В [1]:

```
library("ggplot2")
```

```
Warning message:
"package 'ggplot2' was built under R version 3.6.3"
```

1. Распределение Пуассона

$$a(x) = \frac{1}{x!},\ C(\theta) = e^{\theta},\ \mathbb{P}(\xi = x) = \frac{a(x)\theta^x}{C(\theta)} \Rightarrow$$

$$G(\theta) = log(C(\theta)),\ \mathbb{E}[X_t] = \mu_x = \frac{\theta G'(\theta)}{1-\alpha} = \frac{(}{1-}$$

$$\mathbb{D}[X_t] = \mu_x + \frac{\theta^2 G''(\theta)}{1-\alpha^2} = \frac{\theta}{1-\alpha}$$

В [2]:

```
rho <- function(tms){
    n <- length(tms)
#     n <- 1000
    rh <- rep(0, times=n)
    p2 <- 0
    mn <- mean(tms)
    for (i in 1 : n){
        p2 <- p2 + (tms[i] - mn)^2
    }
    for (k in 0 : n - 1){
        l <- n - k
        for (t in 1 : l){
            rh[k + 1] = rh[k + 1] + (tms[t] - mn) * (tms[t + k] - mn) / p2
        }
    }

    return(rh)
}
```

В [3]:

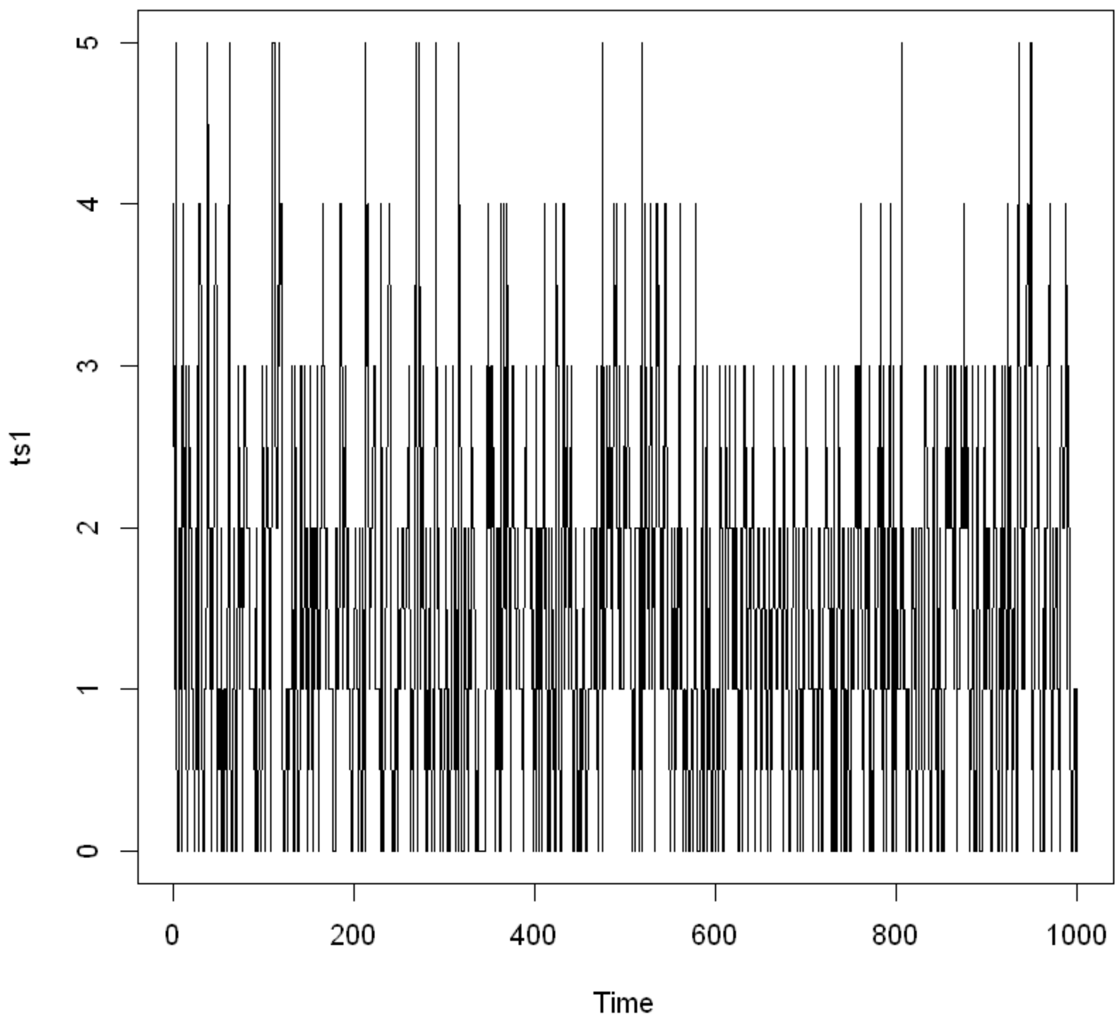
```
lse_alpha <- function(tms){
    p1 <- 0
    p2 <- 0
    p3 <- 0
    p4 <- 0
    n <- length(tms)
    for (t in 2 : n){
        p1 <- p1 + tms[t] * tms[t - 1]
        p2 <- p2 + tms[t]
        p3 <- p3 + tms[t - 1]
        p4 <- p4 + tms[t - 1] ^ tms[t - 1]
    }
    alpha <- (p1 - p2 * p3 / (n - 1)) / (p4 - p3 * p3 / (n - 1))
    return(alpha)
}
```

В [4]:

```
poinar <- function(n, alpha, lambda){
    len <- n + 1
    x <- rep(1, times = len)
    x[1] <- ceiling(lambda / (1 - alpha))
    xi <- rpois(len, lambda)
    for (i in 2 : len){
        x[i] <- sum(rbinom(n=x[i - 1], 1, prob=alpha)) + xi[i]
#         x[i] <- rbinom(1, x[i - 1], alpha) + xi[i]
    }
    return(stats::ts(x[2:len]))
}
```

В [20]:

```
alpha = 0.3
lambda = 1
ts1 <- poinar(n = 1000, alpha = alpha, lambda = lambda)
plot(ts1)
```



В [21]:

```
mean(ts1)
```

1.487

В [22]:

```
var(ts1)
```

1.4672982982983

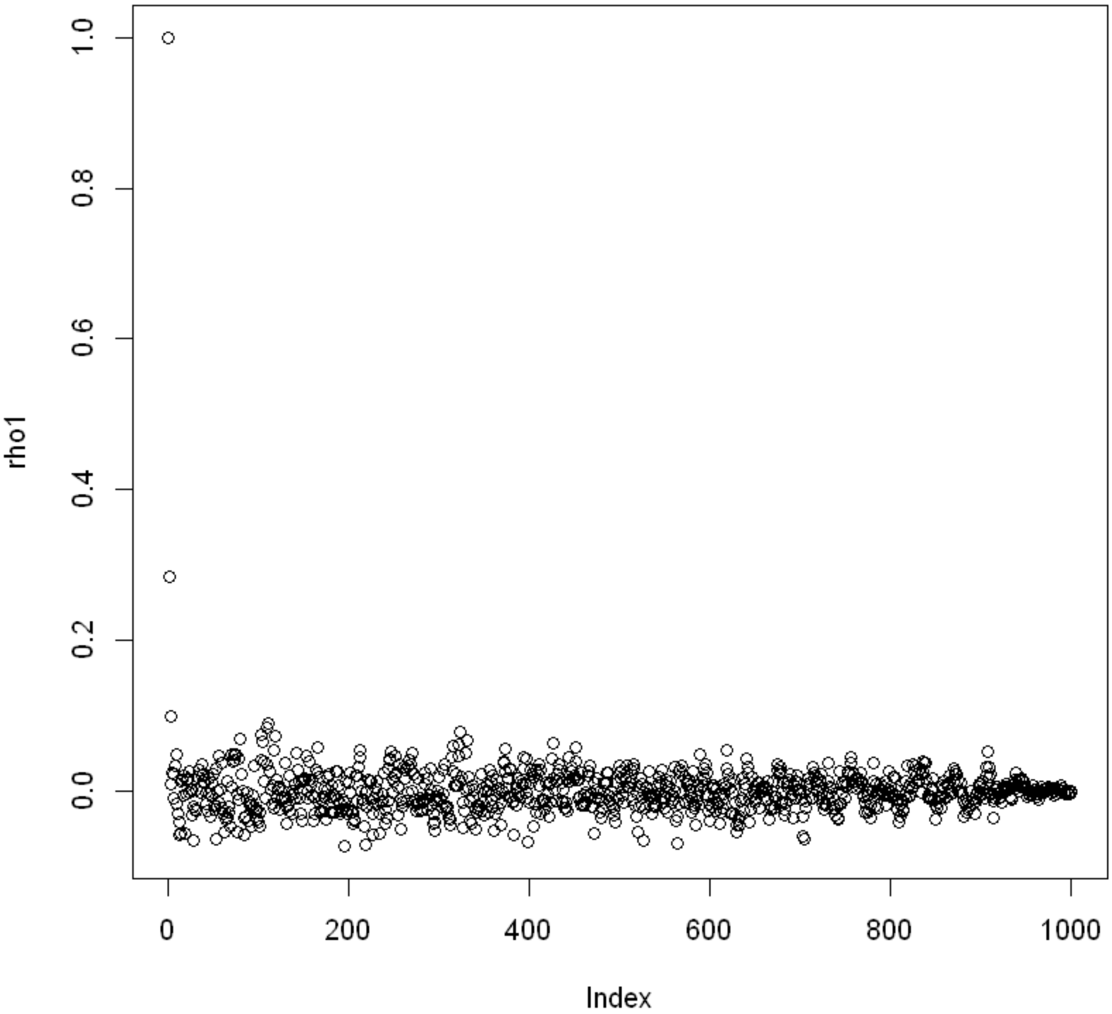
$\mathbb{E}[X_t] = \mathbb{D}[X_t] = \frac{\theta}{1-\alpha} = \frac{\theta}{1-\alpha} = \frac{1}{0.7} = 1.42857142857143$

В [23]:

```
rho1 <- rho(ts1)
```

В [36]:

```
plot(rho1)
```



$\hat{\alpha} = \hat{\rho}(1)$:

В [25]:

```
rho1[2]
```

0.284517677003693

$\hat{\lambda} = \hat{\theta} = (1 - \hat{\alpha})\overline{X}$:

В [26]:

```
(1 - rho1[2] )* mean(ts1)
```

1.06392221429551

В [27]:

```
lse_alpha(ts1)
```

0.00589265359540279

2. Геометрическое распределение

$$a(x) = 1,\ C(\theta) = \frac{1}{1-\theta},\ \mathbb{P}(\xi = x) = \frac{a(x)\theta^x}{C(\theta)} \Rightarrow$$

$$G(\theta) = log(C(\theta)),\ \mathbb{E}[X_t] = \mu_x = \frac{\theta G'(\theta)}{1-\alpha} = \frac{}{(1}$$

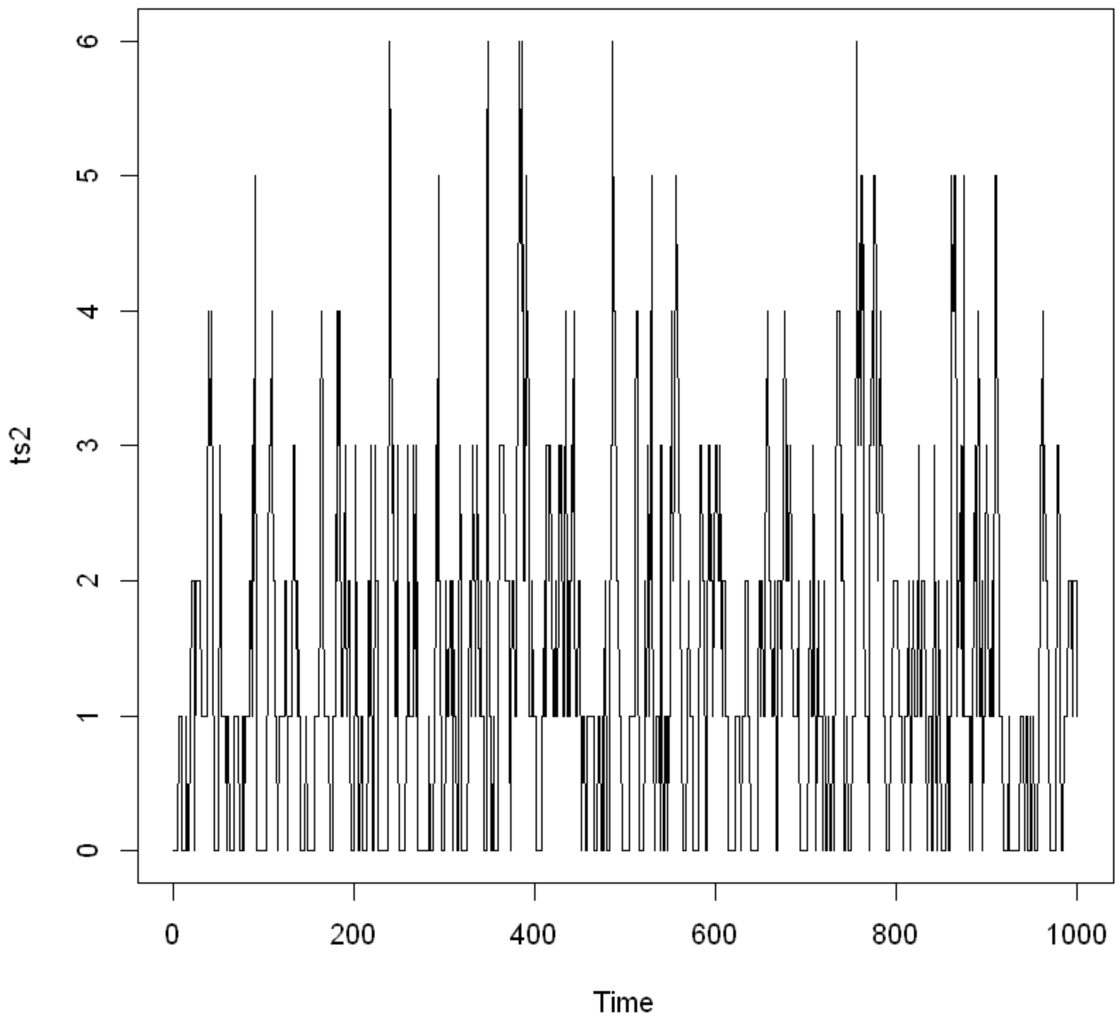
$$\mathbb{D}[X_t] = \mu_x + \frac{\theta^2 G''(\theta)}{1-\alpha^2} = \frac{\theta+\alpha\theta(1-\theta)}{(1-\alpha^2)(1-\theta)^2}$$

В [28]:

```
ginar <- function(n, alpha, p){
    len <- n + 1
    x <- rep(1, times = len)
    xi <- rgeom(len, p)
    for (i in 2 : len){
        x[i] <- sum(rbinom(n=x[i - 1], 1, prob=alpha)) + xi[i]
    }
    return(stats::ts(x[2: len]))
}
```

В [29]:

```
alpha <- 0.7
p <- 0.7
ts2 <- ginar(n = 1000, alpha = 0.7, p = 0.7)
plot(ts2)
```



В [30]:

```
mean(ts2)
```

1.316

$\mathbb{E}[X]$:

В [31]:

```
t <- 1 - p
t / (1 - alpha) / (1 - t)
```

1.42857142857143

В [32]:

```
var(ts2)
```

1.6578018018018

$\mathbb{D}[X]$:

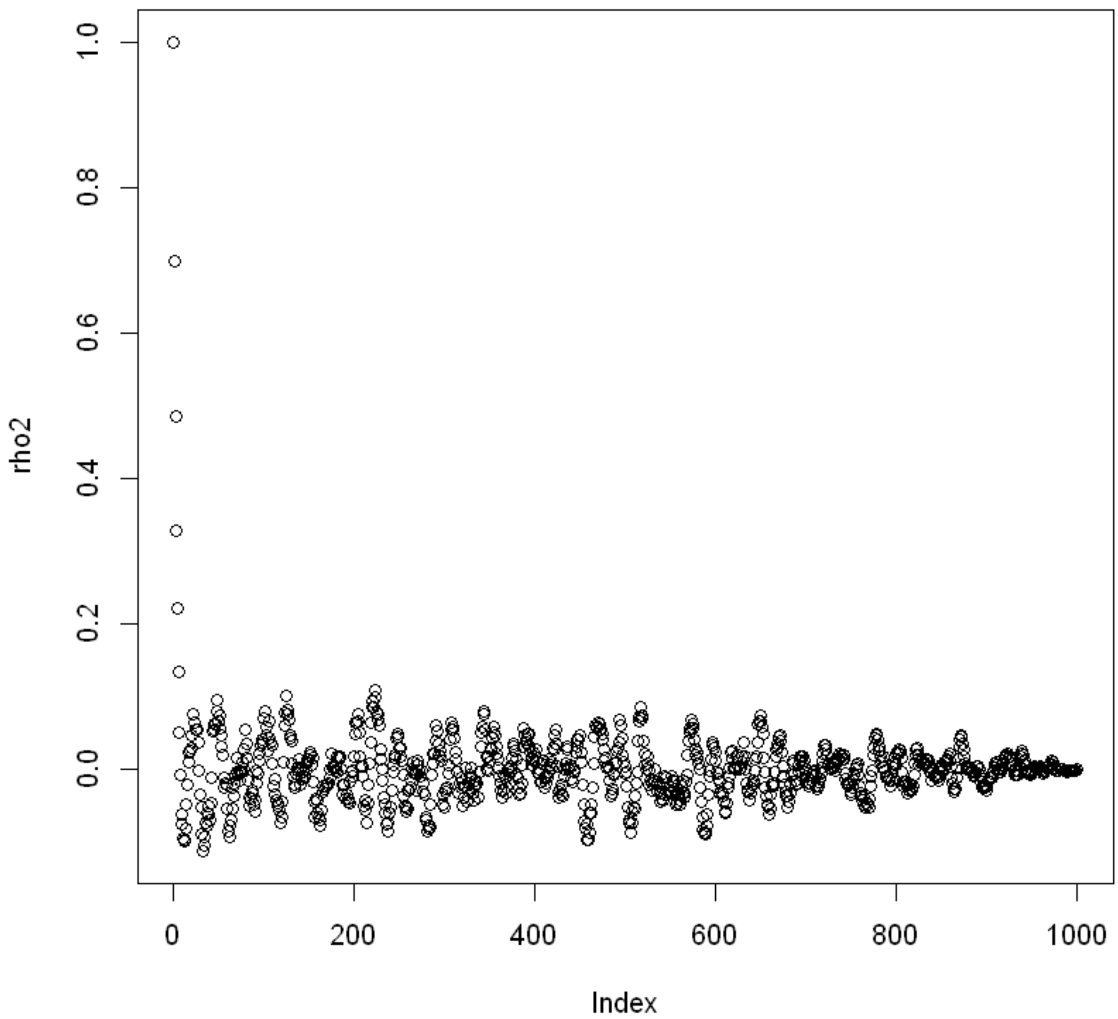
В [33]:

```
(t + alpha * t * p) / (1 - alpha ^ 2) / p^2
```

1.78871548619448

В [35]:

```
rho2 <- rho(ts2)
plot(rho2)
```



$\hat{\alpha} = \hat{\rho}(1)$:

В [37]:

```
rho2[2]
```

0.699050411075362

$$\hat{\theta} = \frac{\overline{X}(1-\hat{\alpha})}{1+\overline{X}(1-\hat{\alpha})} = 1 - \hat{p} :$$

В [38]:

```
mean(ts2) * (1 - rho2[2]) / (1 + mean(ts2) * (1 - rho2[2]))
```

0.283693102508599

В [39]:

```
library('actuar')
```

```
Warning message:
"package 'actuar' was built under R version 3.6.3"
Attaching package: 'actuar'

The following object is masked from 'package:grDevices':

    cm
```

3. Логарифмическое распределение

$$a(x) = \frac{1}{x},\ C(\theta) = -log(1-\theta),$$

$$\mathbb{P}(\xi = x) = \frac{a(x)\theta^x}{C(\theta)} \Rightarrow \theta = p$$

$$G(\theta) = log(C(\theta)),$$

$$\mathbb{E}[X_t] = \mu_x = \frac{\theta G'(\theta)}{1-\alpha} = -\frac{\theta}{(1-\alpha)(1-\theta)log(1-\theta)}$$

$$\mathbb{D}[X_t] = \mu_x + \frac{\theta^2 G''(\theta)}{1-\alpha^2} = \mathbb{E}[X_t] + \frac{\theta^2}{1-\alpha^2}\Bigg(-\frac{\quad}{(1-}$$

B [41]:

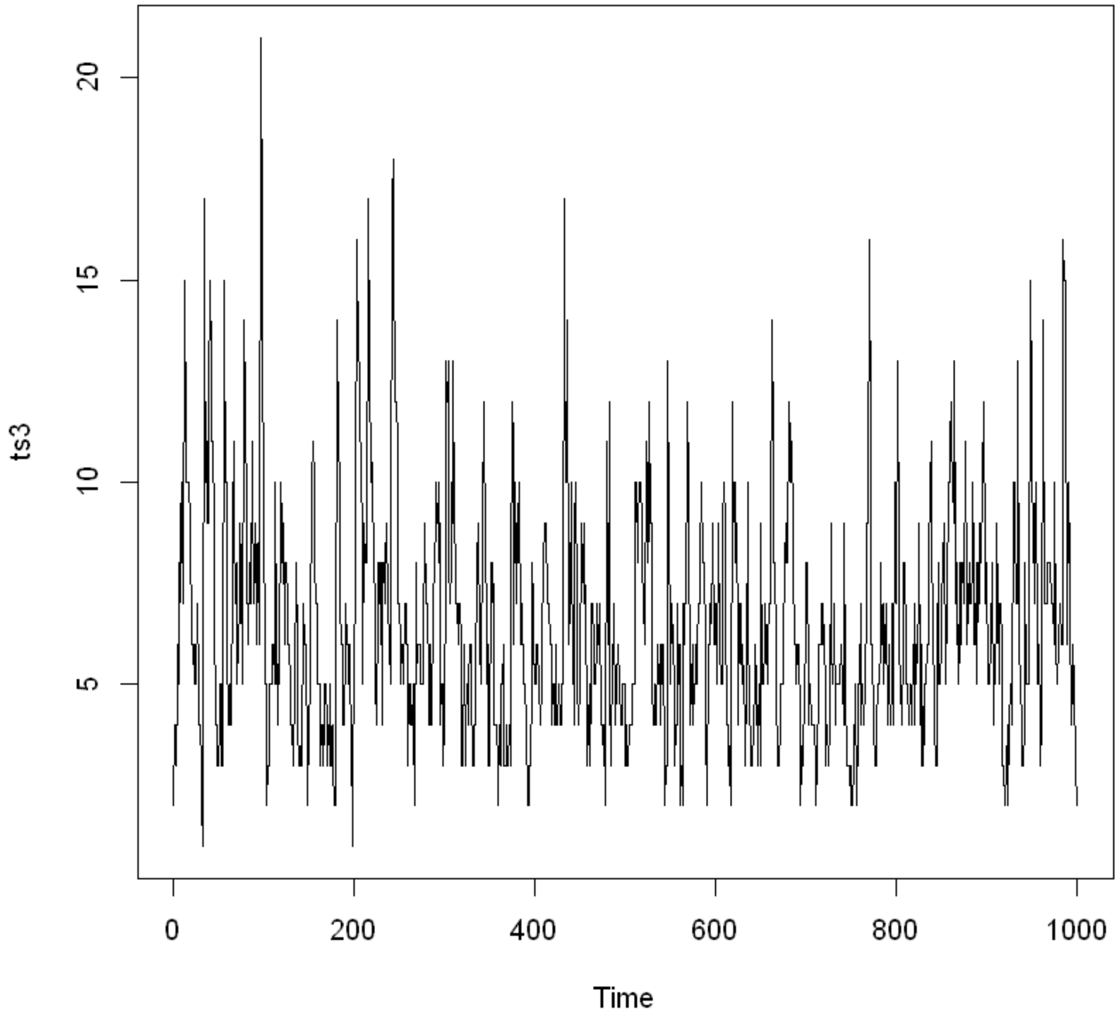
```
loginar <- function(n, alpha, p){
    len <- n + 1
    x <- rep(1, times = len)
    xi <- rlogarithmic(len, p)
    for (i in 2 : len){
        x[i] <- sum(rbinom(n=x[i - 1], 1, prob=alpha)) + xi[i]
    }
    return(stats::ts(x[2: len]))
}
```

B [42]:

```
ts3 <- loginar(n = 1000, alpha = 0.7, p = 0.7)
```

В [43]:

```
plot(ts3)
```



В [44]:

```
mean(ts3)
```

6.502

В [45]:

```
alpha = 0.7
p = 0.7
```

В [46]:

```
E <- -p / (1 - alpha) / log(1-p) / (1-p)
E
```

6.46009423953085

В [47]:

```
var(ts3)
```

8.47647247247247

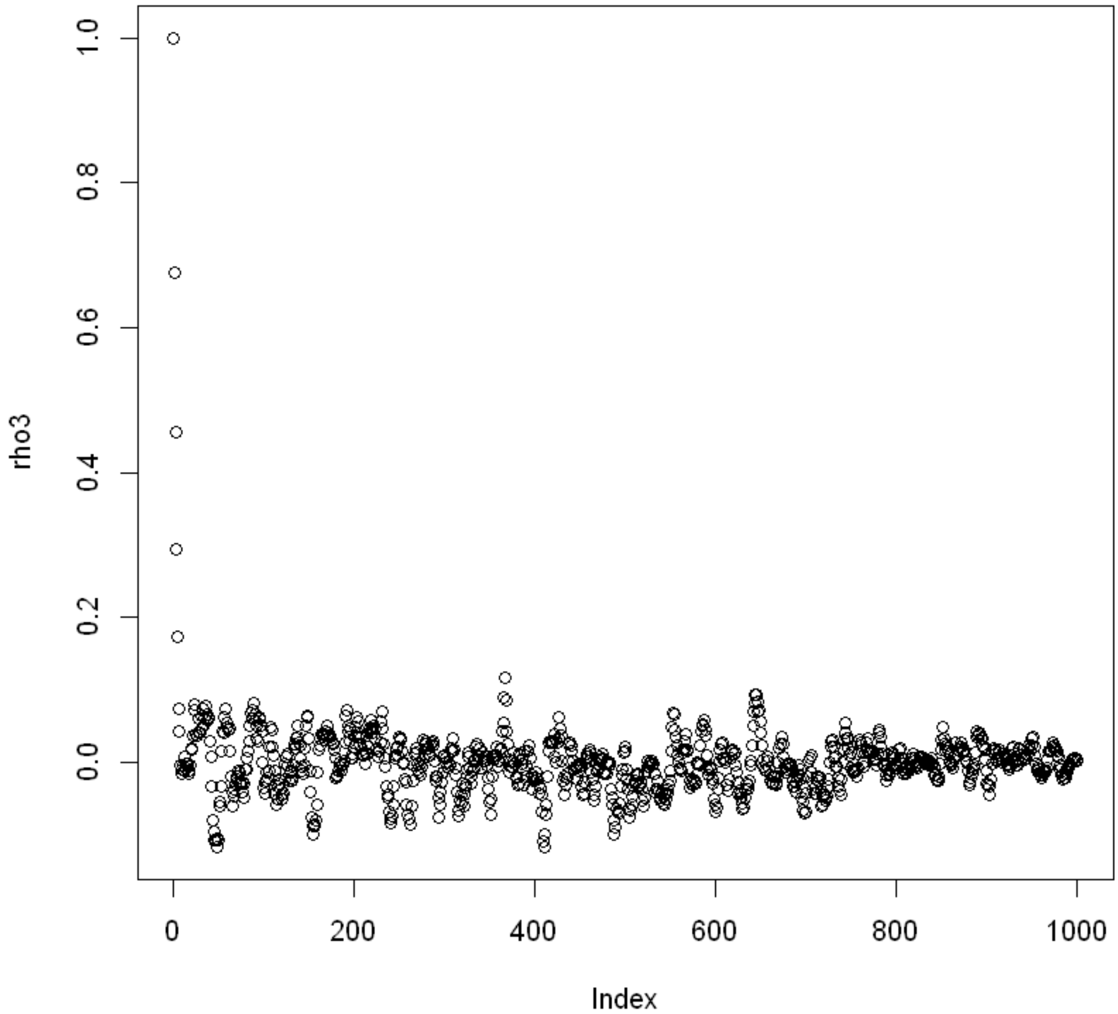
В [48]:

```
V <- E + p^2 / (1 - alpha^ 2)  * (- (1 + log(1 - p)) / ((1 - p) ^ 2 * (log(1 / (1 - p)))^2)
V
```

7.9622753868756

В [49]:

```
rho3 <- rho(ts3)
plot(rho3)
```



В [50]:

```
rho3[2]
```

0.676042831857734

$$\overline{X} = -\frac{\hat{\theta}}{(1-\hat{\alpha})(1-\hat{\theta})log(1-\hat{\theta})} \Rightarrow -\overline{X}(1-\hat{\alpha}) = \frac{}{(1-\hat{\theta}}$$

В [51]:

```
-mean(ts3) * (1 - rho3[2])
```

-2.10636950726101

$\hat{\theta} = 0.738686$

4. Распределение Бернулли

$$a(x) = 1,\ C(\theta) = 1+\theta,$$
$$\mathbb{P}(\xi = x) = \frac{a(x)\theta^x}{C(\theta)} \Rightarrow \theta = \frac{p}{1-p}$$
$$G(\theta) = log(C(\theta)),$$
$$\mathbb{E}[X_t] = \mu_x = \frac{\theta G'(\theta)}{1-\alpha} = \frac{\theta}{(1-\alpha)(1+\theta)}$$
$$\mathbb{D}[X_t] = \mu_x + \frac{\theta^2 G''(\theta)}{1-\alpha^2} = \mathbb{E}[X_t] - \frac{\theta^2}{(1-\alpha^2)(1-\theta)^2}$$

В [49]:

```
install.packages('Rlab')
```

```
package 'Rlab' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
	C:\Users\Ilya\AppData\Local\Temp\RtmpENG9Vy\downloaded_packages
```

В [52]:

```
library("Rlab")
```

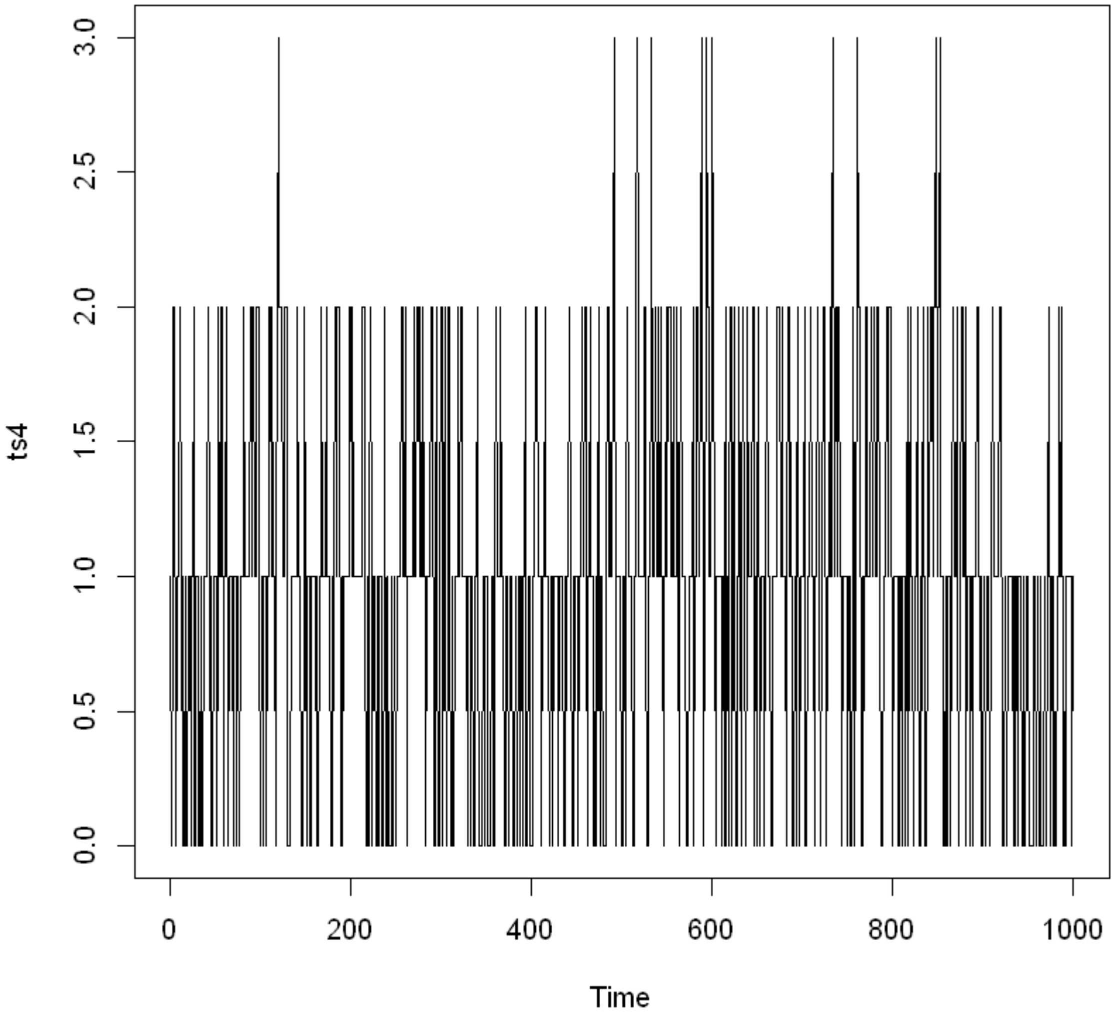
```
Rlab 2.15.1 attached.


Attaching package: 'Rlab'

The following objects are masked from 'package:stats':

    dexp, dgamma, dweibull, pexp, pgamma, pweibull, qexp, qgamma,
    qweibull, rexp, rgamma, rweibull

The following object is masked from 'package:datasets':

    precip
```

В [53]:

```
berinar <- function(n, alpha, p){
    len <- n + 1
    x <- rep(1, times = len)
    xi <- rbern(len, p)
    for (i in 2 : len){
        x[i] <- sum(rbinom(n=x[i - 1], 1, prob=alpha)) + xi[i]
    }
    return(stats::ts(x[2:len]))
}
```

В [61]:

```
alpha = 0.3
p = 0.7
ts4 <- berinar(n = 1000, alpha = alpha, p = p)
plot(ts4)
```



В [62]:

```
mean(ts4)
```

0.98

В [63]:

```
t <- p / (1 - p)
E <- t / (1 - alpha) / (1 + t)
E
```

1

В [64]:

```
var(ts4)
```

0.47007007007007

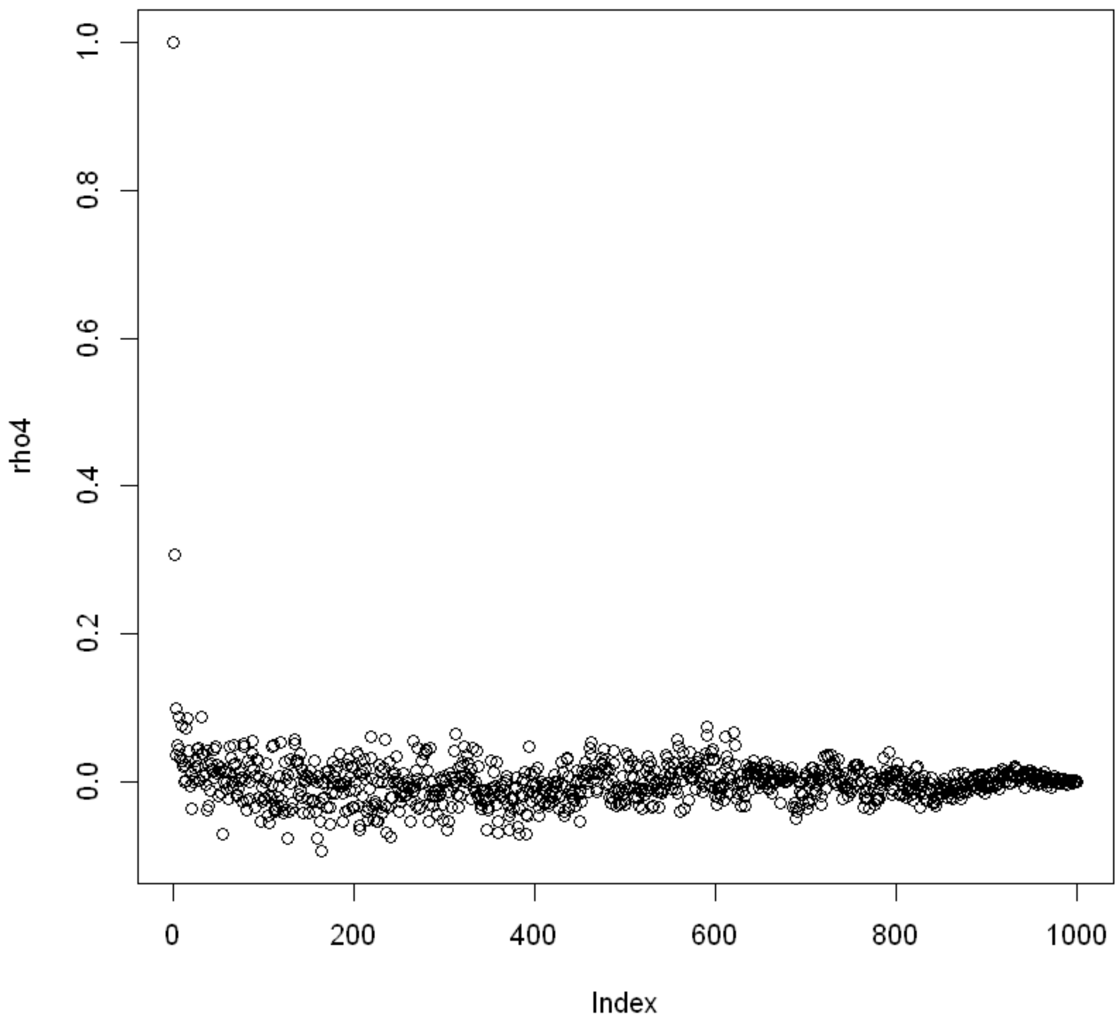
В [65]:

```
V <- E - t^2 / (1 - alpha^2) / (1 + t)^2
V
```

0.461538461538462

В [66]:

```
rho4 <- rho(ts4)
plot(rho4)
```



$\hat{\alpha}$:

В [67]:

```
rho4[2]
```

0.307920783645654

$$\hat{\theta} = \frac{\overline{X}(1-\hat{\alpha})}{1-\overline{X}(1-\hat{\alpha})} = \frac{\hat{p}}{1-\hat{p}}:$$

В [70]:

```
t <- mean(ts4) * (1 - rho4[2]) / (1 - mean(ts4) * (1 - rho4[2]))
t
```

2.10788364189537

$$\hat{p} = \frac{\hat{\theta}}{1+\hat{\theta}}$$

В [71]:

```
t / (1 + t)
```

0.678237632027259

5. Биномиальное распределение

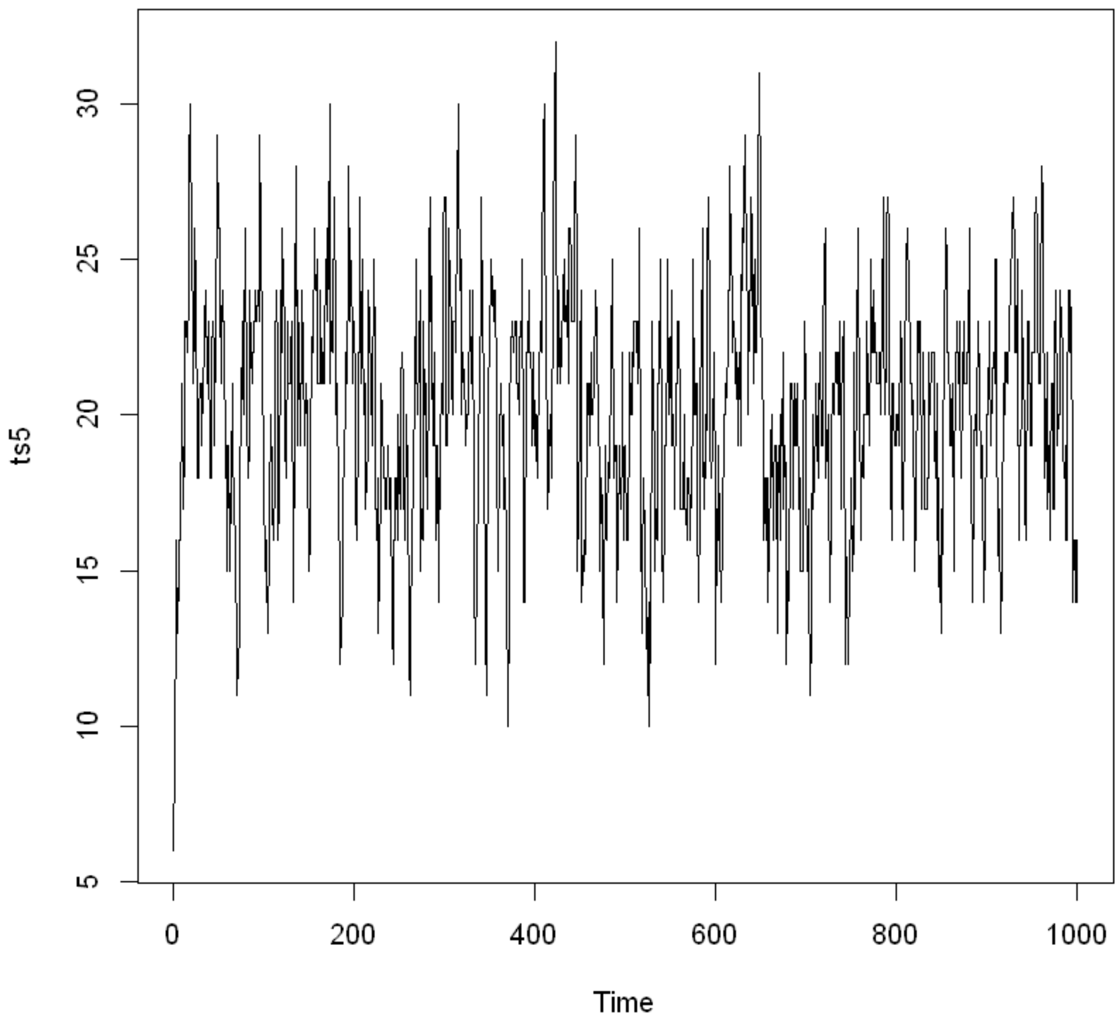
$$a(x) = C_n^k,\ C(\theta) = (1+\theta)^n,$$
$$\mathbb{P}(\xi = x) = \frac{a(x)\theta^x}{C(\theta)} \Rightarrow \theta = \frac{p}{1-p}$$
$$G(\theta) = log(C(\theta)),$$
$$\mathbb{E}[X_t] = \mu_x = \frac{\theta G'(\theta)}{1-\alpha} = \frac{n\theta}{(1-\alpha)(1+\theta)}$$
$$\mathbb{D}[X_t] = \mu_x + \frac{\theta^2 G''(\theta)}{1-\alpha^2} = \mathbb{E}[X_t] - \frac{n\theta^2}{(1-\alpha^2)(1+\theta)^2}$$

В [102]:

```
bininar <- function(n, alpha, p){
    len <- n + 1
    x <- rep(1, times = len)
    xi <- rbinom(len, 10, p)
    for (i in 2 : len){
        x[i] <- sum(rbinom(n=x[i - 1], 1, prob=alpha)) + xi[i]
    }
    return(stats::ts(x[2:len]))
}
```

В [103]:

```
alpha <- 0.7
p <- 0.6
ts5 <- bininar(n=1000, alpha = alpha, p = p)
plot(ts5)
```



В [104]:

```
mean(ts5)
```

20.184

В [105]:

```
t <- p / (1 - p)
E <- t / (1 - alpha) / (1 + t) * 10
E
```

20

В [106]:

```
var(ts5)
```

13.3174614614615

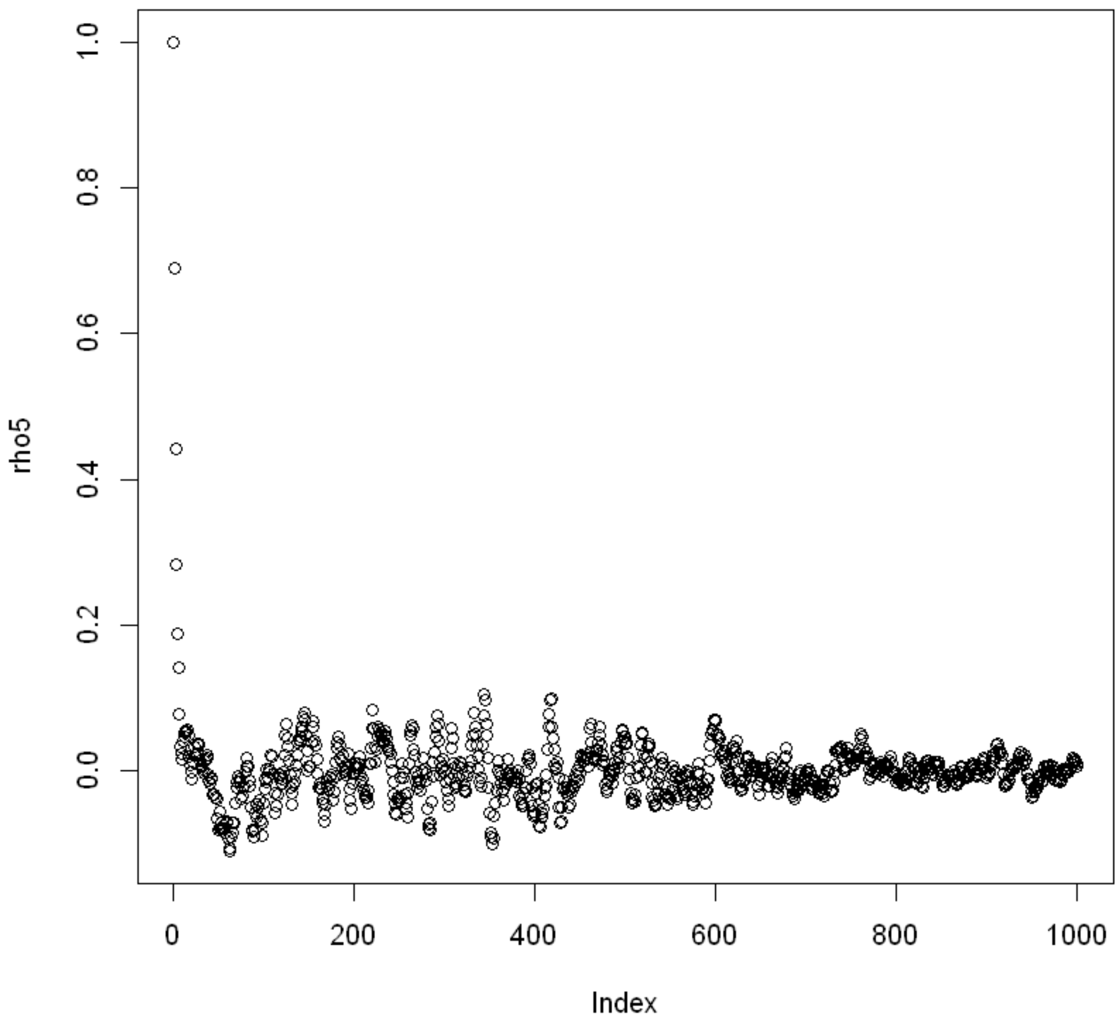
В [107]:

```
V <- E - 10 * t^2 / (1 - alpha^2) / (1 + t)^2
V
```

12.9411764705882

В [108]:

```
rho5 <- rho(ts5)
plot(rho5)
```



$\hat{\alpha}$:

В [109]:

```
rho5[2]
```

0.690493890024042

$$\hat{\theta} = \frac{\overline{X}(1-\hat{\alpha})}{n-\overline{X}(1-\hat{\alpha})} = \frac{\hat{p}}{1-\hat{p}}$$

В [110]:

```
t <- mean(ts5) * (1 - rho5[2]) / (10 - mean(ts5) * (1 - rho5[2]))
t
```

1.6645856776593

$$\hat{p} = \frac{\hat{\theta}}{1+\hat{\theta}}$$

В [111]:

```
t / (t + 1)
```

0.624707132375474

6. Отрицательное биномиальное распределение

$$a(x) = \frac{\Gamma(r+x)}{x!\Gamma(r)}, \ C(\theta) = (1-\theta)^{-r},$$

$$\mathbb{P}(\xi = x) = \frac{a(x)\theta^x}{C(\theta)} \Rightarrow \theta = 1 - p$$

$$G(\theta) = log(C(\theta)),$$

$$\mathbb{E}[X_t] = \mu_x = \frac{\theta G'(\theta)}{1-\alpha} = \frac{r\theta}{(1-\alpha)(1-\theta)}$$

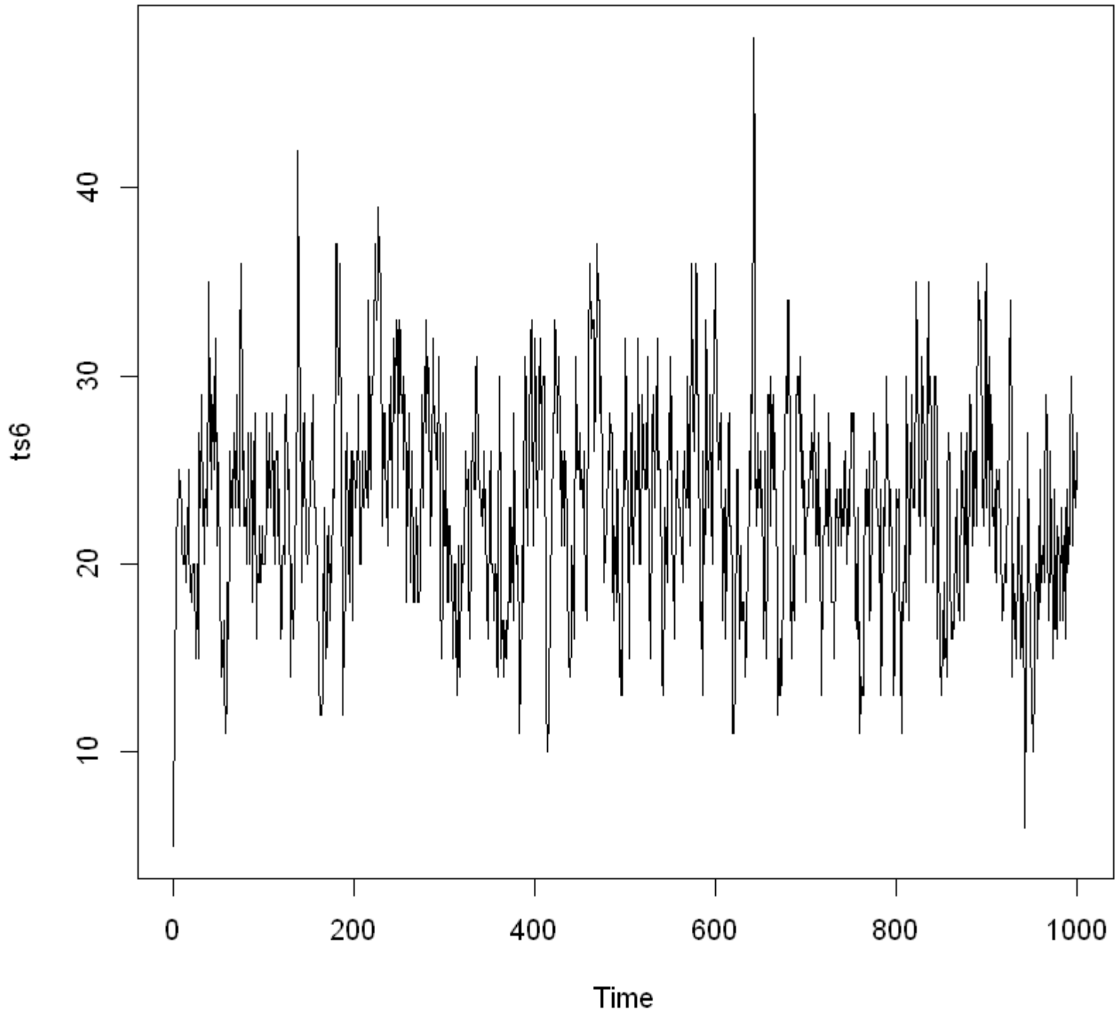
$$\mathbb{D}[X_t] = \mu_x + \frac{\theta^2 G''(\theta)}{1-\alpha^2} = \mathbb{E}[X_t] + \frac{r\theta^2}{(1-\alpha^2)(1-\theta)^2}$$

В [112]:

```
nbininar <- function(n, alpha, p){
    len <- n + 1
    x <- rep(1, times = len)
    xi <- rnbinom(len, 10, p)
    for (i in 2 : len){
        x[i] <- sum(rbinom(n=x[i - 1], 1, prob=alpha)) + xi[i]
    }
    return(stats::ts(x[2:len]))
}
```

В [113]:

```
alpha <- 0.7
p <- 0.6
ts6 <- nbininar(n=1000, alpha=alpha, p = p)
plot(ts6)
```



В [114]:

```
mean(ts6)
```

23.046

В [115]:

```
t <- 1 - p
# E <- -10 * t / (1 - alpha) / (1 + t)
E <- t / (1 - alpha) / (1 - t) * 10
E
```

22.2222222222222

В [116]:

```
var(ts6)
```

30.3622462462462

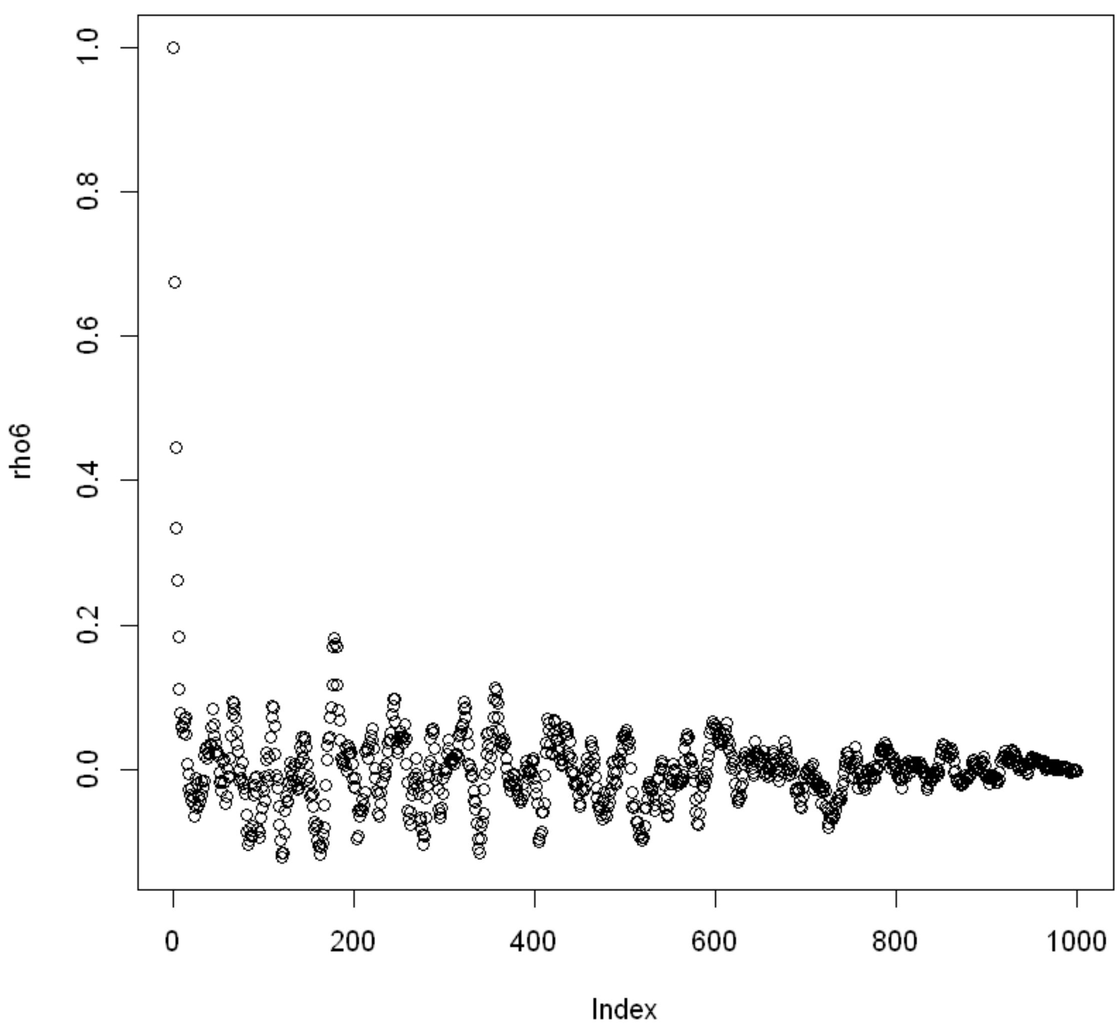
В [117]:

```
V <- E + 10 * t^2 / (1 - alpha^2) / (1 - t)^2
V
```

30.9368191721133

В [118]:

```
rho6 <- rho(ts6)
plot(rho6)
```



$\hat{\alpha}$:

В [119]:

```
rho6[2]
```

0.67395213182274

$$\hat{\theta} = \frac{\overline{X}(1-\hat{\alpha})}{r+\overline{X}(1-\hat{\alpha})} = 1 - \hat{p}:$$

В [120]:

```
mean(ts6) * (1 - rho6[2]) / (10 + mean(ts6) * (1 - rho6[2]))
```

0.429031439017911