

課題

今日の流れ

今日使うファイル類は下記にあります

<http://kurodalab.bs.s.u-tokyo.ac.jp/class/Summer/2019/Day3/>

I. 講義

0. Introduction

1. 細胞運命決定機構におけるシグナル分子Ras, Rap1
2. パラメータ推定
3. 最適化

II. 課題

1. Ras, Rap1のODEモデルを作成
2. 1つ or 2つの未知パラメータを推定
3. 4つの未知パラメータをEPで推定

「課題」「発展課題」ができたならTAのチェックを受けてください。

14:30までに終了したい場合は「発展課題」まで終わらせてください。

14:30以降に終了する場合は「課題」まで終わらせてください。

終わったら、授業の途中で帰っても構いません。

課題への質問に関する注意

質問がある場合には、手を挙げてTAを呼んでください。

その際には、

「課題をどうやったらいいかわからない」

「エラーが出て上手くいかない」

といった漠然とした質問ではなく、

「課題の意図が分からず、最終的にどんな結果が
得られれば良いかがわからない」

「XXXといった操作を実行したいが、どのようにコードを
書けばいいかわからない」

「XXXを実行するために、YYY、ZZZというコードを
書いたが、エラーが出る（予想と違う）」

などできるだけ**具体的に**質問するようにお願いします。

参考：便利なMATLAB関数

[M, id] = min(A)

入力 A : ベクトル or 行列

出力 M : Aの最小値

Aが行列の場合は列ごとの最小値

id: 最小値のインデックス

S = sum(A, dim)

入力 A : ベクトル or 行列

dim : 次元dimに沿って総和を計算

出力 S : Aの総和

詳しくは、Command Window上で”help 関数名”

または、Google等で”MATLAB 関数名”で検索

計算の途中を見たいときは？ → デバッグ機能と変数表示
どこまで上手くいっていて、どこで間違っているかを確認

II. 課題

今日使うファイル類は下記にあります

<http://kurodalab.bs.s.u-tokyo.ac.jp/class/Summer/2019/Day3/>

1. Ras, Rap1のODEモデルを作成

2. 1つ or 2つの未知パラメータを推定

3. 4つの未知パラメータをEPで推定

「課題」「発展課題」ができたならTAのチェックを受けてください。

14:30までに終了したい場合は「発展課題」まで終わらせてください。

14:30以降に終了する場合は「課題」まで終わらせてください。

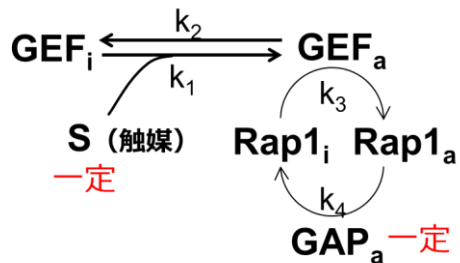
終わったら、授業の途中で帰っても構いません。

課題1-1

コード：なし
データ：なし

Rap1モデルの S , GEF_a , GAP_a , $Rap1_a$ の常微分方程式 (ODE) を記述せよ。また、Rasモデルの S , GEF_a , GAP_a , Ras_a のODEを記述せよ。ただし、 GEF_i , GAP_i , $Rap1_i$, Ras_i は使用しないこと。

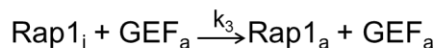
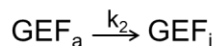
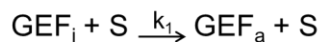
シンプルRap1モデル



不活性型と活性型の総和は保存

$$\begin{aligned} GEF_i + GEF_a &= 1 \\ GAP_i + GAP_a &= 1 \\ Rap1_i + Rap1_a &= 1 \end{aligned}$$

反応式は、



... のように書ける。

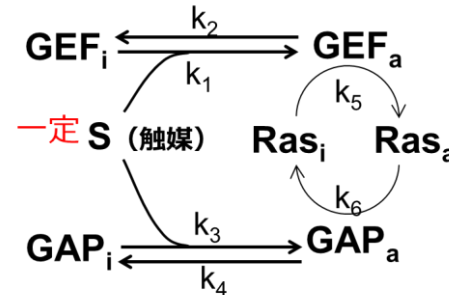
$$\frac{dS}{dt} = ??$$

$$\frac{dGEF_a}{dt} = ??$$

$$\frac{dGAP_a}{dt} = ??$$

$$\frac{dRap1_a}{dt} = ??$$

シンプルRasモデル



不活性型と活性型の総和は保存

$$\begin{aligned} GEF_i + GEF_a &= 1 \\ GAP_i + GAP_a &= 1 \\ Ras_i + Ras_a &= 1 \end{aligned}$$

$$\frac{dS}{dt} = ??$$

$$\frac{dGEF_a}{dt} = ??$$

$$\frac{dGAP_a}{dt} = ??$$

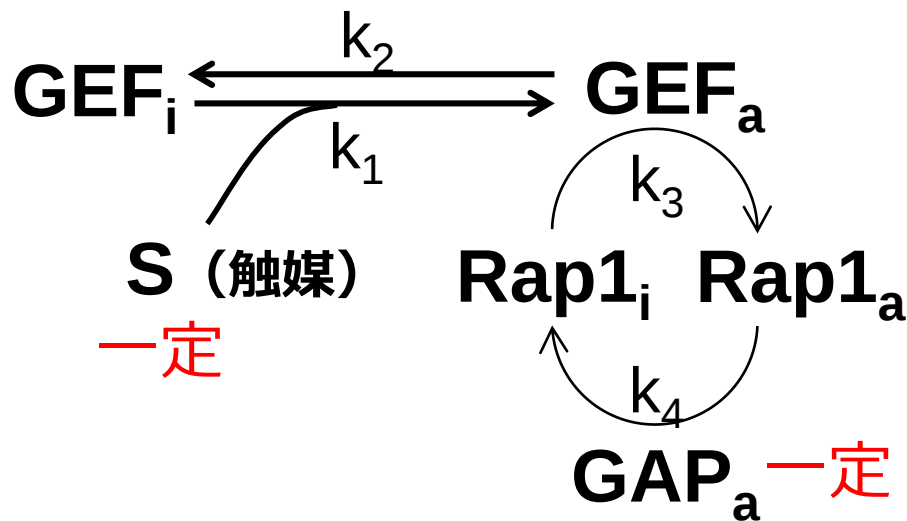
$$\frac{dRas_a}{dt} = ??$$

何の紙でもいいので書いてください。

課題1-2

コード : Rap1_model.m
データ : なし

シンプルRap1モデルのシミュレーション結果を図示せよ。
m-file “Rap1_model.m”の空欄 (□) を埋めて作成せよ。



不活性型と活性型の総和は保存

$$\text{GEF}_i + \text{GEF}_a = 1$$

$$\text{GAP}_i + \text{GAP}_a = 1$$

$$\text{Rap}_i + \text{Rap}_a = 1$$

変数 (分子)	初期濃度
S	1 (定数)
GEF_a	0
GAP_a	0.005 (定数)
Rap1_a	0

パラメータ	値
k_1	0.5
k_2	0.5
k_3	0.2
k_4	10

課題1-2

コード : Rap1_model.m
データ : なし

シンプルRap1モデルのシミュレーション結果を図示せよ。
m-file “Rap1_model.m”の空欄 (□) を埋めて作成せよ。

Rap1_model.mの内容

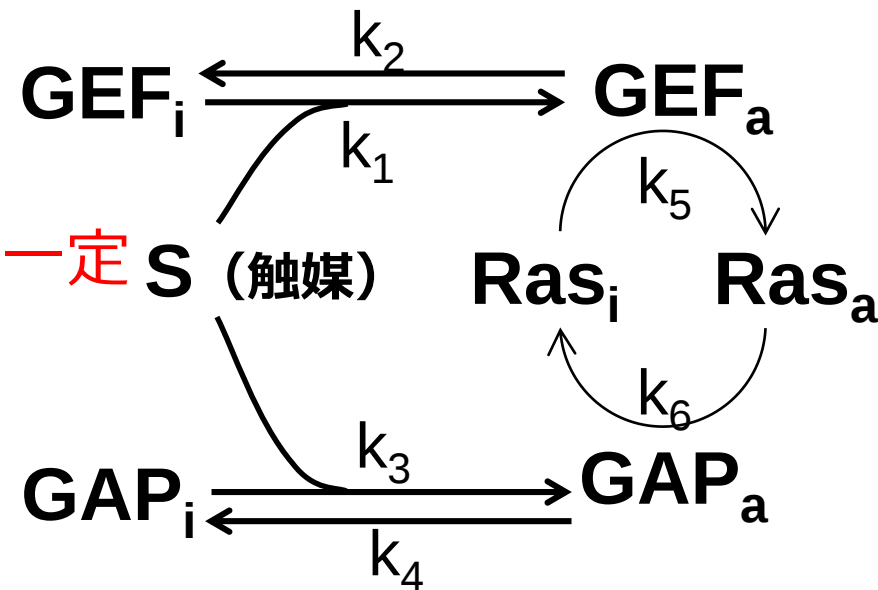
```
function Rap1_model()  
    % Rap1モデル  
    time = 0:100;  
    param = [□]; % [k1, k2, k3, k4]の順に  
    y0 = [□]; % [S, GEFa, GAPa, Rap1a]の順に  
  
    [t_sim, timeCourse_sim] = ...  
        ode15s(@(t, y) ODE(t, y, param), time, y0);  
  
    figure;  
    plot(t_sim, timeCourse_sim(:,4));  
    xlabel('time');  
    ylabel('concentration');  
    title('Rap1 model');  
  
end
```

```
function dydt = ODE(t, y, param)  
    S = y(1);  
    GEFa = y(2); % GEFi = 1 - GEFa  
    GAPa = y(3); % GAPi = 1 - GAPa  
    Rap1a = y(4); % Rap1i = 1 - Rap1a  
  
    k1 = param(1);  
    k2 = param(2);  
    k3 = param(3);  
    k4 = param(4);  
  
    dydt(1, :) = □ ; % dS/dt=??  
    dydt(2, :) = □ ; % dGEFa/dt=??  
    dydt(3, :) = □ ; % dGAPa/dt=??  
    dydt(4, :) = □ ; % dRap1a/dt=??  
  
end
```

課題1-3

コード：Ras_model.m
データ：なし

シンプルRasモデルのシミュレーション結果を図示せよ。
m-file “Ras_model.m”の空欄 (□) を埋めて作成せよ。



不活性型と活性型の総和は保存

$$GEF_i + GEF_a = 1$$

$$GAP_i + GAP_a = 1$$

$$Ras_i + Ras_a = 1$$

変数 (分子)	初期濃度
S	1 (定数)
GEF _a	0
GAP _a	0 (定数)
Ras _a	0
パラメータ	値
k ₁	0.5
k ₂	5
k ₃	0.0005
k ₄	0.005
k ₅	0.05
k ₆	100

課題1-3

コード : Ras_model.m
データ : なし

シンプルRasモデルのシミュレーション結果を図示せよ。
m-file “Ras_model.m”の空欄 (□) を埋めて作成せよ。

Ras_model.mの内容

```
function Ras_model()
    % Rap1モデル
    time = 0:100;
    param = [ ]; % [k1, k2, k3, k4, k5, k6]の順に
    y0 = [ ]; % [S, GEFa, GAPa, Rasa]の順に

    [t_sim, timeCourse_sim] = ...
        ode15s(@(t, y) ODE(t, y, param), time, y0);

    figure;
    plot(t_sim, timeCourse_sim(:,4));
    xlabel('time');
    ylabel('concentration');
    title('Ras model');

end
```

```
function dydt = ODE(t, y, param)
    S = y(1);
    GEFa = y(2); % GEFi = 1 - GEFa
    GAPa = y(3); % GAPi = 1 - GAPa
    Rasa = y(4); % Rasi = 1 - Rasa

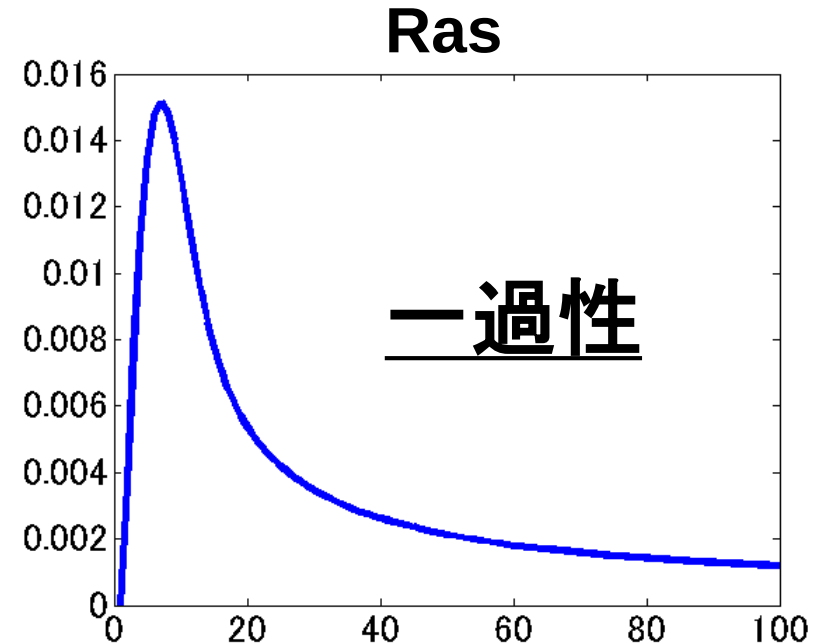
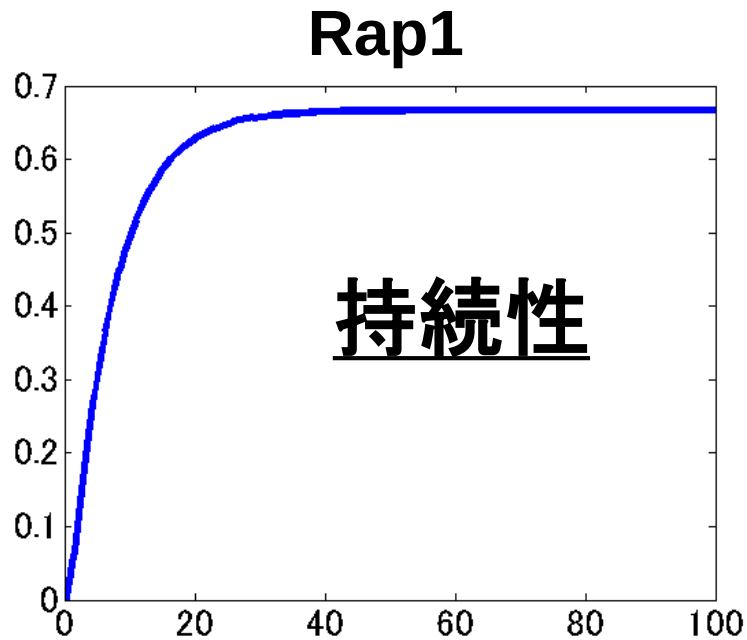
    k1 = param(1);
    k2 = param(2);
    k3 = param(3);
    k4 = param(4);
    k5 = param(5);
    k6 = param(6);

    dydt(1, :) = [ ]; % dS/dt=?
    dydt(2, :) = [ ]; % dGEFa/dt=?
    dydt(3, :) = [ ]; % dGAPa/dt=?
    dydt(4, :) = [ ]; % dRasa/dt=?

end
```

課題1-2, 1-3の結果

波形をプロットして，モデルが正しく作れているか確認する．



II. 課題

今日使うファイル類は下記にあります

<http://kurodalab.bs.s.u-tokyo.ac.jp/class/Summer/2019/Day3/>

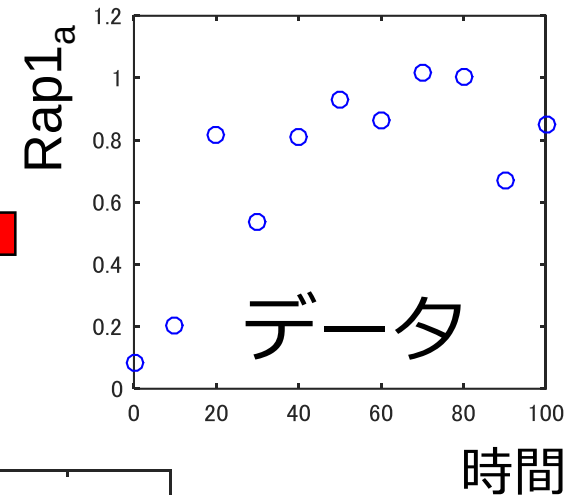
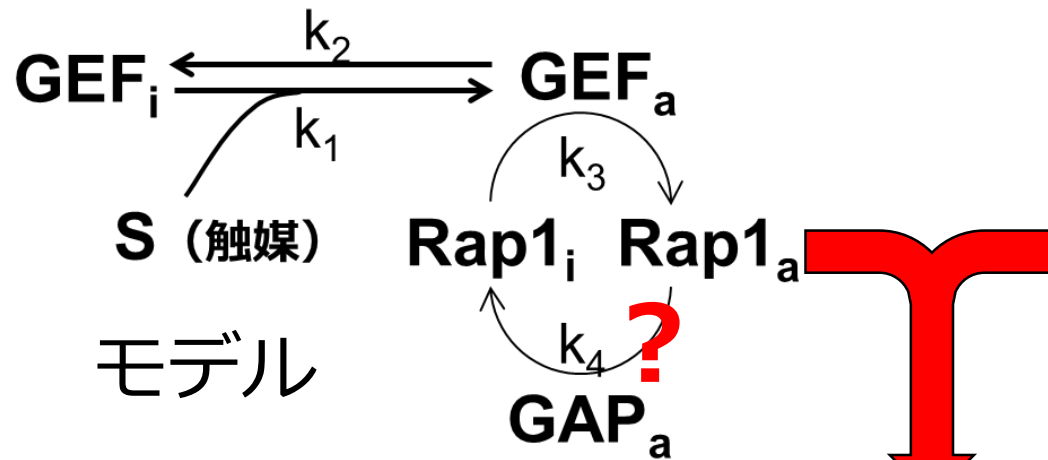
1. Ras, Rap1のODEモデルを作成
2. 1つ or 2つの未知パラメータを推定
3. 4つの未知パラメータをEPで推定

「課題」「発展課題」ができたならTAのチェックを受けてください。
14:30までに終了したい場合は「発展課題」まで終わらせてください。
14:30以降に終了する場合は「課題」まで終わらせてください。
終わったら、授業の途中で帰っても構いません。

課題2-1

コード：Rap1_model.mをコピーして修正
データ： data_Rap1_1para.mat

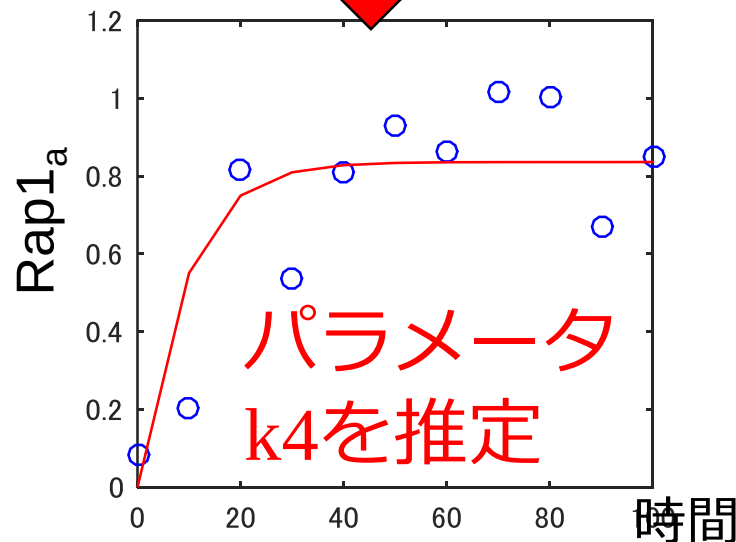
最小二乗法に基づき、Rap1モデルのパラメータk4を手で推定せよ。先ほど作成したm-file “Rap1_model.m”をコピー・修正せよ。



k4を手で色々変えて、
残差二乗和が最小となる値を探す。

ただし、k4以外のパラメータは
既知とする

$k_1 = 0.5$
 $k_2 = 0.5$
 $k_3 = 0.2$



課題2-1

コード : Rap1_model.mをコピーして修正
データ : data_Rap1_1para.mat

```
function x_Rap1_RSS=hand_Rap1_1param(k4)
    param = [0.5, 0.5, 0.2, ]; % [k1, k2, k3, k4]の順に
    y0 = [1, 0, 0.005, 0]; % [S, GEFa, GAPa, Rap1a]の順に

    load('data_Rap1_1para.mat'); % x: Rap1aのデータ、t:時間
    [t_sim, timeCourse_sim] = ode15s(@(t, y) ODE(t, y, param), t, y0);
    x_Rap1_residual = ; % 残差の計算
    x_Rap1_RSS = ; % 残差二乗和の計算

    figure;
    plot(t, x, 'bo'); % データ点の表示
    hold on;
    plot(t_sim, timeCourse_sim(:, 4), 'r-'); % シミュレーション結果の表示
    (略)
end
```

外から、引数でk4を与えると、
残差二乗和(x_Rap1_RSS)が計算される

←追加・修正

←追加、□を記入

←追加

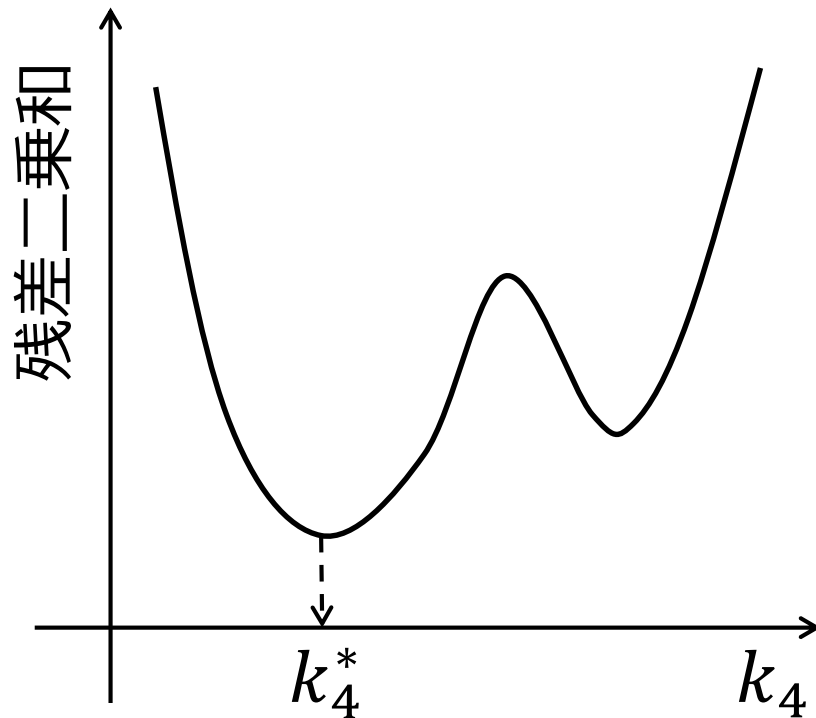
実行時はコマンドウィンドウ上から「hand_Rap1_1param(k4の数値)」

課題2-2

コード： fit_Rap1_1param.m
データ： data_Rap1_1para.mat

最小二乗法に基づき、Rap1モデルのパラメータ k_4 を自動で推定せよ。m-file “fit_Rap1_1param.m”の空欄を埋めて作成せよ。パラメータを変えたときの評価関数の景観を図示せよ。

評価関数(残差二乗和)の景観



パラメータ k_4 の最適解

パラメータ k の値を少しずつ変えながら、残差二乗和を求める。横軸にパラメータ k の値、縦軸に残差二乗和をとってプロットする。

残差二乗和が最小となるパラメータ k_4 を推定する。

課題2-2

コード : fit_Rap1_1param.m
データ : data_Rap1_1para.mat

```
function [k4_est] = fit_Rap1_1param()
```

```
param = [0.5, 0.5, 0.2, NaN]; % [k1, k2, k3, k4]の順に(k4は後で変えるので、仮にNaNを代入)  
y0 = [1, 0, 0.005, 0]; % [S, GEFa, GAPa, Rap1a]
```

```
load('data_Rap1_1para.mat'); % x: Rap1aのデータ、t:時間
```

```
%% k4を振り、残差2乗和(RSS)が最小となるときのk4を求める  
k4 = 0.1:0.05:10; % k4の候補。0.1から10まで0.05刻み。
```

```
[x_Rap1_RSS] = landscape_Rap1_1param(k4, t, x, param, y0); % t及びxはデータ点の時間とRap1aの波形
```

```
[RSS_min, idk4_minRSS] =  ; % 最小値を探す関数minを使って、最小となるRSSとそのときの番号を求める  
k4_est =  ; % RSSが最小となるときのk4を代入。k4は配列であることに注意。
```

```
%% 上で求めた残差2乗和(RSS)が最小となるk4を代入し、シミュレーション
```

```
temp_param = param;
```

```
temp_param(4) =  ; % k4は、RSS最小となるときの値を代入
```

```
[t_sim, timeCourse_sim] = ode15s(@(t, y) ODE(t, y, temp_param), t, y0);
```

k4を0.1から10まで0.05刻みで動かし
残差2乗和(RSS)を計算

次スライドに続く

課題2-2

前スライドからの続き

コード : fit_Rap1_1param.m
データ : data_Rap1_1para.mat

```
%% 描画
figure(1);
%% 時間波形
subplot(1, 2, 1)
% データ点
plot(t, x, 'bo');
hold on
% シミュレーションの結果
plot(t_sim, timeCourse_sim(:, 4), 'r-');
title('データとODEの出力');
xlabel('時間');
ylabel('Rap1a');
%% RSSとk4の関係
subplot(1, 2, 2);
plot(k4, x_Rap1_RSS, 'b-');
hold on
plot(k4_est, RSS_min, 'ro');
title('評価関数の景観');
xlabel('k4');
ylabel('残差二乗和');
end
```

```
function [x_Rap1_RSS] = landscape_Rap1_1param(..
k4, SamplingTime, x_Rap1, param, y0)

% 各要素に各k4のときのRSSが入るような配列を準備
x_Rap1_RSS = zeros(1, length(k4));
for i = 1:length(k4)
    param(4) = ; % 変えているパラメータは？
    [~, tc] = ode15s(@(t, y) ODE(t, y, param), SamplingTime, y0);
    vary_x_Rap1 = tc(:, 4);
    x_Rap1_Residual = vary_x_Rap1 - x_Rap1;
    x_Rap1_RSS(i) = ; % 残差はx_Rap1_Residual。残差2乗和は？
end
end

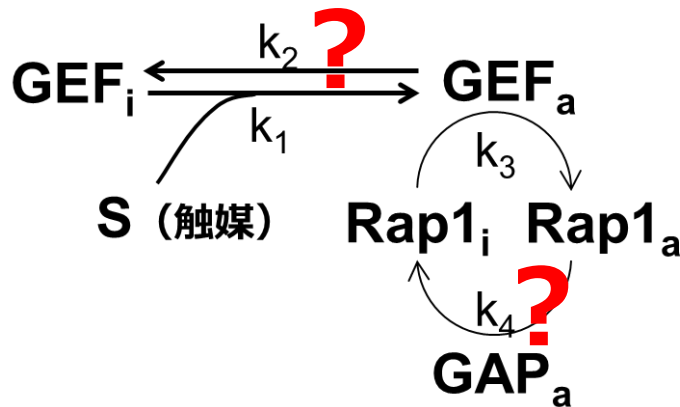
function dydt = ODE(t, y, param)
 % ここにモデルをコピー
end
```

課題2-3

コード : fit_Rap1_2param.m
データ : data_Rap1_2para.mat

最小二乗法に基づき、Rap1モデルのパラメータ k_2 , k_4 を自動で推定せよ。m-file “fit_Rap1_2param.m”の空欄(□)を埋めて作成せよ。

推定したパラメータを用いて、パラメータ k_2 , k_4 をそれぞれ変えたときの評価関数の景観を図示せよ。



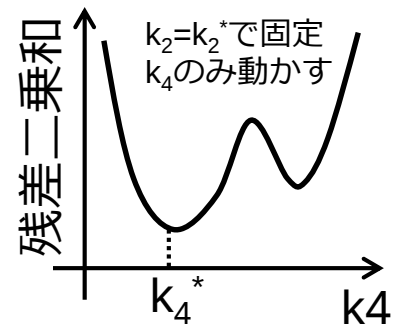
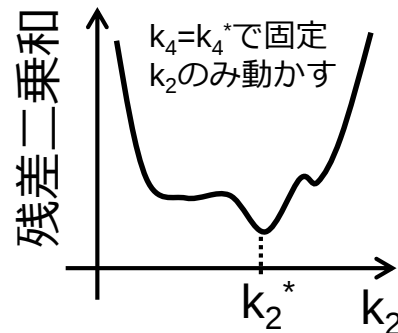
パラメータ推定 $k_2 = k_2^*$
 $k_4 = k_4^*$ k_2, k_4 の最適解

k_2, k_4 のうち、片方は最適解で固定、もう片方だけ動かす

ただし、 k_2, k_4 以外のパラメータは既知とする

$k_1 = 0.05$

$k_3 = 1.5$



課題2-3

コード： fit_Rap1_2param.m
データ： data_Rap1_2para.mat

その1/3 空欄(□)は「その2/3」「その3/3」にあります。

```
function [k_est] = fit_Rap1_2param()
```

```
param = [0.05, NaN, 1.5, NaN]; % [k1, k2, k3, k4]の順に(k2, k4は後で変えるので、仮にNaNを代入)  
y0 = [1, 0, 0.005, 0]; % [S, GEFa, GAPa, Rap1a]
```

```
load('data_Rap1_2para.mat'); % x: Rap1aのデータ、t:時間
```

```
%% k2, k4を振って、残差2乗和(RSS)が最小となるときのk2, k4を求める  
k2 = 0.1:0.05:3; % k2の候補。0.1から3まで0.05刻み。  
k4 = 0.1:0.05:3; % k4の候補。0.1から3まで0.05刻み。
```

```
[x_Rap1_RSS] = landscape_Rap1_2param(k2, k4, t, x, param, y0); % t及びxはデータ点の時間とRap1aの波形
```

次スライドに続く

課題2-3

前スライドからの続き

コード : fit_Rap1_2param.m
データ : data_Rap1_2para.mat

その2/3

% RSSの最小値を見つける。そのときの行と列を保存する。

```
[RSS_min_row, idk2_minRSS_vec] = min(x_Rap1_RSS);
```

```
[RSS_min, idk4_minRSS] = min(RSS_min_row);
```

```
idk2_minRSS = idk2_minRSS_vec(idk4_minRSS);
```

k2_est = k2(idk2_minRSS); % RSSが最小となるときのk2を代入

k4_est = k4(idk4_minRSS); % RSSが最小となるときのk4を代入

k_est = [k2_est, k4_est]; %RSSが最小となるk2とk4

%% 上で求めた残差2乗和(RSS)が最小となるときのk2, k4を代入し、シミュレーション

temp_param =

```
[t_sim, timeCourse_sim] = ode15s(@(t, y) ODE(t, y, temp_param), t, y0);
```

k4

k2

x_Rap1_RSS:
k2, k4に対する
残差二乗和の行列

次スライドに続く

課題2-3

前スライドからの続き

コード : fit_Rap1_2param.m
データ : data_Rap1_2para.mat

その3/3

%% 描画

figure

%% 時間波形

subplot(1, 3, 1)

plot(t, x, 'bo'); % データ点

hold on

% シミュレーション

plot(t_sim, timeCourse_sim(:, 4), 'r-');

title('データとODEの出力');

xlabel('時間'); ylabel('Rap1a');

%% RSSとk2の関係

subplot(1, 3, 2);

plot(, 'b-'); % k2vs RSS, k4は最適解

hold on

plot(k2_est, RSS_min, 'ro');

title('評価関数の景観');

xlabel('k2'); ylabel('残差二乗和');

%% RSSとk4の関係

subplot(1, 3, 3);

plot(, 'b-'); % k4vs RSS, k2は最適解

hold on

plot(k4_est, RSS_min, 'ro');

title('評価関数の景観');

xlabel('k4'); ylabel('残差二乗和');

end

```
function [x_Rap1_RSS] = landscape_Rap1_2param(..  
    k2, k4, SamplingTime, x_Rap1, param, y0)
```

% 残差二乗和は行列として保存する。

x_Rap1_RSS = zeros(length(k2), length(k4));

disp('k2') % disp(X): command windowにXを表示

for r = 1:length(k2)

disp(k2(r))

for c = 1:length(k4)

% 変えているパラメータは?
 % 変えているパラメータは?

[~, tc] = ode15s(@(t, y) ODE(t, y, param), SamplingTime, y0);

vary_x_Rap1 = tc(:, 4);

x_Rap1_Residual = ; % 残差

x_Rap1_RSS(r,c) = ; % 残差二乗和

end

end

end

function dydt = ODE(t, y, param) k2

% ここにモデルをコピー

end

k4

x_Rap1_RSS:
k2, k4に対する
残差二乗和の行列

II. 課題

今日使うファイル類は下記にあります

<http://kurodalab.bs.s.u-tokyo.ac.jp/class/Summer/2019/Day3/>

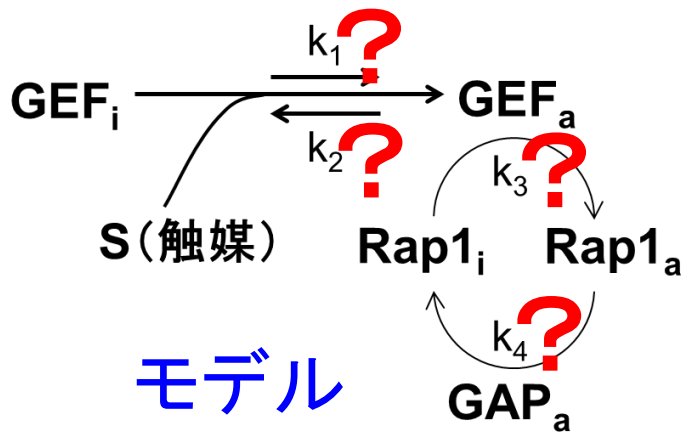
1. Ras, Rap1のODEモデルを作成
2. 1つ or 2つの未知パラメータを推定
3. 4つの未知パラメータをEPで推定

「課題」「発展課題」ができたならTAのチェックを受けてください。
14:30までに終了したい場合は「発展課題」まで終わらせてください。
14:30以降に終了する場合は「課題」まで終わらせてください。
終わったら、授業の途中で帰っても構いません。

課題3-1

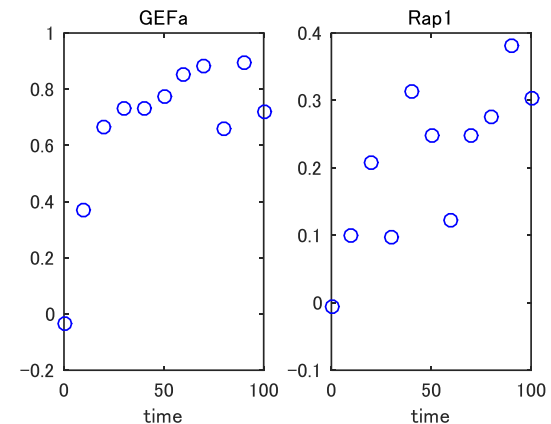
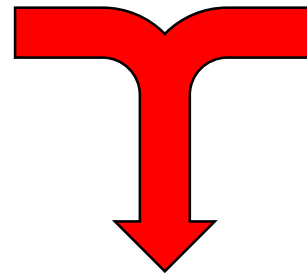
コード： hand_Rap1_1param.mをコピーして修正
データ： data_Rap1_4para.mat

Rap1モデルのパラメータ k_1, k_2, k_3, k_4 を手で推定せよ。m-file
“hand_Rap1_1param.m”をコピー・修正して作成せよ。



モデル

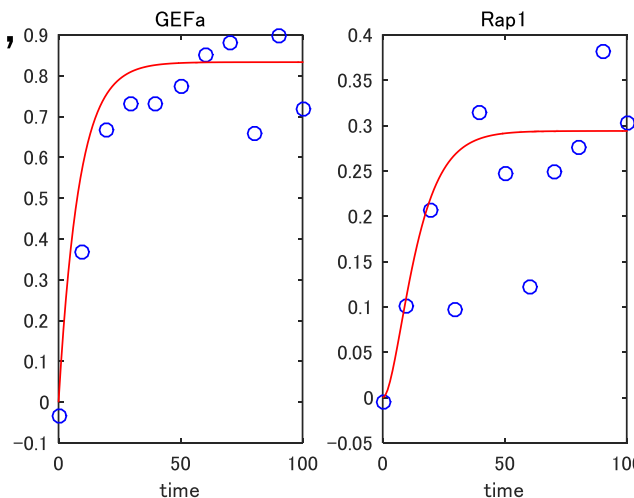
手で $k_1 \sim k_4$ を
変える



GEFa, Rap1の
データ

$k_1 \sim k_4$ を手で色々変えて、
データにもっともよく合う
値を探す。

※ 手で合わせるのが大
変なことを確認する程度
でよい。真面目に合わせる
必要はない。



$k_1 \sim k_4$ を推定

課題3-1

コード： hand_Rap1_1param.mをコピーして修正
データ： data_Rap1_4para.mat

```
function RSS = hand_Rap1_4param(k) ← 表示部分を修正
    param = [ ]; % [k1, k2, k3, k4]の順に
    y0 = [1, 0, 0.005, 0]; % [S, GEFa, GAPa, Rap1a]の順に
    load('data_Rap1_4para.mat'); % x: S, GEFa, GAPa, Rap1aのデータ、t:時間 ← 表示部分を修正

    [t_sim, timeCourse_sim] = ode15s(@(t, y) ODE(t, y, param), t, y0);
    residual = [ ]; % 残差の計算
    RSS = [ ]; % 残差二乗和の計算

    figure;
    for i = 1:2
        subplot(1, 2, i);
        plot(t, x(:, i*2), 'bo'); % データ点の表示
        hold on;
        plot(t_sim, timeCourse_sim(:, i * 2), 'r-'); % シミュレーション結果の表示
    end
    xlabel('time');
    (略)
end
```

GEFaとRap1a ← 表示部分を追加

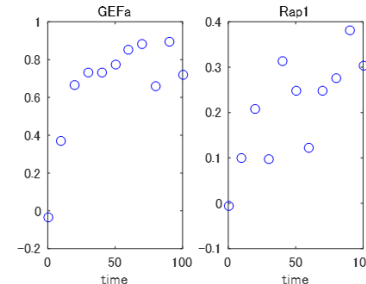
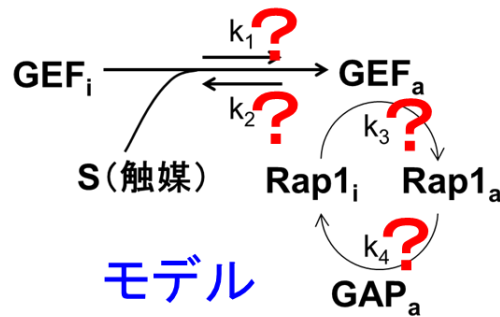
※ 手で合わせるのが大変なことを確認する程度でよい。
真面目に合わせる必要はない。

実行時：hand_Rap1_4param([数値, 数値, 数値, 数値])

課題3-2

コード : fitEP_Rap1_4param.mおよびcopasiep.m
データ : data_Rap1_4para.mat

Rap1モデルのパラメータ k_1 , k_2 , k_3 , k_4 をEPで推定せよ。
“fitEP_Rap1_4param.m”の空欄(□)を埋めて作成せよ。



モデル・データ

EP

EP

EP

...

RSS・パラメータ

RSS・パラメータ

RSS・パラメータ

...

RSSが最小のものを選択

最終的なRSS・パラメータ

課題3-2

その1/2

```
function [res_bestIndv, res_bestScore] = fitEP_Rap1_4para(n, numGeneration)
```

(省略)

```
end
```

```
function [bestIndv, bestScore, report] = work_EP_Rap1(numGeneration)
```

(省略)

%% 各系列で最小のRSSのときのパラメータを用いてシミュレーション

```
time = [0, 100];
```

```
[t_est, x_est] = ode15s(@(t, y) ODE(t, y, bestIndv), time, y0);
```

% データ点とシミュレーション結果の重ね描き

```
figure(1);
```

```
for i = 1:2
```

```
    subplot(1, 2, i);
```

```
    hold on
```

```
    plot(t, x(:, i*2), 'o'); % データ点
```

```
    hold on
```

```
    plot(          , 'r:'); % シミュレーション結果
```

```
    if i == 1
```

```
        title('GEF_a');
```

```
    else
```

```
        title('Rap1');
```

```
    end
```

```
end
```

% EPでRSSが減少して行く様子を描画

```
figure(2);
```

```
subplot(1, 2, 1);
```

```
hold on
```

```
plot(report(:, 1), 'b-');
```

```
title('最も良い個体');
```

```
xlabel('世代数');
```

```
ylabel('残差2乗和');
```

```
subplot(1, 2, 2);
```

```
hold on
```

```
plot(report(:, 2), 'b-');
```

```
title('全個体の平均');
```

```
xlabel('世代数');
```

```
ylabel('残差2乗和');
```

```
end
```

次スライドに続く

課題3-2

前スライドからの続き

その2/2

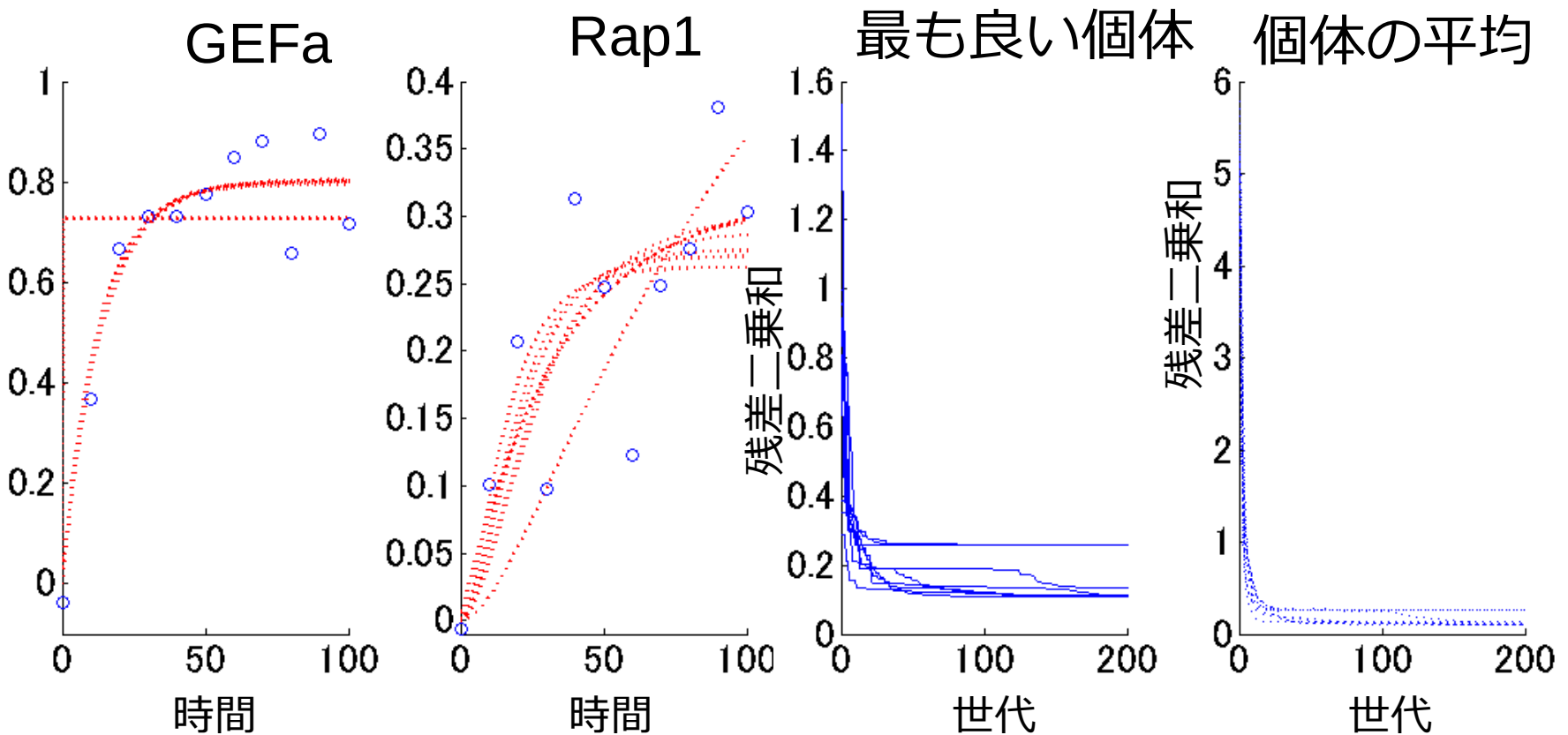
```
function [RSS] = cal_RSS(x_data, SamplingTime, param, y0)
    % 残差2乗和を計算
    [~, tc] = ode15s(@(t, y) ODE(t, y, param), SamplingTime, y0);
    residual = ; % 残差
    RSS = ; % 残差2乗和
end
```

```
function dydt = ODE(t, y, param)
    % ここにモデルをコピー
    
end
```

課題3-2: 実行結果の例

実行結果の例

回数 : $n=10$, 世代数 : $\text{numGeneration}=200$



ワークスペースの変数をクリックし, "res_bestIndv", "res_bestScore"をみて, 全体で最もよい解を確認する. (※実行時に出力を格納するようにする)

発展課題

Rap1のデータに対して、Rasモデルのパラメータ(k_1 - k_6)を推定せよ。

- RasモデルとRap1モデルの関係はどうなっているか？
- Rasモデルを使った場合とRap1モデルを使った場合で、パラメータや残差二乗和にどのような違いが現れたか？
- 残差二乗和が似ている最適化試行では、パラメータの推定値も似ているか？