**Fashion Mnist 분류**

|  |  |
| --- | --- |
| 제 출 일 | 2018.11.29 |
| 과 목 | 인공지능 |
| 작 성 자 | 홍 진 석 : 201202128  임 진 수 : 201203391 |

내용

[2. Background 3](#_Toc530838570)

[2.1 Background 3](#_Toc530838573)

[2.2 Why? 3](#_Toc530838574)

[3. Introduction to Data 3](#_Toc530838575)

[3.1 Image 3](#_Toc530838579)

[3.2 Introduction 4](#_Toc530838580)

[4. Model 4](#_Toc530838581)

[4.1 Neural Network 4](#_Toc530838583)

[4.2 Convolution Neural Network 5](#_Toc530838584)

[5. How to solve 5](#_Toc530838585)

[5.1 Neural Network Training & Testing Result 5](#_Toc530838587)

[5.2 Convolution Neural Network Training & Testing Result 6](#_Toc530838588)

[6. Discussion 7](#_Toc530838589)

# Background



## Background

* 인공지능(AI)쪽에서 가장 많이 사용하고 있는 예제가 MNIST입니다. MNIST는 간단한 컴퓨터 비전 데이터셋입니다. MNIST 데이터베이스는 손으로 쓴 숫자들로 이루어진 대형 데이터베이스로 이 데이터를 이용해 기계 학습 분야의 트레이닝 및 테스트에 많이 사용되고 있습니다.

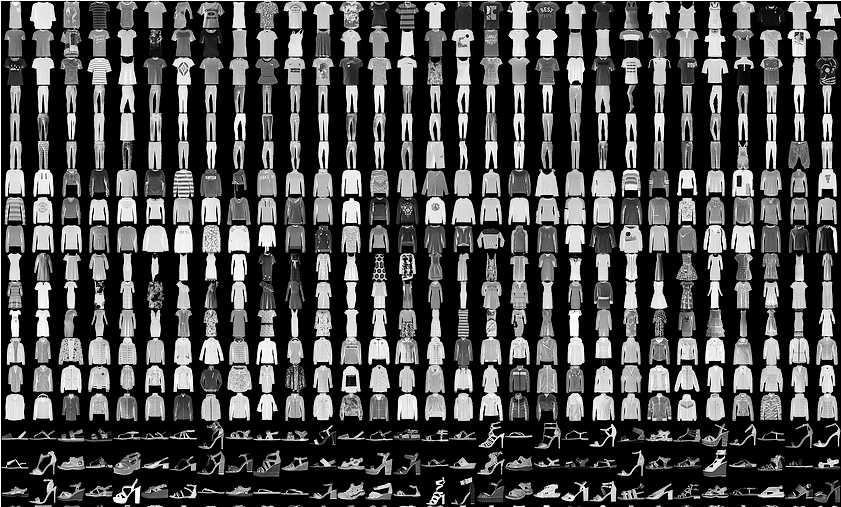
## Why?

* 저희는 손으로 쓴 숫자들로 이루어진 MNIST가 아닌 다른 MNIST 데이터베이스가 존재하지 않을까 생각을 하였고, 그 데이터의 분류도 기존의 MNIST처럼 잘 분류가 되는지 확인을 해보고 싶어서 이 프로젝트를 진행하게 되었습니다.

# Introduction to Data



## Image



## Introduction

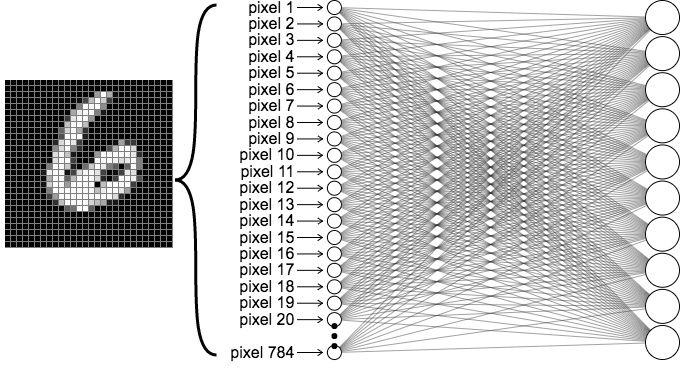
* 28x28 흑백이미지
* Training Data : 60,000개
* Test Data : 10,000개
* Output : 10개의 클래스
  + 0 : T-shirt/top / 1 : Trouser / 2 : Pullover / 3 : Dress / 4 : Coat / 5 : Sandal / 6 : Shirt / 7 : Sneaker / 8 : Bag / 9 : Ankle boot

# Model



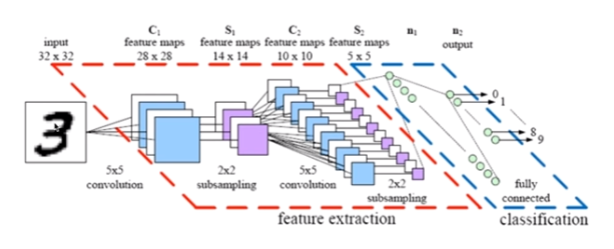
## Neural Network

* Input Layer : 784 Neuron
* Output Layer : 10 Neuron
* Softmax 함수 사용
* Model Example



## Convolution Neural Network

* Input >> Conv Layer(3x3) >> Max Pooling(2x2) >> Conv Layer(3x3) >> Max Pooling(2x2) >> Fully Connected Layer >> Softmax
* Model Example

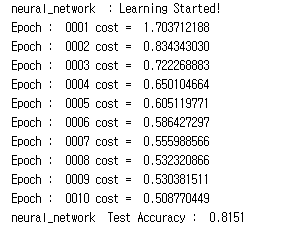


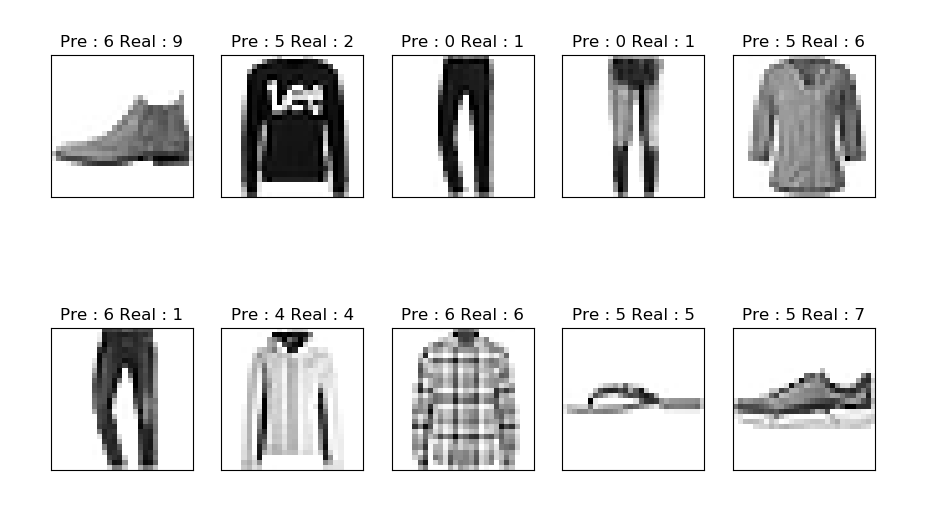
# How to solve

* Total Batch : 100
* Total Epoch : 10

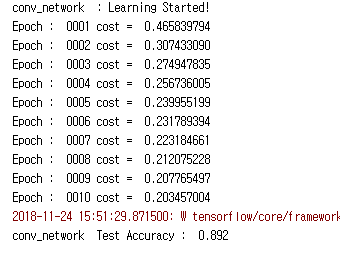


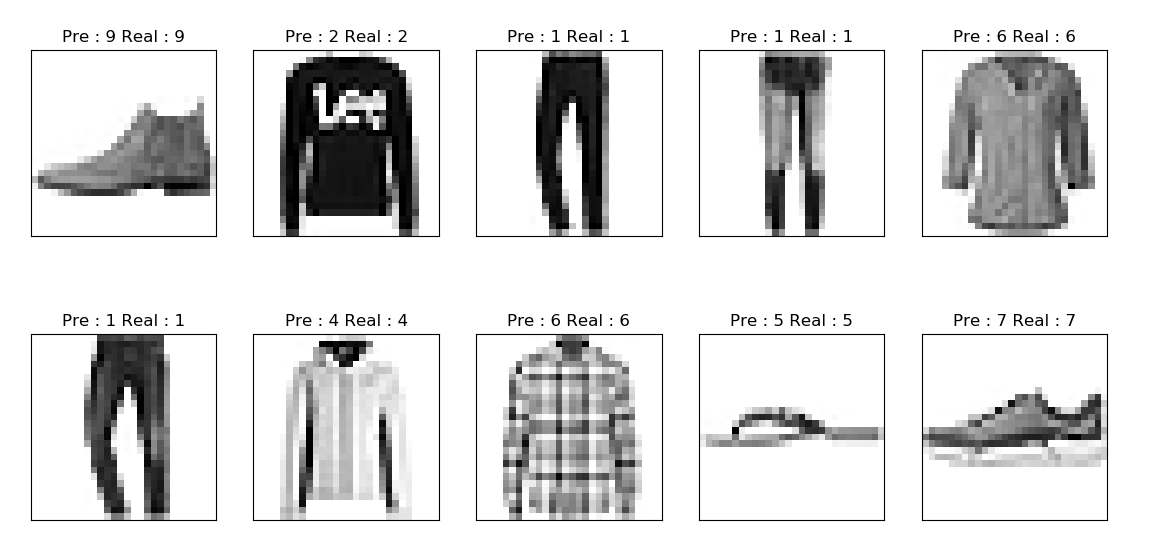
## Neural Network Training & Testing Result





## Convolution Neural Network Training & Testing Result





# Discussion

* 손으로 쓴 숫자데이터의 테스트 정확도가 Neural Network에서는 92%, Convolution Neural Network에서는 99%까지 나오는데, Fashion MNIST 같은 경우는 Neural Network : 81%, Convolution Neural Network : 89%의 성능을 보이며 기존 MNIST보다는 성능이 낮게 나오는 것을 확인할 수 있다.
* 다른 Network 모델을 사용하거나 Hyper Parameter 변경을 통해 성능을 좀더 끌어올릴 수 있지 않을까 생각한다.
* Neural Network모델이 아닌 Clustering을 통해 분류를 해보는 방식도 고려해 볼 수 있다.