MATLAB & Simulink with Python

23rd Feb 2021

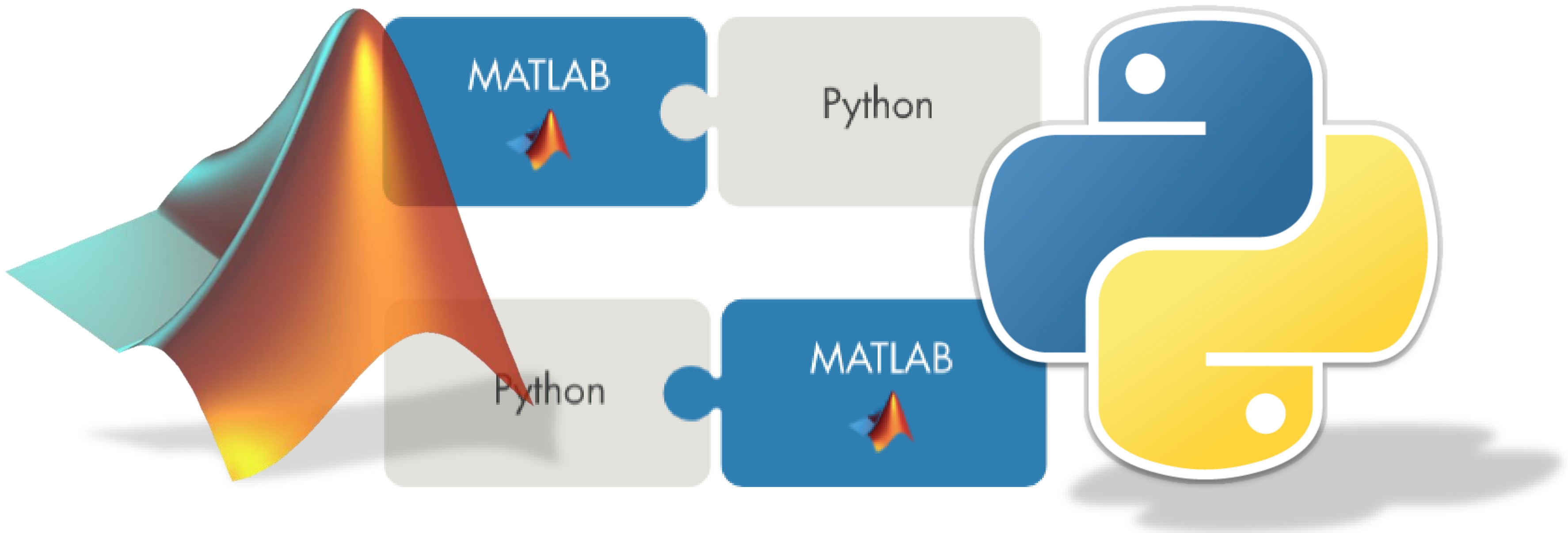
Supamith Sutharojana
Application Engineers Team Lead
ASCENDAS Systems Co.,Ltd
Supamith.Sutharojana@ascendas-asia.com
+662 234 6721 #3009

R2020b

Southeast Asia's sole distributor of
MATLAB & SIMULINK

ASCENDAS SYSTEMS

Who're you . . .



MATLAB Users

MATLAB & Python can work together

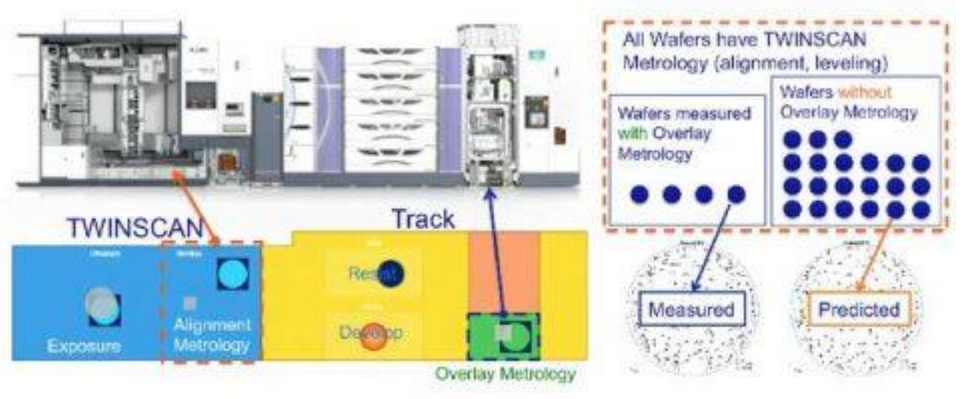
Python Users

Customer References

“As a process engineer I had no experience with neural networks or machine learning. I worked through the MATLAB examples to find the best machine learning functions for generating virtual metrology. I couldn’t have done this in C or Python—it would’ve taken too long to find, validate, and integrate the right packages.”

Emil Schmitt-Weaver, ASML

ASML Develops Virtual Metrology Technology for Semiconductor Manufacturing with Machine Learning



TWINSCAN
Track
Exposure
Alignment Metrology
Resist
Develop
Overlay Metrology
All Wafers have TWINSCAN Metrology (alignment, leveling)
Wafers measured with Overlay Metrology
Wafers without Overlay Metrology
Measured
Predicted

[ASML Develops Virtual Metrology Technology for Semiconductor Manufacturing with Machine Learning - MATLAB & Simulink (mathworks.com)]

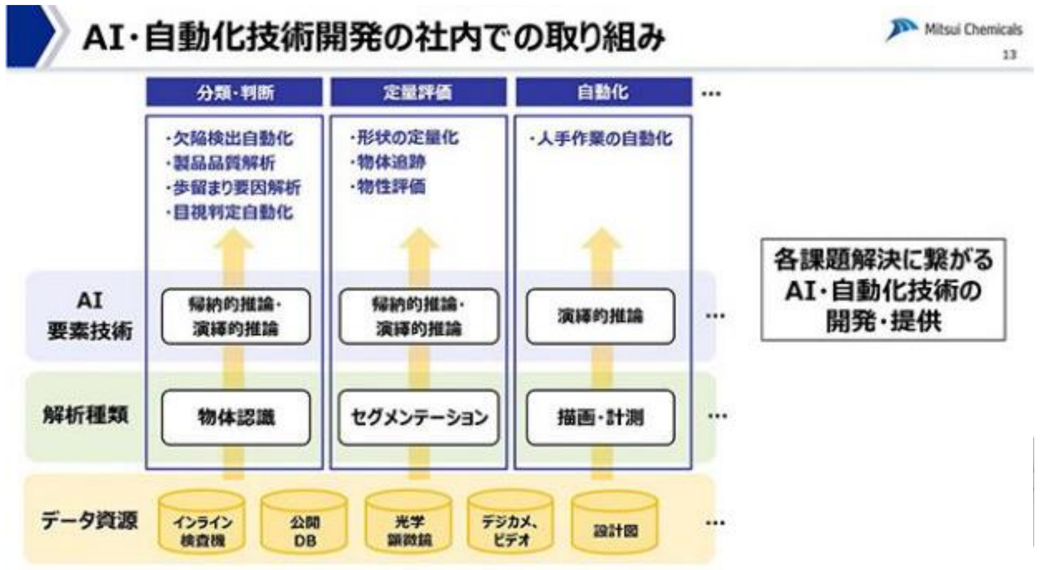
Customer References

"MATLAB solved our problems on the field implementation and saved development time. That led to highly accurate development."

Shintaro Maekawa, Mitsui Chemicals, Inc.

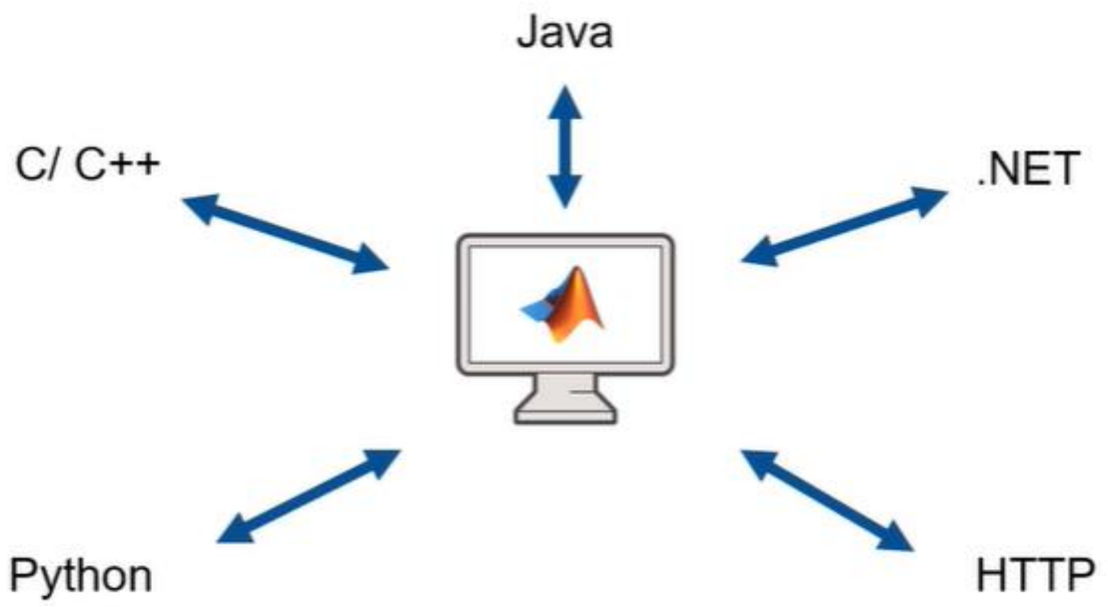
Mitsui Chemicals

Mitsui Chemicals Deploys AI and Automation Systems with TensorFlow and MATLAB



[Mitsui Chemicals Deploys AI and Automation Systems with TensorFlow and MATLAB - MATLAB & Simulink (mathworks.com)]

MATLAB provides flexible integration with multiple languages



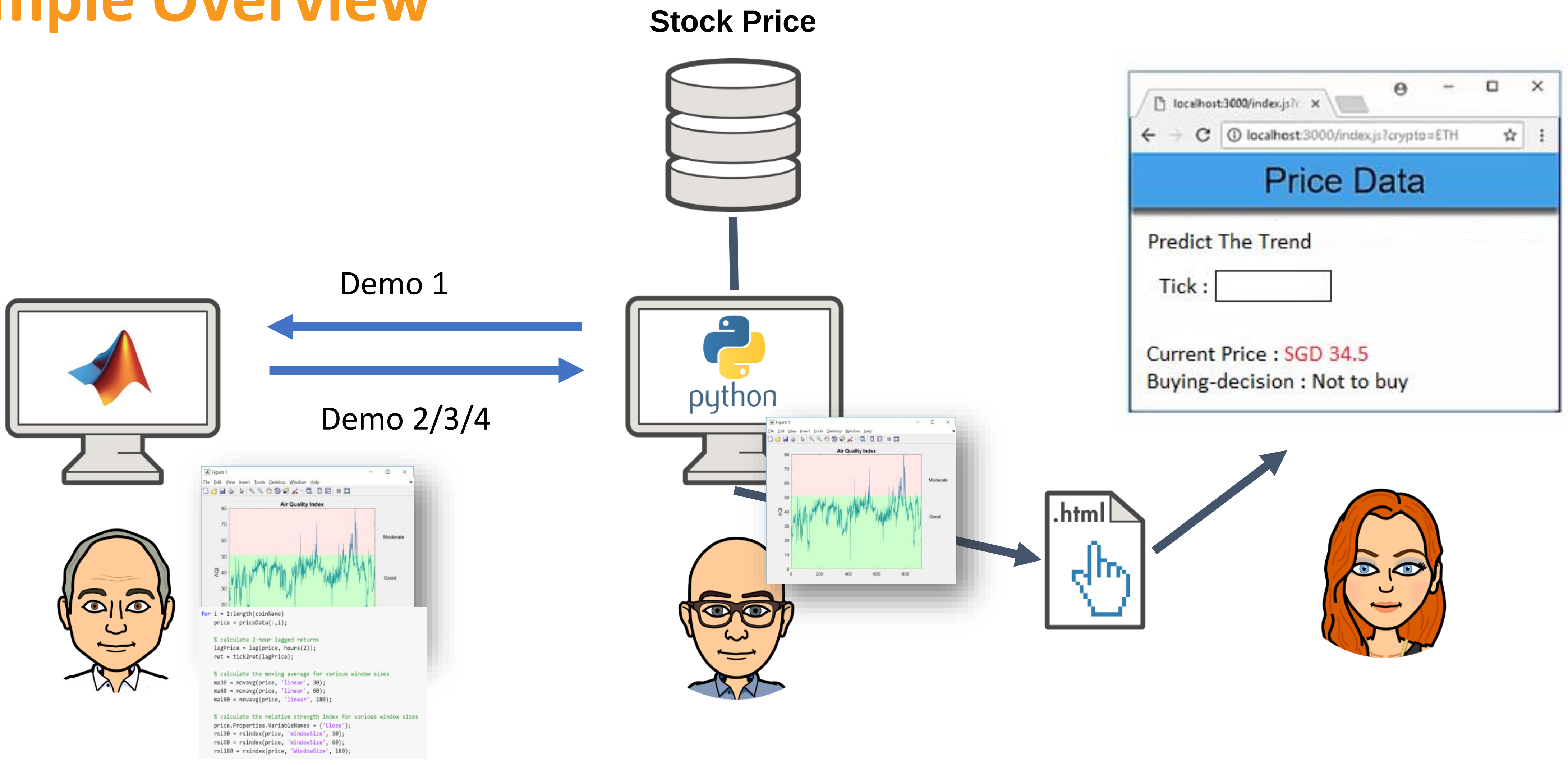
- MATLAB provides API to integrate with *Multiple Programming Language*
- They are *two-way interactive* processes through API, or MATLAB can compile their script to java library, .Net Assembly, etc.

https://www.mathworks.com/products/matlab/matlab-and-other-programming-languages.html

Agenda

- Example Overview
- *Part 1* : Calling Python Libraries from MATLAB
- *Part 2* : Calling MATLAB from Python
  - via MATLAB Engine API
  - via MATLAB Runtime (MATLAB Compiler SDK)
  - via MATLAB Production Server
- Part 3 : Simulink and Python
- Part 4 : Additional info
  - Data management
  - Integration with Anaconda
  - Troubleshooting & Resources

Example Overview



```
for i = 1:length(coinName)
    price = priceData(:,i);

    % calculate 2-hour lagged returns
    lagPrice = lag(price, hours(2));
    ret = tick2ret(lagPrice);

    % calculate the moving average for various window sizes
    ma30 = movavg(price, 'linear', 30);
    ma60 = movavg(price, 'linear', 60);
    ma180 = movavg(price, 'linear', 180);

    % calculate the relative strength index for various window sizes
    price.Properties.VariableNames = {'Close'};
    rsi30 = rsindex(price, 'WindowSize', 30);
    rsi60 = rsindex(price, 'WindowSize', 60);
    rsi180 = rsindex(price, 'WindowSize', 180);
```

localhost:3000/index.js?c

localhost:3000/index.js?crypto=ETH

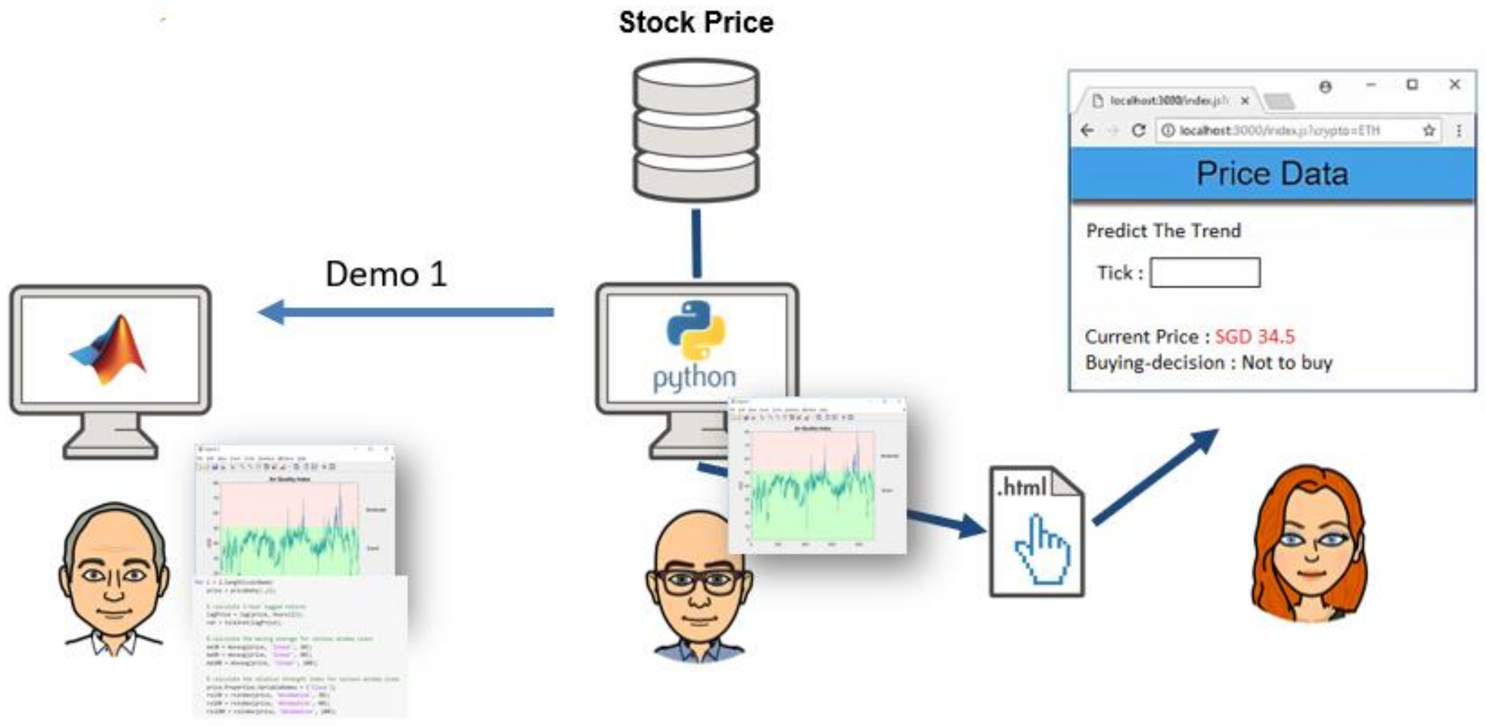
Price Data

Predict The Trend

Tick :

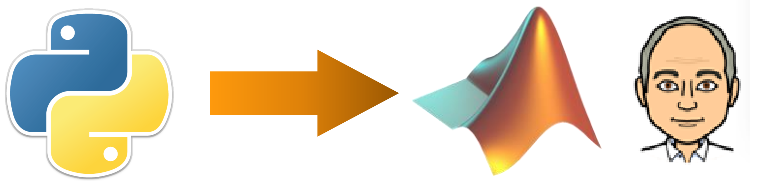
Current Price : SGD 34.5

Buying-decision : Not to buy



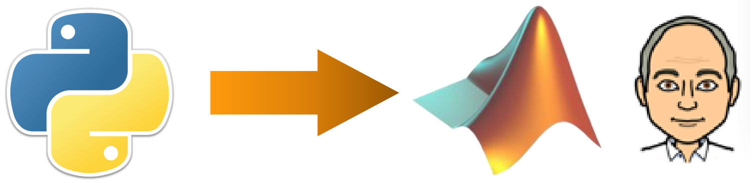
More information :

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



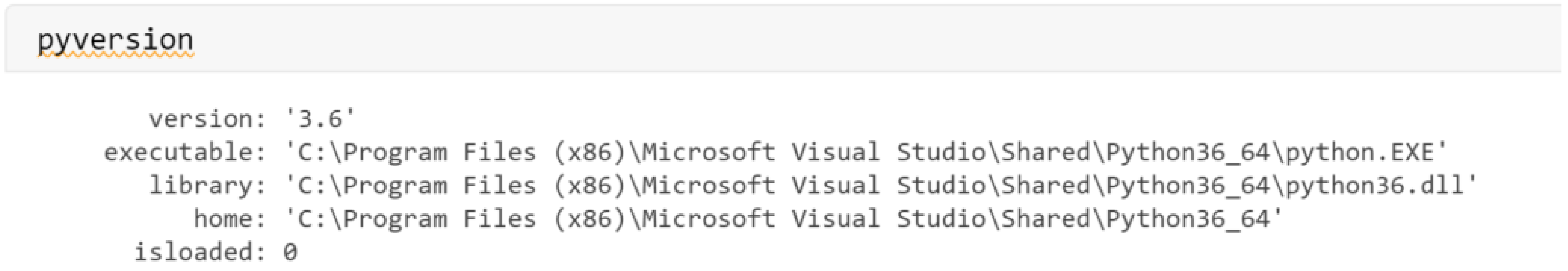
- **Part 1 : Calling Python Libraries from MATLAB**

• *Part 1* : Calling Python Libraries from MATLAB



Let test the connection with Python

- Ensure Python is installed and confirm the version

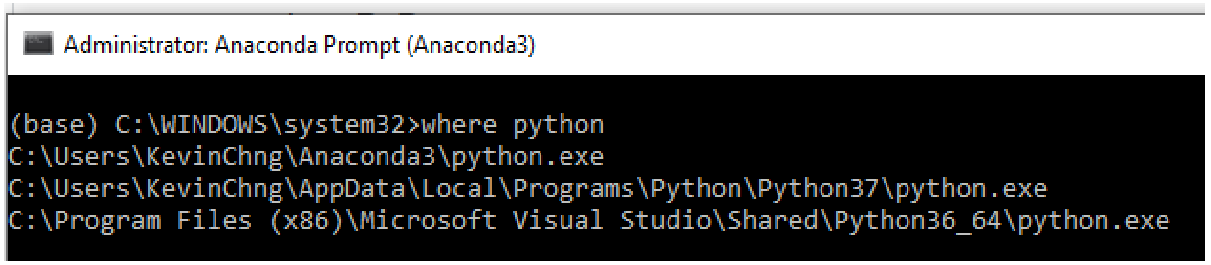


```
pyversion
```

```
     version: '3.6'
  executable: 'C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64\python.EXE'
     library: 'C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64\python36.dll'
        home: 'C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64'
    isloaded: 0
```

Can I call Anaconda's python from MATLAB? Many people ask me

In Anaconda, >> where python



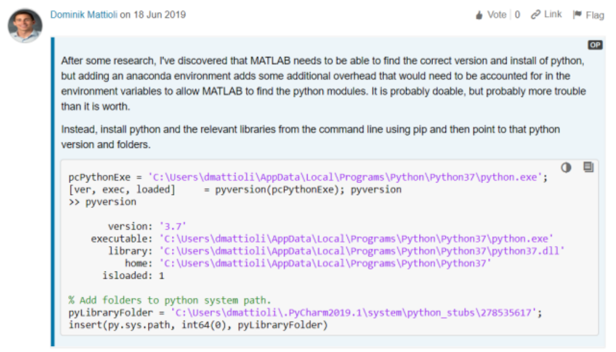
Administrator: Anaconda Prompt (Anaconda3)

```
(base) C:\WINDOWS\system32>where python
C:\Users\KevinChng\Anaconda3\python.exe
C:\Users\KevinChng\AppData\Local\Programs\Python\Python37\python.exe
C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64\python.exe
```

Solution

>> pyversion(path)

https://www.mathworks.com/matlabcentral/answers/466974-how-to-change-python-path



Dominik Mattioli on 18 Jun 2019

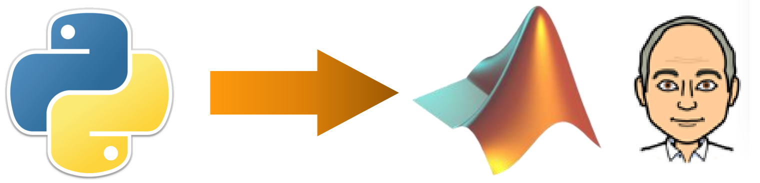
After some research, I've discovered that MATLAB needs to be able to find the correct version and install of python, but adding an anaconda environment adds some additional overhead that would need to be accounted for in the environment variables to allow MATLAB to find the python modules. It is probably doable, but probably more trouble than it is worth.

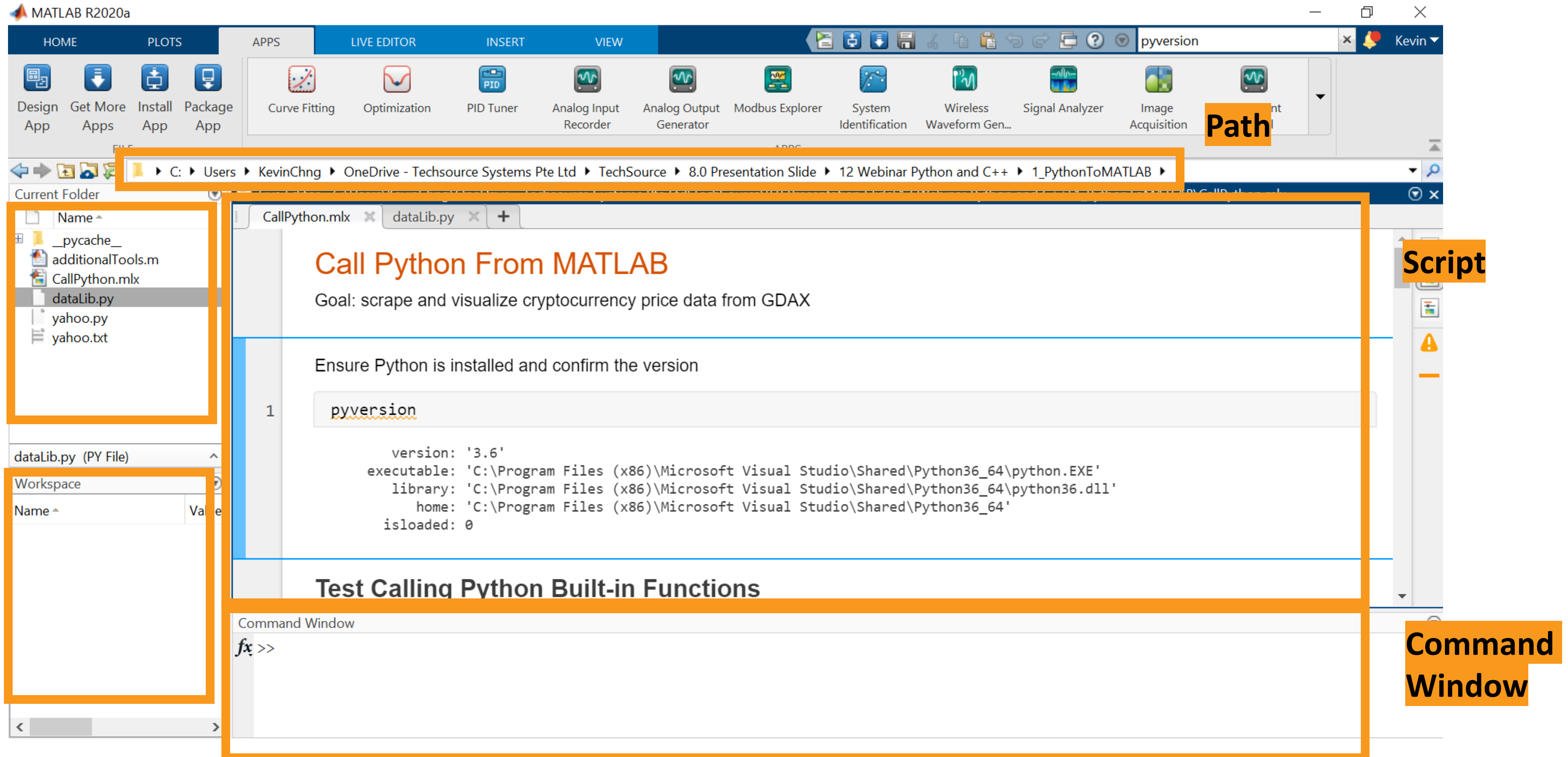
Instead, install python and the relevant libraries from the command line using pip and then point to that python version and folders.

```
pcPythonExe = 'C:\Users\dmattioli\AppData\Local\Programs\Python\Python37\python.exe';
[ver, exec, loaded]     = pyversion(pcPythonExe); pyversion
>> pyversion

       version: '3.7'
    executable: 'C:\Users\dmattioli\AppData\Local\Programs\Python\Python37\python.exe'
       library: 'C:\Users\dmattioli\AppData\Local\Programs\Python\Python37\python37.dll'
          home: 'C:\Users\dmattioli\AppData\Local\Programs\Python\Python37'
      isloaded: 1

% Add folders to python system path.
pyLibraryFolder = 'C:\Users\dmattioli\.PyCharm2019.1\system\python_stubs\278535617';
insert(py.sys.path, int64(0), pyLibraryFolder)
```

- **_Part 1_** : Calling Python Libraries from MATLAB

  - MATLAB Environment (For Python user to understand it)





Call Python From MATLAB

Goal: scrape and visualize cryptocurrency price data from GDAX

Ensure Python is installed and confirm the version

```
pyversion
```

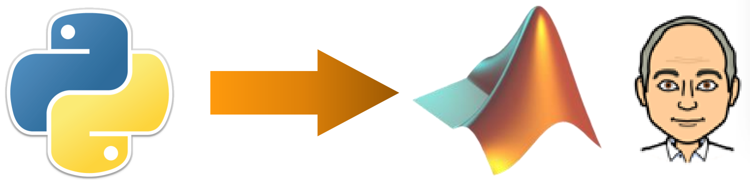
```
       version: '3.6'
    executable: 'C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64\python.EXE'
       library: 'C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64\python36.dll'
          home: 'C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64'
      isloaded: 0
```

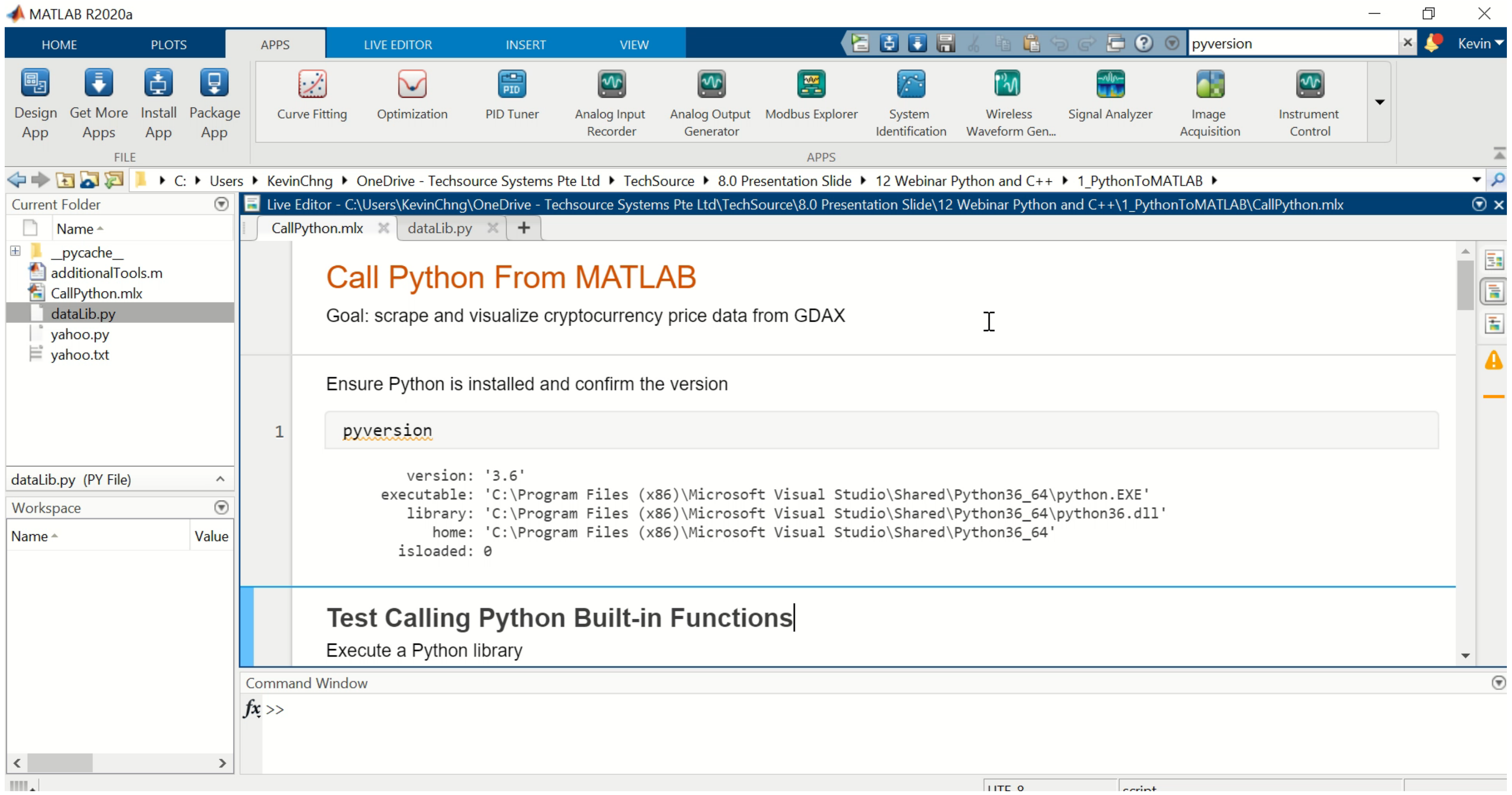
Test Calling Python Built-in Functions

```
fx >>
```

- **Part 1** : Calling Python Libraries from MATLAB

  - Let try the connection and try to call Python built-in function





Call Python From MATLAB

Goal: scrape and visualize cryptocurrency price data from GDAX

Ensure Python is installed and confirm the version

```
pyversion
```

```
       version: '3.6'
    executable: 'C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64\python.EXE'
       library: 'C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64\python36.dll'
          home: 'C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64'
      isloaded: 0
```

Test Calling Python Built-in Functions

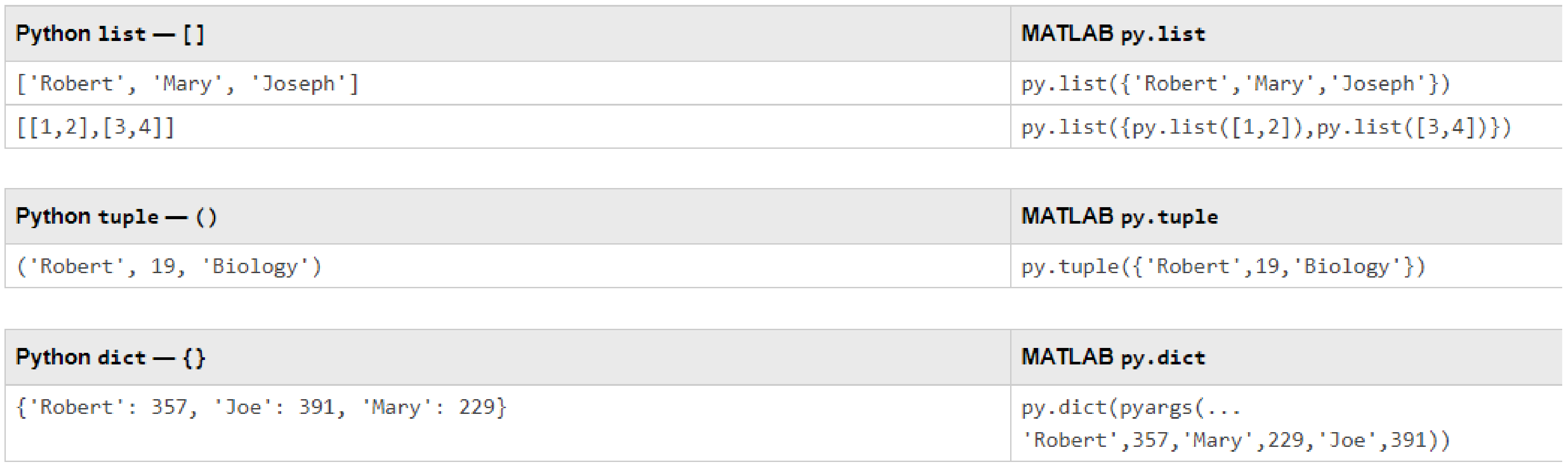
Execute a Python library

```
fx >>
```

Youtube : https://www.youtube.com/watch?v=IpSGlnjmuEg

• Part 1 : Calling Python Libraries from MATLAB

- Create Python Data Type in MATLAB



Python `list` — []	MATLAB `py.list`
`['Robert', 'Mary', 'Joseph']`	`py.list({'Robert','Mary','Joseph'})`
`[[1,2],[3,4]]`	`py.list({py.list([1,2]),py.list([3,4])})`

Python `tuple` — ()	MATLAB `py.tuple`
`('Robert', 19, 'Biology')`	`py.tuple({'Robert',19,'Biology'})`

Python `dict` — {}	MATLAB `py.dict`
`{'Robert': 357, 'Joe': 391, 'Mary': 229}`	`py.dict(pyargs(... 'Robert',357,'Mary',229,'Joe',391))`

https://www.mathworks.com/help/matlab/matlab_external/list-tuple-and-dictionary-types.html

Why do we need to learn create Python Data Type in MATLAB?

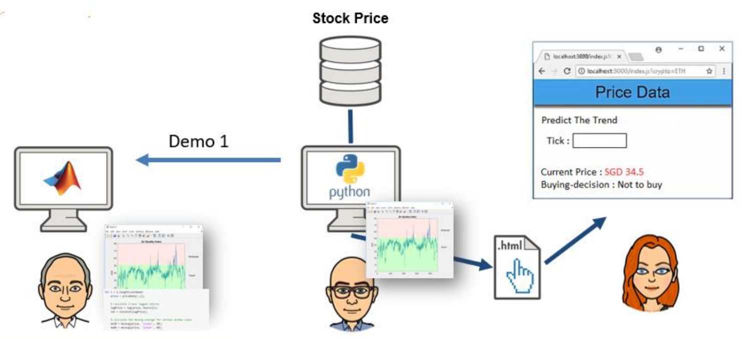
Default datatype mapping : https://www.mathworks.com/help/matlab/matlab_external/passing-data-to-python.html

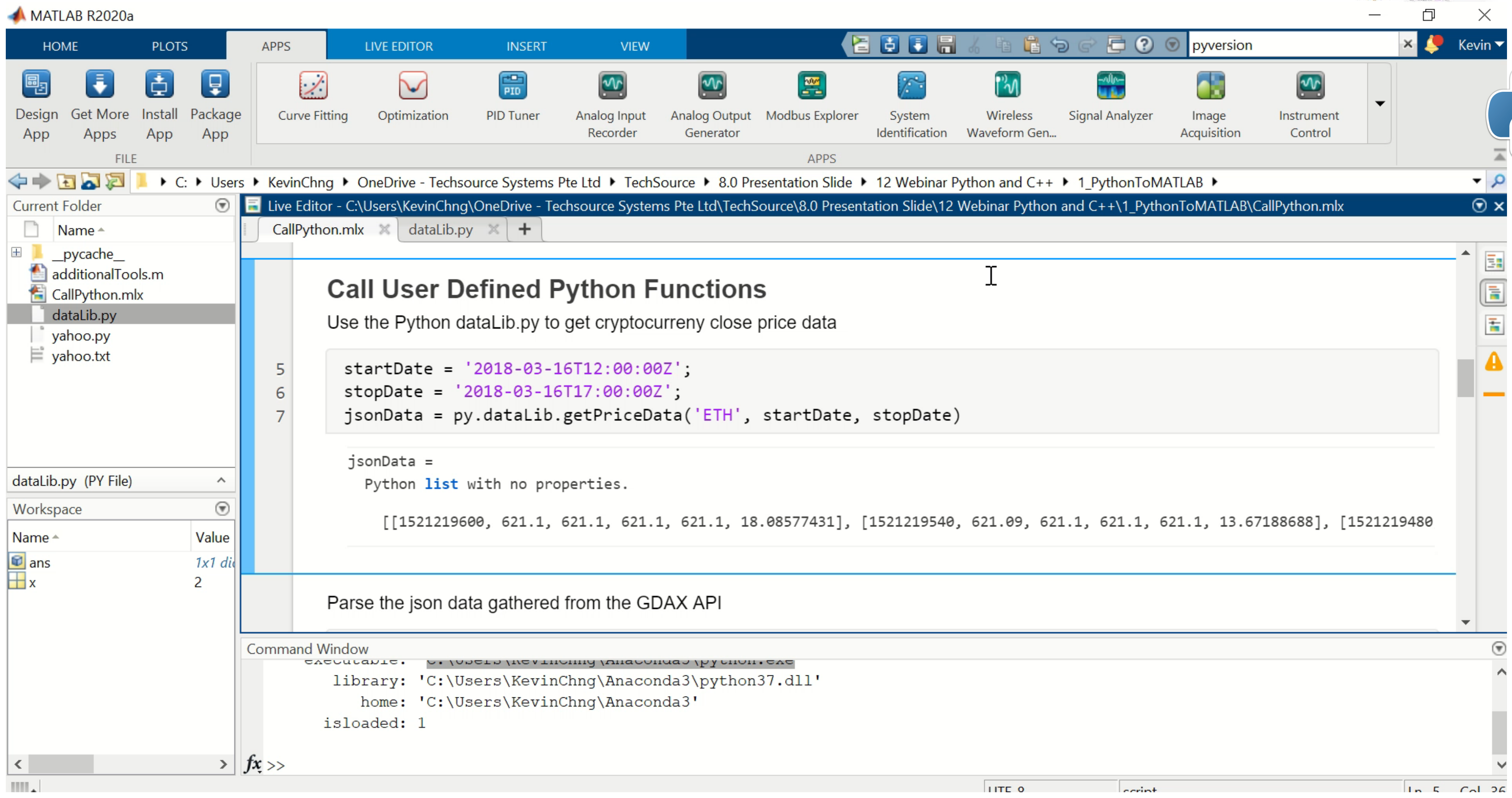
1) Pass appropriate data type to python script

2) Returned Result is in python data type when calling python libraries from MATLAB

- **Part 1** : Calling Python Libraries from MATLAB

  - Calling Python script from MATLAB (Demo 1)





Call User Defined Python Functions

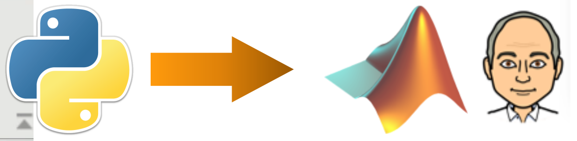
Use the Python dataLib.py to get cryptocurreny close price data

```
startDate = '2018-03-16T12:00:00Z';
stopDate = '2018-03-16T17:00:00Z';
jsonData = py.dataLib.getPriceData('ETH', startDate, stopDate)
```

```
jsonData =
  Python list with no properties.

    [[1521219600, 621.1, 621.1, 621.1, 621.1, 18.08577431], [1521219540, 621.09, 621.1, 621.1, 621.1, 13.67188688], [1521219480
```

Parse the json data gathered from the GDAX API

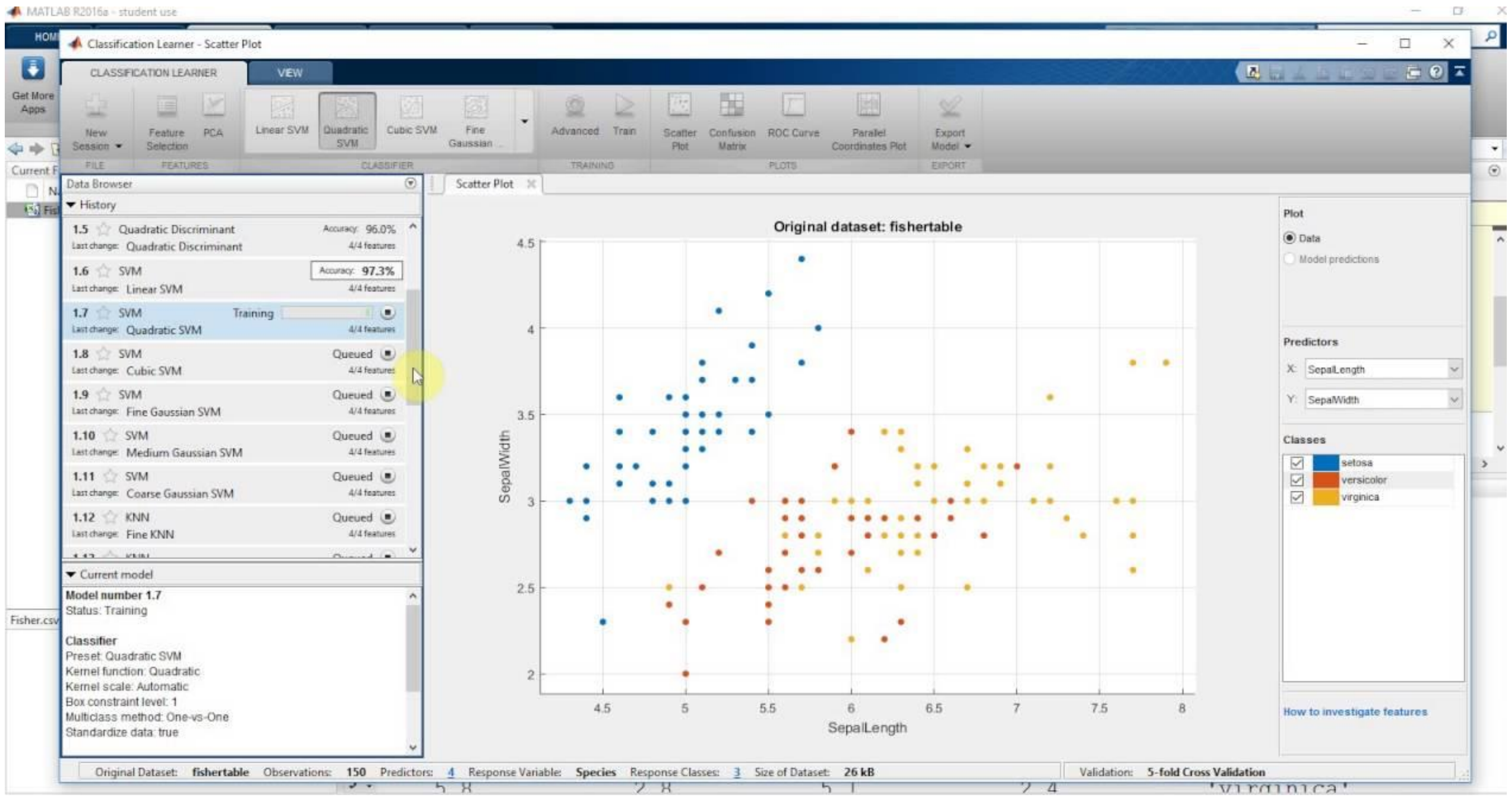
```
     library: 'C:\Users\KevinChng\Anaconda3\python37.dll'
        home: 'C:\Users\KevinChng\Anaconda3'
    isloaded: 1

fx >>
```



Youtube : https://www.youtube.com/watch?v=G7TFbzAwPBw

Build Model in MATLAB to predict the buying-decision for stock

Reference : https://www.mathworks.com/matlabcentral/fileexchange/68637-machine-learning-classification-used-to-predict-stock



History

1.5	Quadratic Discriminant	Last change: Quadratic Discriminant	Accuracy: 96.0%	4/4 features
1.6	SVM	Last change: Linear SVM	Accuracy: 97.3%	4/4 features
1.7	SVM	Last change: Quadratic SVM	Training	4/4 features
1.8	SVM	Last change: Cubic SVM	Queued	4/4 features
1.9	SVM	Last change: Fine Gaussian SVM	Queued	4/4 features
1.10	SVM	Last change: Medium Gaussian SVM	Queued	4/4 features
1.11	SVM	Last change: Coarse Gaussian SVM	Queued	4/4 features
1.12	KNN	Last change: Fine KNN	Queued	4/4 features

Current model

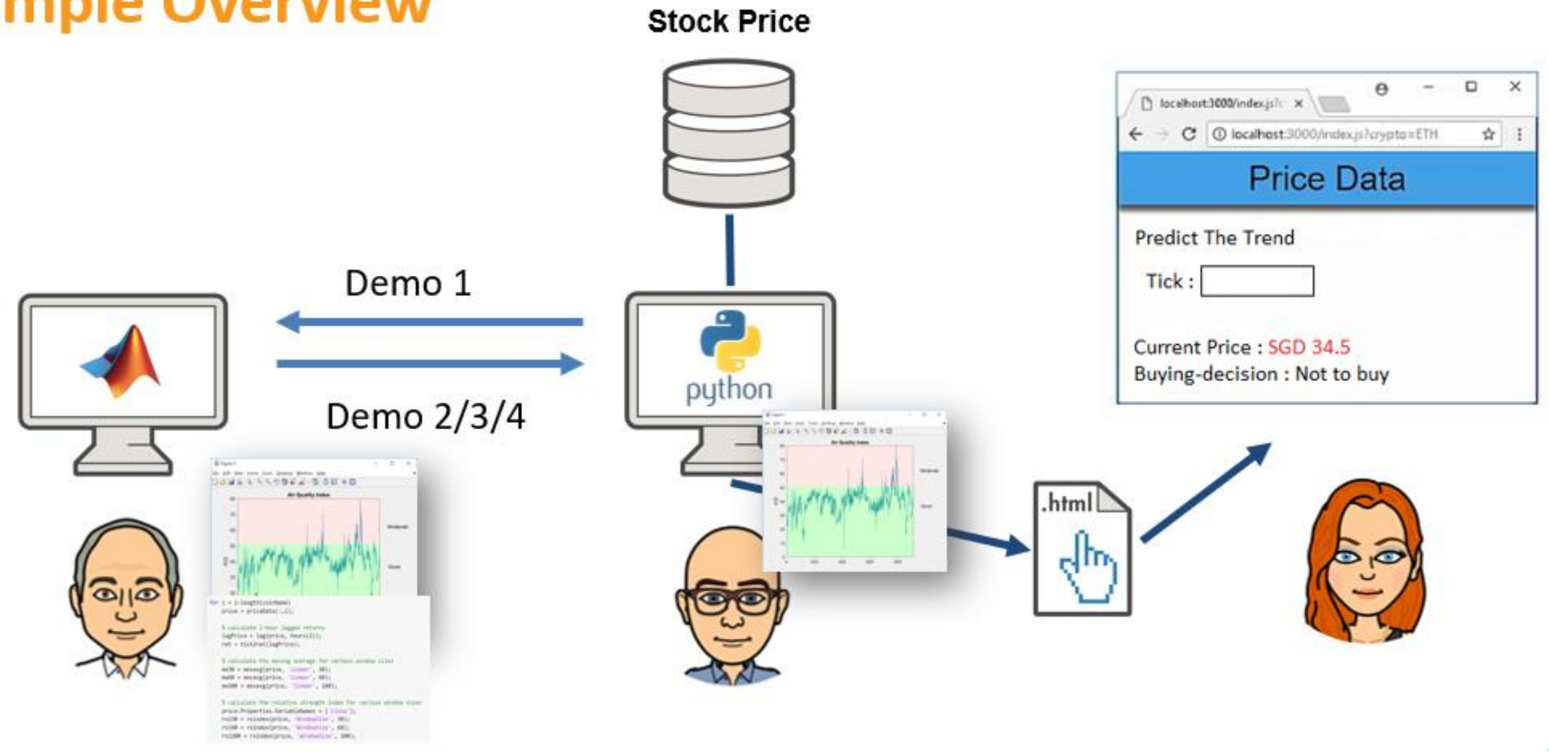
Model number 1.7
Status: Training

Classifier
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Original Dataset: **fishertable** Observations: **150** Predictors: 4 Response Variable: **Species** Response Classes: 3 Size of Dataset: **26 kB** Validation: **5-fold Cross Validation**

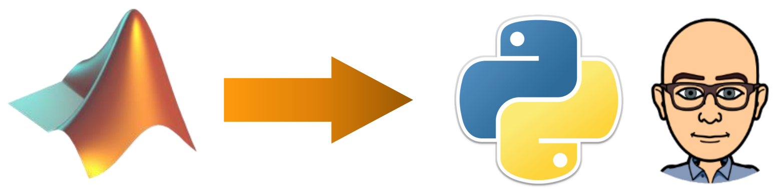
Regression Learner
Classification Learner
Deep Network Designer
Experiment Manager

Example Overview



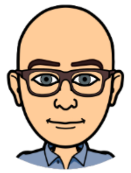
More information :

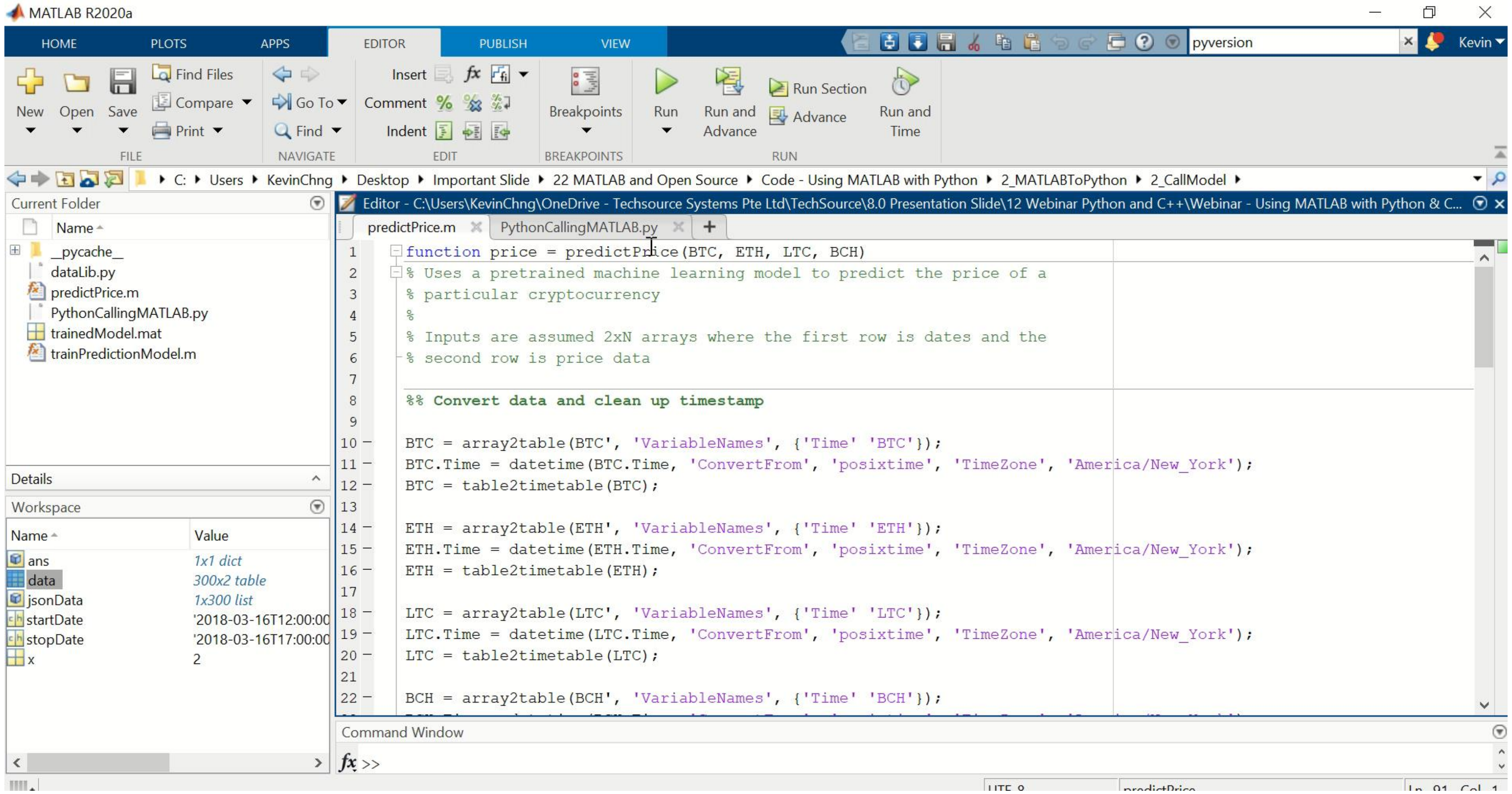
https://www.mathworks.com/help/matlab/matlab-engine-for-python.html?s_tid=CRUX_lftnav



- **Part 2 : Calling MATLAB from Python (via MATLAB Engine API)**

- **Part 2 : Calling MATLAB from Python** (via MATLAB Engine API)





MATLAB R2020a

Editor - C:\Users\KevinChng\OneDrive - Techsource Systems Pte Ltd\TechSource\8.0 Presentation Slide\12 Webinar Python and C++\Webinar - Using MATLAB with Python & C...

Current Folder:
- __pycache__
- dataLib.py
- predictPrice.m
- PythonCallingMATLAB.py
- trainedModel.mat
- trainPredictionModel.m

Workspace:

Name	Value
ans	1x1 dict
data	300x2 table
jsonData	1x300 list
startDate	'2018-03-16T12:00:00
stopDate	'2018-03-16T17:00:00
x	2

predictPrice.m

```matlab
function price = predictPrice(BTC, ETH, LTC, BCH)
% Uses a pretrained machine learning model to predict the price of a
% particular cryptocurrency
%
% Inputs are assumed 2xN arrays where the first row is dates and the
% second row is price data

%% Convert data and clean up timestamp

BTC = array2table(BTC', 'VariableNames', {'Time' 'BTC'});
BTC.Time = datetime(BTC.Time, 'ConvertFrom', 'posixtime', 'TimeZone', 'America/New_York');
BTC = table2timetable(BTC);

ETH = array2table(ETH', 'VariableNames', {'Time' 'ETH'});
ETH.Time = datetime(ETH.Time, 'ConvertFrom', 'posixtime', 'TimeZone', 'America/New_York');
ETH = table2timetable(ETH);

LTC = array2table(LTC', 'VariableNames', {'Time' 'LTC'});
LTC.Time = datetime(LTC.Time, 'ConvertFrom', 'posixtime', 'TimeZone', 'America/New_York');
LTC = table2timetable(LTC);

BCH = array2table(BCH', 'VariableNames', {'Time' 'BCH'});
```

Command Window

```
fx >>
```

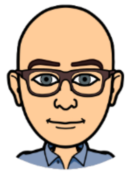
Install MATLAB API package in Python

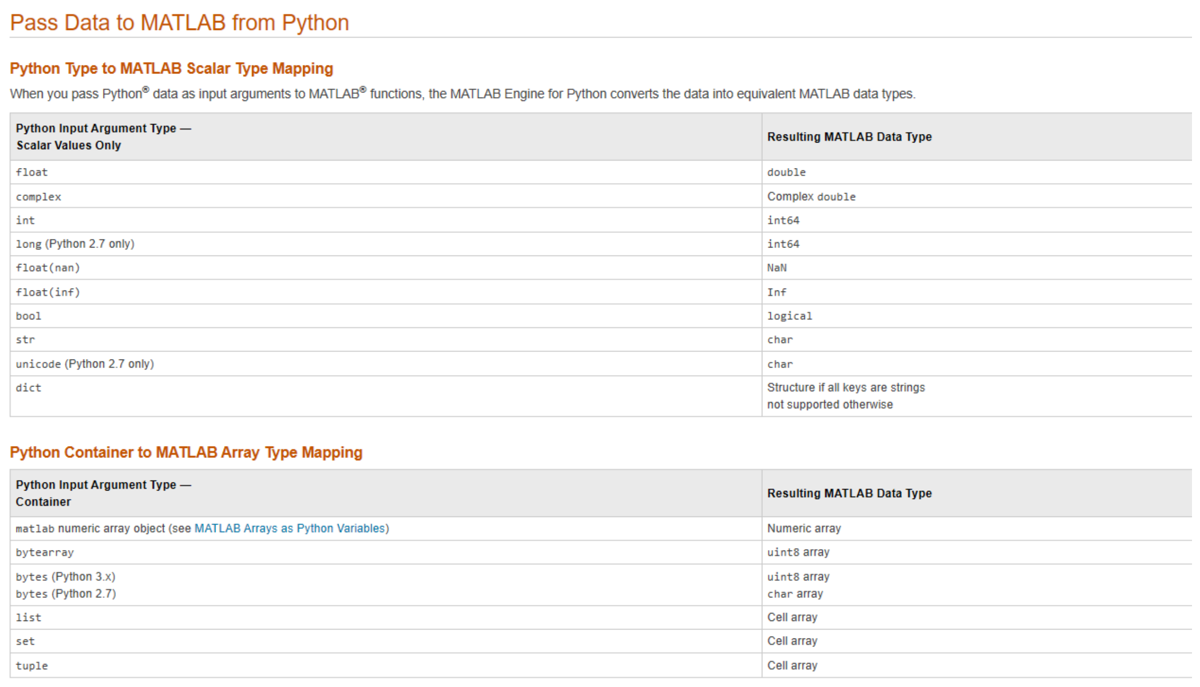
https://www.mathworks.com/help/matlab/matlab_external/install-the-matlab-engine-for-python.html

Youtube : https://www.youtube.com/watch?v=ois6PG5qfnc

• Part 2 : Calling MATLAB from Python (via MATLAB Engine API)

- Pass Data to MATLAB from Python
- Handle data Returned from Python
- Data Type Conversion !!





Pass Data to MATLAB from Python

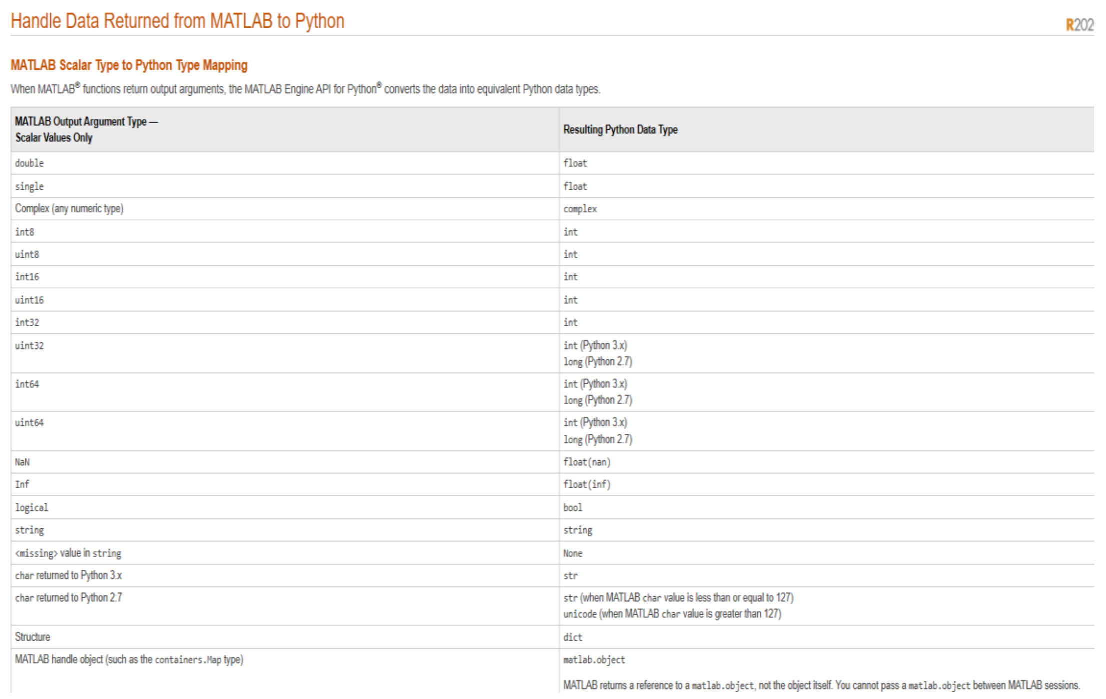
Python Type to MATLAB Scalar Type Mapping

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



Handle Data Returned from MATLAB to Python

R202

MATLAB Scalar Type to Python Type Mapping

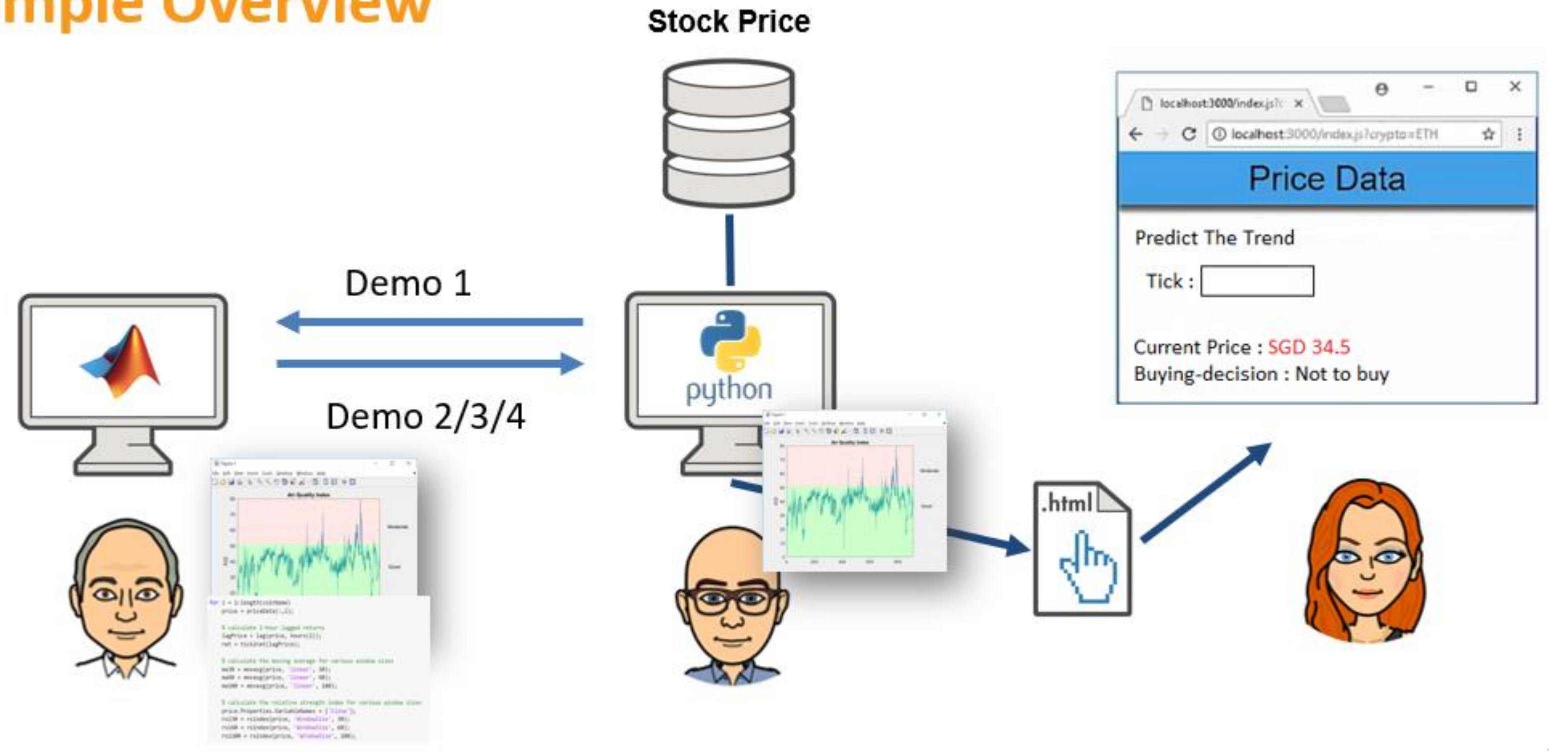
When MATLAB® functions return output arguments, the MATLAB Engine API for Python® converts the data into equivalent Python data types.

MATLAB Output Argument Type — Scalar Values Only	Resulting Python Data Type
double	float
single	float
Complex (any numeric type)	complex
int8	int
uint8	int
int16	int
uint16	int
int32	int
uint32	int (Python 3.x) long (Python 2.7)
int64	int (Python 3.x) long (Python 2.7)
uint64	int (Python 3.x) long (Python 2.7)
NaN	float(nan)
Inf	float(inf)
logical	bool
string	string
<missing> value in string	None
char returned to Python 3.x	str
char returned to Python 2.7	str (when MATLAB char value is less than or equal to 127) unicode (when MATLAB char value is greater than 127)
Structure	dict
MATLAB handle object (such as the containers.Map type)	matlab.object MATLAB returns a reference to a matlab.object, not the object itself. You cannot pass a matlab.object between MATLAB sessions.

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

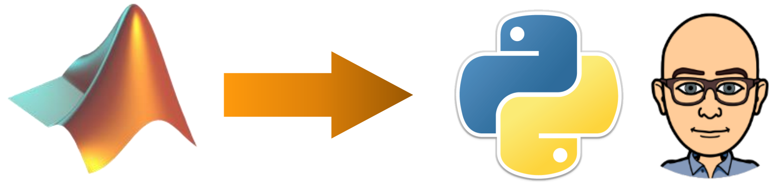
https://www.mathworks.com/help/matlab/matlab_external/handle-data-returned-from-matlab-to-python.html

Example Overview



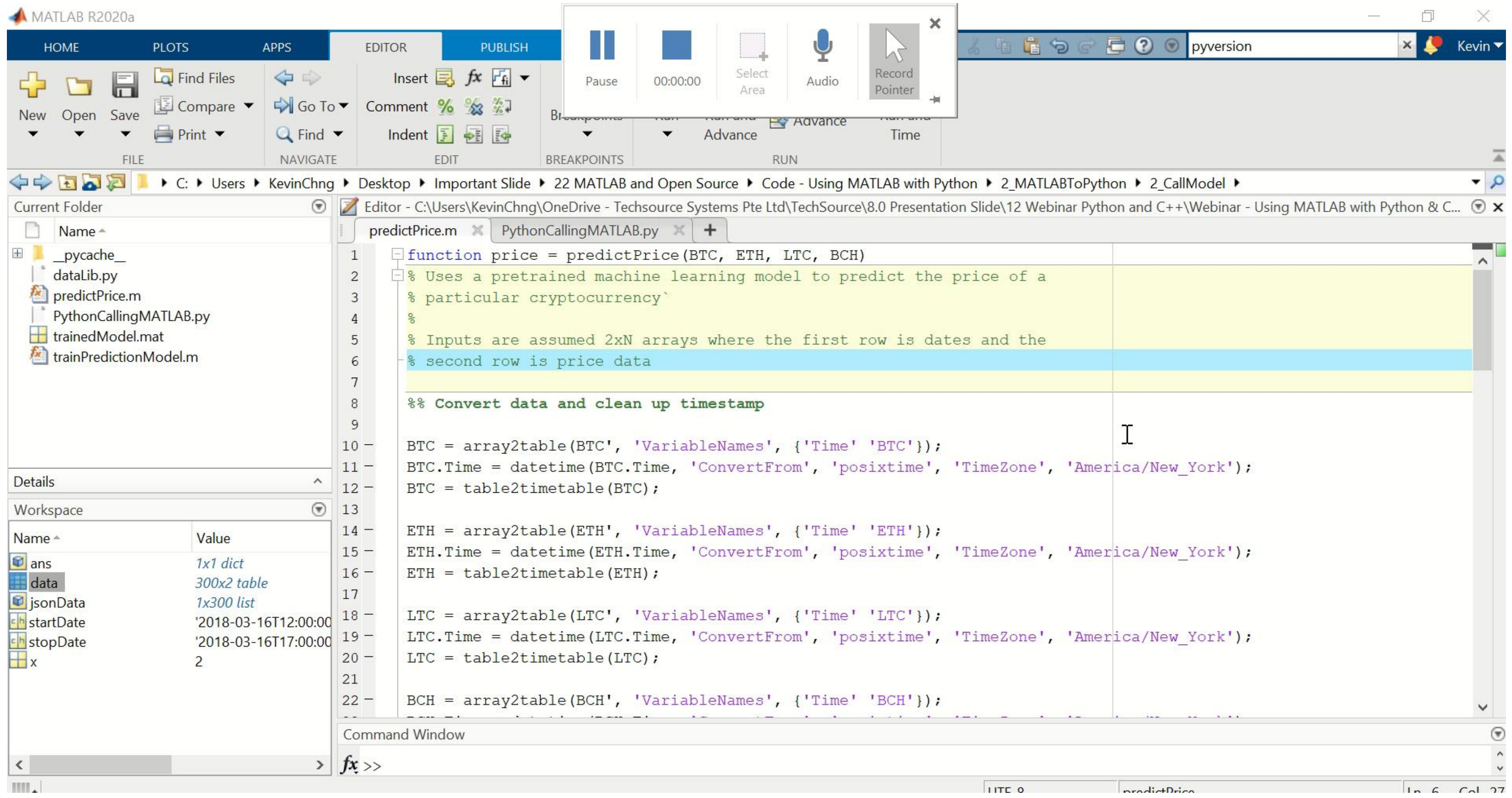
More information :

https://www.mathworks.com/help/compiler_sdk/python_packages.html



- **Part 2 : Calling MATLAB from Python** **(via MATLAB Runtime (MATLAB Compiler SDK)**

- **Part 2 : Calling MATLAB from Python** (via MATLAB Compiler SDK)
  - Compile MATLAB script to Python Packages


MATLAB R2020a
HOME
PLOTS
APPS
EDITOR
PUBLISH
Pause
00:00:00
Select Area
Audio
Record Pointer
pyversion
Kevin
New
Open
Save
Find Files
Compare
Print
FILE
Go To
Find
NAVIGATE
Insert
Comment
Indent
EDIT
BREAKPOINTS
Advance
RUN
Time
C: ▸ Users ▸ KevinChng ▸ Desktop ▸ Important Slide ▸ 22 MATLAB and Open Source ▸ Code - Using MATLAB with Python ▸ 2_MATLABToPython ▸ 2_CallModel ▸
Current Folder
Editor - C:\Users\KevinChng\OneDrive - Techsource Systems Pte Ltd\TechSource\8.0 Presentation Slide\12 Webinar Python and C++\Webinar - Using MATLAB with Python & C...
predictPrice.m
PythonCallingMATLAB.py
Command Window
Details
Workspace
UTF-8
predictPrice
Ln 6
Col 27

Current Folder:

- _pycache_
- dataLib.py
- predictPrice.m
- PythonCallingMATLAB.py
- trainedModel.mat
- trainPredictionModel.m

```matlab
function price = predictPrice(BTC, ETH, LTC, BCH)
% Uses a pretrained machine learning model to predict the price of a
% particular cryptocurrency`
%
% Inputs are assumed 2xN arrays where the first row is dates and the
% second row is price data

%% Convert data and clean up timestamp

BTC = array2table(BTC', 'VariableNames', {'Time' 'BTC'});
BTC.Time = datetime(BTC.Time, 'ConvertFrom', 'posixtime', 'TimeZone', 'America/New_York');
BTC = table2timetable(BTC);

ETH = array2table(ETH', 'VariableNames', {'Time' 'ETH'});
ETH.Time = datetime(ETH.Time, 'ConvertFrom', 'posixtime', 'TimeZone', 'America/New_York');
ETH = table2timetable(ETH);

LTC = array2table(LTC', 'VariableNames', {'Time' 'LTC'});
LTC.Time = datetime(LTC.Time, 'ConvertFrom', 'posixtime', 'TimeZone', 'America/New_York');
LTC = table2timetable(LTC);

BCH = array2table(BCH', 'VariableNames', {'Time' 'BCH'});
```

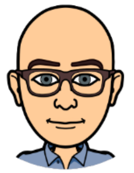
Workspace:

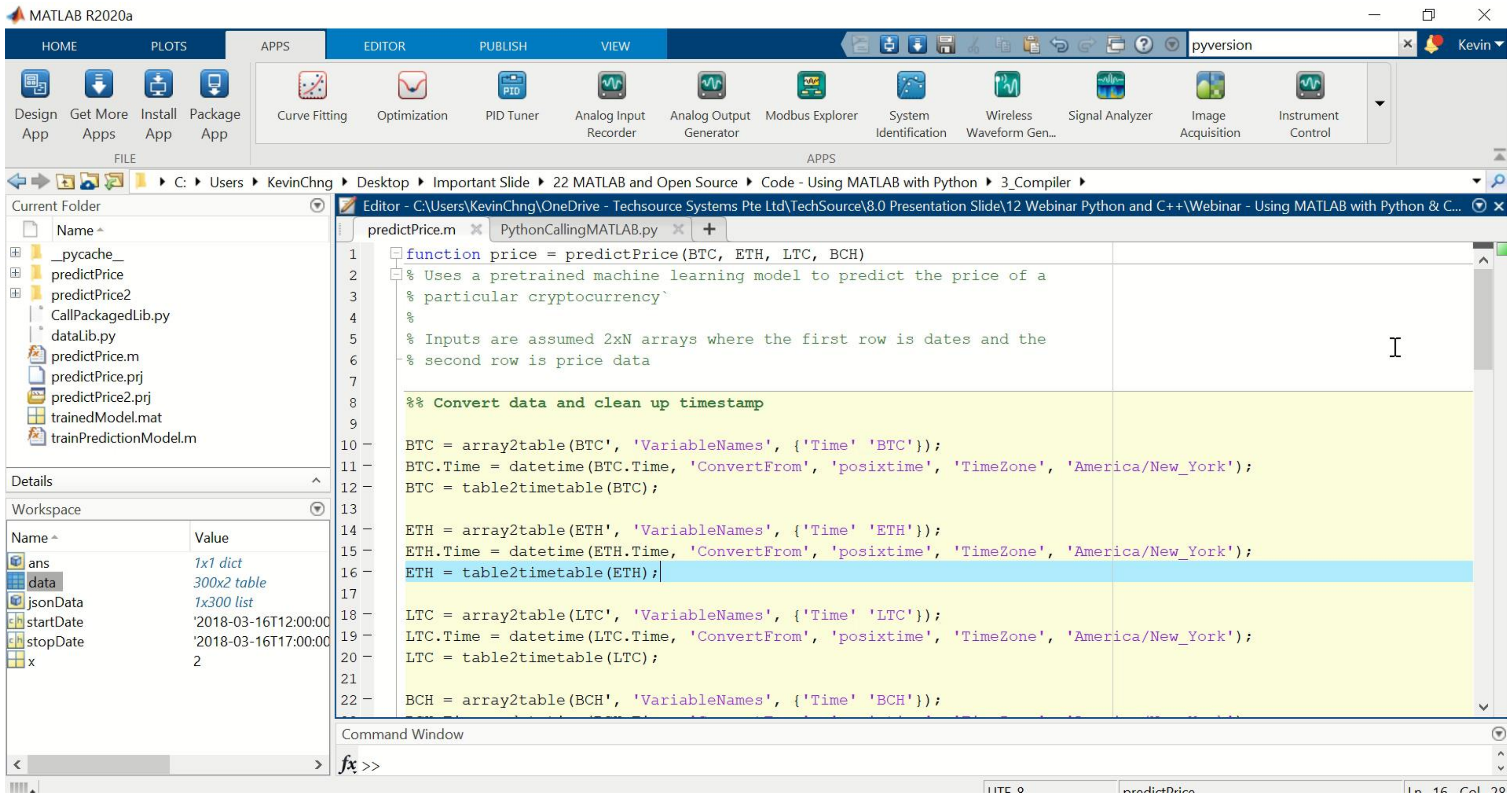
Name	Value
ans	1x1 dict
data	300x2 table
jsonData	1x300 list
startDate	'2018-03-16T12:00:00
stopDate	'2018-03-16T17:00:00
x	2

```
fx >>
```

Youtube : https://www.youtube.com/watch?v=LtdwcAviUN0

- **Part 2** : **Calling MATLAB from Python** (via MATLAB Compiler SDK)





Current Folder

Name
pycache
predictPrice
predictPrice2
CallPackagedLib.py
dataLib.py
predictPrice.m
predictPrice.prj
predictPrice2.prj
trainedModel.mat
trainPredictionModel.m

Workspace

Name	Value
ans	1x1 dict
data	300x2 table
jsonData	1x300 list
startDate	'2018-03-16T12:00:00
stopDate	'2018-03-16T17:00:00
x	2

```
function price = predictPrice(BTC, ETH, LTC, BCH)
% Uses a pretrained machine learning model to predict the price of a
% particular cryptocurrency`
%
% Inputs are assumed 2xN arrays where the first row is dates and the
% second row is price data

%% Convert data and clean up timestamp

BTC = array2table(BTC', 'VariableNames', {'Time' 'BTC'});
BTC.Time = datetime(BTC.Time, 'ConvertFrom', 'posixtime', 'TimeZone', 'America/New_York');
BTC = table2timetable(BTC);

ETH = array2table(ETH', 'VariableNames', {'Time' 'ETH'});
ETH.Time = datetime(ETH.Time, 'ConvertFrom', 'posixtime', 'TimeZone', 'America/New_York');
ETH = table2timetable(ETH);

LTC = array2table(LTC', 'VariableNames', {'Time' 'LTC'});
LTC.Time = datetime(LTC.Time, 'ConvertFrom', 'posixtime', 'TimeZone', 'America/New_York');
LTC = table2timetable(LTC);

BCH = array2table(BCH', 'VariableNames', {'Time' 'BCH'});
```

```
fx >>
```

Install MATLAB Compiled SDK Python Package :

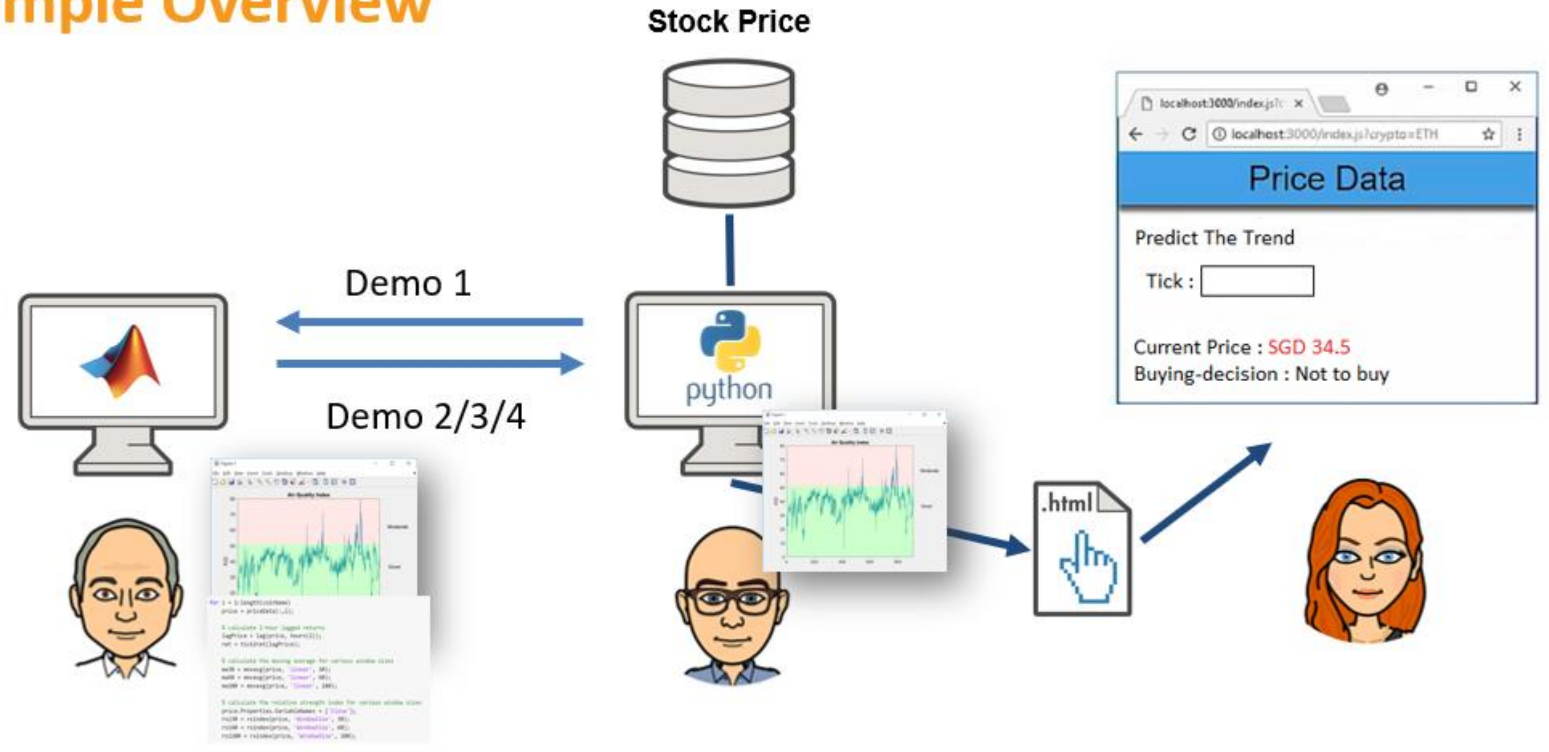
https://www.mathworks.com/help/compiler_sdk/python/install-a-matlab-compiler-sdk-python-package.html

Install MATLAB Runtime :

https://www.mathworks.com/products/compiler/matlab-runtime.html

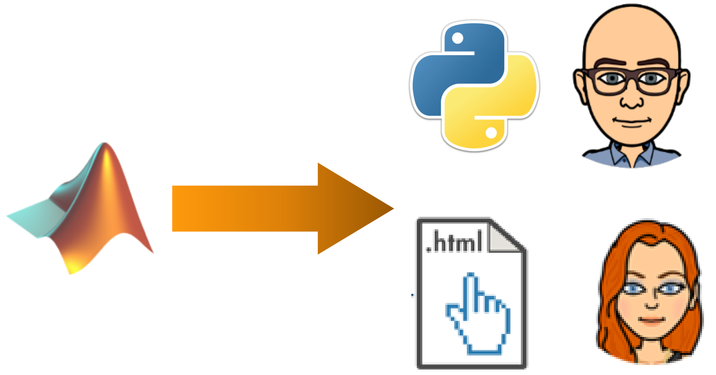
Youtube :
https://www.youtube.com/watch?v=QcBlViM32C8

Example Overview



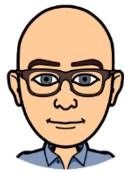
More information :

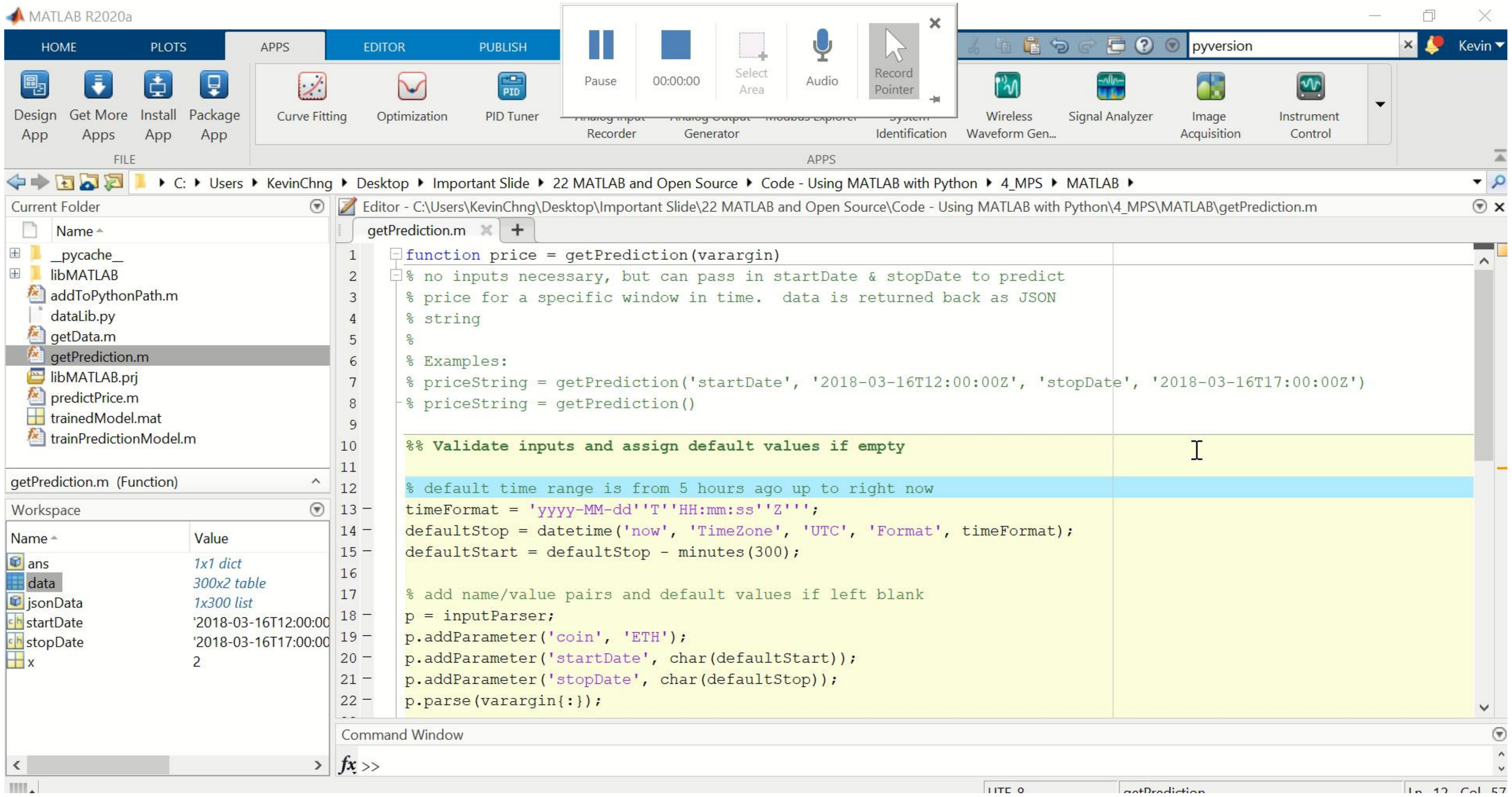
https://www.mathworks.com/help/compiler_sdk/python_packages.html



- **Part 2 : Calling MATLAB from Python (via MATLAB Production Server)**

- **Part 2** : **Calling MATLAB from Python** (via MATLAB Production Server)





C: ▸ Users ▸ KevinChng ▸ Desktop ▸ Important Slide ▸ 22 MATLAB and Open Source ▸ Code - Using MATLAB with Python ▸ 4_MPS ▸ MATLAB ▸

Editor - C:\Users\KevinChng\Desktop\Important Slide\22 MATLAB and Open Source\Code - Using MATLAB with Python\4_MPS\MATLAB\getPrediction.m

- _pycache_
- libMATLAB
- addToPythonPath.m
- dataLib.py
- getData.m
- getPrediction.m
- libMATLAB.prj
- predictPrice.m
- trainedModel.mat
- trainPredictionModel.m

```matlab
function price = getPrediction(varargin)
% no inputs necessary, but can pass in startDate & stopDate to predict
% price for a specific window in time.  data is returned back as JSON
% string
%
% Examples:
% priceString = getPrediction('startDate', '2018-03-16T12:00:00Z', 'stopDate', '2018-03-16T17:00:00Z')
% priceString = getPrediction()

%% Validate inputs and assign default values if empty

% default time range is from 5 hours ago up to right now
timeFormat = 'yyyy-MM-dd''T''HH:mm:ss''Z''';
defaultStop = datetime('now', 'TimeZone', 'UTC', 'Format', timeFormat);
defaultStart = defaultStop - minutes(300);

% add name/value pairs and default values if left blank
p = inputParser;
p.addParameter('coin', 'ETH');
p.addParameter('startDate', char(defaultStart));
p.addParameter('stopDate', char(defaultStop));
p.parse(varargin{:});
```

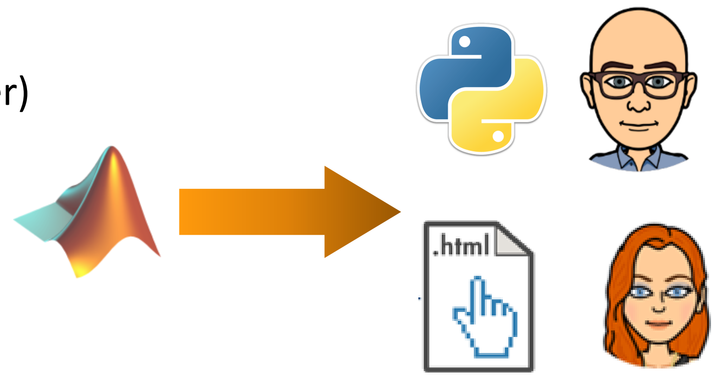
Name	Value
ans	1x1 dict
data	300x2 table
jsonData	1x300 list
startDate	'2018-03-16T12:00:00
stopDate	'2018-03-16T17:00:00
x	2

```
fx >>
```

Youtube : https://www.youtube.com/watch?v=OgYVzVnuR0k

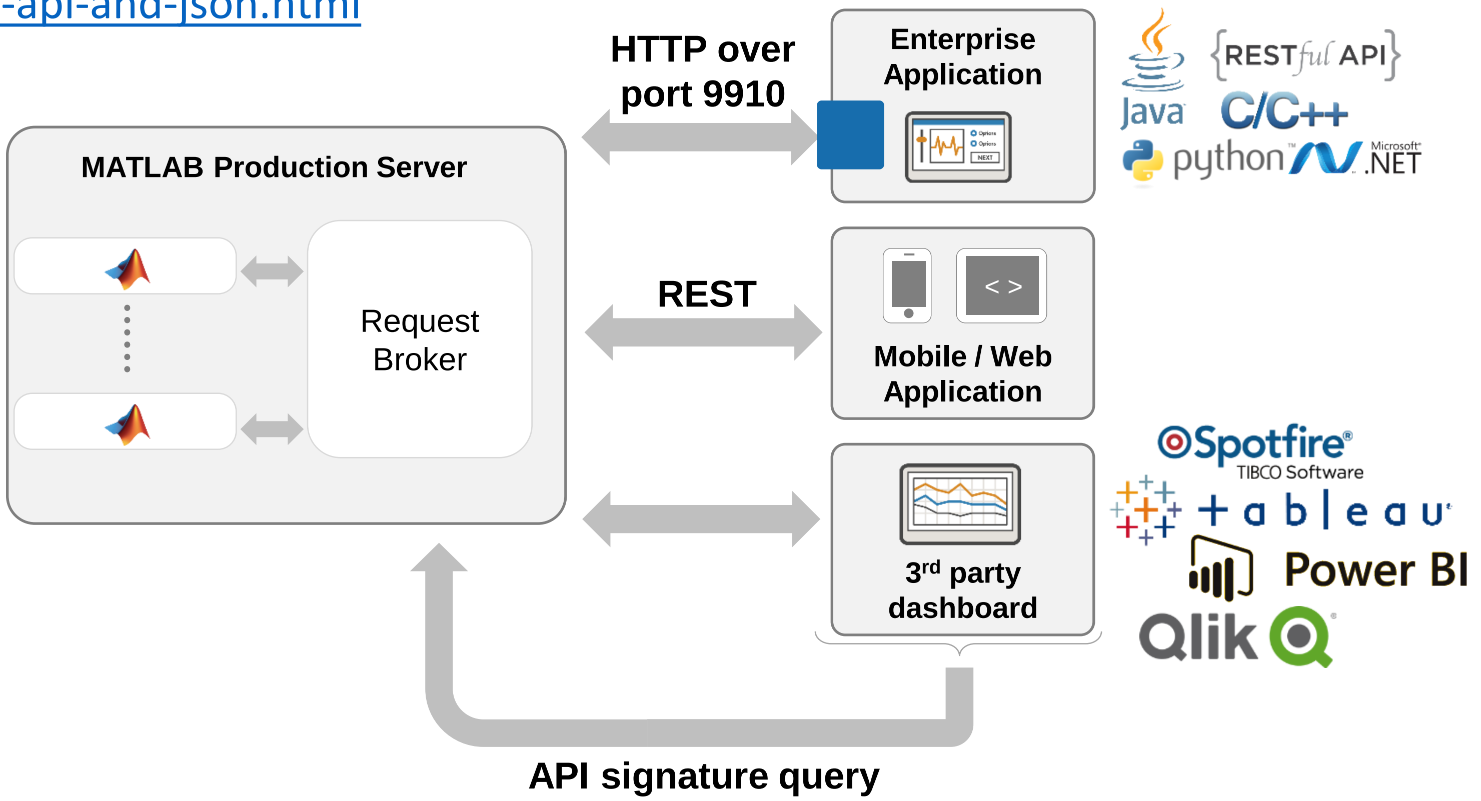
• Part 2 : Calling MATLAB from Python (via MATLAB Production Server)

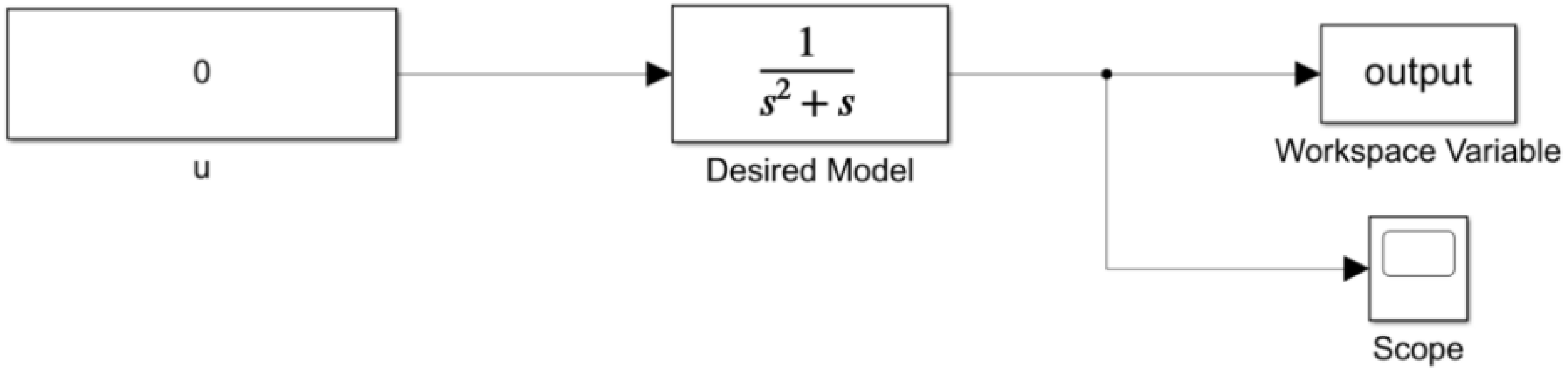
MATLAB Production Server manages multiple MATLAB runtime versions simultaneously

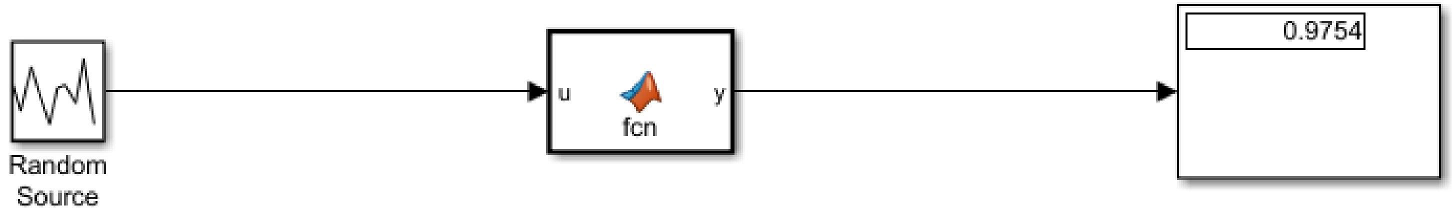


- Restful Api
  https://www.mathworks.com/help/mps/restful-api-and-json.html

- Python Client
  https://www.mathworks.com/help/mps/python-client-programming.html

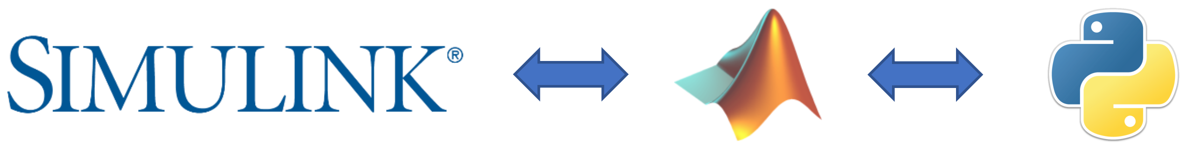






- **Part 3 : Simulink & Python**
    - Call Python from Simulink
    - Call Simulink from Python

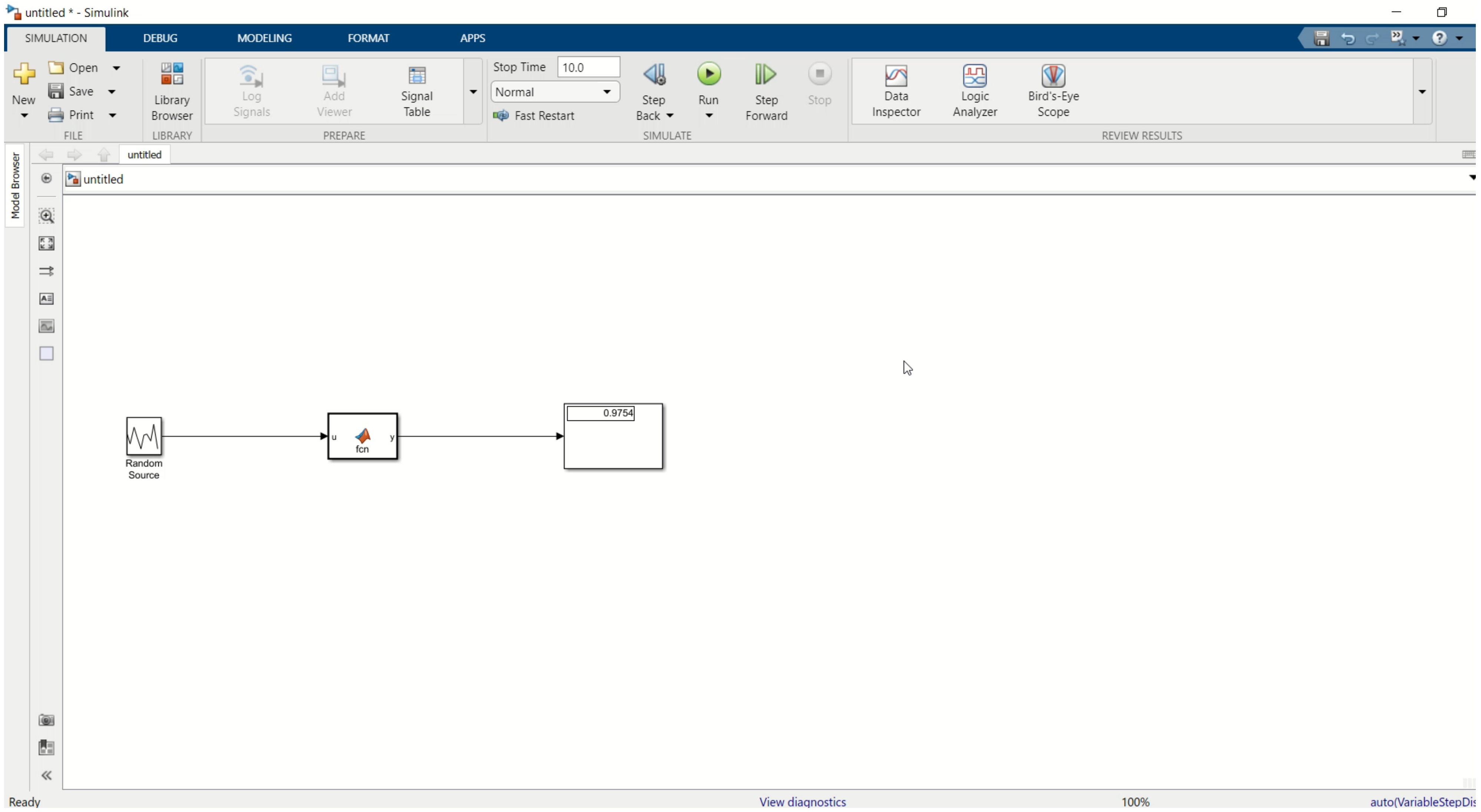
• **Part 3** : Call Python from Simulink



You could try using ***a MATLAB function block that contains MATLAB code to call the Python code***.
This documentation link provides an example of how to integrate a MATLAB function block into a Simulink model,
https://www.mathworks.com/help/simulink/ug/example-calculating-statistical-mean-and-standard-deviation.html

and this documentation link contains useful information on how to use Python code with MATLAB:
https://www.mathworks.com/help/matlab/getting-started-with-python.html

One thing to note is that *not all MATLAB functionalities will be supported for code generation*
so you may need to use 'coder.extrinsic' in the MATLAB function block.
Additional information on 'coder.extrinsic' can be found here:
https://www.mathworks.com/help/simulink/slref/coder.extrinsic.html

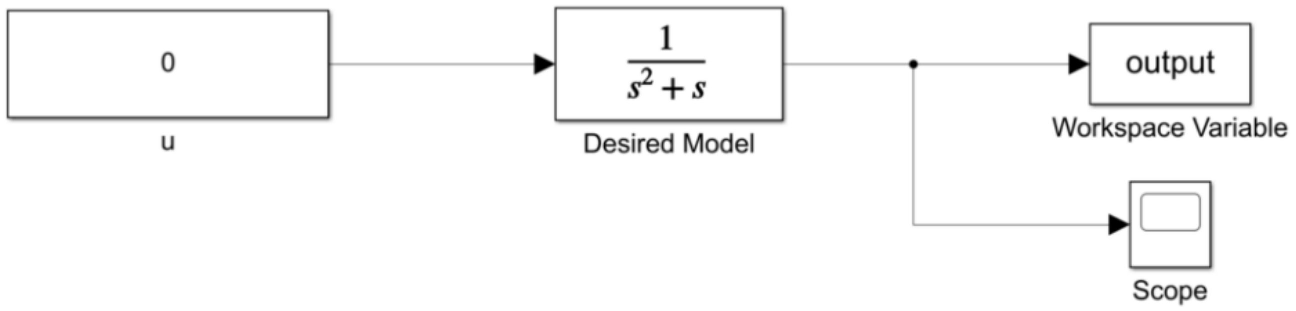


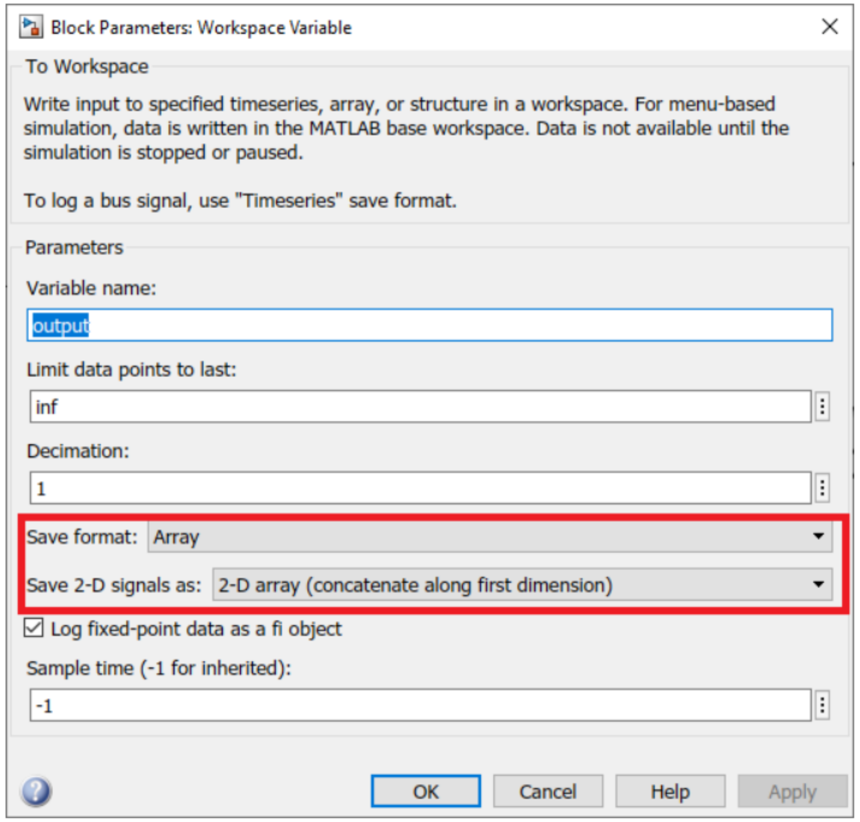
Youtube : https://youtu.be/6Z6I5zgSJ7Y

- **Part 3** : Call Simulink from Python

Example from Soutrik Bandyopadhyay

https://medium.com/@soutrikbandyopadhyay/controlling-a-simulink-model-by-a-python-controller-2b67bde744ee




Block Parameters: Workspace Variable

To Workspace

Write input to specified timeseries, array, or structure in a workspace. For menu-based simulation, data is written in the MATLAB base workspace. Data is not available until the simulation is stopped or paused.

To log a bus signal, use "Timeseries" save format.

Parameters

Variable name: output

Limit data points to last: inf

Decimation: 1

Save format: Array

Save 2-D signals as: 2-D array (concatenate along first dimension)

[✓] Log fixed-point data as a fi object

Sample time (-1 for inherited): -1

Properties for the "To Workspace" block

The Python Code

Importing the Necessary Libraries

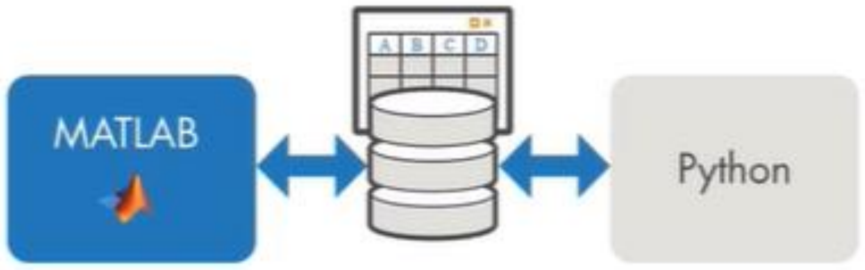
```
import matlab.engine
import matplotlib.pyplot as plt
```

```
#Load the model
self.eng.eval("model = '{}'".format(self.modelName),nargout=0)
self.eng.eval("load_system(model)",nargout=0)
```

```
#Start Simulation and then Instantly pause
self.eng.set_param(self.modelName,'SimulationCommand','start','SimulationCommand','pause',nargout=0)
self.yHist,self.tHist = self.getHistory()
```

- **Part 4 : Additional info**
  - Data management
  - Integration with Anaconda
  - Troubleshooting & Resources

- Use Apache Parquet to store and transfer data between MATLAB and python
- Working with Parquet files : https://www.mathworks.com/help/matlab/parquet-files.html
- MATLAB Library for Apache Arrow on Github : https://github.com/apache/arrow/tree/master/matlab

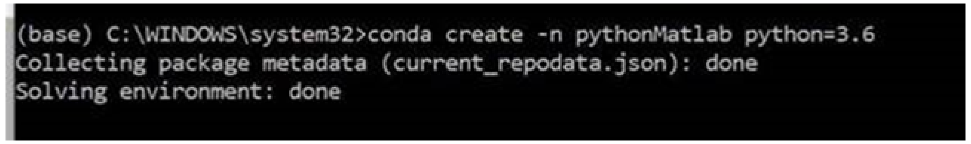


• Part 4 : Additional info

- Data management
- Integration with Anaconda
- Troubleshooting & Resources

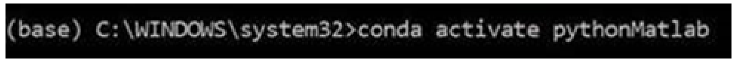
- How to call MATLAB engine from Anaconda? Jupyter notebook, spyder
- Setup besides is how to install MATLAB engine(MATLAB API) in Anaconda
- For MATLAB compiled python package, you can modify step 5 and step 6 to install the MATLAB compiled python package.

1) Open anaconda prompt
2) We create new conda environment with python = 3.6



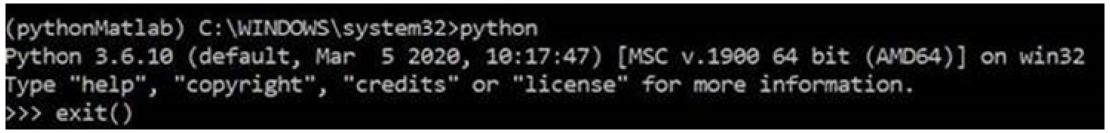
```
(base) C:\WINDOWS\system32>conda create -n pythonMatlab python=3.6
Collecting package metadata (current_repodata.json): done
Solving environment: done
```

3) Activate the conda environment



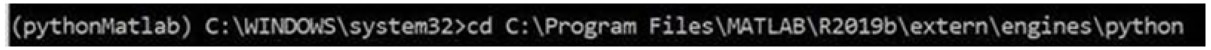
```
(base) C:\WINDOWS\system32>conda activate pythonMatlab
```

4) Now the conda environment (pythonMatlab) is activated and type python to double confirm the version



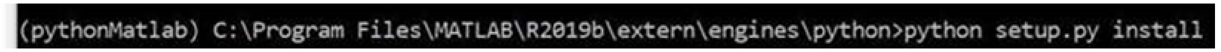
```
(pythonMatlab) C:\WINDOWS\system32>python
Python 3.6.10 (default, Mar  5 2020, 10:17:47) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> exit()
```

5) Now, cd to MATLAB engine path



```
(pythonMatlab) C:\WINDOWS\system32>cd C:\Program Files\MATLAB\R2019b\extern\engines\python
```

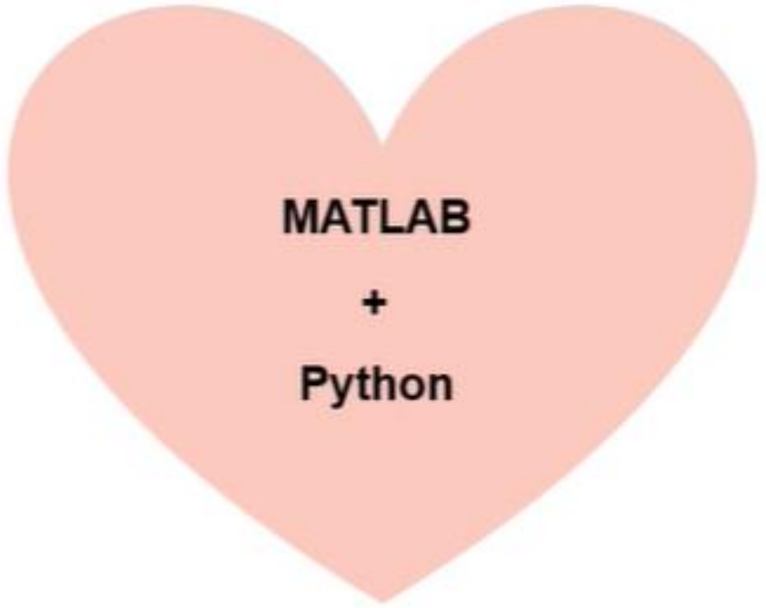
6) Last, install the MATLAB engine



```
(pythonMatlab) C:\Program Files\MATLAB\R2019b\extern\engines\python>python setup.py install
```

Summary

- Example Overview
- ***Part 1*** : Calling Python Libraries from MATLAB
- ***Part 2*** : Calling MATLAB from Python
  - via MATLAB Engine API
  - via MATLAB Runtime (MATLAB Compiler SDK)
  - via MATLAB Production Server
- **Part 3** : Simulink and Python
- **Part 4** : Additional info
  - Data management
  - Integration with Anaconda
  - Troubleshooting & Resources



- **Part 4 : Additional info**

  - Data management
  - Integration with Anaconda
  - Troubleshooting & Resources : Check Documentation & Call Our Tech Support & MATLAB Answer

Ascendas Solutions: Tools & Support

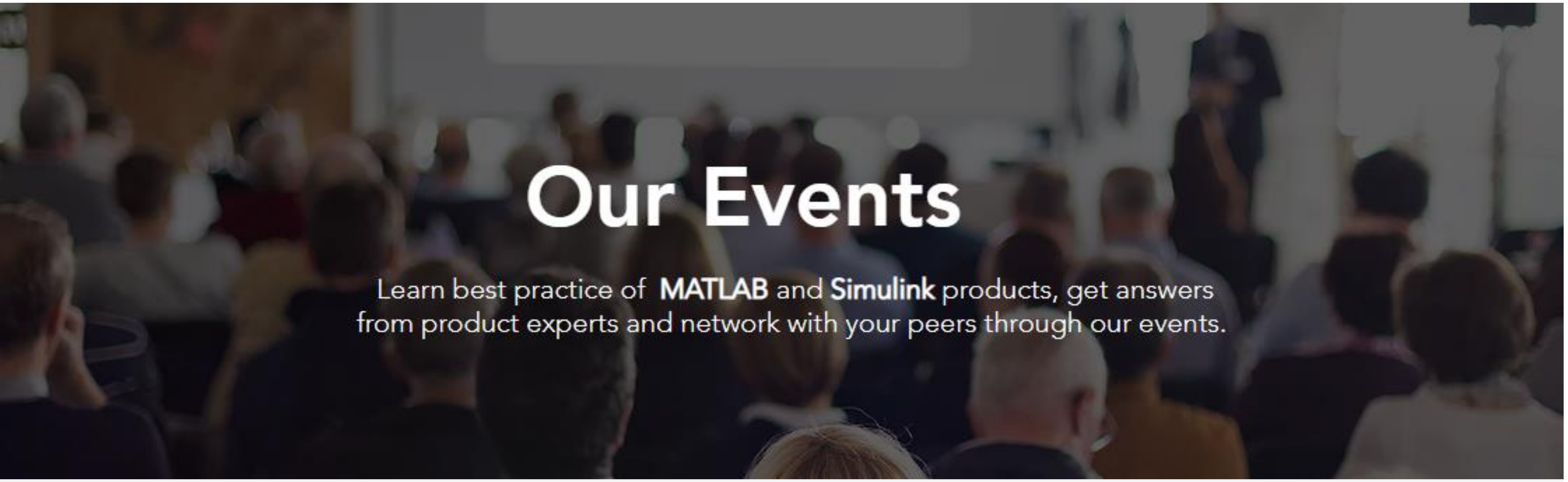
Training Options & Recommendations

Free self-paced, introductory tutorials:

MATLAB Onramp	hands-on tutorial
Deep Learning Onramp	hands-on tutorial
Simulink Onramp	hands-on tutorial
Stateflow Onramp	hands-on tutorial
…	

- **Onsite, hands-on Workshops**
- **Formal Training classes: public, online/onsite**
  - **Public Class**
  - **Online Training**
  - **Custom Onsite Training**

More Upcoming Events & Training, Register at www.techsource-asia.com/events



Our Events
Learn best practice of MATLAB and Simulink products, get answers from product experts and network with your peers through our events.

- *Data Science*
- *Wireless*
- *Power Electronics Control Design*
- *Predictive Maintenance*
- *Embedded Coder, FPGA*
- *Robotics*
- *Signal Processing*
- *Image Processing & Computer Vision*
- *Automated Driving*
- *Finance*

TECHSOURCE ACADEMY

Southeast Asia's sole distributor of MATLAB® & SIMULINK®

Ascendas Solutions: Tools & Support

Consulting Services

- **Technical expertise**
- **Deep product knowledge**
- **Extensive resource access**
- **Broad industry perspective**
- **Customer focus**
- **Ability to work onsite**

- **Innovation**
- **Reduced costs**
- **Faster results**
- **Improved quality**
- **Higher efficiency**

Service Offerings: Get started quickly and effectively with a MathWorks product

Advisory Services: Ongoing, detailed support

- Enterprise Integration and Support
- Release Migration
- Process Audit and Industry best practice
- Software development for custom Apps
- Cloud and edge computing

Special Projects: Extend and customize MathWorks tools

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

Facebook

LinkedIn

ASCENDAS SYSTEMS

Q&A

Feedback

Please scan the QR code to complete a survey so that we can continue to improve our future webinar.

ASCENDAS SYSTEMS

Thank you

MathWorks®

Southeast Asia's sole distributor of

MATLAB® & SIMULINK®