Find your seat

1. Go on to canvas to find your group letter if you haven't done so
2. Find the table corresponding to that letter and take a seat

Probability, matrices, and the apply() function

Brian Kissmer

USU Department of Biology

Sept. 3rd, 2024

Learning objectives

1. Understand how computers produce seemingly random data
2. Learn about various probability distributions
3. Understand what matrices are in R
4. Learn how to use the apply() function

Today's outline

1. Assigned seating
2. Course laptops
3. Lecture
4. Short exercise with new groups

Course laptops

Come grab a laptop if you'd like one, if we don't have enough then let's try to prioritize folks whose laptops may not be able to keep up

Learning objectives

1. Understand how computers produce seemingly random data
2. Learn about various probability distributions
3. Understand what matrices are in R
4. Learn how to use the apply() function

Discussion question

Many applications in computational biology require modeling stochastic processes:

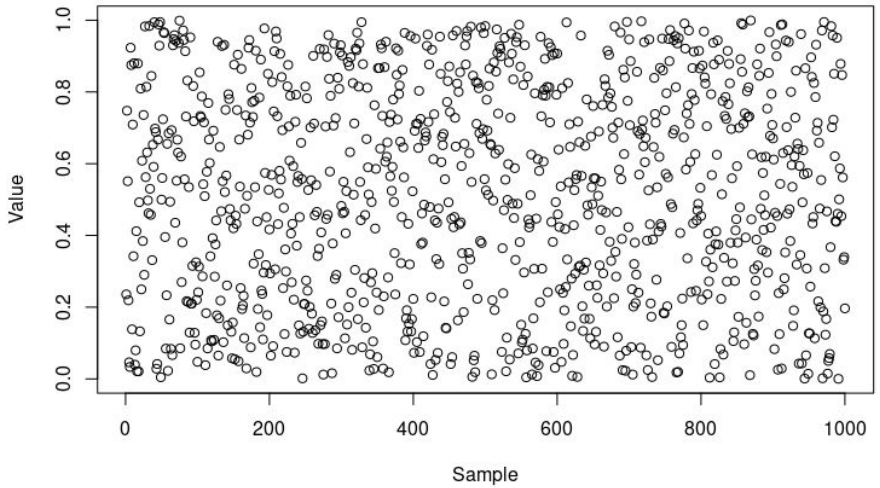
What are some stochastic processes in biology?
What are some processes that can be modeled as stochastic?
What is the difference between these two?

You have ~5 minutes to discuss this in your groups, I will call on one group to volunteer their answer

“Pseudo”random number generation

Rather than generating truly random data, computers generate pseudorandom data. In other words, they use nonrandom (deterministic) generators that create sequences of numbers with properties which approximate idealized random numbers.

“Pseudo”random number generation



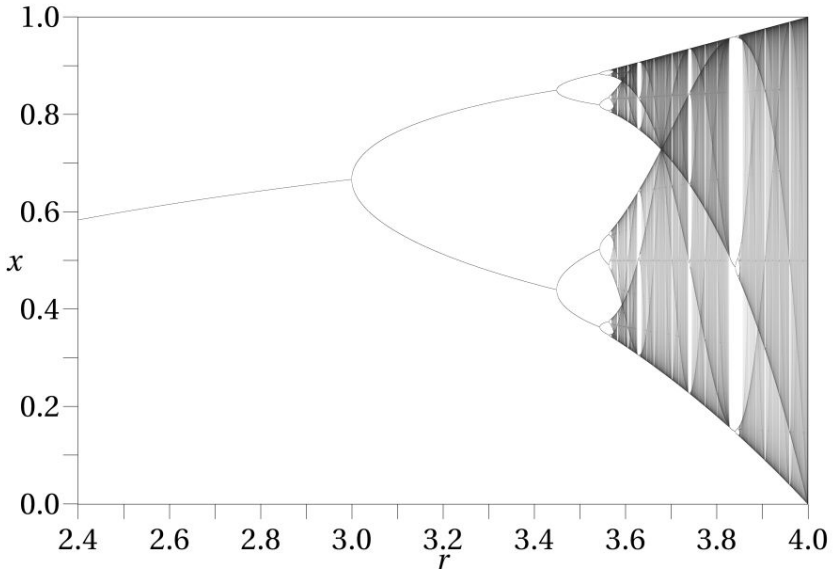
Low autocorrelation, lack of periodicity, uniformity of distribution

Using chaos instead of randomness

Chaos occurs when the present determines the future (deterministic), but the approximate present does not determine the approximate future

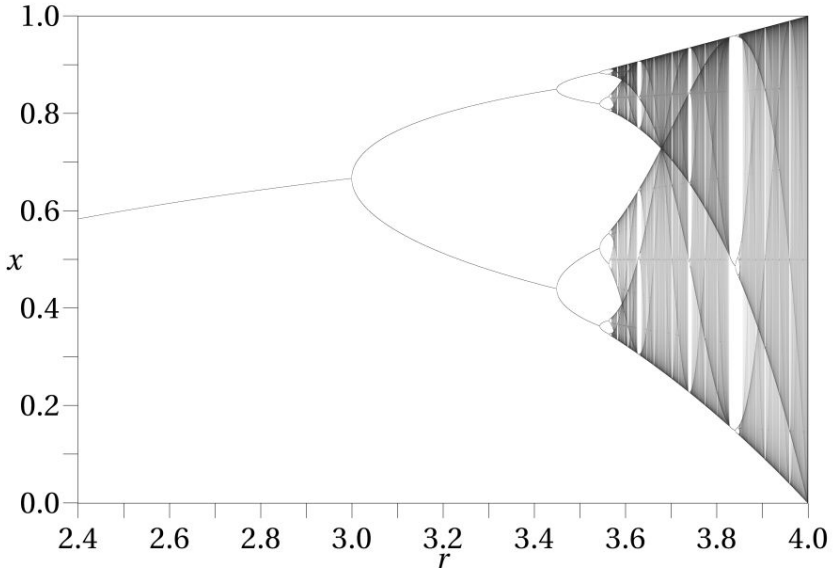
Logistic growth curve

$x_{t+1} = r x_t (1 - x_t)$, r = growth rate



Using chaos instead of randomness

Chaos occurs when the present determines the future (deterministic), but the approximate present does not determine the approximate future



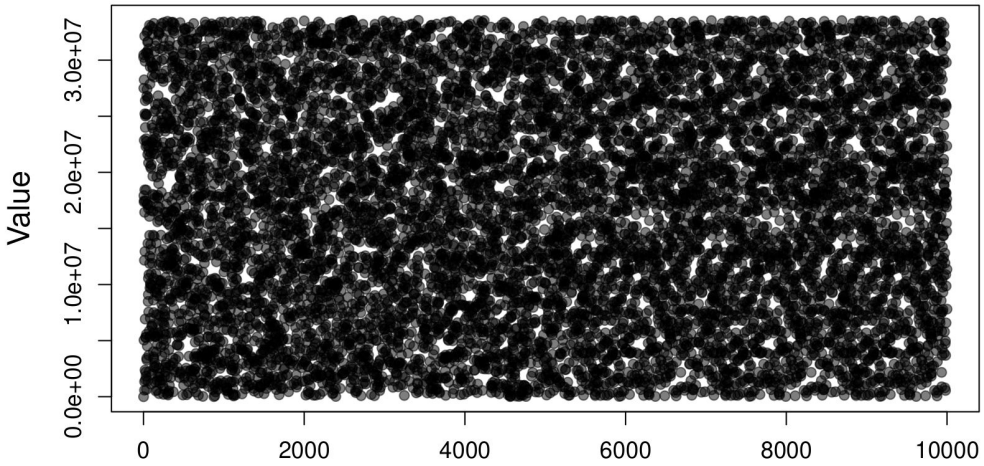
Demonstration of chaos

Due to the laws of physics, there is no randomness guiding the pendulums. However, their trajectories are wildly different due to small differences in their starting conditions.

Note how large the differences in dynamics become over time.

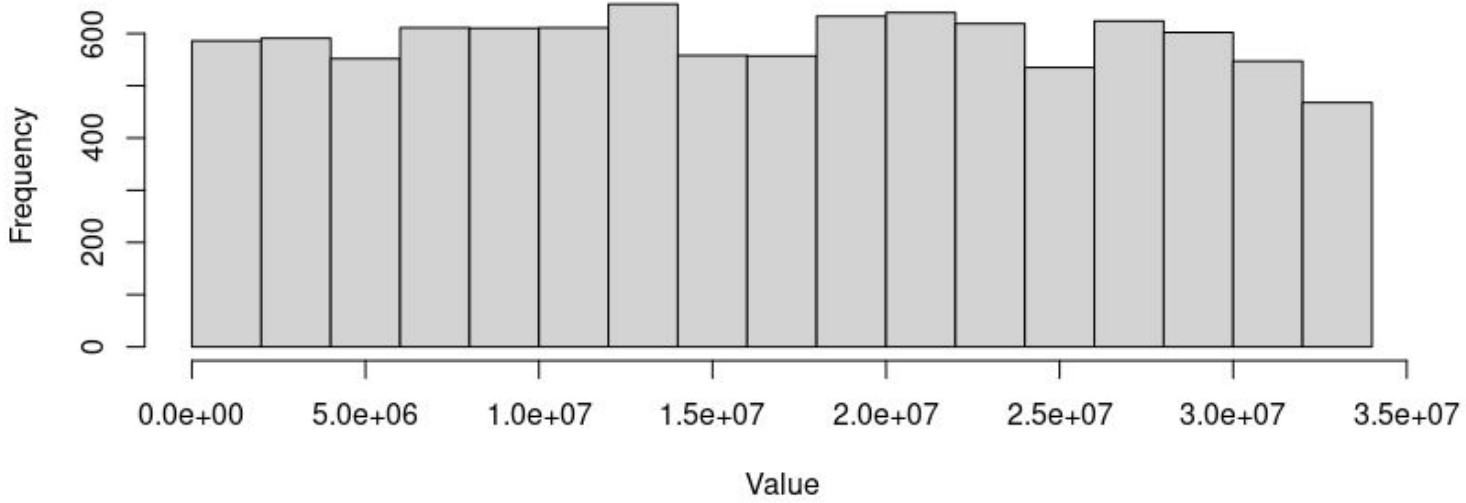
Link to video

Computers use deterministic, chaotic functions to generate pseudorandom numbers



Xn+1 = (aXn + c)modM, a = 25214903917, c = 1, M = 225. Linear Congruential Generator [Li et al., 2020]

Computers use deterministic, chaotic functions to generate pseudorandom numbers



Xn+1 = (aXn + c)modM, a = 25214903917, c = 1, M = 225. Linear Congruential Generator [Li et al., 2020]

Probability is the mathematical language of uncertainty

Where does uncertainty come from?

1. Some processes are just inherently stochastic (quantum mechanics??)
2. Deterministic processes can appear stochastic with limited knowledge
3. Measurement error, incomplete data
4. Inferences based on samples, instead of using the whole population
5. Etc.

Basics of probability functions

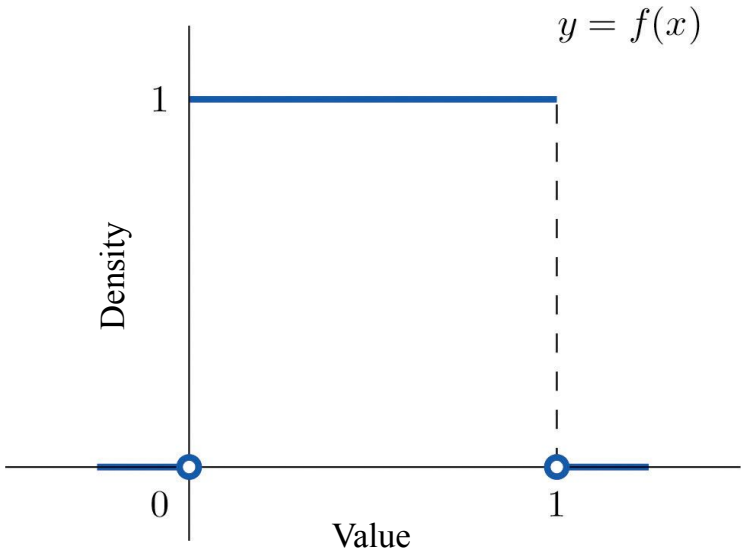
- A random variable is a variable whose value is subject to randomness or uncertainty
- A probability function characterizes uncertainty in the random variable
- Random variables can be discrete or continuous
- Probability functions sum (discrete) or integrate (continuous) to 1

R has functions that allow us to draw samples from common probability distributions

To draw a sample from a uniform distribution:

```
runif(n = 20, min = 0, max = 1)
```

This code samples 20 values from a uniform, bounded by 0 and 1



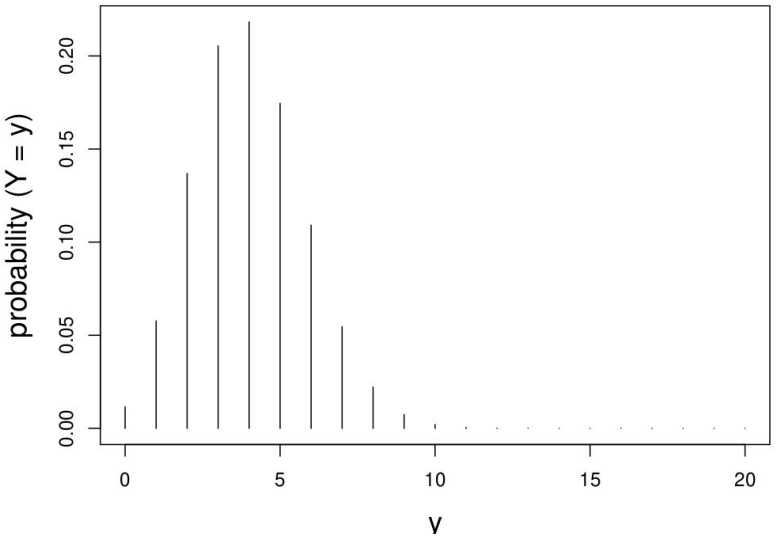
The binomial probability distribution

- A Bernoulli process is a process with two possible outcomes, eg. success or failure
- Each outcome is associated with a probability, denoted *p* and *1-p*.
- A binomial process is a series of independent Bernoulli processes (or trials) with a constant value for *p*

Take a minute to discuss among your group:
Can you think of any Bernoulli processes in biology, or even more broadly?

The binomial probability distribution

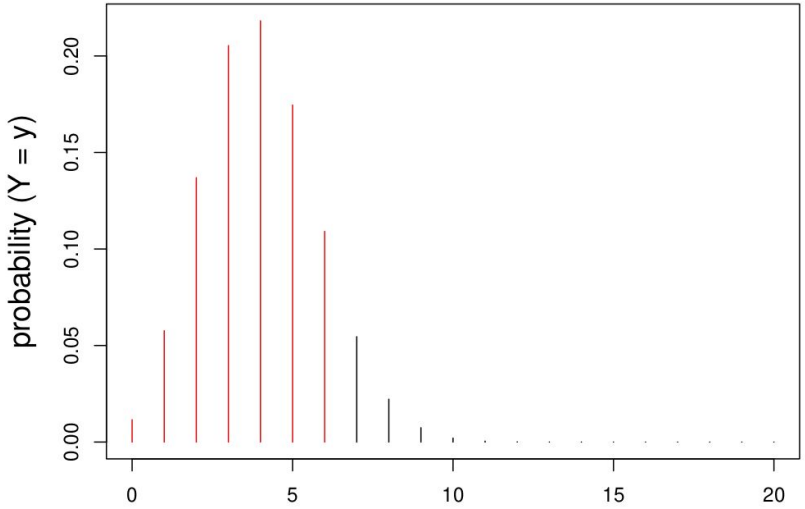
The binomial mass probability function (PMF) gives the probability of getting y successes in n trial



$$P(Y = y|p,n) = \binom{n}{k} p^k * (1-p)^{n-k}$$

The binomial probability distribution

The binomial mass probability function (PMF) gives the probability of getting y successes in n trial

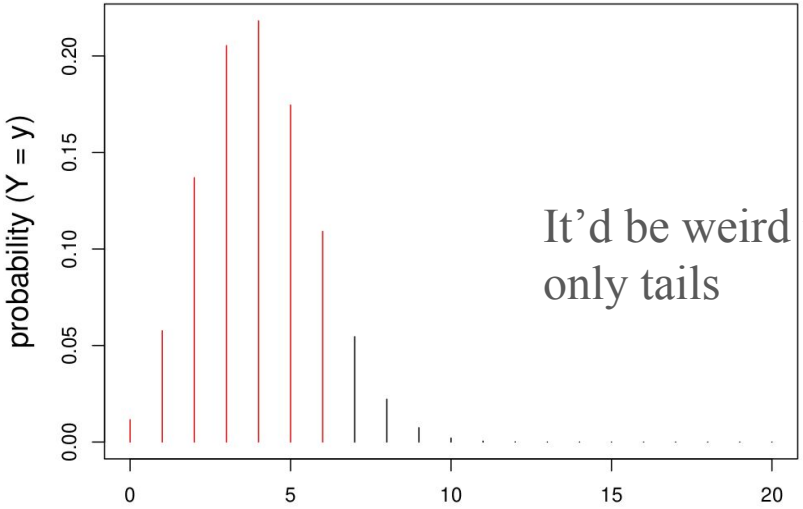


$$P(Y = y|p,n) = \binom{n}{k} p^k * (1-p)^{n-k}$$

The binomial probability distribution

The binomial mass probability function (PMF) gives the probability of getting y successes in n trial

It'd be weird to get only heads

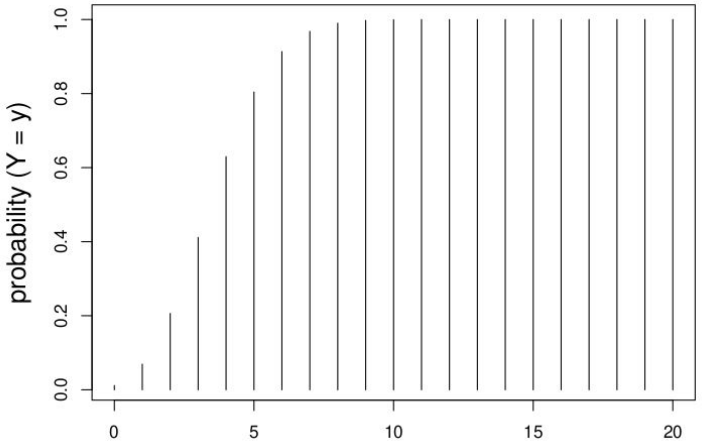


It'd be weird to get only tails

$$P(Y = y|p,n) = \binom{n}{k} p^k * (1-p)^{n-k}$$

The binomial probability distribution

The cumulative distribution function is related to the PMF, but gives the probability that Y ≤ y

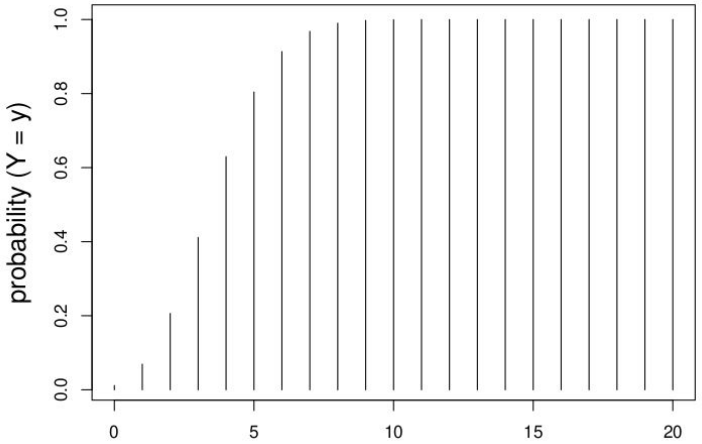


$$P(Y \leq y|p,n) = \sum_{i=0}^{y} \binom{n}{i} p^i (1-p)^{n-i}$$

The binomial probability distribution

The cumulative distribution function is related to the PMF, but gives the probability that Y ≤ y

You should be able to get more than 2 heads



$$P(Y \leq y|p,n) = \sum_{i=0}^{y} \binom{n}{i} p^i (1-p)^{n-i}$$

Working with the binomial distribution in R

- dbinom(x, size, prob) is the probability mass (density for continuous probability distributions)
- pbinom(q, size, prob) is the cumulative distribution function
- rbinom(n, size, prob) is used to draw random deviates from the probability distribution
- qbinom(p, size, prob) is the quantile function, y is the Nth quantile of a distribution if $P(Y \leq y) = N$

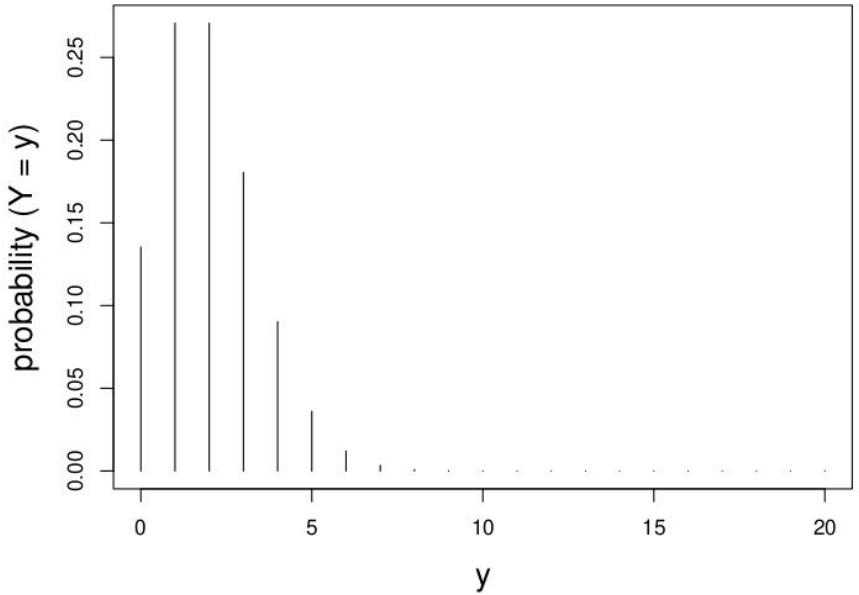
The Poisson probability distribution

A Poisson process arises when events occur at a constant rate per unit time or space (rate = λ)

Can you think of any examples of Poisson processes in biology (or elsewhere)? Discuss among your group

The Poisson probability distribution

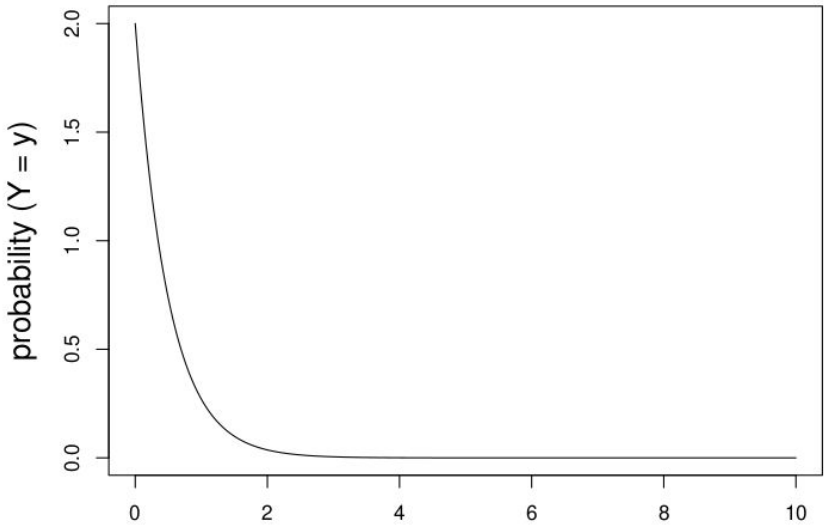
A Poisson process arises when events occur at a constant rate per unit time or space (rate = λ)



$$P(Y = y) = \frac{\lambda^y e^{-\lambda}}{y!}$$

The exponential probability distribution

The exponential distribution describes the time between events in a Poisson process

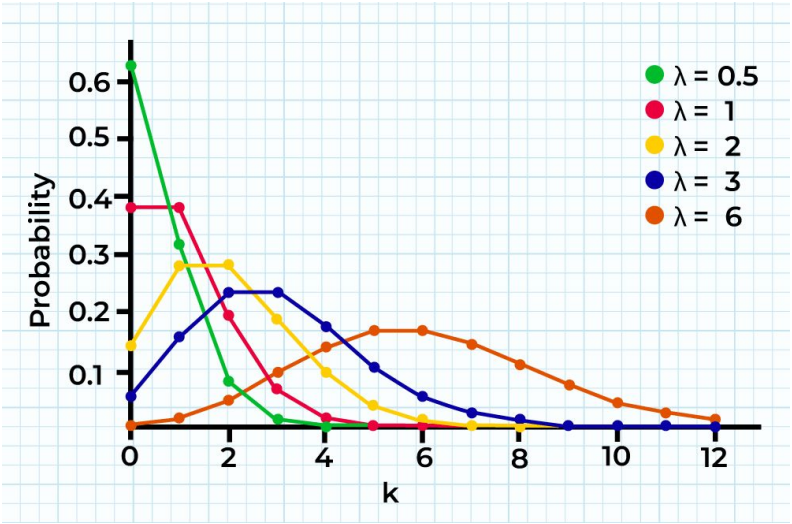


$$P(Y = y) = \lambda e^{-\lambda y}$$

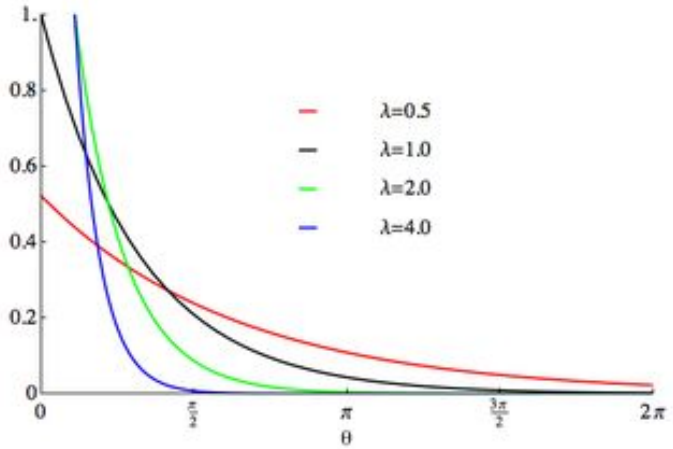
The exponential probability distribution

Changing the event rate (λ) alters our probability distributions

Poisson



Exponential

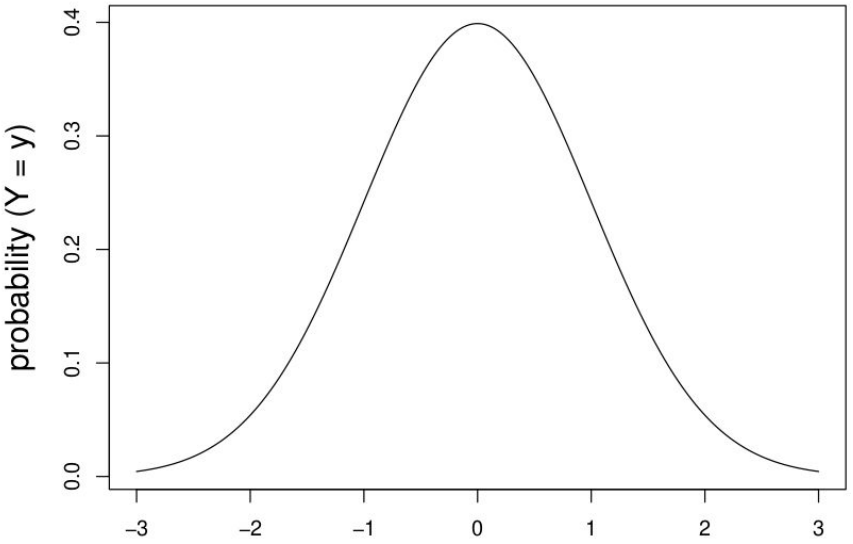


The normal (Gaussian) probability distribution

Normal distributions arise when working with random variables that are the sum of a large number of other random variables regardless of their distribution (central limit theorem).

Can you think of any examples of random variables in biology that should follow a normal distribution based on the description above?

The normal (Gaussian) probability distribution



$$P(Y = y|\mu, \sigma) = \frac{1}{\sigma\sqrt{(2\pi)}} e^{-\frac{1}{2}\left(\frac{y-\mu}{\sigma}\right)^2}$$

Similar syntax for each of the probability distributions we have considered

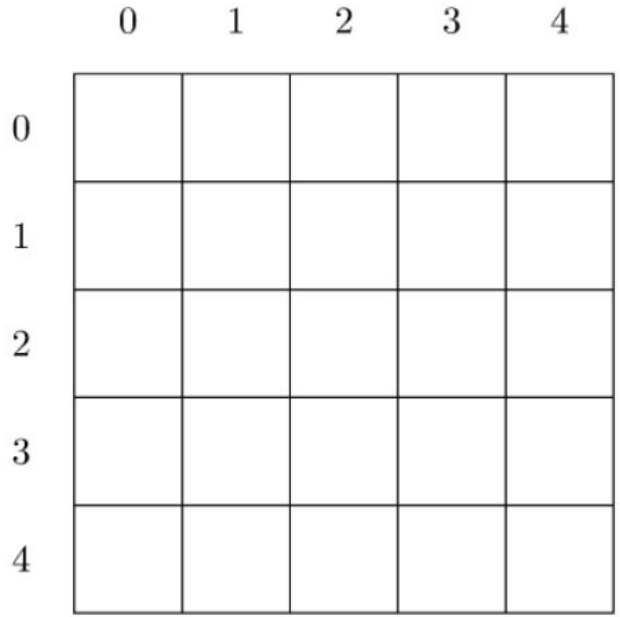
- Uniform: `dunif, punif, runif, qunif`
- Binomial: `dbinom, pbinom, rbinom, qbinom`
- Poisson: `dpois, ppois, rpois, qpois`
- Exponential: `dexp, pexp, rexp, qexp`
- Normal: `dnorm, pnorm, rnorm, qnorm`

Learning objectives

1. Understand how computers produce seemingly random data
2. Learn about various probability distributions
3. Understand what matrices are in R
4. Learn how to use the apply() function

Matrices in R

A matrix is a two dimensional data structure with rows and columns



Working with matrices in R

To create a matrix:

```
X<-matrix(NA, nrow=5, ncol = 5, byrow=TRUE)
```

To query the dimensions of a matrix (row then column):

```
dim(X)
```

Indexing matrices:

```
X[1,2]
X[,1]
X[3:4,]
```

Applying functions to matrices

The apply function applies a function to every row (1) or column (2) in a matrix:

```
X<-matrix(1:25, nrow=5, ncol = 5, byrow=TRUE)
```

Compute mean of each row

```
apply(X,MARGIN=1,mean)
```

Summary function for each column

```
apply(X,MARGIN=2,summary)
```

Closing exercise

Here are three closing questions to check you understanding. Try to work through these in your groups. Then paste the code you used for each into the "Probability and matrices" discussion on canvas.

1. Create a 5 (rows) by 10 (columns) matrix filled with 50 "random" numbers from a uniform distribution bounded by 0 and 100.
2. Compute and display the mean and standard deviation of the 10 numbers in each row of the matrix you created for (1).
3. Create a new matrix from the first 3 rows and first 7 columns of the matrix from (1). Add 10 to all of the values in the matrix.