Zürcher Hochschule
für Angewandte Wissenschaften

School of Engineering
IDP Institut für Datenanalyse und Prozessdesign

FS 2016

Stochastische Prozesse

Woche 2

Aufgabe 1 Diskreter stochastischer Prozess

Betrachten Sie folgenden stochastischen Prozess:

$$X_t = \text{sign}(X_{t-1} + X_{t-2} + 6(\epsilon_t - 0.5)),$$

wobei die ϵ_t unabhängig identisch standardnormalverteilte Zufallsvariable seien. `R`-Tipp: Eine standardnormalverteilte Zufallszahl bekommt man mit `rnorm()`.

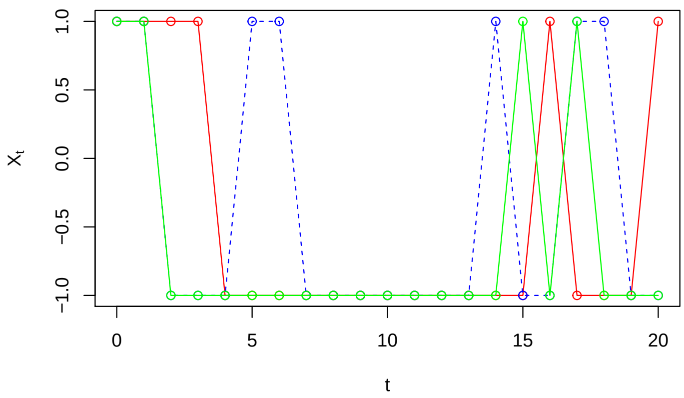
a) Was hat Prozess für einen Zustandsraum?
b) Ist dieser Prozess ein Markov-Prozess?
c) Erzeugen Sie mittels Simulation Trajektorien dieses Prozesses. Plotten Sie drei verschiedene Trajektorien, die alle im Zustand $x_1 = 1$ und $x_0 = 1$ starten.

Tipp: In `R` können Sie mehrere Kurven wie folgt plotten:

```
x <- seq(1, 2 * 3.131, 0.1)
plot(x, sin(x), t = "o")
lines(x, cos(x), col = "red", t = "o")
```

```
# Input: x0: Startwert f<U+00FC>r t=0, x1: Startwert f<U+00FC>r t=1, T:
# Letzter Zeitpunkt f<U+00FC>r Simulation.

# Output: Vektor der L<U+00E4>nge T+1: (x0,x1,x2,...,xT)

sim_proc <- function(x0 = 1, x1 = 1, T = 10) {
    ret <- c(x0, x1, rep(NA, (T - 1)))
    for (t in 3:(T + 1)) {
        ret[t] <- sign(ret[t - 2] + ret[t - 1] + 6 * (rnorm(1) - 0.5))
    }
    return(ret)
}

# Simulation f<U+00FC>r t=0,1,...,20

T <- 20
ret <- sim_proc(1, 1, T)
```

```r
plot(0:T, ret, type = "o", col = "red", xlab = "t", ylab = expression(X[t]))
lines(0:T, sim_proc(1, 1, T), type = "o", col = "blue", lty = 2)
lines(0:T, sim_proc(1, 1, T), type = "o", col = "green")
```



d) Schätzen Sie mittels Simulation die Wahrscheinlichkeit, zum Zeitpunkt $t = 4$ im Zustand $X_{t=4} = 1$ zu sein, wenn der Anfangszustand ($t = 0$ und $t = 1$) durch $x_0 = 1$ und $x_1 = 1$ gegeben ist.

```r
states <- replicate(10000, sim_proc(1, 1, 4)[5])  # Achtung: Wert f<U+00FC>r t=4 steht an f<U+00FC>n
sum(states == 1)/length(states)
```

```
[1] 0.2947
```

Aufgabe 2 Simulation einer Markov-Kette

a) Schreiben Sie eine R-Funktion `markov(P, T, x0)`, die eine Realisation einer Markov-Kette mit Laufzeiten $t = 0, \ldots, T$ für eine beliebige Übergangsmatrix P und Startwert x_0 simuliert. Die Rückgabe soll ein Vektor der Länge T sein, der eine Trajektorie eines Markov-Prozesses (ohne den Startzustand) beschreibt.

```r
markov <- function(P, T = 10, x0 = 1) {
    stopifnot(is.matrix(P))
    stopifnot(dim(P)[1] == dim(P)[2])
    stopifnot(all((P) >= 0))
    states <- dim(P)[1]
    ret <- c(x0, rep(0, T))  # erste Komponente x0, dann x1,...,xT
    for (i in 2:(T + 1)) {
        ret[i] <- sample(states, 1, prob = P[ret[i - 1], ])
    }
    ret[-1]  # Startzustand nicht mit ausgeben
}


P = matrix(c(0, 0.1, 0.9, 0.2, 0.5, 0.3, 0, 0.5, 0.5), ncol = 3, byrow = T)
markov(P)
```

```
 [1] 3 3 3 3 2 3 2 3 2 2
```

b) Schätzen Sie mittels Simulation die Wahrscheinlichkeit, dass der Prozess zum Zeitpunkt $t = 3$ im Zustand 3 ist, der Startwert sei $X_0 = 1$. Verwenden Sie dazu folgende Übergangsmatrix:

```r
(P = matrix(c(0, 0.1, 0.9, 0.2, 0.5, 0.3, 0, 0.5, 0.5), ncol = 3, byrow = T))
```

```
     [,1] [,2] [,3]
[1,]  0.0  0.1  0.9
[2,]  0.2  0.5  0.3
[3,]  0.0  0.5  0.5
```

```r
# Wahrscheinlichkeit, sich zum Zeitpunkt T=3 im Zustand 3 zu befinden
P = matrix(c(0, 0.1, 0.9, 0.2, 0.5, 0.3, 0, 0.5, 0.5), ncol = 3, byrow = T)
states <- replicate(10000, markov(P, T = 3, x0 = 1)[3])
sum(states == 3)/length(states)
```

```
[1] 0.4078
```

```r
# Theoretisches Resultat ist 0.4080
(P %*% P %*% P)[1, 3]
```

```
[1] 0.408
```

c) Der Prozess laufe bis zum Zeitpunkt $t = 5$. Wie hoch ist für diese Laufzeit die erwartete Anzahl von Aufenthalten im Zustand 1?

```r
P = matrix(c(0, 0.1, 0.9, 0.2, 0.5, 0.3, 0, 0.5, 0.5), ncol = 3, byrow = T)
anz_besuche <- replicate(10000, sum(markov(P, T = 5, x0 = 1) == 1))
mean(anz_besuche)  # Z<U+00E4>hlt nur Besuche f<U+00FC>r t=1,2,3,4,5 ohne t=0
```

```
[1] 0.3162
```

```r
mean(anz_besuche) + 1   # Start x0=1
```

```
[1] 1.3162
```

```r
# Theoretisches Resultat ist 1.3104 (mit dem Schritt t0) denn Es gilt laut
# Skript M[1,1] = Anzahl erwarteter Besuche in 1 bei Start in 1

M <- diag(3)
Pt <- P
for (t in 1:5) {
    M <- M + Pt
    Pt <- Pt %*% P
}
M
```

```
        [,1]    [,2]    [,3]
[1,] 1.31040 2.01264 2.67696
[2,] 0.56928 3.30960 2.12112
[3,] 0.38400 2.38320 3.23280
```