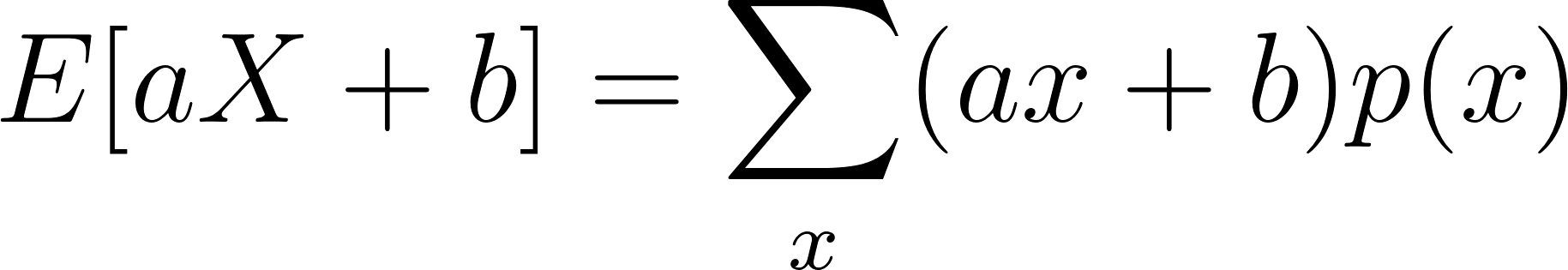
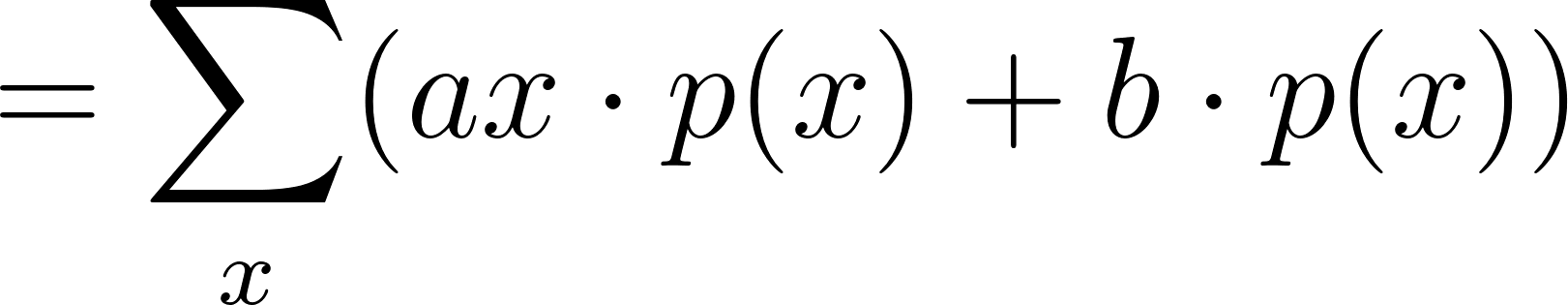
# Expectation and Variance

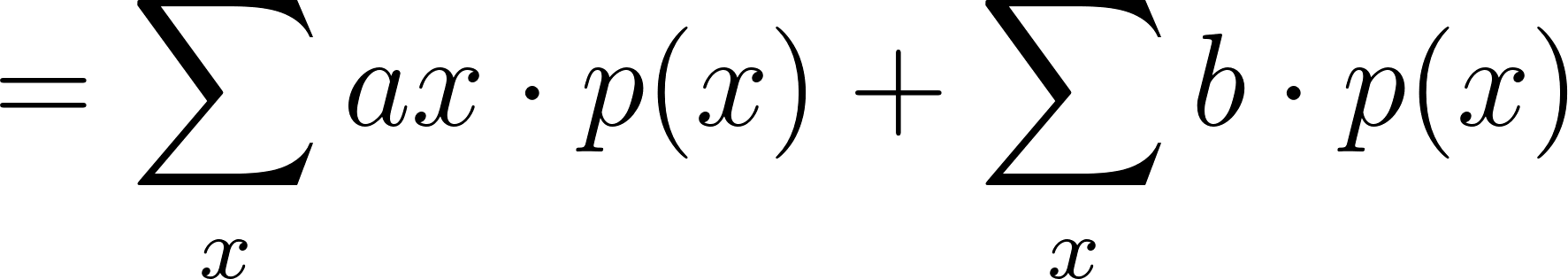
## E[aX + b] = aE[X] + b

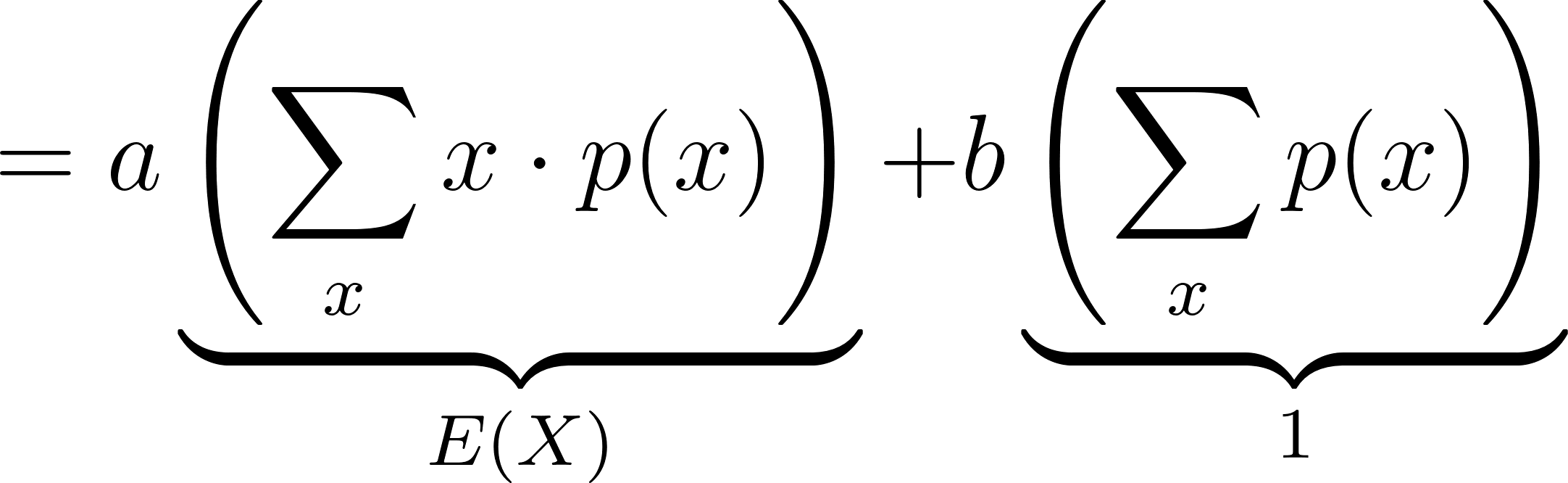
### Proof

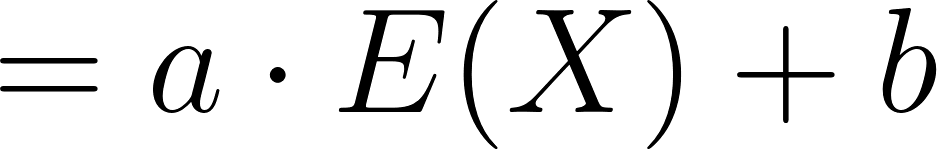
Discrete Random Variable:

[](https://www.codecogs.com/eqnedit.php?latex=E%5BaX%2Bb%5D%3D%20%5Csum_%7Bx%7D(ax%2Bb)p(x)#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3D%5Csum_%7Bx%7D(ax%5Ccdot%20p(x)%2Bb%5Ccdot%20p(x))#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3D%5Csum_%7Bx%7Dax%5Ccdot%20p(x)%20%2B%20%5Csum_%7Bx%7Db%5Ccdot%20p(x)#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3Da%5Cunderbrace%7B%5Cleft(%5Csum_%7Bx%7Dx%5Ccdot%20p(x)%5Cright)%7D_%7BE(X)%7D%20%2B%20b%5Cunderbrace%7B%5Cleft(%5Csum_%7Bx%7Dp(x)%5Cright)%7D_%7B1%7D#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3Da%5Ccdot%20E(X)%20%2B%20b%20%5C%5C#0)

The same process can be done with a continuous random variable and integration.

### R Code

*# Set the constants a and b*  
a <- 2  
b <- 3  
  
*# Set the number of samples*  
n <- 100000  
set.seed(6644)  
  
*# Discrete Distributions*  
  
*# 1. Poisson Distribution*  
X <- rpois(n, lambda = 2) *# generate samples*  
exp\_pois <- mean(a\*X + b) *# calculate expected value*  
exp\_pois\_check <- a\*mean(X) + b *# check property*  
print(paste("Poisson: ", round(exp\_pois, 5), " vs ",   
 round(exp\_pois\_check, 5)))

## [1] "Poisson: 6.9885 vs 6.9885"

*# 2. Binomial Distribution*  
X <- rbinom(n, size = 10, prob = 0.5) *# generate samples*  
exp\_binom <- mean(a\*X + b) *# calculate expected value*  
exp\_binom\_check <- a\*mean(X) + b *# check property*  
print(paste("Binomial: ", round(exp\_binom, 5), " vs ",   
 round(exp\_binom\_check, 5)))

## [1] "Binomial: 13.01288 vs 13.01288"

*# 3. Custom Discrete Distribution*  
probs <- c(0.2, 0.3, 0.5) *# probabilities for each outcome*  
outcomes <- c(1, 2, 3) *# possible outcomes*  
X <- sample(outcomes, size = n, replace = TRUE, prob = probs)*# generate samples*  
exp\_custom <- mean(a\*X + b) *# calculate expected value*  
exp\_custom\_check <- a\*mean(X) + b *# check property*  
print(paste("Custom Discrete: ", round(exp\_custom, 5), " vs ",   
 round(exp\_custom\_check, 5)))

## [1] "Custom Discrete: 7.59696 vs 7.59696"

*# Continuous Distributions*  
  
*# 1. Normal Distribution*  
X <- rnorm(n, mean = 0, sd = 1) *# generate samples*  
exp\_norm <- mean(a\*X + b) *# calculate expected value*  
exp\_norm\_check <- a\*mean(X) + b *# check property*  
print(paste("Normal: ", round(exp\_norm, 5), " vs ",   
 round(exp\_norm\_check, 5)))

## [1] "Normal: 3.00359 vs 3.00359"

*# 2. Exponential Distribution*  
X <- rexp(n, rate = 1) *# generate samples*  
exp\_exp <- mean(a\*X + b) *# calculate expected value*  
exp\_exp\_check <- a\*mean(X) + b *# check property*  
print(paste("Exponential: ", round(exp\_exp, 5), " vs ",   
 round(exp\_exp\_check, 5)))

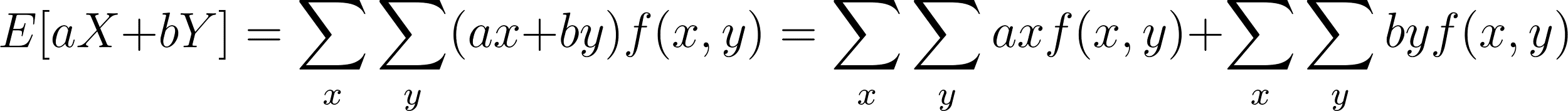
## [1] "Exponential: 4.99792 vs 4.99792"

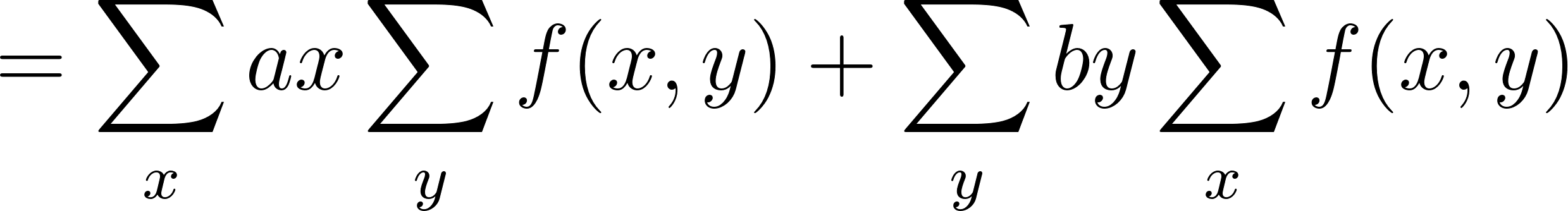
*# 3. Uniform Distribution*  
X <- runif(n, min = 0, max = 1) *# generate samples*  
exp\_unif <- mean(a\*X + b) *# calculate expected value*  
exp\_unif\_check <- a\*mean(X) + b *# check property*  
print(paste("Uniform: ", round(exp\_unif, 5), " vs ",   
 round(exp\_unif\_check, 5)))

## [1] "Uniform: 3.99998 vs 3.99998"

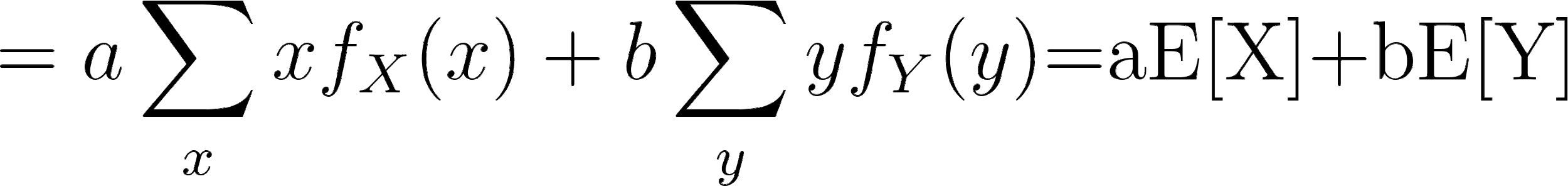
## E[aX + bY] = aE[X] + bE[Y]

### Proof

[](https://www.codecogs.com/eqnedit.php?latex=E%5BaX%2BbY%5D%3D%5Csum_x%20%5Csum_y%20(ax%2Bby)%20f(x%2Cy)%3D%5Csum_x%20%5Csum_y%20axf(x%2Cy)%2B%20%5Csum_x%20%5Csum_y%20byf(x%2Cy)#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3D%5Csum_x%20ax%20%5Csum_y%20f(x%2Cy)%20%2B%20%5Csum_y%20by%20%5Csum_x%20f(x%2Cy)#0)

The marginal PMFs are in this expression, so we can rewrite it as:

[](https://latex-staging.easygenerator.com/eqneditor/editor.php?latex=%3Da%5Csum_x%20xf_X(x)%2Bb%5Csum_y%20y%20f_Y(y)%24%3DaE%5BX%5D%2BbE%5BY%5D#0)

### R Code

*# Set the constants a and b*  
a <- 2  
b <- 3  
  
*# Set the number of samples*  
n <- 100000  
  
*# Set the seed for reproducibility*  
set.seed(123)  
  
*# Discrete Distributions*  
  
*# 1. Poisson and Binomial Distribution*  
X\_discrete <- rpois(n, lambda = 2) *# generate samples from Poisson*  
Y\_discrete <- rbinom(n, size = 10, prob = 0.5) *# generate samples from Binomial*  
exp\_discrete <- mean(a\*X\_discrete + b\*Y\_discrete) *# calculate expected value*  
exp\_discrete\_check <- a\*mean(X\_discrete) + b\*mean(Y\_discrete) *# check property*  
print(paste("Discrete (Poisson & Binomial): ", round(exp\_discrete, 5), " vs ",   
 round(exp\_discrete\_check, 5)))

## [1] "Discrete (Poisson & Binomial): 18.99106 vs 18.99106"

*# Continuous Distributions*  
  
*# 2. Normal and Uniform Distribution*  
X\_cont <- rnorm(n, mean = 0, sd = 1) *# generate samples from Normal*  
Y\_cont <- runif(n, min = 0, max = 1) *# generate samples from Uniform*  
exp\_cont <- mean(a\*X\_cont + b\*Y\_cont) *# calculate expected value*  
exp\_cont\_check <- a\*mean(X\_cont) + b\*mean(Y\_cont) *# check property*  
print(paste("Continuous (Normal & Uniform): ", round(exp\_cont, 5), " vs ",   
 round(exp\_cont\_check, 5)))

## [1] "Continuous (Normal & Uniform): 1.50904 vs 1.50904"

*# Mixed Distributions*  
  
*# 3. Poisson (Discrete) and Normal (Continuous) Distribution*  
X\_mixed <- X\_discrete *# using previously generated Poisson samples*  
Y\_mixed <- X\_cont *# using previously generated Normal samples*  
exp\_mixed <- mean(a\*X\_mixed + b\*Y\_mixed) *# calculate expected value*  
exp\_mixed\_check <- a\*mean(X\_mixed) + b\*mean(Y\_mixed) *# check property*  
print(paste("Mixed (Poisson & Normal): ", round(exp\_mixed, 5), " vs ",   
 round(exp\_mixed\_check, 5)))

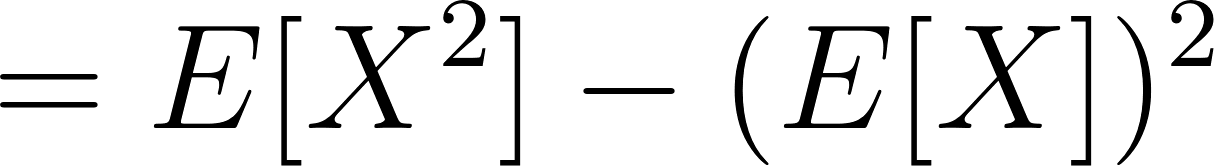
## [1] "Mixed (Poisson & Normal): 4.01115 vs 4.01115"

## Var(X) = E[X^2] - (E[X])^2

### Proof

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BVar%7D(X)%3DE%5B(X%20-%20E%5BX%5D)%5E2%5D%3DE%5BX%5E2%20-%202XE%5BX%5D%2B(E%5BX%5D)%5E2%5D#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5BX%5E2%5D-2E%5BX%5DE%5BX%5D%2B(E%5BX%5D)%5E2%3DE%5BX%5E2%5D-2(E%5BX%5D)%5E2%2B(E%5BX%5D)%5E2#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5BX%5E2%5D-(E%5BX%5D)%5E2#0)

This is equivalent to the covariance of [](https://www.codecogs.com/eqnedit.php?latex=X#0) with itself.

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BCov%7D(X%2CX)%3DE%5B(X-E%5BX%5D)(X-E%5BX%5D)%5D%3DE%5B(X-E%5BX%5D)%5E2%5D%3D%5Ctext%7BVar%7D(X)#0)

### R Code

*# Set the number of samples*  
n <- 100000  
  
*# Set the seed for reproducibility*  
set.seed(123)  
  
*# Generate samples from Normal distribution*  
X <- rnorm(n, mean = 0, sd = 1)  
  
*# Calculate expected value of X*  
E\_X <- mean(X)  
  
*# Calculate variance using the definition:*   
*# Var(X) = Sum((X\_i - E[X])^2) / (n - 1)*  
var\_sum <- sum((X - E\_X)^2) / n  
print(paste("Variance using summation definition: ", round(var\_sum, 5)))

## [1] "Variance using summation definition: 0.99946"

*# Calculate variance using the definition: Var(X) = E[X^2] - (E[X])^2*  
var\_def <- mean(X^2) - E\_X^2  
print(paste("Variance using E[X^2] - (E[X])^2: ", round(var\_def, 5)))

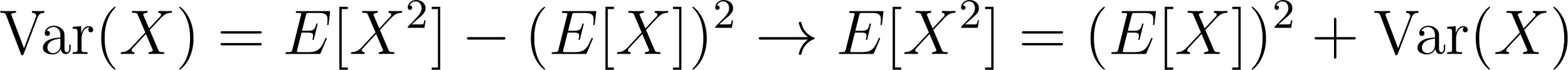
## [1] "Variance using E[X^2] - (E[X])^2: 0.99946"

*# Calculate variance as the average of squared deviations from the mean*  
var\_cov <- sum((X - E\_X) \* (X - E\_X)) / n  
print(paste("Variance as Cov(X, X): ", round(var\_cov, 5)))

## [1] "Variance as Cov(X, X): 0.99946"

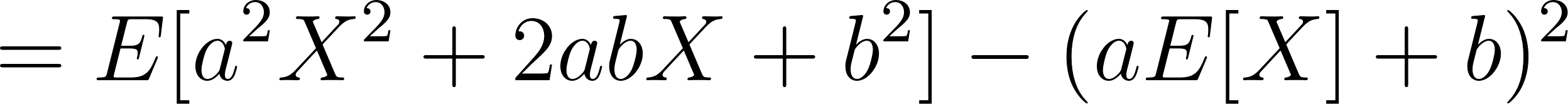
### E[X^2] = (E[X])^2 + Var(X)

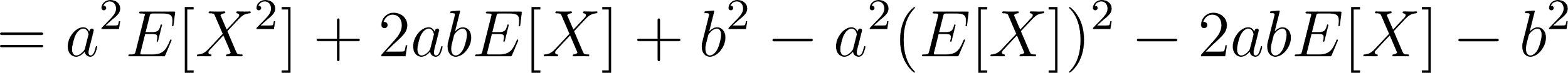
This result is obtained by rearranging the equality obtained in the previous proof.

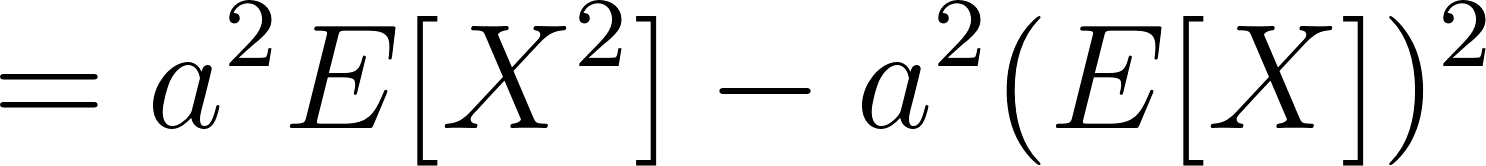
[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BVar%7D(X)%3DE%5BX%5E2%5D-(E%5BX%5D)%5E2%20%5Crightarrow%20E%5BX%5E2%5D%3D(E%5BX%5D)%5E2%20%2B%20%5Ctext%7BVar%7D(X)#0)

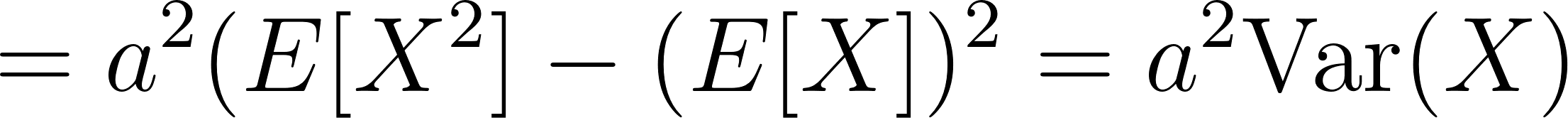
## Var(aX + b) = a^2 Var(X)

### Proof

[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5Ba%5E2X%5E2%2B2abX%2Bb%5E2%5D-(aE%5BX%5D%2Bb)%5E2#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3Da%5E2E%5BX%5E2%5D%2B2abE%5BX%5D%2Bb%5E2-a%5E2(E%5BX%5D)%5E2-2abE%5BX%5D-b%5E2#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3Da%5E2E%5BX%5E2%5D-a%5E2(E%5BX%5D)%5E2#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3Da%5E2(E%5BX%5E2%5D-(E%5BX%5D)%5E2%3Da%5E2%5Ctext%7BVar%7D(X)#0)

### R Code

*# Set the constants a and b*  
a <- 2  
b <- 3  
  
*# Set the number of samples*  
n <- 100000  
  
*# Set the seed for reproducibility*  
set.seed(123)  
  
*# Discrete Distributions*  
  
*# 1. Poisson Distribution*  
X\_discrete <- rpois(n, lambda = 2) *# generate samples*  
var\_discrete <- var(a\*X\_discrete + b) *# calculate variance*  
var\_discrete\_check <- a^2 \* var(X\_discrete) *# check property*  
print(paste("Discrete (Poisson): ", round(var\_discrete, 5), " vs ",   
 round(var\_discrete\_check, 5)))

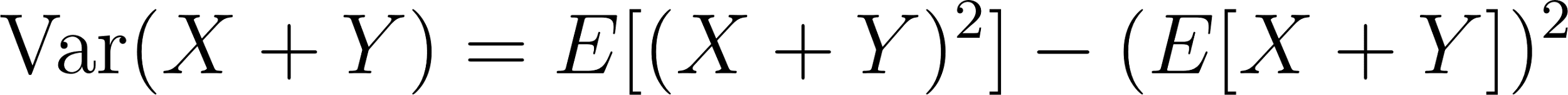
## [1] "Discrete (Poisson): 7.98602 vs 7.98602"

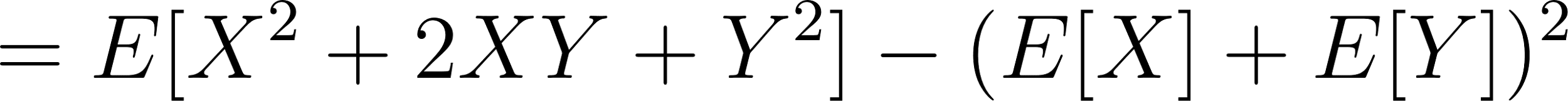
*# Continuous Distributions*  
  
*# 2. Normal Distribution*  
X\_cont <- rnorm(n, mean = 0, sd = 1) *# generate samples*  
var\_cont <- var(a\*X\_cont + b) *# calculate variance*  
var\_cont\_check <- a^2 \* var(X\_cont) *# check property*  
print(paste("Continuous (Normal): ", round(var\_cont, 5), " vs ",   
 round(var\_cont\_check, 5)))

## [1] "Continuous (Normal): 4.01607 vs 4.01607"

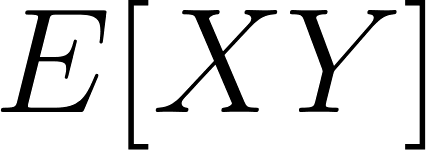
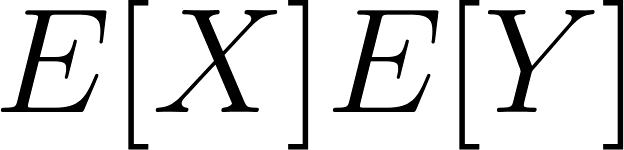
## Var(X + Y) = Var(X) + Var(Y) (X, Y independent)

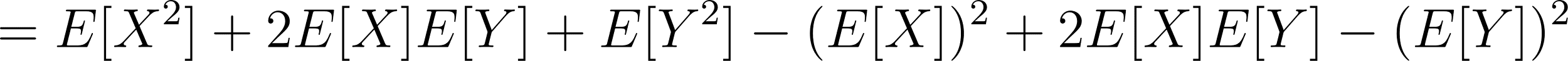
### Proof

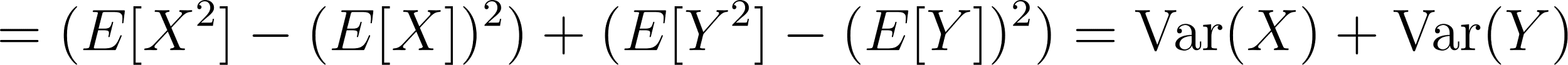
[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BVar%7D(X%2BY)%3DE%5B(X%2BY)%5E2%5D-(E%5BX%2BY%5D)%5E2#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5BX%5E2%20%2B%202XY%20%2B%20Y%5E2%5D-(E%5BX%5D%20%2B%20E%5BY%5D)%5E2#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5BX%5E2%5D%2B2E%5BXY%5D%2BE%5BY%5E2%5D-%5Cbig%5B(E%5BX%5D)%5E2%20%2B%202E%5BX%5DE%5BY%5D%20%2B%20(E%5BY%5D)%5E2%20%5Cbig%5D#0)

The term [](https://www.codecogs.com/eqnedit.php?latex=E%5BXY%5D#0) can be rewritten as [](https://www.codecogs.com/eqnedit.php?latex=E%5BX%5DE%5BY%5D#0) since [](https://www.codecogs.com/eqnedit.php?latex=X#0) and [](https://www.codecogs.com/eqnedit.php?latex=Y#0) are independent.

[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5BX%5E2%5D%2B2E%5BX%5DE%5BY%5D%2BE%5BY%5E2%5D-(E%5BX%5D)%5E2%2B2E%5BX%5DE%5BY%5D-(E%5BY%5D)%5E2#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3D(E%5BX%5E2%5D-(E%5BX%5D)%5E2)%2B(E%5BY%5E2%5D-(E%5BY%5D)%5E2)%3D%5Ctext%7BVar%7D(X)%2B%5Ctext%7BVar%7D(Y)#0)

### R Code

We're assuming X and Y are independent because they are generated separately, and there should be no systematic relationship between them. The covariance between the two sets of random numbers should be approximately zero due to the way they were generated, but due to random sampling error, it might not be exactly zero.

*# Set the number of samples*  
n <- 100000  
  
*# Set the seed for reproducibility*  
set.seed(123)  
  
*# Generate samples from Poisson and Normal distributions*  
X <- rpois(n, lambda = 2)   
Y <- rnorm(n, mean = 0, sd = 1)  
  
*# Calculate variance of X + Y*  
var\_X\_plus\_Y <- var(X + Y)  
  
*# Calculate variance of X and Y*  
var\_X <- var(X)  
var\_Y <- var(Y)  
  
*# Check property Var(X + Y) = Var(X) + Var(Y)*  
var\_check <- var\_X + var\_Y  
  
print(paste("Variance of X + Y: ", round(var\_X\_plus\_Y, 5)))

## [1] "Variance of X + Y: 2.98958"

print(paste("Sum of Variances of X and Y: ", round(var\_check, 5)))

## [1] "Sum of Variances of X and Y: 3.00052"

print(paste("Covariance (random): ", round(cov(X, Y), 5)))

## [1] "Covariance (random): -0.00547"

### Why Do We Always Add Variances?

The following excerpt is taken from [this site](https://apcentral.collegeboard.org/courses/ap-statistics/classroom-resources/why-variances-add-and-why-it-matters).

We bought some cereal. The box says “16 ounces.” We know that’s not precisely the weight of the cereal in the box, just close. After all, one corn flake more or less would change the weight ever so slightly. Weights of such boxes of cereal vary somewhat, and our uncertainty about the exact weight is expressed by the variance (or standard deviation) of those weights.

Next we get out a bowl that holds 3 ounces of cereal and pour it full. Our pouring skill is not very precise, so the bowl now contains about 3 ounces with some variability (uncertainty).

How much cereal is left in the box? Well, we assume about 13 ounces. But notice that we’re less certain about this remaining weight than we were about the weight before we poured out the bowlful. The variability of the weight in the box has increased even though we subtracted cereal.

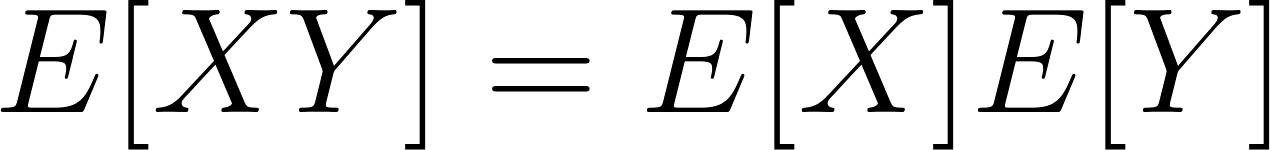
Moral: Every time something happens at random, whether it adds to the pile or subtracts from it, uncertainty (read “variance”) increases.

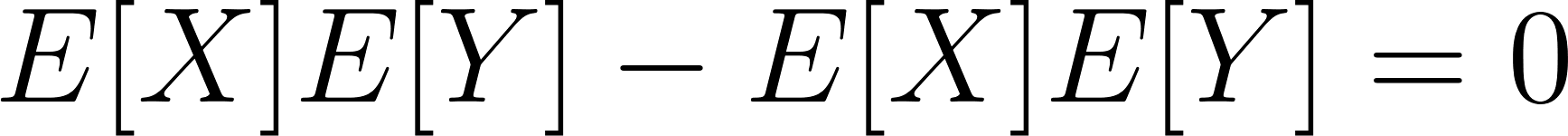
## Cov(X, Y) = E[XY] - E[X]E[Y]

### Proof

[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5BXY-XE%5BY%5D-YE%5BX%5D%2BE%5BX%5DE%5BY%5D#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5BXY%5D-E%5BX%5DE%5BY%5D-E%5BX%5DE%5BY%5D%2BE%5BX%5DE%5BY%5D%3DE%5BXY%5D-E%5BX%5DE%5BY%5D#0)

If [](https://www.codecogs.com/eqnedit.php?latex=X#0) and [](https://www.codecogs.com/eqnedit.php?latex=Y#0) are independent, then [](https://www.codecogs.com/eqnedit.php?latex=E%5BXY%5D%3DE%5BX%5DE%5BY%5D#0) and the expression above becomes

[](https://www.codecogs.com/eqnedit.php?latex=E%5BX%5DE%5BY%5D-E%5BX%5DE%5BY%5D%3D0#0)

### R Code

*# Set the number of samples*  
n <- 100000  
  
*# Set the seed for reproducibility*  
set.seed(123)  
  
*# Generate samples from Normal distribution for X*  
X <- rnorm(n, mean = 0, sd = 1)  
  
*# Generate Y as a function of X plus some independent random noise*  
Y <- 2 \* X + rpois(n, lambda = 2)  
  
*# Calculate covariance using the definition: Cov(X, Y) = E[XY] - E[X]E[Y]*  
cov\_def <- mean(X \* Y) - mean(X) \* mean(Y)  
print(paste("Covariance using definition: ", round(cov\_def, 5)))

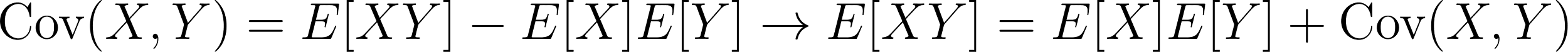
## [1] "Covariance using definition: 2.0037"

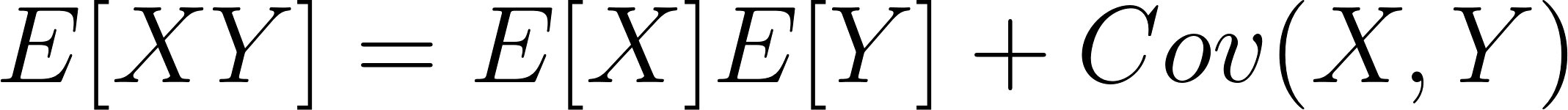
*# Calculate covariance using the built-in R function for comparison*  
cov\_calc <- sum((X - mean(X))\*(Y - mean(Y))) / n  
print(paste("Covariance calculation: ", round(cov\_calc, 5)))

## [1] "Covariance calculation: 2.0037"

### E[XY] = E[X]E[Y] + Cov(X, Y)

This result is obtained by rearranging the equality obtained in the previous proof.

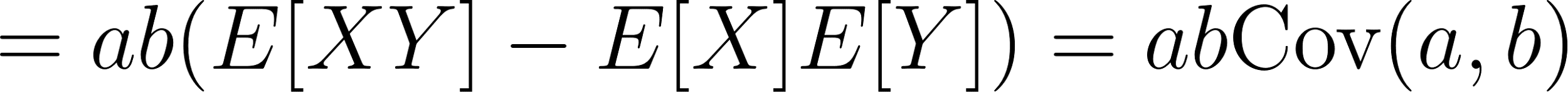
[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BCov%7D(X%2CY)%3DE%5BXY%5D-E%5BX%5DE%5BY%5D%20%5Crightarrow%20E%5BXY%5D%3DE%5BX%5DE%5BY%5D%20%2B%20%5Ctext%7BCov%7D(X%2CY)#0)

[](https://www.codecogs.com/eqnedit.php?latex=E%5BXY%5D%20%3D%20E%5BX%5DE%5BY%5D%2B%20Cov(X%2CY)%20#0)

## Cov(aX, bY) = abCov(X,Y)

### Proof

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BCov%7D(aX%2C%20bY)%3DE%5BaXbY%5D-E%5BaX%5DE%5BbY%5D%3DabE%5BXY%5D-abE%5BX%5DE%5BY%5D#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3Dab(E%5BXY%5D-E%5BX%5DE%5BY%5D)%3Dab%5Ctext%7BCov%7D(a%2Cb)#0)

### R Code

*# Set the constants a and b*  
a <- 2  
b <- 3  
  
*# Set the number of samples*  
n <- 100000  
  
*# Set the seed for reproducibility*  
set.seed(123)  
  
*# Generate samples from Normal distribution for X*  
X <- rnorm(n, mean = 0, sd = 1)  
  
*# Generate Y as a function of X plus some independent random noise*  
Y <- X + rpois(n, lambda = 2)  
  
*# Calculate covariance using the definition: Cov(aX, bY) =*   
*# E[aX \* bY] - E[aX]E[bY]*  
cov\_abXY <- mean(a \* X \* b \* Y) - mean(a \* X) \* mean(b \* Y)  
print(paste("Covariance of aX and bY using definition: ",   
 round(cov\_abXY, 5)))

## [1] "Covariance of aX and bY using definition: 6.02548"

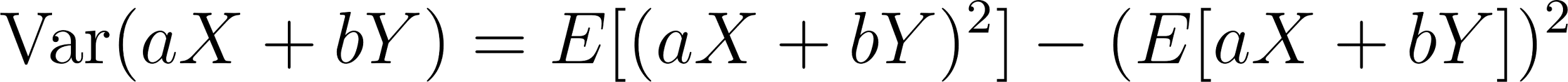
*# Calculate covariance using the property: Cov(aX, bY) = abCov(X, Y)*  
cov\_abXY\_check <- a \* b \* sum((X - mean(X))\*(Y - mean(Y))) / n  
print(paste("Covariance of aX and bY using property: ",   
 round(cov\_abXY\_check, 5)))

## [1] "Covariance of aX and bY using property: 6.02548"

## 

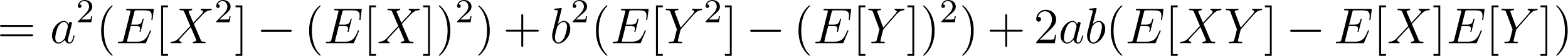
## Var(aX +/- bY) = a^2 Var(X) + b^2 Var(Y) +/- 2ab Cov(X, Y)

### Proof

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BVar%7D(aX%2BbY)%3DE%5B(aX%2BbY)%5E2%5D-(E%5BaX%2BbY%5D)%5E2#0)

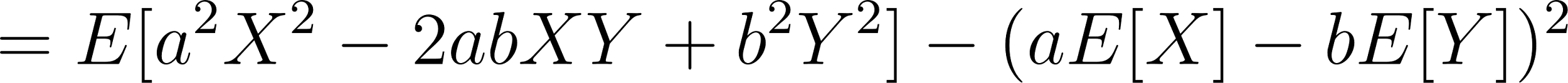
[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5Ba%5E2X%5E2%20%2B%202abXY%20%2B%20b%5E2Y%5E2%5D-(aE%5BX%5D%20%2B%20bE%5BY%5D)%5E2#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3Da%5E2E%5BX%5E2%5D%2B2abE%5BXY%5D%2Bb%5E2E%5BY%5E2%5D-%5Cbig%5B(a%5E2E%5BX%5D)%5E2%20%2B%202abE%5BX%5DE%5BY%5D%20%2B%20b%5E2(E%5BY%5D)%5E2%20%5Cbig%5D#0)

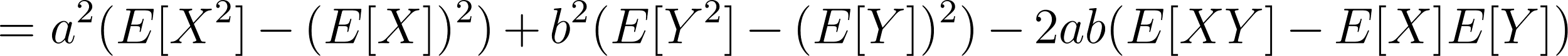
[](https://www.codecogs.com/eqnedit.php?latex=%3Da%5E2(E%5BX%5E2%5D-(E%5BX%5D)%5E2)%2Bb%5E2(E%5BY%5E2%5D-(E%5BY%5D)%5E2)%2B2ab(E%5BXY%5D-E%5BX%5DE%5BY%5D)#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3Da%5E2%5Ctext%7BVar%7D(X)%2B%20b%5E2%5Ctext%7BVar%7D(Y)%20%2B%202ab%5Ctext%7BCov%7D(X%2CY)#0)

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BVar%7D(aX-bY)%3DE%5B(aX-bY)%5E2%5D-(E%5BaX-bY%5D)%5E2#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5Ba%5E2X%5E2%20-%202abXY%20%2B%20b%5E2Y%5E2%5D-(aE%5BX%5D%20-%20bE%5BY%5D)%5E2#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3Da%5E2E%5BX%5E2%5D-2abE%5BXY%5D%2Bb%5E2E%5BY%5E2%5D-%5Cbig%5B(a%5E2E%5BX%5D)%5E2%20-%202abE%5BX%5DE%5BY%5D%20%2B%20b%5E2(E%5BY%5D)%5E2%20%5Cbig%5D#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3Da%5E2(E%5BX%5E2%5D-(E%5BX%5D)%5E2)%2Bb%5E2(E%5BY%5E2%5D-(E%5BY%5D)%5E2)-2ab(E%5BXY%5D-E%5BX%5DE%5BY%5D)#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3Da%5E2%5Ctext%7BVar%7D(X)%2B%20b%5E2%5Ctext%7BVar%7D(Y)%20-%202ab%5Ctext%7BCov%7D(X%2CY)#0)

### R Code

*# Set the constants a and b*  
a <- 2  
b <- 3  
  
*# Set the number of samples*  
n <- 100000  
  
*# Set the seed for reproducibility*  
set.seed(123)  
  
*# Generate samples from Normal distribution for X*  
X <- rnorm(n, mean = 0, sd = 1)  
  
*# Generate Y as a function of X plus some independent random noise*  
Y <- X + rpois(n, lambda = 2)  
  
*# Calculate covariance of X and Y*  
cov\_XY <- sum((X - mean(X))\*(Y - mean(Y))) / n  
var\_X <- sum((X - mean(X))^2) / n  
var\_Y <- sum((Y - mean(Y))^2) / n  
  
*# Calculate variance of aX + bY and aX - bY*  
var\_aX\_plus\_bY <- sum(((a\*X + b\*Y) - mean(a\*X + b\*Y))^2) /n  
var\_aX\_minus\_bY <- sum(((a\*X - b\*Y) - mean(a\*X - b\*Y))^2) /n  
  
*# Check variances using the properties*  
var\_aX\_plus\_bY\_check <- a^2 \* var\_X + b^2 \* var\_Y + 2 \* a \* b \* cov\_XY  
var\_aX\_minus\_bY\_check <- a^2 \* var\_X + b^2 \* var\_Y - 2 \* a \* b \* cov\_XY  
  
print(paste("Variance of aX + bY: ", round(var\_aX\_plus\_bY, 5), " vs ",   
 round(var\_aX\_plus\_bY\_check, 5)))

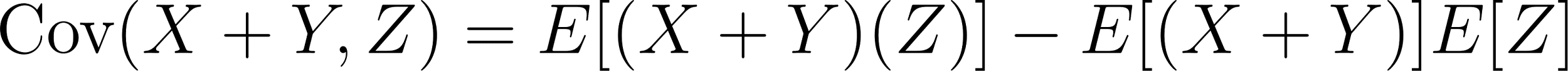
## [1] "Variance of aX + bY: 43.30501 vs 43.30501"

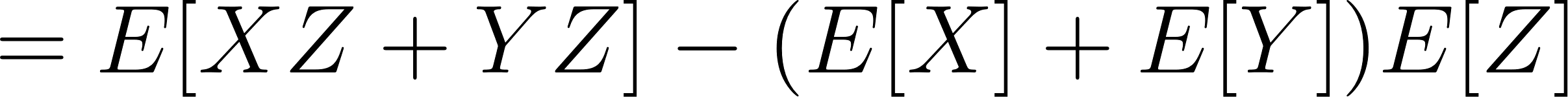
print(paste("Variance of aX - bY: ", round(var\_aX\_minus\_bY, 5), " vs ",   
 round(var\_aX\_minus\_bY\_check, 5)))

## [1] "Variance of aX - bY: 19.20311 vs 19.20311"

## Cov(X + Y, Z) = Cov(X, Z) + Cov(Y, Z)

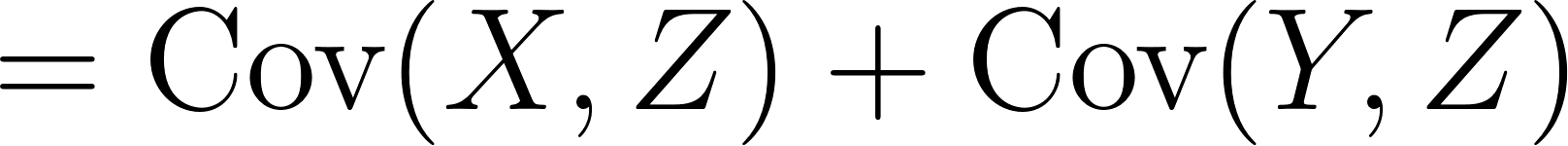
### Proof

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BCov%7D(X%2BY%2CZ)%20%3D%20E%5B(X%2BY)(Z)%5D%20-%20E%5B(X%2BY)%5DE%5BZ%5D%20#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5BXZ%20%2B%20YZ%5D%20-%20(E%5BX%5D%20%2B%20E%5BY%5D)E%5BZ%5D#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5BXZ%5D%20%2B%20E%5BYZ%5D%20-%20E%5BX%5DE%5BZ%5D-E%5BY%5DE%5BZ%5D#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3DE%5BXZ%5D%20-%20E%5BX%5DE%5BZ%5D%20%2B%20E%5BYZ%5D%20-%20E%5BY%5DE%5BZ%5D#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3D%5Ctext%7BCov%7D(X%2C%20Z)%20%2B%20%5Ctext%7BCov%7D(Y%2C%20Z)#0)

### R Code

*# Set the number of samples*  
n <- 100000  
  
*# Set the seed for reproducibility*  
set.seed(123)  
  
*# Generate samples from Normal distribution for X*  
X <- rnorm(n, mean = 0, sd = 1)  
  
*# Generate Y from Poisson distribution*  
Y <- rpois(n, lambda = 2)  
  
*# Generate Z as a function of X and Y plus some independent random noise*  
Z <- X + Y + rnorm(n, mean = 0, sd = 1)  
  
*# Calculate covariance Cov(X + Y, Z)*   
cov\_X\_plus\_Y\_Z <- sum((X + Y - mean(X + Y)) \* (Z - mean(Z))) / n  
print(paste("Covariance of X + Y and Z using definition: ",   
 round(cov\_X\_plus\_Y\_Z, 5)))

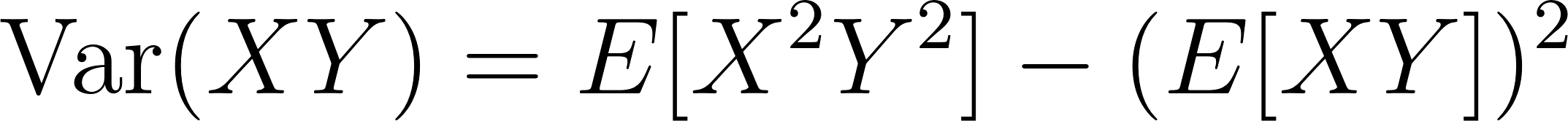
## [1] "Covariance of X + Y and Z using definition: 3.02847"

*# Calculate covariance using the property:*   
*# Cov(X + Y, Z) = Cov(X, Z) + Cov(Y, Z)*  
cov\_XZ <- sum((X - mean(X)) \* (Z - mean(Z))) / n  
cov\_YZ <- sum((Y - mean(Y)) \* (Z - mean(Z))) / n  
cov\_X\_plus\_Y\_Z\_check <- cov\_XZ + cov\_YZ  
print(paste("Covariance of X + Y and Z using property: ",   
 round(cov\_X\_plus\_Y\_Z\_check, 5)))

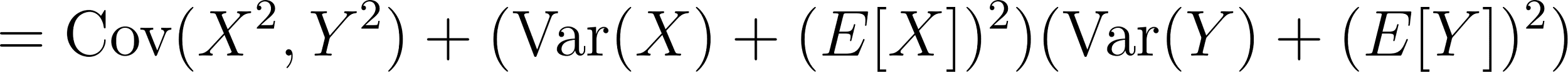
## [1] "Covariance of X + Y and Z using property: 3.02847"

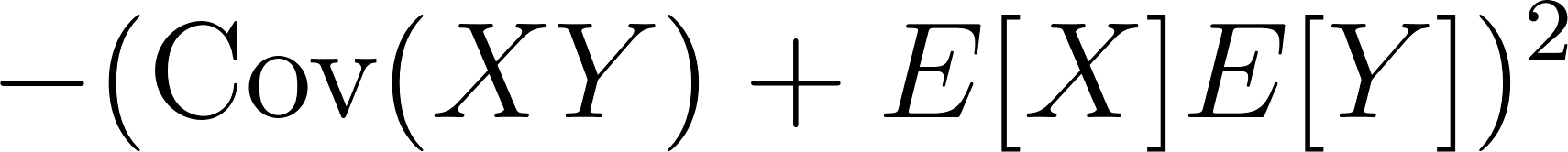
## Var(XY)

### Proof

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BVar%7D(XY)%3DE%5BX%5E2Y%5E2%5D-(E%5BXY%5D)%5E2#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3D%5Ctext%7BCov%7D(X%5E2%2C%20Y%5E2)%20%2B%20E%5BX%5E2%5DE%5BY%5E2%5D%20-(%5Ctext%7BCov%7D(XY)%20%2B%20E%5BX%5DE%5BY%5D)%5E2#0)

[](https://www.codecogs.com/eqnedit.php?latex=%3D%5Ctext%7BCov%7D(X%5E2%2C%20Y%5E2)%20%2B%20(%5Ctext%7BVar%7D(X)%20%2B%20(E%5BX%5D)%5E2)(%5Ctext%7BVar%7D(Y)%20%2B%20(E%5BY%5D)%5E2)#0)

[](https://www.codecogs.com/eqnedit.php?latex=-(%5Ctext%7BCov%7D(XY)%20%2B%20E%5BX%5DE%5BY%5D)%5E2#0)

### R Code

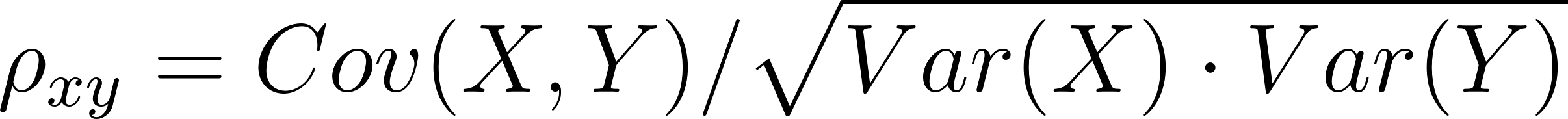
*# Set the number of samples*  
n <- 100000  
  
*# Set the seed for reproducibility*  
set.seed(123)  
  
*# Generate samples from Normal distribution for common variable*  
W <- rnorm(n, mean = 0, sd = 1)  
  
*# Generate X and Y as functions of W plus some independent random noise*  
X <- W + rnorm(n, mean = 0, sd = 1)  
Y <- W + rpois(n, lambda = 2)  
  
*# Calculate variance using the formula: Var(X) = sum((X - mean(X))^2) / n*  
var\_X <- sum((X - mean(X))^2) / n  
var\_Y <- sum((Y - mean(Y))^2) / n  
  
*# Calculate covariance using the formula:*  
*# Cov(X, Y) = sum((X - mean(X)) \* (Y - mean(Y))) / n*  
cov\_XY <- sum((X - mean(X)) \* (Y - mean(Y))) / n  
cov\_X2Y2 <- sum(((X^2) - mean(X^2)) \* ((Y^2) - mean(Y^2))) / n  
  
*# Calculate Var(XY) using the formula:*   
*# Var(XY) = Cov(X^2,Y^2) + [Var(X)+ E(X)^2][Var(Y)+E(Y)^2]*   
*# -[Cov(X,Y) + E(X)E(Y)]^2*  
var\_XY <- sum(((X \* Y) - mean(X \* Y))^2) / n  
var\_XY\_check <- cov\_X2Y2 + (var\_X + mean(X)^2) \* (var\_Y + mean(Y)^2) -   
 (cov\_XY + mean(X) \* mean(Y))^2  
  
print(paste("Variance of XY using definition: ", round(var\_XY, 5)))

## [1] "Variance of XY using definition: 15.04093"

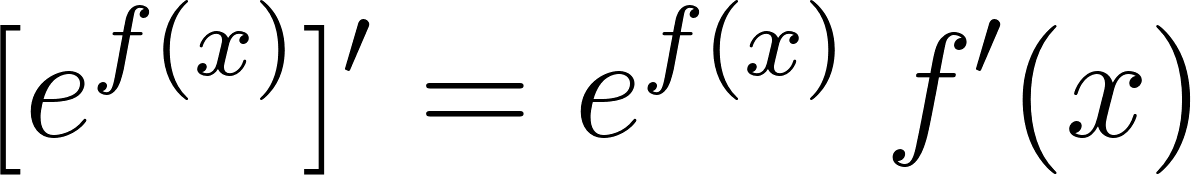
print(paste("Variance of XY using property: ", round(var\_XY\_check, 5)))

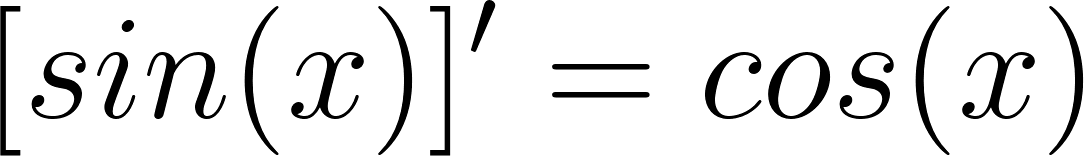
## [1] "Variance of XY using property: 15.04093"

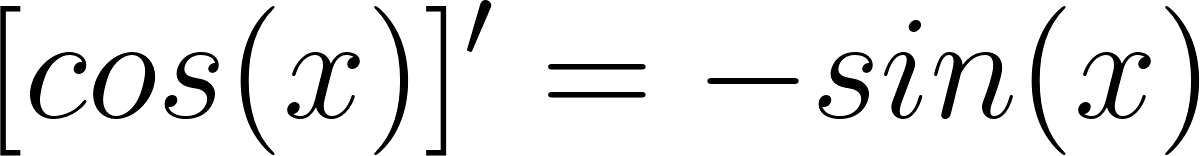
## Correlation of X and Y

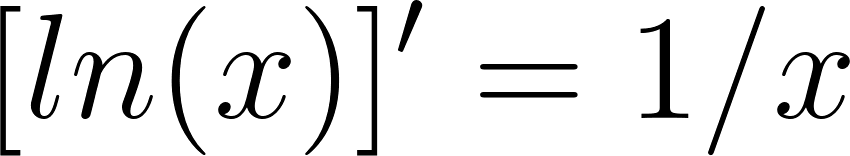
[](https://www.codecogs.com/eqnedit.php?latex=%5Crho_%7Bxy%7D%20%3D%20Cov(X%2CY)%20%2F%20%5Csqrt%7BVar(X)%20%5Ccdot%20Var(Y)%7D%20#0)

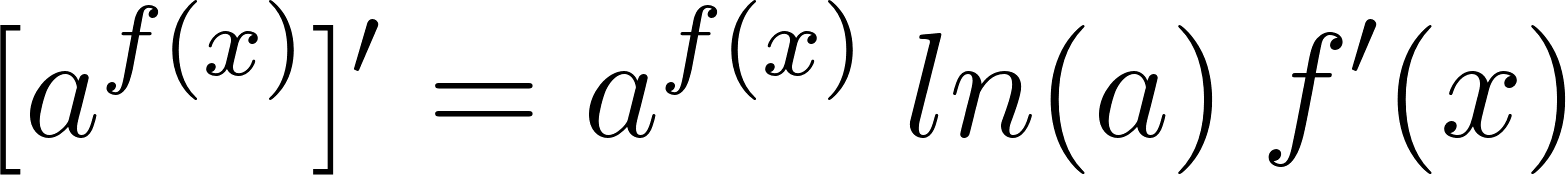
# Differentiation (or Integration if you go from right to left ...)

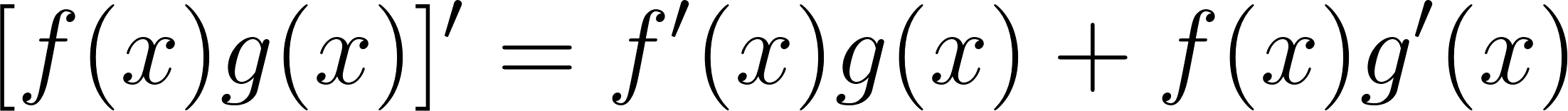
[](https://www.codecogs.com/eqnedit.php?latex=%5Be%5E%7Bf(x)%7D%5D'%20%3D%20e%5E%7Bf(x)%7D%20%5C%3A%20f'(x)%20#0)

[](https://www.codecogs.com/eqnedit.php?latex=%5Bsin%20(x)%20%5D'%20%3D%20cos%20(x)%20#0)

[](https://www.codecogs.com/eqnedit.php?latex=%5Bcos%20(x)%20%5D'%20%3D%20-sin(x)%20#0)

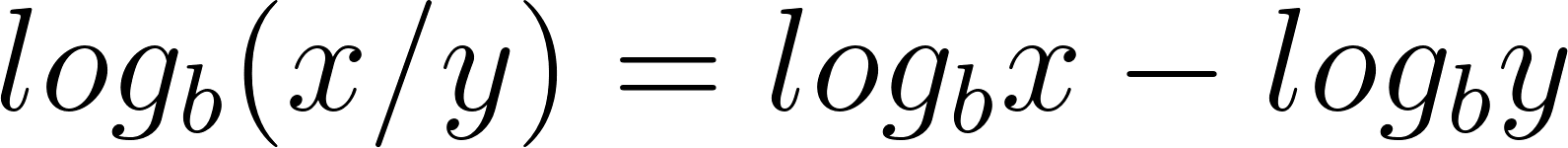
[](https://www.codecogs.com/eqnedit.php?latex=%5Bln%20(x)%20%5D'%20%3D1%2Fx%20#0)

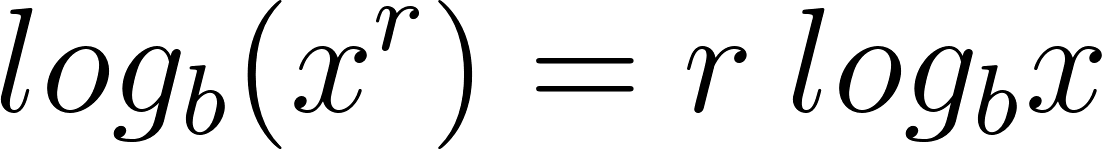
[](https://www.codecogs.com/eqnedit.php?latex=%5Ba%5E%7Bf(x)%7D%5D'%20%3D%20a%5E%7Bf(x)%7D%5C%3A%20ln(a)%20%5C%3A%20f'(x)%20#0)

[](https://www.codecogs.com/eqnedit.php?latex=%5Bf(x)g(x)%5D'%20%3D%20f'(x)g(x)%20%2B%20f(x)g'(x)%20#0)

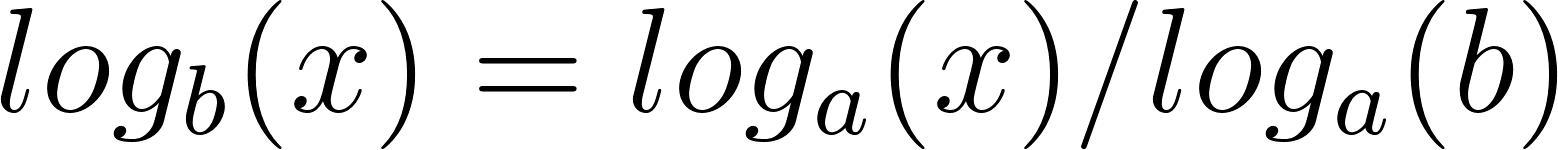
# Integration

# Logarithms

[](https://www.codecogs.com/eqnedit.php?latex=log_b%20(x%2Fy)%20%3D%20log_b%20x%20-%20log_b%20y%20#0)

[](https://www.codecogs.com/eqnedit.php?latex=log_b%20(x%5Er)%20%3D%20r%20%5C%3A%20log_b%20x#0)

change of base:

[](https://www.codecogs.com/eqnedit.php?latex=log_b%20(x)%20%3D%20log_a%20(x)%20%2F%20log_a%20(b)%20#0)