



## 인공지능 TensorFlow (Ch 2)

---

데이터사이언스 & AI 정화민 교수



# TensorFlow 설치

- 일반적 설치 : Anaconda 파이썬 환경 사용 추천
  - CPU만 사용할 때는 상대적 설치가 쉽다.
  - GPU 사용 : CUDA 설치 필요.
    - \* CUDA ("Compute Unified Device Architecture", 쿠다)는 [그래픽 처리 장치](#)(GPU)에서 수행하는 (병렬 처리) 알고리즘을 C 프로그래밍 언어를 비롯한 산업 표준 언어를 사용하여 작성 할 수 있도록 하는 GPGPU(General-Purpose computing on Graphics Processing Units, GPU 상의 범용 계산)기술 필요, CUDA는 [엔비디아](#)가 개발해오고 있으며 지포스 8 시리즈급 이상에서 동작.
- Anaconda : 아나콘다는 Anaconda(이전: Continuum Analytics)라는 곳에서 만든 파이썬 배포판으로, 수백 개의 파이썬 패키지를 포함하고 있습니다. 또한 회사 내에서도 상업용으로 무료로 사용할 수 있다는 장점(Scikit-learn, NumPy, SciPy, Pandas 등 머신러닝 또는 데이터 마이닝 라이브러리 내장)

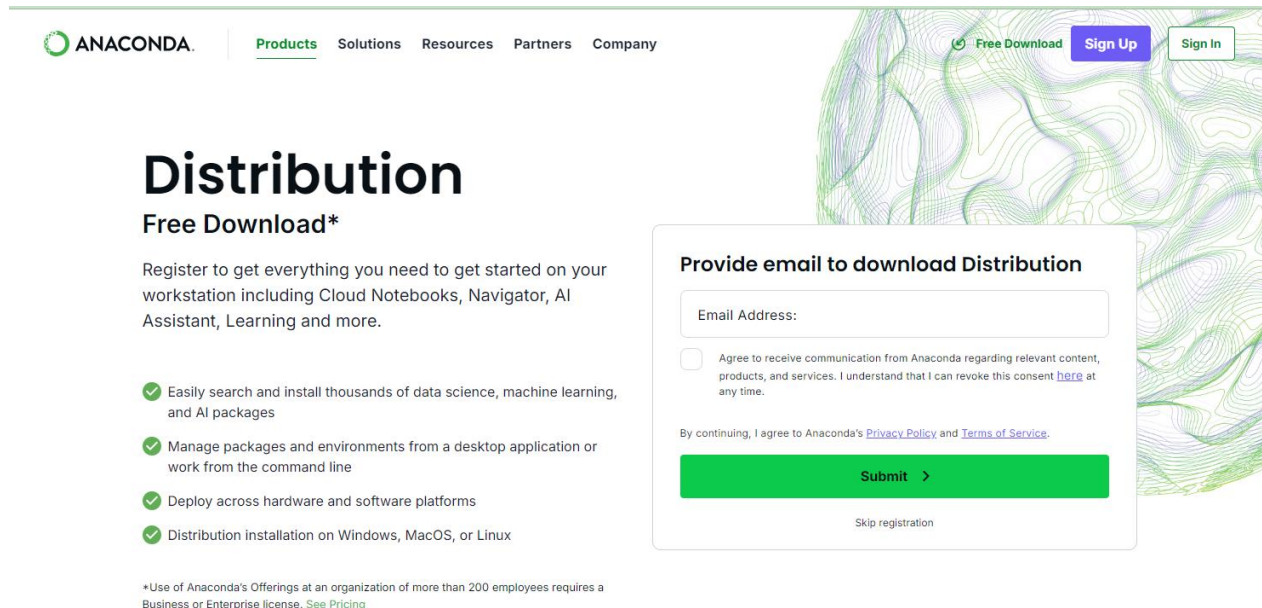
라이브러리	내용
Scikit-learn	기계 학습을 적용하면서 기계 학습의 전반적인 이해를 높이도록 하는 것이 목적으로 파이썬을 기반으로 하며 기계 학습을 쉽게 적용
NumPy	파이썬으로 과학 계산을 하려면 꼭 필요한 패키지. 다차원 배열을 위한 기능과 선형 대수 연산과 푸리에 변환 같은 고수준 수학 함수와 유사 난수 생성기를 포함
SciPy	과학 계산을 함수를 모아놓은 파이썬 패키지. 고성능 선형 대수, 함수 최적화, 신호 처리, 특수한 수학 함수와 통계 분포등을 포함한 많은 기능 제공
Pandas	데이터 처리와 분석을 위한 파이썬 라이브러리. R의 data.frame을 본떠서 설계한 DataFrame이라는 데이터 구조를 기반으로 만들어졌다.

\* 라이브러리 공식 문서 제공 url : <https://hamait.tistory.com/819>

# TensorFlow 설치

- 파이썬 설치 : '파이썬 인터프리터 + 파이썬 패키지' 설치

파이썬 배포판	홈페이지
Anaconda	<a href="https://www.anaconda.com/">https://www.anaconda.com/</a>
Python(x, y)	<a href="http://python-xy.github.io/">http://python-xy.github.io/</a>
Enthought Canopy	<a href="https://assets.enthought.com/downloads/">https://assets.enthought.com/downloads/</a>
WinPython	<a href="http://winpython.github.io/">http://winpython.github.io/</a>

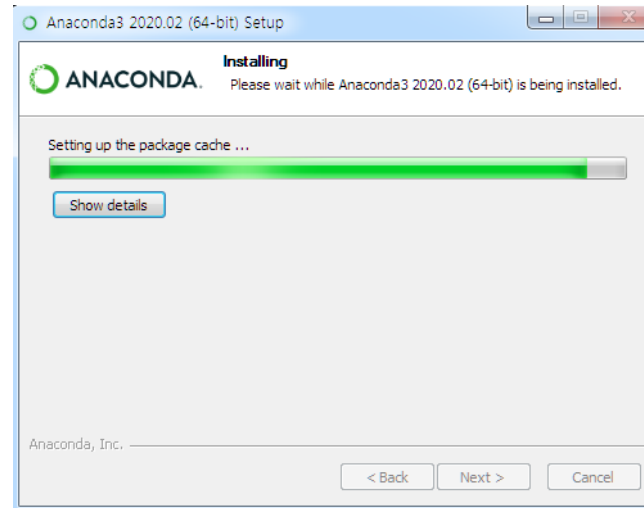
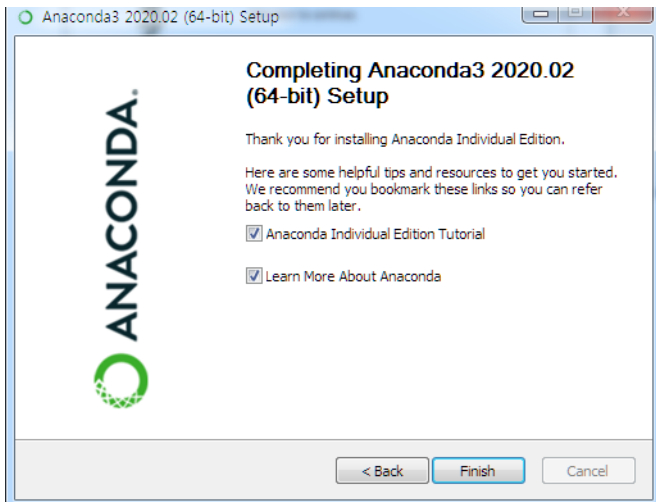
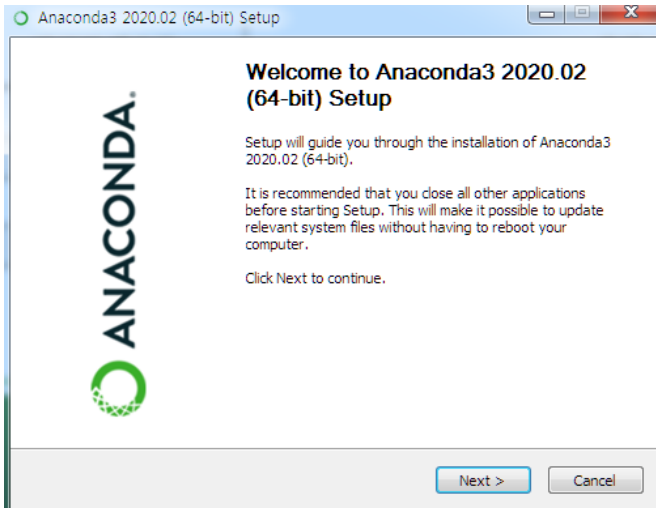


The screenshot shows the Anaconda website's 'Distribution' page. The header includes the Anaconda logo and navigation links: Products, Solutions, Resources, Partners, and Company. On the right, there are buttons for 'Free Download', 'Sign Up', and 'Sign In'. The main heading is 'Distribution' with a subheading 'Free Download\*'. Below this, a paragraph states: 'Register to get everything you need to get started on your workstation including Cloud Notebooks, Navigator, AI Assistant, Learning and more.' A list of four benefits is provided, each with a green checkmark: 1. Easily search and install thousands of data science, machine learning, and AI packages. 2. Manage packages and environments from a desktop application or work from the command line. 3. Deploy across hardware and software platforms. 4. Distribution installation on Windows, MacOS, or Linux. A registration form titled 'Provide email to download Distribution' is on the right, featuring an 'Email Address:' input field, a checkbox for agreeing to communication, and a 'Submit >' button. A link to 'Skip registration' is at the bottom of the form. A small footnote at the bottom left states: '\*Use of Anaconda's Offerings at an organization of more than 200 employees requires a Business or Enterprise license. See Pricing'.

Anoconda download url : <https://www.anaconda.com>

# TensorFlow 설치

- Anaconda 파이썬



Thanks for installing Anaconda!

Anaconda Navigator helps you easily start important Python applications and manage the packages in your local Anaconda installation. It also connects you to online resources for learning and engaging with the Python, SciPy, and PyData community.

To help us improve Anaconda Navigator, fix bugs, and make it even easier for everyone to use Python, we gather anonymized usage information, just like most web browsers and mobile apps.

To opt out of this, please uncheck below (You can always change this setting in the Preferences menu).

☒ Yes, I'd like to help improve Anaconda.

Ok

Ok, and don't show again

# TensorFlow 설치

- Anaconda Prompt 에서 최신 사양으로 업데이트
  - 이전에 설치된 버전이 있으면 업데이터 되는데 시간이 소요
  - Tensorflow, Keras 도 같이 인스톨

Anaconda Prompt	내용
Python --version	아나콘다에 포함돼 있는 파이썬의 버전 확인
conda --version	아나콘다 버전 확인
conda update - all	설치된 파이썬 패키지 모두 최신으로 업데이트
pip install tensorflow	아나콘다에서 텐서플로우 설치(아나콘다에서 케라스 포함)

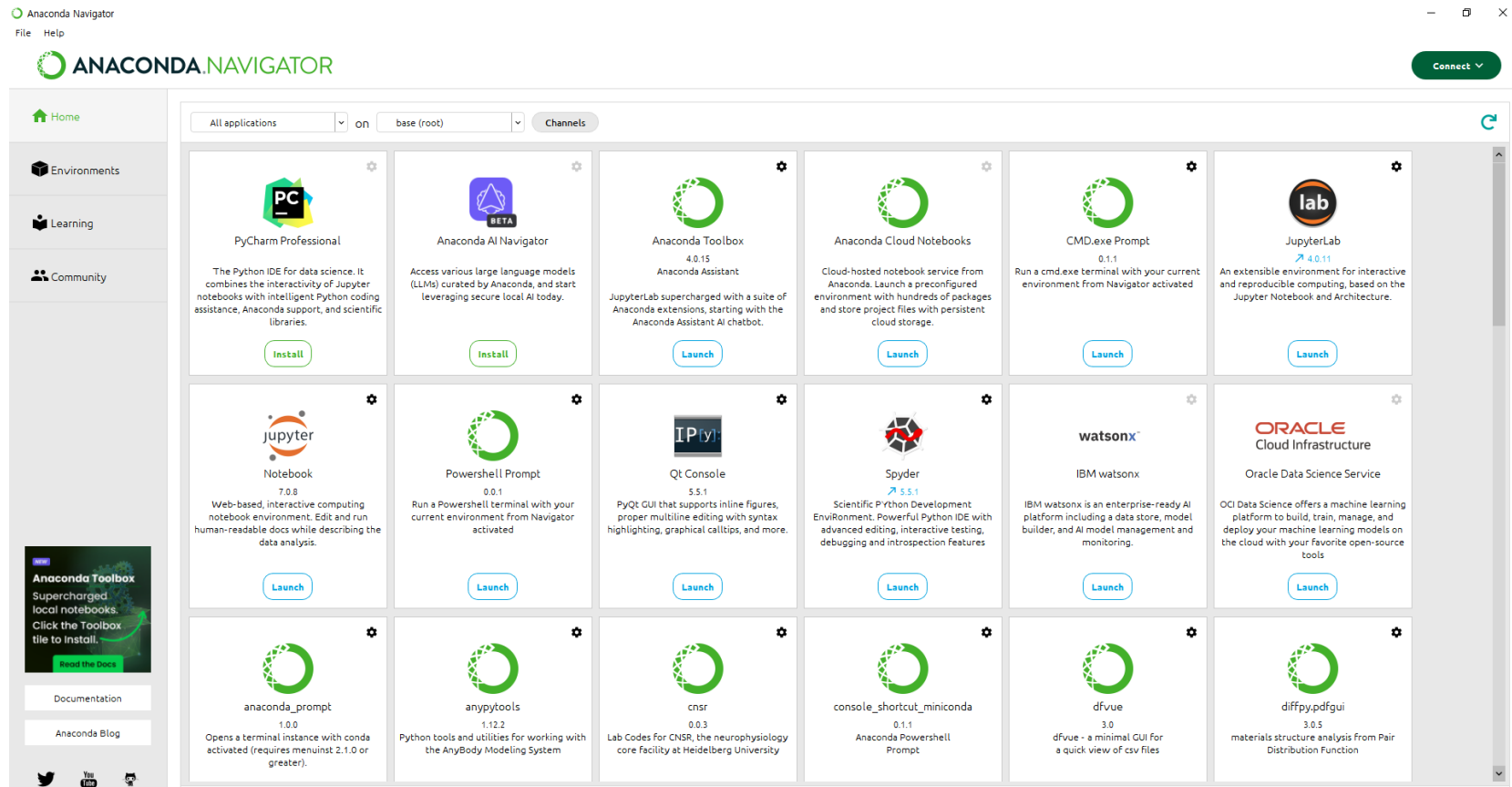
```
Anaconda Prompt - conda update --all

Downloading and Extracting Packages
pyodbc-4.0.30      67 KB ##### 100%
webencodings-0.5.1 12 KB ##### 100%
bzip2-1.0.8       148 KB ##### 100%
pyzmq-19.0.0      428 KB ##### 100%
yapf-0.29.0       119 KB ##### 100%
beautifulsoup4-4.8.2 157 KB ##### 100%
urllib3-1.25.7    160 KB ##### 100%
boto-2.49.0       838 KB ##### 100%
importlib-metadata-1 42 KB ##### 100%
libxml2-2.9.10    3.1 MB ##### 100%
snappy-1.1.8      81 KB ##### 100%
click-7.1.1       64 KB ##### 100%
pluggy-0.13.0     28 KB ##### 100%
mkl_random-1.1.0  265 KB ##### 100%
psutil-5.7.0      346 KB ##### 100%
zeromq-4.3.2      10.0 MB ##### 100%
astroid-2.3.3     282 KB ##### 100%
sphinxcontrib-jsmath 7 KB ##### 100%
jinja2-2.11.1     94 KB ##### 100%
nose-1.3.7        235 KB ##### 100%
cryptography-2.8  564 KB ##### 100%
conda-verify-3.1.1 49 KB ##### 100%
singledispatch-3.4.0 15 KB ##### 100%
py-1.8.1          66 KB ##### 100%
sphinxcontrib-websup 34 KB ##### 100%
mock-3.0.5        44 KB ##### 100%
typing_extensions-3.8 38 KB ##### 100%
sympy-1.5.1       10.3 MB ##### 100%
```

# TensorFlow 설치

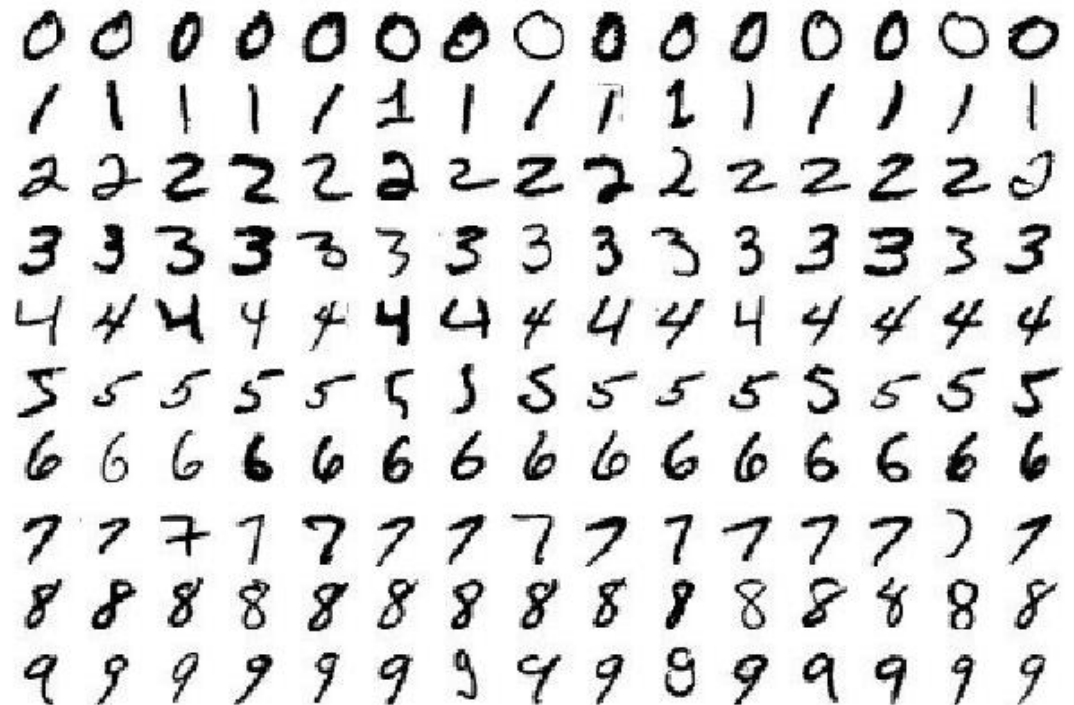
## ■ Anaconda 파이썬

- Jupiter Notebook 실행 ( 인터랙티브 형식의 라이브 코드를 제공하는 웹 어플리케이션, 웹 브라우저 안에서 실행하고 싶은 코드를 입력하고 그 결과를 바로 확인 할 수 있음)



# TensorFlow MNIST 숫자인식

- MNIST 란 ? MNIST (Modified National Institute of Standards and Technology)는 손으로 쓴 숫자들로 이루어진 대형 데이터베이스이며, 다양한 화상 처리 시스템을 트레이닝하기 위해 일반적으로 사용. (출처 : Wikipedia)
- 28 \* 28 크기의 0부터 9사이의 숫자 이미지에 해당하는 레이블로 구성된 숫자 이미지 데이터 베이스
- 각 데이터의 숫자가 무엇인가에 대한 라벨이 함께 붙어 있다. 예) 5, 0, 4, 1 이라는 라벨이 붙어 있다



# TensorFlow MNIST 숫자인식

- 문제 : 이미지를 들여다 보고 그 이미지가 어떤 숫자라고 예측모델이고 가장 높은 수준을 가진 정교한 모델을 만드는 것.
  - 첫 시간에 했던 데이터 분석의 정의에서 정보를 가지고 할 수 있는 것 4가지 꼭 확인
  - 이 문제를 풀기 위하여 예측 알고리즘인 회귀분석을 알아야 하고 소프트맥스 회귀를 알아야 한다.
- 회귀계수를 구하는 방법을 먼저 이해해 보도록 하겠다. 엑셀의 추가 기능을 활성화 한다.
  - 미분값을 구하고 상수를 구해보자

번호	노동력(X)	생산량(Y)	X <sup>2</sup>	XY
1	267	428	71,289	114,276
2	263	430	69,169	113,090
3	238	417	56,644	99,246
4	219	384	47,961	84,096
5	274	432	75,076	118,368
6	257	425	66,049	109,225
7	321	474	103,041	152,154
8	305	462	93,025	140,910
9	285	449	81,225	127,965
10	247	405	61,009	100,035
합계	2,676	4,306	724,488	1,159,365
	A	B	C	D

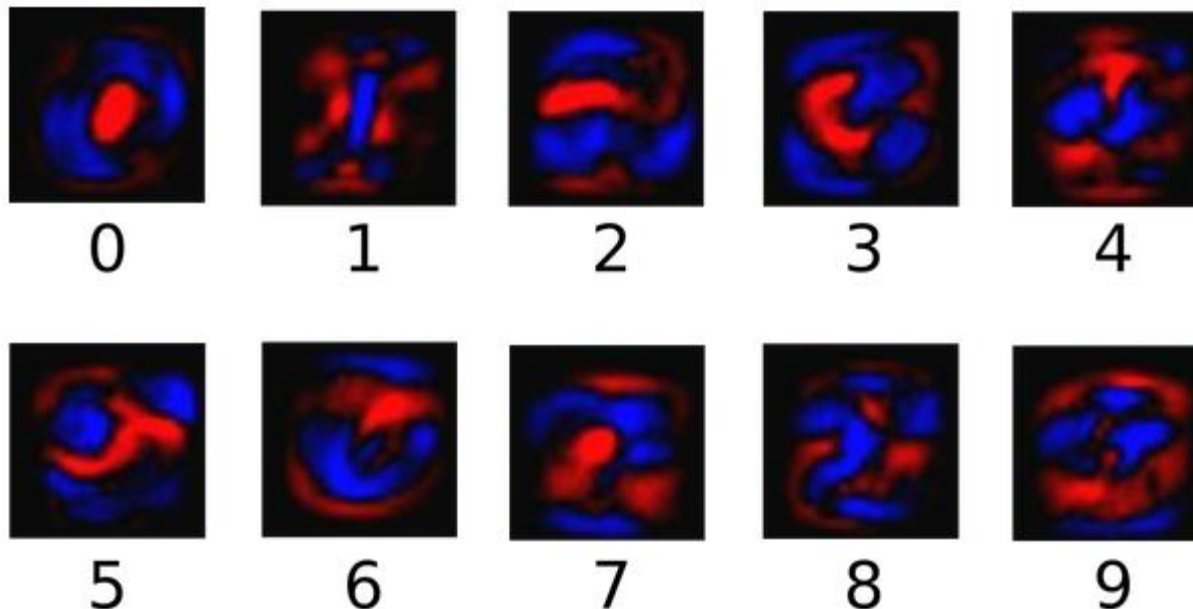
<



# TensorFlow MNIST 숫자인식

- Softmax 회귀 : 이미지를 보았을 때 각 숫자로 판단할 확률을 구함, 구현한 모델이 9라는 숫자의 이미지를 보고 그 숫자가 80% 정도이 확률로 9 라고 확신 할 수 있지만 곡선의 영향으로 8일 가능성도 5%를 주고 나머지 모든 숫자들에는 확신할 수 없으므로 아주 작은 확률로 준다.
  - 소프트맥스 회귀는 자연스럽고 간단한 모델인 고전적인 경우이고 어떤 대상이 여러 다양한 것들에 하나일 확률을 계산하려면 소프트 맥스를 사용함
  - 소프트맥스의 2단계 :
    - 입력이 특정 클래스에 해당되는지에 대한 증거를 더하고 그 다음 증거를 확률로 변환
    - 주어진 이미지가 특정한 클래스에 들어가는 지의 증거를 모아 계산하기 위해 픽셀 농도의 가중합을 사용

아래 그림에서 빨간색은 (+) 가중치 적용, 파란색은 (-) 가중치 적용



# TensorFlow MNIST 숫자인식

- Softmax 회귀 : 주어진 입력  $x$ 에 대한 클래스  $i$ 의 증거는 아래 공식과 같음  
 $W_i$ 는 가중치  $b_i$ 는 클래스  $i$ 에 대한 편향,  $j$ 는 입력한 이미지  $x$ 의 픽셀들에 따른 합을 구하기 위한 인덱스

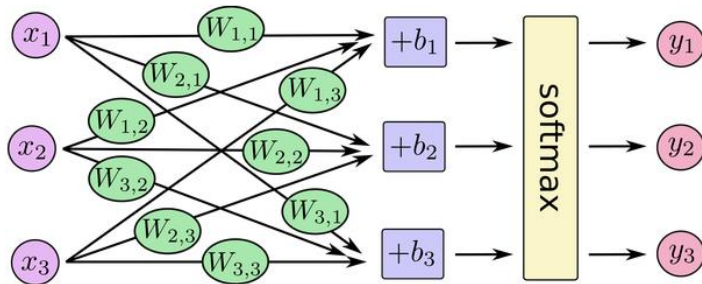
$$\text{evidence}_i = \sum_j W_{i,j} x_j + b_i$$

- Softmax 는 우리가 제공한 선형 함수의 출력 결과를 원하는 형태로 만들기 위해 활성화나 링크 함수의 형태로 적용되고 10가지 경우에 대한 확률 분포를 나타냄

$$\text{softmax}(x)_i = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$$

$$y = \text{softmax}(Wx + b)$$

- 아래 그림은 Softmax 회귀를 나타내는 데 앞에서 배웠던 회귀분석을 이해하기 바란다.



$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \text{softmax} \begin{bmatrix} W_{1,1}x_1 + W_{1,2}x_2 + W_{1,3}x_3 + b_1 \\ W_{2,1}x_1 + W_{2,2}x_2 + W_{2,3}x_3 + b_2 \\ W_{3,1}x_1 + W_{3,2}x_2 + W_{3,3}x_3 + b_3 \end{bmatrix}$$

# TensorFlow MNIST 숫자인식

- Softmax 회귀 : 아래 그림에서 각 이미지는 가로 세로 각 28픽셀, 이것을 숫자로 구성된 행렬로 취급
  - $28 \times 28 = 784$ 개의 숫자를 갖는 벡터로 단순화할 수 있으며 이미지들은 784 차원 벡터 공간안에서 아주 풍부한 구조를 지닌 점들로 구성

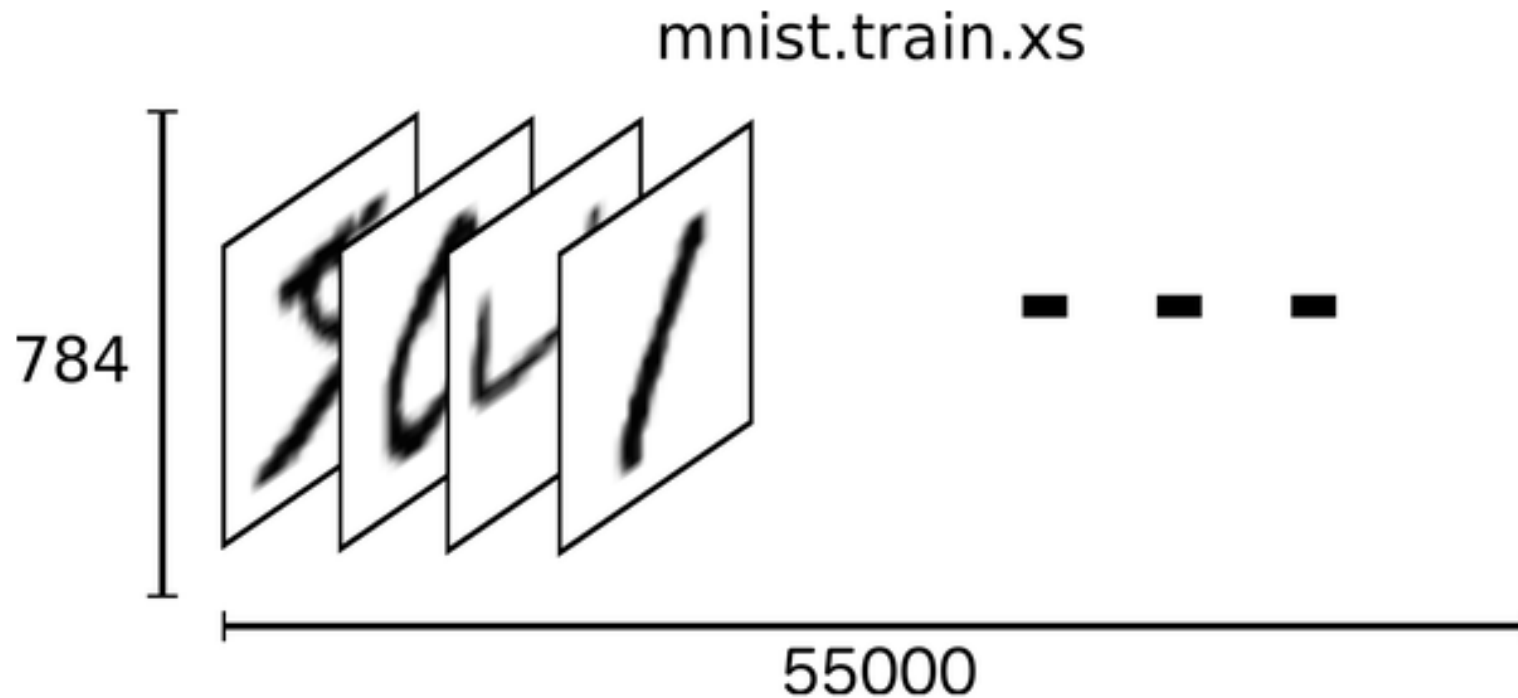


21

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	.6	.8	0	0	0	0	0	0
0	0	0	0	0	0	0	.7	1	0	0	0	0	0	0
0	0	0	0	0	0	0	.7	1	0	0	0	0	0	0
0	0	0	0	0	0	0	.5	1	.4	0	0	0	0	0
0	0	0	0	0	0	0	0	1	.4	0	0	0	0	0
0	0	0	0	0	0	0	0	1	.4	0	0	0	0	0
0	0	0	0	0	0	0	0	1	.7	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	.9	1	.1	0	0	0	0
0	0	0	0	0	0	0	0	.3	1	.1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

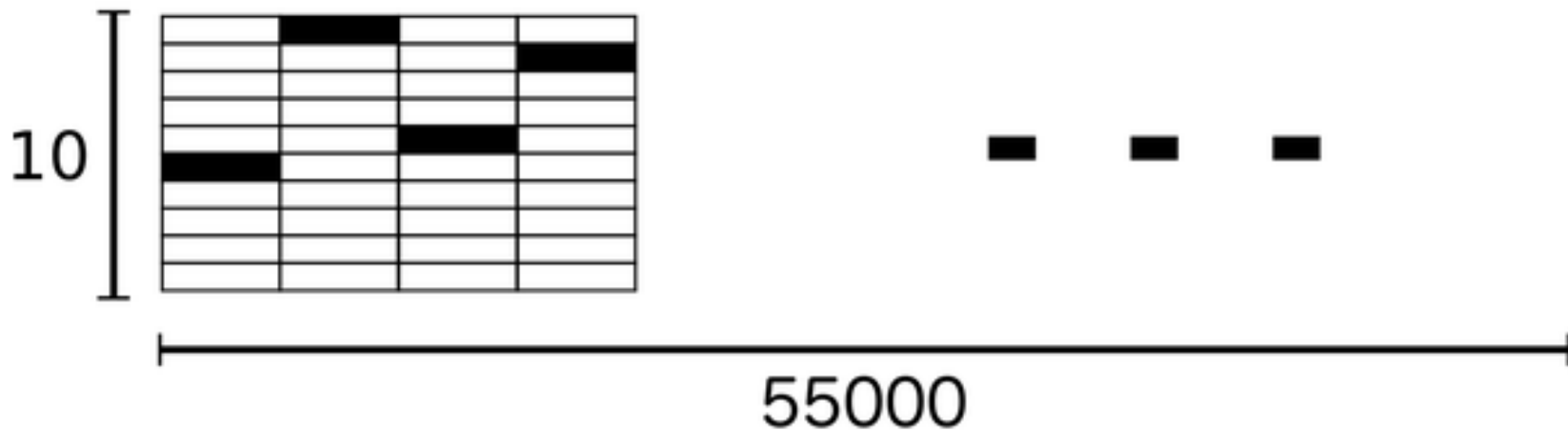
# TensorFlow MNIST 숫자인식

- Softmax 회귀 : 아래 그림에서 [55000, 784] 형태의 텐서 (n차원 행렬)을 얻었고 첫번째 차원 인덱스는 이미지, 두번째는 이미지의 픽셀 대응



# TensorFlow MNIST 숫자인식

- Softmax 회귀 : 아래 그림에서 MNIST 안에서 대응되는 라벨들은 주어진 각 이미지가 어떤 숫자를 나타내는가를 의미하는 0에서 9 까지의 숫자임
  - “one- hot 벡터 “는 하나의 차원만 1이고 나머지는 0으로 채워진 벡터로 n번째 숫자는 n 번째 차원이 1인 벡터로 표시됨, (예를 들어 3은 [0,0,0,1,0,0,0,0,0,0])



- 코딩 및 실습은 다음 시간에 진행함.