Exploring R

2023-11-01

library(ggplot2)

## Warning: replacing previous import 'lifecycle::last\_warnings' by  
## 'rlang::last\_warnings' when loading 'tibble'

## Warning: replacing previous import 'lifecycle::last\_warnings' by  
## 'rlang::last\_warnings' when loading 'pillar'

library(plotly)

## Warning: package 'plotly' was built under R version 4.0.5

##   
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':  
##   
## last\_plot

## The following object is masked from 'package:stats':  
##   
## filter

## The following object is masked from 'package:graphics':  
##   
## layout

library(readr)  
library(ggpointdensity)

## Warning: package 'ggpointdensity' was built under R version 4.0.5

library(viridis)

## Loading required package: viridisLite

## Warning: package 'viridisLite' was built under R version 4.0.5

library(tidyr)  
library(fitdistrplus)

## Warning: package 'fitdistrplus' was built under R version 4.0.5

## Loading required package: MASS

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:plotly':  
##   
## select

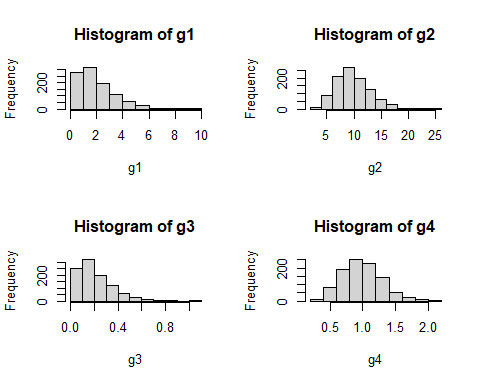
## Loading required package: survival

## Warning: package 'survival' was built under R version 4.0.5

library(rlang)

## Warning: package 'rlang' was built under R version 4.0.5

set.seed(0)  
g1 <- rgamma(1000, shape = 2, rate = 1)  
  
set.seed(1)  
g2 <- rgamma(1000, shape = 10, rate = 1)  
  
set.seed(2)  
g3 <- rgamma(1000, shape = 2, rate = 10)  
  
set.seed(3)  
g4 <- rgamma(1000, shape = 10, rate = 10)  
  
par(mfrow = c(2,2))  
hist(g1)  
hist(g2)  
hist(g3)  
hist(g4)



(function(x, y){ z <- x^2 + y^2; x+y+z })(0:7, 1)

## [1] 2 4 8 14 22 32 44 58

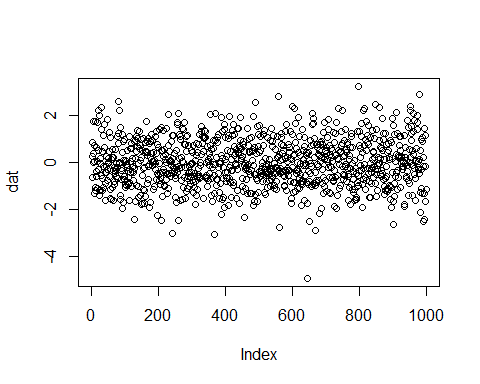
norm <- function(x) sqrt(x%\*%x)  
norm(1:4)

## [,1]  
## [1,] 5.477226

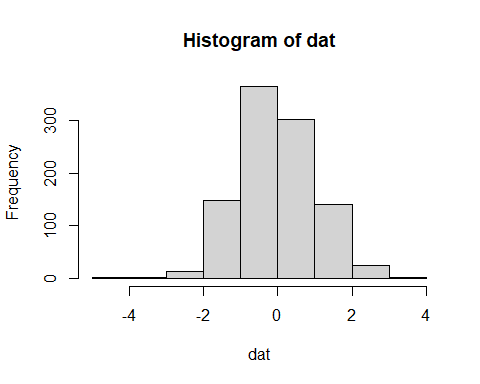
#Generate Random Data  
set.seed(6)  
dat <- rnorm(1000)

#Moffat 1D  
#One Dimensional Moffat Model  
  
#Model Formula  
#f(x) = A(1+(x-mu)^2/gamma^2)^-alpha  
  
#Parameters  
#A = Amplitude  
#mu = the x position of the maximum of the model   
#gamma = core width of the model  
#alpha = power index  
  
#Defaults  
#Amplitude = 1  
#mu = 0  
#gamma = 1  
#alpha = 1  
  
dMoffat1D <- function(x, amplitude, mu, gamma, alpha){amplitude\*(1+((x-mu)/gamma)^2)^-alpha}  
  
Moffat1D\_Deriv <- function(x, amplitude, mu, gamma, alpha){  
 chain = (1 + (x - mu)^2 / gamma^2)  
 d\_A = chain^(-alpha)  
 d\_mu = 2\*amplitude\*alpha\*(x - mu)\*d\_A/(chain\*gamma^2)  
 d\_gamma = 2\*amplitude\*alpha\*((x - mu)^2)\*d\_A/(chain\*gamma^3)  
 d\_alpha = -amplitude\*d\_A\*log(chain)  
   
 return(list(c(d\_A, d\_mu, d\_gamma, d\_alpha)))  
}

plot(dat)



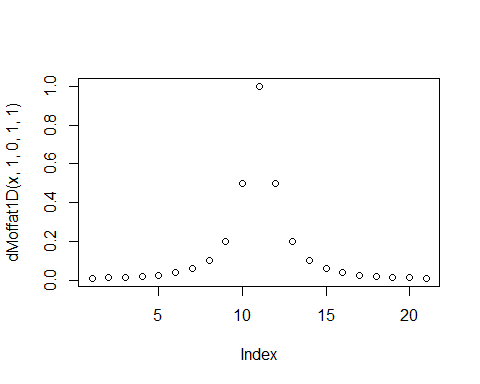
hist(dat)



dMoffat1D(2, 1, 0, 1, 1)

## [1] 0.2

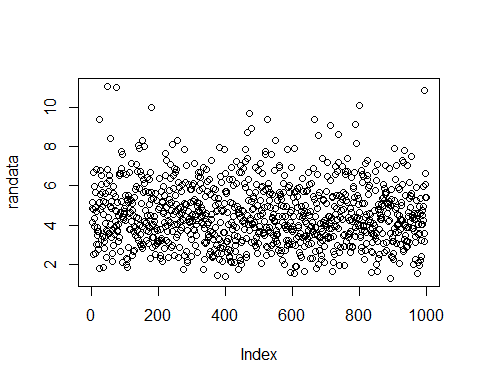
x <- c(-10:10)  
plot(dMoffat1D(x, 1, 0, 1, 1), )



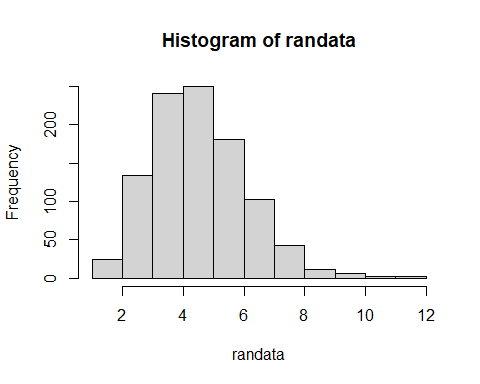
Moffat1D\_Deriv(0.5, 1, 0, 1, 1)

## [[1]]  
## [1] 0.8000000 0.6400000 0.3200000 -0.1785148

set.seed(8)  
  
randata <- rgamma(1000, 9, 2)  
plot(randata)

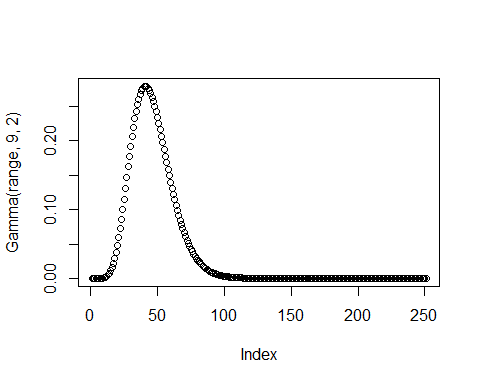


hist(randata)

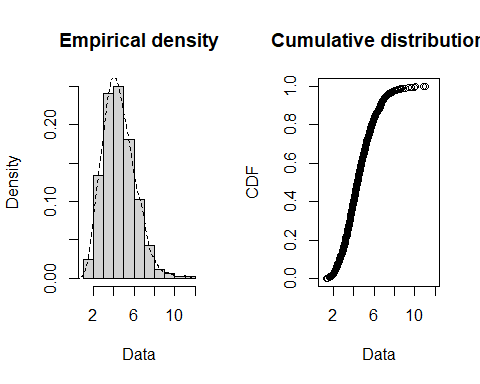


beta = 2  
alpha = 9

Gamma <- function(x, alpha, beta){  
 ((beta^alpha)/factorial(alpha-1))\* x^(alpha-1)\* exp(-beta\*x)  
}  
  
range <- seq(0,25, by=0.1)  
  
  
plot(Gamma(range, 9, 2))



plotdist(randata, demp = TRUE)



#check  
Gamma(5, 9, 2)

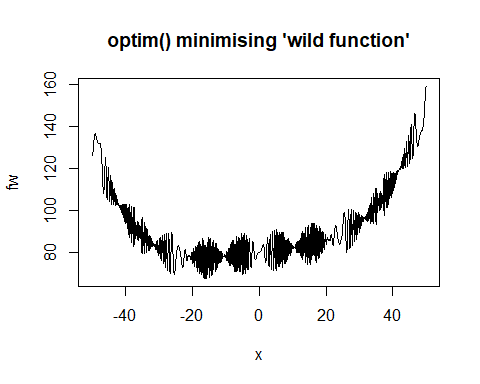
## [1] 0.2251981

dgamma(5, 9, 2)

## [1] 0.2251981

#res <- optim(c(5,9,2), Gamma)

#optim example code  
## "wild" function , global minimum at about -15.81515  
fw <- function (x)  
 10\*sin(0.3\*x)\*sin(1.3\*x^2) + 0.00001\*x^4 + 0.2\*x+80  
plot(fw, -50, 50, n = 1000, main = "optim() minimising 'wild function'")



res <- optim(50, fw, method = "SANN",  
 control = list(maxit = 20000, temp = 20, parscale = 20))  
res

## $par  
## [1] -15.81461  
##   
## $value  
## [1] 67.47023  
##   
## $counts  
## function gradient   
## 20000 NA   
##   
## $convergence  
## [1] 0  
##   
## $message  
## NULL

range2 <- seq(-3, 3, by = 0.1)  
  
par(mfrow = c(2,3))  
plot(dMoffat1D(range2, 1, 0, 1, 1), main = "Default parameters")  
plot(dMoffat1D(range2, 1, 0, 1, 4), main = "Changing alpha")  
plot(dMoffat1D(range2, 1, 1, 1, 1), main = "Changing mu")  
plot(dMoffat1D(range2, 1, 0, 4, 1), main = "Changing gamma")  
plot(dMoffat1D(range2, 4, 0, 1, 1), main = "Changing A")  
plot(dMoffat1D(range2, 4, 1, 4, 4), main = "Changing all")

