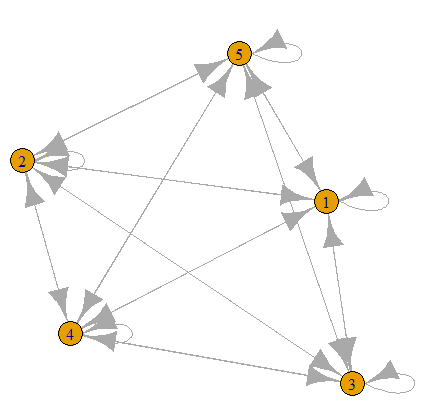
Моделирование марковских цепей

Цель

Изучение дискретных вероятностных моделей на примере марковских цепей.

Ход работы





Код для построения графика:

install.packages("igraph")

library(igraph)

# Задаем матрицу.

P <- matrix(c(0.18, 0.18, 0.36, 0.23, 0.05,

0.43, 0.13, 0.06, 0.27, 0.11,

0.29, 0.20, 0.17, 0.28, 0.06,

0.15, 0.34, 0.03, 0.20, 0.28,

0.22, 0.19, 0.21, 0.18, 0.20),

nrow=5, ncol=5, byrow = TRUE)

# Строим график.

x = graph\_from\_adjacency\_matrix(P, mode = c("directed"), weighted = TRUE)

plot(x)

Код для симуляции работы марковской цепи:

install.packages("markovchain")

library("markovchain")

# Задаем матрицу.

P <- matrix(c(0.18, 0.18, 0.36, 0.23, 0.05,

0.43, 0.13, 0.06, 0.27, 0.11,

0.29, 0.20, 0.17, 0.28, 0.06,

0.15, 0.34, 0.03, 0.20, 0.28,

0.22, 0.19, 0.21, 0.18, 0.20),

nrow=5, ncol=5, byrow = TRUE)

# Функция симуляции работы марковской цепи.

# Параметр steps - число тактов времени.

# Параметр initialState - вектор начального состояния.

# Параметр transitionMatrix - матрица переходов.

simulateMarkov <- function(steps, initialState, transitionMatrix) {

currentState = initialState

for (i in 1:steps) {

currentState = currentState %\*% transitionMatrix

}

return(currentState)

}

# Задаем 3 вектора начальных состояний.

firstState <- c(0, 0.5, 0.1 , 0.3, 0.1)

secondState <- c(1, 0, 0, 0, 0)

thirdState <- c(0.2, 0.2, 0.2, 0.2, 0.2)

#Симуляция работы марковской цепи для 1 состояния:

simulateMarkov(20, firstState, P)

#Симуляция работы марковской цепи для 2 состояния:

simulateMarkov(20, secondState, P)

#Симуляция работы марковской цепи для 3 состояния:

simulateMarkov(20, thirdState, P)

# Находим стационарное распределение вероятностей

# с помощью библиотеки.

chain = new("markovchain", transitionMatrix=P)

steadyStates(chain)

