

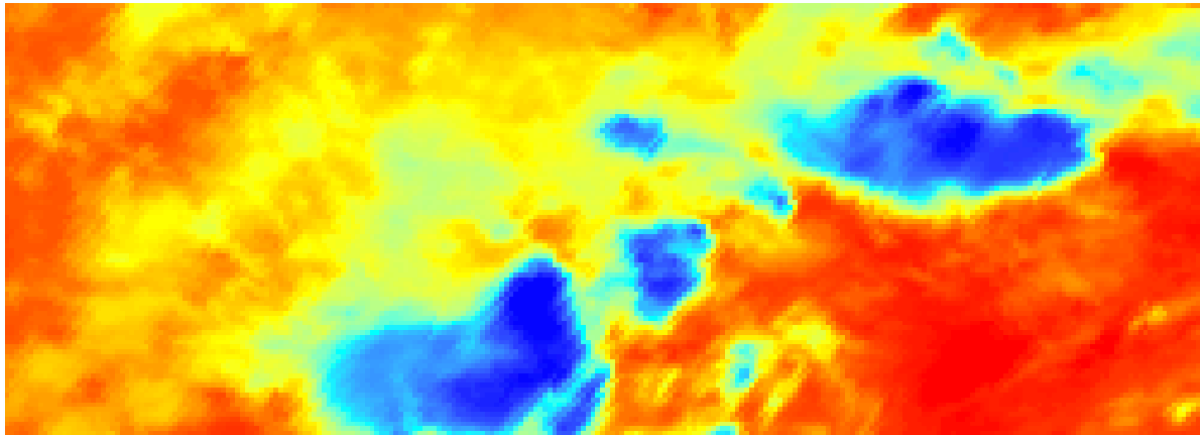
# 1. Mô tả dữ liệu

## 1.1. Các loại dữ liệu

### 1.1.1. Ảnh vệ tinh (Himawari)

Là các ảnh được chụp từ vệ tinh, gồm 14 band phổ.

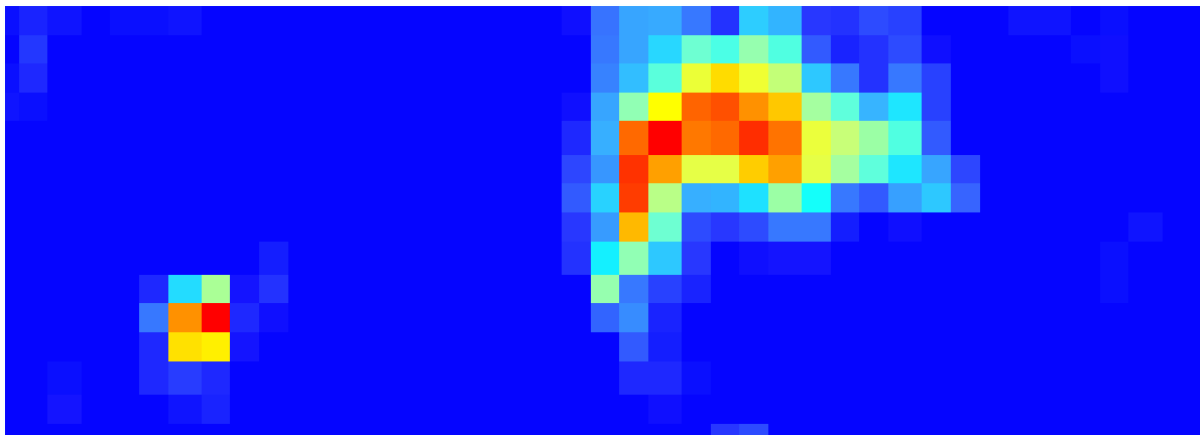
Đối với các ảnh kỹ thuật số, người ta có thể tách thành 3 lớp (Red, Green, Blue). Ảnh vệ tinh cũng tương tự, tuy nhiên số lượng lớp (gọi là band/phổ) sẽ nhiều hơn do có những lớp khác (bước sóng dài, ngắn mà mắt thường không nhìn thấy được, có thể tìm hiểu về ảnh vệ tinh để biết thêm chi tiết).



### 1.1.2. Ảnh khí tượng/phụ trợ (ERA5)

Là các ảnh được sinh ra bởi mô hình học máy, thể hiện các thông tin khí tượng như hướng gió, độ ẩm, v.v.

Tra cứu ý nghĩa dữ liệu: [Parameter Database \(ecmwf.int\)](https://www.ecmwf.int/en/forecasts/parameter-db)



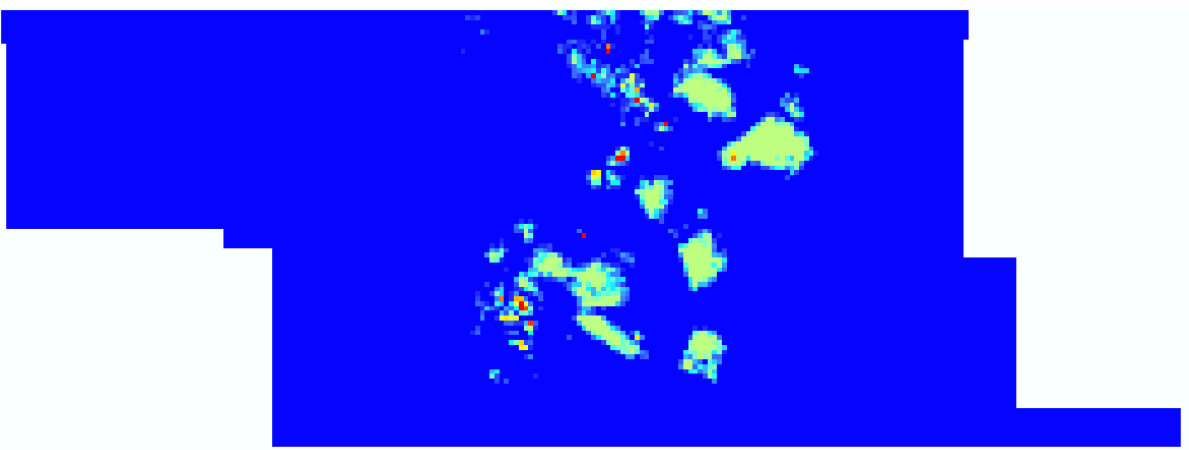
### 1.1.3. Mưa Radar

Là các ảnh dữ liệu mưa được tạo ra từ dữ liệu thu được bởi Radar.

Mỗi điểm ảnh sẽ đại diện cho lượng mưa tại vị trí đó.

Do phạm vi hoạt động của Radar có giới hạn nên ảnh Radar sẽ có những vùng trắng (không có giá trị, được đặt là -inf, nan, hoặc - 9999).

Dữ liệu Radar sẽ được dùng làm NHÃN (Target) cho các bài toán ước tính/dự báo mưa cho Radar (ảnh) và cũng được dùng làm ĐẦU VÀO (Input) cho các bài toán dự báo mưa.



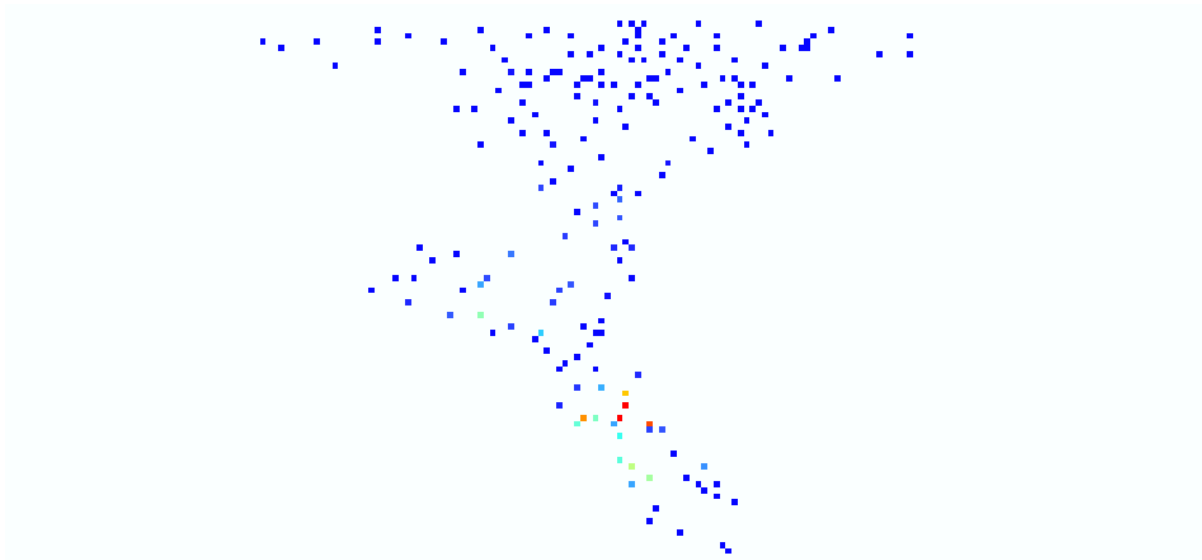
#### 1.1.4. Mưa trạm (AWS)

Dữ liệu trạm gốc là các bản ghi nhận lượng mưa tại những nơi đặt trạm (dữ liệu dạng bảng).

Dữ liệu trạm đã được xử lý và biến đổi thành dạng ảnh.

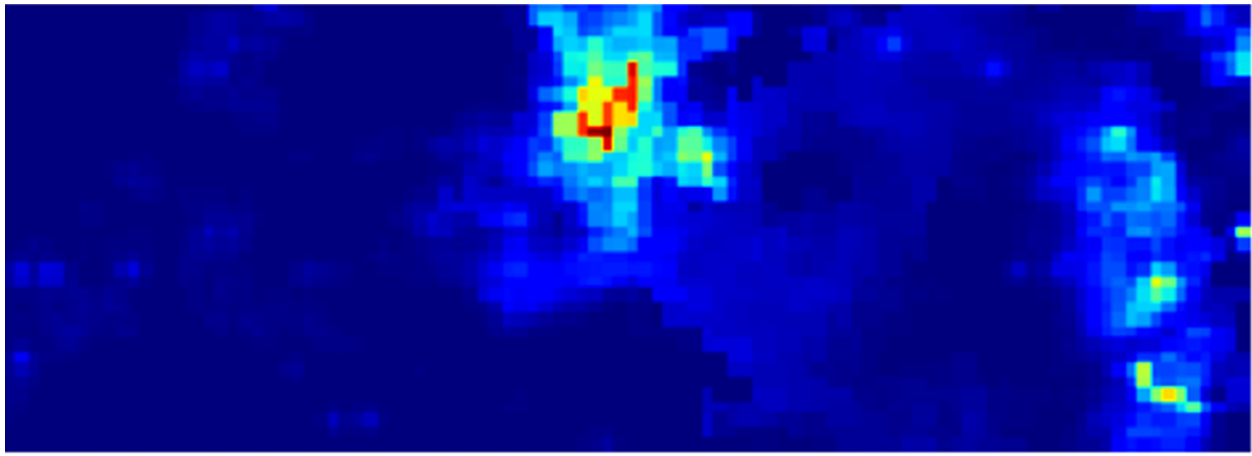
Những điểm ảnh có giá trị  $\geq 0$  là những vị trí có đặt trạm đo (AWS) và có bản ghi lượng mưa từ trạm đó.

Những điểm ảnh trắng (không có giá trị, được đặt là -inf, nan, hoặc - 9999) là những vị trí không có dữ liệu



#### 1.1.5. Mưa vệ tinh (IMERG)

Là một sản phẩm mưa vệ tinh, được tạo ra bởi các mô hình thống kê/học máy với đầu vào là các ảnh vệ tinh (tương tự ảnh vệ tinh Himawari) và cho ra đầu ra là ảnh thể hiện lượng mưa.



## 1.2. Các tính chất chung của các loại dữ liệu trên

- Định dạng (Format): GeoTIFF (.tif)

Có thể sử dụng các công cụ (tool) như Arcmap để visual dữ liệu.

Trong python, có thể sử dụng các thư viện như gdal, xarray (là các thư viện chuyên dùng để đọc, viết, xử lý các dữ liệu liên quan đến địa lý). Hoặc có thể sử dụng các thư viện opencv, numpy (là các thư viện phổ biến để làm việc với các dữ liệu dạng ảnh, số nói chung).

- Kích thước ảnh: 90 x 250

Mỗi file dữ liệu trên sẽ là một ảnh (một mảng 2 chiều, 2d-array, ...) gồm 90 hàng và 250 cột.

- Độ phân giải không gian: 0.04 độ ~ 4KM

Mỗi điểm ảnh (1 pixel) sẽ biểu diễn, đại diện cho thông tin của một khu vực địa lý có kích thước 0.04 độ x 0.04 độ (hoặc 4KM x 4KM)

- Độ phân giải thời gian: 1 giờ

Mỗi file dữ liệu trên biểu diễn thông tin ghi nhận được tại 1 thời điểm (kiểu dữ liệu tức thời ~ *instantaneous*, ví dụ như nhiệt độ tại 0h, 3h, ...) HOẶC dữ liệu tích lũy trong 1 giờ trước đó ~ *cumulative*, ví dụ như lượng mưa ghi nhận lúc 3h là mưa tích lũy từ 2h đến 3h).

Tóm lại: Các file dữ liệu sẽ cách đều nhau 1 giờ. Tuy nhiên có thể sẽ có những thời điểm bị thiếu dữ liệu.

- Khoảng thời gian: tháng 4, 10 năm 2019, 2020

Khoảng thời gian dữ liệu mà giảng viên cung cấp cho sinh viên thực hiện bài tập lớn.

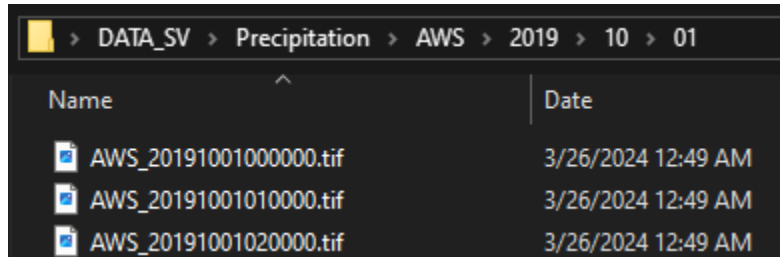
Cần cân nhắc sử dụng dữ liệu để phù hợp với bài toán, phù hợp với điều kiện tính toán (máy cá nhân, colab, ...)

### 1.3. Cấu trúc thư mục/tệp

Dữ liệu lưu theo thư mục con năm/tháng/ngày, dùng chung cho TẤT CẢ các loại dữ liệu kể trên

Tên file: (Biển)\_YYYYmmddHHMMSS.tif

Có thể sử dụng thư viện datetime của Python để xử lý thông tin thời gian từ chuỗi ký tự ở tên file



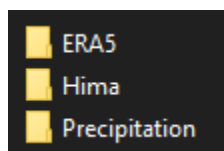
Name	Date
AWS_20191001000000.tif	3/26/2024 12:49 AM
AWS_20191001010000.tif	3/26/2024 12:49 AM
AWS_20191001020000.tif	3/26/2024 12:49 AM

Ba thư mục theo ba nhóm dữ liệu

Hima: Là dữ liệu Ảnh vệ tinh

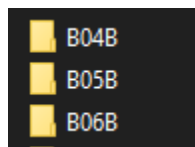
ERA5: Là dữ liệu khí tượng/phụ trợ

Precipitation: Dữ liệu mưa, gồm AWS và Radar



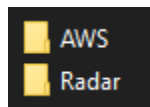
Trong thư mục Hima

Mỗi thư mục chứa một lớp (Band/phổ) của ảnh vệ tinh



Trong thư mục Precipitation

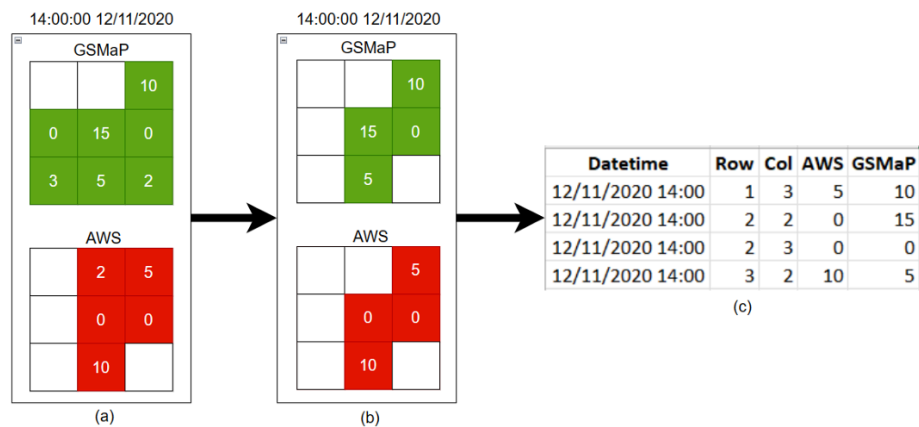
Dữ liệu mưa trạm (AWS) và dữ liệu mưa Radar lưu trong thư mục riêng



## 2. Một vài kỹ thuật xử lý dữ liệu

### 2.1. Xử lý dữ liệu cho các bài toán với trạm

Như đã minh họa ở trên, dữ liệu trạm ban đầu là bản ghi tại những vị trí có trạm, do đó cách cơ bản (dành cho các bài toán trong môn học này) khi làm với loại dữ liệu trạm là biến đổi và tổng hợp TẤT CẢ dữ liệu thành một bảng dữ liệu.



Để biến đổi và tổng hợp một tập hợp các dữ liệu ảnh, cách cơ bản là nhặt các giá trị tại những ô (vị trí) có giá trị (khác nan, -inf, - 9999, ...) tại tất cả các dữ liệu. Ví dụ ở hình minh họa, có 2 loại dữ liệu là AWS, GSMaP tại cùng thời điểm 14h 12/11/2020. Hình (a) là minh họa cho ảnh dữ liệu ban đầu, hình (b) là sau khi loại bỏ các ô CHỈ CÓ dữ liệu ở 1 trong 2 ảnh và giữ lại những ô có dữ liệu ở CẢ HAI ảnh, hình (c) là tổng hợp những ô dữ liệu ở bước trước thành một bảng, với cột datetime là thời gian của dữ liệu, Row, Col là vị trí hàng cột và AWS, GSMaP là các giá trị được trích ra.

*\*Note: Kỹ thuật vừa trình bày chỉ là một Ví DỤ cho một kỹ thuật CƠ BẢN bên cạnh rất nhiều kỹ thuật khác. Thực tế, dữ liệu đã được biến đổi sang ảnh, nên vẫn có thể sử dụng các kỹ thuật, mô hình vốn được dùng cho dữ liệu ảnh để áp dụng với loại dữ liệu này. Khuyến khích sinh viên tìm kiếm các bài báo nghiên cứu (e.g. trên google scholar) với từ khóa liên quan đến deep learning, station, gauge, ...*

## 2.2. Xử lý dữ liệu cho các bài toán với Radar

Dữ liệu chuẩn bị cho môn học đã được xử lý về dạng ảnh với cùng định dạng, kích thước, ... nên về cơ bản, không có kỹ thuật chung để xử lý dữ liệu ảnh, mà sẽ phụ thuộc vào bài toán, mô hình để biến đổi ảnh.

Ví dụ: Thông thường, người ta thường biến đổi/cắt ảnh đầu vào để được một ảnh có kích thước 32x32, 64x64, 128x128, ... trước khi đưa vào các mô hình học sâu mạng tích chập (CNN). Có thể sử dụng thư viện opencv (cv2) để biến đổi kích thước (reshape/resize) ảnh, hoặc cắt nhỏ ảnh ban đầu (kích thước 90x250) thành các kích thước mong muốn.

*\*Note: Các kỹ thuật xử lý ảnh rất đa dạng, cho nên khuyến khích sinh viên tìm kiếm các bài báo nghiên cứu (e.g. trên google scholar) để tìm ra phương pháp làm thích hợp*

Trong quá trình xử lý, biến đổi ảnh, cần lưu ý một vài điều:

- Kích thước dữ liệu: kích thước lý tưởng cho các mô hình học sâu thường là bội số của 2. Tuy nhiên điều này là KHÔNG BẮT BUỘC.
- Thiếu giá trị: như trong minh họa dữ liệu mưa Radar, dữ liệu ảnh hoàn toàn có thể có những vị trí, khu vực bị trống do các điều kiện thực tế không thể đo đạc, tính toán giá trị tại những nơi đó. Do đó cần lưu ý điền (fill) giá trị cho các vị trí này (có thể điền giá trị 0, giá trị mean/median, tùy vào tính chất dữ liệu có được qua các số liệu thống kê).

### 3. Xây dựng mô hình

#### 3.1. Thống kê và chuẩn bị dữ liệu

Trước khi xây dựng các mô hình học máy truyền thống/ học sâu, cần nhìn lại và chuẩn bị dữ liệu. Cụ thể:

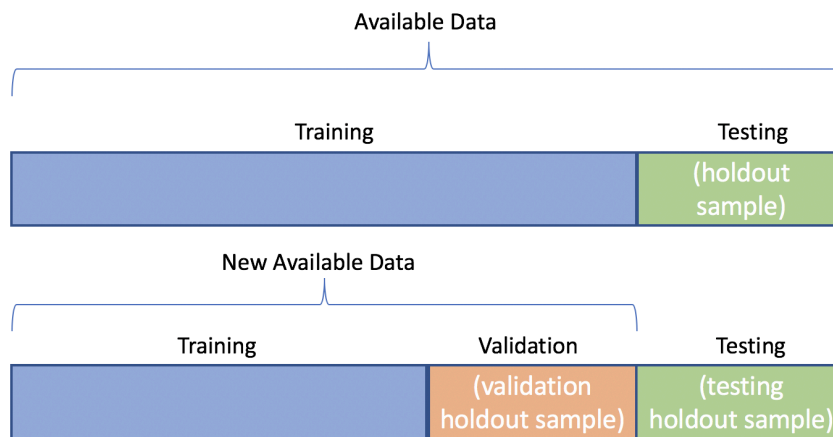
- Cần xác định rõ trong những dữ liệu đã xử lý, đâu là những thông tin (biến, trường dữ liệu, ...) sẽ được sử dụng, đâu là những thông tin không dùng đến, có thể bỏ đi. Ví dụ trong hình sau, các cột datetime, row, col là không cần thiết, có thể bỏ đi.

Datetime	Row	Col	AWS	GSMaP
12/11/2020 14:00	1	3	5	10
12/11/2020 14:00	2	2	0	15
12/11/2020 14:00	2	3	0	0
12/11/2020 14:00	3	2	10	5

- Cần xác định rõ đâu là loại dữ liệu đầu vào, đâu là dữ liệu nhãn. Ví dụ trong hình sau, cột label là nhãn cho bài toán phân loại, các cột B1-B7 là dữ liệu đầu vào.

X	Y	name	B1	B2	B3	B4	B5	B6	B7	label
579914.4	2297104	aquaculture	0.0264075	0.0345475	0.058335	0.046235	0.17114	0.077338	0.0398	1
591552.1	2312693	aquaculture	0.0318525	0.03463	0.05972375	0.042578	0.063863	0.031385	0.020564	1
587696.5	2312823	aquaculture	0.03610125	0.04728	0.08246625	0.056135	0.066461	0.02209	0.015133	1
585916.7	2343845	aquaculture	0.03815	0.043045	0.08413	0.067438	0.207605	0.150818	0.087045	1
599119.4	2295111	aquaculture	0.0337775	0.03969	0.07192	0.05421	0.3071	0.148205	0.070655	1
577655.1	2311412	aquaculture	0.0340525	0.04008875	0.05788125	0.048545	0.047541	0.033283	0.025624	1
555114.7	2315617	aquaculture	0.0425225	0.0567675	0.0788775	0.06279	0.086798	0.094525	0.088008	1

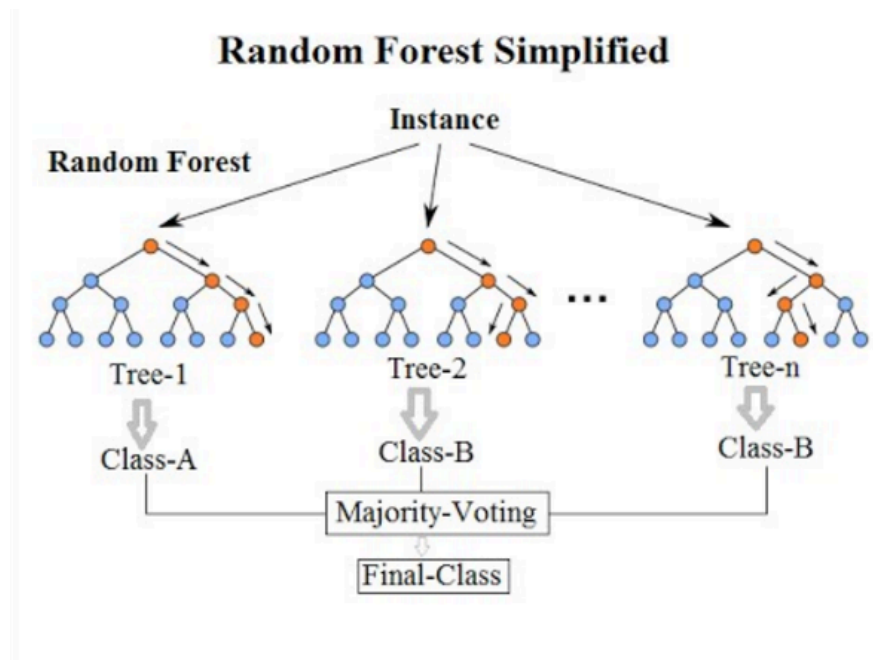
- Thực hiện việc phân chia train/ val/ test: Cơ bản, tất cả dữ liệu sẽ được chia làm 2 nhóm chính là train và test độc lập với nhau (không có mẫu nào ở cả 2 tập). Tiếp đến, để hỗ trợ cho quá trình xây dựng mô hình, tập train có thể chia nhỏ thành tập train và val. Vai trò mỗi tập dữ liệu là kiến thức cơ bản của phần học máy, sinh viên tự tìm hiểu thêm.



- Thực hiện một vài bước thống kê: Cần một vài thống kê cơ bản (e.g. số lượng dữ liệu, số lượng mẫu, số lượng mẫu mưa, ...). Các thống kê này có thể không trực tiếp ảnh hưởng đến việc xây dựng mô hình, tuy nhiên việc này rất cần thiết để có thể “HIỂU” dữ liệu, cũng như hỗ trợ việc xử lý và chuẩn bị dữ liệu.

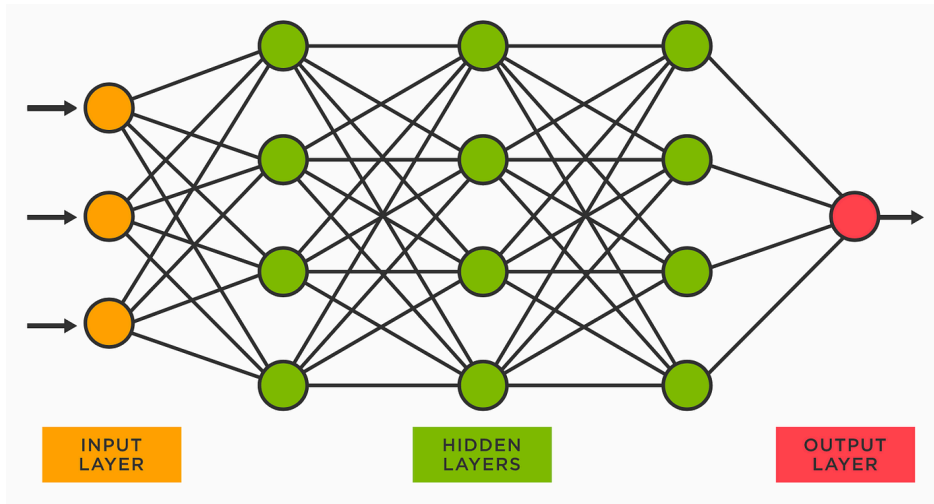
### 3.2. Các mô hình với dữ liệu dạng bảng

Các mô hình học máy cơ bản cho dữ liệu dạng bảng thường được dùng cho dữ liệu dạng bảng nói chung và dữ liệu cho bài toán mưa nói riêng là các mô hình dạng cây (Decision Tree, Random Forest). Sinh viên tự tìm hiểu về nguyên lý của mô hình và cách sử dụng (scikit-learn là thư viện phổ biến nhất cho các mô hình học máy truyền thống).



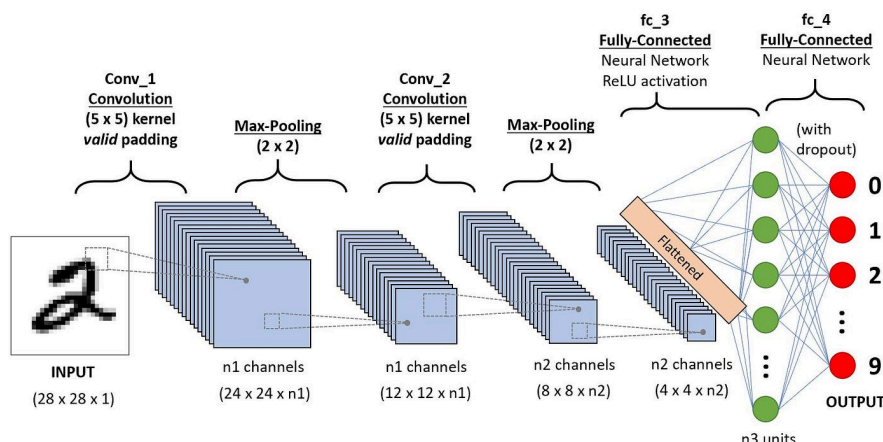
Một mô hình cơ bản khác dùng cho dữ liệu dạng bảng là Neural Network. Khuyến khích các bạn sinh viên sử dụng mô hình này vì đây là kiến thức cơ bản cho quá trình nghiên cứu về Deep Learning (e.g. CNN, RNN, ...) sau này. Hai thư viện phổ biến để thực hiện là Pytorch và Tensorflow. Khuyến khích sử dụng Pytorch đối với những sinh viên muốn đi sâu và phát triển trong lĩnh vực học máy/học sâu.





### 3.3. Các mô hình với dữ liệu dạng ảnh

Mô hình cơ bản cho bài toán với dữ liệu dạng ảnh là Convolution Neural Network. Thực tế, khái niệm CNN là nói chung đến các kiến trúc mạng sử dụng các lớp (layer) mạng tích chập. Việc thiết kế kiến trúc mạng (số lượng lớp, kích thước kernel tích chập, ...) vô cùng đa dạng, phong phú. Khuyến khích sinh viên tìm hiểu về các kiến trúc mạng phổ biến để cài đặt và thực nghiệm. Một số kiến trúc mạng gợi ý gồm LeNet, ResNet, Unet, ...



Bên cạnh các kiến trúc mạng tích chập, một cơ chế khác vốn phổ biến trong lĩnh vực ngôn ngữ tự nhiên (NLP) là cơ chế Attention cũng đang được ứng dụng rộng rãi trong lĩnh vực xử lý ảnh. Sinh viên có thể tìm hiểu thêm về cơ chế này, với mô hình điển hình áp dụng Attention cho phân loại ảnh là Vision Transformer.

### 3.4. Tối ưu tham số

#### Notes on Parameter Tuning

Parameter tuning is a dark art in machine learning, the optimal parameters of a model can depend on many scenarios. So it is impossible to create a comprehensive guide for doing so.

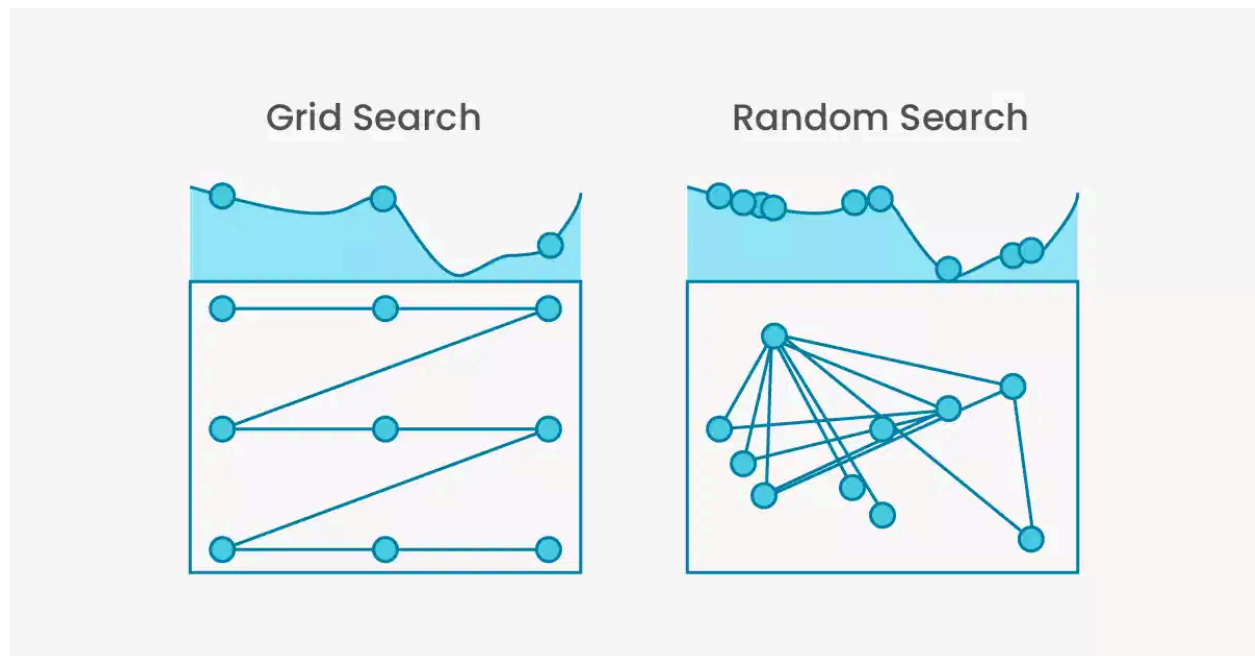
Bất kể mô hình nào cũng đều có những tham số riêng (e.g. các mô hình dạng cây có tham số “max depth”, “max leaf”, ...; các mô hình mạng thần kinh có tham số số lượng lớp, số lượng nodes, ...).



Cần tìm hiểu ý nghĩa của các tham số quan trọng trong mô hình, và thử thay đổi giá trị của các tham số này. Qua đó tìm được bộ tham số tối ưu, cho ra kết quả tốt nhất.

Không có phương pháp cụ thể nào để tìm ra bộ tham số tốt nhất. Sinh viên tự đưa ra chiến lược thực nghiệm và chọn ra kết quả được nhận định là tốt nhất.

Ví dụ với mô hình Random Forest, gồm rất nhiều tham số, trong đó có một vài tham số điển hình như `max_depth`, `n_estimators`, ... Thư viện `scikit-learn` cung cấp một phương pháp tìm kiếm tham số là `GridSearchCV` giúp người dùng tìm ra bộ tham số tối ưu nhất trong các trường hợp thực nghiệm.



### 3.5. Đánh giá mô hình

Sau khi xây dựng, huấn luyện xong mô hình, cần có những thang đo đánh giá để nhận định xem mô hình này tốt hay kém. Cách phổ biến nhất là sử dụng các chỉ số đánh giá, tính toán sai số giữa ĐẦU RA của mô hình với NHÃN trên tập KIỂM TRA (TEST). Lưu ý là việc đánh giá cuối cùng này chỉ làm trên tập Test, là lượng dữ liệu mà mô hình không được học trong quá trình huấn luyện.

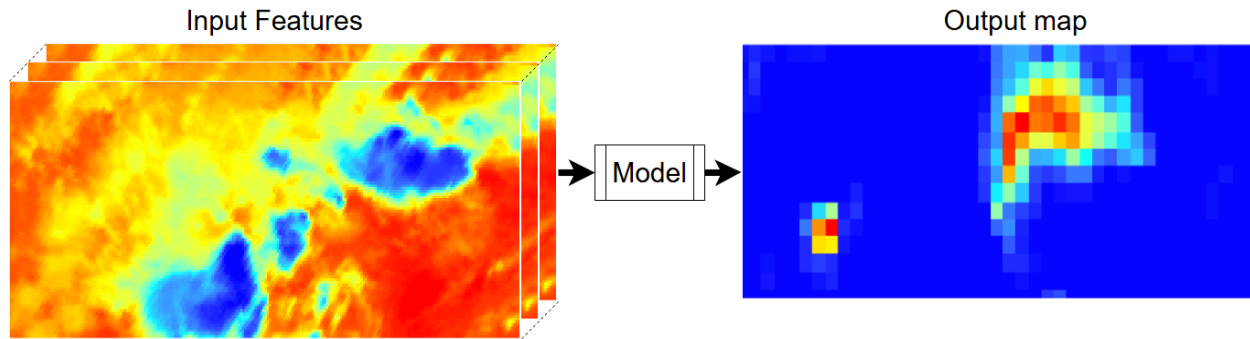
Các chỉ số đánh giá chia làm 2 loại:

- Chỉ số đánh giá phân lớp: Accuracy, Precision, Recall, F1, ...
- Chỉ số đánh giá hồi quy: Pearson R, R<sup>2</sup>, Mean Square Error, ...

Dựa vào kết quả, sinh viên đưa ra mô hình tốt nhất, chuẩn bị cho phần tạo bản đồ kết quả.

### 3.6. Tạo bản đồ kết quả

Dữ liệu đầu vào là dữ liệu dạng ảnh và phần Mô tả dữ liệu đã biểu diễn minh họa một vài hình ảnh của dữ liệu. Do đó, sinh viên cũng cần biểu diễn minh họa một vài hình ảnh kết quả đầu ra của mô hình tốt nhất.



Sinh viên lựa chọn một vài thời điểm để vẽ bản đồ minh họa.

**Đi kèm với bản đồ đầu ra của mô hình, cần có hình ảnh của dữ liệu mưa Radar cùng thời điểm để so sánh trực quan.**

## 4. Chi tiết từng bài toán

### 4.1. Phân loại mưa/Ước tính mưa

Dựa vào dữ liệu ảnh vệ tinh (Hima) để phân loại mưa (mưa/không mưa) hoặc ước tính lượng mưa tại cùng thời điểm với dữ liệu đầu vào.

**Chọn một vài thời điểm để tạo full ảnh kết quả mô hình**

Đầu vào: Ảnh vệ tinh Himawari.

Đầu ra: Dữ liệu mưa trạm / Mưa Radar cùng thời điểm.

Bài toán này phân ra làm 2 loại dựa theo 2 loại dữ liệu đầu ra.

#### **Đầu ra là dữ liệu mưa trạm**

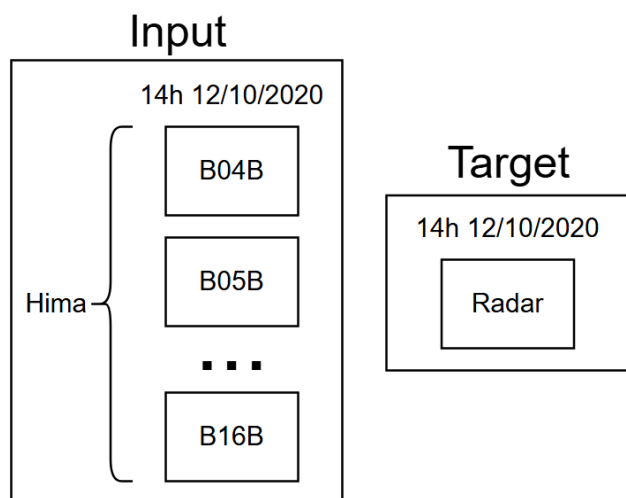
Đối với bài toán sử dụng đầu ra là dữ liệu mưa trạm, sử dụng các kỹ thuật xử lý và mô hình đối với dữ liệu dạng bảng.

Minh họa dữ liệu sau khi xử lý. Đối với bài toán phân loại mưa, nhãn AWS cần được biến đổi về nhãn mưa/không mưa.

Datetime	Row	Col	B04B	B05B	B16B	AWS
14h 12/10/2020	5	7	134	243	975	0.2
20h 09/06/2020	2	3	126	678	852	2.5
05h 11/02/2020	5	100	784	745	865	45

#### **Đầu ra là dữ liệu mưa Radar**

Đối với bài toán sử dụng đầu ra là dữ liệu mưa Radar, sử dụng các kỹ thuật xử lý và mô hình cho dữ liệu dạng ảnh.



## 4.2. Hiệu chỉnh sản phẩm mưa

Dựa vào dữ liệu mưa vệ tinh IMERG, kết hợp với các dữ liệu khí tượng/phụ trợ để đưa ra các bản đồ mưa với chất lượng tốt hơn.

### **Chọn một vài thời điểm để tạo full ảnh kết quả mô hình**

Đầu vào: Mưa vệ tinh IMERG, dữ liệu khí tượng/phụ trợ.

Đầu ra: Dữ liệu mưa trạm / Mưa Radar cùng thời điểm.

Bài toán này phân ra làm 2 loại dựa theo 2 loại dữ liệu đầu ra.

Về cơ bản, cách làm bài toán này tương tự với bài toán Phân loại mưa/Ước tính lượng mưa.

Khác biệt giữa 2 bài toán là kết quả bài toán hiệu chỉnh cần tốt hơn so với dữ liệu mưa vệ tinh. Có nghĩa là khi sử dụng các chỉ số đánh giá của kết quả mô hình trên nhãn cần TỐT HƠN so với các chỉ số đánh giá của IMERG trên nhãn

## 4.3. Dự báo mưa

Dựa vào dữ liệu mưa Radar/mưa trạm ở các giờ trước, kết hợp với dữ liệu ảnh vệ tinh Hima (có thể thêm các giờ trước) để dự báo lượng mưa ở các giờ tiếp theo.

Đầu vào: mưa Radar/mưa trạm ở 3/6/9/12 giờ trước, (có thể) thêm ảnh vệ tinh Hima.

Đầu ra: Dữ liệu mưa trạm / Mưa Radar ở 1/3/6/12 giờ tiếp theo (tùy lựa chọn).

Bài toán này phân ra làm 2 loại dựa theo 2 loại dữ liệu đầu ra.

### **Đầu ra là dữ liệu mưa trạm**

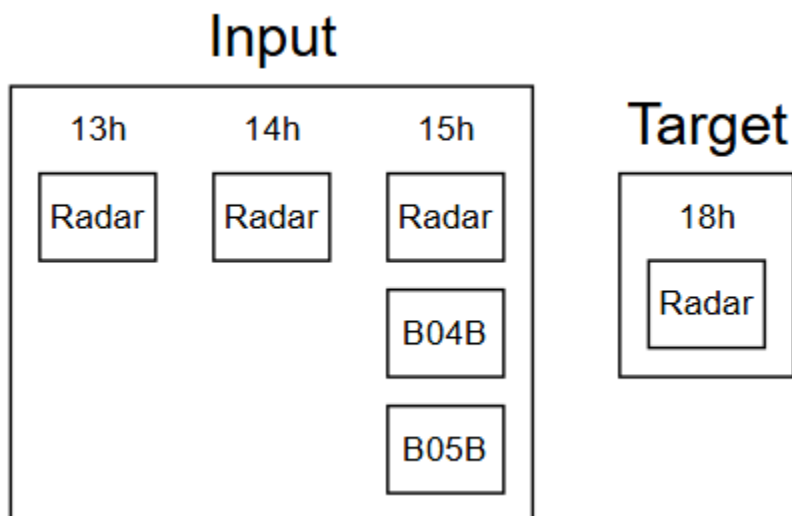
Đối với bài toán sử dụng đầu ra là dữ liệu mưa trạm, sử dụng các kỹ thuật xử lý và mô hình đối với dữ liệu dạng bảng.

Minh họa dữ liệu sau khi xử lý. Trong bảng, dữ liệu AWS+3h là nhãn, các cột AWS còn lại và các cột dữ liệu vệ tinh sẽ là đầu vào/input/feature

Datetime	Row	Col	B04B	AWS-2h	AWS-1h	AWS-0h	AWS+3h
14h 12/10/2020	5	7	353	0	0	2	10
20h 09/06/2020	2	3	345	2	1	2	0
05h 11/02/2020	5	100	541	3	0	10	1

## Đầu ra là dữ liệu mưa Radar

Đối với bài toán sử dụng đầu ra là dữ liệu mưa Radar, sử dụng các kỹ thuật xử lý và mô hình cho dữ liệu dạng ảnh.



**Lưu ý:** Các dữ liệu vệ tinh là OPTIONAL, có thể không có, có thể thêm thông tin của giờ cuối, có thể thêm thông tin của tất cả giờ Input. Cần thực nghiệm các cách đưa dữ liệu vào để tìm ra phương án tốt nhất.