**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное автономное образовательное учреждение

высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа информационных технологий и робототехники

Направление: «Программная инженерия»

Отделение информационных технологий

Отчет по лабораторной работе №5 по дисциплине

**«Искусственный интеллект и логическое программирование»**

Выполнил:

Студент группы 8К61 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Исламов Е. Р.

Проверил:

Доцент ОИТ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Марухина О.В.

**Цель работы**

Изучение дискретных вероятностных моделей на примере марковских цепей.

**Порядок выполнения работы**

1. Выберите переходную матрицу марковской цепи согласно вашему варианту.

2. Постройте граф по заданной матрице.

3. Задайте случайным образом три различных вектора начальных состояний и для каждого из них смоделируйте работу марковской цепи в течение 20 тактов времени.

4. Рассчитайте вектор финальных вероятностей (стационарное распределение вероятностей) системы и сравните его с векторами вероятностей для каждой начальной ситуации на последнем шаге.

5. Сделайте выводы о проделанной работе.

**Задание**



# Решение

**Построение графика**

Построим граф по матрице переходов с использованием языка R.

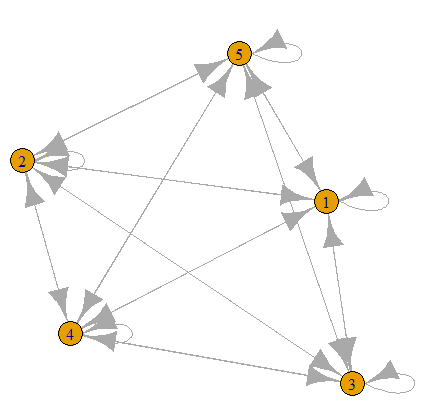


Рисунок 1 - граф переходов

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)

**Моделирование работы марковской цепи.**

Смоделируем работу марковской цепи в течение 20 тактов времени, а также найдем стационарное распределение вероятностей с помощью библиотеки. Исходный код предоставлен ниже.

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)

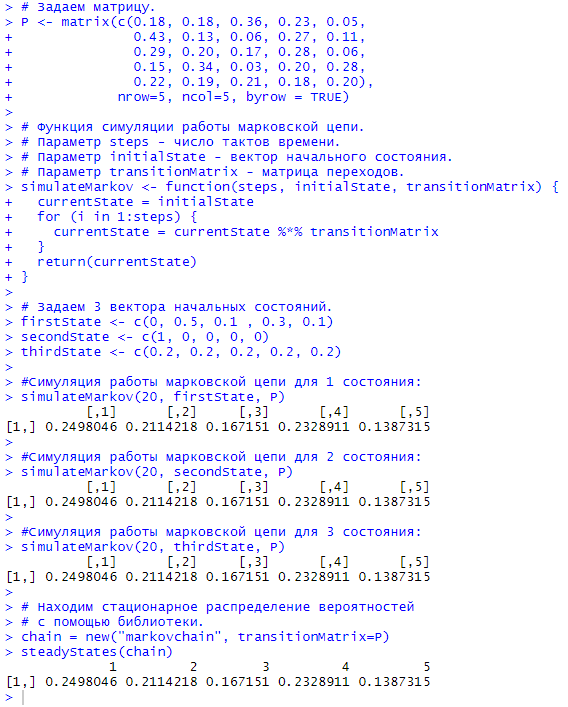


Рисунок 2 – Результат работы программы

Из результата работы программы следует, что симуляция работы марковской цепи проведена правильно, т.к. в результате 20 итераций векторы разных исходных состояний совпали со стационарным распределением вероятностей, найденным с помощью библиотеки.

# Заключение

В результате выполнения лабораторной работы были изучены дискретных вероятностных моделей на примере марковских цепей. Была проведена симуляция работы марковской цепи в течение 20 тактов времени, а также найден вектор финальных вероятностей.