

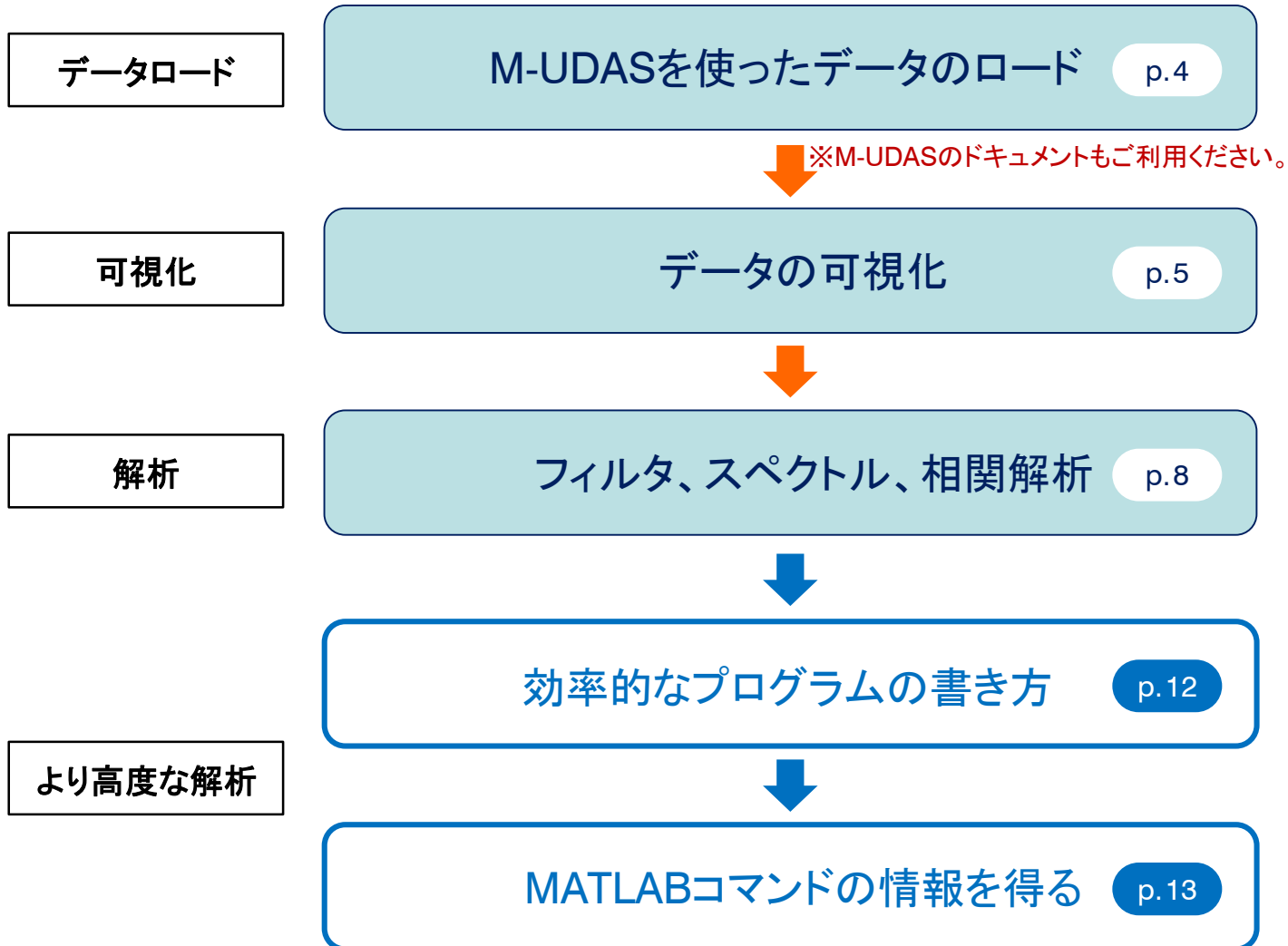
- 科学技術計算ための計算機言語。
- MATLAB = MATrix LABoratory
- 行列演算が得意。
- 対話型の開発環境。
- データ解析、可視化、数値計算、アルゴリズム開発、シミュレーション、GUIアプリケーション開発等に利用。
- 自然科学だけでなく、機械工学、電気・電子工学、情報学、医学、金融等にも応用。
- 豊富な関数ライブラリ、グラフィックツール。
- オプションのToolboxが豊富。
- 変数宣言が不要。

MATLABとIDLの比較(代表的なもの):

	MATLAB	IDL
配列の先頭の要素のインデックス	1	0
配列の表記方法	x(2:9)	x[2:9]
配列の要素の取り出し方	B = A(:, 2:8)	B = A[* , 2:8]
行列の表現方法	i 行 j 列が A(i, j) (一つ目の要素が行)	i 行 j 列が A[j, i] (一つ目の要素が列)
行列の作成	A = [1, 2, 3; 4, 5, 6]	A = [[1, 2, 3], [4, 5, 6]]
行列の要素ごとの掛け算	A.*B	A*B
行列の積	A*B	A#B
コメントアウトの記法	%	;
改行の記法	...	\$
予約語	pi, NaN, inf	!PI, !VALUES.D.NAN, !VALUES.D.INFINIY
if文	<pre>if i == j A(i, j) = 1; elseif i > j A(i, j) = 2; else A(i, j) = 0; end</pre>	<pre>IF (i EQ j) THEN BEGIN A[i, j] = 1; ENDIF ELSE IF (i GT j) THEN BEGIN A[i, j] = 2; ENDIF ELSE BEGIN A[i, j] = 0; ENDELSE</pre>
for文	<pre>for i = 1:length(x) x(i) = i; end</pre>	<pre>FOR i = 0, N_ELEMENTS(x)-1 DO BEGIN x[i] = i; ENDFOR</pre>
実行方法	インタプリタ	コンパイラ

3. M-UDASを利用したデータ解析の流れ

本資料では、MATLAB/M-UDASを利用した以下のデータ解析の流れを簡単に解説します。



4-1. データ読み込み

IUGONET参加機関がウェブ上に公開している多種多様なデータを読み込むには、MATLABライブラリ「M-UDAS」が便利です。M-UDASにより、複数日のデータファイルのダウンロード、読み込みが1コマンドで実行でき、即座に可視化・解析に進めます。

```


データロード
↓
読み込みデータ一覧
↓
メタデータ表示
        
```

```

>> iug_load_XXX(開始日時, 終了日時, 'パラメータ名', '値');

>> whos

>> disp_info(xxx_info);
        
```



ダウンロード元URL: <https://github.com/iugonet/UdasMatlab>

M-UDAS 1.00 に含まれているロード関数:

No.	データの種類	ロード関数名
1	極地研・全天イメージャデータ	iug_load_asi_nipr
2	極地研・全天イメージャ・ケオグラムデータ	iug_load_ask_nipr
3	名大ISEE・広ビームリオメータデータ	iug_load_brio_isee
4	EISCATレーダーデータ	iug_load_eiscat
5	名大ISEE・フラックスゲート磁力計データ	iug_load_gmag_isee_fluxgate
6	名大ISEE・誘導磁力計データ	iug_load_gmag_isee_induction
7	MAGDAS磁力計ネットワークデータ	iug_load_gmag_magdas
8	210度磁気子午線磁力計ネットワークデータ	iug_load_gmag_mm210
9	極地研・フラックスゲート磁力計データ	iug_load_gmag_nipr
10	極地研・誘導磁力計磁力計データ	iug_load_gmag_nipr_induction
11	京大RISH・流星レーダーデータ	iug_load_meteor_rish
12	京大RISH・MFLレーダーデータ	iug_load_mf_rish

4-2. データの可視化(ラインプロット)

Figure ウィンドウの作成

<code>figure ;</code>	新しい Figure ウィンドウを作成します。
<code>f = figure ;</code>	Figure オブジェクト <code>f</code> を返します。 <code>f</code> は、Figure プロパティの変更の際に使用します。

座標軸 (Axes) の作成

<code>subplot(m, n, p) ;</code>	Figure を <code>m</code> 行 <code>n</code> 列のグリッドに分割し、 <code>p</code> で指定された位置に座標軸を作成します。
<code>ax = subplot(m, n, p) ;</code>	Axes オブジェクト <code>ax</code> を返します。 <code>ax</code> は、座標軸の変更の際に使用します。

※R2019b以降は、`subplot` の代わりに、`tiledlayout` コマンドも利用できます。

ラインプロット

<code>plot(X, Y) ;</code>	<code>X, Y</code> の座標データを使って 2 次元ラインプロットを作成します。
<code>plot(X1, Y1, ..., Xn, Yn) ;</code>	同一座標軸上に、複数の <code>X, Y</code> ペアをプロットします。
<code>plot3(X, Y, Z)</code>	<code>X, Y, Z</code> で指定される座標データを 3 次元空間にプロットします。

グラフィックスオブジェクトのプロパティの設定

<code>set(H, Name, Value) ;</code>	オブジェクト <code>H</code> のプロパティ <code>Name</code> の値を <code>Value</code> に設定します。 <code>H</code> に <code>gca</code> (現在の座標軸)、 <code>gcf</code> (現在の Figure ハンドル) を指定することもできます。例: <code>set(gca, 'Color', 'y')</code> で現在の軸が黄色になります。
------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

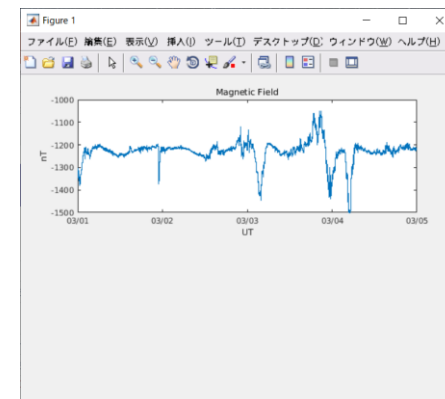
例:

```
>> figure;
>> subplot(2, 1, 1);
>> plot(x, y);
>> set(gca, 'ylim', [-1500, -1000])
```

```
>> title('Magnetic Field')
>> xlabel('UT');
>> ylabel('nT')
```

Figure ウィンドウを開きます。
2行1列の座標軸の1つ目の座標軸を作成します。
データをプロットします。
Y軸の範囲を[-1500, -1000]に設定します。

軸のタイトルを、Magnetic Fieldに指定します。
X軸のラベルを、UTに指定します。
Y軸のラベルを、nTに指定します。



4-2. データの可視化 (カラープロット)

疑似カラープロット

<code>pcolor(Z)</code> ;	行列 Z 内の値を使用して疑似カラープロットを作成します。
<code>pcolor(x, y, Z)</code> ;	ベクトル x, y で色を塗る頂点の X, Y 座標を指定します。x, y の長さは、Zを m 行 n 列とすると、m, nと一致する。
<code>pcolor(X, Y, Z)</code> ;	行列 X, Y で色を塗る頂点の X, Y 座標を指定します。X, Y のサイズは、行列Zのサイズと一致します。

カラーシェーディングプロパティの設定

<code>shading flat</code> ;	カラーシェーディングプロパティを設定します。flat, faceted (デフォルト), interp 等の値があります。
-----------------------------	---------------------------------------------------------------

カラーバーの表示

<code>colorbar</code> ;	座標軸の右側に、カラースケールを示す垂直なカラーバーを表示します。
<code>c = colorbar</code> ;	カラーバーオブジェクト c を返します。c は、カラーバーのプロパティの変更の際に使用します。

カラーマップの設定

<code>colormap(map)</code> ;	map で指定したカラーマップに設定します。parula (デフォルト), jet, hot, cool, gray 等の値があります。
------------------------------	----------------------------------------------------------------------

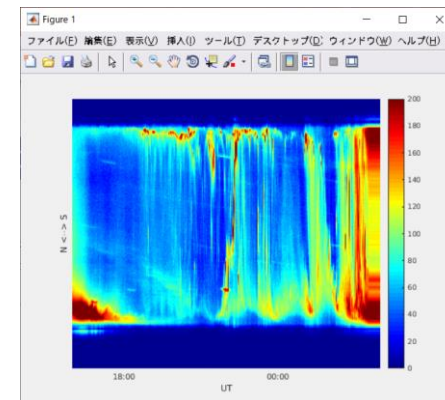
例:

```
>> pcolor(x, y, Z);
>> shading flat;
>> colormap(jet);
>> set(gca, 'clim', [0, 200])
>> colorbar;
```

```
>> set(gca, 'yticklabel', [ ])
>> xlabel('UT');
>> ylabel('N <--> S');
```

カラープロットを作成します。
 カラーシェーディングを flat に設定します。
 カラーマップを jet に変更します。
 カラーの範囲を [0, 200] に設定します。
 カラーバーを表示します。

Y軸の目盛りを表示しないようにします。
 X軸のラベルを、UT に指定します。
 Y軸のラベルを、N<-->S に指定します。



4-2. データの可視化(その他のプロット)

分布図のプロット

<code>histogram(X);</code>	行列 X のヒストグラムプロットを作成します。
<code>scatter(x, y);</code>	ベクトル x および y で指定された位置に円(o)をプロットし、散布図を作成します。

等高線図のプロット

<code>contour(Z);</code>	行列 Z の等高線図を作成します。
<code>contourf(Z);</code>	行列 Z の塗りつぶし等高線図を作成します。
<code>contour3(Z);</code>	行列 Z の3次元等高線図を作成します。

ベクトル場のプロット

<code>quiver(x, y, u, v);</code>	(x, y) で指定される座標に (u, v) で指定されるベクトルを矢印でプロットします。
<code>quiver3(x, y, z, u, v, w);</code>	(x, y, z) で指定される座標に (u, v, w) で指定されるベクトルを矢印でプロットします。

ボリュームデータの可視化

<code>slice(X, Y, Z, V, xs, ys, zs);</code>	座標データ X, Y, Z で指定されるボリュームデータ V のスライス平面を、 xs, ys, zs の位置に表示します。
<code>contourslice(X,Y,Z,V, xs, ys, zs);</code>	座標データ X, Y, Z で指定されるボリュームデータ V のスライス面内の等高線図を、 xs, ys, zs の位置に表示します。

ムービーの表示

<code>movie(M, n);</code>	関数 <code>getframe</code> で作成されたムービーフレームの行列 M を n 回再生します。
<code>F = getframe;</code>	画面に表示されている現在の座標軸をムービーフレームとして 行列 F にキャプチャします。

4-3. フィルタ処理

デジタルフィルターの設計

<code>d = designfilt(resp, Name, Val);</code>	<p>応答タイプ resp の digitalFilter オブジェクト d を設計します。Name, Val のペアのセットを使用してフィルターの特性を規定します。</p> <p>応答タイプ resp には、'lowpassfir', 'highpassfir', 'bandpassfir' 等の値があります。Nameには、resp に応じて、'FilterOrder', 'PassbandFrequency', 'StopbandFrequency', 'DesignMethod' 等を設定できます。</p>
<code>fvtool(d);</code>	デジタルフィルター d の振幅応答を可視化します。

フィルター処理の実行

<code>y = filter(d, x);</code>	デジタル フィルター d を使用して、データ x をフィルター処理を施します。
<code>y = filtfilt(d, x);</code>	データ x を順方向と逆方向の両方で処理して、ゼロ位相デジタルフィルター処理を実行します。

例: ハイパスフィルター

```
>> hpFilt = designfilt('highpassfir', ...
    'PassbandFrequency', 1/150, ...
    'StopbandFrequency', 1/200, ...
    'SampleRate', 1, ...
    'DesignMethod', 'kaiserwin');
```

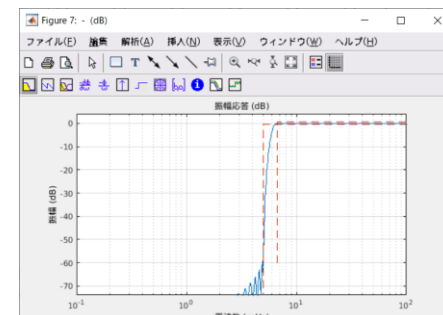
```
>> fvtool(hpFilt);
```

```
>> OutData = filtfilt(hpFilt, InData);
>> plot(time, OutData);
```

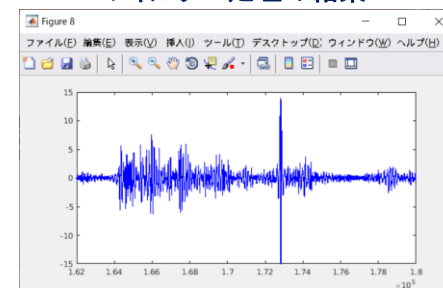
ハイパスフィルタを設計します。
 'PassbandFrequency'(通過帯域周波数)を1/150 Hz、
 'StopbandFrequency'(阻止帯域周波数)を1/200 Hz、
 'SampleRate'(サンプルレート)を1Hz、
 'DesignMethod'(設計法)をカイズー ウィンドウ法に
 設定します。
 デジタルフィルターの特性を可視化します。

データに、ゼロ位相フィルター処理を実行します。
 フィルター処理したデータをプロットします。

※R2018a以降のバージョンでは、
 lowpass(ローパスフィルタ)、highpass(ハイパスフィルタ)、bandpass(バンドパスフィルタ)
 等の関数も利用できます。



フィルター処理の結果



4-3. フィルタ処理

移動平均値、移動中央値

<code>y = movmean(x, k);</code>	データ x に対して、長さ k のスライディングウィンドウを使って移動平均値を計算します。
<code>y = movmedian(x, k);</code>	データ x に対して、長さ k のスライディングウィンドウを使って移動中央値を計算します。

スパイクノイズの除去

<code>y = medfilt1(x, n);</code>	データ x に n 次の 1 次元メディアンフィルターを適用します。
----------------------------------	----------------------------------------

トレンドの除去

<code>y = detrend(x, n);</code>	データ x から n 次の多項式トレンドを除去します。 $n = 0$ のとき平均値を、 $n = 1$ のとき線形トレンドを除去します。
---------------------------------	---------------------------------------------------------------------------

多項式フィッティング

<code>[p, S] = polyfit(x, y, n);</code>	データ (x, y) に対して(最小二乗的に)最適な近似となる n 次の多項式の係数 $p(x)$ 、誤差推定を得るための構造体 S を返します。
<code>[y, delta] = polyval(p, x, S);</code>	<code>polyfit</code> で求めた多項式の係数 p と構造体 S を使って、多項式を x の各点において評価し、誤差推定値 $delta$ を求めます。

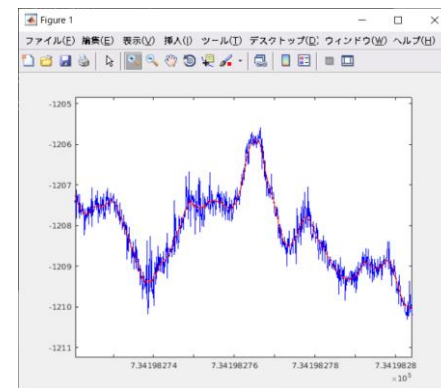
例:

```
>> mx = movmean(x, 20);
```

```
>> plot(time, x, time, mx);
```

データ x の20点の移動平均値 mx を計算します。

x, mx を同時にプロットします。



4-4. スペクトル解析

ダイナミックスペクトル

```
[s, f, t] = spectrogram(x, window, noverlap, nfft, fs);
```

データ x に FFT を施してダイナミックスペクトルを計算します。
 s (ダイナミックスペクトル計算結果)、 f (周波数)、 t (時刻) が出力されます。

$window$: FFT計算に利用するウィンドウを指定します。
 $noverlap$: $noverlap$ 個のデータをオーバーラップさせながら FFT を計算します。
 $nfft$: 離散フーリエ変換のサンプル数です。
 fs : データ x のサンプル周波数 (Hz) を指定します。

例:

```
>> dx = detrend(x);
```

```
>> [s, f, t] = spectrogram(dx, ...
    512, 256, 512, 1);
```

```
>> pcolor(t, f, real(log10(s)));
```

```
>> shading flat;
```

```
>> set(gca, 'ylim', [0, 0.1]);
```

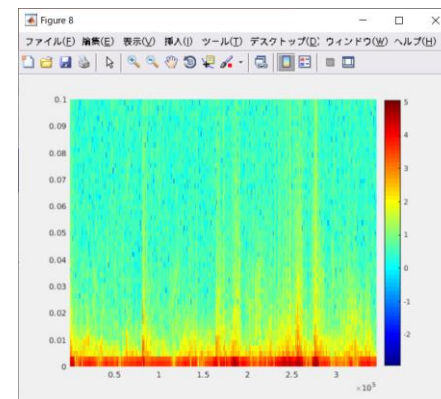
```
>> colorbar;
```

```
>> colormap(jet);
```

データ x から線形トレンドを除去します。

サンプリングレート 1 Hz のデータに対し、長さ 512 の
 ハミングウィンドウ(デフォルト)を利用し、256 サンプ
 ルずつオーバーラップしながら、ダイナミックスペクト
 ルを計算します。

X軸時刻 (t)、Y軸周波数 (f) で、計算結果をカラープ
 ロット表示します。
 カラーシェーディングを flat に設定します。
 Y軸の範囲を $[0, 0.1]$ に設定します。
 カラーバーを表示します。
 カラーマップを jet に変更します。



4-5. 相関解析

線形相関

<code>R = corrcoef(x, y);</code>	データ x, y の相関係数 R を計算します。
<code>[c, lags] = xcorr(x, y);</code>	2つの時系列データ x, y の相互相関 c とラグ $lags$ を計算します。
<code>C = cov(x, y);</code>	データ x, y の共分散を計算します。

例:

```
>> plot(time, x, time, y);

>> [c, lags] = xcorr(x, y, ...
    100, 'normalized');
>> plot(lags, c, 'bx-');

>> y1 = circshift(y, -17);
>> corrcoef(x, y1)
```

データ x, y をプロットします。

x, y の相互相関 c を計算します。
最大ラグを100とし、 c を正規化します。
横軸をラグ $lags$ として、相関係数 c をプロットします。

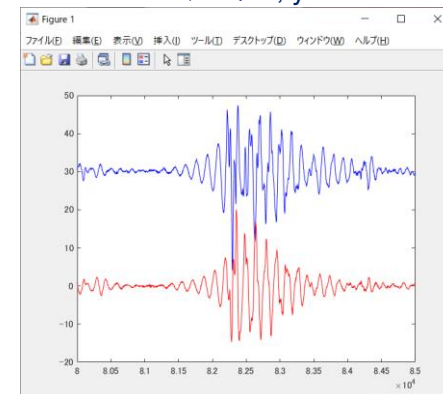
y をラグの分だけシフトさせて、 $y1$ に代入します。
 $x, y1$ の相関係数を計算します。

ans =

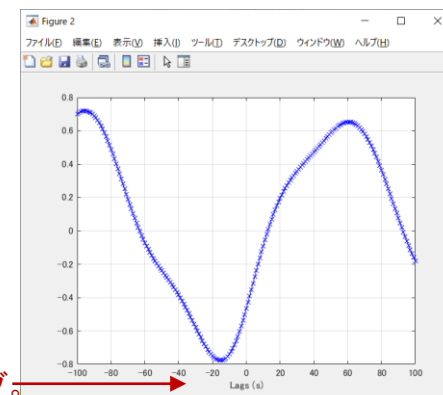
```
1.0000 -0.7740
-0.7740 1.0000
```

← 相互相関係数

データ x, y



x, y の相互相関 c



横軸は、ラグ。

- 繰り返し行列演算(ベクトル演算)を使う。
 - For文の使用は遅くなる原因。
- 予め必要な配列数を確保する。
 - 配列の自動拡張は便利だが、パフォーマンスが落ちる。
- なるべく組込み関数を使う。
 - 上の問題に対応されているものが多い。
- tic,tocを有効に使う。
 - 経過時間を測定するコマンドで時間がかかる部分を見つける。

6. MATLABコマンドの情報の取得方法

情報を得るためのコマンド

<code>help name ;</code>	関数、メソッド、クラス、変数など、name によって指定された機能のヘルプ テキストを表示します。
<code>doc name ;</code>	関数、クラス、ブロックなど、name で指定された機能のドキュメンテーションを表示します。
<code>lookfor keyword ;</code>	関数名がわからない場合に、keywordで指定したキーワードで目的の関数名を検索できます。

ヘルプセンター

<https://jp.mathworks.com/help/>

- Mathworkのウェブサイト内で、MATLABの利用方法、コマンド、プログラミング方法、数学等が、日本語で詳細に解説されています。(最新機能は、英語。)
- 動画やデモも充実しており、初心者から上級者まで、必要な情報をほとんど全てこのウェブサイトで得ることができます。

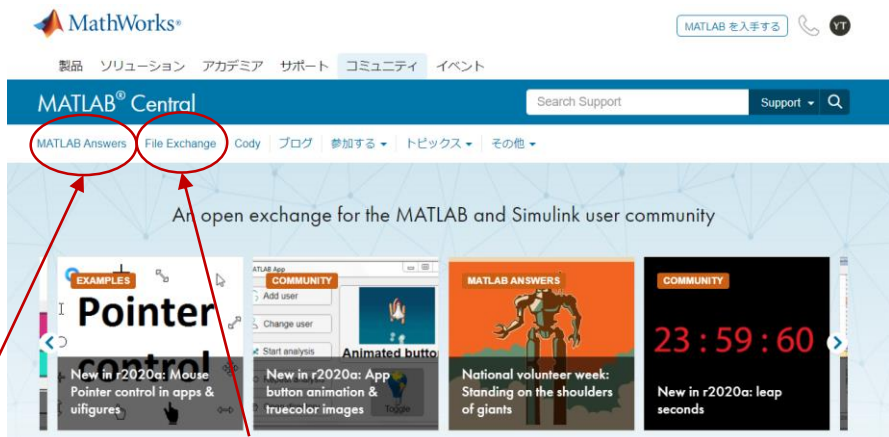


MATLAB Central

<https://jp.mathworks.com/matlabcentral/>

- **MATLAB Answers**から、MATLABに関して質問をすることで、コミュニティからの回答を得ることができます。
- **File Exchange**では、コミュニティから提供された様々な関数やToolboxを利用することができます。
- 例えば、**IGRFモデル**、**MSIS-E-00モデル**、**IRIモデル**等のMATLABプログラムを入手することができます。

※上記を利用するためには、MathWorksのアカウントが必要です。



MATLAB Answers

File Exchange