Introduction to Stochastic Processes:

Definitions, Simulations, and Interesting Applications

By

Dr. Olga Korosteleva, CSULB

SOCAL R USER GROUP
FEBRUARY 11, 2023

About Me

- BS in Mathematics, Wayne State University, Detroit, MI, 1996
- MS in Statistics, Purdue University, West Lafayette, IN, 1998
- Ph.D. in Statistics, Purdue University, West Lafayette, IN, 2002
- Professor of Statistics, CSU, Long Beach, 2002-present

Outline

- ❑Markov Chain
- ❑Random Walk
- ❑Poisson Process
- ❑Birth-and-Death Process
- ❑Branching Process
- ❑Brownian Motion

Greek letters

$\boldsymbol{\pi}$ pi /pai/

$\boldsymbol{\chi}$ chi /kai/

$\boldsymbol{\lambda}$ lambda /lam-da/

$\boldsymbol{\mu}$ mu /myoo/

$\boldsymbol{\sigma}$ sigma /sig-ma/

$\boldsymbol{\theta}$ theta /thei-ta/

MARKOV CHAIN

Definition. A ***Markov chain*** (MC) is a collection of random variables $\{X_n, n = 0, \dots\}$ indexed by step n, with the property that X_{n+1} depends only on X_n and not $X_1, \dots, X_{n-1}$. In mathematical terms,

$$P(X_{n+1} = j \mid X_0 = i_0, \dots, X_{n-1} = i_{n-1}, X_n = i)$$

$$= P(X_{n+1} = j \mid X_n = i).$$

The collection of all possible values that a Markov chain call assume is called the ***state space***. If the state space is finite, the Markov chain is called ***finite***. The collection of all probabilities to transition from state i to state j is called a ***one-step transition probability matrix*****.**

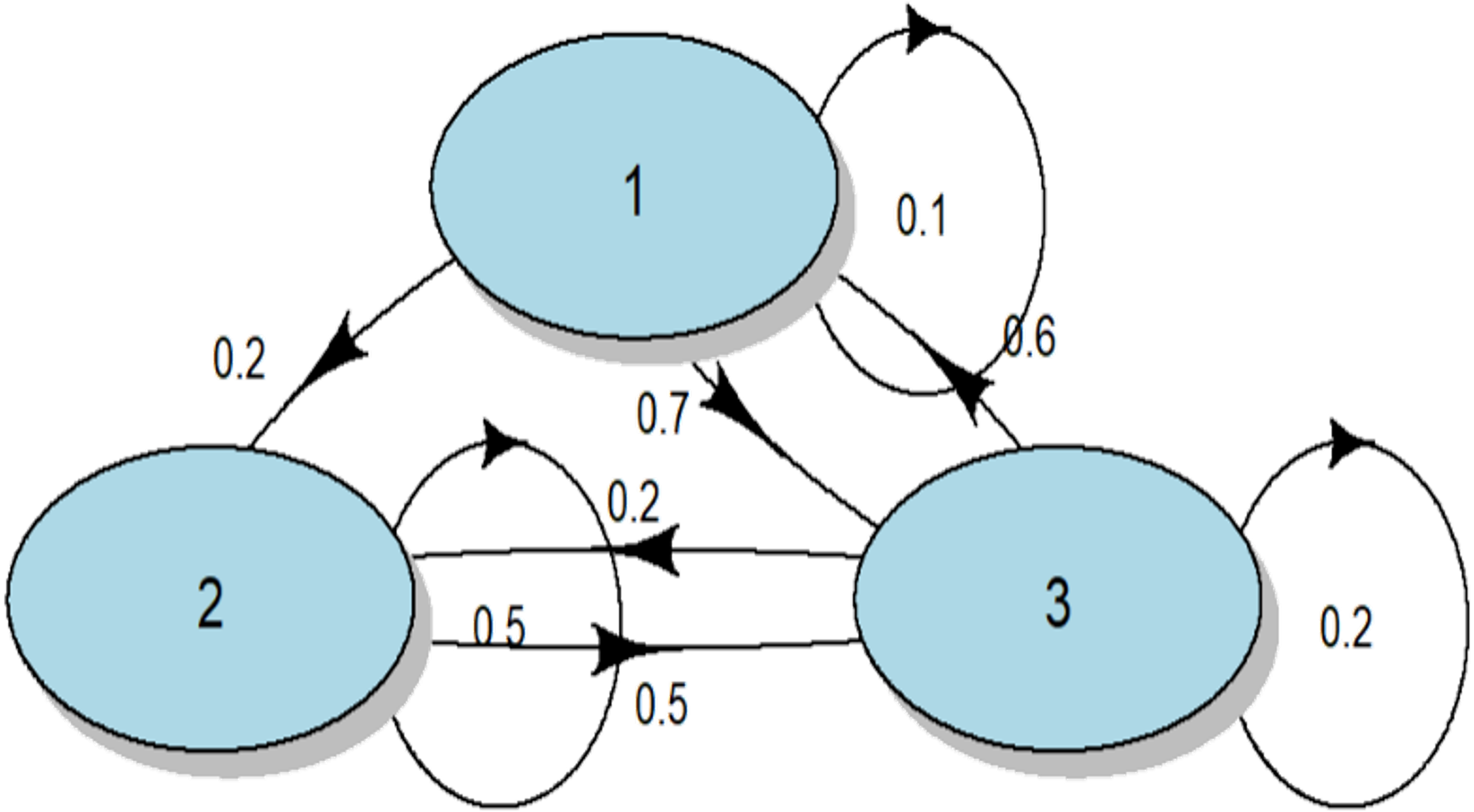
Example of Markov Chain

Consider a Markov chain with states 1, 2, and 3, and a one-step transition probability matrix

$$P = \begin{array}{c c} & \begin{array}{ccc} \mathbf{1} & \mathbf{2} & \mathbf{3} \end{array} \\ \begin{array}{c} \mathbf{1} \\ \mathbf{2} \\ \mathbf{3} \end{array} & \begin{bmatrix} 0.1 & 0.2 & 0.7 \\ 0 & 0.5 & 0.5 \\ 0.6 & 0.2 & 0.2 \end{bmatrix} \end{array}.$$

Note that probabilities in each row must add up to 1 because the chain must transition somewhere.

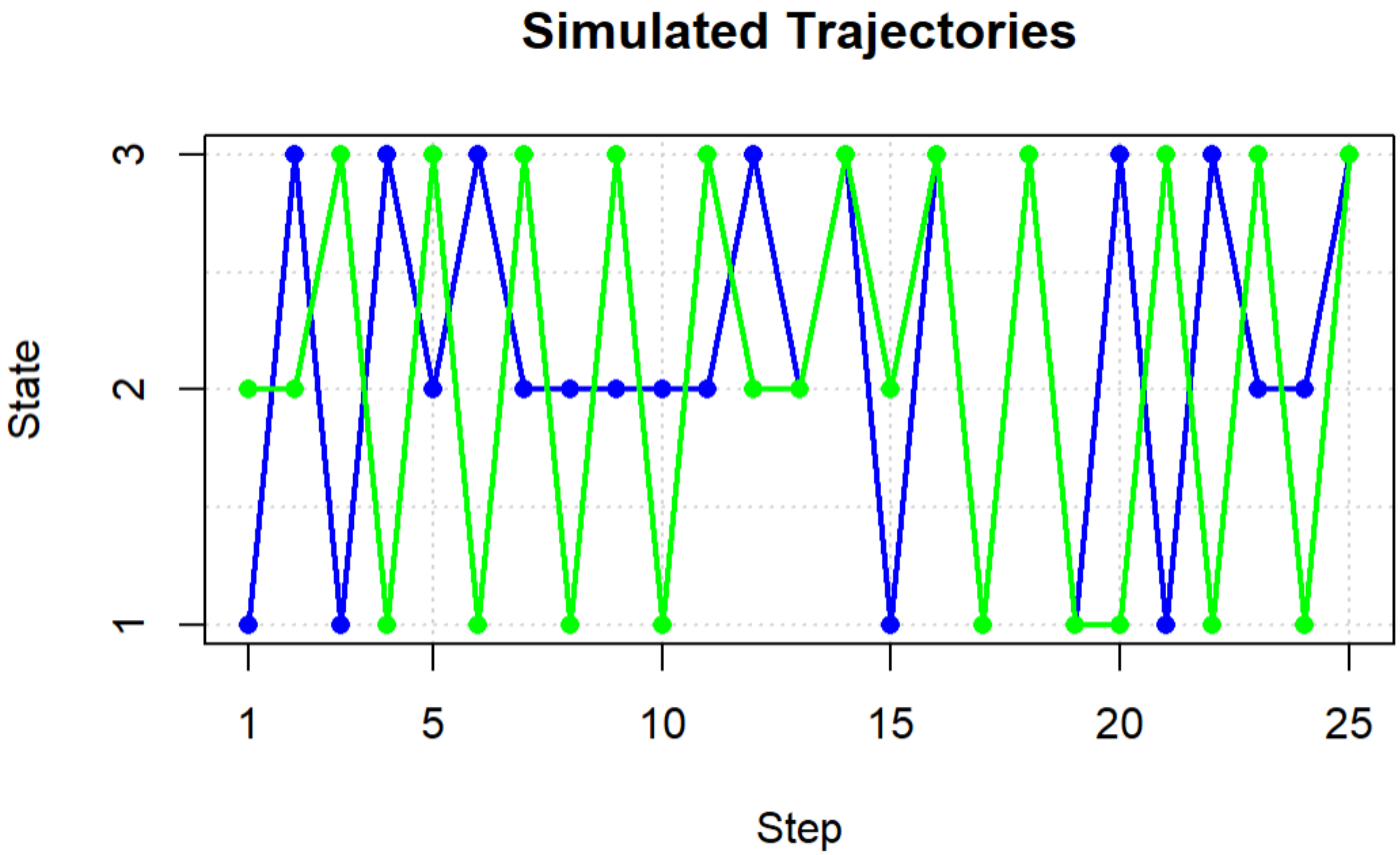
Example of Markov Chain (Cont.)



R CODE

Example of Markov Chain (Cont.)

R CODE



Example of Markov Chain (Cont.)

<u>Definition.</u> ***Steady-state*** (or ***limiting***) ***probabilities*** are long-term proportions that a Markov chain spends in each state. Assume that are k states. Let $\pi_j = \lim_{n\to\infty} P(X_n = j)$ be the steady-state probability of state j, and let $P_{ij} = P(X_{n+1} = j \,|X_n = i)$ be the one-step transition probability from state i to state j. Then steady-state probabilities satisfy the system of linear equations:

$$\pi_j = \sum_{i=1}^{k} P_{ij}\,\pi_i\,, \text{ and } \quad \pi_1 + \pi_2 + \cdots + \pi_k = 1.$$

Proof: $P(X_{n+1} = j)$= $\sum_{i=1}^{k} P(X_{n+1} = j\,|X_n = i)P(X_n = i)$. Passing to the limit when n tends to infinity, we get the result.

Example of Markov Chain (Cont.)

In our example, the one-step transition probability matrix is

$$P = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0.1 & 0.2 & 0.7 \\ 2 & 0 & 0.5 & 0.5 \\ 3 & 0.6 & 0.2 & 0.2 \end{array},$$

therefore, the steady-state probabilities satisfy the system:

$$\begin{cases} \pi_1 = 0.1\pi_1 + 0\pi_2 + 0.6\pi_3, \\ \pi_2 = 0.2\pi_1 + 0.5\pi_2 + 0.2\pi_3, \\ \pi_1 + \pi_2 + \pi_3 = 1. \end{cases}$$

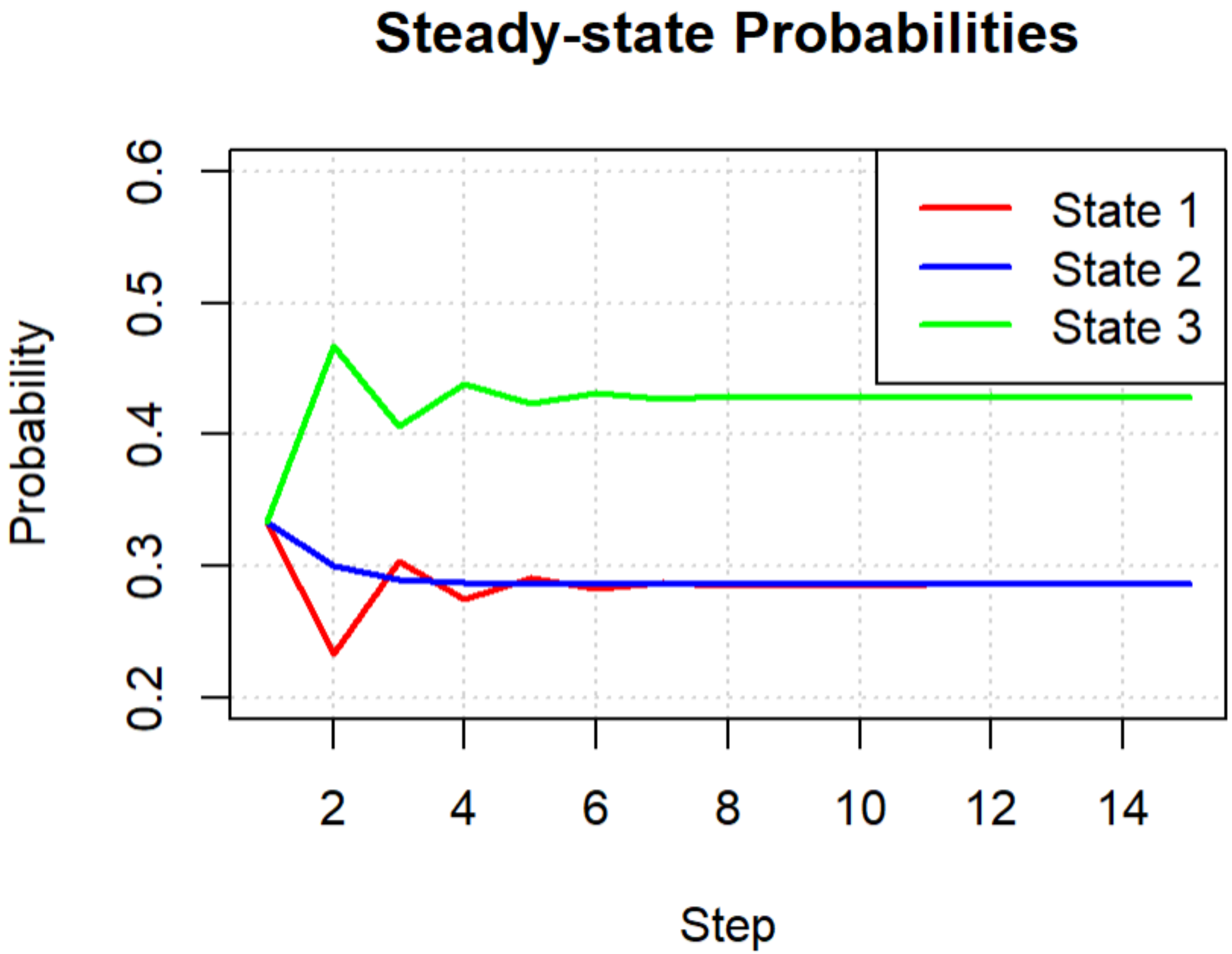
round(steadyStates(mc), digits=4)

1	**2**	**3**
0.2857	**0.2857**	**0.4286**

The chain will spend 28.57% of the time in state 1, 28.57% in state 2, and 42.86% in state 3.

Example of Markov Chain (Cont.)

R CODE



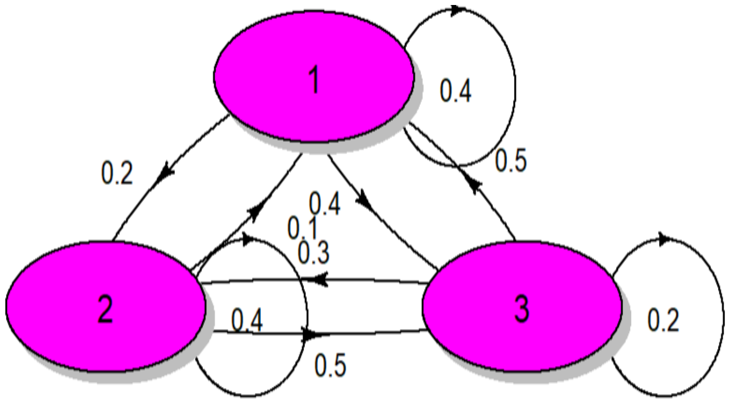
Markov Chain Exercise

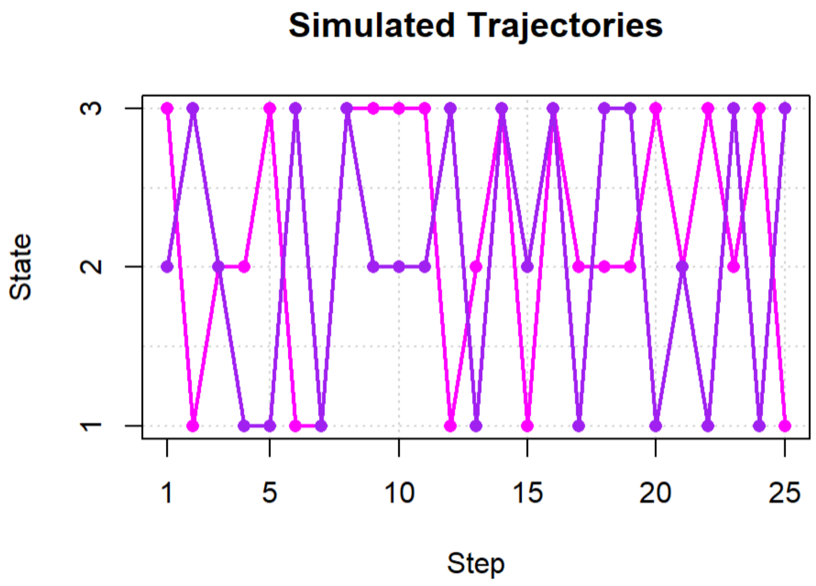
Consider a Markov chain with one-step transition probability matrix

$$P = \begin{array}{c} \\ \mathbf{1} \\ \mathbf{2} \\ \mathbf{3} \end{array} \begin{array}{c} \begin{array}{ccc} \mathbf{1} & \mathbf{2} & \mathbf{3} \end{array} \\ \begin{bmatrix} 0.4 & 0.2 & 0.4 \\ 0.1 & 0.4 & 0.5 \\ 0.5 & 0.3 & 0.2 \end{bmatrix} \end{array}.$$

1. Construct a well-labeled diagram for this Markov chain.
2. Simulate two trajectories starting at a randomly chosen state.
3. Compute theoretical values for the steady-state probabilities. Interpret.
4. Depict convergence to the steady-state probabilities. How many steps are required for the chain to settle down?

Markov Chain Exercise Solution



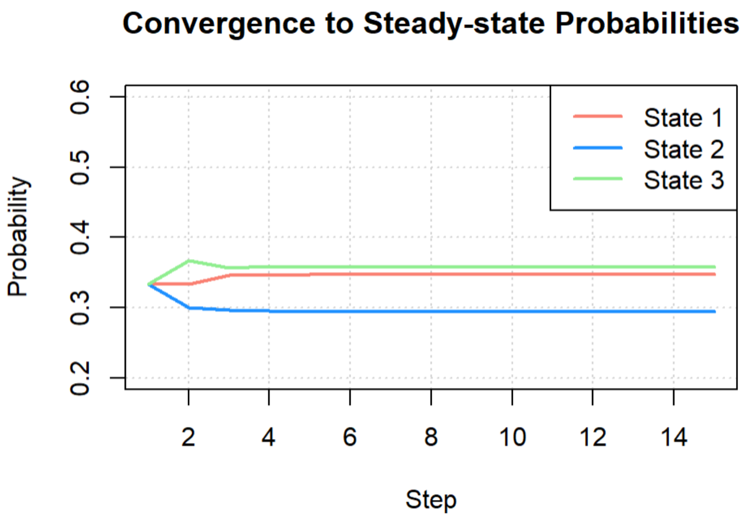


[R CODE]

> round(steadyStates(mc), digits=4)

1	2	3
0.3474	**0.2947**	**0.3579**

34.7% in state 1, 29.5% in state 2, 35.8% in state 3



Application of Markov Chain (1)

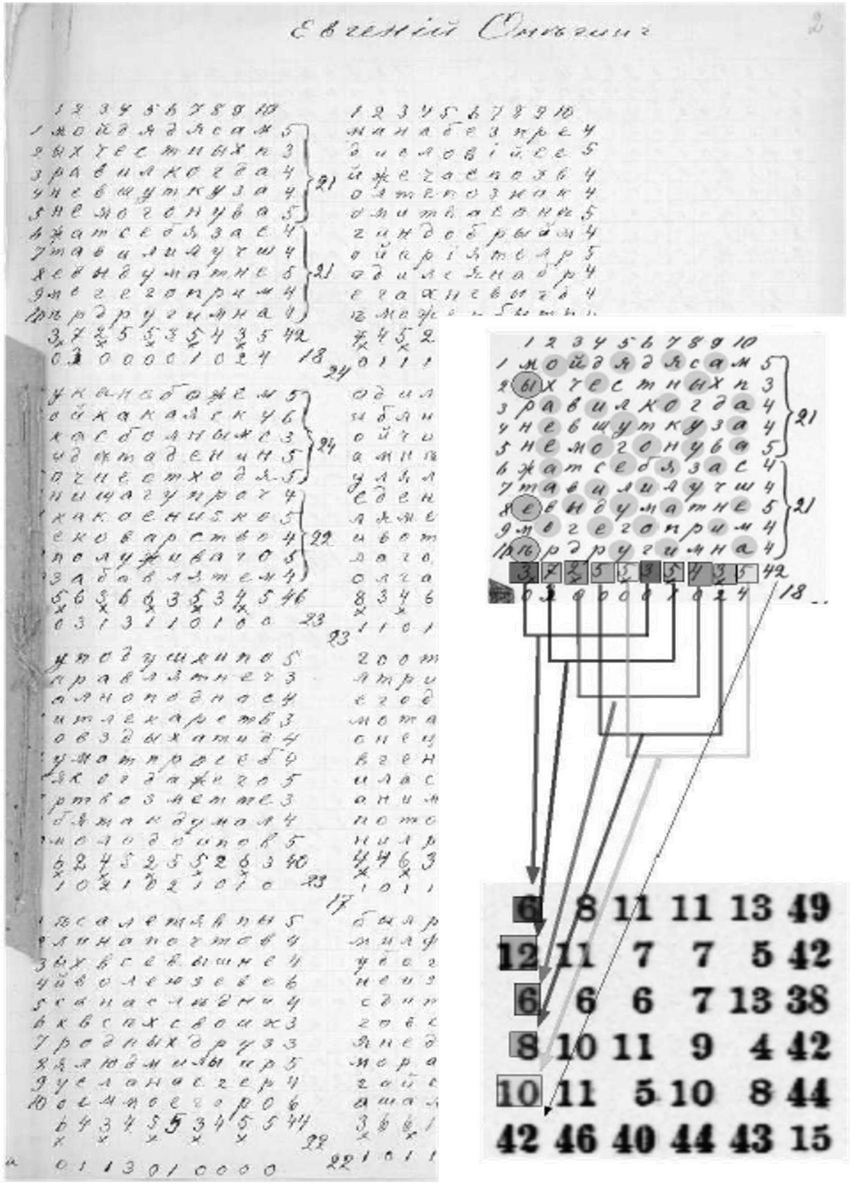
А. А. Марков (1886).

Markov, A. A. (1913). An example of statistical investigation of the text "Eugene Onegin" concerning the connection of samples in chains. (In Russian.) *Bulletin of the Imperial Academy of Sciences of St. Petersburg*, 7(3): 153-162.

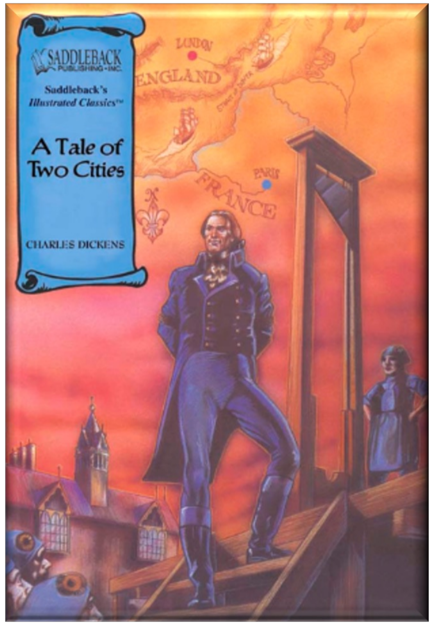
Application of Markov Chain (1) (Cont.)

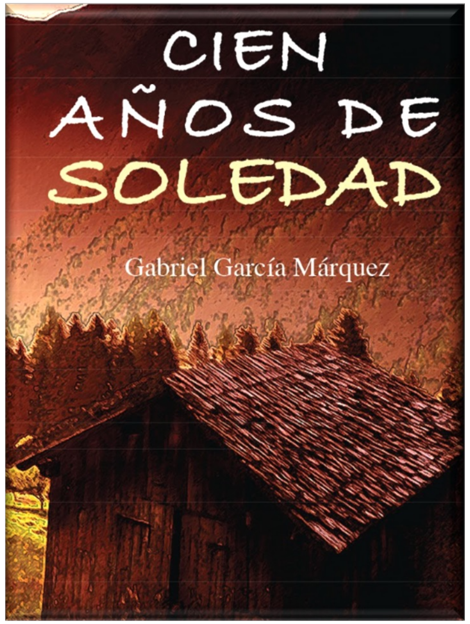
A.A. Markov took the first 20,000 letters of "Eugene Onegin" by A.S. Pushkin (just a string of consonants and vowels of Cyrillic alphabet) and computed a one-step transition probability matrix

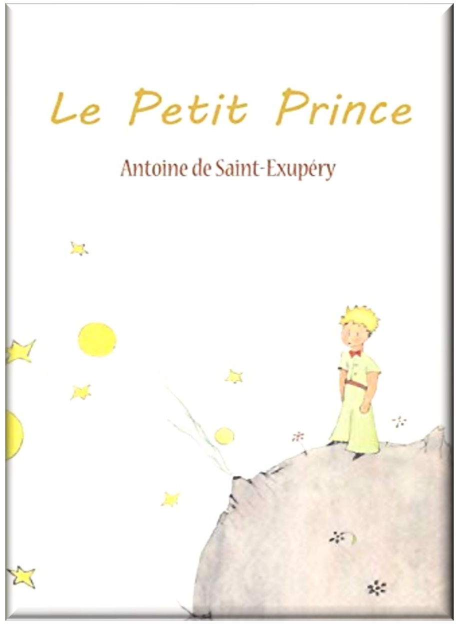
$$
\mathbf{P} = \begin{array}{cc} & \begin{array}{cc} \mathrm{v} & \mathrm{c} \end{array} \\ \begin{array}{c} \mathrm{v} \\ \mathrm{c} \end{array} & \begin{bmatrix} \frac{1104}{8637} = 0.1278 & \frac{7533}{8637} = 0.8722 \\ \frac{7534}{11362} = 0.6631 & \frac{3828}{11362} = 0.3369 \end{bmatrix} \end{array}.
$$



Application of Markov Chain (1) (Cont.)







R CODE

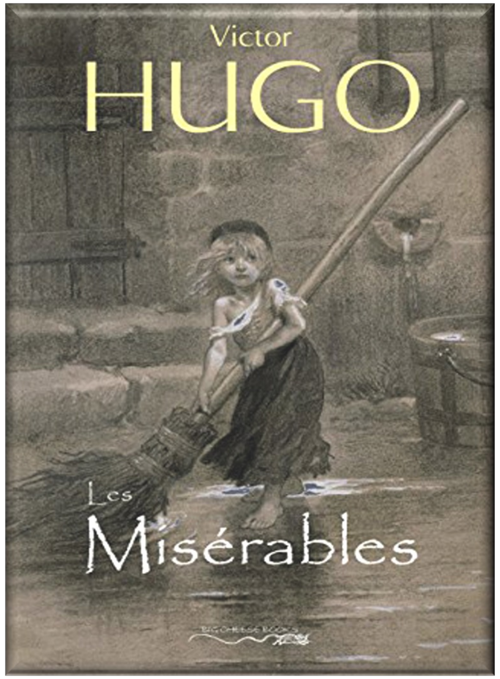
	[v]	[c]
[v]	0.1370	0.8630
[c]	0.6681	0.3319

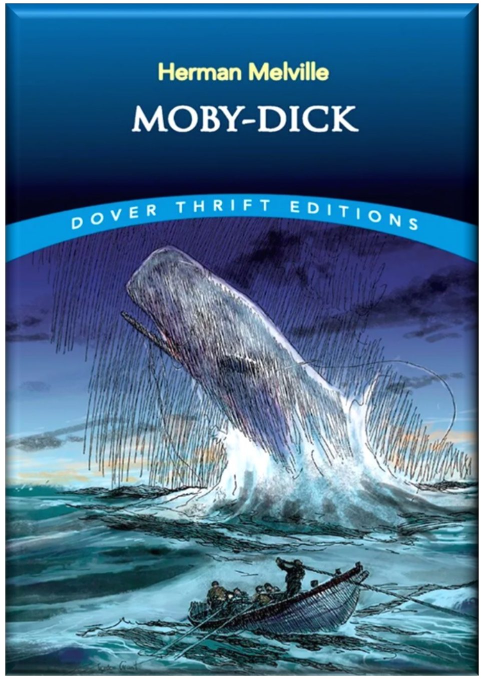
	[v]	[c]
[v]	0.1464	0.8536
[c]	0.5206	0.4794

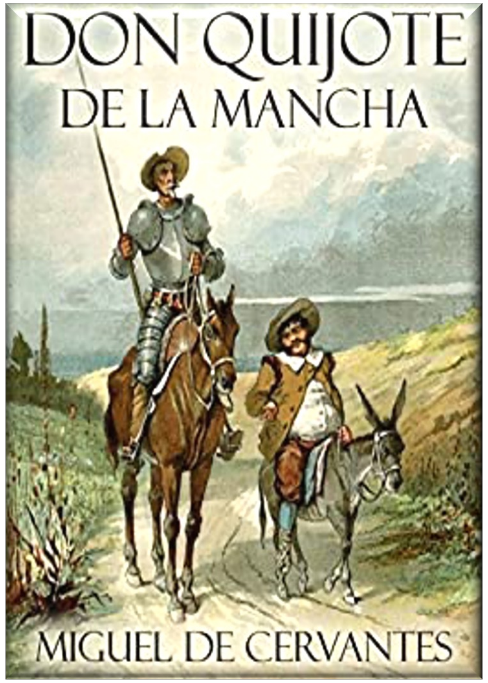
	[v]	[c]
[v]	0.1544	0.8456
[c]	0.7027	0.2973

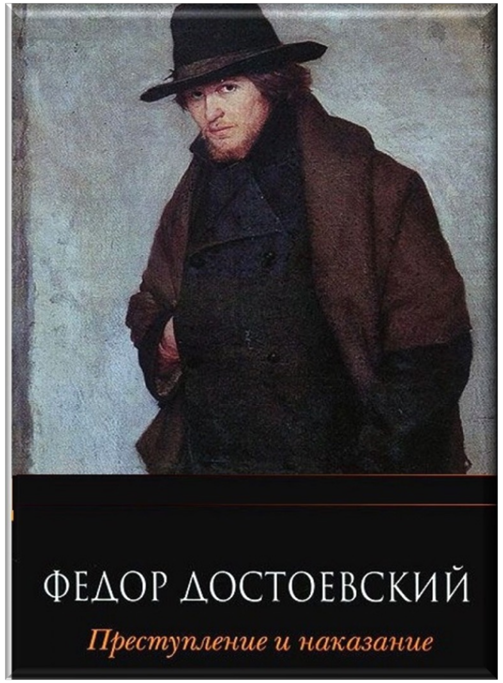
	[v]	[c]
[v]	0.2284	0.7716
[c]	0.6069	0.3931

Exercise: Match book with matrix









	[v]	[c]
[v]	0.1640	0.8360
[c]	0.6996	0.3004

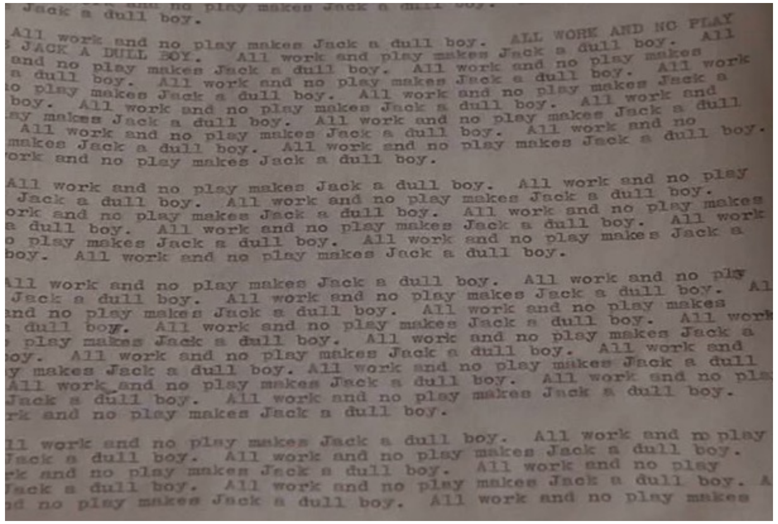
	[v]	[c]
[v]	0.2392	0.7608
[c]	0.6097	0.3903

	[v]	[c]
[v]	0.1422	0.8578
[c]	0.5132	0.4868

	[v]	[c]
[v]	0.1384	0.8616
[c]	0.6866	0.3134

Application of Markov Chain (2)

Staying on the topic of literature, recall that in the famous 1980 movie “Shining”, the main character Jack Torrance (played by Jack Nicholson) repeatedly typed on a typewriter the sentence *All work and no play makes Jack a dull boy*. This “composition” is a deterministic and is NOT a Markov chain.



Jack a dull boy.

All work and no play makes Jack a dull boy. ALL WORK AND NO PLAY
JACK A DULL BOY. All work and play makes Jack a dull boy. All
and no play makes Jack a dull boy. All work and no play makes
a dull boy. All work and no play makes Jack a dull boy. All work
o play makes Jack a dull boy. All work and no play makes Jack a
boy. All work and no play makes Jack a dull boy. All work and
ay makes Jack a dull boy. All work and no play makes Jack a dull
All work and no play makes Jack a dull boy. All work and no
makes Jack a dull boy. All work and no play makes Jack a dull boy.
ork and no play makes Jack a dull boy.

All work and no play makes Jack a dull boy. All work and no play
Jack a dull boy. All work and no play makes Jack a dull boy.
ork and no play makes Jack a dull boy. All work and no play makes
a dull boy. All work and no play makes Jack a dull boy. All work
o play makes Jack a dull boy. All work and no play makes Jack a
boy. All work and no play makes Jack a dull boy.

ll work and no play makes Jack a dull boy. All work and no play
Jack a dull boy. All work and no play makes Jack a dull boy.
and no play makes Jack a dull boy. All work and no play makes
dull boy. All work and no play makes Jack a dull boy. All work
play makes Jack a dull boy. All work and no play makes Jack a
oy. All work and no play makes Jack a dull boy. All work and
y makes Jack a dull boy. All work and no play makes Jack a dull
All work and no play makes Jack a dull boy. All work and no pl
Jack a dull boy. All work and no play makes Jack a dull boy.
rk and no play makes Jack a dull boy.

ll work and no play makes Jack a dull boy. All work and no play
Jack a dull boy. All work and no play makes Jack a dull boy.
k and no play makes Jack a dull boy. All work and no play
Jack a dull boy. All work and no play makes Jack a dull boy.
d no play makes Jack a dull boy. All work and no play makes

Application of Markov Chain (2) (Cont.)

Proof: Suppose he typed it up k times, where k is a very large number. If we turn the letters into vowels (v) (a, e, i, o, u), and consonants (c) (the remaining 21 letters), we get a repeating pattern:

allworkandnoplaymakesjackadullboy|allworkandnoplay…

vcccvccvcccvccvccvcvccvccvcvcccvc|vcccvccvcccvccvc…

Application of Markov Chain (2) (Cont.)

We want to show that the Markovian property doesn't hold, for instance, for the triples $\{ccc\}$ and thus this chain is non-Markov. In the entire composition there are $33k$ letters, of which $11k$ are vowels and $22k$ are consonants. Also, pattern $\{ccc\}$ occurs $3k$ times, $\{ccv\}$ occurs $8k$ times, $\{vcc\}$ occurs $8k$ times, and $\{vcv\}$ occurs $2k + k - 1 = 3k - 1$ times because two such triples lie inside each sentence and one appears at each of $k - 1$ seams. Thus, we find

Application of Markov Chain (2) (Cont.)

$$\widehat{\mathbb{P}}(X_3 = \mathtt{c} \mid X_2 = \mathtt{c},\ X_1 = \mathtt{c}) = \frac{\widehat{\mathbb{P}}(X_1 = \mathtt{c},\ X_2 = \mathtt{c},\ X_3 = \mathtt{c})}{\widehat{\mathbb{P}}(X_1 = \mathtt{c},\ X_2 = \mathtt{c})}$$

$$= \frac{\widehat{\mathbb{P}}(X_1 = \mathtt{c},\ X_2 = \mathtt{c},\ X_3 = \mathtt{c})}{\widehat{\mathbb{P}}(X_1 = \mathtt{c},\ X_2 = \mathtt{c},\ X_3 = \mathtt{v}) + \widehat{\mathbb{P}}(X_1 = \mathtt{c},\ X_2 = \mathtt{c},\ X_3 = \mathtt{c})}$$

$$= \frac{\widehat{\mathbb{P}}(\mathtt{ccc})}{\widehat{\mathbb{P}}(\mathtt{ccv}) + \widehat{\mathbb{P}}(\mathtt{ccc})} = \frac{3k}{8k+3k} = \frac{3}{11}.$$

On the other hand, we estimate

$$\widehat{\mathbb{P}}(X_3 = \mathtt{c} \mid X_2 = \mathtt{c}) = \frac{\widehat{\mathbb{P}}(X_2 = \mathtt{c},\ X_3 = \mathtt{c})}{\widehat{\mathbb{P}}(X_2 = \mathtt{c})}$$

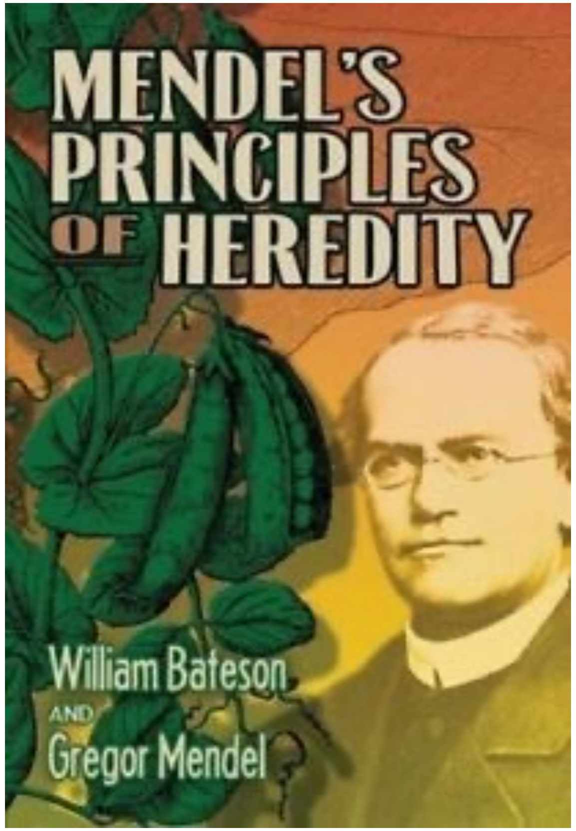
$$= \frac{\widehat{\mathbb{P}}(\mathtt{ccc}) + \widehat{\mathbb{P}}(\mathtt{vcc})}{\widehat{\mathbb{P}}(\mathtt{ccc}) + \widehat{\mathbb{P}}(\mathtt{ccv}) + \widehat{\mathbb{P}}(\mathtt{vcc}) + \widehat{\mathbb{P}}(\mathtt{vcv})} = \frac{3k+8k}{3k+8k+8k+3k-1}$$

$$= \frac{11k}{22k-1} \approx \frac{1}{2}, \quad \text{for large } k.$$

Since $\frac{3}{11} \neq \frac{11k}{22k-1}$, this chain is not a Markov chain.

Application of Markov Chain (3)

According to the Mendelian model of gene inheritance in humans, named after Gregor Johann Mendel (1822-1884), a specific genetic trait is determined by a pair of genes, that can be of three types: AA, Aa, or aa. During reproduction, an offspring inherits one gene of the pair from each parent, and genes are selected at random, independently of each other.



Application of Markov Chain (3) (Cont.)

Suppose gene a is a mutant gene. We consider the genotype of the offspring in successive generations if the **second parent always has genotype $\boldsymbol{AA}$**. This can be presented as a Markov chain with the state space $S = \{AA, Aa, aa\}$ and a one-step transition probability matrix

$$\mathbf{P} = \begin{array}{c} \\ AA \\ Aa \\ aa \end{array} \begin{array}{c} \begin{array}{ccc} AA & Aa & aa \end{array} \\ \begin{bmatrix} 1 & 0 & 0 \\ 0.5 & 0.5 & 0 \\ 0 & 1 & 0 \end{bmatrix} \end{array}.$$

Indeed, if parents have genes (AA, AA), then their offspring are bound to have genes AA with probability 1. If parents have genes (Aa, AA), they are equally likely to spit into AA or Aa, and finally, if parents have genes (aa, AA), their offspring with certainty will have genes Aa.

Application of Markov Chain (3) (Cont.)

In a long-run, gene a will disappear from the population.

```
#creating Markov chain object
library(markovchain)
mc<- new("markovchain", transitionMatrix=tm,
states=c("AA", "Aa", "aa"))

#computing stationary distribution
steadyStates(mc)
```

```
AA  Aa  aa
 1   0   0
```

R CODE

Application of Markov Chain (3) (Cont.)

```
library(expm)
gen1<- c(0.99, 0, 0.01)
gen<- gen1%*%tm
for (n in 2:10) {
print(n)
print(round(gen, digits=10))
gen<- gen%*%tm
}
```

n	p_{AA}	p_{Aa}	p_{aa}
1	0.99	0	0.01
2	0.99	0.01	0
3	0.995	0.005	0
4	0.9975	0.0025	0
5	0.99875	0.00125	0
6	0.999375	0.000625	0
7	0.9996875	0.0003125	0
8	0.9998438	0.00015625	0
9	0.999921875	0.000078125	0
10	0.9999609375	0.0000390625	0

Note that already in the second generation the gene type aa disappears, and is transformed into the hybrid type Aa, and that after as many as ten generations, the gene a is still lingering on in the population.

Application of Markov Chain (4)

In this application, we present yet another chain that is not Markov. It has to do with weather conditions. On an intuitive level, weather tomorrow depends not just on today's weather but on the weather for several past days, if not the entire history of weather conditions in the region. To support this intuitive supposition numerically, we downloaded from kaggle.com an open-access historical hourly weather data for 2012-2017 (file “weather_description.csv”), focused only on the column for Los Angeles, and clumped the weather conditions into the four categories:

```
S = {s="sky clear", c="cloudy", f="fog", r="rain"}.
```

Application of Markov Chain (4) (Cont.)

We then compute empirical conditional probability of clear sky tomorrow, given clear skies yesterday and today,

$$\hat{P}(X_{tomorrow} = s\,|X_{yesterday} = s, X_{today} = s)$$

$$= \frac{\hat{P}(sss)}{\hat{P}(sss) + \hat{P}(ssc) + \hat{P}(ssf) + \hat{P}(ssr)} = 0.9647,$$

and also estimate the conditional probability of clear sky tomorrow given clear sky today,

Application of Markov Chain (4) (Cont.)

$$\hat{P}(X_{tomorrow} = s \,| X_{today} = s) = \frac{\hat{P}(sss) + \hat{P}(css) + \hat{P}(fss) + \hat{P}(rss)}{\hat{P}(sss) + \hat{P}(css) + \cdots + \hat{P}(rsr)} = 0.9356.$$

Since $0.9647 \neq 0.9356$, we conclude that weather conditions are not governed by a Markov chain. R CODE

Application of Markov Chain (4) (Cont.)

However, 0.9647 and 0.9356 are close enough values to hypothesize that statistically speaking weather conditions could constitute a Markov chain. We conduct a chi-squared test to see if it is a Markov chain. We take the first 20% of the data and compute the expected ("theoretical") one-step transition probability matrix. Then we use the remaining 80% of the data to compute an observed transition frequencies and the expected transition frequencies based on the "theoretical" matrix. We then compute a chi-squared statistic $\chi^2 = (obs - exp)^2/exp$, that has a chi-squared distribution with $(4)(3) = 12$ degrees of freedom if the process is a Markov chain.

Application of Markov Chain (4) (Cont.)

Observed Transition Frequencies

	[,1]	[,2]	[,3]	[,4]
[1,]	18819	1143	712	196
[2,]	1056	4995	642	260
[3,]	755	582	4474	231
[4,]	241	233	214	1649

Expected Transition Frequencies

	[,1]	[,2]	[,3]	[,4]
[1,]	18633.9	1193.4	915.8	126.9
[2,]	1013.4	5104.0	570.5	265.1
[3,]	1013.5	740.6	4088.6	199.2
[4,]	205.8	580.6	367.5	1183.2

chi-squared statistic = 703.055

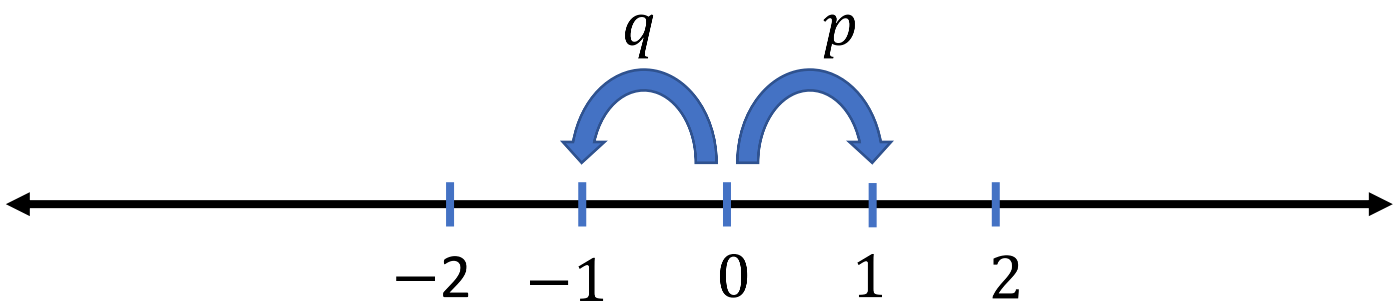
p-value = 9.780797e-143 < 0.05

Conclusion: not a Markov chain

R CODE

RANDOM WALK

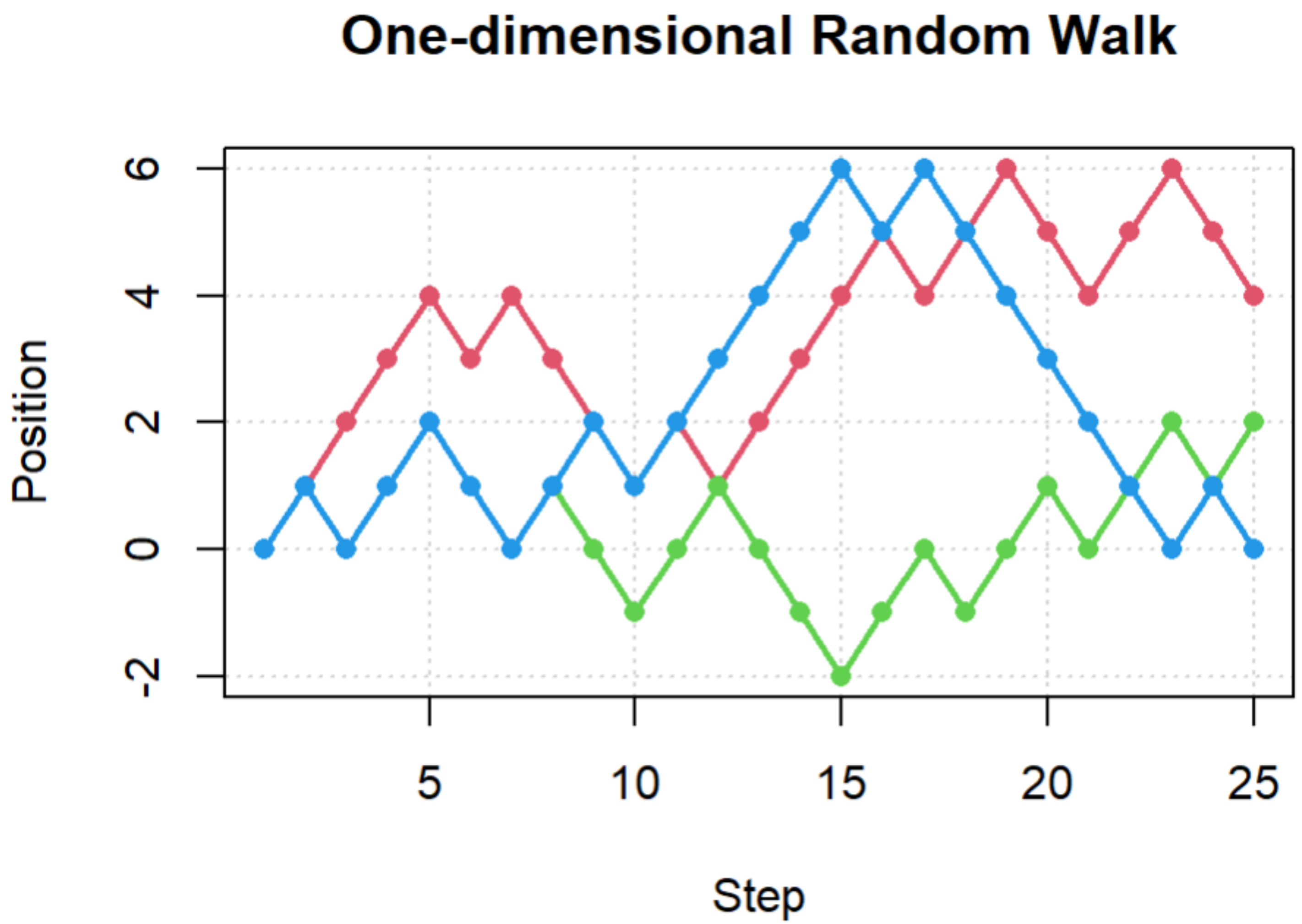
A ***simple one-dimensional random walk*** (RW) is a Markov chain that starts at 0, and allowed transitions are $+1$ with probability p and -1 with probability $q = 1 - p$. So, the state space contains all integers (positive or negative) on a real line.



A simple random walk is called ***symmetric*** if $p = q = 1/2$.

Simulated Trajectories of Random Walk

R CODE

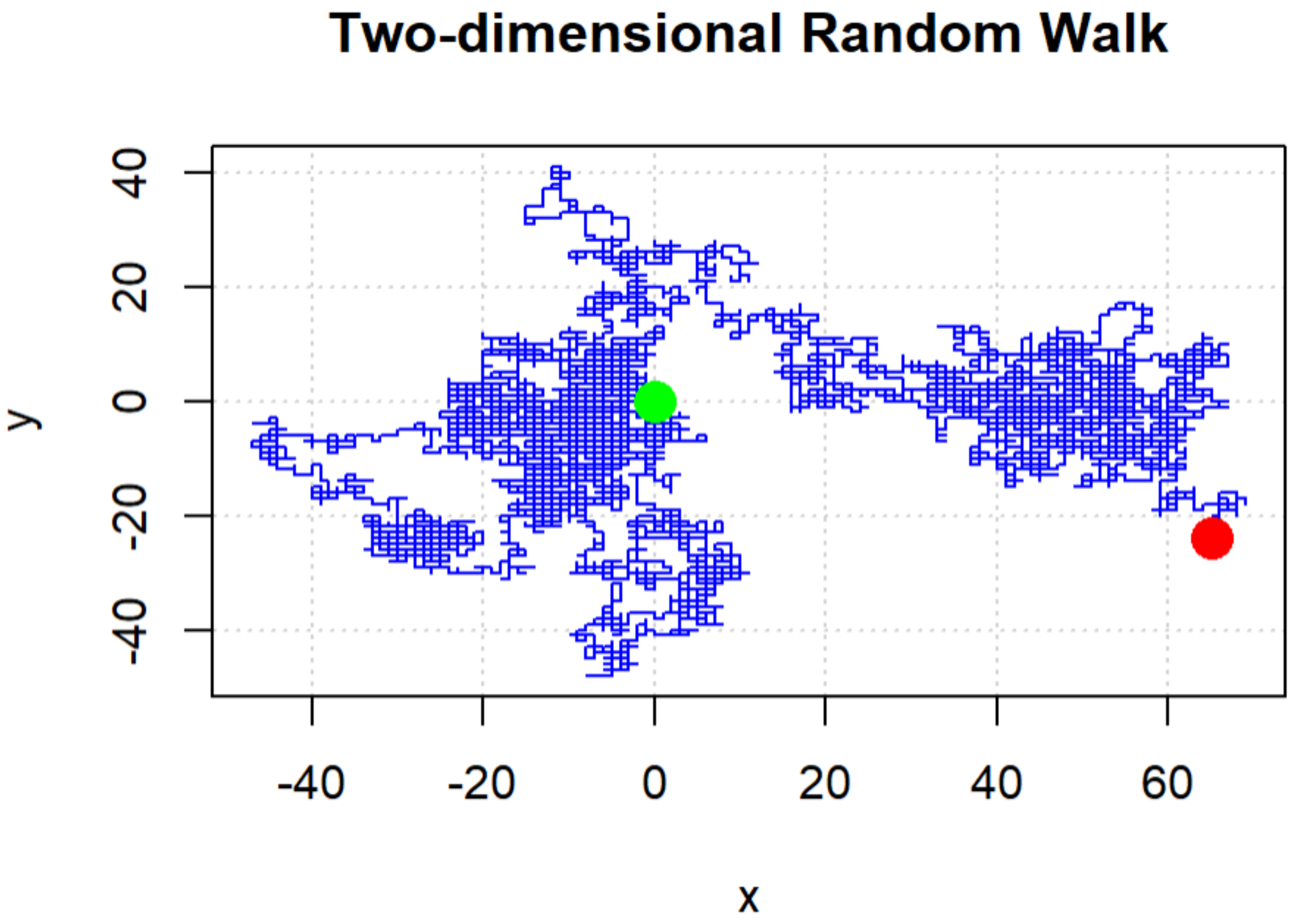


Two-dimensional Random Walk

Definition. A two-dimensional random walk can be defined to move up, down, left, or right only, or diagonal movements may be also allowed.

A ***symmetric two-dimensional random walk*** starts at $(0,0)$, and is equiprobable to move up, down, left, right $(p = 1/4)$.

R CODE

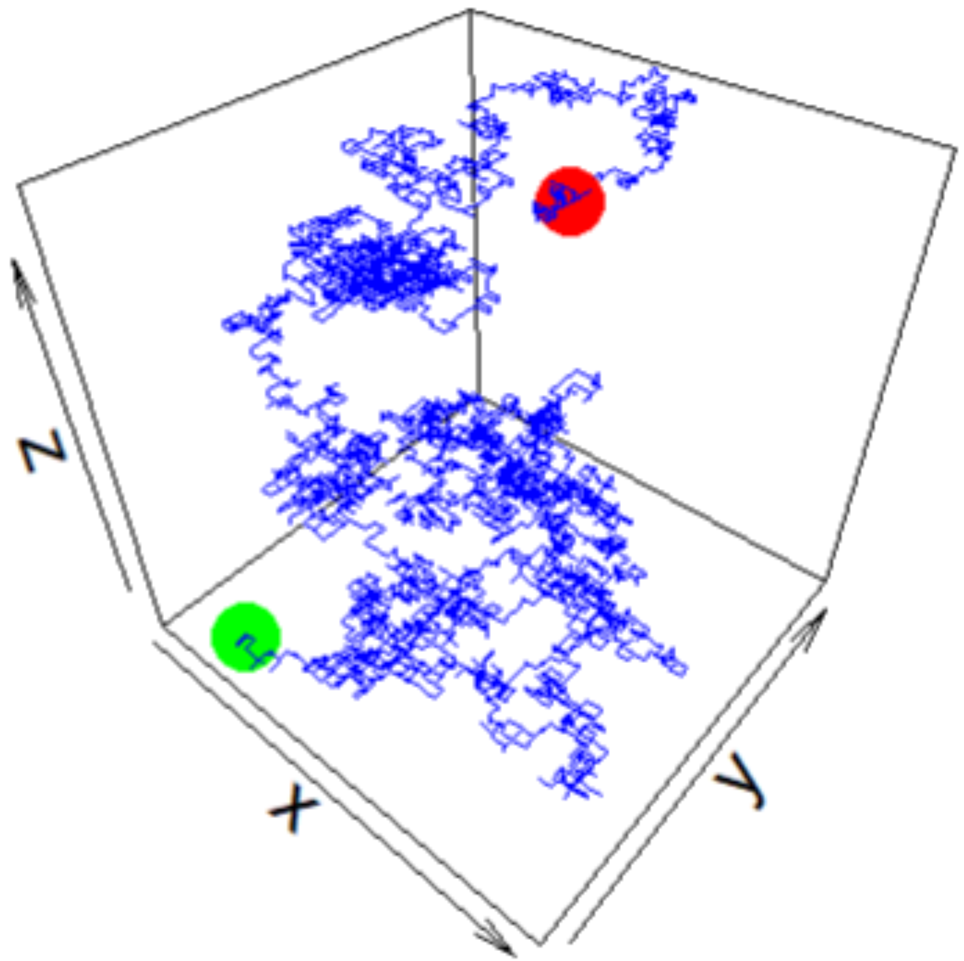


Three-dimensional Random Walk

<u>Definition.</u> A ***symmetric three-dimensional random walk*** starts at (0,0), and moves on three-dimensional integer lattice, with equiprobable moves in six directions ($p = 1/6$).

Three-dimensional Random Walk

R CODE



Must-know Facts About Random Walk

- A symmetric 1D or 2D RW will come back to the origin infinitely many times (is ***recurrent***), whereas an asymmetric 1D or 2D RW will come back only a finite number of times and will eventually wander away (is ***transient***). That is, a symmetric RW is recurrent, and an asymmetric RW is transient.
- Any random walk (symmetric or not) in 3D or a higher dimension is transient.
- For example, a bug crawling randomly along a railroad track will eventually return. A drunk man will find his way home, but a drunk bird will be lost forever.

Application of Random Walk

The most famous application of RW is called ***Gambler's Ruin Problem***. Its version has been formulated as early as 1656, in correspondence between Blaise Pascal and Pierre de Fermat.

Suppose a gambler starts with a fortune of $\$i$ and will move up $\$1$ with probability p or down $\$1$ with probability $q = 1 - p$ until he is either broke or reaches the fortune of $\$N$. What is the probability that he goes broke?

In other words, what's the probability to reach 0 before N, if the RW starts in i?

Gambler's Ruin Problem (cont.)

Denote by P_j the probability of ruin if the RW is in state j. Conditioning on the outcome of the first move, we can write the recurrence relation:

$$P_j = pP_{j+1} + qP_{j-1}$$

with the boundary conditions $P_0 = 1$ and $P_N = 0$. The solution is

$$P_i = \begin{cases} \frac{N-i}{N}, if\ p = q = 1/2, \\ \frac{(q/p)^i - (q/p)^N}{1-(q/p)^N}, \quad if\ p \neq q. \end{cases}$$

Gambler's Ruin Problem (cont.)

Also, it can be shown similarly that the expected number of games that the gambler plays until he reaches $\$N$ or goes bankrupt is

$$E_i = \begin{cases} i(N-i), & if\ p = q = 1/2, \\ \dfrac{i - N(1-P_i)}{q-p}, & if\ p \neq q. \end{cases}$$

Gambler's Ruin Problem (cont.)

Let us see how it plays out with some specific values. Suppose the gambler started with $\$50$, goes up $\$5$ with probability 0.55, or goes down $\$5$ with probability 0.45. He would end up with either $\$100$ or $\$0$. Since the increment is $\$5$, in our notation this translates into $i = 10$, $N = 20$, $p = 0.55$, and $q = 0.45$. The probability of ruin is $P_{10} = \frac{(0.45/0.55)^{10} - (0.45/0.55)^{20}}{1-(0.45/0.55)^{20}} = 0.1185.$ And it takes on average $E_{10} = \frac{10-(20)(1-0.1185)}{0.45-0.55}$ $= 76.3$ games. We can verify these values empirically by simulating one million trajectories and counting the proportion of those that ended in 0 and computing an average lengths of the trajectories.

Gambler's Ruin Problem (cont.)

- Proportion of trajectories that end in 0

  0.118353

- Mean number of games

  76.32965

R CODE

Gambler’s Ruin Problem Exercise

Suppose the gambler started with $\$50$, goes up $\$5$ with probability 0.52, or goes down $\$5$ with probability 0.48. Assume the game ends when the gambler reaches $120 or $0.

1. Compute theoretically the probability of ruin.
2. Compute theoretically the expected number of games.
3. Estimate empirically the probability of ruin.
4. Estimate empirically the expected number of games.

Gambler's Ruin Problem Exercise Solution

In our notation, $i = 10$, $N = 24$, $p = 0.52$, and $q = 0.48$.

1. Compute theoretically the probability of ruin.

$$P_{10} = \frac{(0.48/0.52)^{10} - (0.48/0.52)^{24}}{1 - (0.48/0.52)^{24}} = 0.354616.$$

2. Compute theoretically the expected number of games.

$$E_{10} = \frac{10 - (24)(1 - 0.354616)}{0.48 - 0.52} = 137.2304 \text{ games.}$$

Gambler's Ruin Problem Exercise

3. Estimate empirically the probability of ruin.

   0.354777

R CODE

4. Estimate empirically the expected number of games.

   136.9053

POISSON PROCESS

A stochastic process $\{N(t), t \geq 0\}$ is called a ***Poisson process*** if it counts the total number of events occurring by time t, $N(0) = 0$, and $N(t)$ has a Poisson distribution with rate λt, that is,

$$P(N(t) = n) = \frac{(\lambda t)^n}{n!}\ e^{-\lambda t}, n = 0, 1, 2, \dots,$$

where $\lambda t = E\big(N(t)\big) = Var\big(N(t)\big)$.

Historical Note on Poisson Distribution and Process

- Siméon Poisson (1781 – 1840) was a French mathematician and physicist.
- He is the one who introduced Poisson Distribution.
- The reference to a Poisson process first appeared in two independent publications in 1940.

POISSON PROCESS (CONT.)

Facts: A Poisson process as a special case of a Markov chain. The state space of a Poisson process is $S = \{0, 1, 2, 3, \dots\}$. The process jumps from initial state 0 to state 1, then to state 2, etc. The jumps are always of size 1, and transitions between states are allowed only in the direction of increase. Moreover, a Poisson process includes the time component. It matters how long the process halts between jumps. This type of Markov chain is called a ***continuous-time Markov chain.***

POISSON PROCESS (CONT.)

Definition. An ***interarrival time*** is the time between two consecutive occurrences of events.

Fact: Interarrival times are independent exponentially distributed random variables with the density function $f(t) = \lambda \exp(-\lambda t), t \geq 0$. Note that the mean is $1/\lambda$.

Example: Events occur according to a Poisson process with rate $\lambda = 2$ per minute. Then, on average, one needs to wait for $1/\lambda \ = 1/2$ of a minute for another occurrence.

Fact: In a Poisson process, two events cannot occur at the same time since the interarrival times are exponential and so the probability of exact equality to zero is zero.

EXAMPLES OF POISSON PROCESSES

A Poisson process is used to model occurrences of **rare** events. Here are some instances of Poisson processes: the number of people who enter a store or a bank or a restaurant or a gym or a National Park, the number of cars that pass a certain intersection, the number of auto accidents on a certain stretch of a freeway, the number of births in a hospital, the number of meteors in the night sky, or the number of phone calls to a credit card customer service. Typically, natural disasters occur according to a Poisson process: earthquakes, volcano eruptions, wildfires, etc. Of course, in all the above examples, the considered time period should be short enough for the rate of occurrence to be constant.

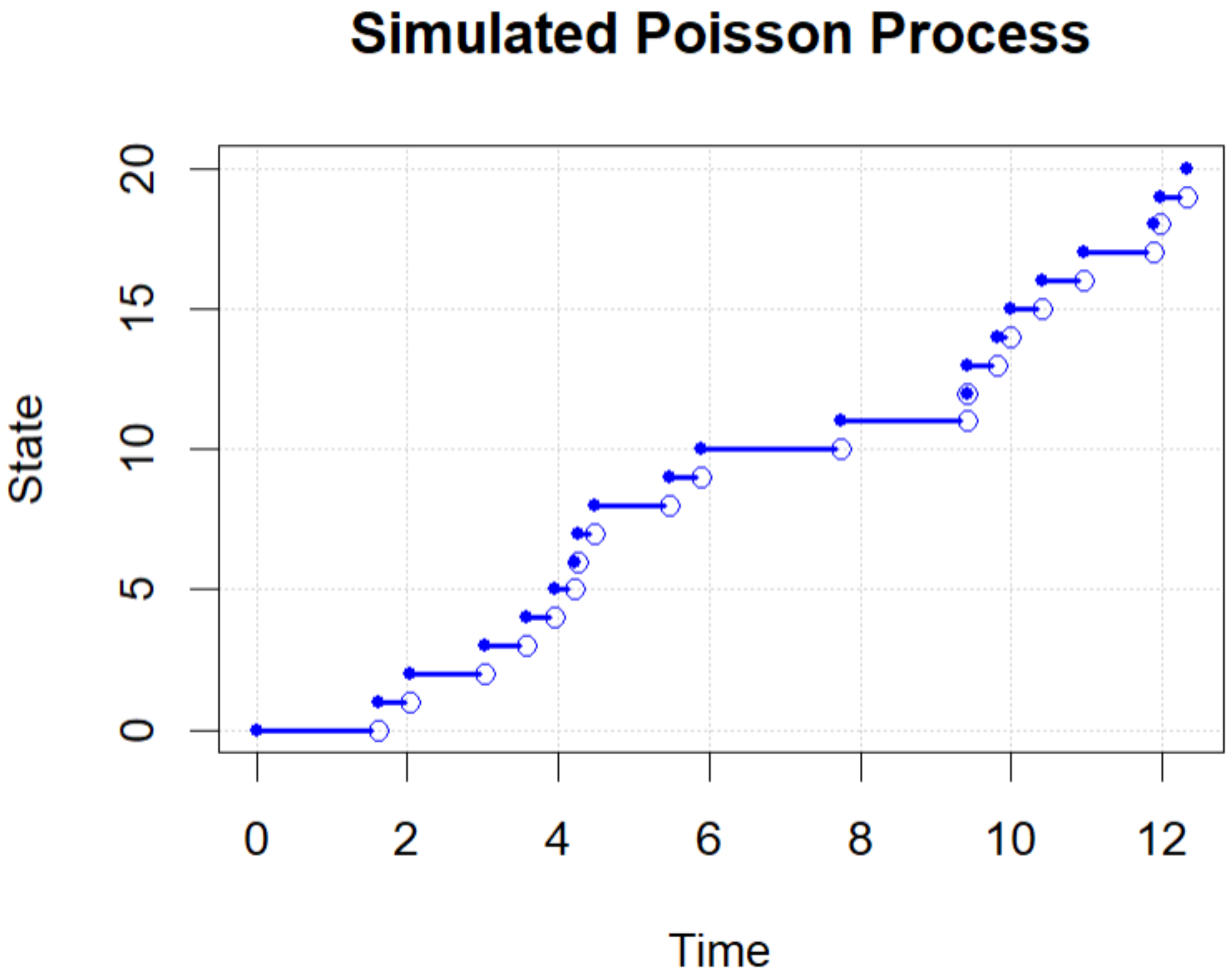
EXAMPLES OF NON-POISSON PROCESSES

Some examples of processes that clearly are not governed by a Poisson law are events that happen according to a schedule, for example, arrival of buses along a certain route, road closures due to construction work, quarry blasts, and building demolitions. Also, events that happen in a competing market, where two rival companies might be scheduling two events at the same time. For instance, two pharmaceutical companies might simultaneously bring to the market two cardiac medications, or two production companies might release two movies on the same day. In addition, some periodic (or seasonal) natural phenomena cannot be modeled as a Poisson process, i.e., eruptions of Old Faithful geyser in Yellowstone National Park, ocean tides, the appearance of sunspots, etc.

SIMULATION OF POISSON PROCESS

To simulate a trajectory of a Poisson process, we generate exponentially distributed interarrival times. And then plot piece-wise constant function, starting at zero and increasing jumps by one.

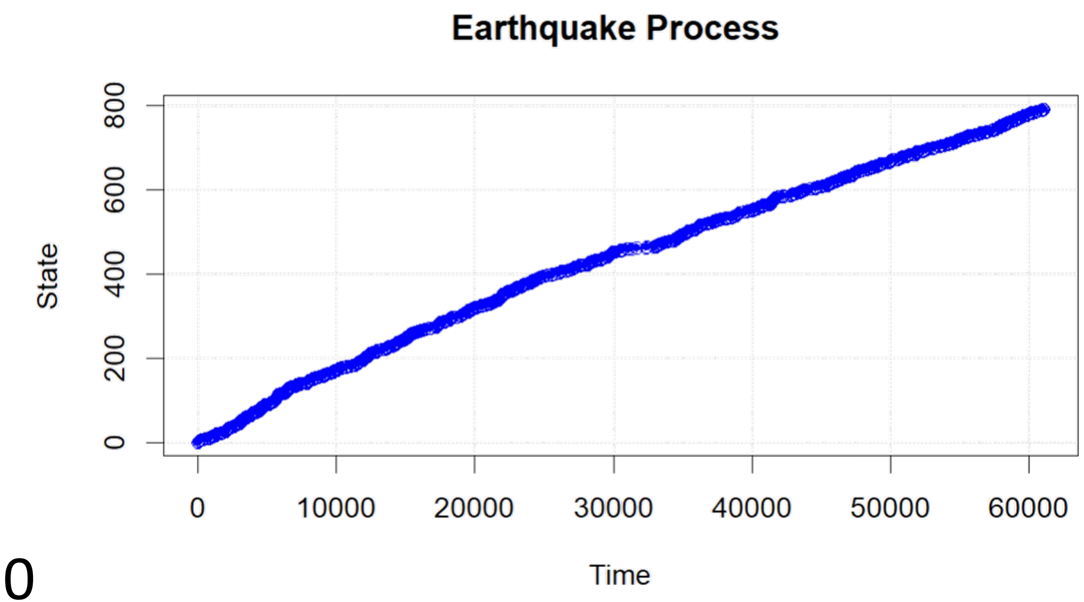
R CODE



APPLICATION OF POISSON PROCESS (1)

In seismology, occurrence of earthquakes is often modeled according to a Poisson process.

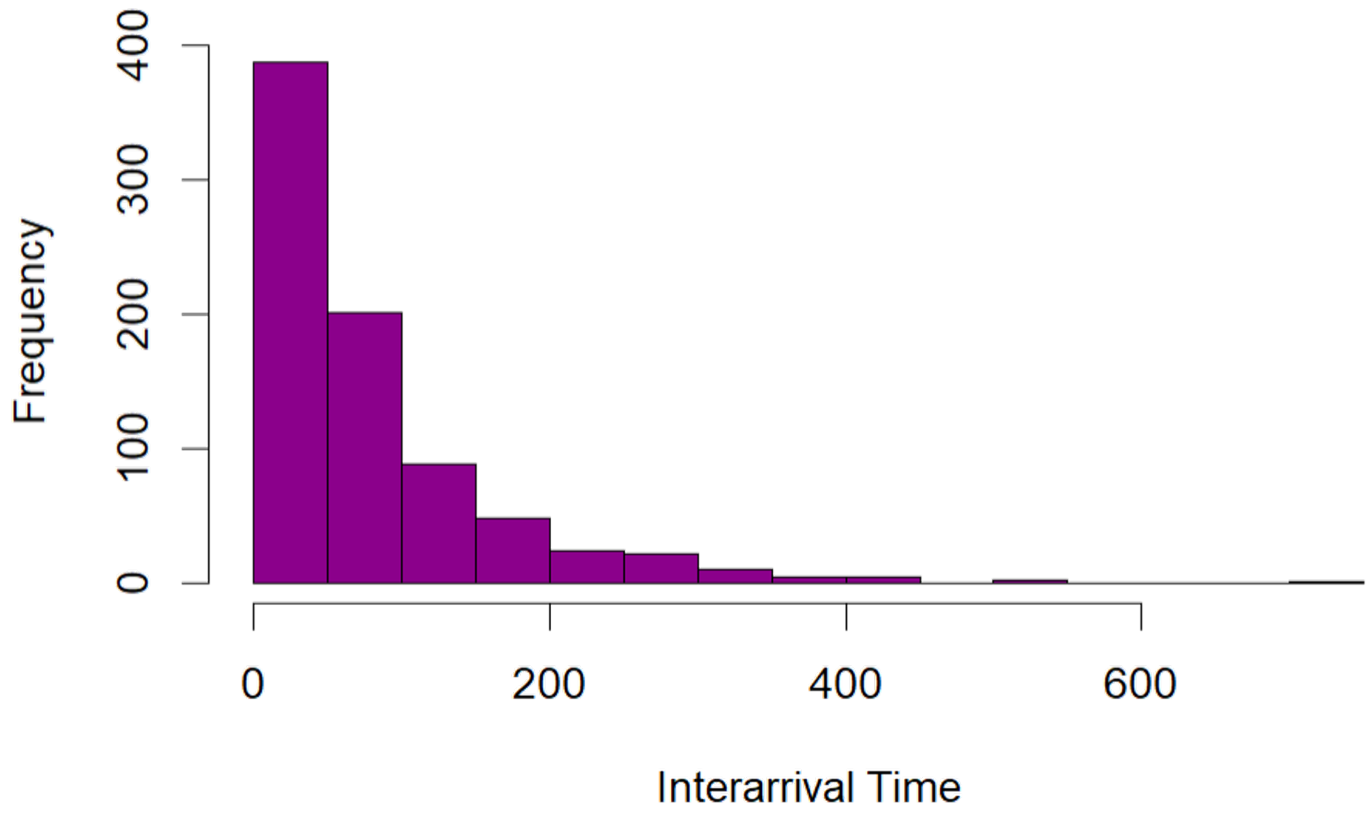
We obtain the data from the Southern California Earthquake Data Center's website ***https://service.scedc.caltech.edu/eq-catalogs/date_mag_loc.php***. The data are on earthquakes in Southern California with a minimum magnitude of 3.0 that occurred between 2012 and 2018. We remove those earthquakes that were registered within three hours of their predecessors (possibly aftershocks).



DATA SET

APPLICATION OF POISSON PROCESS (1) (CONT.)

To see if occurrence of earthquakes is a Poisson process or not, we compute the lengths of the interarrival times and check if they are exponentially distributed.



R CODE

APPLICATION OF POISSON PROCESS (1) (CONT.)

Observed Frequencies

1	2	3	4	5	6	7
342	178	117	49	39	24	42

Expected Frequencies

319.8   190.5   113.5   67.6   40.3   24.0   35.4

Chi-squared statistic

8.883823

P-value

0.1137888 > 0.05, so the interarrival times are exponentially distributed and we conclude that the process is Poisson.

POISSON PROCESS: EXERCISE (1)

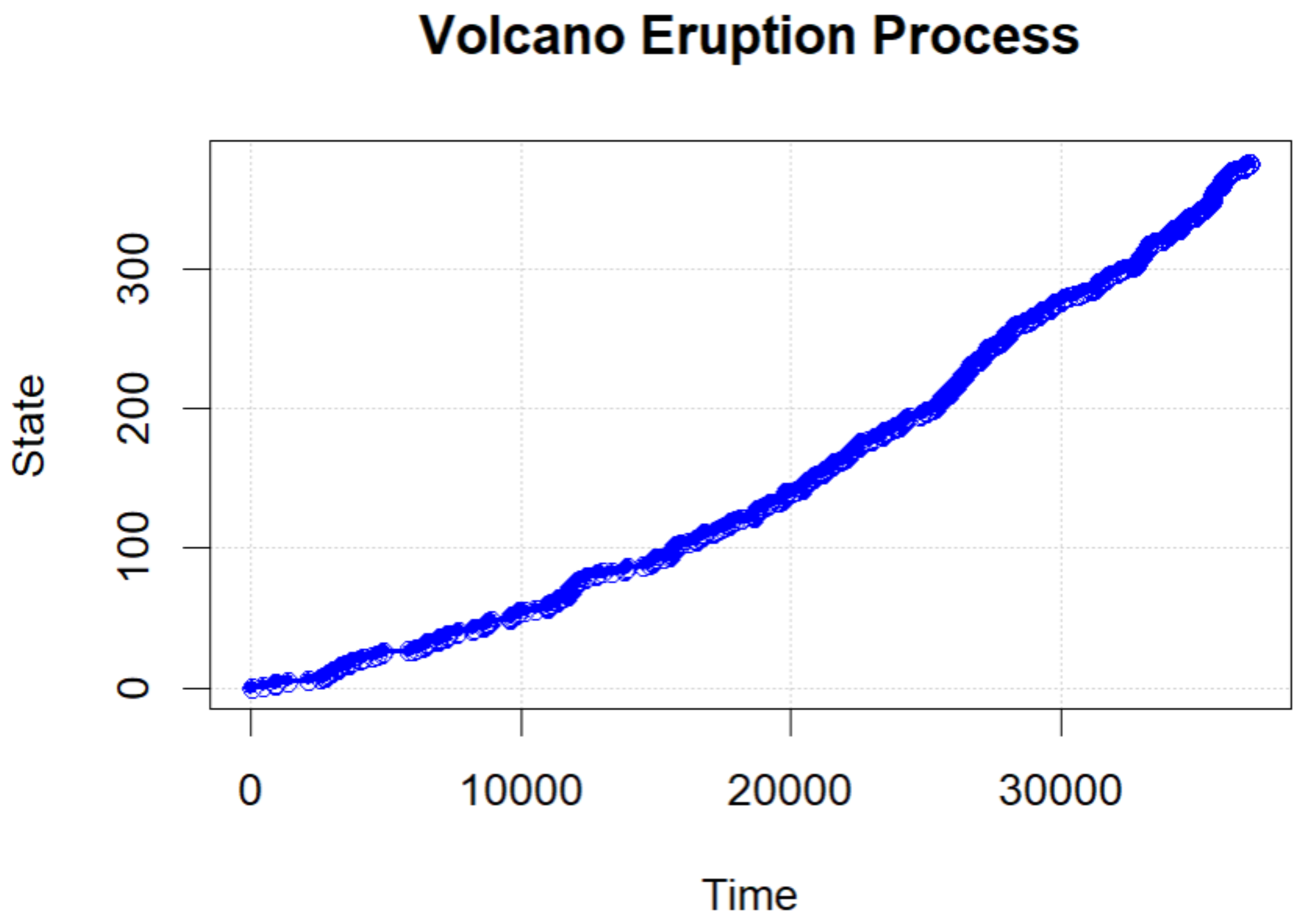
The National Geophysical Data Center's website

https://www.ngdc.noaa.gov/hazel/view/hazards/volcano/event-search/

provides access to the Global Significant Volcanic Eruptions Database. Use the data set “volcanoesdata.csv” to verify that volcanic eruptions in the past 100 years can be modeled as a Poisson process.

DATA SET

POISSON PROCESS: EXERCISE (1) SOLUTION



R CODE

POISSON PROCESS: EXERCISE (1) SOLUTION (CONT.)

Observed Frequencies

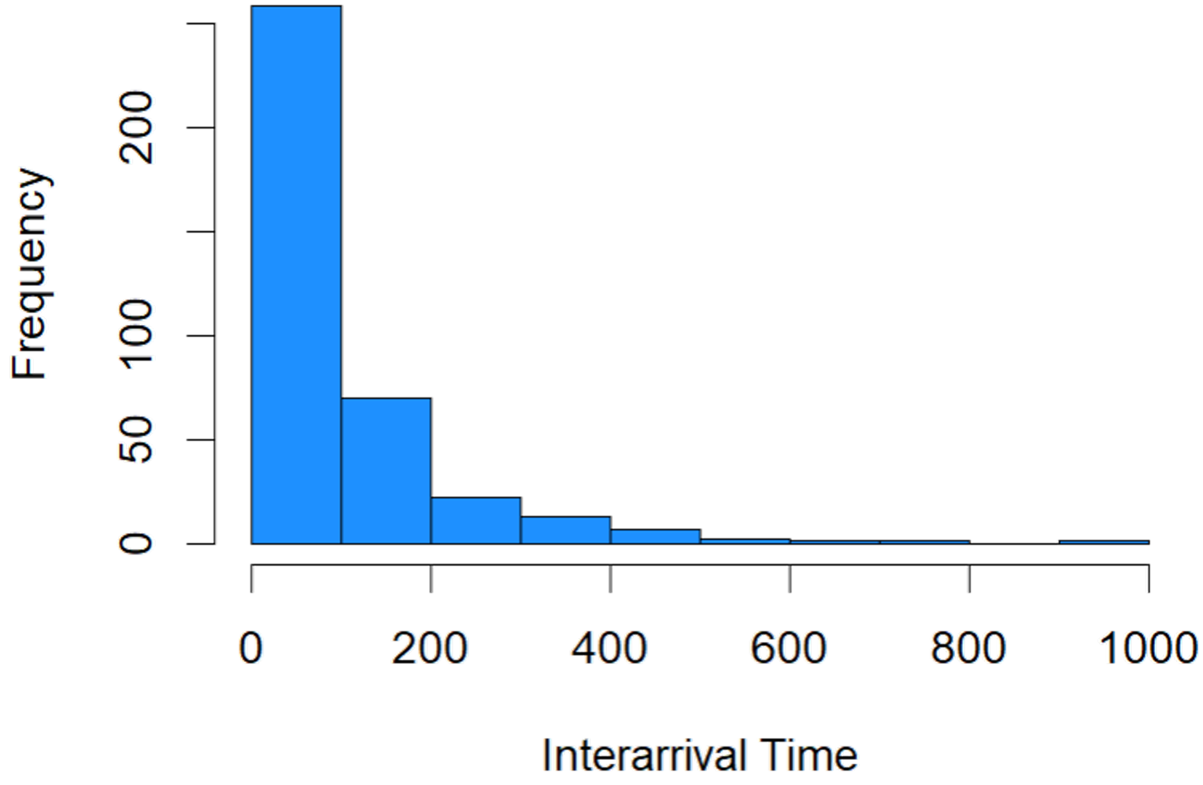
1	2	3	4	5	6	7
96	66	95	43	29	13	34

Expected Frequencies

84.3	65.4	90.1	54.2	32.6	19.6	29.7

Chi-squared statistic

7.492589



P-value

0.1865064 > 0.05, the interarrival times are exponentially distributed and we conclude that the process is Poisson.

APPLICATION OF POISSON PROCESS (2)

In sports analytics, a Poisson process is used to model the process of goal scoring in a game. Consider a team game where players score only one point at a time, for instance, ice hockey. Suppose the points scored by team A follow a Poisson process $\{N_A(t),\ t \geq 0\}$ with rate λ_A, and points scored by team B are governed by a Poisson process $\{N_B(t),\ t \geq 0\}$ with parameter λ_B. Assuming the two processes are independent, we can derive some interesting results.

APPLICATION OF POISSON PROCESS (2) (CONT.)

1. We can find the probability that one team scores ahead of the other team. Denote by T_A and T_B the respective interarrival times. We know that T_A and T_B are independent and exponentially distributed with parameters λ_A and λ_B, respectively. We write $P(team\ A\ scores\ before\ team\ B) = P(T_A < T_B) = \int_0^\infty e^{-\lambda_B t}\lambda_A e^{-\lambda_A t}dt = \frac{\lambda_A}{\lambda_A+\lambda_B}$. Now switching λ_A and λ_B, we get

   $P(team\ B\ scores\ before\ team\ A) = P(T_B < T_A) = \frac{\lambda_B}{\lambda_A+\lambda_B}$.

APPLICATION OF POISSON PROCESS (2) (CONT.)

To see how it works with numbers, suppose team A scores, on average, every 10 minutes, and team B scores every 12 minutes, on average. Then $\lambda_A = \frac{1}{10} = 0.1$ goal per minute, and $\lambda_B = \frac{1}{12} = 0.083$ goals per minute. Thus,

$$P(team\ A\ scores\ before\ team\ B) = \frac{\lambda_A}{\lambda_A + \lambda_B} = \frac{1/10}{\frac{1}{10} + \frac{1}{12}} = 0.545,$$

and

$$P(team\ A\ scores\ before\ team\ B) = 1 - 0.545 = 0.455.$$

APPLICATION OF POISSON PROCESS (2) (CONT.)

2. We can find the probability of a tie at the end of the game, and also the

probability that team A (team B) wins. Let T denote the length of the game. Let $\{N_A(t), t \geq 0\}$ and $\{N_B(t), t \geq 0\}$ be the Poisson processes of scoring for teams A and B, respectively. Then, we can write

$$P(game\ is\ tied) = P\big(N_A(T) = N_B(T)\big) = \sum_{n=0}^{\infty} P(N_A(T) = n, N_B(T) = n)$$

$$= \sum_{n=0}^{\infty} \frac{(\lambda_A T)^n}{n!} e^{-\lambda_A T} \frac{(\lambda_B T)^n}{n!} e^{-\lambda_B T} = e^{-(\lambda_A + \lambda_B)T} \sum_{n=0}^{\infty} \frac{(\lambda_A \lambda_B T^2)^n}{(n!)^2} \ ,$$

APPLICATION OF POISSON PROCESS (2) (CONT.)

$$P(team\ A\ wins) = P\big(N_A(T) > N_B(T)\big) = \sum_{n=0}^{\infty}\sum_{k=1}^{\infty} P(N_A(T) = n+k, N_B(T) = n)$$

$$= \sum_{n=0}^{\infty}\sum_{k=1}^{\infty} \frac{(\lambda_A T)^{n+k}}{(n+k)!} e^{-\lambda_A T} \frac{(\lambda_B T)^n}{n!} e^{-\lambda_B T} = e^{-(\lambda_A+\lambda_B)T} \sum_{n=0}^{\infty}\left[\frac{(\lambda_A\lambda_B T^2)^n}{n!}\sum_{k=1}^{\infty}\frac{(\lambda_A T)^k}{(n+k)!}\right],$$

and

$$P(team\ B\ wins) = P\big(N_A(T) < N_B(T)\big) = 1 - P(game\ is\ tied) - P(team\ A\ wins).$$

APPLICATION OF POISSON PROCESS (2) (CONT.)

In our example, ice hockey game lasts for $T = 60$ minutes. We compute

$$P(game\ is\ tied) = e^{-(\lambda_A+\lambda_B)T} \sum_{n=0}^{\infty} \frac{(\lambda_A \lambda_B T^2)^n}{(n!)^2}$$

R CODE

$$= e^{-(\frac{1}{10}+\frac{1}{12})(60)} \sum_{n=0}^{\infty} \frac{((\frac{1}{10})(\frac{1}{12})\,(60)^2)^n}{(n!)^2} = e^{-11} \sum_{n=0}^{\infty} \frac{30^n}{(n!)^2} = 0.1166,$$

$P(team\ A\ wins) = e^{-11} \sum_{n=0}^{\infty} \left[\frac{30^n}{n!} \sum_{k=1}^{\infty} \frac{6^k}{(n+k)!}\right] = 0.5590$, and

- $P(team\ B\ wins) = 1 - 0.1166 - 0.5590 = 0.3244.$

BIRTH-AND-DEATH PROCESS

Definition. A continuous-time Markov chain $\{X(t), t \geq 0\}$ with state space $S = \{0, 1, 2, \dots\}$ is called a ***birth-and-death process*** if, given that the chain is in state n, the time to transition to state $n + 1$ (***birth*** or ***arrival***) is exponentially distributed with mean $1/\lambda_n$, and the time to transition to state $n - 1$ (***death*** or ***departure***) is exponentially distributed with mean $1/\mu_n$. The two waiting times are independent. Simply put, births occur as a Poisson process, and deaths occur as a Poisson process and the two processes are independent.

BIRTH-AND-DEATH PROCESS (CONT.)

Example. A Poisson process is an example of a birth-and-death process with $\lambda_n = \lambda$ and $\mu_n = 0,\ n = 0, 1, 2, \ldots$. Because there are no deaths, it is a ***pure birth*** process. The probability that there are n particles in the system at time t is

$$P_n(t) = \frac{(\lambda t)^n}{n!}\, e^{-\lambda t}, n = 0,1,2, \ldots, t \geq 0.$$

BIRTH-AND-DEATH PROCESS (CONT.)

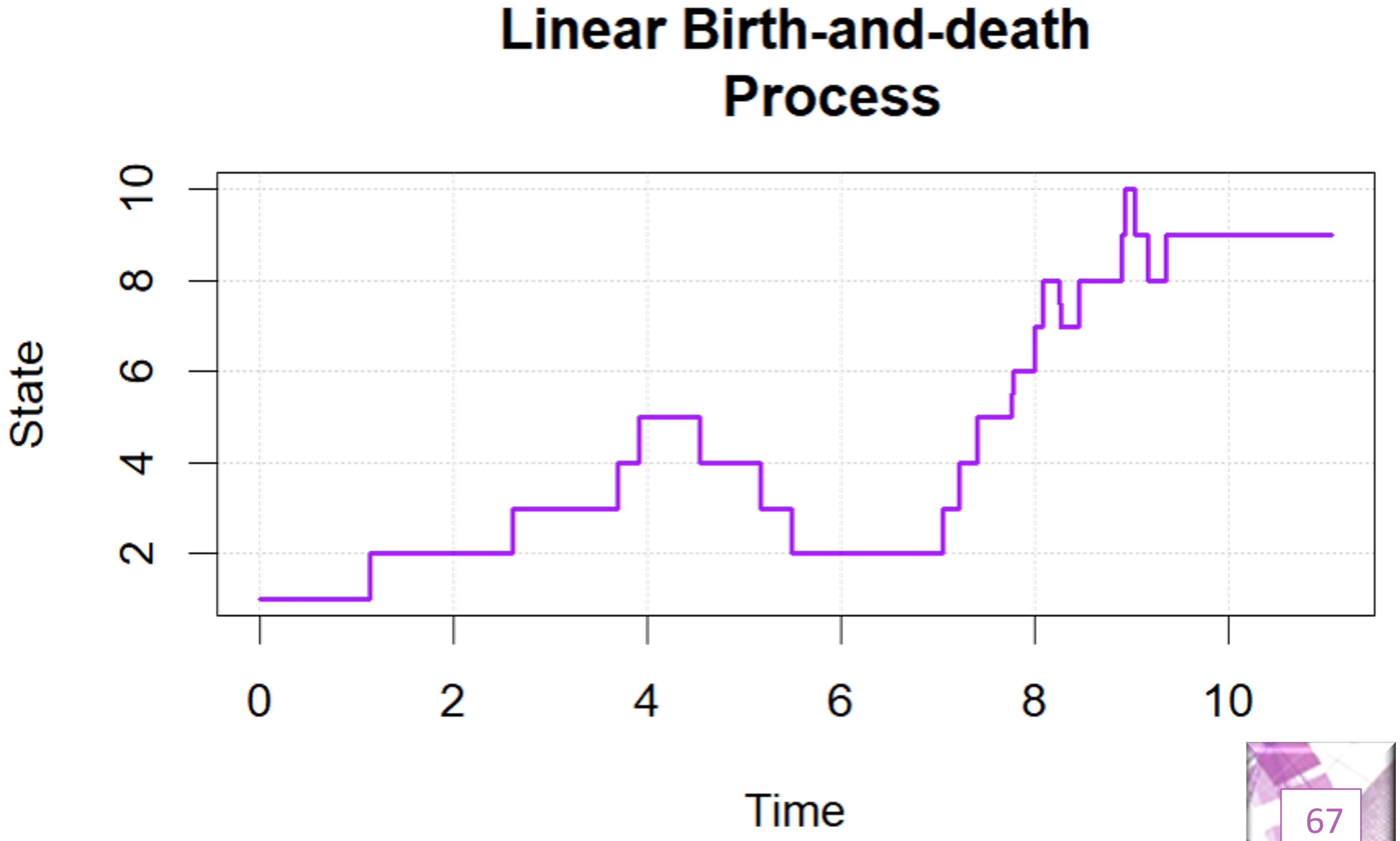
Example. A ***linear birth-and-death process*** is a birth-and-death process with parameters

$\lambda_n = \lambda n$ and $\mu_n = \mu n, \quad n = 0,1,2, \dots$. Simply put, there are n particles in the system, and each of them independently of the others can give birth to one particle or die. The probability that there are n particles in the system at time t is

$$P_n(t) = (1 - P_0)\left(1 - \frac{\lambda}{\mu} P_0\right)\left(\frac{\lambda}{\mu} P_0\right)^{n-1} n = 1,2, \dots, \text{where } P_0 = \frac{\mu e^{(\lambda-\mu)t} - \mu}{\lambda e^{(\lambda-\mu)t} - \mu} .$$

SIMULATION OF LINEAR BIRTH-AND-DEATH PROCESS

We simulate a trajectory of the linear birth-and-death process with parameters $\lambda = 0.3$ and $\mu = 0.1$.



R CODE

APPLICATION OF BIRTH-AND-DEATH PROCESS (1)

<u>Definition.</u> An **M/M/1 queue** is a birth-and-death process with $\lambda_n = \lambda$ and $\mu_n = \mu, n = 0,1,2,3,\ldots$. We assume $\lambda < \mu$, meaning that deaths occur more frequently than births, or, otherwise, there is an accumulation of particles in the system. In this process, customers join a queue (representing births) at independent exponential times with parameter λ, and leave the system (representing deaths) after going through the service, which time is exponentially distributed with parameter μ. All customers act independently. In the name of the process, M/M/1, "M"s stand for Markovian arrival and departure processes and "1" means a single server.

APPLICATION OF BIRTH-AND-DEATH PROCESS (1) (CONT.)

Examples. An example of M/M/1 queue is the number of customers in a bank with a single bank teller: customers enter the bank and join the line to the bank teller, then when it is their turn, they get a service from the teller and leave the bank. Another example is the number of broken cars in a repair shop with a single repairman: broken cars are added to the line to get a service from the repairman, and leave the shop once they are repaired.

APPLICATION OF BIRTH-AND-DEATH PROCESS (1) (CONT.)

Queueing theory is a well-developed area of stochastic processes. Of interest are:

- The ***limiting*** (or ***steady-state***) probabilities that there will be n customers in the system $P_n = \lim_{t\to\infty} P_n(t) = \left(1-\frac{\lambda}{\mu}\right)\left(\frac{\lambda}{\mu}\right)^n, n = 0,1,2, \ldots.$

- The ***average number of customers in the system in a long run*** is $\frac{\lambda}{\mu-\lambda}$.

APPLICATION OF BIRTH-AND-DEATH PROCESS (1) (CONT.)

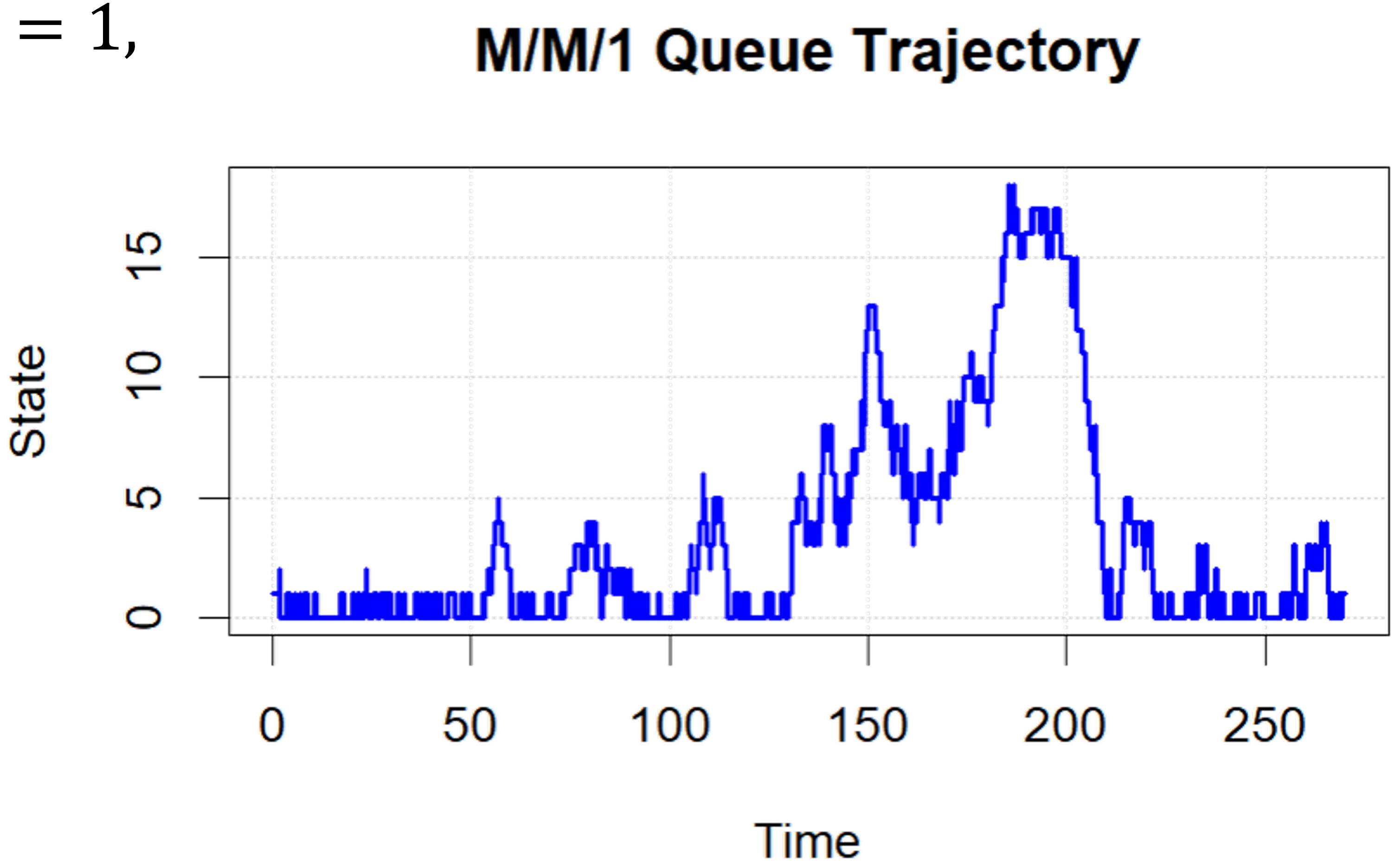
- To see how it works with numbers, consider M/M/1 system in which arrivals occur with the rate $\lambda = 1$ per minute, and departures occurs with the rate $\mu = 1.5$ per minute. The steady-state probabilities are $P_n = \left(1 - \frac{\lambda}{\mu}\right)\left(\frac{\lambda}{\mu}\right)^n = \left(1 - \frac{1}{1.5}\right)\left(\frac{1}{1.5}\right)^n = \left(\frac{1}{3}\right)\left(\frac{2}{3}\right)^n, n = 0,1,2, \dots .$

- In the long run, there will be, on average, $\frac{\lambda}{\mu-\lambda} = \frac{1}{1.5-1} = 2$ customers in the system at any given time.

APPLICATION OF BIRTH-AND-DEATH PROCESS (1) (CONT.)

- We plot a trajectory with $\lambda = 1$, $\mu = 1.5$, and 500 jumps.
- We simulate 50,000 jumps and compute the mean number of customers **1.823064**

[R CODE]



M/M/1 QUEUE: EXERCISE

Simulate a trajectory of an M/M/1 queue with 500 jumps for:

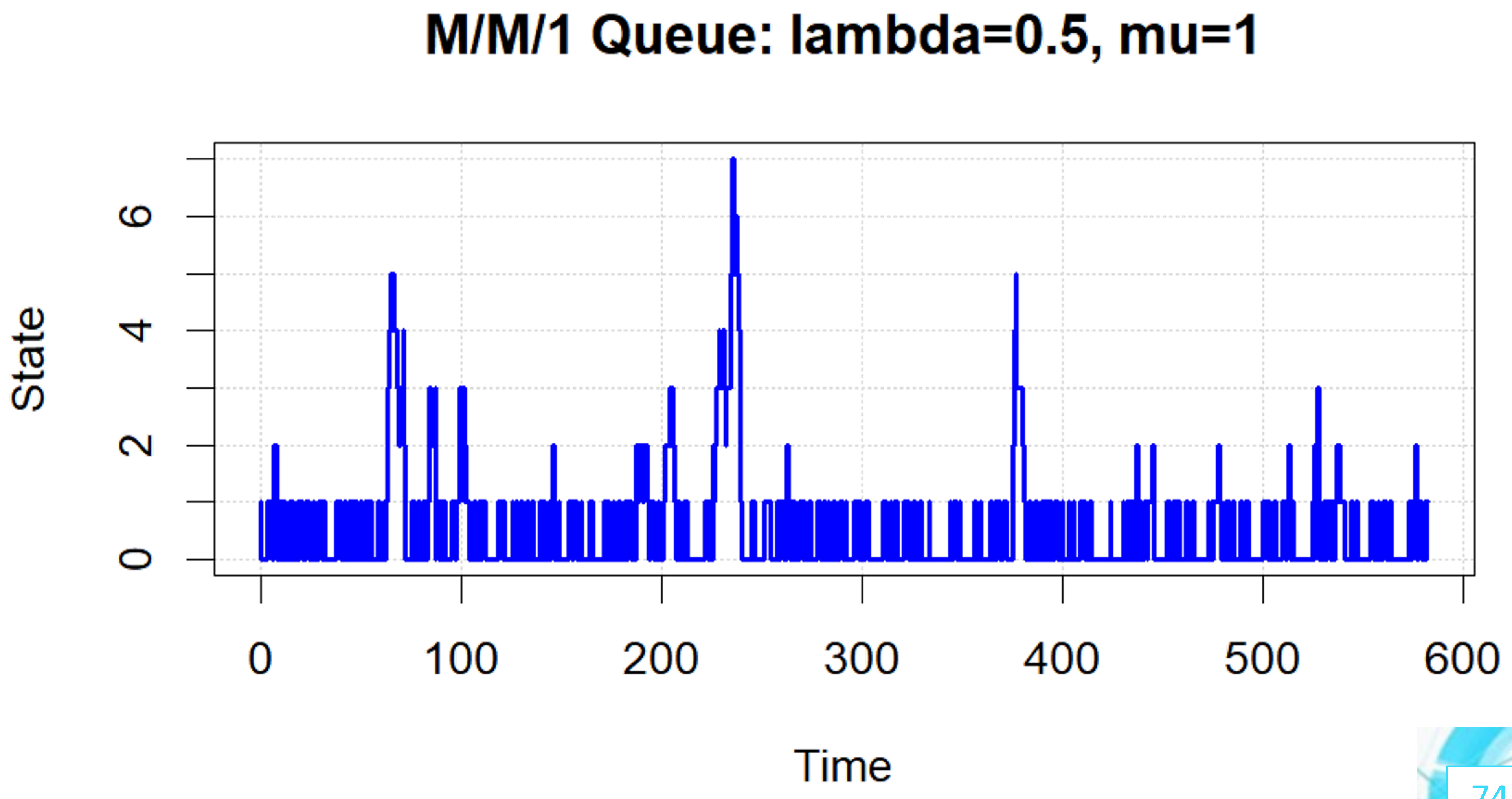
- $\lambda = 0.5$ and $\mu = 1$,
- $\lambda = 1$ and $\mu = 1$,

and

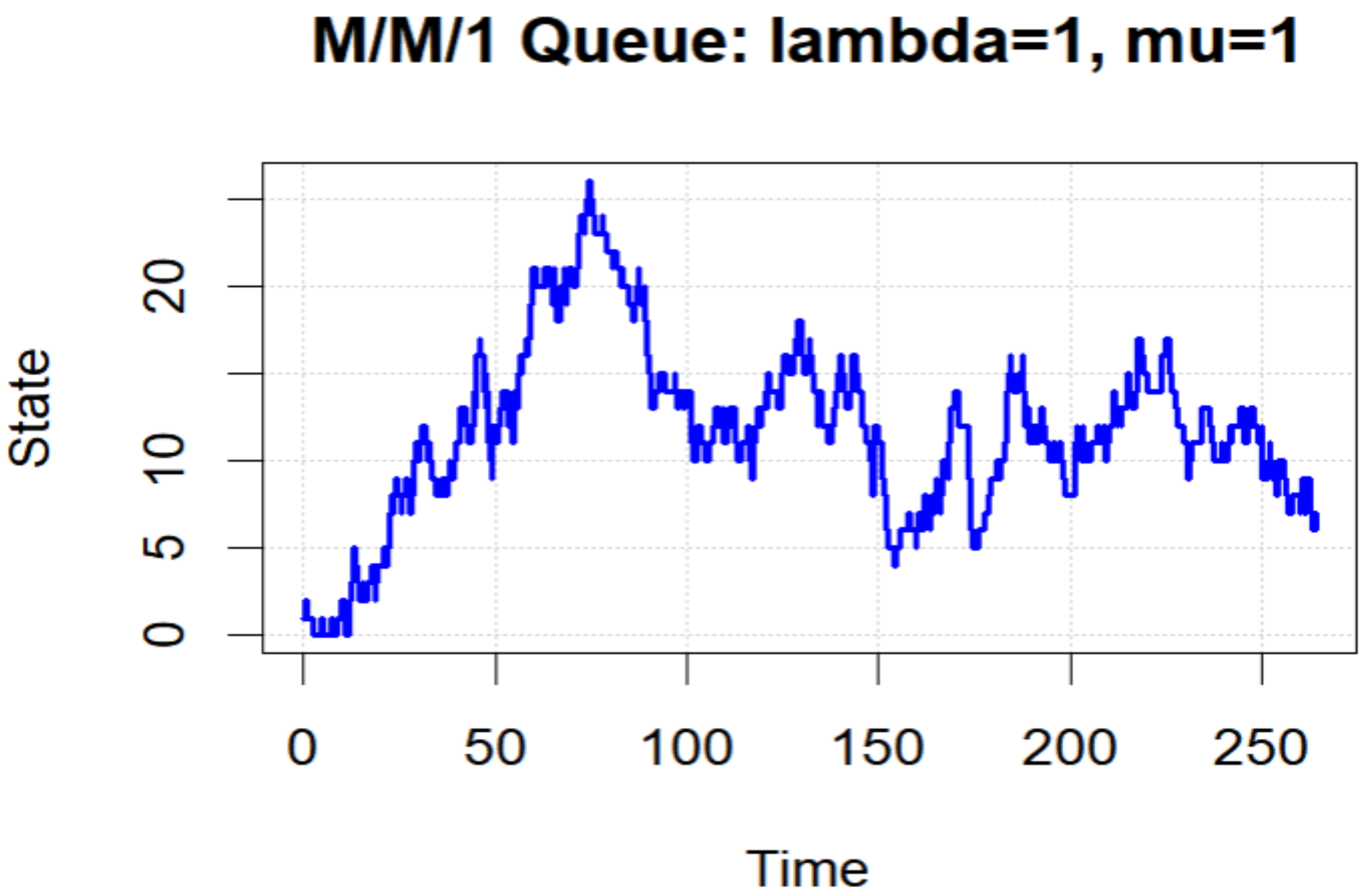
- $\lambda = 1.5$ and $\mu = 1$.

M/M/1 QUEUE: EXERCISE SOLUTION

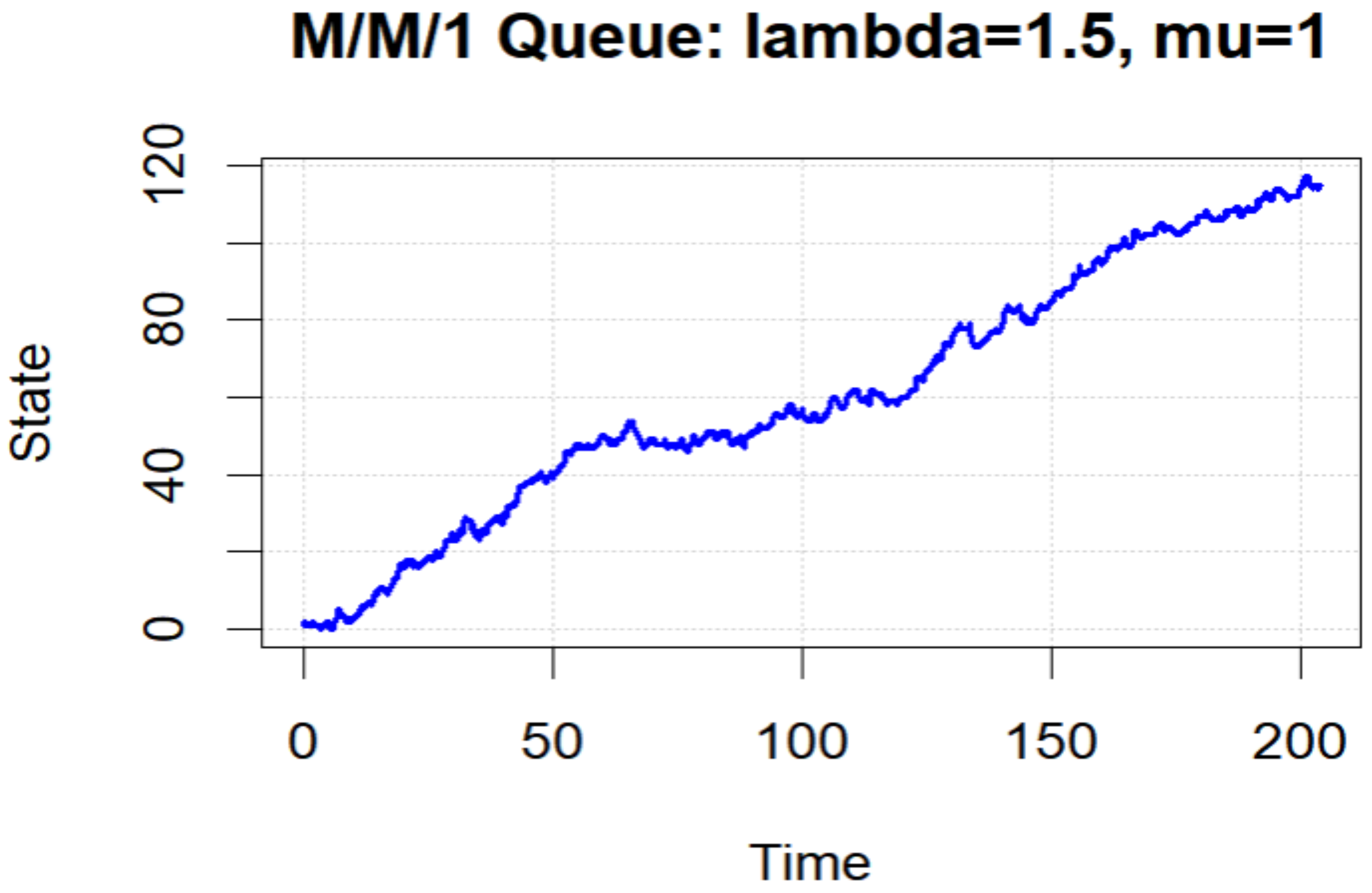
R CODE



M/M/1 QUEUE: EXERCISE SOLUTION (CONT.)



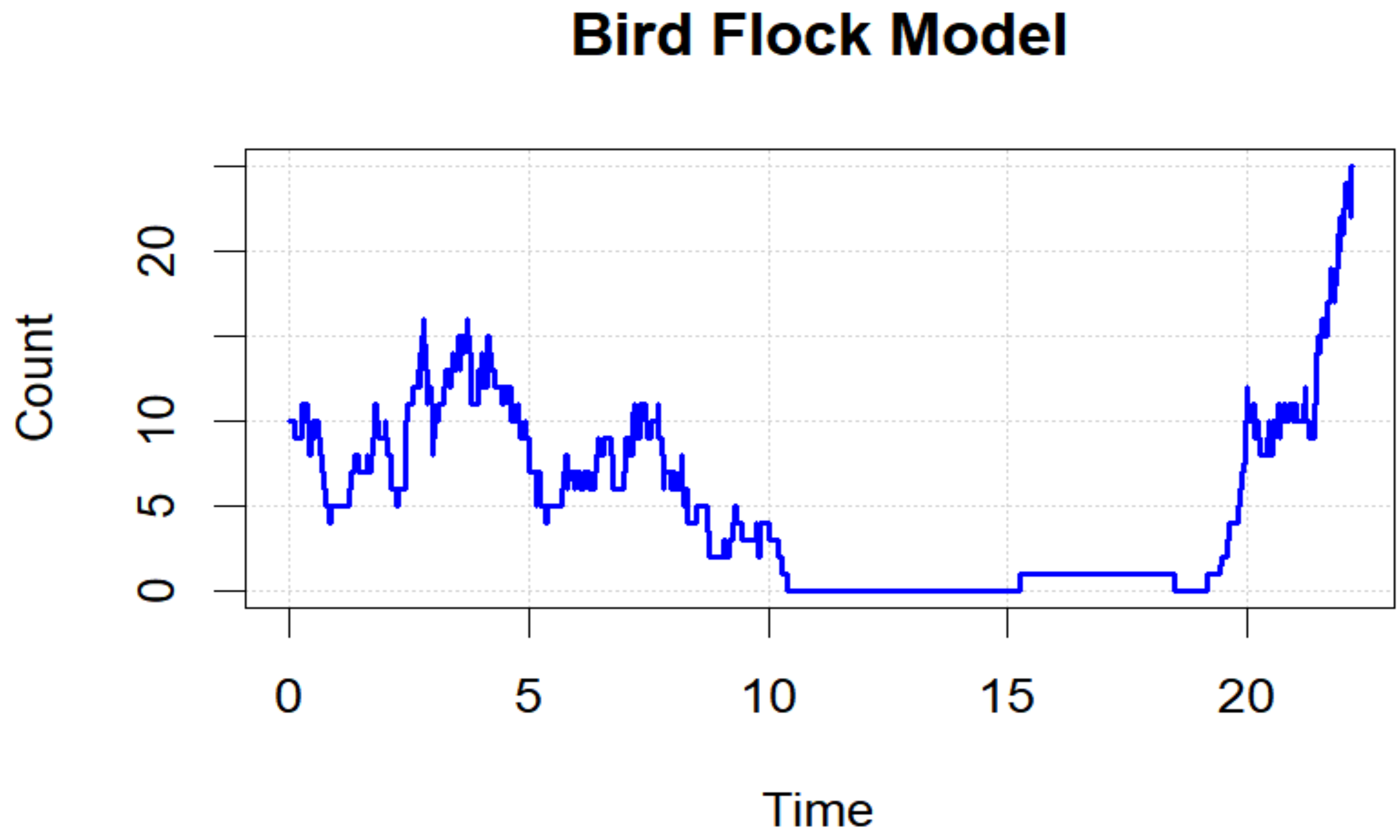
M/M/1 QUEUE: EXERCISE SOLUTION (CONT.)



APPLICATION OF BIRTH-AND-DEATH PROCESS (2)

Bird flock size is often modeled as a ***birth-and-death process with immigration and emigration***. In this model, $\lambda_n = \lambda n + \alpha$ and $\mu_n = \mu n + \beta$ where α is the ***rate of immigration*** (joining the flock) and β is the ***rate of emigration*** (leaving the flock). Note that because of immigration, the flock would never die out. We generate a trajectory of the process until the flock size increases from 10 to 25 birds, assuming $\lambda = 1.4, \mu = 1.2, \alpha = 0.3$, and $\beta = 0.1$.

APPLICATION OF BIRTH-AND-DEATH PROCESS (2) (CONT.)

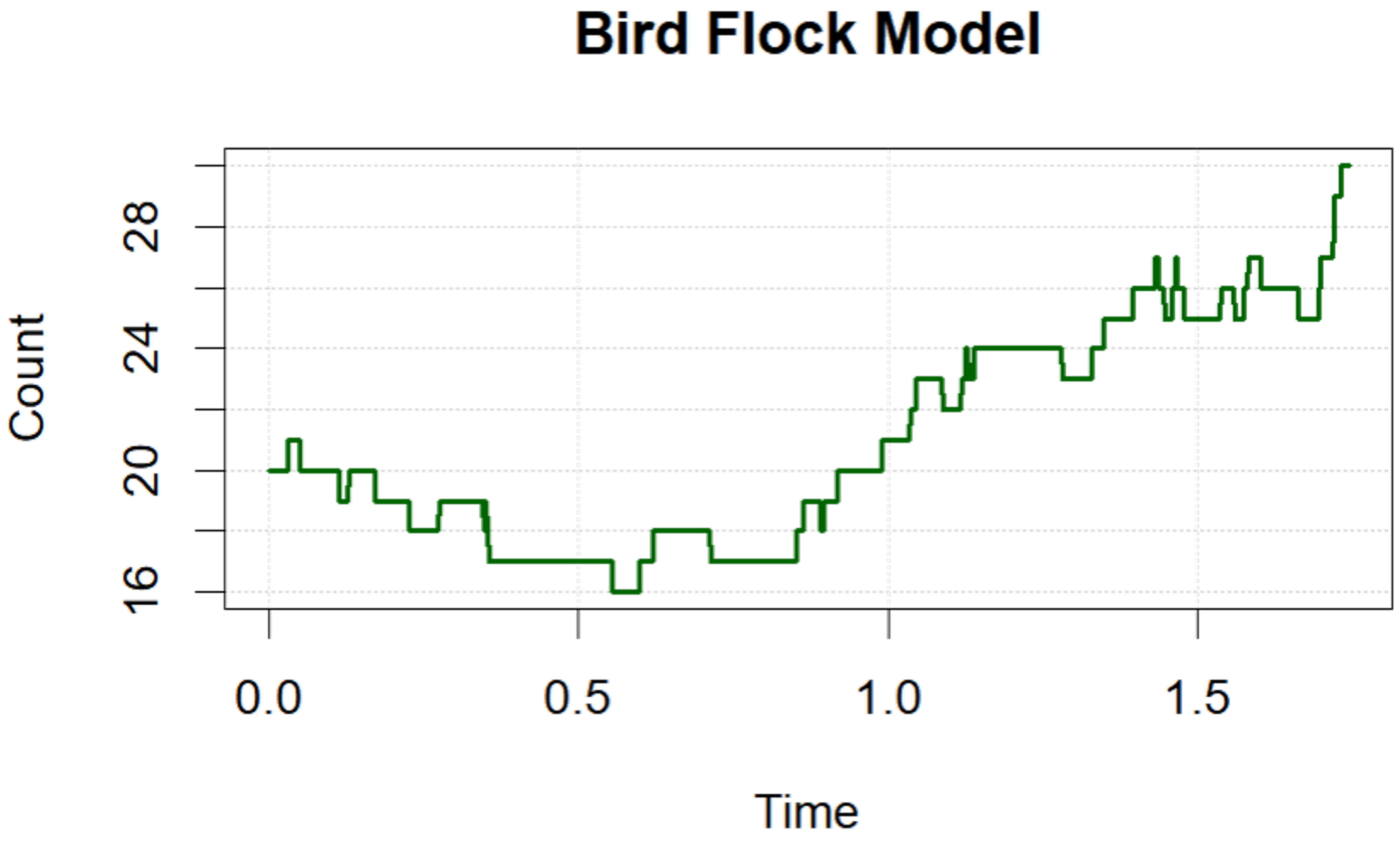


[R CODE]

BIRD FLOCK SIZE EXERCISE

Plot a trajectory of a birth-and-death process with immigration and emigration with $\lambda = 0.8, \mu = 1.1, \alpha = 0.1$, and $\beta = 0.2$. Assume it starts with 20 birds and ends with 30 birds.

BIRD FLOCK SIZE EXERCISE SOLUTION

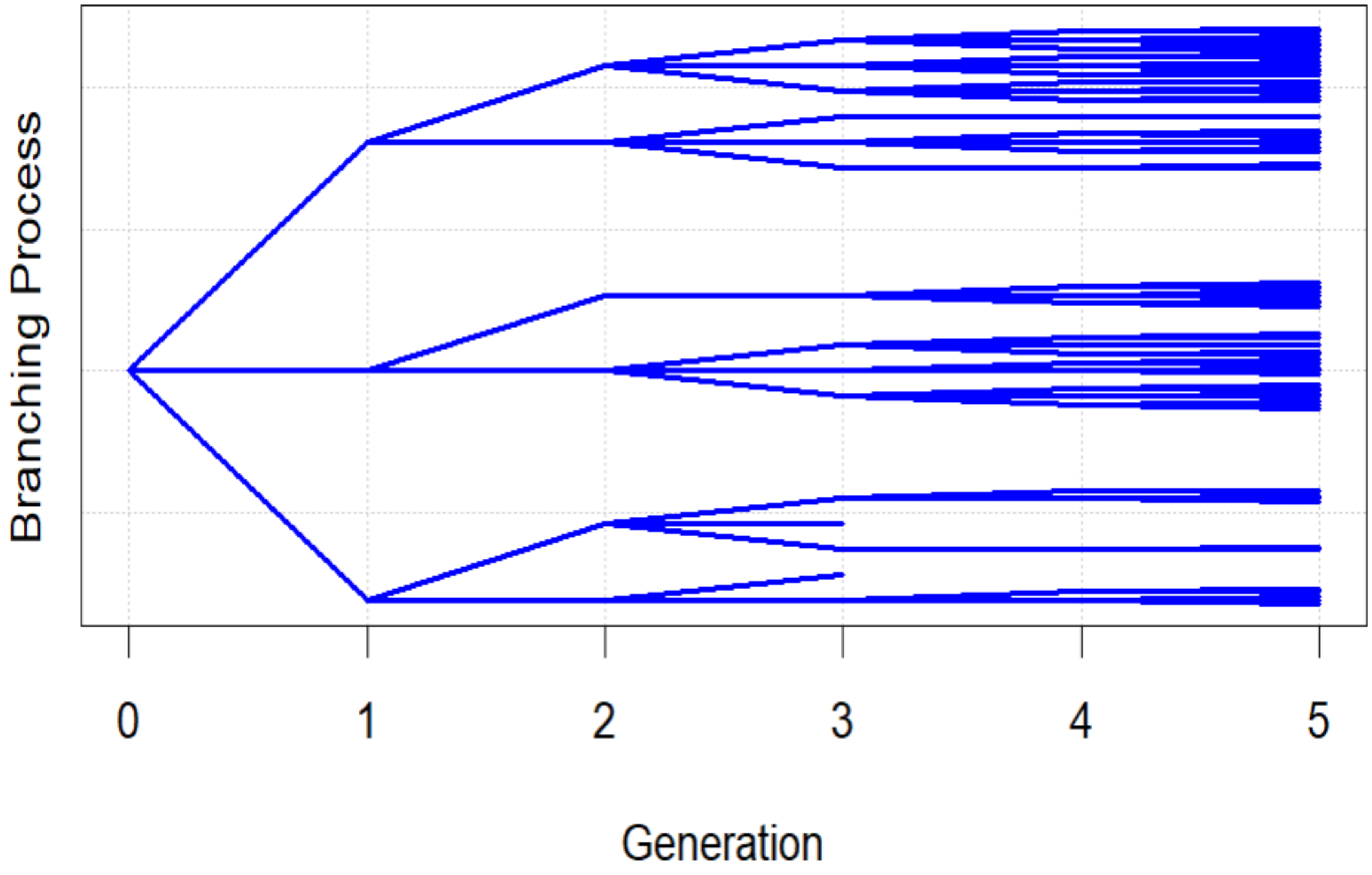


R CODE

BRANCHING PROCESS

Definition. A branching process is a discrete-time stochastic process that keeps track of the size of the nth generation of multiplying particles. It starts with some number of particles in the 0th generation. Each particle survives for one-time unit, at the end of which it splits into a random number of particles with a known probability distribution. The offspring particles survive for one-time unit, and produce a random number of offspring, independently from each other, and the process continues.

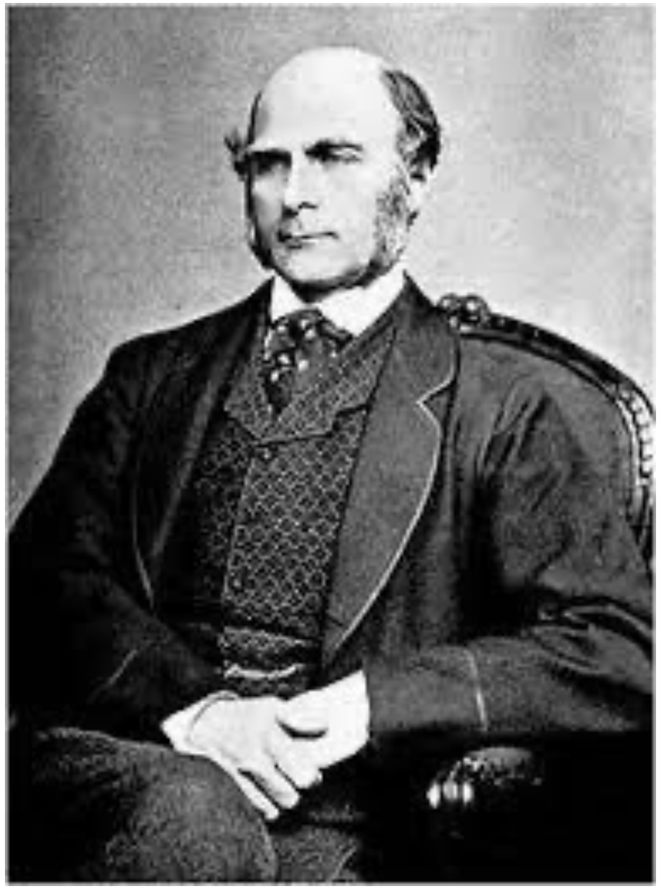
SIMULATION OF BRANCHING PROCESS



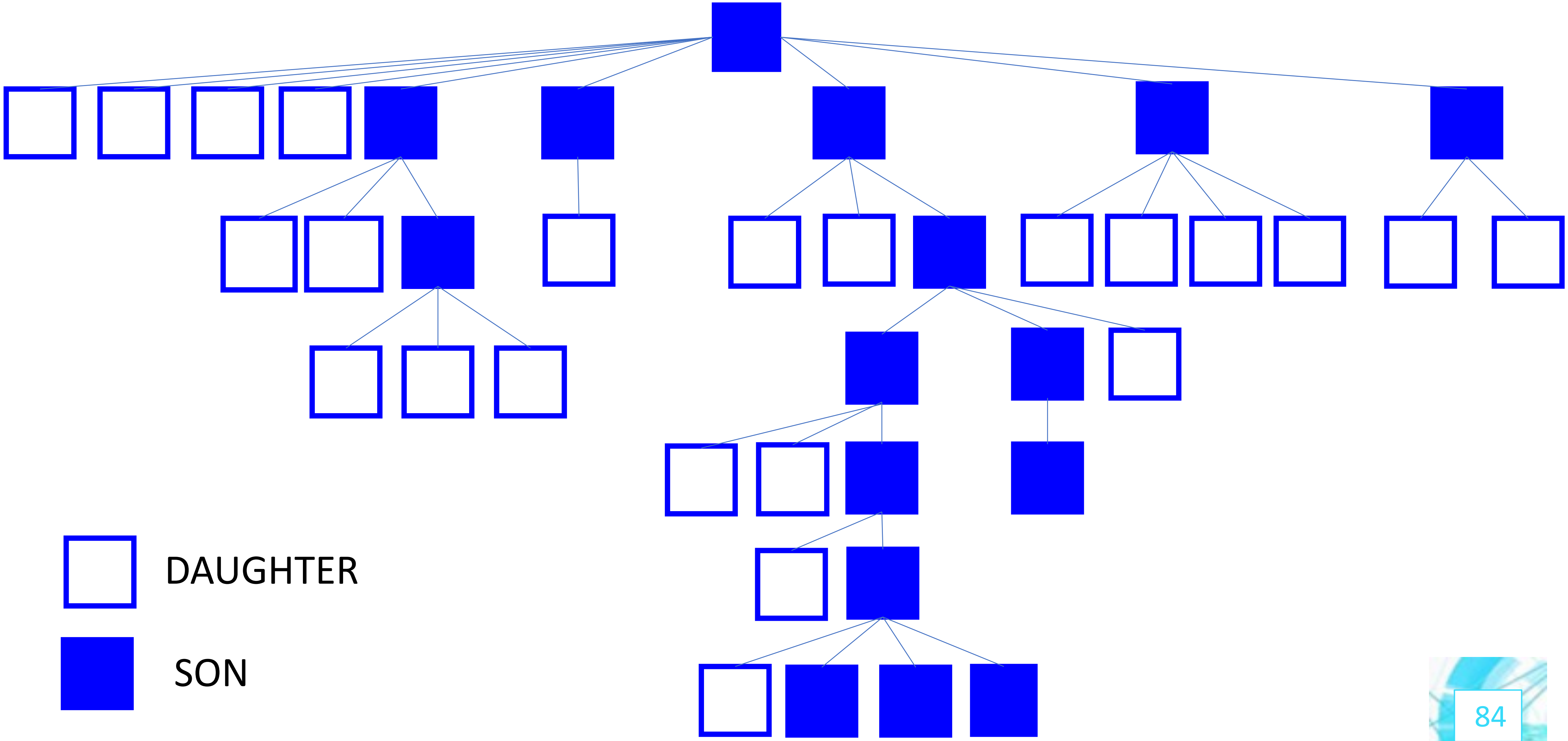
R CODE

Historical Note

Branching process is also called ***Galton process*** after Sir Francis Galton (1822-1911), an Englishman, who made a significant contribution to many scientific fields. He was interested in survival of surnames of noble English families when he came up with a branching process.



DR. OLGA'S GENEALOGY TREE



DAUGHTER

SON

BRANCHING PROCESS (CONT.)

Fact: Branching process is a Markov chain.

Definition. Denote by Z the random number of offspring of one particle. Let μ denote the expected number of Z. If $\mu < 1$, the process is called ***subcritical***; if $\mu = 1$, the process is called ***critical***, and if $\mu > 1$, the process is termed ***supercritical***.

Fact: If $\mu \leq 1$, the process will eventually die out with probability 1.

If $\mu > 1$, the process can still die out with a positive probability π_0 but

BRANCHING PROCESS (CONT.)

$\pi_0 < 1$ and is the smallest positive solution of the equation:

$$\pi_0 = \sum_{k=0}^{\infty} \pi_0^k P(Z = k).$$

Example. Suppose a particle produces no particles 20% of the time, one particle 50% of the time, and two particles 30% of the time. Then $P(Z = 0) = 0.2, P(Z = 1) = 0.5$, and $P(Z = 2) = 0.3$. Its mean is $\mu = EZ = (0)(0.2) + (1)(0.5) + (2)(0.3) = 1.1 > 1$, the process is supercritical, and the probability of extinction π_0 solves $\pi_0 = 0.2\pi_0^0 + 0.5\pi_0^1 + 0.3\pi_0^2$, or $0.3\,\pi_0^2 - 0.5\,\pi_0 + 0.2 = 0.3(\pi_0 - 1)\left(\pi_0 - \frac{2}{3}\right) = 0$. So, $\pi_0 = 2/3$. R CODE

We simulate 1,000 trajectories and compute empirical probability of 0.662.

APPLICATION OF BRANCHING PROCESS

Based on work by Alfred J. Lotka in 1931 who analyzed the data from the 1920 U.S. Census, suppose that male offspring has the distribution: $p(0) = 0.4828$ and $p(n) = (0.228292)(0.5586)^{n-1}$, $n = 1, 2, 3, \ldots$. The expected size of male offspring is $\mu = (0.228292)\sum_{n=1}^{\infty} n\,(0.5586)^{n-1} = \frac{0.228292}{(1-0.5586)^2} = 1.171726 > 1$, so the population growth is a supercritical process. The probability π_0 of extinction is the smallest positive solution to the equation $\pi_0 = 0.4828 + \sum_{n=1}^{\infty} \pi_0^n\,(0.228292)(0.5586)^{n-1} = 0.4828 + \frac{0.228292\pi_0}{1-0.5586\pi_0}$, which is a quadratic equation $0.5586\pi_0^2 - 1.0414\pi_0 + 0.4828 = 0$. The solution is $\pi_0 = 0.864304$.

MORE EXAMPLES OF BRANCHING PROCESSES

Here are more examples of processes that are modeled well by a branching process:

1. Growth of a colony of bacteria
2. Spread of an infectious diseases
3. Spread of a computer virus
4. Parlaying strategy in gambling

BRANCHING PROCESS EXERCISE

Parlaying in gambling is defined as a series of bets in which winnings are used as a stake for further bets. Suppose a gambler starts with a stake of \$1, and can win \$1 with probability 0.3, or \$5 with probability 0.2, or \$10 with probability 0.1, or \$0 with probability 0.4.

- Show that this is a supercritical process.
- Find the probability that the gambler's stake eventually turns into \$0.
- Simulate 1,000 trajectories and estimate the probability of ruin.

BRANCHING PROCESS EXERCISE SOLUTION

- Show that this is a supercritical process.

$$\mu = (1)(0.3) + (5)(0.2) + (10)(0.1) + (0)(0.4) = 2.3 > 1.$$

- Find the probability that the gambler's stake eventually turns into $0.

Let π_0 be the probability that the gambler's stake eventually turns into $0. It is the smallest positive solution to the equation $\pi_0 = 0.4 + 0.3\pi_0 + 0.2\pi_0^5 + 0.1\pi_0^{10}$. Solved numerically, $\pi_0 = 0.5931886$.

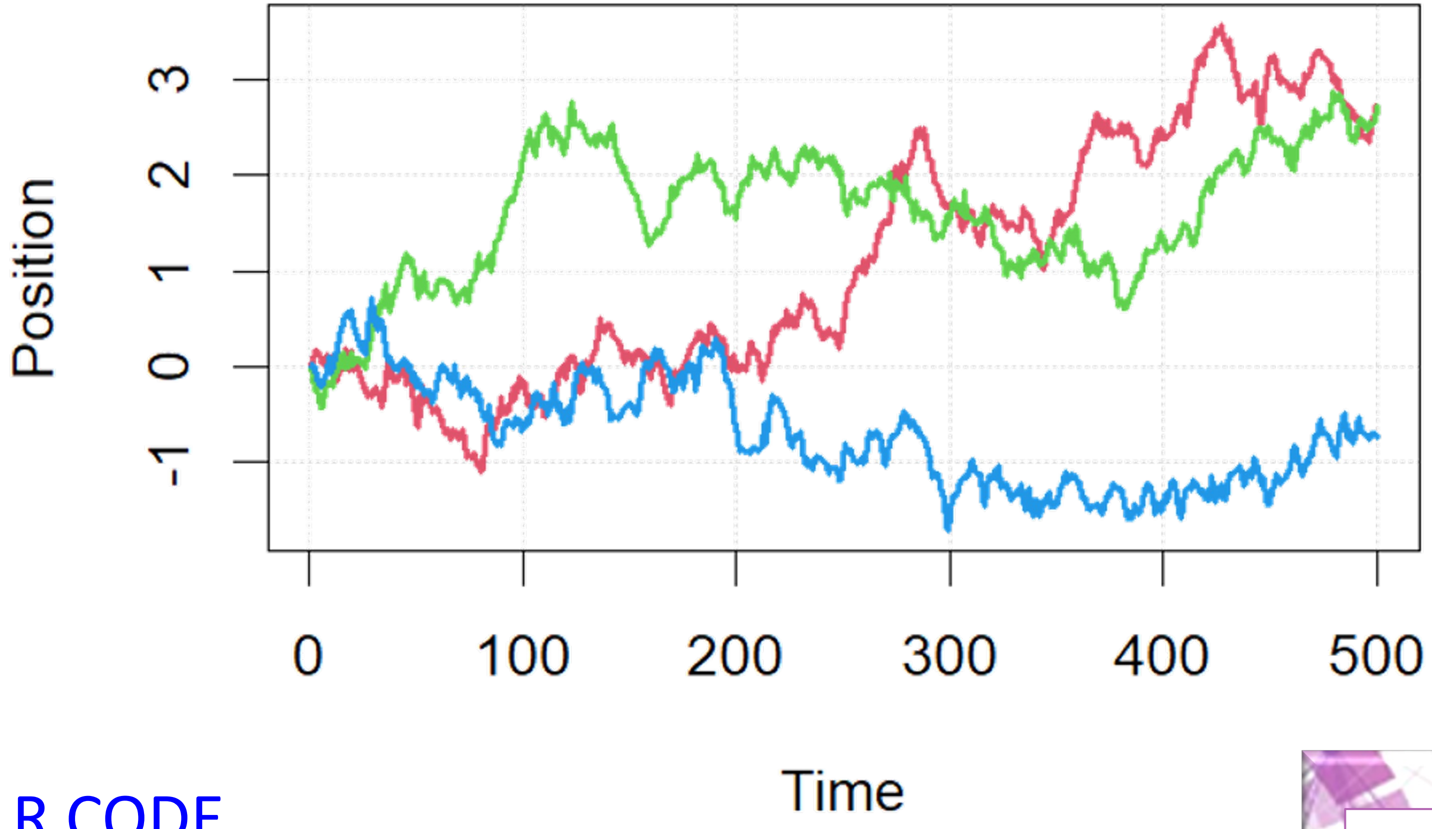
R CODE

- Simulate 1,000 trajectories and estimate the probability of ruin.

0.6

BROWNIAN MOTION

Definition. A standard one-dimensional Brownian motion $\{B(t), t \geq 0\}$ is a stochastic process that starts at zero, that is, $B(0) = 0$, and $B(t) \sim N(0, t), t > 0$.



R CODE

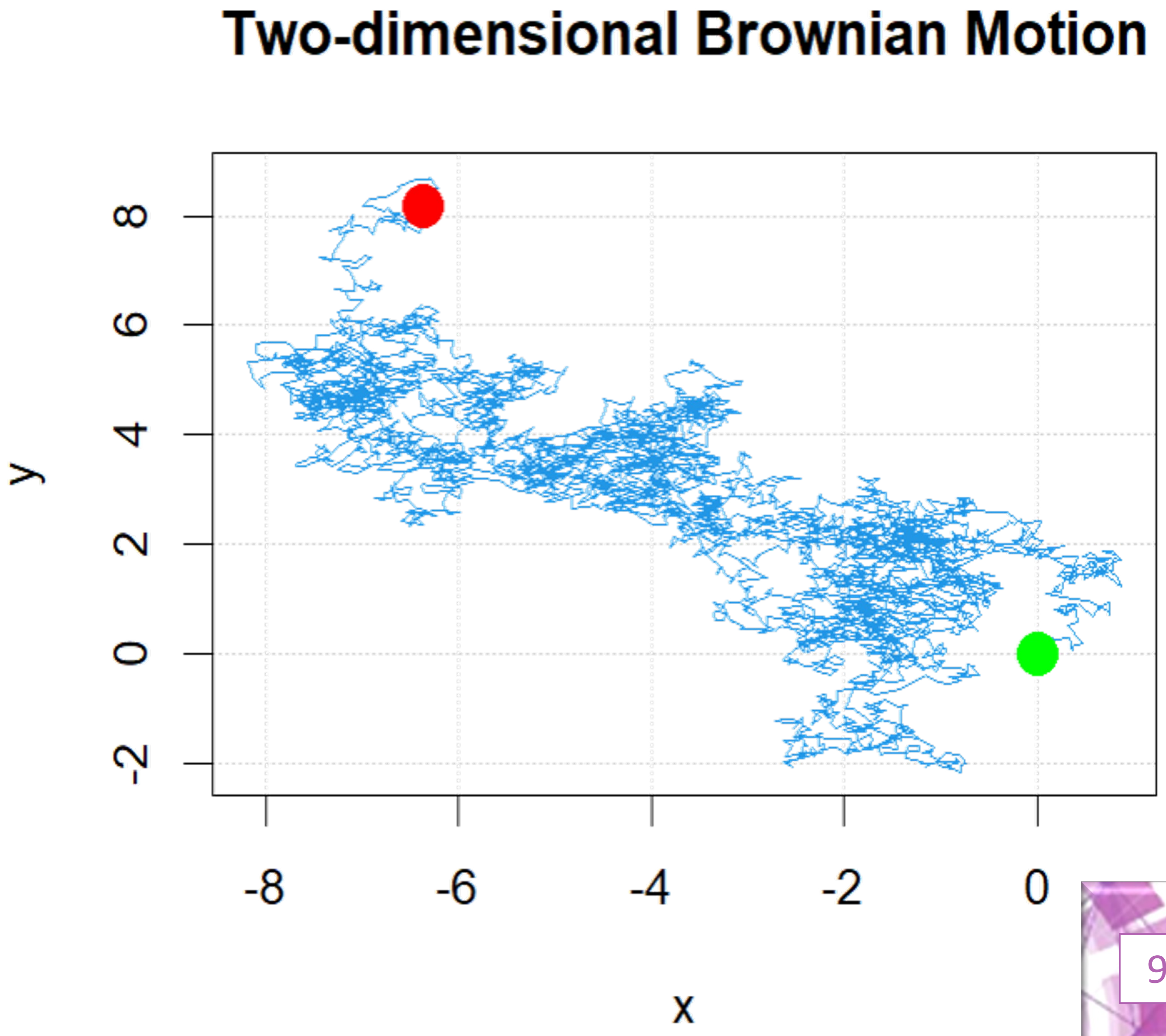
HISTORICAL NOTE

- In 1828, an English botanist Robert Brown (1773-1858) observed an erratic movement of dust particles in liquid (on a microscopic level) and described the motion in a publication.
- In 1921, an American mathematician Norbert Wiener (1894 – 1964) proposed a rigorous mathematical model explaining this motion. The erratic motion is explained by the particle being hit by the molecules of the liquid.
- Brownian motion is often called a Wiener process.

BROWNIAN MOTION (CONT.)

Definition. A ***two-dimensional Brownian motion*** is a stochastic process that keeps track of two coordinates, both of which are independent Brownian motions.

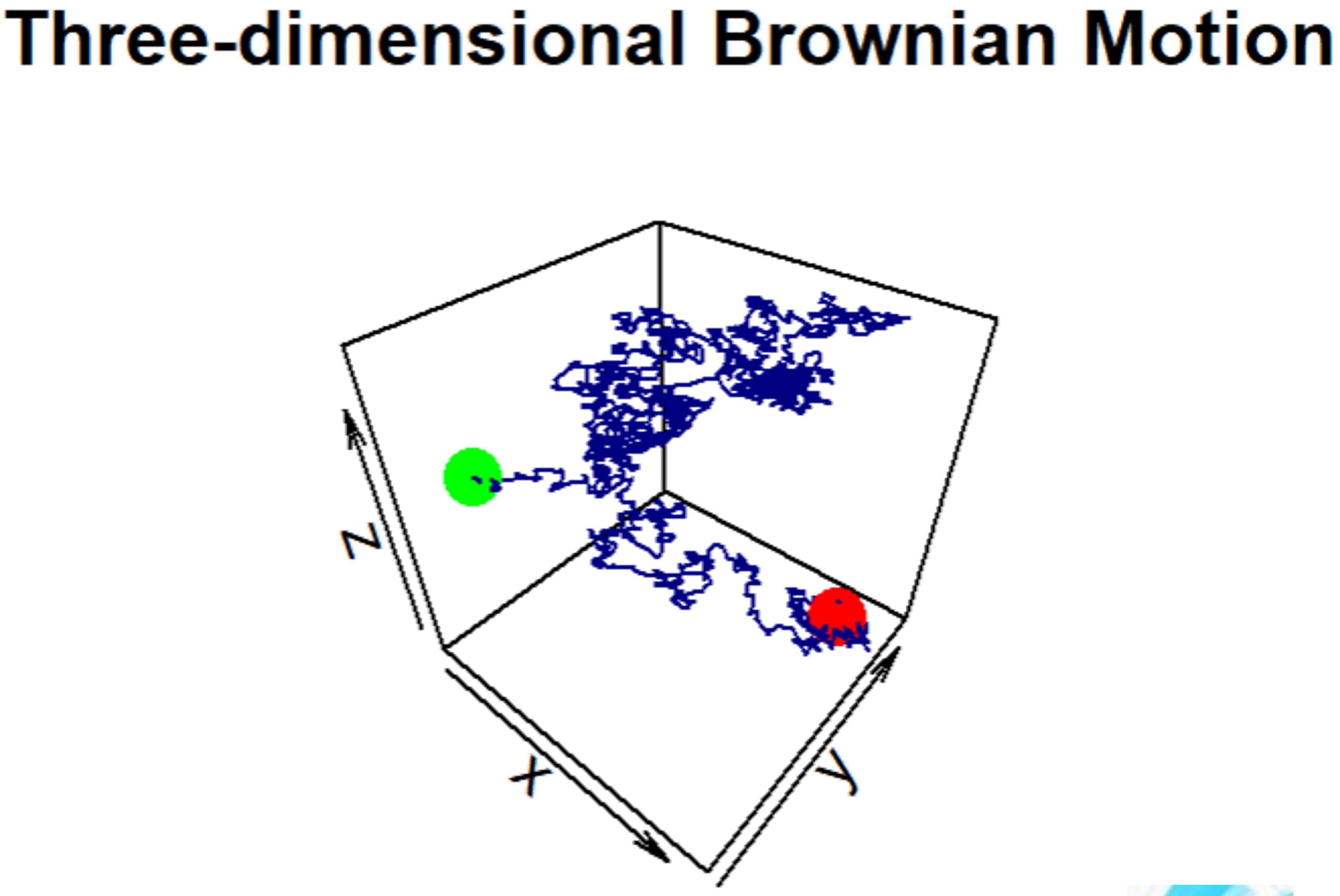
[R CODE]



BROWNIAN MOTION (CONT.)

Definition. A ***three-dimensional Brownian motion*** is a stochastic process that models position by three coordinates, defined by three independent Brownian motions.

R CODE



BROWNIAN MOTION (CONT.)

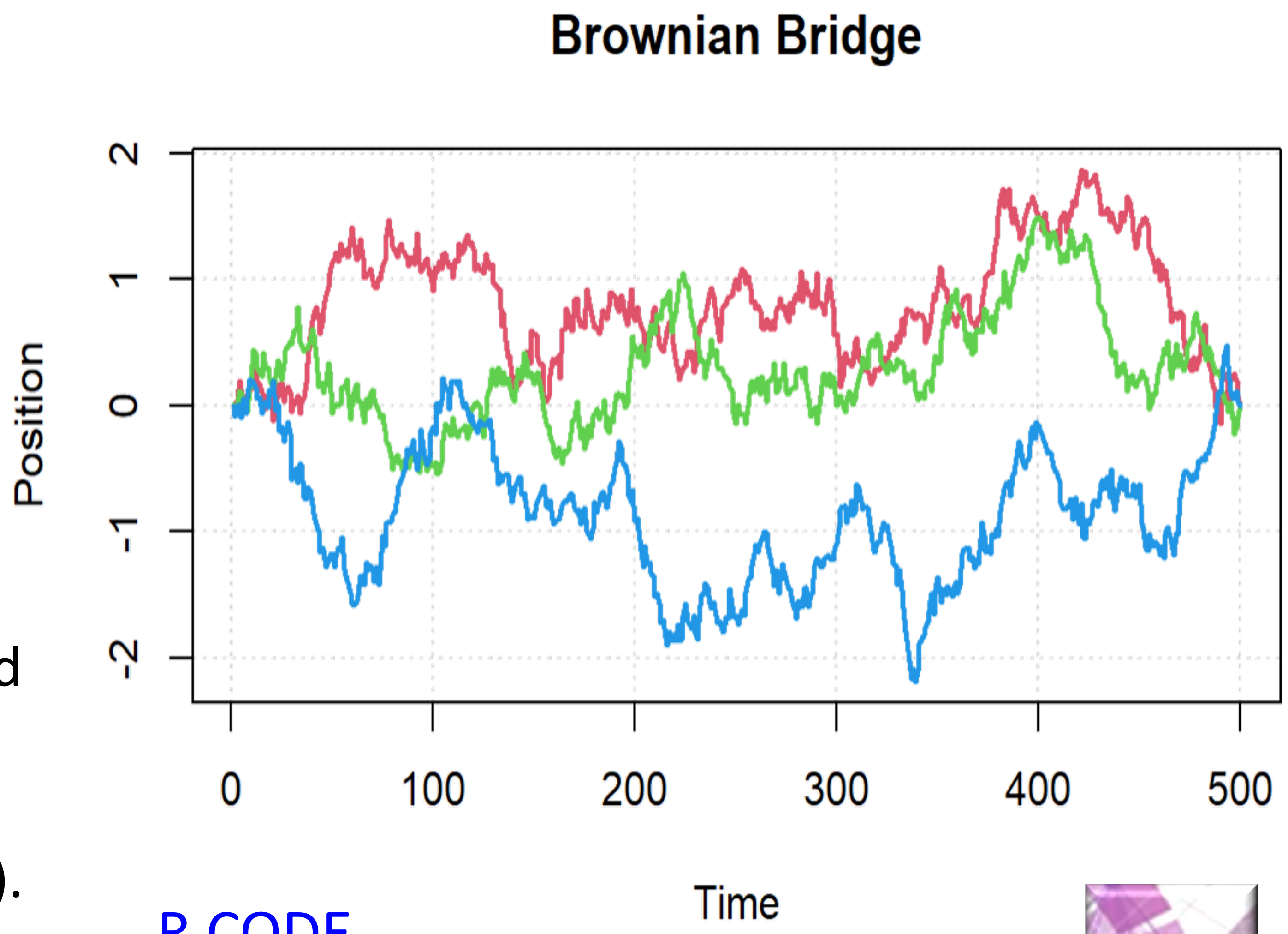
<u>Definition.</u> Suppose that $\{B(t), t \geq 0\}$ is a standard Brownian motion.

The process

$$X(t) = B(t) - tB(1), 0 \leq t \leq 1,$$

is called a ***Brownian bridge***.

Note that $X(t)$ behaves like a Brownian Motion, but is limited to [0,1] and is tied at both ends since $X(0) = X(1) = 0$ (resembling a bridge).

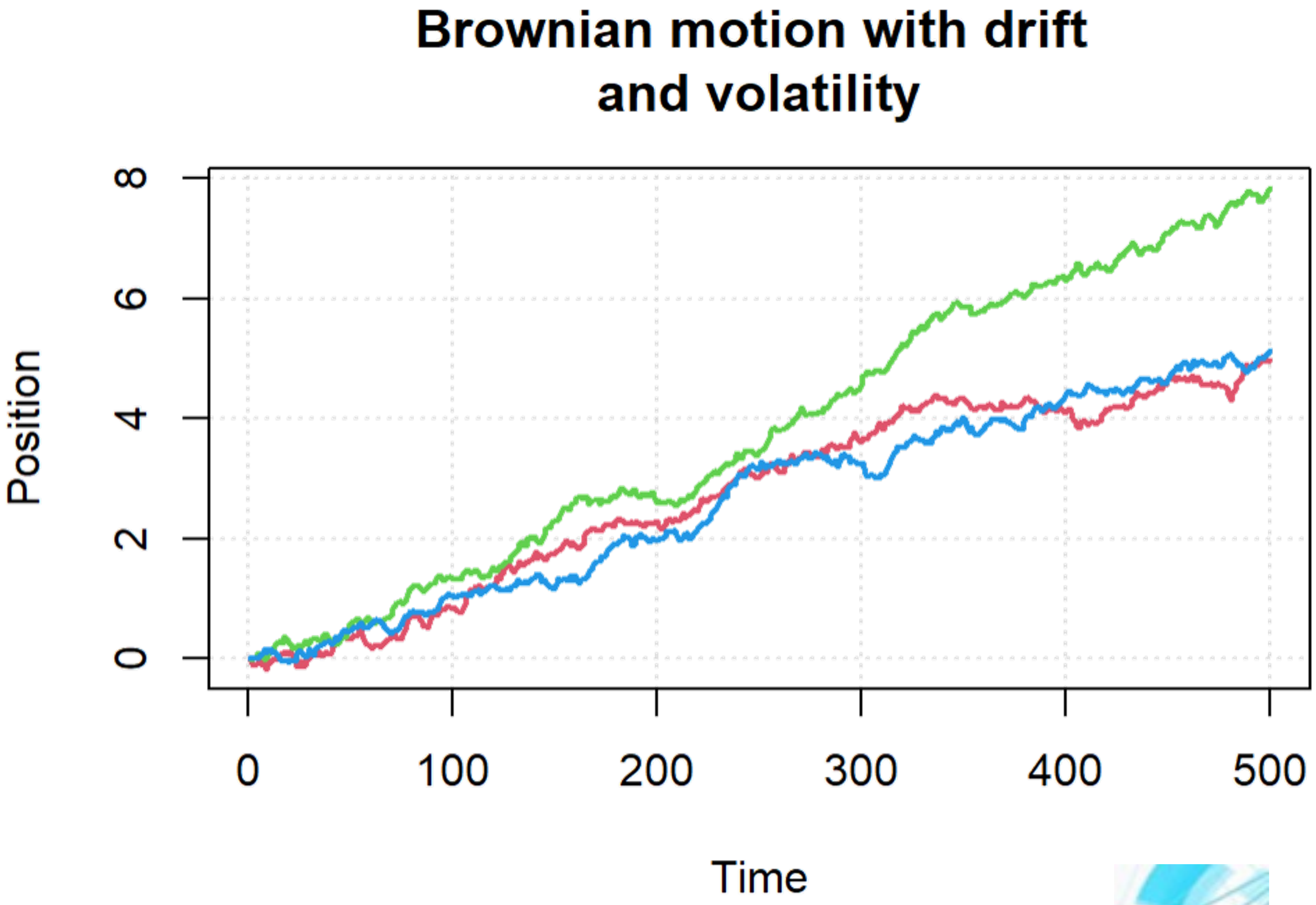


[R CODE]

BROWNIAN MOTION (CONT.)

Definition. Let $\{B(t), t \geq 0\}$ denote a standard Brownian motion. A stochastic process $\{X(t) = \mu t + \sigma B(t), t \geq 0\}$ *is* called a Brownian motion with the **drift coefficient μ** and **volatility coefficient σ**. Note that $X(t) \sim N(\mu t, \sigma^2 t)$.

R CODE

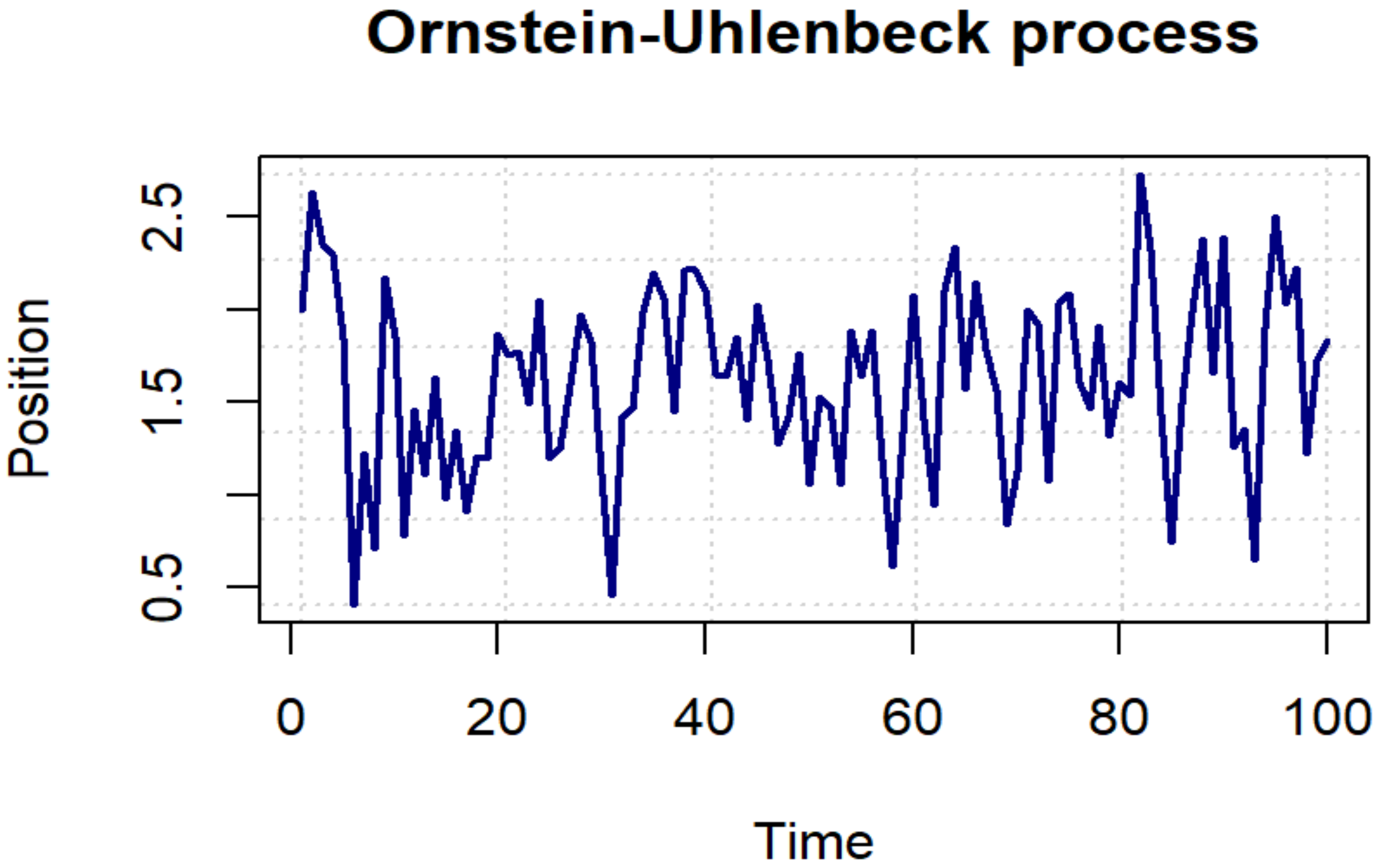


BROWNIAN MOTION (CONT.)

Definition. The stochastic process $\left\{X(t) = X(0)e^{-\theta t} + \mu\left(1 - e^{-\theta t}\right) + \frac{\sigma}{\sqrt{2\theta}}e^{-\theta t}B\left(e^{2\theta t} - 1\right), t \geq 0\right\}$ is called an ***Ornstein-Uhlenbeck process***.

Facts: The mean tends to μ as t increases, so the drift μ is the ***long-term mean***, and the process is called ***mean-reverting***. The parameter θ represents the ***rate*** by which the process reverts towards the mean. In addition, the variance of this process is bounded by a constant $\frac{\sigma^2}{2\theta}$ and in the long run, approaches this constant.

BROWNIAN MOTION (CONT.)



R CODE

APPLICATION OF BROWNIAN MOTION (1)

Brownian bridge is used to model movements of herds as they walk on their trails during daylight time and return to their designated lodging for an overnight stay (maybe for source of water). The main goal of the research is to estimate the distance between the north-most and south-most points that the deer have reached. This distance approximates the ***diameter*** of the deer ***home range***. Assuming that the unit of measurement is one-tenth of a mile, we simulate 1,000 trajectories to give an empirical estimate of the diameter of home range that deer cover within 8-hour day light time (480 minutes). The answer is $(26.80793)(0.1) = 2.68$ miles. R CODE

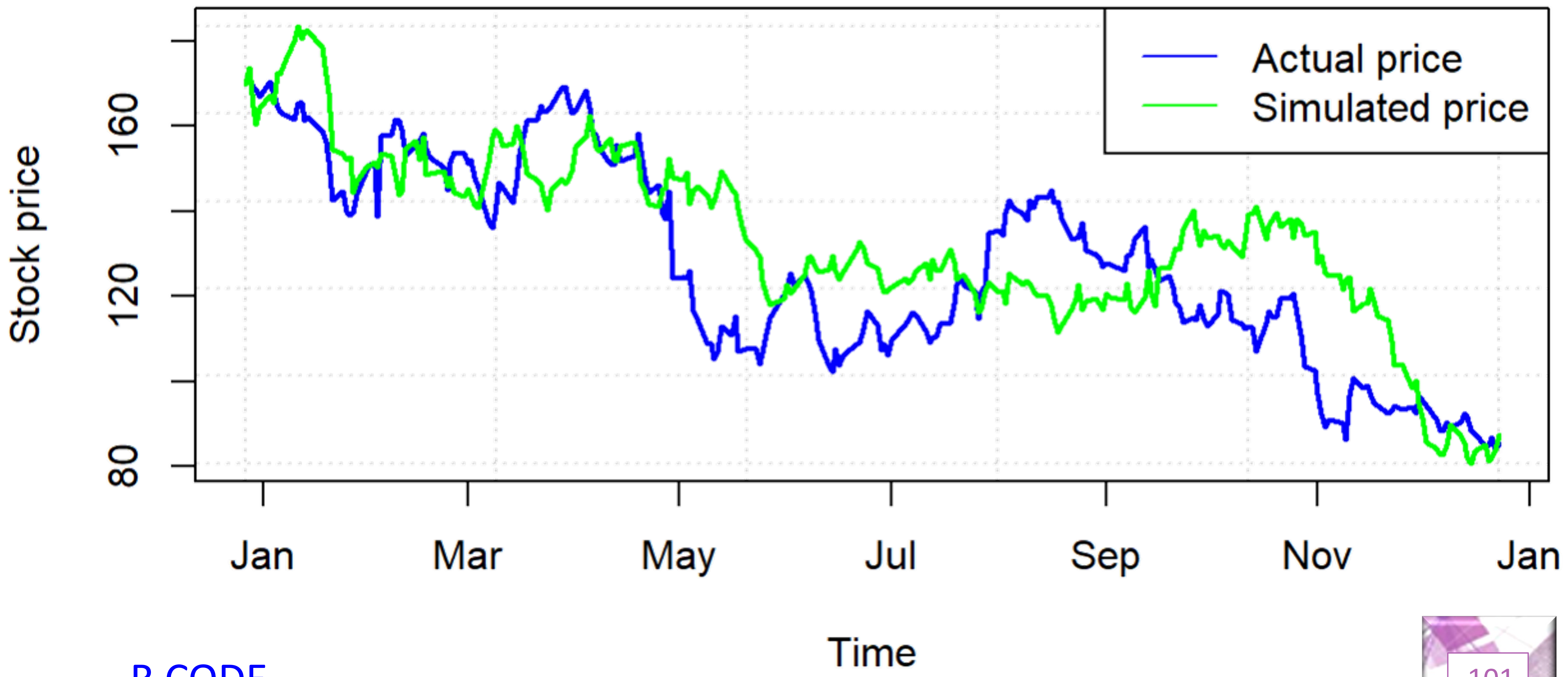
APPLICATION OF BROWNIAN MOTION (2)

A Brownian motion with drift and volatility is used to model the behavior of a stock price over time. The data set downloaded from

https://nance.yahoo.com/quote/AMZN/history/

contains Amazon.com, Inc. daily stock prices at the closing time of stock market exchange between 12/27/2021 and 12/23/2022. We estimate drift and volatility by noticing that $\Delta X = X(t+1) - X(t) = \mu + \sigma B(1) \sim N(\mu, \sigma^2)$, so the estimate of μ is the average of ΔX and the estimate of σ is the standard deviation of ΔX.

APPLICATION OF BROWNIAN MOTION (2) (CONT.)



[R CODE]()

APPLICATION OF BROWNIAN MOTION (3)

As opposed to stock that can rise indefinitely, commodity prices move in a limited range in a free market. If the price is high, the demand drops, and consequently, the price drops. Likewise, if the price is low, demand increases, and eventually, the price increases. This characteristic is called a reversion to a long-run mean and, as we know, an Ornstein-Uhlenbeck (OU) process is a good mathematical model that captures this mean-reversion property. We use it to model natural gas prices between 1/4/2010 and 8/11/2020 (downloaded from *kaggle.com*).

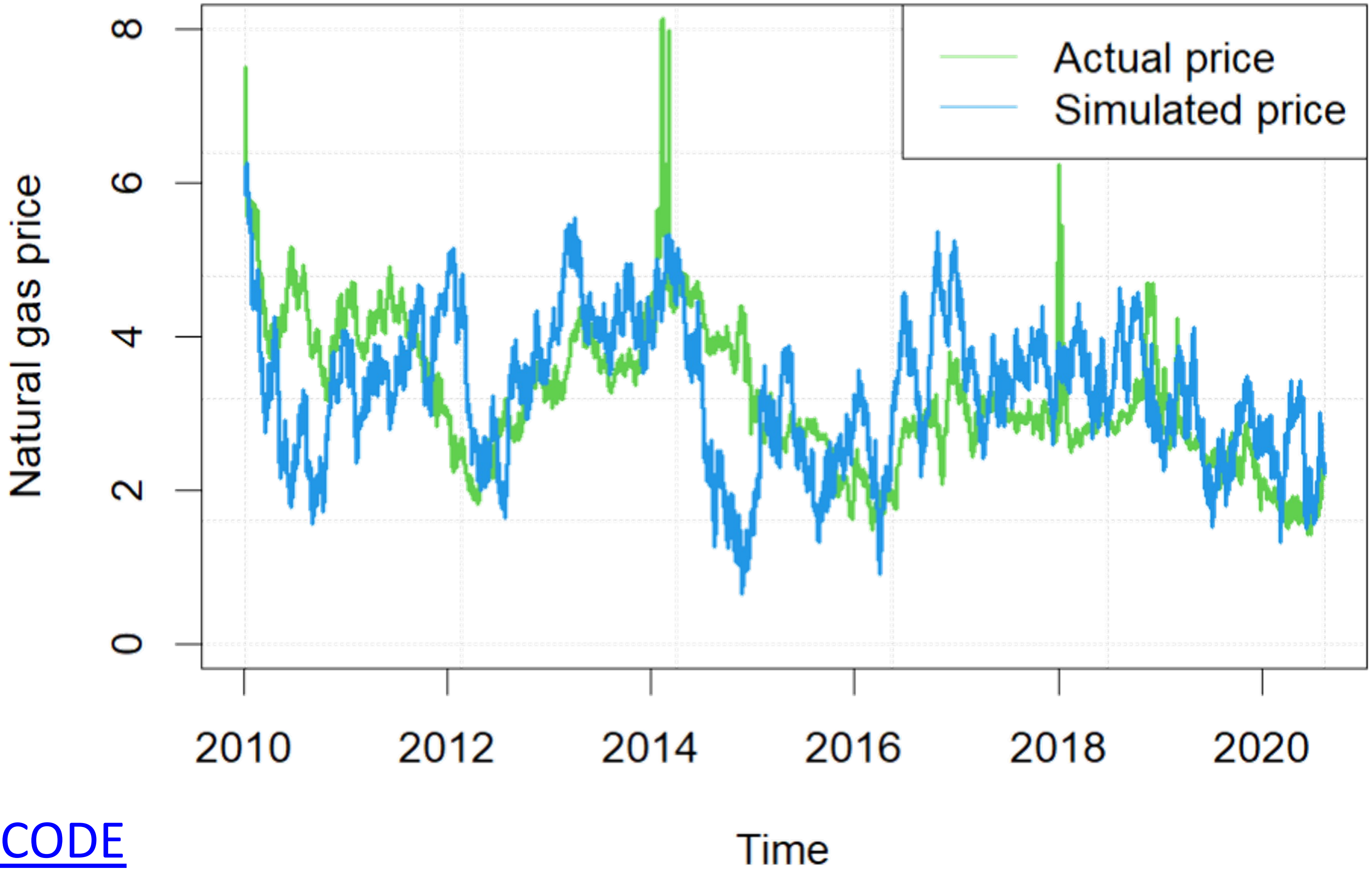
APPLICATION OF BROWNIAN MOTION (3) (CONT.)

Ornstein-Uhlenbeck process $\{X(t), t \geq 0\}$ satisfies the equation

$$\Delta X = X(t+1) - X(t) = \theta\mu - \theta X(t) + \sigma B(1).$$

It means that we can regress ΔX on $X(t)$ and, denoting the fitted intercept and slope by a and b, respectively, we see that the estimate of θ is $-b$, estimate of μ is $-a/b$, and σ is estimated as the sample standard deviation of the regression error.

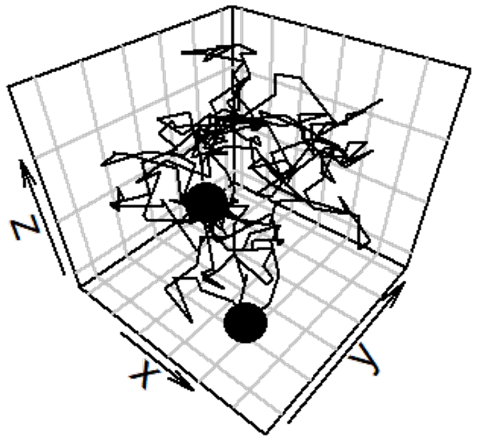
APPLICATION OF BROWNIAN MOTION (3) (CONT.)



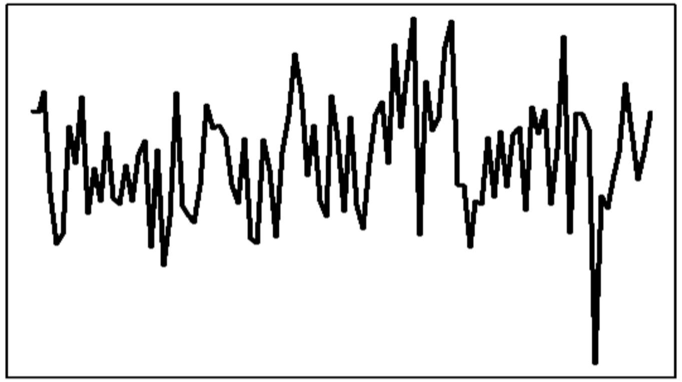
R CODE

BROWNIAN MOTION EXERCISE

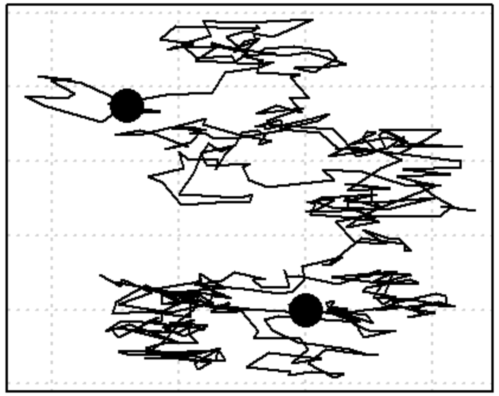
- Name the depicted processes.



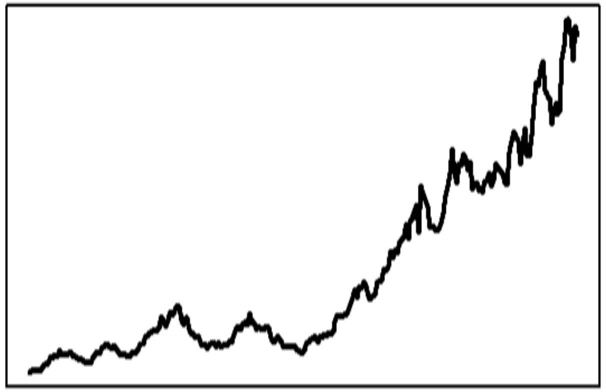
3D Brownian motion



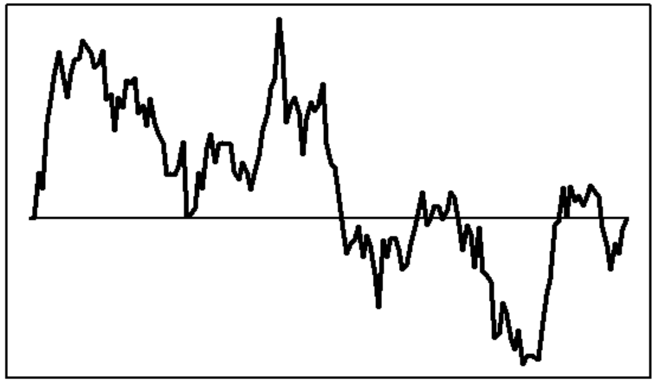
Ornstein-Uhlenbeck



2D Brownian motion



BM w/drift and volatility



Brownian bridge

BROWNIAN MOTION EXERCISE (CONT.)

- Match each description with the process name.
  - movement of a second grader after school
  - price of gold, sugar, cotton, soybean
  - price of the stock of Johnson & Johnson, Starbucks, Disney, Boeing, and Tesla
  - movement of a restless second-grader along a hallway during recess (between two recess bells)
  - exchange rate of Japanese yens and Canadian dollars

2D Brownian motion

Ornstein-Uhlenbeck process

Brownian motion with drift and volatility

Brownian bridge

Ornstein-Uhlenbeck process

BROWNIAN MOTION EXERCISE (CONT.)

- movement of bacterium on the water surface — 2D Brownian motion
- movement of aerosol particles in the air — 3D Brownian motion
- daily movements of a pelican within its home range on Grande Isle, Louisiana, as measured north-south — Brownian bridge
- movement of molecules in boiling water — 3D Brownian motion


Thank You!