CS 696 Intro to Big Data: Tools and Methods
Fall Semester, 2017
Doc 6 Sampling
Sep 11, 2017

Sampling - Motivation

How to find mean and median of 1 Billion values?

Web browser wants to warn user when they request a known malicious website
- Could be millions of malicious websites
- Don't want to check server for each URL

Web Crawler
- Visit page A
- Extract all links from page A
- Repeat process on all links from page A
- How to know if you have already visited a page?
- Google indexes ~45 Billion web pages

Descriptive Statistics

mean
median
mode
variance
standard variation
quantiles

Descriptive Statistics

Arithmetic mean

mean(numbers) = sum(numbers)/length(numbers)

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

mean([1,7,3,8,5]) == 4.80

median

Middle value of sorted list of numbers

If even number of values then mean of middle two values

median([1,7,3,8,5]) == 5.00

mode

Value that appears the most in the data

Descriptive Statistics

Variance

Measures the spread in the numbers

$$s^2 = \frac{1}{n}\sum_{i=1}^{n}\left(x_i - \overline{x}\right)^2$$

Standard Deviation, (SD, s, σ)

square root of the variance

Quantiles

q-quantiles

Cutpoints that divide the sorted data into q equal sized groups

4-quantile, quartile

1 1 4 7 7 8 10 15 17 17 25 26

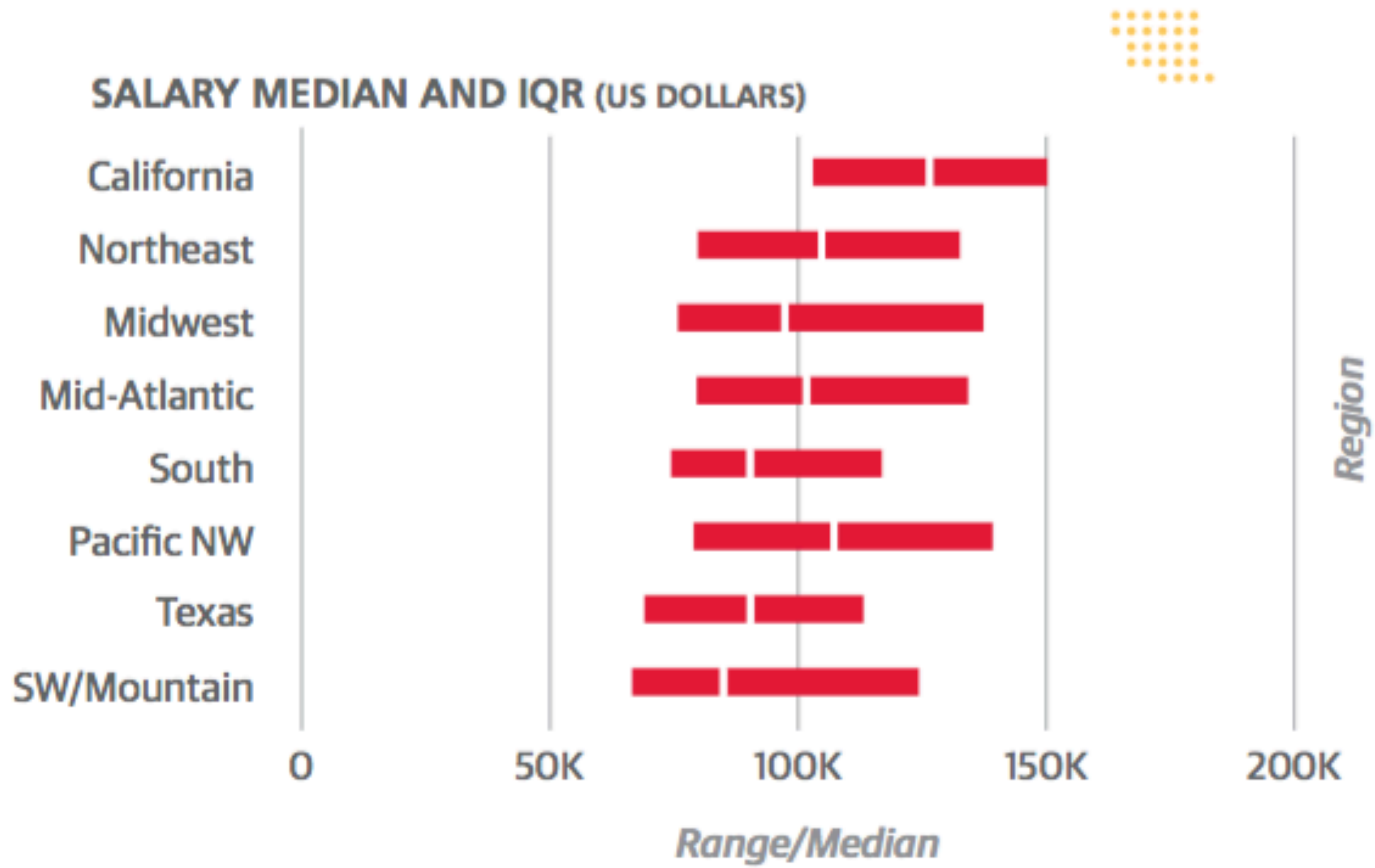
first quartile
Q1

third quartile
Q3

second quartile
median
Q2

Red Bar shows middle two quartiles

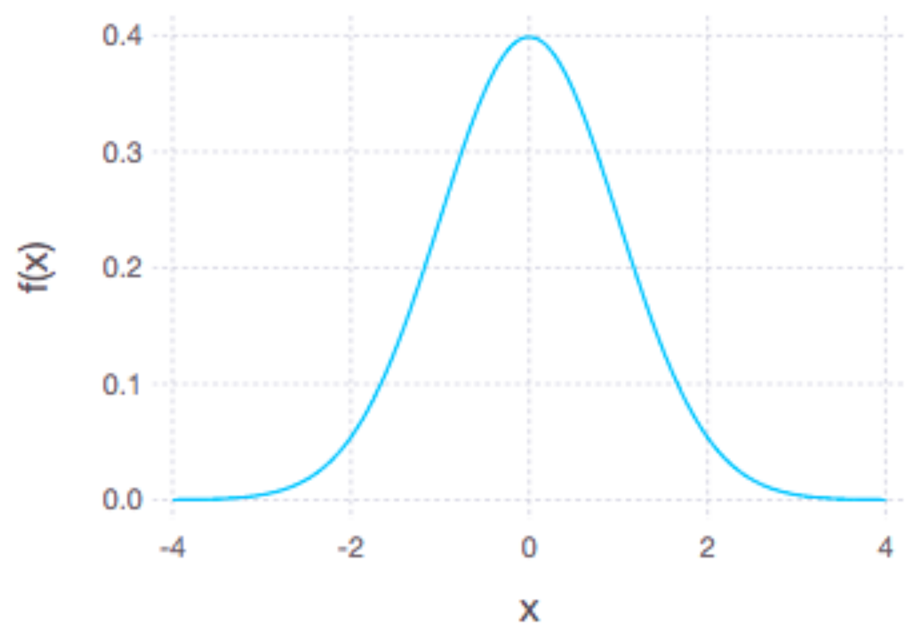
White bar is median

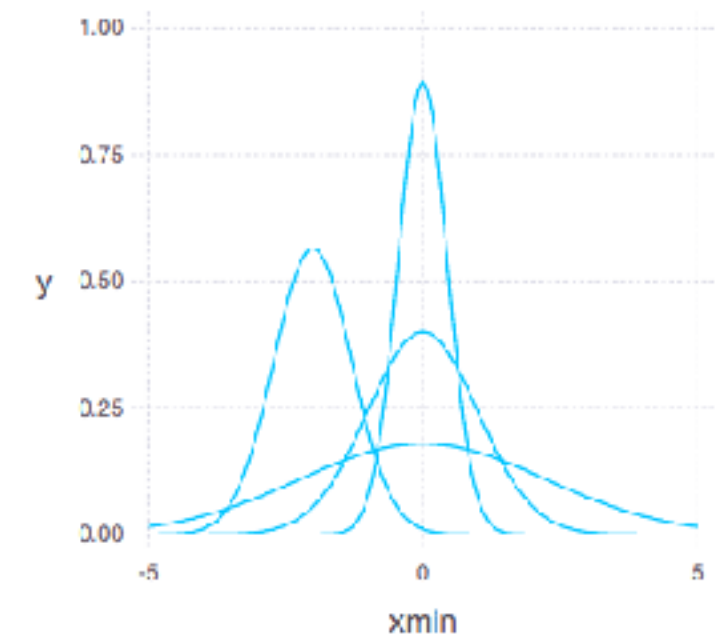


Distributions

Think in distributions not numbers

Normal (Gaussian) Distribution





$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\sigma^2\pi}}\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Normal distribution is specified by

μ - mean, central point

σ - standard deviation

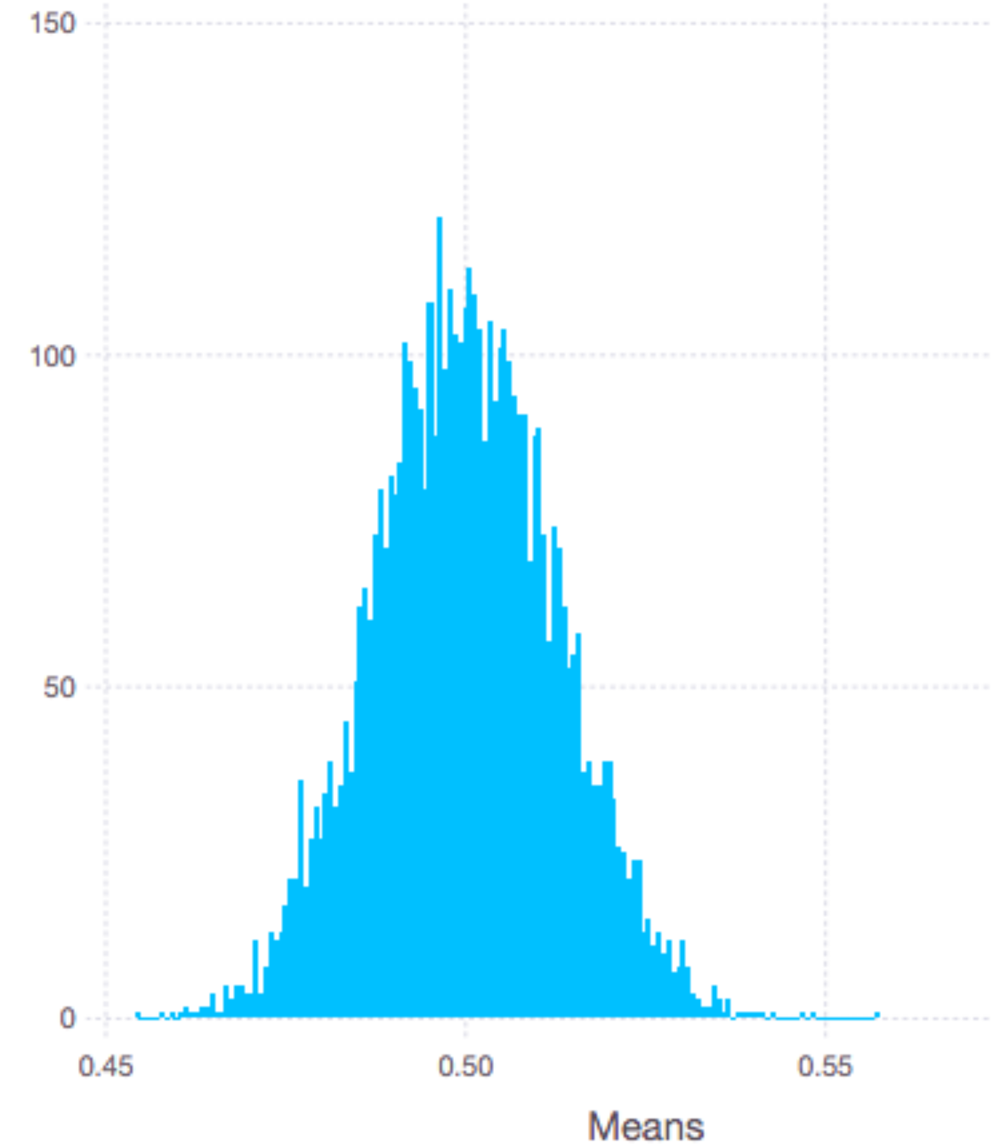
Central Limit Theorem

Let

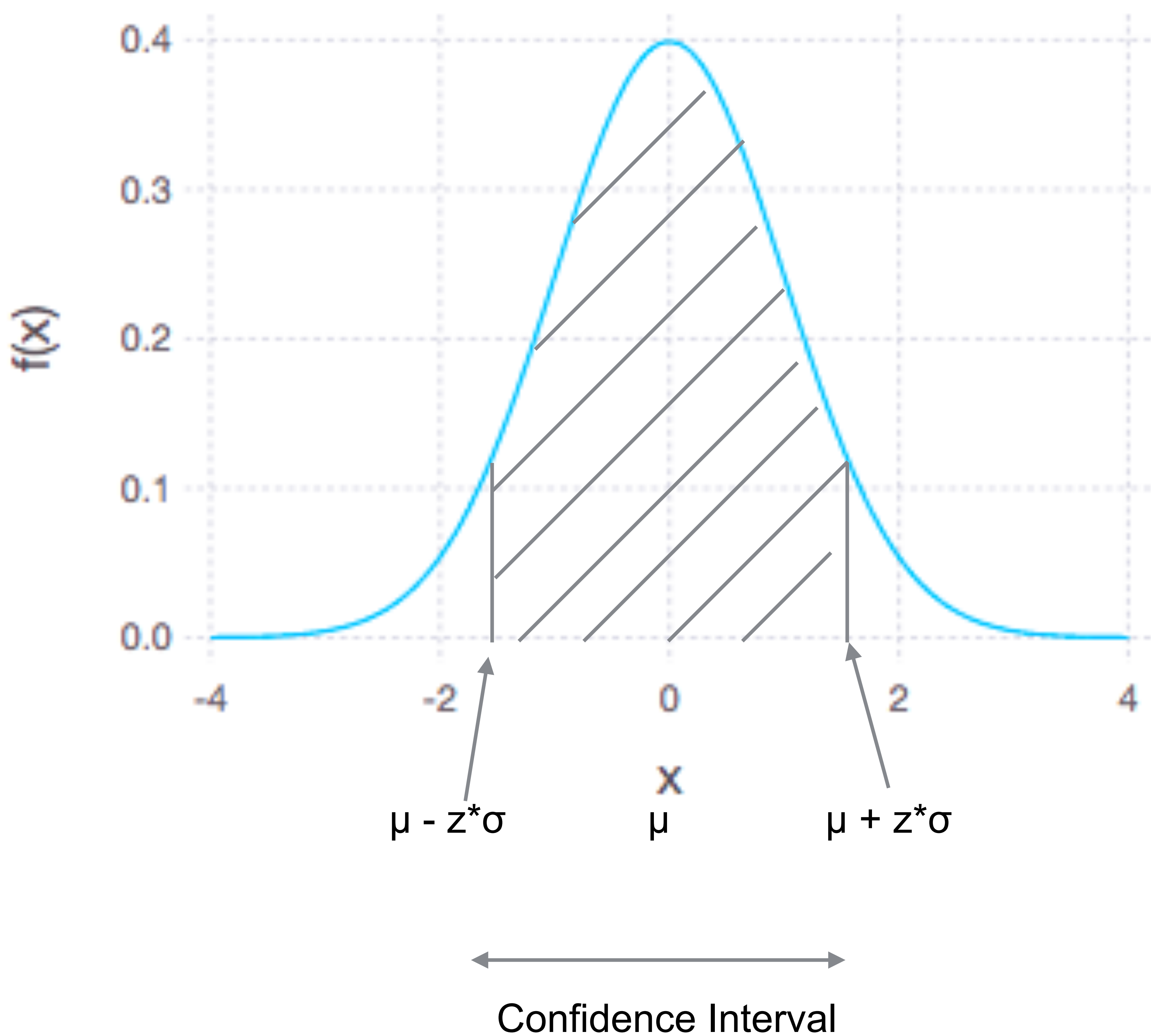
$X_1, X_2, ..., X_N$ random sample

$S_N = (X_1 + ... + X_N)/N$

Then as N gets large S_N approximates the normal distribution



Area in Shaded Part



Area	Z*
99%	2.576
98%	2.326
95%	1.96
90%	1.645

Populations & Samples

Measure	Sample statistic	Population parameter
Number of items	n	N
Mean	$\bar{x}$	μ_x
Standard deviation	S_x	σ_x
Standard error	$S_{\bar{x}}$	

Populations - all the items

Sample - set of representative items

Standard Error of sample = σ_x/sqrt(n)

Standard Error of mean (SEM)

Standard deviation of the sample-mean estimate of a population mean

Note to decrease the SE by 2 we need to increase the sample size by factor of 4

Sampling

100,000 data points

    Compute the average

Take random sample of 1000 compute average

    How close will sample average be to actual average?

Let   s = average of the sample

        n = sample size = 1000

Standard Error = standard deviation = s/sqrt(n)

Sampling

Let s = average of the sample
n = sample size = 1000

Standard Error = standard deviation = s/sqrt(n)

Confidence Interval (s - z*s/sqrt(n), s + z*s/sqrt(n))

Width of confidence interval = s + z*s/sqrt(n) - (s - z*s/sqrt(n))
= s + z*s/sqrt(n) - s + z*s/sqrt(n)
= z*s/sqrt(n) + z*s/sqrt(n)
= 2z*s/sqrt(n)

Sampling

Confidence Interval (s - z*s/sqrt(n), s + z*s/sqrt(n))

Experiment

100,000 random integer between 0 and 1000

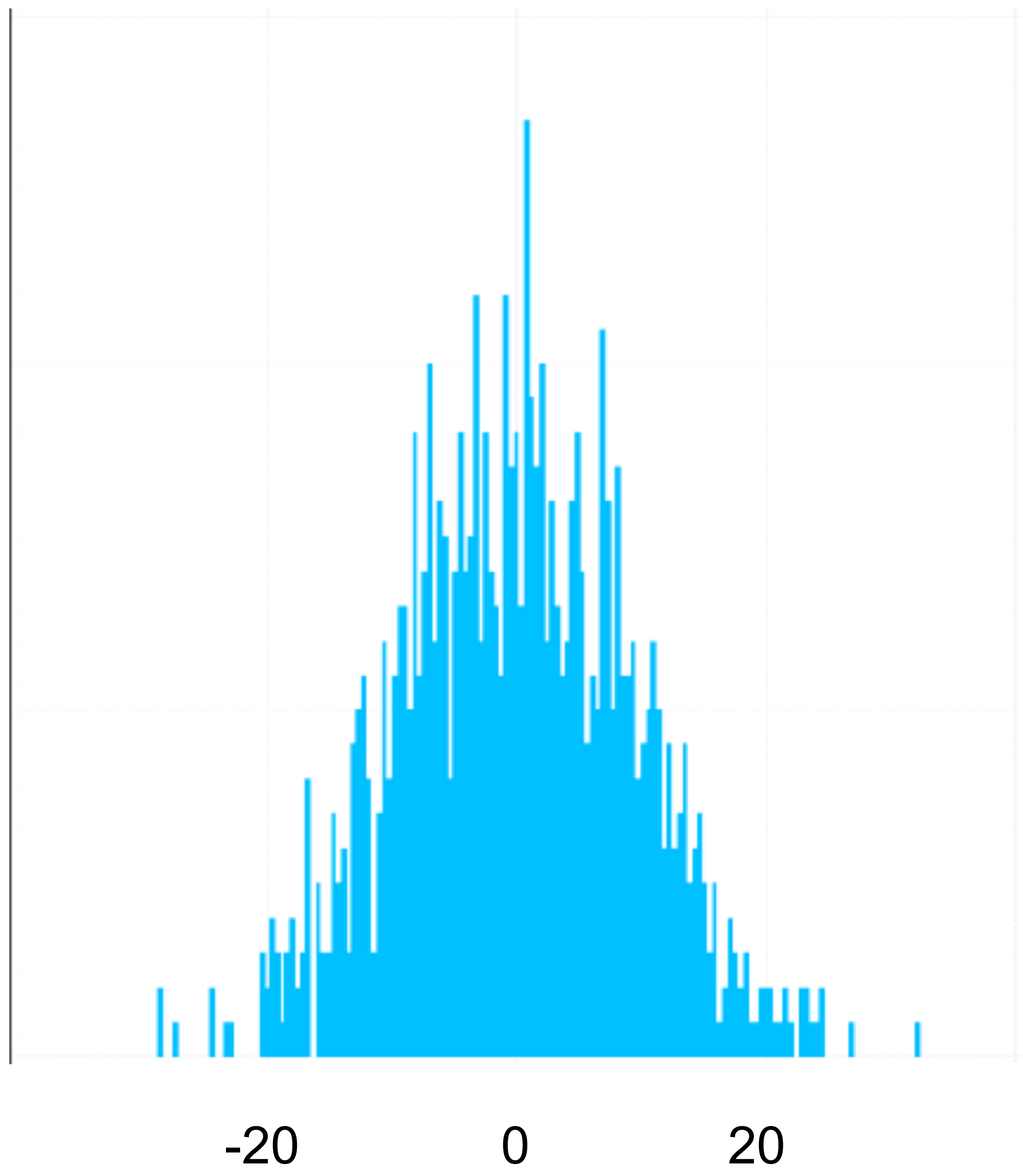
Sample size 1,000

Sample mean (s) = 532.33

Confidence Interval at 95% = (499.3, 565.3)

Actual mean = 501.4

Sample Mean - Population Mean



Sample Size = 1000
Number of Sample = 1000

What if we want sample to be within 10?

Width of confidence interval = W = 2z*s/sqrt(n)

$n = 4z^{*2}s^2/W^2$
$= 4 * 1.96^2 * 501.4^2/10^2$
≈ 39000

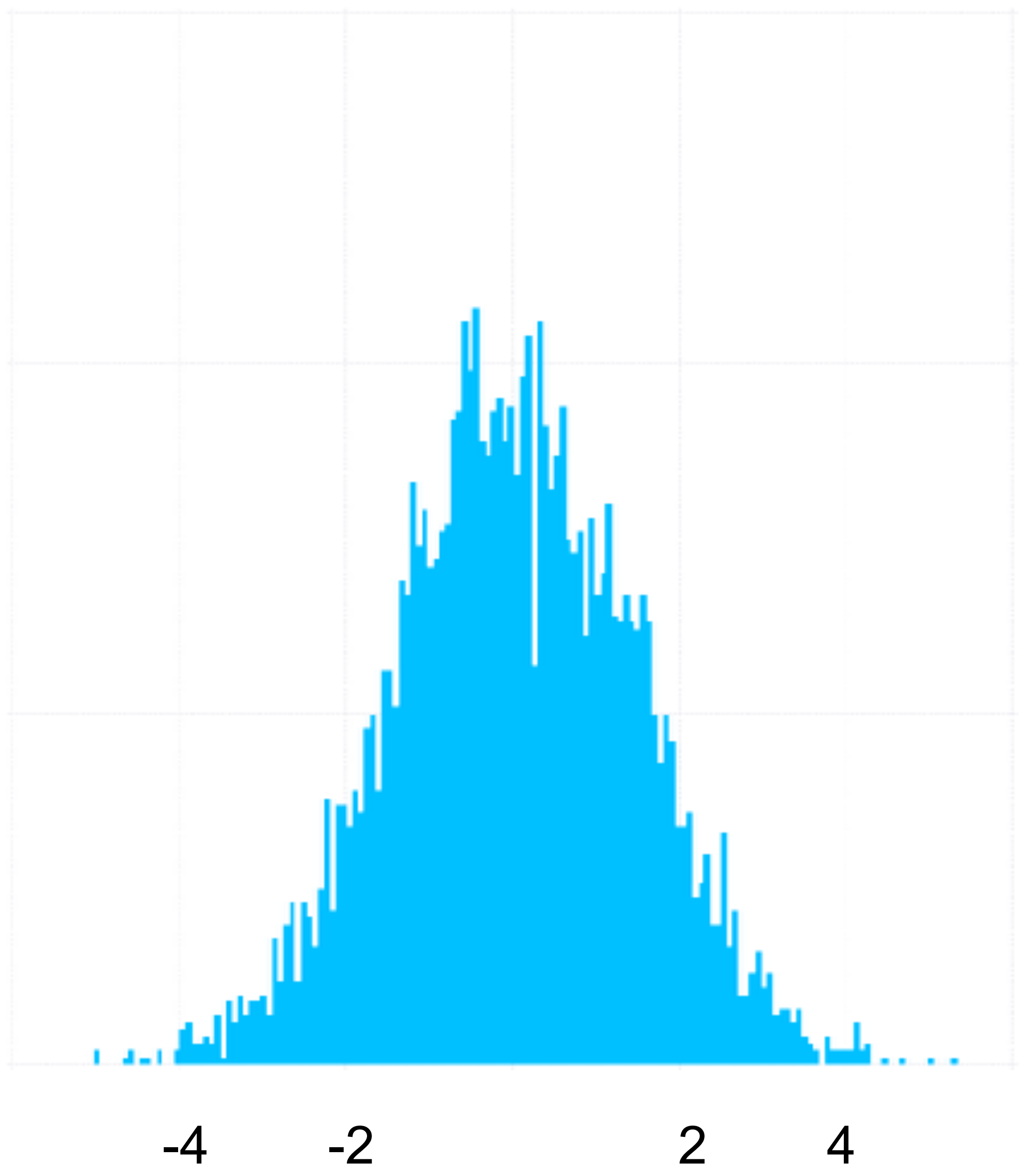
Mean of samples of size 39000

502.37
500.795
503.108
502.488
499.351
499.907
500.791
501.248
501.814
501.707
⋮
504.143
500.595

Population mean

501.4

Sample Mean - Population Mean



Sample Size = 39000

Number of Sample = 5000

Bloom Filter

Burton Bloom - 1970

Space-efficient probabilistic data structure

Test whether an element is in a set

Bloom filter does not contain the elements in the set

False positive matches are possible
    Possibly in set

False negatives are not possible
    Definitely not in set

Types of Errors

False Positive (FP), type I error

Accepting a statement as true when it is not true

False Negative (FN), type II error

Accepting a statement as false when it is true

Bloom Filter - How it works

Empty Bloom filter

- m bits all 0
- k different hash functions

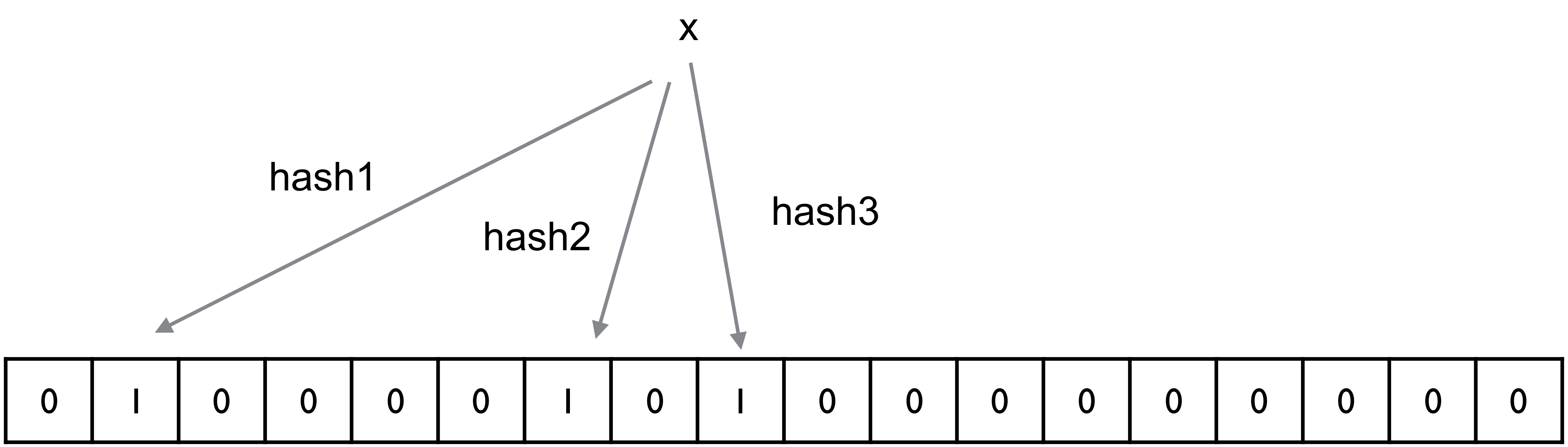
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bloom Filter - How it works

m = 18

k = 3

Insert x

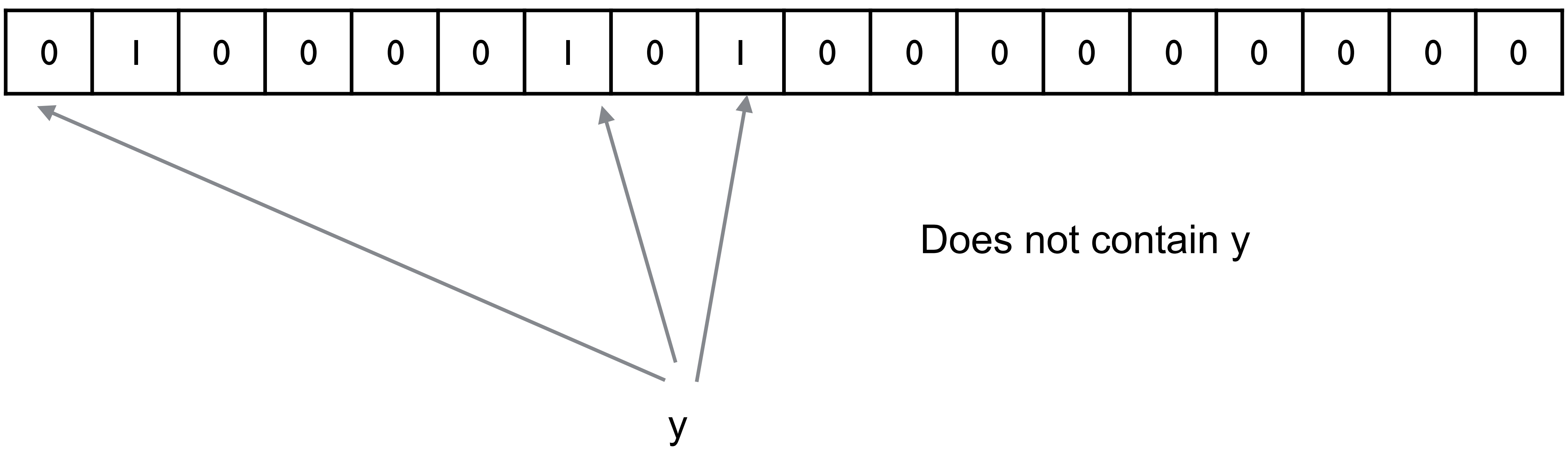


Bloom Filter - How it works

m = 18
k = 3

Contains y?

{x}



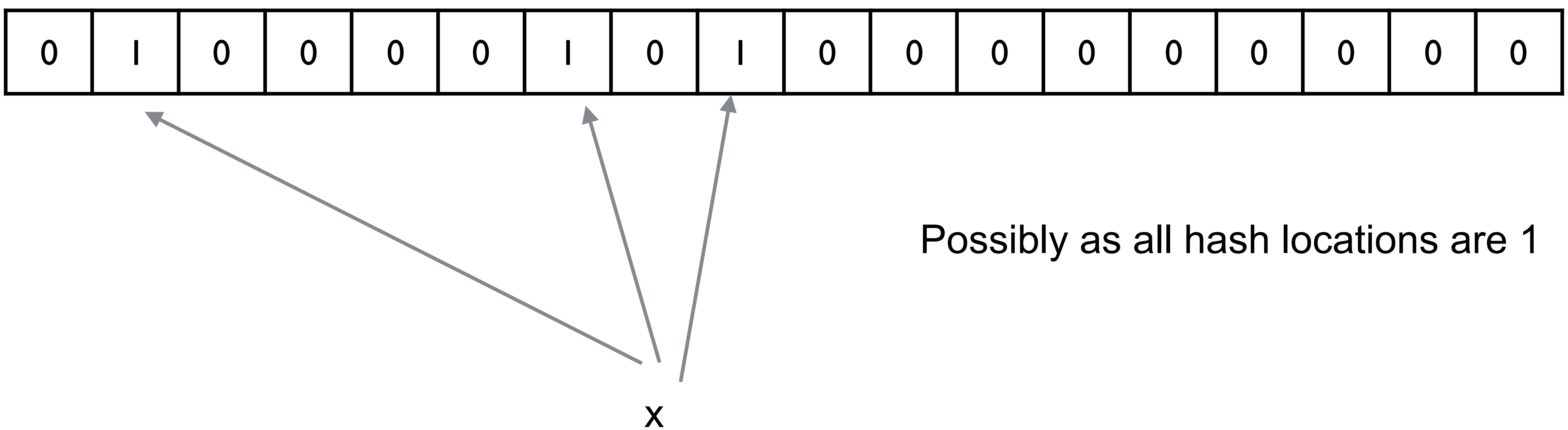
Does not contain y

Bloom Filter - How it works

m = 18
k = 3

Contains x?

{x}



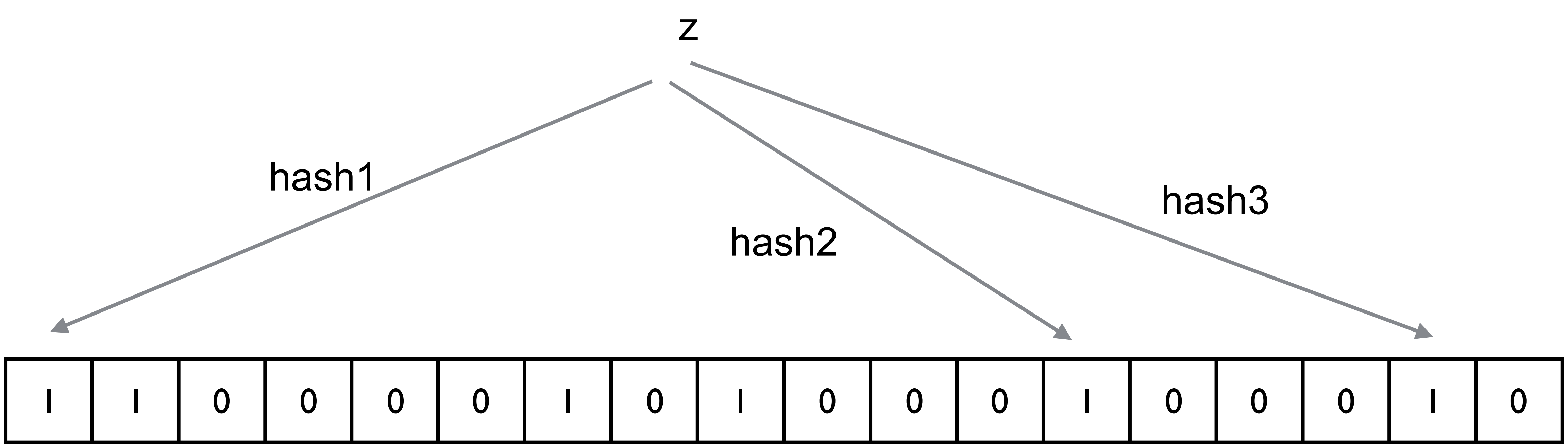
Possibly as all hash locations are 1

Bloom Filter - How it works

m = 18
k = 3

Insert z

{x}

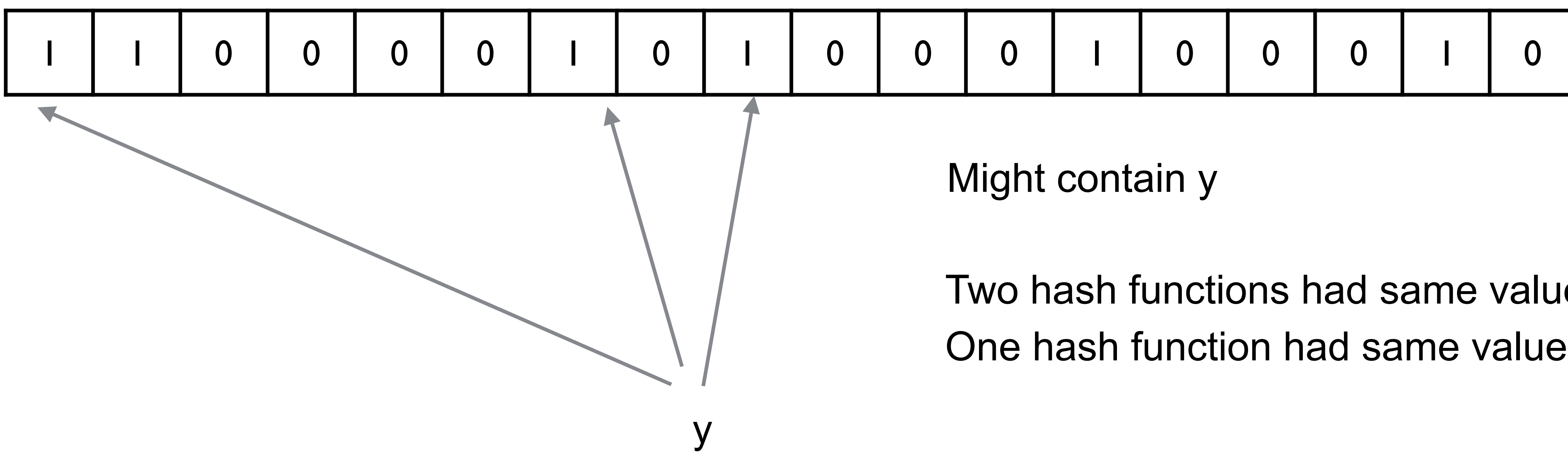


Bloom Filter - How it works

m = 18
k = 3

Contains y?

{x, z}



Might contain y

Two hash functions had same value as x
One hash function had same value of z

Bloom Filter - How it works

Larger m

- Decreases false positives
- Increases table size - fewer collisions

Larger k

- Decreases false positives up to a point
- But fills table faster

Bloom filter for Scala

https://github.com/alexandrnikitin/bloom-filter-scala

```
// Create a Bloom filter
val expectedElements = 1000000
val falsePositiveRate = 0.1
val bf = BloomFilter[String](expectedElements, falsePositiveRate)

// Put an element
bf.add(element)

// Check whether an element in a set
bf.mightContain(element)

// Dispose the instance
bf.dispose()
```

Bloom Filter - Sample Uses

Akamai’s web servers

- Some pages are only accessed once - One-hit-wonders
- Only cache web page after second time it is accessed
- Use bloom filter to determine if page has been seen before

Google BigTable, Apache HBase and Apache Cassandra, and Postgresql

- Use Bloom filters to see if rows or columns exist
- Avoid costly disk access on nonexistent rows

Google Chrome web browser

- Use Bloom filter to identify malicious URLs
- If filter contains the url then check server to make sure

Medium

- Uses Bloom filters to avoid recommending articles a user has previously read

Heavy Hitters Problem

Streaming

Real time

Computing popular products

- Given the page views on Amazon which products are viewed the most?

Computing frequent search queries

- Given the stream of Google searches what are the popular searches
- 3.5 billion searches per day

View Tweets

- How often are trees viewed? What the most popular tweets?

Heavy Network flows

- Given packet count source and destination through switch
- Where is the traffic the heaviest?
- Cisco Nexus 9500 - 172.8 Tbps
- Useful to detect DoS attacks

Volatile Stocks

- Given stream of stock transactions which stocks are
  - Traded the most
  - Change prices the most

Count-Min Sketch

Graham Cormode and S. Muthu Muthukrishnan - 2003

Consume a stream of events

Count the frequency of the different types of events in the stream

Does not store the events

Counts for each event type

- Estimate of actual count
- Within given range of actual count with given probability

Count-Min Sketch - How it works

Initial count-min sketch

- w - columns
- d - rows
- d different hash functions
- All entries integers = 0

w determines

- Interval length containing actual count

d determines

- Probability that actual count is in interval

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

Count-Min Sketch - How it works

Event x

x

hash1

hash2

hash3

0	1	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0

Count-Min Sketch - How it works

Event y

y

0	1	1	0	0	0	0	0	0	0
0	0	0	0	2	0	0	0	0	0
0	0	0	0	0	1	0	0	1	0

Count-Min Sketch - How it works

Event x

x

0	2	1	0	0	0	0	0	0	0
0	0	0	0	3	0	0	0	0	0
0	0	0	0	0	2	0	0	1	0

Count-Min Sketch - How it works

Event z

z

0	2	2	0	0	0	0	0	0	0
0	0	0	1	3	0	0	0	0	0
0	1	0	0	0	2	0	0	1	0

Count-Min Sketch - How it works

How often did x occur?

Look at counts for x in each row

Return the minimum count

x

0	2	2	0	0	0	0	0	0	0
0	0	0	1	3	0	0	0	0	0
0	1	0	0	0	2	0	0	1	0