MATHEMATICAL AND COMPUTER MODELING OF NONLINEAR BIOSYSTEMS I

COMPUTER LABORATORY VI: Kinetics of single enzyme and product, kinetics of multiple enzymes/prodicts/substrates

Ph. D. Programme 2013/2014

PhD STUDIES

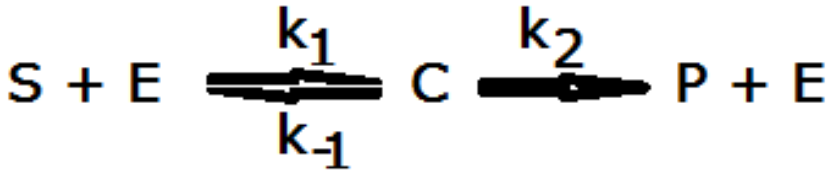
KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Project co-financed by European Union within the framework of European Social Fund

Michaelis-Menten kinetics

Let us consider the following biochemical reaction



where S is the substrate, E is the enzyme, C is the enzyme-substrate complex, and P is the reaction product.

Michaelis-Menten kinetics is one of the best-known models of the above reaction

$$\frac{d[P]}{dt} = \frac{V_m[S]}{K_m + [S]},$$

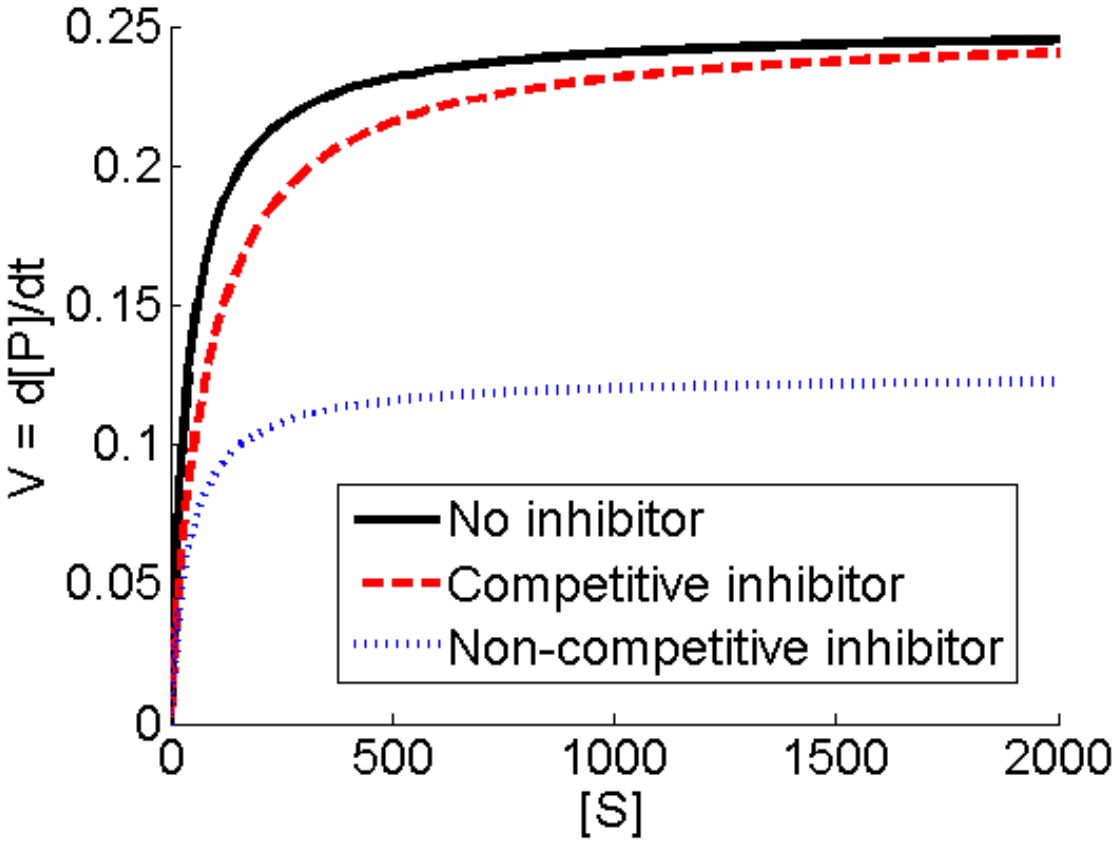
where $V_m = k_2 E_0$ (maximum reaction rate) and $K_m = \frac{k_{-1}+k_2}{k_1}$ (substrate concentration at which reaction rate is half of V_m).

We have that $\frac{d[P]}{dt} = -\frac{d[S]}{dt}$.

Influence of inhibitors

Enzymes can be inhibited in two ways:

- competitively — substrate and the inhibitor compete for the biding site of the enzyme (K_m increases);
- non-competitively — inhibitor binds to other site of the enzyme and reduce its efficacy (V_m decreases).



Estimation of kinetic constants I

Lineweaver-Burk plot

Taking the reciprocal of the reaction rate in Michaelis-Menten kinetics gives

$$\frac{1}{V} = \frac{K_m + [S]}{V_m[S]} = \frac{K_m}{V_m}\frac{1}{[S]} + \frac{1}{V_m}.$$

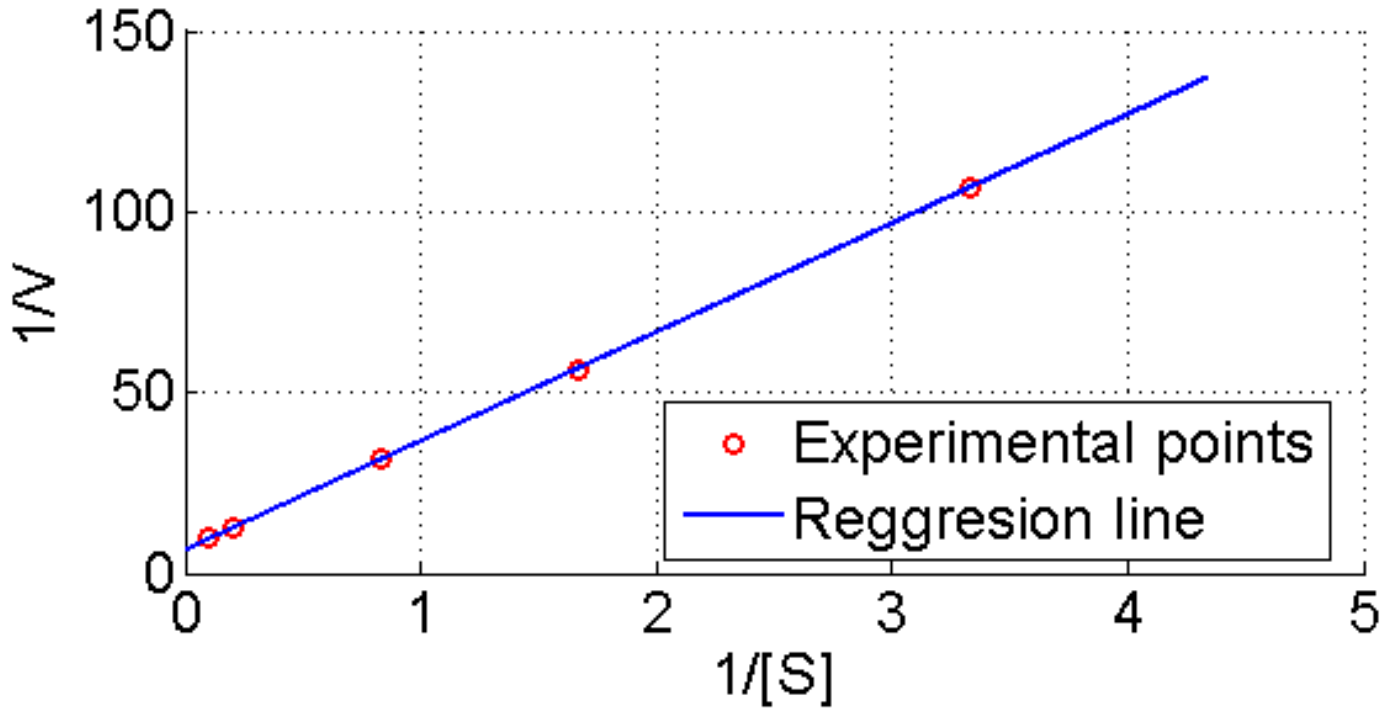
the above relationship is linear with respect to the $1/V$ and $1/[S]$.

We perform experiments with the same concentration of enzyme and with different amount of substrate. We measure concentration of the product frequently and approximate the V (the derivative).

We plot experimental points in $1/V$ and $1/[S]$ coordinates and perform linear regression.

Estimation of kinetic constants I

Lineweaver-Burk plot

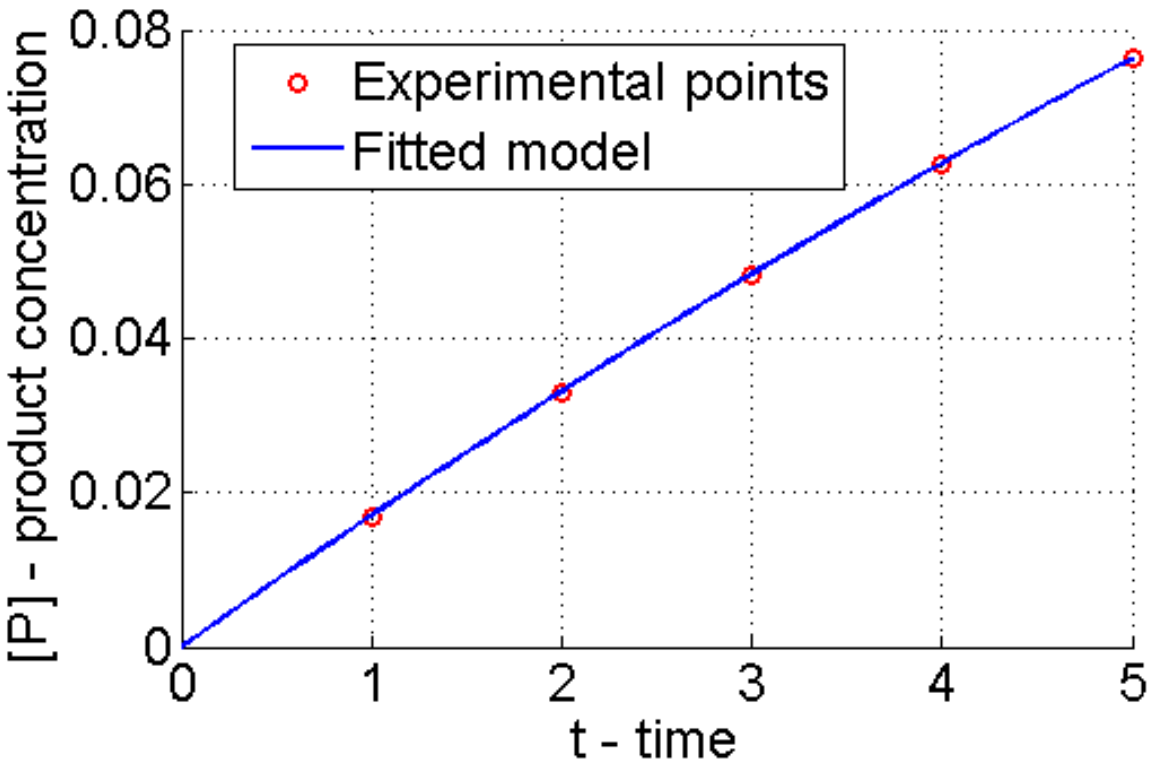


Estimation of kinetic constants II

We may also directly fit the solution of the Michaelis-Menten kinetics

$$\frac{d[P]}{dt} = \frac{V_m[S]}{K_m + [S]},$$

to the experimental measurements.



Case study

We are looking for the best inhibitor for enzymatic reaction that we use in our biochemical reactor.

Our lab technicians performed experiments using 100 different substances that might have inhibitory effect.
In each case they used initially 9.6, 4.8, 1.2, 0.6, 0.3 mM of the substrate, same amount of the enzyme and same amount of the substance suspected to be inhibitor.
They always measured the product amount at 1, 2, 3, 4, 5 minutes.

They've send us the file with all the results. **We need to decide which of the substances are true inhibitors, what is the pathway of inhibition (competitive, non-competitive) and pick the strongest competitve inhibitor.**

Step 1 - read data to MATLAB

1. Download a csv file containing experimental measurements from:

   `http://www.mimuw.edu.pl/~poleszczuk/enzymeData.csv`

   File structure:
   - each experiment in separate column
   - control experiment (no inhibitor) in the first column
   - measurements for 9.6, 4.8, 1.2, 0.6, 0.3 mM in consecutive rows (blocks of five).

2. Implement MATLAB function which reads data from downloaded file
   - input arguments: file name, number of measurements for each experimental setting, and experiment number;
   - output variable: matrix where columns are for different initial concentrations and measurements are in the rows.

Step 1 - solution

```
function data = loadData( fileName, noMeasurements, expNum )

    data = csvread(fileName);
    if expNum<=size(data,2)
        data = reshape(data(:,expNum),noMeasurements,[]);
    else
        data = [];
        warning('Experiment with a specified number...
         does not exist!')
    end

end
```

Step 2 - Lineweaver-Burk plot based estimation

Implement MATLAB function which estimates the kinetic constants using Lineweaver-Burk method.

Input arguments:

- considered initial substrate concentration,
- moments of product measurements,
- measured values (matrix from step 1),
- optional argument indicating if to plot Lineweaver-Burk plot

Output arguments: kinetic contants (K_m and V_m)

Step 2 - solution (part I)

```
function [Km, Vmax] = LBcalc(initC,measurements,data,plotLB)

if nargin<4
   plotLB = false;
end

data = [zeros(size(initC)); data];
measurements = [0 measurements];

%estimate derivatives
derivs = mean((data(2:end,:)-data(1:end-1,:))...
          ./repmat(diff(measurements)',1,length(initC)),1);
Y = 1./derivs;
X = 1./initC;

%fitting
F = polyfit(X,Y,1);
```

Step 2 - solution (part II)

```
Km = 1/(F(2)/F(1));
Vmax = 1/F(2);
if plotLB
     plotLBplot(X,Y,F)
end
end

function plotLBplot(X,Y,F)
    clf
    hold on
    plot(X,Y,'LineStyle','none','Marker','o','Color','r');
    plot(xm, F(1)*linspace(0,max(X),2)+F(2),'LineWidth',2)
    hold off
    xlabel('1/[S]')
    ylabel('1/V')
    grid on;
    legend('Experimental points','Reggresion line')
end
```

Step 3 - performing the analyis

Write MATLAB script in which for each experimental settings kinetics constants ar estimated.

Plot LB plot for the control experiment.

Decide which of the considered substances are true inhibitors (make the division into types) — make a bar plot with classification.

Find the best competitive inhibitor.

Step 3 - solution (part I)

```
clear all

experiments = 1:101;

initC = [9.6 4.8 1.2 0.6 0.3];
measurements = 1:5;

Km = zeros(1,length(experiments));
Vmax = Km;

for i=1:length(experiments)
   data = loadData( 'enzymeData.csv', length(measurements),...
          experiments(i) );
   if i==1
    [Km(i), Vmax(i)]=LBcalc(initC, measurements, data, true);
   else
    [Km(i), Vmax(i)]=LBcalc(initC, measurements, data);
   end
end
```

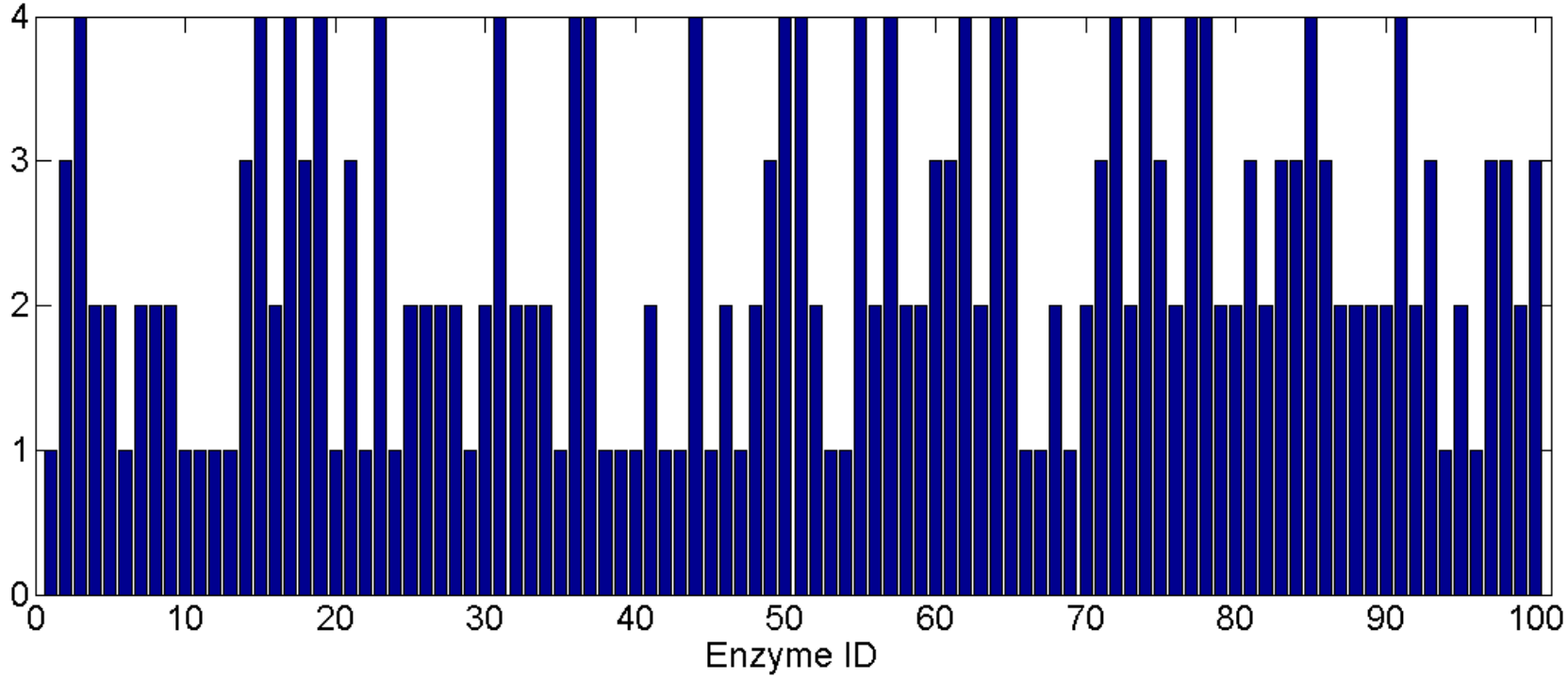
Step 3 - solution (part II)

```
tolK=1; tolV = 0.1;
Out = (abs(Km-Km(1))<tolK) & ...
      (abs(Vmax-Vmax(1))<tolV);
Out = Out+2*((abs(Km-Km(1))>=tolK) & ...
      (abs(Vmax-Vmax(1))<tolV));
Out = Out+3*((abs(Km-Km(1))<tolK) & ...
      (abs(Vmax-Vmax(1))>=tolV));
Out = Out+4*((abs(Km-Km(1))>=tolK) & ...
      (abs(Vmax-Vmax(1))>=tolV));

figure(2)
bar(Out(2:end))
```

Result

1 — no inhibition, 2 — competitive inhibitor, 3 — non-competitive inhibitor, 4 — mixed type inhibitor



The best competitive inhibitor ID: 4

Found using command: [m, id]=max(Km(Out==2)))