Probability and Random Processes

ECS 315

Asst. Prof. Dr. Prapun Suksompong

prapun@siit.tu.ac.th

Discrete Random Variable

Office Hours:
BKD 3601-7

Monday	**14:00-16:00**
Wednesday	**14:40-16:00**

Discrete Random Variable

- X is a **discrete** random variable if it has a countable support.
  - Recall that countable sets include finites set and countably infinite sets.
- For X whose support is uncountable, there are two types:
  - **Continuous** random variable
  - ***Mixed*** random variable

Probabilities involving discrete RV

- Back to example of rolling a dice
- The "important" probabilities are

$$P[X=1]=P[X=2]=\cdots=P[X=6]=\frac{1}{6}$$

- In tabular form:

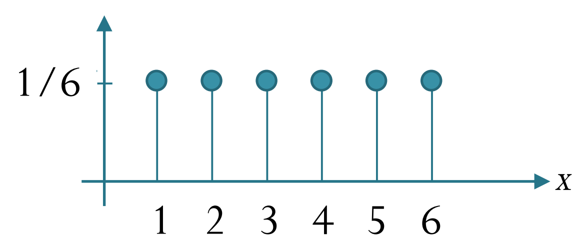
Dummy variable

x	$P[X=x]$
1	1/6
2	1/6
3	1/6
4	1/6
5	1/6
6	1/6

- **Probability mass function (PMF)**:

$$p_X(x)=\begin{cases}1/6, & x=1,2,3,4,5,6,\\ 0, & \text{otherwise.}\end{cases}$$

  - In general, $p_X(x)\equiv P[X=x]$
- Stem plot:



Probabilities involving discrete RV

To find P[some condition(s) on X] from the pmf $p_X(x)$ of X:

1. Find the support of X.
2. Look only at values x inside the support.
   Find all x that satisfies the condition(s).
3. Evaluate the pmf at x found in the previous step.
4. Add the pmf values from the previous step.

Back to the dice roll example. Suppose we want to find $P[X > 4]$.

1. The support of X is $\{1,2,3,4,5,6\}$.
2. The members which satisfies the condition ">4" is 5 and 6.
3. The pmf values at 5 and 6 are all $1/6$.
4. Adding the pmf values gives $2/6 = 1/3$.

Probabilities involving discrete RV

- Back to example of rolling a dice
- The "important" probabilities are

$$P[X=1]=P[X=2]=\cdots=P[X=6]=\frac{1}{6}$$

- In tabular form:

Dummy variable

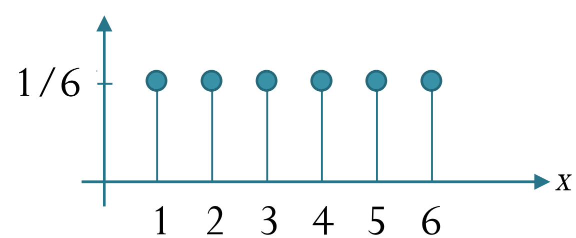
x	$P[X=x]$
1	1/6
2	1/6
3	1/6
4	1/6
5	1/6
6	1/6

- **Probability mass function (PMF)**:

$$p_X(x)=\begin{cases}1/6, & x=1,2,3,4,5,6,\\ 0, & \text{otherwise.}\end{cases}$$

  - In general, $p_X(x)\equiv P[X=x]$

- Stem plot:



Roll a fair dice. Record the result.

$$X \sim \text{Uniform}(\{1,2,\ldots,6\})$$


```
>> X = randi(6)

X =

     5
```

Again, roll a fair dice. Record the result.

```
>> X = randi(6)

X =

     6
```

Again, roll a fair dice. Record the result.

```
>> X = randi(6)

X =

     1
```

Again, roll a fair dice. Record the result.

```
>> X = randi(6)

X =

     6
```

Again, roll a fair dice. Record the result.

```
>> X = randi(6)

X =

     4
```

Again, roll a fair dice. Record the result.

```
>> X = randi(6)

X =

     1
```

```
>> X = randi(6,20,10)

X =
```

Generate X 200 times. Put the results in a table of size 20×10

2	5	3	4	5	2	1	2	6	3
4	3	4	1	5	4	3	1	6	4
6	4	5	1	3	5	2	2	4	5
6	2	5	2	4	5	5	2	1	1
1	5	2	6	1	3	3	3	2	6
6	1	5	2	1	1	6	1	3	5
6	2	4	5	4	2	2	6	5	3
3	1	1	2	5	6	2	6	1	3
5	1	1	6	6	1	1	3	1	3
1	5	3	3	1	5	1	3	2	2
3	5	6	2	4	4	6	3	4	4
6	2	3	2	3	6	4	6	5	4
5	6	4	4	1	1	4	3	4	5
6	1	2	3	3	3	1	1	3	5
4	3	5	3	1	1	6	5	4	4
1	3	2	5	5	6	4	3	2	3
6	5	4	4	2	1	3	2	5	5
6	5	5	4	4	5	4	3	2	4
5	2	6	6	1	5	3	1	5	3
5	3	6	2	4	6	1	1	2	6

`randi` function

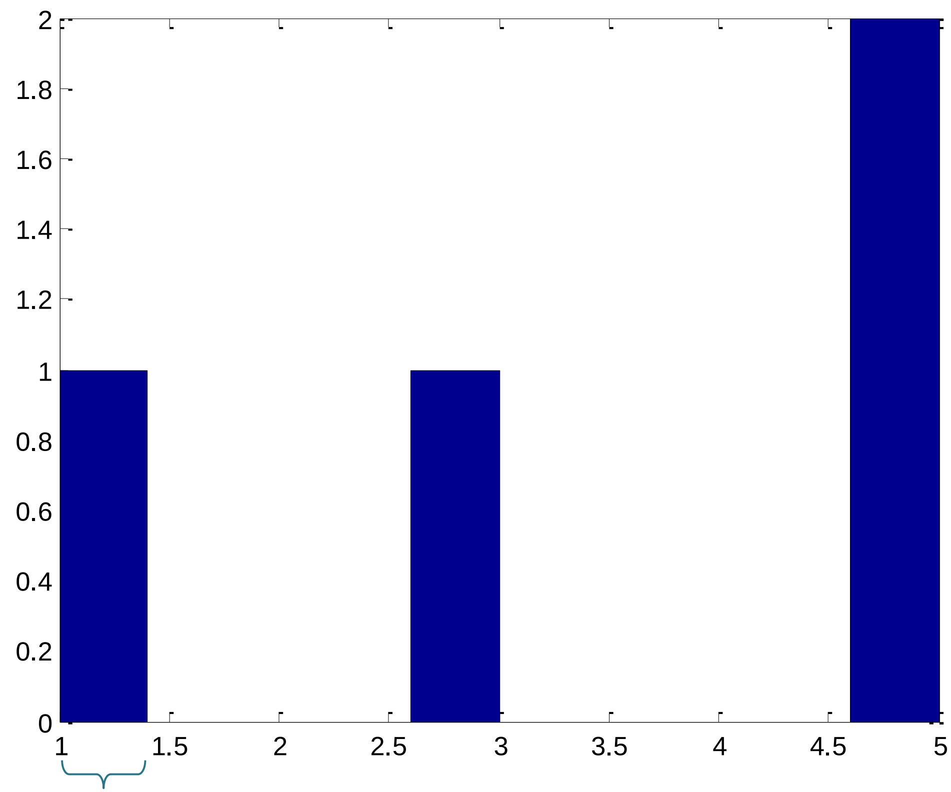
- Generate uniformly distributed pseudorandom **integers**
- `randi(imax)` returns a scalar value between 1 and `imax`.
- `randi(imax,m,n)` and `randi(imax,[m,n])` return an *m*-by-*n* matrix containing pseudorandom integer values drawn from the discrete uniform distribution on the interval [1,`imax`].
  - `randi(imax)` is the same as `randi(imax,1)`.
- `randi([imin,imax],...)` returns an array containing integer values drawn from the discrete uniform distribution on the interval [`imin`,`imax`].

`hist` function

- Create histogram plot
- `hist(data)` creates a histogram bar plot of data.
  - Elements in data are sorted into **10 equally spaced bins** along the x-axis **between the minimum and maximum** values of data.
  - Bins are displayed as rectangles such that the height of each rectangle indicates the number of elements in the bin.
  - If data is a vector, then one histogram is created.
  - If data is a matrix, then a histogram is created separately for each column.
    - Each histogram plot is displayed on the same figure with a different color.
- `hist(data,nbins)` sorts data into the number of bins specified by `nbins`.
- `hist(data,xcenters)`
  - The values in `xcenters` **specify the centers** for each bin on the x-axis.

`hist` function: Example

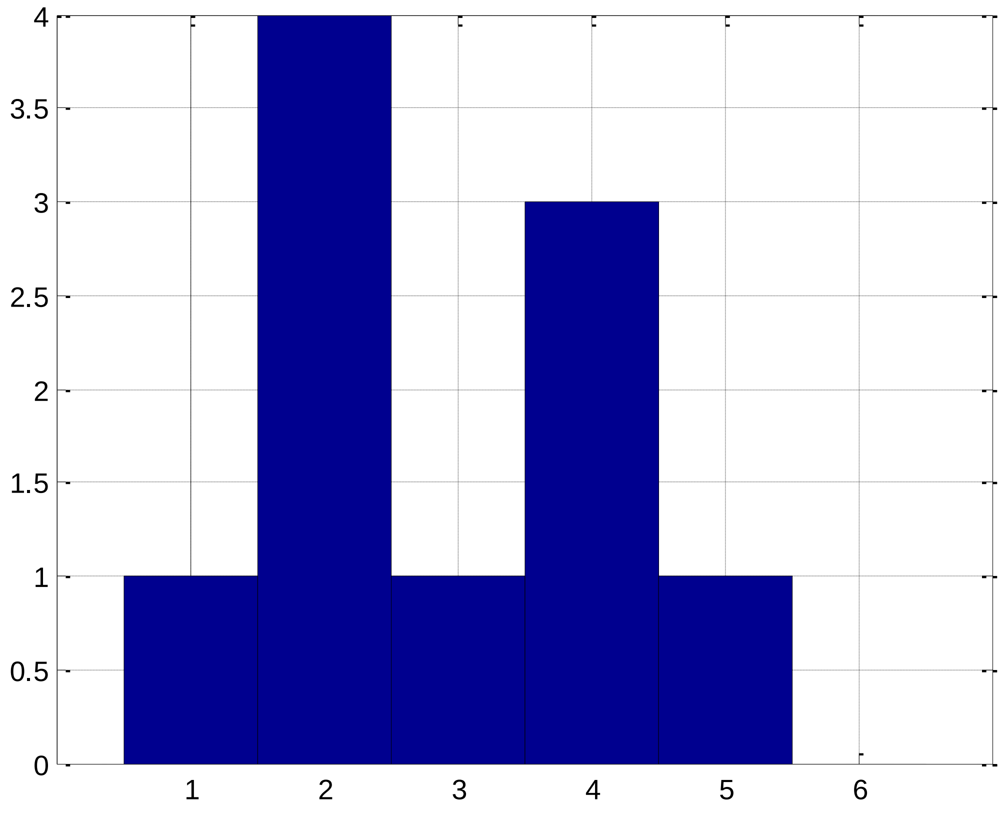
```
>> x = [1 3 5 5]

x =

     1     3     5     5

>> hist(x)
```



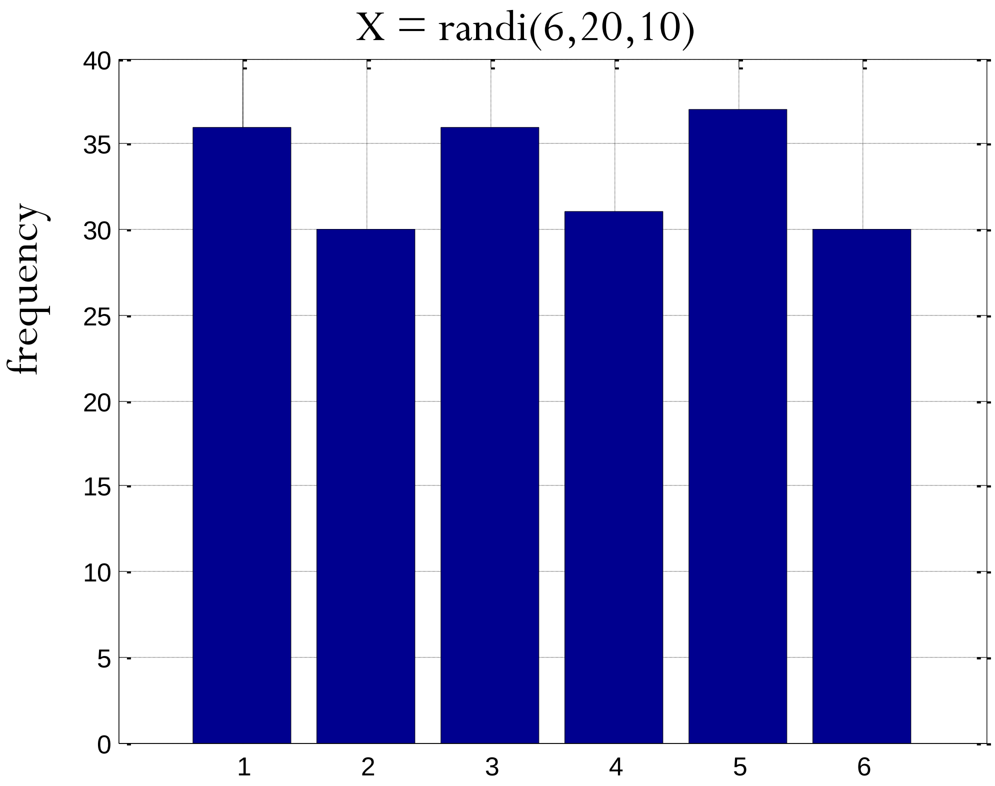
The width of each bin is $\frac{\overset{\text{max}}{5}-\overset{\text{min}}{1}}{10}=0.4$

`hist` function: Example

```
>> hist(reshape(X,1,prod(size(X))))
>> X = randi(6,1,10)
X =
     4     2     4     5     2     1     2     2     3     4
>> hist(reshape(X,1,prod(size(X))),1:6)
>> grid on
```



X ~ Uniform({1,2,...,6})



[N, x] = hist(reshape(X,1,prod(size(X))),1:6)
bar(x,N)
Grid on

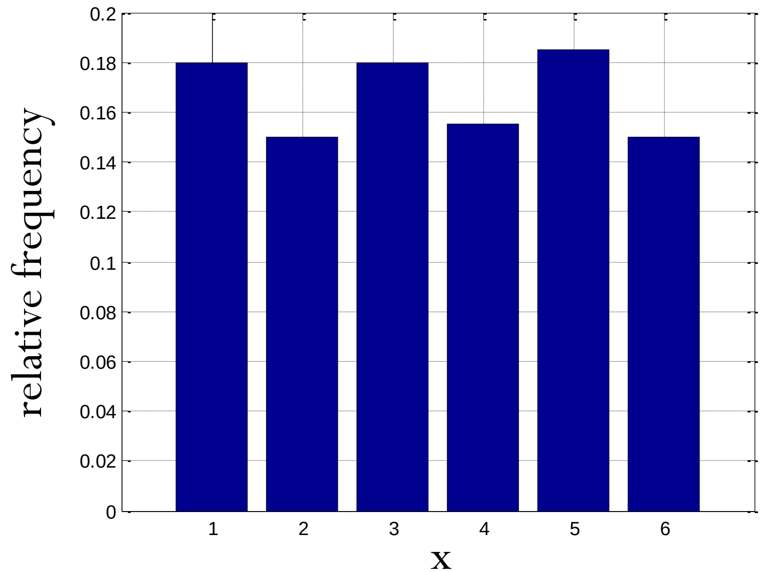
`histc` vs `hist`

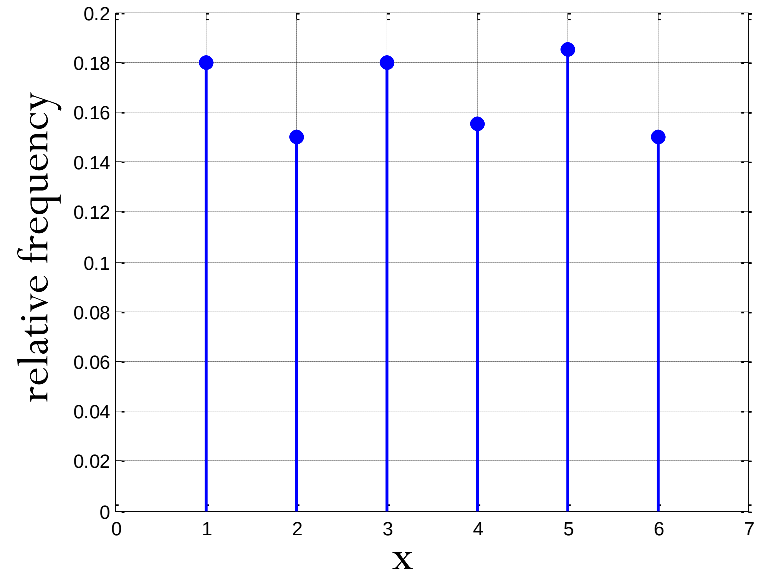
- `N = hist(U,centers)`
  - Bins' centers are defined by the vector `centers`.
    - The first bin includes data between -inf and the first center and the last bin includes data between the last bin and inf.
  - `N(k)` count the number of entries of vector `U` whose values falls inside the *k*th bin.
- `N = histc(U,edges)`
  - Bins' edges are defined by the vector `edges`.
  - `N(k)` count the value `U(i)` if `edges(k) ≤ U(i) < edges(k+1)`.
  - The last (additional) bin will count any values of `U` that match `edges(end)`.
  - Values outside the values in `edges` are not counted.
  - May use `-inf` and `inf` in `edges`.
- `[N,BIN_IND] = histc(U,EDGES)` also returns vector `BIN_IND` indicating the bin index that each entry in `U` sorts into.

Example: histc

```
>> p_X = [1/6 1/3 1/2];
>> F_X = cumsum(p_X)

F_X =

    0.1667    0.5000    1.0000

>> U = rand(1,5)

U =

    0.2426    0.9179    0.9409    0.1026    0.8897

>> [dum,V] = histc(U,[0 F_X])

dum =

     1     1     3     0


V =

     2     3     3     1     3
```

Relative Frequency

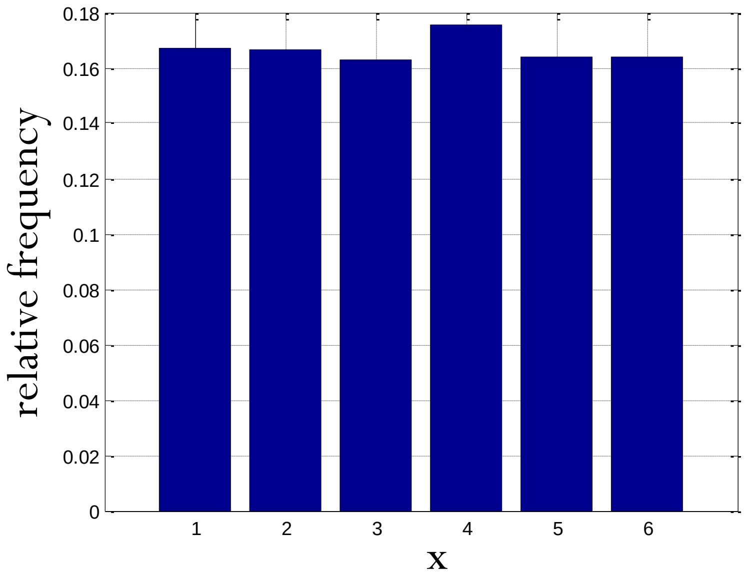


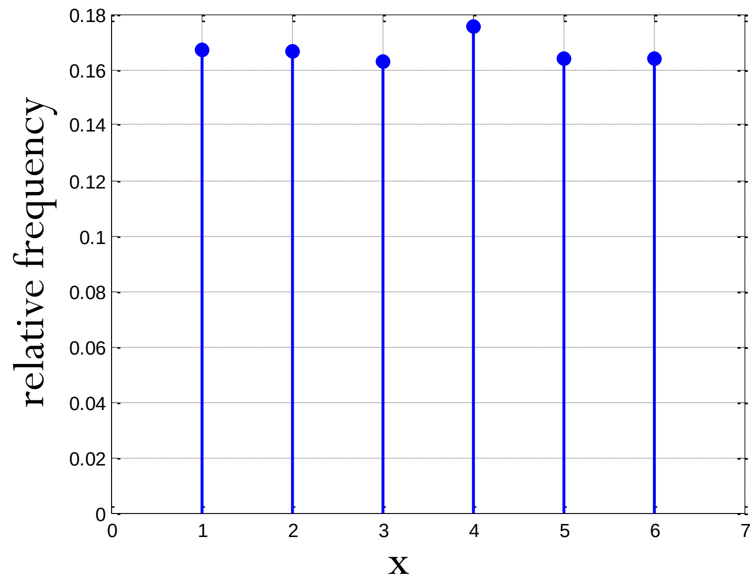



```
rf = N/prod(size(X))
bar(x,rf)
grid on
```

```
stem(x,rf,'filled','LineWidth',1.5)
grid on
```

With larger number of samples





rf = N/prod(size(X))
bar(x,rf)
grid on

stem(x,rf,'filled','LineWidth',1.5)
grid on

X = randi(6,100,100);

20-Sided Dice

Dice in Dungeons & Dragons

- A fantasy tabletop role-playing game (RPG)
- First published in 1974
- Widely regarded as the beginning of modern role-playing games and the role-playing game industry

D&D uses polyhedral dice to resolve random events. These are abbreviated by a 'd' followed by the number of sides. Shown counter-clockwise from the bottom are: d4, d6, d8, d10, d12 and d20 dice.

DUNGEONS & DRAGONS®

http://en.wikipedia.org/wiki/File:Dice_(typical_role_playing_game_dice).jpg
http://en.wikipedia.org/wiki/Dungeons_%26_Dragons

D20 Bowl Set

[http://geekologie.com/2014/09/dungeons-dragons-approved-d20-serving-bo.php]

Flip an unfair coin 10 times. (The probability of getting heads for each time is 0.3.)
Count the number of heads.

n p

X ~ binomial(10,0.3)

```
>> X = binornd(10,0.3)

X =

     3
```

Again, flip an unfair coin 10 times. Count #H.

```
>> X = binornd(10,0.3)

X =

     2
```

Again, flip an unfair coin 10 times. Count #H.

```
>> X = binornd(10,0.3)

X =

     2
```

Again, flip an unfair coin 10 times. Count #H.

```
>> X = binornd(10,0.3)

X =

     5
```

Again, flip an unfair coin 10 times. Count #H.

```
>> X = binornd(10,0.3)

X =

     1
```

Again, flip an unfair coin 10 times. Count #H.

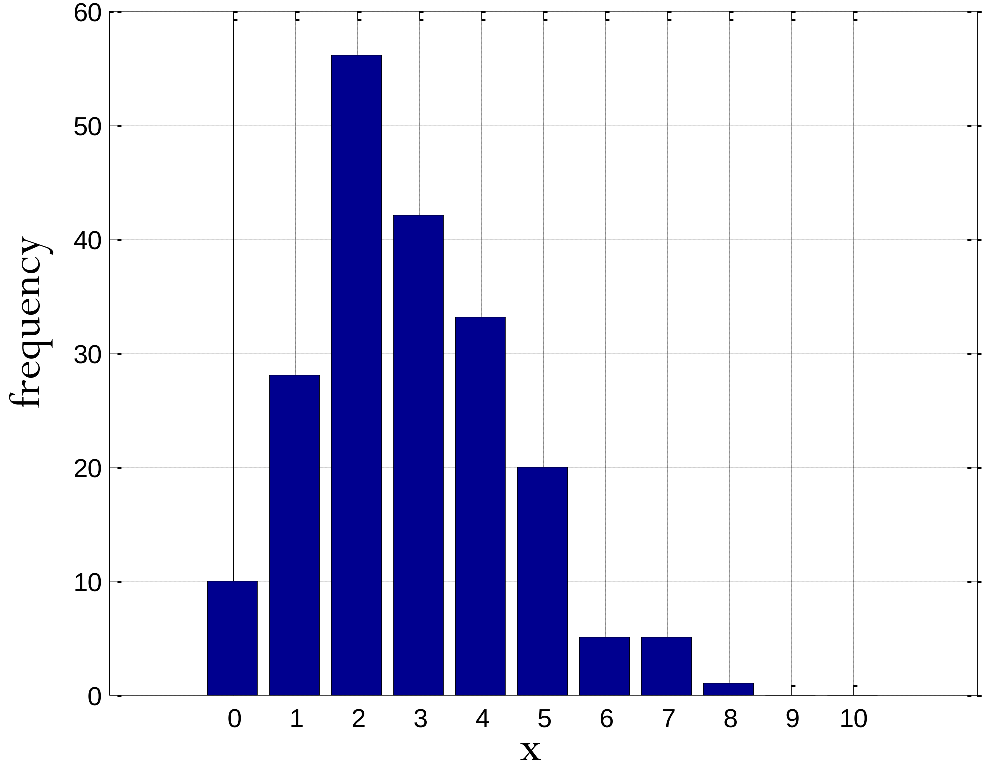
```
>> X = binornd(10,0.3)

X =

     4
```

```
>> X = binornd(10,0.3,20,10)

X =
```

Generate X 200 times. Put the results in a table of size 20×10

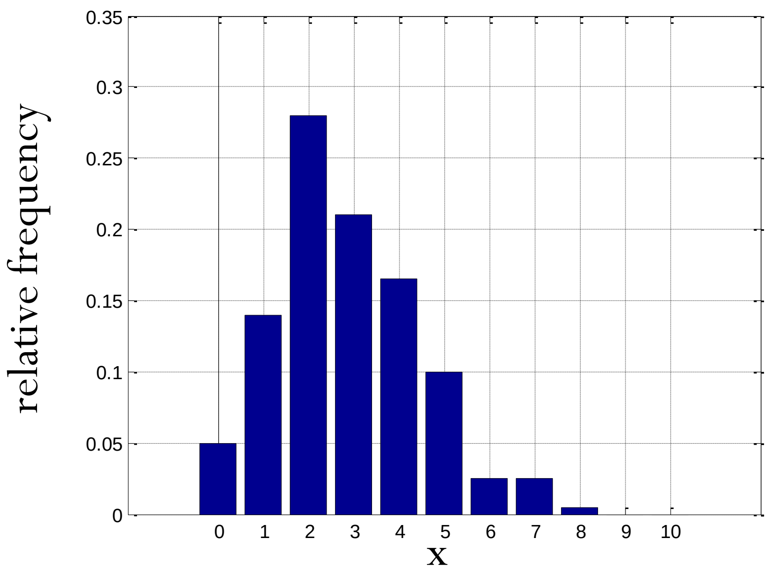
3	4	4	5	7	2	2	2	2	1
3	5	3	1	0	4	2	1	2	3
5	2	2	6	4	2	2	4	3	1
1	2	2	4	2	4	3	3	3	5
4	1	4	3	3	4	2	2	2	2
2	1	3	1	5	2	5	2	1	2
4	0	3	3	2	1	2	1	3	1
4	4	0	2	3	6	2	3	1	1
5	0	3	3	7	1	3	1	3	8
1	2	4	4	1	5	2	4	5	1
5	2	4	6	3	2	3	3	5	0
2	4	0	0	2	2	3	2	0	2
4	3	3	2	2	2	1	2	7	4
2	4	2	1	3	3	4	3	5	2
5	3	2	3	4	2	3	3	1	2
2	6	2	3	4	4	4	5	6	7
5	1	2	4	3	3	0	5	0	2
1	4	1	3	1	4	2	4	2	4
5	2	2	3	3	5	3	5	2	1
4	2	4	3	2	5	7	2	3	1

Histogram: X ~ binomial(10,0.3)

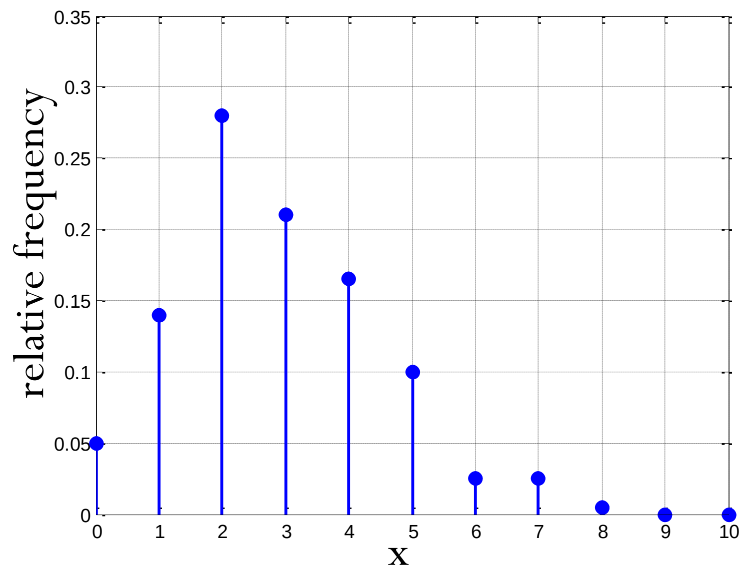


[N, x] = hist(reshape(X,1,prod(size(X))),0:10)
bar(x,N)
Grid on

Relative Freq.: X ~ binomial(10,0.3)



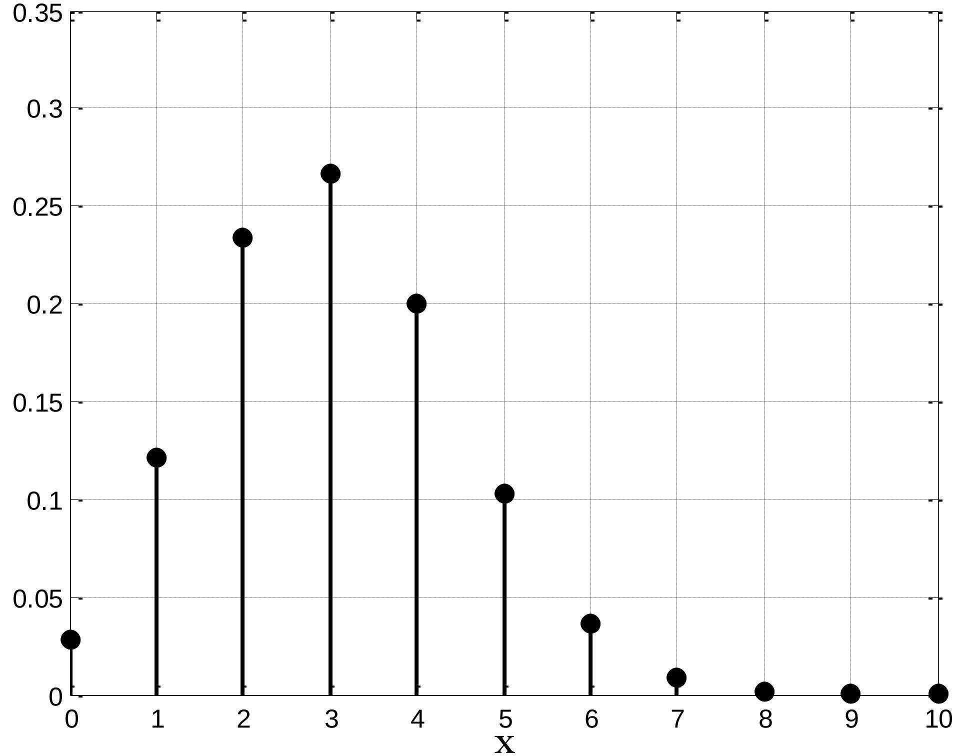
rf = N/prod(size(X))
bar(x,rf)
grid on



stem(x,rf,'filled','LineWidth',1.5)
grid on

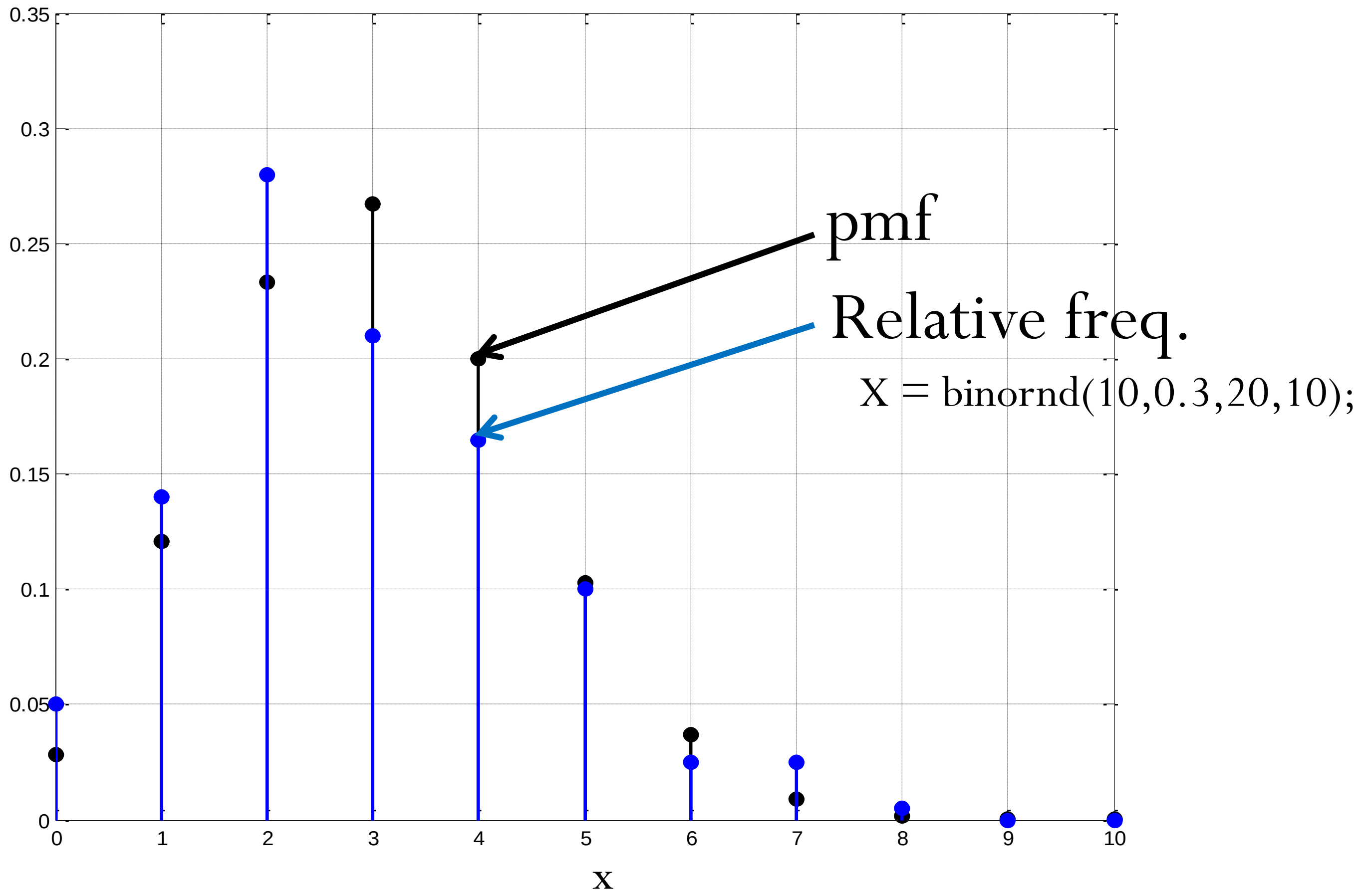
pmf for X ~ binomial(10,0.3)

$$p_X(x) = \binom{10}{x} 0.3^x (1 - 0.3)^{10-x}$$

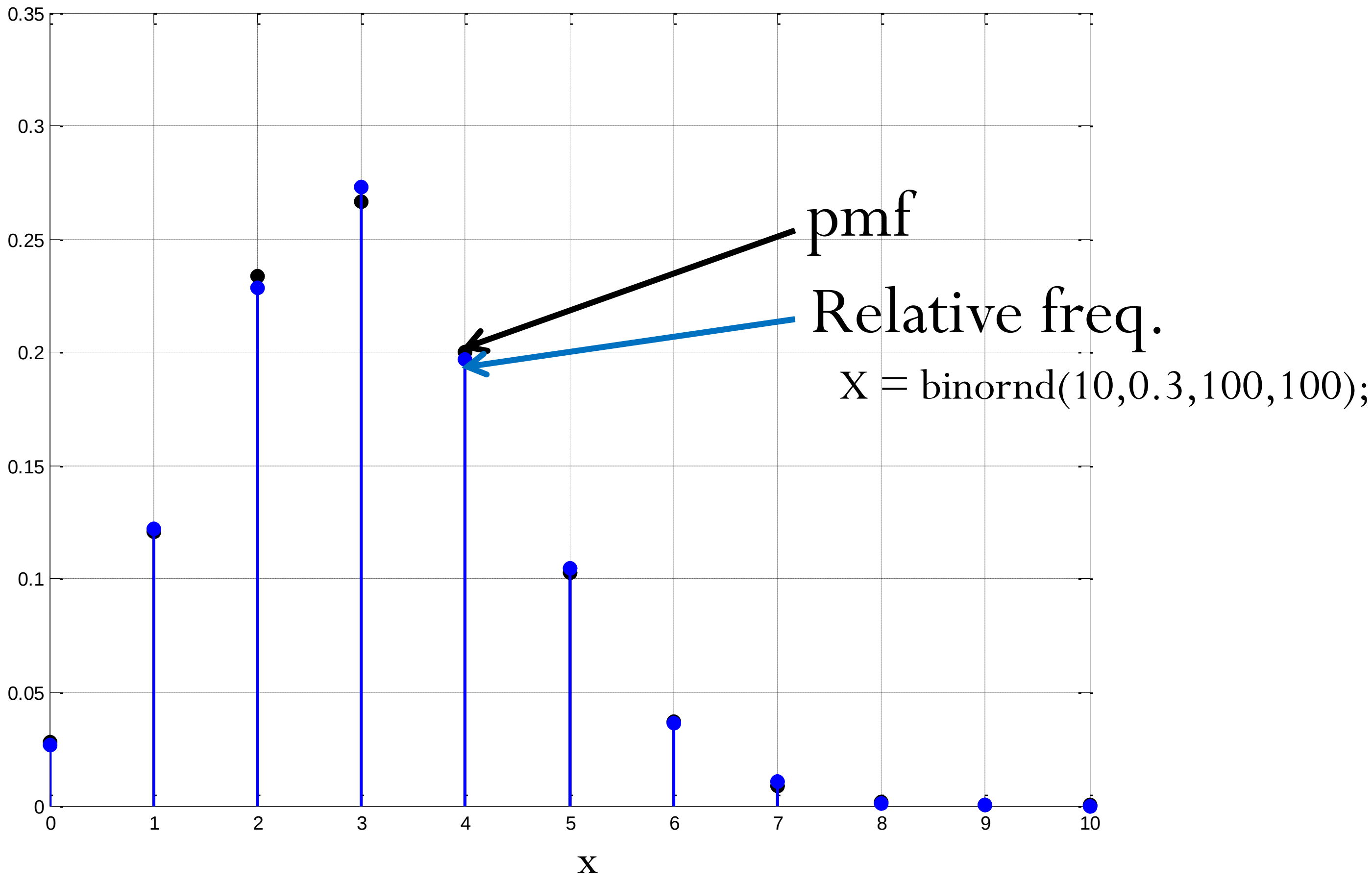


p = binopdf(x,10,0.3)
stem(x,p,'k','filled','LineWidth',1.5); grid on

X ~ binomial(10,0.3)



X ~ binomial(10,0.3)



Bernoulli Trials

$0 1 0 0 0 1 0 1 1 0 0 0 1 0 1 1 1 \underbrace{0 0 0 0 1 0 1 0 1 1 1 0 0} \ldots$

The number of trials until the next 1 is a geometric$_1$ random variable.

The number of 0 until the next 1 is a geometric$_0$ random variable.

The number of 1s in n trials is a binomial random variable with parameter (n,p)

In the limit, as $n \to \infty$ and $p \to 0$ while $np = \alpha$,

The number of 1s is a Poisson random variable with parameter $\alpha = np$.

Publishing Success

- **Publishing success** is so **unpredictable** that even if our novel is destined for the best-seller list, numerous publishers could miss the point and send those letters that say thanks but no thanks.
- In fact, many books destined for great success had to survive not just rejection, but repeated rejection.
- J. K. **Rowling**'s first **Harry Potter** manuscript was rejected by **nine** publishers.
- Lesson: Suppose four publishers have rejected your manuscript.
  - Your intuition and the bad feeling in the pit of your stomach might say that the rejections by all those publishing experts mean your manuscript is no good.
  - We all know from experience that if several tosses of a coin come up heads, it doesn't mean we are tossing a two-headed coin.

Box Office Success

- Hollywood's unpredictability
- Does luck play a far more important role in box office success (and failure) than people imagine?
- There are reasons for a film's box office performance
  - but those reasons are so complex and the path from green light to opening weekend so vulnerable to unforeseeable and uncontrollable influences that
  - educated guesses about an unmade film's potential aren't much better than flips of a coin.
- Studio executive David Picker:
  - "If I had said yes to all the projects I turned down, and no to all the other ones I took, it would have worked out about the same."

Don’t give up

Successful people in every field are almost universally members of a certain set—the set of people who don’t give up.

Bernoulli Trials

$0100010110001011100001010111 00\ldots$

The number of trials until the next 1 is a geometric$_1$ random variable.

The number of 0 until the next 1 is a geometric$_0$ random variable.

The number of 1s in n trials is a binomial random variable with parameter (n,p)

In the limit, as $n \to \infty$ and $p \to 0$ while $np = \alpha$,

The number of 1s is a Poisson random variable with parameter $\alpha = np$.

Poisson Approximation

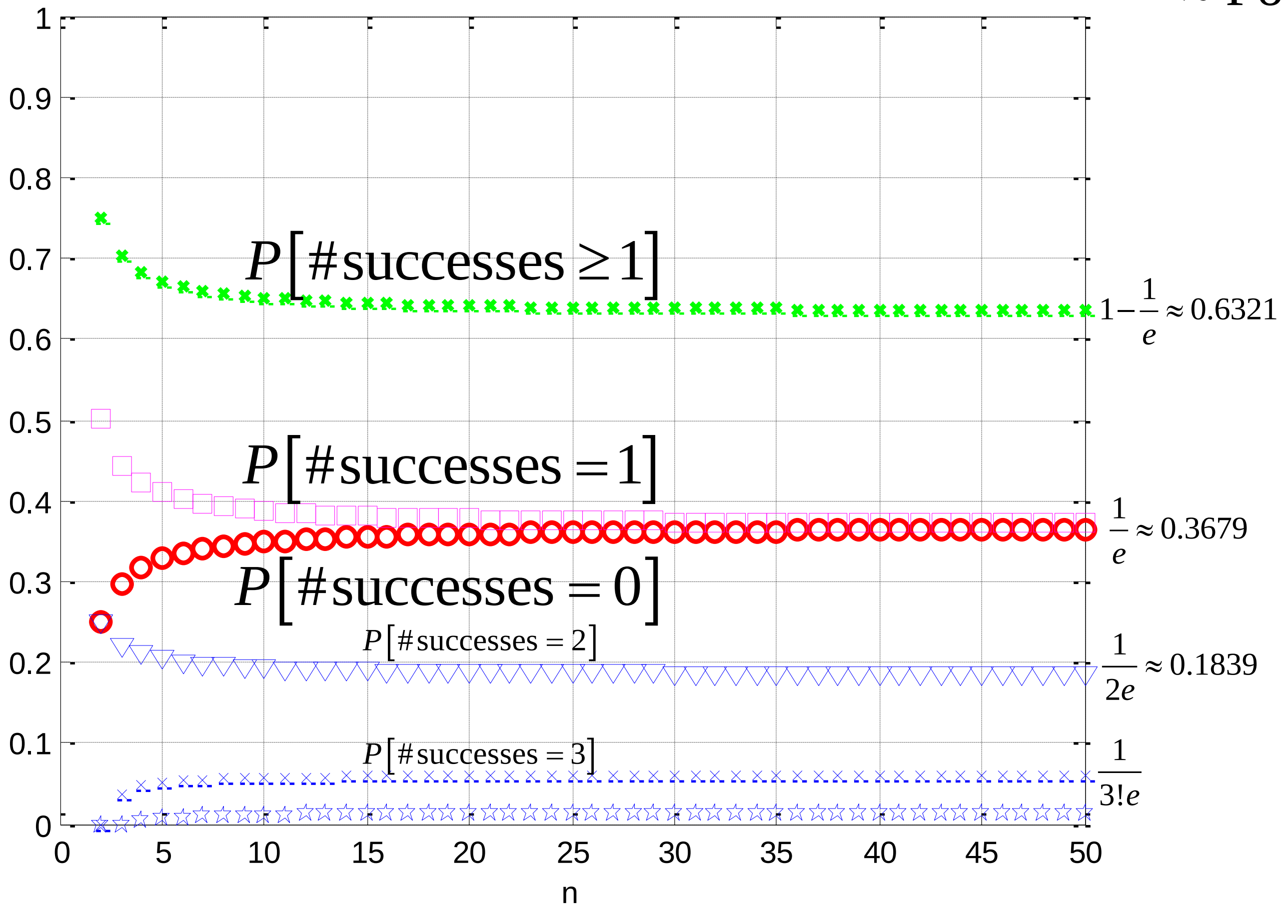
- Consider n Bernoulli trials.
- Assume success probability for each trial is 1/n.

$$\#\text{successes} \sim \text{binomial}\left(n, \frac{1}{n}\right)$$

when n is large

$$\approx \text{Poisson}(1)$$

$$e^{-\alpha}\frac{\alpha^k}{k!} \overset{\alpha=1}{=} \frac{1}{k!e}$$



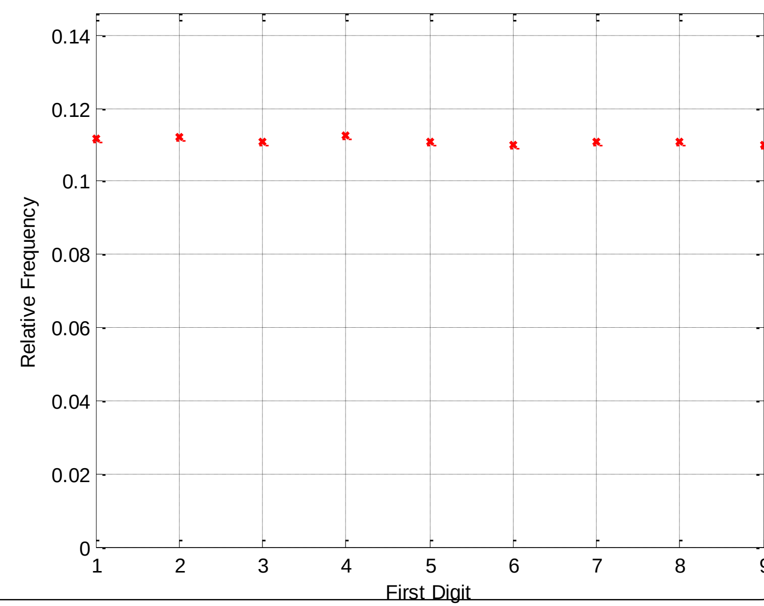
Benford's law: Introduction

For example, the first digit of 110,364 is a 1.

- Consider the distribution of the **first digit** in real-life sources of data.
- Suppose you start reading through a particular issue of a publication like the New York Times or The Economist, and each time you encounter any number (the amount of donations to a particular political candidate, the age of an actor, the number of members of a union, and so on), you record the first digit of that number. Possible first digits are 1, 2, 3, … , or 9. In the long run, how frequently do you think each of these nine possible first digits will be encountered?
- It might be quite natural to assume that all digits are equally likely to show up in most random data sets.

```
X = randi(1e6,1e5,1);
```

560447
845196
901480
639879
449454
41875
365825
41551
976613
706264
164932
88515
452648
820554



Benford's law: Introduction

- One of the following columns contains the value of the closing stock index as of Aug. 8, 2012 for each of a number of countries, and the other column contains fake data obtained with a random number generator.
- Just by looking at the numbers, without considering context, can you tell which column is fake and which is real?

China	2264	3058
Japan	8881	9546
Britain	5846	7140
Canada	11,781	6519
Euro area	797	511
Austria	2053	4995
France	3438	2097
Germany	6966	4628
Italy	14,665	8461
Spain	722	598
Norway	480	1133
Russia	1445	4100
Sweden	1080	2594
Turkey	64,699	35,027
Hong Kong	20,066	42,182
India	17,601	3388
Pakistan	14,744	10,076
Singapore	3052	5227
Thailand	1214	7460
Argentina	2459	2159
⋮	⋮	⋮

Benford's law: Introduction

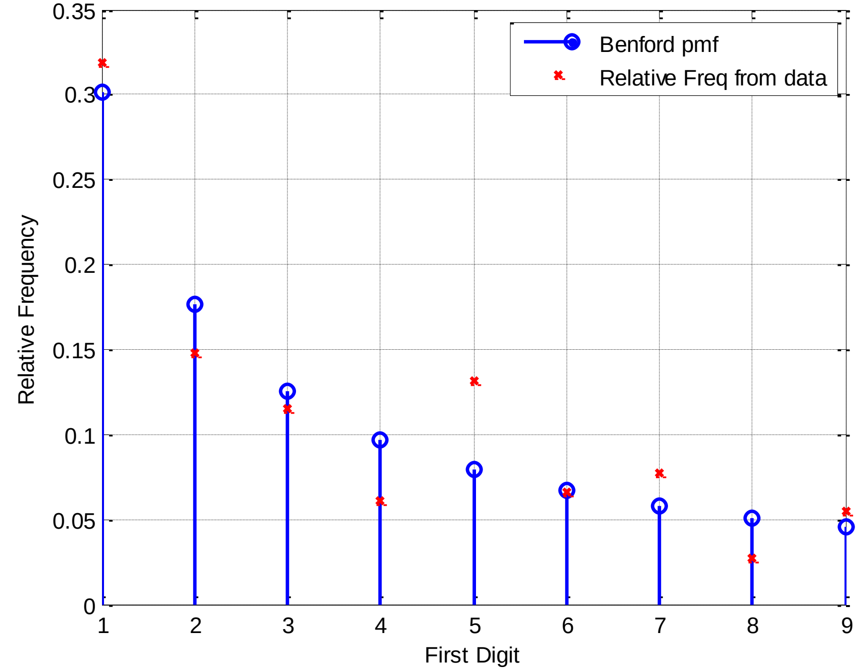
- Examination of the foregoing lists of numbers shows that
  the first column conforms much more closely to Benford's Law than does the second column.
- In fact, the first column is real, whereas the second one is fake.

	real	fake
China	2264	3058
Japan	8881	9546
Britain	5846	7140
Canada	11,781	6519
Euro area	797	511
Austria	2053	4995
France	3438	2097
Germany	6966	4628
Italy	14,665	8461
Spain	722	598
Norway	480	1133
Russia	1445	4100
Sweden	1080	2594
Turkey	64,699	35,027
Hong Kong	20,066	42,182
India	17,601	3388
Pakistan	14,744	10,076
Singapore	3052	5227
Thailand	1214	7460
Argentina	2459	2159
⋮	⋮	⋮

http://www.census.gov/population/international/data/countryrank/rank.php

Countries and Areas Ranked by Population:2013

Rank	Country Or Area	Population
1	China	1,349,585,838
2	India	1,220,800,359
3	United States	316,438,601
4	Indonesia	251,160,124
5	Brazil	201,009,622
6	Pakistan	193,238,868
7	Nigeria	172,831,537
8	Bangladesh	163,654,860
9	Russia	142,500,482
10	Japan	127,253,075
11	Mexico	118,818,228
12	Philippines	105,720,644
13	Ethiopia	93,877,025
14	Vietnam	92,477,857
15	Egypt	85,294,388
16	Germany	81,147,265
17	Turkey	80,694,485
18	Iran	79,853,900



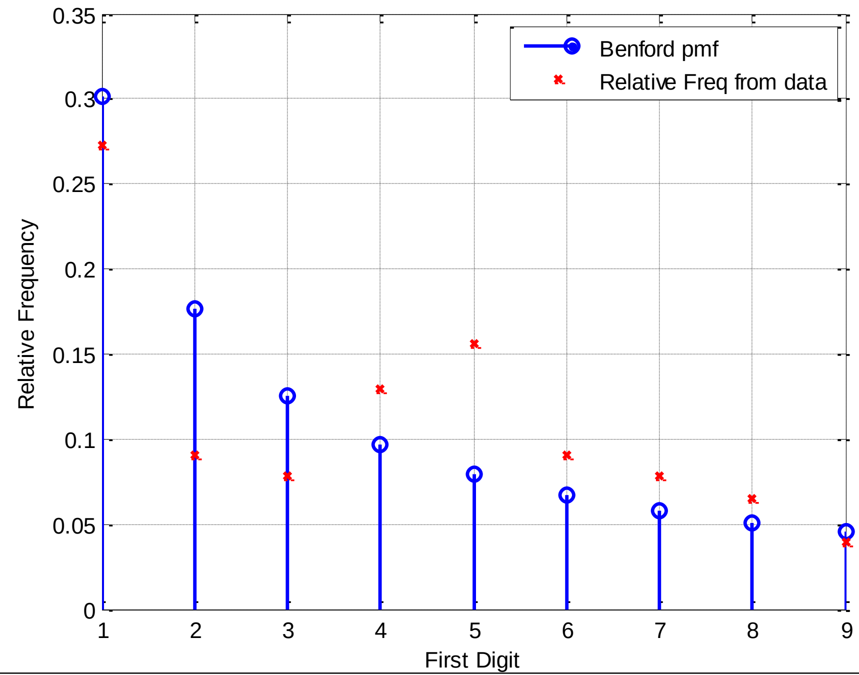
http://stat.bora.dopa.go.th/stat/y_stat55.html

ประกาศสำนักทะเบียนกลาง กรมการปกครอง

เรื่อง จำนวนราษฎรทั่วราชอาณาจักร แยกเป็นกรุงเทพมหานครและจังหวัดต่าง ๆ
ตามหลักฐานการทะเบียนราษฎร ณ วันที่ ๓๑ ธันวาคม ๒๕๕๕

อาศัยอำนาจตามความในมาตรา ๔๕ แห่งพระราชบัญญัติการทะเบียนราษฎร พ.ศ. ๒๕๓๔ จึงประกาศจำนวนราษฎรทั่วราชอาณาจักร แยกเป็นกรุงเทพมหานครและจังหวัดต่าง ๆ ตามหลักฐานการทะเบียนราษฎร ณ วันที่ ๓๑ ธันวาคม ๒๕๕๕ ดังต่อไปนี้

ลำดับ	จังหวัด	จำนวนราษฎร		
		ชาย	หญิง	รวม
	ทั่วประเทศ	๓๑,๗๐๐,๗๒๗	๓๒,๗๕๕,๙๖๘	๖๔,๔๕๖,๖๙๕
๑	กรุงเทพมหานคร	๒,๖๙๐,๗๕๔	๒,๙๘๒,๘๐๖	๕,๖๗๓,๕๖๐
๒	จังหวัดกระบี่	๒๒๑,๙๐๖	๒๒๓,๐๖๑	๔๔๔,๙๖๗
๓	จังหวัดกาญจนบุรี	๔๒๐,๔๖๗	๔๑๗,๘๐๒	๘๓๘,๒๖๙
๔	จังหวัดกาฬสินธุ์	๔๙๐,๖๑๗	๔๙๔,๔๖๗	๙๘๕,๐๘๔
๕	จังหวัดกำแพงเพชร	๓๖๑,๕๑๑	๓๖๖,๐๔๔	๗๒๗,๕๕๕
๖	จังหวัดขอนแก่น	๘๗๙,๓๖๐	๘๙๕,๔๕๖	๑,๗๗๔,๘๑๖
๗	จังหวัดจันทบุรี	๒๕๖,๖๙๐	๒๖๕,๑๒๒	๕๒๑,๘๑๒
๘	จังหวัดฉะเชิงเทรา	๓๓๕,๙๘๓	๓๔๙,๗๓๘	๖๘๕,๗๒๑
๙	จังหวัดชลบุรี	๖๖๘,๗๔๔	๖๙๕,๒๕๘	๑,๓๖๔,๐๐๒
๑๐	จังหวัดชัยนาท	๑๖๐,๗๖๙	๑๗๒,๔๐๓	๓๓๓,๑๗๒
๑๑	จังหวัดชัยภูมิ	๕๖๓,๓๒๓	๕๖๙,๗๑๑	๑,๑๓๓,๐๓๔
๑๒	จังหวัดชุมพร	๒๔๕,๙๗๓	๒๔๙,๓๓๗	๔๙๕,๓๑๐
๑๓	จังหวัดเชียงราย	๕๙๐,๔๔๖	๖๐๙,๙๗๗	๑,๒๐๐,๔๒๓
๑๔	จังหวัดเชียงใหม่	๘๐๖,๗๒๐	๘๔๘,๙๒๒	๑,๖๕๕,๖๔๒
๑๕	จังหวัดตรัง	๓๐๙,๗๒๙	๓๒๒,๑๙๑	๖๓๑,๙๒๐



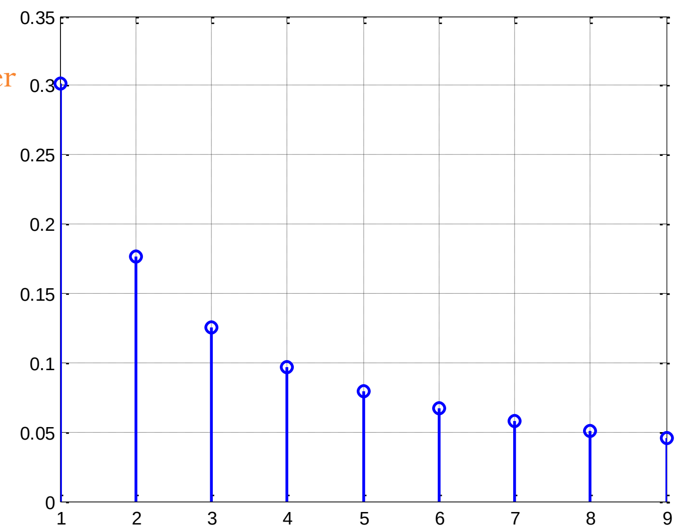
Benford's law

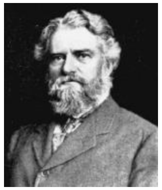
Zero is inadmissible as a first digit. which means that there are nine possible first digits (1, 2, . . . , 9).
The signs of negative numbers are ignored.

- The distribution of the **first digit** in many (but not all) real-life sources of data.

1 is the most likely first digit with a probability of about 30% rather than the 11.1% we would get if all nine digits were equally likely.

$$p_X(x) = \begin{cases} \log_{10}\left(1+\dfrac{1}{x}\right), & x = 1, 2, 3, \ldots 9, \\ 0, & \text{otherwise.} \end{cases}$$





- Named after an American physicist Frank Benford, who stated it in 1938, although it had been previously stated by Simon Newcomb in 1881.

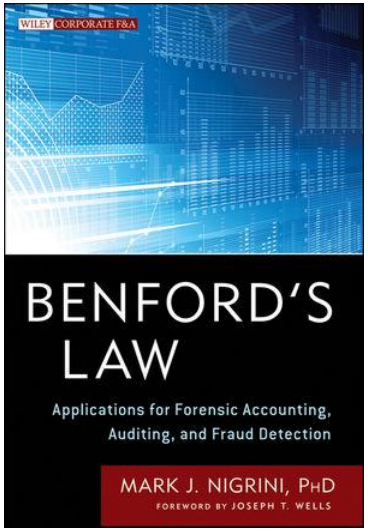
[Benford, "The law of anomalous numbers", Proceedings of the American Philosophical Society, vol. 78, pp. 551–572, 1938.]

- There is a large bias towards the lower digits, so much so that nearly one-half of all numbers are expected to start with the digits 1 or 2.

Benford's law

- Applicable to a wide variety of data sets, including electricity bills, street addresses, stock prices, population sizes, death rates, lengths of rivers, physical and mathematical constants.
- It tends to be most accurate when values are distributed across multiple orders of magnitude.
- Today, Benford's law is routinely applied in several areas in which naturally occurring data arise.
- Perhaps the most practical application of Benford's law is in **detecting fraudulent** data (or unintentional errors) in accounting reports, and in particular to detect fraudulent tax returns.

An application pioneered by Prof. Mark Nigrini (http://www.nigrini.com/).



[http://as.wiley.com/WileyCDA/WileyTitle/productCd-1118152859.html]

Generating Discrete RV in MATLAB

```
clear all; close all;

S_X = [1 2 3 4]; p_X = [1/2 1/4 1/8 1/8]; n = 1e6;

SourceString = randsrc(1,n,[S_X;p_X]);
```

Alternatively, we can also use

```
SourceString = datasample(S_X,n,'Weights',p_X);
```

```
rf = hist(SourceString,S_X)/n; % Ref. Freq. calc.
stem(S_X,rf,'rx','LineWidth',2) % Plot Rel. Freq.
hold on
stem(S_X,p_X,'bo','LineWidth',2) % Plot pmf
xlim([min(S_X)-1,max(S_X)+1])
legend('Rel. freq. from sim.','pmf p_X(x)')
xlabel('x')
grid on
```


[GenRV_Discrete_datasample_Ex.m]

Example

$$\mathcal{S}_X = \{1,2,3,4\}$$

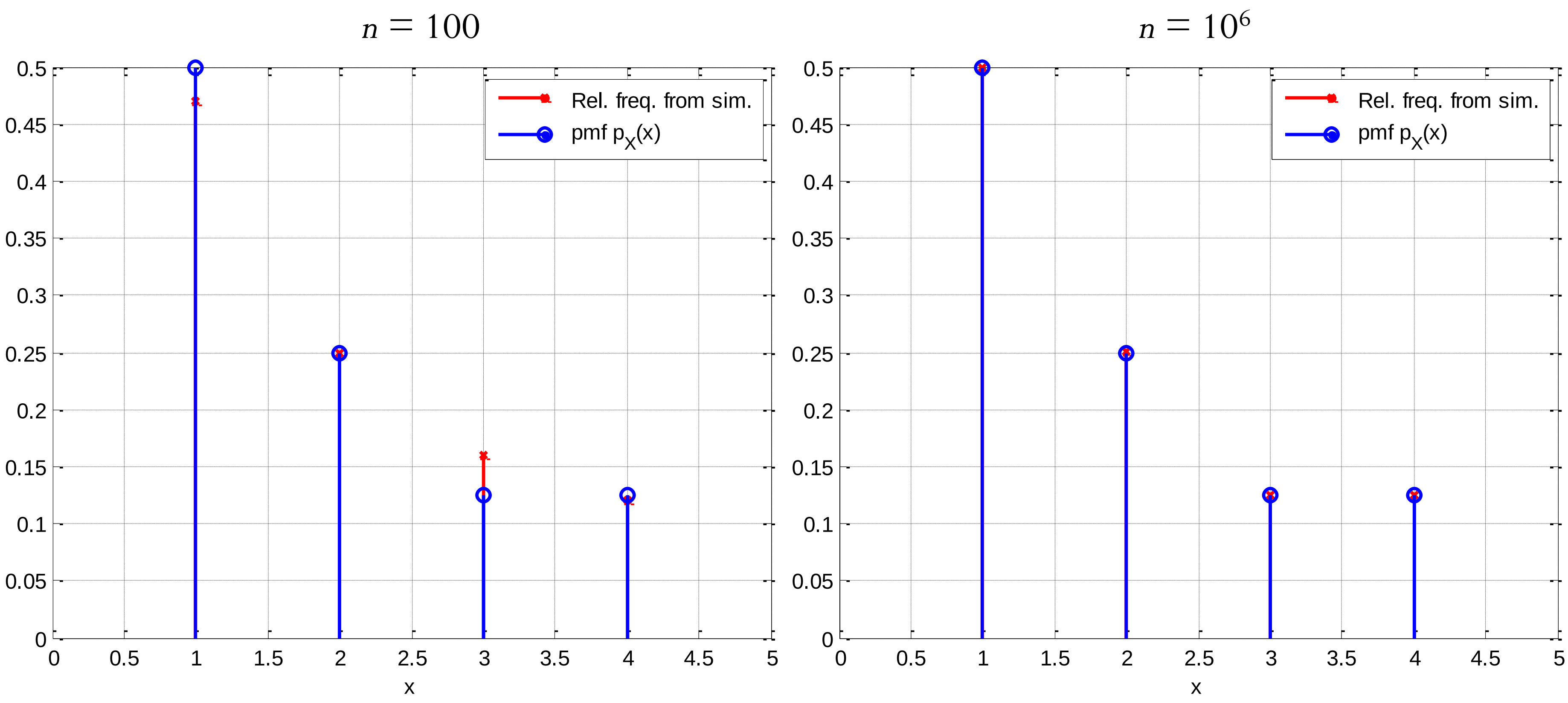
$$p_X(x) = \begin{cases} \frac{1}{2}, & x = 1, \\ \frac{1}{4}, & x = 2, \\ \frac{1}{8}, & x \in \{3,4\} \\ 0, & \text{otherwise} \end{cases}$$

2	1	1	2	1	4	1	1	1	1
1	1	4	1	1	2	4	2	2	1
3	1	1	2	3	2	4	1	2	4
2	1	1	2	1	1	3	3	1	1
1	3	4	1	4	1	1	2	4	1
4	1	4	1	2	2	1	4	2	1
4	1	1	1	1	2	1	4	2	4
2	1	1	1	2	1	2	1	3	2
2	1	1	1	1	1	1	2	3	2
2	1	1	2	1	4	2	1	2	1

Approximately 50% are number ‘1’s

[GenRV_Discrete_finite_support.m]

Example



$n = 100$
$n = 10^6$
Rel. freq. from sim.
pmf $p_X(x)$
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
0.5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
x