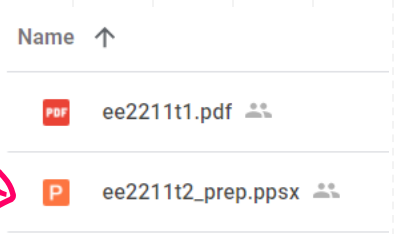
Briefing on pandas and matplotlib is available in tiny.cc/ee2211tut.

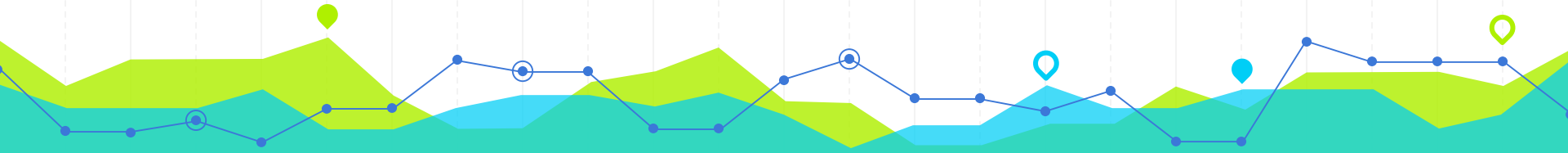


We'll start at 1205PM, let me know which questions we should spend more time on!

EE2211 Introduction to Machine Learning

T14 & T22, Chua Dingjuan elechuad@nus.edu.sg

Slides @ tiny.cc/ee2211tut



Tutorial 2

Plotting, Manipulation of Data & Matrices in Python

Brief Summary of Key Points

- Python for **Data manipulation**, **Plotting** and **ML**
  - **pandas**, **matplotlib**, **scikit** (sklearn)

- Data Wrangling (Normalization along x-axis)
  - Min-max

$$x_i = \frac{x_i^{raw} - x^{min}}{x^{max} - x^{min}}, i = 1,2, \dots, m$$

range of values

  - Z score

$$x_i = \frac{x_i^{raw} - E[X]}{\sigma(X)}, i = 1,2, \dots, m$$

mean

Standard deviation

IDLE Library Installation Instructions...

How to install libraries for Python IDLE :

1) Open command prompt as administrator. (Search for cmd, right click and "Run as Administrator")

2) Using command prompt, navigate to your python installation folder. (eg. cd C:\Python37)

3) Navigate to the Scripts folder. (eg cd Scripts)

4) In the Scripts folder, type the following to install libraries.

"pip install <libraryname>”

- pip install numpy
- pip install scipy
- pip install matplotlib
- pip install ipython jupyter

Note...

- Often for programming based questions, there are multiple solutions possible.
- NOT restricted to a single solution!
- Try!
- Reference to official documentation and different tutorials and examples online would be helpful in understanding how certain functions / codes work.

pandas

- Built on top of numpy, data analysis / manipulation tool library
- 2 primary data structures :

Series (1-dimensional) and DataFrame (2-dimensional)

```
a     0.469112
b    -0.282863
c    -1.509059
d    -1.135632
e     1.212112
dtype: float64
```

	SepalLength	SepalWidth	PetalLength
0	5.1	3.5	1.4
1	4.9	3.0	1.4
2	4.7	3.2	1.3
3	4.6	3.1	1.5
4	5.0	3.6	1.4

https://pandas.pydata.org/docs/getting_started/10min.html

pandas –

Accessing Columns

```
>>> import pandas as pd
>>> df=pd.read_csv('titanic.csv')
>>> df
```

This is a dataframe

	PassengerId	Survived	Pclass	Name	Sex	Age
0	1	0	3	Braund, Mr. Owen Harris	male	22
1	2	1	1	Cumings, Mrs. John Bradley	female	38
2	3	1	3	Heikkinen, Miss. Laina	female	26
3	4	1	1	Futrelle, Mrs. Jacques Heath	female	35
4	5	0	3	Allen, Mr. William Henry	male	35

```
>>>
>>> df['Age']
0    22
1    38
2    26
3    35
4    35
Name: Age, dtype: int64
>>> df['Age'].tolist()
[22, 38, 26, 35, 35]
>>>
>>> df[['Age','Pclass']]
   Age  Pclass
0   22       3
1   38       1
2   26       3
3   35       1
4   35       3
>>>
>>> df['Age']>32
0    False
1     True
2    False
3     True
4     True
Name: Age, dtype: bool
>>>
```

← Accessing column

← Converting to Python list

← Accessing multiple columns

← Conditional Expressions

pandas –

Accessing Rows

```
>>> df
   PassengerId  Survived  Pclass                                          Name     Sex  Age
0            1         0       3                       Braund, Mr. Owen Harris    male   22
1            2         1       1           Cumings, Mrs. John Bradley (Florence  female   38
2            3         1       3                        Heikkinen, Miss. Laina  female   26
3            4         1       1  Futrelle, Mrs. Jacques Heath (Lily May Peel)  female   35
4            5         0       3                      Allen, Mr. William Henry    male   35
>>> df[0:3]  ← Accessing by rows...
   PassengerId  Survived  Pclass                                          Name     Sex  Age
0            1         0       3                       Braund, Mr. Owen Harris    male   22
1            2         1       1           Cumings, Mrs. John Bradley (Florence  female   38
2            3         1       3                        Heikkinen, Miss. Laina  female   26
>>> df['Age']>32  ← Recall column conditional expression
0    False
1     True
2    False
3     True
4     True
Name: Age, dtype: bool
>>> df[ df['Age']>32 ]    ← Combining the two
   PassengerId  Survived  Pclass                                          Name     Sex  Age
1            2         1       1           Cumings, Mrs. John Bradley (Florence  female   38
3            4         1       1  Futrelle, Mrs. Jacques Heath (Lily May Peel)  female   35
4            5         0       3                      Allen, Mr. William Henry    male   35
>>>
```

pandas –

Using loc

```
>>> df
   PassengerId  Survived  Pclass                                Name     Sex  Age
0            1         0       3             Braund, Mr. Owen Harris    male   22
1            2         1       1   Cumings, Mrs. John Bradley (Flo...  female   38
2            3         1       3              Heikkinen, Miss. Laina  female   26
3            4         1       1        Futrelle, Mrs. Jacques Heath  female   35
4            5         0       3             Allen, Mr. William Henry    male   35
>>> df.loc[ 2:4, ['Sex', 'Age'] ]
      Sex  Age
2  female   26
3  female   35
4    male   35
>>>
>>> df.loc[ df['Age'] == 35 ]
   PassengerId  Survived  Pclass                                Name     Sex  Age
3            4         1       1        Futrelle, Mrs. Jacques Heath  female   35
4            5         0       3             Allen, Mr. William Henry    male   35
>>>
>>> df.loc[ df['Age'] == 35 ].index
Int64Index([3, 4], dtype='int64')
>>>
```

← Accessing by rows and columns

← Accessing by rows with condition

← index returns index (number label) of rows

pandas

- `df = pd.read_csv('_.csv')`
  - Import into dataframe

	PassengerId	Survived	Pclass	Name	Sex	Age
0	1	0	3	Braund, Mr. Owen Harris	male	22
1	2	1	1	Cumings, Mrs. John Bradley	female	38
2	3	1	3	Heikkinen, Miss. Laina	female	26
3	4	1	1	Futrelle, Mrs. Jacques Heath	female	35
4	5	0	3	Allen, Mr. William Henry	male	35

- `df.describe()`
  - Provides data statistics such as mean, standard deviation, min, max, percentiles

	PassengerId	Survived	Pclass	Age
count	5.000000	5.000000	5.000000	5.00000
mean	3.000000	0.600000	2.200000	31.20000
std	1.581139	0.547723	1.095445	6.83374
min	1.000000	0.000000	1.000000	22.00000
25%	2.000000	0.000000	1.000000	26.00000
50%	3.000000	1.000000	3.000000	35.00000
75%	4.000000	1.000000	3.000000	35.00000
max	5.000000	1.000000	3.000000	38.00000

- `df ['Age']`
  - Selecting a column by name, returns series
  - `df.['Age'].tolist()`, returns python list (no labels)

```
>>> df['Age']
0    22
1    38
2    26
3    35
4    35
```

- `df [['Age','Pclass']]` Selecting multiple columns

```
>>> df[['Age','Pclass']]
   Age  Pclass
0   22       3
1   38       1
2   26       3
3   35       1
4   35       3
```

https://pandas.pydata.org/pandas-docs/stable/getting_started/index.html

pandas

	PassengerId	Survived	Pclass	Name	Sex	Age
0	1	0	3	Braund, Mr. Owen Harris	male	22
1	2	1	1	Cumings, Mrs. John Bradley	female	38
2	3	1	3	Heikkinen, Miss. Laina	female	26
3	4	1	1	Futrelle, Mrs. Jacques Heath	female	35
4	5	0	3	Allen, Mr. William Henry	male	35

- `df['Age']>32`
  - Conditions returns rows with true / false

```
>>> df['Age']>32
0    False
1     True
2    False
3     True
4     True
Name: Age, dtype: bool
```

- `df [0:3]`
  - Selecting via rows, returns df
  - `df [ df['Age']>32 ]`

```
>>> df[0:3]
   PassengerId  Survived  Pclass                                               Name     Sex  Age
0            1         0       3                            Braund, Mr. Owen Harris    male   22
1            2         1       1  Cumings, Mrs. John Bradley (Florence Briggs Th...  female   38
2            3         1       3                             Heikkinen, Miss. Laina  female   26
```

https://pandas.pydata.org/pandas-docs/stable/getting_started/index.html

pandas df.loc

	PassengerId	Survived	Pclass	Name	Sex	Age
0	1	0	3	Braund, Mr. Owen Harris	male	22
1	2	1	1	Cumings, Mrs. John Bradley	female	38
2	3	1	3	Heikkinen, Miss. Laina	female	26
3	4	1	1	Futrelle, Mrs. Jacques Heath	female	35
4	5	0	3	Allen, Mr. William Henry	male	35

- `df.loc[2:4, ['Sex', 'Age']]`

```
>>> df.loc[2:4,['Sex','Age']]
      Sex  Age
2  female   26
3  female   35
4    male   35
```

- `df.loc[df['Age'] == 35]`
  - Returns rows (in df) that matches condition in column

```
>>> df['Age']==35
0    False
1    False
2    False
3     True
4     True
```

```
>>> df.loc[df['Age'] == 35]
   PassengerId  Survived  Pclass                         Name     Sex  Age
3            4         1       1  Futrelle, Mrs. Jacques Heath  female   35
4            5         0       3       Allen, Mr. William Henry    male   35
```

- `df.loc[___].index`
  - Returns index(s) (row label(s))

```
>>> df.loc[df['Age'] == 35].index
Int64Index([3, 4], dtype='int64')
```

https://pandas.pydata.org/docs/getting_started/10min.html

matplotlib.pyplot

- `import matplotlib.pyplot as plt`
- `plt.plot( <x-data>,<y-data>,<colour and marker>,label='')`
  - Colour and marker is also called fmt (format)
  - 'bx' – plot blue line with x markers
- `plt.xlabel('')`
- `plt.ylabel('')`
- `plt.title('Title')`
- `plt.legend()`
- `plt.show()`

Markers

character	description
'.'	point marker
','	pixel marker
'o'	circle marker
'v'	triangle_down marker
'^'	triangle_up marker
'<'	triangle_left marker
'>'	triangle_right marker
'1'	tri_down marker
'2'	tri_up marker
'3'	tri_left marker
'4'	tri_right marker
's'	square marker
'p'	pentagon marker
'*'	star marker
'h'	hexagon1 marker
'H'	hexagon2 marker
'+'	plus marker
'x'	x marker
'D'	diamond marker
'd'	thin_diamond marker
'|'	vline marker
'_'	hline marker

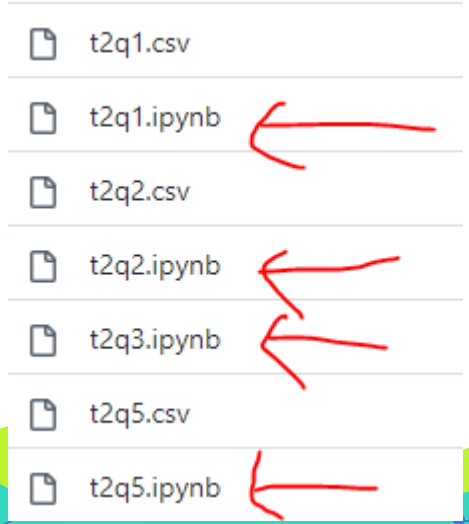
https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.html

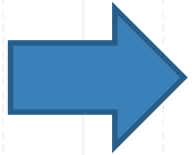
For codes.. Go to

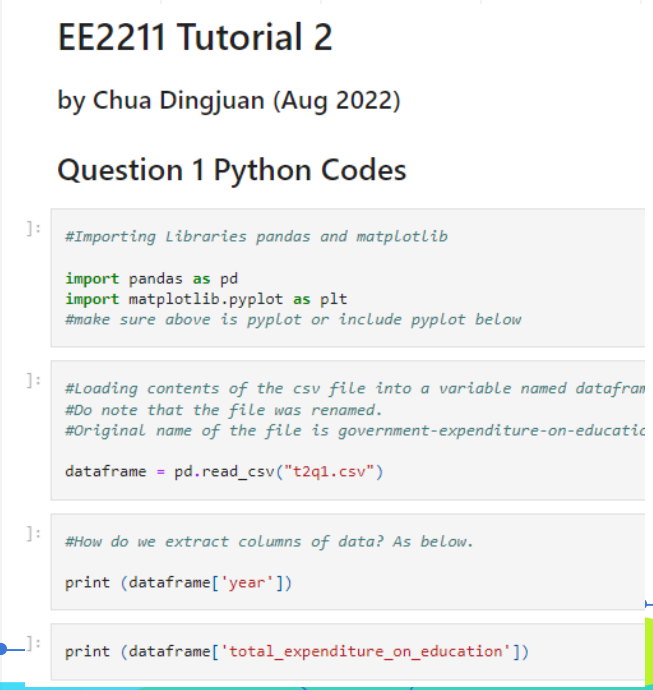
github.com/elechuad/mltutdj

Option 1 :

You can access codes and comments via the .ipynb files





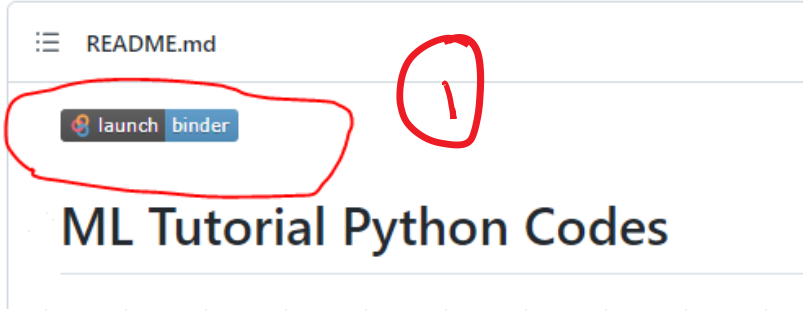


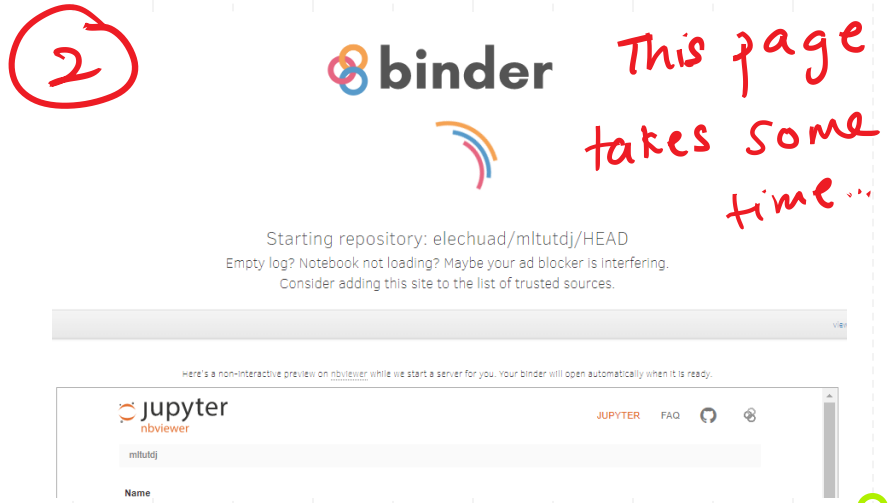
For codes.. Go to

`github.com/elechuad/mltutdj`

Option 2 :

You can INTERACT with codes via the binder.



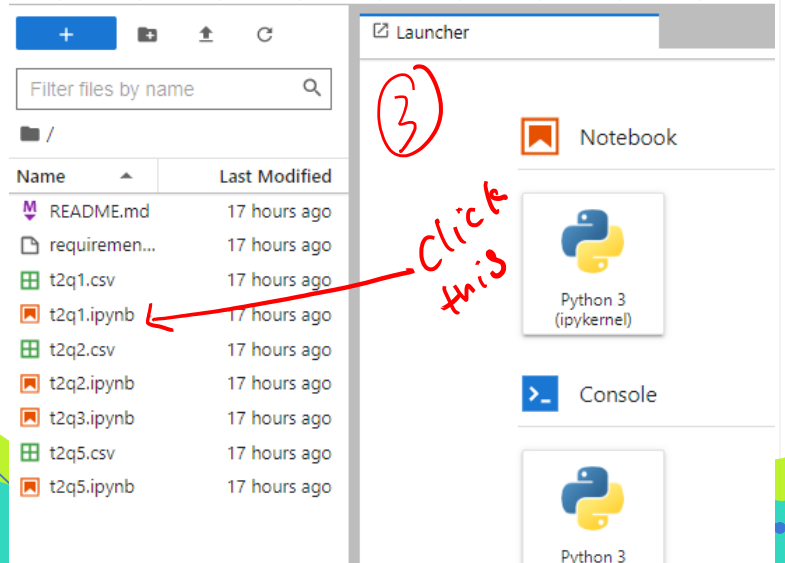


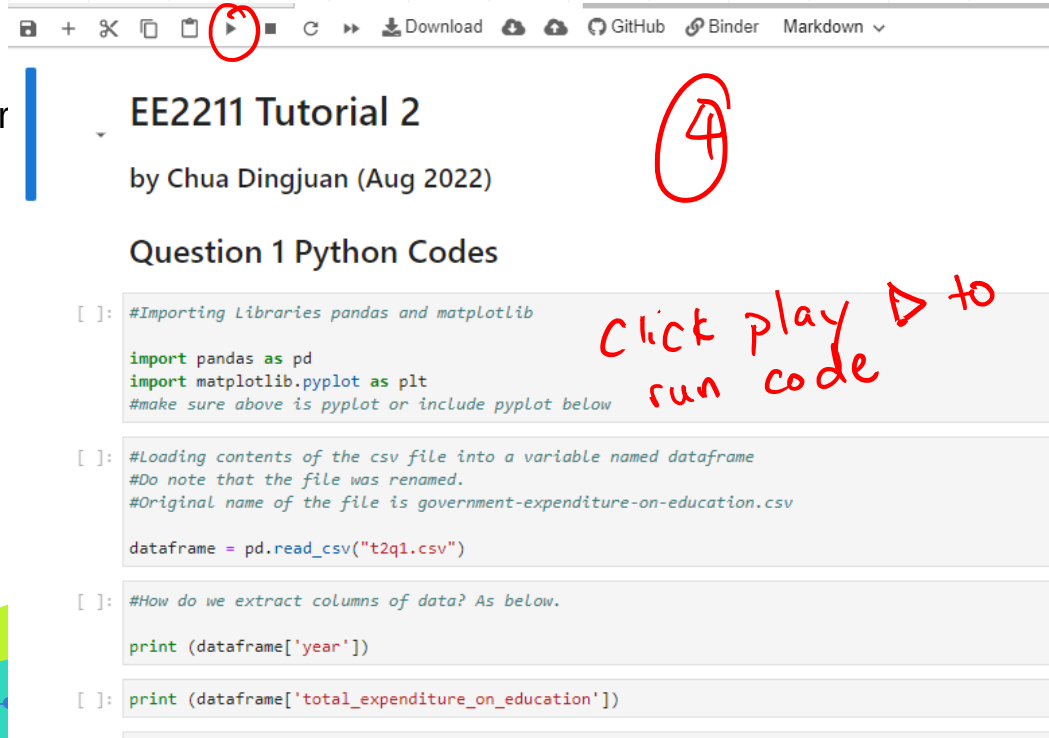
For codes.. Go to

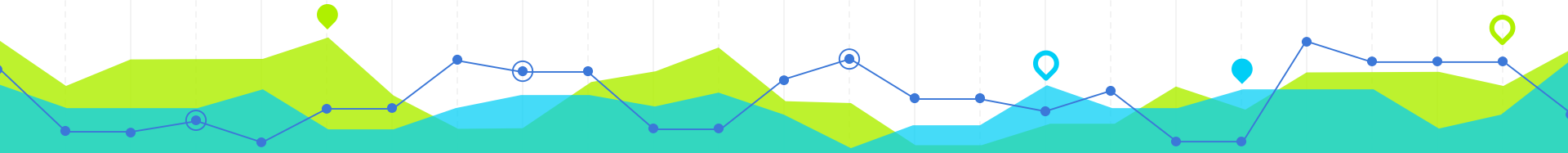
github.com/elechuad/mltutdj

Option 2 :

You can INTERACT/MODIFY codes via the binder







Discussion of Solutions 2

Q 1, 2, 4, 5, 3, 6, 7, 8

Question1

A Comma Separated Values (CSV) file is a plain text file that contains a list of data. These files are often used for exchanging data between different applications. Download the file "government-expenditure-on-education.csv" from https://data.gov.sg/dataset/government-expenditure-on-education. Plot the educational expenditure over the years. (Hint: you might need "`import pandas as pd`" and "`import matplotlib.pyplot as plt`".)

	A	B
1	year	total_expenditure_on_education
2	1981	942517
3	1982	1358430
4	1983	1611647
5	1984	1769728
6	1985	1812376
7	1986	1641893
8	1987	1654115
9	1988	1604473
10	1989	1765250
11	1990	2056374
12	1991	2816371
13	1992	2597894
14	1993	2902886
15	1994	3318956
16	1995	3443857
17	1996	3771955
18	1997	4449754
19	1998	4853120
20	1999	4857488
21	2000	5867507

STEPS:

- Import the .csv file
- Extract the Year data
- Extract the Expenditure data

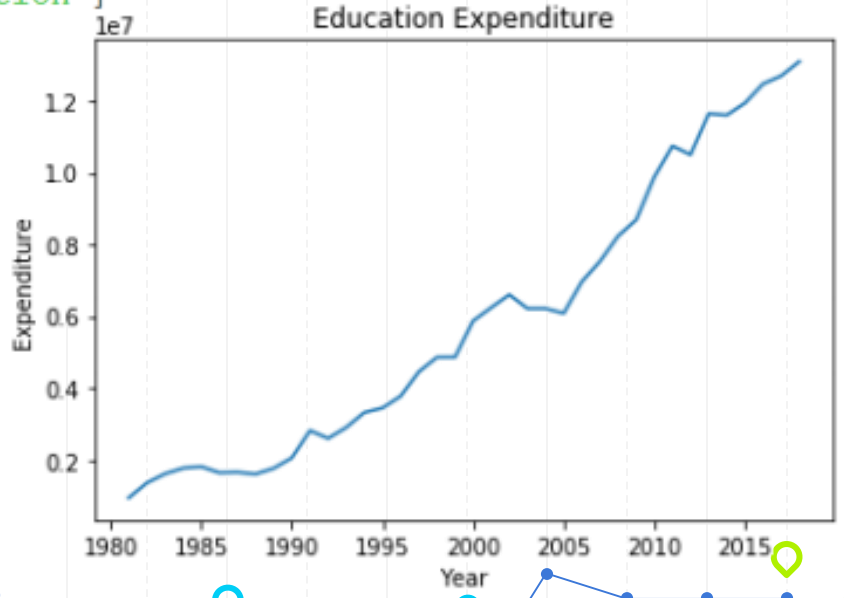
- Plot year on x-axis and expenditure on y-axis

Question1

```
import pandas as pd
import matplotlib.pyplot as plt
#make sure above is pyplot or include pyplot below

dataframe = pd.read_csv("t2q1.csv")

expenditure = dataframe['total_expenditure_on_education']
year = dataframe['year']

plt.plot(year,expenditure)

plt.xlabel('Year')
plt.ylabel('Expenditure')
plt.title('Education Expenditure')
plt.show()
```

import libraries

.tolist()

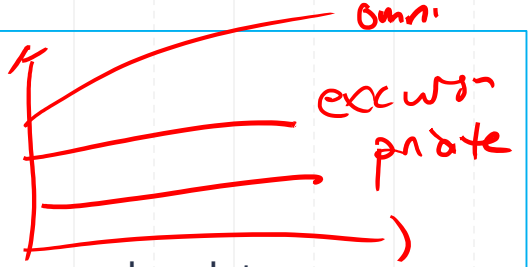


Question2

Download the CSV file from https://data.gov.sg/dataset/annual-motor-vehicle-population-by-vehicle-type. Extract and plot the number of Omnibuses, Excursion buses and Private buses over the years as shown below. (Hint: you might need "`import pandas as pd`" and "`import matplotlib.pyplot as plt`".)

110	2013	Goods & Other Vehicles	Very Heavy Goods Vehicles (VHGVs)	16170
111	2014	Goods & Other Vehicles	Very Heavy Goods Vehicles (VHGVs)	16712
112	2005	Buses	Omnibuses	3599
113	2006	Buses	Omnibuses	3785
114	2007	Buses	Omnibuses	3761
115	2008	Buses	Omnibuses	3854
116	2009	Buses	Omnibuses	4045
117	2010	Buses	Omnibuses	3981
118	2011	Buses	Omnibuses	4112
119	2012	Buses	Omnibuses	4212
120	2013	Buses	Omnibuses	4552
121	2014	Buses	Omnibuses	4756
122	2005	Buses	School buses (CB)	1873
123	2006	Buses	School buses (CB)	1881
124	2007	Buses	School buses (CB)	1851
125	2008	Buses	School buses (CB)	1852
126	2009	Buses	School buses (CB)	1849
127	2010	Buses	School buses (CB)	1845
128	2011	Buses	School buses (CB)	1844
129	2012	Buses	School buses (CB)	1839
130	2013	Buses	School buses (CB)	1847
131	2014	Buses	School buses (CB)	1845
132	2005	Buses	Private buses	2557
133	2006	Buses	Private buses	2577
134	2007	Buses	Private buses	2628
135	2008	Buses	Private buses	2739
136	2009	Buses	Private buses	2795
137	2010	Buses	Private buses	2842

STEPS:

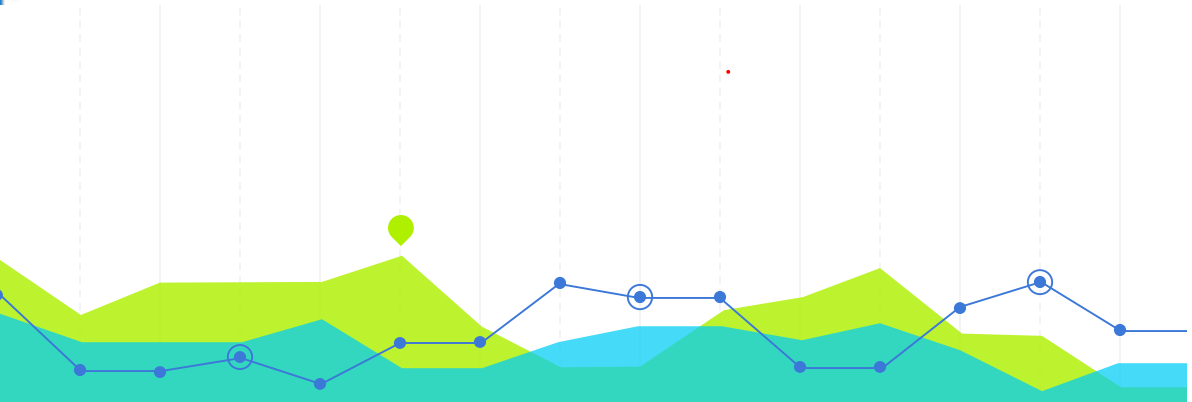


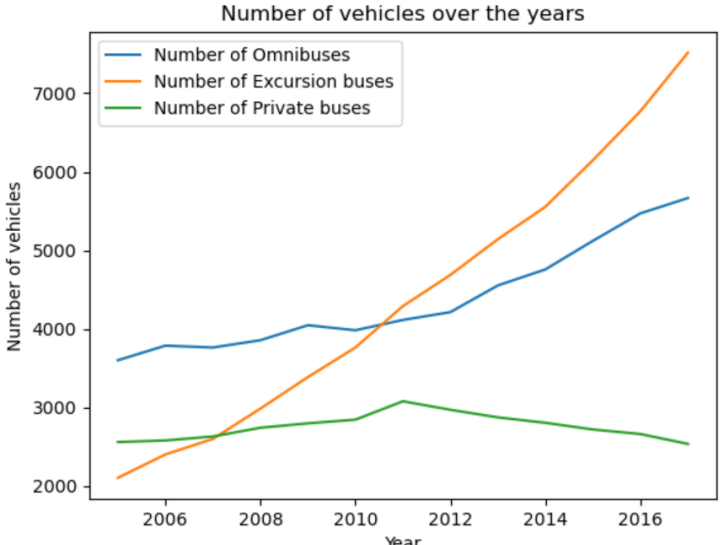
- Import the .csv file
- Extract the Omnibuses number data
- Extract the Excursion buses number data
- Extract the Private buses number data

- Plot year on x-axis and different buses on y-axis

Question2

```
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv("t2q2.csv")

List1 = df.loc[df['type']=='Omnibuses']
List2 = df.loc[df['type']=='Excursion buses']
List3 = df.loc[df['type']=='Private buses']

plt.plot(List1['year'], List1['number'], label = 'Number of Omnibuses')
plt.plot(List2['year'], List2['number'], label = 'Number of Excursion buses')
plt.plot(List3['year'], List3['number'], label = 'Number of Private buses')
plt.xlabel('Year')
plt.ylabel('Number of vehicles')
plt.title('Number of vehicles over the years')
plt.legend()
plt.show()
```





Question4

You are given a set of data for supervised learning. A sample block of data looks like this:

“

1.2234, 0.3302, 123.50, 0.0081, 30033.81, 1
1.3456, 0.3208, 113.24, 0.0067, 29283.18, -1
0.9988, 0.2326, 133.45, 0.0093, 36034.33, 1
1.1858, 0.4301, 128.55, 0.0077, 34037.35, 1
1.1533, 0.3853, 116.70, 0.0066, 22033.58, -13
1.2755, 0.3102, 118.30, 0.0098, 30183.65, 1
1.0045, 0.2901, 123.52, 0.0065, 31093.98, -1
1.1131, 0.3912, 113.15, 0.0088, 29033.23, -1

”

anomaly?

Each row corresponds to a sample data measurement with 5 input features and 1 response.

(a) What kind of undesired effect can you anticipate if this set of raw data is used for learning?

(b) How can the data be preprocessed to handle this issue?

Question4

You are given a set of data for supervised learning. A sample block of data looks like this:

“ 1.2234, 0.3302, 123.50, 0.0081, 30033.81, 1
1.3456, 0.3208, 113.24, 0.0067, 29283.18, -1
0.9988, 0.2326, 133.45, 0.0093, 36034.33, 1
1.1858, 0.4301, 128.55, 0.0077, 34037.35, 1
1.1533, 0.3853, 116.70, 0.0066, 22033.58, -13
1.2755, 0.3102, 118.30, 0.0098, 30183.65, 1
1.0045, 0.2901, 123.52, 0.0065, 31093.98, -1
1.1131, 0.3912, 113.15, 0.0088, 29033.23, -1 ”

Each row corresponds to a sample data measurement with 5 input features and 1 response.

(a) What kind of undesired effect can you anticipate if this set of raw data is used for learning?

(b) How can the data be preprocessed to handle this issue?

Ans:

(a) Those features with very large values may overshadow those with very small values.

(b) We can either use min-max or z-score normalization to resolve the problem.

Question5

The Pima Indians Diabetes Dataset involves predicting the onset of diabetes within 5 years in Pima Indians given medical details. Download the Pima-Indians-Diabetes data from https://raw.githubusercontent.com/jbrownlee/Datasets/master/pima-indians-diabetes.data.csv.

It is a binary (2-class) classification problem. The number of observations for each class is not balanced. There are 768 observations with 8 input variables and 1 output variable. The variable names are as follows:

0. Number of times pregnant.
1. Plasma glucose concentration a 2 hours in an oral glucose tolerance test.
2. Diastolic blood pressure (mm Hg).
3. Triceps skinfold thickness (mm).
4. 2-Hour serum insulin (mu U/ml).
5. Body mass index (weight in kg/(height in m)^2).
6. Diabetes pedigree function.
7. Age (years).
8. Class variable (0 or 1).

	A	B	C	D	E	F	G	H	I
1	6	148	72	35	0	33.6	0.627	50	1
2	1	85	66	29	0	26.6	0.351	31	0
3	8	183	64	0	0	23.3	0.672	32	1
4	1	89	66	23	94	28.1	0.167	21	0
5	0	137	40	35	168	43.1	2.288	33	1
6	5	116	74	0	0	25.6	0.201	30	0
7	3	78	50	32	88	31	0.248	26	1
8	10	115	0	0	0	35.3	0.134	29	0
9	2	197	70	45	543	30.5	0.158	53	1
10	8	125	96	0	0	0	0.232	54	1
11	4	110	92	0	0	37.6	0.191	30	0
12	10	168	74	0	0	38	0.537	34	1

(a) Print the summary statistics of this data set.
(b) Count the number of "0" entries in columns [1,2,3,4,5].
(c) Replace these "0" values by "NaN".

(Hint: you might need the "`.describe()`" and "`.replace(0, numpy.NaN)`" functions "`from pandas import read_csv`".)

Question5

(a) Print the summary statistics of this data set.

(a)
```
df = pd.read_csv('t2q5.csv', header=None)
df.describe()
```

(b) Count the number of "0" entries in columns [1,2,3,4,5].

(b)
```
( df[[1,2,3,4,5]] == 0 ).sum()
```

(c) Replace these "0" values by "NaN".

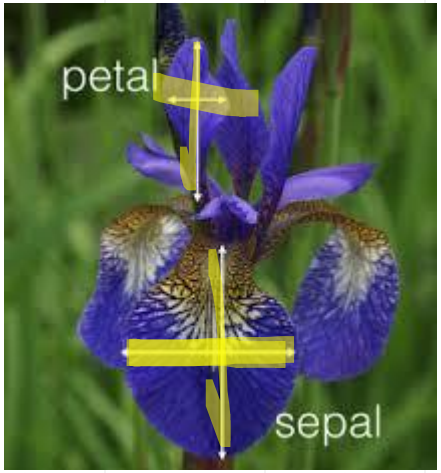
(c)
```
import numpy
df[[1,2,3,4,5]] = df[[1,2,3,4,5]].replace(0, numpy.NaN)
```

```
>>> df
     0      1     2     3      4     5      6   7  8
0    6  148.0  72.0  35.0    NaN  33.6  0.627  50  1
1    1   85.0  66.0  29.0    NaN  26.6  0.351  31  0
2    8  183.0  64.0   NaN    NaN  23.3  0.672  32  1
```

```
print(df.isnull().sum())
```

Question3

The “iris” flower data set consists of measurements such as the length, width of the petals, and the length, width of the sepals, all measured in centimeters, associated with each iris flower. Get the data set “from sklearn.datasets import load_iris” and do a scatter plot as shown below. (Hint: you might need “`from pandas.plotting import scatter_matrix`”)



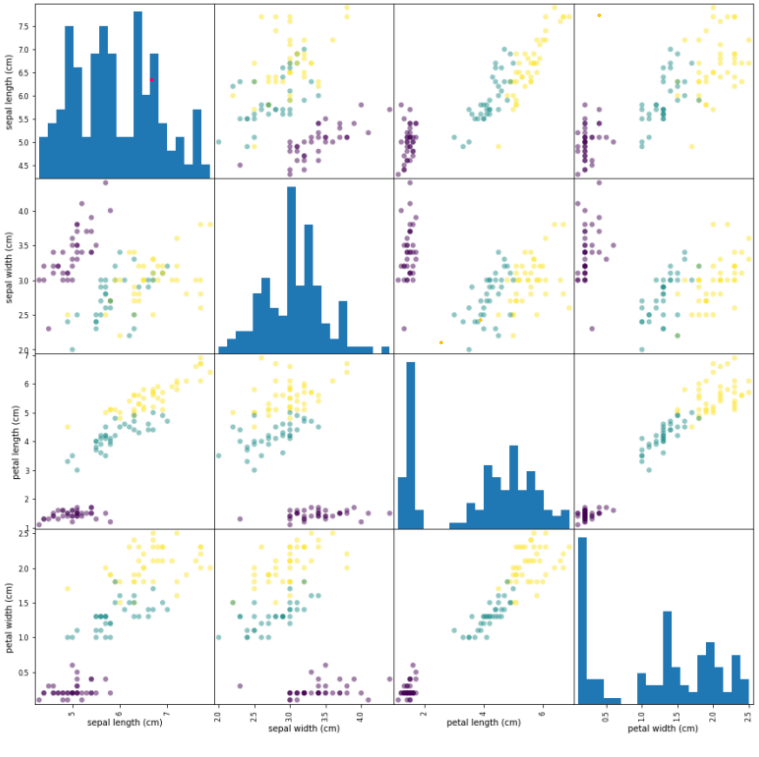
- scikit-learn is a python-based popular and free ML library
- Supports many ML models for exploration w tuning parameters
- Active community / support

- Famous Iris dataset
  - 1936, Edgar Anderson collected 50 samples of 3 different species of iris (150 samples total)
  - Each flower sample, he noted the sepal length and width, petal length and width and the species.
  - Fisher’s paper : linear discriminant analysis to id species based on measurements (supervised)

```
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
```

Question3

The “iris” flower data set consists of measurements such as the length, width of the petals, and the length, width of the sepals, all measured in centimeters, associated with each iris flower. Get the data set “from sklearn.datasets import load_iris” and do a scatter plot as shown below. (Hint: you might need “`from pandas.plotting import scatter_matrix`”)



STEPS:

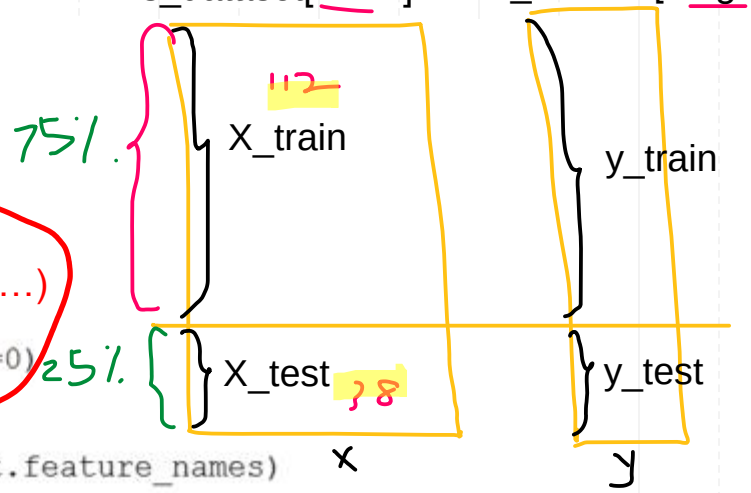
- Load the iris dataset
- Split and extract the data using train_test_split
- Convert to dataframe for plotting
- Plot in scatter matrix form
  - Scatter Matrix (pairs plot) plots all numeric variables in dataset against each other
  - Diagonal : dist of variables
  - Other cells : correlation plot (scatter plot)

Question3

```
import pandas as pd
import sklearn
from sklearn.datasets import load_iris
iris_dataset = load_iris()

from sklearn.model_selection import train_test_split

(this next step is extra, not really needed in the context of plotting…)
X_train, X_test, y_train, y_test = train_test_split(
iris_dataset['data'], iris_dataset['target'], random_state=0)

iris_dataframe = pd.DataFrame(X_train, columns=iris_dataset.feature_names)


from pandas.plotting import scatter_matrix
grr = pd.plotting.scatter_matrix(iris_dataframe, c=y_train, figsize=(15, 15), marker='o',
hist_kwds={'bins': 20})

import matplotlib.pyplot as plt
plt.show()
```

5.1,3.5,1.4,0.2, Iris-setosa

Iris_dataset['data'] Iris_dataset['target']



Question6

We collect data to train an AI system for face recognition. When the collected data reflect the structure of populations, AI systems that learn from the data are not biased.

(1) True

(2) False

Ans: False (such collected data do not describe the minority of the populations equally well with the majority)

Question7

Disease Outbreak Response System Condition (DORSCON) in Singapore is a colour-coded framework that shows the current disease situation. The framework provides us with general guidelines on what needs to be done to prevent and reduce the impact of infections. There are 4 statuses – Green, Yellow, Orange and Red, depending on the severity and spread of the disease. Which type of data does DORSCON belong to ?

(1) Categorical; (2) Ordinal; (3) Continuous; (4) Interval

Ans: (2) Ordinal

Question8

A boxplot is a standardized way of displaying the dataset based on a five-number summary: the minimum, the maximum, _BLANK1_, and the first and third quartiles, where the number of data points that fall between the first and third quartiles amounts to _BLANK2_ percent of the total number of data on display.

Ans:
BLANK1: Median
BLANK2: 50%