

level up

o phai dinh, hoi cho ho them option

Machine learning 3 loai

supervised (đặc trung, biên mục)

unsupervised (không)

reinforcement

Xay dung 1 mo hinh

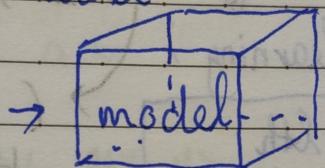
B1: chon đặc trung

B2: chia bo du thanh train, validation, test
80% 10% 10%

B3: chon model

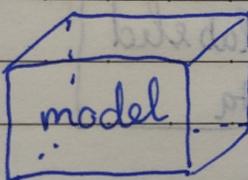
B4:

val



Độ chính xác hợp lý?
↓ Xem ↓ chia

B5: test



B5

B6.1

Độ chính xác.

Artificial Intelligence > Machine learning > Deep learning
nguồn, pt đt' mt có
thể lnr các v. của
con người

Tự học từ dataset

lquán đtn

x 1 cần code

các thuật toán

thể

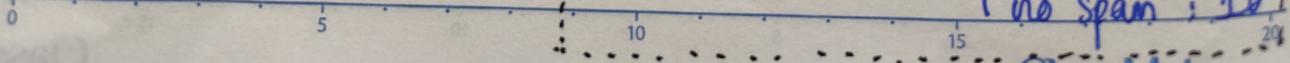
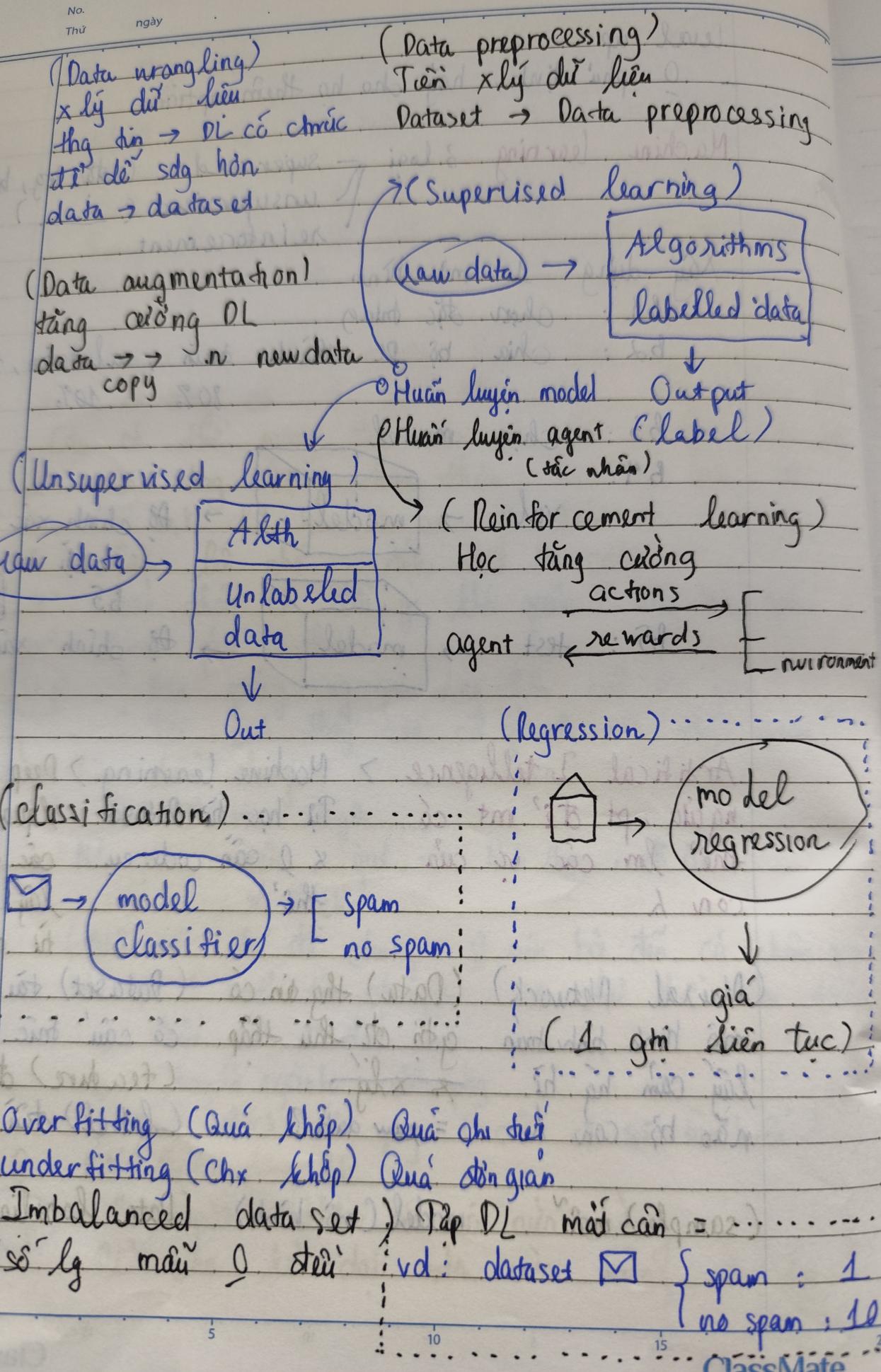
lấy cảm hứng

từ não bộ con

nh

(Neural Network) (Data) thg tin có (Dataset) tài liệu
mô hình tính toán g tin đt thu thập có cấu trúc
lấy cảm hrg hrg (feature) đặc trng
não bộ con người x lý (label) dấu ra

(sample) mẫu model (mô hình) (Data labeling / annotation)

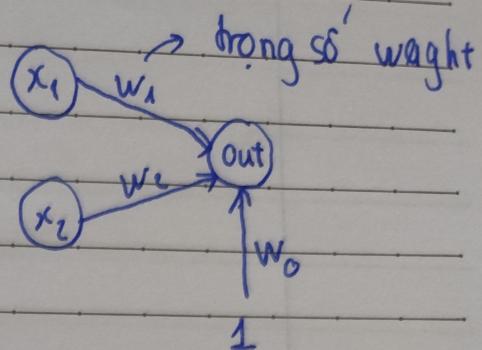


All layer trong neural network

Fully connected layer

mỗi neuron connect all neuron
của lớp trước và lớp sau

$$\text{out} = \sum_{i=1}^n w_i x_i + w_0 \text{ (bias)}$$



input : vector

tối TN tính toán, dễ bị overfitting

output : vector - new

Convolutional layer

thực hiện tích chập giữa img đầu vào & bộ lọc để
tìm đặc trưng

pooling layer
giảm kích thước

kết hợp

fully connected layer

ClassMate

$$\text{output} = (\text{số d'nh} + 1) \times \text{số nhan}$$

$$(5408 + 1) \times 128$$

Tích chập là quá kernel qua mt ảnh

Tích chập
padding, \rightarrow KT ảnh

Input: $n \times n$

Kernel: $K \times K$

Tích chập
(convolution): $(n - K + 1) \times (n - K + 1)$

Ảnh đang trống

padding

: thêm pixel 0 vào Input (p vòng)
 $(n - K + 1 + 2p) \times (n - K + 1 + 2p)$

bên hông strate: $(\frac{n - K + 2p}{s} + 1) \times (\frac{n - K + 2p}{s})$

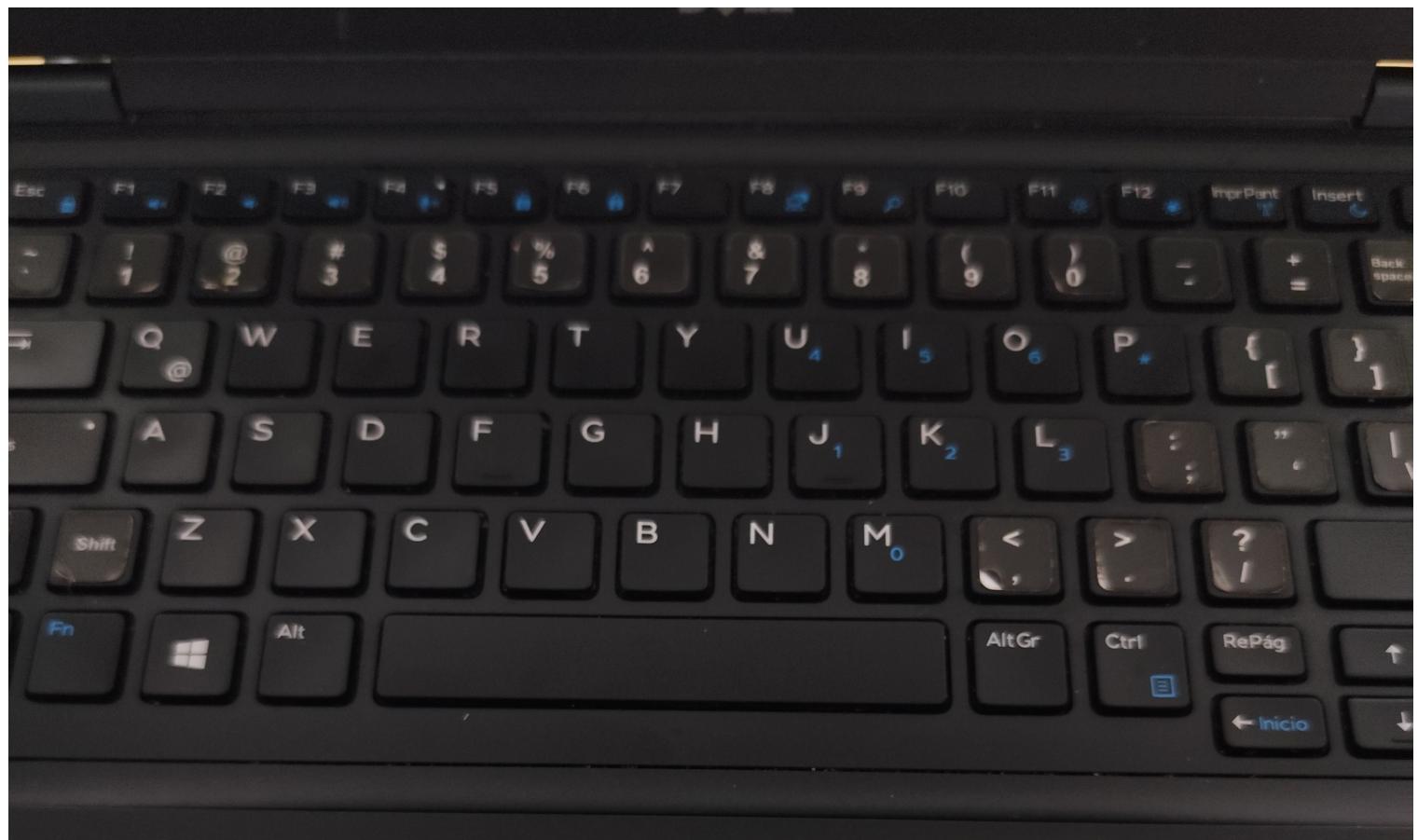
Ảnh màu RGB

Chiều sâu của Kernel = Số kênh ~~màu~~ Input

Output: ~~và~~ ảnh (tùy ~~đi~~ kernel có chiều sâu)

HAPLUS

Muốn Out ~~1~~ chiều \rightarrow n^2 Kernel



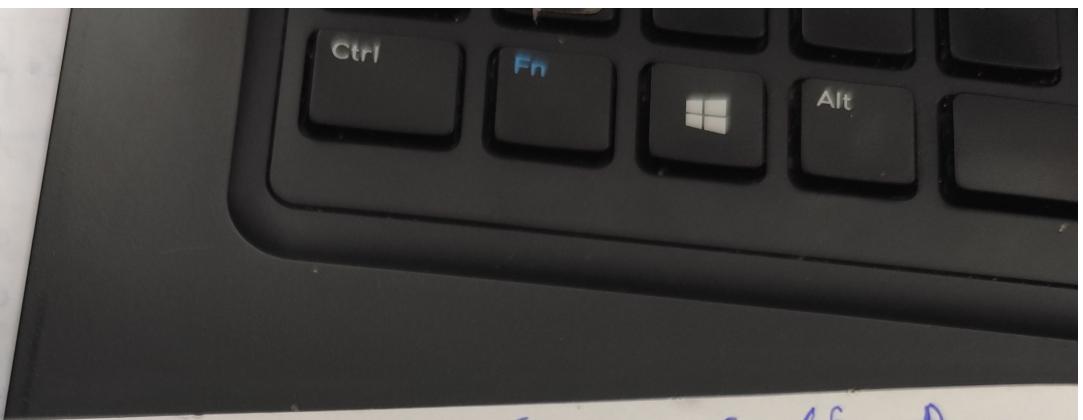
Input : $n \times n \times n_c$ \rightarrow số kênh

Kernel : $K \times K \times n_c \times n_k$ \rightarrow số kernel (b8/bc)

Out : $(n - K + 1) \times () \times n_k$

Pooling : ảnh xg 1 yg pixel thành 1 ghi duy nhất

- 1, Lớp tích chập 2D chiều \rightarrow kernel: 3×3 ($K=3$) thêm các lớp
Conv2D ('32, (3,3), activation = 'relu', input_shape=(28,28,1)) \rightarrow số lọc $n_K=32$ \rightarrow số lượng $n_C=1$
Hàm kích hoạt: ReLU max(0,2) Input \leftarrow $28 \times 28 \times 1$
 $n=28, n_C=1$
Output: $26 \times 26 \times 32$ (None, 26, 26, 32)
 $(n-K+1) \times () \times n_K$ (\downarrow kíp batch, lọc, ngang, số lọc)
 $(K \times K \times n_C + 1) \times n_K$
 \downarrow 1 gtn bias cho mỗi bộ lọc
- 2, Lớp Max - pooling (gửi đặc trưng) Kernel: 2×2 ($K=2$)
 Gram đặc trưng
Max Pooling 2D (pool_size = (2,2))
 Output: $13 \times 13 \times 32$ (None, 13, 13, 32)
 $\left(\frac{n}{k}\right) \times () \times n_C$
- 3, Flatten () Chuyển 2D \rightarrow 1D (None, 5408)
 nhieu \rightarrow vecto 1 chiều $13 \times 13 \times 32$
- 4, Lớp fully connect số neuron = 128
 Dense (128, activation = 'relu')
 Output: $(\text{số đ'nh} + 1) \times \text{số neuron}$
 $(5408 + 1) \times 128$



- chỉ có 50% số node () lớp Dense
5, Dropout (No. 5). dt, kích hoạt
- Ngừa HT overfitting \hookrightarrow Giảm CNN phụ thuộc quá nhiều
- Bỏ qua 50% số node 1 cách ngẫu nhiên

bản dịch mờ hình (create)

Huấn luyện
fit (x-train, y-train, epochs = 10, batch_size = 64),
validation_data = (x-test, y-test))

sử dụng mảng xly dc
khi cập nhật wnew