

# PREVIOUS HOMEWORK

- Bonus points, explain the reason based on what you learned in lecture

```
# Why are these two elements almost the same? additional 10 points
print(f'pi : {EM_model.pi}')
print(f'count / total : {np.bincount(EM_pred) / 150}')
```

- The answer lies in online lecture and lecture note

## EM Algorithm

➤ M-step: find  $z^k(x)$  and  $\theta^{k+1}$  with maximizing the EXPECTATION from E-step

1) Calculate  $z^k(x)$

Let us now define

$$w(x|y; \theta^k) \triangleq \frac{q(x, y; \theta^k)}{f(y; \theta^k)} = \frac{q(x, y; \theta^k)}{\int_x q(x, y; \theta^k) dx}$$

then note that

$$\begin{aligned} F(z, \theta^k) &\triangleq \int_x z(x) \log \left[ \frac{w(x|y; \theta^k) f(y; \theta^k)}{z(x)} \right] dx \\ &= \log f(y; \theta^k) - D(z(x) \| w(x|y; \theta^k)) \end{aligned}$$

where

$$D(z_1(x) \| z_2(x)) \triangleq \int_x z_1(x) \log \left[ \frac{z_1(x)}{z_2(x)} \right] dx$$

(D is the Kullback-Leibler distance)

In order to maximize  $F(z, \theta^k)$ ,

$$z^k(x) = w(x|y; \theta^k) = P(X|Y, \theta^k),$$

a priori = a posteriori

Bio Computing & Machine Learning (BCML) Lab

## <과제물 작성시 주의사항>

### [공통]

과제물 제출시 완성된 **소스파일 및 보고서**를 반드시 '**HW\_03\_학번.zip**' 형식으로 압축하여 첨부합니다.

(이름 약어.py, HW\_03\_학번.pdf )

### [소스파일] - 40점

1. 소스파일은 **.py파일만 작성**하며 반드시 문제에서 지시 또는 요구한 조건에 맞추어서 작성합니다.  
(jupyter로 작성하였어도 코드를 제출 시 py파일로 작성하여 제출하여야 합니다.)
2. 각 코드마다 **반드시 주석을 달아 주셔야 합니다.** 주석을 달지 않을 경우, 부분적으로 감점이 있을 수 있습니다.
3. 결과가 올바르게라도 과정이 옳지 않을 경우, 부분적으로 감점이 있을 수 있습니다.
4. 제출한 파일이 실행되지 않을 경우, 제출한 과제물은 0점 처리됩니다.

### [보고서] - 60점

1. **PDF**로 제출하며, 표지를 포함해야 합니다.
2. 보고서에는 **#1(데이터에 대한 설명과 목적), #2(네트워크 구조에 대한 설명), #3(소스 코드에 대한 설명), #4(실행 결과 + Plot), #5(참고문헌)**이 포함되어야 합니다.
3. 자신의 코드 혹은 오픈소스 코드에 대한 설명이 부족할 시 감점 당할 수 있습니다.
4. **실행 결과는 실행 결과를 캡처하여 첨부하도록 합니다.**
5. 참고문헌은 반드시 적어도 한 개 이상을 명시하여야 합니다.



# CONVOLUTION NEURAL NETWORK USING TENSORFLOW

Machine learning homework-3

Assistant : Choongseop Lee, cndtjq97@gmail.com

# CNN USING TF

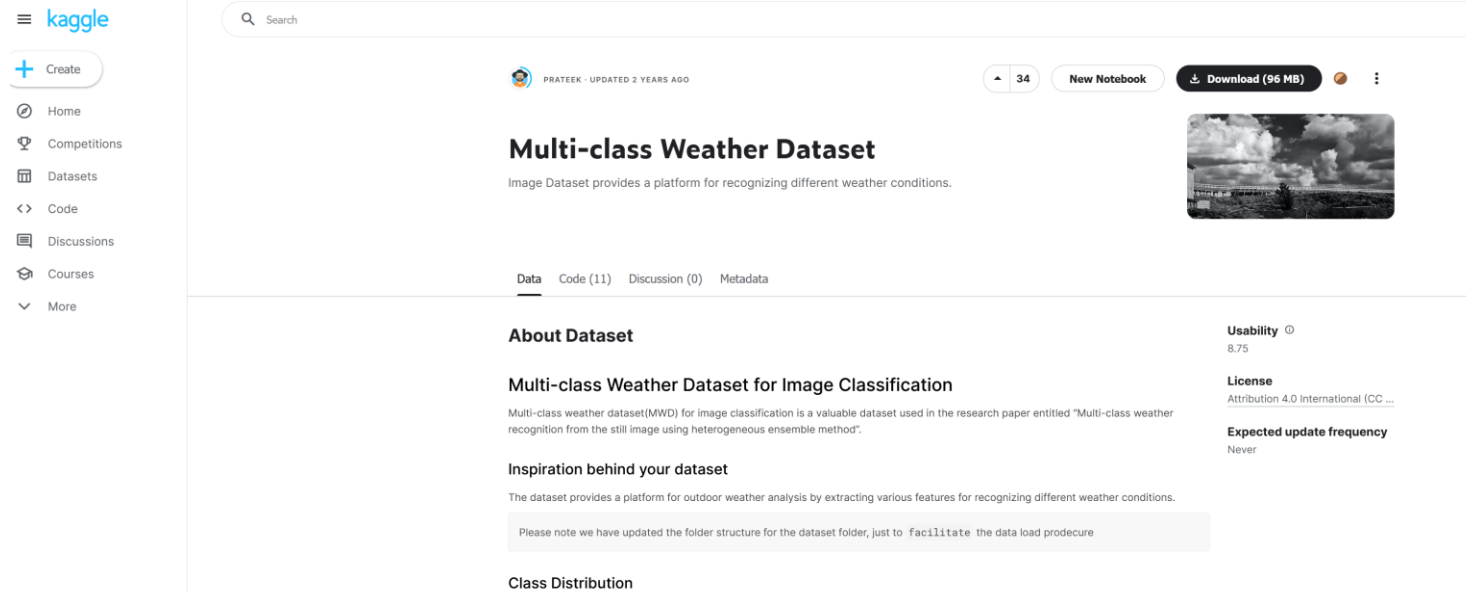
- Implementation of Convolution Neural Network Using TensorFlow



- You can implement your own network or use open source (Make sure to state the source)
- All CNN-based models may be used to increase accuracy
- However, a **deep understanding** of CNN and the **progress of the code** should be shown in the report

# DATASET #1

- <https://www.kaggle.com/datasets/pratik2901/multiclass-weather-dataset>



The screenshot shows the Kaggle interface for the 'Multi-class Weather Dataset' by Prateek, updated 2 years ago. The dataset is an image dataset for recognizing different weather conditions. It has 34 versions, a 'New Notebook' button, and a 'Download (96 MB)' button. The page includes a sidebar with navigation options like Home, Competitions, Datasets, Code, Discussions, Courses, and More. The main content area has tabs for Data, Code (11), Discussion (0), and Metadata. The 'About Dataset' section describes it as a multi-class weather dataset (MWD) for image classification, used in a research paper. It also mentions the inspiration behind the dataset and a note about updated folder structure. On the right, there are sections for Usability (8.75), License (Attribution 4.0 International), and Expected update frequency (Never).

**Multi-class Weather Dataset**  
Image Dataset provides a platform for recognizing different weather conditions.

**About Dataset**

**Multi-class Weather Dataset for Image Classification**  
Multi-class weather dataset(MWD) for image classification is a valuable dataset used in the research paper entitled "Multi-class weather recognition from the still image using heterogeneous ensemble method".

**Inspiration behind your dataset**  
The dataset provides a platform for outdoor weather analysis by extracting various features for recognizing different weather conditions.

Please note we have updated the folder structure for the dataset folder, just to facilitate the data load procedure

**Class Distribution**

**Usability** 8.75

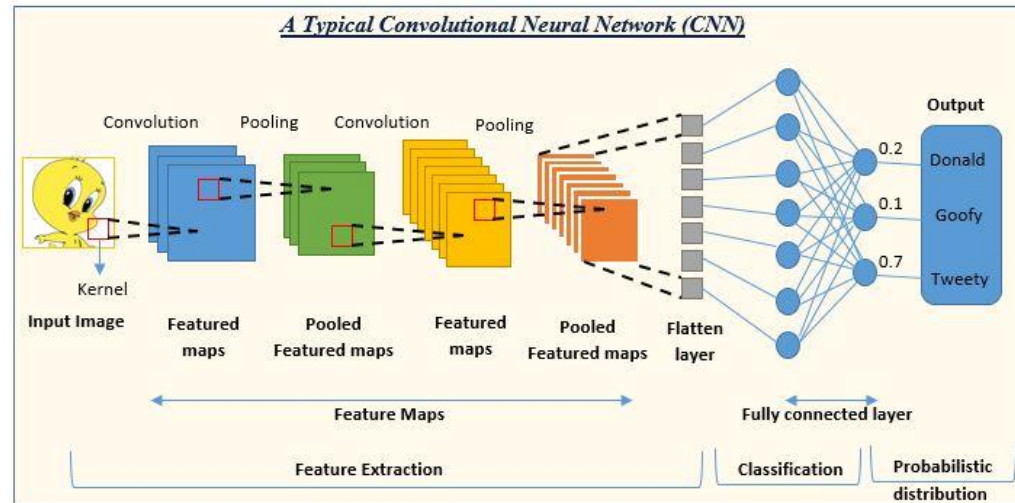
**License**  
Attribution 4.0 International (CC ...)

**Expected update frequency**  
Never

- Using a kaggle open dataset
- The report should state the description and objectives of the dataset.

# NETWORK STRUCTURE #2

- The network structure and each layer need to be explained



Operation Layer		Number of Filters	Size of Each Filter	Stride Value	Padding Value	Size of Output Image
<b>Input image</b>						
		-	-	-	-	$224 \times 224 \times 3$
<b>Convolution Layer (two times)</b>	Convolution	64	$3 \times 3 \times 3$	$1 \times 1$	$1 \times 1$	$224 \times 224 \times 64$
	ReLU	-	-	-	-	$224 \times 224 \times 64$
<b>Pooling Layer</b>	Max pooling	1	$2 \times 2$	$2 \times 2$	0	$112 \times 112 \times 64$
<b>Convolution Layer (two times)</b>	Convolution	128	$3 \times 3 \times 64$	$1 \times 1$	$1 \times 1$	$112 \times 112 \times 128$
	ReLU	-	-	-	-	$112 \times 112 \times 128$
<b>Pooling Layer</b>	Max pooling	1	$2 \times 2$	$2 \times 2$	0	$56 \times 56 \times 128$
<b>Convolution Layer (four times)</b>	Convolution	256	$3 \times 3 \times 128$	$1 \times 1$	$1 \times 1$	$56 \times 56 \times 256$
	ReLU	-	-	-	-	$56 \times 56 \times 256$
<b>Pooling Layer</b>	Max pooling	1	$2 \times 2$	$2 \times 2$	0	$28 \times 28 \times 256$
<b>Convolution Layer (four times)</b>	Convolution	512	$3 \times 3 \times 256$	$1 \times 1$	$1 \times 1$	$28 \times 28 \times 512$
	ReLU	-	-	-	-	$28 \times 28 \times 512$
<b>Pooling Layer</b>	Max pooling	1	$2 \times 2$	$2 \times 2$	0	$14 \times 14 \times 512$
<b>Convolution Layer (four times)</b>	Convolution	512	$3 \times 3 \times 512$	$1 \times 1$	$1 \times 1$	$14 \times 14 \times 512$
	ReLU	-	-	-	-	$14 \times 14 \times 512$
<b>Pooling Layer</b>	Max pooling	1	$2 \times 2$	$2 \times 2$	0	$7 \times 7 \times 512$
<b>Inner Product Layer</b>	Fully connected	-	-	-	-	4096
	ReLU	-	-	-	-	4096

- You have to make structural figure and table with your hands
- Write down the selected layers and functions
- And explain how they work, **respectively**
- Conv, pool, FC layers and regularization, activation functions must be described

## SOURCE CODE #3

- Open source is available but must be **explainable**
- Comments are also required and explain process of forward and back propagation
- Explain with an example how an image example is transformed when forwarding
- Describe **how and why** you chose mini-batch, epoch, loss function, optimization function, and so on

# RESULTS AND PLOTS #4

- Experiment with **modifying hyperparameters** and measure accuracy
- Explain the results through a visible **plot**, such as a confusion matrix
- **Discuss why** such results came about



# POINT ALLOCATION

- **Code score – 40 points**

Quantitative evaluation

- **40** – Top 15 accuracy model
- **30** – Works well
- **20** – Works

- 10 points if report description is insufficient
- Code copy is not allowed among students

- **Report score – 60 points**

Qualitative evaluation

- **60** – Excellent
- **45** – Good
- **30** – Fair

- Additionally, the score may be deducted

# REFERENCES

- <https://becominghuman.ai/what-are-convolutional-neural-network-cnn-d065414546a3>
- <https://www.analyticsvidhya.com/blog/2022/01/convolutional-neural-network-an-overview/>
- [https://www.researchgate.net/publication/326916041\\_Deep\\_Learning-Based\\_Enhanced\\_Presentation\\_Attack\\_Detection\\_for\\_Iris\\_Recognition\\_by\\_Combining\\_Features\\_from\\_Local\\_and\\_Global\\_Regions\\_Based\\_on\\_NIR\\_Camera\\_Sensor/figures?lo=1](https://www.researchgate.net/publication/326916041_Deep_Learning-Based_Enhanced_Presentation_Attack_Detection_for_Iris_Recognition_by_Combining_Features_from_Local_and_Global_Regions_Based_on_NIR_Camera_Sensor/figures?lo=1)