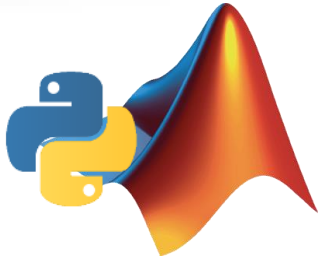
Using MATLAB with Python

Kantika Wongkasem
Application Engineer
Ascendas Systems Co.,Ltd
Kantika@ascendas-asia.com



Date : 27th Jan 2023

Time : 09:00 – 11:00

R2022b

MathWorks Authorized Reseller | MATLAB & SIMULINK

TECHSOURCE

ASCENDAS SYSTEMS

MathWorks® Authorized Reseller | MATLAB® & SIMULINK®

As the **Authorized Reseller** in Southeast Asia for MathWorks Inc, developer of the MATLAB® and Simulink® family of products since 1996, we provide organizations and businesses with a wide spectrum of the best tools, products and services to facilitate innovations.

SINGAPORE'S OUTSTANDING ENTERPRISE
SME 150/2013

TÜV SÜD ISO 9001
CERT NO.: 2006-2-1559
ISO 9001 : 2008

Headquarters

Natick, MA USA

North America

United States

Europe

Finland
France
Germany
Ireland
Italy
Netherlands
Spain
Sweden
Switzerland
UK

Asia-Pacific

Australia
China
India
Japan
Korea

MathWorks®

4 million+ users

in more than 180 countries

5000+ staff

in 31 offices around the world

$1+ billion

in 2019 revenues with 60% from outside the US

Privately held

and profitable every year

DATA QUADRANT AWARDS 2022
Machine Learning

MathWorks®

MathWorks

MathWorks Matlab

MATLAB is a high-level language and interactive environment for numerical computation, visualization, and programming.

8.8	+98	92%
COMPOSITE SCORE	EMOTIONAL FOOTPRINT	LIKELINESS TO RECOMMEND

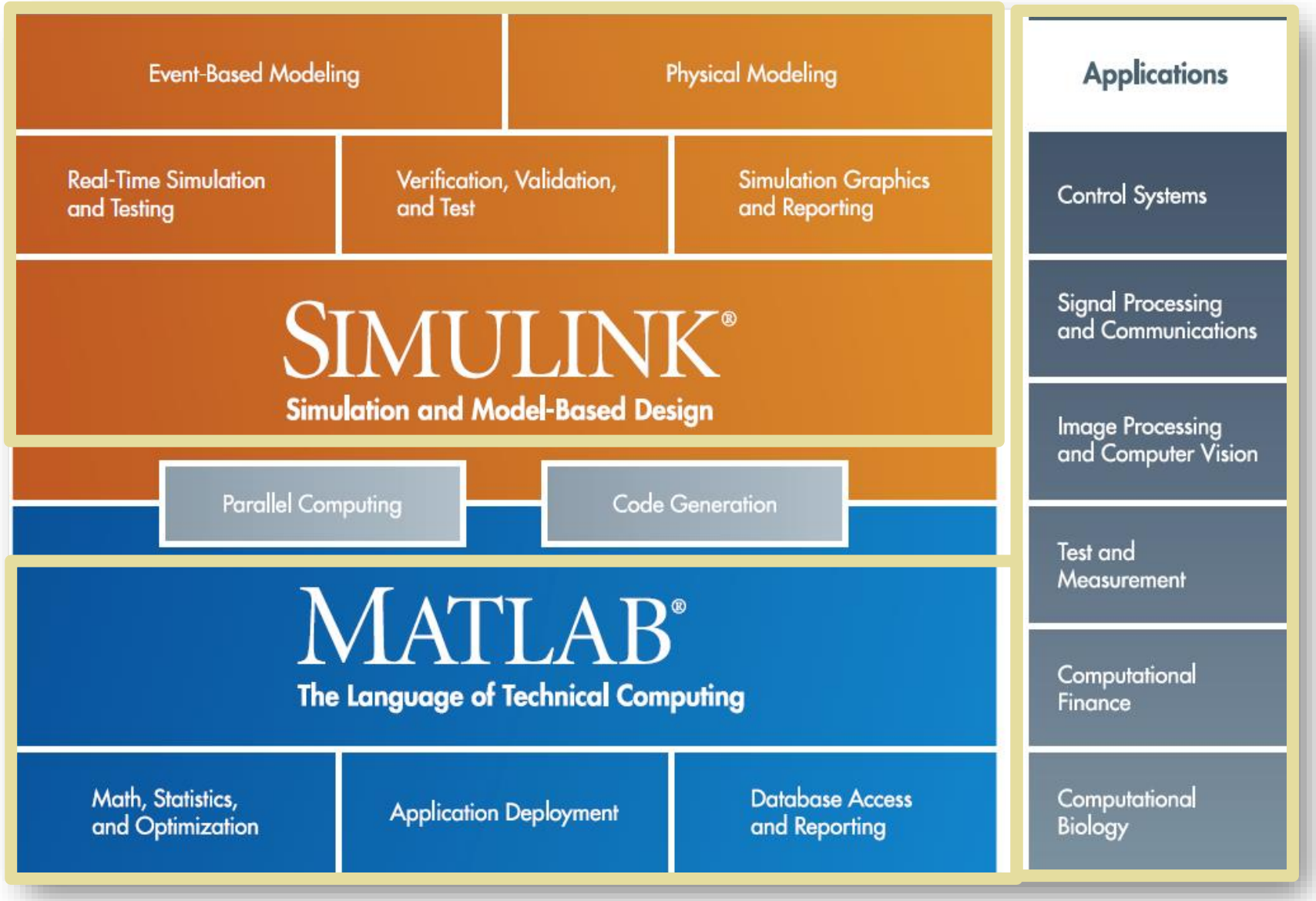
Review Software | Product Report 15+ pages

MATLAB® is the enterprise engineering platform for AI.

- Empower your team, including those with limited AI or data science experience
- Apply complete workflows for data preparation, AI modeling, system design, and production
- Deploy AI models on embedded devices, edge, enterprise systems, and the cloud
- Tackle integration challenges and reduce risk in designing AI-driven systems with Simulink®

*Top Machine Learning Software Awards 2022 | Software Reviews

MathWorks Product Overview



MathWorks offers nearly 119 products for technical computing and Model-Based Design. Widely used throughout industry, government and academia, these products are accelerating the pace of discovery, innovation, development, and learning in engineering and science

5

Our Solutions


Artificial Intelligence

Wireless Systems


Computational Biology


Image Processing & Computer Vision


Predictive Maintenance


Control Systems

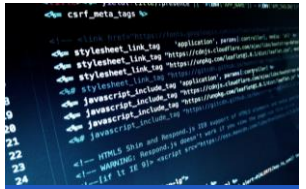

Machine Learning

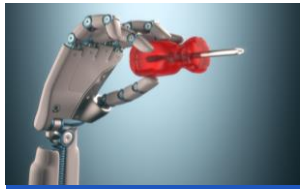

Robotics


Power Electronics Control Design

Internet of Things (IoT)


Power System Analysis & Design


Data Science


Mechatronics


Signal Processing


Automated Driving Systems


Deep Learning


Test & Measurement


Embedded Systems


FPGA, ASIC & SoC Development


Mixed Signal System

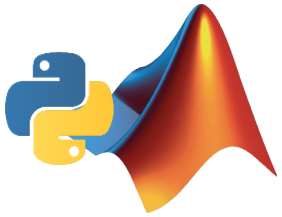

Enterprise & IT Systems

Our Locations

- Singapore (Headquarter)
- Malaysia
- Thailand
- Philippines
- Vietnam
- Indonesia
- Brunei

Contact us at

www.techsource-asia.com/contact-us



“Using MATLAB with Python“

Agenda (2hr)

- Introduction MATLAB and Python	09:00 - 09:30 (15min)
- Co-execution multiple languages	09:30 - 09:45 (15min)
- Call Python from MATLAB	09:45 - 10:00 (15min)
- Short Break	10:00 - 10:15 (15min)
- Call MATLAB from Python	10:15 - 10:35 (30min)
- Resources for further learning	10:35 - 10:50 (15min)
- Q&A	10:50 – 11:00 (10min)

What’s new in MATLAB

Teach with MATLAB Live Editor

MATLAB in an Executable Notebook

Use live scripts to create engaging lectures that combine explanatory text, mathematical equations, code, and results

Share live scripts directly with colleagues or students

Work in a single environment to eliminate context switching

Use data to tell a story – Live Scripts

```
% pulse of input DC current (stimulation parameters)
params.stm.pulseWidth = 0.8;
params.stm.pulseAmpl = 70;
params.mem.C = 140;          % membrane capacitance
params.mem.vr = -60;         % resting membrane potential
params.mem.vt = -52;         % instantaneous threshold potential
params.mem.a = 0.01;         % recovery time constant
params.mem.b = -0.9;         % determines whether u is amplifying (b<0) or resonar
params.mem.c = -40;          % voltage reset value
params.mem.d = 170;          % total outward-inward current during spike
params.mem.vpeak = 17;       % spike cutoff

% initialize/calculate some parameters
T = 1000; tau = 1; % total time of simulation and time step
n = round(T/tau); % simulation time steps
```

- MATLAB code and related output together
- Text, pictures, graphics and equations inline
- UI controls for users to select options
- Export to html, latex, word, pdf

```
    end
```

- Communicate your research (Lab meetings, collaborators)
- Allow others to interact with your code
- Upload code, documentation & results together

```
ylabel('membrane potential')
```

Spiking model of cortical neurons

This live script is an attempt to reproduce the neuronal dynamics of most cortical neurons by using the Izhikevich model of spiking neurons. It is biologically as plausible as the Hodgkin-Huxley model, yet computationally viable like the integrate-and-fire models. For more information, please see the original paper of Eugene Izhikevich [*IEEE Trans on Neural Networks, 14(6), 2003*] as well as the project report of Ahmed Salem from the University of Edinburgh, School of Informatics.

1 The spiking model

1.1 Spiking model equation of Izhikevich

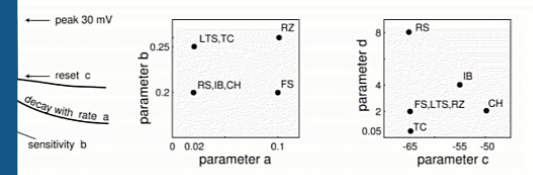
The spiking model of cortical neurons is based on the following equations

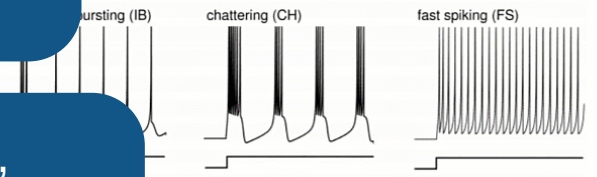
$C\dot{v} = k(v - v_r)(v - v_t) - u + I$ .....[1]

$\dot{u} = a\{b(v - v_r) - u\}$ .....[2]

If $v \geq v_{peak}$, then $v = c, u = u + d$ .....[3]

...the membrane potential, C is the membrane capacitance and u is the membrane recovery variable. v_r is the resting membrane potential and v_t is the instantaneous spiking ...arge diversity of cortical neurons can be simulated by setting the membrane parameters a, b, c & d appropriately (see figure below).





...s of the model

...rrent parameters

...hat is injected

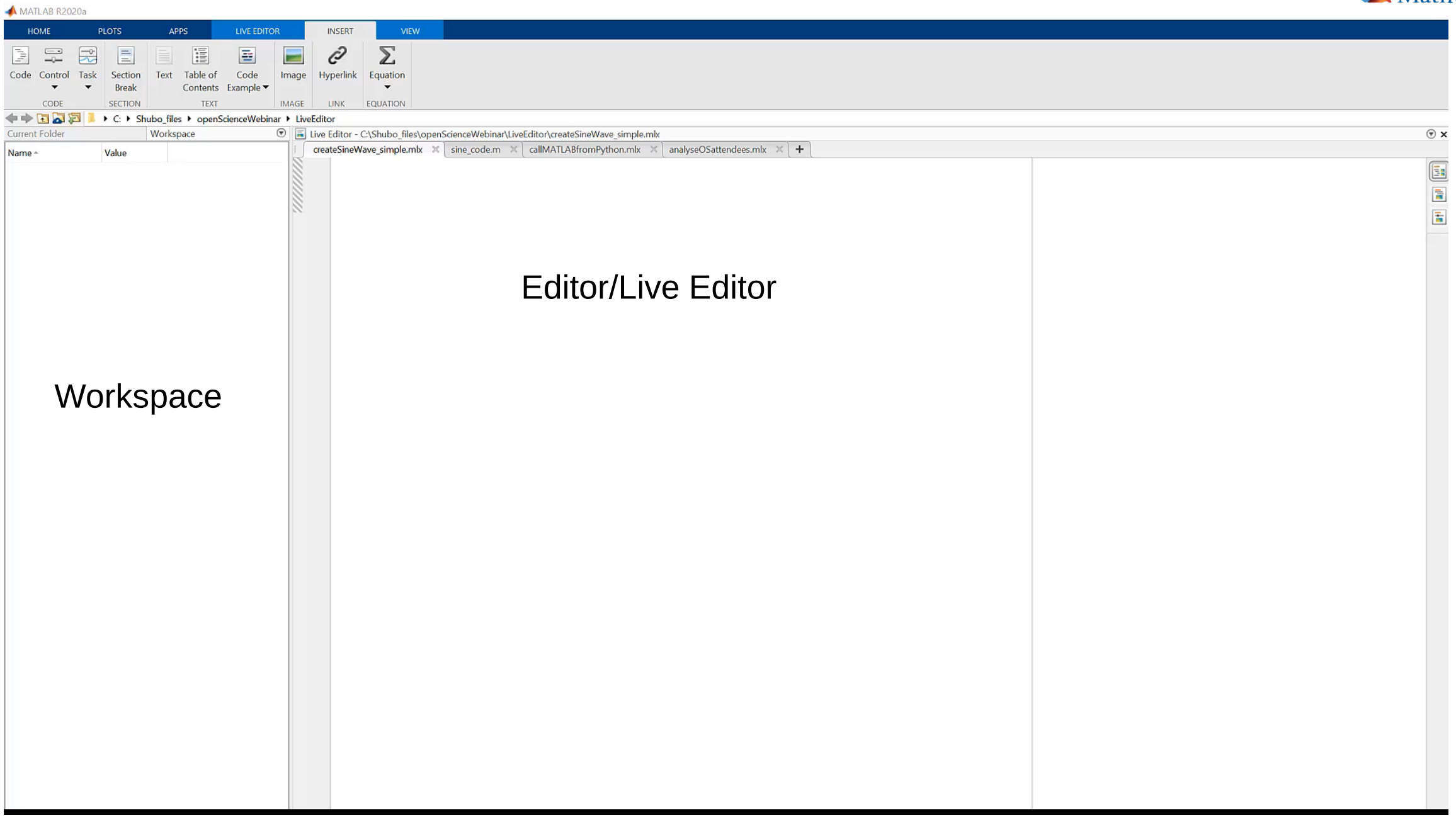
```
...idth =  0.7 ;
...mpl = 70;
```

...rameters

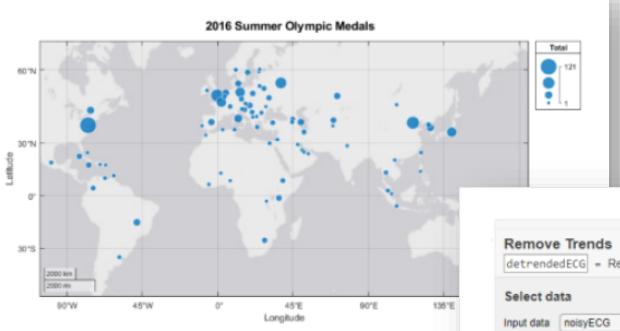
...eters of the model a, b, c & d control various membrane properties and affect spiking behavior. They are

- a = recovery time constant
- b = determines whether recovery variable, u, is an amplifying (b < 0) or a resonant (b > 0) variable.
- c = voltage reset value following spiking

mathworks.com/products/matlab/live-editor.html


MATLAB R2020a
HOME
PLOTS
APPS
LIVE EDITOR
INSERT
VIEW
Code
Control
Task
Section Break
Text
Table of Contents
Code Example
Image
Hyperlink
Equation
CODE
SECTION
TEXT
IMAGE
LINK
EQUATION
C: ▸ Shubo_files ▸ openScienceWebinar ▸ LiveEditor
Current Folder
Workspace
Name
Value
Live Editor - C:\Shubo_files\openScienceWebinar\LiveEditor\createSineWave_simple.mlx
createSineWave_simple.mlx
sine_code.m
callMATLABfromPython.mlx
analyseOSattendees.mlx
Workspace
Editor/Live Editor

The Live Script Gallery

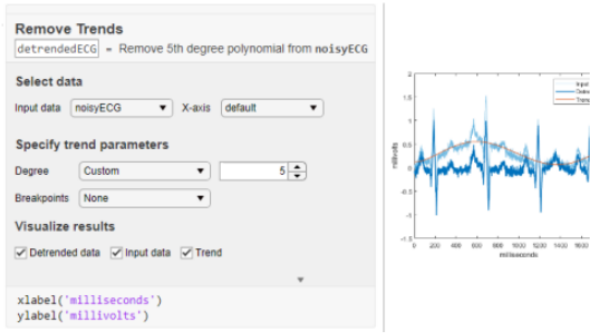


GDP and Olympic Success

by David Garrison

Run this example

View on File Exchange

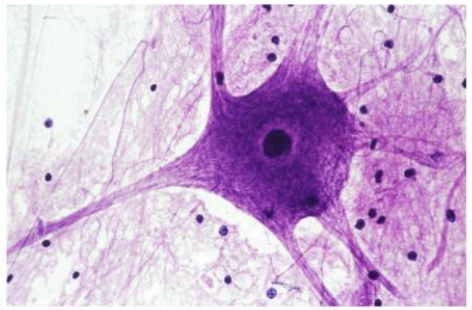


Calculate Heart Rate from Electrocardiogram Data

by Heather Gorr

Run this example

View on File Exchange

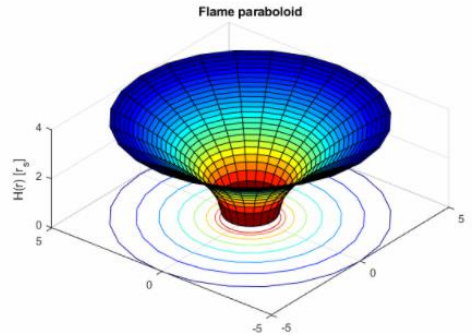


Inside the Brain: Modelling the Neuron

by Davide Borra

Run this example

View on File Exchange



Schwarzschild metric and black holes (Computational example)

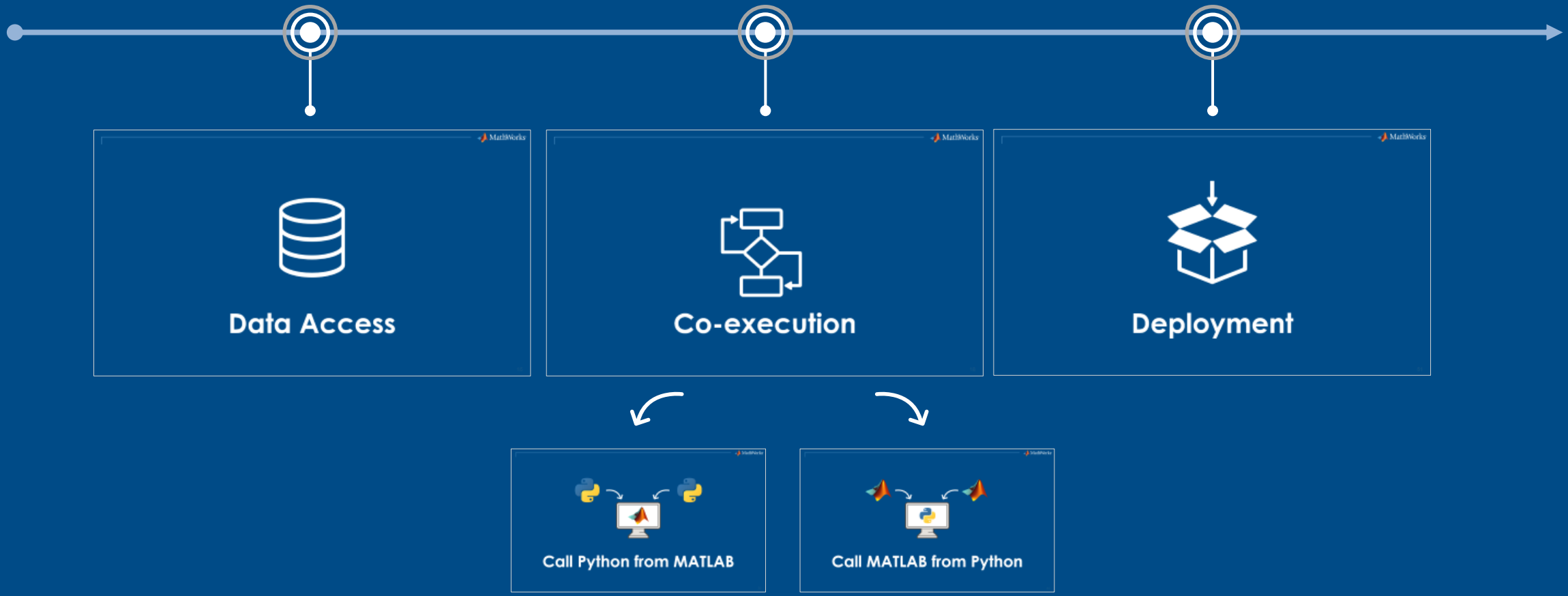
by Kurt Braeuer

Run this example

View on File Exchange

https://www.mathworks.com/products/matlab/live-script-gallery.html

Plan


Data Access
Co-execution
Deployment
Call Python from MATLAB
Call MATLAB from Python

Data Access

Setup tips

- Use **`pyenv`** to update settings (**`pyversion`** before R2019b)
- Ensure all code is on path
- Check environment settings, depending on how you set up python

- How to select the MATLAB engine version if I have multiple MATLAB releases to interface?

Setup

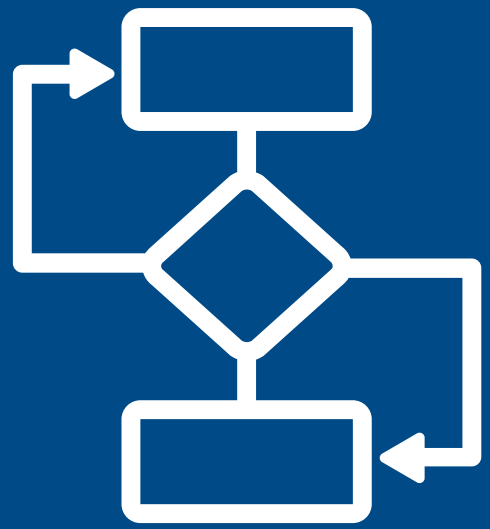
Ensure Python is installed and confirm the version with `pyenv`.

We will provide the path to the correct executable for MATLAB Online.

```
pyenv
```

```
ans =
  PythonEnvironment with properties:

         Version: "3.9"
      Executable: "C:\Users\KantikaWongkasem\anaconda3\python.EXE"
         Library: "C:\Users\KantikaWongkasem\anaconda3\python39.dll"
            Home: "C:\Users\KantikaWongkasem\anaconda3"
          Status: Loaded
   ExecutionMode: OutOfProcess
       ProcessID: "4668"
     ProcessName: "MATLABPyHost"
```



Co-execution

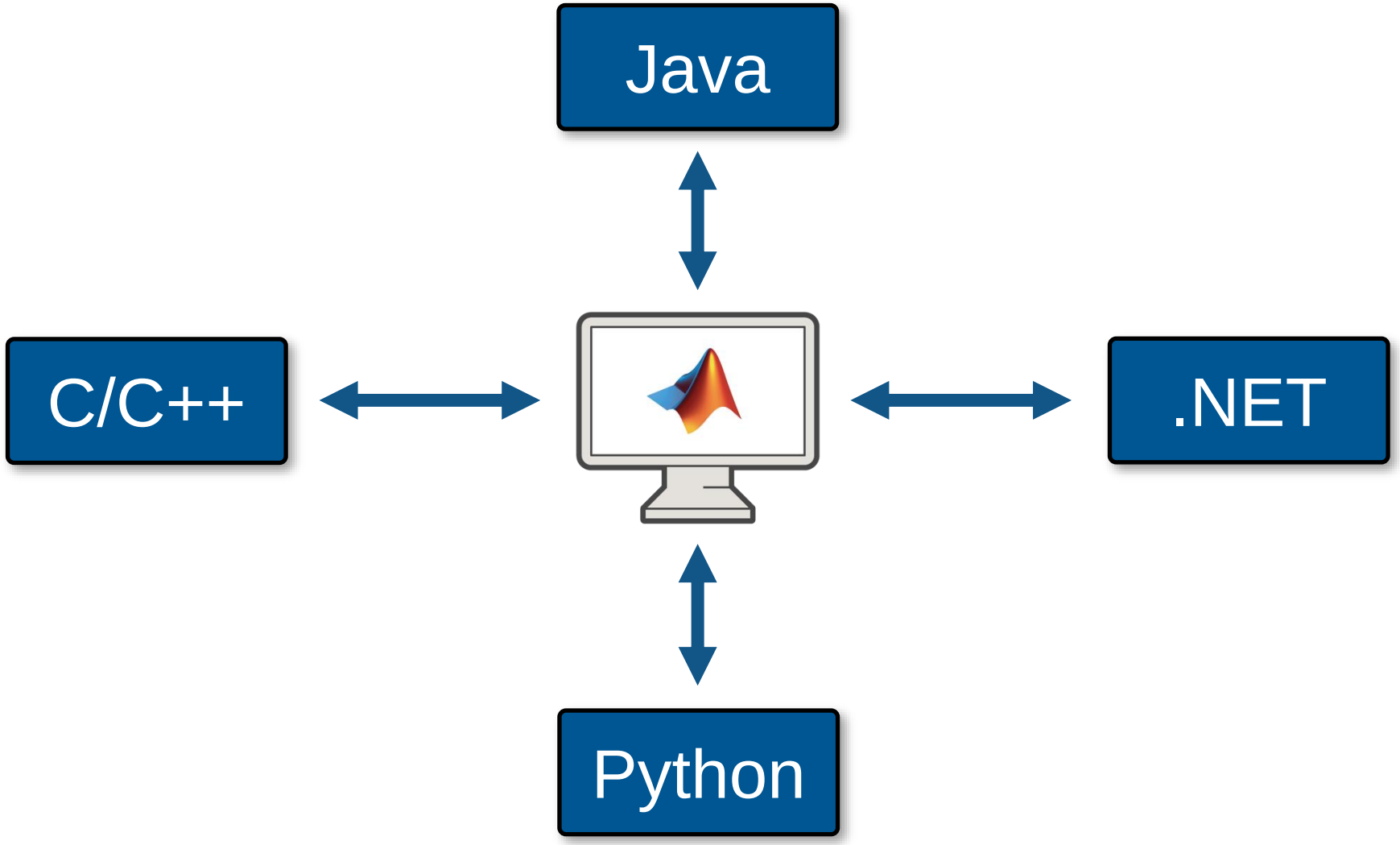
Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

MATLAB provides flexible integration with multiple languages



Given: Existing Python Code accessing & preparing weather data

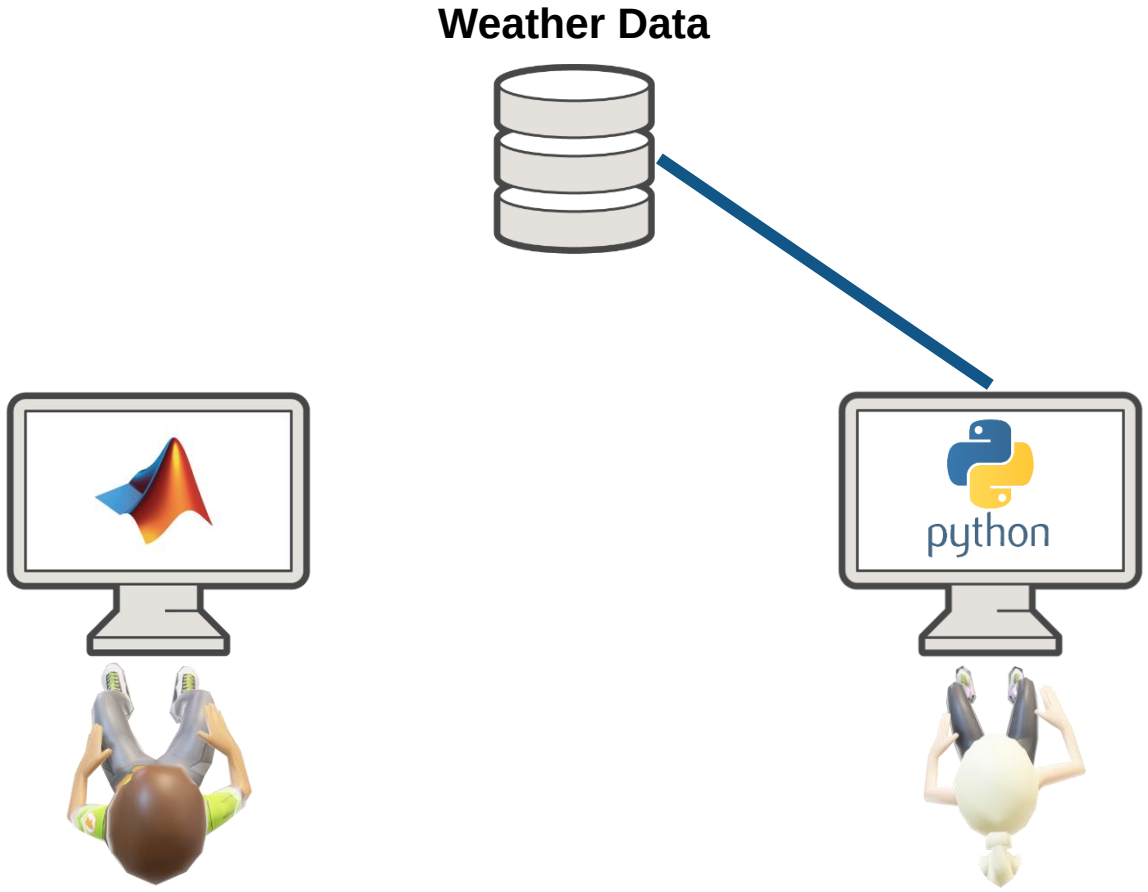
Data preparation | Modeling | Deployment

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Call Python from MATLAB

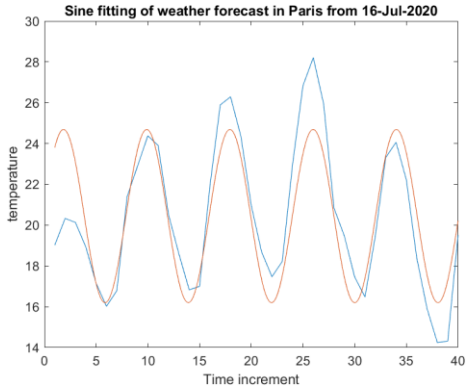
Data preparation | Modeling | Deployment

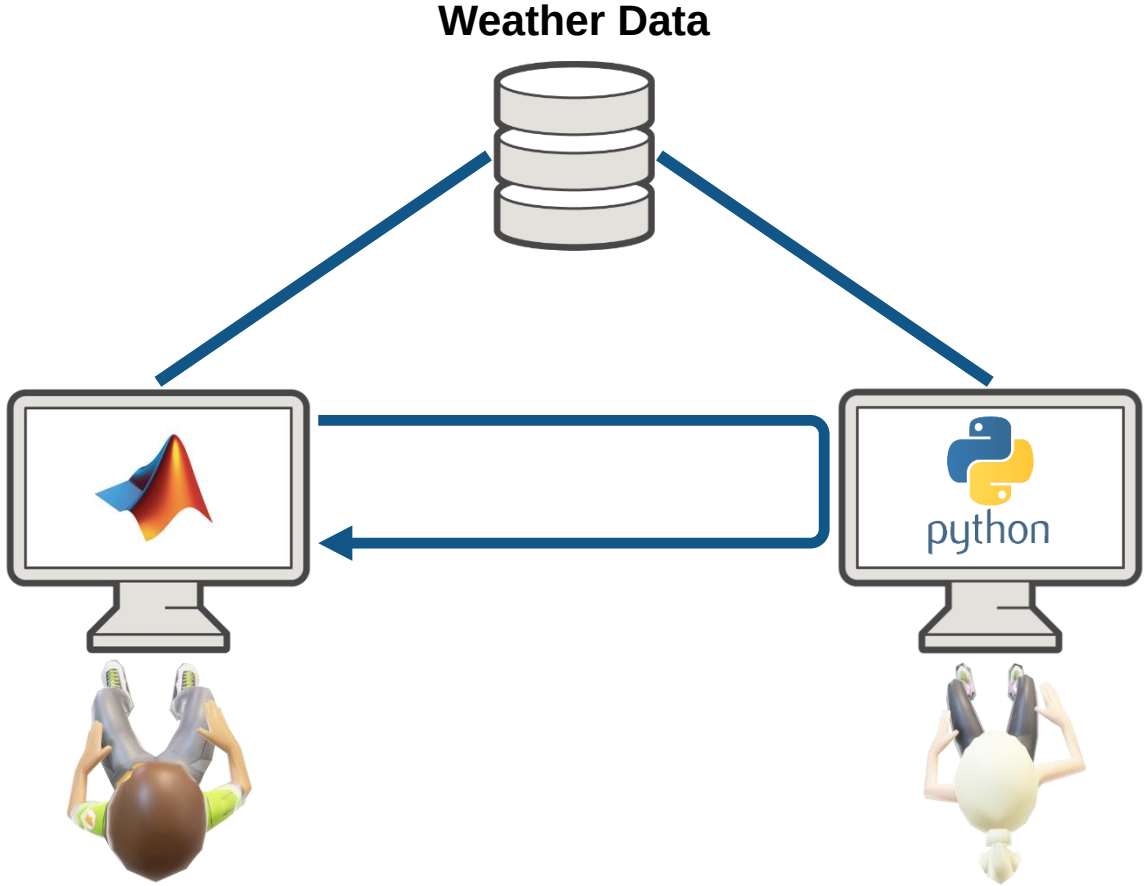
Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment





Call MATLAB from Python

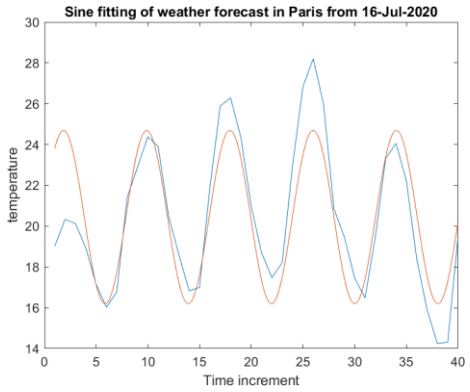
Data preparation → Modeling → Deployment

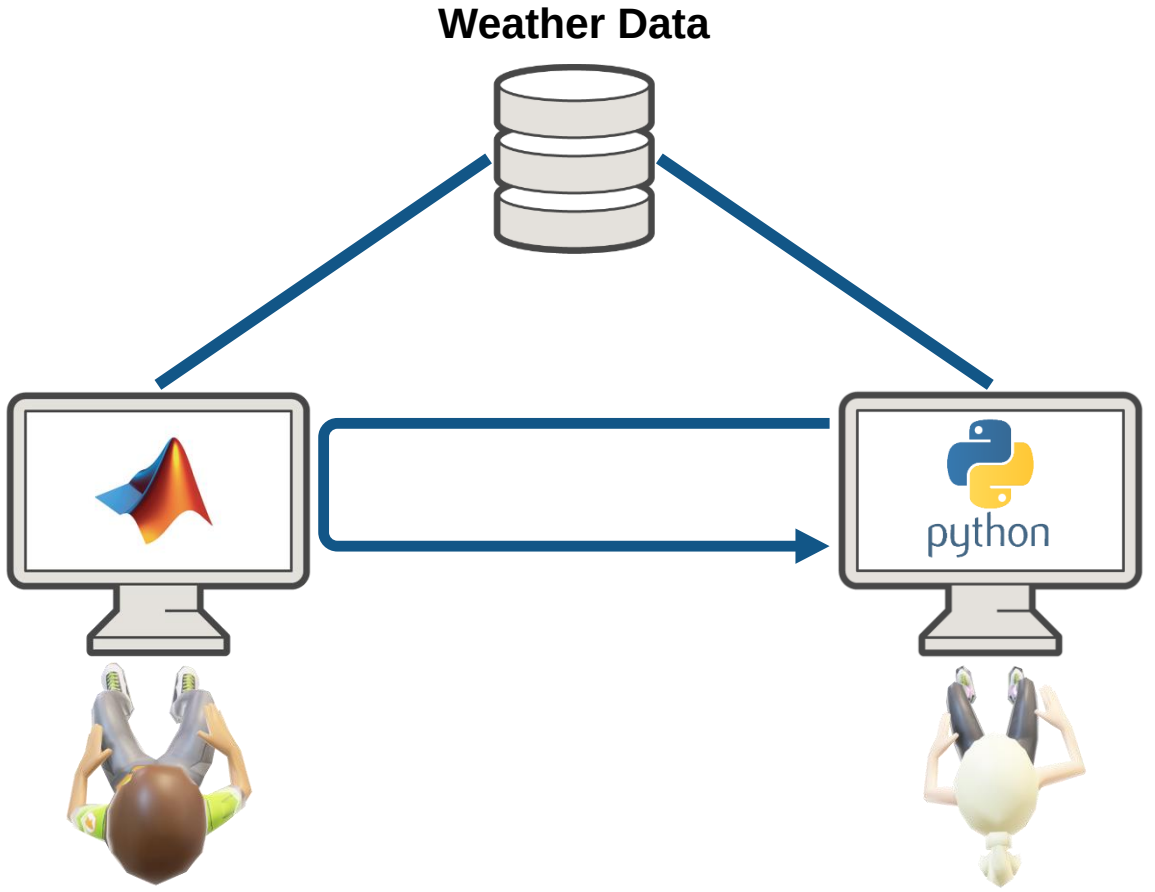
Data Access

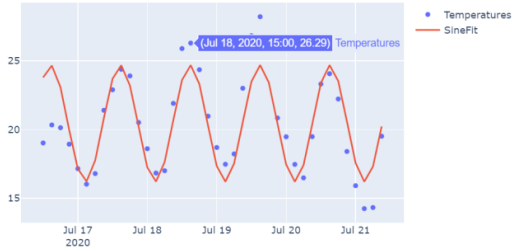
Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment







Deploy:
MATLAB Analytics into Python

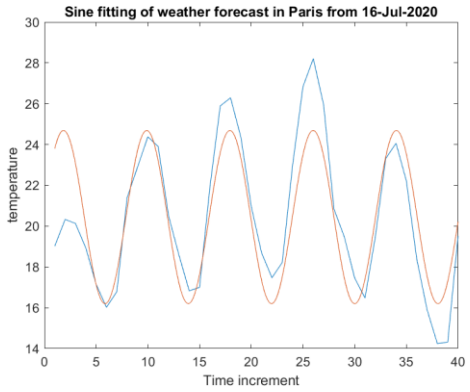
Data preparation → Modeling → Deployment

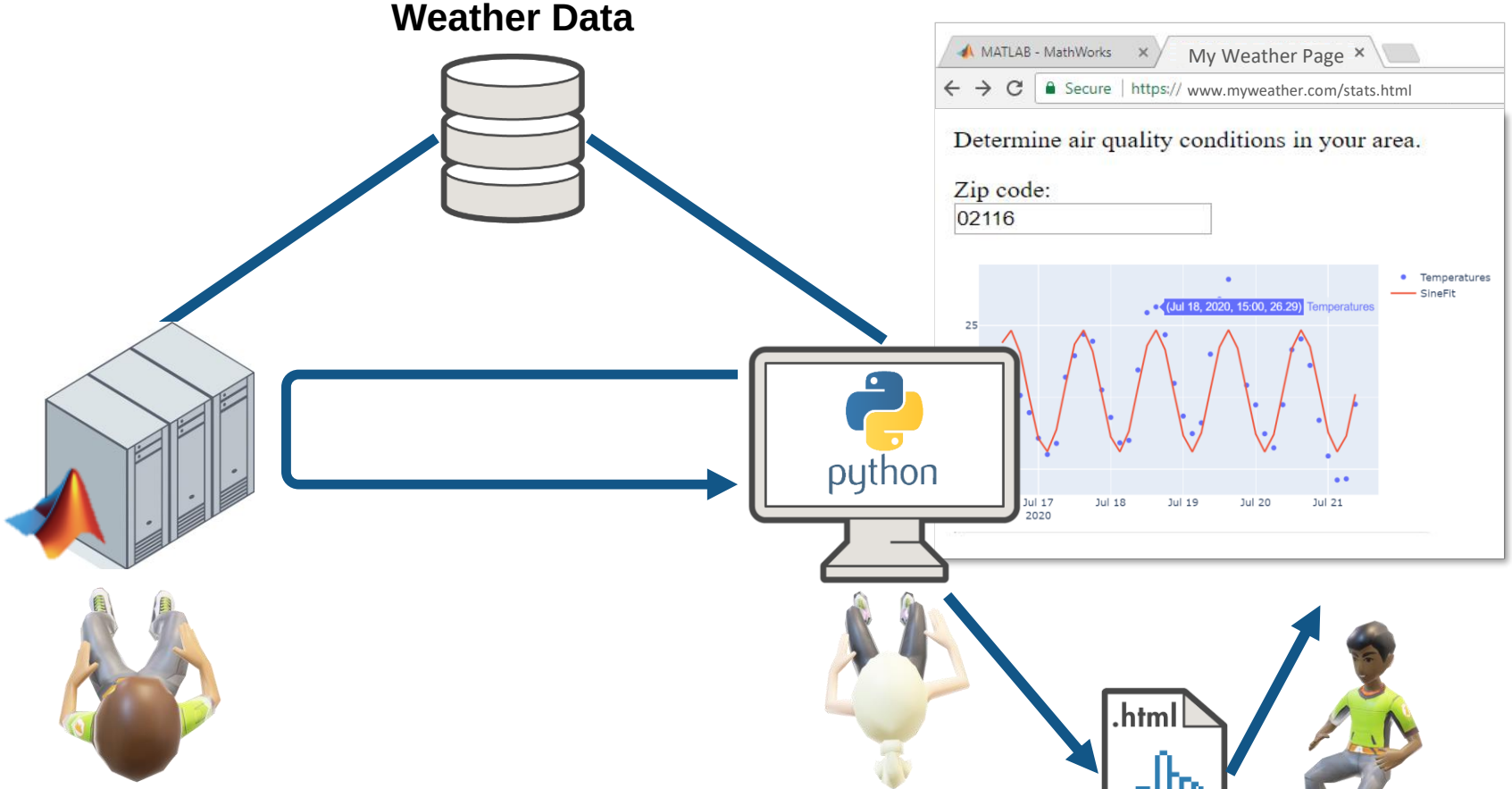
Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment





Deploy:
MATLAB Analytics into Python

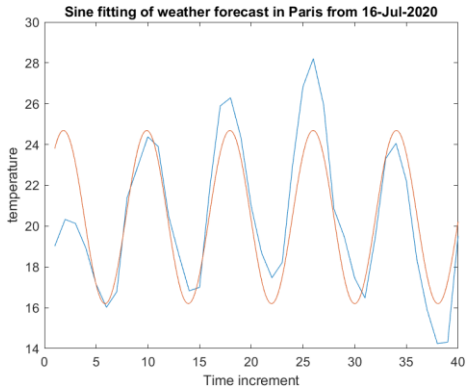
Data preparation → Modeling → Deployment

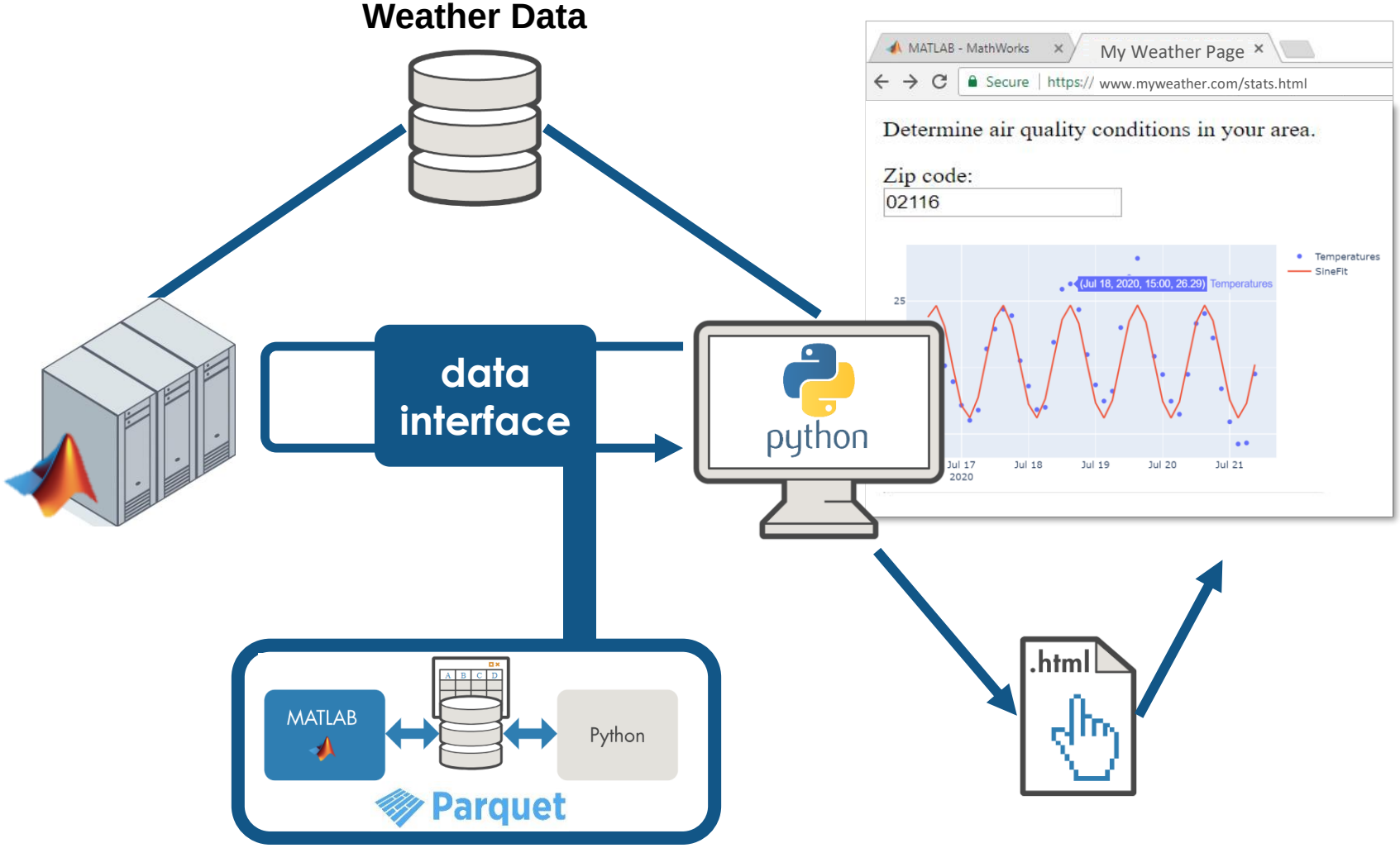
Data Access

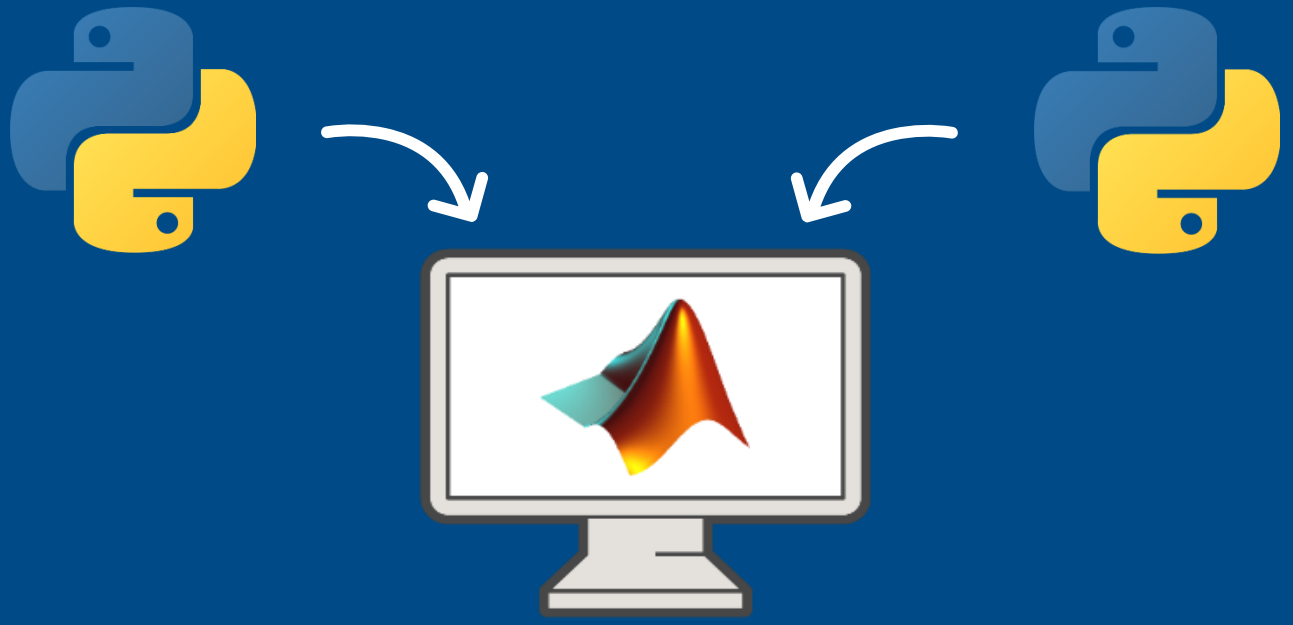
Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment







Call Python from MATLAB

Create List, Tuple, and Dictionary Types

- *This table shows the commands for creating `list, tuple, dict` types. The commands on the left are run from the Python interpreter. The commands on the right are MATLAB commands*

Python list — []	**MATLAB py.list**
>>> [‘Kantika’, ‘Wongkasem’, ‘Fah’]	>> py.list({‘Kantika’,’Wongkasem’,’Joseph’})
>>> [[1,2],[3,4]]	>> py.list({py.list([1,2]),py.list([3,4])})

Python tuple — ()	**MATLAB py.tuple**
>>> (‘Kantika’, 27, ‘Engineer’)	>> py.tuple({‘Kantika’,19,’Engineer’})

Python dict — {}	**MATLAB py.dict**
>>> {‘Ink’: 424 , ‘Pink’: 543 , ‘Blue’: 320}	>>py.dict(pyargs(‘Ink’,424,’Pink’,543,’Blue’,320))
	For information about passing keyword arguments, see ***`pyargs`***

Access Python Standard Library Modules in MATLAB

- Create a Python `list` datatype:

```
l = py.list({'Kantika','Wongka','Fah'})
```

```
l =
  Python list with values:

    ['Kantika', 'Wongka', 'Fah']

    Use string, double or cell function to convert to a MATLAB array.
```

```
l.append('Engineer')
l
```

```
l =
  Python list with values:

    ['Kantika', 'Wongka', 'Fah', 'Engineer']

    Use string, double or cell function to convert to a MATLAB array.
```

- MATLAB recognizes Python objects and automatically converts the MATLAB cell array to the appropriate Python type.

- To convert the list variable to a MATLAB variable, call `cell` on the list and `string` on the elements of the list

```
myl = cellfun(@string,cell(l))
```

```
myl = 1×4 string
   "Kantika"    "Wongka"    "Fah"       "Engineer"
```

```
cell(l)
```

```
ans = 1×4 cell
```

	1	2	3	4
1	1×7 str	1×6 str	1×3 str	1×8 str

Call User-Defined Python Module(1)

This example shows how to call methods from following Python module:

Editor - D:\DriveD\Kantika W\Edu\Python Webinar\MATLABwithPythonWorkshopExpo\mymod.py

```
# mymod.py
"""Python module demonstrates passing MATLAB types to Python functions"""
def search(words):
    """Return list of words containing 'ka'"""
    newlist = [w for w in words if 'ka' in w]
    return newlist

def engineer(words):
    """Append 'Engineer' to list of words"""
    words.append('Engineer')
    return words
```

Command Window

New to MATLAB? See resources for Getting Started.

```
>> l = py.list({'Kantika','Kittika','Fah'})

l =

  Python list with values:

    ['Kantika', 'Kittika', 'Fah']

    Use string, double or cell function to convert to a MATLAB array.

>> py.mymod.search(l)

ans =

  Python list with values:

    ['Kantika', 'Kittika']

    Use string, double or cell function to convert to a MATLAB array.
```

- ✓ How to use call mymod.py and To convert the list variable to a MATLAB variable

Command Window

New to MATLAB? See resources for Getting Started.

```
>> py.mymod.engineer(l)

ans =

  Python list with values:

    ['Kantika', 'Kittika', 'Fah', 'Engineer']

    Use string, double or cell function to convert to a MATLAB array.

>> cellfun(@string,cell(l))

ans =

  1×4 string array

    "Kantika"    "Kittika"    "Fah"    "Engineer"

>> cell(l)

ans =

  1×4 cell array

    {1×7 py.str}    {1×7 py.str}    {1×3 py.str}    {1×8 py.str}
```

Display Python Documentation in MATLAB

You can display help text for Python functions in MATLAB. For example

```
>> py.help('list.append')
Help on method_descriptor in list:

list.append = append(self, object, /)
    Append object to the end of the list.
```

- Tab completion when typing `py.` Does not display available Python functionality. For more information,
- see Limitations to Python Support - MATLAB & Simulink (mathworks.com)

Call User-Defined Python Module(2)

If you create `mymod.py` in a Python editor, be sure that the module is on the Python search path

- From the MATLAB command prompt, add the current folder to the Python search path: `>> py.sys.path()`

```
if count(py.sys.path,pwd()) == 0
    insert(py.sys.path,int32(0),pwd());
end
```

- To learn how to call the function, read the function signature for the search function in the `mymod.py` source file. The function takes one input argument, `words`.

```
def search(words):
```

Reload Modified User-Defined Python Module

- **Create Python Module**

  Save the file `mymod.py`:

```
def myfunc():
    return 'version 1'
```

- **Modify Module**

  Modify the function:

```
    return 'version 2'
```

- **Unload Module**

```
>> clear classes
```

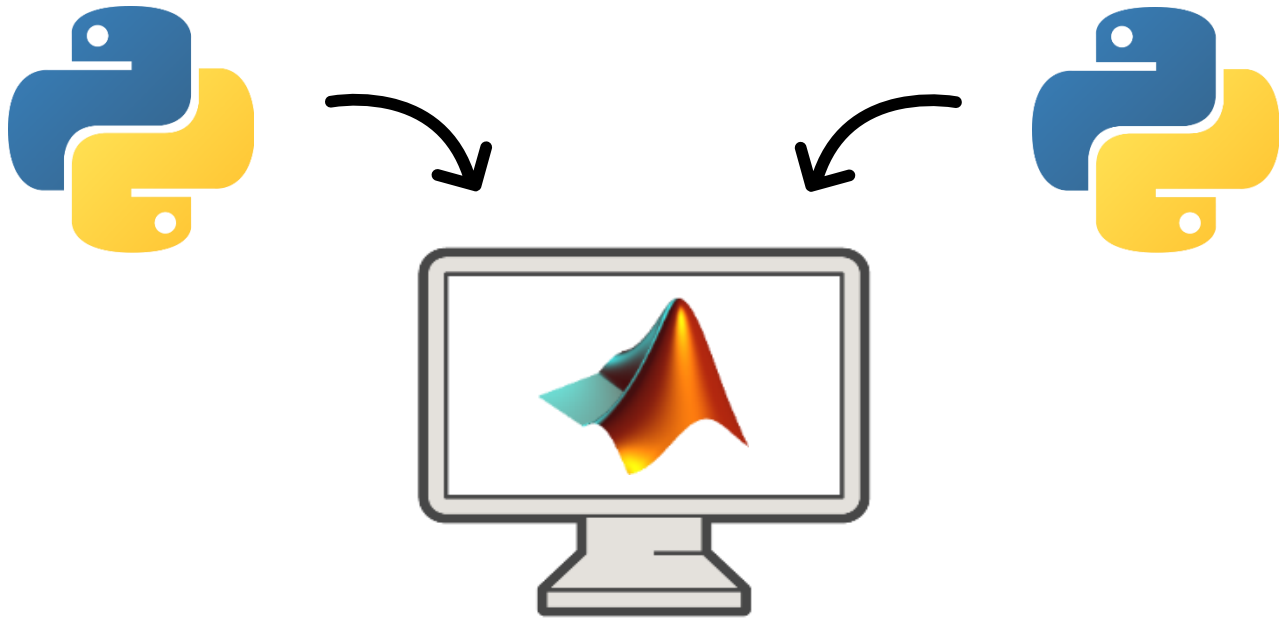
- **Import & Reload Modified Module**

```
>> mod = py.importlib.import_module('mymod');
>> py.importlib.reload(mod);
```

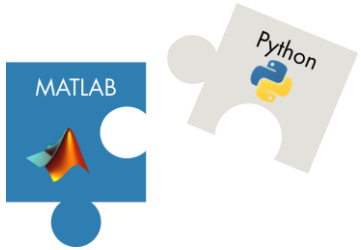
Why Call Python from MATLAB?

Already working in MATLAB, and:

- Want to reuse existing Python code
- Need functionality that is only available in Python



Calling Python libraries from MATLAB



Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

Use the weather.py module to get the air quality for Paris. This is a user-defined Python module which includes functions to read and parse the current and forecasted weather data by location.

```
jsonData = py.weather.get_current_weather("Paris","France",apikey.Key)
```

```
jsonData =
  Python dict with no properties.

    {'coord': {'lon': 2.35, 'lat': 48.85}, 'weather': [{'id': 803, 'main': 'Cloud
```

```
def get_current_weather(city, country, apikey):
    # get current conditions in specified location
    # get_current_weather('boston','us',key)
    import urllib.request
    import json
    # read current conditions
    try:
        url = "https://api.openweathermap.org/data/2.5/weather?q="+city+","+country+"&appid="+apikey
        response = urllib.request.urlopen(url)
        html = response.read()
        json_data = json.loads(html)

    except urllib.error.URLError:
        # if weather API doesnt work, read the file
        json_data = read_backup(city)

    return json_data
```

Parse the json data returned from the weather API.

The Python dictionary can be represented as a MATLAB struct.

```
weatherData = py.weather.parse_json(jsondata);
struct(weatherData)
```

```
ans = struct with fields:
            temp: 18.7100
      feels_like: 17.3000
        temp_min: 17.7800
        temp_max: [1×1 py.int]
```

Use a function (`prepData.m`) to prepare data for machine learning (create a table with the expected variable names, preprocessing steps, etc).

```
currentData = prepData(weatherData)
```

currentData = 1×12 table

	DateLocal	city	StateName	T	P	DP	RH	WindDir	WindSpd
1	01-Jul-2020 11:...	"Paris"	Ile de France	21.6200	20.2600	349.2200	1010	5.1000	73

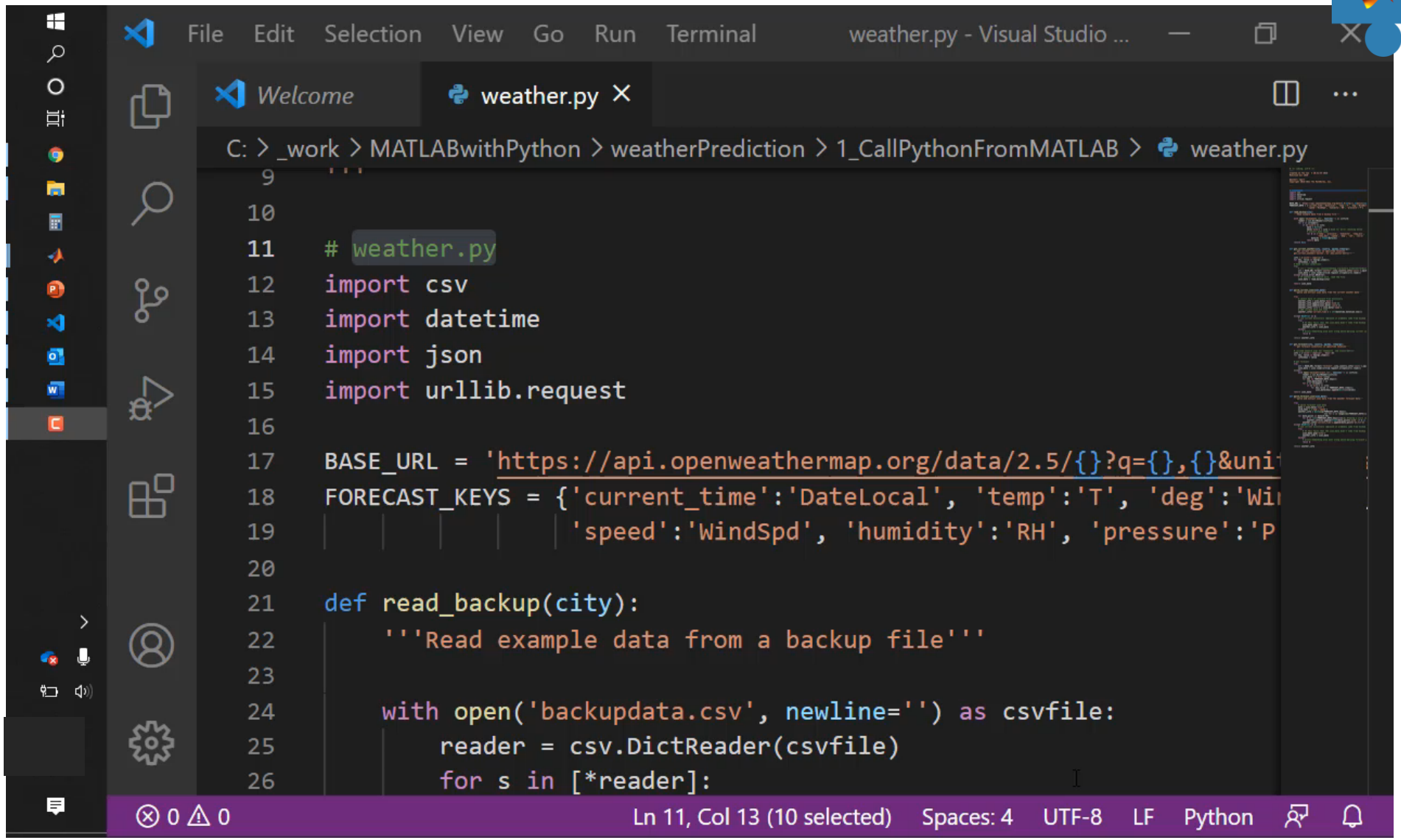
Calling Python libraries from MATLAB

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



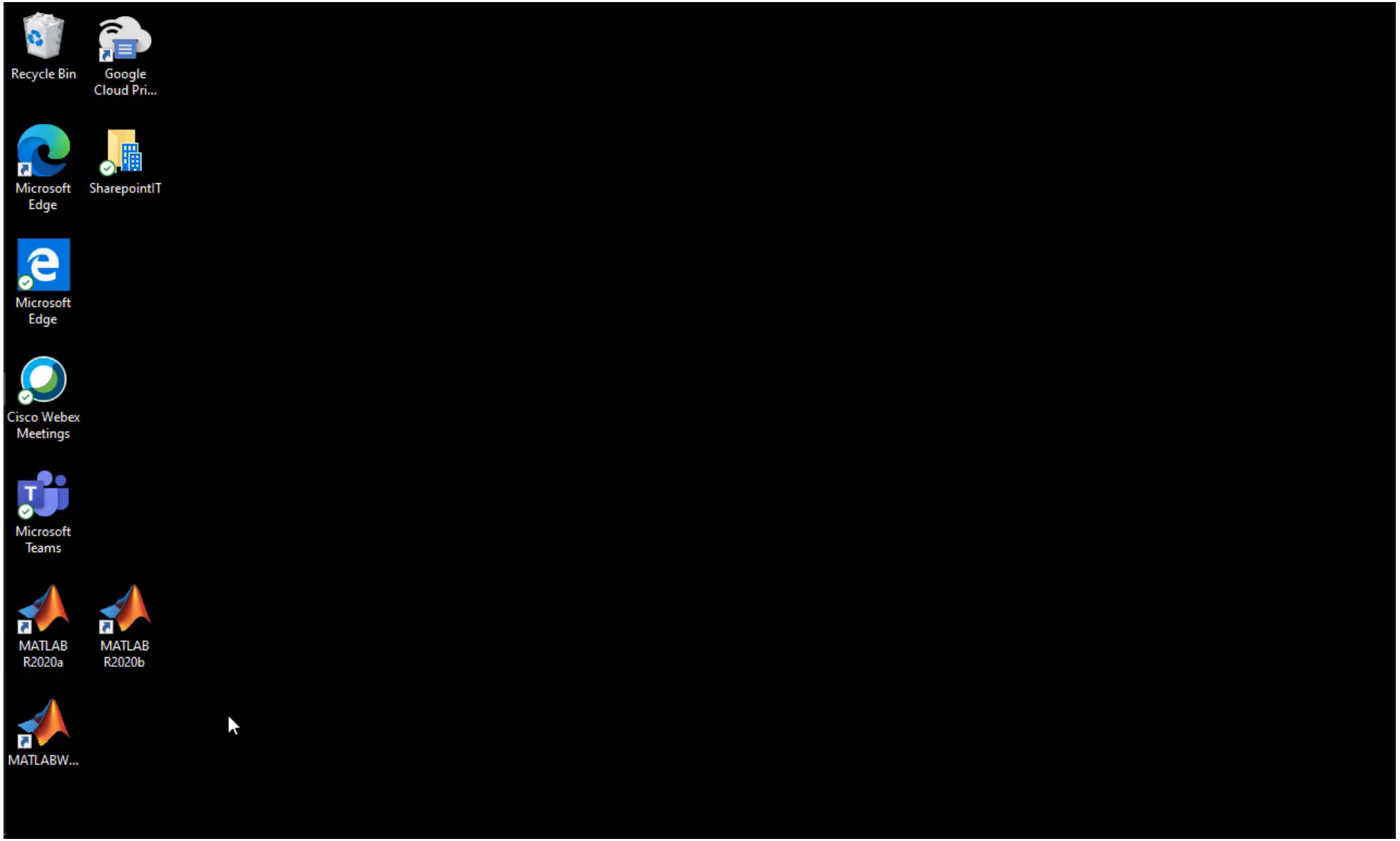
Connect to Python

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

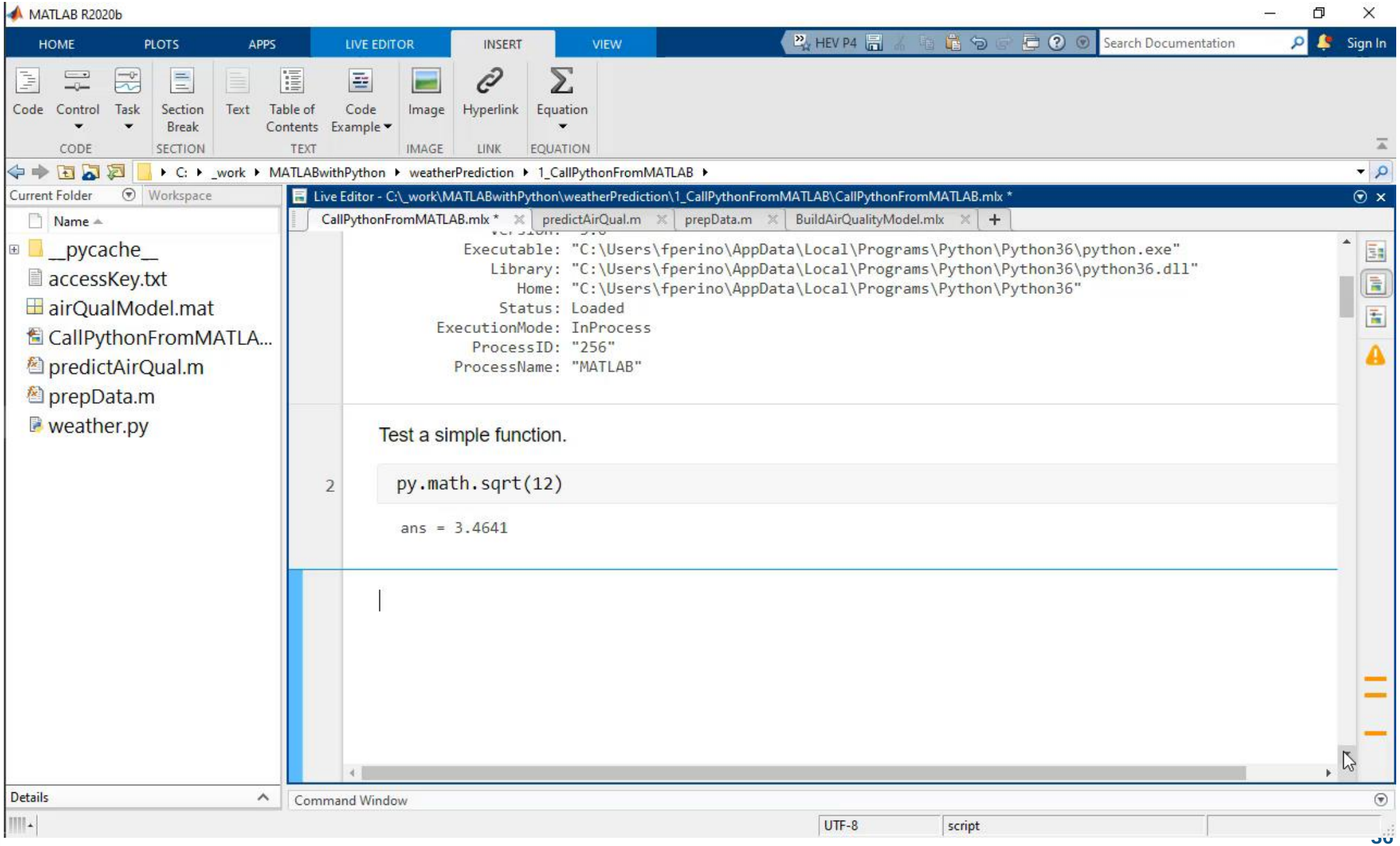


Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



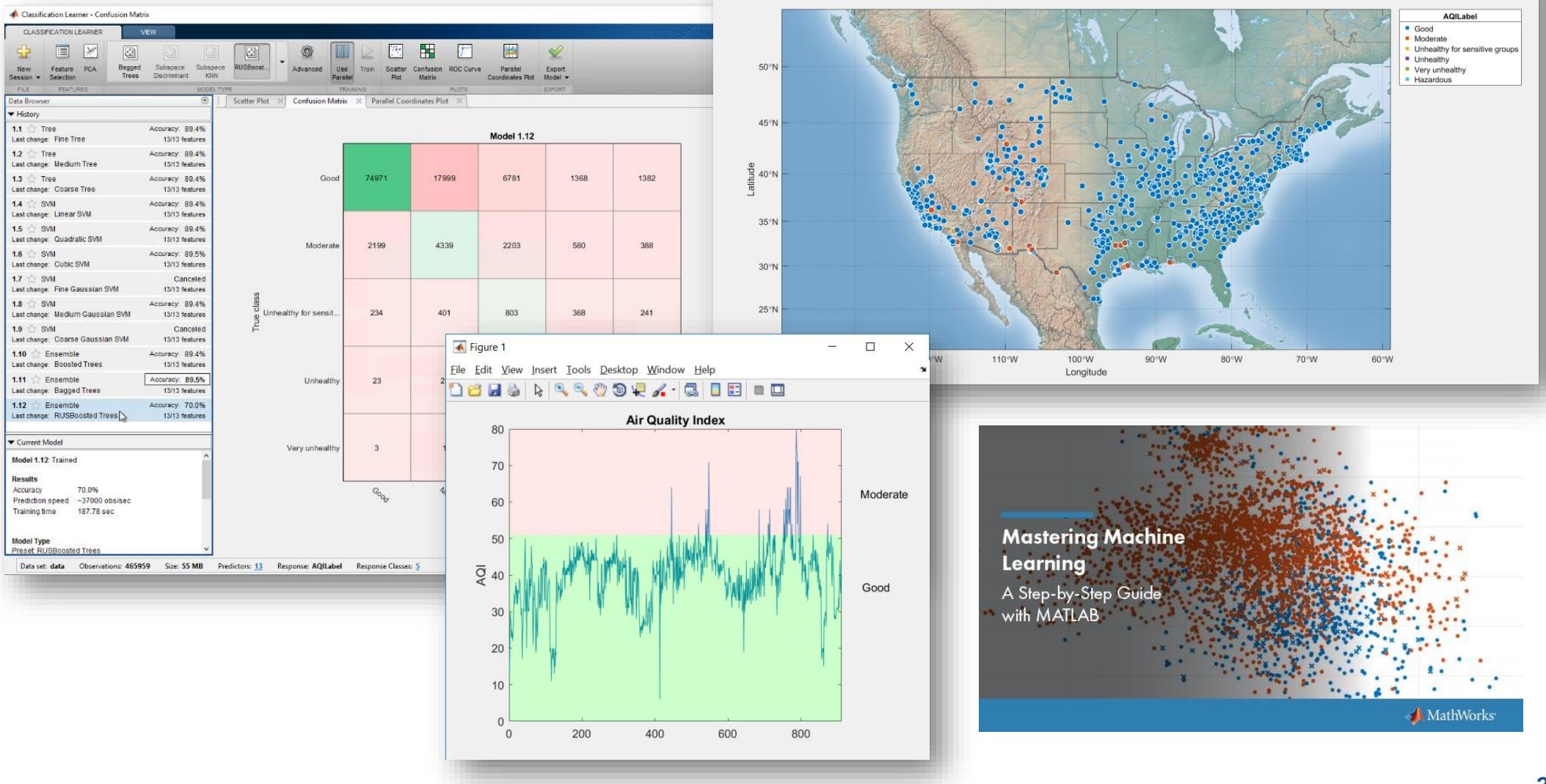
Air Quality Prediction Model

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



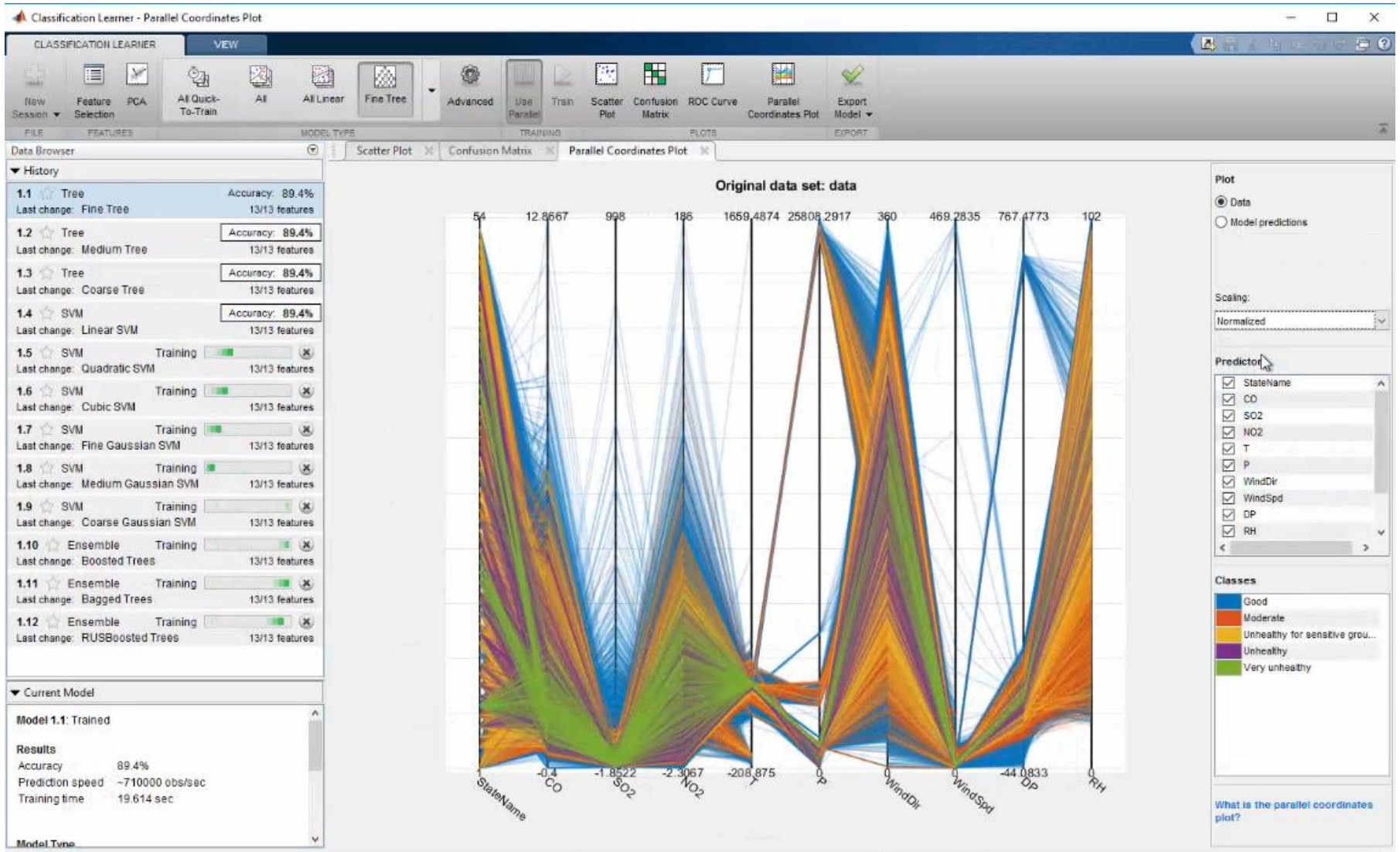
Train the Air Quality Prediction Model

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Model development is illustrated in Heather's webinar [MATLAB with Data Science]

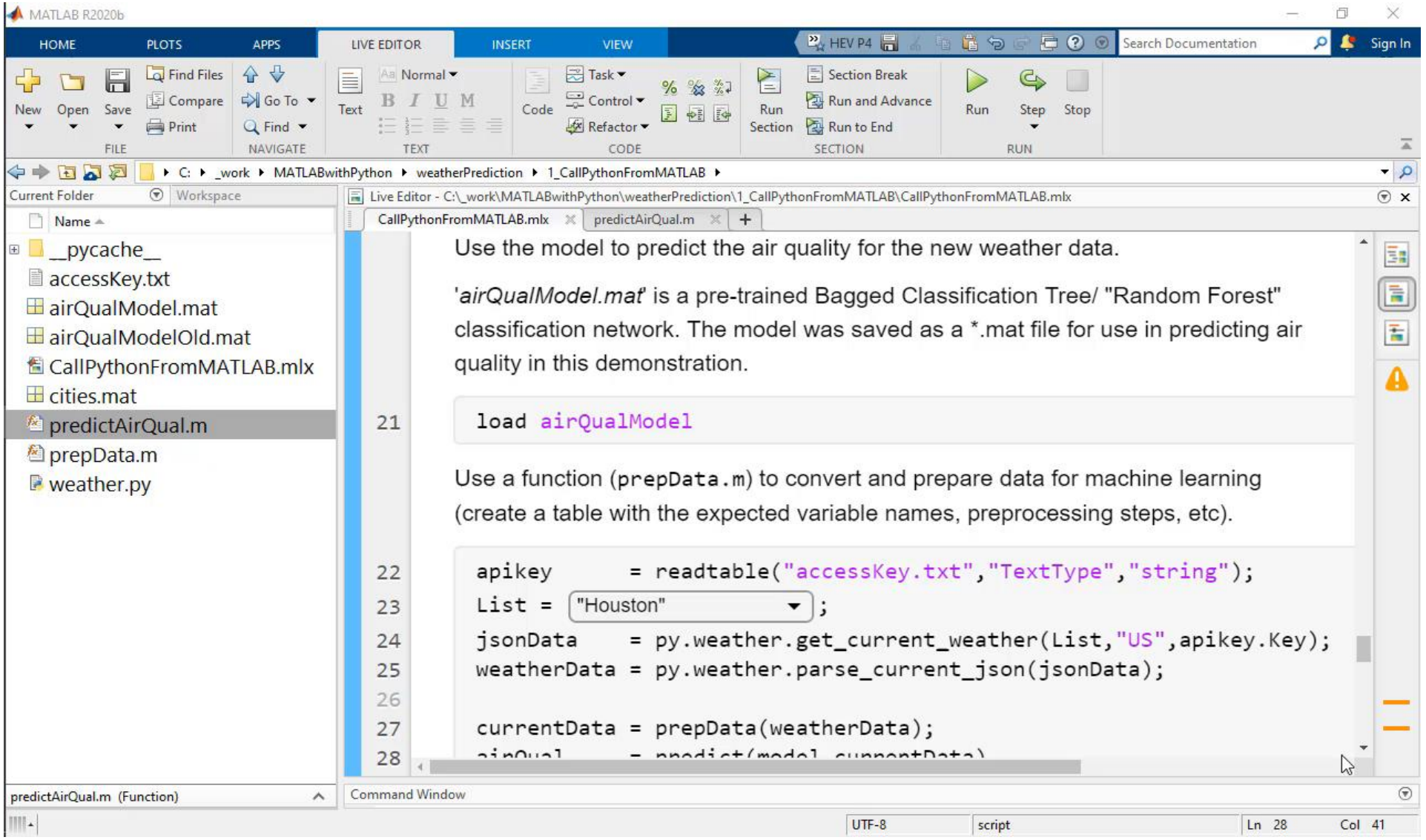
Call the Air Quality Prediction Model

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

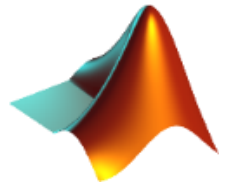
Deployment



Recap: Calling Python from MATLAB

Syntax differences when calling Python from MATLAB

Python



MATLAB

```
>>> import math
>>> math.sqrt(42)
```

```
>> py.math.sqrt(42)
```

```
>>> print('hello','world',sep=', ')
```

```
>> py.print('hello','world',...
       pyargs('sep',', '))
```

Data are automatically converted where possible

Otherwise convert explicitly

MATLAB to Python Data Type Mapping

R2021a

When calling a Python® function, MATLAB® converts MATLAB data into types that best represent the data to the Python language.

Pass Scalar Values to Python

MATLAB Input Argument Type — Scalar Values Only	Resulting Python `py.` Type	Examples
`double` `single`	`float`	Use Python Numeric Variables in MATLAB

Complex `single`
Complex `double`
	`complex`	`z = complex(1,2);`
		`py.cmath.polar(z)`
		
		`ans =`
		`Python tuple with no properties.`
		
		`(2.23606797749979, 1.1071487177940904)`

`int8`
`uint8`
`int16`
`uint16`
`int32`
	`int`	

`uint32`
`int64`
`uint64`
	`int`
	`long` (version 2.7 only)	

`NaN`	`float("nan")`	
`Inf`	`float("inf")`	
`string` scalar	`str`	Use Python str Variables in MATLAB

https://mathworks.com/help/matlab/matlab_external/passing-data-to-python.html

Data Access

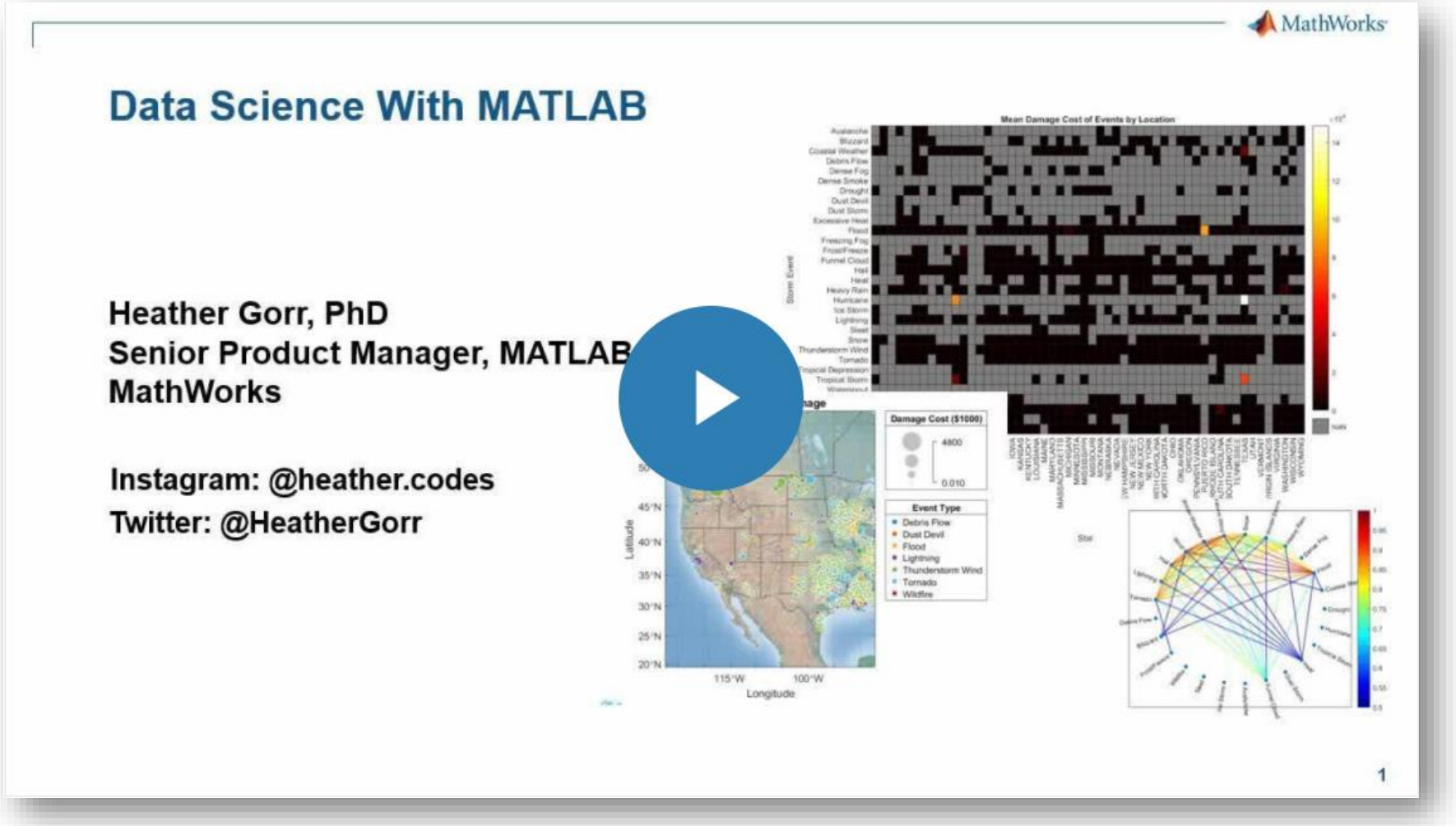
Co-Execution

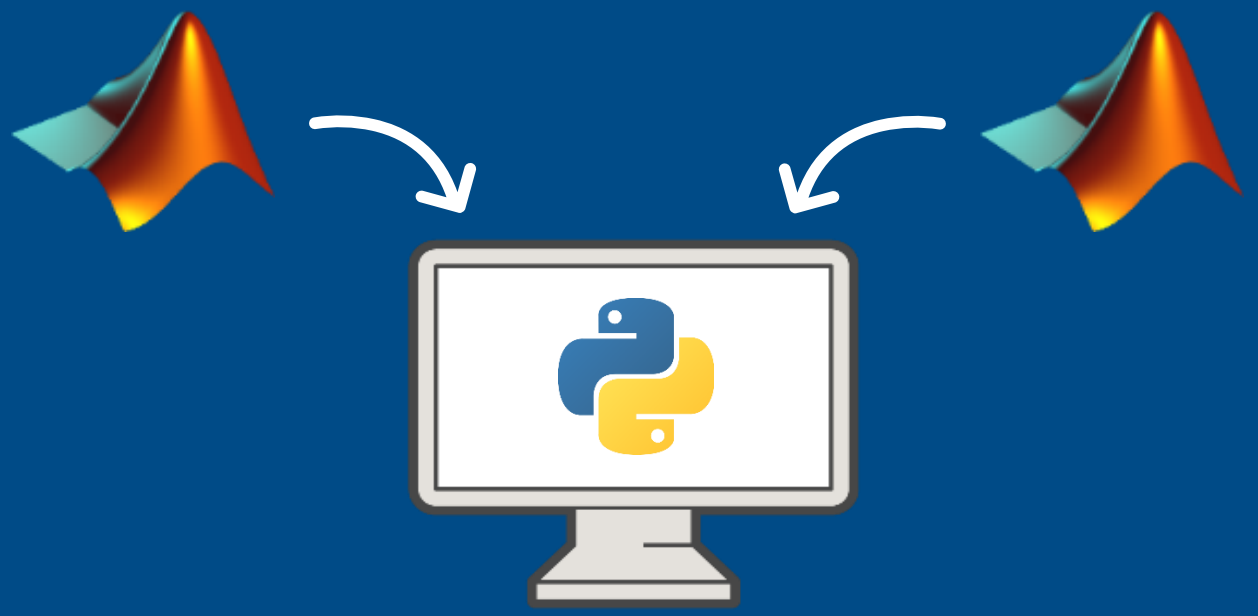
- Call Python from MATLAB
- Call MATLAB from Python

Deployment

Learn more about

Data Science with MATLAB



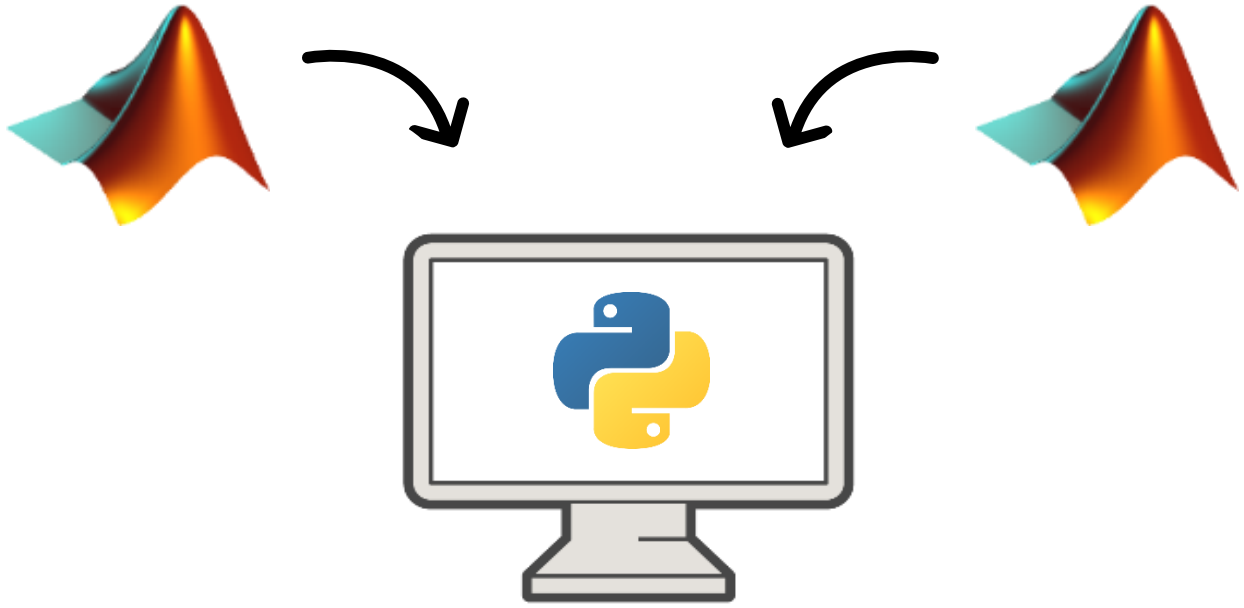


Call MATLAB from Python

Why call MATLAB from Python?

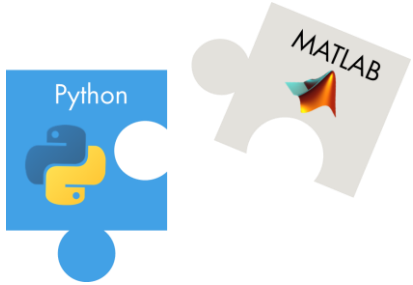
Already working in Python, and:

- Want to reuse existing MATLAB code
- Need functionality available in MATLAB
- Want to collaborate with MATLAB users



Call MATLAB from Python

MATLAB Engine API



- Install MATLAB Engine API for Python

```
$ cd "C:\Program Files\MATLAB\R2021b\extern\engines\python“
$ python setup.py install
```

- Start a MATLAB process

```
>>> import matlab.engine
>>> m = matlab.engine.start_matlab()
```

- Call MATLAB functions

```
>>> x = m.sqrt(float(42))
```

Integrate the MATLAB model in a Python environment

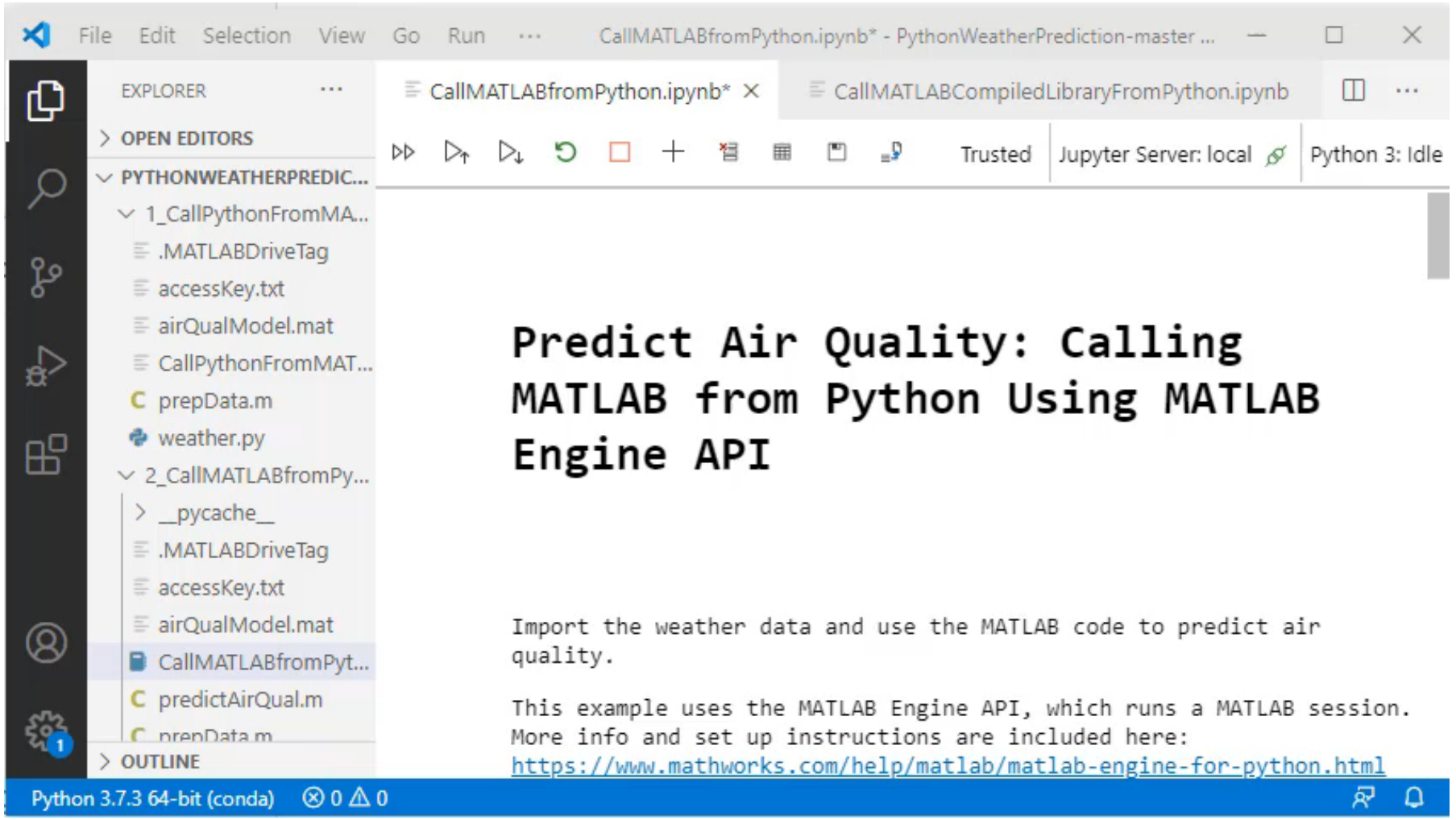
CallMATLABfromPython.ipynb

Data Access

Co-Execution

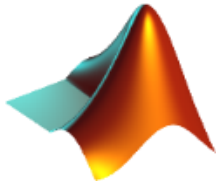
- Call Python from MATLAB
- **Call MATLAB from Python**

Deployment



Recap: Calling MATLAB from Python

Syntax differences when calling MATLAB from Python



MATLAB

Python

```
>> [s,sidx] = sort(x)
```

→

```
>>> s = eng.sort(x,nargout=2)
```

```
>> foo(x)
```

```
>>> eng.foo(x,nargout=0)
```

```
>> C = A + B
```

→

```
>>> C = eng.plus(A,B)
```

Data are automatically converted where possible

Data Access

Co-Execution

- Call Python from MATLAB
- **Call MATLAB from Python**

Deployment

Pass Data to MATLAB from Python

R2021a

Python Type to MATLAB Scalar Type Mapping

When you pass Python® data as input arguments to MATLAB® functions, the MATLAB Engine for Python converts the data into equivalent MATLAB data types.

Python Input Argument Type — Scalar Values Only	Resulting MATLAB Data Type
`float`	`double`
`complex`	Complex `double`
`int`	`int64`
`long` (Python 2.7 only)	`int64`
`float(nan)`	`NaN`
`float(inf)`	`Inf`
`bool`	`logical`
`str`	`char`
`unicode` (Python 2.7 only)	`char`
`dict`	Structure if all keys are strings not supported otherwise

Python Container to MATLAB Array Type Mapping

Python Input Argument Type — Container	Resulting MATLAB Data Type
`matlab` numeric array object (see MATLAB Arrays as Python Variables)	Numeric array
`bytearray`	`uint8` array
`bytes` (Python 3.x) `bytes` (Python 2.7)	`uint8` array `char` array
`list`	Cell array
`set`	Cell array
`tuple`	Cell array

https://mathworks.com/help/matlab/matlab_external/pass-data-to-matlab-from-python.html

Deployment

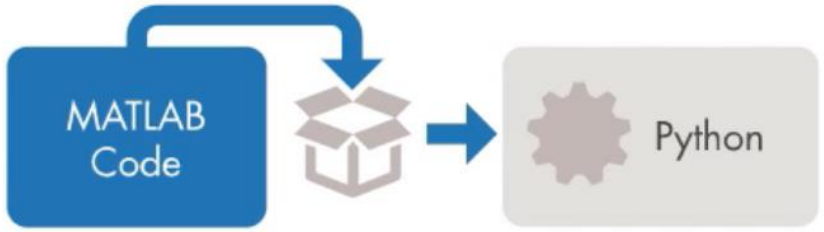
Generate Python library from MATLAB functions

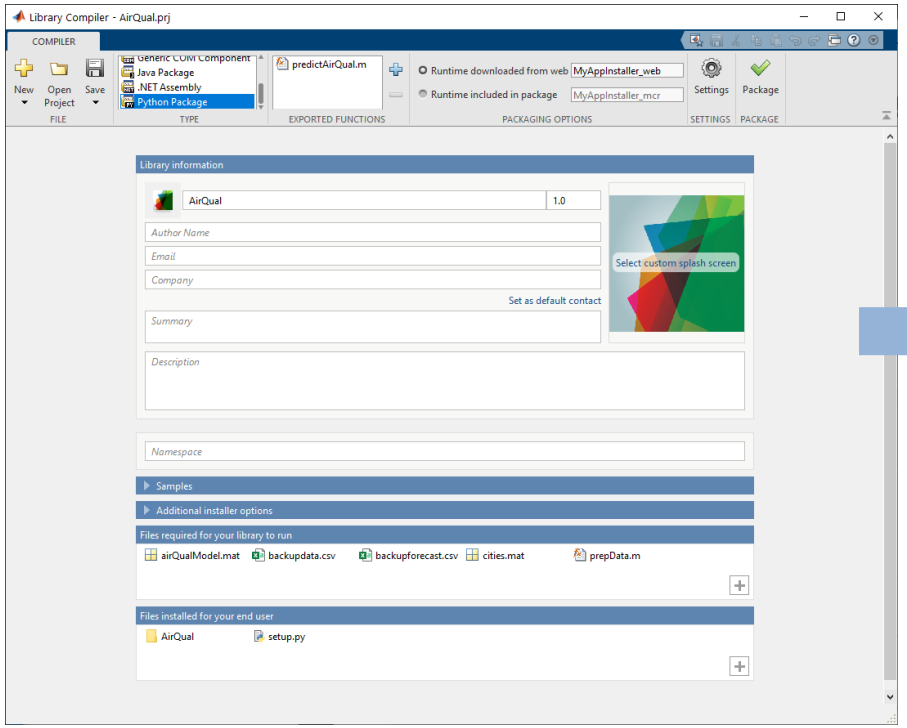
Data Access

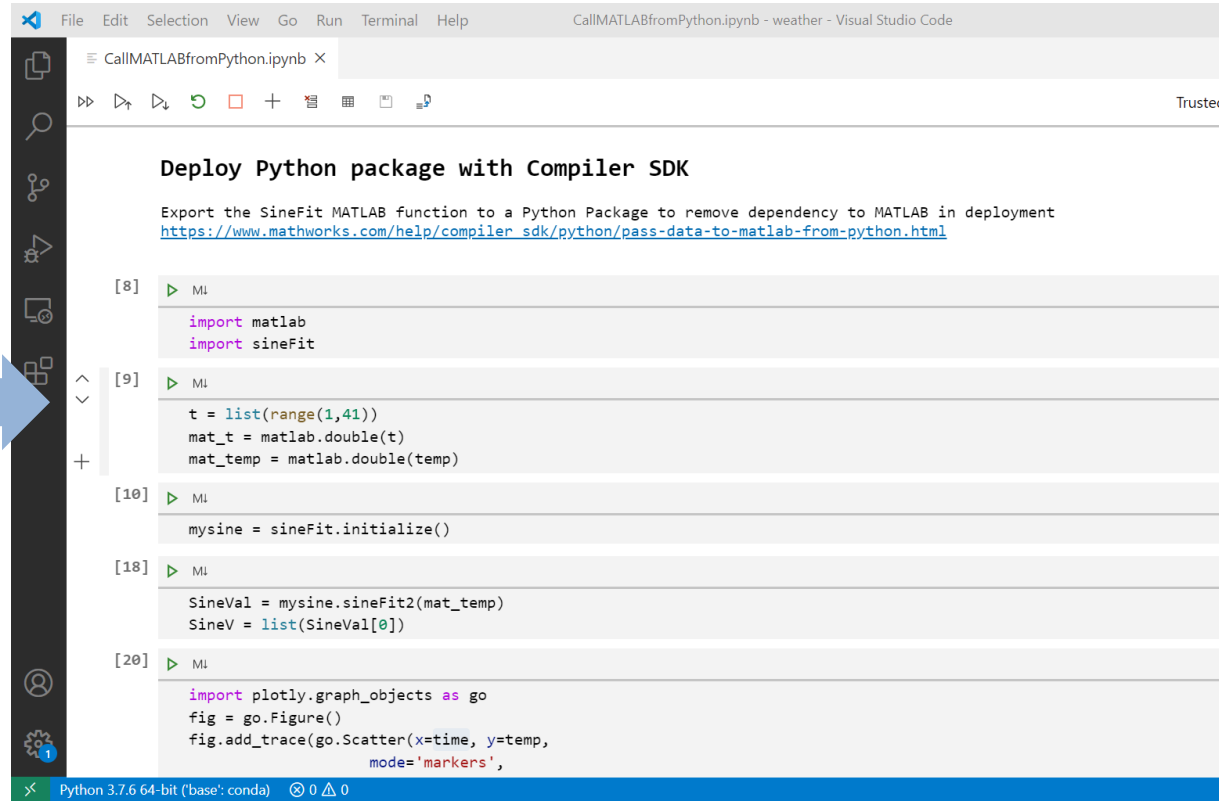
Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment







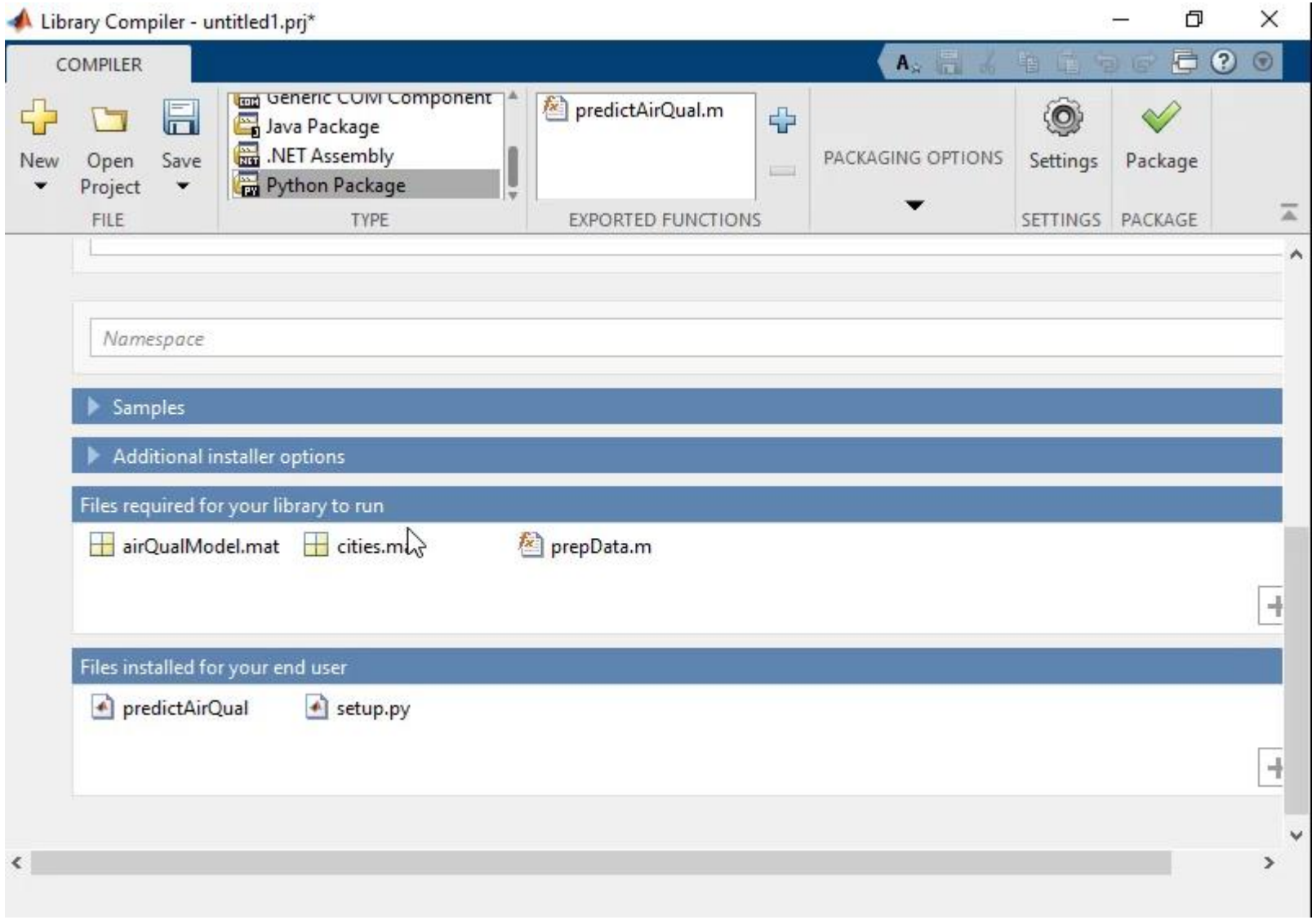
Data Access

Co-Execution

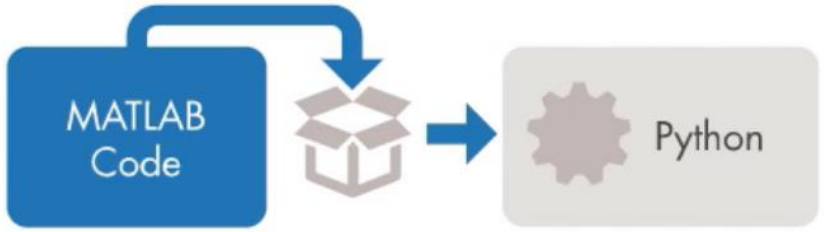
- Call Python from MATLAB
- Call MATLAB from Python

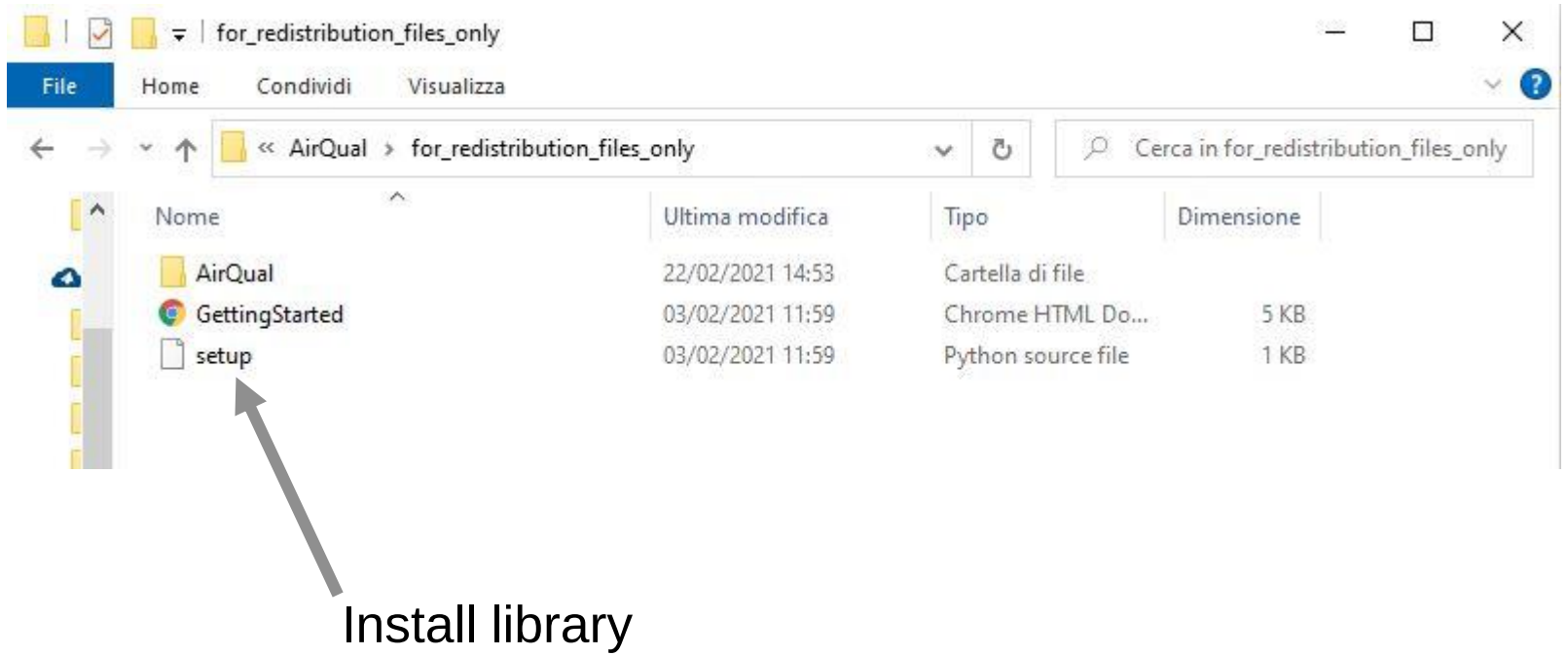
Deployment

Generate Python library from MATLAB functions



Generate Python library from MATLAB functions





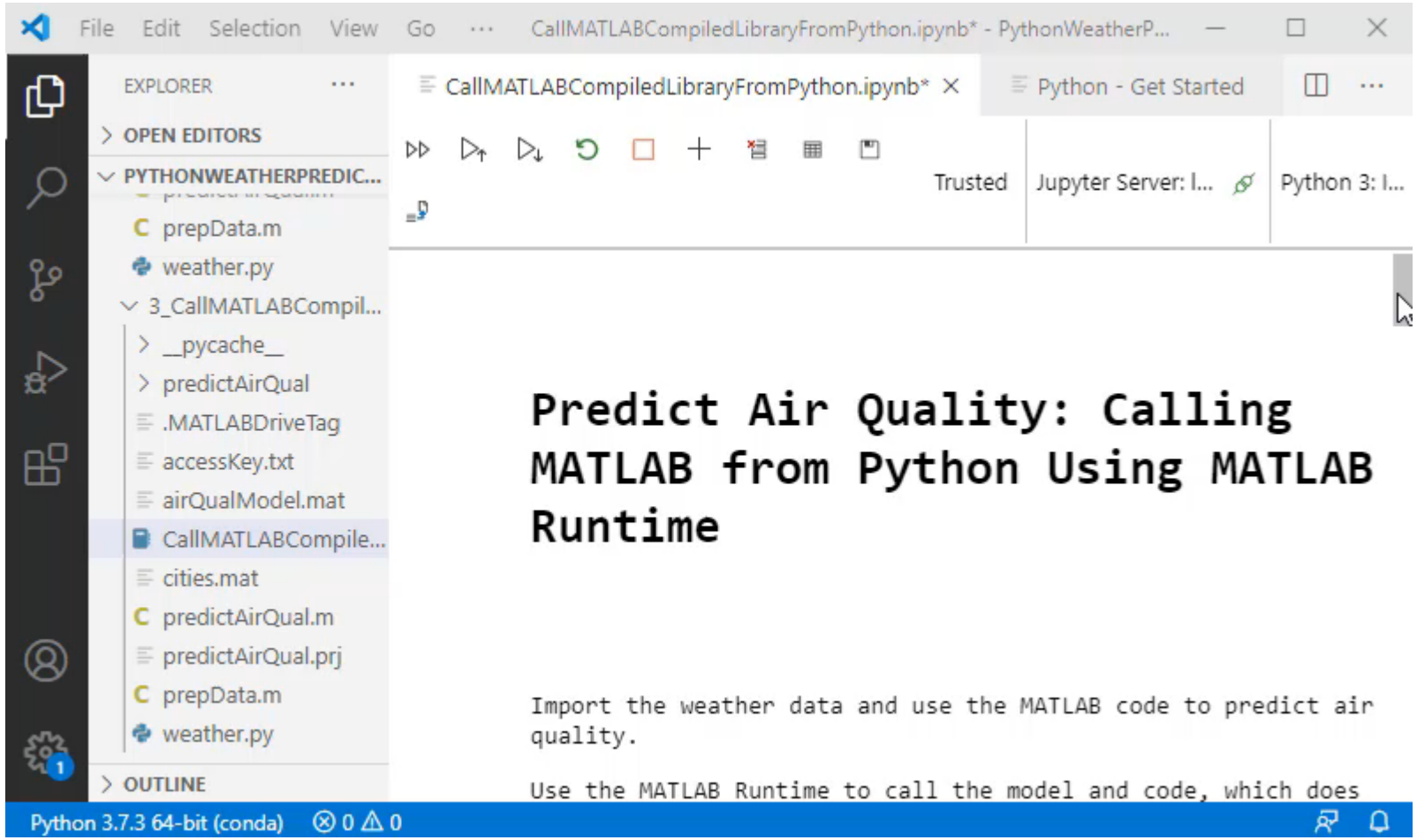
Execute Python library from MATLAB functions

Data Access

Co-Execution

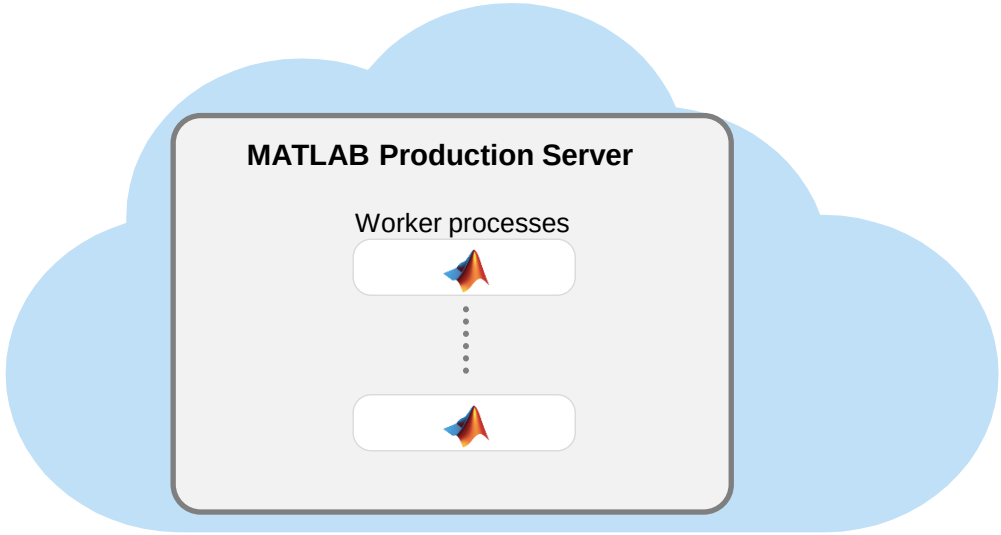
- Call Python from MATLAB
- Call MATLAB from Python

Deployment



MATLAB Production Server

Access functions as web services



Calling our function:

```
{"nargout":1,"rhs":["input"]}
```

Getting the result:

```
{"lhs":[{"mwdata":["output"],"mwsize":[1,6],"mwtype":"char"}]}
```

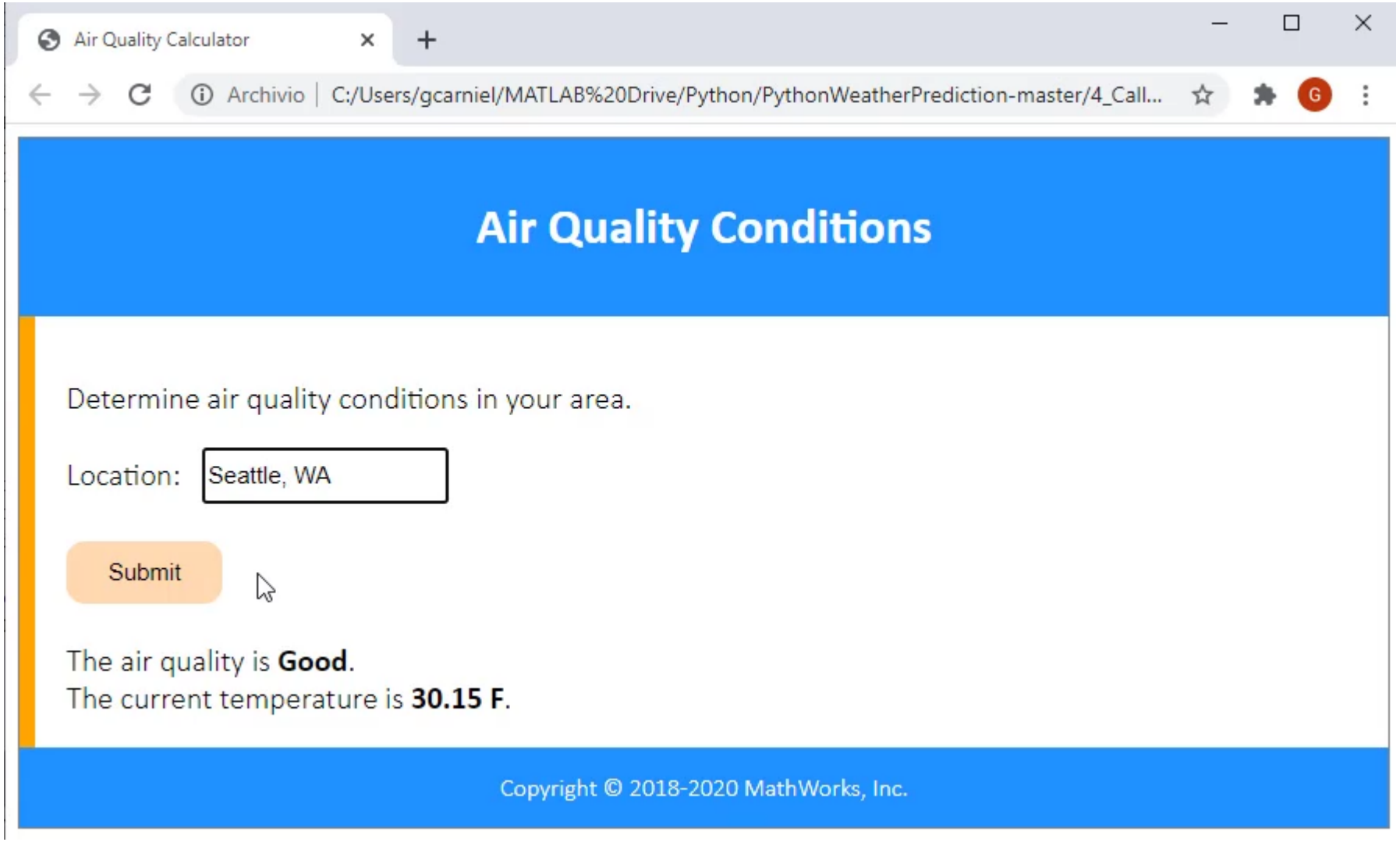
Execute Python library from MATLAB functions

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



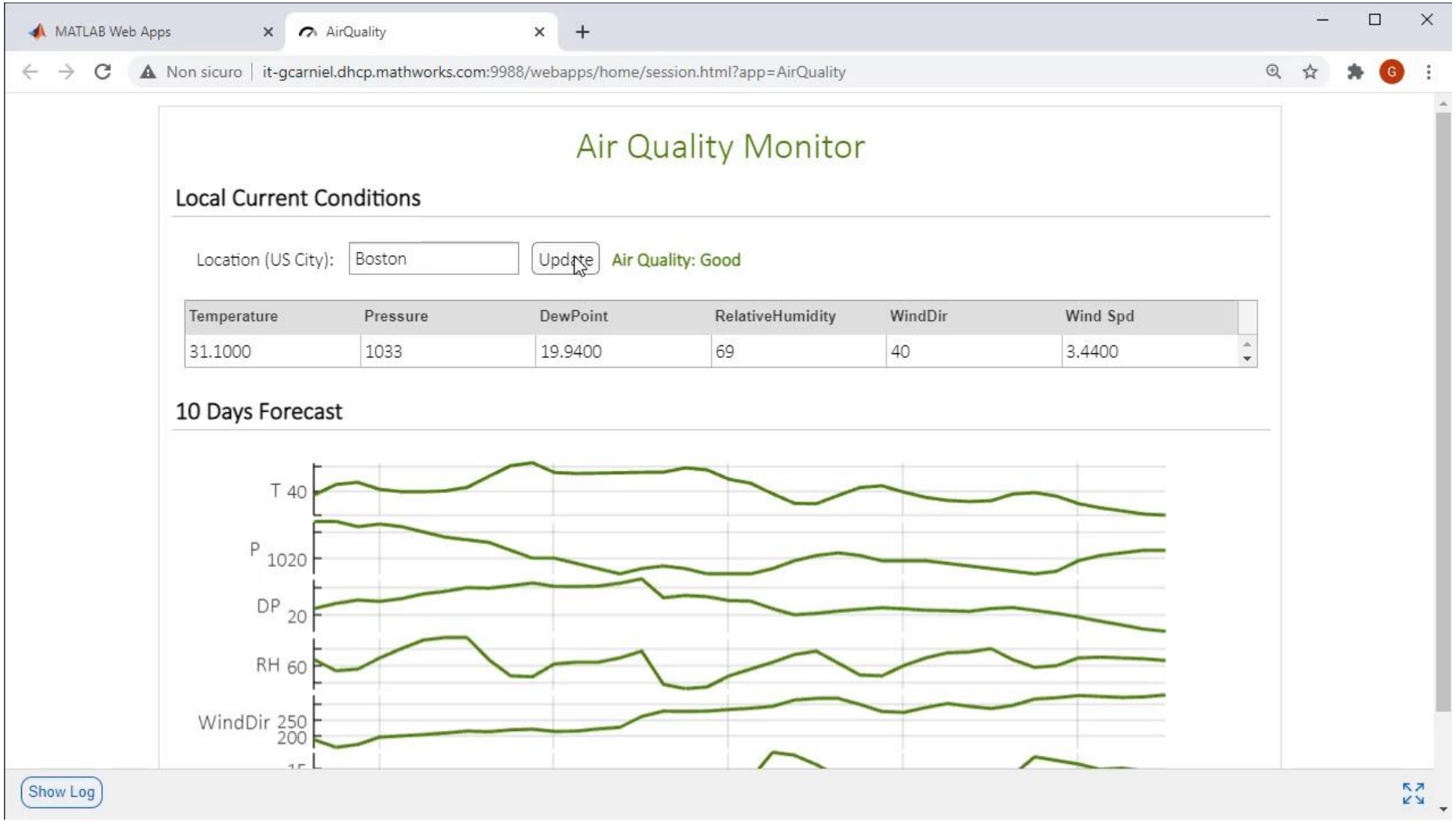
Share MATLAB App in the Web – Central Deployment

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Data Access

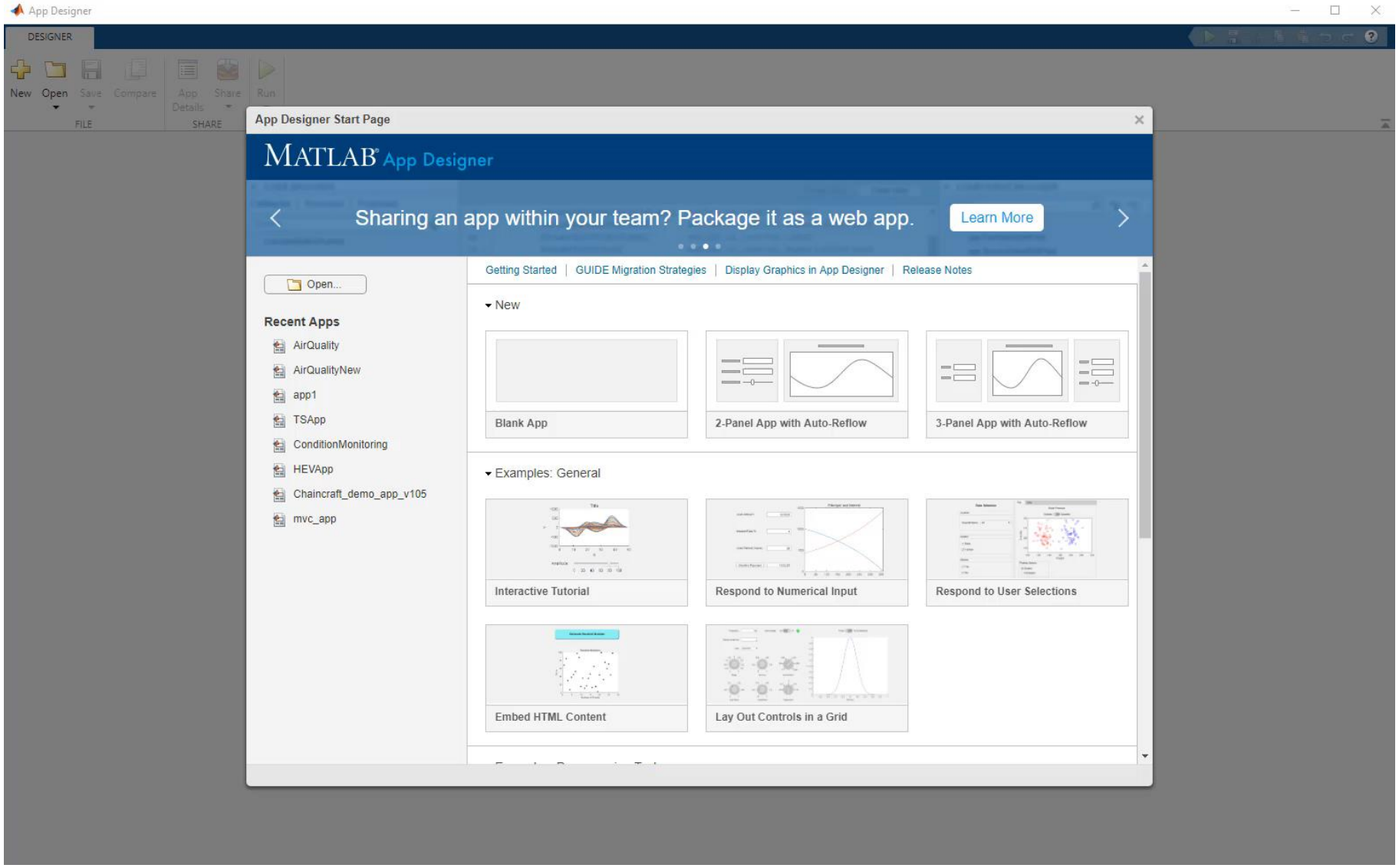
Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

MATLAB App Designer

App development for Non-Programmers



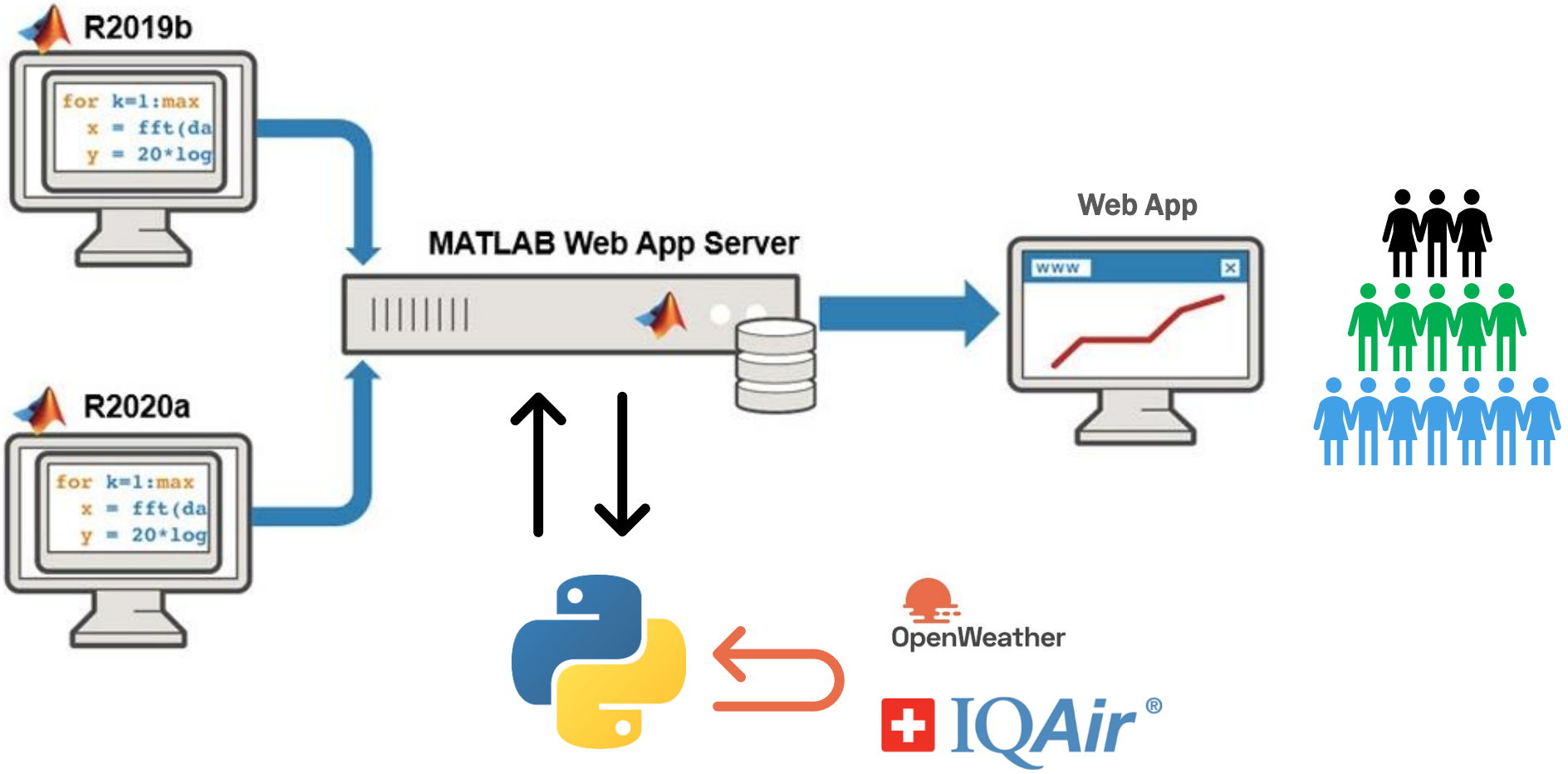
Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

MATLAB Web App Server – Central Deployment



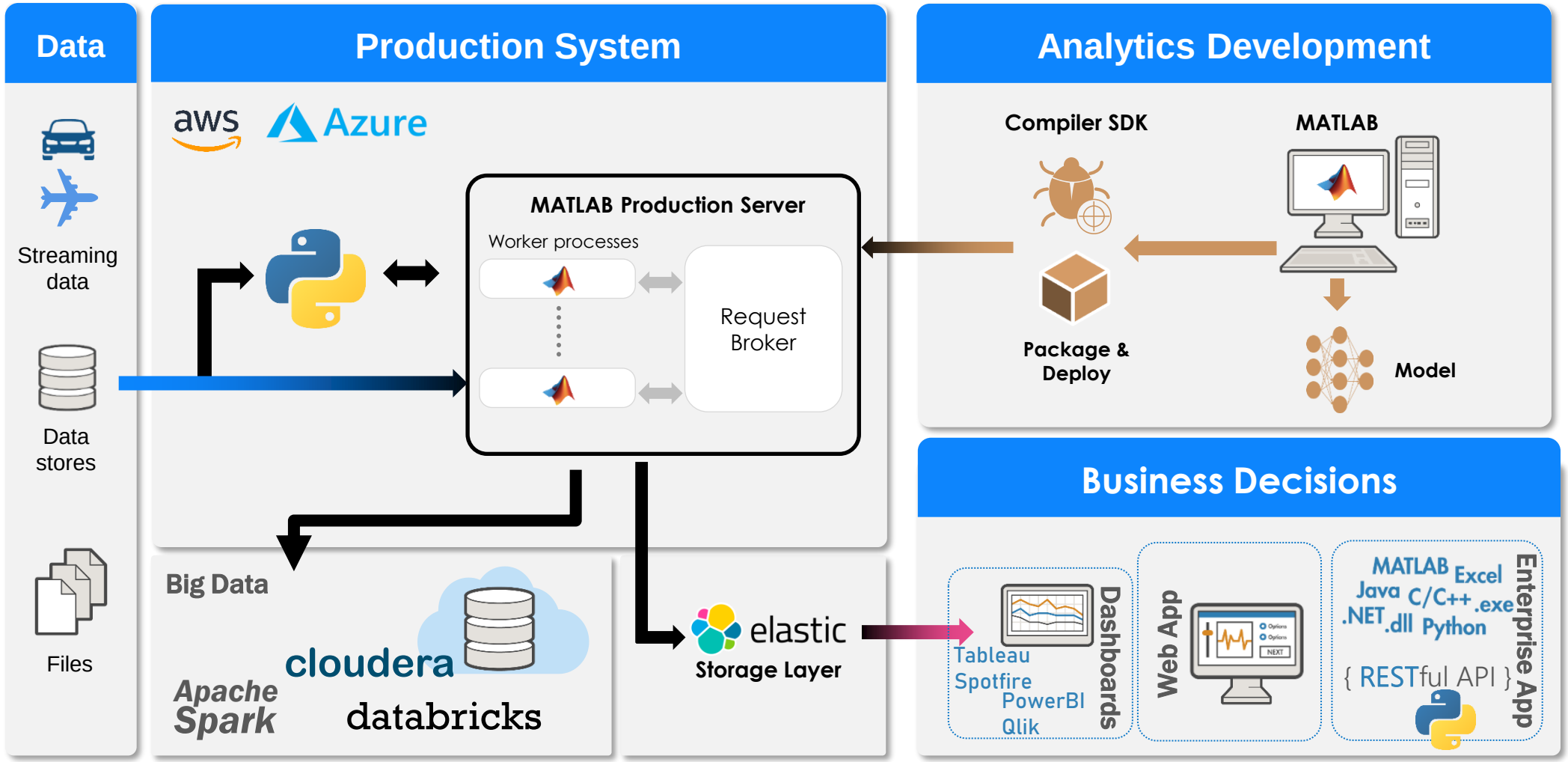
Integrate your Production System in an IT ecosystem

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Data Access

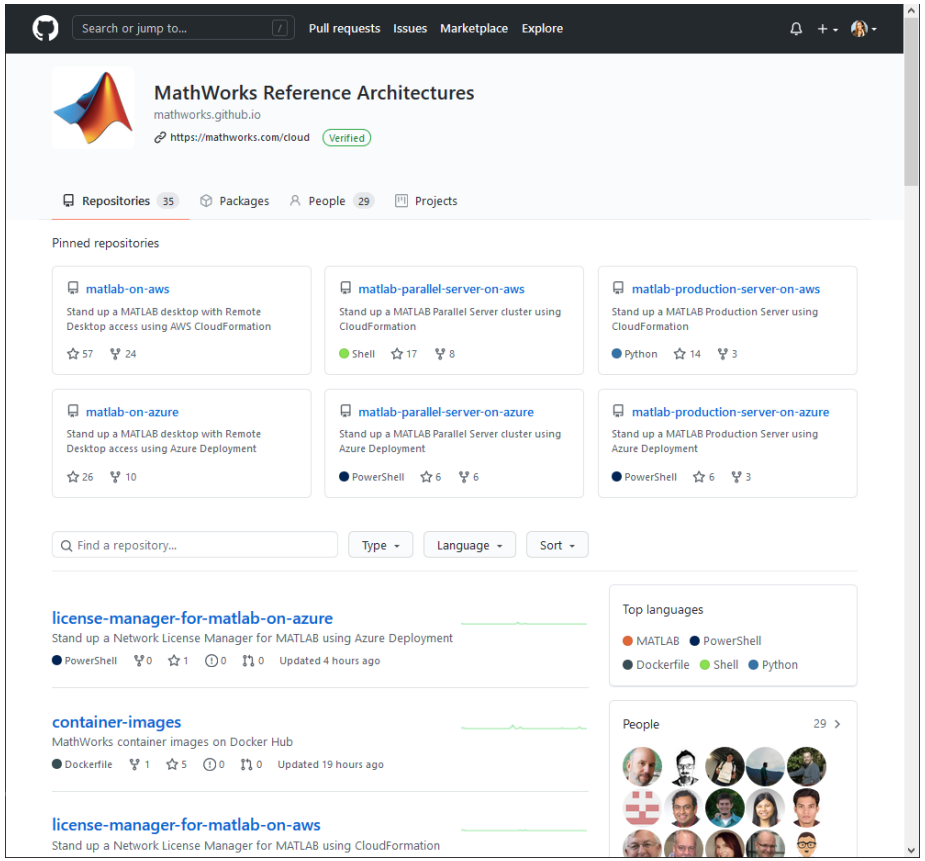
Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

Use MATLAB Reference Architectures for easy cloud setup, Dockerfiles, and interfaces to OSS

- https://github.com/mathworks-ref-arch/matlab-dockerfile



https://github.com/mathworks-ref-arch

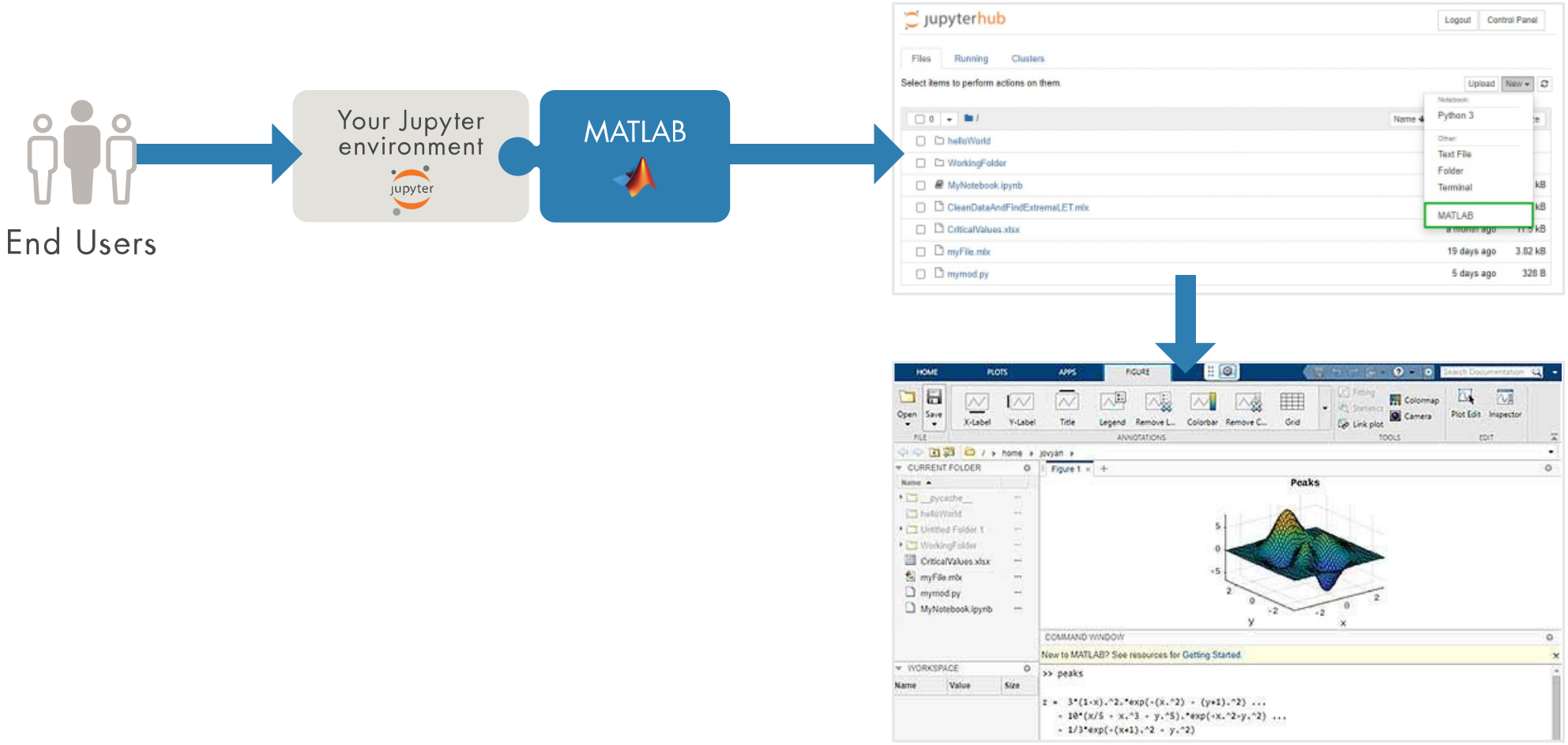
MATLAB Integration for Jupyter

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

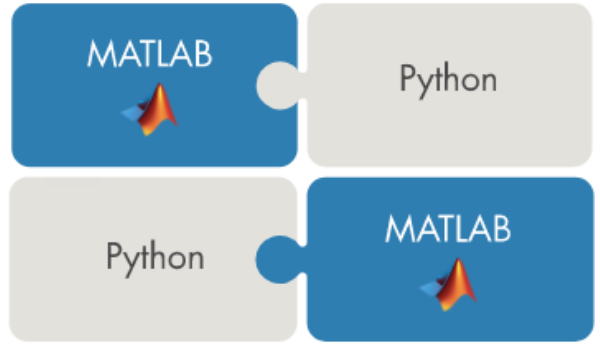


[MATLAB Integration for Jupyter (mathworks.com)]

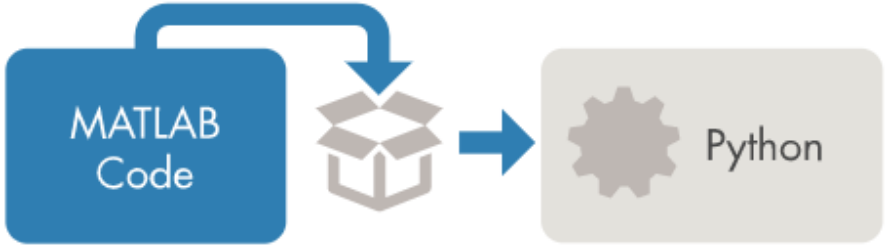
Summary:
Using MATLAB with Python

- Access Data
  - Weather App example

- Interoperability
  - Calling libraries written in Python from MATLAB
  - Calling MATLAB from Python



- Deploy Apps & Algos
  - Web App
  - Production API



Python libraries in MATLAB

Documentation

Python libraries in MATLAB (1)
Directly call Python® functionality from MATLAB®

Using Python Libraries

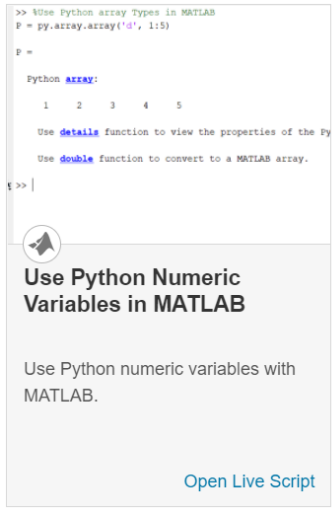
- **Access Python Modules from MATLAB - Getting Started**
  - How to create and use a Python object in MATLAB.
- **Configure Your System to Use Python**
  - How to verify you have installed a supported version of Python.
- **Call User-Defined Python Module**
  - Create a Python module used by examples in this documentation.
- **Understand Python Function Arguments**
  - Python method syntax which might be unfamiliar to MATLAB users.
- **Advanced Topics**
  - Code pattern differences you should be aware of.
- **Out-of-Process Execution of Python Functionality**
  - Execute Python scripts in processes that are separate from the MATLAB process.
- **Reload Out-of-Process Python Interpreter**
  - Reload out-of-process Python interpreter without restarting MATLAB.

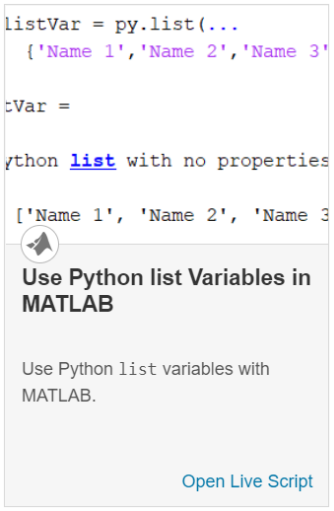
https://www.mathworks.com/help/matlab/call-python-libraries.html

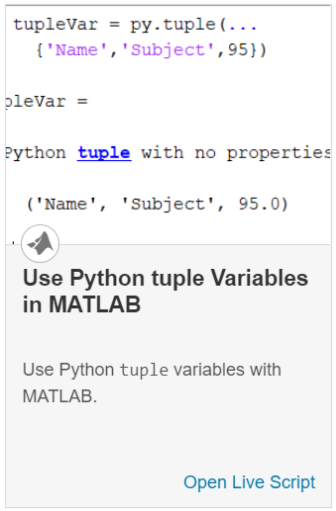
Python libraries in MATLAB (2)
Directly call Python® functionality from MATLAB®

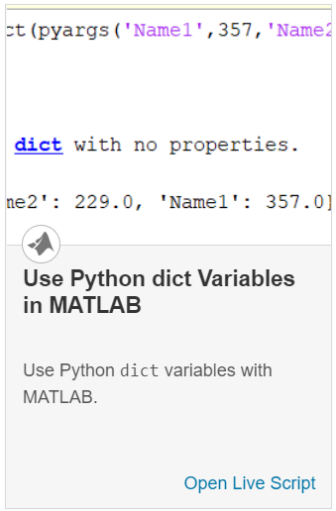
Passing Data

- **MATLAB to Python Data Type Mapping**
  - How MATLAB converts MATLAB data into compatible Python data types.
- **Access Elements in Python Container Types**
  - A Python container is typically a sequence type (`list` or `tuple`) or a mapping type (`dict`).
- **Pass Python Function to Python map Function**
  - This example shows how to display the length of each word in a list.









Additional resources

Resources

- General:
  - https://www.mathworks.com/products/matlab/matlab-and-python.html
- Python from MATLAB:
  - https://www.mathworks.com/help/matlab/call-python-libraries.html
- MATLAB from Python:
  - MATLAB Engine API:
    - https://www.mathworks.com/help/matlab/matlab-engine-for-python.html
  - MATLAB Compiler SDK:
    - https://www.mathworks.com/help/compiler_sdk/python_packages.html
  - Data type conversions:
    - https://www.mathworks.com/help/matlab/python-data-types.html
- Example:
  - https://github.com/mathworks/matlab-with-python

Cheatsheet

MathWorks®

Accelerating the pace of engineering and science

Using MATLAB® and Python® Together

The ≥ icon provides links to relevant sections of the MATLAB documentation to learn more.

Call Python in MATLAB

Access settings and status of Python interpreter:

```
>> pe = pyenv
```

Specify version to use:

```
>> pe = pyenv("Version",3.7)
```

Call Python modules and functions:

```
py.module_name.function_name
```

```
>> py.math.sqrt(42)
```

Pass keyword arguments

Use pyargs to pass keyword arguments

```
>>> foo(5,bar=42)
>> py.foo(5,pyargs('bar',42))
```

Reload modules

Reload the module after making updates:

```
>> py.importlib.reload(module)
```

Call MATLAB in Python

Install MATLAB Engine API for Python ≥

Run `setup.py` from an OS command window

```
$ cd [matlabroot]/extern/engines/python
$ python setup.py install
```

Call MATLAB functions

Import the module and start the engine

```
>>> import matlab.engine
>>> eng =
    matlab.engine.start_matlab()
```

Call functions through the engine

```
>>> x = eng.sqrt(42.0)
```

Capture multiple outputs

```
>>> x = eng.gcd(42.0,8.0,nargout=3)
```

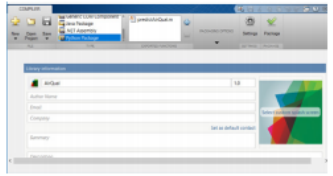
Stop the engine

```
>>> eng.exit()
```

Create Python Package

Package MATLAB functions ≥

Use the Library Compiler App to create a Python package for MATLAB functions



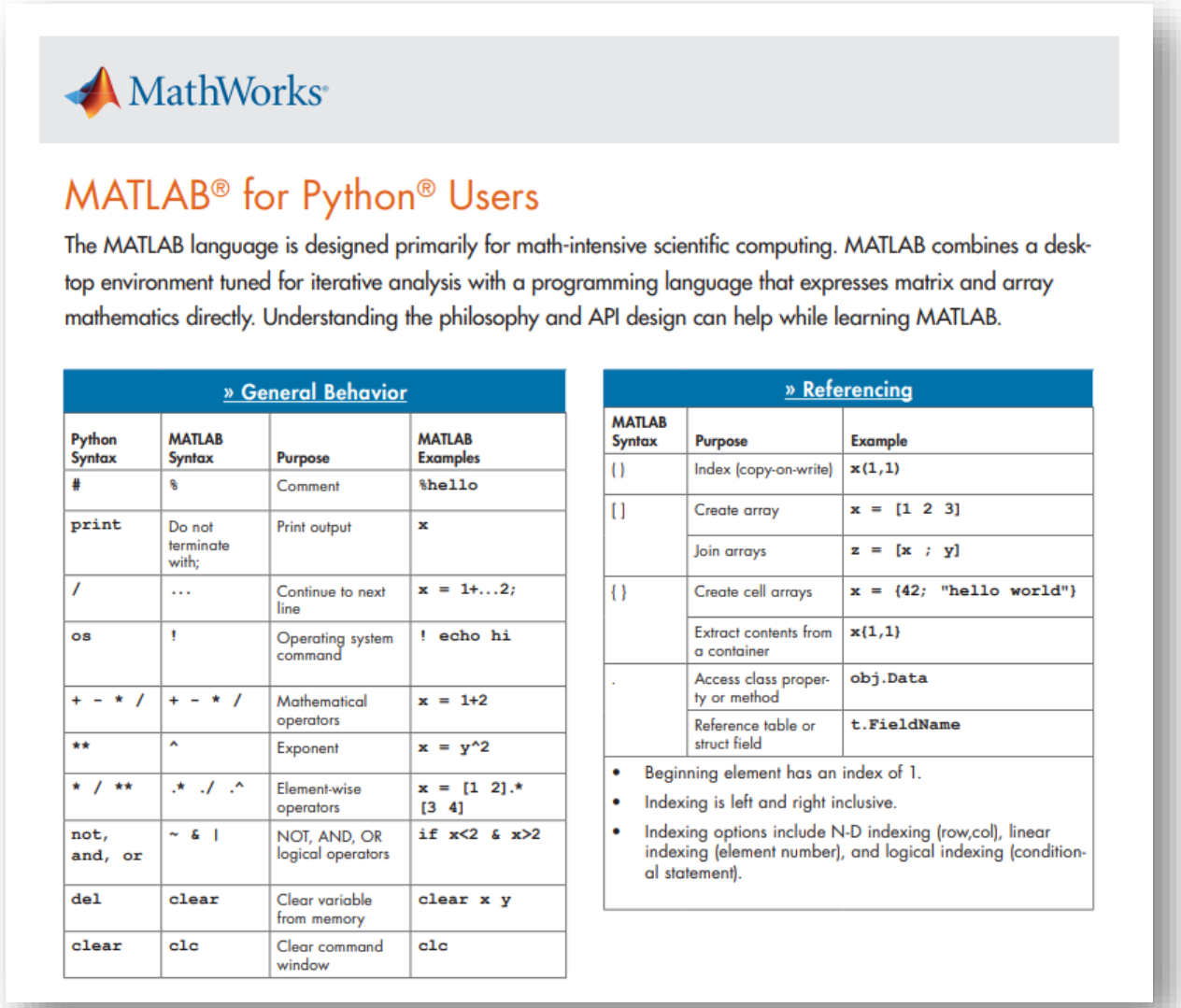
Invoke MATLAB functions from the Python package

```
>>> import PackageName
>>> pkg =
PackageName.initialize()
>>> result = pkg.foo()
```

Close package

```
>>> pkg.terminate()
```

NEW MATLAB for Python Users Cheat Sheet



MathWorks

MATLAB® for Python® Users

The MATLAB language is designed primarily for math-intensive scientific computing. MATLAB combines a desktop environment tuned for iterative analysis with a programming language that expresses matrix and array mathematics directly. Understanding the philosophy and API design can help while learning MATLAB.

» General Behavior

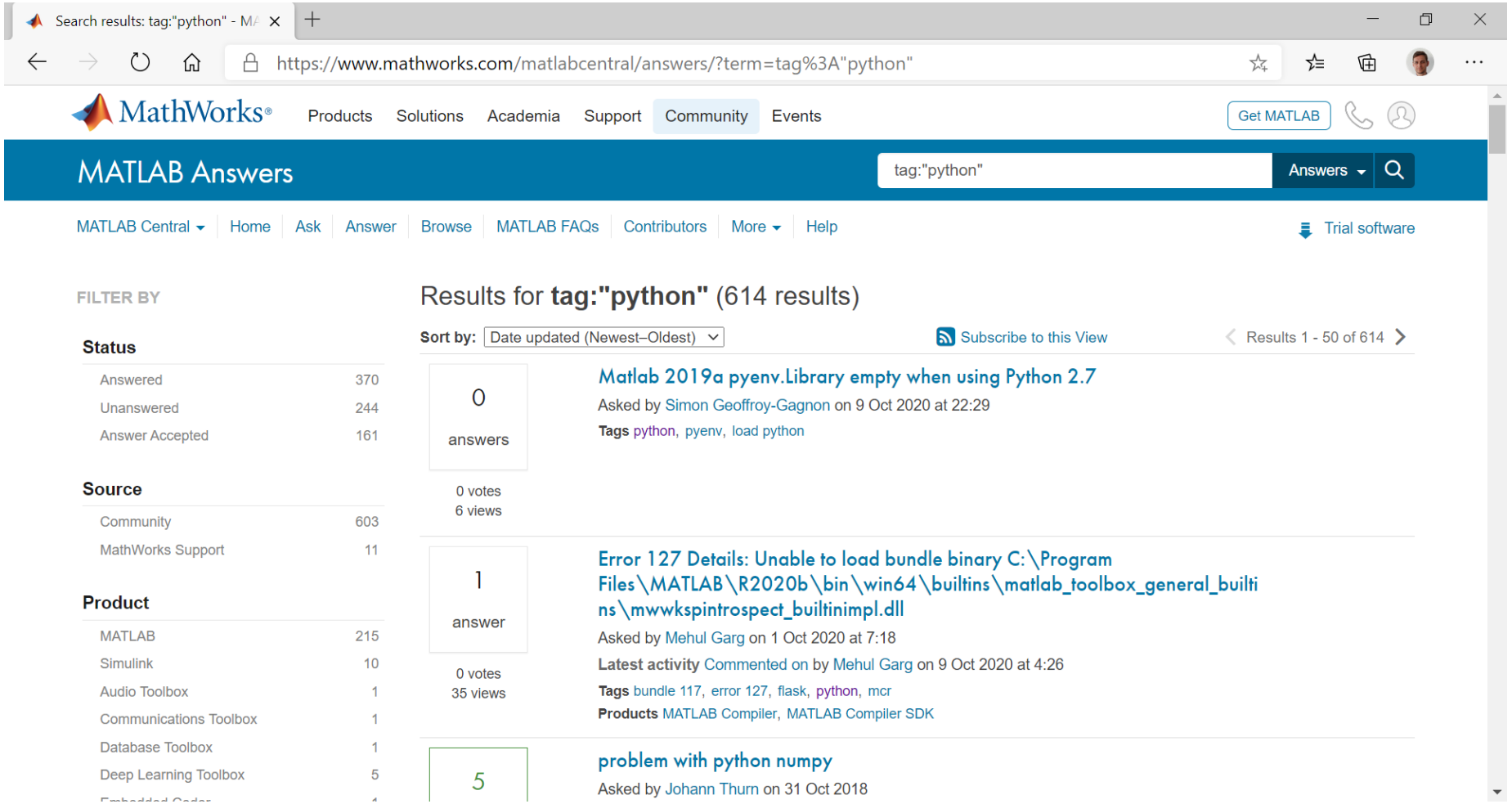
Python Syntax	MATLAB Syntax	Purpose	MATLAB Examples
`#`	`%`	Comment	`%hello`
`print`	Do not terminate with;	Print output	`x`
`/`	`...`	Continue to next line	`x = 1+...2;`
`os`	`!`	Operating system command	`! echo hi`
`+ - * /`	`+ - * /`	Mathematical operators	`x = 1+2`
`**`	`^`	Exponent	`x = y^2`
`* / **`	`.* ./ .^`	Element-wise operators	`x = [1 2].* [3 4]`
`not, and, or`	`~ & |`	NOT, AND, OR logical operators	`if x<2 & x>2`
`del`	`clear`	Clear variable from memory	`clear x y`
`clear`	`clc`	Clear command window	`clc`

» Referencing

MATLAB Syntax	Purpose	Example
`( )`	Index (copy-on-write)	`x(1,1)`
`[ ]`	Create array	`x = [1 2 3]`
	Join arrays	`z = [x ; y]`
`{ }`	Create cell arrays	`x = {42; "hello world"}`
	Extract contents from a container	`x{1,1}`
`.`	Access class property or method	`obj.Data`
	Reference table or struct field	`t.FieldName`

- Beginning element has an index of 1.
- Indexing is left and right inclusive.
- Indexing options include N-D indexing (row,col), linear indexing (element number), and logical indexing (conditional statement).

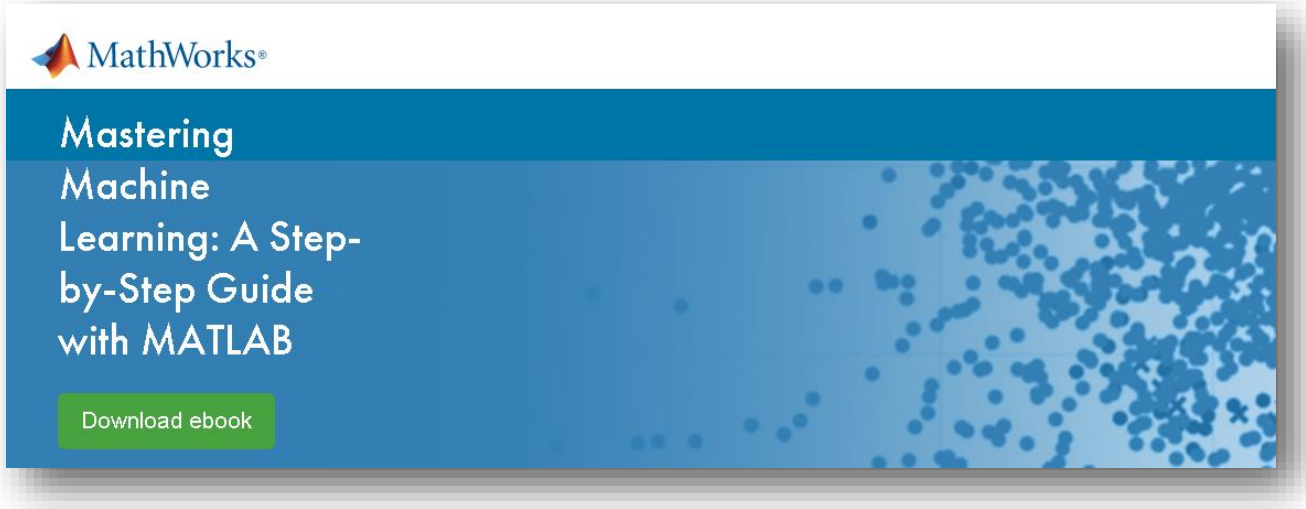
MATLAB Answers – tag:"python"

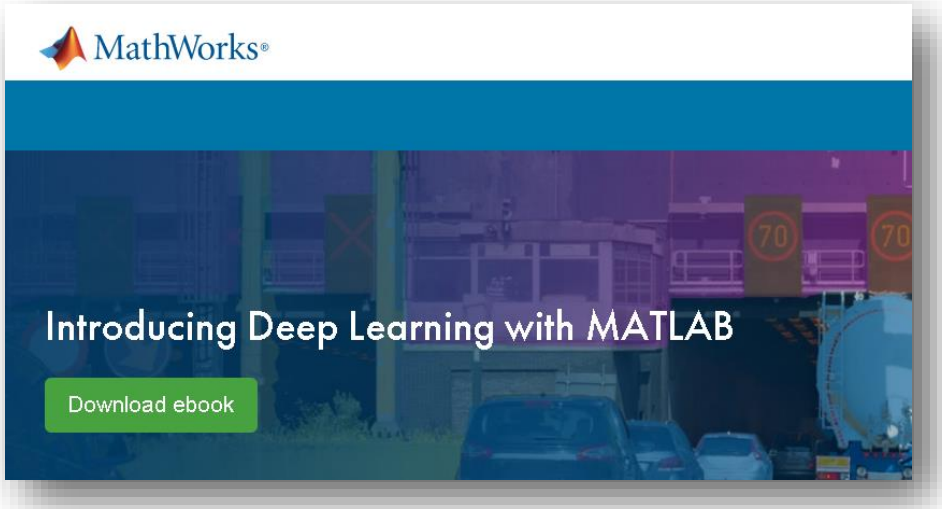


https://www.mathworks.com/matlabcentral/answers/?term=tag%3A%22python%22

Additional Resources

- Predictive Analytics
  - https://www.mathworks.com/discovery/predictive-analytics.html
- Deep Learning
  - https://www.mathworks.com/discovery/deep-learning.html
- Reinforcement Learning
  - https://www.mathworks.com/products/reinforcement-learning.html





Learn MATLAB for Free

Hands-on practice sessions and demonstrations

FREE

MATLAB Onramp

Get started quickly with the basics of MATLAB.

FREE

Simulink Onramp

Get started quickly with the basics of Simulink.

FREE

Machine Learning Onramp

Learn the basics of practical machine learning methods for classification problems.

FREE

Deep Learning Onramp

Get started quickly using deep learning methods to perform image recognition.

NEW FREE

Reinforcement Learning Onramp

Master the basics of creating intelligent controllers that learn from experience.

NEW FREE

Image Processing Onramp

Learn the basics of practical image processing techniques in MATLAB.

NEW FREE

Signal Processing Onramp

An interactive introduction to signal processing methods for spectral analysis.

NEW FREE

Simscape Onramp

Learn the basics of simulating physical systems in Simscape.

FREE

Stateflow Onramp

Learn the basics of creating, editing, and simulating state machines in Stateflow.

NEW FREE

Control Design Onramp with Simulink

Get started quickly with the basics of feedback control design in Simulink.

NEW FREE

Optimization Onramp

Learn the basics of solving optimization problems in MATLAB using the problem-based approach.

NEW

Circuit Simulation Onramp

7 modules | 2 hours | Languages

Learn the basics of simulating electrical circuits in Simscape.

https://matlabacademy.mathworks.com/#getting-started

Campus-Wide MATLAB Training

Search MathWorks.com

Available Courses

https://matlabacademy.mathworks.com/

MATLAB Fundamentals

Details

Machine Learning with MATLAB

Details

Deep Learning with MATLAB

Details

MATLAB for Data Processing and Visualization

Details

Solving Ordinary Differential Equations with MATLAB

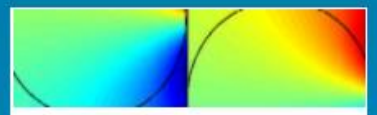
Details

Introduction to Linear Algebra

Details

Introduction to Statistical Methods with MATLAB

Details



Solving Nonlinear Equations with MATLAB

Details

MATLAB Programming Techniques

Details

MATLAB for Financial Applications

Details

Introduction to Symbolic Math with MATLAB

Image Processing with MATLAB

MathWorks

MATLAB and Simulink Training

Training Overview | Find a Course | Get Certified | Training at Your Facility | More | » My Courses | Contact Training

MATLAB and Simulink Course Schedule

Results 1 - 50 of 223

FILTERED BY English ✕ Clear Filters

Dates	Course	Location	Language	Price	Register
On Demand	**MATLAB Programming Techniques** 180 days of full access from the day of purchase	Self-Paced	English	USD 350	
On Demand	**Machine Learning with MATLAB** 180 days of full access from the day of purchase	Self-Paced	English	USD 350	
On Demand	**MATLAB Fundamentals** 180 days of full access from the day of purchase	Self-Paced	English	USD 500	
On Demand	**Image Processing with MATLAB** 180 days of full access from the day of purchase	Self-Paced	English	USD 350	
On Demand	**Deep Learning with MATLAB** 180 days of full access from the day of purchase	Self-Paced	English	USD 350	
On Demand	**MATLAB for Data Processing and Visualization** 180 days of full access from the day of purchase	Self-Paced	English	USD 200	

Campus Wide License Program
Offer FREE !!

https://www.mathworks.com/training-schedule/training_cart/add_class

MATLAB Certification

Certification sets individuals apart in the job market and can help accelerate professional growth

Two paid levels of certification are offered:

Certified MATLAB Associate

Certified MATLAB Professional

Share credentials on platforms such as

LinkedIn and Facebook

Rupesh Gosavi • 2nd

MathWorks Certified MATLAB Associate. Serial Entrepreneur. Chief Product Strategist. Technology Mentor.

Pune, Maharashtra, India

Connect | Message | More...

Professional MATLAB Instructor

College of Engineering Pune

See contact info

500+ connections

I am a passionate and enthusiastic collaborative engineer. My mission is to build innovations through collaborative research technique. Currently, I am pursuing a role of chief product strategist at Bluemark. I am responsible to manage technical and commercial aspects of a product. Having a good knowledge ...

1 Certification

MATHWORKS CERTIFIED MATLAB ASSOCIATE

Jul 2016 – Present • License 1-2475804532

The MathWorks

Certificates Learning Course

MathWorks | Training Services

Course Completion Certificate

Kantika Wongkasem

has successfully completed 100% of the self-paced training course

Deep Learning Onramp

DIRECTOR, TRAINING SERVICES

06 April 2021

MathWorks | Training Services

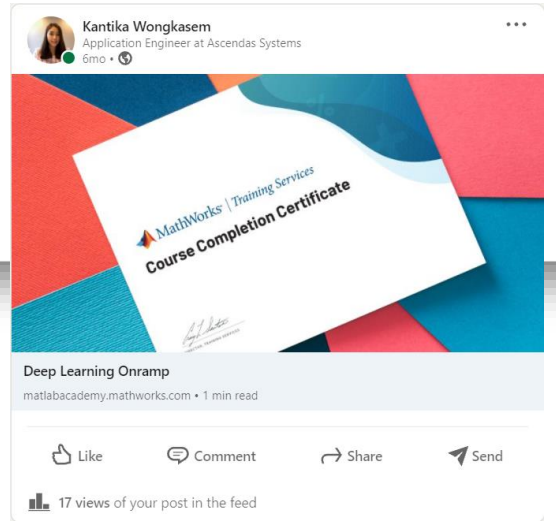
Progress Report

Name: Kantika Wongkasem
Course: Deep Learning Onramp
Progress: 100% complete (as of 06 April 2021)

Chapters

1. Introduction 100%
2. Using Pretrained Networks 100%
3. Managing Collections of Image Data 100%
4. Performing Transfer Learning 100%
5. Conclusion 100%

Release: R2020b | Language: English



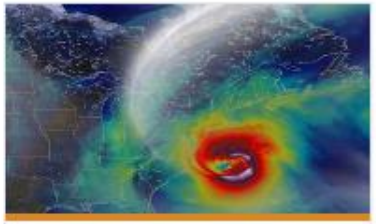
Share Certificates upgrade profile on facebook LinkedIn

MATLAB Courseware

Free sets of course materials developed by faculty from top universities Curricula available for all STEM disciplines and at multiple levels

Teaching Quantitative Finance and Risk Management with MATLAB

» Integrate MATLAB into your Quantitative Finance and Risk Management curriculum



Teaching Computational Science Using MATLAB

» Integrate MATLAB into your robust data analysis, data visualization and exploration curriculum

Teaching Biology with MATLAB

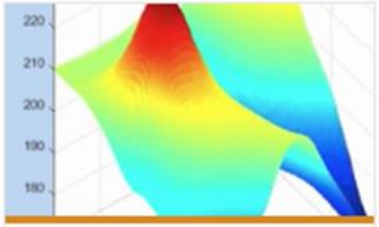
» Integrate MATLAB into your Biology curriculum

Teaching Calculus with MATLAB

» Integrate MATLAB into your Calculus curriculum

Teaching Chemistry with MATLAB

» Integrate MATLAB into your Chemistry curriculum



Teaching Geoscience with MATLAB

» Integrate MATLAB into your Geoscience curriculum

Teaching Physics with MATLAB

» Integrate MATLAB into your Physics curriculum

Teaching Psychology and Neuroscience with MATLAB

» Integrate MATLAB into your Psychology and Neuroscience curriculum

Teaching Data Science with MATLAB

» Integrate MATLAB into your Data Science curriculum

Teaching Deep Learning with MATLAB

» Integrate MATLAB into your Deep Learning curriculum

Teaching Econometrics with MATLAB

» Integrate MATLAB into your Econometrics curriculum

MATLAB and Simulink Based Books

- More than 2000 titles in 26 languages for educational and professional
- Subjects include:
  - Biosciences and Biomedical
  - Chemistry and Chemical Engineering
  - Control Systems
  - Digital Signal Processing
  - Earth Sciences
  - Economics and Computational Finance
  - Image and Video Processing
  - Mathematics
  - Mechanical Engineering
  - Neural Networks and Fuzzy Logic
  - Physics
  - Programming and Computer Science
  - Robotics
  - System Modeling and Simulation

Ascendas Solutions: Tools & Support

Technical Support

Technical Support:

✉ : support@techsource-asia.com

Contact Us:

📱 : https://www.ascendas-asia.com
✉ : th-events@ascendas-asia.com
☎ : +66 2 234 6721

TechSource Systems | Events (techsource-asia.com)