Table of Contents

Leslie matrix

In this example we will consider a seven-stage model for the life cycle of an animal according to the following table:

Stage number	Description (age in years)	Annual Survivorship	Eggs Laid per Year
1	Eggs, hatchlings (< 1)	0.6747	0
2	Small juveniles (1 – 7)	0.7857	0
3	Large juveniles (8 – 15)	0.6758	0
4	Subadults (16 – 21)	0.7425	0
5	Novice breeders (22)	0.8091	127
6	First-year remigrants (23)	0.8091	4
7	Mature breeders (24 – 54)	0.8091	80

This will result in the following Leslie matrix:

$$L = \begin{pmatrix} 0 & 0 & 0 & 0 & 127 & 4 & 80 \\ 0.6747 & 0.7370 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.0486 & 0.6610 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0147 & 0.6907 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0518 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.8091 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.8091 & 0.8089 \end{pmatrix}$$

With initial condition:

$$X_0 = [200000, 130000, 100000, 70000, 500, 400, 1100]^T$$

```
A = [0      0      0      0      127    4      80;
     0.6747 0.7370 0      0      0      0      0 ;
     0      0.0486 0.6610 0      0      0      0 ;
     0      0      0.0147 0.6907 0      0      0 ;
     0      0      0      0.0518 0      0      0 ;
     0      0      0      0      0.8091 0      0 ;
     0      0      0      0      0      0.8091 0.8089];
x0 = [200000 130000 100000 70000 500 400 1100].';
```

Note that we will consider the unforced system in this example.

Defining data

We will start by generating new data. This can be done as follows $x(t+1) = Ax(t) + Bu(t)$. Since we are considering the unforced system we get $x(t+1) = Ax(t)$.

```
% Number of data points we want to consider
datapoints = 20;
% Allocating memory for data
X = zeros(size(x0,1), datapoints);
% Generating the data
X(:,1) = x0;
for idx = 2:datapoints
    X(:,idx) = A * X(:,idx - 1);
end
```

System identification

Complete data

To check if the data is informative for system identification we can use the `[bool, A] = isIdentifiable(X)` function.

```
[bool, A_s] = isInformIdentification(X);
disp("Is the data informative for system identification?");
```

```
Is the data informative for system identification?
```

```
disp(bool);
```

```
     1
```

As we can see the data is informative for system identification. Now we can verify that the returned system is indeed the same as the provided system. We will verify this by considering the error / difference between the systems.

```
disp("Difference between actual A and calculated A_s");
```

```
Difference between actual A and calculated A_s
```

```
disp(A-A_s);
```

```
   1.0e-12 *

   -0.0167    0.0153   -0.0840    0.1670   -0.2274    0.1315    0.0568
   -0.0016    0.0012   -0.0054    0.0178    0.0103    0.3073   -0.0568
   -0.0001   -0.0001   -0.0003    0.0024    0.0035    0.0065   -0.0036
   -0.0000   -0.0000    0.0000   -0.0001   -0.0027    0.0108   -0.0036
   -0.0000   -0.0000    0.0000    0.0000   -0.0002    0.0007   -0.0001
   -0.0001    0.0001   -0.0006    0.0012   -0.0014    0.0005    0.0008
   -0.0000    0.0000   -0.0000    0.0001    0.0004    0.0019    0.0007
```

As we can see the matrices are equal up to working precision.

What is the impact of sampling moment?

We will now consider what happence when we skip the first few entries. We want that the result stays consistent independant of when we sample the system.

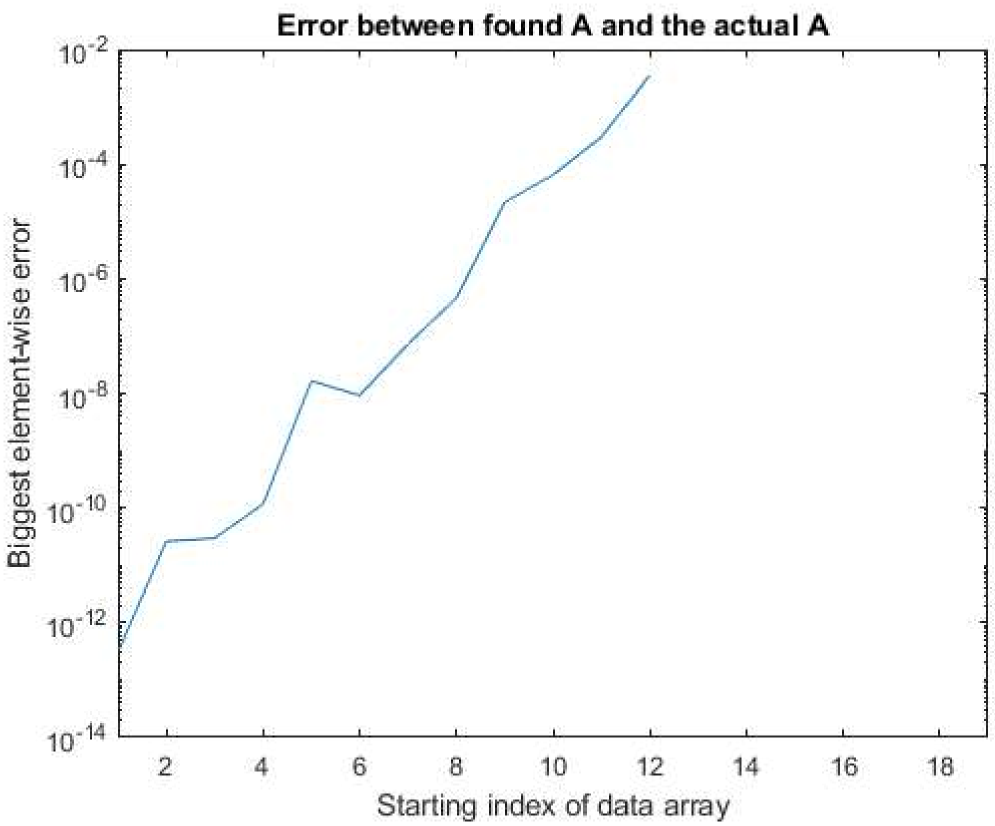
```
% Number of data points we want to consider
m = 19;
elements_skipped = 1:m;
% Allocating memory for data
max_difference = zeros(1,m);
% Generating the data
```

```
for n = elements_skipped
    [bool, A_moment] = isInformIdentification(X(:, n:end));
    if bool
        max_difference(n) = max(max(abs(A - A_moment)));
    else
        max_difference(n) = inf;
    end
end
```

TODO: Find out why the error is increasing (note that after 12 the data is not informative for identification)

However informativity is not enough, since the solution should be unique independant of when we sample out data. To verify this we will look at the biggest element-wise error between the true A and the generated A_{moment}.

```
plot(elements_skipped, max_difference)
set(gca, 'YScale', 'log')
title(gca,"Error between found A and the actual A")
xlabel('Starting index of data array')
ylabel('Biggest element-wise error')
xlim([1 m])
```



As we can see the calulated A will get less and less ***accurate due to ....***

What is the impact of sampling rate?

For this part of the example we will consider what happence if we sample our data on a fixed time, e.g. every 2 iterations. We whould assume this will result in the matrix A^2.

```
[bool, A_rate] = isInformIdentification(X(:,1:2:end));
disp("Is the data informative for system identification?");
```

```
Is the data informative for system identification?
```

```
disp(bool);
```

```
1
```

```
disp("Largest difference between A^2 and calculated A_rate");
```

```
Largest difference between A^2 and calculated A_rate
```

```
larges_difference = max(max(abs(A^2 - A_rate)))
```

```
larges_difference = 1.5859e-10
```

As we can see, this does indeed result in the `A^2` matrix. However, we do need the sampling rate to be consistent. ***add (text or example)***

Stability

We can also check if the data is informative for stability. To do this we use the function `bool = isInformStable(X)`. Since we also have the true system we can also check the result by comparing it to `bool = isStableD(A)`, which returns if the descrete time system is stable.

```
disp("Is the true system stable?");
```

```
Is the true system stable?
```

```
disp(isStableD(A));
```

```
1
```

```
disp("Is the data informative for stability?")
```

```
Is the data informative for stability?
```

```
disp(isInformStable(X));
```

```
1
```