TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH

KHOA CHẤT LƯỢNG CAO



BỘ MÔN TRÍ TUỆ NHÂN TẠO

ĐỀ TÀI: SỬ DỤNG THUẬT TOÁN CNN ĐỂ NHẬN DIỆN CHỮ SỐ VIẾT TAY TRONG THƯ VIỆN MNIST

GIÁO VIÊN: NGUYỄN TRƯỜNG THỊNH

SINH VIÊN THỰC HIỆN: NGUYỄN MINH CHÂU

MSSV: 18146086

LỚP:18146CL1B – LỚP SÁNG THỨ 3 - NHÓM 06CLC

TPHCM, NGÀY 12/6/2021

**MỤC LỤC**

[Chương 1. TỔNG QUAN 2](#_Toc74406644)

[1.1 Đặt vấn đề 3](#_Toc74406645)

[1.2 Mục tiêu: 3](#_Toc74406646)

[1.3 Các bước thực hiện. 3](#_Toc74406647)

[**Chương 2. CƠ SỞ LÝ THUYẾT** 4](#_Toc74406648)

[2.1 Lý thuyết. 4](#_Toc74406649)

[2.2 Công thức. 4](#_Toc74406650)

[**Sơ qua về Logistic Regression:** Logistic Regression có activation function là hàm sigmoid : 4](#_Toc74406651)

[**Cost function:** 5](#_Toc74406652)

[**FORWARD PROPAGATION** 6](#_Toc74406653)

[**BACK PROPAGATION** Như đã nói ở trên, mục tiêu của back propagation là đi tính 7](#_Toc74406654)

[2.3 Mạng thần kinh tích chập (Convolutional neural network) 9](#_Toc74406655)

[2.4 Ưu nhược điểm. 9](#_Toc74406656)

[2.5 Ứng dụng. 9](#_Toc74406657)

[Chương 3. DỮ LIỆU 10](#_Toc74406658)

[3.1 Mô tả tập dữ liệu ( cách thu thập dữ liệu hoặc trình bày việc tham khảo nguồn dữ liệu từ đâu) 10](#_Toc74406659)

[3.2 Các tiền xử lý dữ liệu 10](#_Toc74406660)

[Chương 4. GIẢI THUẬT VÀ CHƯƠNG TRÌNH 12](#_Toc74406661)

[4.1 Chương trình (Giải thích chi tiết các dòng lệnh trong chương trình Machine learning) 13](#_Toc74406662)

[4.2 Cách đánh giá chất lượng của huấn luyện 14](#_Toc74406663)

[Chương 5. KẾT QUẢ PHÂN TÍCH 14](#_Toc74406664)

[5.1 Kết quả đạt được 14](#_Toc74406665)

[5.2 Phân tích kết quả và các cải tiến đã thực hiện để cải thiện 14](#_Toc74406666)

[Chương 6. KẾT LUẬN 15](#_Toc74406667)

[6.1 Kết quả đạt được ( tóm tắt kết quả thu được sau khi huấn luyện và kiến thức thu được sau khi làm project) 15](#_Toc74406668)

# Chương 1. TỔNG QUAN

## 1.1 Đặt vấn đề

Các vấn đề về phân loại ảnh được ứng dụng rất nhiều trong cuộc sống hiện nay ví dụ như :

+ Chuẩn đoán ảnh X-ray của bệnh nhân để xem có bị ưng thư hay không

+ Phân loại, nhận diện được các chữ, số viết tay => tự động đọc được biển số xe, văn bản.

+ Phân loại được các biển báo giao thông => hỗ trợ cho ô tô tự lái

+ Và còn rất nhiều ứng dụng khác nữa.

Ở Project này em sử dụng thuật ANN(artificial neural network) để nhận diện chữ số viết tay trong tập dữ liệu MNIST

## 1.2 Mục tiêu:

Mục tiêu của ANN: Mạng nơ-ron nhân tạo là một chuỗi các [thuật toán](https://vietnambiz.vn/giao-dich-thuat-toan-algorithmic-trading-la-gi-nhung-dac-diem-can-luu-y-20191205025413985.htm" \o "thuật toán )được đưa ra để nỗ lực tìm kiếm các [mối quan hệ](https://vietnambiz.vn/tuong-quan-dong-bien-positive-correlation-la-gi-nhung-dac-diem-can-luu-y-20191227113029107.htm" \o "mối quan hệ) cơ bản trong một tập hợp dữ liệu, thông qua quá trình bắt chước cách thức hoạt động của bộ não con người.

## 1.3 Các bước thực hiện.

+ Giới thiêu cơ sở lý thuyết thuật toán

+ Mô tả dữ liệu

+ Giải thuật và chương trình

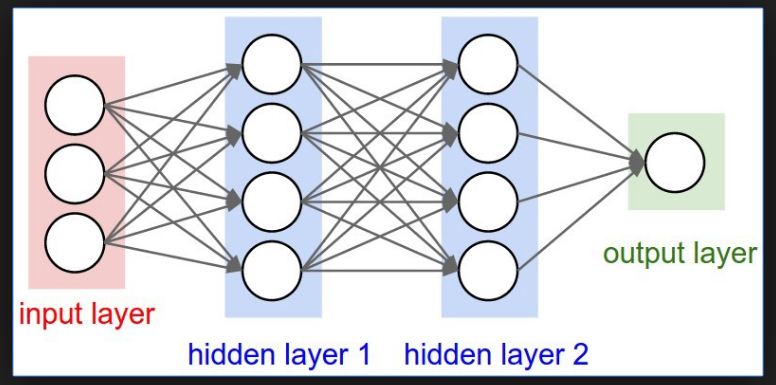
+ Kết quả phân tích và kết luận

+ Kết luận

## **Chương 2. CƠ SỞ LÝ THUYẾT**

## 2.1 Lý thuyết.

Artificial Neural Network (ANN) gồm 3 thành phần chính: Input layer và output layer chỉ gồm 1 layer , hidden layer có thể có 1 hay nhiều layer tùy vào bài toán cụ thể. ANN hoạt động theo hướng mô tả lại cách hoạt động của hệ thần kinh với các neuron được kết nối với nhau  
Trong ANN, trừ input layer thì tất cả các node thuộc các layer khác đều full-connected với các node thuộc layer trước nó. Mỗi node thuộc hidden layer nhận vào ma trận đầu vào từ layer trước và kết hợp với trọng số để ra được kết quả.



## 2.2 Công thức.

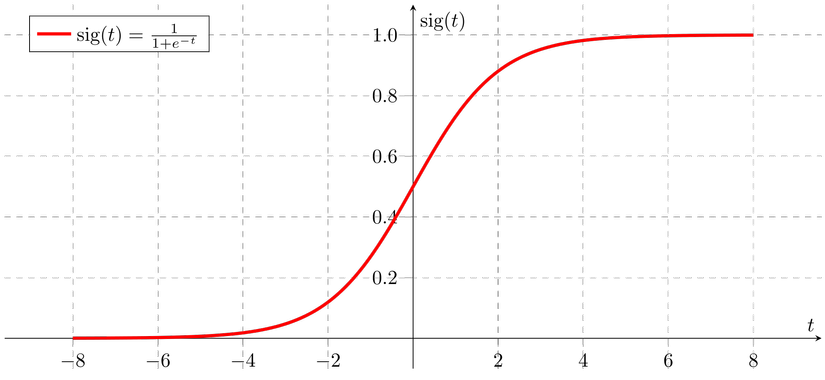
### **Sơ qua về Logistic Regression:** Logistic Regression có activation function là hàm sigmoid :

https://images.viblo.asia/fa2bd0ea-42e5-448b-a4ab-1bdb5d0cb9de.PNG

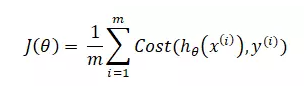
Hàm hypothesys :

https://images.viblo.asia/eafd4e1e-de8f-4085-bedc-0e1ca3fd6c17.PNG

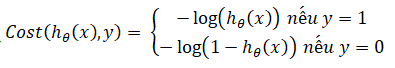
Đồ thị có dạng :



### **Cost function:**

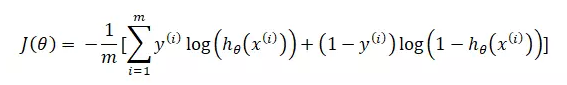


Với :

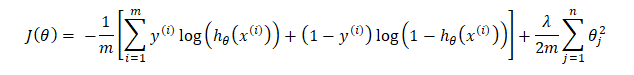




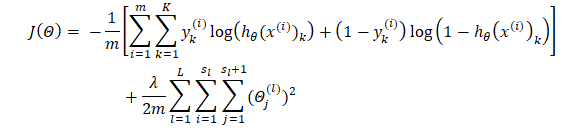
Vậy ta có cost function :



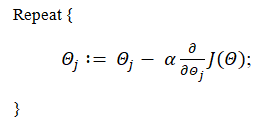
Kết hợp với Regurlarization:



Vậy **với ANN** với mỗi node thuộc layer khác input layer đều là một Logistic Regression ta sẽ có :



Công việc của chúng ta hiện tại là tìm ra được Θ sao cho  *J*(Θ) min.  
Để tìm cực tiểu của *J*(Θ) ta áp dụng thuật toán Gradient Descent.

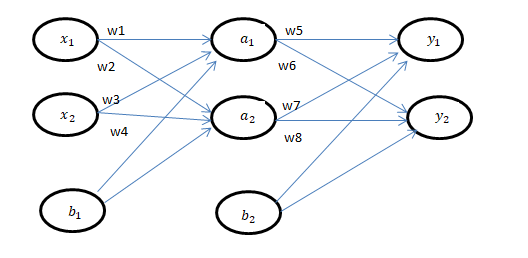


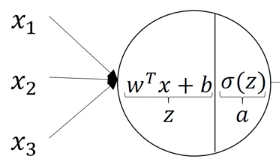
Với α là learning rate.

### **FORWARD PROPAGATION**

Ta có mạng neural như sau :

Phần này em thay thế ký hiệu trọng số nên



**Chú thích :**  
**,** là các features của input.  
**,**​ là các output.  
**,**​​ là các bias.  
**,**​,…, là các trọng số.  
Như cái tên của forward propagation , ta sẽ tiến hành tính toán **,**​*y*1​,*y*2​ từ trái qua phải.  
Mô hình tổng quát  


Với:  
z = = + b

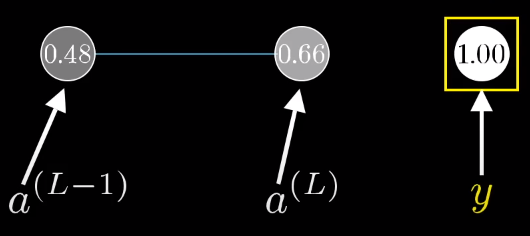
a =

Forward propagation là một công đoạn tính toán giá trị tại từng node để phục vụ việc tính toán trong Back propagation.

### **BACK PROPAGATION** Như đã nói ở trên, mục tiêu của back propagation là đi tính

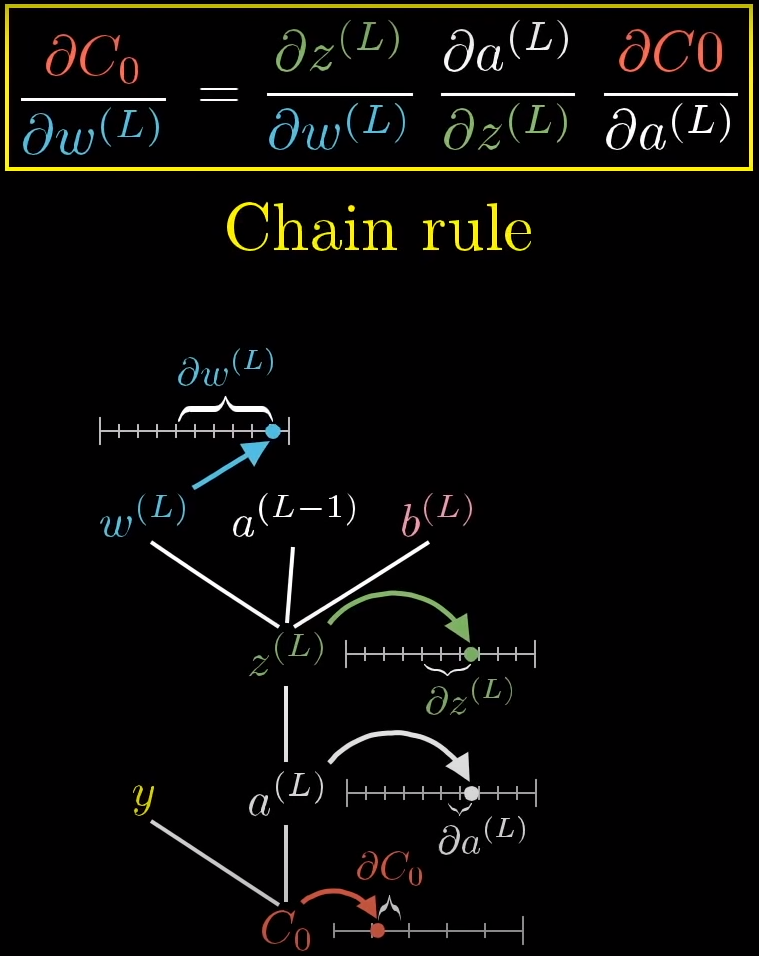
Ảnh hưởng của sự thay đổi các thành phần đối với hàm cost function:

Ở đây em sử dụng 1 mạng có 2 nút để diễn dãi (tham khảo từ video Backpropagation calculus | Chapter 4, Deep learning , của tác giả 3Blue1Brown)



Hàm mất mát của mạng này là ,(L là lớp cuối cùng)

=



Các giá trị , , đều có thể tính được thông qua kết quả thu được từ forward propagation. Vậy ta có thể tính được

Tổng quát hóa cho mạng gồm nhiều neuron ở các layer thì ta có thể tính được back propagation cho nó

## 2.3 Mạng thần kinh tích chập (Convolutional neural network)

Các convolutional layer có các parameter(kernel) đã được học để tự điều chỉnh lấy ra những thông tin chính xác nhất mà không cần chọn các feature.



## 2.4 Ưu nhược điểm.

Một ưu điểm vượt trội của mô hình mạng nơron nhân tạo là khả năng tự học và điều chỉnh các trọng số để kết quả tính toán phù hợp với thực tế mà không phụ thuộc vào ý kiến chủ quan.

Nhược điểm: ANN chỉ phát huy tốt khi sử dụng một tập dữ liệu lớn. Với những tập dự liệu nhỏ ANN có kết quả có thể kém hơn các thuật toán khác

## 2.5 Ứng dụng.

Mạng nơ-ron nhân tạođược sử dụng rộng rãi ở nhiều lĩnh vực, có thể ứng dụng cho [tài chính](https://vietnambiz.vn/he-thong-tai-chinh-financial-system-la-gi-dac-trung-va-y-nghia-2019081016163084.htm" \o "tài chính), lập kế hoạch doanh nghiệp, giao dịch, [phân tích kinh doanh](https://vietnambiz.vn/tri-tue-doanh-nghiep-business-intelligence-bi-la-gi-loi-ich-cua-bi-20191115104115757.htm" \o "phân tích kinh doanh) và bảo trì sản phẩm.

# Chương 3. DỮ LIỆU

## 3.1 Mô tả tập dữ liệu ( cách thu thập dữ liệu hoặc trình bày việc tham khảo nguồn dữ liệu từ đâu)

Em dùng tập dữ liệu về chữ số viết tay MNIST, chứa khoảng 70,000 ảnh đen trắng phân thành 10 loại. Mỗi một ảnh là một số từ 0-9 với độ phân giải thấp (28 by 28 pixel), như hình minh hoạ bên dưới:

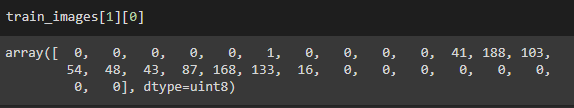


Với tập dữ liệu này, 60.000 ảnh sẽ được dùng để huấn luyện và 10.000 ảnh sẽ đường dùng để đánh giá khả năng phân loại nhận diện ảnh của mạng neuron.

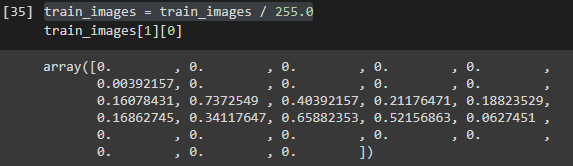
## 3.2 Các tiền xử lý dữ liệu

Ban đầu ảnh có kích cỡ 28\*28 và các phần từ này là 1 pixel có độ lớn từ 0-255

Em thử lấy 1 hàng trong 28 của tấm ảnh (dòng code này em sử dụng ảnh có shape(28,28) thay vì shape(28,28,1) để dễ hình dung).



Sau đó em chuyển các pixel này có giá trị từ 0-1.



# Chương 4. GIẢI THUẬT VÀ CHƯƠNG TRÌNH

Các bước xây dựng NN đơn giản bằng Keras:

+ Load dữ liệu

+ Xây dựng Model

+ Compile Model

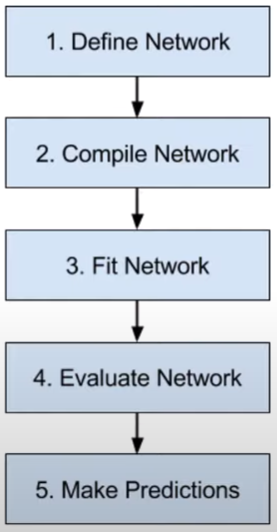
+ Train Model

+ Đánh giá Model

+ Dữ đoán dữ liệu mới

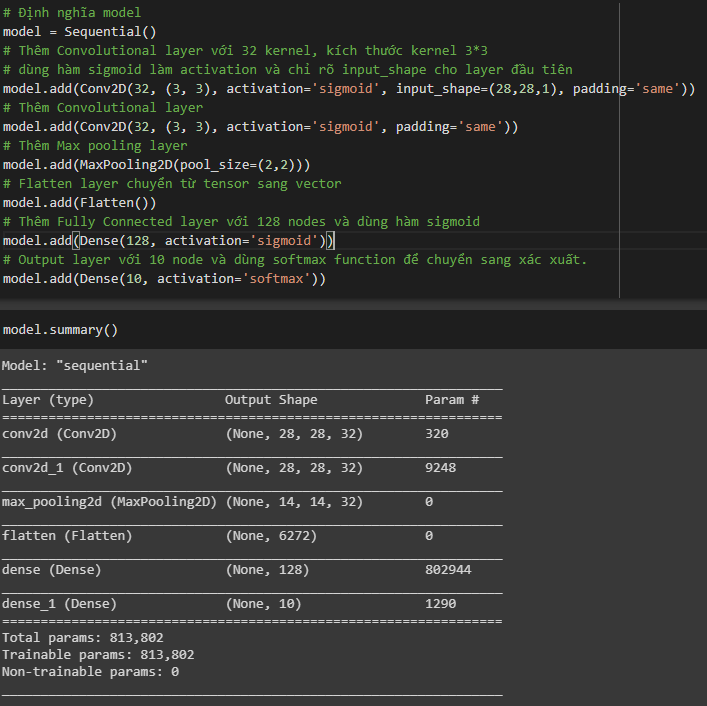
+ Lưu Model

+ Load Model

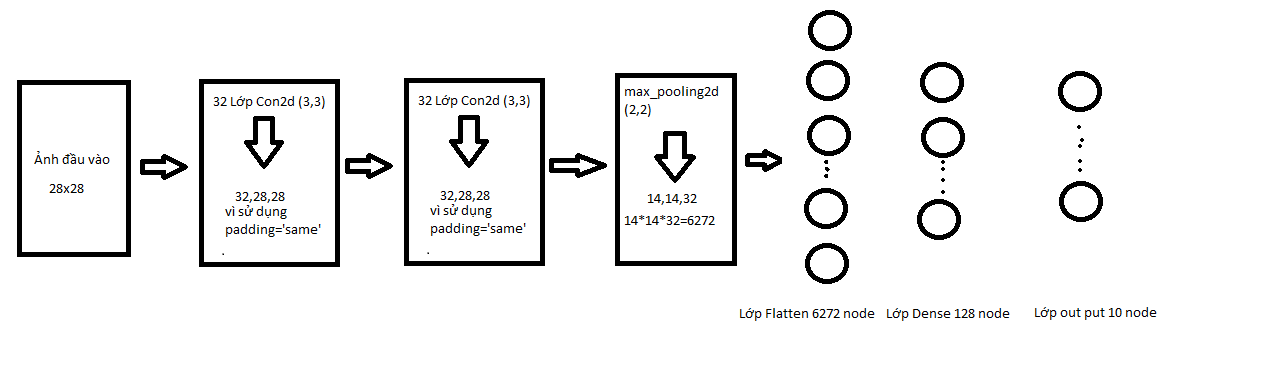


## 4.1 Chương trình (Giải thích chi tiết các dòng lệnh trong chương trình Machine learning)

+ Define network:



Em có vẽ hình để minh họa:



## 4.2 Cách đánh giá chất lượng của huấn luyện

+ Compile Network:

model.compile(loss='CategoricalCrossentropy',optimizer='adam',metrics=['accuracy'])

* Loss: Dùng để tính hàm mất mác đề cập ở chương 3. Ở đây em sử dụng CategoricalCrossentropy do đó ở phần nhãn của dữ liệu phải chuyển về onehot encoding

# One hot encoding label (Y)

train\_labels = np\_utils.to\_categorical(train\_labels, 10)

test\_labels = np\_utils.to\_categorical(test\_labels, 10)

* Optimizer: thuật toán dung để tối ưu hàm Loss
* Metric: Thước đo để đo độ chính xác sau mỗi lần train

# Chương 5. KẾT QUẢ PHÂN TÍCH

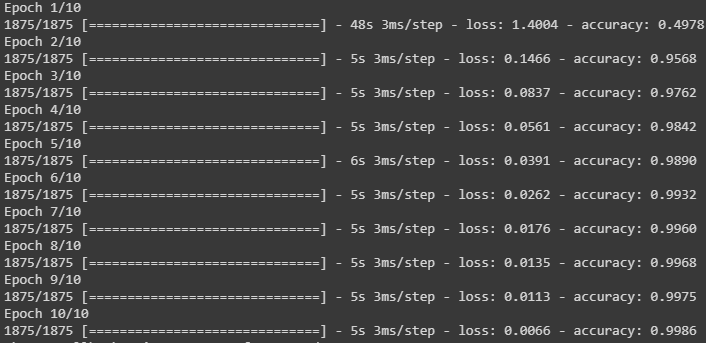
## 5.1 Kết quả đạt được

Nếu ta xét trên tập kiểm thử (tập test)

Hàm mất mác sau 10 epoch: 0.0521

Đọ chính xác sau 10 epoch: 0.9846

## 5.2 Phân tích kết quả và các cải tiến đã thực hiện để cải thiện



Ta thấy sau 10 epoch thì quá trình của hàm Loss đang giảm và độ chính xác accuracy điều đó nghĩa là nếu ta tiếp tục train thì có thể chính xác hơn nữa.

# Chương 6. KẾT LUẬN

## 6.1 Kết quả đạt được ( tóm tắt kết quả thu được sau khi huấn luyện và kiến thức thu được sau khi làm project)

Sau khi hoàn thành project, em đã học hỏi thêm được nhiều kiến thức về Machine Learning cũng như Deep Learning. Hiểu sâu hơn về các thuật toán Gradient Decent,

ANN, CNN,…

*Tài liệu tham khảo (bao gồm các sách tham khảo, các website, các link code tham khảo):*

+ List video về DeepLearning của 3blue1brown : <https://www.youtube.com/watch?v=aircAruvnKk&list=PL_h2yd2CGtBHEKwEH5iqTZH85wLS-eUzv>

+ Sườn bài hướng dẫn em tham khảo của Tensor Flow:

<https://www.tensorflow.org/tutorials/keras/classification#compile_the_model>

+ List video hướng dẫn cơ bản về Machine Learning:

<https://www.youtube.com/playlist?list=PLQY2H8rRoyvxNqk9EV5VP5fS0cWEXW5QQ>

+ Phần toán học ANN em tham khảo của đường link này:

<https://viblo.asia/p/tong-quan-ve-artificial-neural-network-1VgZvwYrlAw>