Assignment 2

Solutions to Linear Dynamical Systems in discrete and continuous time

Juan Carlos Villaseñor-Derbez

January 23, 2018

```
suppressPackageStartupMessages({
  library(deSolve)
  library(tidyverse)
})
```

1 Discrete Time

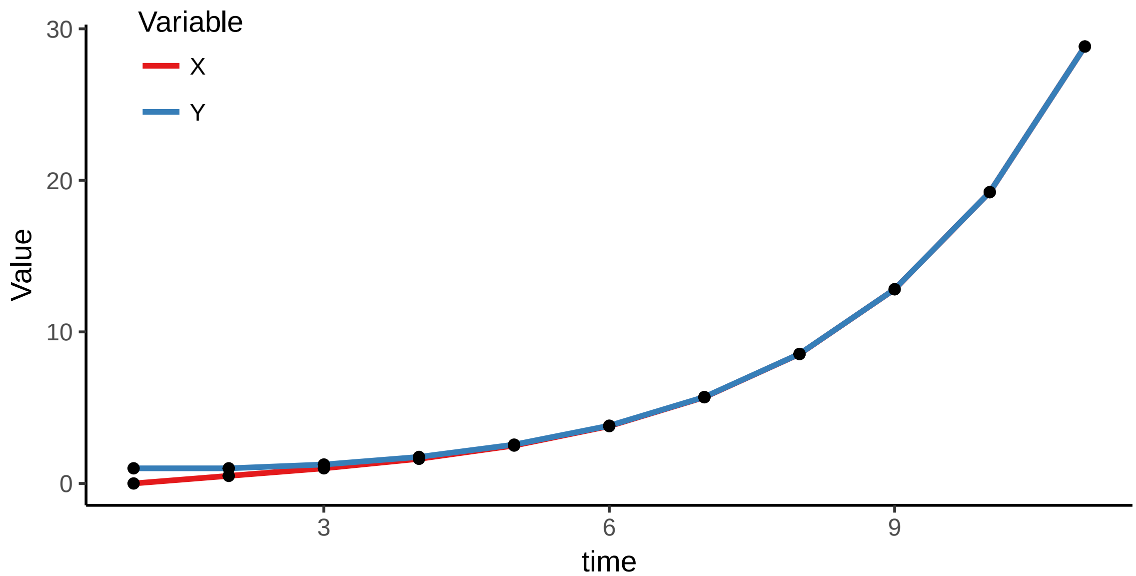
Consider the system of linear discrete-time equations:

$$
\begin{aligned}
x_{t+1} &= x_t + 0.5y_t \\
y_{t+1} &= 0.5x_t + y_t
\end{aligned}
\tag{1}
$$

with initial conditions:
$x_0 = 0$
$y_0 = 1$

1.1 Calculate and plot the *numerical solution* to Eq 1 with $t \in (0, 10)$

```
X_0 <- 0 # Initial conditions for X
Y_0 <- 1 # Initial conditons for Y
nyears <- 10 #Number of years
time <- seq(0:nyears) #Vector of years
X <- rep(NULL, nyears + 1) #Pre-define vector of X
Y <- rep(NULL, nyears + 1) #Pre-define vctor of Y
X[1] <- X_0 #Assign initial conditions to the vectors
Y[1] <- Y_0

# Iterate over years, calculating each state variable
for(i in 1:nyears){
  X[i + 1] <- X[i] + 0.5 * Y[i]
  Y[i + 1] <- 0.5 * X[i] + Y[i]
}
# Plot the solution
data.frame(time, X, Y) %>%
  gather(Variable, Value, -time) %>%
  ggplot(aes(x = time, y = Value, color = Variable)) +
  geom_line(size = 1) +
  geom_point(color = "black") +
  theme_classic() +
  scale_color_brewer(palette = "Set1") + theme(legend.position = c(0.1, 0.9))
```



1.2 Calculate and plot the *analytical solution* to Eq 1 with $t \in (0, 10)$

In matrix form, Eq 1 can be written as:

$$\begin{bmatrix} X_{t+1} \\ Y_{t+1} \end{bmatrix} = A \times \begin{bmatrix} X_t \\ Y_t \end{bmatrix} \tag{2}$$

Where A is the 2×2 matrix:

$$A = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$$

Therefore, the system is:

$$\begin{bmatrix} X_{t+1} \\ Y_{t+1} \end{bmatrix} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix} \times \begin{bmatrix} X_t \\ Y_t \end{bmatrix}$$

This particular system has a general solution of the form:

$$\begin{bmatrix} X(t) \\ Y(t) \end{bmatrix} = C_1 \lambda_1^t \bar{v_1} + C_2 \lambda_2^t \bar{v_2} \tag{3}$$

Where C_1 and C_2 represent variables specific to the particular problem, based on the initial conditions. λ_1 and λ_2 are the eigenvalues of A, and $\bar{v_1}$ and $\bar{v_2}$ are the eigenvectors of A.

```
A <- matrix(c(1, 0.5, 0.5, 1), nrow = 2)

lambda_1 <- eigen(A)$values[1]
lambda_2 <- eigen(A)$values[2]

v_1 <- eigen(A)$vectors[,1]

v_2 <- eigen(A)$vectors[,2]
```

The eigenvalues are $\lambda_1 = 1.5$ and $\lambda_2 = 0.5$. The eigenvectors are $\bar{v_1} = \begin{bmatrix} 0.7071 \\ 0.7071 \end{bmatrix}$, and $\bar{v_2} = \begin{bmatrix} -0.7071 \\ 0.7071 \end{bmatrix}$.

Substituting these values into Eq 3, and simplifying we obtain:

$$\begin{aligned} X(t) &= C_1(1.5)^t(1) + C_2 0.5^t(-1) \\ Y(t) &= C_1(1.5)^t(1) + C_2 0.5^t(1) \end{aligned} \tag{4}$$

Using the known initial conditions ($X(0) = 0$ and $Y(0) = 1$), we can evaluate Eq 4 at $t = 0$ and solve for C_1 and C_2. Solving the equation for state variable N:

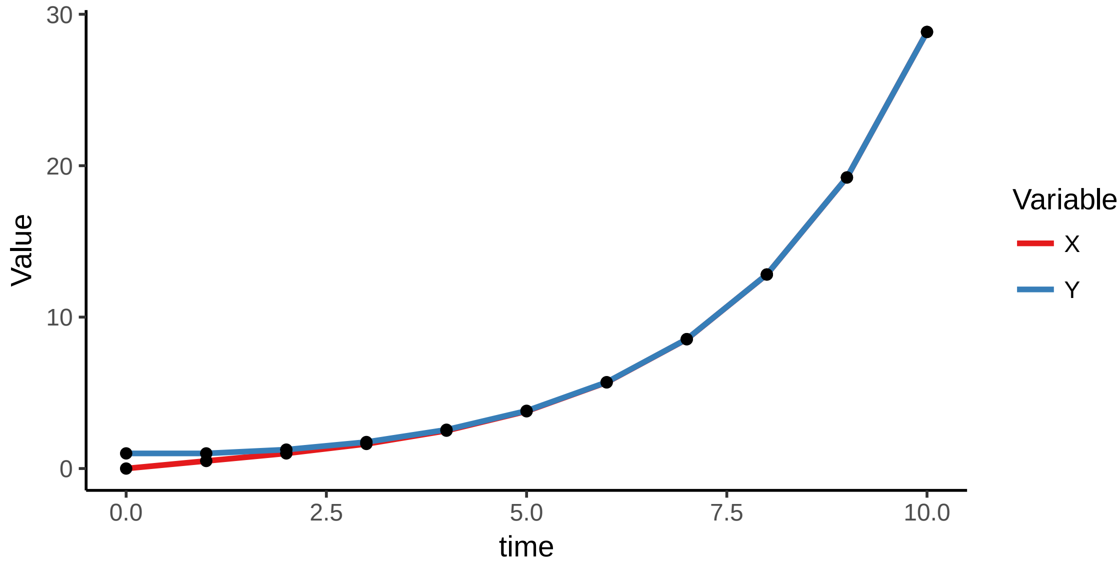
$$\begin{aligned} 0 &= C_1(1.5)^0(1) + C_2 0.5^0(-1) \\ 0 &= C_1(1)(1) + C_2(1)(-1) \\ 0 &= C_1 - C_2 \\ C_1 &= C_2 \end{aligned} \tag{5}$$

Substituting this in the equation for state variable Y:

$$\begin{aligned} 1 &= C_1(1.5)^0(1) + C_2 0.5^0(1) \\ 1 &= C_1(1)(1) + C_2(1)(1) \\ 1 &= C_1 + C_2 \\ 1 &= C_2 + C_2 \\ 1 &= 2C_2 \\ C_2 &= \frac{1}{2} \end{aligned} \tag{6}$$

Returning to Eq 5, we obtain that $C_1 = \frac{1}{2}$

```r
C1 <- 0.5
C2 <- C1

t <- seq(0, nyears)

XY <- cbind(C1*(lambda_1^t)*1 + C2*(lambda_2^t)*(-1),
            C1*(lambda_1^t)*1 + C2*(lambda_2^t)*(1))

cbind(t, XY) %>%
  as.data.frame() %>%
  magrittr::set_colnames(c("time", "X", "Y")) %>%
  gather(Variable, Value, -time) %>%
  ggplot(aes(x = time, y = Value, color = Variable)) +
  geom_line(size = 1) +
  geom_point(color = "black") +
  theme_classic() +
  scale_color_brewer(palette = "Set1")
```



Value
30
20
10
0
0.0
2.5
5.0
7.5
10.0
time
Variable
X
Y

2 Continuous Time

Consider the systems of linear ordinary differential equations:

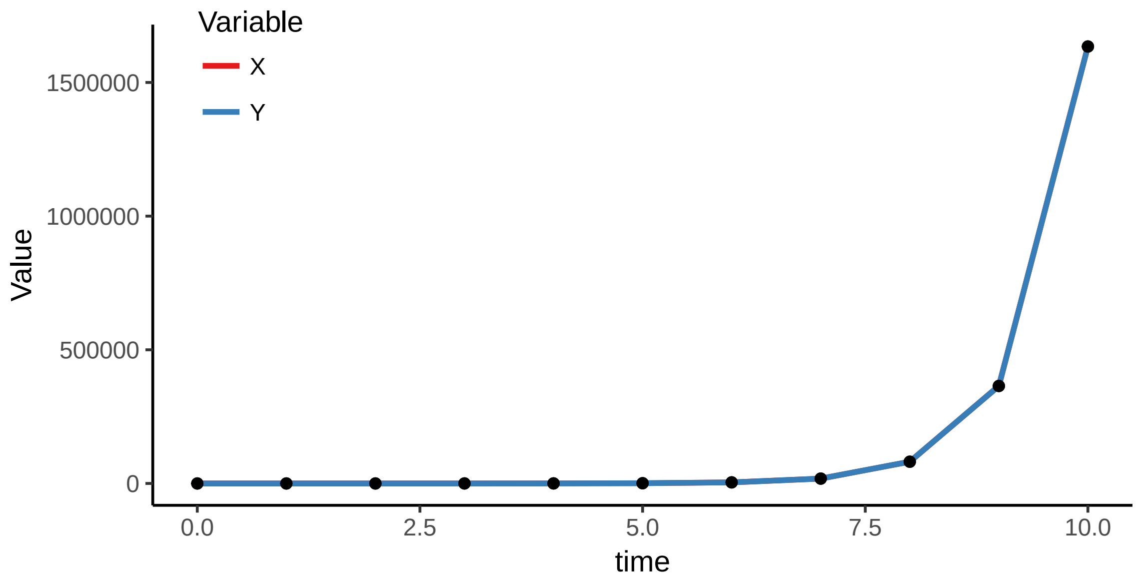
$$\begin{aligned}\frac{dx}{dt} &= x + 0.5y \\ \frac{dy}{dt} &= 0.5x + y\end{aligned} \tag{7}$$

with initial conditions:
$x_0 = 0$
$y_0 = 1$

2.1 Calculate and plot the *numerical solution* to Eq 7 with $t \in (0, 10)$

```
# Define a function to pass to lsoda
system <- function(t, values, params){

  x <- values[1]
  y <- values[2]

  a <- params[1]
  b <- params[2]
  c <- params[3]
  d <- params[4]

  dxdt <- a*x + b*y
  dydt <- c*x + d*y

  return(list(c(dxdt, dydt)))
}
```

```
t <- seq(0, 10)
x_0 <- 0
Y_0 <- 1
initial_values <- c(X_0, Y_0)

a <- 1
b <- 0.5
c <- 0.5
d <- 1
params <- c(a, b, c, d)

lsoda(y = initial_values, times = t, func = system, parms = params) %>%
  as.data.frame() %>%
  magrittr::set_colnames(c("time", "X", "Y")) %>%
  gather(Variable, Value, -time) %>%
  ggplot(aes(x = time, y = Value, color = Variable)) +
  geom_line(size = 1) +
  geom_point(color = "black") +
  theme_classic() +
  scale_color_brewer(palette = "Set1") + theme(legend.position = c(0.1, 0.9))
```



2.2 Calculate and plot the *analytical solution* to Eq 7 with $t \in (0, 10)$