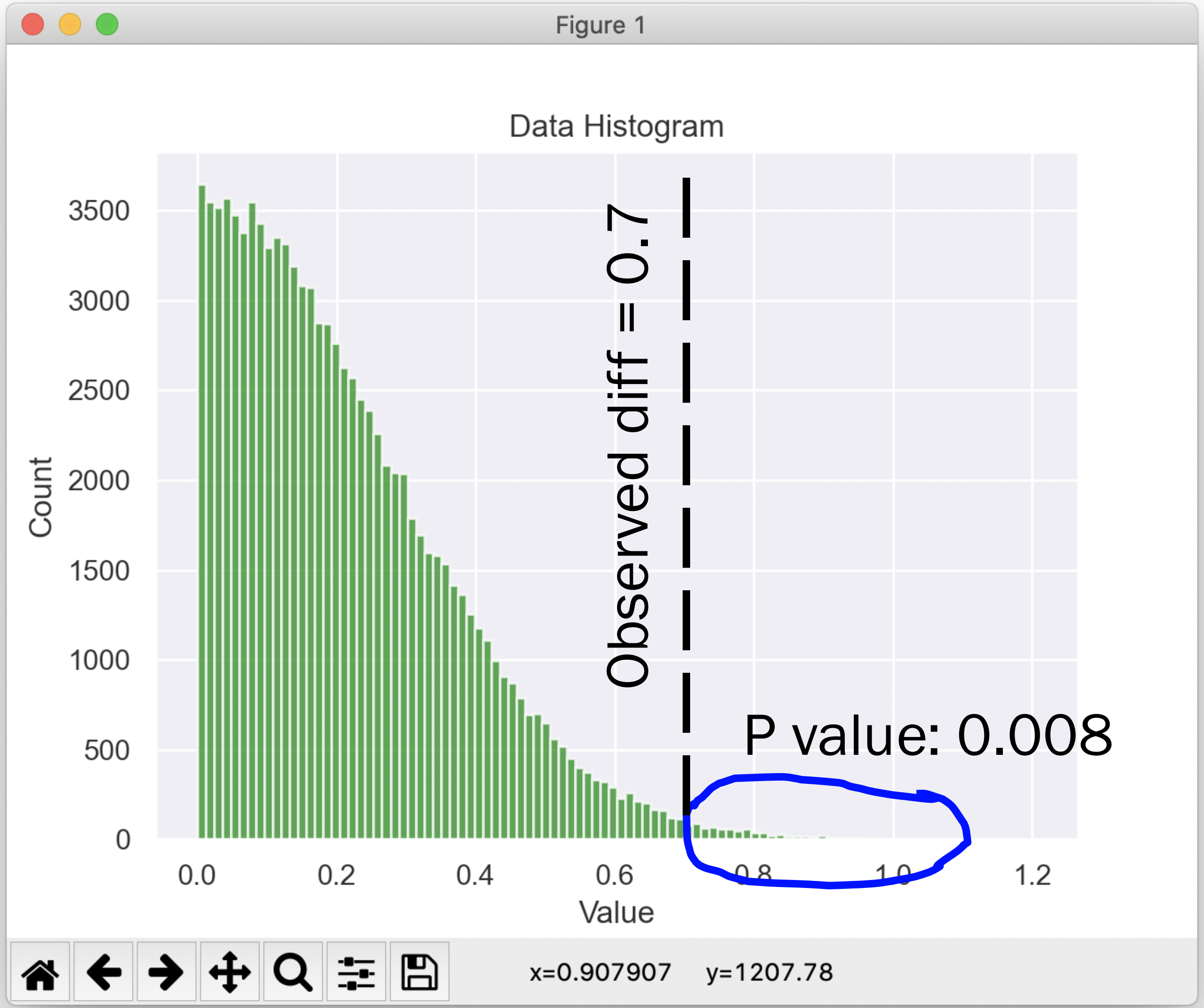
To the code!

Distribution of Mean Diffs under Null Hypothesis



Every* Science Result needs a p-value!

* almost



Food For Thought

Puzzle

Results of flipping a coin 20 times. Give your belief distribution of p:

H, H, H, T, H, T, H, H, H, H, H, T, H, H, H, H, H, H, T, H

4 tails, 16 heads

How can you build distribution for p without using a prior?

Two Opinions on Distributions

Results of flipping a coin 20 times. Give your belief distribution of p:

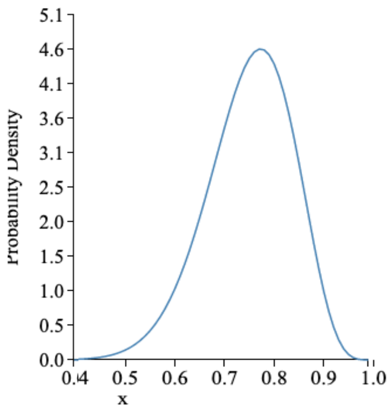
H, H, H, T, H, T, H, H, H, H, H, T, H, H, H, H, H, H, T, H

4 tails, 16 heads

Bayesian:

Let's use Laplace prior X ~ Beta(2, 2)

X ~ Beta(a = 18, b = 6)



Frequentist:

Let's bootstrap