Convertirse en un equipo de ingenieros centrado en datos: ponerse al día con el aluvión de datos

Paula Poza
A path for how your team can better work with and utilize data.
Data Science Maturity Levels

- Ad-hoc Individual Analysis
- Generally Useful Tools for Analysis
- Common Infrastructure; Tested and Documented

Ease of scaling to more people
Overhead when asking a new question
Data Science Maturity Levels

- **Ad-hoc Individual Analysis**
- **Generally Useful Tools for Analysis**
- **Common Infrastructure; Tested and Documented**

• Goal is to be fast: reduce time to insight
Getting Started: Exploring a New Dataset
Getting Started: **Exploring a New Dataset**

![MATLAB interface](image)

**Exploring Flight Data**

**Import Data**

```
flightData1Hz = 4636x89 table
```

<table>
<thead>
<tr>
<th>Time</th>
<th>ABRK</th>
<th>ACMT</th>
<th>AIL_1</th>
</tr>
</thead>
<tbody>
<tr>
<td>10-May-2022</td>
<td>119.9836</td>
<td>59</td>
<td>91.8789</td>
</tr>
<tr>
<td>10-May-2022</td>
<td>119.9836</td>
<td>59</td>
<td>91.8994</td>
</tr>
<tr>
<td>10-May-2022</td>
<td>119.9836</td>
<td>60</td>
<td>91.8994</td>
</tr>
<tr>
<td>10-May-2022</td>
<td>119.9836</td>
<td>59</td>
<td>91.8789</td>
</tr>
<tr>
<td>10-May-2022</td>
<td>119.9836</td>
<td>60</td>
<td>91.9607</td>
</tr>
</tbody>
</table>
Getting Started: Exploring a New Dataset

Access variables by type:

```matlab
flightData1Hz(:,vartype("numeric"))
```

```matlab
ans = 4636×88
```

```
0.0001    0.0001    0.0001    0.0001    0.0001    0.0060 ...
0.0001    0.0001    0.0001    0.0001    0.0001    0.0000 ...
```

```matlab
10^6 * ...
```
Getting Started: **Exploring a New Dataset**

- **Missing Data**
  - `ismissing`
  - `rmmissing`
  - `fillmissing`

- **Outliers**
  - `isoutlier`
  - `rmoutliers`
  - `filloutliers`

- **Change Points**
  - `ischange`

- **Noisy Data**
  - `smoothdata`

  and more...

Getting Started: **Exploring a New Dataset**

**Load 4Hz Data**
Load data from same flight that was sampled at 4Hz.

```matlab
Import4HzData

\text{t4hz} = \text{table2timetable}(\text{flightData4Hz})
```

**Synchronize 1Hz and 4Hz Data**
Join the 1Hz and 4Hz data using the synchronize command. The synchronize command gives us flexibility in how the synchronize occurs. Here, we use the default synchronize method which synchronizes the
Getting Started: Exploring a New Dataset

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

Visualizing the Flight Path

```matlab
figure;
inFlight = t.WOW==1;
geoplot(t.LATP(inFlight),t.LONP(inFlight), 'LineWidth',3);
```
Data Science Maturity Levels

- Explore and understand data
- Document analysis
- Tools will be re-used in next steps
Data Science Maturity Levels

Ad-hoc
Individual Analysis

Generally Useful
Tools for Analysis

Common
Infrastructure; Tested and Documented

• Apply to different datasets
  • Functions/Scripts
  • MATLAB Apps

• Trend: Work with **BIG DATA**
Overview of Flight Data

- **35** unique aircraft
- **180,000** unique flights
- **300 GB** of data

**Source:**
- NASA Dash Link: Sample Flight Data
- [https://c3.nasa.gov/dashlink/projects/85/](https://c3.nasa.gov/dashlink/projects/85/)
Big Data Creates Opportunities

- Find rare events, then dive deeper
- Build and validate test scenarios that match real-world conditions
- Perform fleet-wide calculations
Big Data Requires New Tools

Create a datastore from all CSV files

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

Read a single file of data

```matlab
data = read(ds);
```

Reset the datastore back to the first file

```matlab
reset(ds);
```

Find the maximum value of “Y” in each file

```matlab
X = [];  
while hasdata(ds)  
    data = read(ds);  
    X(end+1) = max(data.Y);  
end
```

<table>
<thead>
<tr>
<th>Built-In Datastores</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General</strong></td>
</tr>
<tr>
<td>datastore</td>
</tr>
<tr>
<td>spreadsheetDatastore</td>
</tr>
<tr>
<td>tabularTextDatastore</td>
</tr>
<tr>
<td>fileDatastore</td>
</tr>
<tr>
<td><strong>Database</strong></td>
</tr>
<tr>
<td>databaseDatastore</td>
</tr>
<tr>
<td><strong>Image</strong></td>
</tr>
<tr>
<td>imageDatastore</td>
</tr>
<tr>
<td>denoisingImageDatastore</td>
</tr>
<tr>
<td>randomPatchExtractionDatastore</td>
</tr>
<tr>
<td>pixelLabelDatastore</td>
</tr>
<tr>
<td>augmentedImageDatastore</td>
</tr>
<tr>
<td><strong>Audio</strong></td>
</tr>
<tr>
<td>audioDatastore</td>
</tr>
<tr>
<td><strong>Predictive Maintenance</strong></td>
</tr>
<tr>
<td>fileEnsembleDatastore</td>
</tr>
<tr>
<td>simulationEnsembleDatastore</td>
</tr>
<tr>
<td><strong>Simulink</strong></td>
</tr>
<tr>
<td>SimulationDatastore</td>
</tr>
<tr>
<td><strong>Automotive</strong></td>
</tr>
<tr>
<td>mdfDatastore</td>
</tr>
</tbody>
</table>
Big Data Requires New Tools

Custom Datastore

- Customize a datastore to work with your dataset

- Gives you control over how data is loaded and formatted

- MATLAB subclass: “fill-in-the-blanks”

- Build a piece of infrastructure, then re-use it in your analyses

```matlab
function [data,info] = read(ds)
... end

function tf = hasdata(ds)
... end

function reset(ds)
... end

function p = progress(ds)
... end

function data = readall(ds)
... end
```
A Custom Datastore for Flight Data

```matlab
% Basic datastore implementation
function [data,info] = read(fds)

function tf = hasdata(fds)

function reset(fds)

function frac = progress(fds)

function data = readall(fds,iterDisp)

% Custom methods for reading individual flight data
function data = readFullFlight(fds,flightIdentifier)
```
Find Rare Events, then Dive Deeper

Incident Analysis
The data for tail 666 captures an engine fire that happened shortly after takeoff. This example analyzes data from that tail number to see if there is any interesting information that can be learned about how when the fire occurred.

Search for Flights where the Engine 2 Fire Sensor Went True
Loop through all flights for this aircraft, and display some information if there is a fire.

```matlab
ds = flightDataStore('/Volumes/Ext/Data/FlightData/666/');
ds.SelectedVariableNames = {'FIRE_2'};
ds.AddFlightIdentifier = true;

results = table('Size',[0 2], 'VariableTypes', {'categorical', 'double'}, ...
    'VariableNames', {'Flight', 'Fire_Seconds'});
while hasdata(ds)
    t = read(ds);
    if any(t.FIRE_2)
        results = [results; {t.Flight(1), sum(t.FIRE_2)}];
    end
end
results
```
Perform Fleet-Wide Calculations

Fleet-Wide Calculations
Calculate fuel efficiency across all of the flights in the dataset.

Tall Arrays
Creating a tall array from the datastore gives us access to many built-in functions.

1

\[
ds = \text{flightDataStore}('Volumes/Ext/Data/FlightData/');\]

TAS: True Airspeed (knots)
FF_N: Fuel Flow N (lbs/hr)
WOW: Weight On Wheels (logical)
Data Science Maturity Levels

- Make it easy to navigate the data
- Re-use each time you analyze the dataset
Data Science Maturity Levels

- **Ad-hoc Individual Analysis**
- **Generally Useful Tools for Analysis**
- **Common Infrastructure; Tested and Documented**

- **Collaborate**: Work with others on a common code base
- **Verify**: Write well-tested software
- **Share**: Build tools for others
MATLAB Projects
Testing
Creating a Toolbox
Data Science Maturity Levels

- Ad-hoc Individual Analysis
- Generally Useful Tools for Analysis
- Common Infrastructure; Tested and Documented

- Scale-out to larger group of users
- Easier to maintain and share
What’s Next?

Advanced Analytics and Machine Learning

Build and Test Algorithms for Embedded Systems

Deploy Apps and Analytics to Enterprise IT Systems