Wiener process generating

Vitaliy Pozdnyakov

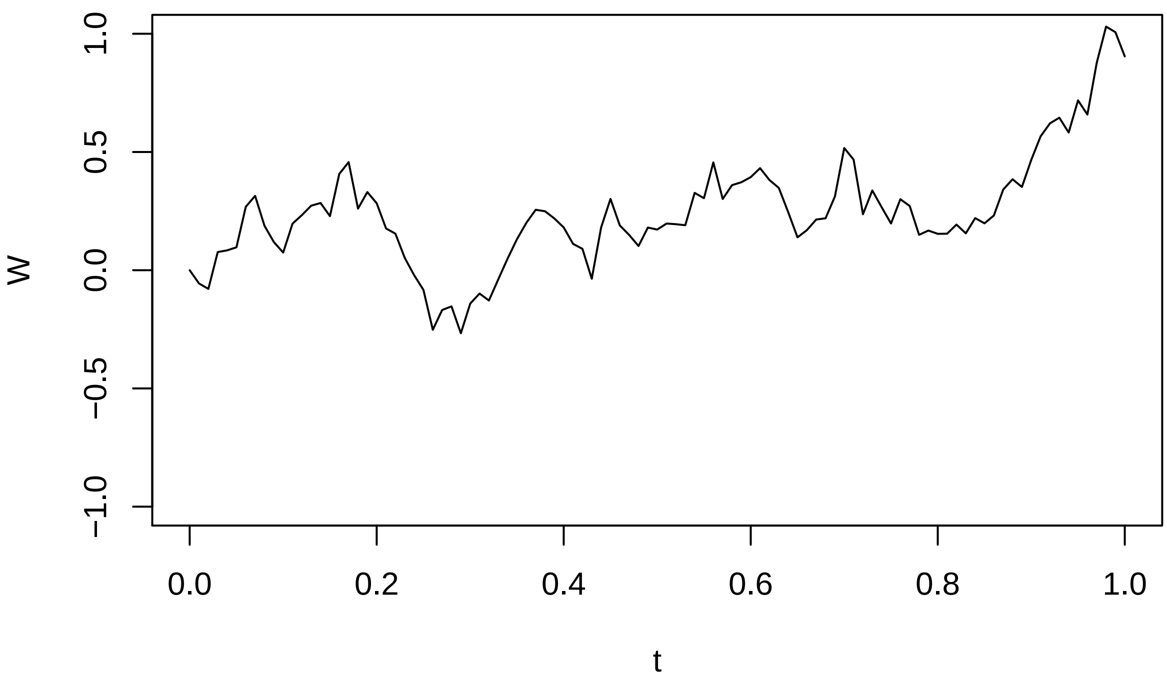
January 2020

By definition

$$W(t + \Delta t) = W(t) + \sqrt{n}z$$

where z is a realization from $Z \sim N(0, 1)$

```
set.seed(123)
N <- 100
T <- 1
Delta <- T/N
W <- numeric(N+1)
t <- seq(0,T, length=N+1)
for(i in 2:(N+1))
  W[i] <- W[i-1] + rnorm(1) * sqrt(Delta)
plot(t,W, type="l", ylim=c(-1,1))
```



As the limit of a random walk

$$W(t) = \frac{S_{[nt]}}{\sqrt{n}}$$

where

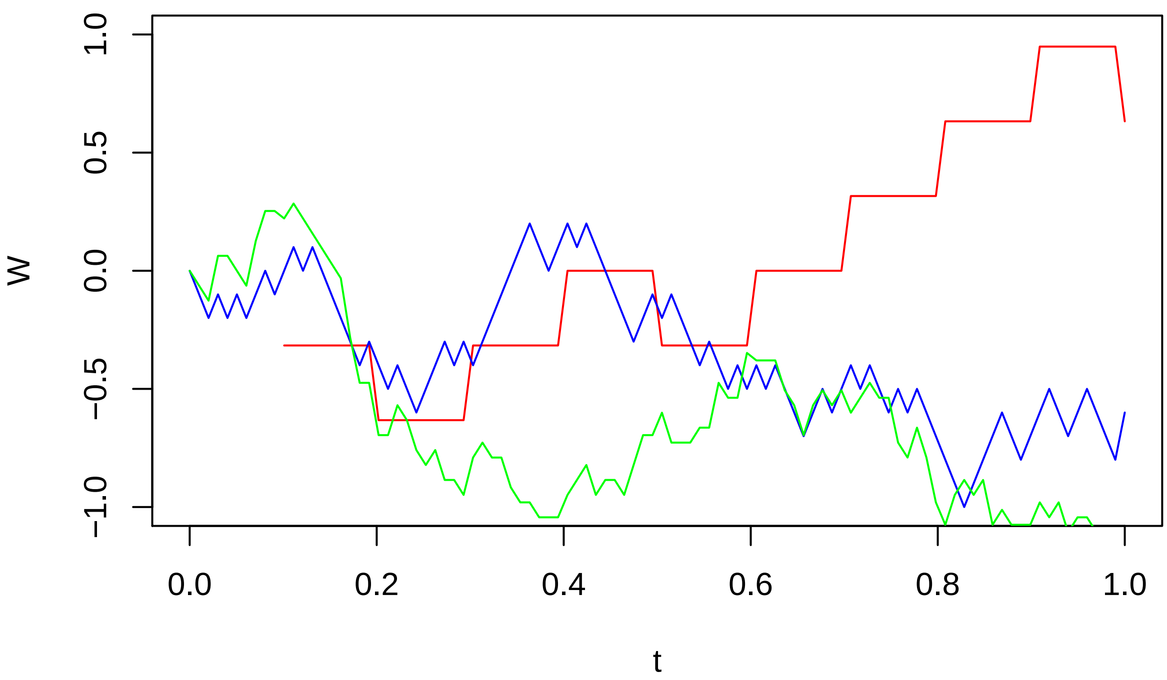
$$S_n = X_1 + \cdots + X_n$$

```
set.seed(1)
n <- 10
```

```r
T <- 1
t <- seq(0, 1, length=100)
S <- cumsum(2*(runif(n)>0.5)-1)
W <- numeric(100)
for(i in (2:100))
  W[i] <- ifelse(t[i]*n > 0, S[t[i]*n] / sqrt(n), 0)
palette(rainbow(6))
plot(t, W, type = "l", ylim = c(-1, 1), col=palette()[1])

n <- 100
S <- cumsum(2*(runif(n)>0.5)-1)
W <- numeric(100)
for(i in (2:100))
  W[i] <- ifelse(t[i]*n > 0, S[t[i]*n] / sqrt(n), 0)
lines(t, W, col=palette()[5])

n <- 1000
S <- cumsum(2*(runif(n)>0.5)-1)
W <- numeric(100)
for(i in (2:100))
  W[i] <- ifelse(t[i]*n > 0, S[t[i]*n] / sqrt(n), 0)
lines(t, W, col=palette()[3])
```



As L^2 expansion

$$W(t) = \sum_{i=0}^{+\infty} Z_i(\omega)\phi_i(t), 0 \leq t \leq T$$

where

$$\phi_i(t) = \frac{2\sqrt{2T}}{(2i+1)\pi} \sin \frac{(2i+1)\pi t}{2T}$$

```
set.seed(2)
phi <- function(i, t, T){
  (2*sqrt(2*T))/((2*i+1)*pi) * sin(((2*i+1)*pi*t)/(2*T))
}
T <- 1
N <- 100
t <- seq(0, T, length = N+1)
W <- numeric(N+1)
n <- 10
Z <- rnorm(n)
for(i in (2:N+1))
  W[i] <- sum(Z*sapply(1:n, function(x) phi(x, t[i], T)))
plot(t, W, type = "l", col = 'black', ylim=c(-1,1))

n <- 50
Z <- rnorm(n)
for(i in (2:N+1))
  W[i] <- sum(Z*sapply(1:n, function(x) phi(x, t[i], T)))
lines(t, W, col = 'red')

n <- 100
Z <- rnorm(n)
for(i in (2:N+1))
  W[i] <- sum(Z*sapply(1:n, function(x) phi(x, t[i], T)))
lines(t, W, col = 'blue')
```