



ARTIFICIAL  
INTELLIGENCE SOCIETY

# Introduction to UCL AI Society Machine Learning tutorials

22nd Oct 2020

Presented by Danny Toeun Kim

# Tutorials Team

**Danny Toeun Kim**



**Head of Tutorials**

MEng  
Computer Science

**Kamen Brestnichki**



**ML Officer**

MSc  
Computational Statistics and Machine Learning

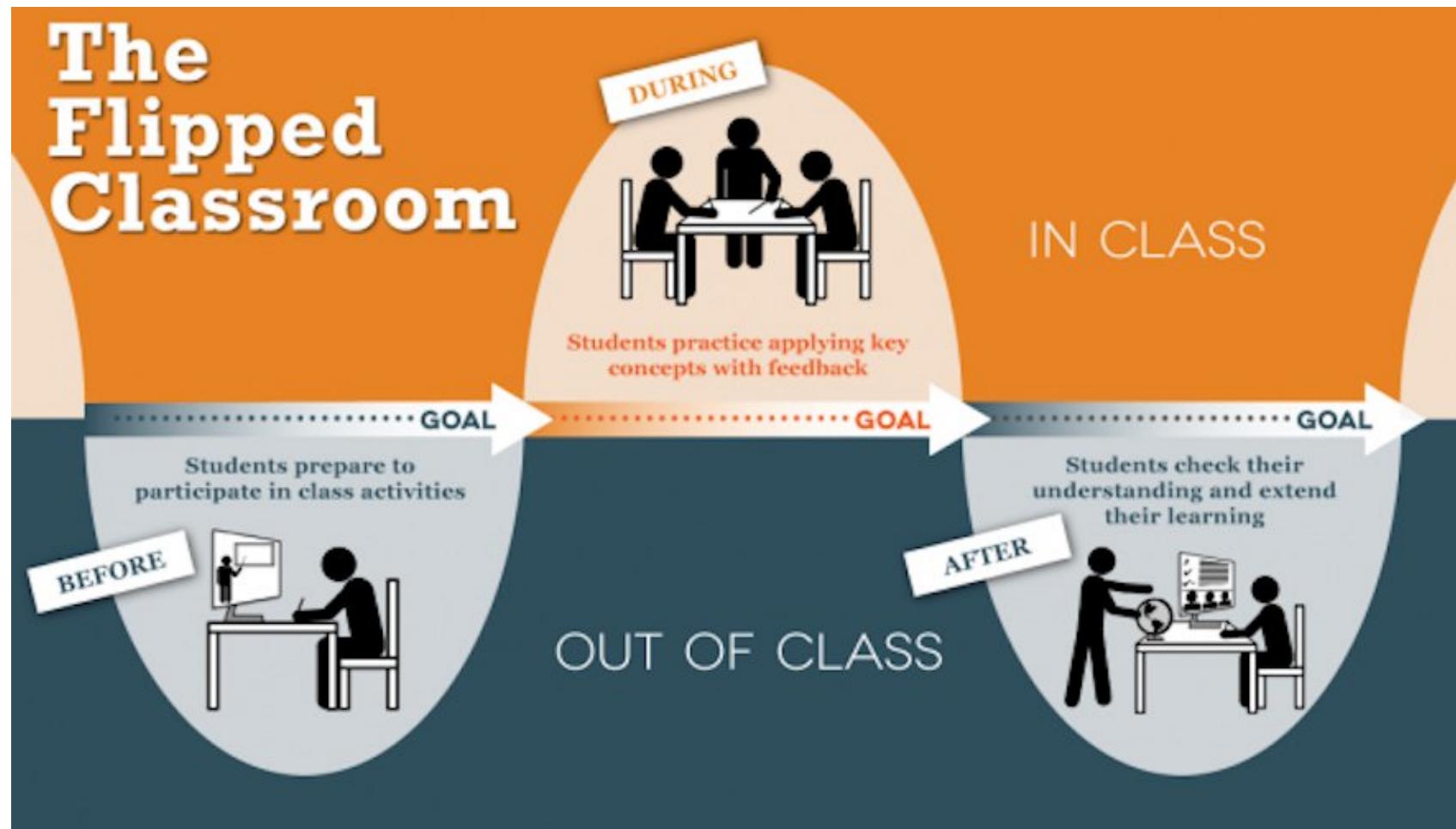
**Jianqiao Mao**



**ML Officer**

MSc  
Integrated Machine Learning System

# Flipped Tutorials



# Tutorial Schedule

WEEK	MON	TUES	WED	THURS	FRI	SAT	SUN
(Oct) 8	19	20	21	<b>OCT 22</b> Introduction video Set-up video *Notebook 1 RELEASE			
9	26			29			
(Nov) 10	2			<b>NOV 5</b> TUT1 live class + Solution 1 *Notebook2 RELEASE			
11	9			12			
12	16			<b>NOV 19</b> TUT2 live class + Solution 2 *Notebook3 RELEASE			
13	23			26			
(Dec) 14	30	1	2	<b>DEC 3</b> TUT3 live class + Solution 3 *Notebook4 RELEASE			
15				10			
16				<b>DEC 17</b> TUT4 live class + Solution 4 *Notebook5 RELEASE (subject to change)	18 TERM1 END		
17	21						

Numerical Computation and Visualisation for ML

Machine Learning algorithms and practices

Introduction to Deep Neural Networks

Applications of Deep Learning

# Problem notebook

## How do you initialise numpy arrays / matrix?

```
"""
TODO: Replace 'None's with appropriate answers
e.g) b = np.None((2, 2)) --> np.ones((2, 2))
"""

# create a matrix full of ones
b = np.None((2, 2))
print("Matrix b")
print(b)
```

```
# create a matrix full of zeros
c = np.None((2, 3))
print("\nMatrix c")
print(c)
```

```
# create an identity matrix
d = np.None(3)
print("\nMatrix d")
print(d)
```

```
# create a matrix filled with random numbers between 0 and 1
e = np.None((2, 2))
print("\nMatrix e")
print(e)
```

```
# create an array which has 0-9 as its element in sorted order
# expected output: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
f = np.None(10)
print("\nMatrix f")
print(f)
```

```
# create a matrix placeholder, without initializing entries (elements in
g = np.None((5, 3))
print("\nMatrix g")
print(g)
```

## 1.2 Matrix Calculation

- `np.transpose()` : Transpose of an array
- `np.dot(a, b)` : Dot product of two arrays
- `np.linalg.inv()` : Inverse matrix of an array (only valid to square matrix, whose dimension is  $n * n$ )
- `np.diagonal()` : Diagonal components of an array
- `a.reshape(row = x, column = y)` : Reshape an array to the given dimension

```
# To Do: By using a masking operation, Extract the movies whose 'Metascore' is bigger than 95,
# and sort the result from the most recent to the least recent ones
None
```

```
# To Do: Extract movies who are directed by one of UCL's alumni ---> Hint: Tenet, Inception
None
```

### 2.2.1 Pandas Exercise

To Do: Extract the movie list that meets these requirements:

- 1. Released after 2010 (key = 'Year') (including the year 2010)
- 1. Runtime is shorter than 150 minutes (key = 'Runtime (Minutes)')
- 1. Rating is above 8.0 (key = 'Rating')

Print out only the first 3 movies from the result.

```
None
```

### 2.3 How to deal with Missing Data

To represent missing data, Pandas uses `np.nan` (this is the `np` from the NumPy tutorial). Data scientists and machine learning engineers sometimes just remove missing data. However, it heavily depends on which data are missing, how big the missing data are and so on. You can fill the missing part with 0, with the mean value of the column, with the mean value of only the 10 closest values in the column or anything else that might seem appropriate. It is important for you to choose the way you are going to deal with missing data. Here are some methods to help you with that:

- `isnull()`: returns True or False, depending on the cell's null status.
- `sum()`: This can be used as a trick when you count the number of Trues. Once the Dataframe is filtered through the `isnull()` function, the sum of all Trues in a column gives you how many fields have missing data in them.
- `dropna()`: deletes any row that contains at least one null value.
- `fillna(value)`: Fills missing values with the given values.

```
movie.isnull()
```



Thank you!

Happy Coding!

---

Tutorial Repository:

<https://github.com/UCLAIS/Machine-Learning-Tutorials>

