

知って得する！データサイエンスを快適にする MATLAB基本テクニック

MathWorks Japan
アプリケーションエンジニアリング部
田口美紗

データサイエンス？

ニューラルネット？

K-NN？



ランダムフォレスト？



SVM？



ワークフロー全体を考える

データアクセス



センサー



ファイル



データベース

データ解析



データ 探索



前処理



専門的な
アルゴリズム

開発



AI モデル



アルゴリズム
開発



モデリング &
シミュレーション

実装&展開



デスクトップ
アプリ



エンタープライ
ズシステム



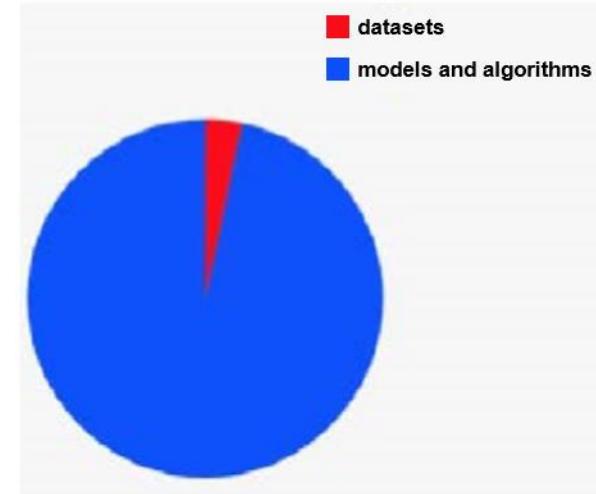
組み込み
デバイス

AIの作業で最も労力を要しているデータの準備…

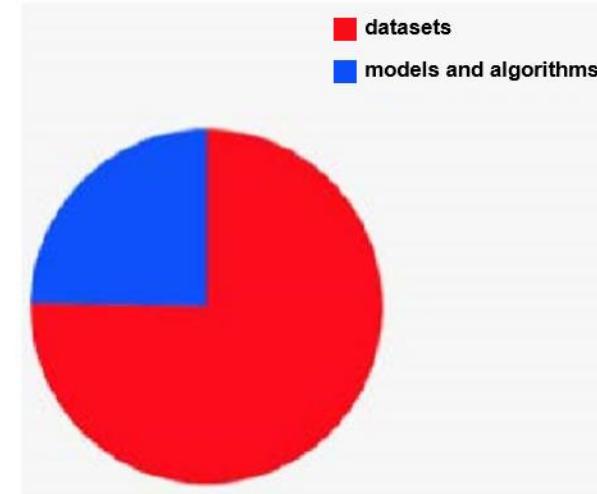
生データをモデリングや解析のために変換することは非常に重要なステップです。

何に時間をかけるか… **データセット**
モデル・アルゴリズム

PhD



Tesla



Source: Andrej Karpathy slide from TrainAI 2018

データサイエンス成功への鍵:

```
# -*- coding: utf-8 -*-
"""
Created on Wed Aug 19 14:04:47 2020

@author: hyoshino
"""

import unittest

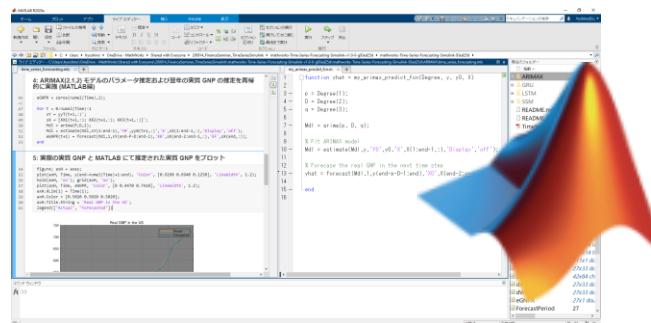
def fun(x):
    return x + 1

class MyTest(unittest.TestCase):
    def test(self):
        self.assertEqual(fun(3), 4)
```

本質的な課題

非本質的作業:

ツールの使い方・コマンド、プログラミング、ライブラリの整合性、実装、バージョン管理、知財…

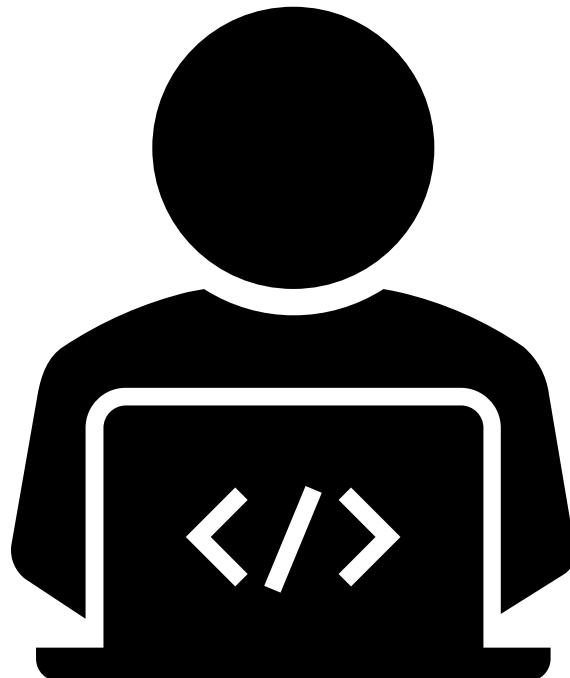


本質的な課題

最小限の
非本質的作業

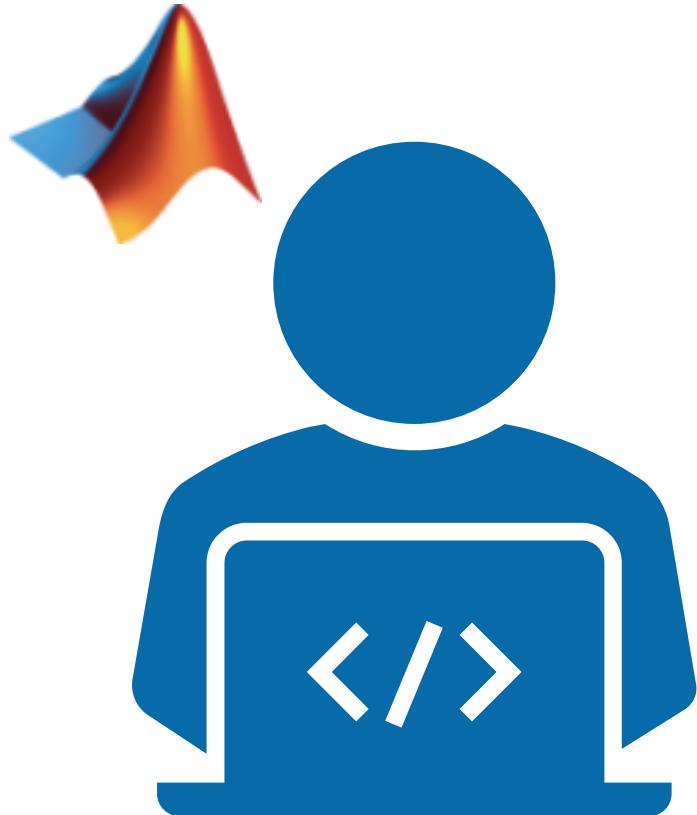
改良 / 次のプロジェクト

天才エンジニアの先輩の話



- 音声合成の研究者
- AIアルゴリズムをオープンソーススプラットフォームを使って開発
- C++, C の使い手
- アルゴリズム開発 → ハードウェア実装 (C++) もできてしまう天才

私はこうします



Screenshot of the MATLAB Coder interface showing the generation of C code from a MATLAB function.

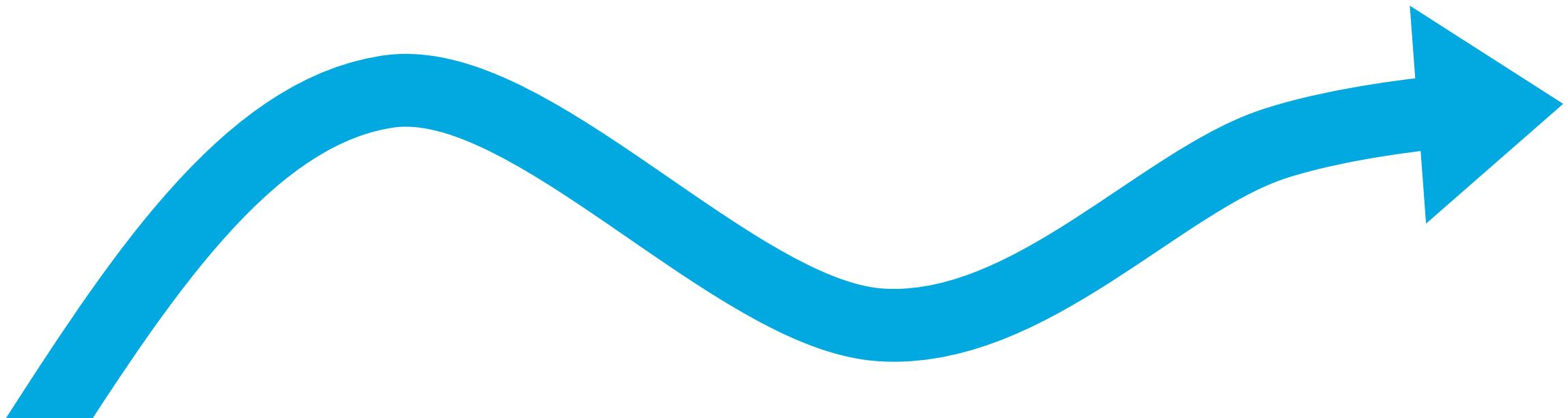
The MATLAB Coder interface shows the following details:

- Project:** MATLAB Code - classifyAirplane.prj
- Function:** classifyAirplane
- Output Files:** A list of generated files including: classifyAirplane_da, classifyAirplane_err, classifyAirplane_en, classifyAirplane_ini, classifyAirplane_ter, classifyAirplane.c, CompactClassificationTree, main.c, rt_nonfinite.c, rtGetInf.c, rtNaN.c, classifyAirplane_da, classifyAirplane_en, and classifyAirplane_err.
- Code Preview:** The generated C code for classifyAirplane.c is displayed, starting with the header block.
- Variables:** A table showing variables and their types:

Variable	Type	Size
X	double	:Inf x 14
label	cell	:Inf x 1
label(:)	char	1 x 6
Mdl		1 x 1
- Status:** A message at the bottom indicates "Source Code generation succeeded." with a "View Report" link.

講演のメッセージ

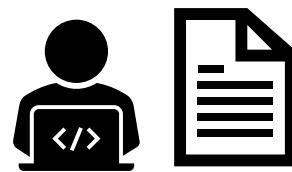
“本質的な課題” の解決
こそが価値があること



本日のアジェンダ

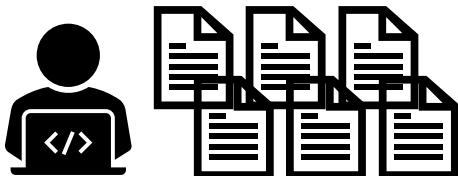
①アドホックな解析

- データ読み込み
- データアクセス
- 前処理
- 可視化



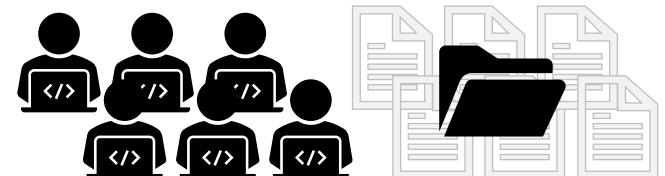
②ビッグデータ解析

- ビッグデータの読み込み
- ビッグデータの解析・深堀り



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



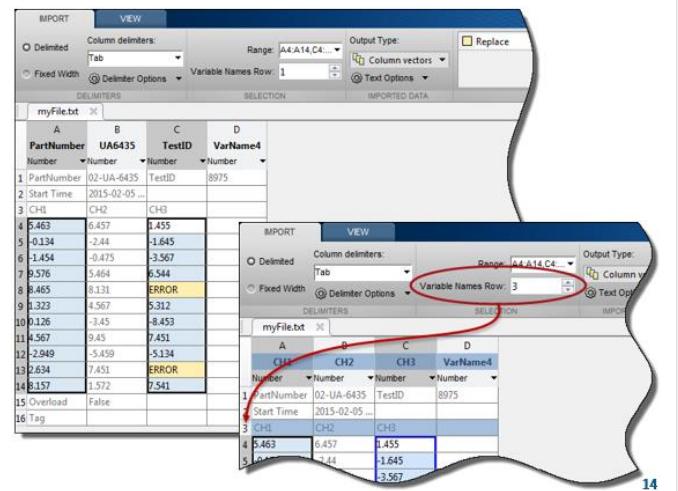
データインポートツールアプリ

MathWorks

DEMO

- GUIで即座にインポート
 - 取り込み範囲の指定
 - 変数型の指定
 - 欠損値処理の指定
- MATLABコード生成機能
 - 2回目以降はコマンド

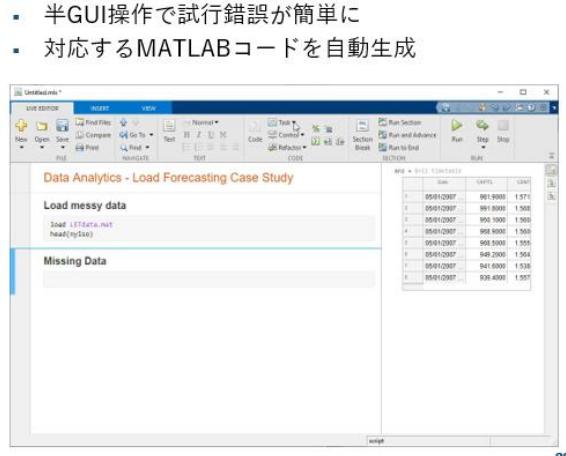
(1)



ライブタスクから最適な前処理設定を素早く探索

MathWorks

DEMO



(3)

[紹介] ライブエディター

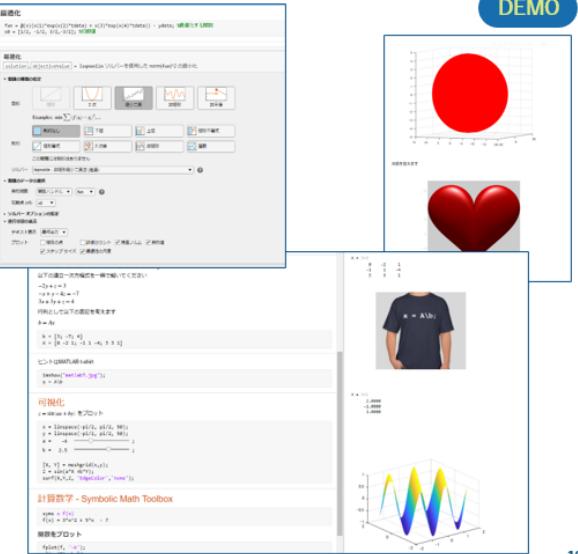
MathWorks

DEMO

- 高機能スクリプトエディター
 - 数式 (LaTeX対応)
 - 文字、画像を埋め込み
 - 計算の出力、グラフを表示

- 半GUIを実装
 - ライブタスク, e.g., 最適化、前処理
 - ライブコントロール, e.g., スライダー
- レポート機能
 - Word, pdf, LaTeX, ...

(2)



16

プロジェクト化 共同作業・ヴァージョン管理

MathWorks

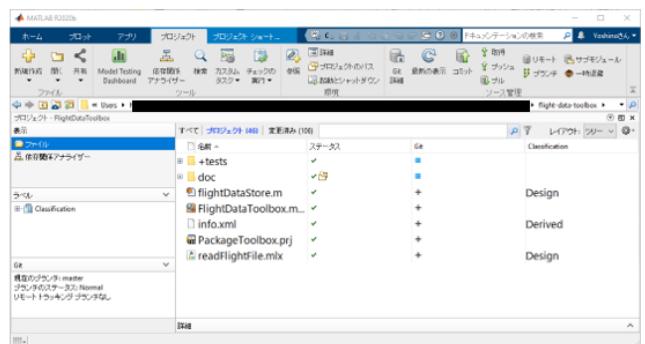
DEMO



- ファイル、データ、テスト等を1か所で管理
- 作業の整理、チームでの共同作業を円滑にする
- Git™, SVN®を使用したソース管理ツール (GitはMATLABに同梱)

(4)

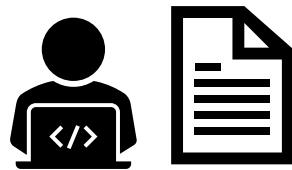
MATLABプロジェクトについて



41

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



フライトデータの解析例

振動センサ



緯度・経度

燃料センサ

火災センサ

時間

データの読み込み

低水準インポート関数

- ファイルポインタ
- ループで読み込み
- 詳細に設定が必要

```
filename = 'mymeas.dat';
measrows = 4;
meascols = 4;

% open the file
fid = fopen(filename);

% read the file headers, find N (one value)
N = fscanf(fid, '%*s %*s\nN=%d\n\n', 1);

% read each set of measurements
for n = 1:N
    mystruct(n).mtime = fscanf(fid, '%s', 1);
    mystruct(n).mdate = fscanf(fid, '%s', 1);

    % fscanf fills the array in column order,
    % so transpose the results
    mystruct(n).meas = ...
        fscanf(fid, '%f', [measrows, meascols]);
end

% close the file
fclose(fid);
```

(例) テキストファイルからのデータの読み込み

特定ファイル形式 インポート関数

- 比較的簡単に読み込み
- ループ要らず

```
opts = delimitedTextImportOptions("NumVariables", 89);

% 範囲と区切り記号の指定
opts.DataLines = [3, Inf];
opts.Delimiter = ",";

% 列名と型の指定
opts.VariableNames = ["Time", "ABRK", "ACMT", "AIL_1", "AIL_2"];
opts.VariableTypes = ["string", "double", "double", "double"];

% ファイル レベルのプロパティを指定
opts.ExtraColumnsRule = "ignore";
opts.EmptyLineRule = "read";

% 変数プロパティを指定
opts = setvaropts(opts, "WhitespaceRule", "preserve");
opts = setvaropts(opts, "Time", "EmptyFieldRule", "auto");

% データのインポート
flightData1Hz = readtable("mydata.dat", opts);
```

データインポート ツールアプリ

- GUIで読み込み

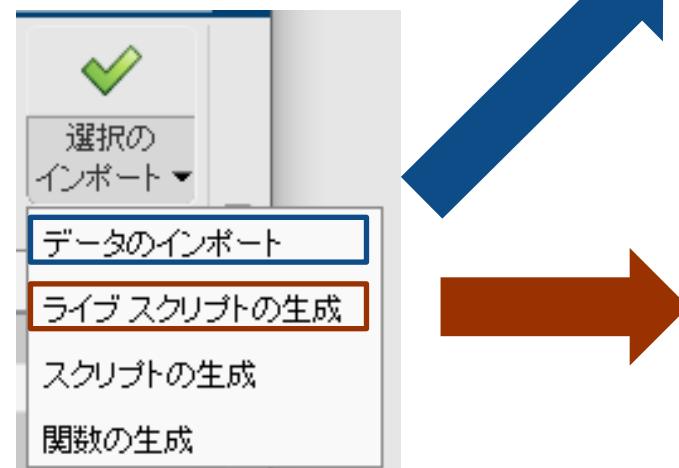


The image shows the 'Import Data' tool window in MATLAB. It displays a preview of the 'flightData1Hz' table. The table has columns labeled Time, ABRK, ACMT, AIL_1, AIL_2, ALTS, APFD, ATEN, A_T, BLV, BPGR_1, BPGR_2, and BPYR_1. The first few rows of data are shown, including dates like '10-May-20...' and numerical values for the other columns. A large red arrow points from the 'Import Data' icon in the ribbon down to this window.

	A	B	C	D	E	F	G	H	I	J	K	L	M
Time	ABRK	ACMT	AIL_1	AIL_2	ALTS	APFD	ATEN	A_T	BLV	BPGR_1	BPGR_2	BPYR_1	
1	Time	ABRK	ACMT	AIL_1	AIL_2	ALTS	APFD	ATEN	A_T	BLV	BPGR_1	BPGR_2	P
2	DEG	DEG	DEG	DEG	FEET					PSI	PSI	PSI	
3	10-May-20...	119.983558...	59	91.8788909...	91.5924835...	6000	2	0	1	0	0	43.9454231...	2373.05297...
4	10-May-20...	119.983558...	59	91.8993530...	91.5924835...	6000	2	0	1	0	0	43.9454231...	2373.05297...
5	10-May-20...	119.983558...	60	91.8993530...	91.5924835...	6000	2	0	1	0	0	43.9454231...	2368.7016...
6	10-May-20...	119.983558...	59	91.8993530...	91.5720214...	6000	2	0	1	0	0	43.9454231...	2368.7016...
7	10-May-20...	119.983558...	60	91.8788909...	91.5720214...	6000	2	0	1	0	0	43.9454231...	2368.7016...
8	10-May-20...	119.983558...	60	91.9607238...	91.5720214...	6000	2	0	1	0	0	43.9454231...	2368.7016...
9	10-May-20...	119.983558...	59	91.9198150...	91.5515670...	6000	2	0	1	0	0	43.9454231...	2368.7016...
10	10-May-20...	119.983558...	59	91.8788909...	91.5924835...	6000	2	0	1	0	0	43.9454231...	2368.7016...
11	10-May-20...	119.983558...	59	91.8175201...	91.5515670...	6000	2	0	1	0	0	48.8282470...	2368.7016...
12	10-May-20...	119.983558...	60	91.8175201...	91.5311126...	6000	2	0	1	0	0	43.9454231...	2368.7016...
13	10-May-20...	119.983558...	60	91.8175201...	91.5311126...	6000	2	0	1	0	0	43.9454231...	2368.7016...
14	10-May-20...	119.983558...	59	91.8175201...	91.5106506...	6000	2	0	1	0	0	43.9454231...	2363.2875...

データ読み込み用のコード生成

- データのインポート
- コードの生成
 - ライブスクリプト
 - スクリプト
 - 関数化



ワークスペース

名前	値	クラス	サイズ	バイト
flightData1Hz	4636x89 t...	table	4636x89	3682582

untitled.mlx *

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\hyoshino\OneDrive - MathWorks\Shared with Everyone\201023_MATLAB_Preprocessing\datacentricengineeringteams\expl...

MATLABからの自動生成日: 2020/10/23 19:40:43

インポート オプションの設定およびデータのインポート

```
1 opts = delimitedTextImportOptions("NumVariables", 89);
2
3 % 論理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", "BPGR_2"];
9 opts.VariableTypes = ["string", "double", "double", "double", "double", "double", "double", "double", "double", "double"];
10
11 % ファイル レベルのプロパティを指定
12 opts.ExtraColumnsRule = "ignore";
13 opts.EmptyLineRule = "read";
14
15 % 変数プロパティを指定
16 opts = setvaropts(opts, "Time", "WhitespaceRule", "preserve");
17 opts = setvaropts(opts, "Time", "EmptyFieldRule", "auto");
18
19 % データのインポート
20 flightData1Hz = readtable("C:\Users\hyoshino\OneDrive - MathWorks\Shared with Everyone\201023_MATLAB_Preprocessing\datacentricengineeringteams\expl...
```

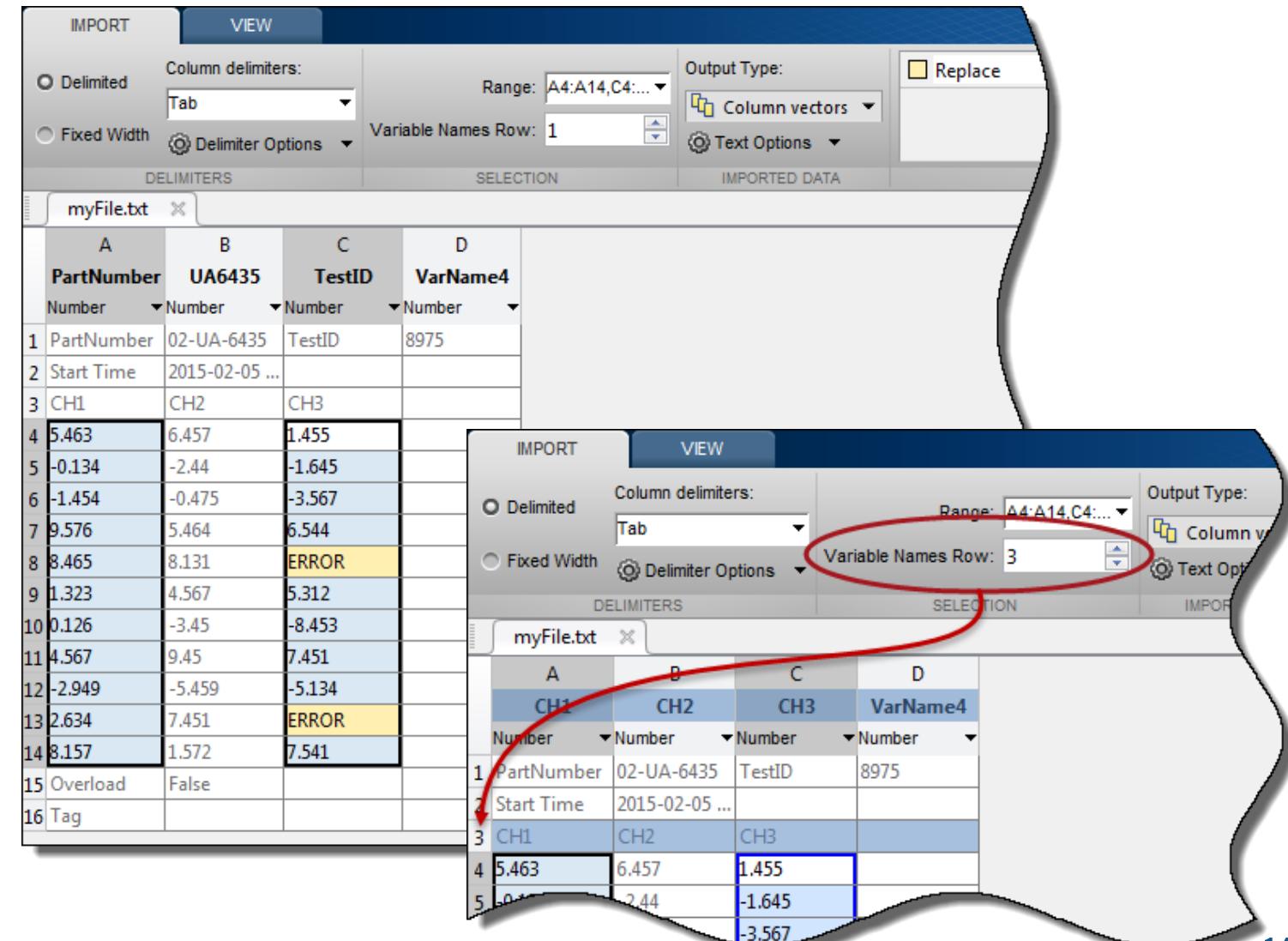
一時変数のクリア

```
21 clear opts
```

データインポートツールアプリ

- GUIで即座にインポート
 - 取り込み範囲の指定
 - 変数型の指定
 - 欠損値処理の指定
- MATLABコード生成機能
 - 2回目以降はコマンド

データインポートツールアプリ



[紹介] ライブエディター

DEMO

- 高機能スクリプトエディター
 - 数式 (LaTeX対応)
 - 文字、画像を埋め込み
 - 計算の出力、グラフを表示
- 半GUIを実装
 - ライブタスク, e.g., 最適化、前処理
 - ライブコントロール, e.g., スライダー
- レポート機能
 - Word, pdf, LaTeX, …



ライブスクリプトで微分方程式論の授業

LiveEditor



微分方程式論(Introduction of Differential Equations)

version 2.0.0 (5.19 MB) by Ben T. Nohara

第1部：基礎編 第1章：微分方程式とは 第2章：積分法、解析解と数値解 第3章：ベクトル場、相図：線形微分方程式と非線形微分方程式の違い、Hartman-Grobman定理 第4章：数値解法の基礎事項：オイラー法。。。その他

[+ Follow](#) [Download](#)

[Overview](#) [Examples](#)

学問としての微分方程式論の重要性は、諸現象の核心部分を抽出しモデル化して、物理量の動態を把握することにあり、これにより未 来時間での動態を予測できることにある。インストラクターはつきの事項に注力する。すなわち、微分方程式の基礎理論をマスター すること(勿論であるが、ともすれば理学部数学科の理論倒れになるのを防ぐため、理論と応用のバランスに心掛け(理論のない学 問は発展せず、応用のない学問は空虚である)、MATLABを駆使して専門科目(機械工学、電気工学、化学、建築など)の持つ課題解決 への発展を目指す。

受講生はこの講義の内容を修得すれば、

- (1) 線形常微分方程式の解の導出や解軌跡を相図で表現できる。
- (2) 解くことが不可能な非線形常微分方程式の厳密解をその線形化により表現できるようになり、大局的な解の動態を把握できる。
- (3) 歴史上重要な方程式(van der Pol方程式やLotka-Volterra方程式など)のモデル化を学び、安定性の概念や非線形の取り扱いを修得 できる。
- (4) MATLABを通して微分方程式の数値解法を修得でき、解の正確さを吟味できるようになる。
- (5) エンジニアの将来として、未解決問題を微分方程式によりモデル化が可能になり、技術革新の基盤形成ができる。

ともすれば無味乾燥した微分方程式という対象の面白さを伝えるためにはどうしてもコンピュータ援用学習が必要になります。その 点、MATLABのLive Scriptは理にかなった構造と機能を持っています。Live Scriptは説明文(平文)とコードが混在した形態であり、 理論を(概念的に)理解した後に、実際にコードを実行させて概念的な理解を具現化するよう設計されています。

この講座の原稿はすべてLive Scriptで書いてあり、例や問題を皆さん実行し、考察した結果を印刷すれば皆さん一人一人の独立した 教科書になるでしょう。Live Scriptというヨットに乗って微分方程式の大海上に漕ぎ出してみませんか。皆さんがLive Scriptを操作することにより、安全な航海ができ微分方程式の提供する実りある目的にきっと入港できるでしょう。Bon voyage! どうぞ、よい!

[Cite As](#)

Ben T. Nohara (2019). 微分方程式論(Introduction of Differential Equations) (<https://www.mathworks.com/matlabcentral/fileexchange/71912-introduction-of-differential-equations>), MATLAB Central File Exchange. Retrieved December 9, 2019.



Ben T. Nohara
Tokyo City Univ.

9.1 van der Pol方程式

van der Pol方程式は、 $x = x(t)$ として

$$(9.1) \quad \ddot{x} - \epsilon(1 - x^2)\dot{x} + x = 0$$

という非線形の常微分方程式で表せられる。ここに、 ϵ は正の定数である。式(9.1)のような非線形微分方程式は一般に解くのが難しい。

さて、これをつきのように書き改めるとリミットサイクルの現象を理解するのに都合がよい。

$$(9.2) \quad \begin{cases} \dot{x} &= y - f(x) \\ \dot{y} &= -x \end{cases}$$

ここに、

$$f(x) = -\epsilon \left(x - \frac{1}{3}x^3 \right)$$

である。

9.2 シミュレーション

まず、シミュレーションにより式(9.2)の動態を観察してみよう。

```
% CODE_9100
% van der Pol 方程式
clc;clear;
epsilon = 10;%epsilonの設定
tRange = [0 20];
x0 = [-4;1];
```

データアクセス

様々なデータを一緒に扱う

読み込んだデータ

時間 × (時間列+センサ)

名前	値	クラス
flightData1Hz	4636x89 table	table

	A	B	C	D
1	Time	ABRK	ACMT	AIL_1
2		DEG		DEG
3	2001/5/10 17:03	119.9836	59	91.87889
4	2001/5/10 17:03	119.9836	59	91.89935
5	2001/5/10 17:03	119.9836	60	91.89935
6	2001/5/10 17:03	119.9836	59	91.89935
7	2001/5/10 17:03	119.9836	60	91.87889
8	2001/5/10 17:03	119.9836	60	91.96072
9	2001/5/10 17:04	119.9836	59	91.91982

テーブルデータへアクセス

```
T(rows,vars)
T{rows,vars}
T.var
T.(varindex)
T.var(rows)
T.Variables
```

S = vartype(type);
T(rows,S)
S = vartype(type);
T{rows,S}

データ.変数名

flightData1Hz.A_T
ABRK
ACMT
AIL_1
AIL_2

table → timetable 型へ

時系列データを扱うための専用の型

```
myTimeTable = table2timetable(myTable)
```

Time	Humidity_indoors	AirQuality	Humidity_outdoors	TemperatureF	PressureHg
2015-11-15 00:00:24	36	80	49	51.3	29.61
2015-11-15 01:13:35	36	80	NaN	NaN	NaN
2015-11-15 01:30:24	NaN	NaN	48.9	51.5	29.61
2015-11-15 02:26:47	37	79	NaN	NaN	NaN
2015-11-15 03:00:24	NaN	NaN	48.9	51.5	29.61

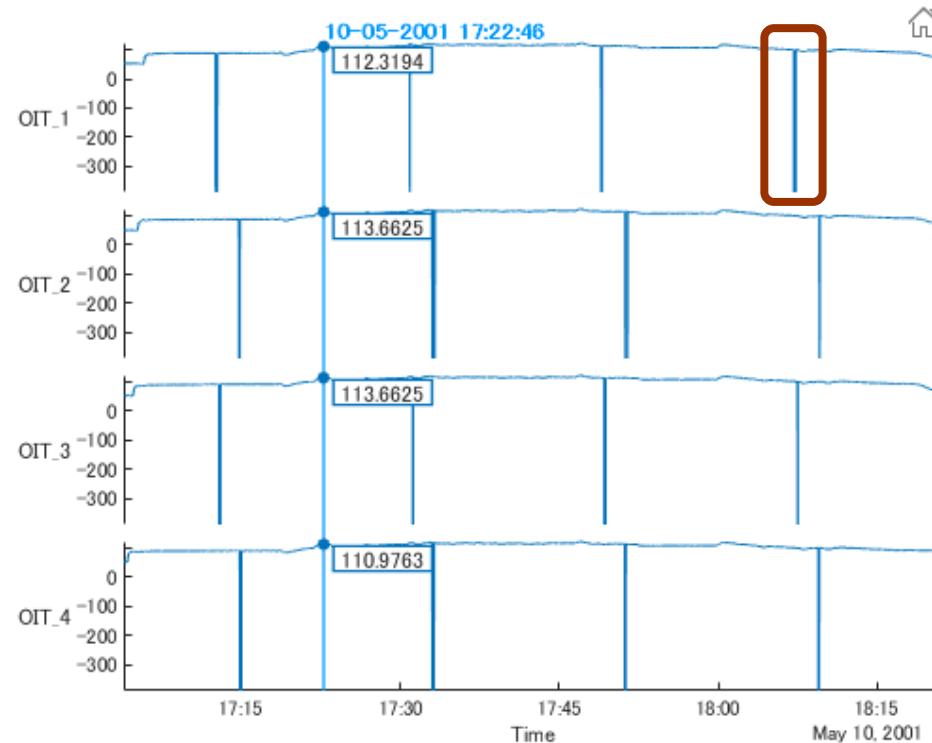
t3=synchronize(t1, t2)



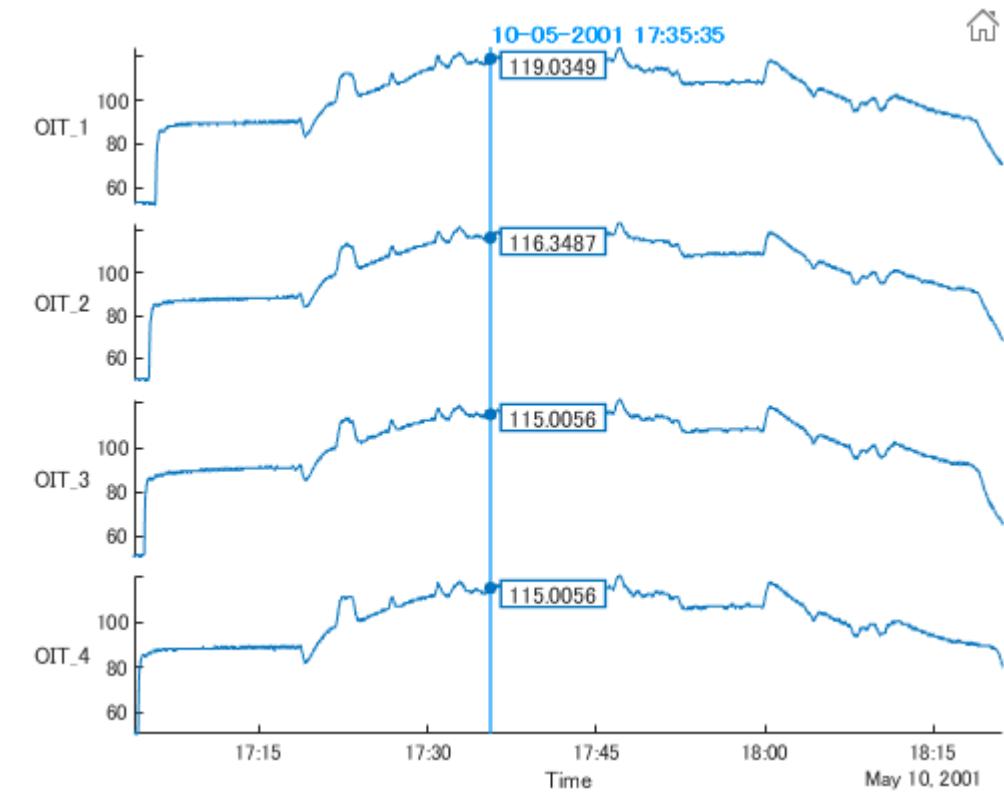
t3,t1,t2,:timetable

Time	Humidity_indoors	AirQuality	Humidity_outdoors	TemperatureF	PressureHg
2015-11-15 00:00:24	36	80	49	51.3	29.61
2015-11-15 01:13:35	36	80	48.919	51.463	29.61
2015-11-15 01:30:24	36.23	79.77	48.9	51.5	29.61
2015-11-15 02:26:47	37	79	48.9	51.5	29.61
2015-11-15 03:00:24	37	80.378	48.9	51.5	29.61

可視化と前処理



filloutliers



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

timetable

テーブル内のセンサー名

便利な前処理関数

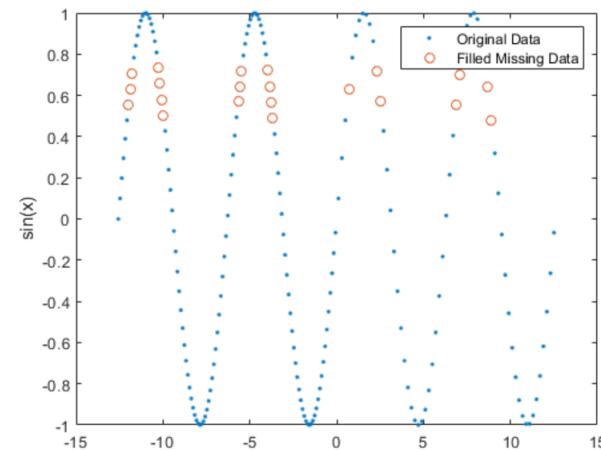
コマンド一行で前処理は完結させる

欠損値

`ismissing`

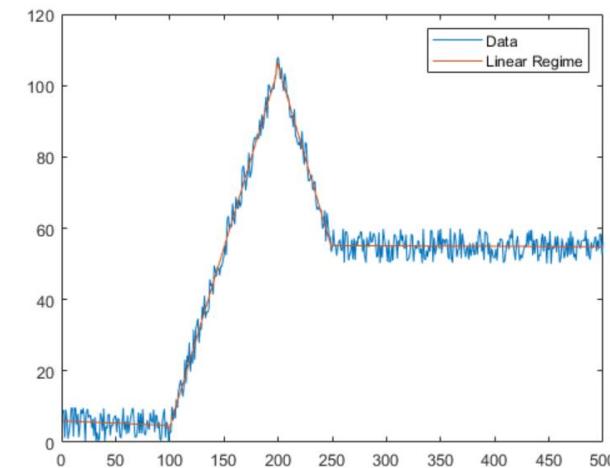
`rmmissing`

`fillmissing`



急激な変化点

`ischange`

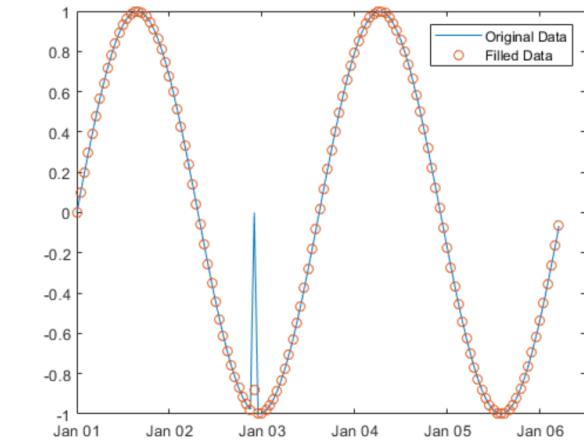


外れ値

`isoutlier`

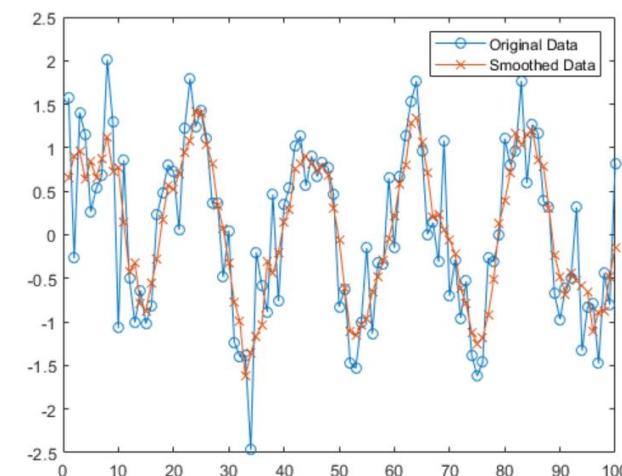
`rmoutliers`

`filloutliers`



平滑化

`smoothdata`



ライブラリタスクから最適な前処理設定を素早く探索

[DEMO](#)

データ前処理



Table
Timetable

制御システム
設計と解析

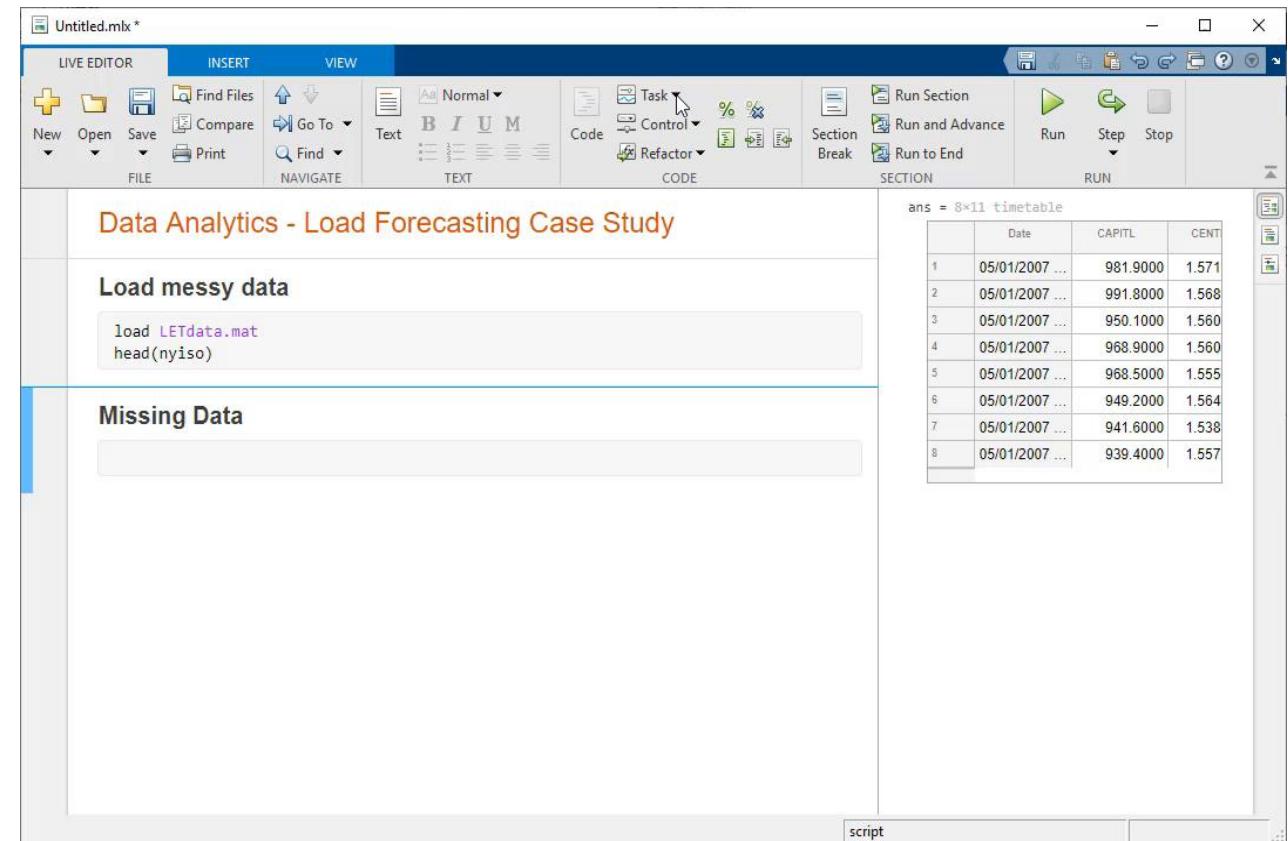
予知保全

システム同定

信号処理と通信

シンボリック
数学

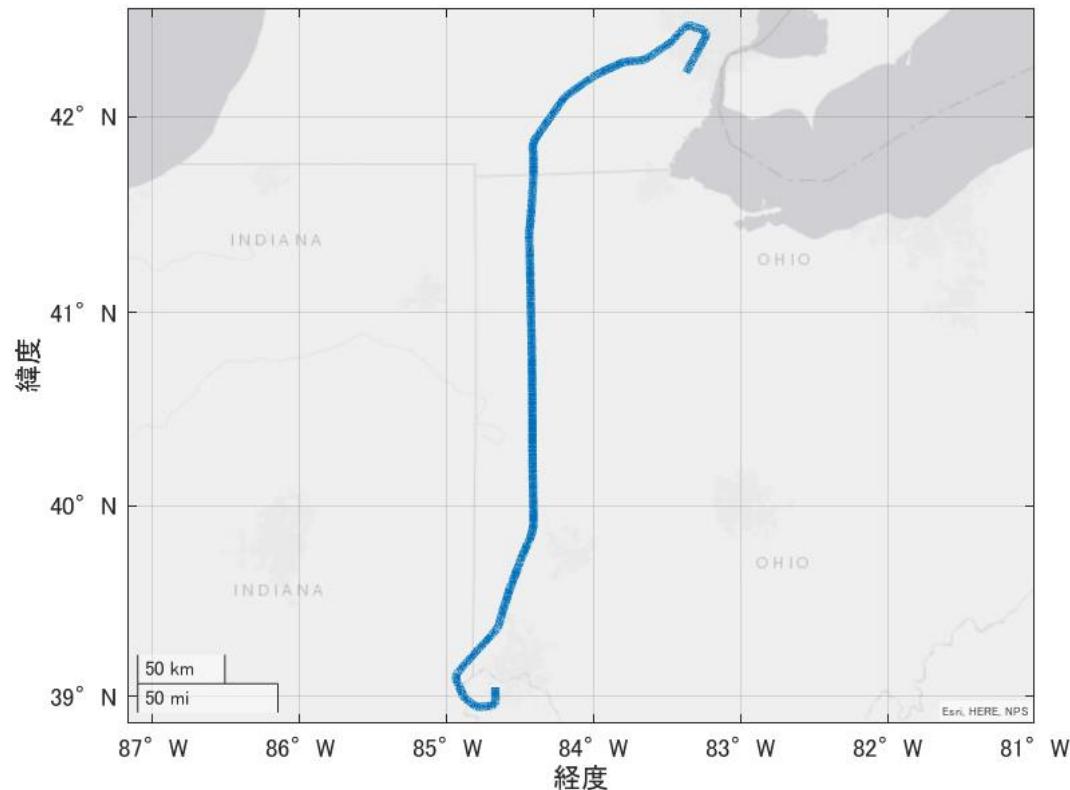
- 半GUI操作で試行錯誤が簡単に
- 対応するMATLABコードを自動生成



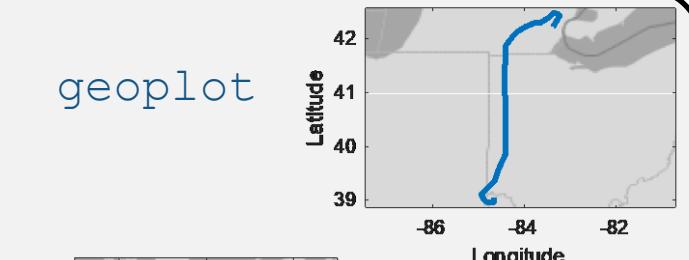
可視化 – 地図を利用する

geoplot: 地理座標に軌跡をプロット

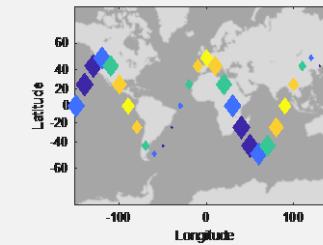
geoplot(lat, lon)



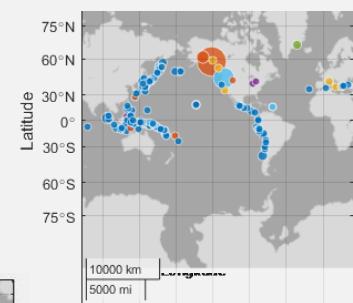
geoplot



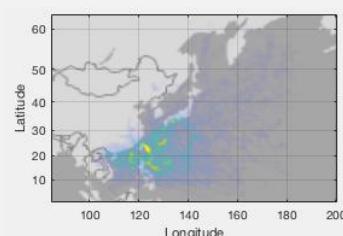
geoscatte



geobubble



geodensityplot

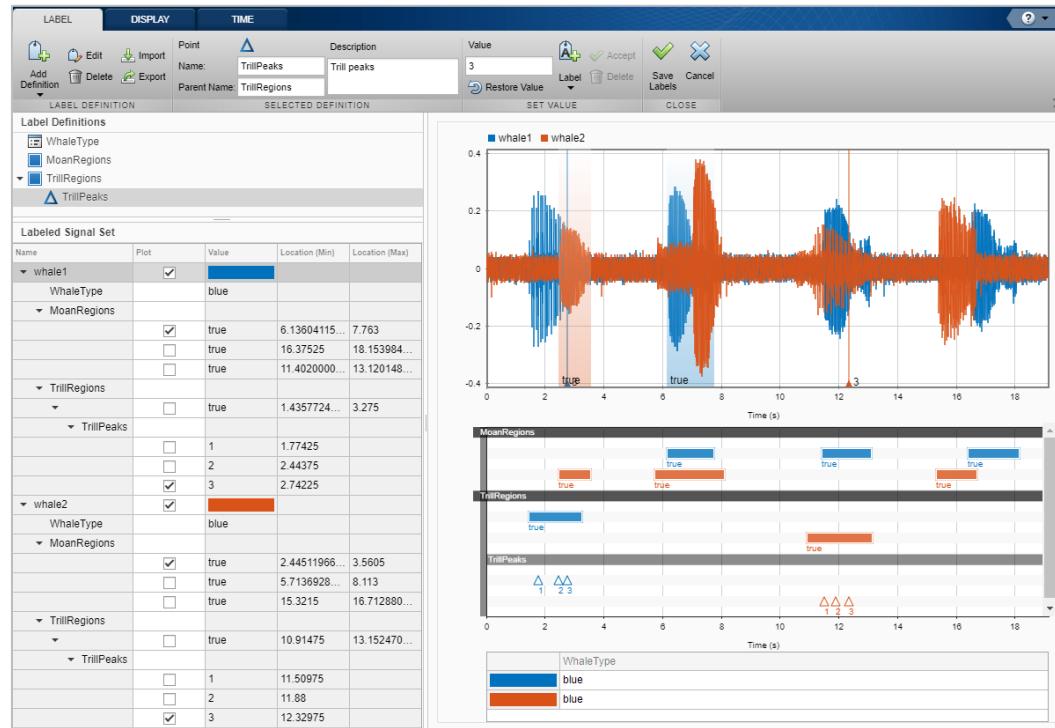


<https://www.mathworks.com/help/matlab/geographic-plots.html>

機械学習等で特に必要な前処理の高速化支援ツール
(本編とは無関係)

ラベリング作業はアプリで高速化

- マウス操作で高速ラベリング
- トラッキングアルゴリズムを使った自動ラベリング機能



Signal Labeler app
(例) イルカの鳴き声の分類

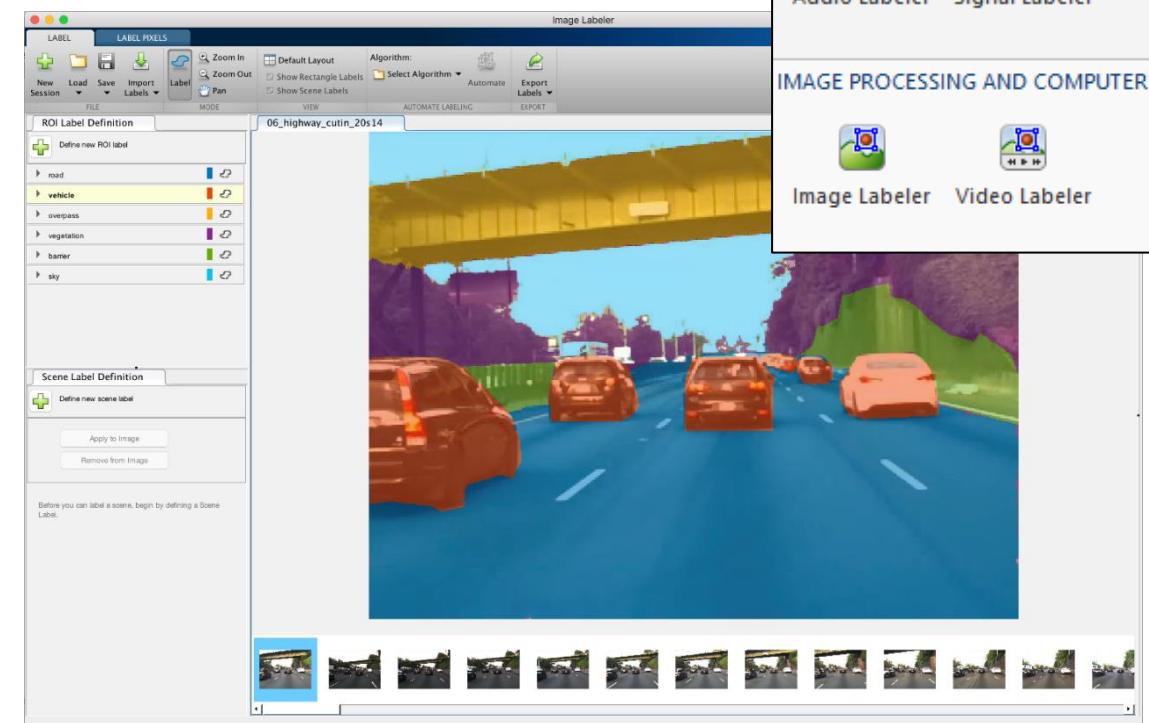


Image Labeler app
(例) セマンティックセグメンテーション

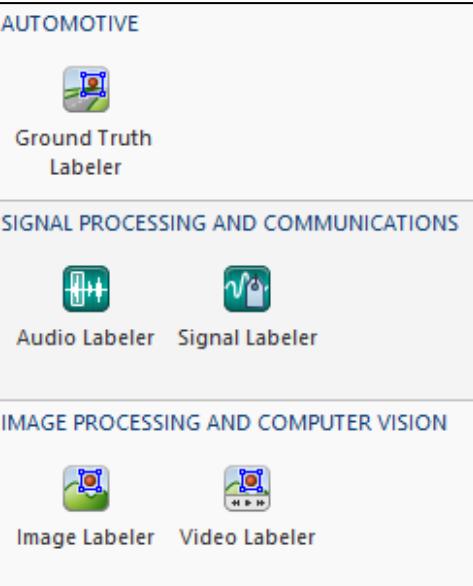
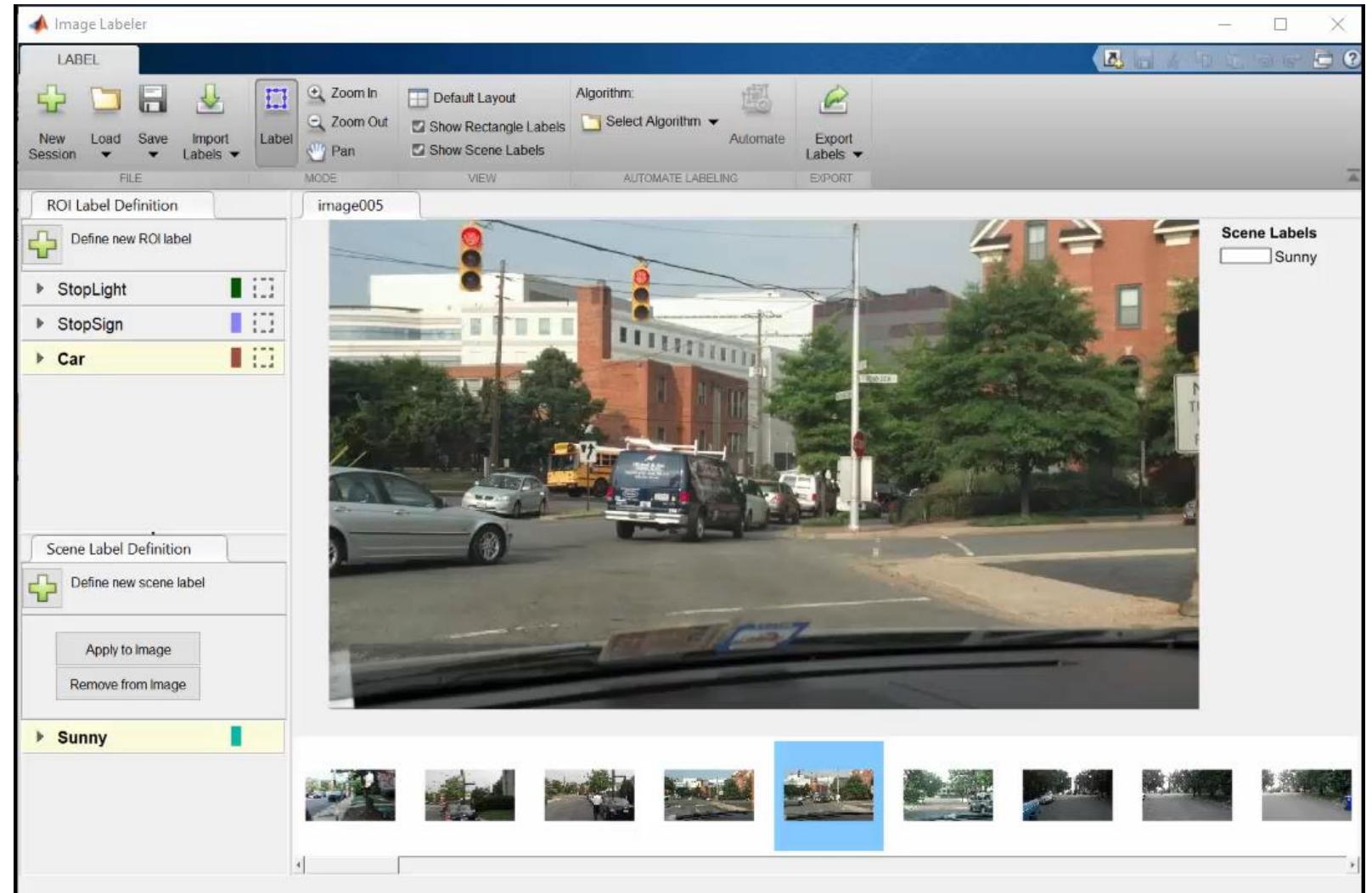


Image Labeler App

- Image Labeler App
 - マウス操作でROI指
 - イメージフレーム毎にラベリング
- ト rackingアルゴリズム
 - イメージフレーム全体に自動ラベリング

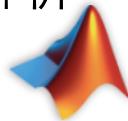


テキストデータの前処理もコマンド一行で!

アクセス



HTMLをパース
HTMLの要素検出
HTMLの構文解析
トークン化
語幹, レンマ化
特定単語削除



```
htmlTree  
findElement  
extractHTMLText  
tokenizedDocument  
normalizeWords  
removeWords
```

探索 & 発見



エクジ
アリ 特徴 テキスト 分類
アルゴリズム 処理 センサー 作成 テスト
Toolbox 複雑 から でき および 手法
コード 展開 れする が が 化 故障 で よう 運転 ため 場合
エンターライズ 組み込み も の TM れる 用途 生成
世 こと ますと に の て 使用 実行
また 画像 し を、 た 可能 (ウド) シミュレーション
機械 重要 し たり 的 な システム 時間
前 學習 は や AI まし 必要 Simulink
MATLAB 合成 により データ い は 予測 など 分野
データ and 向け 自動 モデル です エンジニア
クラ 設計 これ う Learning
最適 デバイス
ドライブインシステム

テキストデータ用の前処理関数の一例

- 不要なコンテンツの削除を可能にするフィルター処理関数

➤ トーカン化

`tokenizedDocument`

吾輩は猫である。

吾輩 は 猫 で ある 。

➤ ストップワード削除

`removeStopWords`

吾輩 は 猫 で ある 。

吾輩 猫 。

➤ 句読点の削除

`erasePunctuation`

吾輩 猫 。

吾輩 猫

➤ レンマ化

`normalizeWords`

輝き を 増し て いる

輝く を 増す て いる

➤ 小文字化

`lower`

Hello World

hello world

その他、

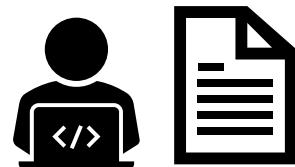
- 数値変換
- 特定の語の削除
- 長い/短い語の削除
- URL/HTMLタグの削除
- スペル修正（英独韓）etc…

ポイントのまとめ

-  GUIでデータインポート操作、等価な関数も1クリックで作成
-  テーブルでデータを集約、`table`.変数名でデータアクセス
-  時系列解析に便利な`timetable`
-  データの前処理は関数1行, ライブタスク or アプリで実施
-  系列データは`stackedplot`で簡単可視化

①アドホックな解析

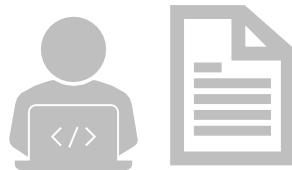
- データ読み込み
- データアクセス
- データ結合
- 前処理
- 可視化



アドホック解析における
データの前処理をマスター

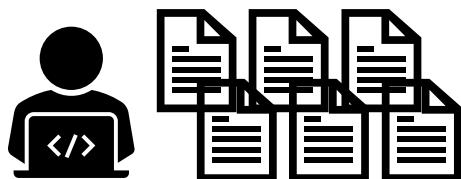
①アドホックな解析

- データ読み込み
- データアクセス
- 前処理
- 可視化



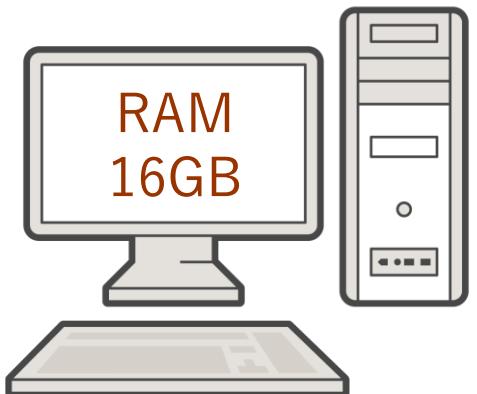
②ビッグデータ解析

- ビッグデータの読み込み
- ビッグデータの解析・深堀り

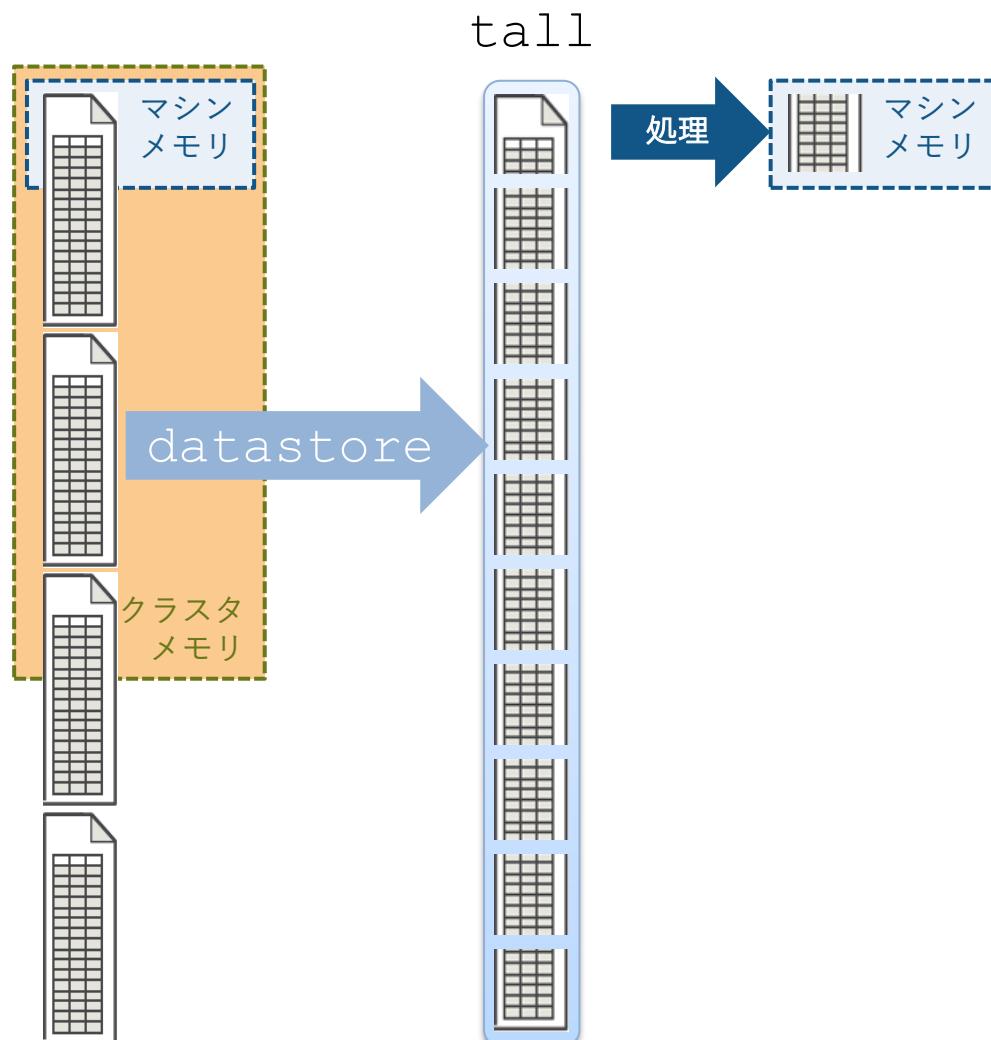


フライトデータ概要

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



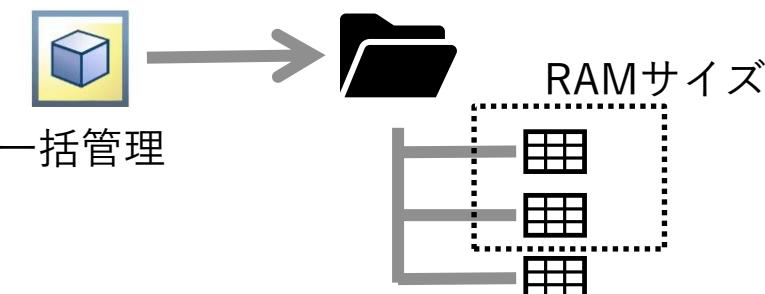
メモリに収まらないデータ取り扱いのためのテクニック



datastore型変数

メモリに収まらないデータの集まりのリポジトリ

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



tall配列

datastoreに格納されたデータを操作する

```
t = tall(ds)
```

datastoreで大きなデータ集合を扱う

必要な項目(センサー等)の選択

```
varnames = {'Time', 'EngineSpeedRPM'}
```

選択項目についてのdatastore型変数の作成

```
ds = datastore('EngineData*.csv',...
    'SelectedVariableNames', varnames)
```

“メモリに乗る”サイズのデータの読み込み

```
data = read(ds)
```



マシン
メモリ

利用可能な datastore 一覧	
一般	datastore
	spreadsheetDatastore
	tabularTextDatastore
	fileDatastore
データベース	databaseDatastore
画像	imageDatastore
	denoisingImageDatastore
	randomPatchExtractionDatastore
	pixelLabelDatastore
	augmentedImageDatastore
音声	audioDatastore
予知保全	fileEnsembleDatastore
	simulationEnsembleDatastore
Simulink	SimulationDatastore
自動車	mdfDatastore
カスタム	subclass matlab.io.Datastore
変換	transform 存在するdatastoreの変換

“全体”の計算はtallで実行

1個のファイル

データにアクセス

```
data = readtable("EngineData1.csv");
data = table2timetable(data);
```

データの前処理

興味のあるデータの選択

```
data = data(:, "EngineSpeedRPM");
```

欠損データ処理

```
data = fillmissing(data, "linear");
```

統計量計算

```
m = mean(data.EngineSpeedRPM);
s = std(data.EngineSpeedRPM);
```

1,000個のファイル

データにアクセス

```
data = datastore("EngineData*.csv");
data = tall(data);
data = table2timetable(data);
```

データの前処理

興味のあるデータの選択

```
data = data(:, "EngineSpeedRPM");
```

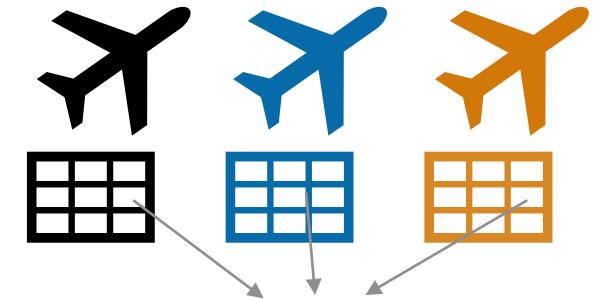
欠損データ処理

```
data = fillmissing(data, "linear");
```

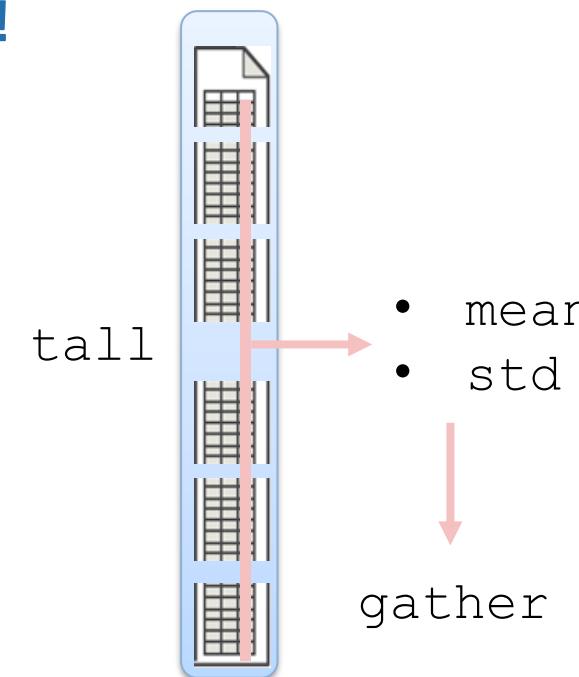
統計量計算

```
m = mean(data.EngineSpeedRPM); • 平均
s = std(data.EngineSpeedRPM); • 分散
```

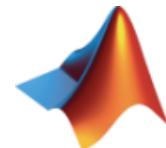
```
[m,s] = gather(m,s);
```



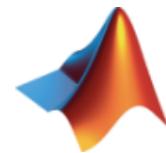
2行付け加えるだけ!



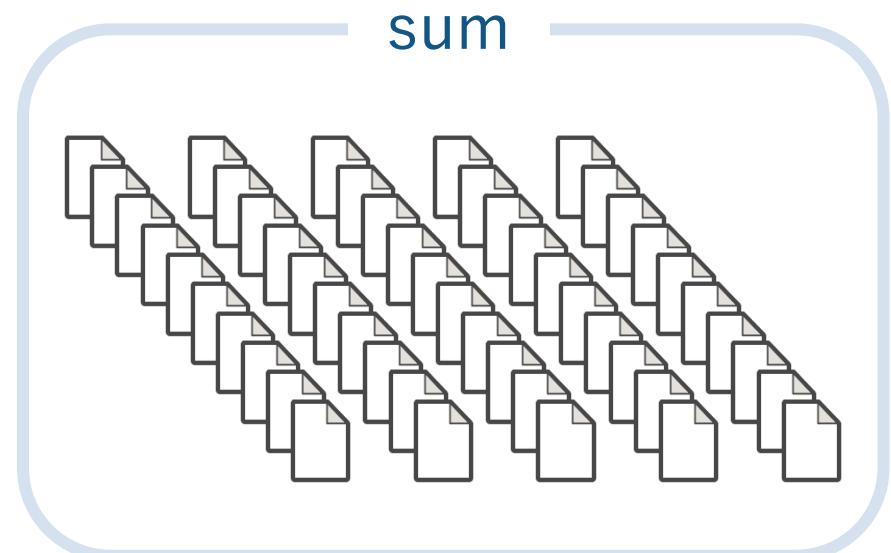
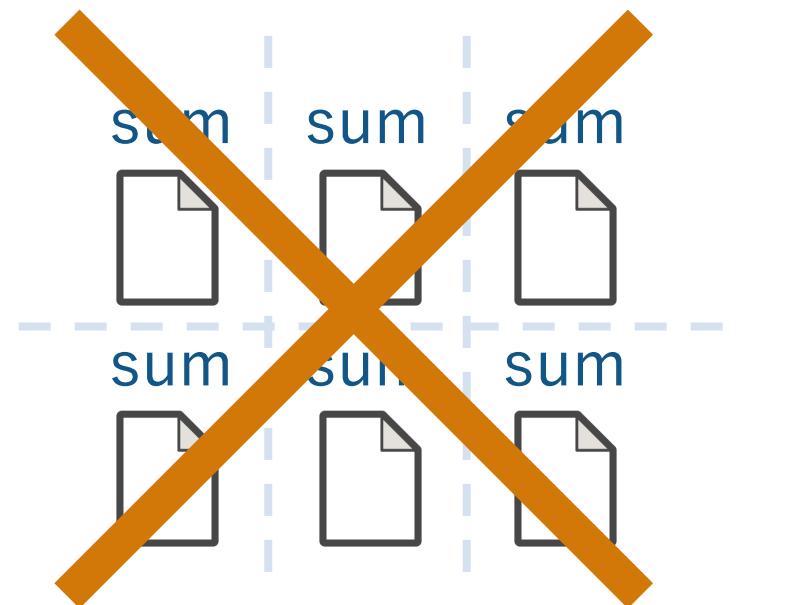
ポイントのまとめ



datastoreによりRAMサイズ以上の大容量データを一括管理

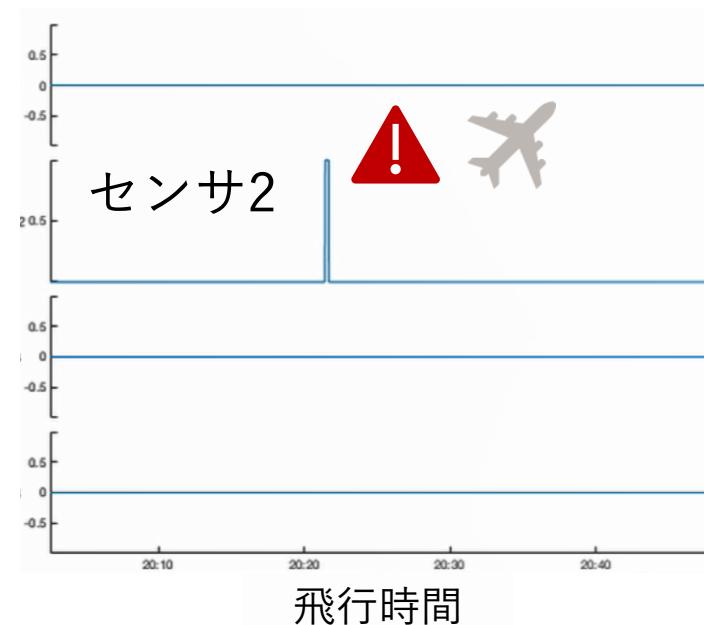


tallによってビッグデータも従来通りの操作が可能



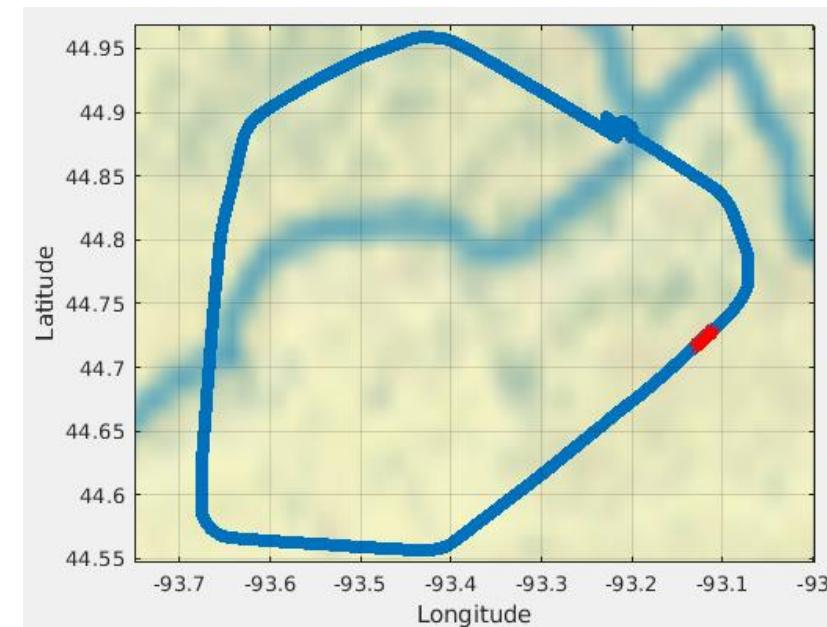
ビッグデータの活用例 – 事象の深堀り

4つの火災センサーを探索



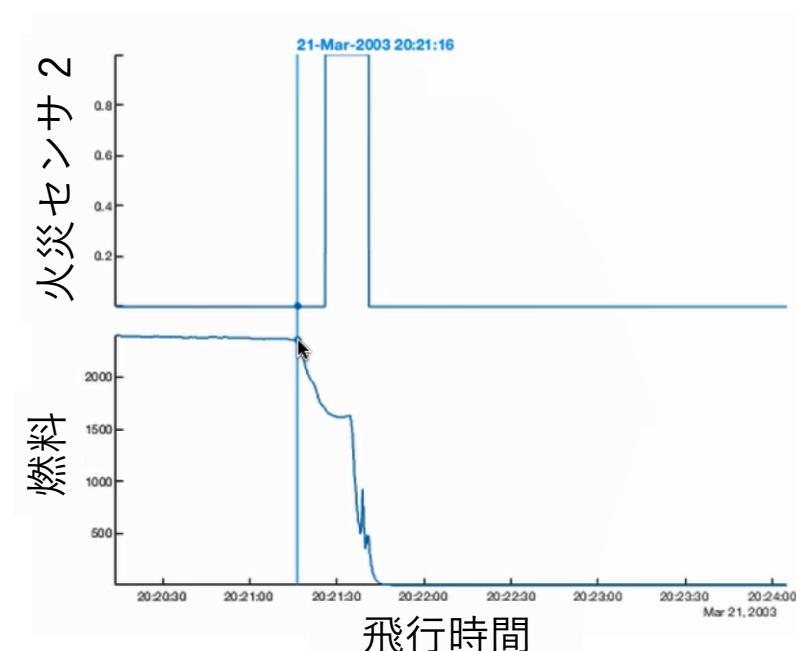
stackedplot

フライト軌跡の確認



geoplot

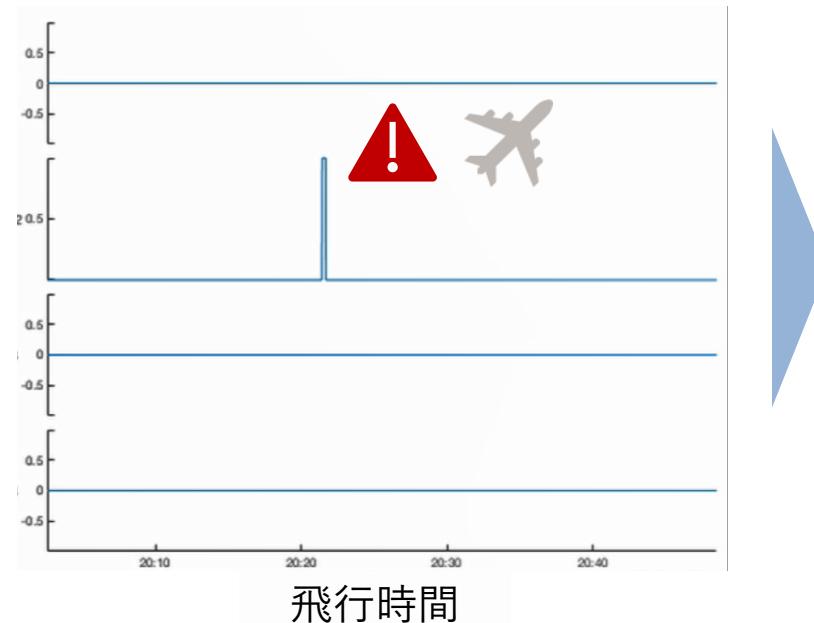
燃料データの確認



stackedplot

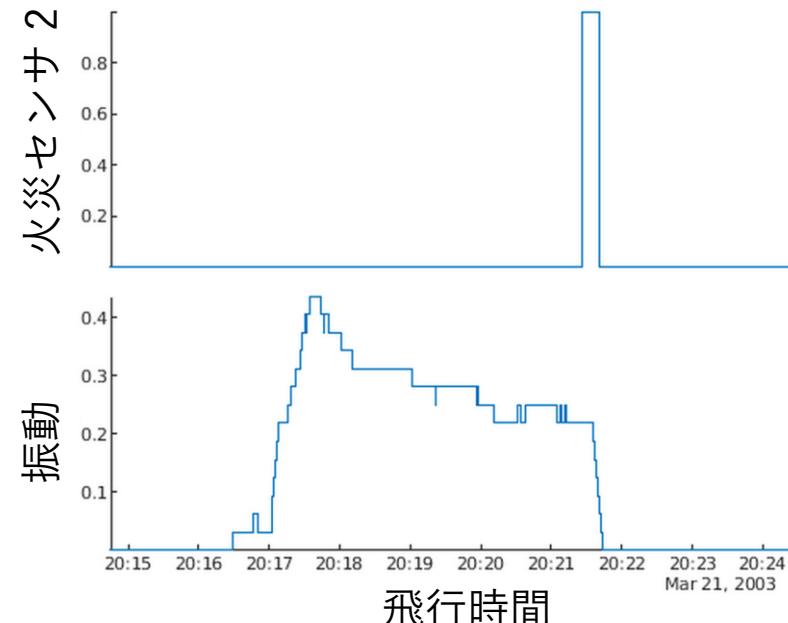
ビッグデータの活用例 – 事象の深堀り

4つの火災センサーを探索



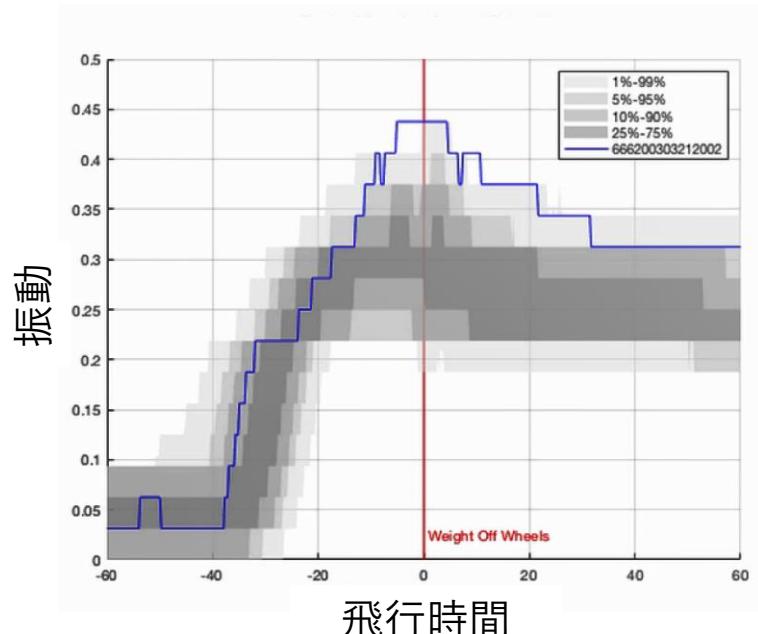
stackedplot

振動データの確認



stackedplot

全フライトとの比較



火災が発生したフライトは
振動レベルが高かった

ビッグデータも通常通り解析を継続可能

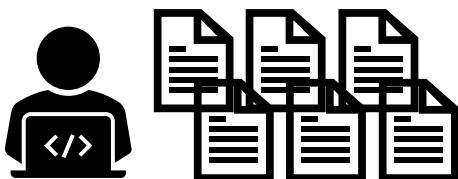
①アドホックな解析

- データ読み込み
- データアクセス
- データ結合
- 前処理
- 可視化



②ビッグデータ解析

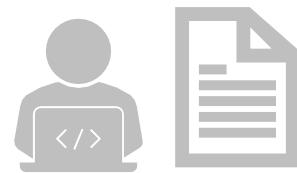
- ビッグデータの読み込み
- ビッグデータの解析・深堀り



ビッグデータの効率の良い
解析方法をマスター

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



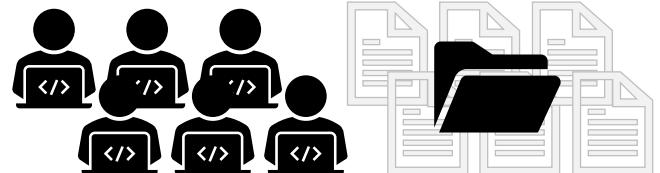
②ビッグデータ解析

- 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

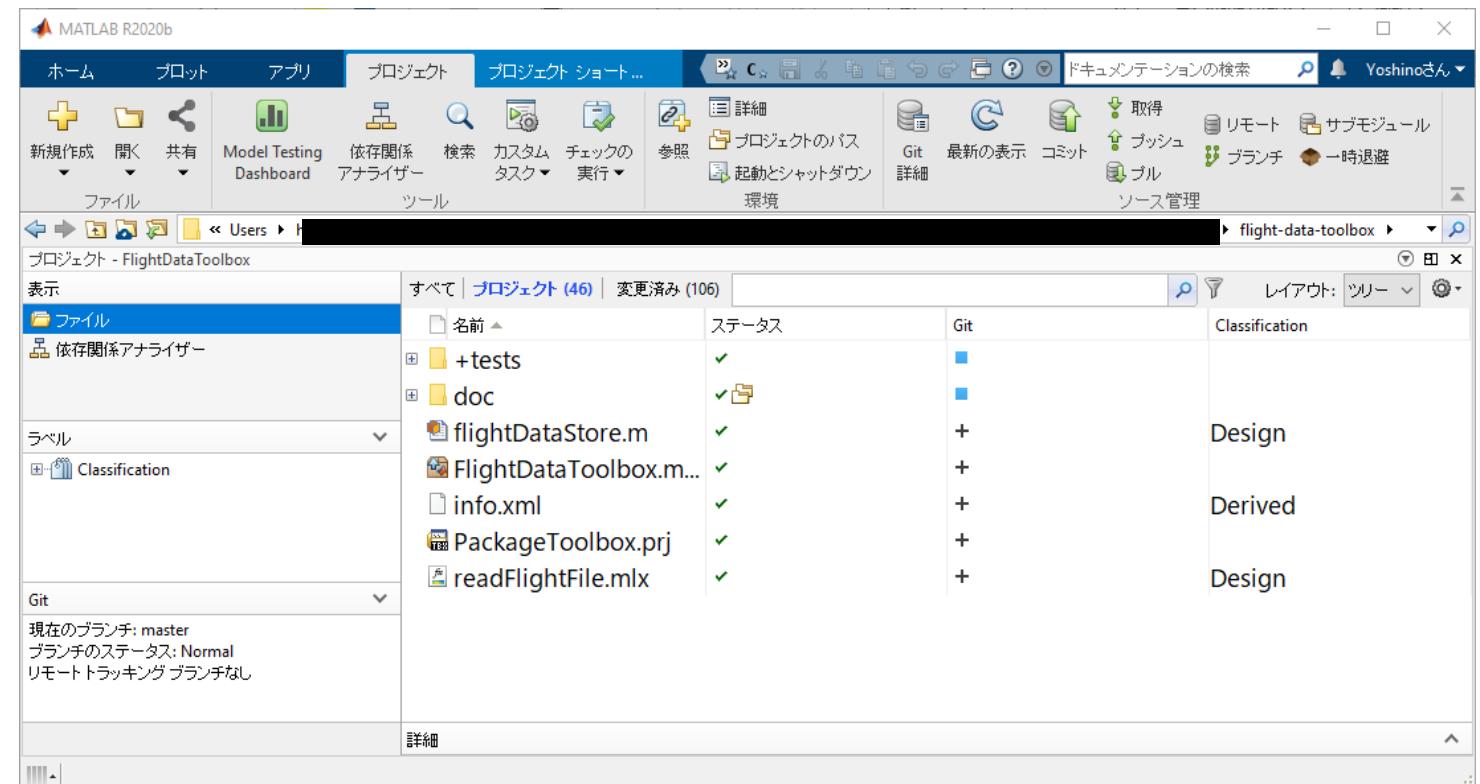
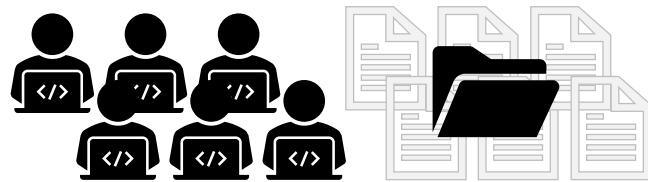
- プロジェクトの作成・共有
- 大人数で解析



プロジェクト化

共同作業・バージョン管理

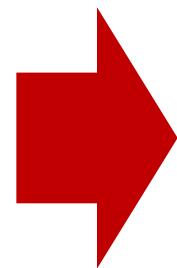
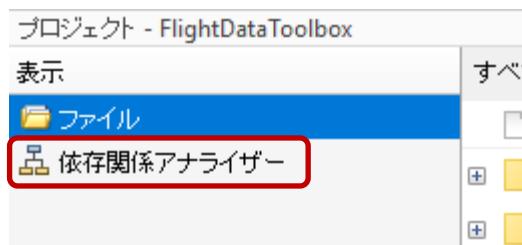
- ファイル、データ、テスト等を1か所で管理
- 作業の整理、チームでの共同作業を円滑にする
- Git™, SVN®を使用したソース管理ツール (GitはMATLABに同梱)
- 依存関係の解析



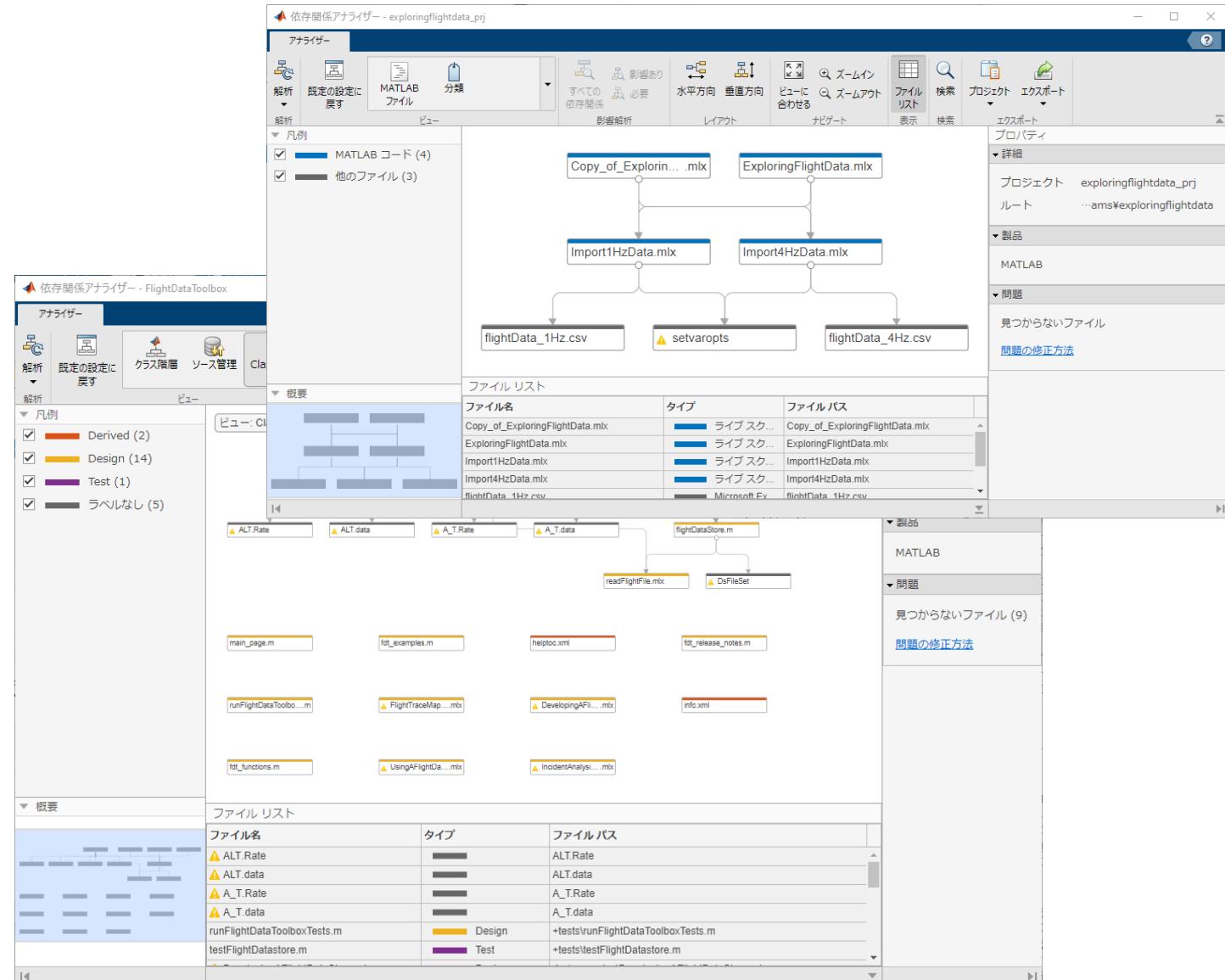
MATLABプロジェクトについて

プロジェクトによる業務一元化

コードの依存関係解析して可視化



- ファイル、データ間の依存関係を可視化
- クラスの階層なども表示可能
(オブジェクト指向プログラミング)



プロジェクトによる業務一元化

ソースコード管理

Gitを使ったヴァージョン管理の例

- Git 対応 (GitHubなどにも使用できる!)
- SVN 対応
- ボタンクリックで操作

名前	ステータス	分類	Git
GoogleNet	✓		.
openFiles	✓		■
appDesignerDe...	✓	設計	●
appDesignerSa...	✓	設計	●
classificationDe...	✓	設計	●
Data.csv	✓	アーティファクト	●
dataImportTool...	✓	設計	●
GoogleNet.prj	✓		●
GoogleNet_Ca...	✓	設計	●
importfileSampl...	✓	設計	●
LivetaskDemo.m...	✓	設計	●
MATLAB_IntroD...	✓	設計	■



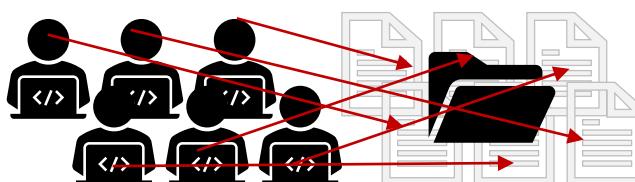
名前	ステータス	分類	Git
GoogleNet	✓		.
openFiles	✓		■
appDesignerDe...	✓	設計	●
appDesignerSa...	✓	設計	●
classificationDe...	✓	設計	●
Data.csv	✓	アーティファクト	●
dataImportTool...	✓	設計	●
GoogleNet.prj	✓		●
GoogleNet_Ca...	✓	設計	●
importfileSampl...	✓	設計	●
LivetaskDemo.m...	✓	設計	●
MATLAB_IntroD...	✓	設計	●



プロジェクトによる業務一元化

ソースコード管理 (Cont.)

- Revisionとの相違
 - 挿入
 - 削除
 - 変更
- 複数のRevision間を比較可能
- 複数人での作業の効率が向上



Comparison tool interface showing differences between two MATLAB files:

```

MATLAB_IntroDemo_70d160230f5823d308bc867954425e16c58a0f78.mlx and MATLAB_IntroDemo_b240544451591b5aaa983f56b05e1ce92e7f736b.mlx

```

Left pane (Original):

```

MATLABの雰囲気ご紹介
MATLAB as MATrix LABoratory
可視化
計算数学 - Symbolic Math Toolbox
関数をプロット
の解
微分
積分

```

Right pane (Revised):

```

MATLABの雰囲気ご紹介: MATrix LABratory
MATLAB as MATrix LABoratory
可視化

```

Bottom pane (Differences):

```

sample_GoogleNet_CameraRecognition_8fd21e4525be1d9a0823bff46db178a3862469d2.m and sample_GoogleNet_CameraRecognition_b240544451591b5aaa983f56b05e1ce92e7f736b.m

```

Detailed description of the bottom pane:

問題：以下の媒介変数表示は何を示すでしょうか？

```

clear; close all;
n = 100; t=linspace(-pi,pi,n);
X=zeros(n,n); Y=zeros(n,n); Z=zeros(n,n); zns

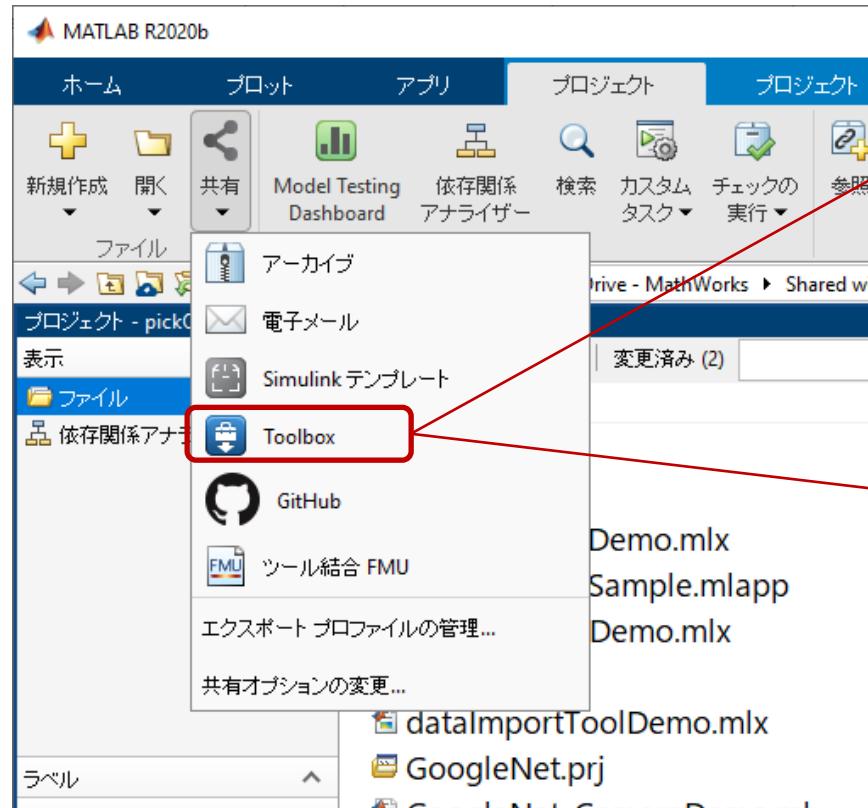
```

一致した行数: 6

左のファイル内の一致しない行数: 1
右のファイル内の一致しない行数: 1

検証済みツールとして共有

プロジェクト終了 →
ツールボックス化して共有



ツールボックス

The screenshot shows the MATLAB Toolstrip with the 'Packaging' tab selected. A window titled 'ツールボックスのパッケージ化 - PackageToolbox.prj' is open, showing the 'flight-data-toolbox' project. The right panel displays the 'ツールボックスの情報' (Toolbox Information) for 'FlightDataToolbox', which includes details like author ('Seth DeLand'), email ('sdeland@mathworks.com'), and company ('MathWorks'). Below this, a preview image of a flight map is shown. The bottom panel contains a code snippet for creating a FlightDataStore and a table of signal meta-information.

ツールボックスの情報

FlightDataToolbox
Seth DeLand
sdeland@mathworks.com
MathWorks

Flight Data Toolbox provides functions for working with the large repository of flight recorder data. It includes a datastore for accessing flight recorder data, and examples of analysis that can be performed.

Using a FlightDataStore

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

ds =
    flightDataStore with properties:
```

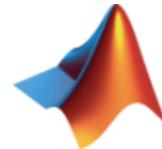
CurrentPageIndex:	1
NumberOfFiles:	6
VariableNames:	{1x186 cell}
SelectedVariableNames:	{1x186 cell}
AddFlightIdentifier:	0

Data for a Subset of Signals

```
signalMetaInfo(ds)
```

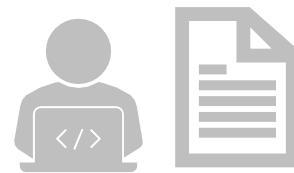
	Signals	Rate	Units	Description	Alpha
1	'ABRK'	1.0000	'DEG'	'AIRBRAKE POS...'	'ABRK'
2	'ACID'	0.2500	"	'AIRCRAFT NUM...'	'ACID'
3	'ACMT'	1.0000	"	'ACMS TIMING ...'	'ACMT'
4	'AIL_1'	1.0000	'DEG'	'AILERON POSI...'	'AIL_1'
5	'AIL_2'	1.0000	'DEG'	'AILERON POSI...'	'AIL_2'
6	'ALT'	4.0000	'FEET'	'PRESSURE ALT...'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

ポイントのまとめ

-  MATLABプロジェクトによる簡単配布・作業標準化
-  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の作業を円滑化
-  ツールボックス化して配布、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



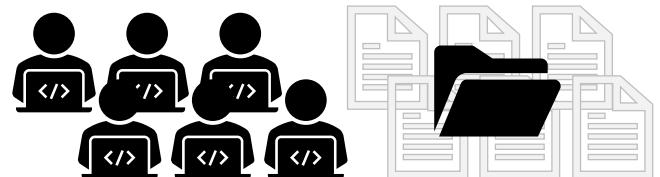
②ビッグデータ解析

- ビッグデータの読み込み
- ビッグデータの解析

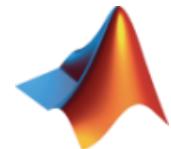


③プロジェクト化

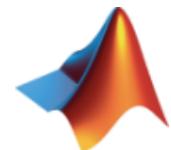
- プロジェクトの作成・共有
- 大人数で解析



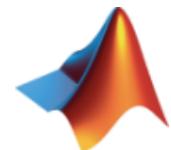
まとめ



データ解析の鍵を握る前処理をMATLABで効率よく実施可能



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

本日ご紹介したツール – MATLAB R2021a



更に学びたい方へ



MATLAB 入門

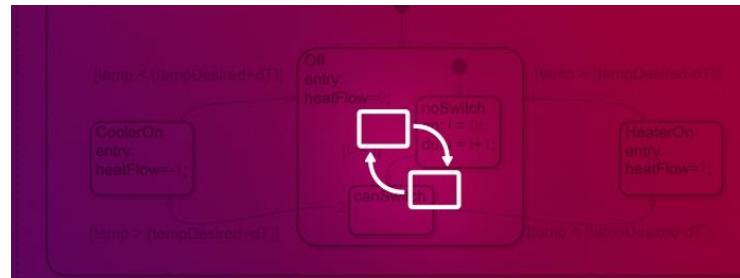


機械学習入門



ディープラーニング入門

- 無料の入門コース
- 受講にライセンス不要
- Webブラウザで操作



Stateflow 入門



Simulink 入門

<https://jp.mathworks.com/services/training.html>

機械学習における前処理の効率化

ebookにアクセス

MATLAB®の高水準のツール・可視化・ドメイン固有ツールとアプリ・ライブエディタタスクを使用して、機械学習アルゴリズムに使用される表形式データおよび時系列データの前処理を高速化する方法を学びます。これらのツールを前処理スクリプトと一緒に使用することで、機械学習モデルの精度を迅速に評価することができます。

ebookには次の内容が含まれています。

- ✓ データの調査
- ✓ 一般的な前処理
- ✓ MATLABアプリによる前処理

ebookにアクセス

リンクはこちら





© 2021 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.