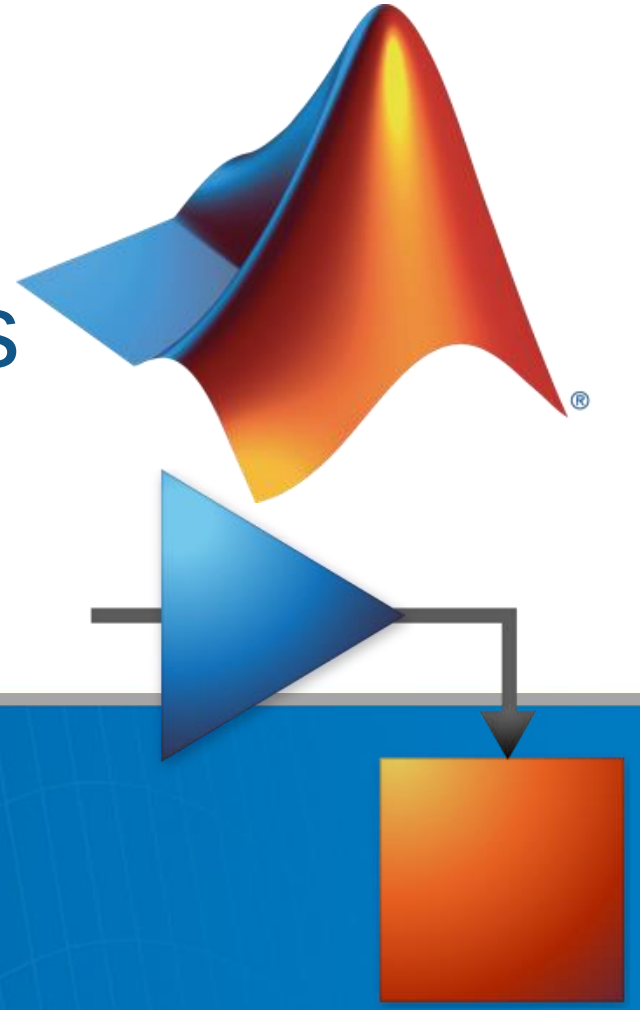
MATHACK 2022 Training Sessions

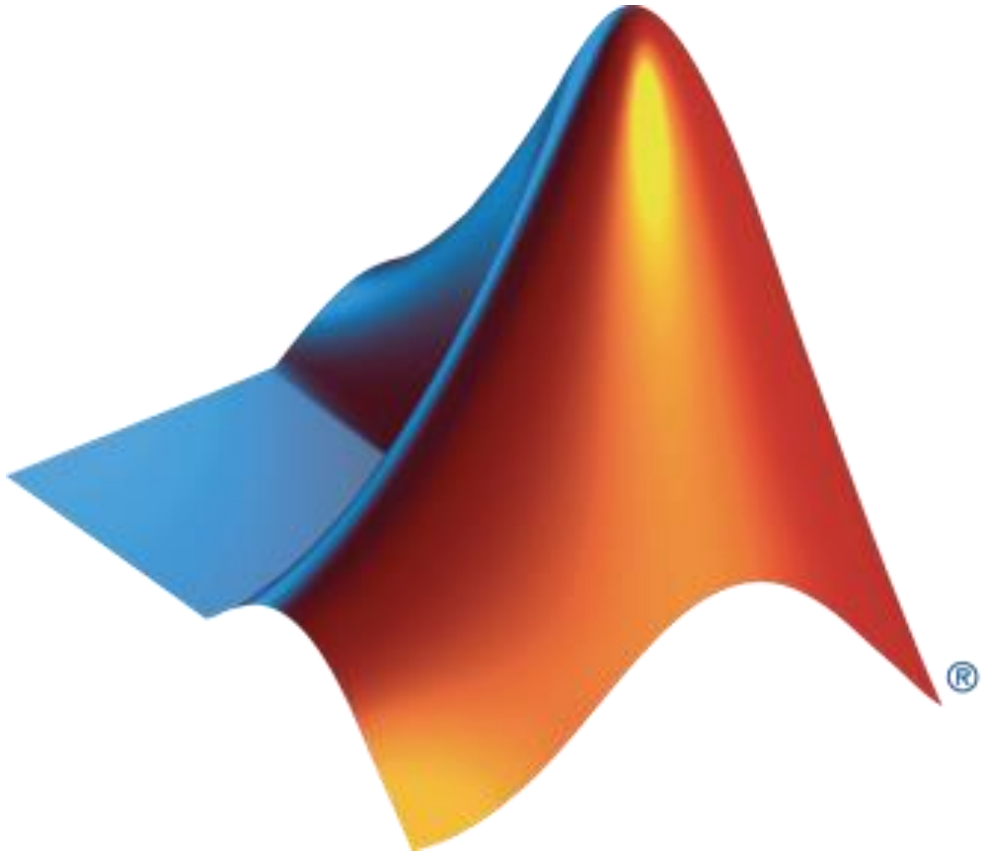


Contact

- Arturo Fernández
  - 4th year Bachelor Telecom Eng.
  - Polytechnic University of Valencia (UPV)
- Giulio Mattera
  - Phd in Intelligent Robotics
  - University of Naples Federico II
- Alessio Tumminello
  - 2th year Master's Degree Robotics Eng.
  - University of Pisa
- Javier Crespo
  - 2nd year Aerospace Engineering
  - Carlos III University

Today:

- 19.00 – 19.25 : Set-up
  - Welcome presentation
  - Introduction to the MATHACK Challenge
  - MATHACK Challenge resources
- 19.25 – 19.45 : MATLAB Mobile Data Acquisition and Processing Demo
  - Log data locally and load on MATLAB Drive
  - Stream data from MATLAB Mobile to MATLAB
  - Fitness Tracker Demo with steps counter
- 19.45 – 20.05 : Machine Learning Mobile Data Demo
  - Using data collected from your phone for Run or walking application
  - Development of neural network in MATLAB
- 20.05 – 20.30 : Greetings
  - Git Fundamentals
  - Submissions and Judging Criteria

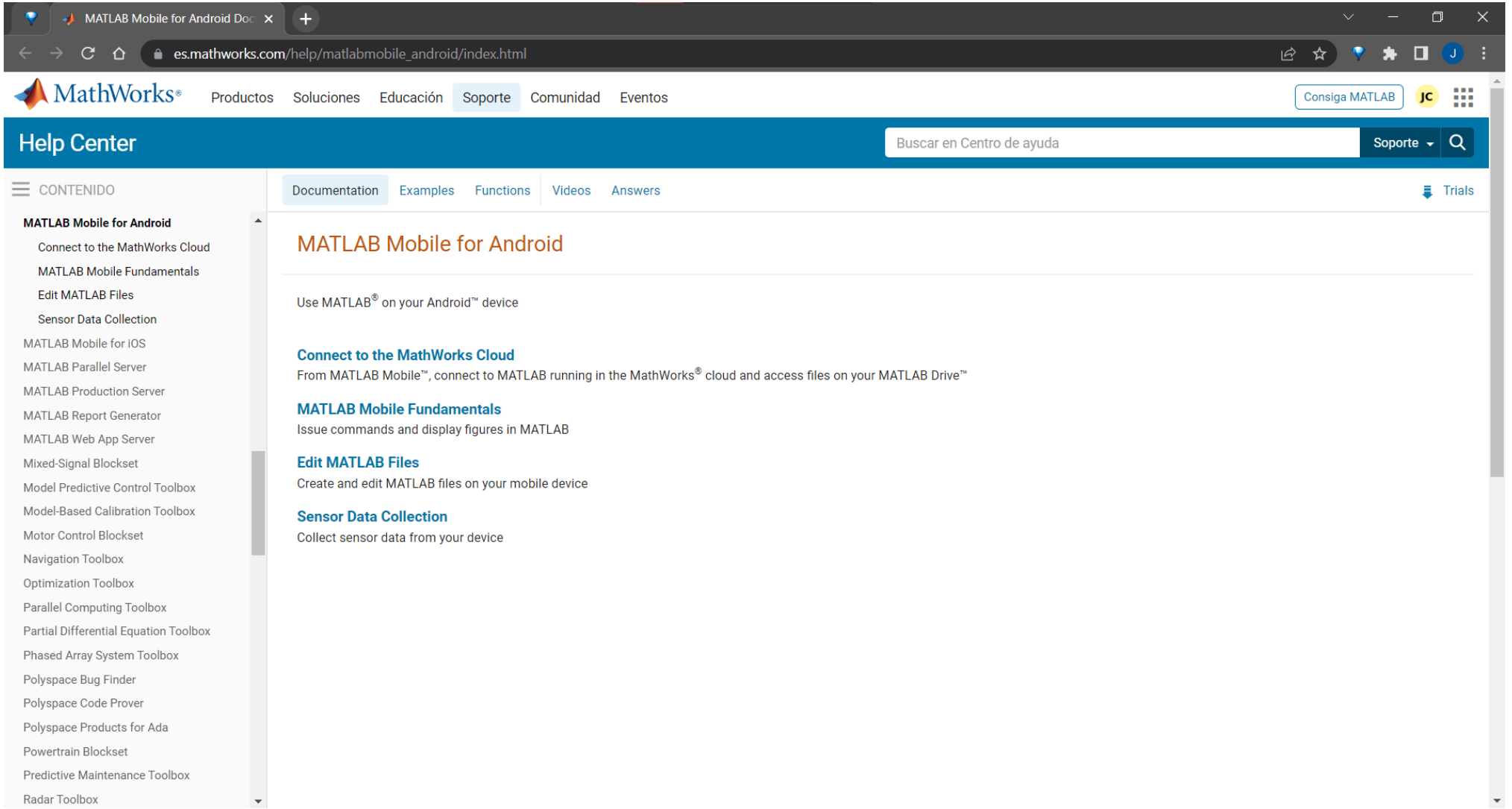


MATHACK 2022 Training Sessions
MATHACK Challenge Resources

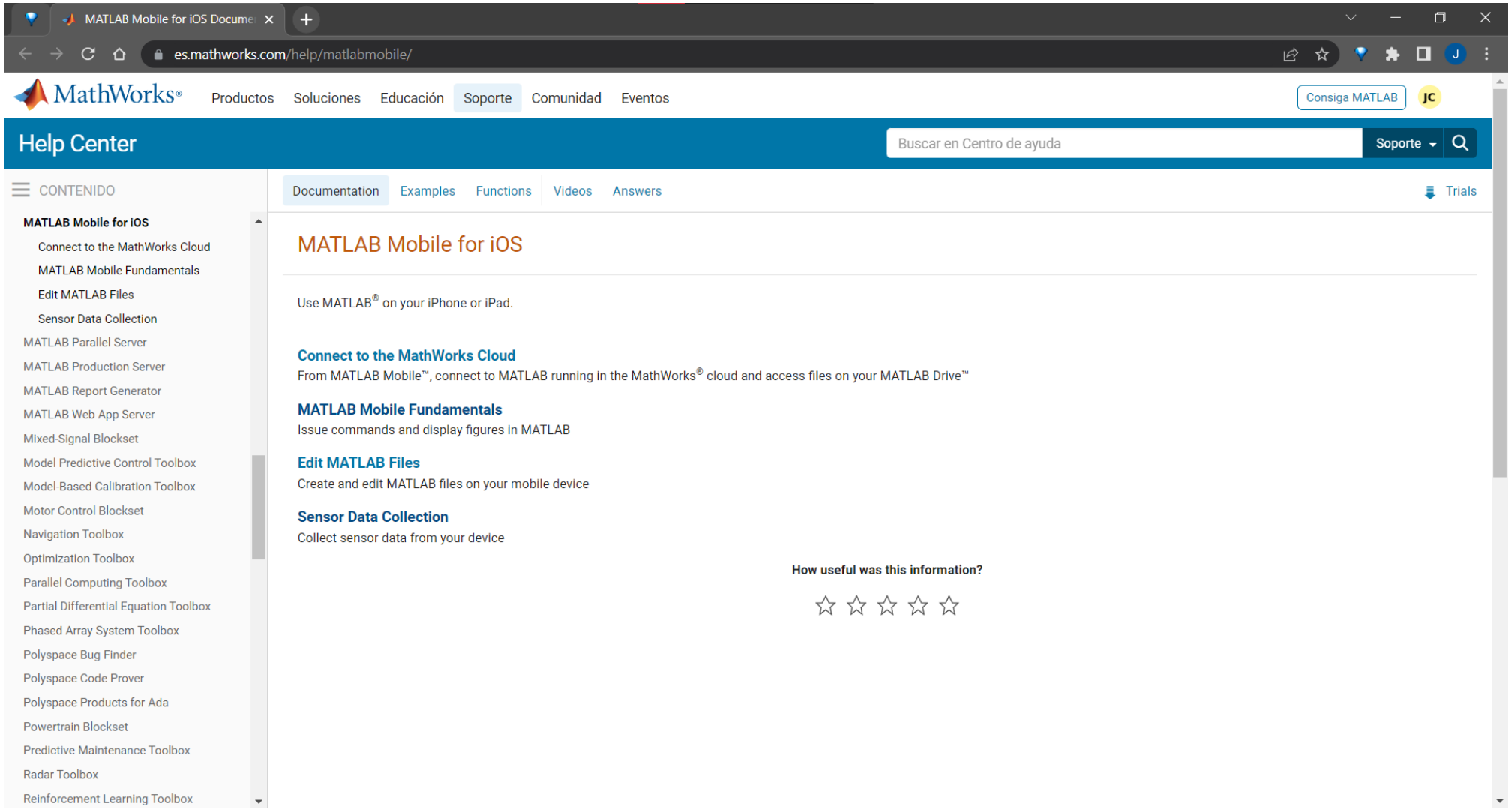
JAVIER CRESPO
UC3M Ambassador

Universidad
Carlos III de Madrid

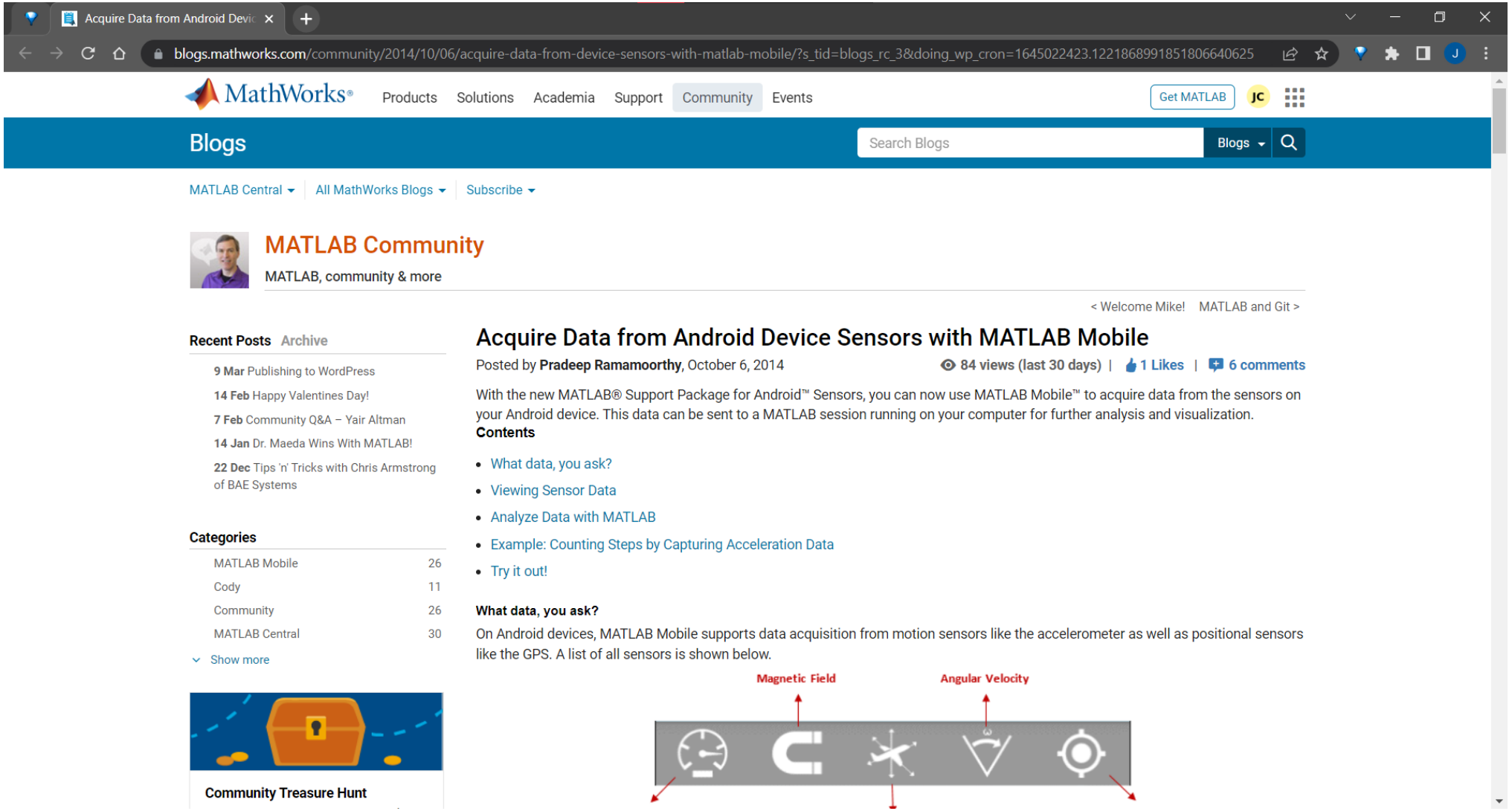
MATLAB Mobile Android documentation



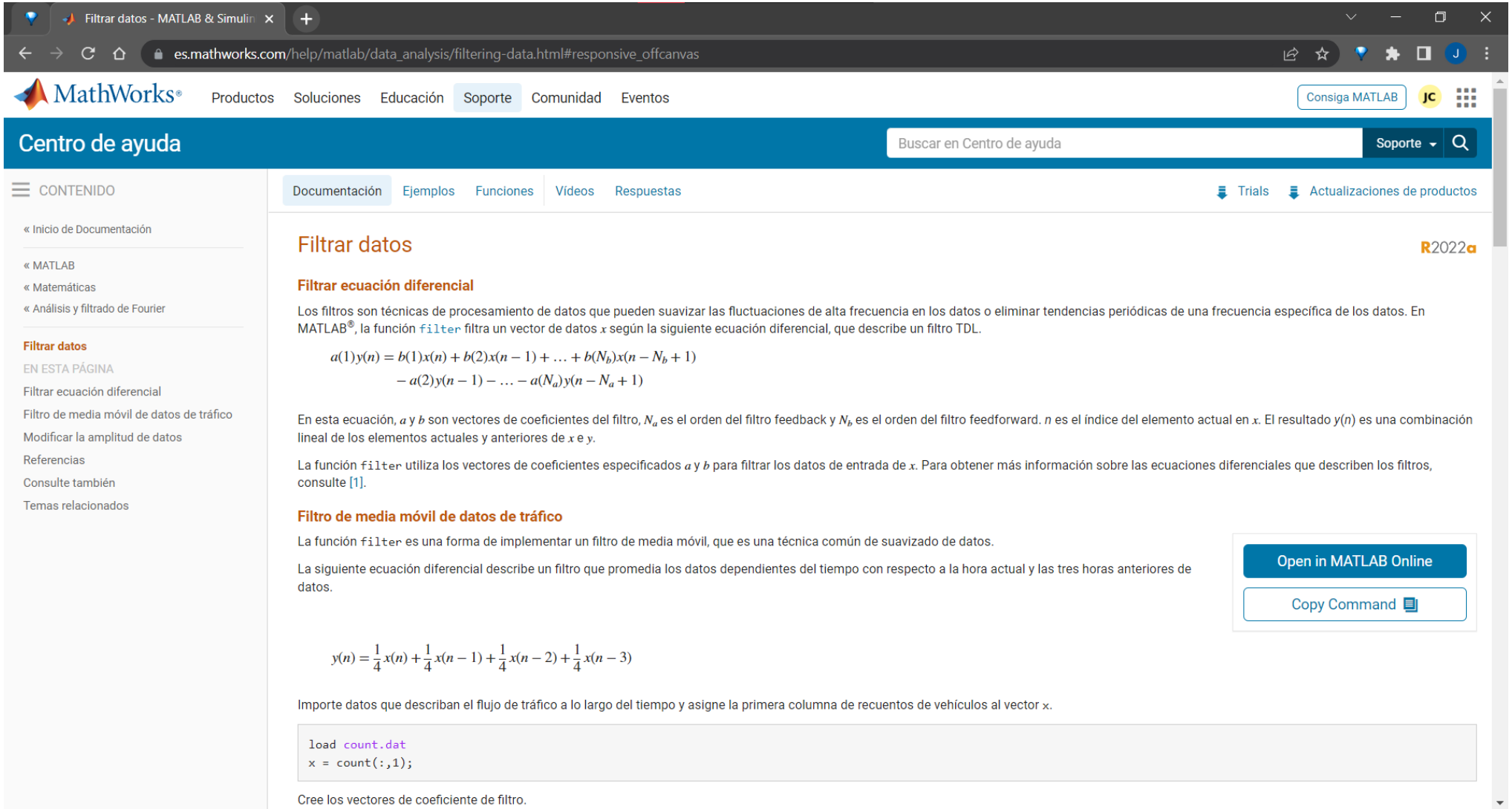
MATLAB Mobile IOS documentation



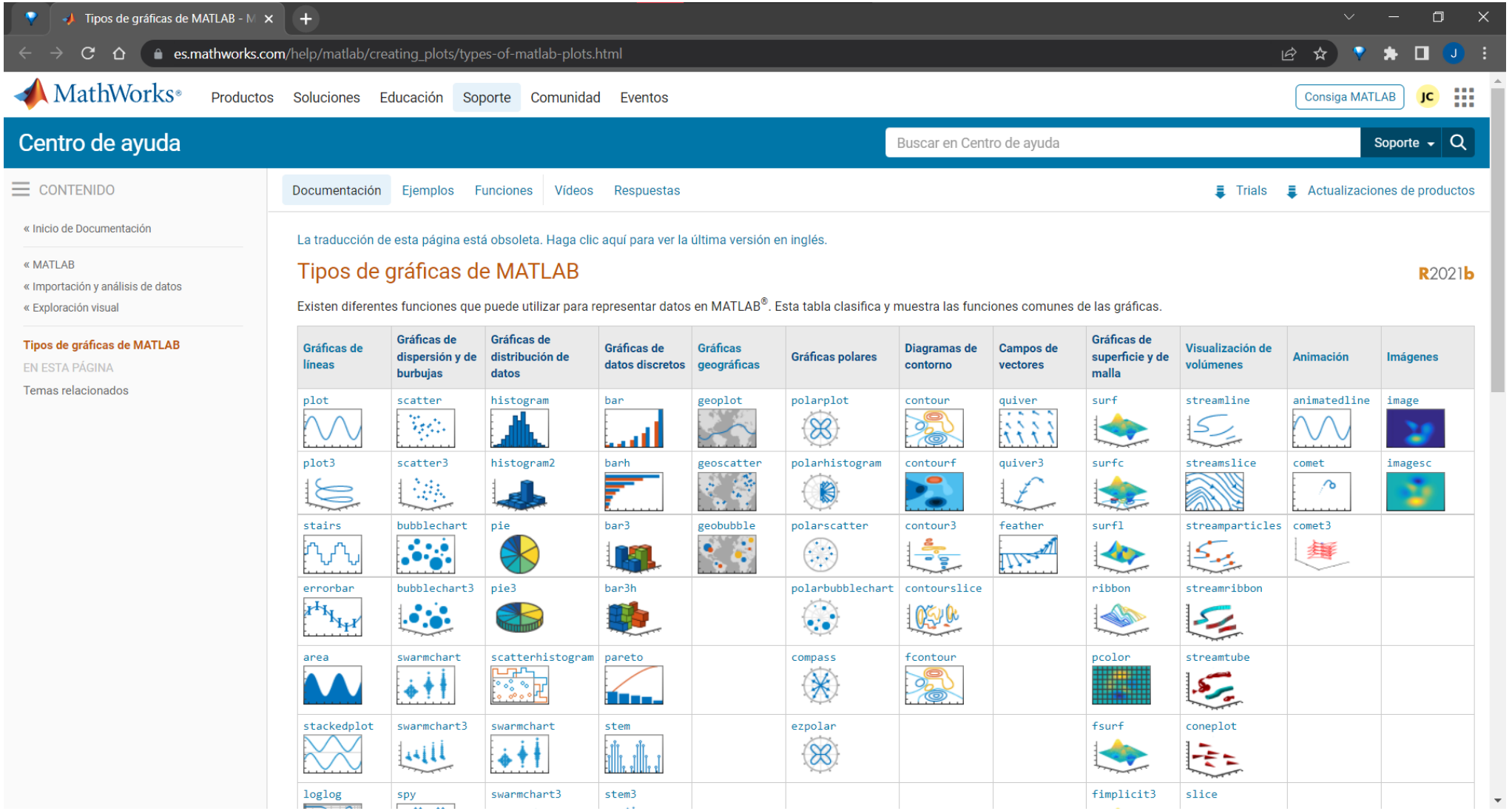
Acquire Data from Android Device Sensors with MATLAB Mobile



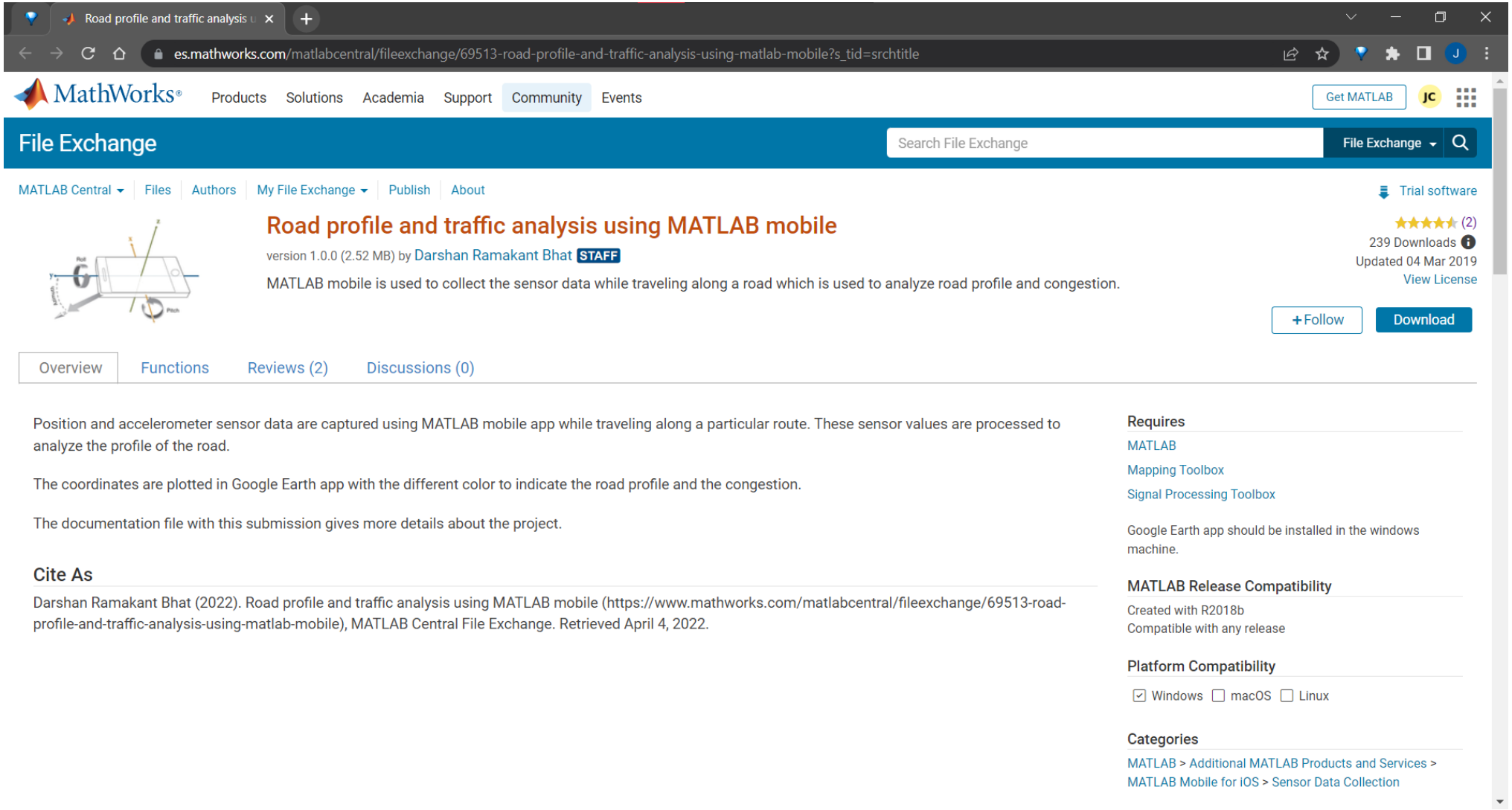
Data Filtering



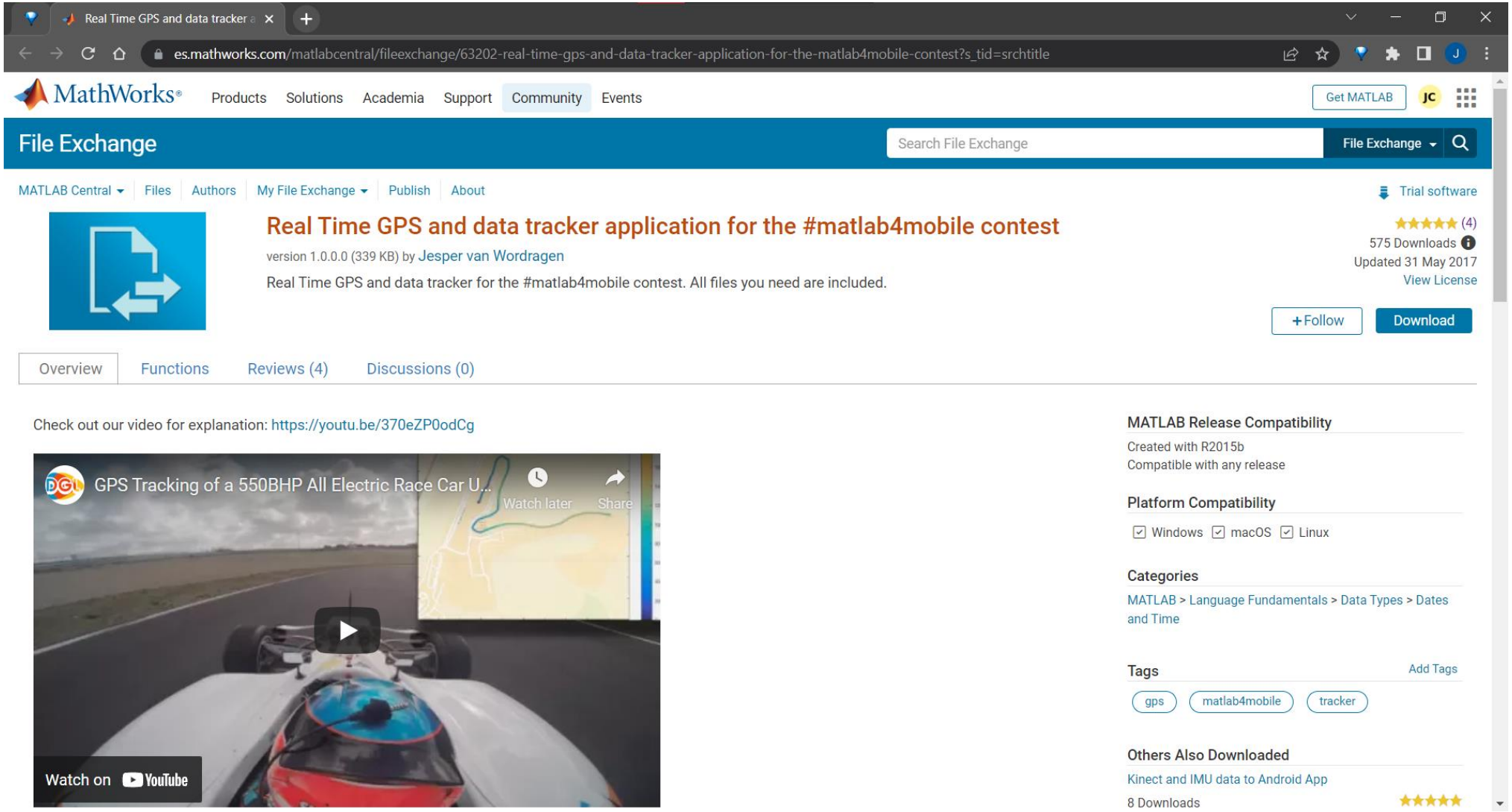
MATLAB Plots



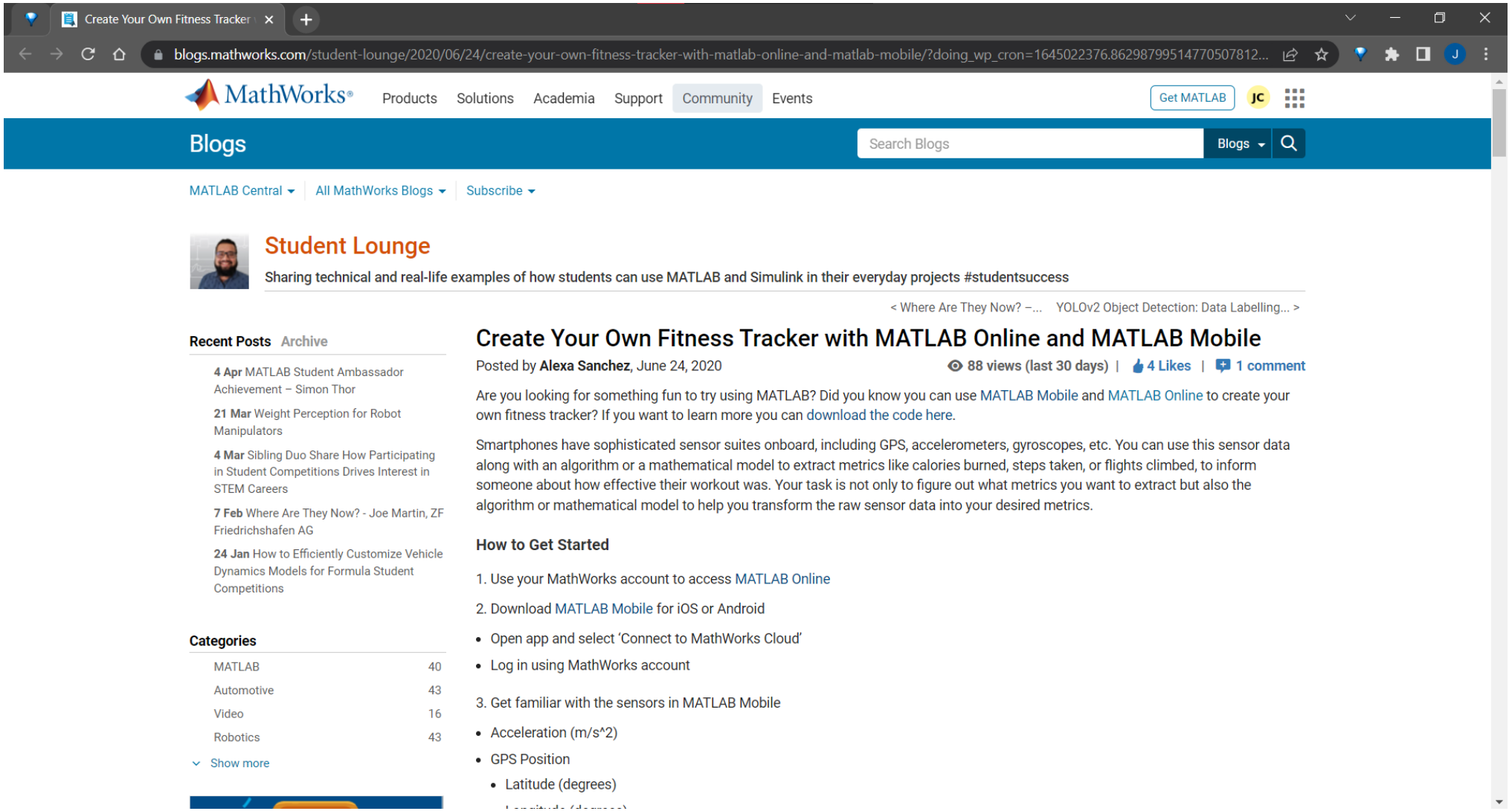
Example: Road profile and traffic analysis using MATLAB mobile



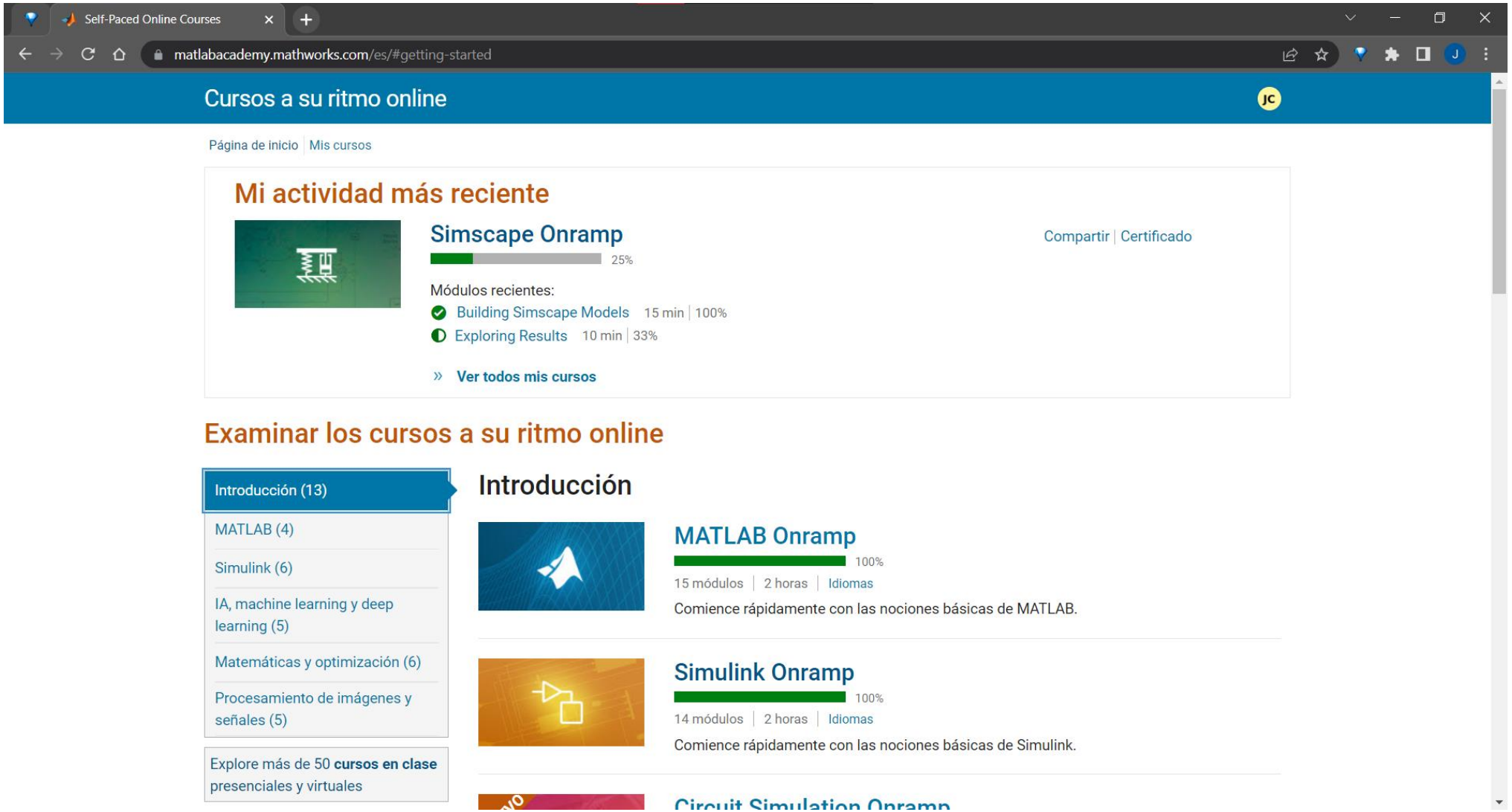
Example: Real Time GPS and data tracker application



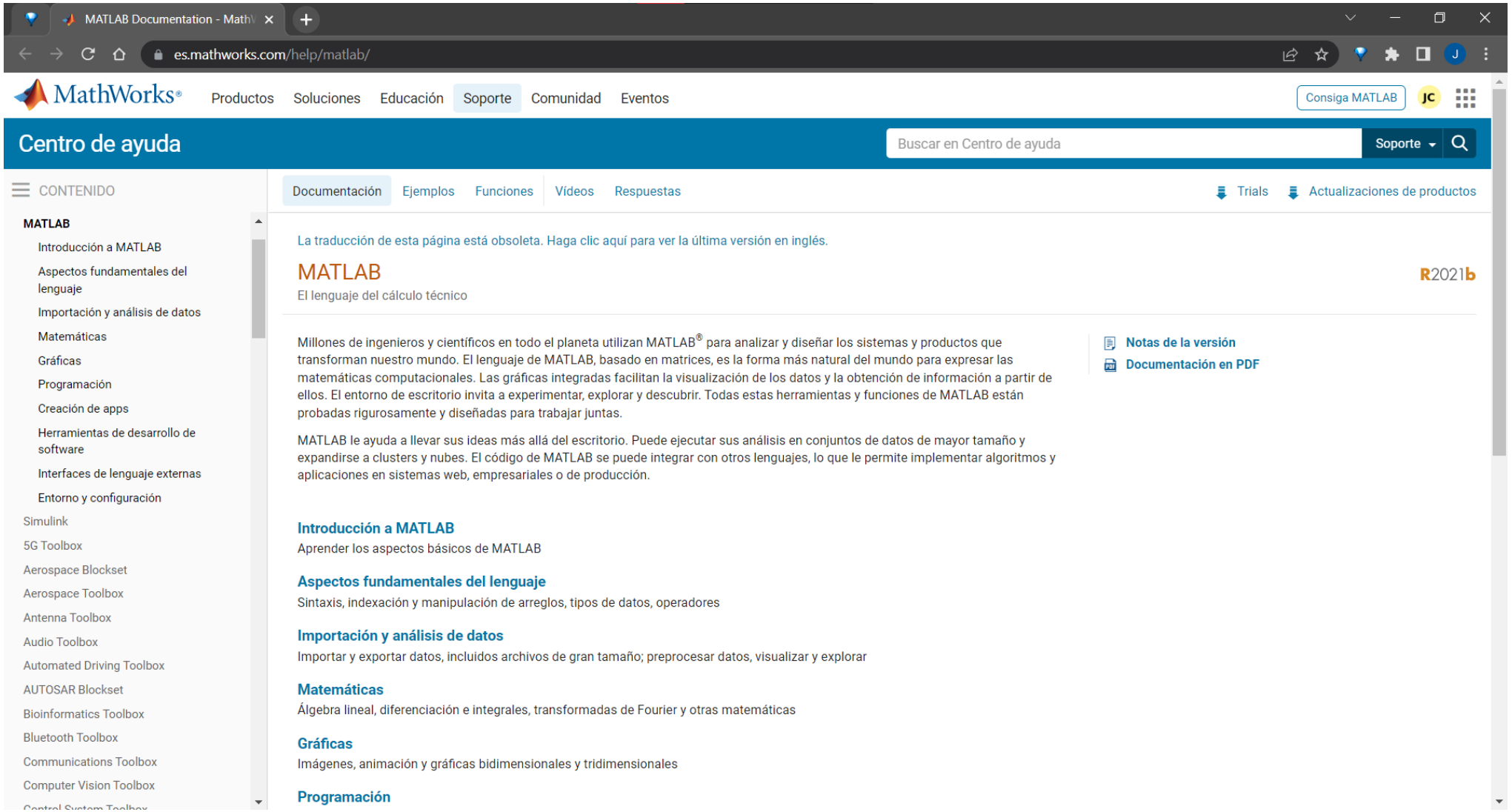
Example: Fitness Tracker with MATLAB Mobile



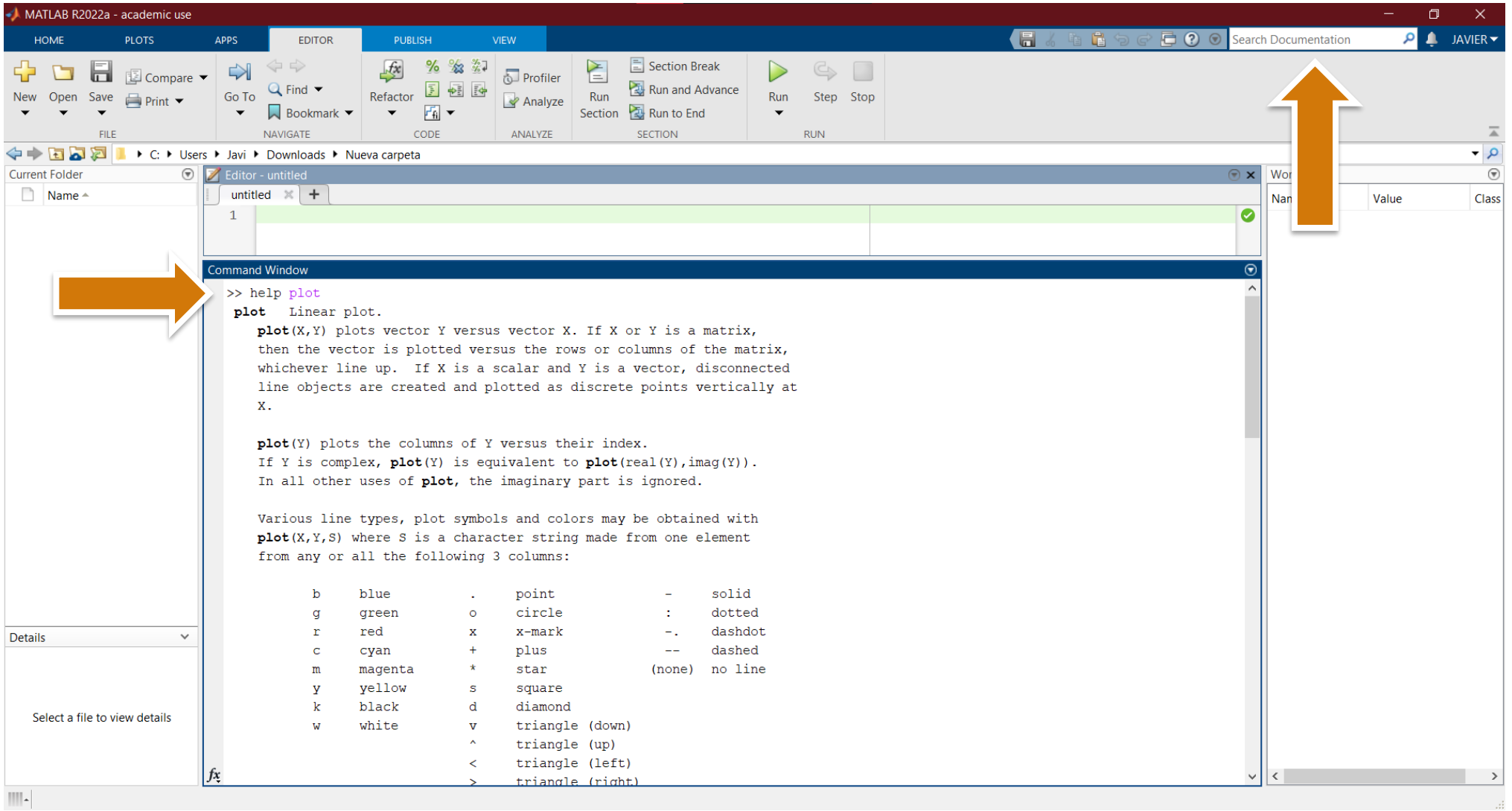
MATLAB Academy

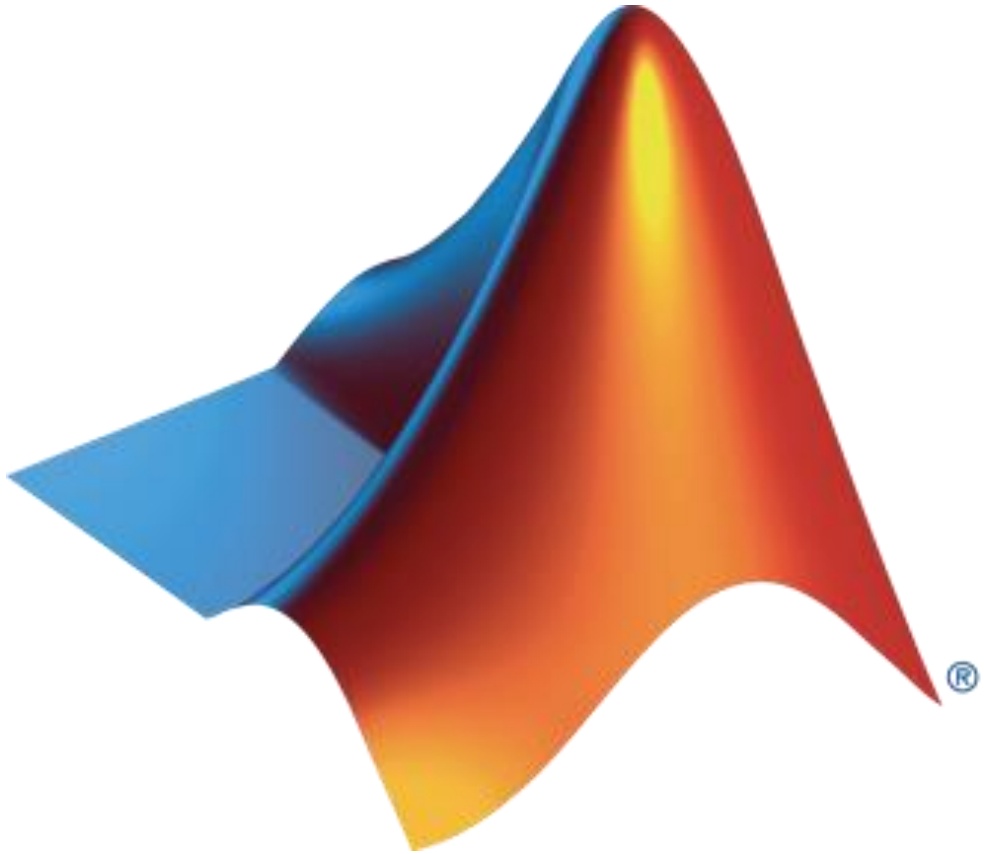


MATLAB Help


MATLAB Documentation - Math
es.mathworks.com/help/matlab/
MathWorks
Productos
Soluciones
Educación
Soporte
Comunidad
Eventos
Consiga MATLAB
JC
Centro de ayuda
Buscar en Centro de ayuda
Soporte
CONTENIDO
MATLAB
Introducción a MATLAB
Aspectos fundamentales del lenguaje
Importación y análisis de datos
Matemáticas
Gráficas
Programación
Creación de apps
Herramientas de desarrollo de software
Interfaces de lenguaje externas
Entorno y configuración
Simulink
5G Toolbox
Aerospace Blockset
Aerospace Toolbox
Antenna Toolbox
Audio Toolbox
Automated Driving Toolbox
AUTOSAR Blockset
Bioinformatics Toolbox
Bluetooth Toolbox
Communications Toolbox
Computer Vision Toolbox
Documentación
Ejemplos
Funciones
Vídeos
Respuestas
Trials
Actualizaciones de productos
La traducción de esta página está obsoleta. Haga clic aquí para ver la última versión en inglés.
MATLAB
El lenguaje del cálculo técnico
R2021b
Millones de ingenieros y científicos en todo el planeta utilizan MATLAB® para analizar y diseñar los sistemas y productos que transforman nuestro mundo. El lenguaje de MATLAB, basado en matrices, es la forma más natural del mundo para expresar las matemáticas computacionales. Las gráficas integradas facilitan la visualización de los datos y la obtención de información a partir de ellos. El entorno de escritorio invita a experimentar, explorar y descubrir. Todas estas herramientas y funciones de MATLAB están probadas rigurosamente y diseñadas para trabajar juntas.
MATLAB le ayuda a llevar sus ideas más allá del escritorio. Puede ejecutar sus análisis en conjuntos de datos de mayor tamaño y expandirse a clusters y nubes. El código de MATLAB se puede integrar con otros lenguajes, lo que le permite implementar algoritmos y aplicaciones en sistemas web, empresariales o de producción.
Notas de la versión
Documentación en PDF
Introducción a MATLAB
Aprender los aspectos básicos de MATLAB
Aspectos fundamentales del lenguaje
Sintaxis, indexación y manipulación de arreglos, tipos de datos, operadores
Importación y análisis de datos
Importar y exportar datos, incluidos archivos de gran tamaño; preprocesar datos, visualizar y explorar
Matemáticas
Álgebra lineal, diferenciación e integrales, transformadas de Fourier y otras matemáticas
Gráficas
Imágenes, animación y gráficas bidimensionales y tridimensionales
Programación

MATLAB Help


MATLAB R2022a - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Search Documentation
JAVIER
New
Open
Save
Compare
Print
Go To
Find
Bookmark
Refactor
Profiler
Analyze
Run Section
Section Break
Run and Advance
Run to End
Run
Step
Stop
FILE
NAVIGATE
CODE
ANALYZE
SECTION
RUN
C: ▸ Users ▸ Javi ▸ Downloads ▸ Nueva carpeta
Current Folder
Name
Editor - untitled
untitled
1
Command Window
>> help plot
plot Linear plot.
plot(X,Y) plots vector Y versus vector X. If X or Y is a matrix,
then the vector is plotted versus the rows or columns of the matrix,
whichever line up. If X is a scalar and Y is a vector, disconnected
line objects are created and plotted as discrete points vertically at
X.
plot(Y) plots the columns of Y versus their index.
If Y is complex, plot(Y) is equivalent to plot(real(Y),imag(Y)).
In all other uses of plot, the imaginary part is ignored.
Various line types, plot symbols and colors may be obtained with
plot(X,Y,S) where S is a character string made from one element
from any or all the following 3 columns:
b blue . point - solid
g green o circle : dotted
r red x x-mark -. dashdot
c cyan + plus -- dashed
m magenta * star (none) no line
y yellow s square
k black d diamond
w white v triangle (down)
^ triangle (up)
< triangle (left)
> triangle (right)
Details
Select a file to view details
Value
Class



MATHACK 2022 Training Sessions
MATLAB Mobile Data Acquisition

ALESSIO TUMMINELLO
UNIPI Ambassador

UNIVERSITÀ DI PISA

MATLAB Mobile Data Acquisition

The idea of this MATLAB and MATLAB Mobile example is count the steps as a fitness tracker using the data acquired from smartphone.

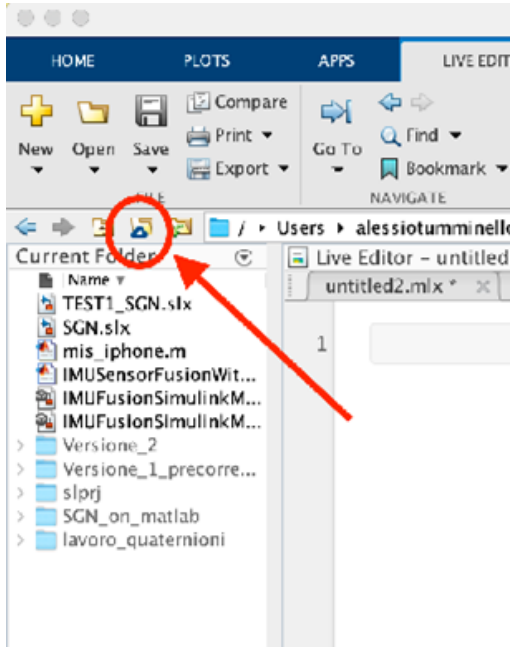
To realize our task we have 2 options:

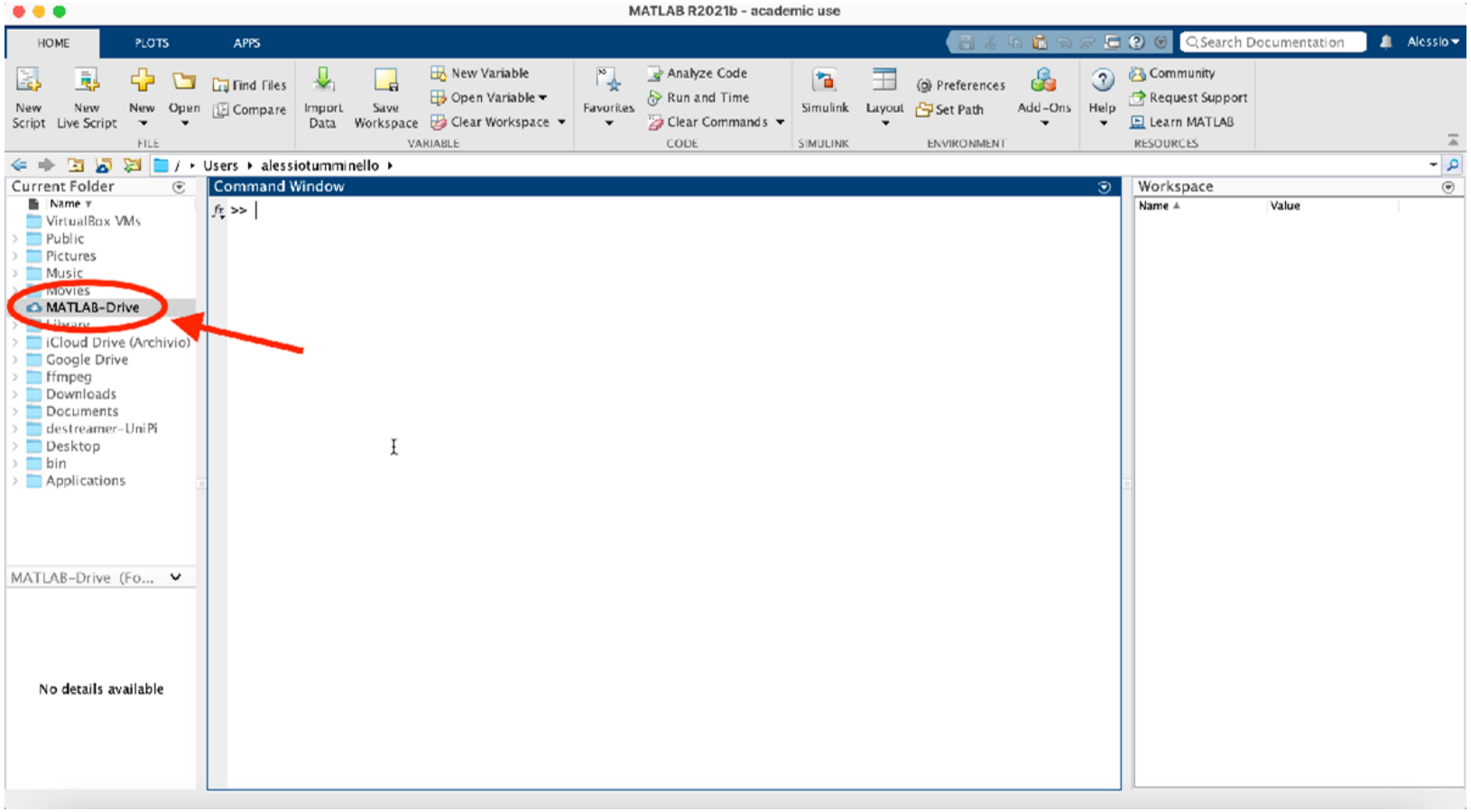
- Log data locally on your smartphone and load automatically on MATLAB Drive, after that we can move on MATLAB laptop app and load file with all the measurements from MATLAB Drive folder

- Stream data from MATLAB Mobile to MATLAB and acquire immediately the data

MATLAB Mobile Data Acquisition – **Logging Data**

1) MATLAB Drive Connector

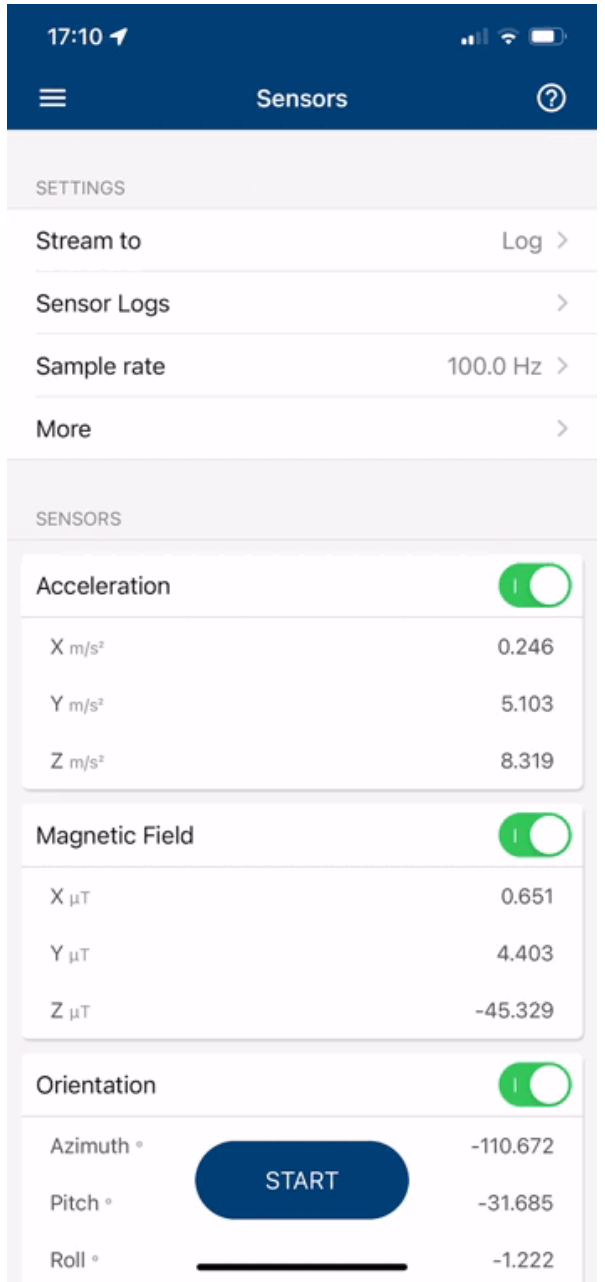




MATLAB Mobile Data Acquisition – **Logging Data**

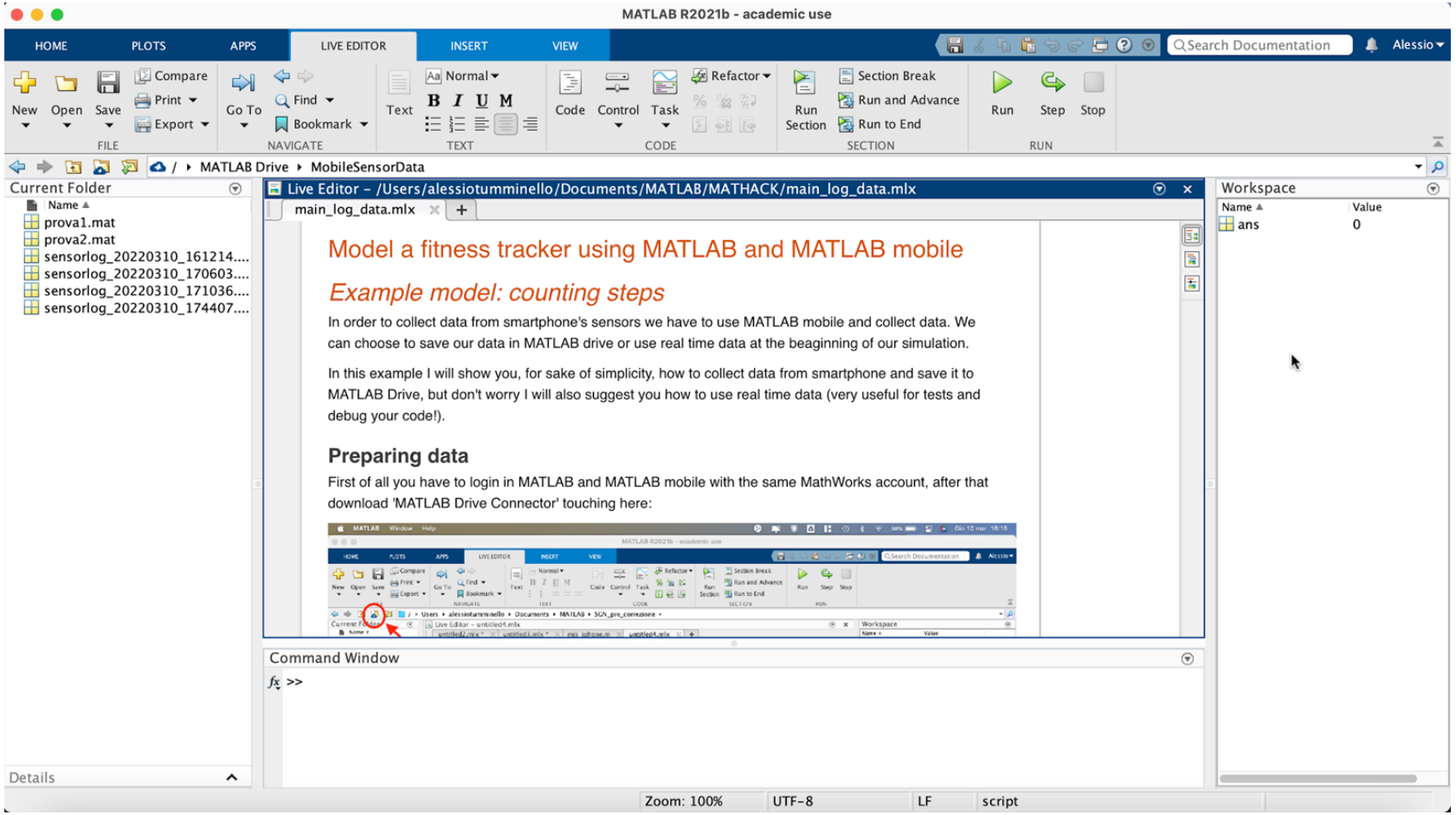
2) Let's move to MATLAB Mobile !

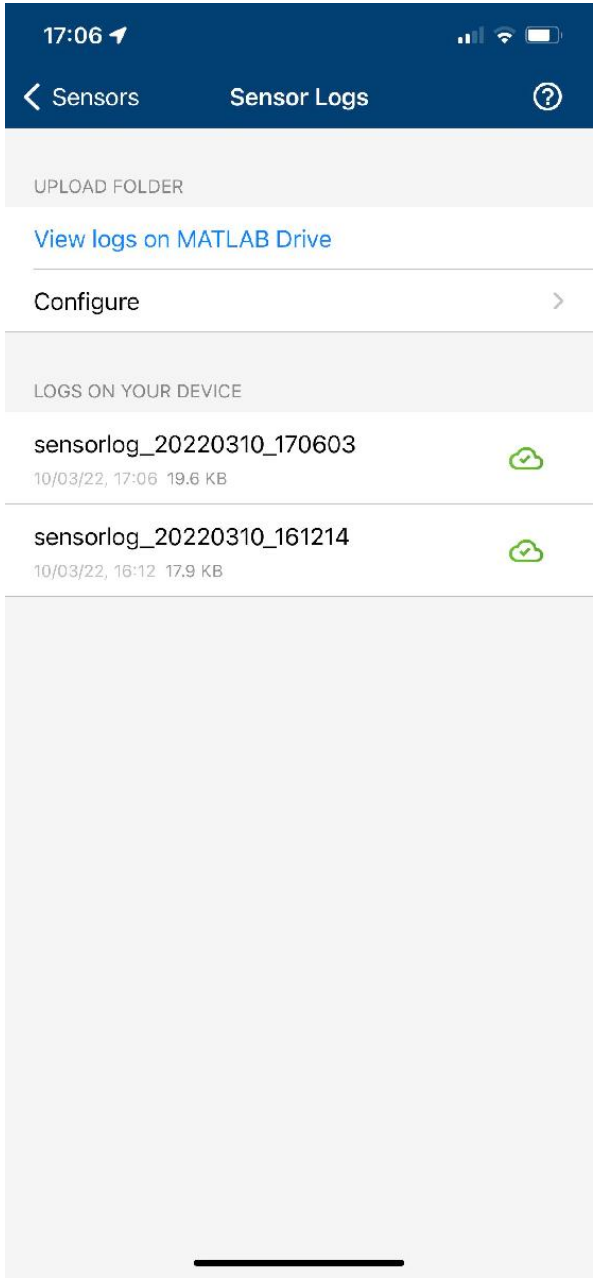
- In the sensor tab you have to decided where you want to **stream your data** (log locally in your smartphone and automatically load to MATLAB Drive folder) and also choose the sensor's **sample rate** (usually 100Hz for a realistic IMU and GPS).



MATLAB Mobile Data Acquisition – **Logging Data**

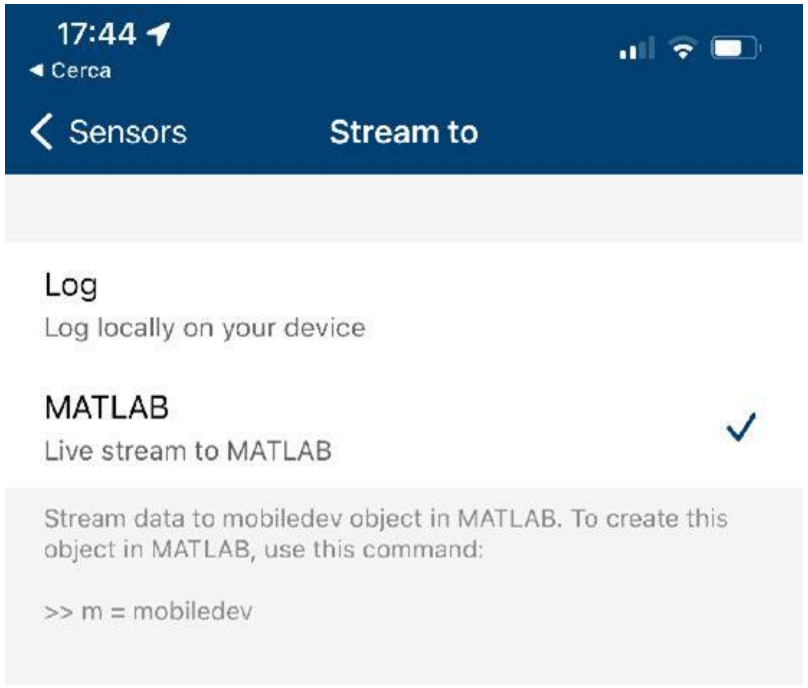
3) Save data on MATLAB Drive folder

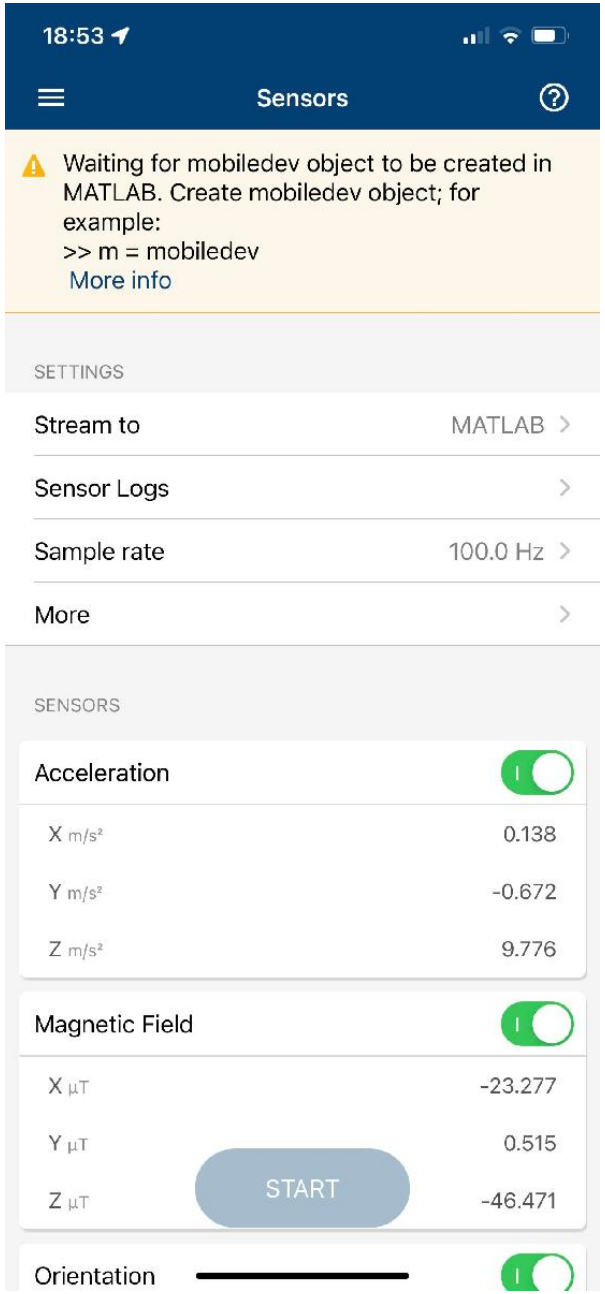




MATLAB Mobile Data Acquisition – **Stream Data**

1) Choose to stream data

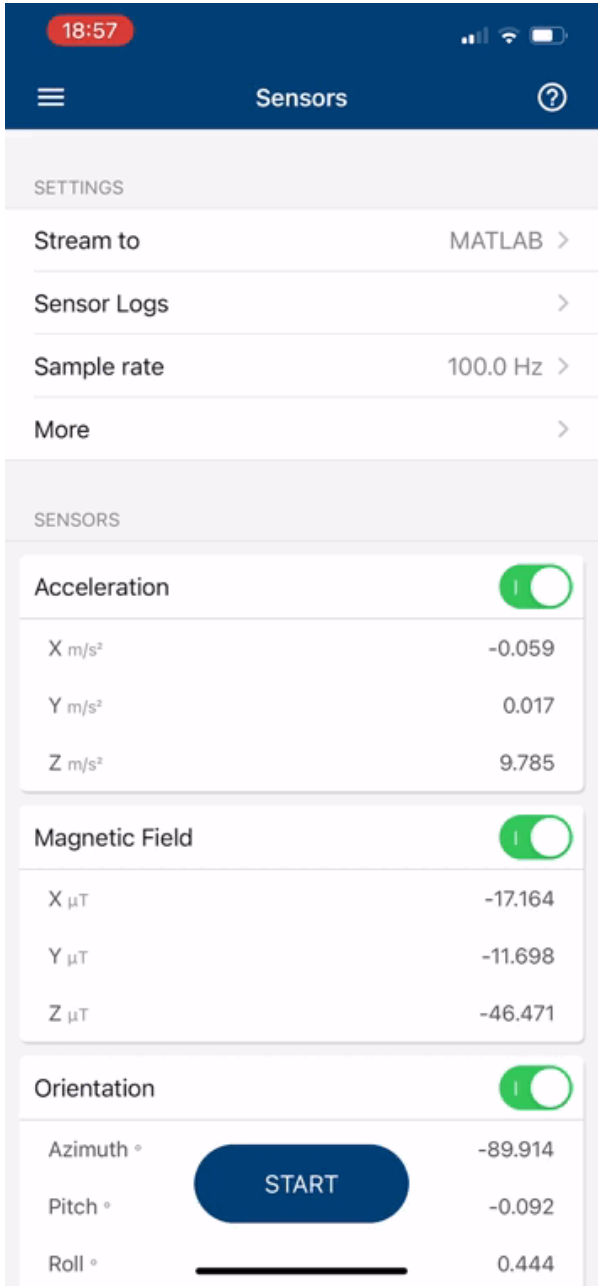




MATLAB Mobile Data Acquisition – **Stream Data**

2) Create a 'mobiledev object' on MATLAB

```
iPhonediAlessio = mobiledev;
iPhonediAlessio.SampleRate = 100; %Sets sample rate at which device will acquire the data

%%Flag which sensor you want to use
iPhonediAlessio.AngularVelocitySensorEnabled = 1;
iPhonediAlessio.OrientationSensorEnabled = 1;
iPhonediAlessio.AccelerationSensorEnabled = 1;
iPhonediAlessio.PositionSensorEnabled = 1;
iPhonediAlessio.MagneticSensorEnabled = 1;

loggingTime = 10; % seconds

iPhonediAlessio.Logging = 1; % start the transmission of data from all selected sensors
pause(loggingTime);
```



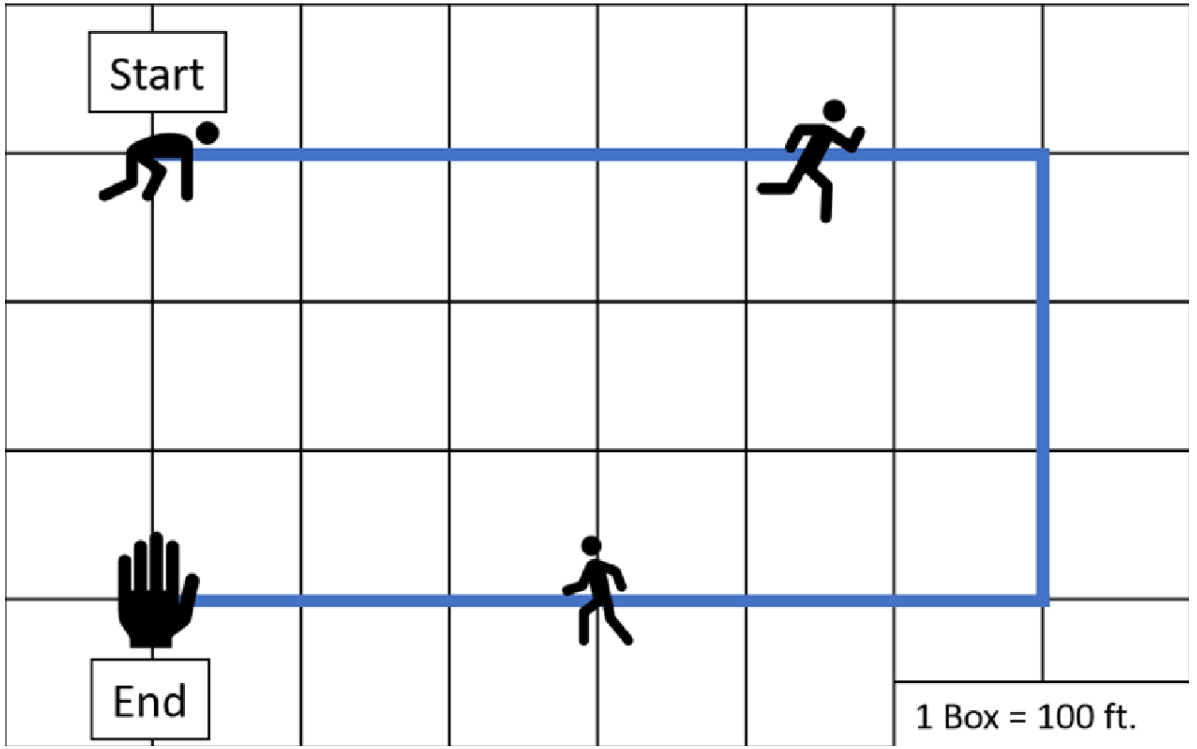
Processing Demo

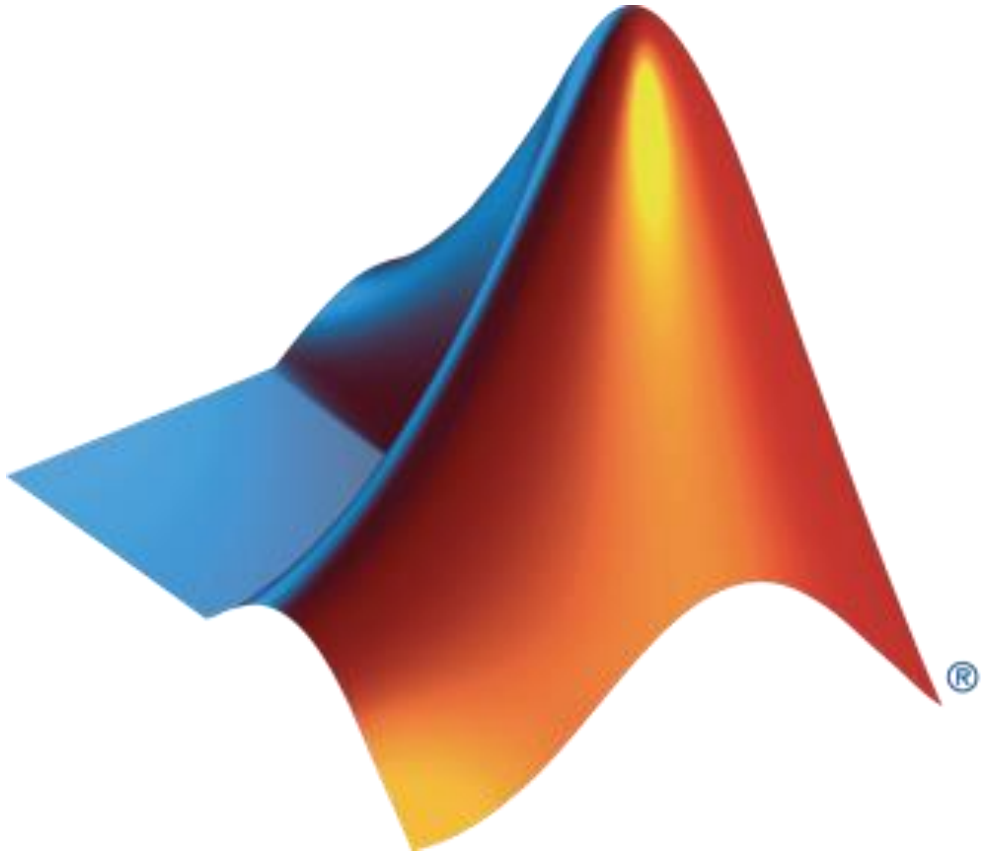
Fitness Tracker Demo with steps counter

$$\frac{\text{totalDistance(ft)}}{\text{stride}\left(\frac{\text{ft}}{\text{step}}\right)} = \frac{\text{ft}}{\frac{\text{ft}}{\text{step}}} = \text{ft} * \frac{\text{step}}{\text{ft}} = \text{steps}$$

$$\frac{\text{traveledDistance}}{\text{earth'sCircumference}} = \frac{\text{degreesTraveled}}{\text{earth'sDegrees}}$$

```
for i = 1:(length(lat)-1)

    lat1 = lat(i);     %The first latitude
    lat2 = lat(i+1);   %The second latitude
    lon1 = lon(i);
    lon2 = lon(i+1);

    diff = distance(lat1, lon1, lat2, lon2);
    dis = (diff/360)*earthCirc;

    totaldis = totaldis + dis;

end
```





MATHACK 2022 Training Sessions
DNN in MATLAB - Running detection

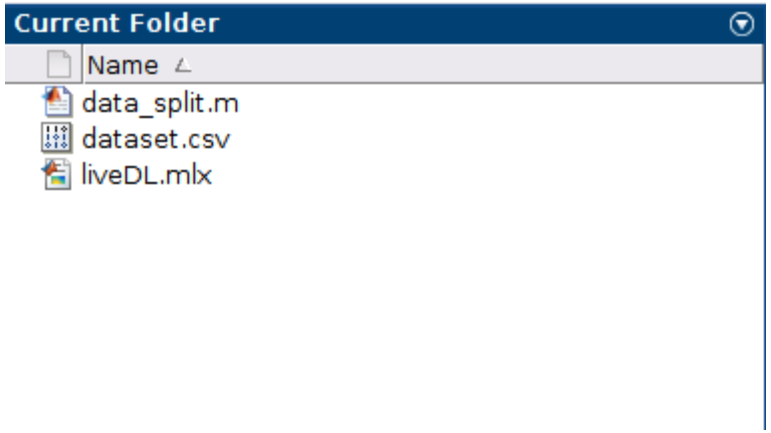
GIULIO MATTERA
UNINA Ambassador

UNIVERSITÀ DEGLI STUDI DI NAPOLI
FEDERICO II

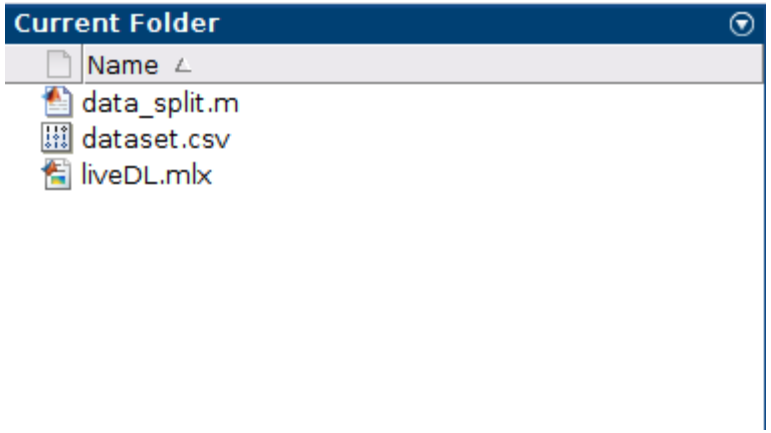
DNN application (I) : Dataset

- Download Dataset form **Kaggle:**
  - https://www.kaggle.com/datasets/vmalyi/run-or-walk
- This dataset contain:
  - Gyro x,y,z
  - Accelerometer x,y,z
    - Filtered
    - Scaled
    - Normalized
  - Run is labeled as 1, otherwise 0 (walking)

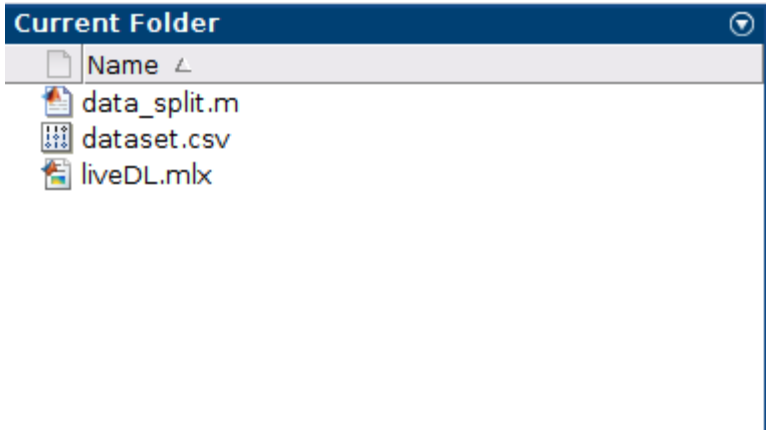
DNN application (II) : Dataset importing

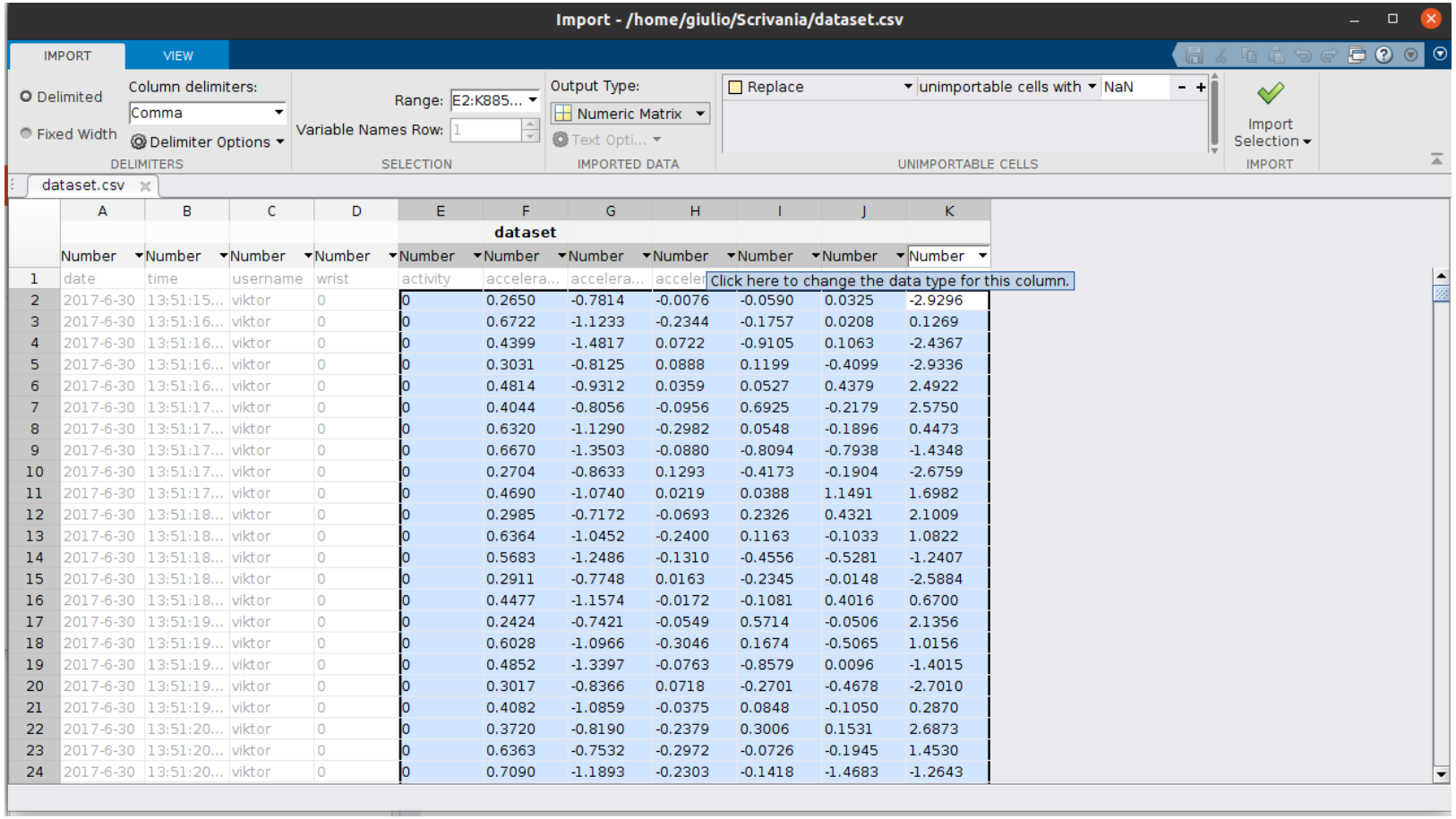


DNN application (II) : Dataset importing



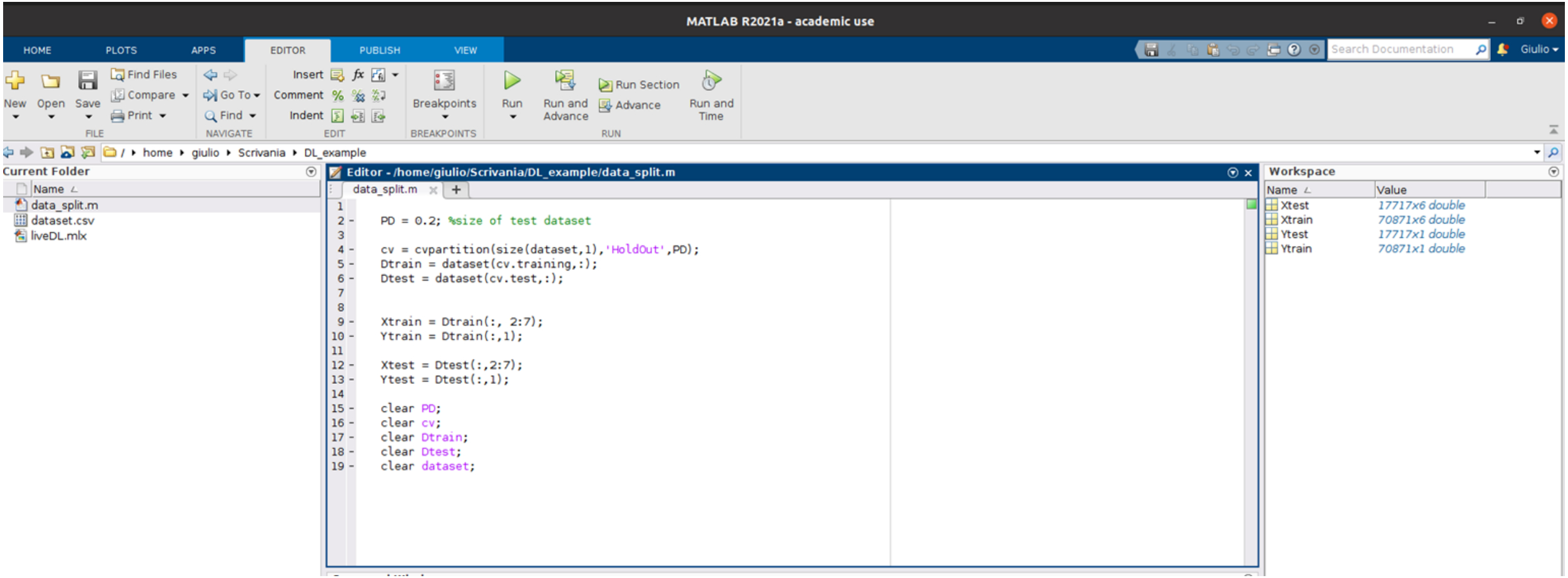
DNN application (II) : Dataset importing



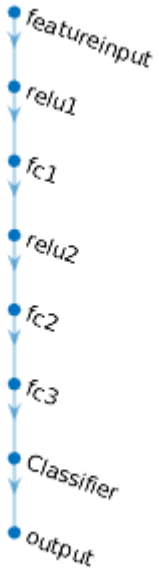


	A	B	C	D	E	F	G	H	I	J	K
	Number	Number	Number	Number	Number	Number	Number	Number	Number	Number	Number
1	date	time	username	wrist	activity	accelera...	accelera...	acceler			
2	2017-6-30	13:51:15...	viktor	0	0	0.2650	-0.7814	-0.0076	-0.0590	0.0325	-2.9296
3	2017-6-30	13:51:16...	viktor	0	0	0.6722	-1.1233	-0.2344	-0.1757	0.0208	0.1269
4	2017-6-30	13:51:16...	viktor	0	0	0.4399	-1.4817	0.0722	-0.9105	0.1063	-2.4367
5	2017-6-30	13:51:16...	viktor	0	0	0.3031	-0.8125	0.0888	0.1199	-0.4099	-2.9336
6	2017-6-30	13:51:16...	viktor	0	0	0.4814	-0.9312	0.0359	0.0527	0.4379	2.4922
7	2017-6-30	13:51:17...	viktor	0	0	0.4044	-0.8056	-0.0956	0.6925	-0.2179	2.5750
8	2017-6-30	13:51:17...	viktor	0	0	0.6320	-1.1290	-0.2982	0.0548	-0.1896	0.4473
9	2017-6-30	13:51:17...	viktor	0	0	0.6670	-1.3503	-0.0880	-0.8094	-0.7938	-1.4348
10	2017-6-30	13:51:17...	viktor	0	0	0.2704	-0.8633	0.1293	-0.4173	-0.1904	-2.6759
11	2017-6-30	13:51:17...	viktor	0	0	0.4690	-1.0740	0.0219	0.0388	1.1491	1.6982
12	2017-6-30	13:51:18...	viktor	0	0	0.2985	-0.7172	-0.0693	0.2326	0.4321	2.1009
13	2017-6-30	13:51:18...	viktor	0	0	0.6364	-1.0452	-0.2400	0.1163	-0.1033	1.0822
14	2017-6-30	13:51:18...	viktor	0	0	0.5683	-1.2486	-0.1310	-0.4556	-0.5281	-1.2407
15	2017-6-30	13:51:18...	viktor	0	0	0.2911	-0.7748	0.0163	-0.2345	-0.0148	-2.5884
16	2017-6-30	13:51:18...	viktor	0	0	0.4477	-1.1574	-0.0172	-0.1081	0.4016	0.6700
17	2017-6-30	13:51:19...	viktor	0	0	0.2424	-0.7421	-0.0549	0.5714	-0.0506	2.1356
18	2017-6-30	13:51:19...	viktor	0	0	0.6028	-1.0966	-0.3046	0.1674	-0.5065	1.0156
19	2017-6-30	13:51:19...	viktor	0	0	0.4852	-1.3397	-0.0763	-0.8579	0.0096	-1.4015
20	2017-6-30	13:51:19...	viktor	0	0	0.3017	-0.8366	0.0718	-0.2701	-0.4678	-2.7010
21	2017-6-30	13:51:19...	viktor	0	0	0.4082	-1.0859	-0.0375	0.0848	-0.1050	0.2870
22	2017-6-30	13:51:20...	viktor	0	0	0.3720	-0.8190	-0.2379	0.3006	0.1531	2.6873
23	2017-6-30	13:51:20...	viktor	0	0	0.6363	-0.7532	-0.2972	-0.0726	-0.1945	1.4530
24	2017-6-30	13:51:20...	viktor	0	0	0.7090	-1.1893	-0.2303	-0.1418	-1.4683	-1.2643

DNN application (III) : Dataset splitting

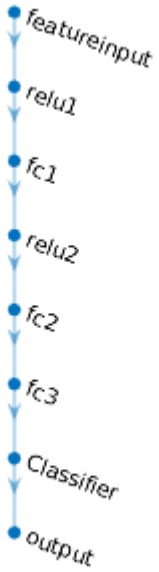


DNN application (IV) : Architecture




```
layers = [
    featureInputLayer(6,"Name","featureinput")
    reluLayer("Name","relu1")
    fullyConnectedLayer(128,"Name","fc1")
    reluLayer("Name","relu2")
    fullyConnectedLayer(64,"Name","fc2")
    fullyConnectedLayer(1,"Name","fc3")
    sigmoidLayer("Name", "Classifier")
    regressionLayer("Name","output")
    ];
```

DNN application (V) : Options

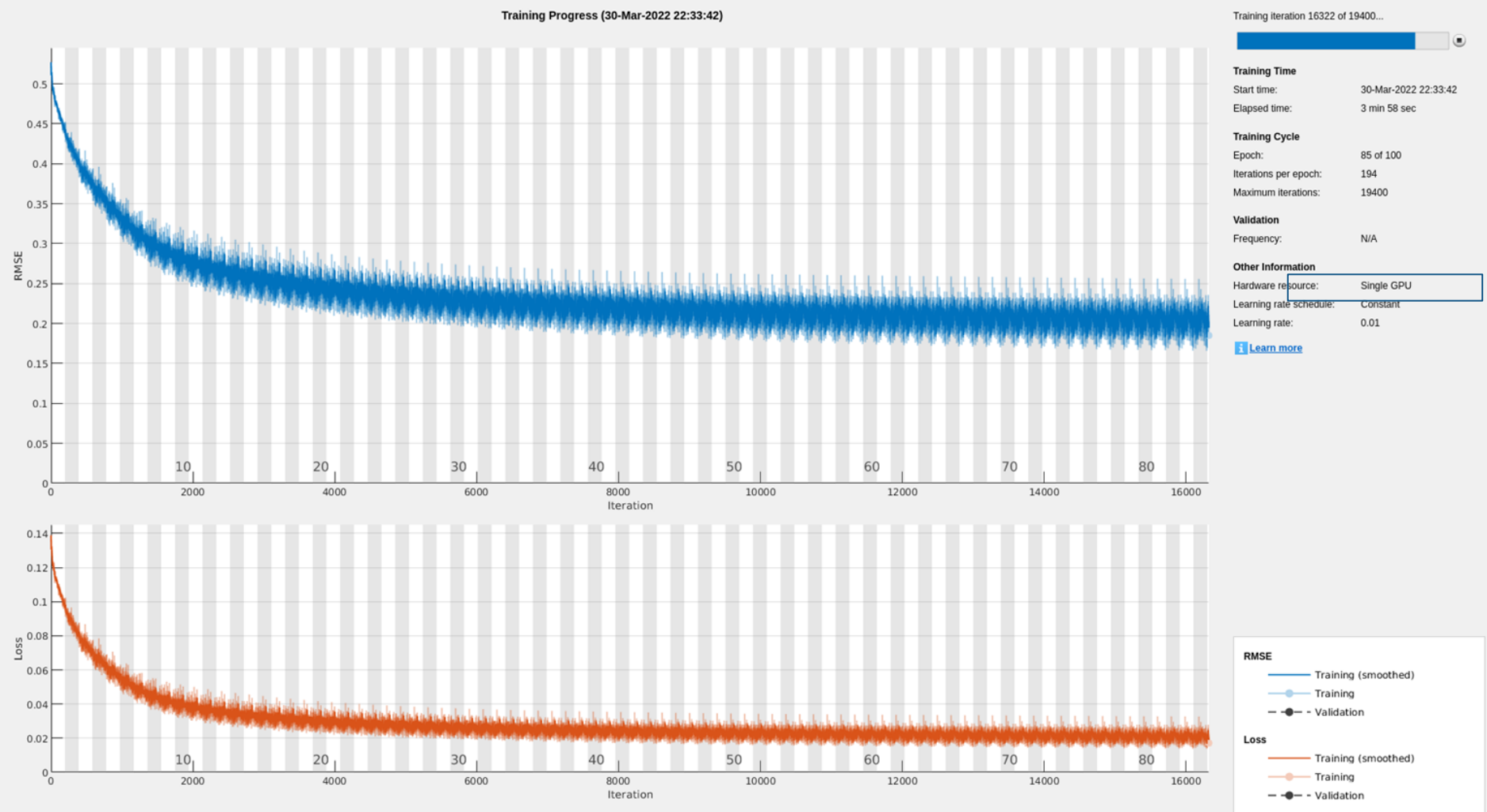



```
layers = [
    featureInputLayer(6,"Name","featureinput")
    reluLayer("Name","relu1")
    fullyConnectedLayer(128,"Name","fc1")
    reluLayer("Name","relu2")
    fullyConnectedLayer(64,"Name","fc2")
    fullyConnectedLayer(1,"Name","fc3")
    sigmoidLayer("Name", "Classifier")
    regressionLayer("Name","output")
    ];
```

```
options = trainingOptions("sgdm", ...
    MaxEpochs=100, ...
    InitialLearnRate=0.01, ...
    Plots="training-progress", ...
    MiniBatchSize=364, ...
    Verbose=1)
```

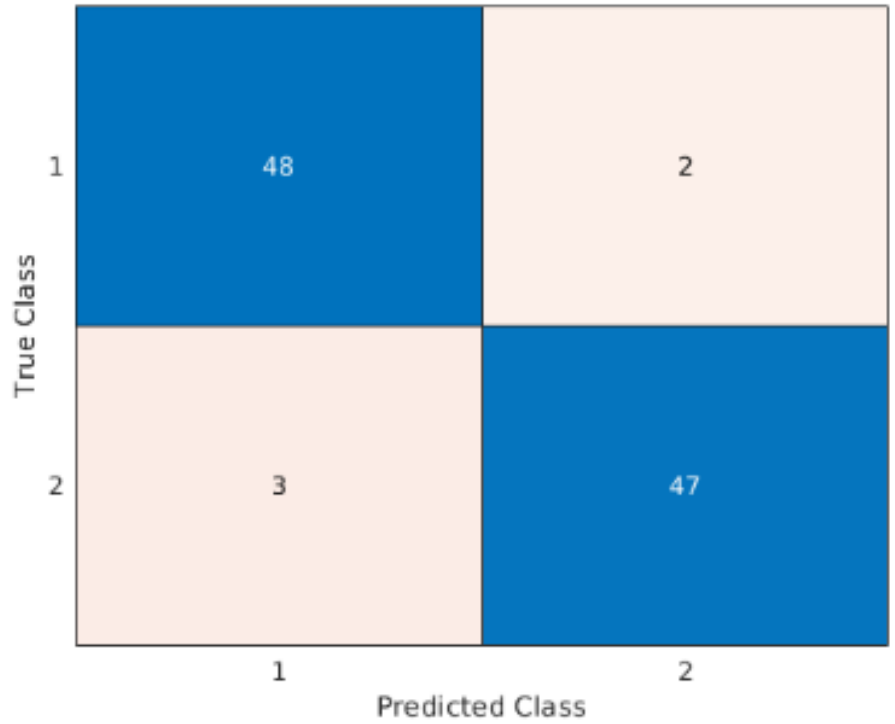
DNN application (VI) : Training

```
net = trainNetwork(Xtrain,Ytrain,layers,options);
```

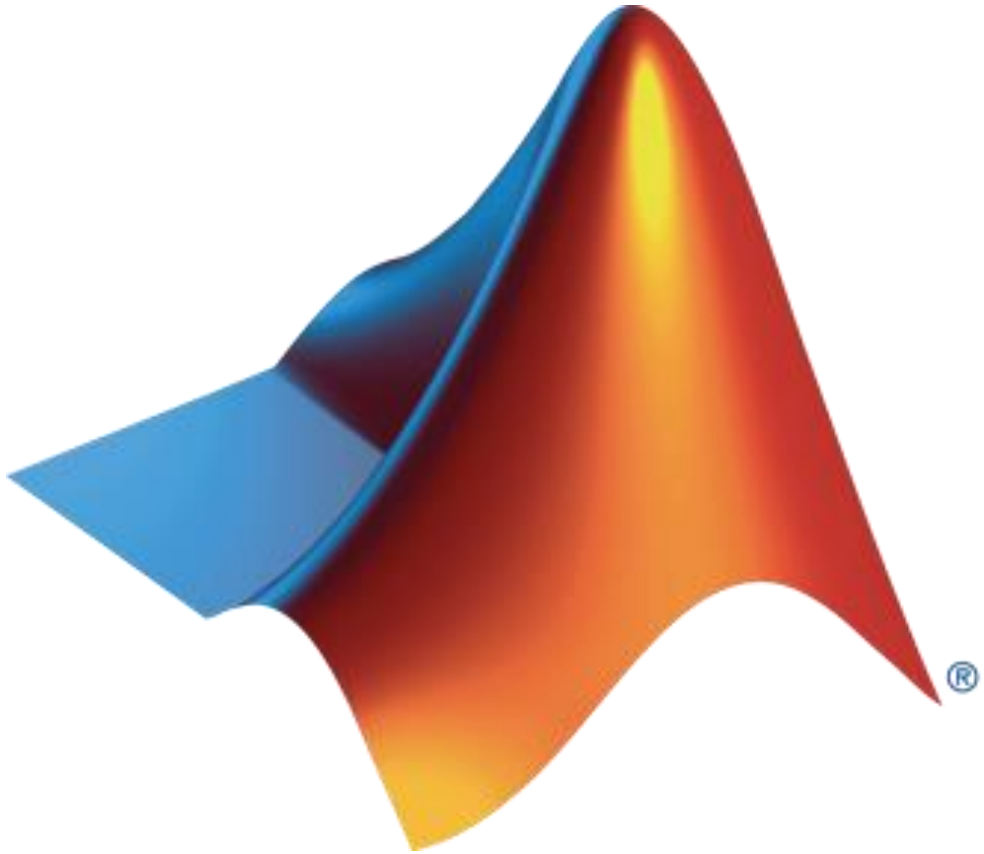


DNN application (VII) : Testing and measuring performance

```
Y = predict(net,Xtest);
for i=1:size(Y,1)
if Y(i,1) > 0.5
    Y(i,1) = 1;
else
    Y(i,1) = 0;
end
end

c = confusionmat(Ytest, double(Y));
c = (c/size(Y,1).*100);
c = round(c)
confusionchart(c)
acc = (c(1,1)+c(2,2))
disp('Clssficiation accuracy is in percent: ')
disp(acc)
```




```
acc = 95
Clssficiation accuracy is in percent:
    95
```



MATHACK 2022 Training Sessions
Intro to Git and Code Submission

ARTURO FERNÁNDEZ
UPV Ambassador

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Project Submission

- **Submission Deadline → 7/04 19:30h CEST**
- **What to Submit?**
  - **URL to your Code Repository (e.g., GitHub) or File Exchange**

What is Git?

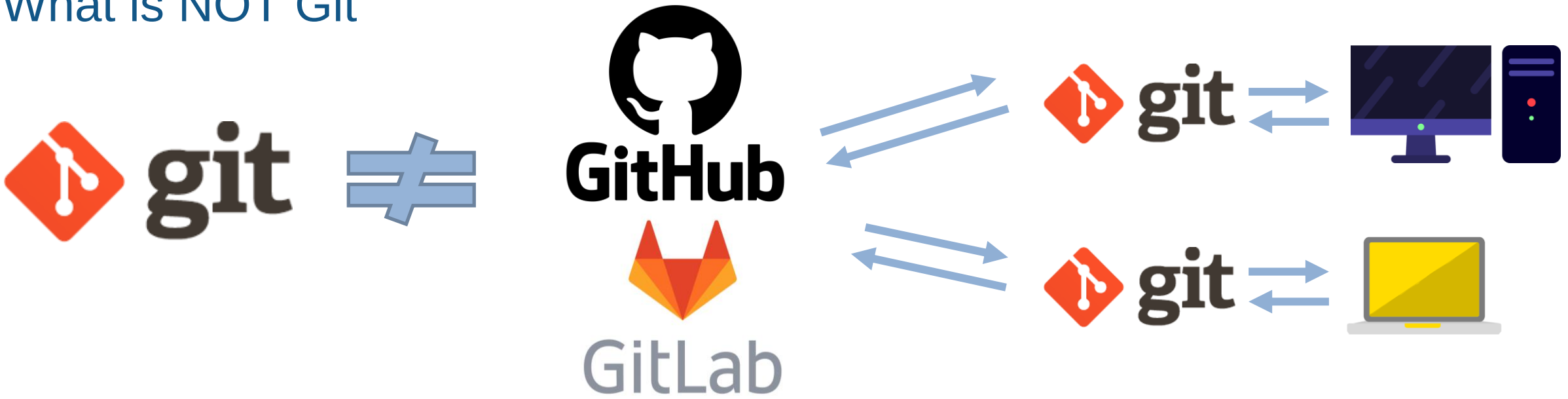
"Free and open-source distributed version control system (**VCS**)"

Purpose: Share code and collaborate on Software Projects

Git's core functionality can be simplified into 3 basic parts:

1. Allow developers to keep track of updates to their code over time
2. Allow developers to easily combine their code updates with previous updates or new updates made by other people
3. Allow developers to easily share code over the Internet

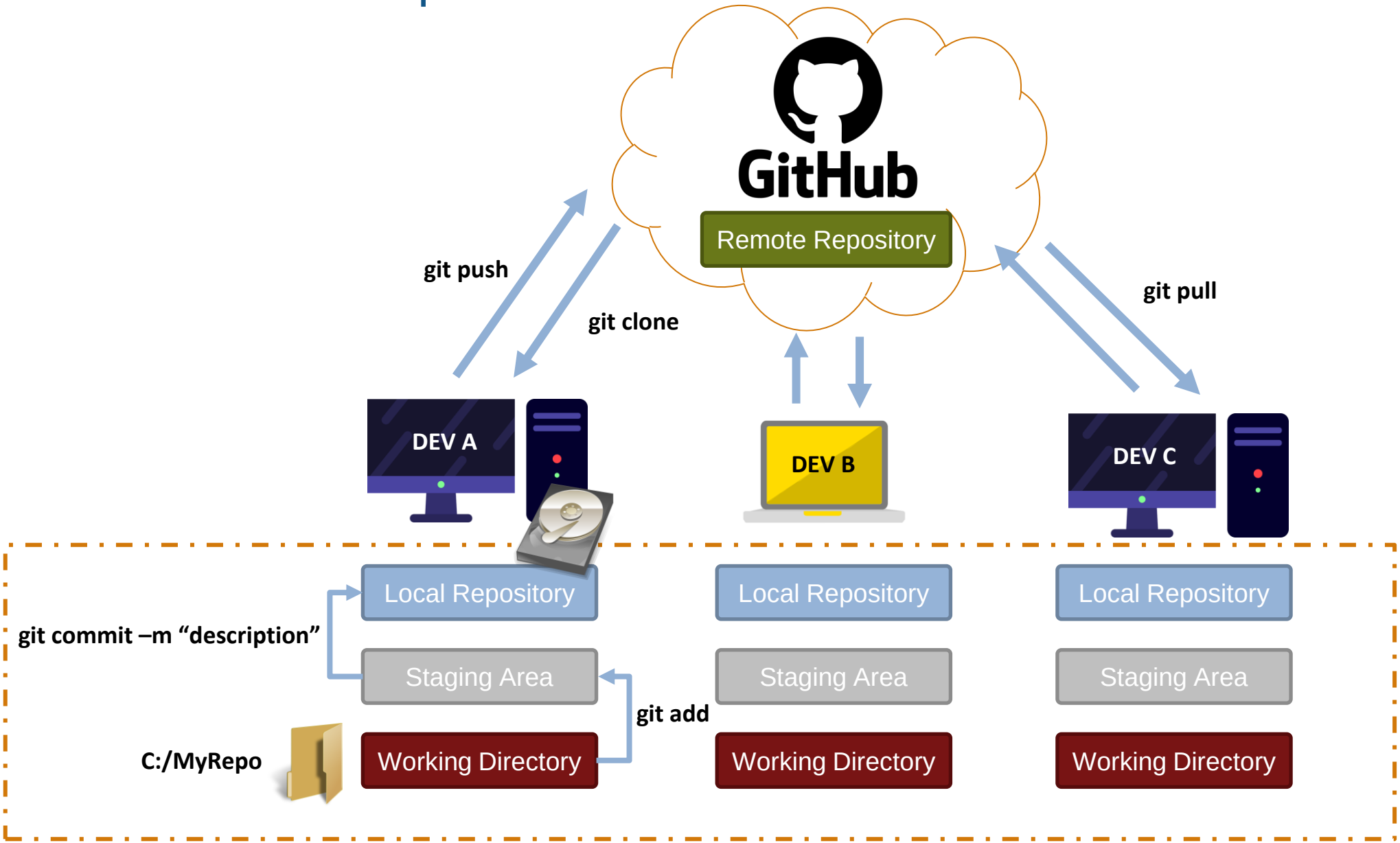
What is NOT Git



¿Are there other alternatives to Git?

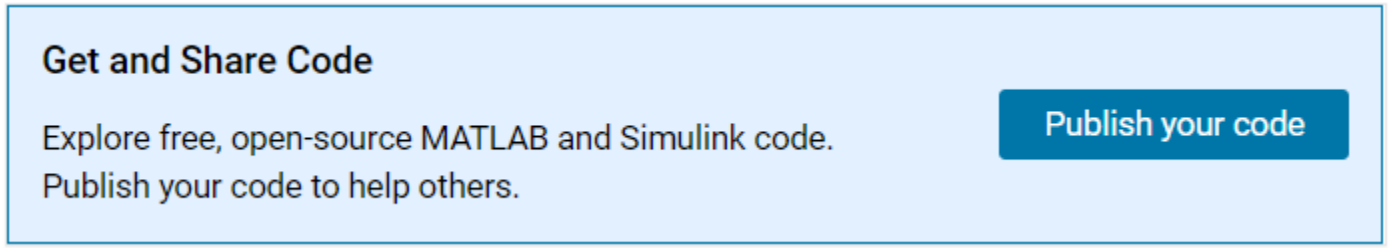
PERFORCE

Basic Git Concepts and Commands

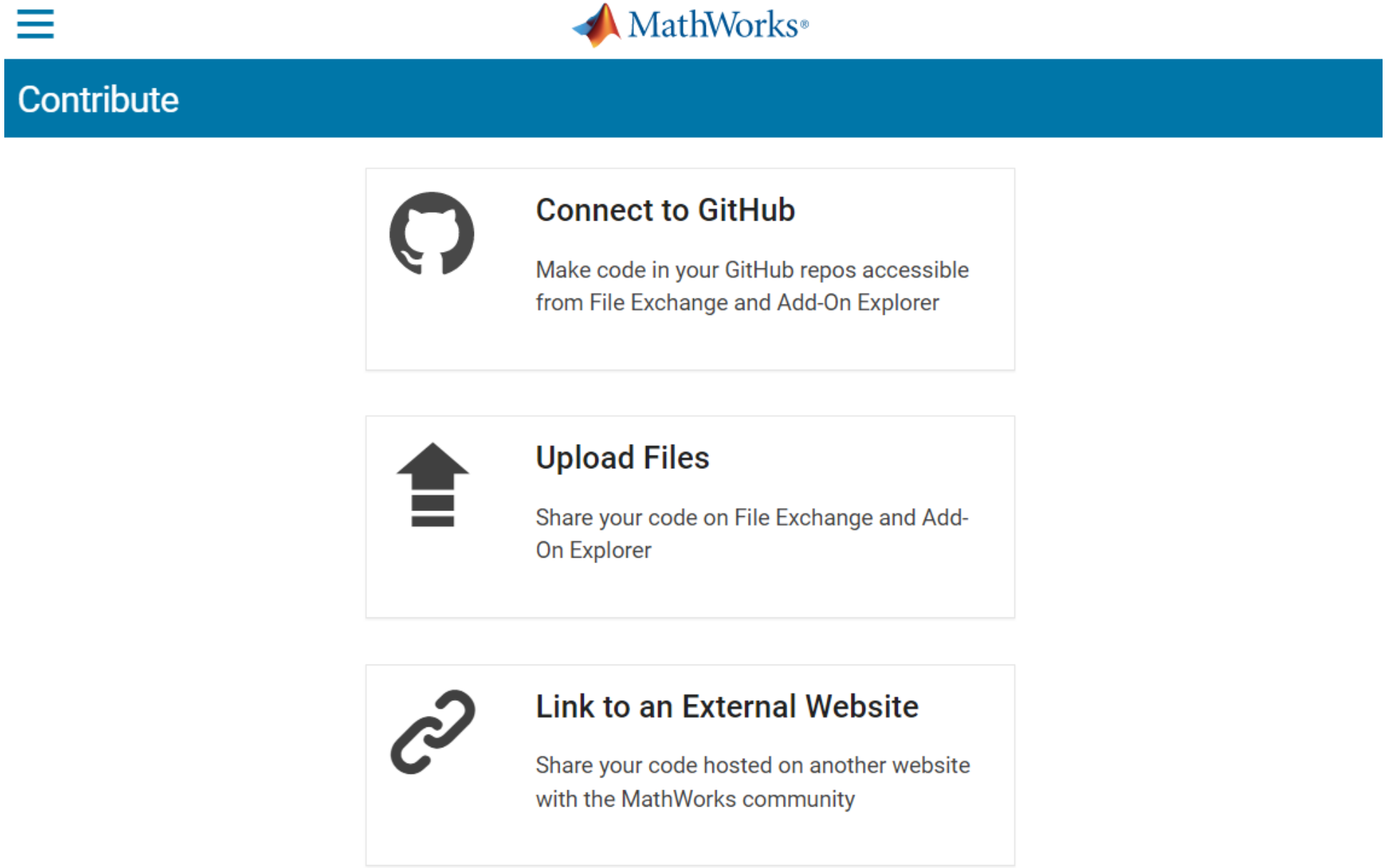


File Exchange Code Submission

- Steps:
  1. Login to www.mathworks.com
  2. Access to File Exchange
  3. Click Publish your Code Button →



File Exchange Code Submission



File Exchange Code Submission

File Exchange Code Submission