

# Understanding Probability Distributions

Nik Bear Brown

## Lesson 2: Understanding probability distributions

In the second lesson, we study the theory behind probability distributions, variables, random variables, and learn some notation. We then use R to graph, and identify several common probability distributions.

### Additional packages needed

To run the code in M01\_Lesson\_02.Rmd you may need additional packages.

- If necessary install ggplot2 package.

```
`install.packages("ggplot2");
```

```
require(ggplot2)
```

```
## Loading required package: ggplot2
```

### Topics

- Probability
- Random variables
- Probability distributions
  - Uniform
  - Normal
  - Binomial
  - Poisson
  - Fat Tailed

### Probability

- Probability is a measure of the likelihood of a random phenomenon or chance behavior. Probability describes the long-term proportion with which a certain outcome will occur in situations with short-term uncertainty.
- Probability is expressed in numbers between 0 and 1. Probability = 0 means the event never happens; probability = 1 means it always happens.
- The total probability of all possible event always sums to 1.

### Sample Space

- Coin Toss = {head, tail}
- Two coins  $S = \{HH, HT, TH, TT\}$

- Inspecting a part = {good, bad}
- Rolling a die  $S = \{1, 2, 3, 4, 5, 6\}$

## Random Variables

In probability and statistics, a random variable, or stochastic variable is a variable whose value is subject to variations due to chance (i.e. it can take on a range of values)

- Coin Toss = {head, tail}
- Rolling a die  $S = \{1, 2, 3, 4, 5, 6\}$

### Discrete Random Variables

- Random variables (RVs) which may take on only a countable number of distinct values  
E.g. the total number of tails  $X$  you get if you flip 100 coins
- $X$  is a RV with arity  $k$  if it can take on exactly one value out of  $\{x_1, \dots, x_k\}$  E.g. the possible values that  $X$  can take on are 0, 1, 2, ..., 100

### Continuous Random Variables

- Probability density function (pdf) instead of probability mass function (pmf)
- A pdf is any function  $f(x)$  that describes the probability density in terms of the input variable  $x$ .

## Probability distributions

- We use probability distributions because they model data in real world.
- They allow us to calculate what to expect and therefore understand what is unusual.
- They also provide insight in to the process in which real world data may have been generated.
- Many machine learning algorithms have assumptions based on certain probability distributions.

### *Cumulative distribution function*

A probability distribution  $\Pr$  on the real line is determined by the probability of a scalar random variable  $X$  being in a half-open interval  $(-\infty, x]$ , the probability distribution is completely characterized by its cumulative distribution function:

$$F(x) = \Pr[X \leq x] \quad \forall \quad x \in \mathbb{R}.$$

## Uniform Distribution

$$X \equiv U[a, b]$$

$$f(x) = \frac{1}{b-a} \quad \text{for } a < x < b$$

$$f(x) = 0 \quad \text{for } a \leq x \quad \text{or} \quad \geq b$$

$$F(x) = \frac{x-a}{b-a} \quad \text{for } a \leq x < b$$

$$F(x) = 0 \quad \text{for } x < a \quad F(x) = 1 \quad \text{for } x \geq b$$

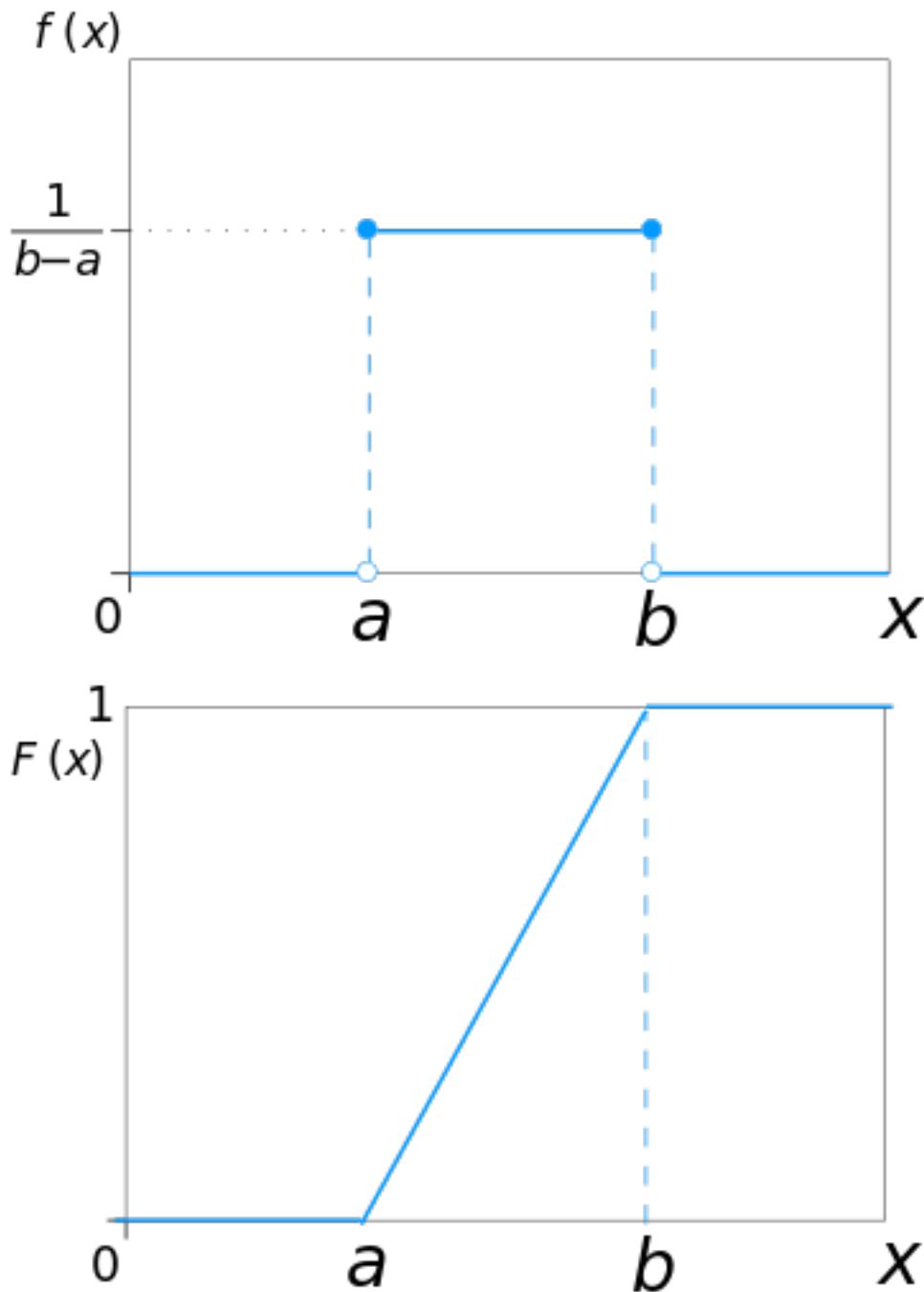


*image Uniform Distribution“*

#### *Continuous Uniform Distribution*

In probability theory and statistics, the continuous uniform distribution or rectangular distribution is a family of symmetric probability distributions such that for each member of the family, all intervals of the same length on the distribution's support are equally probable.

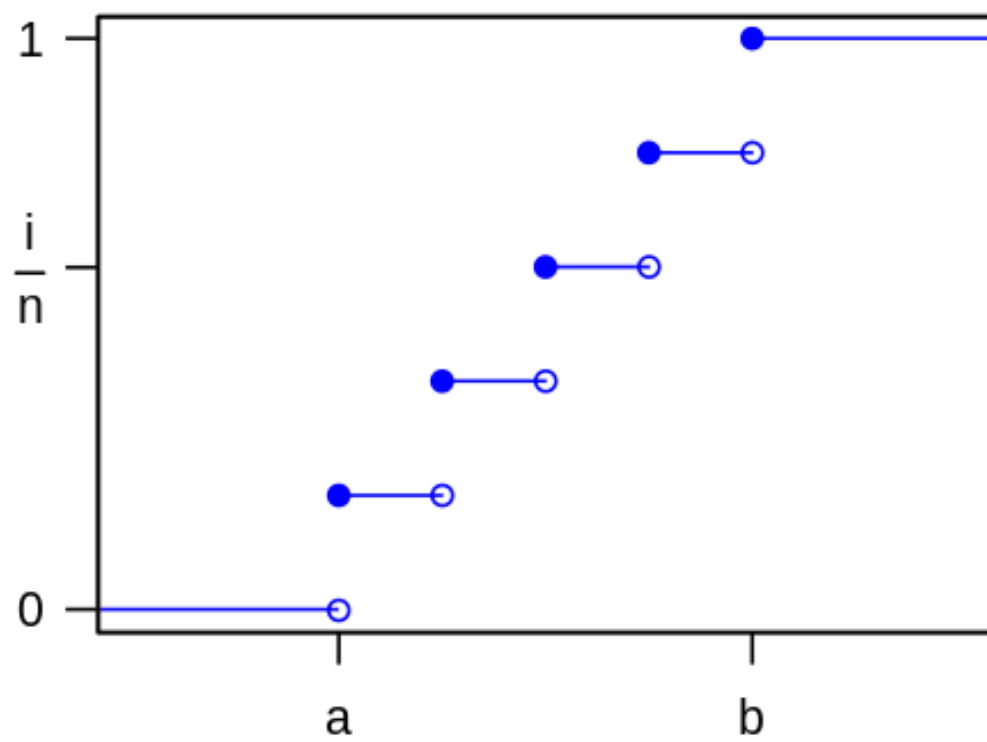
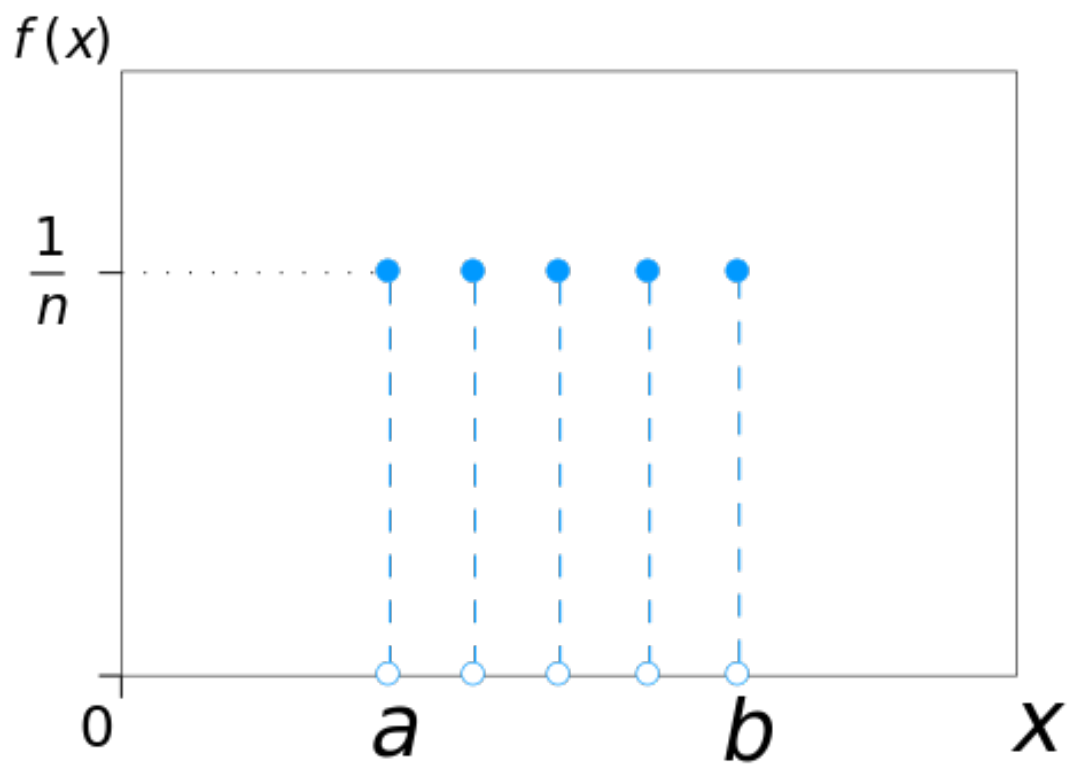
- from [Uniform distribution \(continuous Wikipedia\)](#)



### *Discrete Uniform Distribution*

In probability theory and statistics, the discrete uniform distribution is a symmetric probability distribution whereby a finite number of values are equally likely to be observed; every one of  $n$  values has equal probability  $1/n$ . Another way of saying “discrete uniform distribution” would be “a known, finite number of outcomes equally likely to happen”.

- from [Uniform distribution \(discrete\) Wikipedia](#))



## Uniform Distribution in R

runif function in R. Type ?runif

### *The Uniform Distribution*

#### *Description*

These functions provide information about the uniform distribution on the interval from min to max. dunif gives the density, punif gives the distribution function qunif gives the quantile function and runif generates random deviates.

#### *Usage*

- `dunif(x, min = 0, max = 1, log = FALSE)` //dunif gives the density
- `punif(q, min = 0, max = 1, lower.tail = TRUE, log.p = FALSE)` // punif gives the distribution function
- `qunif(p, min = 0, max = 1, lower.tail = TRUE, log.p = FALSE)` // qunif gives the quantile function
- `runif(n, min = 0, max = 1)` // runif generates random deviates. (a random sample)

```
set.seed(333)
trails <- 3333
min <- 1
max <- 100
uni_dist<-runif(trails,min,max+1)
uni_dist<-data.frame(x=uni_dist)
head(uni_dist)

##           x
## 1 47.700066
## 2  9.459815
## 3 98.348527
## 4 58.130558
## 5  3.011937
## 6 73.355739

str(uni_dist)

## 'data.frame':   3333 obs. of  1 variable:
##  $ x: num  47.7 9.46 98.35 58.13 3.01 ...

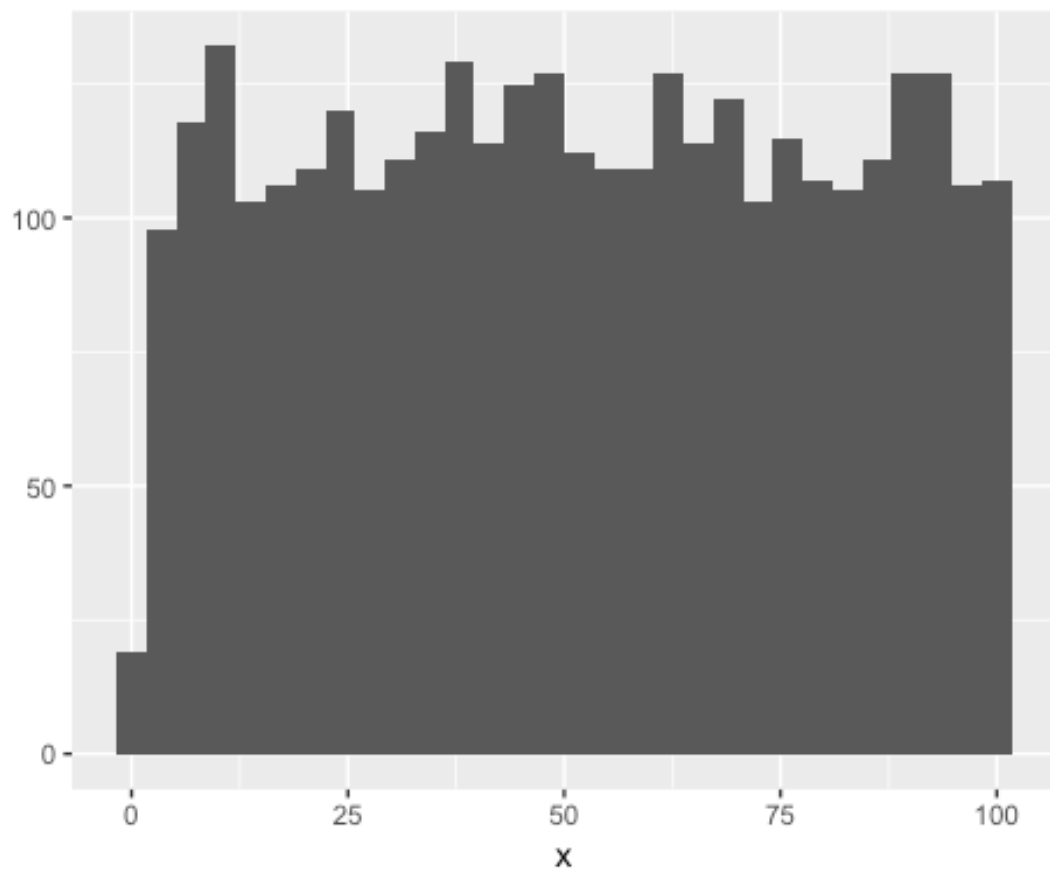
names(uni_dist)

## [1] "x"
```

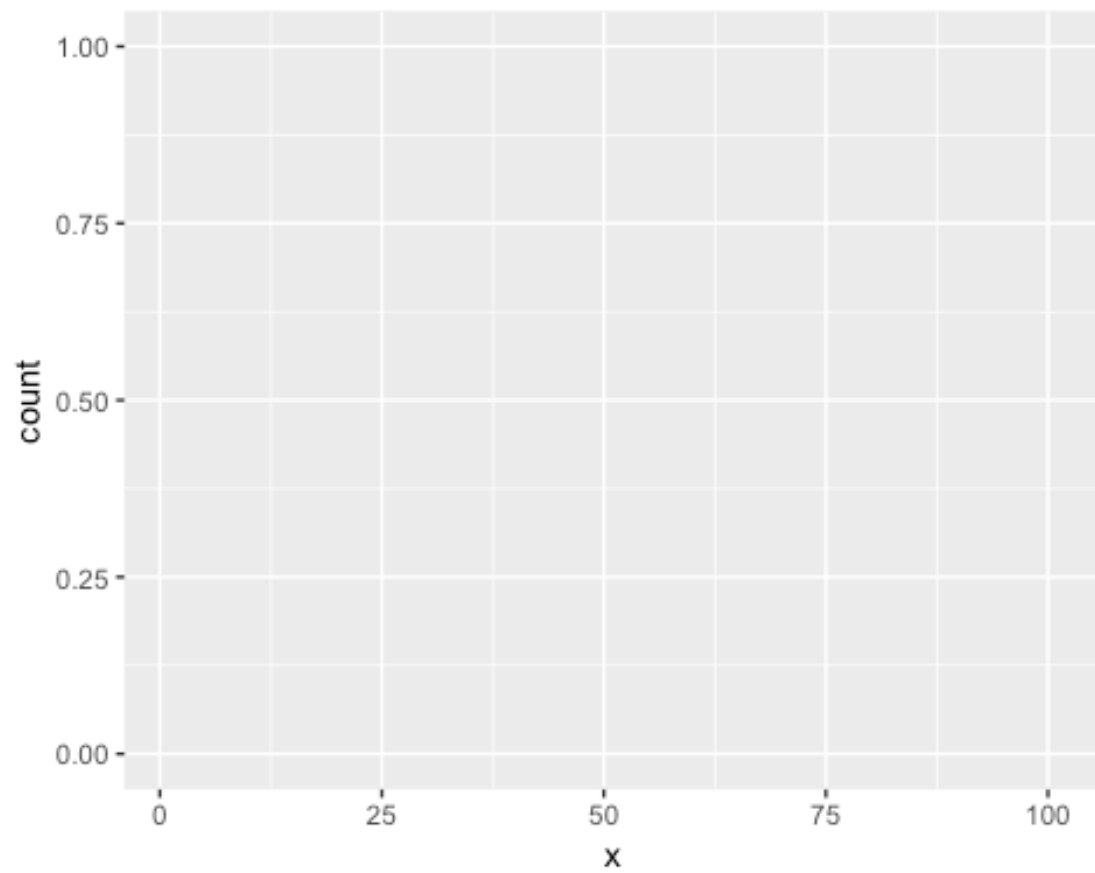
## Plotting a Uniform Distribution

```
qplot(x=x,data=uni_dist)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

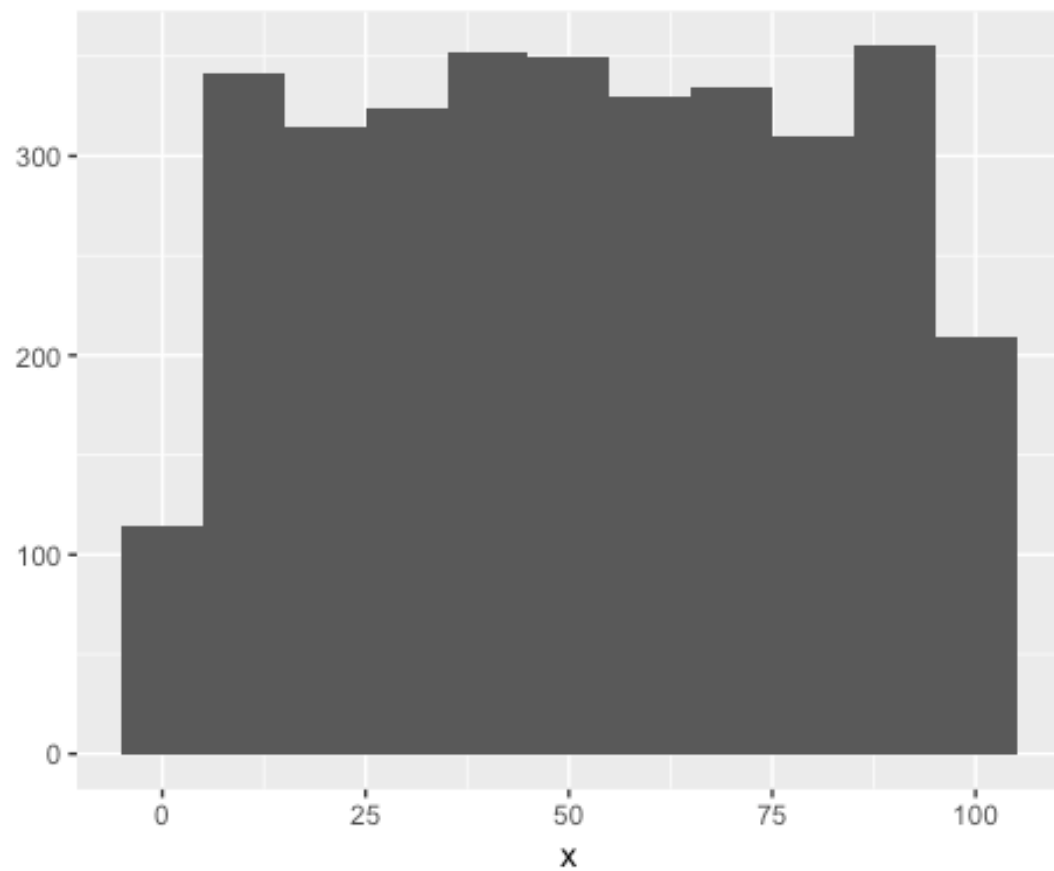


```
ggplot(uni_dist) + aes(x=x) + geom_bar()
```

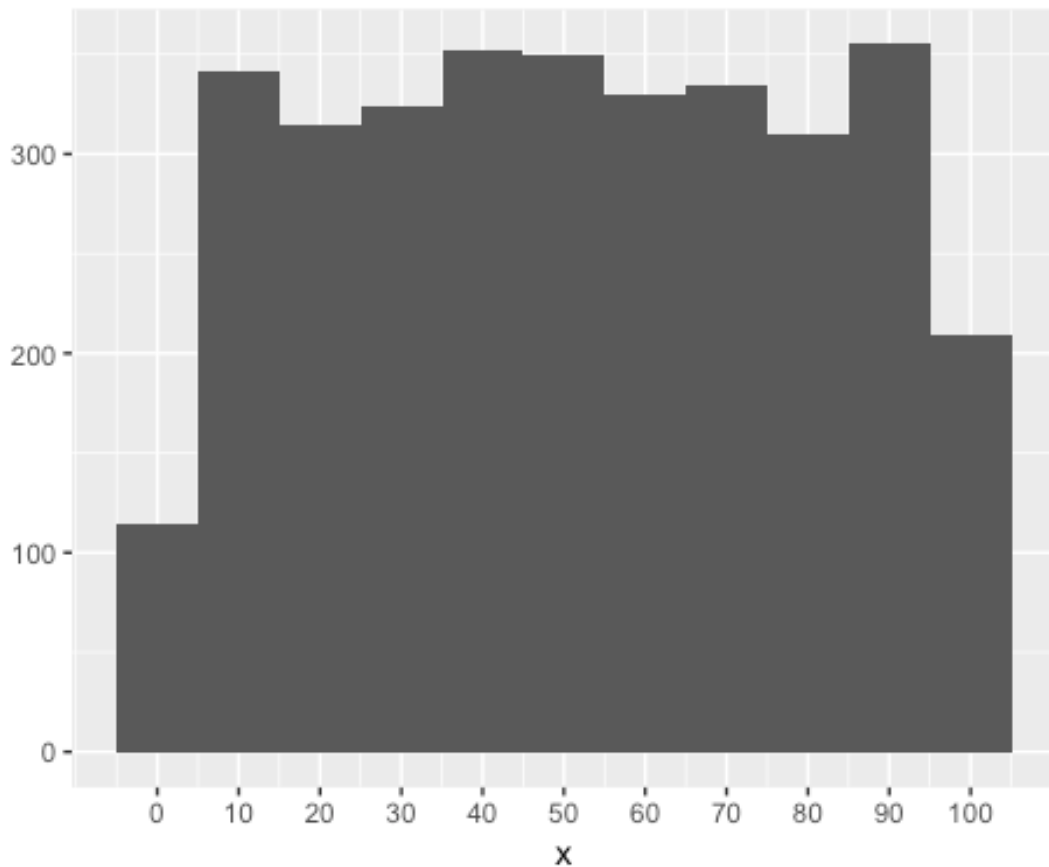


```
qplot(x=x,data=uni_dist,binwidth=10)
```





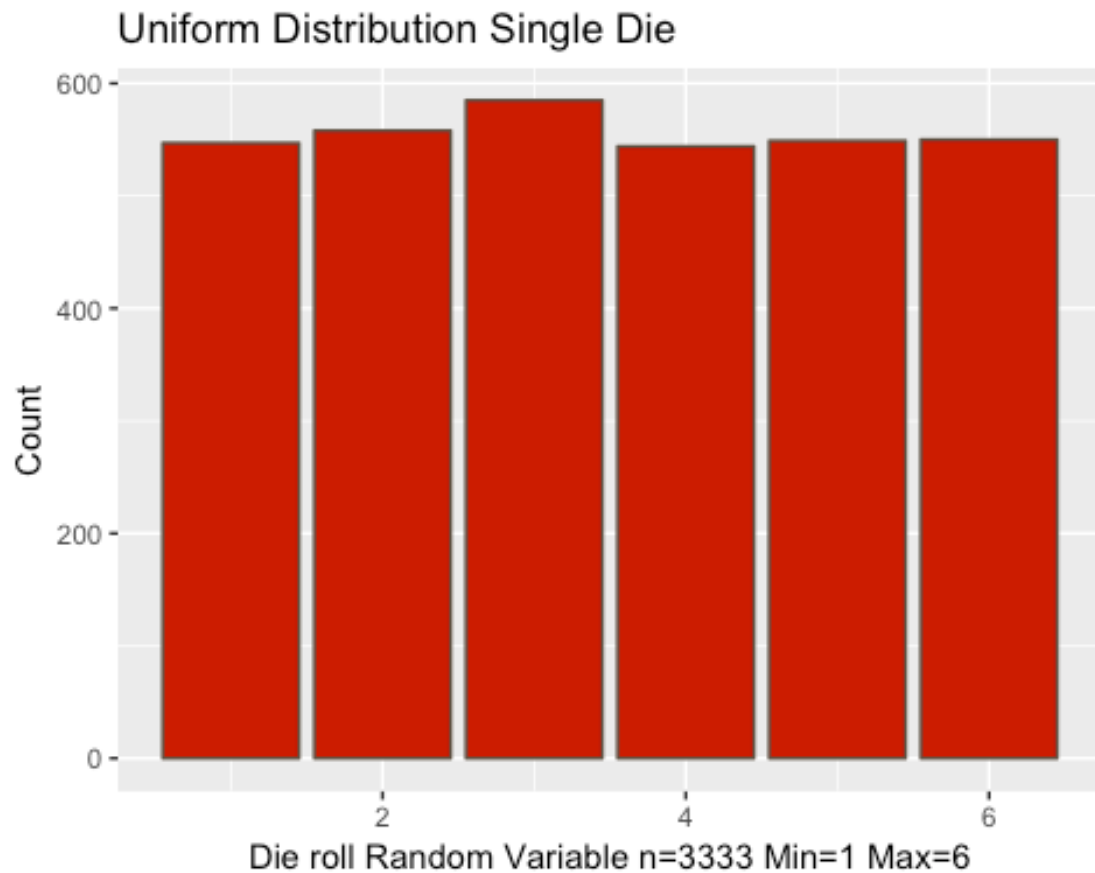
```
qplot(x=x,data=uni_dist,binwidth=10)+  
scale_x_continuous(breaks=seq(0,100,10))
```



```
min <- 1
max <- 6
die <- as.integer(runif(trails,min,max+1))
die<-data.frame(roll=die)
head(die)

##    roll
## 1     1
## 2     4
## 3     6
## 4     3
## 5     2
## 6     1

ggplot(die) + aes(x=roll) + geom_bar(color=I('#615445'), fill=I('#CC0000')) +
labs(title="Uniform Distribution Single Die", y="Count", x="Die roll Random
Variable n=3333 Min=1 Max=6")
```



## Quiz Distribution of two dice

See if you can generate a distribution that models the output that would be generated by the sum of two dice.

## Quiz Distribution of two dice answer

```
dice <- as.integer(runif(trails,min,max+1)+runif(trails,min,max+1))
dice<-data.frame(rolls=dice)
head(dice)
```

```
##    rolls
## 1     11
## 2      6
## 3      7
## 4     10
## 5      3
## 6     12
```

```
summary(dice)
```

```
##      rolls
##  Min.   : 2.000
## 1st Qu.: 6.000
```

```

## Median : 7.000
## Mean   : 7.449
## 3rd Qu.: 9.000
## Max.   :13.000

dice <- as.integer(runif(trails,min,max+0.5)+runif(trails,min,max+0.5))
dice<-data.frame(rolls=dice)
head(dice)

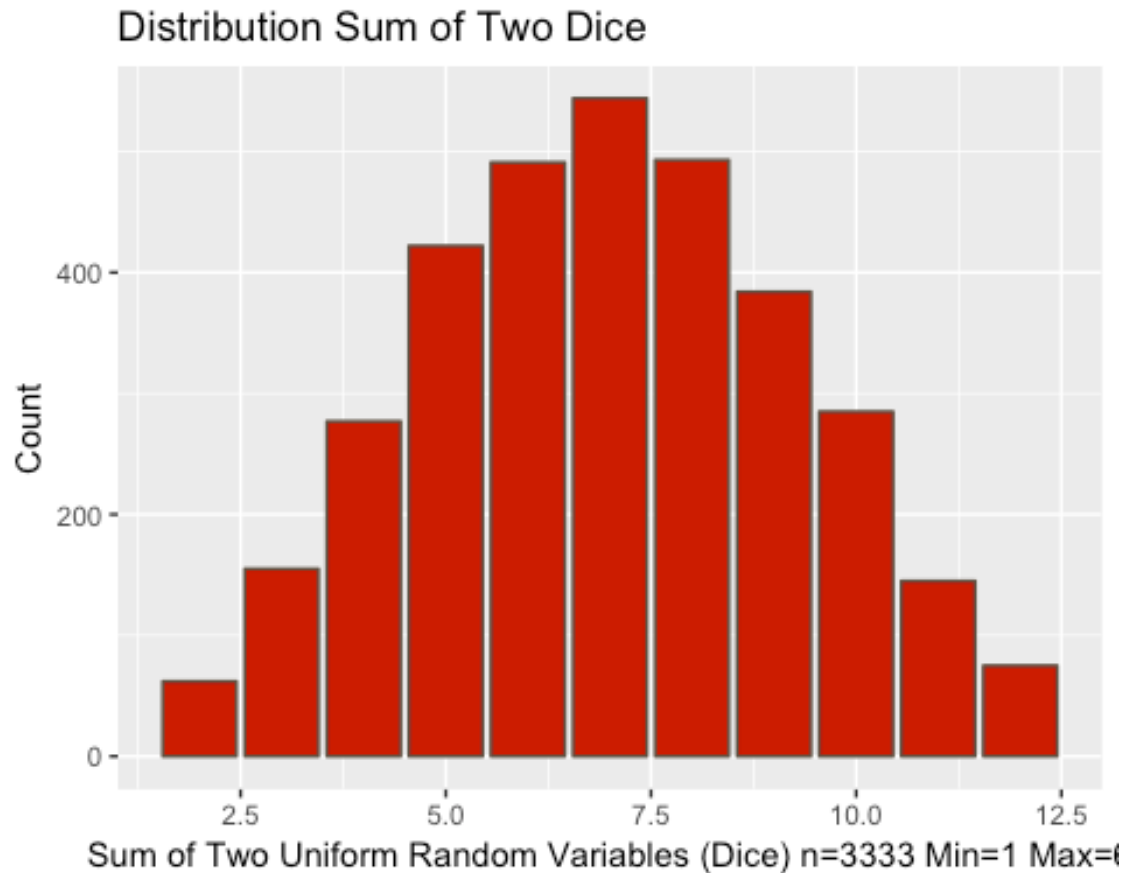
##    rolls
## 1      4
## 2     10
## 3      2
## 4      6
## 5      5
## 6      7

summary(dice)

##      rolls
## Min.     : 2.000
## 1st Qu.: 5.000
## Median : 7.000
## Mean    : 6.992
## 3rd Qu.: 9.000
## Max.     :12.000

ggplot(dice) + aes(x=rolls) + geom_bar(color=I('#615445'), fill=I('#CC0000'))
+ labs(title="Distribution Sum of Two Dice", y="Count", x="Sum of Two
Uniform Random Variables (Dice) n=3333 Min=1 Max=6")

```



## Normal Distribution

In probability theory, the normal (or Gaussian) distribution is a very common continuous probability distribution. The normal distribution is remarkably useful because of the central limit theorem. In its most general form, under mild conditions, it states that averages of random variables independently drawn from independent distributions are normally distributed. Physical quantities that are expected to be the sum of many independent processes (such as measurement errors) often have distributions that are nearly normal.

- from [Normal Distribution - Wikipedia](#))

$$X \sim N(\mu, \sigma^2)$$

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

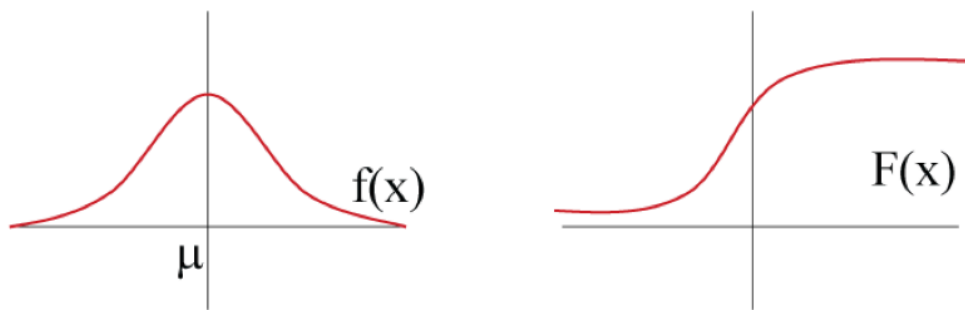


image Normal Distribution“

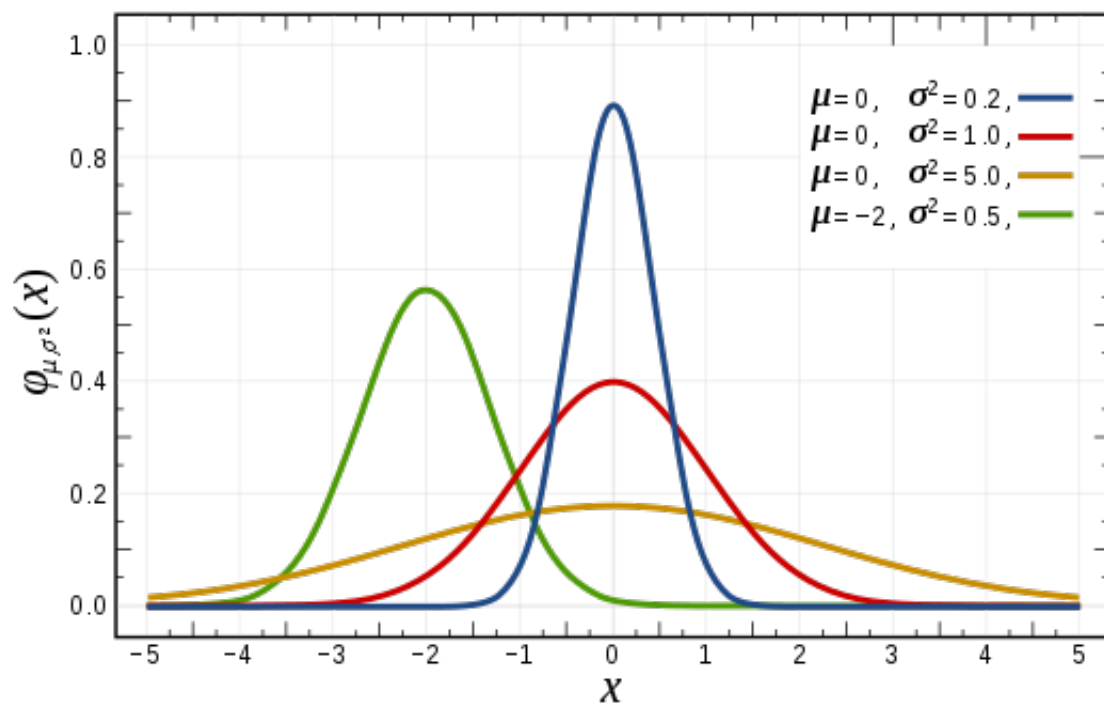
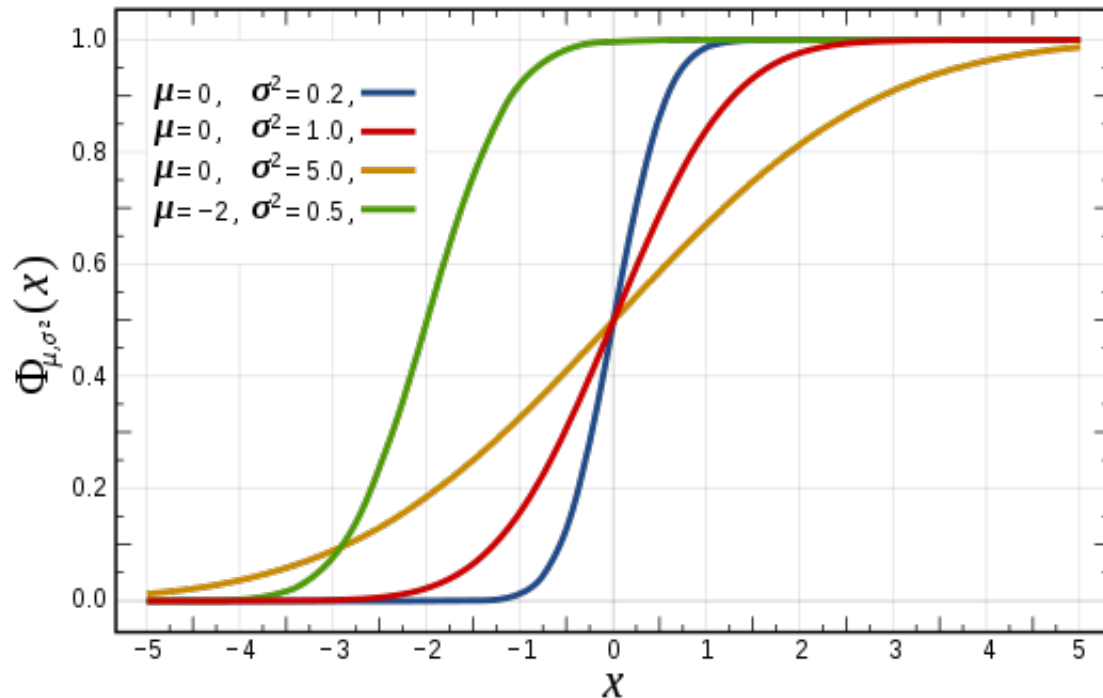


image Normal Distribution “

### Normal cumulative distribution function



### Properties of normal distribution

- symmetrical, unimodal, and bell-shaped
- on average, the error component will equal zero, the error above and below the mean will cancel out
- Z-Score is a statistical measurement is (above/below) the mean of the data
- important characteristics about z scores:
  1. mean of z scores is 0
  2. standard deviation of a standardized variable is always 1
  3. the linear transformation does not change the *form* of the distribution

The normal (or Gaussian) distribution was discovered in 1733 by Abraham de Moivre as an approximation to the binomial distribution when the number of trials is large.



*image Abraham de Moivre “*

- from [Abraham de Moivre - Wikipedia](#))

The Gaussian distribution was derived in 1809 by Carl Friedrich Gauss.





*image Carl Friedrich Gauss “*

- from [Carl Friedrich Gauss - Wikipedia](#))

Importance lies in the Central Limit Theorem, which states that the sum of a large number of independent random variables (binomial, Poisson, etc.) will approximate a normal distribution

## Central Limit Theorem

In probability theory, the central limit theorem (CLT) states that, given certain conditions, the arithmetic mean of a sufficiently large number of iterates of independent random variables, each with a well-defined expected value and well-defined variance, will be approximately normally distributed, regardless of the underlying distribution. The central limit theorem has a number of variants. In its common form, the random variables must be identically distributed.

- from [Central Limit Theorem - Wikipedia](#))

The Central Limit Theorem tells us that when the sample size is large the average  $\bar{Y}$  of a random sample follows a normal distribution centered at the population average  $\mu_Y$  and with standard deviation equal to the population standard deviation  $\sigma_Y$ , divided by the square root of the sample size  $N$ .

This means that if we subtract a constant from a random variable, the mean of the new random variable shifts by that constant. If  $X$  is a random variable with mean  $\mu$  and  $a$  is a constant, the mean of  $X - a$  is  $\mu - a$ .

This property also holds for the spread, if  $X$  is a random variable with mean  $\mu$  and SD  $\sigma$ , and  $a$  is a constant, then the mean and SD of  $aX$  are  $a\mu$  and  $|a| \sigma$  respectively. This implies that if we take many samples of size  $N$  then the quantity

$$\frac{\bar{Y} - \mu}{\sigma_Y / \sqrt{N}}$$

is approximated with a normal distribution centered at 0 and with standard deviation 1.

## The t-distribution

In probability and statistics, Student's t-distribution (or simply the t-distribution) is any member of a family of continuous probability distributions that arises when estimating the mean of a normally distributed population in situations where the sample size is small and population standard deviation is unknown. Whereas a normal distribution describes a full population, t-distributions describe samples drawn from a full population; accordingly, the t-distribution for each sample size is different, and the larger the sample, the more the distribution resembles a normal distribution. The t-distribution plays a role in a number of widely used statistical analyses, including the Student's t-test for assessing the statistical significance of the difference between two sample means, the construction of confidence intervals for the difference between two population means, and in linear regression analysis. The Student's t-distribution also arises in the Bayesian analysis of data from a normal family.

- from [The t-distribution - Wikipedia](#))

When the CLT does not apply (i.e. as the number of samples is large), there is another option that does not rely on large samples. When a the original population from which a random variable, say  $Y$ , is sampled is normally distributed with mean 0 then we can calculate the distribution of

number of variants. In its common form, the random variables must be identically distributed.

$$\sqrt{N} \frac{\bar{Y}}{s_Y}$$

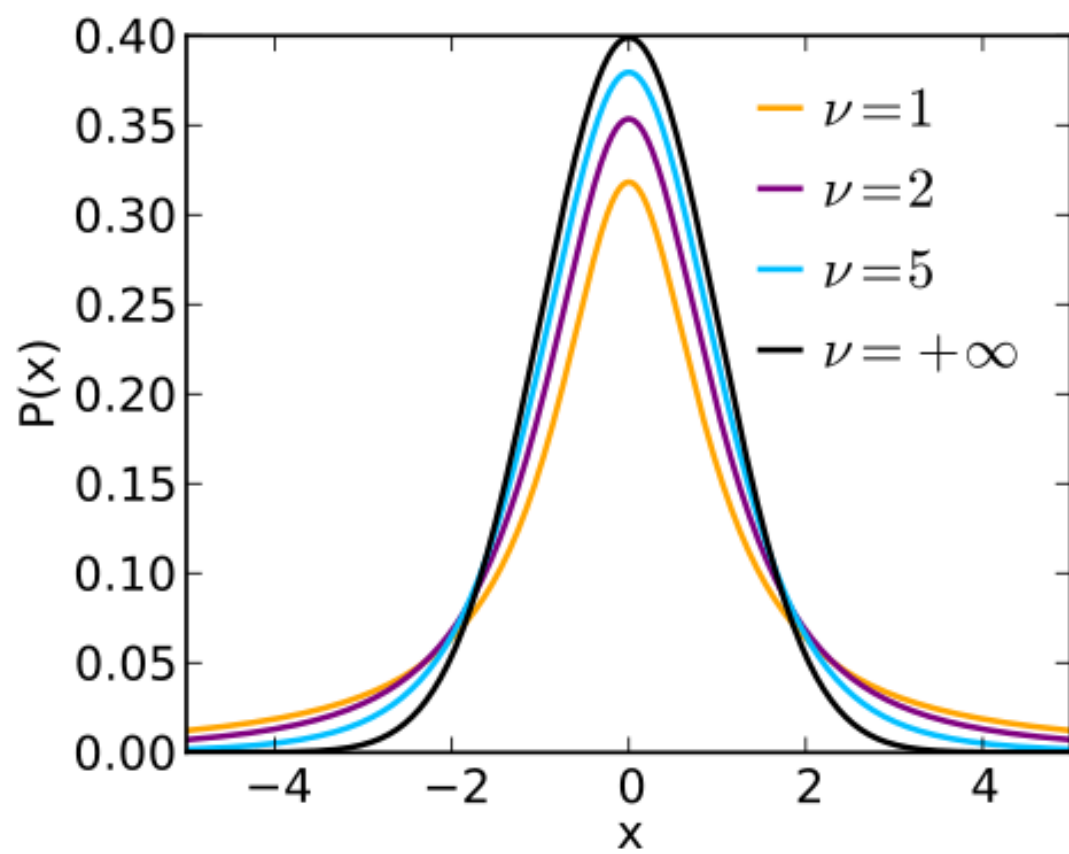
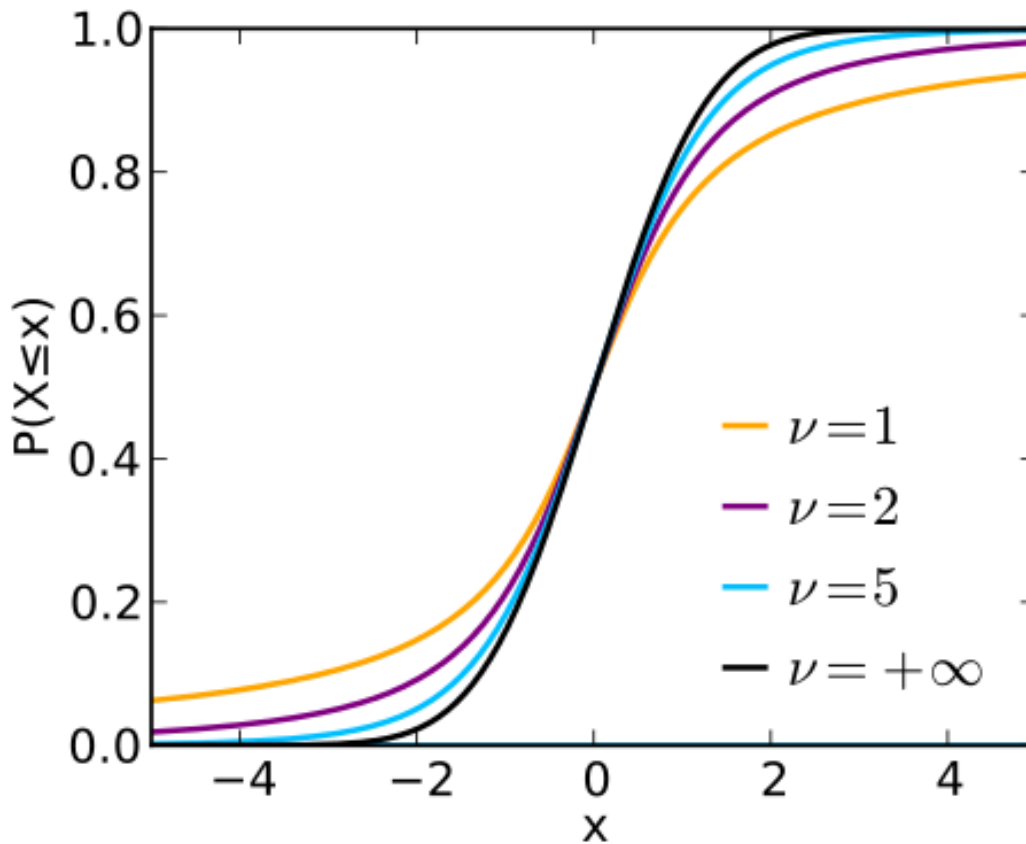


image Student's  $t$ -distribution “

Normal cumulative distribution function



## Normal Distribution in R

rnorm function in R. Type ?rnorm

*The Normal Distribution*

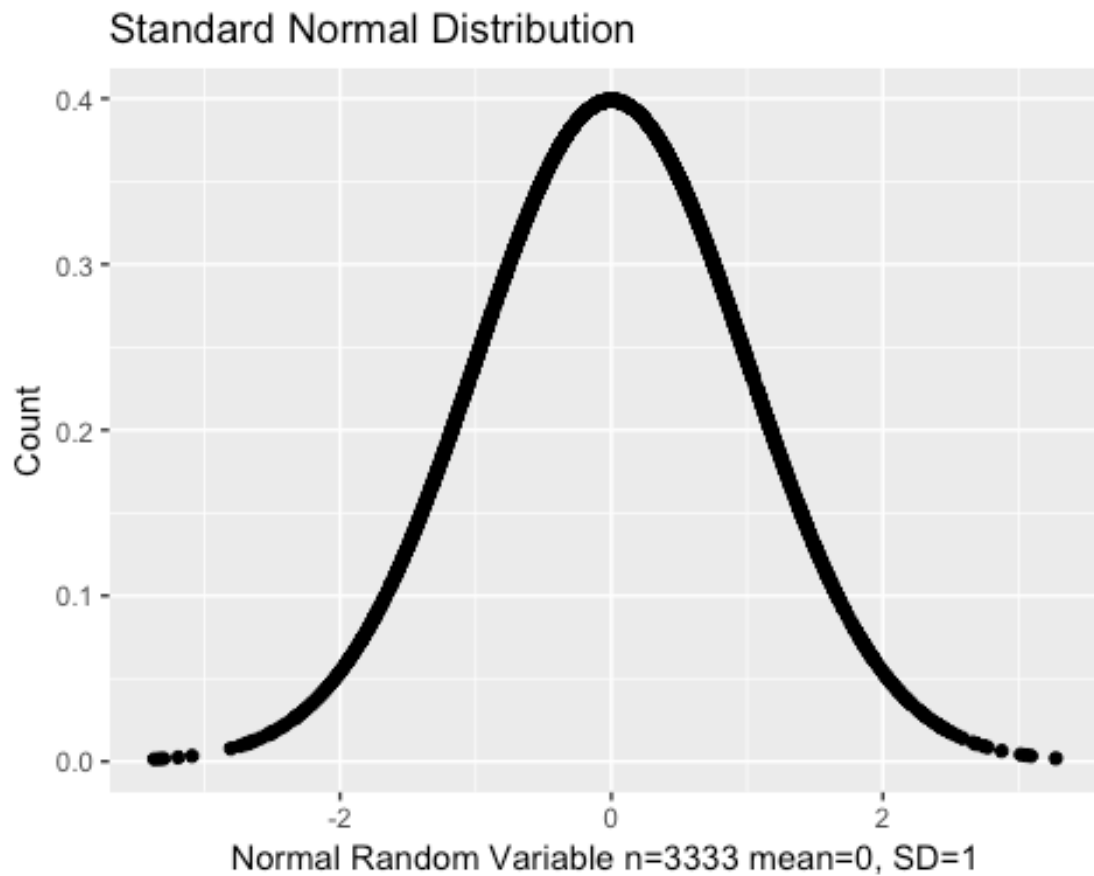
*Description*

These functions provide information about the Normal Distribution. Density, distribution function, quantile function and random generation for the normal distribution with mean equal to mean and standard deviation equal to sd.

*Usage*

- `dnorm(x, mean = 0, sd = 1, log = FALSE)` // gives the density
- `pnorm(q, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)` // gives the distribution function
- `qnorm(p, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)` // gives the quantile function
- `rnorm(n, mean = 0, sd = 1)` // generates random deviates. (a random sample)

```
norm_dist<-rnorm(trails) # norm_dist<-rnorm(33333, mean=33, sd=13)
norm_denisty<-dnorm(norm_dist)
ggplot(data.frame(x=norm_dist,y=norm_denisty)) + aes(x=x,y=y) + geom_point()
+ labs(title="Standard Normal Distribution", y="Count", x="Normal Random
Variable n=3333 mean=0, SD=1")
```



```
pnorm(c(-3,-2,-1,0,1,2,3))
## [1] 0.001349898 0.022750132 0.158655254 0.500000000 0.841344746
0.977249868
## [7] 0.998650102

head(pnorm(norm_dist))
## [1] 0.9888572 0.8988909 0.7278853 0.3655243 0.4505659 0.4768617

pnorm(1)-pnorm(-1) # +/- sd=1
## [1] 0.6826895

sqrt(0.2) # sd^2=var=0.2 sqrt(0.2)
## [1] 0.4472136
```

```

r_normal <-data.frame(A=rnorm(n=trails,mean=0,sd=sqrt(0.2)),
                     B=rnorm(n=trails,mean=0,sd=sqrt(1.0)),
                     C=rnorm(n=trails,mean=0,sd=sqrt(5.0)),
                     D=rnorm(n=trails,mean=-2,sd=sqrt(0.5)))

head(r_normal)

##           A           B           C           D
## 1  0.54155509 -2.22714526 -0.1040236 -2.351942
## 2 -0.47483148 -0.53423550 -2.8076141 -1.356664
## 3  0.10344855 -0.33500942 -3.6522453 -1.930220
## 4 -0.07541431  0.07897436  2.1417539 -2.093454
## 5 -0.15449491  0.07004269 -3.6008573 -2.239520
## 6 -0.60500401  2.13413287 -2.0847004 -1.856297

summary(r_normal)

##           A           B           C
## Min.      :-1.523699  Min.      :-3.016040  Min.      :-7.73420
## 1st Qu.: -0.305862   1st Qu.: -0.678949   1st Qu.: -1.59105
## Median : -0.009247   Median : -0.012332   Median : -0.05598
## Mean     :-0.006160   Mean     :-0.009371   Mean     :-0.02873
## 3rd Qu.:  0.291037   3rd Qu.:  0.639749   3rd Qu.:  1.50373
## Max.      : 1.861096   Max.      :  3.700725   Max.      :  8.25200
##           D
## Min.      :-4.6922
## 1st Qu.: -2.4903
## Median : -1.9967
## Mean     :-1.9958
## 3rd Qu.: -1.5104
## Max.      :  0.3461

require(reshape2)

## Loading required package: reshape2

rnd <- melt(data=r_normal)

## No id variables; using all as measure variables

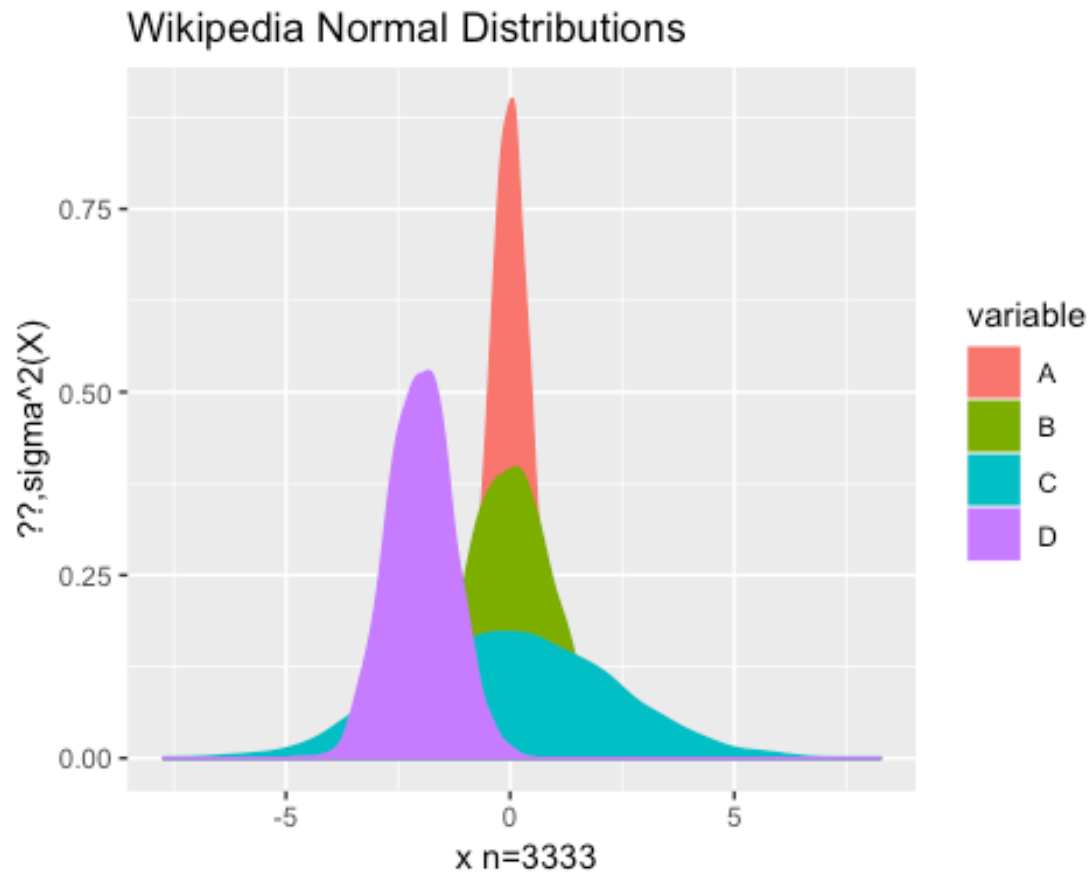
head(rnd)

##   variable      value
## 1         A  0.54155509
## 2         A -0.47483148
## 3         A  0.10344855
## 4         A -0.07541431
## 5         A -0.15449491
## 6         A -0.60500401

ggplot(rnd, aes(x=value)) +
geom_density(aes(group=variable,color=variable,fill=variable)) +

```

```
labs(title="Wikipedia Normal Distributions", y="??,sigma^2(X)", x=" x n=3333")
```



## Binomial Distribution

$$X \sim B(n, p)$$

$$P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k} \quad k = 1, 2, \dots, n$$

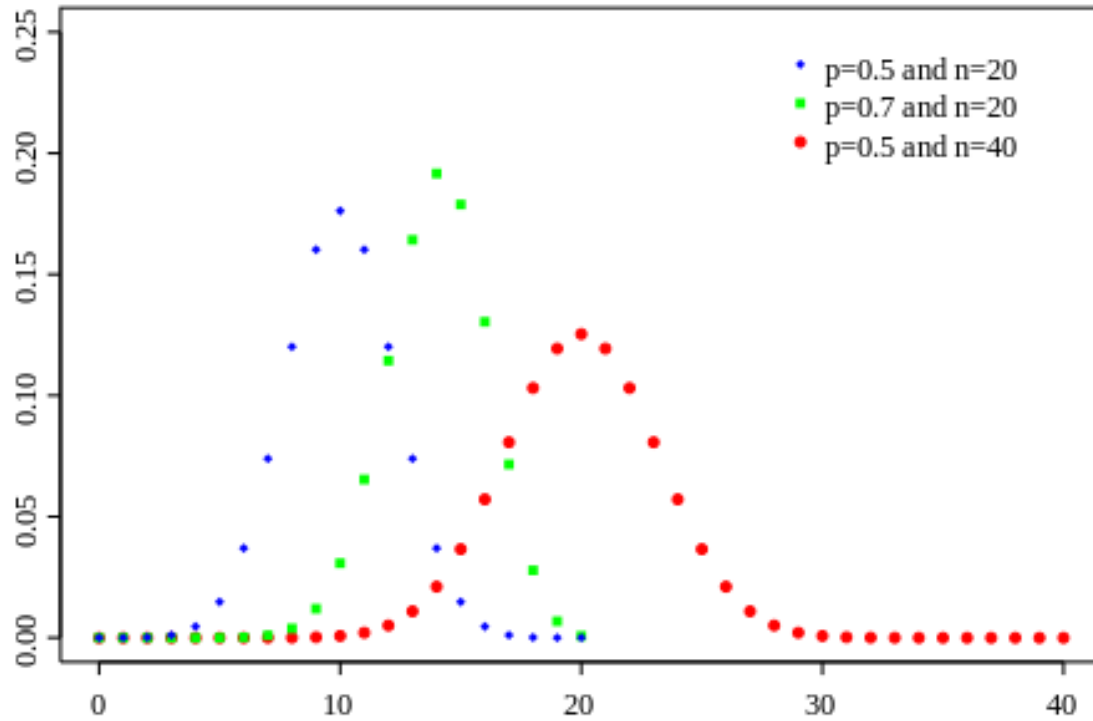
$$\binom{n}{k} = \frac{n!}{k! (n - k)!}$$

### *Binomial Distribution*

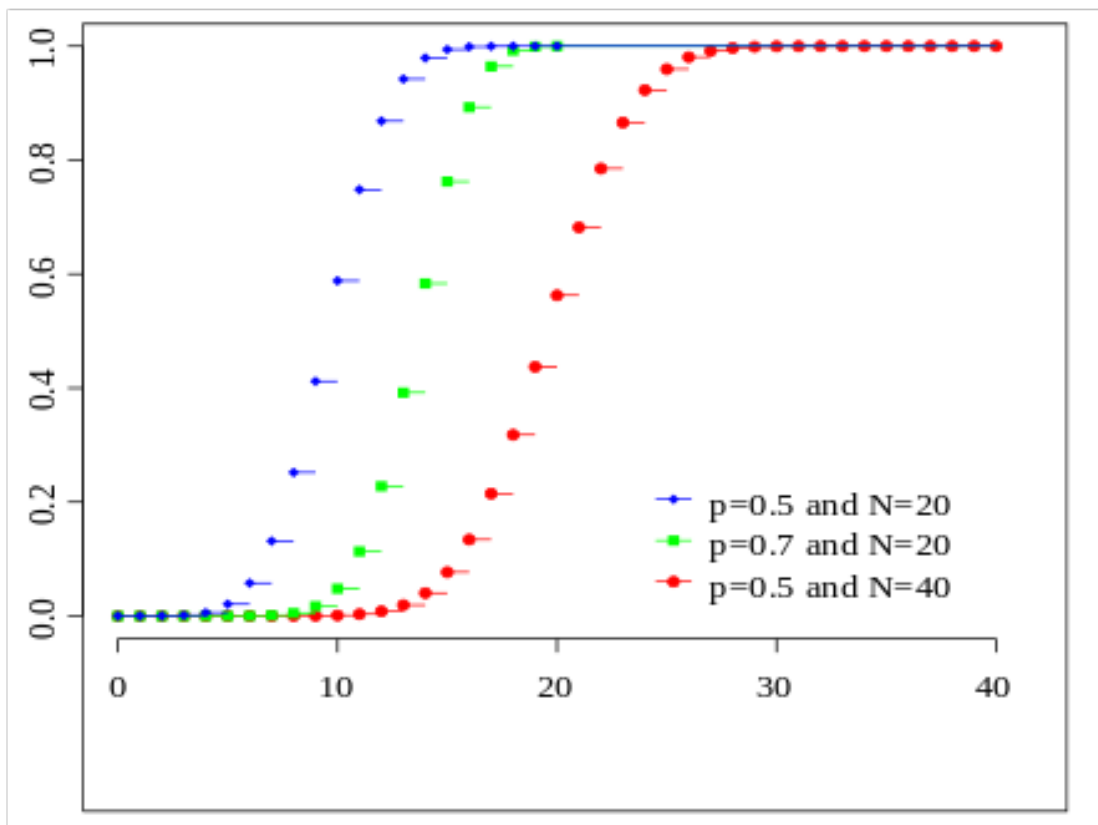
In probability theory and statistics, the binomial distribution with parameters  $n$  and  $p$  is the discrete probability distribution of the number of successes in a sequence of  $n$  independent yes/no experiments, each of which yields success with probability  $p$ . A success/failure experiment is also called a Bernoulli experiment or Bernoulli trial; when  $n = 1$ , the binomial distribution is a Bernoulli distribution.

- from [Binomial Distribution - Wikipedia](#)

Binomial Distribution



Binomial cumulative distribution function





- The data arise from a sequence of  $n$  independent trials.
- At each trial there are only two possible outcomes, conventionally called success and failure.
- The probability of success,  $p$ , is the same in each trial.
- The random variable of interest is the number of successes,  $X$ , in the  $n$  trials.
- The assumptions of independence and constant  $p$  are important. If they are invalid, so is the binomial distribution

### *Bernoulli Random Variables*

- Imagine a simple trial with only two possible outcomes
  - Success (S) with probability  $p$ .
  - Failure (F) with probability  $1-p$ .
- Examples
  - Toss of a coin (heads or tails)
  - Gender of a newborn (male or female)

## Binomial Distribution in R

rbinom function in R. Type ?rbinom

### *The Binomial Distribution*

#### *Description*

These functions provide information about the Binomial Distribution. Density, distribution function, quantile function and random generation for the binomial distribution with parameters size and prob.

This is conventionally interpreted as the number of 'successes' in size trials.

#### *Usage*

- `dbinom(x, size, prob, log = FALSE)` // gives the density
- `pbinom(q, size, prob, lower.tail = TRUE, log.p = FALSE)` // gives the distribution function
- `qbinom(p, size, prob, lower.tail = TRUE, log.p = FALSE)` // gives the quantile function
- `rbinom(n, size, prob)` // generates random deviates. (a random sample)

```
bi_dist<-rbinom(n=1,size=9,prob=0.33) # n=1 means one draw size=9 means nine trails
```

```
bi_dist # we expect 0.33* 9 or about 3 success
```

```
## [1] 5
```

```
bi_dist<-rbinom(n=5,size=9,prob=0.33) # n=5 means five draws size=9 means nine trails
```

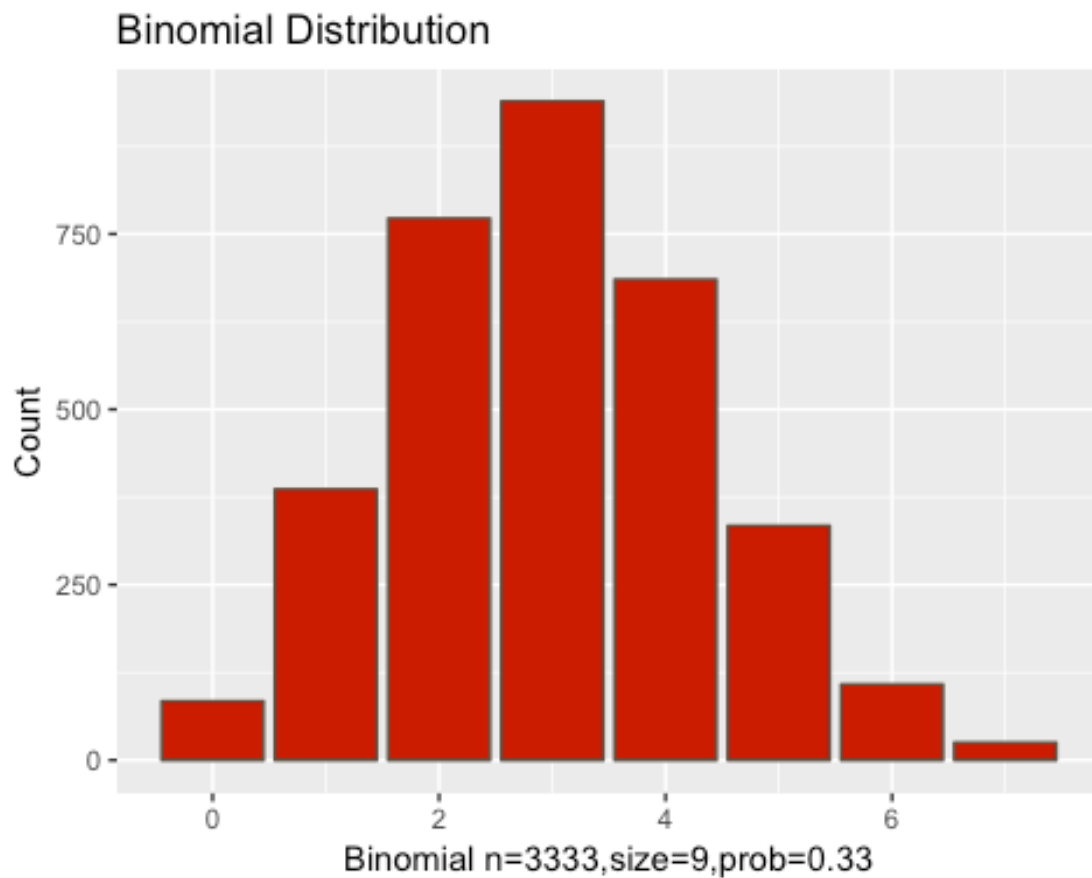
```
bi_dist # we expect each trial to be around 3 success
```

```
## [1] 3 4 4 1 1
```

```
bernoulli_dist<-rbinom(n=1,size=1,prob=0.33) # Bernoulli distribution
bi_dist<-rbinom(n=trails,size=9,prob=0.33) # same as bi_dist<-rbi(3333)
summary(bi_dist)

##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.000  2.000  3.000  2.994  4.000  7.000

data=data.frame(x=bi_dist)
ggplot(data) + aes(x=x) + geom_bar(color=I('#615445'), fill=I('#CC0000')) +
labs(title="Binomial Distribution", y="Count", x="Binomial
n=3333,size=9,prob=0.33")
```



```
r_binomial <-data.frame(A=rbinom(n=trails,size=20,prob=0.5),
                        B=rbinom(n=trails,size=20,prob=0.7),
                        C=rbinom(n=trails,size=40,prob=0.5))
```

```
head(r_binomial)
```

```
##      A  B  C
## 1 14 15 23
## 2 11 16 12
## 3  9 12 18
## 4 13 12 22
## 5  8 11 23
## 6 10 15 19
```

```
summary(r_binomial)
```

```
##           A           B           C
## Min.      : 3.000   Min.      : 5.00   Min.      : 9.00
## 1st Qu.: 8.000   1st Qu.:13.00   1st Qu.:18.00
## Median :10.000   Median :14.00   Median :20.00
## Mean      : 9.994   Mean      :14.02   Mean      :19.94
## 3rd Qu.:12.000   3rd Qu.:15.00   3rd Qu.:22.00
## Max.      :17.000   Max.      :20.00   Max.      :32.00
```

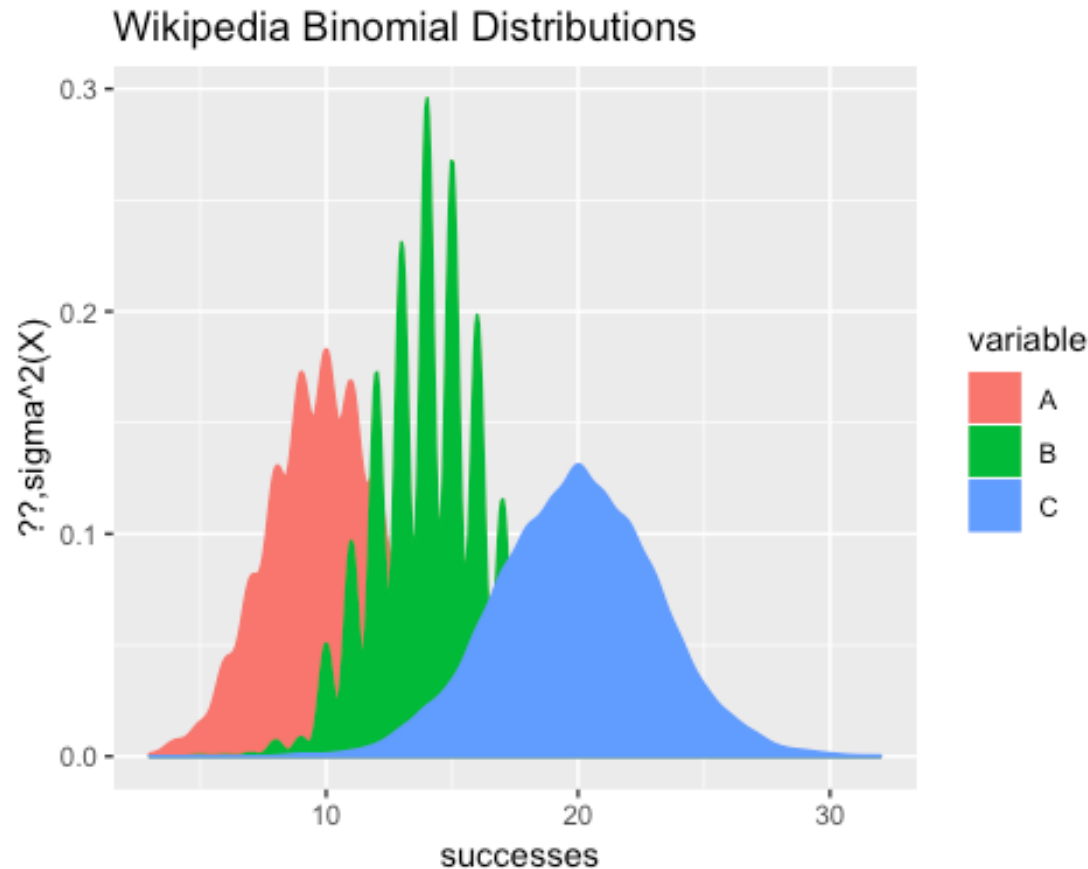
```
rnd <- melt(data=r_binomial)
```

```
## No id variables; using all as measure variables
```

```
head(rnd)
```

```
##   variable value
## 1         A    14
## 2         A    11
## 3         A     9
## 4         A    13
## 5         A     8
## 6         A    10
```

```
ggplot(rnd, aes(x=value)) +
  geom_density(aes(group=variable,color=variable,fill=variable)) +
  labs(title="Wikipedia Binomial Distributions", y="??,sigma^2(X)", x="
successes")
```



## Poisson Distribution

$X$  expresses the number of “rare” events

$$X \sim P(\lambda) \quad \lambda > 0$$

$$P(X = x) = \frac{e^{-\lambda} \lambda^x}{x!} \quad x = 1, 2, \dots, n$$

### *Poisson Distribution*

In probability theory and statistics, the Poisson distribution, named after French mathematician Siméon Denis Poisson, is a discrete probability distribution that expresses the probability of a given number of events occurring in a fixed interval of time and/or space if these events occur with a constant rate per time unit and independently of the time since the last event. The Poisson distribution can also be used for the number of events in other specified intervals such as distance, area or volume.

For instance, an individual keeping track of the amount of mail they receive each day may notice that they receive an average number of 4 letters per day. If receiving any particular piece of mail doesn't affect the arrival times of future pieces of mail, i.e., if pieces of mail from a wide range of sources arrive independently of one another, then a reasonable assumption is that the number of pieces of mail received per day obeys a Poisson

distribution. Other examples that may follow a Poisson: the number of phone calls received by a call center per hour, the number of decay events per second from a radioactive source, or the number of taxis passing a particular street corner per hour.

The Poisson distribution gives us a probability mass for discrete natural numbers  $k$  given some mean value  $\lambda$ . Knowing that, on average,  $\lambda$  discrete events occur over some time period, the Poisson distribution gives us the probability of seeing exactly  $k$  events in that time period.

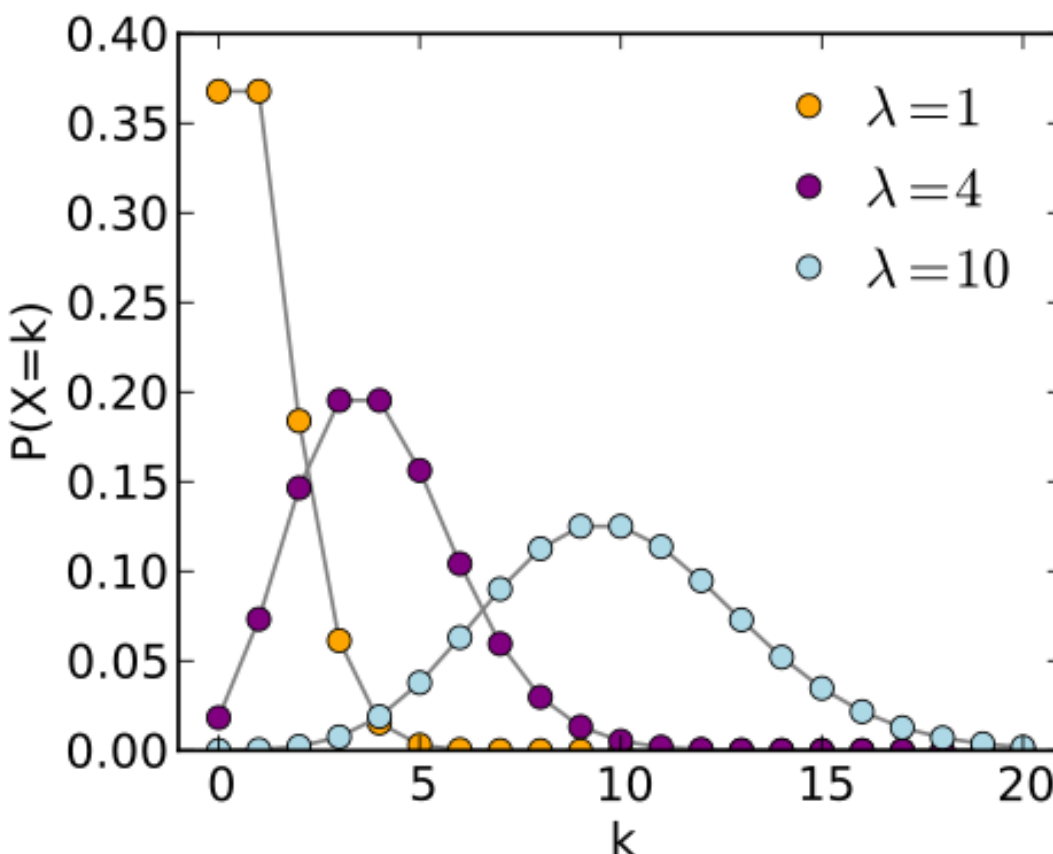
For example, if a call center gets, on average, 100 customers per day, the Poisson distribution can tell us the probability of getting exactly 150 customers today.

$k \in \mathbf{N}$  (i.e. is a natural number) because, on any particular day, you can't have a fraction of a phone call. The probability of any non-integer number of people calling in is zero. E.g.,  $P(150.5) = 0$ .

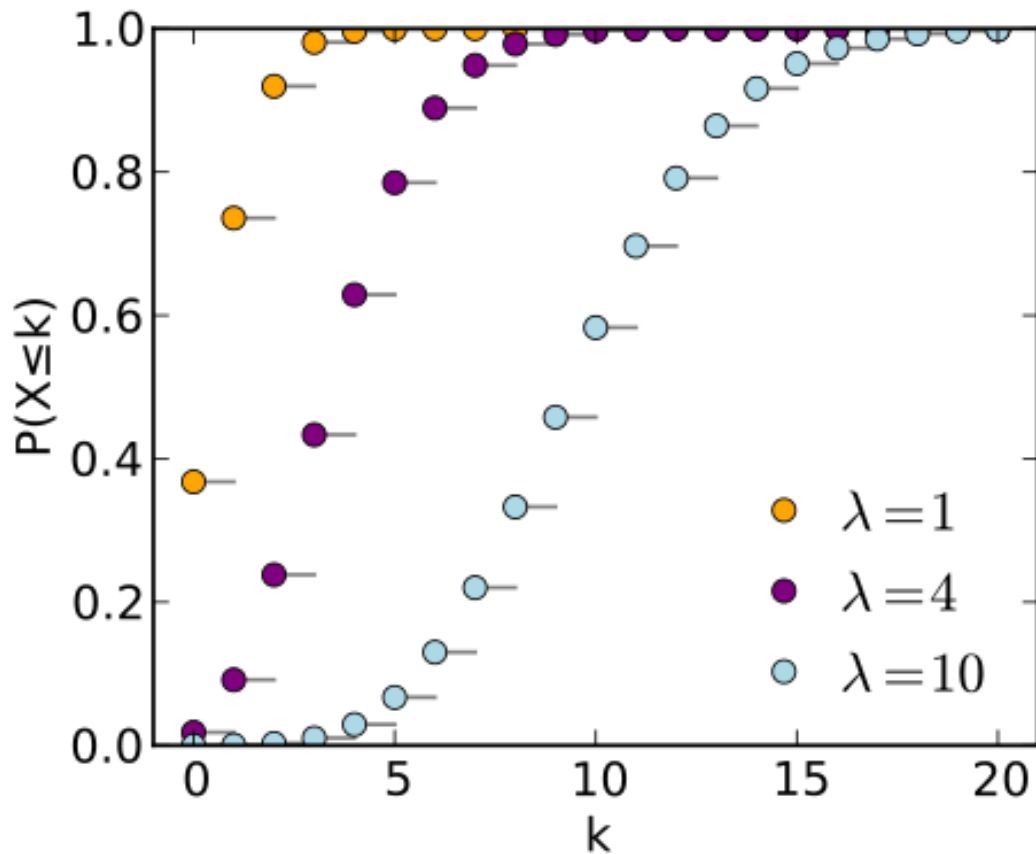
$\lambda \in \mathbf{R}$  (i.e. is a real number) because, even though any *particular* day must have an integer number of people, the *mean* number of people taken over many days can be fractional (and usually is). It's why the "average" number of phone calls per day could be 3.5 even though half a phone call won't occur.

- from [Poisson Distribution - Wikipedia](#))

Poisson Distribution



## Poisson cumulative distribution function



## Properties of Poisson distribution

- The mean number of successes from  $n$  trials is  $\mu = np$
- If we substitute  $\mu/n$  for  $p$ , and let  $n$  tend to infinity, the binomial distribution becomes the Poisson distribution.
- Poisson distributions are often used to describe the number of occurrences of a 'rare' event. For example
  - The number of storms in a season
  - The number of occasions in a season when river levels exceed a certain value
- The main assumptions are that events occur
  - at random (the occurrence of an event doesn't change the probability of it happening again)
  - at a constant rate
- Poisson distributions also arise as approximations to binomials when  $n$  is large and  $p$  is small.
- When there is a large number of trials, but a very small probability of success, binomial calculation becomes impractical

## Poisson Distribution in R

rpois function in R. Type ?rpois

### *The Poisson Distribution*

#### *Description*

These functions provide information about the Poisson Distribution. Density, distribution function, quantile function and random generation for the Poisson distribution with parameter lambda.

#### *Usage*

- dpois(x, lambda, log = FALSE) // gives the density
- ppois(q, lambda, lower.tail = TRUE, log.p = FALSE) // gives the distribution function
- qpois(p, lambda, lower.tail = TRUE, log.p = FALSE) // gives the quantile function
- rpois(n, lambda) // generates random deviates. (a random sample)

```
poisson_dist <- data.frame(lambda1=rpois(n=trials,lambda=1),
lambda3=rpois(n=trials,lambda=3), lambda5=rpois(n=trials,lambda=5),
lambda10=rpois(n=trials,lambda=10))
head(poisson_dist)
```

```
##   lambda1 lambda3 lambda5 lambda10
## 1      0      3      4      8
## 2      0      4      9     18
## 3      0      5      4      7
## 4      1      3      9     14
## 5      1      4      5      6
## 6      2      3      7      5
```

```
str(poisson_dist)
```

```
## 'data.frame':   3333 obs. of  4 variables:
## $ lambda1 : int  0 0 0 1 1 2 0 1 1 1 ...
## $ lambda3 : int  3 4 5 3 4 3 5 1 5 4 ...
## $ lambda5 : int  4 9 4 9 5 7 8 4 5 6 ...
## $ lambda10: int  8 18 7 14 6 5 9 7 12 11 ...
```

```
poisson <- melt(data=poisson_dist, variable.name = "lambda", value.name =
"count")
```

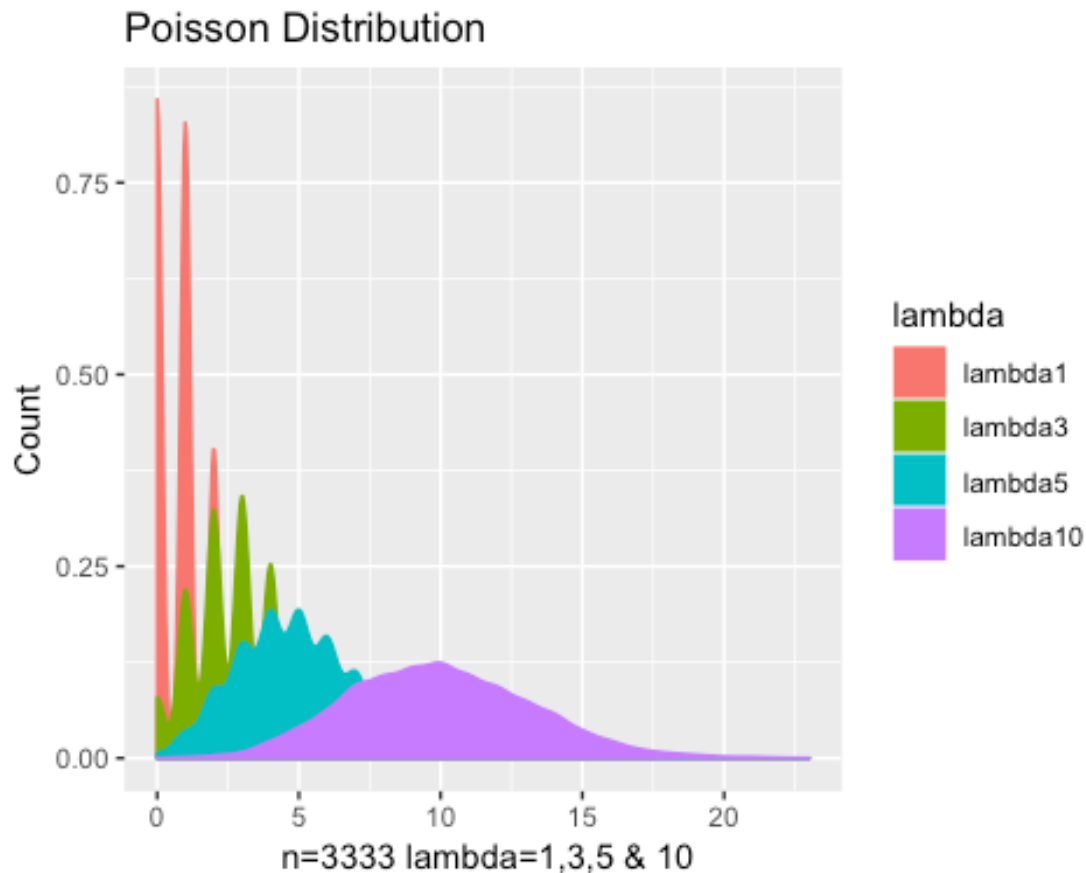
```
## No id variables; using all as measure variables
```

```
head(poisson)
```

```
##   lambda count
## 1 lambda1     0
## 2 lambda1     0
## 3 lambda1     0
## 4 lambda1     1
```

```
## 5 lambda1      1
## 6 lambda1      2

ggplot(poisson, aes(x=count)) +
  geom_density(aes(group=lambda,color=lambda,fill=lambda)) +
  labs(title="Poisson Distribution", y="Count", x=" n=3333 lambda=1,3,5 & 10")
```



```
#ggplot(poisson, aes(x=value)) +
  geom_density(aes(group=variable,color=variable,fill=variable)) +
  labs(title="Poisson Distribution", y="Count", x=" n=3333 Lambda=1,3,5 & 10")
```

## Poisson and Binomial Distributions

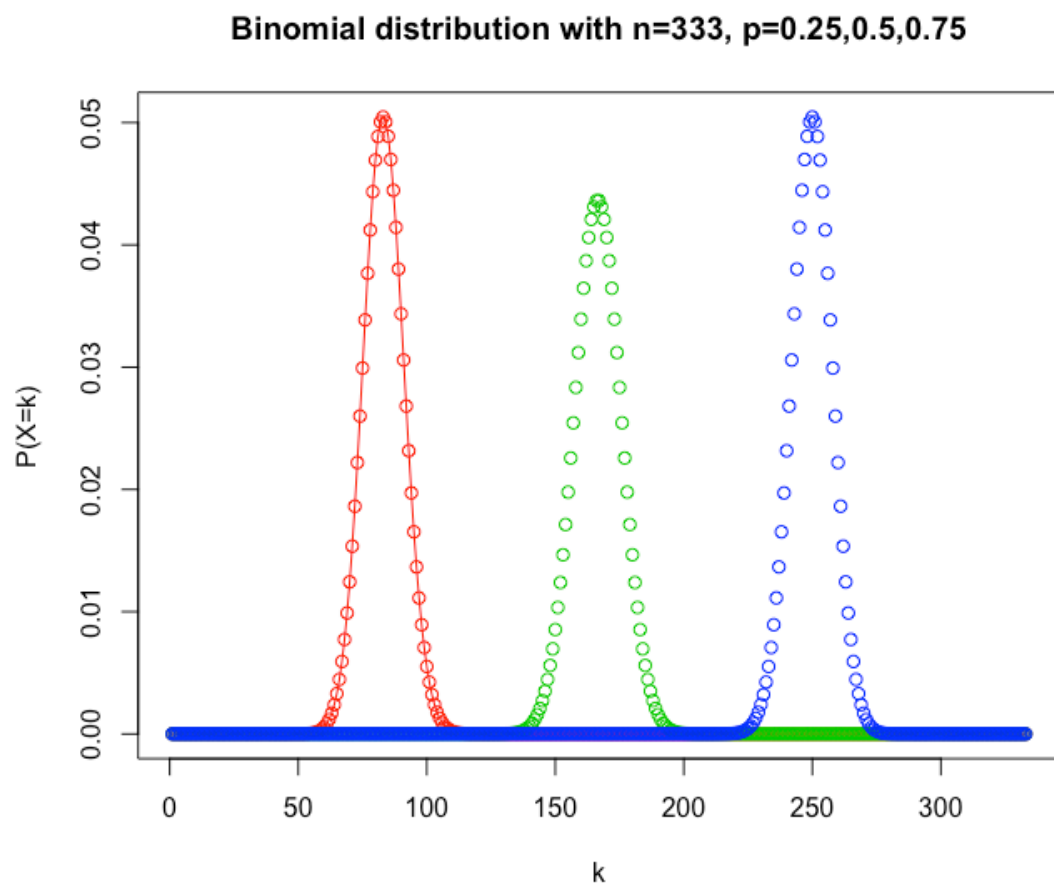
The binomial distribution is usually shown with a fixed  $n$ , with different values of  $p$  that will affect the  $k$  successes from the fixed  $n$  trials. This supposes we know the number of trials beforehand. We can graph the binomial distribution as a set of curves with a fixed  $n$ , and varying probabilities of the probability of success,  $p$ , below.

```
trails<-333
d_binomial <-data.frame(A=dbinom(1:trails, size=trails, prob=0.25),
                        B=dbinom(1:trails, size=trails, prob=0.5),
                        C=dbinom(1:trails, size=trails, prob=0.75))
head(d_binomial)
```



```
##           A           B           C
## 1 2.758822e-40 1.903074e-98 3.262784e-198
## 2 1.526548e-38 3.159103e-96 1.624867e-195
## 3 5.614305e-37 3.485543e-94 5.378309e-193
## 4 1.543934e-35 2.875573e-92 1.331131e-190
## 5 3.386362e-34 1.892127e-90 2.627653e-188
## 6 6.170703e-33 1.034363e-88 4.309351e-186
```

```
plot(1:trails, d_binomial$A, type="o", col=2, ylab="P(X=k)", xlab="k",
main="Binomial distribution with n=333, p=0.25,0.5,0.75")
points(1:trails,d_binomial$B,col=3)
points(1:trails,d_binomial$C,col=4)
```

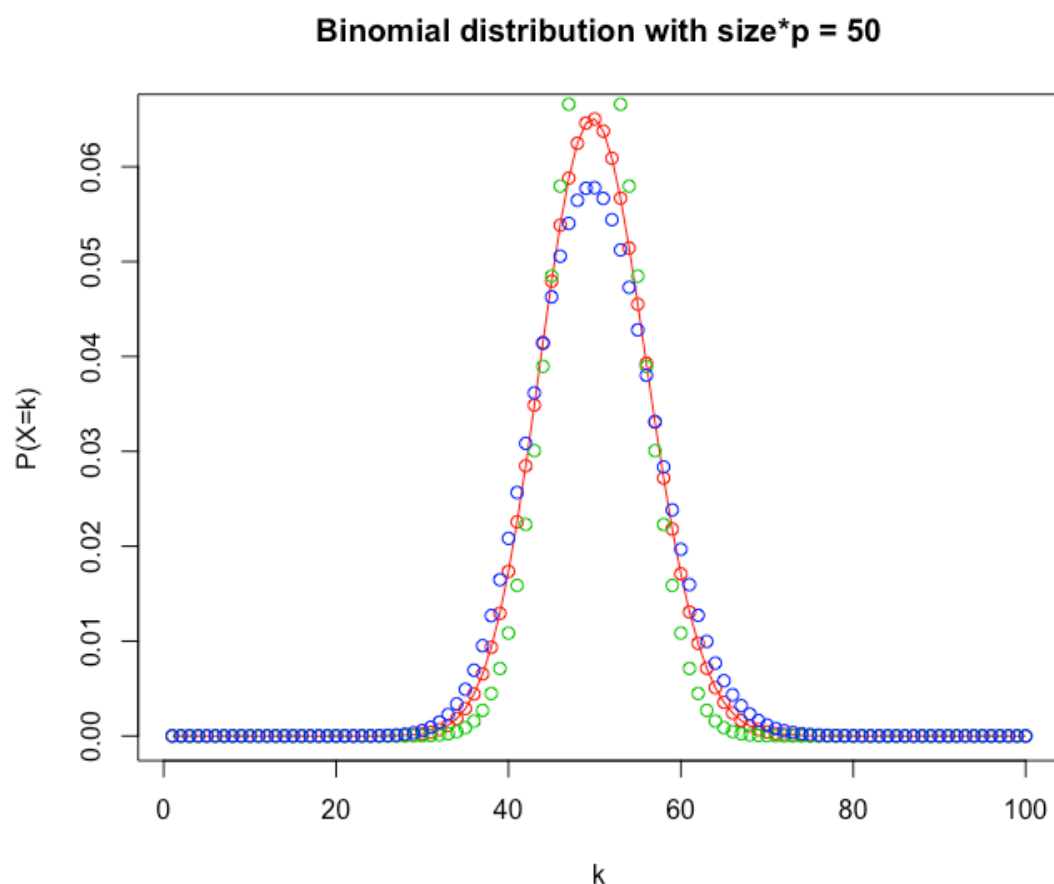


**What if we knew the rate but not the probability,  $p$ , or the number of trails,  $n$ ?**

But what if we were to invert the problem? What if we knew only the number of heads we observed, but not the total number of flips? If we have a known expected number of heads but an unknown number of flips, then we don't really know the true probability for each individual head. Rather we know that, on average,  $p = \text{mean}(k)/n$ . However if we were to

plot these all on the same graph in the vicinity of the same  $k$ , we can make them all have a convergent shape around  $\text{mean}(k)$  because, no matter how much we increase  $n$ , we decrease  $p$  proportionally so that, for all  $n$ , the peak stays at  $\text{mean}(k)$ .

```
trails<-100
d_binomial <-data.frame(A=dbinom(1:trails, size=200, prob=0.25),
                        B=dbinom(1:trails, size=100, prob=0.5),
                        C=dbinom(1:trails, size=1000, prob=0.05))
plot(1:trails, d_binomial$A, type="o", col=2, ylab="P(X=k)", xlab="k",
     main="Binomial distribution with size*p = 50")
points(1:trails,d_binomial$B,col=3)
points(1:trails,d_binomial$C,col=4)
```



Notice that the curves get just a little flatter as  $n$  gets higher. This is because the rightward tail keeps getting longer and longer. As  $n$  gets large, binomial distribution gives us an approximate Poisson distribution since, as  $n$  tends toward infinity, but  $p$  tends toward 0.

## Deriving the Poisson Distribution from the Binomial Distribution

Let's make this a little more formal. The binomial distribution works when we have a fixed number of events  $n$ , each with a constant probability of success  $p$ . In the Poisson

Distribution, we don't know the number of trials that will happen. Instead, we only know the average number of successes per time period, the rate  $\lambda$ . So we know the rate of successes per day, or per minute but not the number of trials  $n$  or the probability of success  $p$  that was used to estimate to that rate.

If  $n$  is the number of trials in our time period, then  $np$  is the success rate or  $\lambda$ , that is,  $\lambda = np$ . Solving for  $p$ , we get:

$$p = \frac{\lambda}{n} \quad (1)$$

Since the Binomial distribution is defined as below

$$P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k} \quad k = 1, 2, \dots, n \quad (2)$$

or equivalently

$$P(X = k) = \frac{n!}{k! (n - k)!} p^k (1 - p)^{n-k} \quad k = 1, 2, \dots, n \quad (3)$$

By substituting the above  $p$  from (1) into the binomial distribution (3)

$$P(X = k) = \frac{n!}{k! (n - k)!} \frac{\lambda^k}{n^k} \left(1 - \frac{\lambda}{n}\right)^{n-k} \quad (4)$$

For  $n$  large and  $p$  small:

$$P(X = k) \equiv \frac{e^{-\lambda} \lambda^k}{k!} \quad k = 1, 2, \dots, n \quad (5)$$

Which is the probability mass function for the Poisson distribution.

## Poisson approximation to the Binomial in R

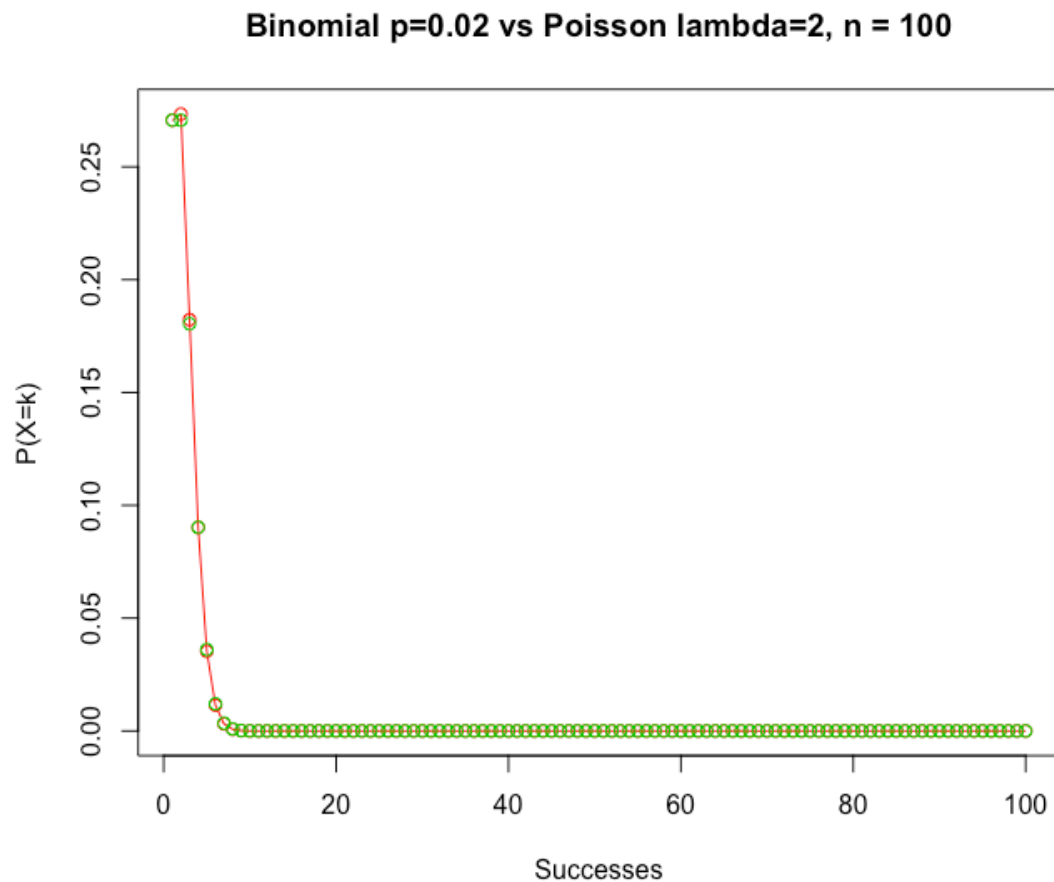
From the above derivation, it is clear that as  $n$  approaches infinity, and  $p$  approaches zero, a Binomial( $n, p$ ) will be approximated by a Poisson( $n \cdot p$ ). What is surprising is just how quickly this happens. That is how small a "large  $n$ " really is. The graph below has "large  $n$ " values as low as  $n = 100$ , and "small  $p$ " values of 0.02.

```
trails<-100
d_binomial <-data.frame(A=dbinom(1:trails, size=trails, prob=0.02),
                        B=dpois(1:trails,2))
head(d_binomial)

##           A           B
## 1 0.27065215 0.27067057
## 2 0.27341391 0.27067057
## 3 0.18227594 0.18044704
## 4 0.09020799 0.09022352
```

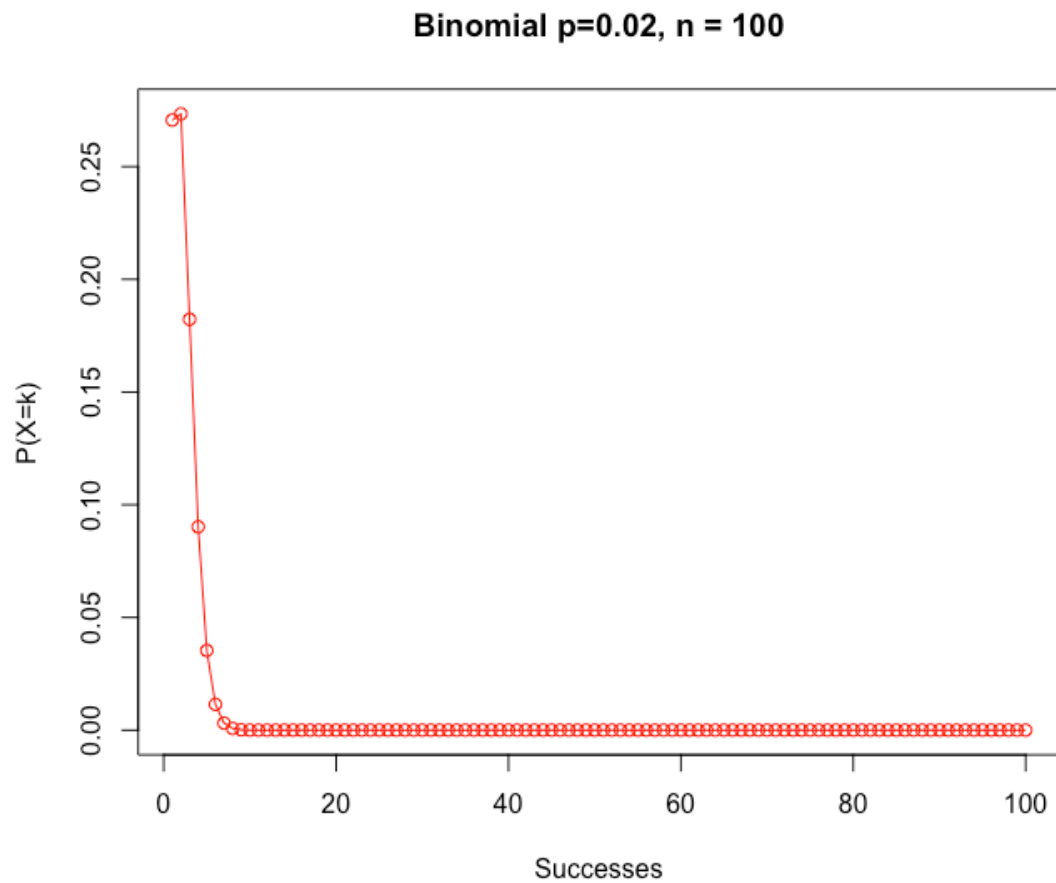
```
## 5 0.03534680 0.03608941
## 6 0.01142159 0.01202980
```

```
plot(1:trails, d_binomial$A, type="o", col=2, ylab="P(X=k)",
xlab="Successes", main="Binomial p=0.02 vs Poisson lambda=2, n = 100")
points(1:trails,d_binomial$B,col=3)
```



“Large n” Binomial p=0.02, n = 100

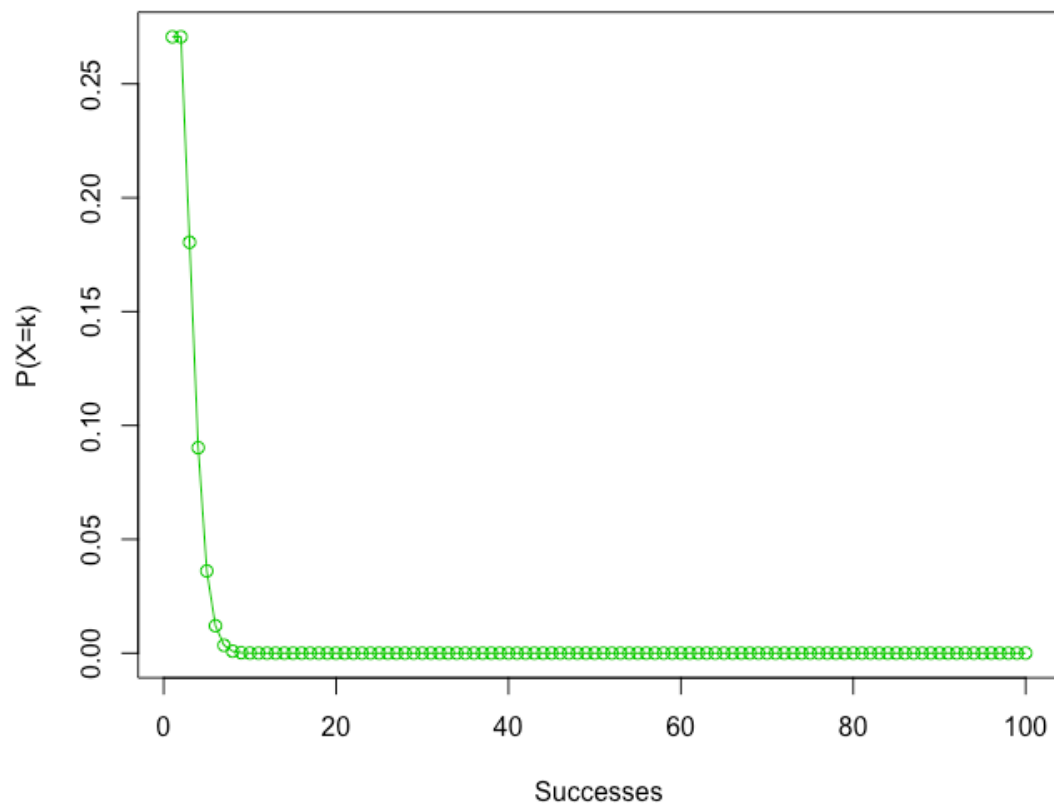
```
plot(1:trails, d_binomial$A, type="o", col=2, ylab="P(X=k)",
xlab="Successes", main="Binomial p=0.02, n = 100")
```



“Large  $n$ ” Poisson  $\lambda=2$ ,  $n = 100$

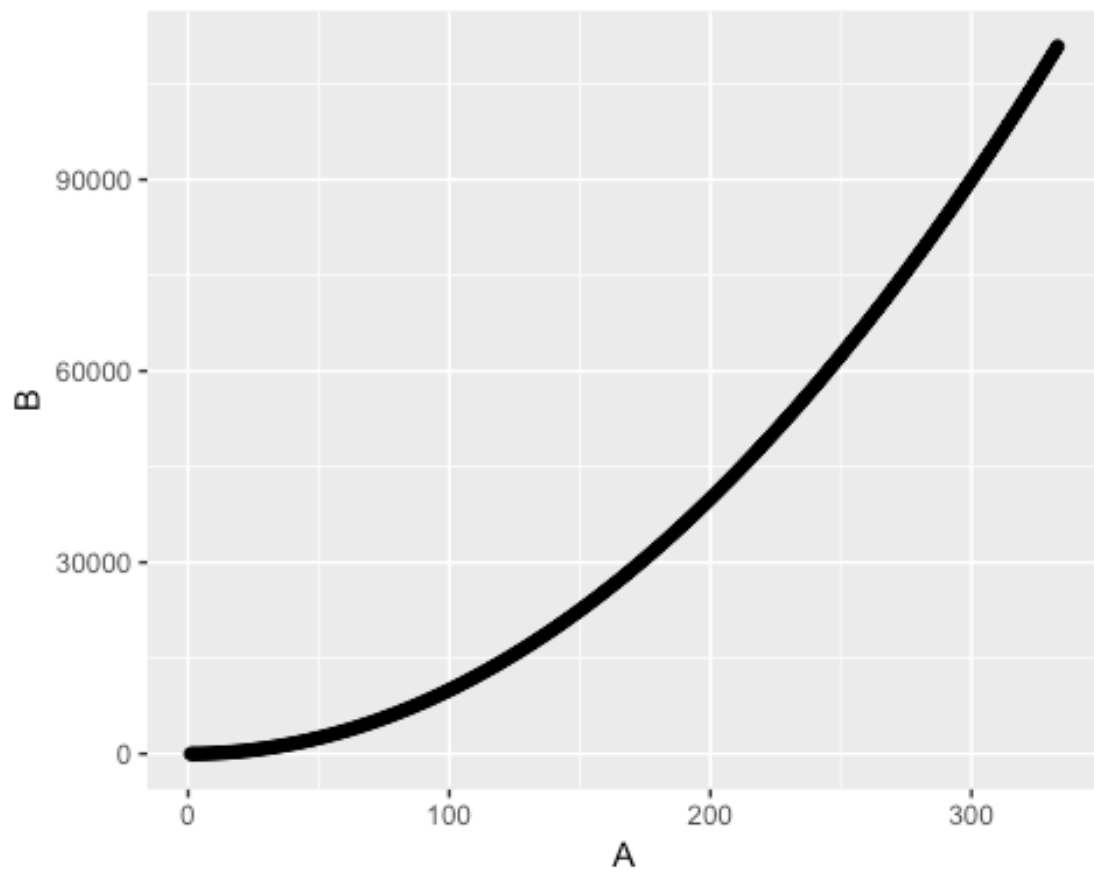
```
plot(1:trials, d_binomial$B, type="o", col=3, ylab="P(X=k)",  
xlab="Successes", main="Poisson lambda=2, n = 100")
```

### Poisson lambda=2, n = 100

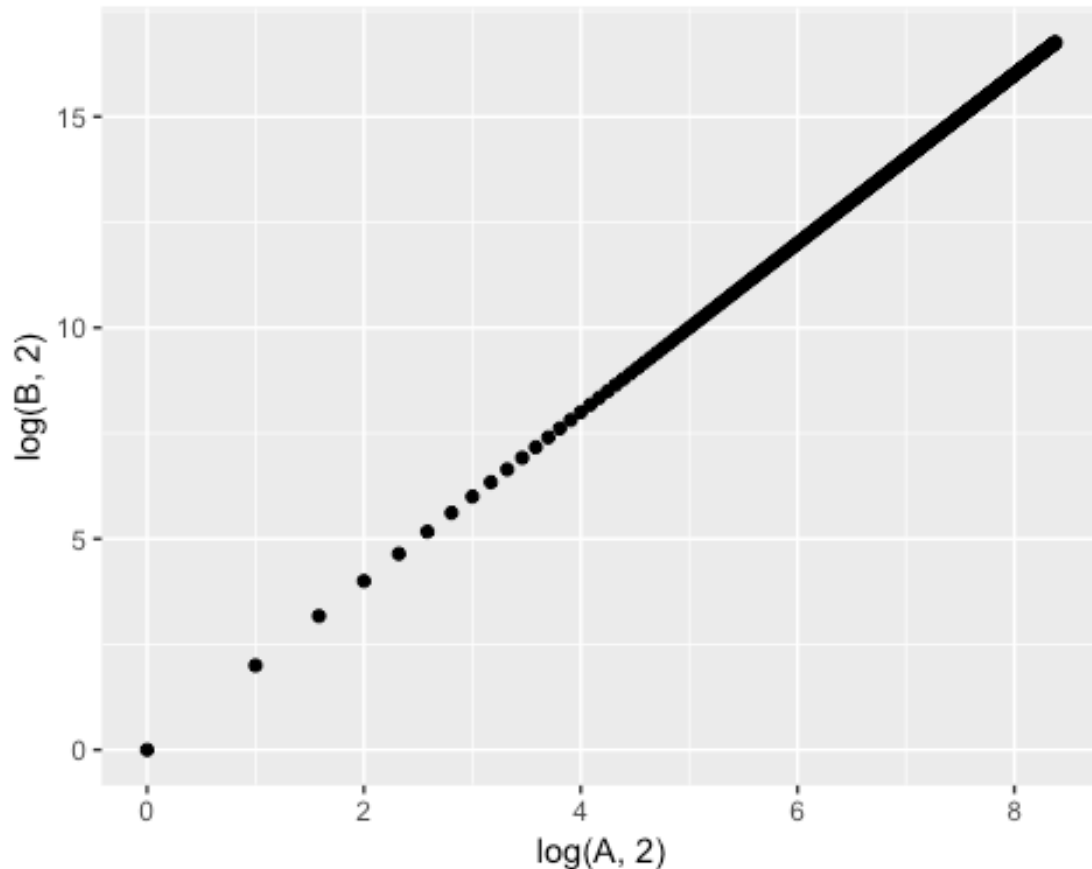


### Fat-Tailed Distribution

```
n<-333
A<-1:n
B<-A^2
llog<-data.frame(A,B)
qplot(A,B, data=llog)
```



```
qplot(log(A,2),log(B,2), data=llog)
```



### *Fat-Tailed Distribution*

In probability theory, the Fat-Tailed (or Gaussian) distribution is a very common continuous probability distribution. The Fat-Tailed distribution is remarkably useful because of the central limit theorem. In its most general form, under mild conditions, it states that averages of random variables independently drawn from independent distributions are Fat-Tailedly distributed. Physical quantities that are expected to be the sum of many independent processes (such as measurement errors) often have distributions that are nearly Fat-Tailed.

- from [Fat-Tailed Distribution - Wikipedia](#))

### *Properties of Fat-Tailed distribution*

- Power law distributions:
  - for variables assuming integer values  $> 0$
  - $\text{Prob} [X=k] \sim Ck^{-\alpha}$
  - typically  $0 < \alpha < 2$ ; smaller  $\alpha$  gives heavier tail
- For binomial, normal, and Poisson distributions the tail probabilities approach 0 exponentially fast
- What kind of phenomena does this distribution model?



- What kind of process would generate it?

## Cauchy Distribution

An example of a Fat-tailed distribution is the Cauchy distribution.

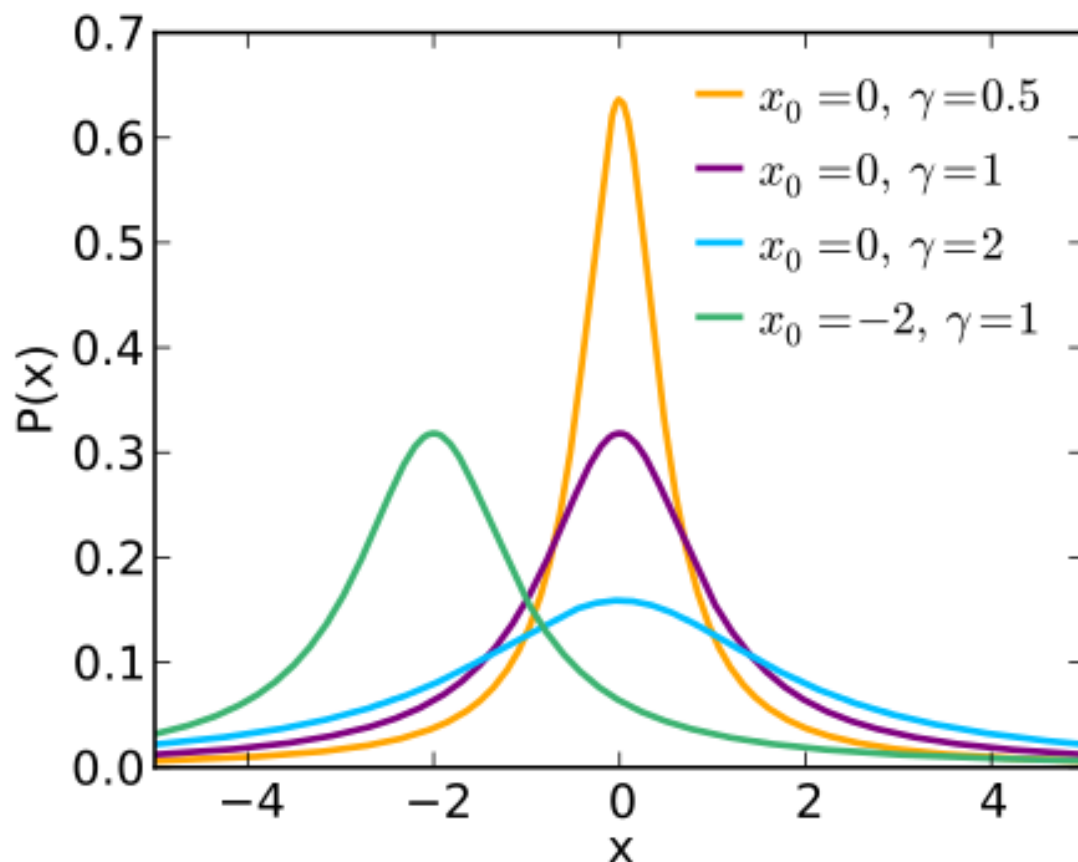
### *Cauchy Distribution*

The Cauchy distribution, named after Augustin Cauchy, is a continuous probability distribution. It is also known, especially among physicists, as the Lorentz distribution (after Hendrik Lorentz), Cauchy–Lorentz distribution, Lorentz(ian) function, or Breit–Wigner distribution. The simplest Cauchy distribution is called the standard Cauchy distribution. It is the distribution of a random variable that is the ratio of two independent standard normal variables and has the probability density function

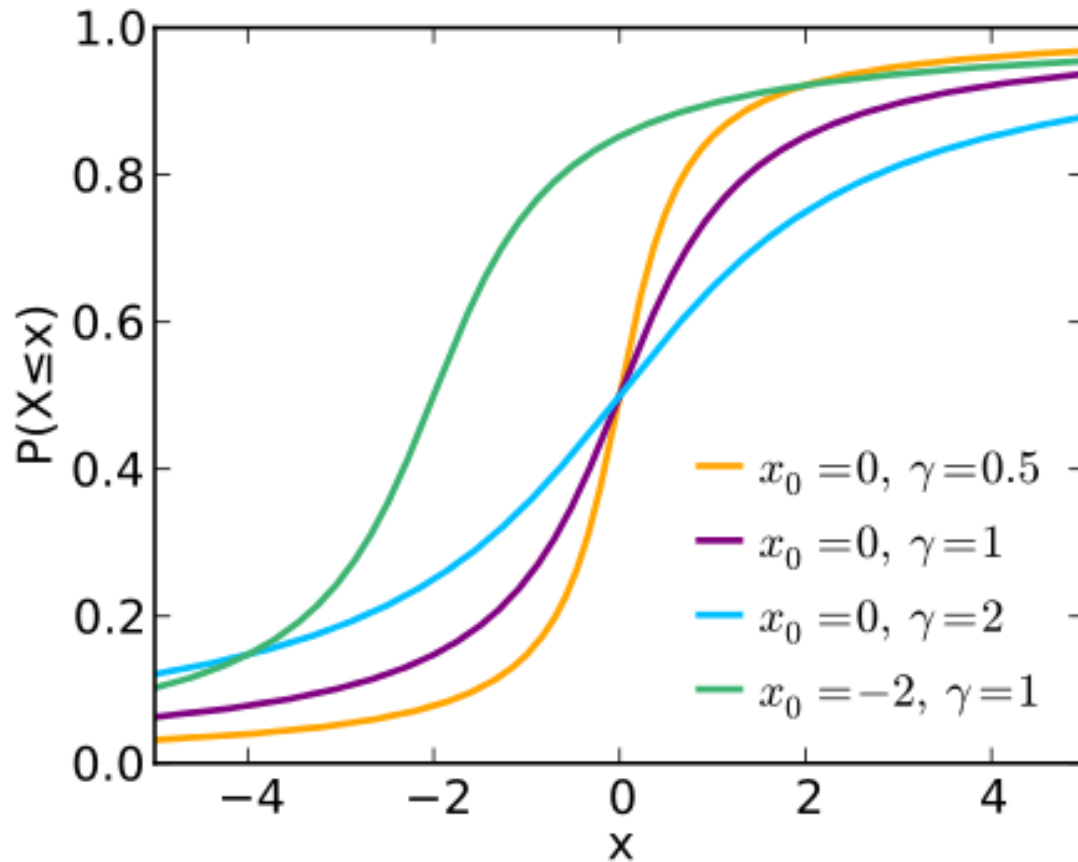
The Cauchy distribution is often used in statistics as the canonical example of a “pathological” distribution since both its mean and its variance are undefined. (But see the section Explanation of undefined moments below.) The Cauchy distribution does not have finite moments of order greater than or equal to one; only fractional absolute moments exist.[1] The Cauchy distribution has no moment generating function.

- from [Cauchy Distribution - Wikipedia](#))

### Cauchy Distribution



## Cauchy cumulative distribution function



## Homework Question 1

### *Probability distributions*

- Replicate and plot the fat-tailed Cauchy distributions from [https://en.wikipedia.org/wiki/Cauchy\\_distribution](https://en.wikipedia.org/wiki/Cauchy_distribution)
- Load the file M01\_Lesson\_02\_Q1.csv
- Answer the following questions for the data in each column:
  - How is the data distributed?
  - What are the summary statistics?
  - Are there anomalies/outliers?
  - Try to regenerate the data in each column.
  - Plot your regenerated data versus the original data. How does it compare?

