Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Институт Информационных Технологий и Анализа Данных

**Название работы** – “Анализ Марковской цепи с дискретным временем”

Отчет по лабораторной работе “Лабораторная работа №4”

по дисциплине Моделирование систем и процессов

Вариант 21

Выполнил

Студент, номер группы ИСМб-19-1 \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д.Д.Солопов

Дата: \_\_\_\_\_\_\_\_\_\_

Принял

Должность \_\_\_\_\_\_\_\_\_\_\_\_\_\_ О.С.Бучнев

Дата: \_\_\_\_\_\_\_\_\_\_

Иркутск 2021 г.

Содержание

[Цель работы 3](#_Toc69887780)

[Постановка задач 4](#_Toc69887781)

[Решение задач 5](#_Toc69887782)

[1. Построить размеченный граф состояний 5](#_Toc69887783)

[2. Нахождение вероятности состояний: 7](#_Toc69887784)

[3. Решение СЛАУ и проверка на регулярность Марковской цепи 9](#_Toc69887785)

[4 Выполнение моделирования Марковской цепи 11](#_Toc69887786)

[Исходный код 14](#_Toc69887787)

[Ответы на контрольные вопросы 17](#_Toc69887788)

[Вывод 19](#_Toc69887789)

Цель работы

Целью лабораторной работы является ознакомление с теорией цепей Маркова с дискретным временем, а так же применение теории Марковских цепей для ранжирования веб-страниц и моделирования переходов по веб-страницам.

Постановка задач

Задание

Для своего номера варианта (от 1 до 30) получить матрицу вероятностей переходов Марковской цепи (файл Варианты\_R.xls, Лист 3).

Используя возможности языка R и библиотеки markovchain, провести расчет и моделирование заданной Марковской цепи. При этом необходимо получить:

3.1. размеченный граф состояний;

3.2. вероятности состояний после первых 15 переходов (приняв, что изначально система находится в первом состоянии), построить график изменения вероятностей состояний;

3.3. решив систему 4.1, найти стационарное распределение вероятностей состояний, сравнить с вероятностями состояний, полученными на предыдущем этапе, сделать вывод о регулярности Марковской цепи;

3.4. поверить, регулярна ли Марковская цепь, используя соответствующую функцию языка R;

3.5. выполнить моделирование Марковской цепи, осуществив 50 переходов, по полученным статистическим данным оценить матрицу вероятностей переходов, используя квадратичный критерий, оценить отклонение значений полученной матрицы от истинных значений, проверить;

3.6. повторить предыдущий пункт для количества переходов n=100, 200, 500,1000, используя квадратичный критерий оценить точность расчета матрицы вероятностей переходов, построить график зависимости точности от длины реализации.

Составить отчет по лабораторной работе, в который включить все результаты статистического анализа, снабдив их развернутыми пояснениями и иллюстрациями. Пояснения должны содержать описание всех данных, полученных при выполнении пунктов 2.1 – 2.6.

Решение задач

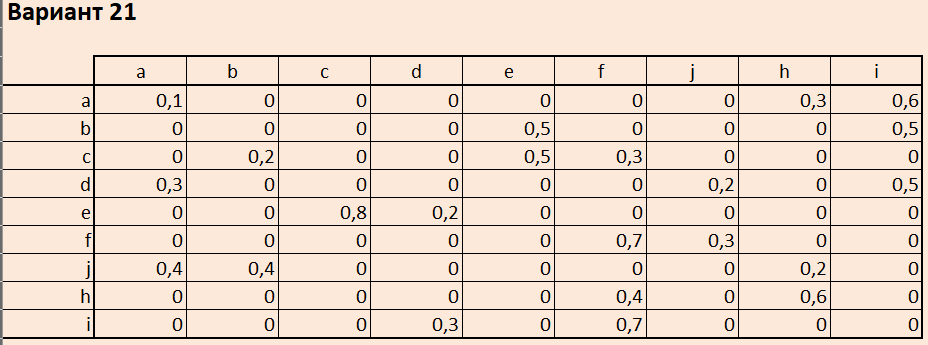


Рисунок 1 – Данные индивидуального варианта

На рисунке 1 видно, что Марковская цепь имеет 9 состояний.

1. Построить размеченный граф состояний

**Код:**

library("markovchain")

t\_matrix=c(

0.1, 0, 0, 0, 0, 0, 0, 0.3, 0.6,

0, 0, 0, 0, 0.5, 0, 0, 0, 0.5,

0, 0.2, 0, 0, 0.5, 0.3, 0, 0, 0,

0.3, 0, 0, 0, 0, 0, 0.2, 0, 0.5,

0, 0, 0.8, 0.2, 0, 0, 0, 0, 0,

0, 0, 0, 0, 0, 0.7, 0.3, 0, 0,

0.4, 0.4, 0, 0, 0, 0, 0, 0.2, 0,

0, 0, 0, 0, 0, 0.4, 0, 0.6, 0,

0, 0, 0, 0.3, 0, 0.7, 0, 0, 0)

#задать цепь

statesNames=c("a","b","c","d","e","f","g","h","i")

true\_matrix=matrix(t\_matrix, nrow=9,

byrow=TRUE, dimnames=list(statesNames, statesNames))

mcA<-new("markovchain", states=statesNames,

transitionMatrix=true\_matrix)

mcA

library(diagram)

plot(mcA, package = "diagram", box.size = 0.06,

main = "Граф состояний Марковской цепи")

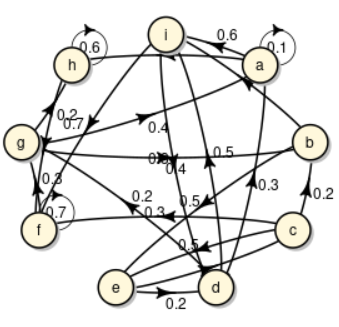


Рисунок 2 – Граф состояний Марковской цепи

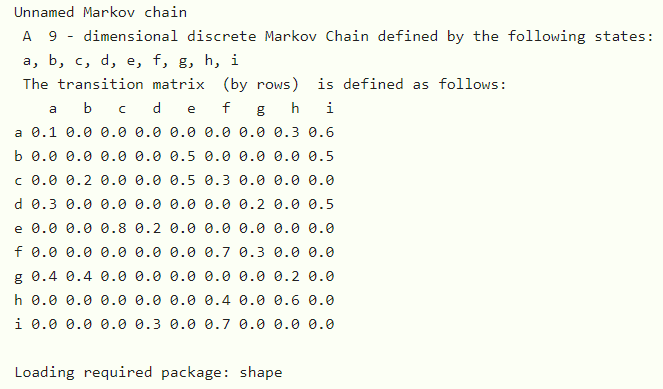


Рисунок 3 – Результат выполнения кода

2. Нахождение вероятности состояний:

**Код для нахождения вероятности состояний после первых 15 переходов:**

time=seq(0,15,1)

p1=(1)

p2=(0)

p3=(0)

p4=(0)

p5=(0)

p6=(0)

p7=(0)

p8=(0)

p9=(0)

n\_=c(1,0,0,0,0,0,0,0,0)

for (i in 1:15){

n\_i=n\_\*mcA^i

c=cat(n\_i[1]," ",n\_i[2]," ",n\_i[3]," ",n\_i[4]," ",n\_i[5]," ",n\_i[6]," ",n\_i[7]," ",n\_i[8]," ",n\_i[9])

print(c)

p1=c(p1,n\_i[1])

p2=c(p2,n\_i[2])

p3=c(p3,n\_i[3])

p4=c(p4,n\_i[4])

p5=c(p5,n\_i[5])

p6=c(p6,n\_i[6])

p7=c(p7,n\_i[7])

p8=c(p8,n\_i[8])

p9=c(p9,n\_i[9])

}

**Код для постройки графиков изменения вероятностей состояний:**

plot(time,p1,ylim=range(c(0,1)), type="l",col="red")

lines(time,p2,col="green")

lines(time,p3,col="blue")

lines(time,p4,col="yellow")

lines(time,p5,col="pink")

lines(time,p6,col="black")

lines(time,p7,col="gray")

lines(time,p8,col="brown")

lines(time,p9,col="orange")

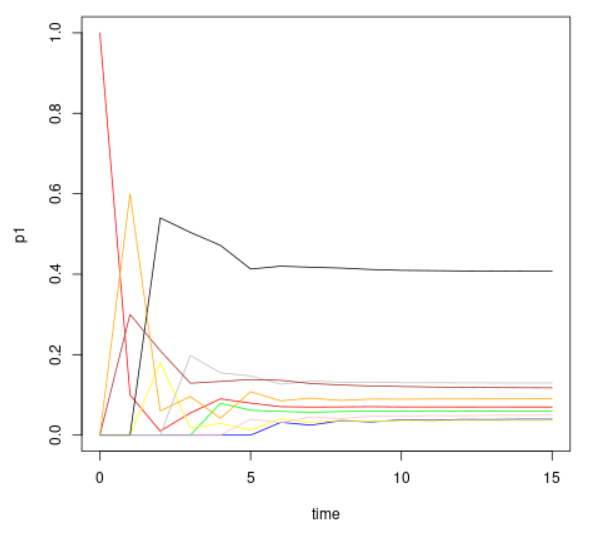


Рисунок 4 - Графики изменения вероятностей состояний

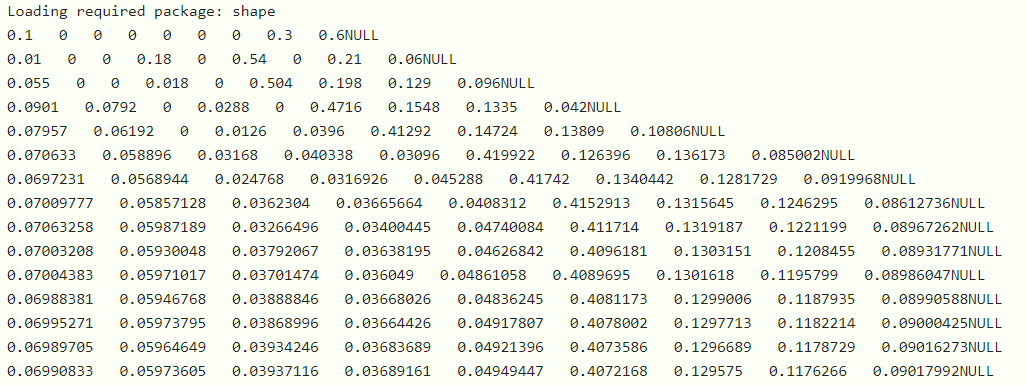


Рисунок 5 – Результат выполнения кода

3. Решение СЛАУ и проверка на регулярность Марковской цепи

**Код для нахождения стационарного распределение вероятностей состояний:**

steadyStates(mcA)

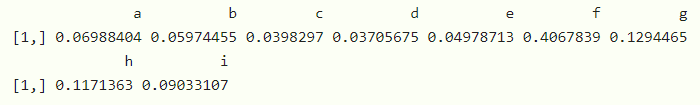


Рисунок 6 - Стационарные вероятности состояний

Вычислим стационарные вероятности состояний. Выпишем систему:

Преобразуем:

Заменим последнее уравнение на условие нормировки :

Решив систему, получим:

В рассматриваемом примере вероятности состояний после 15 переходов совпадают со стационарными. Поэтому есть основания полагать, что цепь регулярна. Проверим это.

**Код для проверки на регулярность:**

is.regular(mcA)



Рисунок 7 – Результат проверки на регулярность

4 Выполнение моделирования Марковской цепи

**Код для моделирования Марковской цепи:**

data=markovchainSequence(n=50, markovchain=mcA, include=TRUE)

data

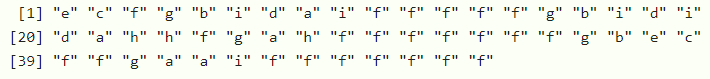


Рисунок 8 – Результат моделирования Марковской цепи

Оценим по имеющейся реализации Марковского случайного процесса матрицу вероятностей переходов .

**Код для оценки реализации Марковского случайного процесса:**

#оценка матрицы

estimated=markovchainFit(

data,

method = "map",

byrow = TRUE,

nboot = 10L,

laplacian = 0,

name = "",

parallel = FALSE,

confidencelevel = 0.99,

confint = FALSE,

hyperparam = matrix(),

sanitize = FALSE,

possibleStates = character()

)##===============

h=estimated$estimate

h

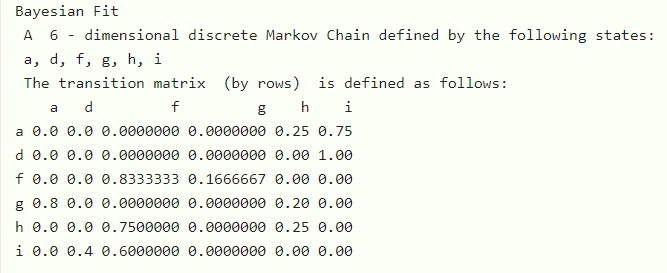


Рисунок 9 - Оценка матрицы вероятностей переходов

По формуле оценим отклонение эмпирических значений от истинных. Для этого сначала преобразуем объект h в одномерный массив est\_matrix:

est\_matrix=NULL

for (i in 1:9){

for (j in 1:9){

est\_matrix=c(est\_matrix,h[i,j])

}

}

est\_matrix

Затем найдем отклонение:

sum=0

for (i in 1:81){

sum=sum+(t\_matrix[i]-est\_matrix[i])^2

}

sum

q=sqrt(sum)

q



Рисунок 10 - Оценка отклонения истинных значений матрицы вероятностей переходов от эмпирических

Повторим эксперименты при длинах эмпирических реализаций, равных n=100, 200, 500,1000:

q\_=q

n\_=c(100,200,500,1000)

for (n in n\_) {

#моделирование

len=n

data=markovchainSequence(n=len, markovchain=mcA, include=TRUE)

data

#оценка матрицы

estimated=markovchainFit(

data,

method = "map",

byrow = TRUE,

nboot = 10L,

laplacian = 0,

name = "",

parallel = FALSE,

confidencelevel = 0.99,

confint = FALSE,

hyperparam = matrix(),

sanitize = FALSE,

possibleStates = character()

)##===============

h=estimated$estimate

h

est\_matrix=NULL

for (i in 1:9){

for (j in 1:9){

est\_matrix=c(est\_matrix,h[i,j])

}

}

sum=0

for (i in 1:81){

sum=sum+(t\_matrix[i]-est\_matrix[i])^2

}

sum

q\_=c(q\_,sqrt(sum))

}

n\_=c(50,n\_)

n\_

q\_

plot(n\_,q\_, type="l",col="red")

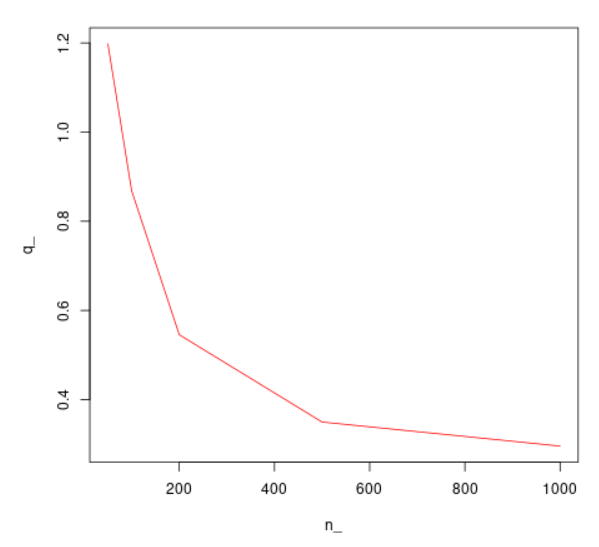


Рисунок 11 - Значения отклонений для оценки матрицы вероятностей переходов



Рисунок 12 - Значения отклонений для оценки матрицы вероятностей переходов

Исходный код

library("markovchain")

t\_matrix=c(

0.1, 0, 0, 0, 0, 0, 0, 0.3, 0.6,

0, 0, 0, 0, 0.5, 0, 0, 0, 0.5,

0, 0.2, 0, 0, 0.5, 0.3, 0, 0, 0,

0.3, 0, 0, 0, 0, 0, 0.2, 0, 0.5,

0, 0, 0.8, 0.2, 0, 0, 0, 0, 0,

0, 0, 0, 0, 0, 0.7, 0.3, 0, 0,

0.4, 0.4, 0, 0, 0, 0, 0, 0.2, 0,

0, 0, 0, 0, 0, 0.4, 0, 0.6, 0,

0, 0, 0, 0.3, 0, 0.7, 0, 0, 0)

#задать цепь

statesNames=c("a","b","c","d","e","f","g","h","i")

true\_matrix=matrix(t\_matrix, nrow=9,

byrow=TRUE, dimnames=list(statesNames, statesNames))

mcA<-new("markovchain", states=statesNames,

transitionMatrix=true\_matrix)

mcA

library(diagram)

plot(mcA, package = "diagram", box.size = 0.06,

main = "Граф состояний Марковской цепи")

time=seq(0,15,1)

p1=(1)

p2=(0)

p3=(0)

p4=(0)

p5=(0)

p6=(0)

p7=(0)

p8=(0)

p9=(0)

n\_=c(1,0,0,0,0,0,0,0,0)

for (i in 1:15){

n\_i=n\_\*mcA^i

c=cat(n\_i[1]," ",n\_i[2]," ",n\_i[3]," ",n\_i[4]," ",n\_i[5]," ",n\_i[6]," ",n\_i[7]," ",n\_i[8]," ",n\_i[9])

print(c)

p1=c(p1,n\_i[1])

p2=c(p2,n\_i[2])

p3=c(p3,n\_i[3])

p4=c(p4,n\_i[4])

p5=c(p5,n\_i[5])

p6=c(p6,n\_i[6])

p7=c(p7,n\_i[7])

p8=c(p8,n\_i[8])

p9=c(p9,n\_i[9])

}

plot(time,p1,ylim=range(c(0,1)), type="l",col="red")

lines(time,p2,col="green")

lines(time,p3,col="blue")

lines(time,p4,col="yellow")

lines(time,p5,col="pink")

lines(time,p6,col="black")

lines(time,p7,col="gray")

lines(time,p8,col="brown")

lines(time,p9,col="orange")

steadyStates(mcA)

is.regular(mcA)

data=markovchainSequence(n=50, markovchain=mcA, include=TRUE)

data

#оценка матрицы

estimated=markovchainFit(

data,

method = "map",

byrow = TRUE,

nboot = 10L,

laplacian = 0,

name = "",

parallel = FALSE,

confidencelevel = 0.99,

confint = FALSE,

hyperparam = matrix(),

sanitize = FALSE,

possibleStates = character()

)##===============

h=estimated$estimate

h

est\_matrix=NULL

for (i in 1:9){

for (j in 1:9){

est\_matrix=c(est\_matrix,h[i,j])

}

}

est\_matrix

sum=0

for (i in 1:81){

sum=sum+(t\_matrix[i]-est\_matrix[i])^2

}

sum

q=sqrt(sum)

q

q\_=q

n\_=c(100,200,500,1000)

for (n in n\_) {

#моделирование

len=n

data=markovchainSequence(n=len, markovchain=mcA, include=TRUE)

data

#оценка матрицы

estimated=markovchainFit(

data,

method = "map",

byrow = TRUE,

nboot = 10L,

laplacian = 0,

name = "",

parallel = FALSE,

confidencelevel = 0.99,

confint = FALSE,

hyperparam = matrix(),

sanitize = FALSE,

possibleStates = character()

)##===============

h=estimated$estimate

h

est\_matrix=NULL

for (i in 1:9){

for (j in 1:9){

est\_matrix=c(est\_matrix,h[i,j])

}

}

sum=0

for (i in 1:81){

sum=sum+(t\_matrix[i]-est\_matrix[i])^2

}

sum

q\_=c(q\_,sqrt(sum))

}

n\_=c(50,n\_)

n\_

q\_

plot(n\_,q\_, type="l",col="red")

Ответы на контрольные вопросы

1. Цепь Маркова — последовательность случайных событий с конечным или счётным числом исходов, где вероятность наступления каждого события зависит только от состояния, достигнутого в предыдущем событии. Характеризуется тем свойством, что, говоря нестрого, при фиксированном настоящем будущее независимо от прошлого.

Важным классом случайных процессов являются Марковские случайные процессы (процессы без последействия). Особенность Марковских процессов заключается в том, что прогноз о будущем состоянии системы зависит только от того, в каком состоянии находится система в текущий момент, и не зависит от того, в каких состояниях находилась система в прошлом.

В том случае, когда множество состояний конечно, а переход происходит в строго определенные, изолированные моменты времени , такой случайный процесс называют Марковской цепью с дискретным временем и конечным числом состояний.

Одна из главных целей исследования Марковских цепей это нахождение закона распределения вероятностей состояний – т.е. найти вероятность для каждого состояния, в котором будет находиться система через конечное число переходов.

Распределение вероятностей состояний Марковской цепи, не изменяющееся со временем, называется стационарным.

Одной из первых научных дисциплин, в которой цепи Маркова нашли практическое применение, стала лингвистика (в частности текстология).

2. Распределение вероятностей состояний Марковской цепи, не изменяющееся со временем, называется стационарным.

Распределение вероятностей состояний Марковской цепи, называется предельным (финальным) (иными словами, распределение вероятностей состояний после «бесконечного» числа переходов).

Состояние называется существенным, если выйдя из этого состояния, система может в него вернуться за один или несколько переходов. В противном случае, состояние называется несущественным.

Марковская цепь называется регулярной тогда, когда из любого существенного состояния можно попасть в любое другое существенное состояние за конечное число переходов.

Если система S может переходить из одного состояния в другое не в строго определенные, а в произвольные моменты времени, причем множество таких моментов является непрерывным, и выполняется Марковское свойство, то такая система называется Марковским процессом с непрерывным временем. Здесь, как и прежде, исследуются вероятности состояний системы, но не за конечное число переходов, а через определенный момент времени.

Однородную Марковскую цепь обозначают в виде размеченного графа состояний. Для Марковской цепи с тремя состояниями: a, b, c, заданной матрицей вероятностей переходов

3. Для того, чтобы найти вероятности состояний системы после одного, двух и большего числа переходов, рассмотрим матрицу , где – вероятность того, что после переходов, система находится в состоянии Для элементов этой матрицы при любом выполняется равенство . При имеем матрицу – начальное распределение вероятностей.

Матрицы удовлетворяют следующим матричным уравнениям:

или . (4.2)

Эти уравнения позволяют определить вероятности состояний после k-го шага по известному начальному распределению и заданной матрице переходных вероятностей за один шаг.

Для нахождения стационарного распределения вероятностей состояний необходимо решить матричное уравнение:

, (4.3)

или в координатной форме:

. (4.4)

Для получения единственного решения к уравнению (4.4) необходимо добавлять условие нормировки:

. (4.5)

Если Марковская цепь регулярна, то предельные вероятности совпадают со стационарными вероятностями.

Вывод

Было осуществлено ознакомление с теорией цепей Маркова с дискретным временем, применение теории Марковских цепей для ранжирования веб-страниц и моделирование переходов по веб-страницам.

С помощью языка R и библиотеки markovchain были решены практические задачи. Был построен размеченный граф состояний Марковской цепи, были построены вероятности состояний после первых 15 переходов и график изменения вероятностей состояний. Было найдено стационарное распределение вероятностей после решения системы уравнений и осуществлено сравнение с вероятностями состояний, после чего был сделан вывод о регулярности Марковской цепи, так предельные вероятности Марковской цепи совпали со стационарными. Средствами языка R была осуществлена проверка регулярности цепи Маркова, которая подтвердила вывода о регулярности при сравнении стационарных вероятностей с предельными вероятностями, полученными практическим путём (решение СЛАУ). Было выполнено моделирование Марковской цепи для количества переходов 50, 100, 200, 500 и 1000. Была оценена точность расчёта матрицы вероятностей переходов и построен график зависимости точности от длины реализации.