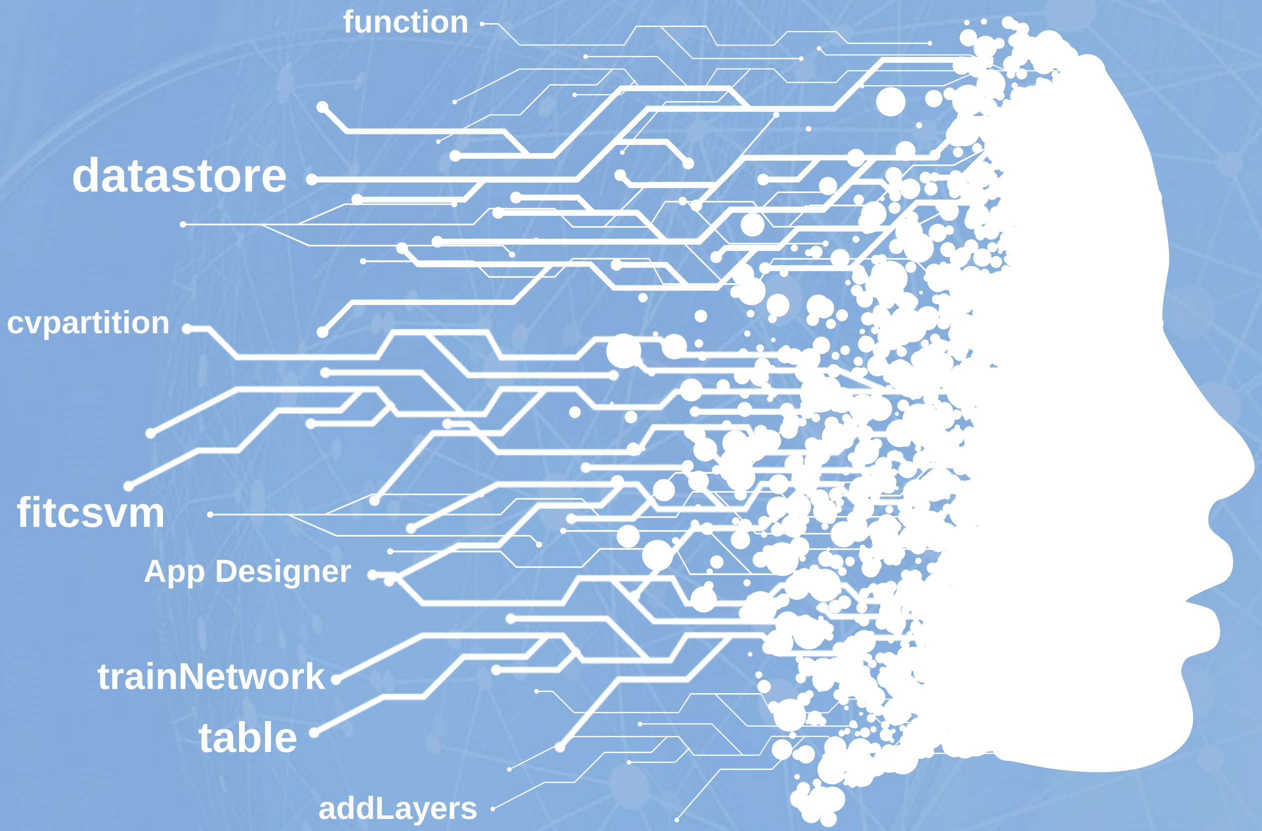


資料結構

MATLAB進階程式語言與實作

盧家鋒 副教授
生物醫學影像暨放射科學系
alvin4016@nycu.edu.tw

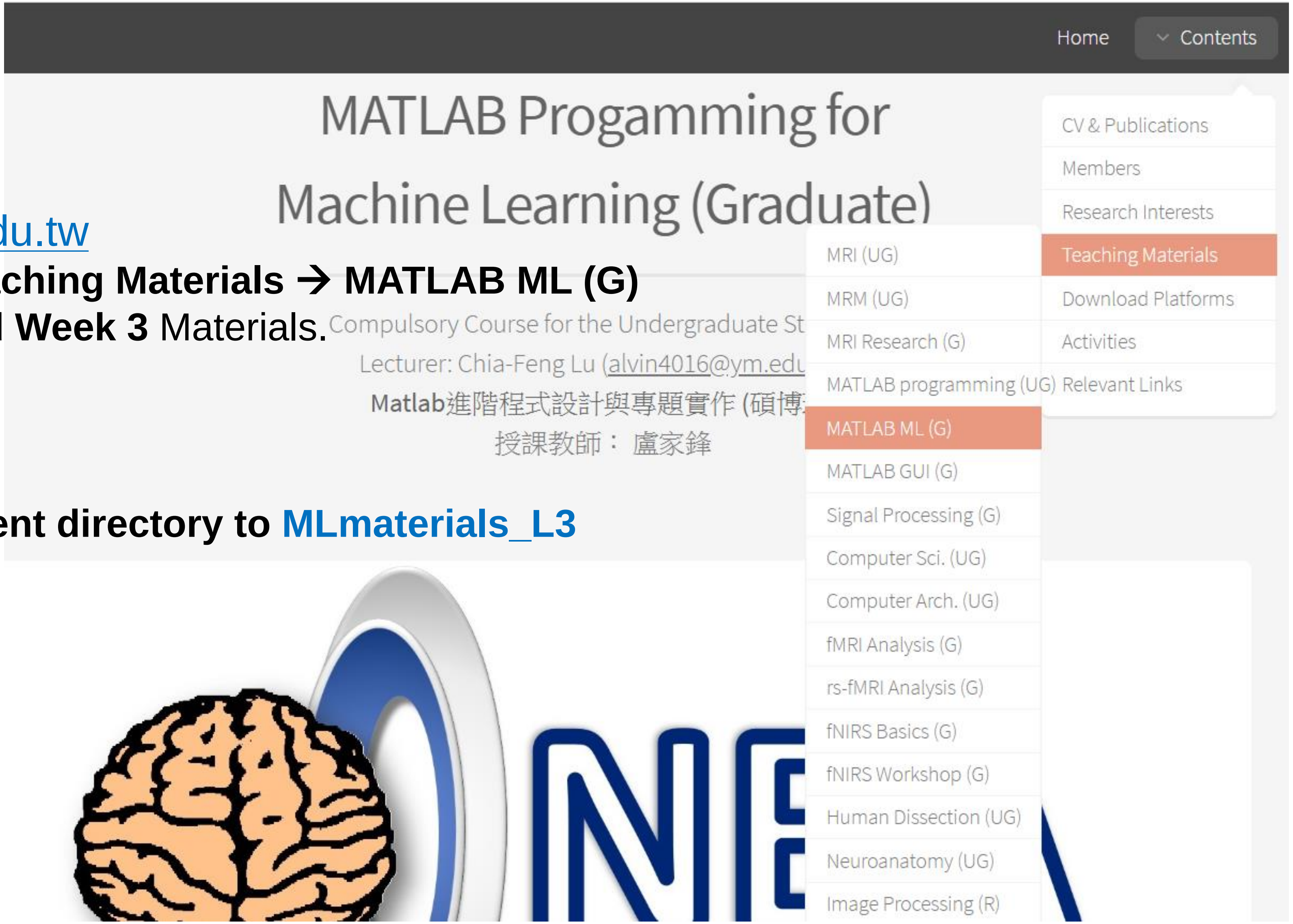
Teaching Materials

cflu.lab.nycu.edu.tw

Contents → Teaching Materials → MATLAB ML (G)

Please download **Week 3** Materials.

Please set current directory to MLmaterials_L3



MATLAB Progamming for Machine Learning (Graduate)

Compulsory Course for the Undergraduate St

Lecturer: Chia-Feng Lu (alvin4016@ym.edu

Matlab進階程式設計與專題實作 (碩博

授課教師： 盧家鋒

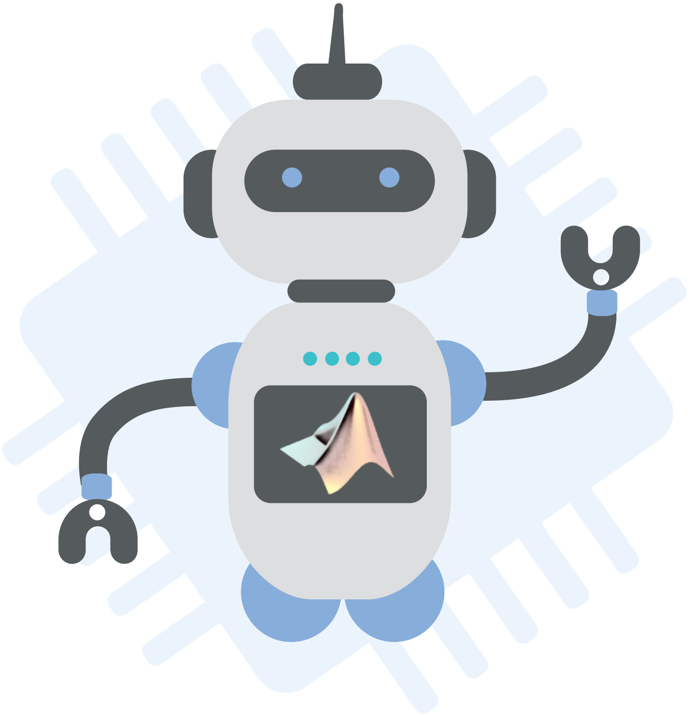
Contents in this Week

01 MATLAB Data Types

Images, datastore, table arrays

02 Datastore and Table Usage

A hands-on example



Reference

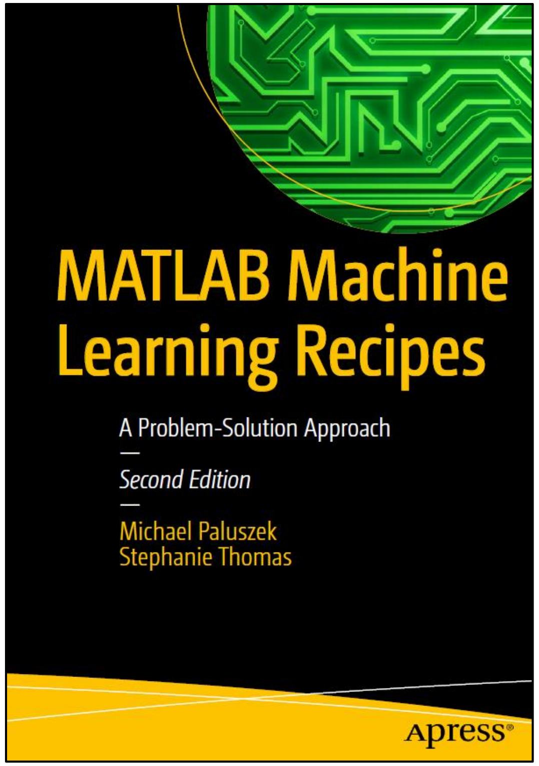
[Textbook 2]

- **MATLAB Machine Learning Recipes, 2nd Ed, 2018**

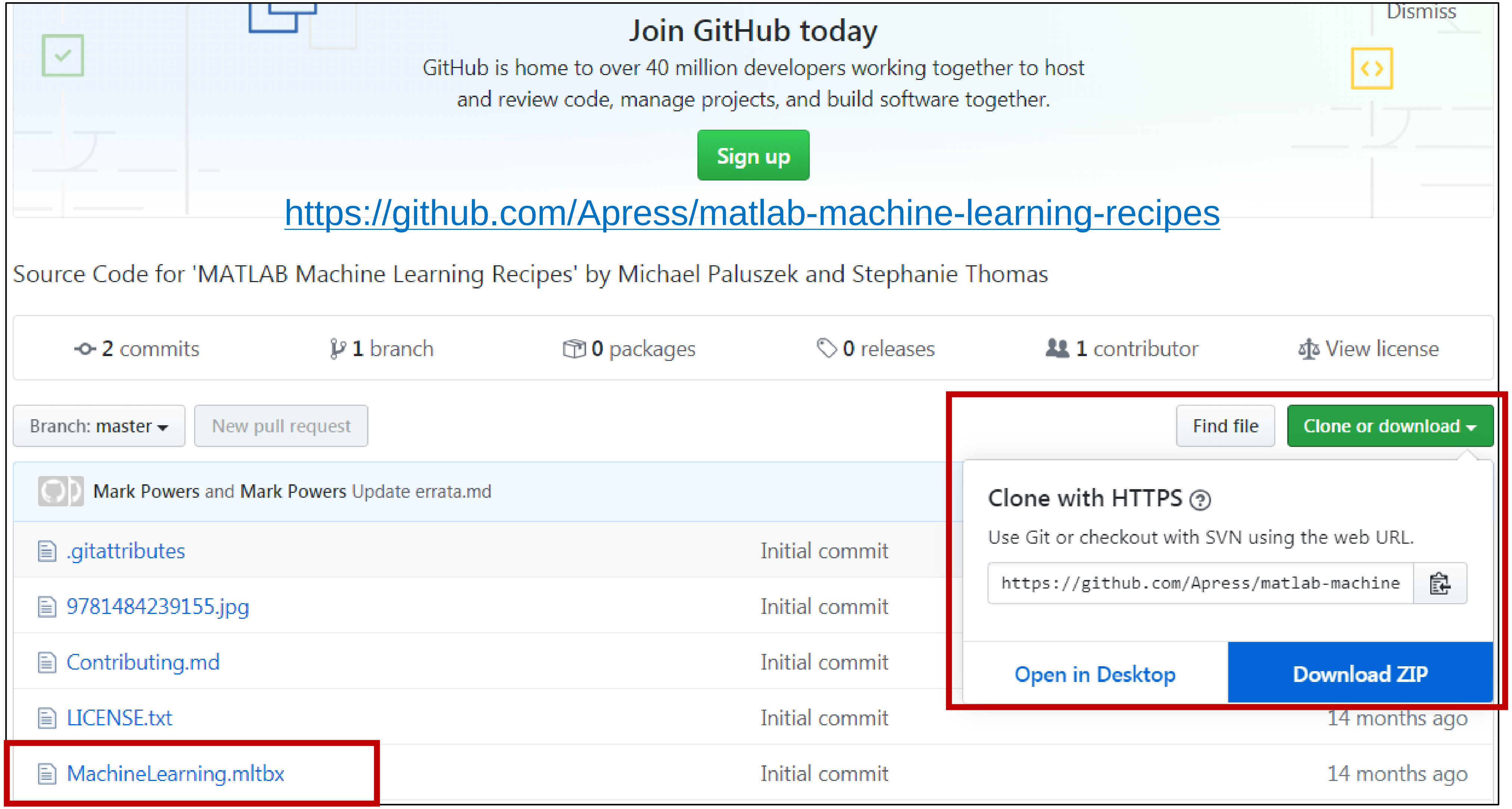
  Michael Paluszek, Stephanie Thomas
- **Online resources:**

  https://github.com/Apress/matlab-machine-learning-recipes

- **Chapter 2: Representation of Data for Machine Learning in MATLAB**



Download MATLAB Add-On



Dismiss

Join GitHub today

GitHub is home to over 40 million developers working together to host and review code, manage projects, and build software together.

Sign up

https://github.com/Apress/matlab-machine-learning-recipes

Source Code for 'MATLAB Machine Learning Recipes' by Michael Paluszek and Stephanie Thomas

2 commits | 1 branch | 0 packages | 0 releases | 1 contributor | View license

Branch: master ▾ | New pull request | Find file | Clone or download ▾

Mark Powers and Mark Powers Update errata.md

.gitattributes	Initial commit	
9781484239155.jpg	Initial commit	
Contributing.md	Initial commit	
LICENSE.txt	Initial commit	14 months ago
MachineLearning.mltbx	Initial commit	14 months ago

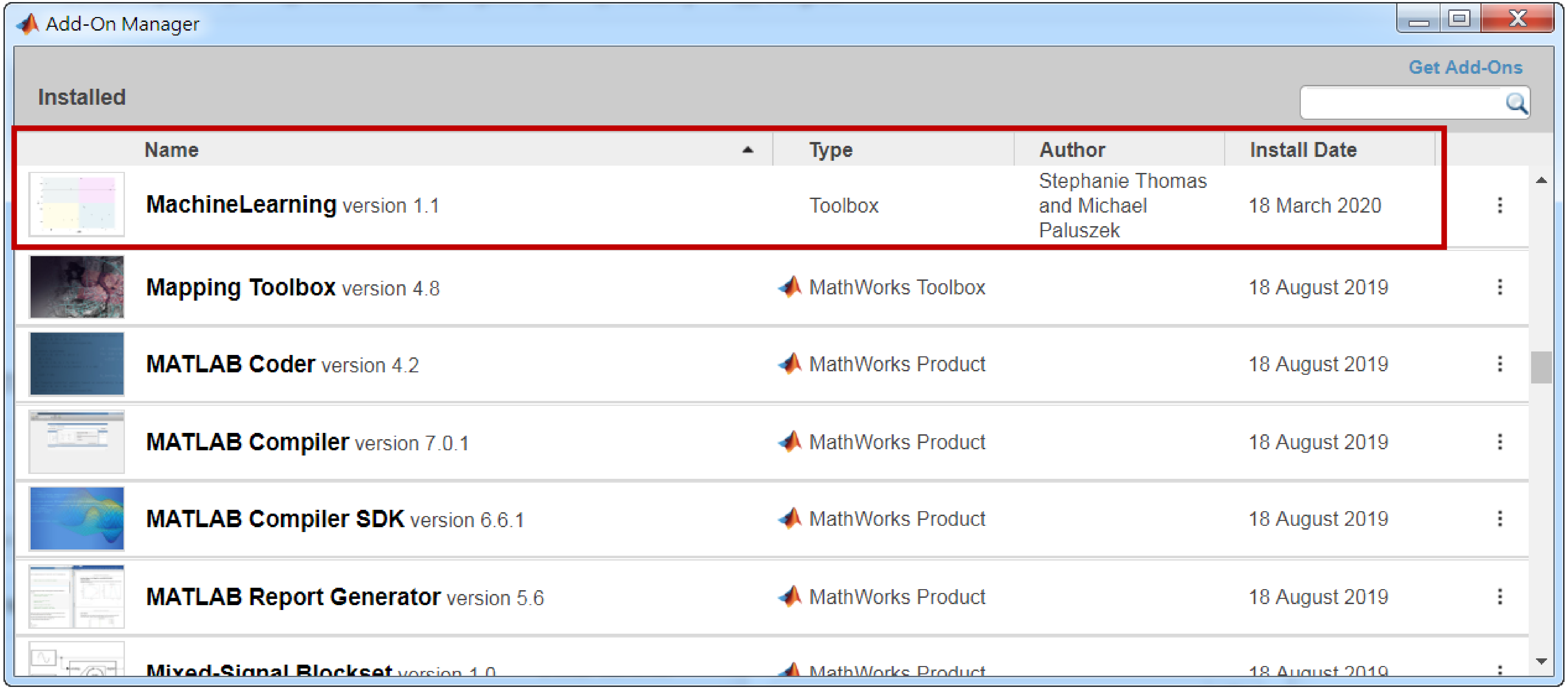
Clone with HTTPS ?

Use Git or checkout with SVN using the web URL.

```
https://github.com/Apress/matlab-machine
```

Open in Desktop | Download ZIP

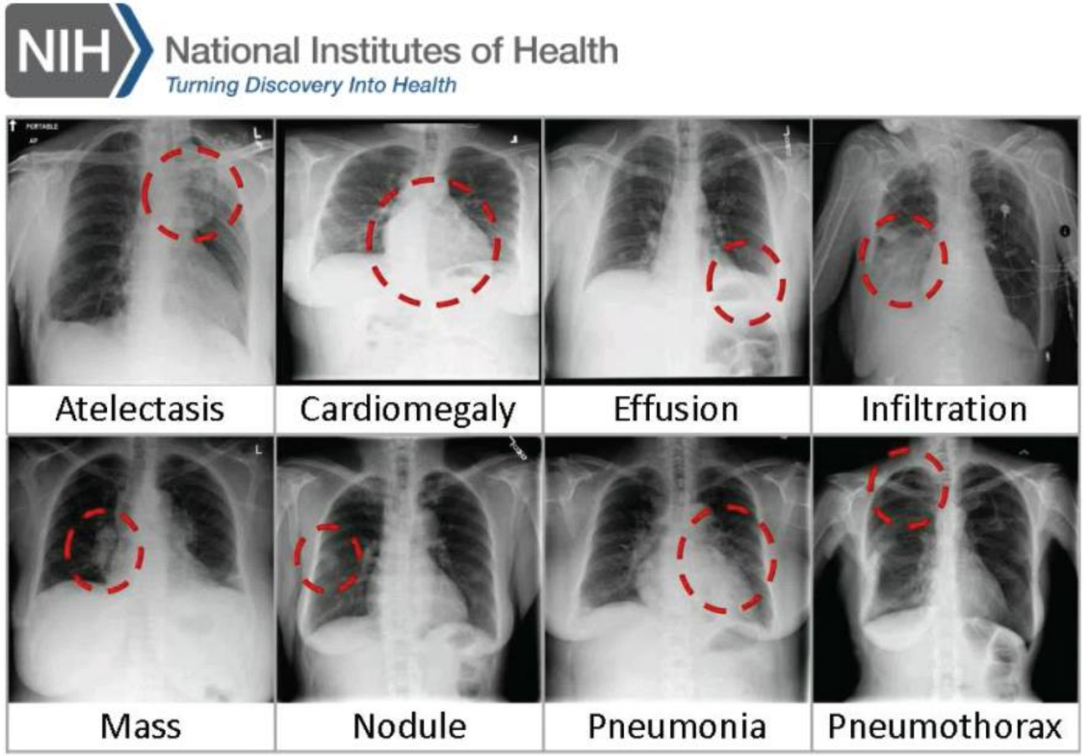
Install MachineLearning.mltbx



Name	Type	Author	Install Date
MachineLearning version 1.1	Toolbox	Stephanie Thomas and Michael Paluszek	18 March 2020
Mapping Toolbox version 4.8	MathWorks Toolbox		18 August 2019
MATLAB Coder version 4.2	MathWorks Product		18 August 2019
MATLAB Compiler version 7.0.1	MathWorks Product		18 August 2019
MATLAB Compiler SDK version 6.6.1	MathWorks Product		18 August 2019
MATLAB Report Generator version 5.6	MathWorks Product		18 August 2019
Mixed-Signal Blockset version 1.0	MathWorks Product		18 August 2019

Disease Classification/Detection in Chest X-ray

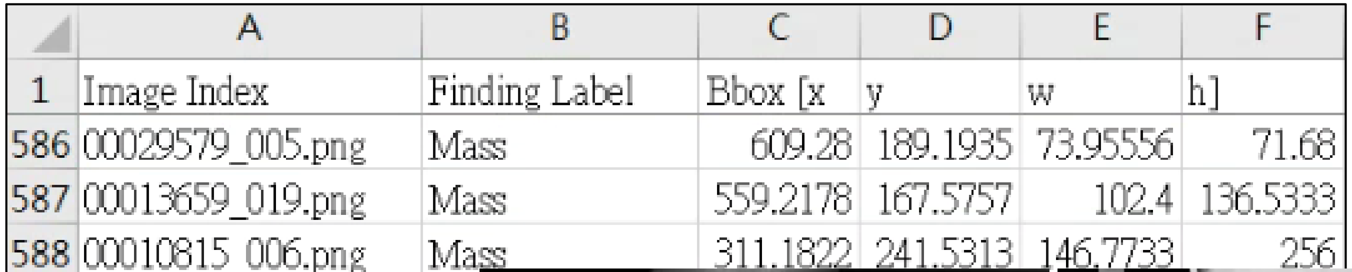
- Provided by NIH Clinical Center, 2017
  - https://nihcc.app.box.com/v/ChestXray-NIHCC/folder/36938765345
- Overall 112,120 chest x-ray images from 30805 patients
- Covering 14 common chest diseases
  - Atelectasis; Cardiomegaly; Effusion; Infiltration; Mass; Nodule; Pneumonia; Pneumothorax; Consolidation; Edema; Emphysema; Fibrosis; Pleural Thickening; Hernia
- With bounding box (BBox) locating lesions in 984 images



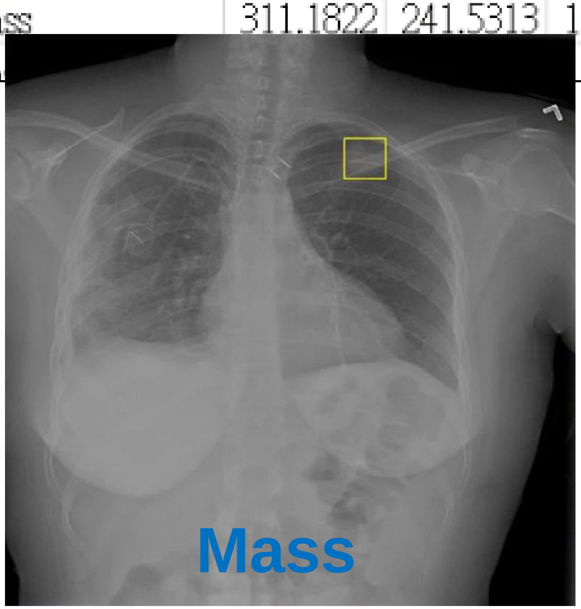
CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning
Wang et al. ChestX-ray8. IEEE CVPR 2017

Demo Dataset - ChestX-ray14

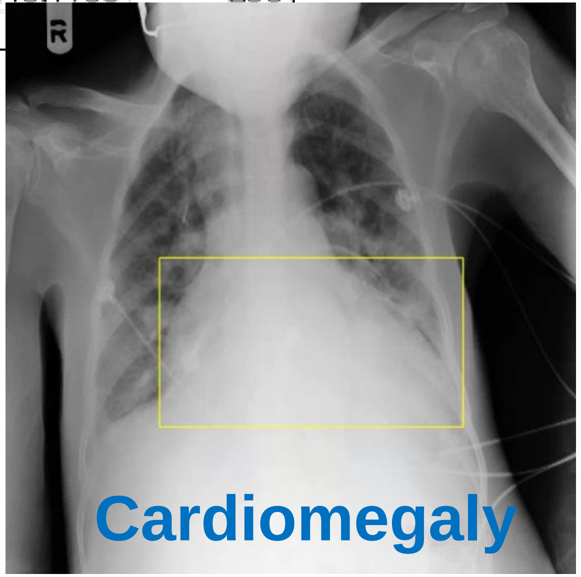
- MLmaterials_L3\BBox_List_2017.csv
- MLmaterials_L3\bbox_display.m



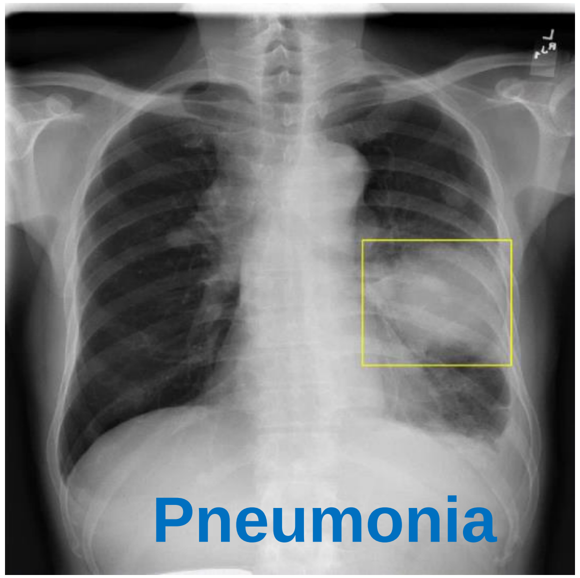
	A	B	C	D	E	F
1	Image Index	Finding Label	Bbox [x	y	w	h]
586	00029579_005.png	Mass	609.28	189.1935	73.95556	71.68
587	00013659_019.png	Mass	559.2178	167.5757	102.4	136.5333
588	00010815_006.png	Mass	311.1822	241.5313	146.7733	256
589	00026695_000.png	Ma				



Mass



Cardiomegaly



Pneumonia

MATLAB Data Types

Images, datastore, table arrays

MATLAB Data Types

- Matrices
- Cell arrays
- Structure arrays
- **Images**
- **Datastore**
- **Tables**
- Categoricals
- Tall arrays
- Sparse matrices
- Large MAT-files

Images

- MATLAB supports a variety of formats including GIF, JPG, TIFF, PNG, HDF, FITS, and BMP.

>> **imread** Read image from graphics file.

- **A = imread(FILENAME,FMT)** reads a grayscale or color image from the file specified by the character vector or string scalar FILENAME.

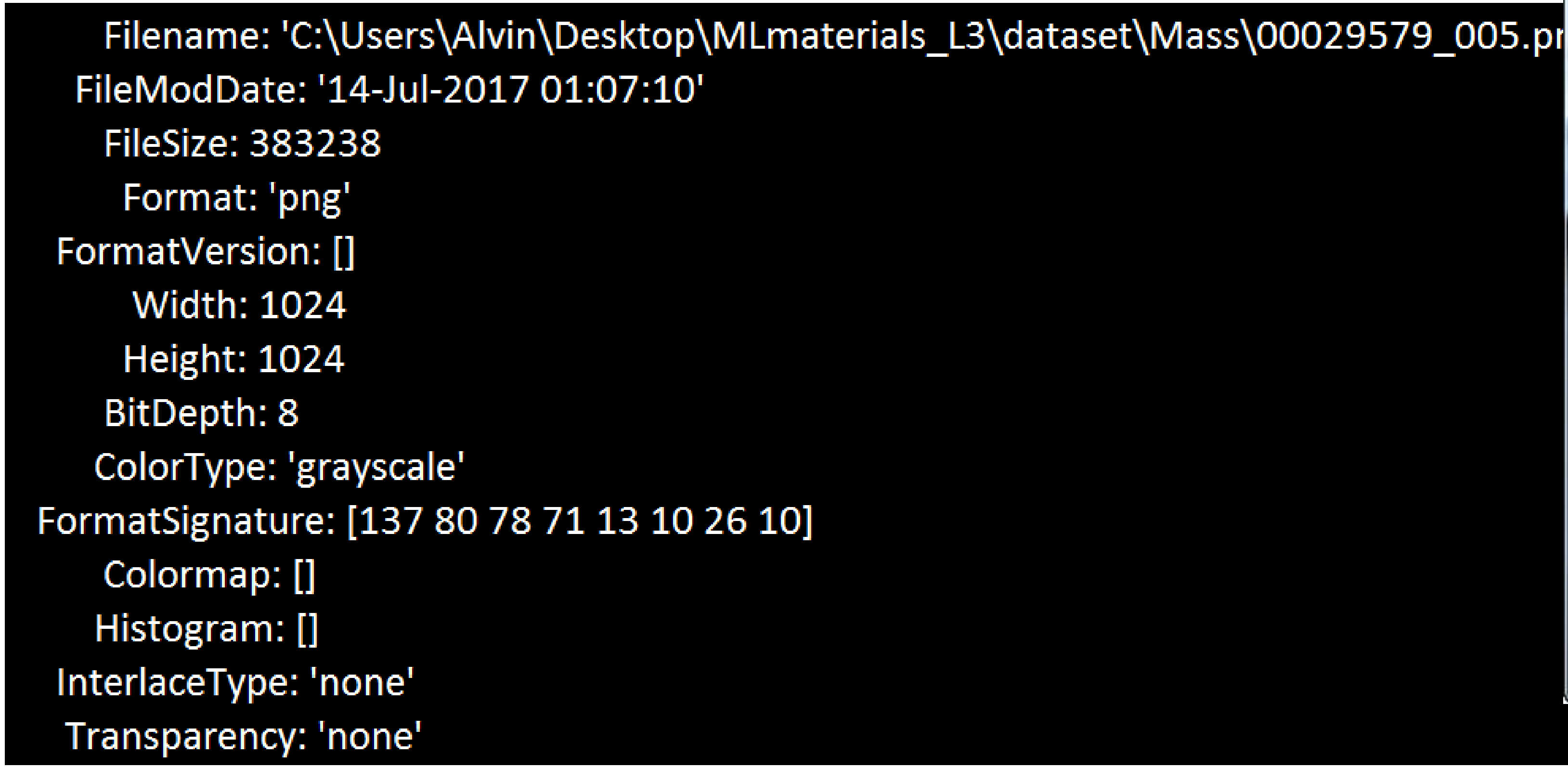
>> imfinfo Information about graphics file.

- **INFO = imfinfo(FILENAME,FMT)** returns a structure whose fields contain information about an image in a graphics file.

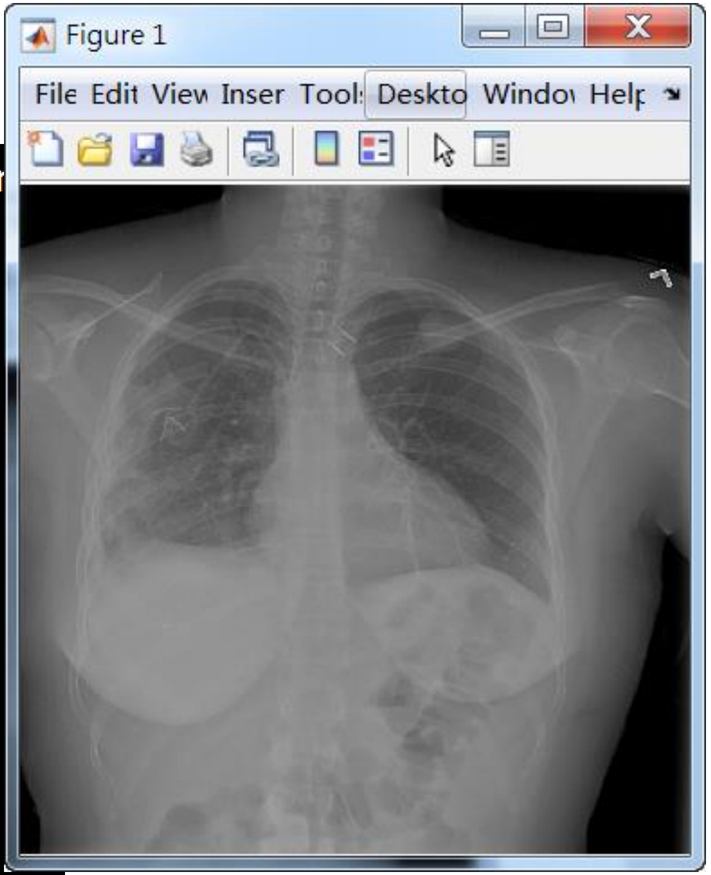
Exercise

For Mac users, please replace '\' by '/' in the file path.

- imdata = imread('.\dataset\Mass\00029579_005.png');
- figure, imshow(imdata,[],'border','tight')
- info = imfinfo('.\dataset\Mass\00029579_005.png')



```
      Filename: 'C:\Users\Alvin\Desktop\MLmaterials_L3\dataset\Mass\00029579_005.pr
   FileModDate: '14-Jul-2017 01:07:10'
      FileSize: 383238
        Format: 'png'
 FormatVersion: []
         Width: 1024
        Height: 1024
      BitDepth: 8
     ColorType: 'grayscale'
FormatSignature: [137 80 78 71 13 10 26 10]
      Colormap: []
     Histogram: []
 InterlaceType: 'none'
  Transparency: 'none'
```



Useful Functions for Images

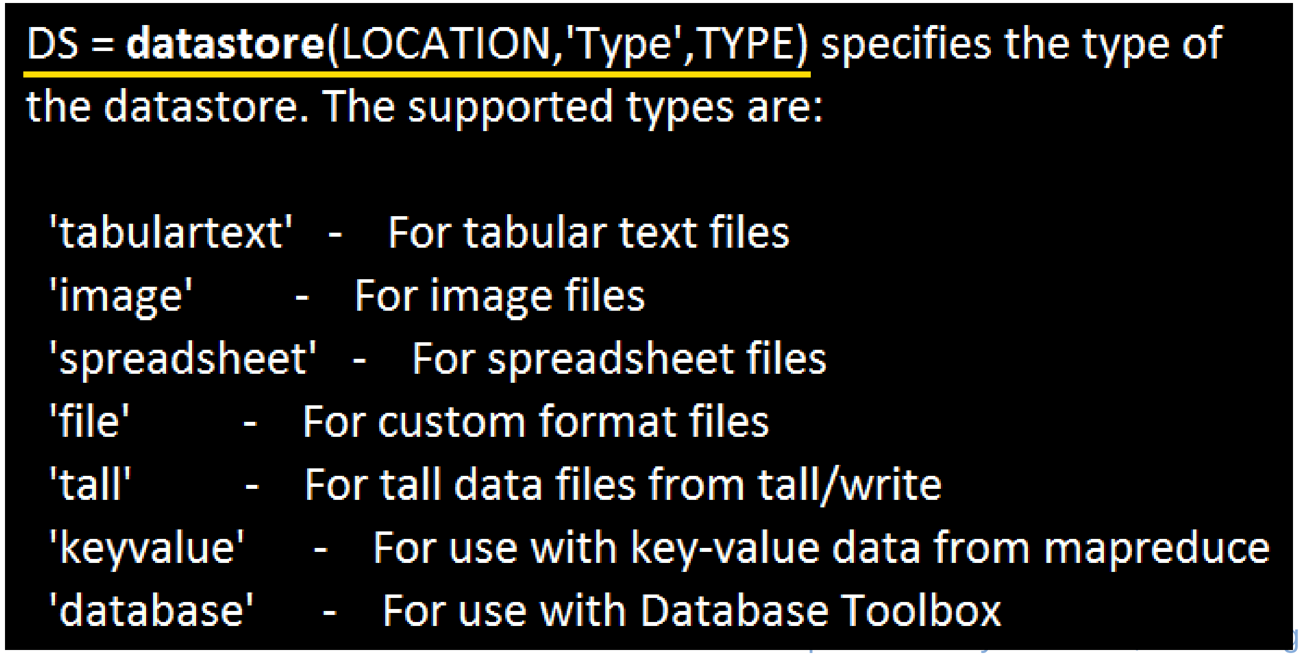
Function	Purpose
imread	Read an image in a variety of formats
imfinfo	Gather information about an image file
imformats	Manage file format registry
imwrite	Write data to an image file
image	Display image from array
imagesc	Display image data scaled to the current colormap
imshow	Display an image, optimizing figure, axes, and image object properties, and taking an array or a filename as an input
rgb2gray	Convert RGB image or colormap to grayscale
ind2rgb	Convert index data to RGB
rgb2ind	Convert RGB data to indexed image data
fitsread	Read a FITS file
fitswrite	Write data to a FITS file
fitsinfo	Information about a FITS file returned in a data structure
fitsdisp	Display FITS file metadata for all HDUs in the file

Datastore

- Datastores allow you to interact with files containing data that are too large to fit in memory.

>> datastore Create a datastore for working with collections of data.

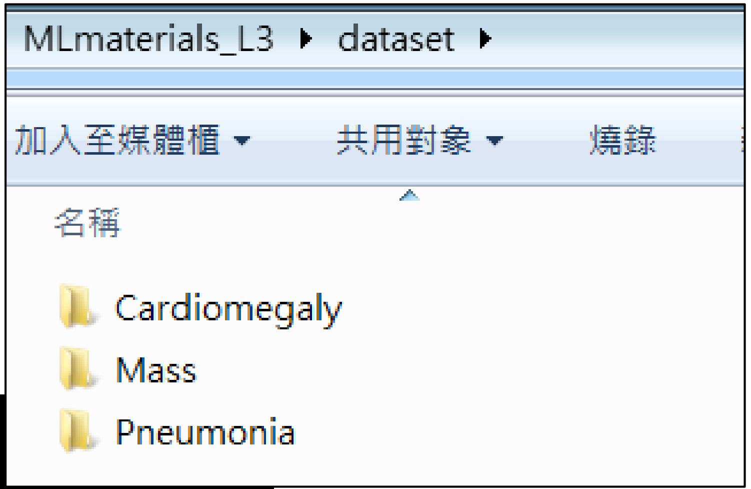
- **DS = datastore(LOCATION)** creates a datastore DS based on the LOCATION of the data.



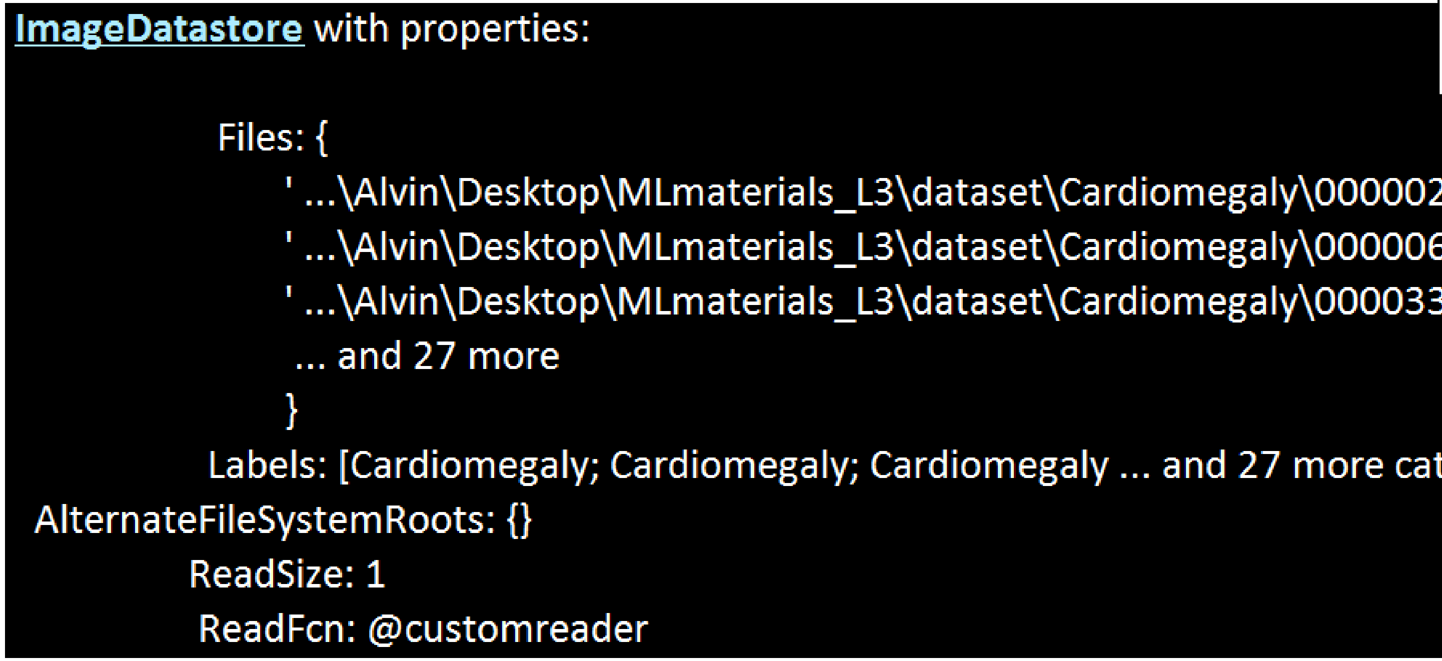
```
DS = datastore(LOCATION,'Type',TYPE) specifies the type of
the datastore. The supported types are:

  'tabulartext'   -   For tabular text files
  'image'         -   For image files
  'spreadsheet'   -   For spreadsheet files
  'file'          -   For custom format files
  'tall'          -   For tall data files from tall/write
  'keyvalue'      -   For use with key-value data from mapreduce
  'database'      -   For use with Database Toolbox
```

Exercise – image datastore

- dirname='.\dataset';
- ds = datastore(dirname,'type','image',...
  'IncludeSubfolders',true,...
  'LabelSource','foldernames',...
  'ReadFcn',@customreader);



Cardiomegaly
Mass
Pneumonia

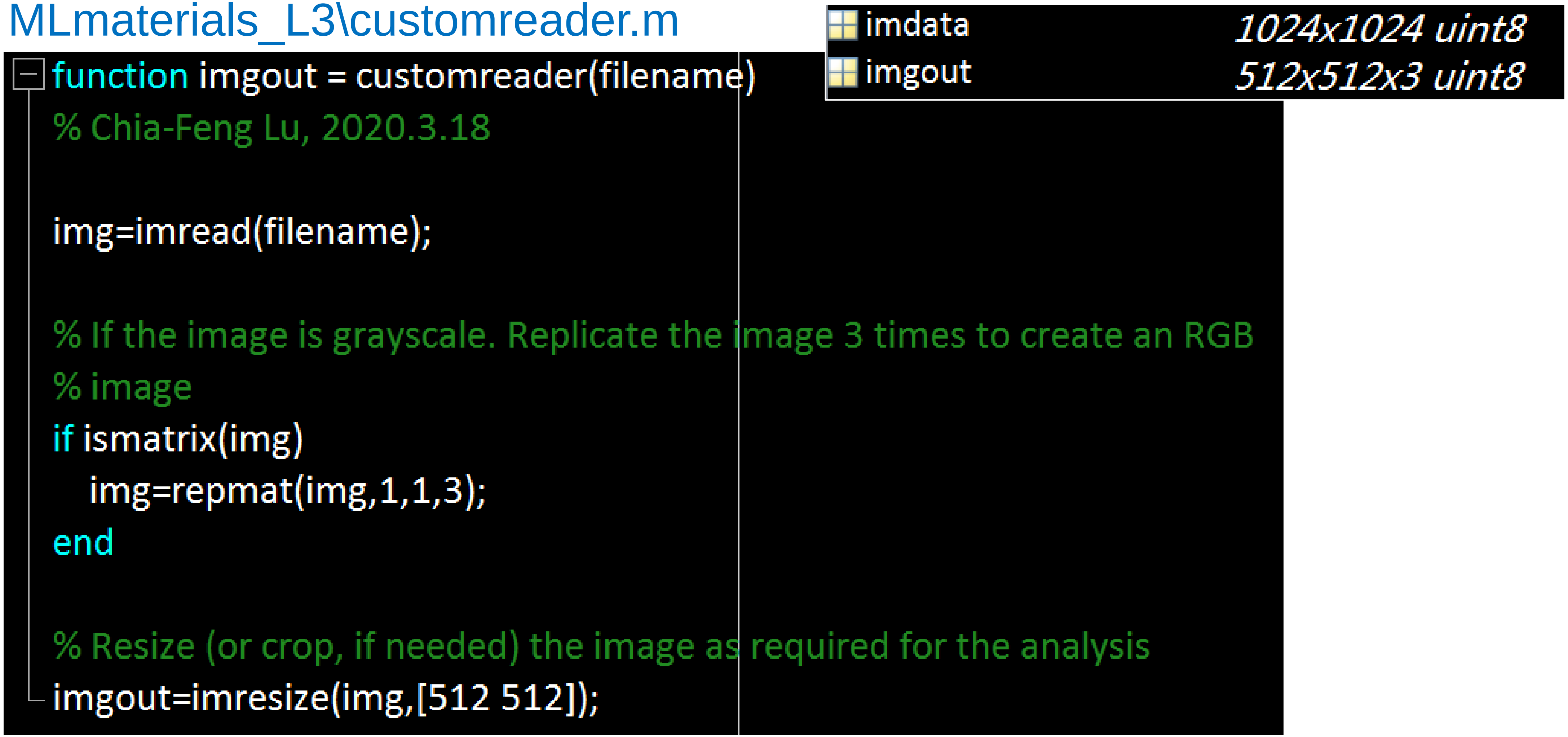


```
ImageDatastore with properties:

                  Files: {
                         ' ...\Alvin\Desktop\MLmaterials_L3\dataset\Cardiomegaly\00000211_041.png';
                         ' ...\Alvin\Desktop\MLmaterials_L3\dataset\Cardiomegaly\00000661_000.png';
                         ' ...\Alvin\Desktop\MLmaterials_L3\dataset\Cardiomegaly\00003394_006.png'
                          ... and 27 more
                         }
                 Labels: [Cardiomegaly; Cardiomegaly; Cardiomegaly ... and 27 more categorical]
 AlternateFileSystemRoots: {}
               ReadSize: 1
                ReadFcn: @customreader
```

>> doc **datastore**
>> doc **imageDatastore**
for more detailed information

customreader – ReadFcn

- MLmaterials_L3\customreader.m



```
function imgout = customreader(filename)
% Chia-Feng Lu, 2020.3.18

img=imread(filename);

% If the image is grayscale. Replicate the image 3 times to create an RGB
% image
if ismatrix(img)
    img=repmat(img,1,1,3);
end

% Resize (or crop, if needed) the image as required for the analysis
imgout=imresize(img,[512 512]);
```

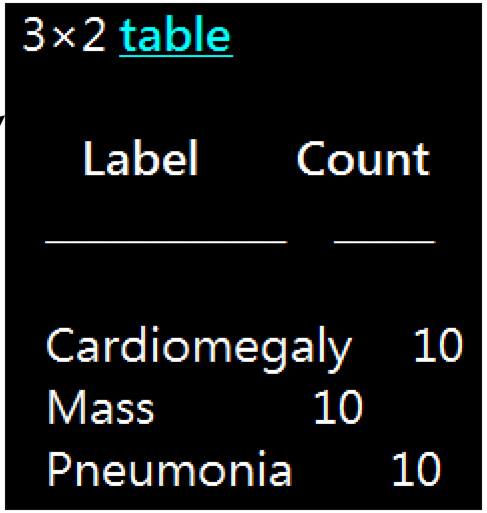
```
imdata        1024x1024 uint8
imgout        512x512x3 uint8
```

Useful Functions for Datastore

Function	Purpose
datastore	Create a datastore
read	Read a subset of data from the datastore
readall	Read all of the data in the datastore
hasdata	Check to see if there are more data in the datastore
reset	Initialize a datastore with the contents of a folder
partition	Excerpt a portion of the datastore
numpartitions	Estimate a reasonable number of partitions
ImageDatastore	Datastore of a list of image files
TabularTextDatastore	A collection of one or more tabular text files
SpreadsheetDatastore	Datastore of spreadsheets
FileDatastore	Datastore for files with a custom format, for which you provide a reader function
KeyValueDatastore	Datastore of key-value pairs
DatabaseDatastore	Database connection, requires the Database Toolbox

Using a Datastore

- ds.Files
- ds.Labels
- ds.countEachLabel
- shuffle_ds=ds.shuffle
- ds.ReadSize



```
3×2 table

    Label        Count
 ____________    _____

 Cardiomegaly     10
 Mass             10
 Pneumonia        10
```

- [img,info]=ds.read; **or** img=read(ds);
  % read next image set from ds ImageDatastore
- ds.reset; **or** reset(ds);
  % reset the counter to the first image file
- imgall=ds.readall; **or** imgall=readall(ds);
  % read all image files and output as a cell array: 30 x 1 cell

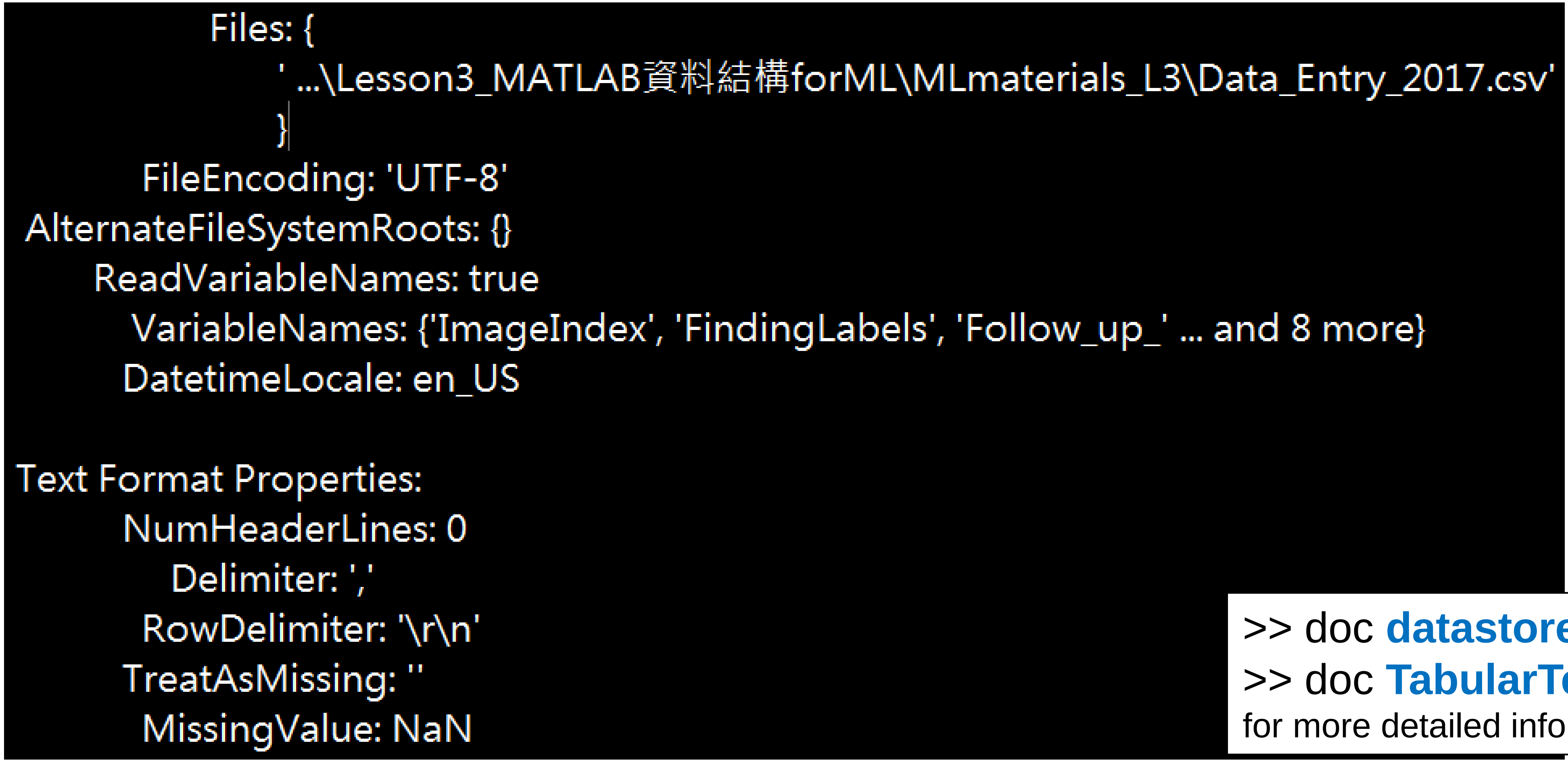
- ds.hasdata;

Using a Datastore

```
% Partition the datastore into three parts on three workers in a parallel pool.
n = 3;
p = parpool('local',n);

parfor ii=1:n
   subds = partition(ds,n,ii);
      while hasdata(subds)
         img = read(subds);
      end
end
```

Exercise – Tabular Datastore

- ds = datastore('Data_Entry_2017.csv','type','tabular');

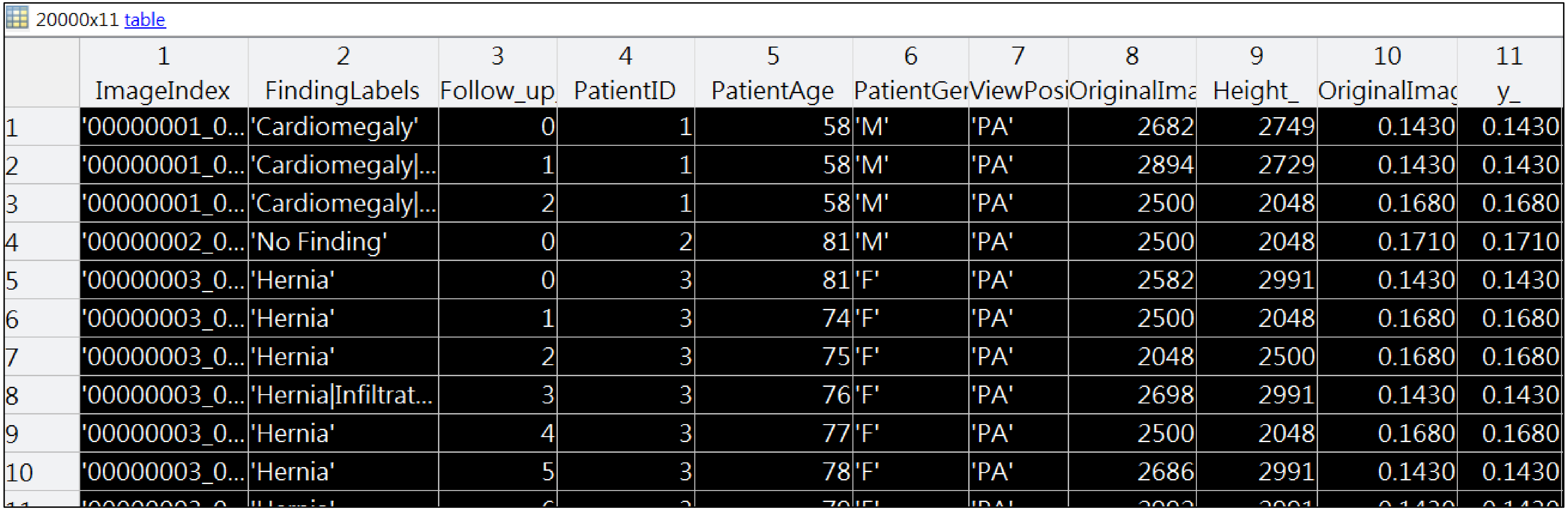


```
                   Files: {
                          ' ...\Lesson3_MATLAB資料結構forML\MLmaterials_L3\Data_Entry_2017.csv'
                          }
            FileEncoding: 'UTF-8'
 AlternateFileSystemRoots: {}
       ReadVariableNames: true
           VariableNames: {'ImageIndex', 'FindingLabels', 'Follow_up_' ... and 8 more}
          DatetimeLocale: en_US

Text Format Properties:
          NumHeaderLines: 0
               Delimiter: ','
            RowDelimiter: '\r\n'
          TreatAsMissing: ''
            MissingValue: NaN
```

>> doc **datastore**
>> doc **TabularTextDatastore**
for more detailed information

Using a Tabular Datastore

- ds.VariableNames
- ds.ReadSize % 20000 in default
- tabledata=ds.read; % Output as a table array



20000x11 table

	1 ImageIndex	2 FindingLabels	3 Follow_up	4 PatientID	5 PatientAge	6 PatientGe	7 ViewPosi	8 OriginalIma	9 Height_	10 OriginalImag	11 y_
1	'00000001_0...	'Cardiomegaly'	0	1	58	'M'	'PA'	2682	2749	0.1430	0.1430
2	'00000001_0...	'Cardiomegaly|...	1	1	58	'M'	'PA'	2894	2729	0.1430	0.1430
3	'00000001_0...	'Cardiomegaly|...	2	1	58	'M'	'PA'	2500	2048	0.1680	0.1680
4	'00000002_0...	'No Finding'	0	2	81	'M'	'PA'	2500	2048	0.1710	0.1710
5	'00000003_0...	'Hernia'	0	3	81	'F'	'PA'	2582	2991	0.1430	0.1430
6	'00000003_0...	'Hernia'	1	3	74	'F'	'PA'	2500	2048	0.1680	0.1680
7	'00000003_0...	'Hernia'	2	3	75	'F'	'PA'	2048	2500	0.1680	0.1680
8	'00000003_0...	'Hernia|Infiltrat...	3	3	76	'F'	'PA'	2698	2991	0.1430	0.1430
9	'00000003_0...	'Hernia'	4	3	77	'F'	'PA'	2500	2048	0.1680	0.1680
10	'00000003_0...	'Hernia'	5	3	78	'F'	'PA'	2686	2991	0.1430	0.1430
11	'00000003_0...	'Hernia'	6	3	79	'F'	'PA'	[illegible]	2991	0.1430	0.1430

Table Array

- Tables were introduced in release R2013 of MATLAB and allow tabular data to be stored with metadata in one workspace variable.
- The table columns can be named, assigned units and descriptions, and accessed as one would fields in a data structure.

- tabledata.PatientAge → 20000x1 double
- tabledata(:,3:4) → 20000x2 table
- tabledata(:,3:4).Variables → 20000x2 double
- tabledata.Properties.VariableNames
- tabledata.Properties.VariableUnits
- tabledata.Properties.VariableDescriptions

Useful Functions for Table

Function	Purpose
table	Create a table from workspace variables
readtable	Create a table by reading from a file
join	Merge two tables by matching up rows using key variables
innerjoin	Join tables A and B retaining only the rows that match
outerjoin	Join tables including all rows
stack	Stack data from multiple table variables into one variable
unstack	Unstack data from a single variable into multiple variables
summary	Calculate and display summary data for the table
struct2table	Convert structure array to table
table2struct	Convert table to structure array
table2array	Convert table to a homogeneous array

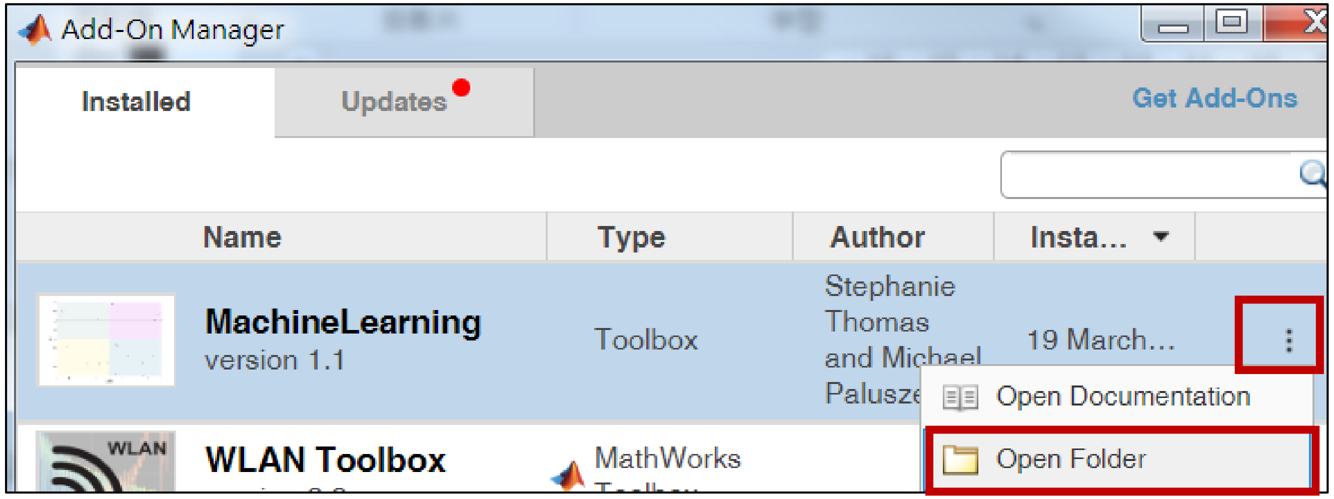
- structdata=table2struct(tabledata);
- tablesummary=summary(tabledata);

Datastore and Table Usage

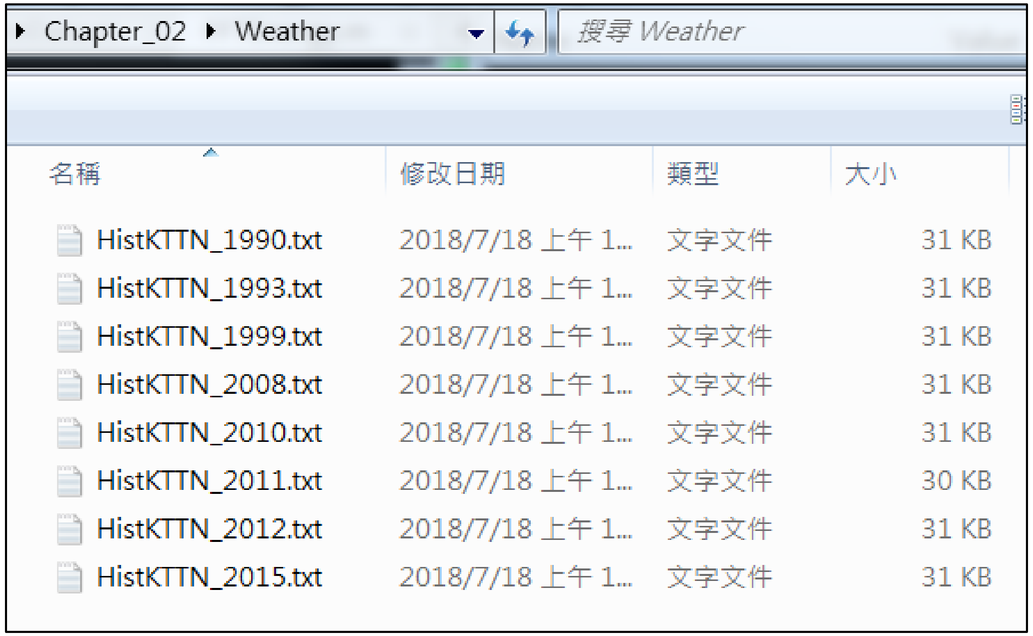
A hands-on example

Problem to be Solved

- **Problem:** We want to compare temperature frequencies in 1993 and 2015 using weather data from a table.
- **Solution:** Use **tabularTextDatastore** to load the data and perform the fast Fourier transform (FFT) on the data.
- **Switch current directory**
  - ~\MachineLearning\Chapter_02



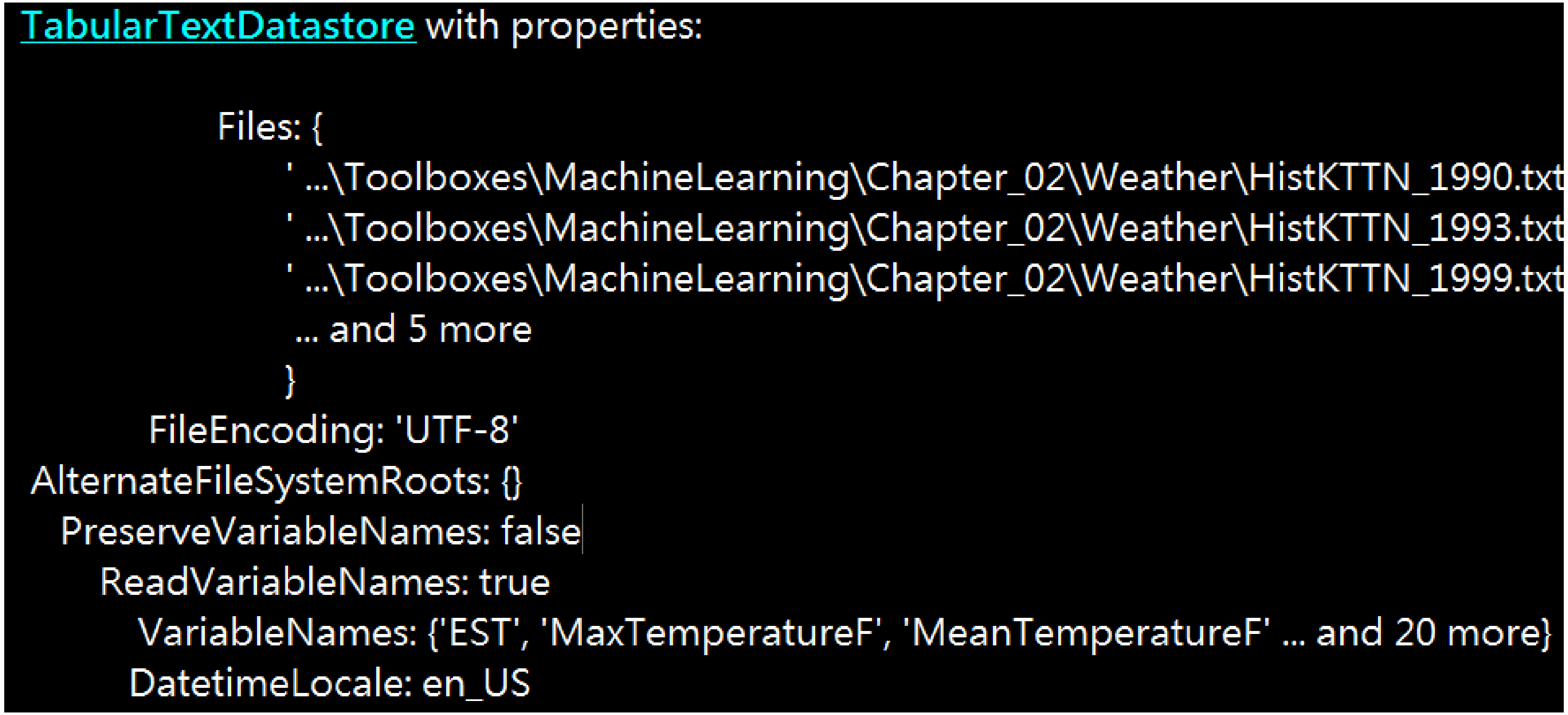
Name	Type	Author	Insta...
MachineLearning version 1.1	Toolbox	Stephanie Thomas and Michael Palusze	19 March...
WLAN Toolbox	MathWorks Toolbox		



名稱	修改日期	類型	大小
HistKTTN_1990.txt	2018/7/18 上午 1...	文字文件	31 KB
HistKTTN_1993.txt	2018/7/18 上午 1...	文字文件	31 KB
HistKTTN_1999.txt	2018/7/18 上午 1...	文字文件	31 KB
HistKTTN_2008.txt	2018/7/18 上午 1...	文字文件	31 KB
HistKTTN_2010.txt	2018/7/18 上午 1...	文字文件	31 KB
HistKTTN_2011.txt	2018/7/18 上午 1...	文字文件	30 KB
HistKTTN_2012.txt	2018/7/18 上午 1...	文字文件	31 KB
HistKTTN_2015.txt	2018/7/18 上午 1...	文字文件	31 KB

Create Tabular Datastore

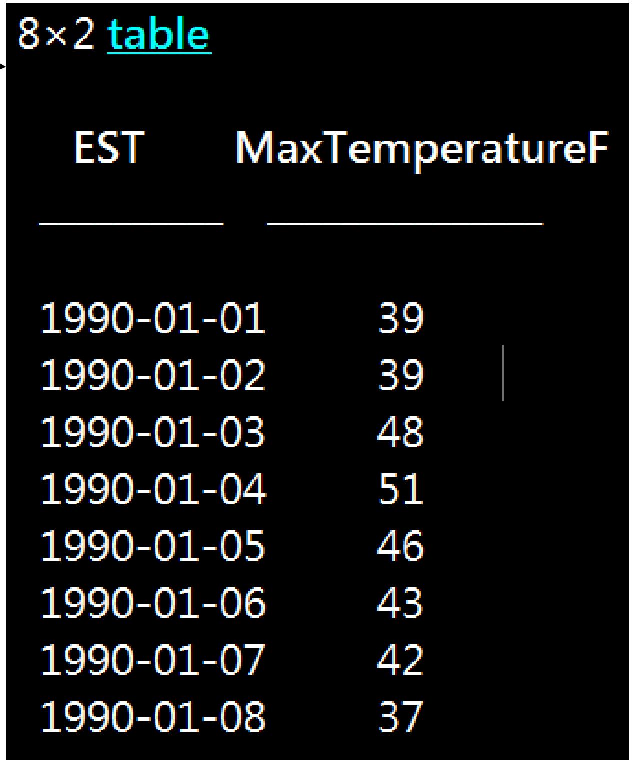
- tableds=datastore('.\Weather','type','tabular')
- tableds.VariableNames   % 23 variables



```
TabularTextDatastore with properties:

                      Files: {
                             ' ...\Toolboxes\MachineLearning\Chapter_02\Weather\HistKTTN_1990.txt
                             ' ...\Toolboxes\MachineLearning\Chapter_02\Weather\HistKTTN_1993.txt
                             ' ...\Toolboxes\MachineLearning\Chapter_02\Weather\HistKTTN_1999.txt
                              ... and 5 more
                             }
               FileEncoding: 'UTF-8'
  AlternateFileSystemRoots: {}
     PreserveVariableNames: false
         ReadVariableNames: true
             VariableNames: {'EST', 'MaxTemperatureF', 'MeanTemperatureF' ... and 20 more}
            DatetimeLocale: en_US
```

Select Variables and Read

```
tableds.SelectedVariableNames={'EST','MaxTemperatureF'};
tableds.preview
AllTemp=tableds.readall;

%% identify 1993 and 2015 data indices
dateinfo=AllTemp{:,1};
% tempdata=AllTemp.EST; datatime format
y=year(dateinfo);
ind_1993=find(y==1993);
ind_2015=find(y==2015);
```

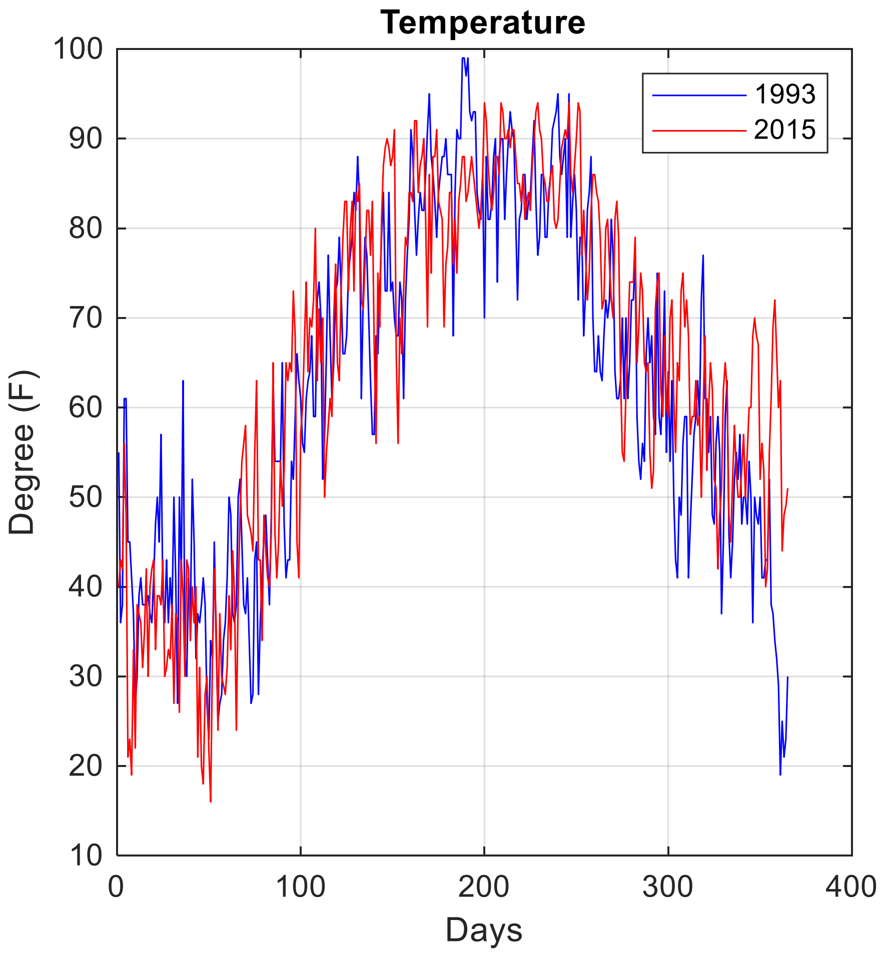


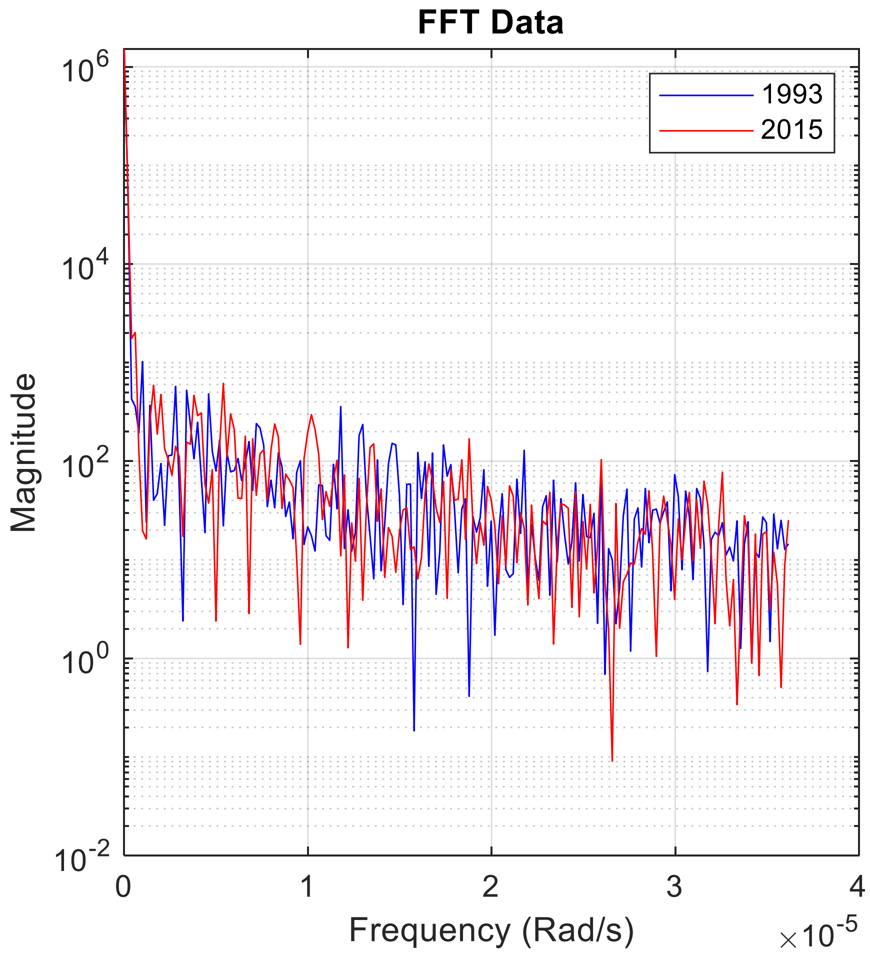
```
8×2 table

    EST        MaxTemperatureF
  __________   _______________

  1990-01-01         39
  1990-01-02         39
  1990-01-03         48
  1990-01-04         51
  1990-01-05         46
  1990-01-06         43
  1990-01-07         42
  1990-01-08         37
```

Perform FFT on Temperature Data

```
%% Extract temperature data in 1993 and 2015, respectively
tempdata=AllTemp{:,2};  %tempdata=AllTemp.MaxTemperatureF;
temp1993=tempdata(ind_1993);
temp2015=tempdata(ind_2015);
figure, subplot(1,2,1), plot(1:365, temp1993','b', 1:365, temp2015','r'), grid on
title('Temperature'),xlabel('Days'),ylabel('Degree (F)'),legend('1993','2015')

SampTime=86400;  % interval between days in seconds
[p1993,f1993]=FFTEnergy(temp1993',SampTime);
[p2015,f2015]=FFTEnergy(temp2015',SampTime);
subplot(1,2,2), plot(f1993, p1993','b', f2015, p2015','r'), set(gca,'YScale','log'), grid on
title('FFT Data'),xlabel('Frequency (Rad/s)'),ylabel('Magnitude'),legend('1993','2015')
```

Result Plots





THE END

Contact:

盧家鋒 alvin4016@nycu.edu.tw