pruebas iid

Eduardo Rubio M

01-09-2024

Primero estamos probando generar series IID para ver como se comporta el estadístico X_n(r,d) esta vez con un mayor numero de observaciones.

```
# Generar las series IID
set.seed(125)
N <- 10000  # Longitud de la serie
data <- rnorm(N)
```

Generamos la función para calcular la Corelación integral

```
# Definir la función para calcular la Corelación integral
correlation_integral <- function(data, r, d) {
  N <- length(data)
  count <- 0

  # Iterar sobre todos los pares (i, j) tal que 1 <= i < j <= N-d+1
  for (i in 1:(N-d)) {
    for (j in (i+1):(N-d+1)) {
      # Crear los vectores u_i^d y u_j^d
      u_i_d <- data[i:(i+d-1)]
      u_j_d <- data[j:(j+d-1)]

      # Calcular la distancia euclidiana entre u_i^d y u_j^d
      distance <- sqrt(sum((u_i_d - u_j_d)^2))

      # Verificar si la distancia es menor que r
      if (distance < r) {
        count <- count + 1
      }
    }
  }

  # Calcular C_N(r, d)
  C_N <- (2 / (N^2)) * count
  return(C_N)
}
```

Probamos para distintos casos de d y r como funciona la función.

```
# Parámetros
r <- 0.5  # Umbral para la norma
```

```r
d <- 2    # Dimensión del espacio fase

# Calcular la correlación integral
C_N_value <- correlation_integral(data, r, d)
C_N_value
```

```
## [1] 0.05993166
```

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d <- 3    # Dimensión del espacio fase

# Calcular la correlación integral
C_N_value <- correlation_integral(data, r, d)
C_N_value
```

```
## [1] 0.01117144
```

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d <- 4    # Dimensión del espacio fase

# Calcular la correlación integral
C_N_value <- correlation_integral(data, r, d)
C_N_value
```

```
## [1] 0.00184548
```

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d <- 5    # Dimensión del espacio fase

# Calcular la correlación integral
C_N_value <- correlation_integral(data, r, d)
C_N_value
```

```
## [1] 0.00027478
```

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d <- 7    # Dimensión del espacio fase

# Calcular la correlación integral
C_N_value <- correlation_integral(data, r, d)
C_N_value
```

```
## [1] 5e-06
```

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d <- 10    # Dimensión del espacio fase

# Calcular la correlación integral
C_N_value <- correlation_integral(data, r, d)
C_N_value
```

```
## [1] 2e-08
```

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d <- 15    # Dimensión del espacio fase

# Calcular la correlación integral
C_N_value <- correlation_integral(data, r, d)
C_N_value
```

```
## [1] 0
```

```r
# Parámetros
r <- 0.1  # Umbral para la norma
d <- 2    # Dimensión del espacio fase

# Calcular la correlación integral
C_N_value <- correlation_integral(data, r, d)
C_N_value
```

```
## [1] 0.00247884
```

```r
# Parámetros
r <- 1  # Umbral para la norma
d <- 2    # Dimensión del espacio fase

# Calcular la correlación integral
C_N_value <- correlation_integral(data, r, d)
C_N_value
```

```
## [1] 0.219405
```

```r
# Parámetros
r <- 5  # Umbral para la norma
d <- 2    # Dimensión del espacio fase

# Calcular la correlación integral
C_N_value <- correlation_integral(data, r, d)
C_N_value
```

```
## [1] 0.997598
```

```r
# Parámetros
r <- 10  # Umbral para la norma
d <- 2    # Dimensión del espacio fase

# Calcular la correlación integral
C_N_value <- correlation_integral(data, r, d)
C_N_value
```

```
## [1] 0.9997
```

Verificamos que la función se comporta según lo esperado, casi no se percibe cambio a las pruebas anteriores con menos observaciones, por lo que se seguirá con las pruebas

Ahora generamos la función para el X__n(r,d)

```r
# Definir la función X_N(r, d)
X_N <- function(data, r, d) {
  C_N_d <- correlation_integral(data, r, d)
  C_N_d_minus_1 <- correlation_integral(data, r, d-1)
  C_N_d_plus_1 <- correlation_integral(data, r, d+1)

  X_N_value <- (C_N_d^2) / (C_N_d_minus_1 * C_N_d_plus_1)
  return(X_N_value)
}

# Caso especial para d = 1
X_N_d_1 <- function(data, r) {
  C_N_1 <- correlation_integral(data, r, 1)
  C_N_2 <- correlation_integral(data, r, 2)

  X_N_value <- (C_N_1^2) / C_N_2
  return(X_N_value)
}
```

Probemos calcular para d=2 y d=1

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d <- 2    # Dimensión del espacio fase

# Calcular X_N(r, d)
X_N_value <- X_N(data, r, d)
X_N_value
```

```
## [1] 1.170082
```

```r
# Calcular X_N(r, 1)
X_N_value_d_1 <- X_N_d_1(data, r)
X_N_value_d_1
```

```
## [1] 1.259847
```

probemos como va con distintos d y r

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d <- 3    # Dimensión del espacio fase

# Calcular X_N(r, d)
X_N_value <- X_N(data, r, d)
X_N_value
```

```
## [1] 1.128373
```

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d <- 4    # Dimensión del espacio fase

# Calcular X_N(r, d)
X_N_value <- X_N(data, r, d)
X_N_value
```

```
## [1] 1.109493
```

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d <- 5    # Dimensión del espacio fase

# Calcular X_N(r, d)
X_N_value <- X_N(data, r, d)
X_N_value
```

```
## [1] 1.07496
```

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d <- 6    # Dimensión del espacio fase

# Calcular X_N(r, d)
X_N_value <- X_N(data, r, d)
X_N_value
```

```
## [1] 1.054344
```

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d <- 10    # Dimensión del espacio fase

# Calcular X_N(r, d)
X_N_value <- X_N(data, r, d)
X_N_value
```

```
## [1] Inf
```

nuevamente se generan los problemas para d muy grandes dado que se obtienen ceros en el paso anterior al calcular la correlación integral.

Veamos como funciona para múltiples series IID, esta vez con 500 series de 1000 observaciones.

```r
# Generar múltiples series IID
set.seed(125)
n_series <- 500  # Número de series
length_series <- 1000  # Longitud de cada serie

# Crear una matriz donde cada fila es una serie IID
data_matrix <- matrix(rnorm(n_series * length_series), nrow = n_series, ncol = length_series)
```

```r
# Definir parámetros
r <- 0.5  # Umbral para la norma
d <- 2    # Dimensión del espacio fase

# Inicializar vectores para almacenar los resultados
X_N_values <- numeric(n_series)
X_N_values_d_1 <- numeric(n_series)

# Calcular X_{N}(r,d) para cada serie
for (i in 1:n_series) {
  series <- data_matrix[i, ]
  X_N_values[i] <- X_N(series, r, d)
  X_N_values_d_1[i] <- X_N_d_1(series, r)
}

# Ver los resultados
head(X_N_values)
```

```
## [1] 1.157207 1.153408 1.171539 1.180872 1.170530 1.193548
```

```r
head(X_N_values_d_1)
```

```
## [1] 1.271478 1.250571 1.253229 1.272058 1.263005 1.250287
```

```r
# Parámetros
r <- 0.5  # Umbral para la norma
d_values <- 1:5  # Valores de d

# Inicializar listas para almacenar los resultados
X_N_results <- matrix(0, nrow = n_series, ncol = length(d_values))
colnames(X_N_results) <- paste("d =", d_values)

# Calcular X_{N}(r,d) para cada serie y para cada valor de d
for (i in 1:n_series) {
  series <- data_matrix[i, ]

  # Caso especial para d = 1
  X_N_results[i, 1] <- X_N_d_1(series, r)

  # Calcular X_{N}(r,d) para d = 2, 3, 4, 5
```

```r
  for (d in 2:length(d_values)) {
    X_N_results[i, d] <- X_N(series, r, d_values[d])
  }
}

# Ver los resultados
head(X_N_results)
```

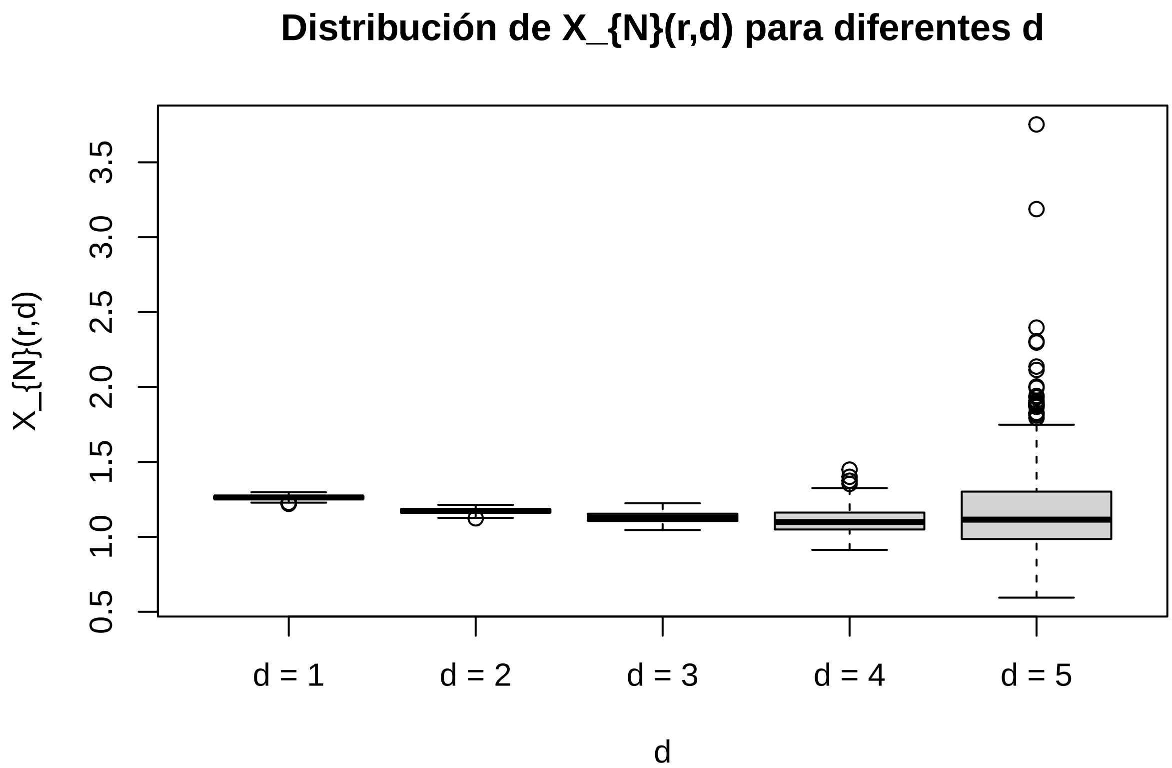
```
##         d = 1    d = 2    d = 3    d = 4    d = 5
## [1,] 1.271478 1.157207 1.140453 1.101400 1.436829
## [2,] 1.250571 1.153408 1.148975 1.118246 1.612384
## [3,] 1.253229 1.171539 1.112096 1.114135 1.130591
## [4,] 1.272058 1.180872 1.045644 1.070135 1.004490
## [5,] 1.263005 1.170530 1.197670 1.092133 1.306241
## [6,] 1.250287 1.193548 1.159229 1.069199 1.040414
```

Veamos como se comporta X_{N}(r,d) para este caso

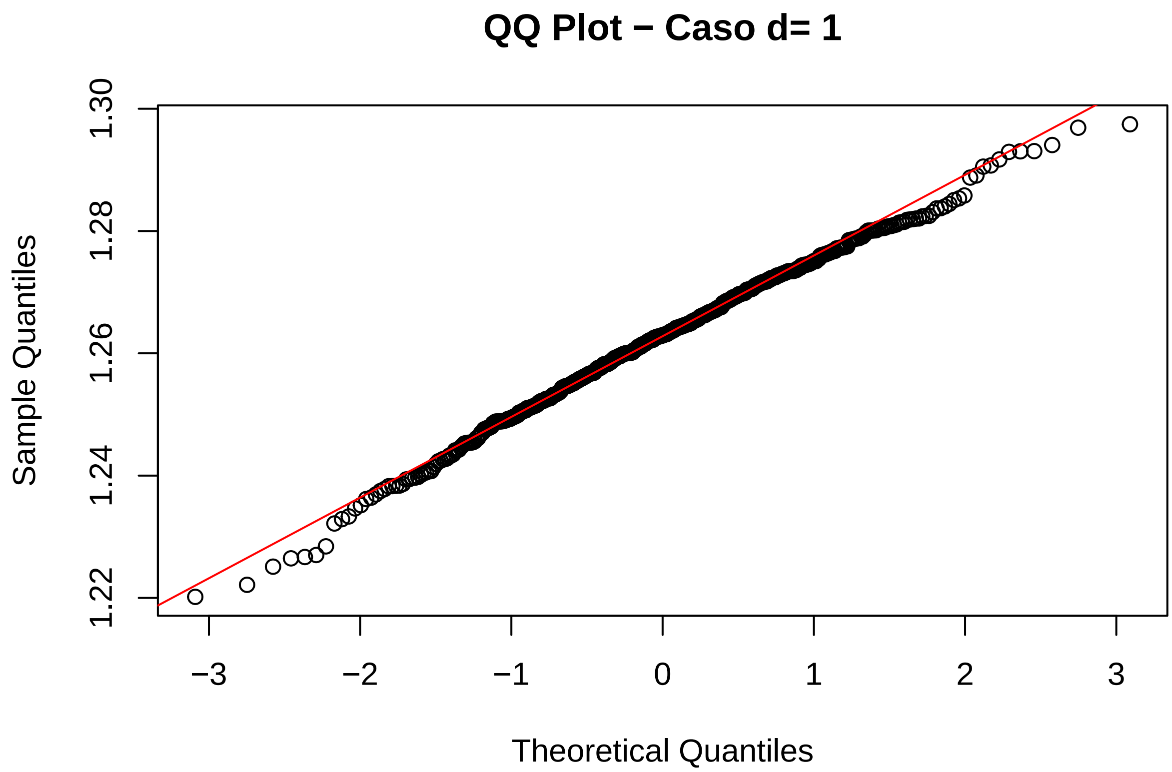
```r
# Resumen de los resultados
summary(X_N_results)
```

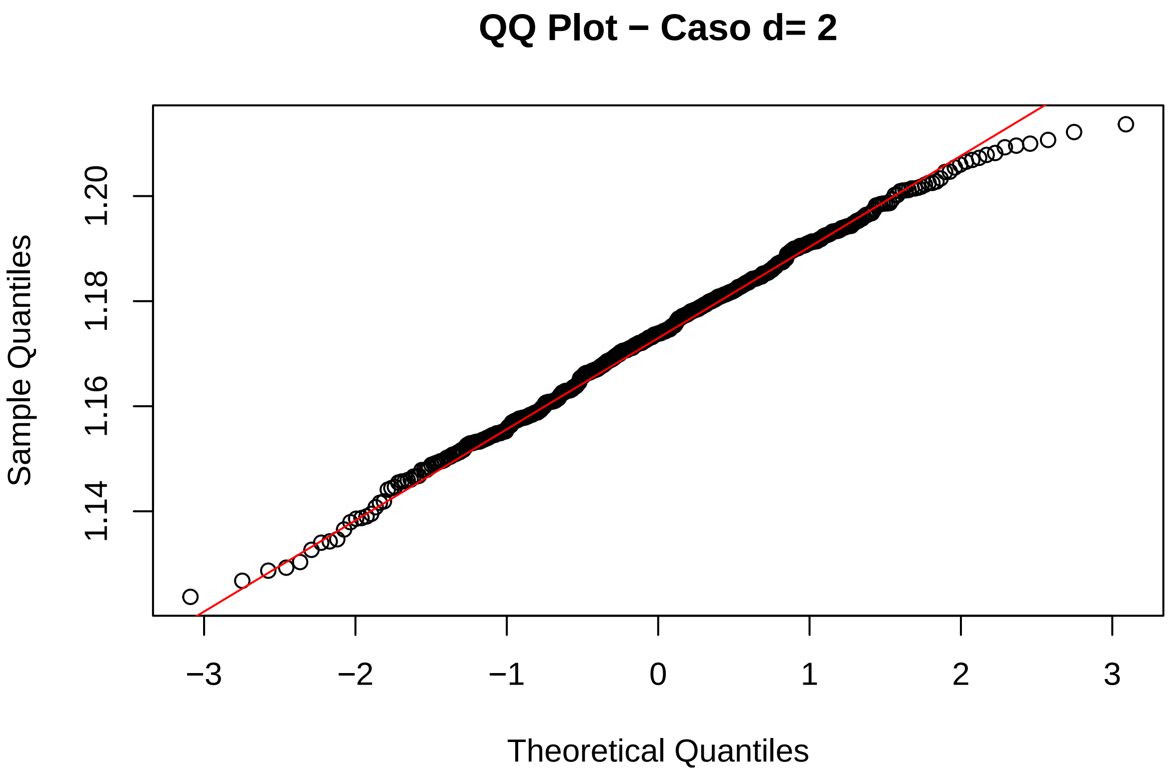
```
##      d = 1           d = 2           d = 3           d = 4
##  Min.   :1.220   Min.   :1.124   Min.   :1.046   Min.   :0.9136
##  1st Qu.:1.254   1st Qu.:1.161   1st Qu.:1.106   1st Qu.:1.0494
##  Median :1.263   Median :1.174   Median :1.129   Median :1.0992
##  Mean   :1.262   Mean   :1.173   Mean   :1.130   Mean   :1.1090
##  3rd Qu.:1.272   3rd Qu.:1.185   3rd Qu.:1.155   3rd Qu.:1.1620
##  Max.   :1.297   Max.   :1.214   Max.   :1.224   Max.   :1.4488
##      d = 5
##  Min.   :0.5944
##  1st Qu.:0.9859
##  Median :1.1150
##  Mean   :1.1799
##  3rd Qu.:1.3015
##  Max.   :3.7528
```

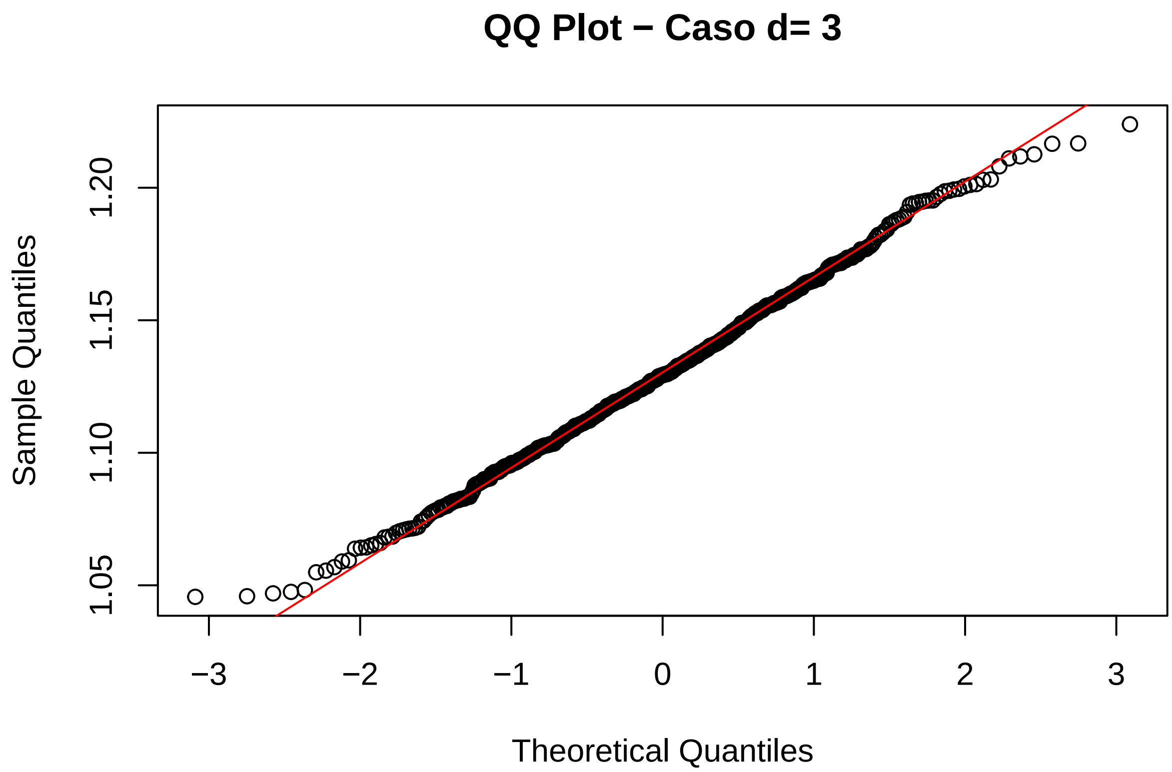
```r
# Graficar los resultados
boxplot(X_N_results, main="Distribución de X_{N}(r,d) para diferentes d", ylab="X_{N}(r,d)", xlab="d")
```

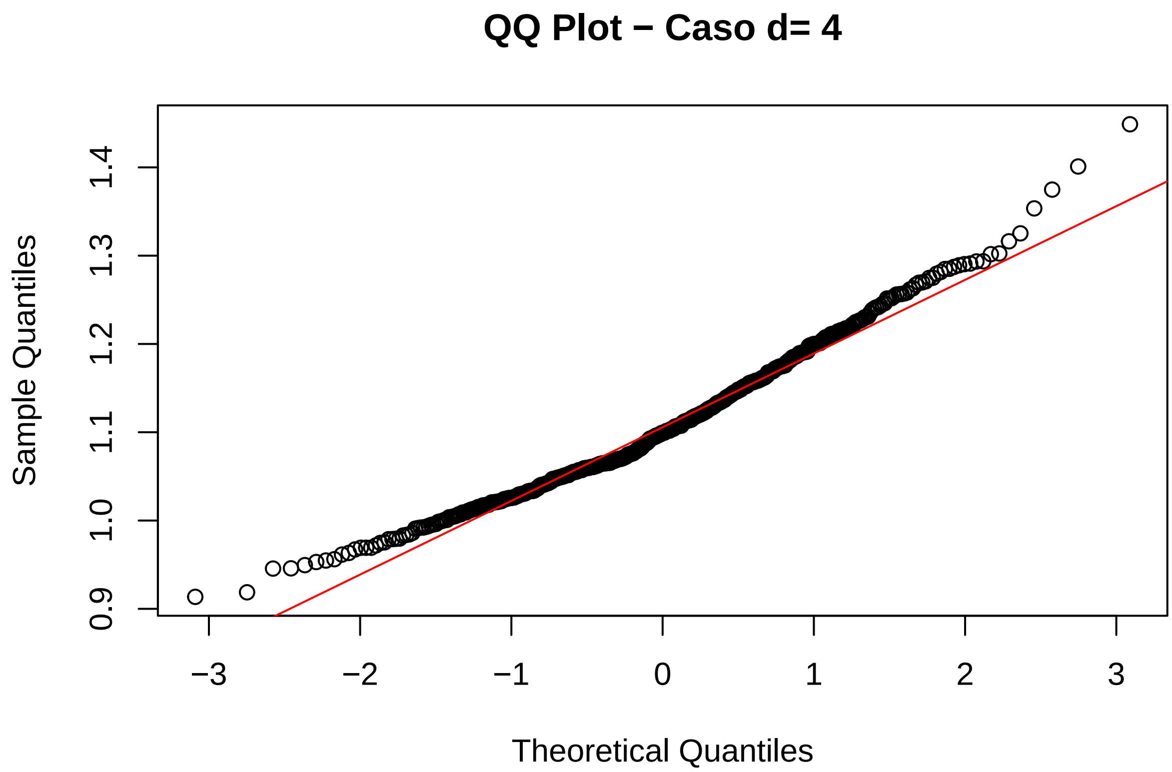


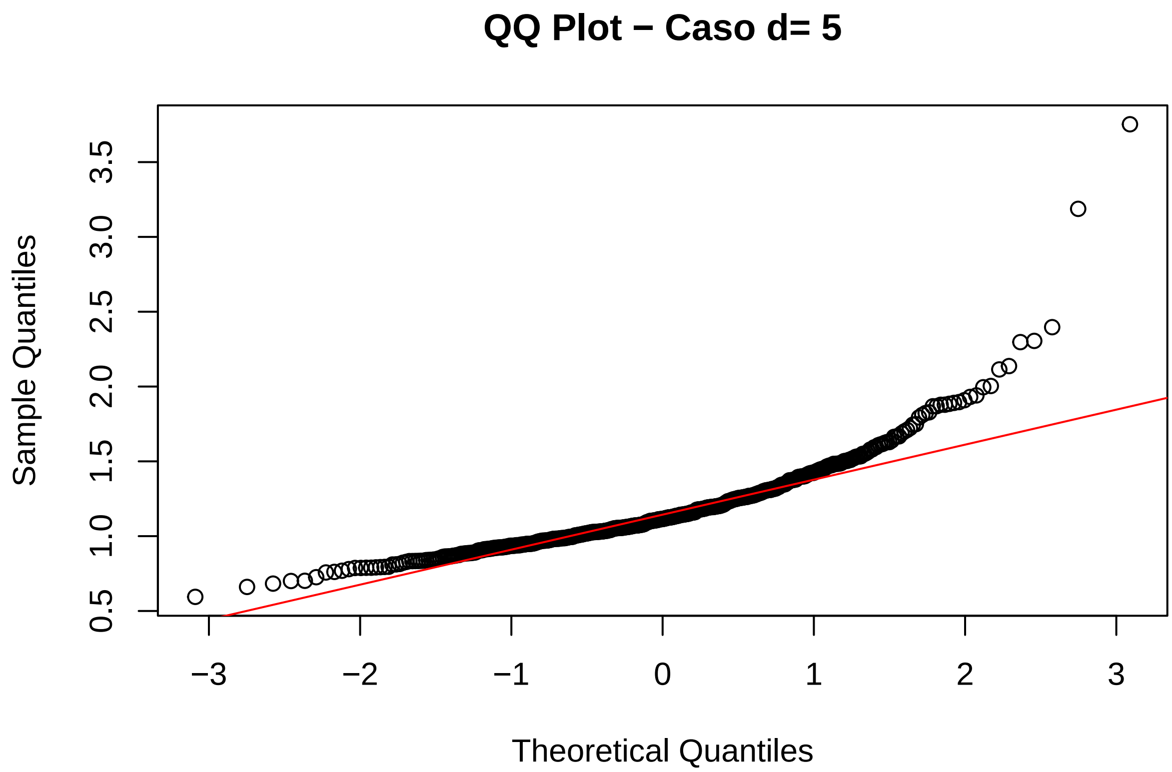

```r
# Crear el QQ plot

for (i in 1:ncol(X_N_results)) {
  qqnorm(X_N_results[, i], main = paste("QQ Plot - Caso d=", i))
  qqline(X_N_results[, i], col = "red")
}
```


QQ Plot − Caso d= 1
Sample Quantiles
1.22
1.24
1.26
1.28
1.30
−3
−2
−1
0
1
2
3
Theoretical Quantiles


QQ Plot − Caso d= 2
Sample Quantiles
1.14
1.16
1.18
1.20
−3
−2
−1
0
1
2
3
Theoretical Quantiles


QQ Plot − Caso d= 3
Sample Quantiles
1.05
1.10
1.15
1.20
−3
−2
−1
0
1
2
3
Theoretical Quantiles


QQ Plot − Caso d= 4
Sample Quantiles
0.9
1.0
1.1
1.2
1.3
1.4
−3
−2
−1
0
1
2
3
Theoretical Quantiles



Esta vez al aumentar en numero de series se puede observar de manera mas clara la tendencia.

Calculamos $\sqrt{N} ln(X_N(r,d))$

```r
# Longitud de cada serie
N <- length_series  # Esto es igual a 1000 en nuestro caso

# Calcular sqrt(N)
sqrt_N <- sqrt(N)

# Calcular sqrt(N) * ln(X_{N}(r,d)) para cada serie y cada valor de d
sqrt_N_ln_X_N <- sqrt_N * log(X_N_results)

# Ver los resultados
head(sqrt_N_ln_X_N)
```

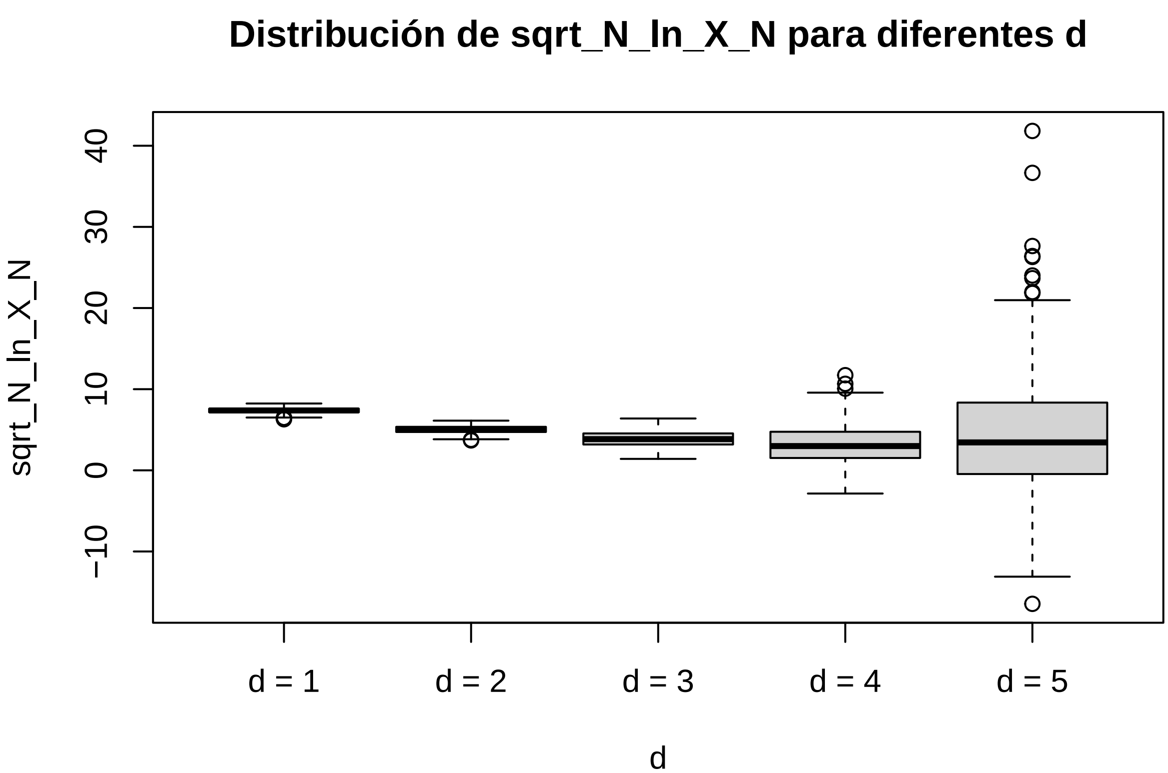
```
##         d = 1    d = 2    d = 3    d = 4      d = 5
## [1,] 7.595157 4.617217 4.156038 3.054196 11.4613071
## [2,] 7.070848 4.513228 4.391462 3.534200 15.1066396
## [3,] 7.138013 5.006463 3.359822 3.417745  3.8813903
## [4,] 7.609576 5.257377 1.411429 2.143540  0.1416631
## [5,] 7.383722 4.979228 5.704060 2.786997  8.4481352
## [6,] 7.063680 5.595034 4.672435 2.115886  1.2528633
```

y vemos como se comporta:

```r
# Resumen de los resultados
summary(sqrt_N_ln_X_N)
```

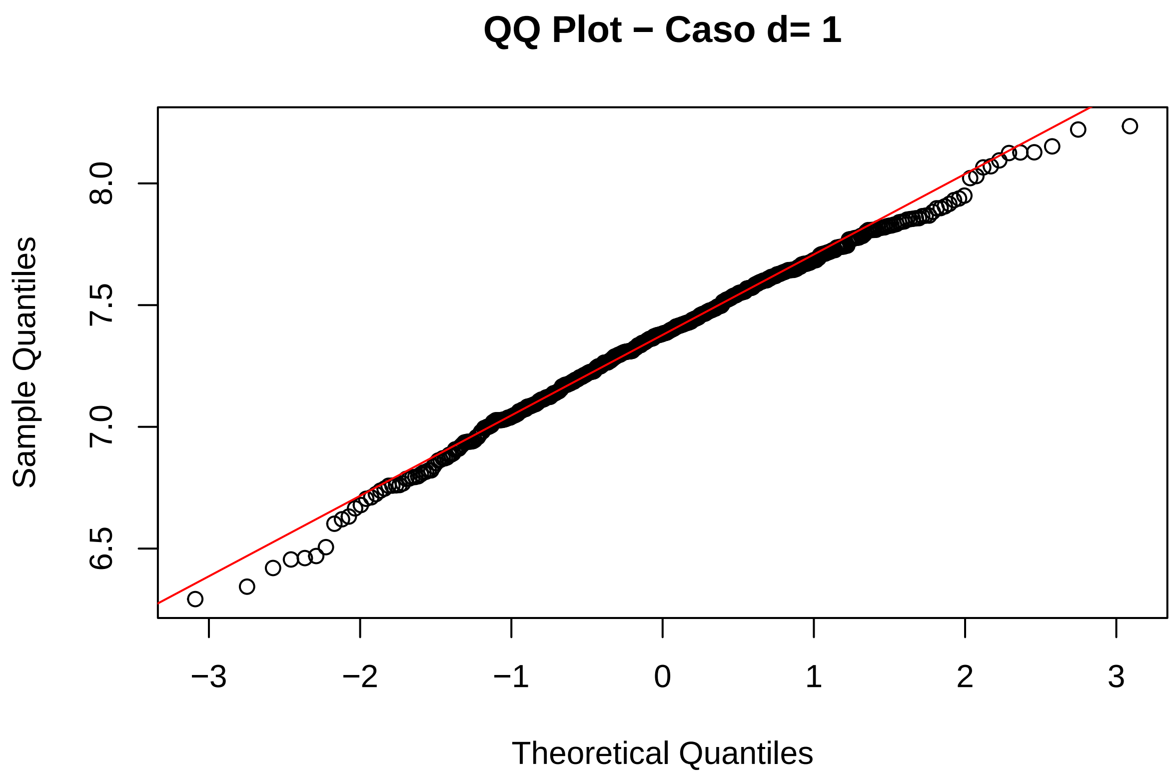
```
##      d = 1           d = 2           d = 3           d = 4
##  Min.   :6.293   Min.   :3.689   Min.   :1.411   Min.   :-2.859
##  1st Qu.:7.155   1st Qu.:4.728   1st Qu.:3.188   1st Qu.: 1.526
##  Median :7.383   Median :5.068   Median :3.844   Median : 2.991
##  Mean   :7.367   Mean   :5.055   Mean   :3.854   Mean   : 3.182
##  3rd Qu.:7.601   3rd Qu.:5.359   3rd Qu.:4.546   3rd Qu.: 4.748
##  Max.   :8.235   Max.   :6.124   Max.   :6.390   Max.   :11.724
##      d = 5
##  Min.   :-16.4512
##  1st Qu.: -0.4494
##  Median :  3.4414
##  Mean   :  4.3424
##  3rd Qu.:  8.3329
##  Max.   : 41.8215
```

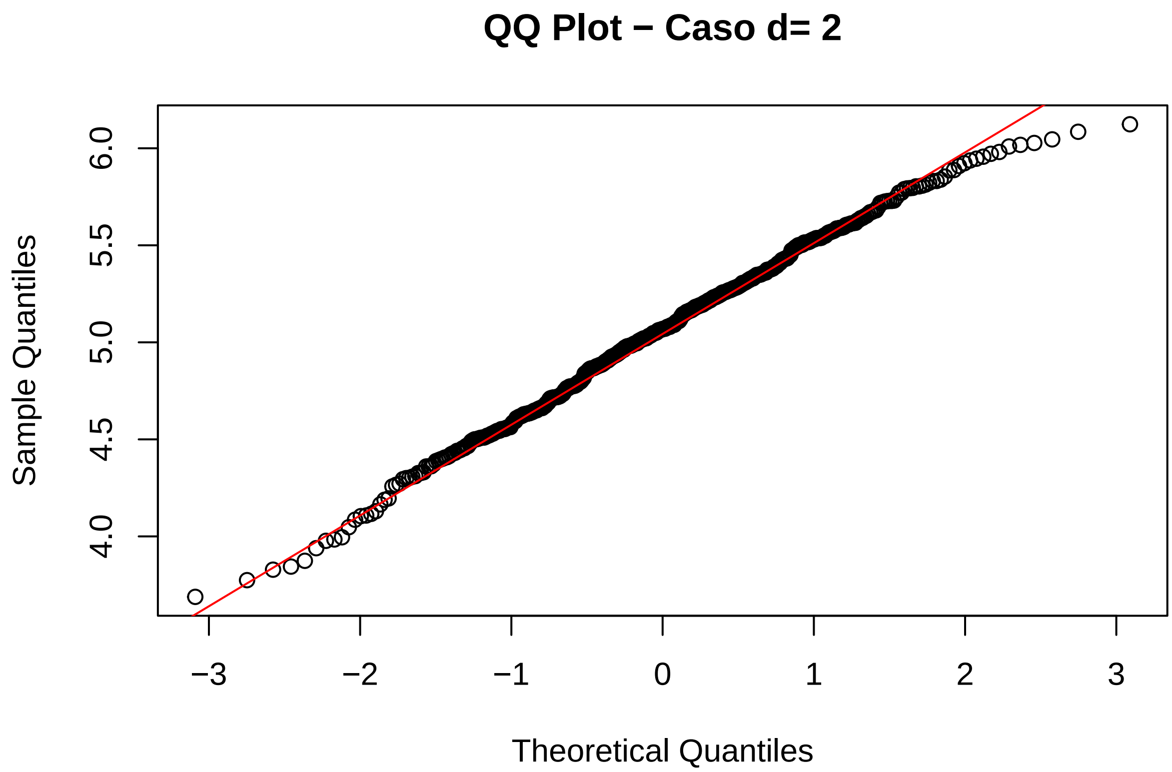
```r
# Graficar los resultados
boxplot(sqrt_N_ln_X_N, main="Distribución de sqrt_N_ln_X_N para diferentes d", ylab="sqrt_N_ln_X_N", xla
```

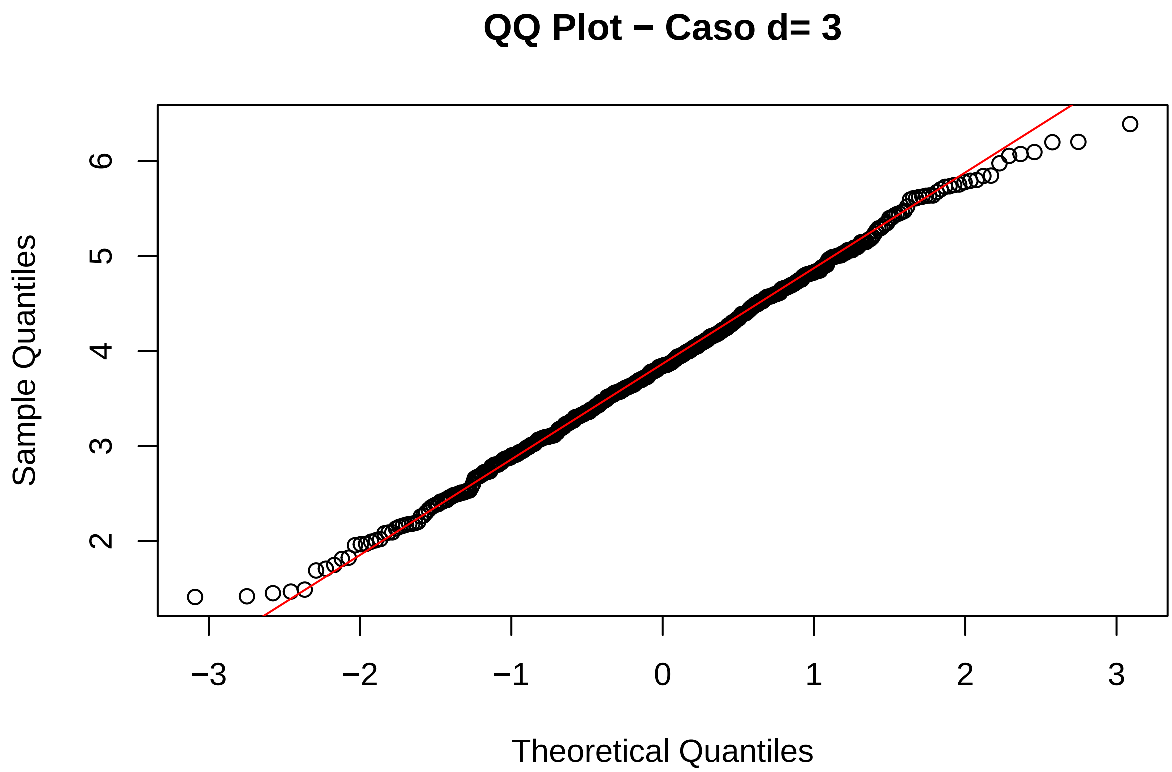


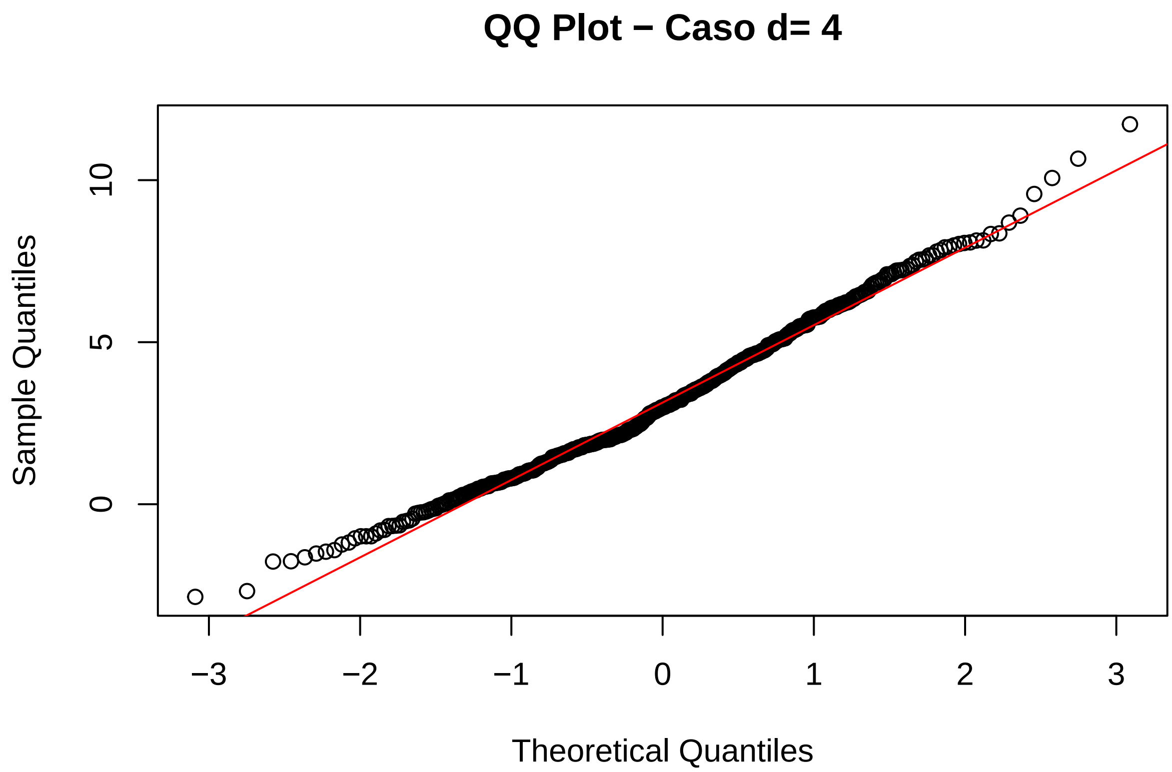

```r
# Crear el QQ plot

for (i in 1:ncol(sqrt_N_ln_X_N)) {
```

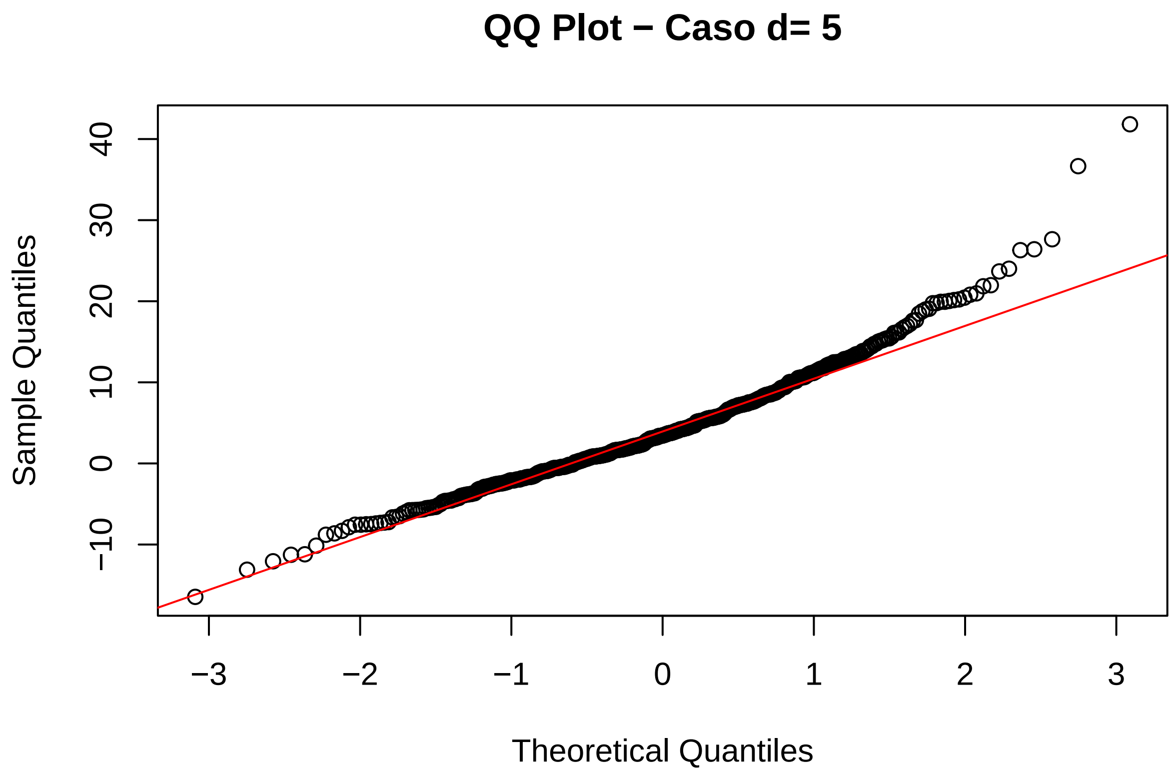
```
  qqnorm(sqrt_N_ln_X_N[, i], main = paste("QQ Plot - Caso d=", i))
  qqline(sqrt_N_ln_X_N[, i], col = "red")
}
```




QQ Plot − Caso d= 2
Sample Quantiles
4.0
4.5
5.0
5.5
6.0
−3
−2
−1
0
1
2
3
Theoretical Quantiles


QQ Plot − Caso d= 3
Sample Quantiles
2
3
4
5
6
−3
−2
−1
0
1
2
3
Theoretical Quantiles


QQ Plot − Caso d= 4
Sample Quantiles
Theoretical Quantiles
10
5
0
−3
−2
−1
0
1
2
3



De la misma manera para $\sqrt{N} ln(X_N(r,d))$ se tiene mucho mas clara la tendencia de los datos.

pruebas con caso AR(1)

```r
set.seed(125)  # Fijamos la semilla

# Definimos parámetros
n_series <- 500   # Número de series a generar
n_obs <- 1000      # Número de observaciones por serie
phi <- 0.5        # Coeficiente AR(1)
sigma <- 1        # Desviación estándar del ruido

# Matriz para almacenar las series AR(1)
ar1_series <- matrix(0, nrow = n_obs, ncol = n_series)

# Generamos las series AR(1)
for (i in 1:n_series) {
  ar1_series[, i] <- arima.sim(n = n_obs, list(ar = phi), sd = sigma)
}
```

```r
# Definir parámetros
r <- 0.5  # Umbral para la norma
d <- 2    # Dimensión del espacio fase

# Inicializar vectores para almacenar los resultados
X_N_values_ar <- numeric(n_series)
X_N_values_ar_d_1 <- numeric(n_series)
```

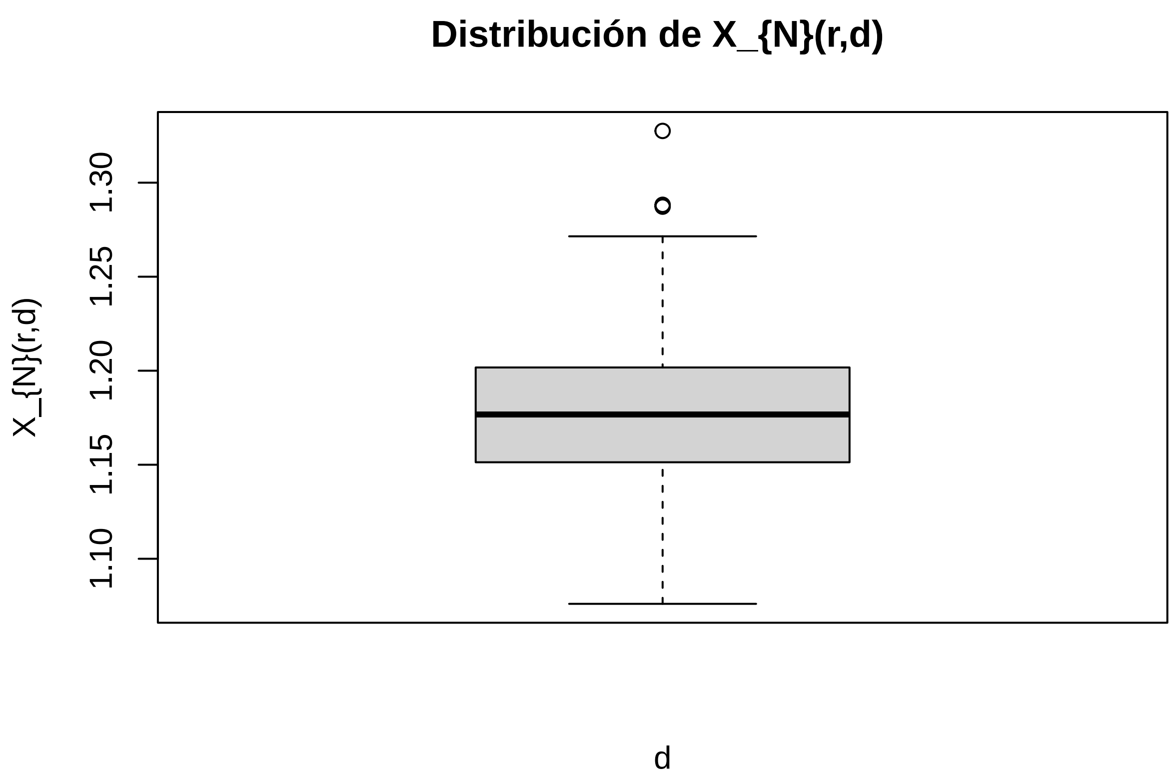
```r
# Calcular X_{N}(r,d) para cada serie
for (i in 1:n_series) {
  series <- ar1_series[i, ]
  X_N_values_ar[i] <- X_N(series, r, d)
  X_N_values_ar_d_1[i] <- X_N_d_1(series, r)
}
```

Veamos como se comporta X_{N}(r,d) para este caso

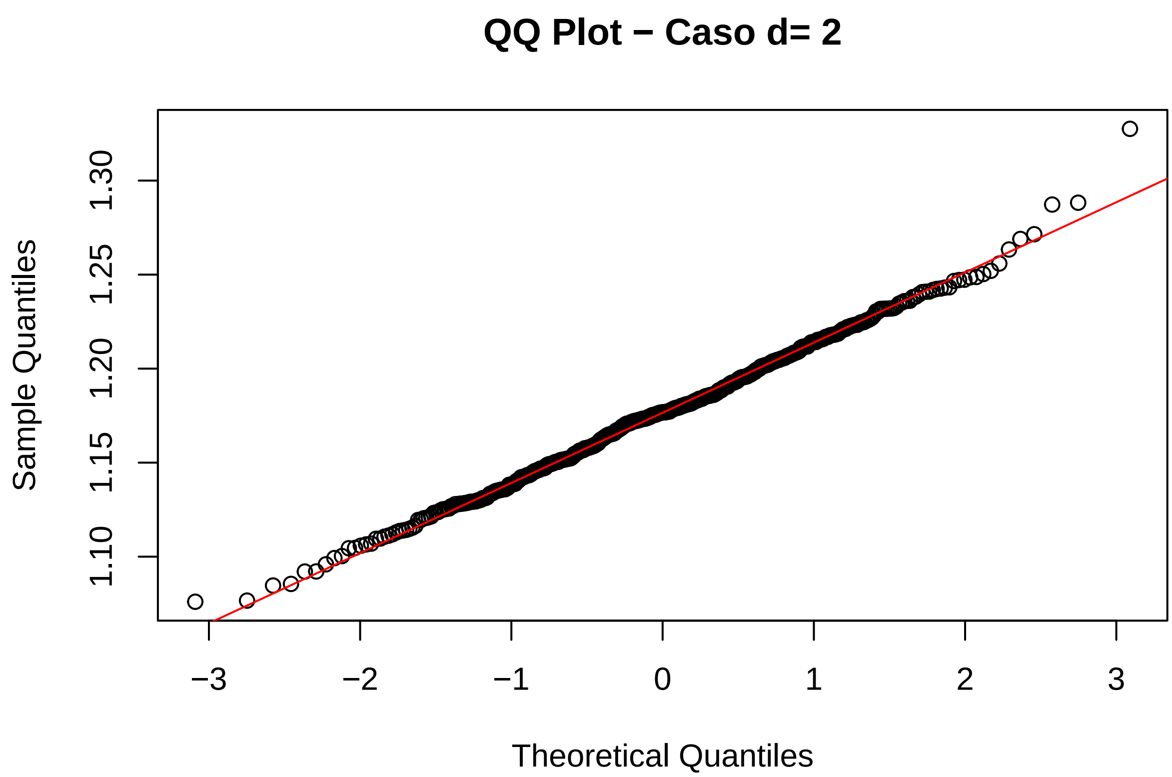
```r
# Resumen de los resultados
summary(X_N_values_ar)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.076   1.151   1.177   1.177   1.202   1.328
```

```r
# Graficar los resultados
boxplot(X_N_values_ar, main="Distribución de X_{N}(r,d) ", ylab="X_{N}(r,d)", xlab="d")
```




```r
# Crear el QQ plot


  qqnorm(X_N_values_ar, main = paste("QQ Plot - Caso d=", 2))
  qqline(X_N_values_ar, col = "red")
```



calculamos $\sqrt{N}ln(X_N(r,d))$

```
# Longitud de cada serie
N <- length_series  # Esto es igual a 1000 en nuestro caso

# Calcular sqrt(N)
sqrt_N <- sqrt(N)

# Calcular sqrt(N) * ln(X_{N}(r,d)) para cada serie
sqrt_N_ln_X_N_ar <- sqrt_N * log(X_N_values_ar)

# Ver los resultados
head(sqrt_N_ln_X_N_ar)
```

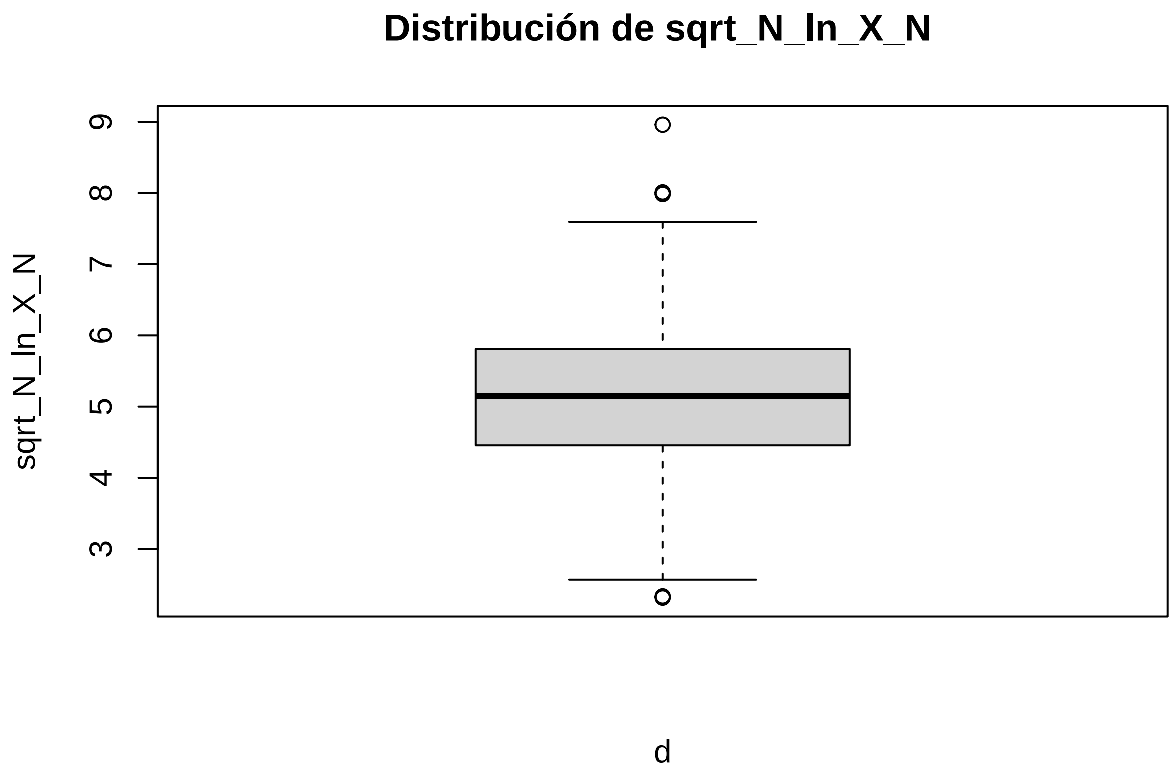
```
## [1] 3.354642 6.557705 5.709658 3.975086 5.022267 3.728973
```

y vemos como se comporta:

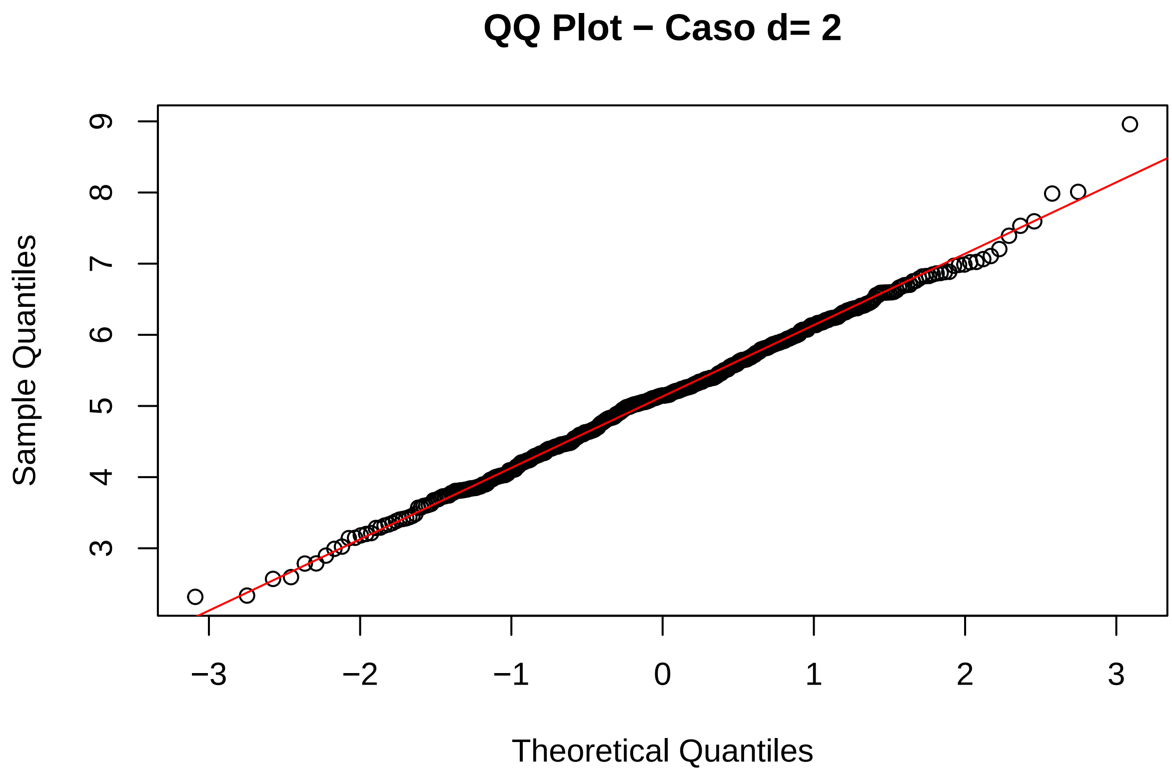
```
# Resumen de los resultados
summary(sqrt_N_ln_X_N_ar)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.318   4.457   5.146   5.127   5.810   8.959
```

```r
# Graficar los resultados
boxplot(sqrt_N_ln_X_N_ar, main="Distribución de sqrt_N_ln_X_N ", ylab="sqrt_N_ln_X_N", xlab="d")
```




```r
# Crear el QQ plot

  qqnorm(sqrt_N_ln_X_N_ar, main = paste("QQ Plot - Caso d=", 2))
  qqline(sqrt_N_ln_X_N_ar, col = "red")
```




```r
# Parámetros
r <- 0.5  # Umbral para la norma
d_values <- 1:5   # Valores de d

# Inicializar listas para almacenar los resultados
X_N_results_ar <- matrix(0, nrow = n_series, ncol = length(d_values))
colnames(X_N_results_ar) <- paste("d =", d_values)

# Calcular X_{N}(r,d) para cada serie y para cada valor de d
for (i in 1:n_series) {
  series <- ar1_series[i, ]

  # Caso especial para d = 1
  X_N_results_ar[i, 1] <- X_N_d_1(series, r)

  # Calcular X_{N}(r,d) para d = 2, 3, 4, 5
  for (d in 2:length(d_values)) {
    X_N_results_ar[i, d] <- X_N(series, r, d_values[d])
  }
}

# Ver los resultados
head(X_N_results_ar)
```

```
##         d = 1    d = 2    d = 3     d = 4     d = 5
## [1,] 1.238919 1.111914 1.313662 1.3424537 1.6410256
```

```
## [2,] 1.275556 1.230441 1.023353 1.1132155 0.6521739
## [3,] 1.242251 1.197882 1.107570 0.7265877 1.8000000
## [4,] 1.244096 1.133946 1.097088 0.9224959 1.1462264
## [5,] 1.264516 1.172125 1.097172 0.8772016 2.1404110
## [6,] 1.289999 1.125155 1.212834 1.0901953 0.7824074
```

```
X_N_results_ar
```

```
##          d = 1    d = 2     d = 3     d = 4     d = 5
##   [1,] 1.238919 1.111914 1.3136615 1.3424537 1.6410256
##   [2,] 1.275556 1.230441 1.0233526 1.1132155 0.6521739
##   [3,] 1.242251 1.197882 1.1075704 0.7265877 1.8000000
##   [4,] 1.244096 1.133946 1.0970881 0.9224959 1.1462264
##   [5,] 1.264516 1.172125 1.0971725 0.8772016 2.1404110
##   [6,] 1.289999 1.125155 1.2128340 1.0901953 0.7824074
##   [7,] 1.265481 1.222888 1.0459549 0.9164386 0.6016667
##   [8,] 1.288363 1.146185 1.1833654 0.8846652 1.5625000
##   [9,] 1.274089 1.185513 1.1722445 1.3335979 1.5845070
##  [10,] 1.256836 1.210946 1.1981380 1.4691541 0.4959016
##  [11,] 1.271937 1.195368 1.1363986 1.1133833 2.3142857
##  [12,] 1.252422 1.176120 1.0973334 0.8758020 1.1758065
##  [13,] 1.256807 1.140210 1.1687146 0.9525581 0.9765625
##  [14,] 1.237335 1.142903 1.3671877 1.0804297       Inf
##  [15,] 1.267807 1.204977 1.2059402 1.1548738 0.3993399
##  [16,] 1.254253 1.125640 1.1546056 0.9216167 1.2787611
##  [17,] 1.294637 1.143749 1.1096019 1.3443983       Inf
##  [18,] 1.282123 1.192686 0.9934455 1.0734434 1.2151899
##  [19,] 1.255130 1.100330 1.1344652 0.6932544 3.2374101
##  [20,] 1.278670 1.177174 1.3273869 1.0581948       Inf
##  [21,] 1.281999 1.151626 1.1918028 1.0340823       Inf
##  [22,] 1.241072 1.154688 1.1917250 1.2989753       Inf
##  [23,] 1.262631 1.202135 1.1080560 1.4190931       Inf
##  [24,] 1.250427 1.157603 1.2251905 0.8415386 1.3241379
##  [25,] 1.260118 1.176206 1.0616387 1.3325047 0.5639098
##  [26,] 1.259961 1.144170 1.0310942 1.3307707 0.8698225
##  [27,] 1.231204 1.175664 1.1560139 1.0633681 1.7857143
##  [28,] 1.288406 1.160354 1.0171459 1.2022727 0.7826087
##  [29,] 1.309157 1.216780 1.0210890 1.0299338       Inf
##  [30,] 1.254120 1.248562 1.0775614 0.9792178 1.1327434
##  [31,] 1.224849 1.180936 1.0120412 1.0705501 1.1011236
##  [32,] 1.284999 1.187172 1.2566905 1.0495081 1.9844961
##  [33,] 1.295756 1.153269 1.2114850 0.8423251 0.6568182
##  [34,] 1.249713 1.218149 1.0969339 1.0170081 0.8437500
##  [35,] 1.274539 1.224884 0.9921413 1.2721963 1.9393939
##  [36,] 1.296856 1.165215 1.2041012 1.4551417 1.1162791
##  [37,] 1.225087 1.133434 1.1030648 1.0378079 0.7245283
##  [38,] 1.254941 1.212780 1.2251279 1.3117836       Inf
##  [39,] 1.270202 1.179170 1.1061828 1.1318823 2.8089172
##  [40,] 1.297022 1.186456 1.2063126 0.6790981 2.8764045
##  [41,] 1.251994 1.241804 1.1272811 1.2080973 0.4671533
##  [42,] 1.291527 1.248746 1.0809322 0.9661603 1.0514019
##  [43,] 1.232872 1.149565 1.1398202 1.1560128 3.2359551
##  [44,] 1.231773 1.243256 1.1106737 1.5307971 0.5538462
##  [45,] 1.267986 1.238031 1.1879429 1.3526298 0.3595506
```

```
##  [46,] 1.261765 1.227198 1.1655768 1.1570513 0.7412281
##  [47,] 1.290785 1.172252 1.0833975 1.4064646       Inf
##  [48,] 1.280373 1.128732 1.1472145 1.3376658 0.4579946
##  [49,] 1.277888 1.134518 1.0663429 0.9472925 1.2367647
##  [50,] 1.227629 1.184356 1.2728356 0.7323095 1.0251938
##  [51,] 1.251890 1.182689 1.0990229 1.1014404 0.6400709
##  [52,] 1.279194 1.164198 1.3814747 2.1291566       Inf
##  [53,] 1.300180 1.146809 1.1815834 2.1089152       Inf
##  [54,] 1.254195 1.170721 1.0934135 0.9678410 0.8521127
##  [55,] 1.266861 1.181293 1.0982801 1.4670061 0.5102041
##  [56,] 1.286890 1.176723 0.9667976 1.8379722 0.4961240
##  [57,] 1.253446 1.133637 1.0968730 1.3812531 2.1746988
##  [58,] 1.206323 1.106952 1.1447218 0.8693623 0.5619883
##  [59,] 1.285220 1.246608 1.1476636 0.8293610 1.1479592
##  [60,] 1.260501 1.169991 1.0596857 1.0412322 0.8000000
##  [61,] 1.245242 1.185430 1.0070881 1.8585012       Inf
##  [62,] 1.239945 1.189315 1.1108173 1.0641727 1.2112403
##  [63,] 1.275220 1.181200 1.1458469 0.9713977 1.2041667
##  [64,] 1.257654 1.152047 1.0092981 1.4706847 1.1203704
##  [65,] 1.261876 1.151738 1.1995743 1.1432738 1.5089286
##  [66,] 1.277393 1.252007 1.0965693 0.9906367       Inf
##  [67,] 1.286295 1.125567 1.3088252 1.1549494 1.2972973
##  [68,] 1.283478 1.215301 1.1111767 1.1379240 1.9844961
##  [69,] 1.264084 1.169540 1.1948743 1.4805339 1.0000000
##  [70,] 1.246303 1.236023 1.2033804 2.5001858       Inf
##  [71,] 1.250505 1.195800 1.1528711 0.9623815 0.4491228
##  [72,] 1.282020 1.215215 1.1979978 1.3944836       Inf
##  [73,] 1.280285 1.131218 1.0563557 1.4309882 0.8767123
##  [74,] 1.309394 1.179099 1.0621641 0.8674396 0.9869403
##  [75,] 1.244063 1.114823 1.1767551 1.3345146 0.8121547
##  [76,] 1.254094 1.214194 1.0848146 2.1301775       Inf
##  [77,] 1.251864 1.190017 1.2020154 1.4302600 0.9090909
##  [78,] 1.274974 1.218091 1.1450961 1.0848813 1.2448276
##  [79,] 1.261529 1.159757 1.1978370 1.1909796 0.5769231
##  [80,] 1.249312 1.192659 1.2110598 1.0454780       Inf
##  [81,] 1.263908 1.126807 1.0835909 0.9132738 1.1118421
##  [82,] 1.288158 1.178977 1.2367145 0.9831759 0.5688976
##  [83,] 1.257728 1.188096 1.0268743 1.3727829       Inf
##  [84,] 1.266459 1.138825 1.2409645 1.4475470       Inf
##  [85,] 1.262741 1.158051 1.1918765 0.9966973 2.4923077
##  [86,] 1.272861 1.219444 1.2058096 1.4037123 0.4545455
##  [87,] 1.316925 1.231836 1.2926633 1.0425532 0.9642857
##  [88,] 1.244686 1.231963 1.1935492 0.9434698 0.6818182
##  [89,] 1.238339 1.190561 1.1039117 1.2224946 1.9104478
##  [90,] 1.318687 1.169399 1.1317309 1.0557269 0.8203883
##  [91,] 1.250877 1.230445 1.0985354 0.9032067 0.6923077
##  [92,] 1.237033 1.204559 1.1018522 1.1564820 1.0703704
##  [93,] 1.281623 1.176759 1.1127184 1.0272222 2.5116279
##  [94,] 1.255841 1.138493 1.0750165 1.5097262 1.9171598
##  [95,] 1.242766 1.207457 1.3127863 1.2742055 1.2900763
##  [96,] 1.259390 1.242612 0.9649716 1.3042095 0.5547945
##  [97,] 1.277788 1.223325 1.2287221 1.2426471 0.6153846
##  [98,] 1.333467 1.247115 1.0660505 0.8450914 4.9620253
##  [99,] 1.272499 1.221070 1.1765987 0.8823696 0.7220000
```

```
## [100,] 1.253592 1.154974 1.1906038 1.2495352       Inf
## [101,] 1.253228 1.211624 1.1901198 1.0690305 1.3955696
## [102,] 1.248174 1.165003 1.0353426 0.8807220 0.7135802
## [103,] 1.307415 1.170270 1.3177986 1.5001025       Inf
## [104,] 1.238372 1.175505 1.2179391 1.0109515 0.9735099
## [105,] 1.247708 1.184866 1.1535684 0.9872747 0.6785714
## [106,] 1.260659 1.172821 1.0811831 0.9960614 0.9290323
## [107,] 1.280575 1.263345 1.0751055 1.5824176       Inf
## [108,] 1.285037 1.126978 1.0984852 0.9856840 1.1282051
## [109,] 1.273767 1.128581 1.2803339 1.1237311 0.5859375
## [110,] 1.293468 1.171775 1.1422205 1.6070950       Inf
## [111,] 1.294431 1.224596 1.2395805 0.8389877 1.0922330
## [112,] 1.242319 1.115441 1.1267772 1.1587187 2.1729323
## [113,] 1.277939 1.167780 1.1033910 1.0940325       Inf
## [114,] 1.277875 1.151525 1.2714914 1.1300098 1.8823529
## [115,] 1.273915 1.250308 1.0616502 0.8742629 2.0281690
## [116,] 1.272169 1.179387 1.1526046 0.8059317 1.2150000
## [117,] 1.234094 1.129496 1.1052698 0.8465745 1.1578313
## [118,] 1.259961 1.191331 1.4001166 0.9264138 0.9533898
## [119,] 1.306572 1.199758 1.2763892 1.2108642 1.1100917
## [120,] 1.248995 1.211637 1.0421446 1.7232673 0.8474576
## [121,] 1.268471 1.195836 1.0527728 0.9893878 0.9475066
## [122,] 1.293953 1.172087 1.1514543 1.0285714 2.1333333
## [123,] 1.285609 1.179110 1.1521815 0.8067509 0.5975610
## [124,] 1.258105 1.164484 1.0923496 1.3655462 0.7538462
## [125,] 1.244246 1.138201 1.1900589 1.1225116 0.5512500
## [126,] 1.268362 1.188536 1.0024304 1.0431759 0.5840278
## [127,] 1.287442 1.206812 1.2383794 0.9328524 2.3495935
## [128,] 1.259760 1.150468 1.1234288 1.2209351 0.9851024
## [129,] 1.267859 1.173375 1.0810090 1.3039254 0.6372832
## [130,] 1.290431 1.131643 0.9912923 1.3198368 2.2344828
## [131,] 1.275876 1.129387 1.2103624 1.2666667 1.2631579
## [132,] 1.252761 1.176311 1.1866508 1.2065997 0.6643678
## [133,] 1.256106 1.203270 1.1182297 1.2644704 0.6486486
## [134,] 1.246991 1.145593 1.1742066 1.2870196       Inf
## [135,] 1.265592 1.113949 1.1748967 1.5348669 0.3932039
## [136,] 1.244445 1.152161 1.1946539 1.0301072       Inf
## [137,] 1.282134 1.213994 1.1456824 2.1588946       Inf
## [138,] 1.274187 1.131189 1.1924009 1.6723596       Inf
## [139,] 1.258443 1.145126 1.0692219 1.4883659 0.6467066
## [140,] 1.261379 1.225645 1.1228316 1.1777347 0.4985251
## [141,] 1.240342 1.185481 1.0336736 1.0483763 0.5184000
## [142,] 1.295147 1.204582 1.0725939 1.4711202       Inf
## [143,] 1.266561 1.225917 1.2268880 2.0227395       Inf
## [144,] 1.261562 1.208518 1.0558839 0.8253904 2.8613861
## [145,] 1.266581 1.201392 1.1163561 1.2219655 0.8698225
## [146,] 1.177855 1.148725 1.0487622 1.1123669 1.4202899
## [147,] 1.257252 1.217945 1.0803040 1.1678528       Inf
## [148,] 1.304024 1.111206 1.1557270 1.2392693 1.5934959
## [149,] 1.274524 1.158919 1.2032964 1.1062849       Inf
## [150,] 1.266627 1.123503 1.1831552 0.9044565 0.8705036
## [151,] 1.292639 1.110665 1.0464007 1.1188701 0.7908497
## [152,] 1.296922 1.165599 1.1768780 0.8951646 2.5350877
## [153,] 1.269082 1.109487 1.1763867 1.0457809 1.1468254
```

```
## [154,] 1.264076 1.175221 1.1539497 1.0964088 0.6772487
## [155,] 1.267111 1.174317 1.1844262 1.2707248 0.6926230
## [156,] 1.263075 1.175104 1.1645464 1.9363337       Inf
## [157,] 1.262260 1.092083 1.0436724 0.9664430 1.3333333
## [158,] 1.263701 1.243085 1.0402381 1.5346839 0.8100000
## [159,] 1.239440 1.136308 1.1067563 1.1597660 2.6490066
## [160,] 1.230917 1.173615 1.1999426 1.0301772 0.5245902
## [161,] 1.276475 1.175803 1.1261920 1.5239708 0.5041667
## [162,] 1.237504 1.201252 1.4671932 0.5485075 1.5238095
## [163,] 1.281946 1.152072 1.0802817 1.2783022 1.0104895
## [164,] 1.242305 1.136700 1.0092723 1.0225269 1.6111111
## [165,] 1.298526 1.189974 1.1844031 1.4432699 0.5625000
## [166,] 1.266569 1.119519 1.1425329 1.2016477 0.3772321
## [167,] 1.274687 1.189479 1.1902676 0.8701412 0.9152542
## [168,] 1.271320 1.184035 1.1718761 0.8507812       Inf
## [169,] 1.298452 1.202865 1.1049622 0.9540703 0.8233618
## [170,] 1.254422 1.121441 1.1303888 1.2940299 2.1176471
## [171,] 1.279399 1.146121 1.1514470 1.1862259 1.9248120
## [172,] 1.287679 1.240987 1.1320946 0.8448352 1.6129032
## [173,] 1.301767 1.199177 1.1915732 1.6351726       Inf
## [174,] 1.259616 1.183127 1.1820623 1.2308561 0.7100840
## [175,] 1.269027 1.235869 1.1968845 0.6919285 2.8131868
## [176,] 1.246569 1.203876 1.1022669 1.1029729 0.7562500
## [177,] 1.217399 1.195564 1.0277819 1.1399036 0.8163580
## [178,] 1.245063 1.191776 1.1135819 1.0519033 0.6993007
## [179,] 1.276521 1.149274 1.0502435 2.0683347       Inf
## [180,] 1.259837 1.180168 1.2566283 1.0472759 1.7043478
## [181,] 1.202440 1.204664 1.2387663 0.8400521 0.6299213
## [182,] 1.295141 1.190399 1.2079175 1.0437283 2.1893939
## [183,] 1.257153 1.170817 1.1447001 1.2334199 0.7152318
## [184,] 1.271997 1.167582 1.1954365 0.9296149 1.0500000
## [185,] 1.291098 1.247290 1.1517430 0.7408009       Inf
## [186,] 1.240943 1.123793 1.0816144 1.0923567 1.0000000
## [187,] 1.259724 1.146942 1.0896977 1.2861492 0.5769231
## [188,] 1.261931 1.179258 1.0643054 0.9004329 1.5576923
## [189,] 1.238133 1.116350 1.0433076 1.2613749       Inf
## [190,] 1.225566 1.171040 1.2083787 1.0455795 0.8474178
## [191,] 1.251665 1.156419 1.0996708 0.8442872 0.7986726
## [192,] 1.269660 1.148437 1.0447819 0.9456662 1.6961538
## [193,] 1.259224 1.201791 1.0925107 1.2211643 0.4776119
## [194,] 1.237590 1.195697 1.0620217 1.1672774       Inf
## [195,] 1.276393 1.181284 1.0402931 1.3965030 0.5474453
## [196,] 1.254621 1.238308 1.1351439 1.1373940 0.9770992
## [197,] 1.241434 1.185967 1.2067812 0.6511410 0.8135593
## [198,] 1.298914 1.106537 1.1963080 0.9944310 0.5067568
## [199,] 1.262359 1.163852 1.1091038 1.3398954 1.9139073
## [200,] 1.252546 1.154440 1.0463085 1.0389344 0.9230769
## [201,] 1.257984 1.149353 1.1918070 1.0153426 0.9324009
## [202,] 1.281473 1.176602 1.3067568 1.3413544       Inf
## [203,] 1.302460 1.095943 1.0268814 1.8208188 1.0510949
## [204,] 1.230985 1.158917 1.0468014 1.2273482 1.5377907
## [205,] 1.295229 1.186181 1.1652314 1.4313272 1.2611940
## [206,] 1.245419 1.143372 1.1442055 1.6397376 1.3888889
## [207,] 1.247091 1.163278 1.1206059 0.9584158 3.0303030
```

```
## [208,] 1.253902 1.164876 0.9545352 1.5204790 1.6304348
## [209,] 1.305414 1.223138 1.1661765 0.8135754 0.7731959
## [210,] 1.248478 1.138231 1.1636794 1.0772183 0.9070295
## [211,] 1.246209 1.199973 1.1199152 0.9377990 0.9550265
## [212,] 1.218805 1.138595 1.0918550 1.0987596 1.1020833
## [213,] 1.277576 1.180141 1.2071781 1.6496281 0.5268817
## [214,] 1.234860 1.206963 1.2484607 1.0360355 0.6719160
## [215,] 1.277735 1.201182 1.1635606 0.7671114       Inf
## [216,] 1.294300 1.215706 1.1306173 1.4273894 1.0431034
## [217,] 1.273560 1.190535 1.2219083 1.2640644       Inf
## [218,] 1.259994 1.174631 1.1828774 1.3979915 1.0521739
## [219,] 1.318473 1.224637 1.0660589 1.1673611 0.9146341
## [220,] 1.268436 1.242370 1.3063423 0.9920853 1.2736842
## [221,] 1.234704 1.151243 1.0559044 1.2147094 0.5000000
## [222,] 1.269412 1.192575 1.0410381 1.5094196       Inf
## [223,] 1.270786 1.187194 1.1080669 1.6628599 0.6322581
## [224,] 1.257924 1.152262 1.0756475 1.2033842 2.4391892
## [225,] 1.259004 1.202442 1.2563913 0.9359909 1.5921053
## [226,] 1.330877 1.194874 1.0668728 1.2640760 0.9343066
## [227,] 1.253389 1.194600 1.1552030 0.8537409 1.0112045
## [228,] 1.281396 1.213949 1.2878230 0.9853042 1.3010753
## [229,] 1.250064 1.076040 0.9794616 1.3890251 1.8044554
## [230,] 1.240124 1.140210 1.1831711 1.0922310 1.4134615
## [231,] 1.317108 1.207798 1.1001510 1.7575928 0.4000000
## [232,] 1.265565 1.135040 0.9804180 1.2096062 0.6490798
## [233,] 1.268504 1.150066 1.2133686 1.3377926 1.2000000
## [234,] 1.263295 1.222092 1.1922528 1.4659706       Inf
## [235,] 1.254446 1.196096 1.2005644 1.7679558 0.6750000
## [236,] 1.218239 1.171473 1.1462318 1.1959971 0.6880952
## [237,] 1.271276 1.176967 1.1743391 0.9574625 1.7287582
## [238,] 1.274650 1.150750 1.1511500 1.1111598 1.3245033
## [239,] 1.276488 1.183198 1.1089618 1.4081645 1.6953642
## [240,] 1.285981 1.209057 0.9803639 0.7401015 4.0000000
## [241,] 1.267466 1.162135 1.1825223 0.8485826 1.4727669
## [242,] 1.254619 1.154766 1.2353157 1.0474860 1.3333333
## [243,] 1.261185 1.130315 1.0369613 0.9218110 2.0828829
## [244,] 1.280027 1.172387 1.2856996 1.2658228       Inf
## [245,] 1.275886 1.158893 1.3128701 0.8501707 0.7378049
## [246,] 1.264190 1.186029 1.1688277 1.0595369 0.6881720
## [247,] 1.282133 1.142312 1.2326884 1.2594905 1.3739837
## [248,] 1.280736 1.155606 1.1410383 1.0878125 1.1770245
## [249,] 1.275199 1.205237 0.9601772 0.9053118 1.1160714
## [250,] 1.235979 1.232256 1.1485536 1.8572324       Inf
## [251,] 1.269699 1.165029 1.1816078 1.4497653 0.6259259
## [252,] 1.274048 1.142717 1.0928001 0.9970344 0.8780488
## [253,] 1.261770 1.217803 1.1374829 1.1071649 2.1567164
## [254,] 1.283670 1.157653 1.1870255 1.0251938       Inf
## [255,] 1.287844 1.123213 1.1788461 0.9846949 0.9661836
## [256,] 1.263299 1.222631 1.1680072 1.1753658       Inf
## [257,] 1.269522 1.178678 1.2247089 0.9972301 0.4495413
## [258,] 1.277814 1.146758 1.1821536 4.9481928       Inf
## [259,] 1.241748 1.173385 1.1357468 0.9809783 3.4802632
## [260,] 1.223648 1.166956 1.0881404 1.0363120 0.6870130
## [261,] 1.272119 1.156829 1.2439261 0.8333333 1.1250000
```

```
## [262,] 1.274963 1.114262 1.3090899 1.0876344 0.5490196
## [263,] 1.242822 1.104499 1.0901897 1.2832733 3.4141414
## [264,] 1.291655 1.149074 1.0327581 0.8172637 0.7110215
## [265,] 1.295867 1.170928 1.0945007 1.5444947       Inf
## [266,] 1.284867 1.127948 1.1134764 0.9583052 1.1442080
## [267,] 1.278533 1.177042 1.1908893 0.8984720 1.4325397
## [268,] 1.244133 1.161208 1.1653782 1.0712178       Inf
## [269,] 1.275838 1.147362 1.1912320 0.9261168 0.6233766
## [270,] 1.248688 1.232001 1.1252747 1.2140805 1.7307692
## [271,] 1.226150 1.170614 1.0651171 0.8984009 1.6436059
## [272,] 1.256002 1.135207 1.0754420 0.9375000 1.0416667
## [273,] 1.227488 1.162365 1.0338361 0.8813068 1.6308725
## [274,] 1.276889 1.176744 1.2096767 0.9513575 1.1034483
## [275,] 1.255847 1.193385 0.9761117 1.2775518 0.8044693
## [276,] 1.307367 1.165385 1.2139525 1.1034594 1.9541985
## [277,] 1.253173 1.157656 1.1038030 0.9143184 2.2781250
## [278,] 1.290817 1.176473 0.9300230 1.6857251       Inf
## [279,] 1.289275 1.151458 1.1468005 1.9102273 0.6585366
## [280,] 1.298710 1.128424 1.2133852 1.0083333 0.8181818
## [281,] 1.262470 1.220800 1.1930248 0.6727442 1.1134021
## [282,] 1.287305 1.099244 1.0895592 1.1946960 0.9297521
## [283,] 1.261445 1.160155 0.9434286 1.3462560 0.8355615
## [284,] 1.319336 1.145768 1.0427227 1.2178417 0.5661232
## [285,] 1.293289 1.188421 1.1651955 1.4366385 0.9174312
## [286,] 1.282834 1.185274 1.1603371 1.0195222 2.2936508
## [287,] 1.310155 1.153333 1.0646755 0.9139241 0.9473684
## [288,] 1.257719 1.173032 1.3290462 3.3654933 0.1022727
## [289,] 1.279459 1.190115 1.2464546 0.8976064       Inf
## [290,] 1.263028 1.211357 1.0029904 1.4810294 0.7313433
## [291,] 1.258436 1.180975 1.1981392 1.7063721       Inf
## [292,] 1.290941 1.197071 1.0493777 0.8552826 0.6779661
## [293,] 1.298176 1.172672 1.2342655 1.1384731       Inf
## [294,] 1.217577 1.169271 1.1312873 0.9034920 1.2367647
## [295,] 1.250283 1.128230 1.1270426 0.9138340 1.7794118
## [296,] 1.290146 1.213327 1.0849279 1.2030060 0.6416040
## [297,] 1.268515 1.240889 1.0463207 1.2359387 1.7716535
## [298,] 1.281228 1.198230 1.1694347 2.3563636       Inf
## [299,] 1.258295 1.142266 1.1805700 1.0983982 0.9375000
## [300,] 1.252389 1.128005 1.1680161 1.2720216       Inf
## [301,] 1.280028 1.156575 1.0015104 1.1940019 1.2380952
## [302,] 1.247436 1.179553 1.0668817 1.1193238 2.0523560
## [303,] 1.292580 1.194224 1.0478888 1.0866935 3.1428571
## [304,] 1.292937 1.184017 1.1817532 1.2923899       Inf
## [305,] 1.287503 1.183977 1.2336951 0.6214590 1.6420455
## [306,] 1.266133 1.214214 1.1048651 1.1438336 1.5504587
## [307,] 1.253206 1.173868 1.2339494 0.8616300 1.0103093
## [308,] 1.296886 1.120388 1.1400201 1.2740541 0.3920000
## [309,] 1.257137 1.145506 1.0699871 1.1192736       Inf
## [310,] 1.244457 1.121010 0.9900711 0.9393565 0.9655629
## [311,] 1.227627 1.135459 1.0636272 0.8944622 0.9358289
## [312,] 1.279584 1.161709 1.1662713 1.1972329 0.8032787
## [313,] 1.302648 1.193285 1.3405849 0.9009097 1.6732673
## [314,] 1.257792 1.174397 1.3130689 1.0327249 1.1235955
## [315,] 1.284557 1.157863 1.0334260 1.6714573 0.2596154
```

```
## [316,] 1.300488 1.208389 1.0855846 0.8351648 1.5833333
## [317,] 1.257578 1.193103 1.2608728 1.3056140       Inf
## [318,] 1.271248 1.165737 1.0368238 0.8514371 0.7751152
## [319,] 1.261181 1.255890 1.1966159 1.5859756       Inf
## [320,] 1.262813 1.182593 1.0832142 1.1265372 1.9067797
## [321,] 1.260837 1.201945 1.0624638 0.9480551 0.7223114
## [322,] 1.257896 1.129286 1.0369859 0.8705478 1.2469775
## [323,] 1.248455 1.135890 1.0487241 0.8754301 0.6405229
## [324,] 1.293024 1.195437 1.2859616 1.0616667 1.0989011
## [325,] 1.234509 1.162835 1.1463310 1.1501255       Inf
## [326,] 1.317232 1.235140 1.1128094 1.0797080 0.2325581
## [327,] 1.251173 1.236116 1.0886325 1.2830479 1.4201681
## [328,] 1.255132 1.192163 1.1830089 1.3698630 1.3000000
## [329,] 1.224521 1.205344 1.1577409 1.3086935 0.4383562
## [330,] 1.279509 1.152305 1.2634510 1.3767092       Inf
## [331,] 1.274913 1.165370 1.1730548 1.2896825       Inf
## [332,] 1.298864 1.180322 1.0442879 0.9211003 0.6342926
## [333,] 1.275078 1.192092 1.1478954 0.9001815 2.9112903
## [334,] 1.280883 1.185246 1.2292348 0.9662254 2.1512605
## [335,] 1.278466 1.174380 1.0824186 1.1014454 1.1289062
## [336,] 1.269716 1.173354 1.2832582 0.9715370 1.1289062
## [337,] 1.240639 1.156502 1.1854342 1.0452247 0.6994536
## [338,] 1.258371 1.201667 1.1717907 1.3878613 1.5306122
## [339,] 1.252215 1.178206 1.0706544 1.0619146 0.9768786
## [340,] 1.231625 1.206380 1.2757134 1.0883266 0.8858268
## [341,] 1.265395 1.203226 1.0696674 0.7764558 1.3557423
## [342,] 1.242266 1.113635 1.2483577 1.1984646 0.5328947
## [343,] 1.282028 1.205893 1.3821103 0.9049145       Inf
## [344,] 1.266862 1.167049 1.1044570 0.6658765 1.5220126
## [345,] 1.254446 1.208812 0.9682099 1.2599031 2.6143791
## [346,] 1.245126 1.182186 1.1366594 1.3143557       Inf
## [347,] 1.257963 1.158316 1.3011308 1.0142795 0.4979310
## [348,] 1.272386 1.183333 1.0072705 1.2718305 0.4152542
## [349,] 1.264893 1.231881 1.1724305 0.8298851 0.7105263
## [350,] 1.252504 1.177240 1.2212375 1.1897167       Inf
## [351,] 1.301808 1.186708 1.2192208 1.1960336 1.2857143
## [352,] 1.268811 1.204184 1.0227038 0.8584754 1.6418919
## [353,] 1.282035 1.172294 1.0598235 1.1276676 0.8064024
## [354,] 1.243609 1.159861 1.1681872 1.2464093 1.1281250
## [355,] 1.285855 1.173529 1.3010918 0.8003807 1.3965517
## [356,] 1.281081 1.152572 1.1551647 1.0782325 1.1911765
## [357,] 1.222643 1.173875 0.9964129 0.8376923 2.3674242
## [358,] 1.288712 1.231717 1.1634404 0.8996328 1.0714286
## [359,] 1.272131 1.178698 1.0902119 1.2387761       Inf
## [360,] 1.265003 1.150378 1.0339945 0.9973404 0.6400000
## [361,] 1.272688 1.163549 1.1908227 0.8851596 1.0335917
## [362,] 1.244679 1.171745 1.1335278 1.3579138 2.1360947
## [363,] 1.269801 1.172404 1.0212308 0.9366516 3.8333333
## [364,] 1.278235 1.228438 0.9580432 0.8553418 1.3994709
## [365,] 1.243683 1.214189 1.2470273 1.3797035 0.9090909
## [366,] 1.298299 1.143396 1.2134181 1.0560030 1.8907563
## [367,] 1.293750 1.175412 1.1312638 1.7217247       Inf
## [368,] 1.255062 1.144750 1.0876152 1.1176370 2.5785714
## [369,] 1.294703 1.176745 1.2164051 0.9962335 1.9565217
```

```
## [370,] 1.270213 1.173121 1.0354703 1.0018680 0.8730539
## [371,] 1.248337 1.209455 1.0747008 1.0084469 1.6462585
## [372,] 1.294685 1.177954 1.1479300 1.3414634 1.1000000
## [373,] 1.228676 1.218507 1.1672694 1.2882448 0.6750000
## [374,] 1.200808 1.215686 1.2558821 1.1377910 1.8686131
## [375,] 1.255411 1.128224 1.0907508 1.2216867 0.6410256
## [376,] 1.210427 1.129218 1.0726372 1.2899425 0.3184080
## [377,] 1.263680 1.194946 1.3039302 1.4470402 0.8547009
## [378,] 1.270518 1.185960 1.3647753 2.2864799       Inf
## [379,] 1.261135 1.175471 1.1870019 1.4408254 1.2518519
## [380,] 1.286889 1.218619 1.1256684 1.2382392 0.4774011
## [381,] 1.240226 1.206800 1.1185423 0.8726447 1.0752688
## [382,] 1.305478 1.171552 1.0895139 1.5541111 1.2335766
## [383,] 1.260009 1.158189 1.1892424 1.1127820 1.2195946
## [384,] 1.253608 1.220090 1.3315553 0.9312154 1.4845361
## [385,] 1.235592 1.287259 1.1523631 1.1340369 0.8166667
## [386,] 1.288654 1.208554 1.1425371 1.3007530 0.8804348
## [387,] 1.275511 1.327516 1.2977788 0.9057508 0.3888889
## [388,] 1.248595 1.203870 1.1144654 1.2494718 0.7903226
## [389,] 1.266168 1.130027 1.1665059 1.7211604 0.8130081
## [390,] 1.190902 1.207590 1.1539398 0.7983368 1.5648148
## [391,] 1.281428 1.216926 1.2193794 1.3125000       Inf
## [392,] 1.241061 1.192778 1.0657211 1.2378947 0.4464286
## [393,] 1.266838 1.151815 1.2059452 1.4990391 1.0992366
## [394,] 1.275182 1.181730 1.0641001 1.0280632 1.1525054
## [395,] 1.267456 1.239491 1.1463290 1.0536512 0.9533898
## [396,] 1.212861 1.125238 0.9991010 1.6551724 1.1124339
## [397,] 1.263962 1.177293 1.1606378 0.9263080 1.3185654
## [398,] 1.273970 1.076644 1.0425151 1.3267851 0.8855422
## [399,] 1.233192 1.199160 1.0715391 1.3524471 0.4475524
## [400,] 1.265359 1.140746 1.1274152 0.7406918 0.7831978
## [401,] 1.263093 1.223936 0.9874089 0.7442639 1.4019231
## [402,] 1.258059 1.164667 1.0339304 1.1387280 1.2180451
## [403,] 1.242482 1.105836 1.1506469 1.4963872 1.3900709
## [404,] 1.258780 1.135607 1.1520530 0.9740276 0.8709677
## [405,] 1.276163 1.186045 1.4125168 1.3243243       Inf
## [406,] 1.275223 1.135912 1.0155280 1.3650024 1.0728477
## [407,] 1.261574 1.185950 1.2309695 1.1796001       Inf
## [408,] 1.242788 1.211809 1.2021780 1.2813679       Inf
## [409,] 1.269486 1.150370 1.2142555 1.2734203 0.8152174
## [410,] 1.249489 1.176663 1.0614143 0.9177841 0.8836601
## [411,] 1.244358 1.167955 1.1343723 1.0981549 3.1867470
## [412,] 1.276256 1.268947 1.0797515 1.2509138 1.1747573
## [413,] 1.273439 1.141687 1.0879028 1.1605263 2.0317460
## [414,] 1.266308 1.131104 1.0983005 1.2282393       Inf
## [415,] 1.251408 1.156655 1.1550571 0.9040997 0.7130802
## [416,] 1.284997 1.271480 1.1976442 2.1954566       Inf
## [417,] 1.272450 1.184716 1.1718528 1.2845186       Inf
## [418,] 1.285690 1.167163 1.1808370 1.5197388       Inf
## [419,] 1.225766 1.143281 1.0533763 1.1080892 0.9245562
## [420,] 1.258216 1.234526 0.9917793 0.8725954 0.9603175
## [421,] 1.229930 1.216942 1.0895262 1.1172070 0.5833333
## [422,] 1.270656 1.196855 1.0104502 1.2280528 0.9221311
## [423,] 1.284359 1.147158 1.1172916 0.8542453 0.8820000
```

```
## [424,] 1.229671 1.112871 1.0643998 1.1239212 1.1421320
## [425,] 1.265138 1.149179 1.2039200 1.2250000 0.9142857
## [426,] 1.256740 1.134875 1.1718575 1.5906546       Inf
## [427,] 1.251808 1.120632 1.1864241 1.2224467 1.5934959
## [428,] 1.264569 1.203948 1.0660030 1.2389061 1.1020408
## [429,] 1.283416 1.170773 1.0931599 0.9787920 2.8203125
## [430,] 1.234251 1.162654 1.1060164 1.3063902 0.4914966
## [431,] 1.244132 1.159242 1.0706410 2.1737394       Inf
## [432,] 1.275704 1.180264 1.3375471 1.0126957 1.3980583
## [433,] 1.266486 1.085527 1.1012058 0.9142184 4.0563380
## [434,] 1.236962 1.129065 1.1268569 1.1048951 1.5316456
## [435,] 1.238421 1.205573 1.1936904 0.9537919       Inf
## [436,] 1.265576 1.119583 1.3017383 0.9868720 0.9698276
## [437,] 1.287957 1.197591 1.0708756 1.1838955       Inf
## [438,] 1.247151 1.179722 1.1169593 1.1757815 1.0547445
## [439,] 1.269604 1.176966 1.1589107 1.8389048 0.9473684
## [440,] 1.250973 1.109672 1.0646131 1.1119949 1.1952736
## [441,] 1.243295 1.185538 1.1832130 1.5156946 0.8304598
## [442,] 1.292642 1.240696 1.0289133 1.2131000       Inf
## [443,] 1.266401 1.157913 1.1650723 1.1345332       Inf
## [444,] 1.259771 1.172754 1.0769725 1.7822334 0.6125000
## [445,] 1.321138 1.149766 1.0923206 0.9065680 1.2595238
## [446,] 1.278675 1.168344 1.1651191 0.8912467 1.5555556
## [447,] 1.276378 1.211666 1.0841055 1.7347319       Inf
## [448,] 1.275938 1.180606 1.0873051 1.1752119 0.3968992
## [449,] 1.281882 1.158512 1.0697721 0.9160779 0.6342926
## [450,] 1.297324 1.216095 1.1882124 2.2508013       Inf
## [451,] 1.270622 1.181493 1.2197264 0.9646291       Inf
## [452,] 1.283520 1.221127 1.0950110 1.3227727       Inf
## [453,] 1.245288 1.196724 1.2605734 1.3084411       Inf
## [454,] 1.272755 1.150323 1.2116734 0.8417441 2.0973154
## [455,] 1.238097 1.195469 1.0874992 1.2926622 1.9794521
## [456,] 1.253393 1.142294 1.0220889 1.3385052 0.6914286
## [457,] 1.286915 1.196464 1.2477040 1.1445652 0.7222222
## [458,] 1.238686 1.151400 1.0835463 1.4713515 0.5186782
## [459,] 1.268085 1.233071 1.0266054 1.2609551 1.9104478
## [460,] 1.275018 1.222053 1.1822225 1.6696901 0.5212766
## [461,] 1.284848 1.084664 1.1167412 1.0252772 0.9803922
## [462,] 1.281652 1.173248 1.0769163 0.9685907 1.3403141
## [463,] 1.257653 1.155499 1.2312000 1.0930391 0.8602151
## [464,] 1.233299 1.166172 1.0516814 1.0999703 1.3272059
## [465,] 1.271669 1.177873 1.0049029 1.1759440 0.8470588
## [466,] 1.279518 1.184018 1.1621596 1.5756252       Inf
## [467,] 1.280706 1.156295 1.0070132 1.2323502 1.0078740
## [468,] 1.276737 1.124583 1.2875139 1.2711312 0.7313433
## [469,] 1.268727 1.135727 1.2455657 0.7233562 2.9489796
## [470,] 1.230864 1.205669 1.2370206 1.9674119 0.6090226
## [471,] 1.266391 1.138640 1.1064233 1.3672624 0.9506579
## [472,] 1.271475 1.199409 1.0586998 1.6142857       Inf
## [473,] 1.287173 1.198160 1.2401010 0.7161965 1.0384615
## [474,] 1.290693 1.146972 1.0557725 1.0166272 0.7931655
## [475,] 1.270020 1.132653 1.0705882 1.4358974 1.6071429
## [476,] 1.281080 1.223334 1.1802878 1.7773208       Inf
## [477,] 1.299741 1.188464 1.2028048 1.0314074 1.9067797
```

```
## [478,] 1.266769 1.183035 1.1587313 0.9601440 1.0975057
## [479,] 1.243164 1.181592 1.0808073 1.2402556 1.3283133
## [480,] 1.239675 1.129969 1.3569338 1.5438278       Inf
## [481,] 1.265335 1.231766 1.3137257 1.2053455 0.7441860
## [482,] 1.253085 1.288233 0.9360150 1.2713478 0.6302083
## [483,] 1.300586 1.168578 1.1373851 1.4697237 0.8100000
## [484,] 1.227435 1.182422 1.1876026 1.0832272 1.0971429
## [485,] 1.280901 1.092150 1.2299660 1.0394747 0.4849138
## [486,] 1.269365 1.211806 1.0553024 1.4285512       Inf
## [487,] 1.288620 1.104664 1.0524075 1.4224501 2.0506329
## [488,] 1.256502 1.175262 1.1672037 0.7839153 0.9680000
## [489,] 1.281281 1.215328 1.0939990 0.9045825       Inf
## [490,] 1.278976 1.188449 0.9872472 1.2075007       Inf
## [491,] 1.291587 1.201870 1.2363474 0.9532661 1.9911504
## [492,] 1.282120 1.180686 1.1853109 0.8002561 1.2800000
## [493,] 1.253006 1.127876 1.1134489 0.8845078 0.9406452
## [494,] 1.271548 1.178378 1.1323231 0.9370840 1.3388430
## [495,] 1.268177 1.217570 1.2318884 2.1124019 0.4224138
## [496,] 1.302050 1.226477 1.0024937 0.9900990       Inf
## [497,] 1.222387 1.183903 1.0331185 1.0465174 2.1191860
## [498,] 1.299127 1.176931 1.0556992 1.4316891       Inf
## [499,] 1.283299 1.180963 1.2181672 2.2290713       Inf
## [500,] 1.272879 1.200847 1.1878739 1.0258940       Inf
```

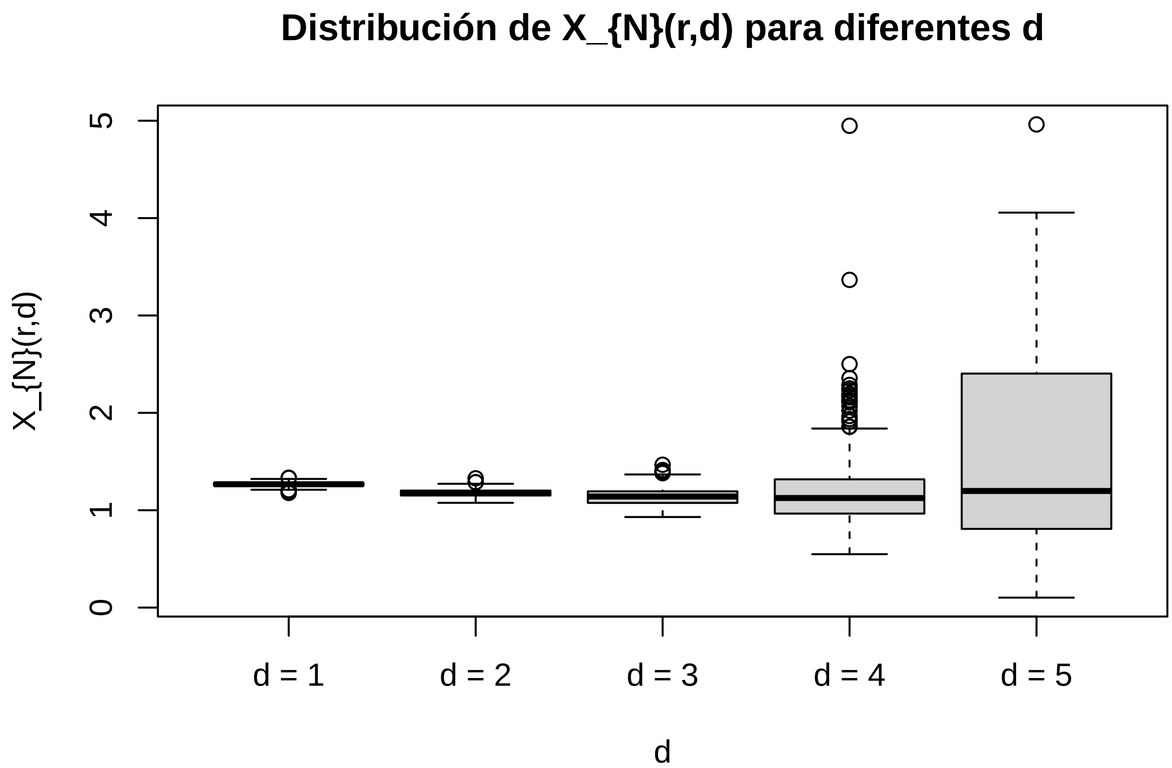
Veamos como se comporta X_{N}(r,d) para este caso

```r
# Resumen de los resultados
summary(X_N_results_ar)
```

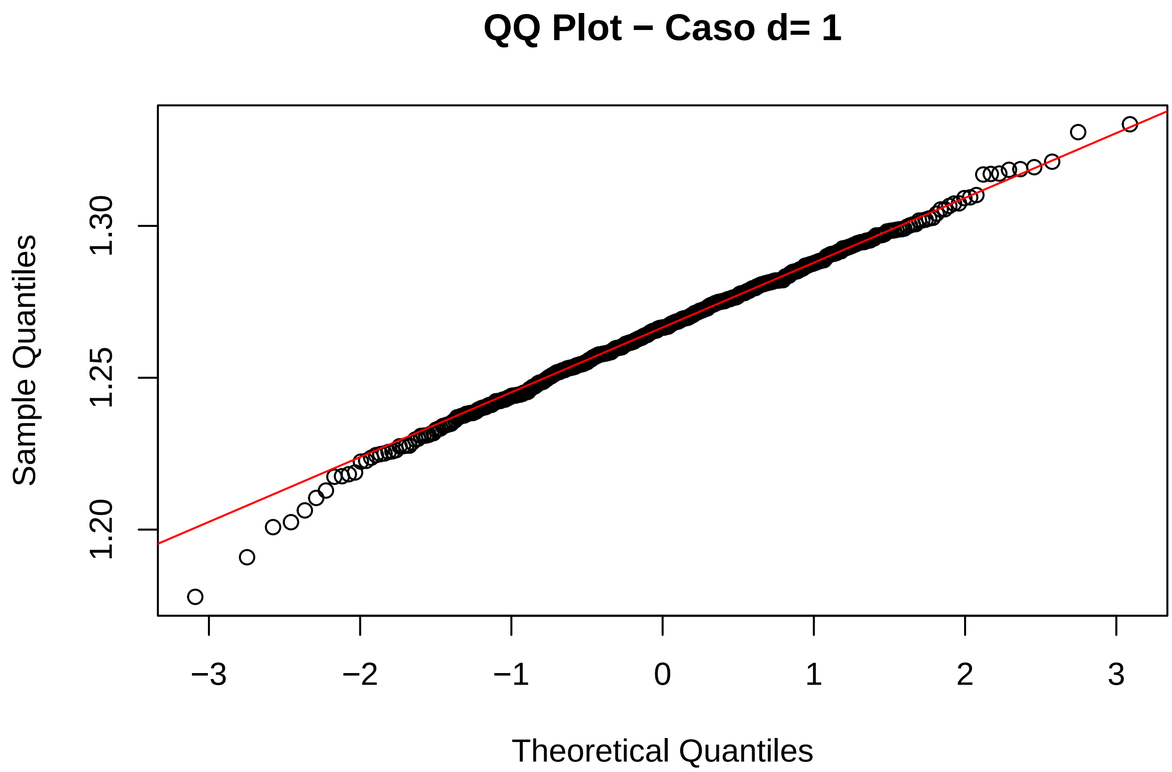
```
##      d = 1           d = 2           d = 3           d = 4       
##  Min.   :1.178   Min.   :1.076   Min.   :0.930   Min.   :0.5485  
##  1st Qu.:1.252   1st Qu.:1.151   1st Qu.:1.075   1st Qu.:0.9658  
##  Median :1.267   Median :1.177   Median :1.141   Median :1.1252  
##  Mean   :1.266   Mean   :1.177   Mean   :1.140   Mean   :1.1908  
##  3rd Qu.:1.281   3rd Qu.:1.202   3rd Qu.:1.194   3rd Qu.:1.3157  
##  Max.   :1.333   Max.   :1.328   Max.   :1.467   Max.   :4.9482  
##      d = 5       
##  Min.   :0.1023  
##  1st Qu.:0.8091  
##  Median :1.1976  
##  Mean   :   Inf  
##  3rd Qu.:2.3854  
##  Max.   :   Inf  
```

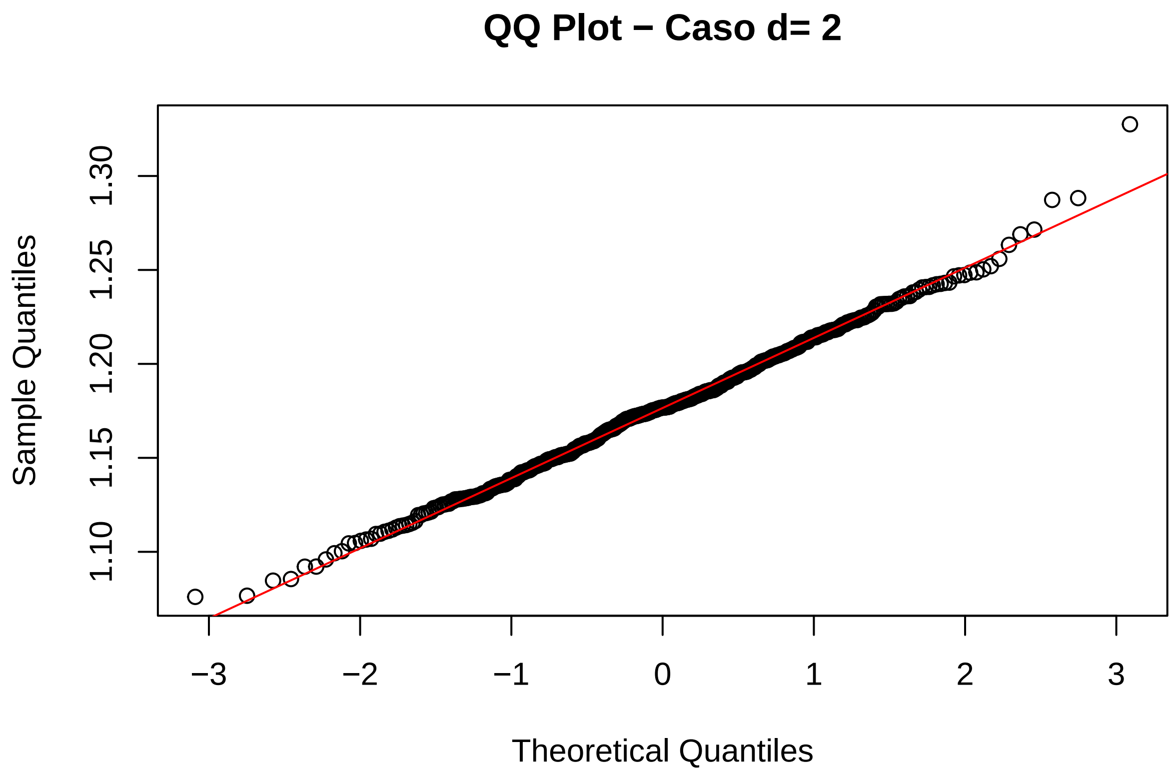
```r
# Graficar los resultados
boxplot(X_N_results_ar, main="Distribución de X_{N}(r,d) para diferentes d", ylab="X_{N}(r,d)", xlab="d
```

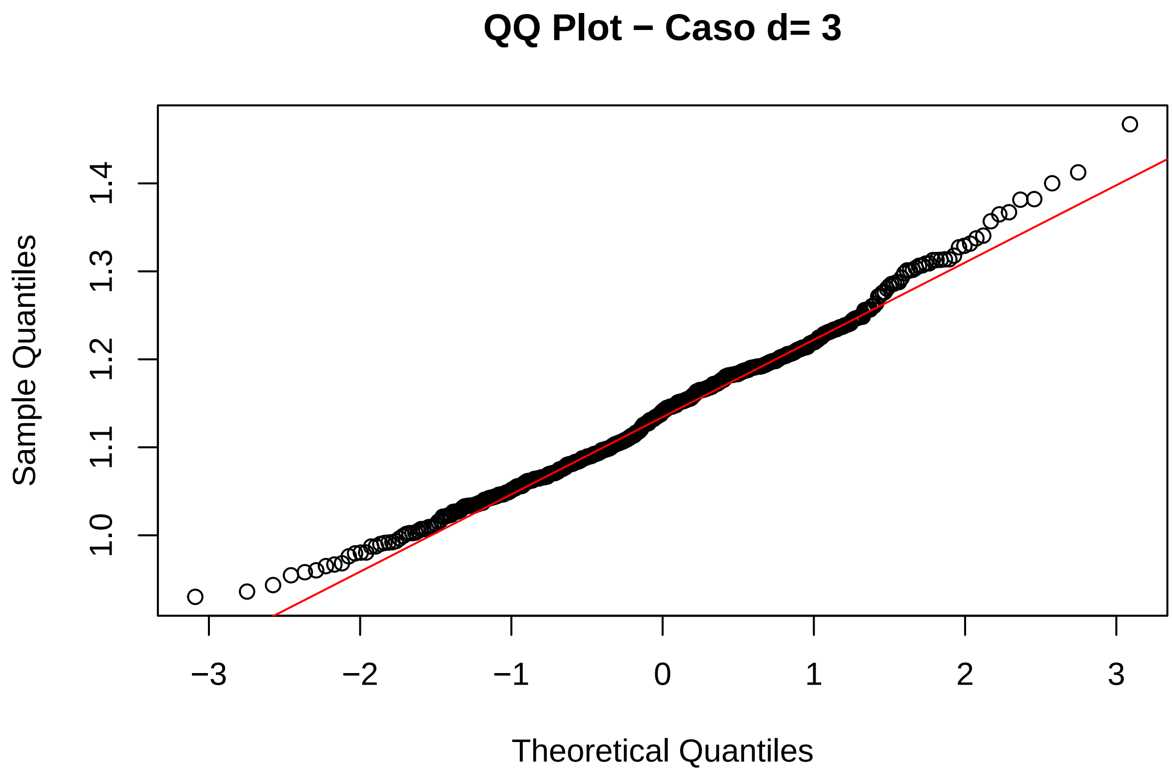
```
## Warning in bplt(at[i], wid = width[i], stats = z$stats[, i], out =
## z$out[z$group == : Outlier (Inf) in boxplot 5 is not drawn
```

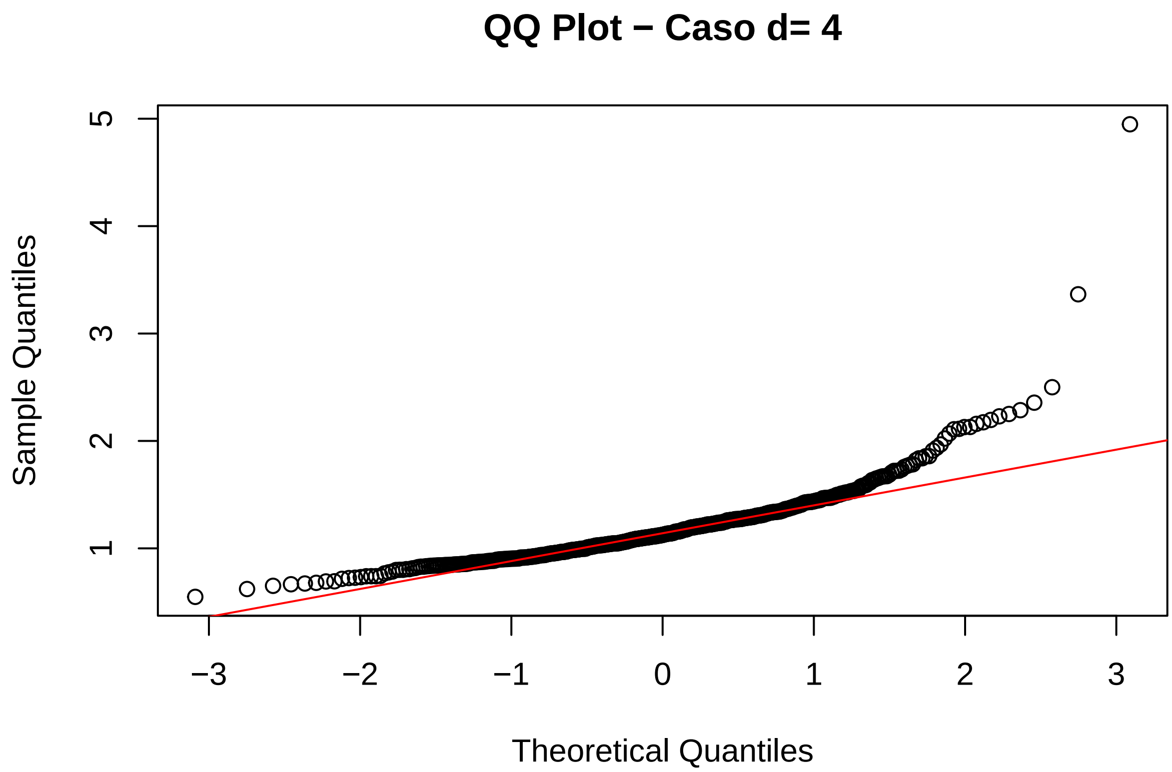



```r
# Crear el QQ plot

for (i in 1:4) {
  qqnorm(X_N_results_ar[, i], main = paste("QQ Plot - Caso d=", i))
  qqline(X_N_results_ar[, i], col = "red")
}
```


QQ Plot − Caso d= 1
Sample Quantiles
1.20
1.25
1.30
−3
−2
−1
0
1
2
3
Theoretical Quantiles


QQ Plot − Caso d= 2
Sample Quantiles
1.10
1.15
1.20
1.25
1.30
−3
−2
−1
0
1
2
3
Theoretical Quantiles


QQ Plot − Caso d= 3
Sample Quantiles
1.0
1.1
1.2
1.3
1.4
−3
−2
−1
0
1
2
3
Theoretical Quantiles




```r
# Longitud de cada serie
N <- length_series  # Esto es igual a 1000 en nuestro caso

# Calcular sqrt(N)
sqrt_N <- sqrt(N)

# Calcular sqrt(N) * ln(X_{N}(r,d)) para cada serie y cada valor de d
sqrt_N_ln_X_N_ar <- sqrt_N * log(X_N_results_ar)

# Ver los resultados
head(sqrt_N_ln_X_N_ar)
```

```
##         d = 1    d = 2     d = 3      d = 4      d = 5
## [1,] 6.774831 3.354642 8.6272714   9.312877  15.663439
## [2,] 7.696417 6.557705 0.7299845   3.391627 -13.516967
## [3,] 6.859764 5.709658 3.2308605 -10.100190  18.587446
## [4,] 6.906696 3.975086 2.9301513  -2.551084   4.315724
## [5,] 7.421520 5.022267 2.9325824  -4.143165  24.064865
## [6,] 8.052480 3.728973 6.1019247   2.730845  -7.759587
```

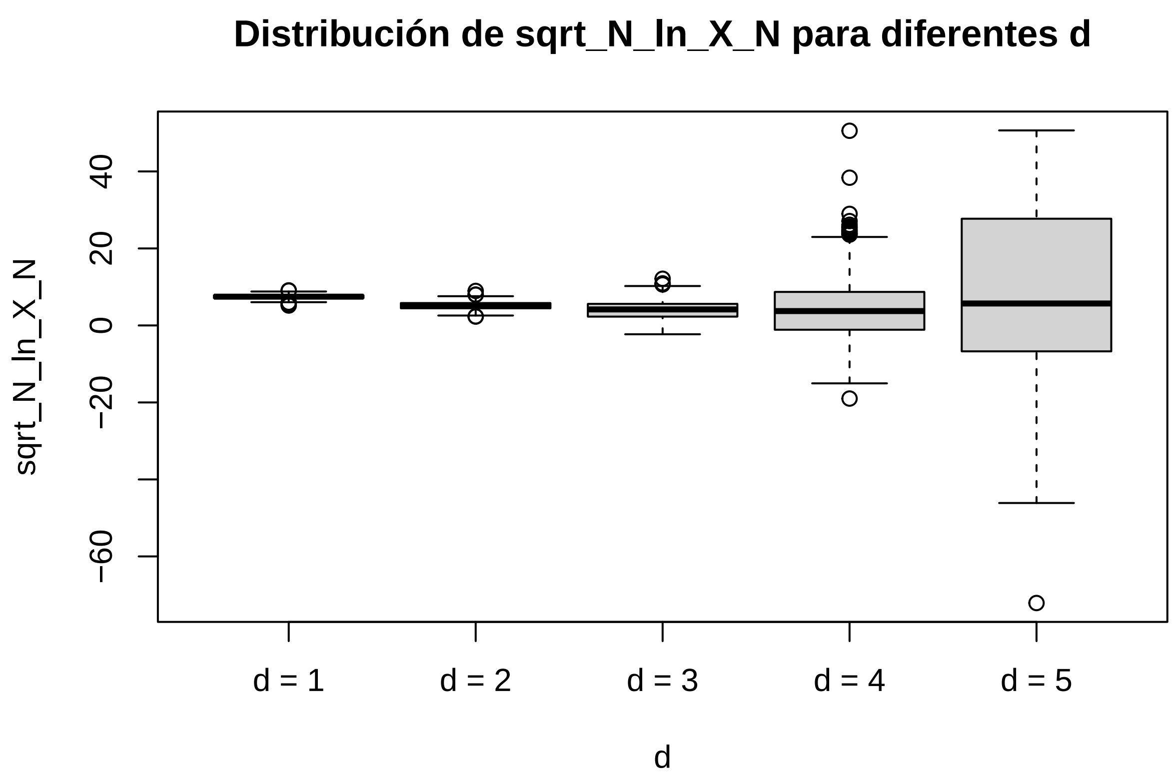
y vemos como se comporta:

```r
# Resumen de los resultados
summary(sqrt_N_ln_X_N_ar)
```

```
##       d = 1           d = 2           d = 3            d = 4
##  Min.   :5.176   Min.   :2.318   Min.   :-2.294   Min.   :-18.991
##  1st Qu.:7.111   1st Qu.:4.457   1st Qu.: 2.289   1st Qu.: -1.101
##  Median :7.472   Median :5.146   Median : 4.158   Median :  3.731
##  Mean   :7.452   Mean   :5.127   Mean   : 4.041   Mean   :  4.420
##  3rd Qu.:7.830   3rd Qu.:5.810   3rd Qu.: 5.596   3rd Qu.:  8.677
##  Max.   :9.100   Max.   :8.959   Max.   :12.123   Max.   : 50.566
##       d = 5
##  Min.   :-72.103
##  1st Qu.: -6.699
##  Median :  5.703
##  Mean   :    Inf
##  3rd Qu.: 27.489
##  Max.   :    Inf
```

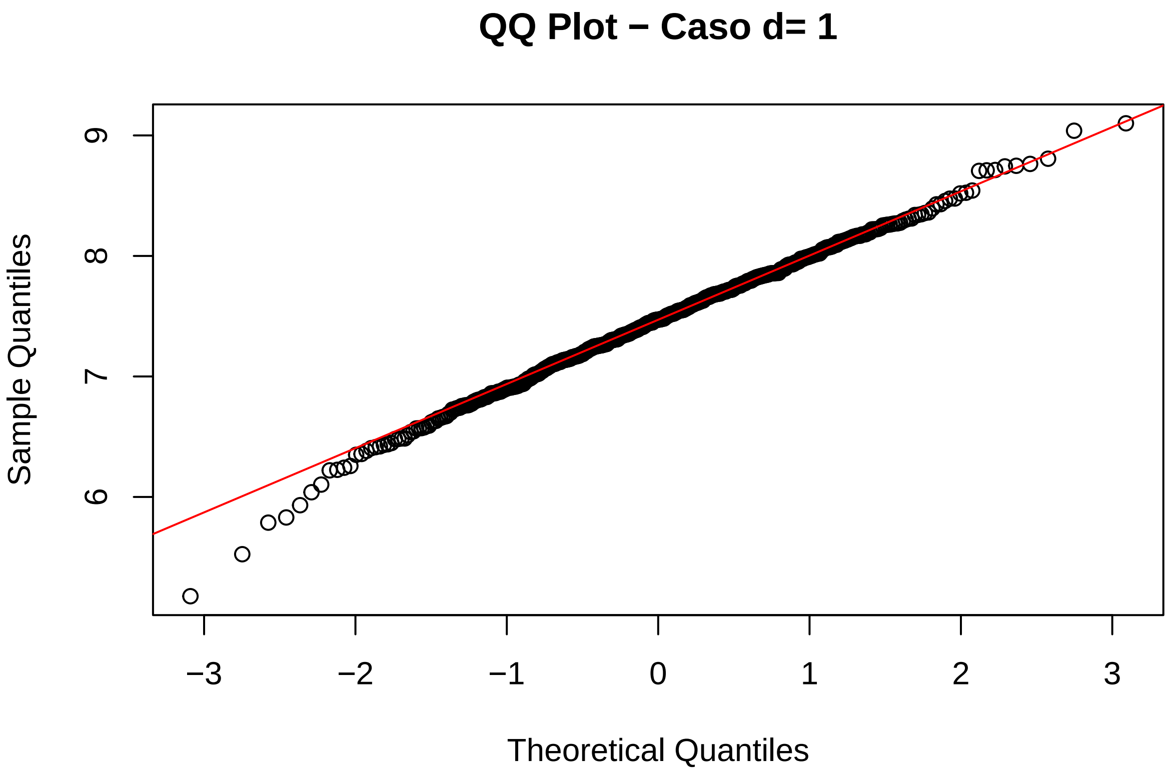
```r
# Graficar los resultados
boxplot(sqrt_N_ln_X_N_ar, main="Distribución de sqrt_N_ln_X_N para diferentes d", ylab="sqrt_N_ln_X_N",
```

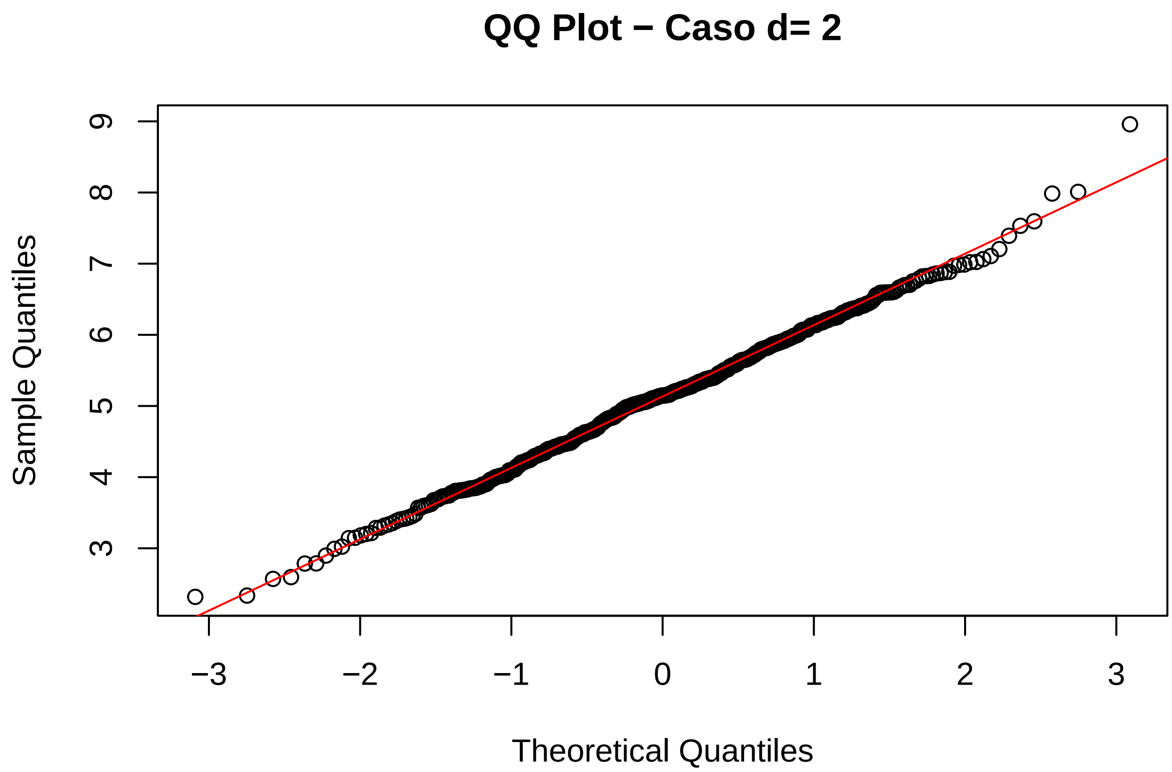
```
## Warning in bplt(at[i], wid = width[i], stats = z$stats[, i], out =
## z$out[z$group == : Outlier (Inf) in boxplot 5 is not drawn
```



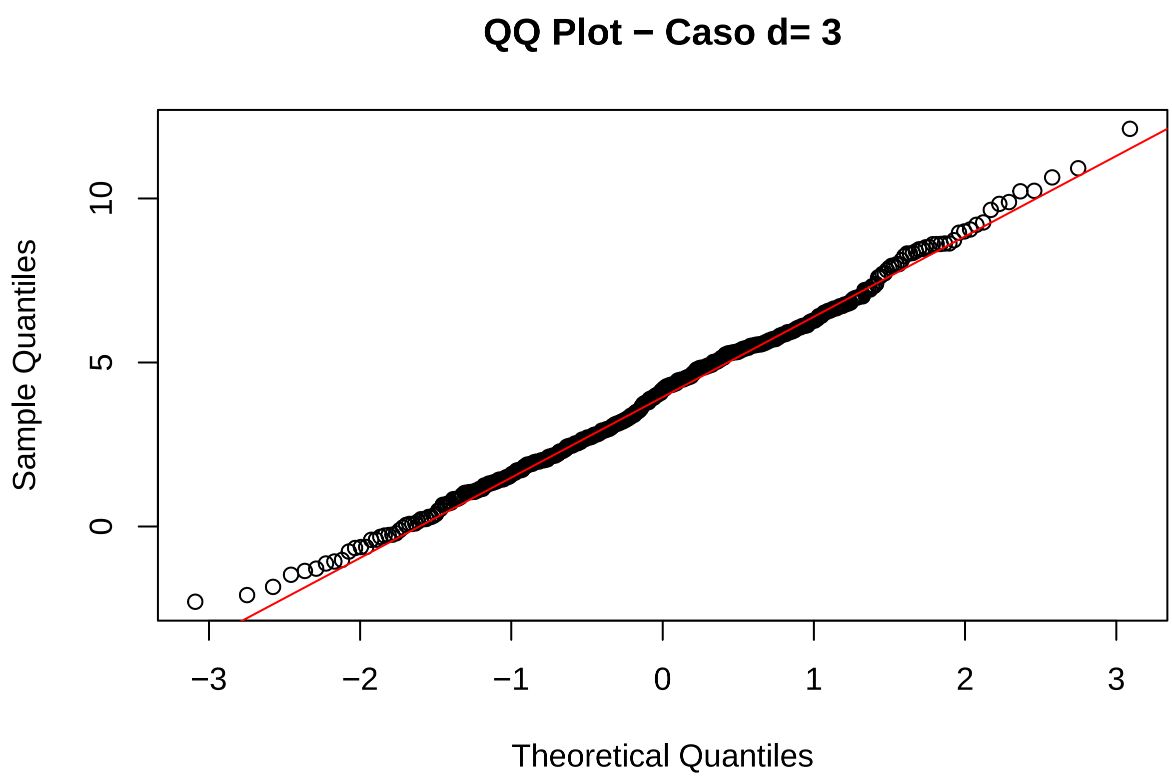

```r
# Crear el QQ plot

for (i in 1:4) {
  qqnorm(sqrt_N_ln_X_N_ar[, i], main = paste("QQ Plot - Caso d=", i))
```

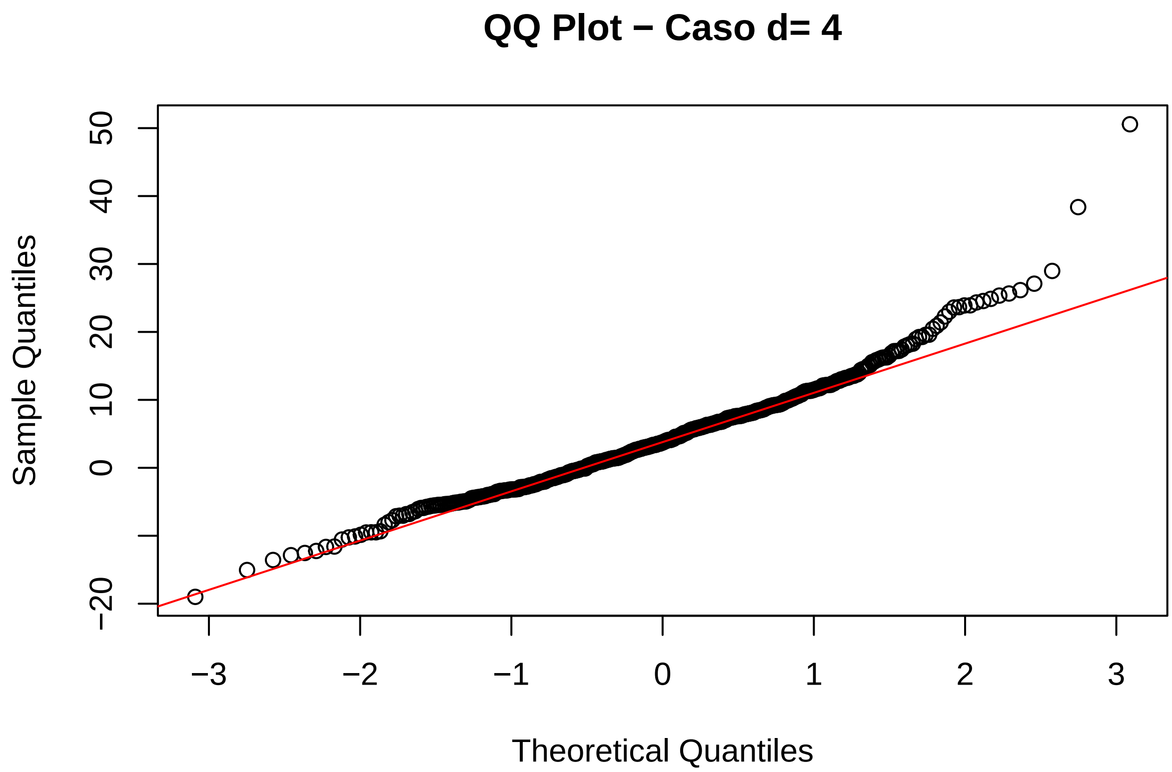
```r
  qqline(sqrt_N_ln_X_N_ar[, i], col = "red")
}
```





QQ Plot − Caso d= 2
Sample Quantiles
3
4
5
6
7
8
9
−3
−2
−1
0
1
2
3
Theoretical Quantiles


QQ Plot − Caso d= 3
Sample Quantiles
0
5
10
−3
−2
−1
0
1
2
3
Theoretical Quantiles



Se mantiene la tendencia en los casos d=1 a d=4 con problemas de datos indefinidos en el caso d=5