Homework Assignment № 5

Kevin Mack, Clarkson University 11/08/2017

Problem Statement:

In this exercise we consider an Euler beam with pinned boundary conditions on both ends. A chaotic element is bouncing in the middle of the beam. The chaotic element, which is described by Equation 1, is a "hanging from the beam" and is derived from the solution of the Rossler equations. The equation for the beam is given by,

$$EI\left(\frac{d^4w}{dx^4}\right) = -\mu\left(\frac{d^2w}{dt^2}\right) + q(x,t) \tag{1}$$

with boundary conditions,

$$w(0,t) = w(L,t) = 0, \tag{2}$$

and initial conditions,

$$w(x,0) = f(x). \tag{3}$$

DMD Solution:

The Dynamic Mode Decomposition (DMD) solution is designed to allow for equation-free modeling of a dynamic system. The DMD uses only a data matrix that is created by sampling the system across a determined space and time. The DMD requires only that the method of sampling uses a uniform time between samples, Δt, and that the data matrix $\mathbf{X}$ is of low dimensionality (low rank). Given such conditions are met, the DMD algorithm can give the state of a dynamic system at any time, t (even predicting future states of the system). However, if a system is not sampled for long enough in time, the error between the true state of the system and the prediction from the DMD can grow as time increases. The growth of this error is shown to be exponential in Figure 20.2 of Kutz's *Data-Driven Modeling and Scientific Computation*.

Mathematical Formulation:

First, create a data matrix $\mathbf{X}$ by deciding how often the data will be sampled in time and space. Given N number of spatial points, and M number of "snapshots" taken in time, then $\mathbf{X} \in \mathbb{R}^{N\times M}$. The DMD method is used to approximate the modes of the Koopman operator, which is a linear and infinite dimensional operator that represents nonlinear and finite dimensional dynamics. This means that the decomposition will give the growth rates and frequencies associated with each mode of the dynamics, even if they are nonlinear. The Koopman Operator, given by $\mathbf{A}$, is used to map the data from one snapshot to the next. This is defined by,

$$\mathbf{x}_{j+1} = \mathbf{A}\mathbf{x}_j, \tag{4}$$

where the vector $\mathbf{x}_j$ is an N-dimensional vector of the data points collected at time j. To construct the Koopman operator that represents the collected data, the matrix $\mathbf{X}_1^{M-1}$ is given by:

$$\mathbf{X}_1^{M-1} = \left[\mathbf{x}_1 \; \mathbf{x}_2 \; \mathbf{x}_3 \ldots \mathbf{x}_{M-1}\right]. \tag{5}$$

Using (4) and (5) we can define,

$$\mathbf{X}_1^{M-1} = \left[\mathbf{x}_1 \; \mathbf{A}\mathbf{x}_1 \; \mathbf{A}^2\mathbf{x}_1 \ldots \mathbf{A}^{M-2}\mathbf{x}_1\right]. \tag{6}$$

With this formulation we attempt to fit the first $M-1$ data collection points using the Koopman Operator. Further, we use the SVD in order to perform dimensionality reduction on $\mathbf{X}_1^{M-1}$. Here it can be seen that if the data matrix is of full rank, then the DMD will fail at reconstructing the dynamics. However, if the data is of low dimensionality then we are able to project a future state for the system. The Koopman operator again allows for the prediction future states by solving a matrix inversion of the form,

$$\mathbf{A}\mathbf{X}_1^{M-1} = \mathbf{X}_2^{M}. \tag{7}$$

Unfortunately, $\mathbf{A}$ is a matrix with unknown elements, so instead the low-rank matrix $\widetilde{\mathbf{S}}$ is to be computed and used instead,

$$\widetilde{\mathbf{S}} = \mathbf{U}^*\mathbf{X}_2^{M}\mathbf{V}\mathbf{\Sigma}^{-1}. \tag{8}$$

The initial conditions can be projected onto the eigenvectors and the eigenvalues of $\widetilde{\mathbf{S}}$, using the pseudo-inverse, in order to reconstruct the dynamics of the system.

DMD Approximation of a Chaotic Beam:

The chaotic beam problem is solved by a set of partial differential equations, and in this section the equation-free DMD solution is attempted. The number of time "snapshots" will be set to $N = 17$, and the number of spacial samples is set to $M = 200$. The code that solves the PDE for the chaotic beam is given in the first listing, and the DMD solution is given by the second listing.

Listing 1: Matlab code – PDE solver for chaotic beam

```
% Forced damped beam vibration solver via finite-difference
% Author: Leonardo Antonio de Araujo
% e-mail: leonardo.aa88@gmail.com
clear; close all
```

```
clc;

E=2.1E11; % Young modulus
rho=76.8; % Density
A=0.01*0.01; % Cross-section area
I=(0.01*0.01^3)/12; % Second moment of inertia
L=5; % Length
c=1; % Damping

nx=17; % Number of points
dx=L/nx;

dt=1E-5; % Time step
tf=2.5E-1; % Final time

x=linspace(0,L,nx);

% Initial conditions
for i=1:(nx)
    w(i,1)=0;
    w(i,2)=0;
end

%Rossler
xx=1; yy=1; zz=1; a=0.2; b=0.2; c=5.7; h=500*dt; xk=[];
% Acting force
for t=1:(tf/dt)
    for i=1:nx
     f(i,t)=-10;
    end
end

for t=1:(tf/dt-2)
    [t (tf/dt-2)];
    % Boundary Conditions
    w(1,t)=0;
    w(2,t)=0;
    w(nx,t)=0;
    w(nx-1,t)=0;

    xxt=xx+h*(-yy-zz); yyt=yy+h*(xx+a*yy); zzt=zz+h*(b+zz*(xx-c)); xx=xxt; yy=yyt; zz=zzt;
        w(11,t)=w(11,t)+5*h*xx;
        w(6,t)=w(14,t)-.5*h*yy;
    xk=[xk; xx];

    for i=3:(nx-2)
```

```
        w(i,t+2)=((dt^2)/(rho*A))*(f(i,t)-(E*I/(dx^4))*(w(i-2,t)-4*w(i-1,t)+6*w(i,t)-4*w(i+1,t)+w(i+2,t)))+2*w(i,t+1)-w(i,t)-((c*dt)/(rho*A))*(w(i,t+1)-w(i,t));
    end

    xlabel('length (m)');
    ylabel('displacement (m)');
    set(gca,'FontSize',12)
    axis([0 L -0.1 0.1]);
end

figure; imagesc(w)
figure; surf(w(:,1:20:end))
w=w(:,1:200:end);

t = (1:200:t)*dt;
dt = 200*dt;
x = (1:nx)*dx;
save('beamChaotic_new.mat','w','t','dt','x','dx')
```

Listing 2: Matlab code – DMD solution

```
clear
load('beamChaotic_new.mat')

X = w;
[N,M] = size(X);
u  = X(:,1);          % intial condition
X1 = X(:,1:end-1);
X2 = X(:,2:end);

%DMD implementation
[U,Sigma,V] = svd(X1,'econ');
Sigma = Sigma + diag(10^-10*ones(size(Sigma,1),1));
Sigma_inv = diag(1./diag(Sigma));
S =  U'*X2*V*Sigma_inv;
[ev,D] = eig(S);
mu = diag(D);
omega = log(mu)/(dt);
Phi = U*ev;

y0 = pinv(Phi)*u; %pseudo-inverse intial conditions
u_modes = zeros(size(V,2), length(t));
for iter = 1:M
    u_modes(:,iter) = (y0.*exp(omega*t(iter)));
end
```

```
max_modes = 5;
Phi_new = Phi(:, 1:max_modes);
u_modes_new = u_modes(1:max_modes,:);

u_dmd_trunc = Phi_new*u_modes_new;
u_dmd = Phi*u_modes;
gamma = 19;
N = 6;
u_dmd_test = N*u_dmd.*exp(gamma*t);

D_diag = diag(D);
D_new = D_diag(1:max_modes);

figure
hold all
subplot(2,3,1)
waterfall(real(w'))
xlabel('x')
ylabel('time')
zlabel('|u|      ')
set(get(gca,'zlabel'),'rotation',0)
title('PDE')

subplot(2,3,2)
waterfall(real(u_dmd_trunc.'))
xlabel('x')
ylabel('time')
zlabel('|u|      ')
set(get(gca,'zlabel'),'rotation',0)
title('DMD')

subplot(2,3,3)
waterfall(abs(Phi_new'))
xlabel('x')
ylabel('modes')

th = 0:pi/50:2*pi;
x = 0; y = 0; r = 1;
x_circ = r * cos(th) + x;
y_circ = r * sin(th) + y;

subplot(2,3,4)
hold on
grid on
grid minor
axis([-1.25 1.25 -1.25 1.25 ])
```

```
axis square
plot(D_new,‘ko‘)
plot(x_circ, y_circ,‘r-‘)
xlabel(‘Real (\mu)‘)
ylabel(‘Im (\mu)‘)

subplot(2,3,5)
hold on
for i = 1:max_modes
    plot(0:6,ones(7)*abs(u_modes(i)))
end
xlabel(‘t‘)

subplot(2,3,6)
hold on
plot(t, real(u_modes_new(1,:)), ‘k-‘)
plot(t, imag(u_modes_new(1,:)), ‘k--‘)
xlabel(‘t‘)
axis([0 0.25 -100 100])
```

From Figure 1, it can be seen that although the DMD algorithm seems to capture the dynamics of the system, the reconstructed system does not accurately represent the original model. Upon further inspection it is clear that the modes of the DMD reconstruction are subject to decay over time, causing the reconstructed signal to lose its accuracy. In order to further inspect what may be causing this decay, the example from Kutz‘s book will compared to the chaotic beam problem. Figure 2 shows the example from the book, which models the non-linear Schrödinger equation. In particular, the DMD modes in the bottom left image of the figure show that most of the modes are located close to, or just outside of the unit circle. In contrast, for the chaotic beam problem the DMD modes are mostly located closer to the center of the unit circle, with at least one falling directly at the origin. It is known that modes inside the unit circle can cause the system to decay significantly. In order to remove the decay from the model, several of the modes that are closest to the origin will be removed. However, it is important not remove modes with high amounts of energy from the system, since the model will suffer from not including the dominant modes. The results from this attempted fix are displayed in Figures 3 and 4. It can be seen that although most of the modes with closest distance to the origin have been removed, the model is still subject to decay. This is most likely due to the fact that all of the modes are still inside of the unit circle, with none standing outside to balance the system.

It is clear from Figures 3 and 4 that removing the DMD modes with the smallest eigenvalues will not fix the problem of decay in the reconstruction of the system. The issue remains because if the eigenvalues of all the modes are inside the unit circle, then there are no modes that can balance and stabilize the system. It can be seen by comparing the original system (Figure 1) with all of the modes included, and the system with only three modes (Figure 4) decay at different rates, with the three mode system decaying slower.

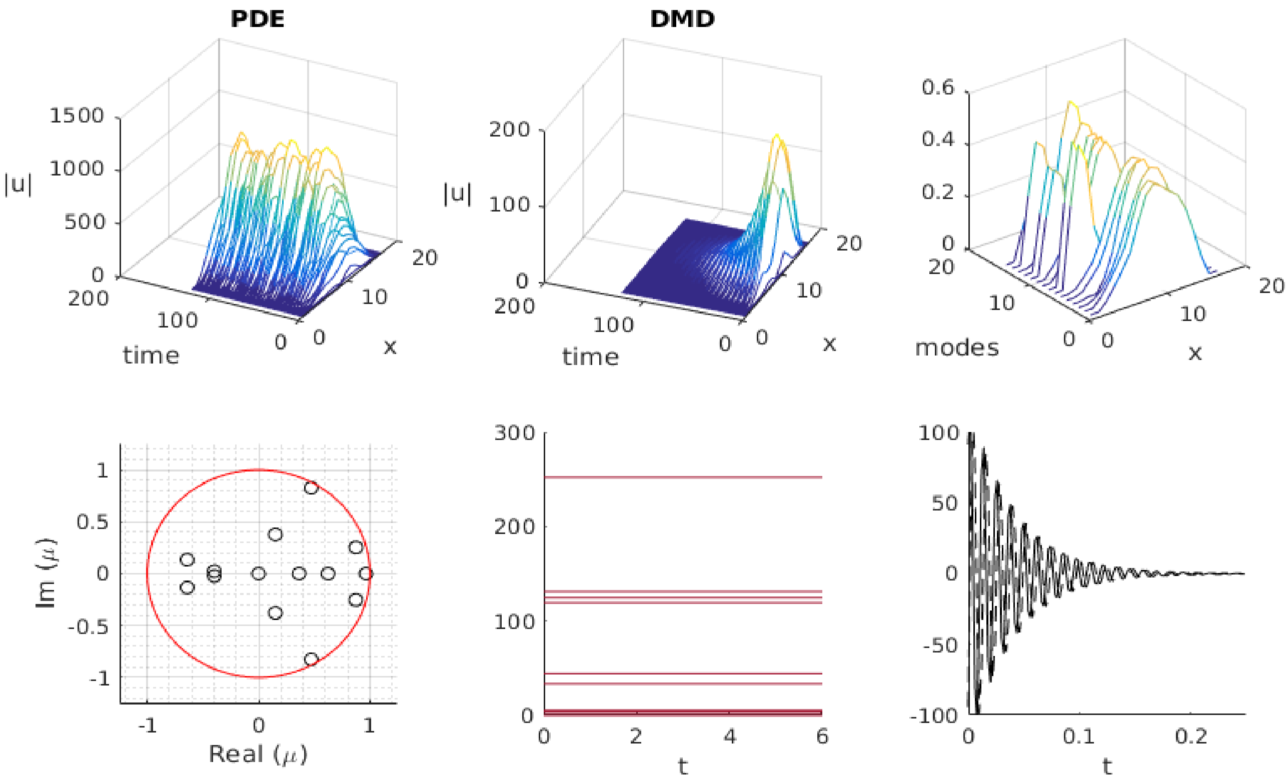

Figure 1: The shows the reconstruction of the chaotic beam system by DMD. The absolute value of the original PDE (top left) is a chaotic system that has been modeled using the DMD (top middle). The DMD modes are represented in the top right image, with the time dynamics represented in the bottom middle. The eigenvalues of the DMD modes are represented in the bottom left image, with the bottom right image being the time evolution of the first mode (real is solid, imaginary is dotted). In comparison with 2 the eigenvalues of the DMD modes show that the modes will decay. This is what causes the reconstruction of the system to decay over time.

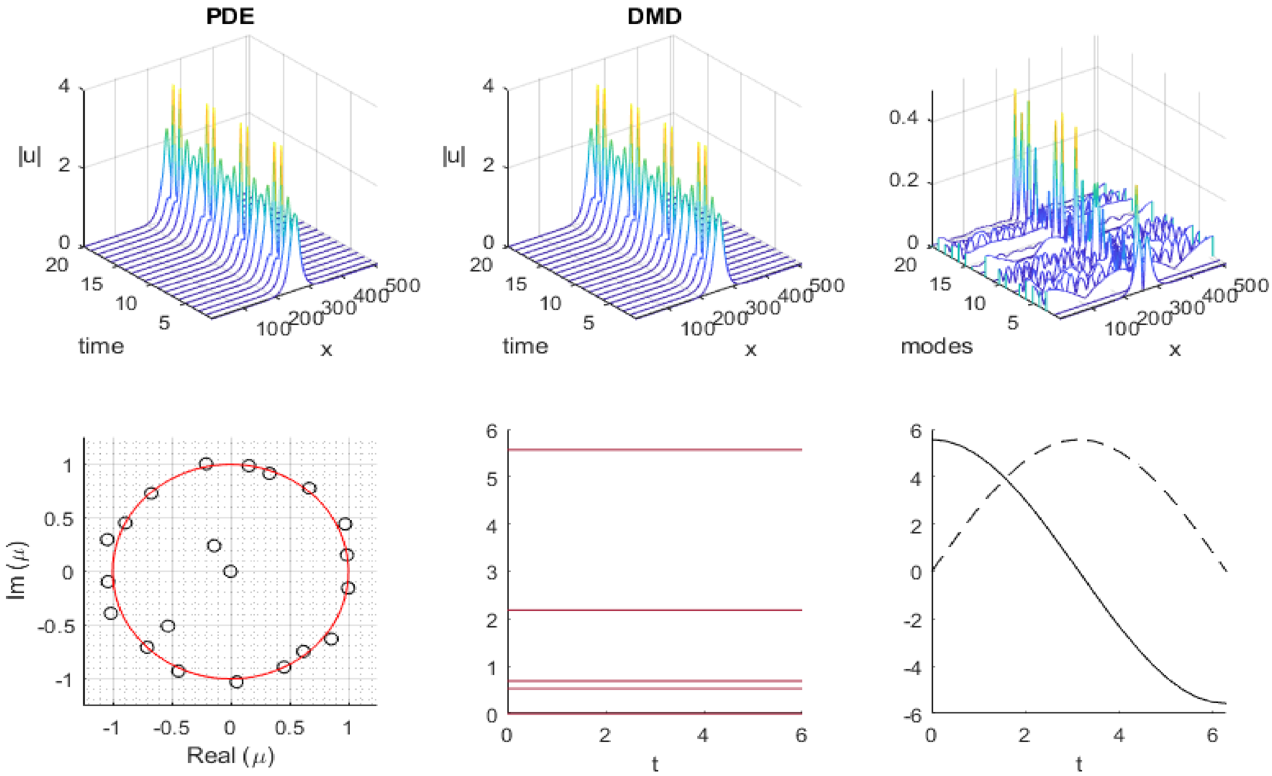

Figure 2: This figure shows the effectiveness of the DMD method for reconstructing a chaotic system based on the non-linear Schrodinger equation. The individual figures are the same as in Figure 1, with clear differences in the eigenvalues of the DMD modes (bottom left). In this case, DMD works phenomenally well at reconstructing the dynamics of the system.

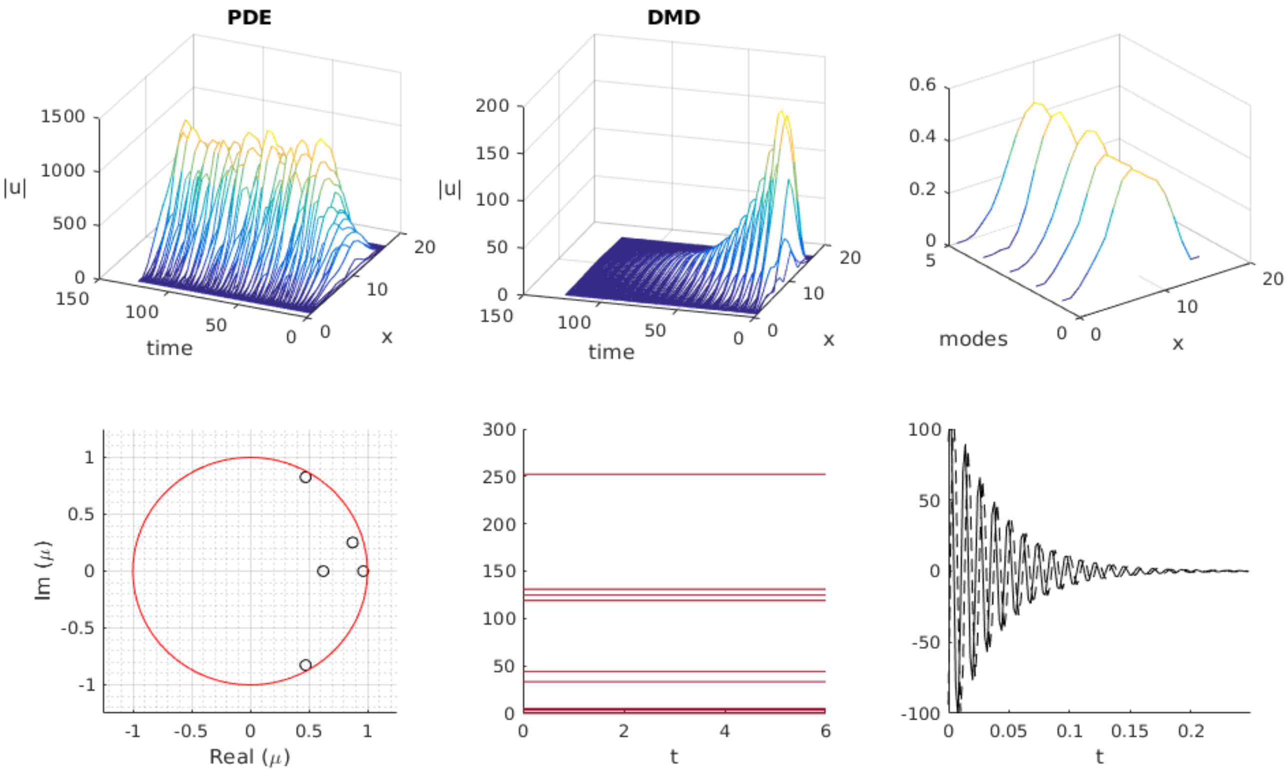

Figure 3: Here the DMD analysis only uses the most prevalent 5 modes to reconstruct the original PDE. Since all of the modes are still inside the unit circle, the reconstructed system still suffers from the issue of decay over time.

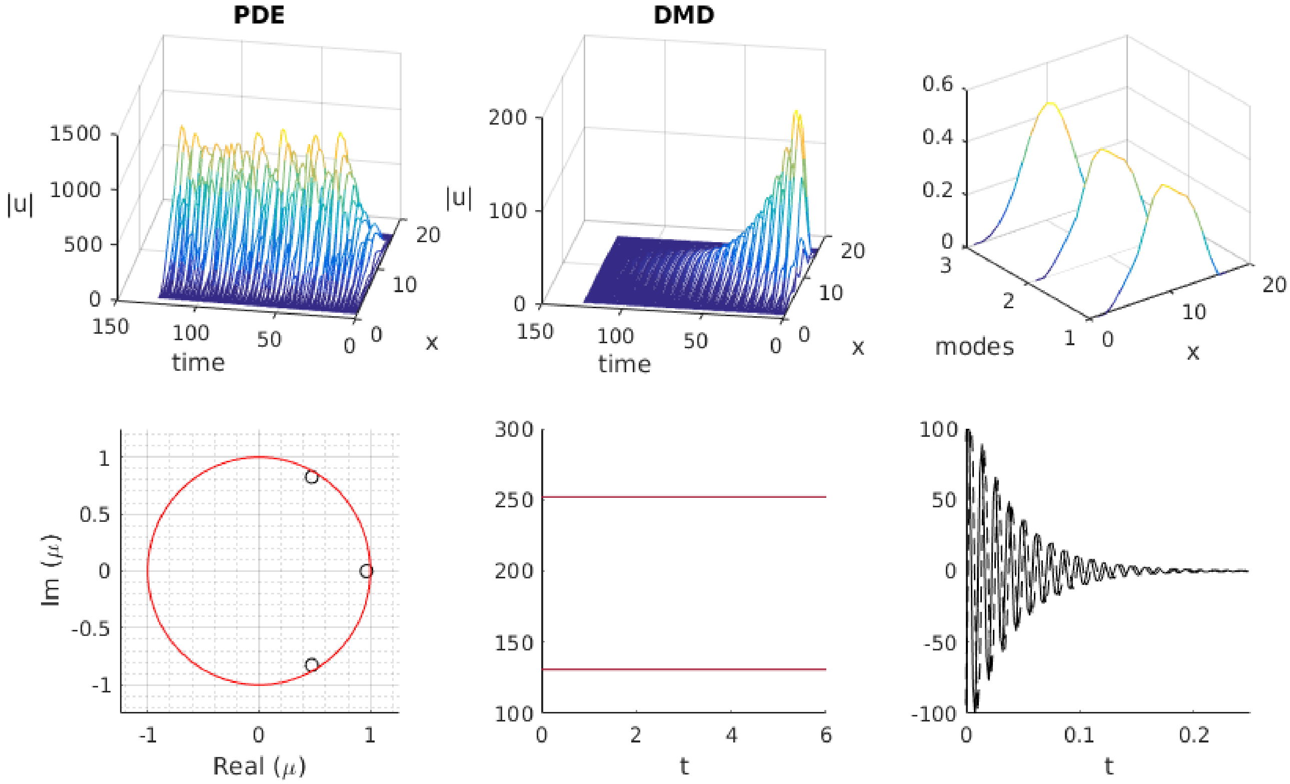

Figure 4: In this case the DMD analysis only uses the 3 strongest modes to reconstruct the PDE for the chaotic beam. However, it suffers from the same problem as the previous figures where all of the modes lay inside the unit circle. Even though the only remaining modes have eigenvalues near the unit circle, they are still causing decay over time.