

MATLAB EXPO 2019

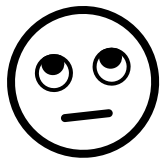
MATLAB® × データサイエンス
～最新機能紹介と業務一元化の手引き～

田口 美紗

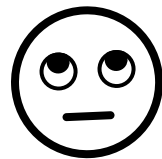


データサイエンス？

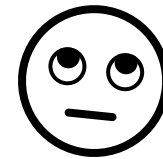
機械学習？



AI？



ディープラーニング？



データサイエンスで大切なこと

💡 可視化・異常値チェックなど前処理が不可欠

😞 しかし、前処理には非常に時間がかかる

講演の目的

**MATLABを用いて効率よく
データサイエンスのプロセスを
進めて頂きたい！**

本日のアジェンダ

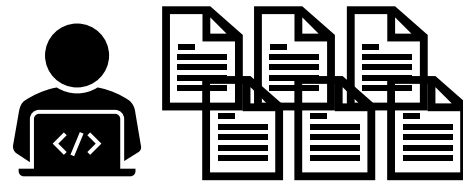
① アドホックな解析

- データ読み込み
- データアクセス
- 前処理
- 可視化



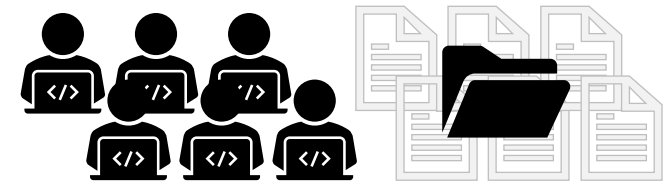
② ビッグデータ解析

- ビッグデータの読み込み
- ビッグデータの解析・深掘り



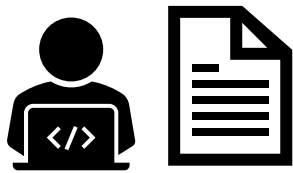
③ プロジェクト化

- プロジェクトの作成・共有
- 大人数で解析



①アドホックな解析

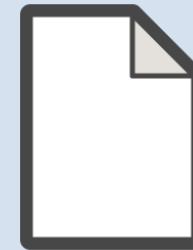
- データ読み込み
- データアクセス
- データ結合
- 可視化



フライトデータの解析
を例にご紹介

振動センサ

燃料センサ



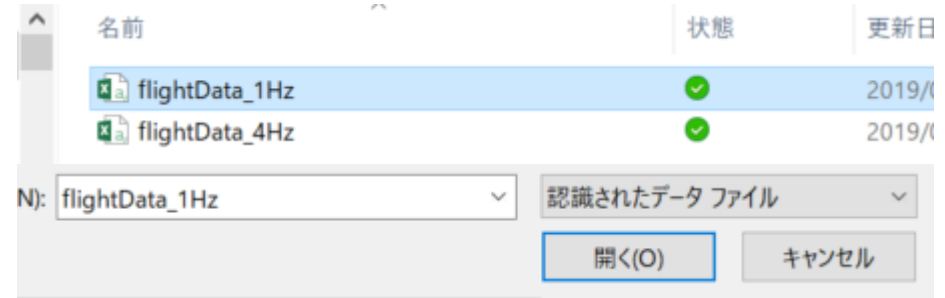
時間

緯度・経度

火災センサ

データ読み込み

① GUIからのデータ読み込み



Import Wizard (インポート) - Display (表示) tab

Separator (区切り): Column (コンマ) | Range (範囲): A3:CK4... | Variable Name (変数名): 1 | Output Type (出力タイプ): table | Replace (置き換え) | Next (次を含むインポートできないセル) | NaN

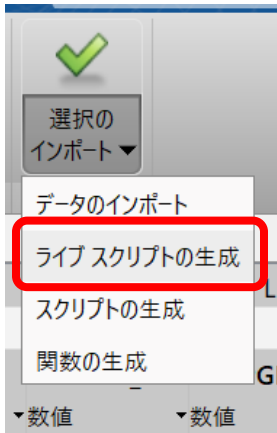
flightData_1Hz.csv

flightData1Hz1															
	Time	ABRK	ACMT	AIL_1	AIL_2	ALTS	APFD	ATEN	A_T	BLV	BPGR_1	BPGR_2	BPYR_1	BPYR_2	CALT
テキスト	数値	数値	数値	数値	数値	数値	数値	数値	数値	数値	数値	数値	数値	数値	数値
1	Time	ABRK	ACMT	AIL_1	AIL_2	ALTS	APFD	ATEN	A_T	BLV	BPGR_1	BPGR_2	BPYR_1	BPYR_2	CALT
2		DEG		DEG	DEG	FEET					PSI	PSI	PSI	PSI	
3	10-May-20...	119.983558...	59	91.8788909...	91.5924835...	6000	2	0	1	0	0	43.9454231...	2373.05297...	0	0
4	10-May-20...	119.983558...	59	91.8993530...	91.5924835...	6000	2	0	1	0	0	43.9454231...	2373.05297...	0	0
5	10-May-20...	119.983558...	60	91.8993530...	91.5924835...	6000	2	0	1	0	0	43.9454231...	2368.17016...	0	0
6	10-May-20...	119.983558...	59	91.8993530...	91.5720214...	6000	2	0	1	0	0	43.9454231...	2368.17016...	0	0
7	10-May-20...	119.983558...	60	91.8788909...	91.5720214...	6000	2	0	1	0	0	43.9454231...	2368.17016...	0	0
8	10-May-20...	119.983558...	60	91.9607238...	91.5720214...	6000	2	0	1	0	0	43.9454231...	2368.17016...	0	0
9	10-May-20...	119.983558...	59	91.9198150...	91.5515670...	6000	2	0	1	0	0	43.9454231...	2368.17016...	0	0
10	10-May-20...	119.983558...	60	91.8788909...	91.5924835...	6000	2	0	1	0	0	43.9454231...	2368.17016...	0	0
11	10-May-20...	119.983558...	59	91.8175201...	91.5515670...	6000	2	0	1	0	0	48.8282470...	2368.17016...	0	0
12	10-May-20...	119.983558...	60	91.8175201...	91.5311126...	6000	2	0	1	0	0	43.9454231...	2368.17016...	0	0
13	10-May-20...	119.983558...	60	91.8175201...	91.5311126...	6000	2	0	1	0	0	43.9454231...	2368.17016...	0	0
14	10-May-20...	119.983558...	59	91.8175201...	91.5106506...	6000	2	0	1	0	0	43.9454231...	2363.28735...	0	0
15	10-May-20...	119.983558...	60	91.8175201...	91.5311126...	6000	2	0	1	0	0	48.8282470...	2368.17016...	0	0
16	10-May-20...	119.983558...	59	91.8379821...	91.5106506...	6000	2	0	1	0	0	43.9454231...	2363.28735...	0	0
17	10-May-20...	119.983558...	60	91.8379821...	91.5311126...	6000	2	0	1	0	0	43.9454231...	2363.28735...	0	0
18	10-May-20...	119.983558...	60	91.8379821...	91.4901885...	6000	2	0	1	0	0	43.9454231...	2910.16357...	0	0
19	10-May-20...	119.983558...	59	91.8584365...	91.5311126...	6000	2	0	1	0	0	43.9454231...	2910.16357...	0	0



データ読み込み

② GUI操作と等価のライブスクリプト 自動生成



テキスト ファイルからのデータのインポート

次のテキスト ファイルからデータをインポートするスクリプト:

ファイル名: C:\Users\mtaguchi\OneDrive - MathWorks\Training\datacentricengineeringteams-master\exploringflightdata\flightDa

MATLAB からの自動生成日: 2019/04/11 17:51:50

インポート オプションの設定

```

1  opts = delimitedTextImportOptions("NumVariables", 89);
2
3  % 範囲と区切り記号の指定
4  opts.DataLines = [3, Inf];
5  opts.Delimiter = ",";
6
7  % 列名と型の指定
8  opts.VariableNames = ["Time", "ABRK", "ACMT", "AIL_1", "AIL_2", "ALTS", "APFD", "ATEN", "A_T", "BLV", "BPGR_1",
9  opts.VariableTypes = ["string", "double", "double", "double", "double", "double", "double", "double", "double", "double",
10 opts = setvaropts(opts, 1, "WhitespaceRule", "preserve");
11 opts = setvaropts(opts, 1, "EmptyFieldRule", "auto");
12 opts.ExtraColumnsRule = "ignore";
13 opts.EmptyLineRule = "read";
14
15 % データのインポート
16 flightData1Hz1 = readtable("C:\Users\mtaguchi\OneDrive - MathWorks\Training\datacentricengineeringteams-master\

```





データ読み込み

③ ライブスクリプトの関数化 → 以降の読み込みを自動化

ファイル名(N):

ファイルの種類(T):

保存(S) キャンセル

コマンドウィンドウ

MATLAB のご利用がはじめての場合は、[入門](#)をご覧ください。

```
>> Import1HzData
flightData1Hz =
4636×89 table
    Time    ABRK    ACMT    AIL_1    AIL_2    ALTS    APFD    ATEN    A_T    BLV    BPGR_1    BPGR_2    BPYR_1    BPYR_2    CALT    CASS    CRSS    DFGS    DWPT
    _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____
    NaT    119.98    59    91.879    91.592    6000    2    0    1    0    0    43.945    2373.1    0    0    128    33.926    1    42188
    NaT    119.98    59    91.899    91.592    6000    2    0    1    0    0    43.945    2373.1    0    0    128    33.926    1    42188
    NaT    119.98    60    91.899    91.592    6000    2    0    1    0    0    43.945    2368.2    0    0    128    33.926    1    42188
    NaT    119.98    59    91.899    91.572    6000    2    0    1    0    0    43.945    2368.2    0    0    128    33.926    1    42188
    NaT    119.98    60    91.879    91.572    6000    2    0    1    0    0    43.945    2368.2    0    0    128    33.926    1    42188
    NaT    119.98    60    91.961    91.572    6000    2    0    1    0    0    43.945    2368.2    0    0    128    33.926    1    42188
    NaT    119.98    59    91.92    91.552    6000    2    0    1    0    0    43.945    2368.2    0    0    128    33.926    1    42188
    NaT    119.98    60    91.879    91.592    6000    2    0    1    0    0    43.945    2368.2    0    0    128    33.926    1    42188
    NaT    119.98    59    91.818    91.552    6000    2    0    1    0    0    48.828    2368.2    0    0    128    33.926    1    42188
    NaT    119.98    60    91.818    91.531    6000    2    0    1    0    0    43.945    2368.2    0    0    128    33.926    1    42188
    NaT    119.98    60    91.818    91.531    6000    2    0    1    0    0    43.945    2368.2    0    0    128    33.926    1    42188
    NaT    119.98    59    91.818    91.511    6000    2    0    1    0    0    43.945    2363.3    0    0    128    33.926    1    42188
    NaT    119.98    60    91.818    91.531    6000    2    0    1    0    0    43.945    2368.2    0    0    128    33.926    1    42188
```




データアクセス

読み込んだデータ  `flightData1Hz` 時間 × センサ `4636x89 table`

データ.変数名

```
flightData1Hz.FIRE_1
```

変数型でソート

```
flightData1Hz{: ,vartype("numeric")}
```

```
ans = 4636x88
106 ×
    0.0001    0.0001    0.0001    0.0001    0.0060 ...
    0.0001    0.0001    0.0001    0.0001    0.0060
    0.0001    0.0001    0.0001    0.0001    0.0060
    0.0001    0.0001    0.0001    0.0001    0.0060
    0.0001    0.0001    0.0001    0.0001    0.0060
    0.0001    0.0001    0.0001    0.0001    0.0060
    0.0001    0.0001    0.0001    0.0001    0.0060
    0.0001    0.0001    0.0001    0.0001    0.0060
    0.0001    0.0001    0.0001    0.0001    0.0060
    0.0001    0.0001    0.0001    0.0001    0.0060
    ⋮
```



タイムテーブルへの変換

タイムテーブルへ変換

```
t1hz = table2timetable(flightData1Hz)
```

```
t1hz = 4636x88 timetable
```

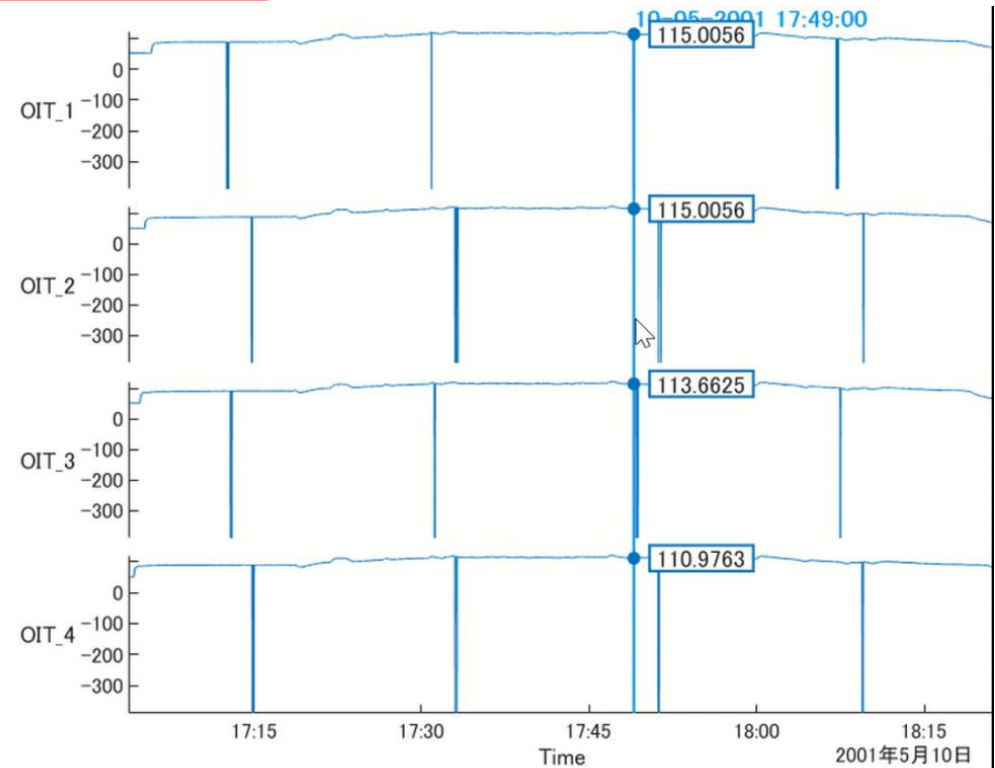
	Time	ABRK	ACMT	AIL_1
1	10-May-20...	119.9836	59	91.8789
2	10-May-20...	119.9836	59	91.8994
3	10-May-20...	119.9836	60	91.8994
4	10-May-20...	119.9836	59	91.8994
5	10-May-20...	119.9836	60	91.8789
6	10-May-20...	119.9836	60	91.9607
7	10-May-20...	119.9836	59	91.9198
8	10-May-20...	119.9836	60	91.8789
9	10-May-20...	119.9836	59	91.8175

時間によるデータアクセス

```
starttime = datetime(2001,5,10,17,30,0);
endtime = datetime(2001,5,10,17,30,5);
t1hz(timerange(starttime,endtime),:)
```

可視化

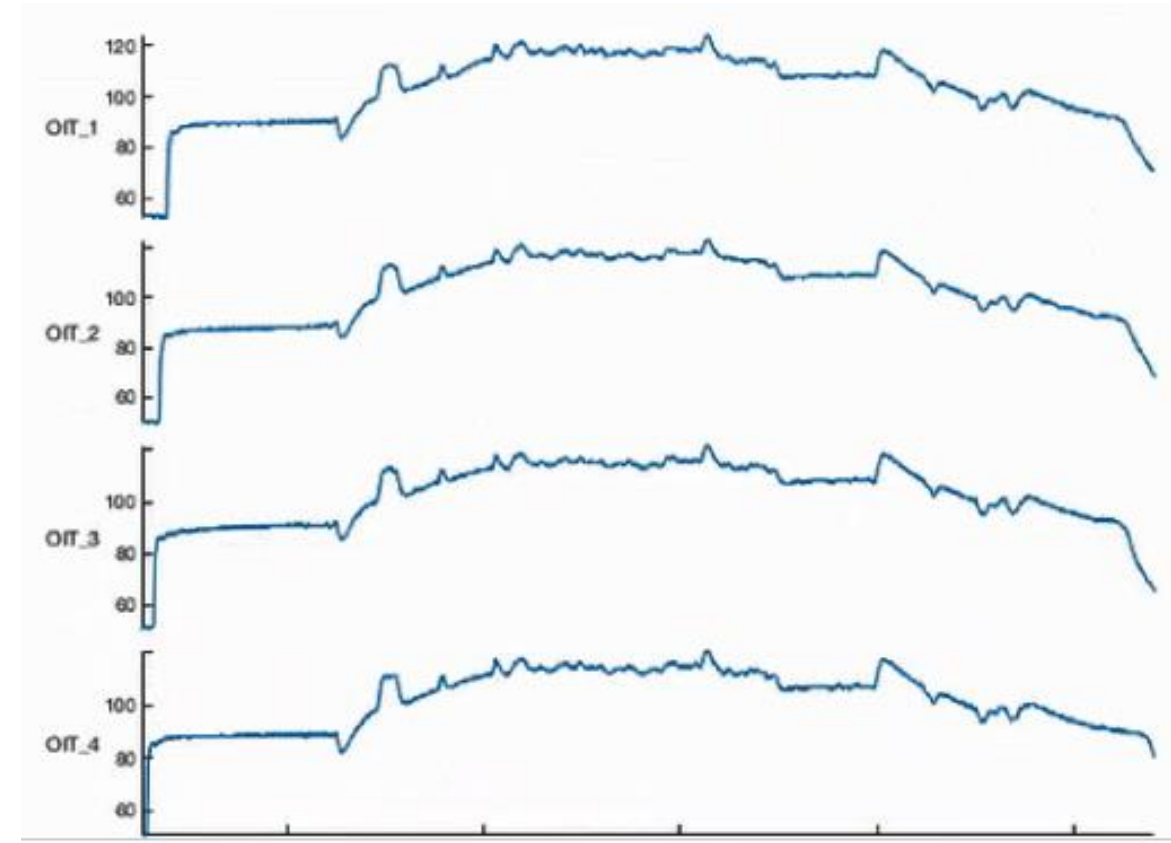
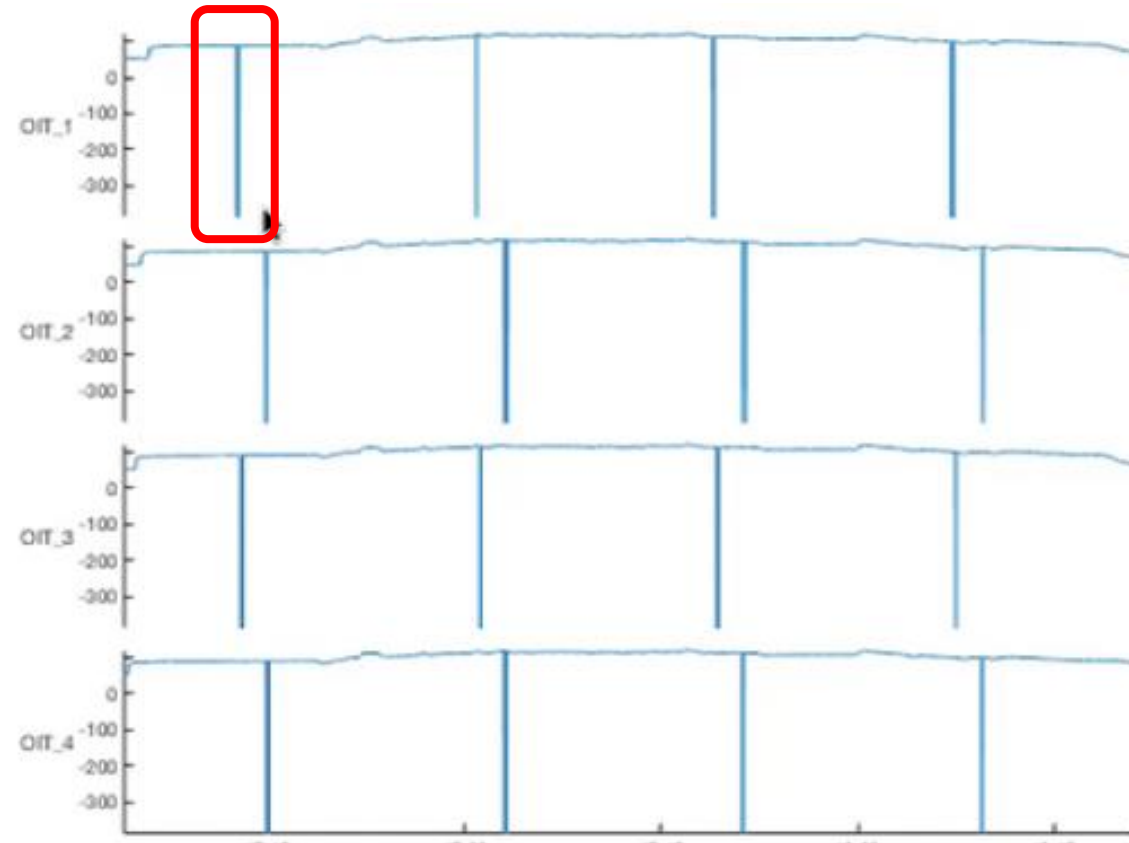
```
stackedplot(t1hz,{'OIT_1','OIT_2','OIT_3','OIT_4'});
```





異常値の削除・補間処理

```
filloutliers()
```

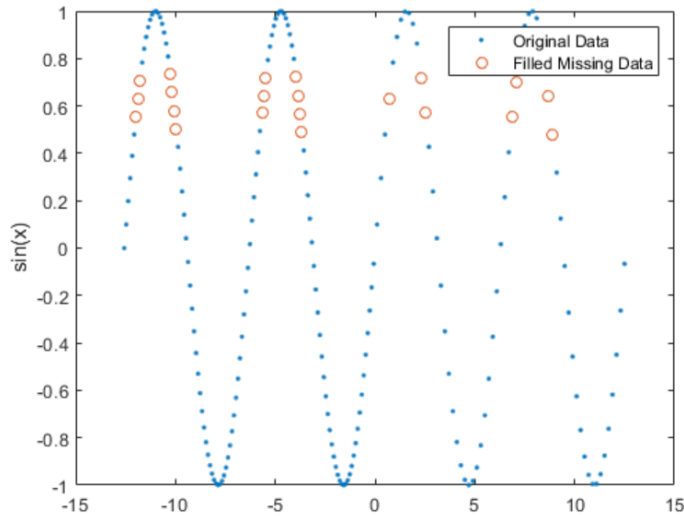




便利な前処理関数

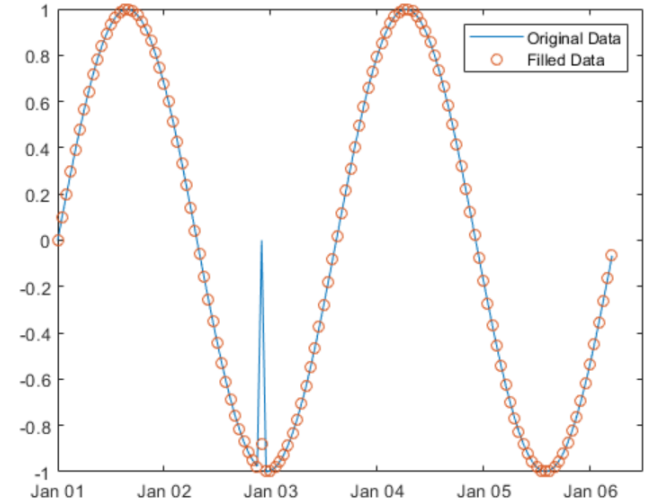
欠損値

`ismissing`
`rmissing`
`fillmissing`



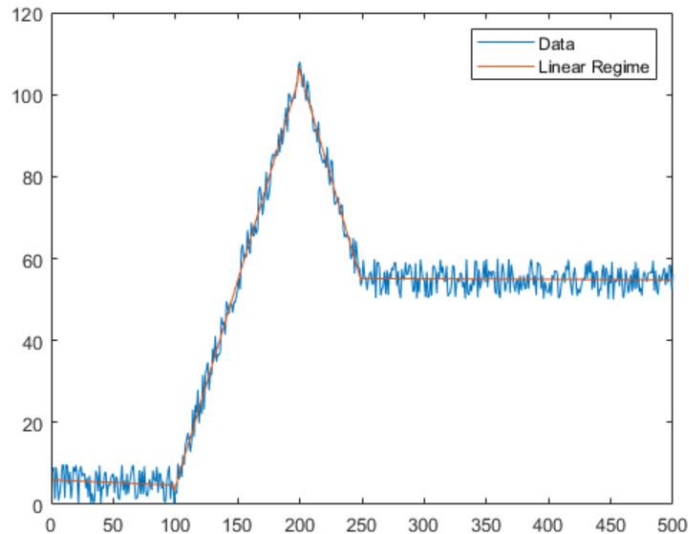
外れ値

`isoutlier`
`rmoutliers`
`filloutliers`



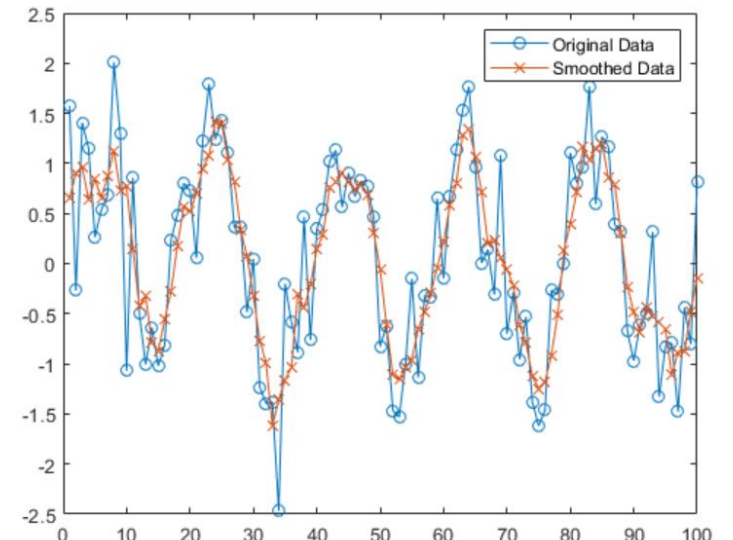
急激な変化点

`ischange`



平滑化

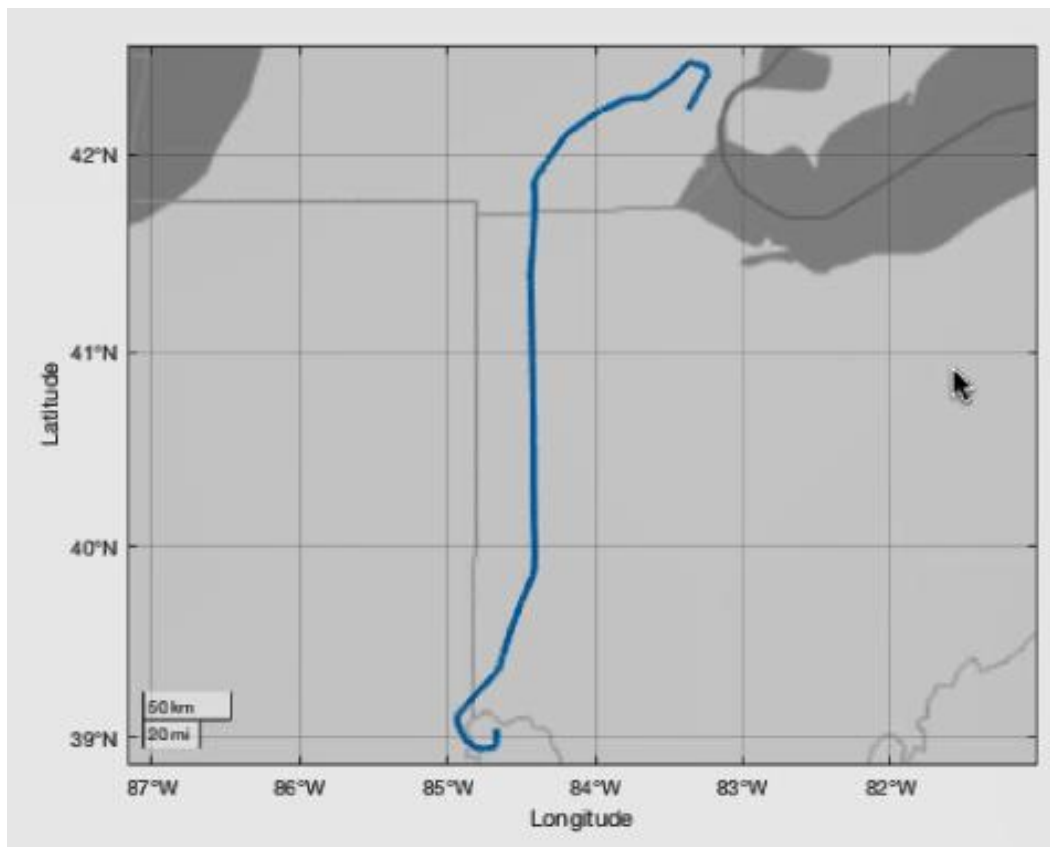
`smoothdata`



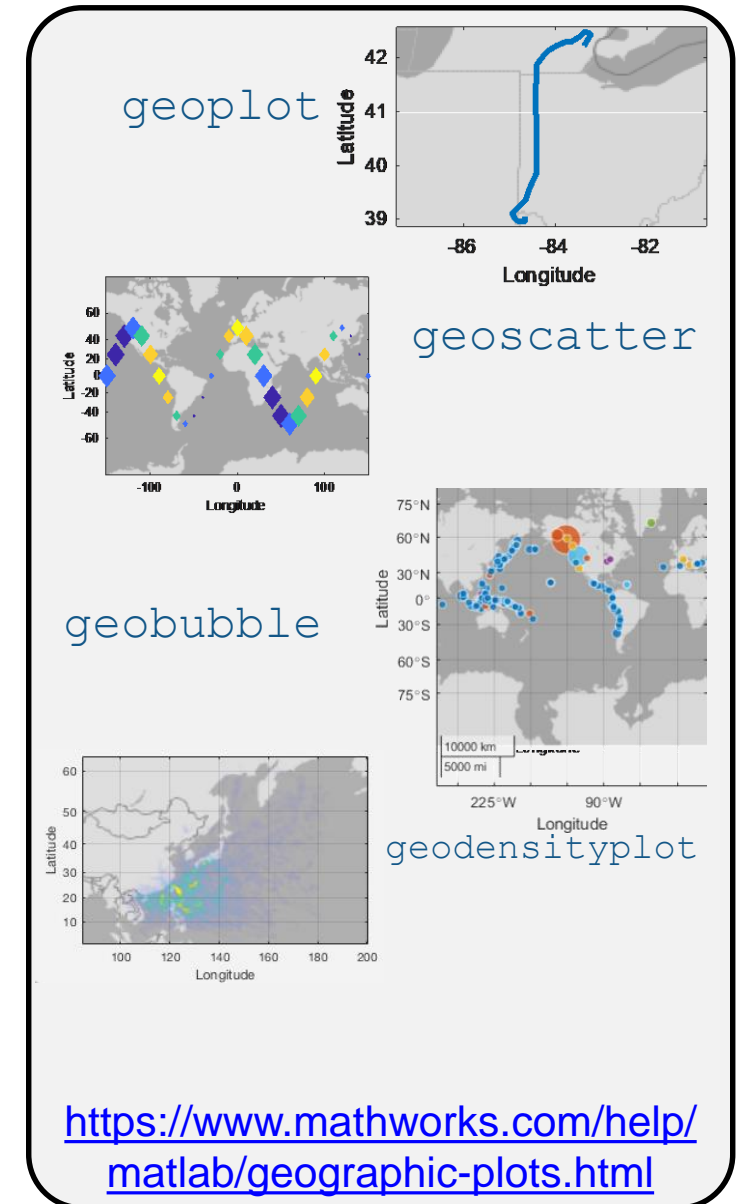
可視化

地図上に軌跡をプロット

```
geoplot(t.LATP(inFlight),t.LONP(inFlight),'LineWidth',3);
```



MATLAB EXPO 2019





ポイントまとめ



GUIで**簡単操作**・操作と**等価の関数も1クリック**で作成



行列数を使わないデータアクセス・データソート



時系列解析に便利な**タイムテーブル**



データの**前処理は関数1行**で実施



地理プロットで簡単可視化

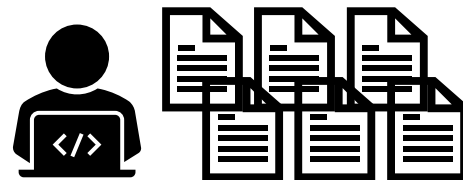
①アドホックな解析

- データ読み込み
- データアクセス
- 前処理
- 可視化



②ビッグデータ解析

- ビッグデータの読み込み
- ビッグデータの解析・深掘り



フライトデータ概要

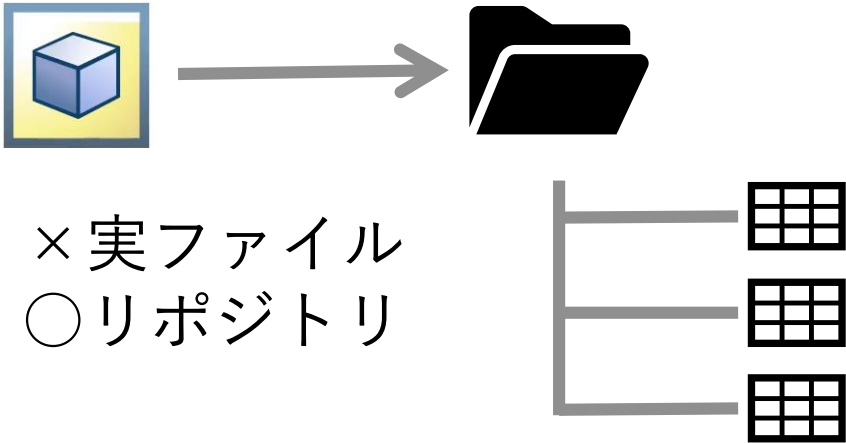
- 35 機
- 180,000 飛行
- 容量 300 GB
- 出典:
 - NASA Dash Link: Sample Flight Data
 - <https://c3.nasa.gov/dashlink/projects/85/>



ビッグデータの読み取り

大規模なデータ集合の読取り

```
ds = datastore('*.*csv')
```



⇒ PCのRAMメモリ以上の
データ読み取りが可能

大規模なデータを1変数で管理

```
t = read(ds);
```

MATLAB EXPO 2019

Built-In Datastores	
General	datastore
	spreadsheetDatastore
	tabularTextDatastore
	fileDatastore
Database	databaseDatastore
Image	imageDatastore
	denoisingImageDatastore
	randomPatchExtractionDatastore
	pixelLabelDatastore
	augmentedImageDatastore
Audio	audioDatastore
Predictive Maintenance	fileEnsembleDatastore
	simulationEnsembleDatastore
Simulink	simulationDatastore
Automotive	mdfDatastore

  データストアのカスタマイズ (カスタムデータストア)

R2019a

自作のカスタムデータストアでデータストア読み込み

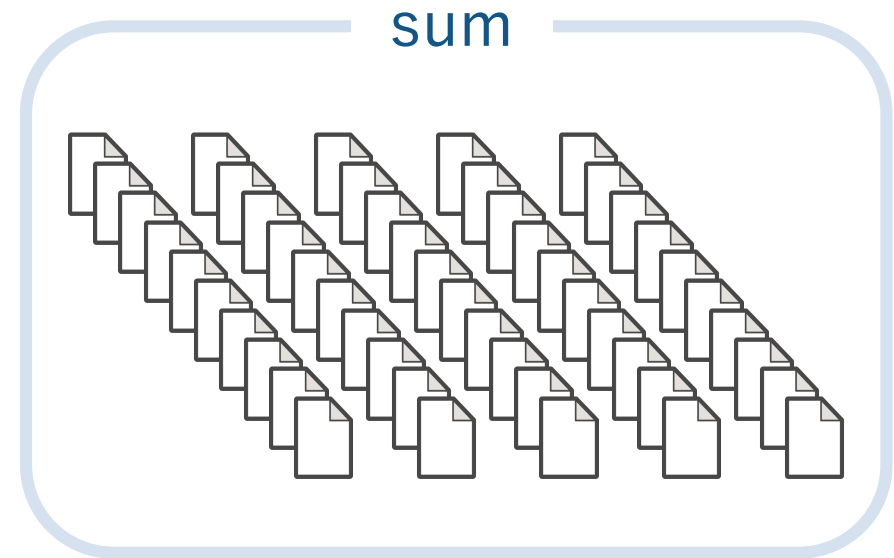
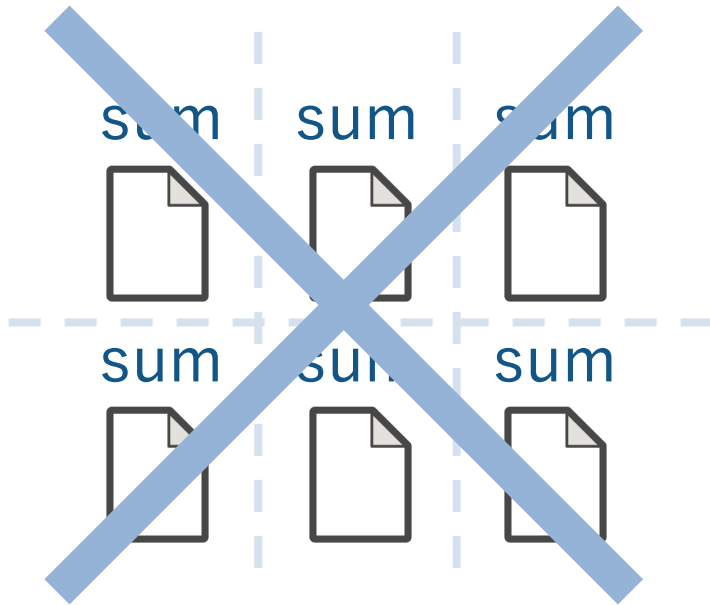
```
>> ds = flightDataStore('/Volumes/Ext/Data/FlightData/675/')  
  
ds =  
  
flightDataStore with properties:  
  
    CurrentFileIndex: 1  
      NumberOfFiles: 5700  
    VariableNames: {1x186 cell}  
SelectedVariableNames: {1x186 cell}  
    AddFlightIdentifier: 0
```

オリジナルフォーマットのビッグデータ解析にも対応可能



ポイントまとめ

- 😊 データストアによりRAMサイズ以上の**大容量データを一括読取り**
- 😊 独自のファイルもデータストア化し一元操作

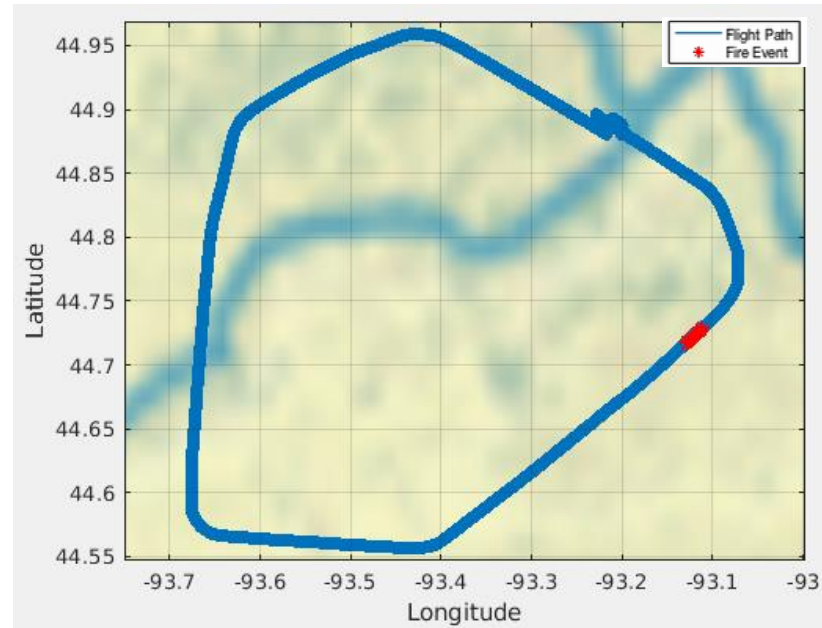


ビッグデータ活用例 – 事象の深掘り

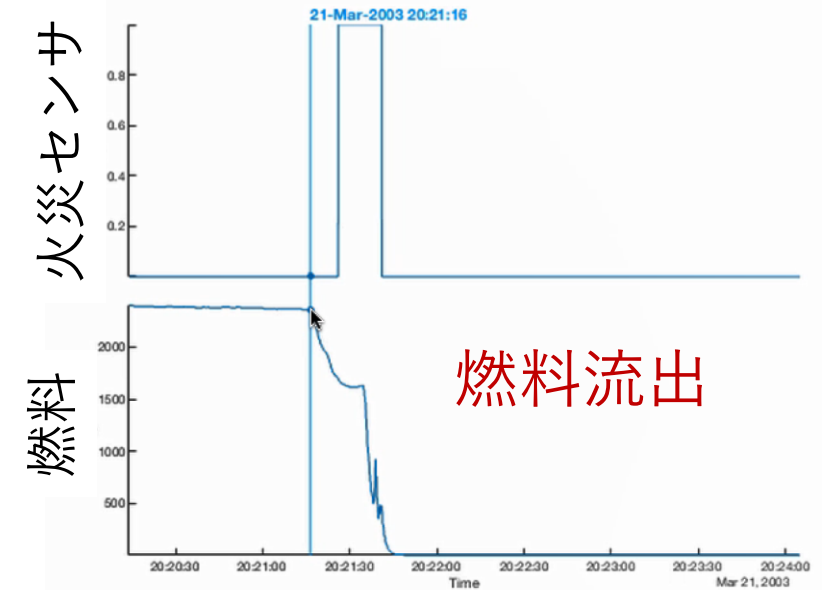
センサ2で火災を検知



フライト軌跡の確認



燃料データの確認

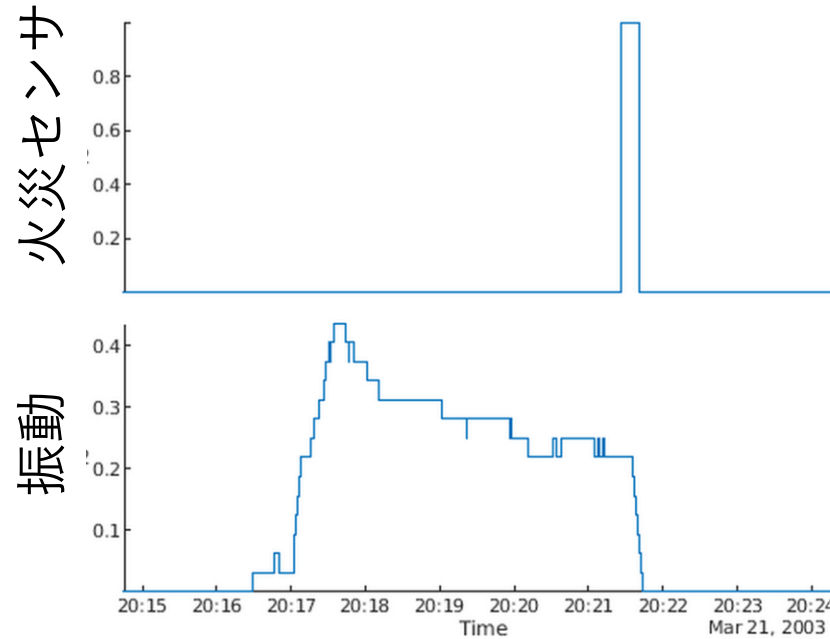


ビッグデータ活用例 – 事象の深掘り

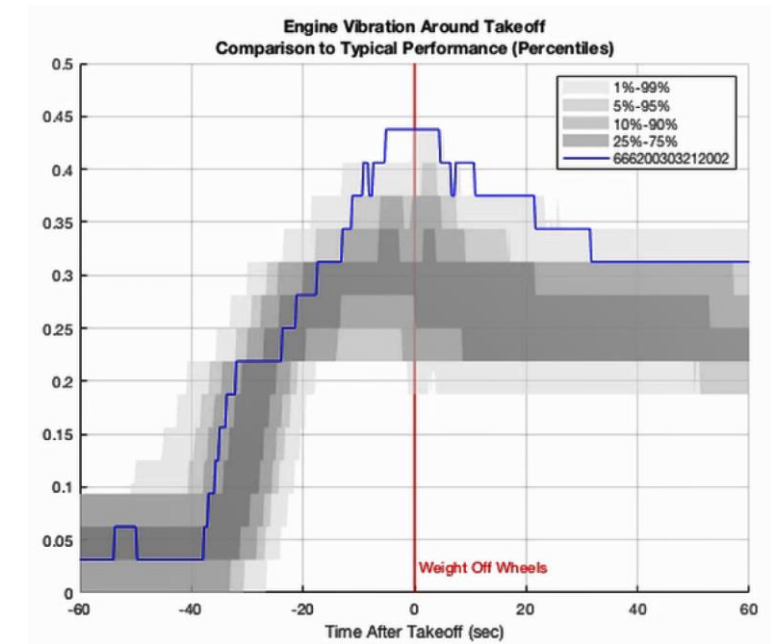
センサ2で火災を検知



振動データの確認



振動確率分布と比較



- 火災の原因をデータから検討
- 再発防止に役立つ知見を得る

振動による燃料流出が火災の原因か？

①アドホックな解析

- データ読み込み
- データアクセス
- 前処理
- 可視化



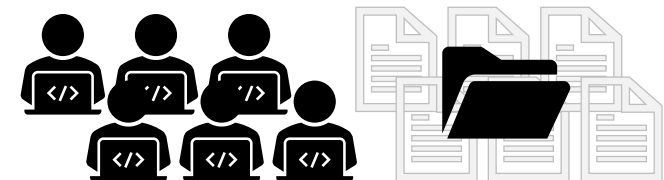
②ビッグデータ解析

- ビッグデータの読み込み
- ビッグデータの解析



③プロジェクト化

- プロジェクトの作成・共有
- 大人数で解析



プロジェクト化 R2019a

必要なファイルをまとめてプロジェクト化 → 簡単配布・作業自動化

The screenshot shows the MATLAB Project Explorer window for a project named 'FlightDataToolbox'. The window displays a list of files and folders with their status and classification. The 'Dependency Analysis' (依存関係の解析) icon is highlighted with a red box.

名前	ステータス	Classification
+ tests	✓	
doc	✓	
flightDataStore.m	✓	Design
FlightDataToolbox.mltbx	✓	
info.xml	✓	Derived
PackageToolbox.prj	✓	
readFlightFile.mlx	✓	Design



プロジェクトによる業務一元化 お役立ち機能 R2019a

ファイルの依存関係を可視化 → プロジェクトの簡単解読

The screenshot displays the MATLAB R2019a interface with the 'Dependency Analysis' tool active. The main window shows a project named 'Airframe Example' with a dependency graph. The graph illustrates the relationships between various files, including MATLAB files, Simulink models, and C source files. The 'Dependency Type' legend indicates the following categories:

- Function (6)
- Class (1)
- Simulink Model (7)
- Simulink Library (1)
- C Source (1)
- Coder TLC Source (1)
- MEX-file (1)
- Simulink Data Dictionary (3)

The 'Impact View' on the right provides a summary of the project's dependencies:

- Airframe Example**
- Root: /.../airframe
- Analyzed: 3/6/19, 8:0...
- Products (3)
- MATLAB 9.7
- Simulink 9.3
- Simulink Coder 9.1
- External Toolboxes (1)
- C/C++ Library Header
- Problems (1)
- Not in project (1)

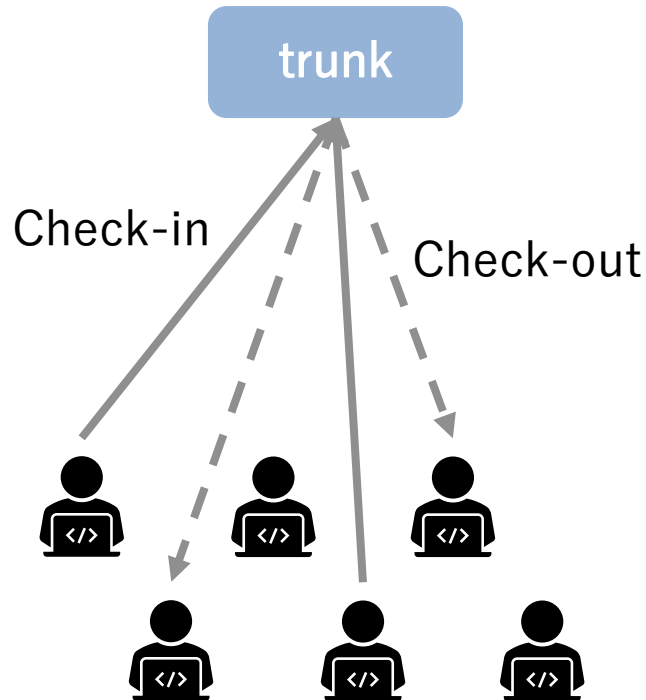
The 'Files' pane on the left shows the project structure, and the 'Git' pane at the bottom indicates the current branch is 'master' with a normal status.



プロジェクトによる業務一元化 お役立ち機能

ソース管理 → 複数人での作業支援・バージョン管理・差分表示

Subversion (SVN) Gitとの連携



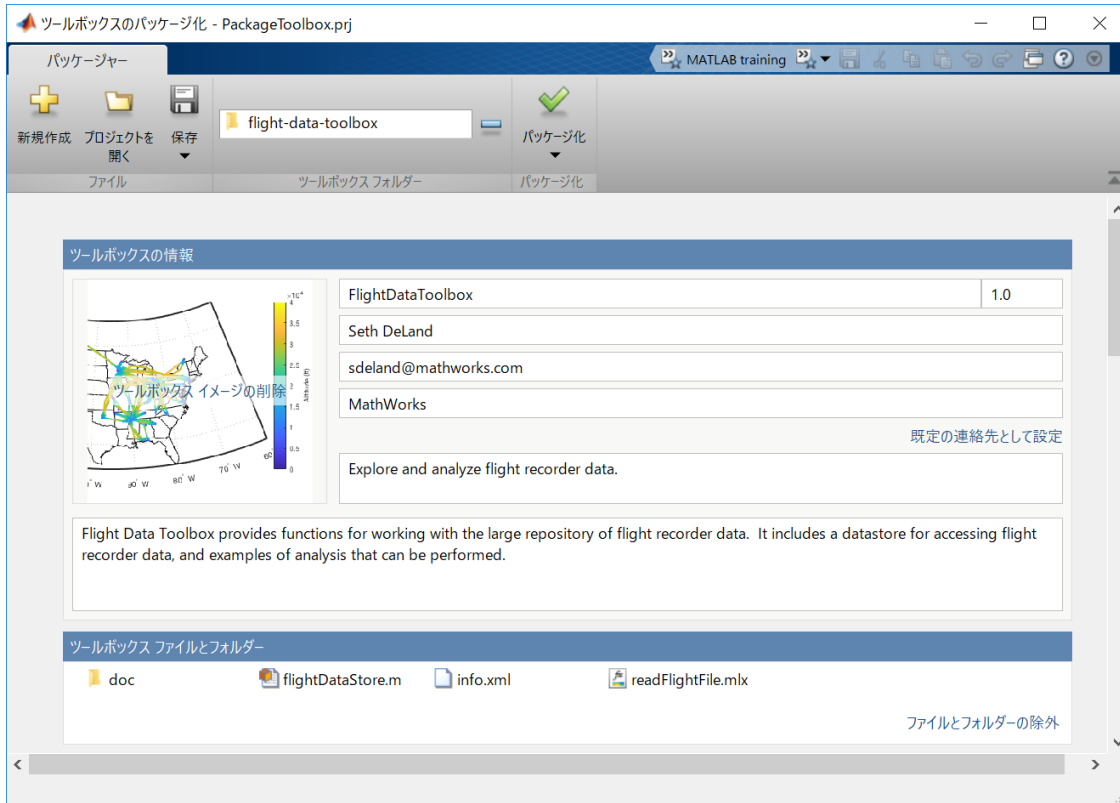
```

23 methods                                     [22 unmodified lines hidden]
24                                             . methods
25 function fds = flightDataStore(location)    . function fds = flightDataStore(location)
26                                             > %FLIGHTDATASTORE Dastore for reading flight data
27                                             > % DS = FLIGHTDATASTORE(LOCATION) creates a flightDataStore DS base
28                                             > % of the data. LOCATION is a folder that contains .MAT files
29                                             > % that contain flight data. Alternatively, LOCATION can be
30                                             > % an individual .MAT file of flight data. If LOCATION is a
31                                             > % folder, .MAT files are searched for recursively (there can
32                                             > % be multiple levels of folders).
33                                             > %
34                                             > % flightDataStore supports the following functions:
35                                             > %
36                                             > % read - Read one file's worth of data.
37                                             > % readall - Read all of the data.
38                                             > % hasdata - Returns true if there is more data.
39                                             > % reset - Reset the flightDataStore to the start of the data
40                                             > % signalMetaInfo - Return meta-data about each signal
41                                             > %
42                                             > % Example:
43                                             > % Create a flightDataStore
44                                             > % ds = flightDataStore('DataFolder')
45                                             > % We are only interested in the Altitude data
46                                             > % ds.SelectedVariableNames = {'ALT'}
47                                             > % Find the maximum altitude achieved
48                                             > % maxAlt = 0;
49                                             > % while hasdata(ds)
50                                             > %     tt = read(ds);
51                                             > %     maxAlt = max(maxAlt,max(tt.ALT));
52                                             > % end
53                                             > % maxAlt
54                                             .
55 % Use the DsFileSet class to find and keep track of files
56 fds.FileSet = matlab.io.datastore.DsFileSet(location,...
27 % Use the DsFileSet class to find and keep track of files
28 fds.FileSet = matlab.io.datastore.DsFileSet(location,...
[279 unmodified lines hidden]

```

検証済みツールとして共有 R2019a

ツールボックス



ツール説明書

Using a FlightDataStore

場所: sers/mtaguchi/AppData/Roaming/MathWorks/MATLAB%20Add-Ons/Toolboxes/FlightDataToolb

Using a FlightDataStore

All the Flight Data

```
ds = flightDataStore('/Users/sdeland/MATLAB/projects/FlightDataToolbox/+tests/sampleData/')

ds =
    flightDataStore with properties:
        CurrentFileIndex: 1
        NumberOfFiles: 6
        VariableNames: {1x186 cell}
        SelectedVariableNames: {1x186 cell}
        AddFlightIdentifier: 0
```

Data for a Subset of Signals

```
signalMetaInfo(ds)
```

ans = 186x5 table

	Signals	Rate	Units	Description	Alpha
1	'ABRK'	1.0000	'DEG'	'AIRBRAKE POS...	'ABRK'
2	'ACID'	0.2500	"	'AIRCRAFT NUM...	'ACID'
3	'ACMT'	1.0000	"	'ACMS TIMING ...	'ACMT'
4	'AIL_1'	1.0000	'DEG'	'AILERON POSI...	'AIL_1'
5	'AIL_2'	1.0000	'DEG'	'AILERON POSI...	'AIL_2'
6	'ALT'	4.0000	'FEET'	'PRESSURE ALT...	'ALT'
7	'ALTR'	4.0000	'FT/MIN'	'ALTITUDE RATE'	'ALTR'
8	'ALTS'	1.0000	'FEET'	'SELECTED ALT...	'ALTS'
9	'AOA1'	4.0000	'DEG'	'ANGLE OF ATT...	'AOA1'
10	'AOA2'	4.0000	'DEG'	'ANGLE OF ATT...	'AOA2'

```
ds.SelectedVariableNames = {'LATP', 'LONP'}
```

```
ds =
    flightDataStore with properties:
```



ポイントまとめ

- 😊 **MATLABプロジェクト**による簡単配布・作業自動化
- 😊 **プロジェクト可視化**機能による簡単解読
- 😊 **ソース管理機能連携**により複数人で作業が可能
- 😊 **ツールボックス化**し配布・さらなる協業を加速

①アドホックな解析

- データ読み込み
- データアクセス
- 前処理
- 可視化



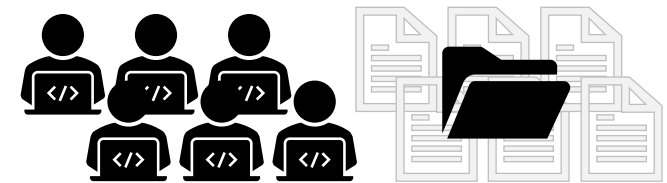
②ビッグデータ解析

- ビッグデータの読み込み
- ビッグデータの解析



③プロジェクト化

- プロジェクトの作成・共有
- 大人数で解析



まとめ



- データ解析のカギを握る **前処理 MATLAB**で効率良く**実施**可能
- MATLABは**大容量データ解析**に対応、ビッグデータ活用の橋渡し
- **プロジェクト化**により組織横断プロジェクトへ拡大

本日正式紹介したツール

- MATLAB **R2019a**

**A3 信号処理アプリケーション
のためのAI技術**
Time:15:30-16:00