

# 파이썬 텐서플로 기반 딥러닝

컴퓨터정보공학과  
강 환수 교수

# 파이썬 기반 딥러닝 소개

- 인공지능 개요
  - 머신러닝과 딥러닝
  - 딥러닝
    - 인공신경망
  - 머신러닝 딥러닝 라이브러리
    - 텐서플로, theano, cntk
- 가상 환경
  - 독립적인 파이썬 개발 환경
    - 하나의 컴퓨터에 여러 개 생성 가능

# 수업 개요

- 2020-7/16(목), 7/17(금)에서 8/14(금)까지 4주 동안 매주 목 금
  - 총 48시간
  - 2시간 수업, 1시간 질의 및 자기주도학습
- **교재**
  - 시작하세요! 텐서플로 2.0 프로그래밍
  - 코딩 참조:
    - 파이썬으로 시작하는 컴퓨터과학입문
    - 파이썬프로그래밍개론(An Introduction to Programming using Python)
    - Realpython.com
- **일정(변경될 수 있음)**

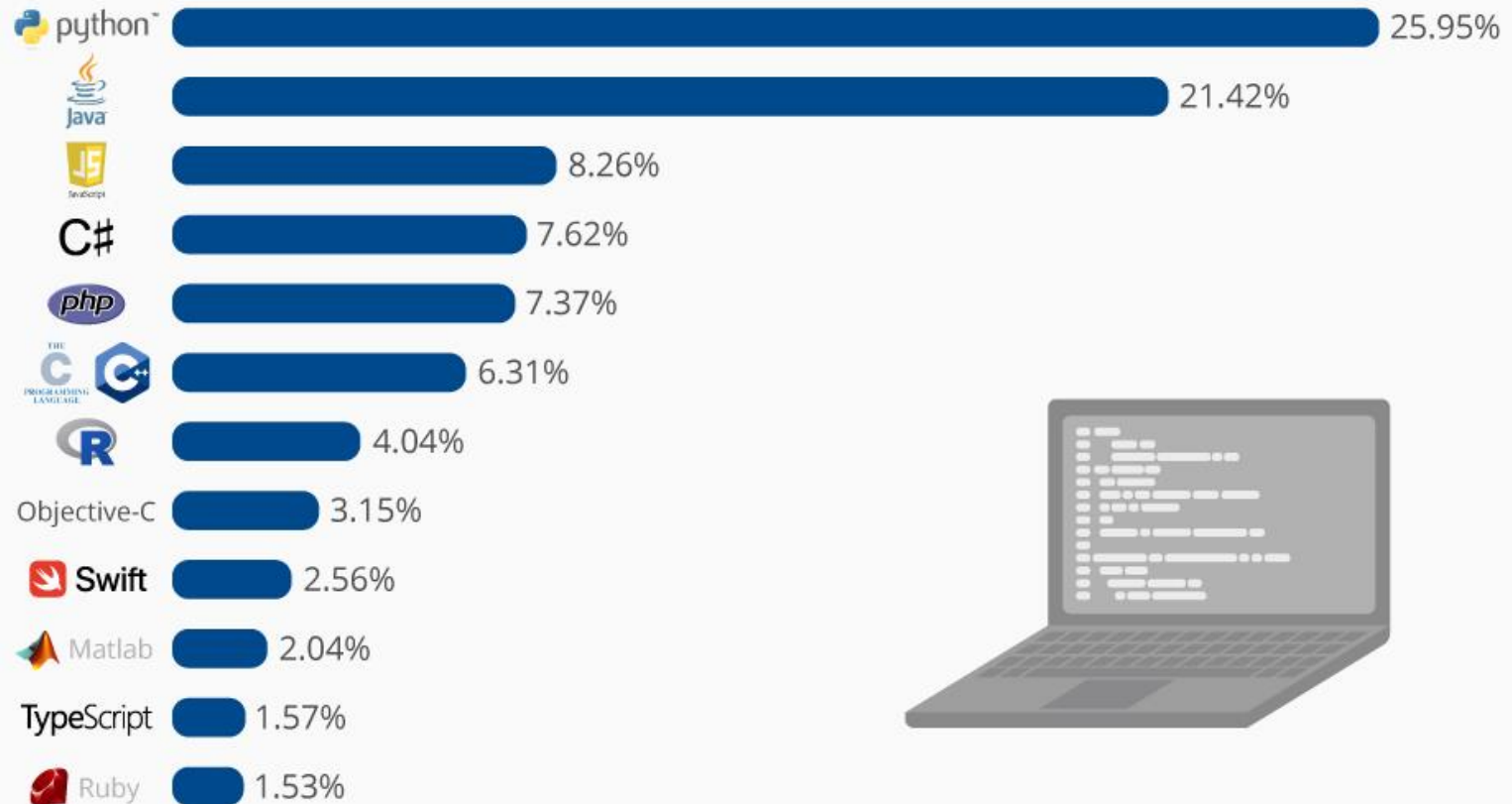
|               |     | 오전                  | 오후                          |
|---------------|-----|---------------------|-----------------------------|
| 2020-05-07(목) | 1일차 | 파이썬 기본 개발 환경        | 파이썬 문법 코딩하기(기본과 함수, 리스트)    |
| 2020-05-14(목) | 2일차 | 아나콘다 설치, 주피터 노트북 사용 | 파이썬 문법 코딩하기(딕셔너리와 튜플, 집합)   |
| 2020-05-21(목) | 3일차 | 모듈 기본, 파이참 설치       | 파이썬 문법 코딩하기, import this 코딩 |
| 2020-05-28(목) | 4일차 | 가상환경 개요와 생성         | 직접 가상환경 생성과 파이참에서 생성        |
| 2020-06-04(목) | 5일차 | 주피터 노트북과 파이참 심화     | 구글 코랩                       |
| 2020-06-11(목) | 6일차 | 비주얼 스튜디오 코드         | 서브라임텍스트와 아톰                 |

# 데이터과학 분야의 핵심 언어 파이썬

# 파이썬 언어의 인기

## The Most Popular Programming Languages

Share of the most popular programming languages in the world\*



# 파이썬의 활용 영역

- 데이터 과학 학습 순서



# 파이썬의 간결성

## 자바

```
static void main(String args[]){
    ...

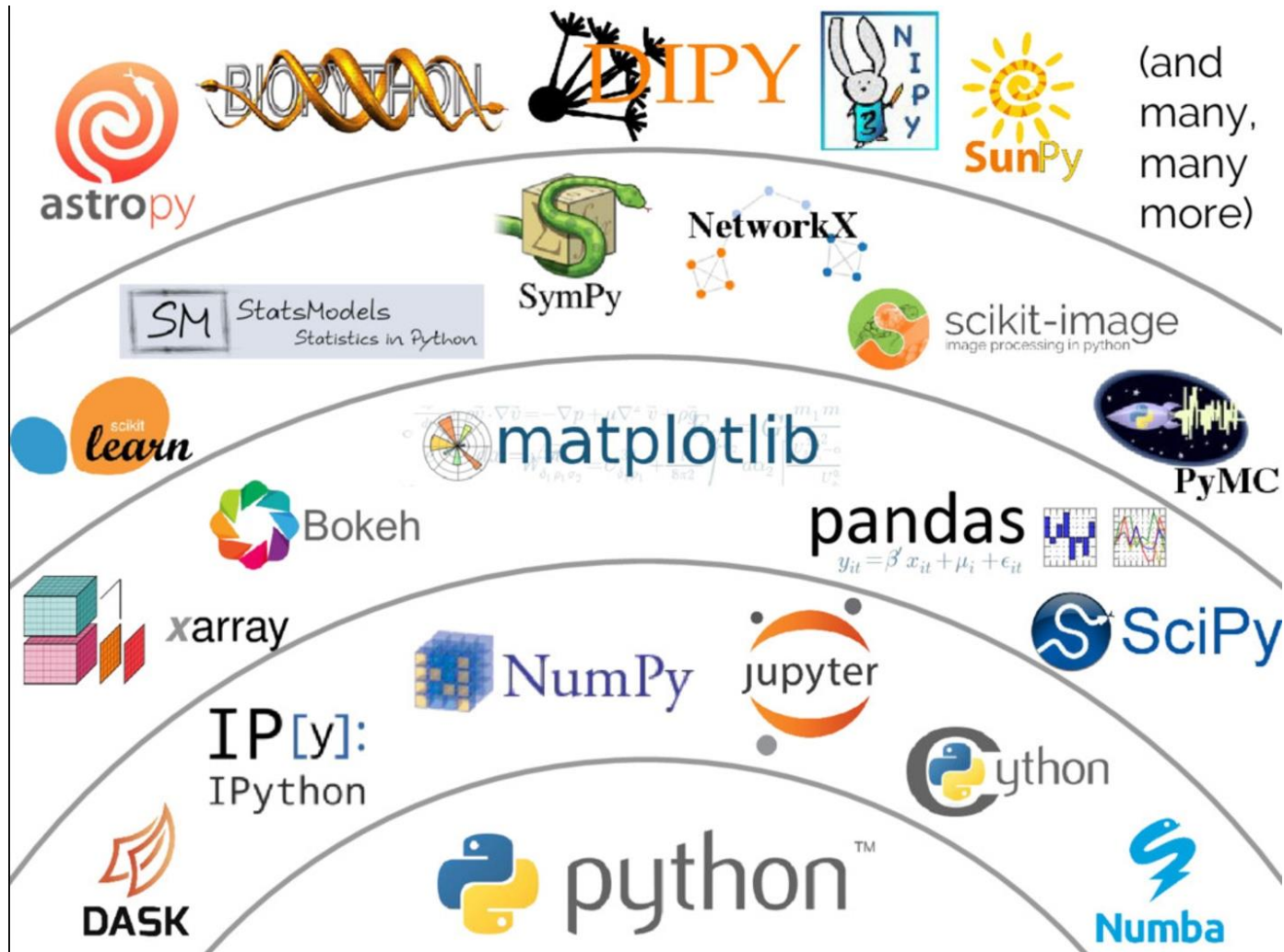
    int 횟수 = 0

    for(int i=0; i< VOC.length; i++){
        String 의견 = VOC.get(i);
        boolean 포함 = 의견.matches("한글");
        if(포함){
            횟수 += 1
        }
    }
}
```

## 파이썬

```
횟수 = 0
for 의견 in VOC:
    if '한글' in 의견:
        횟수 += 1
print(횟수)
```

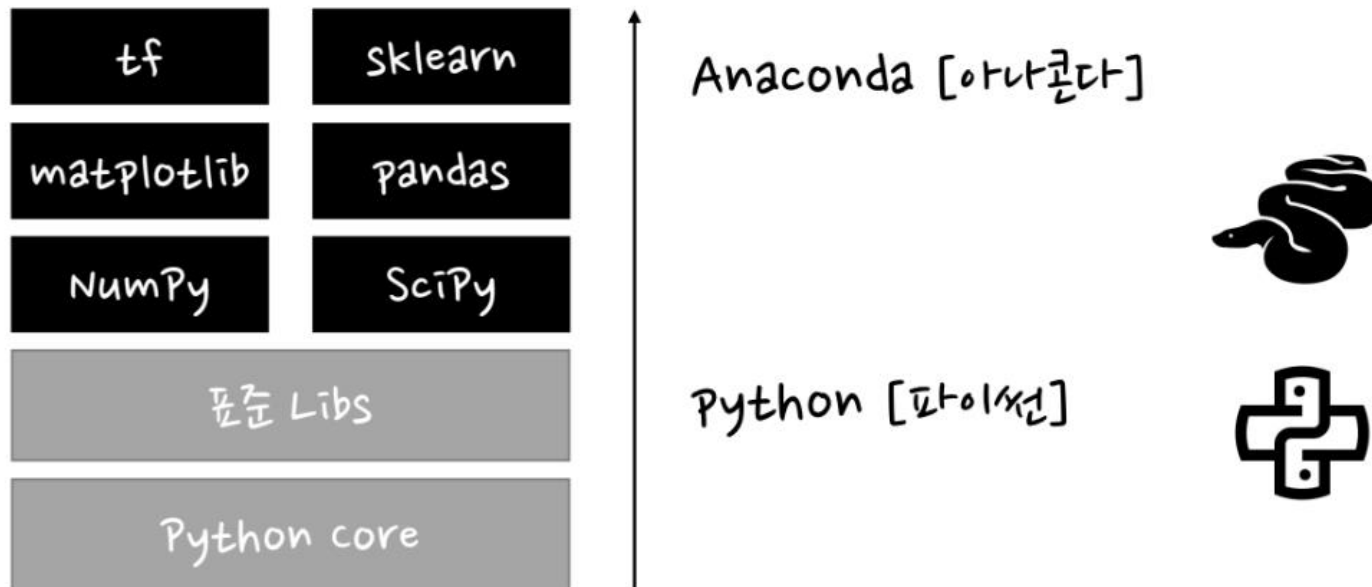
# 파이썬의 확장성





# 주 개발환경

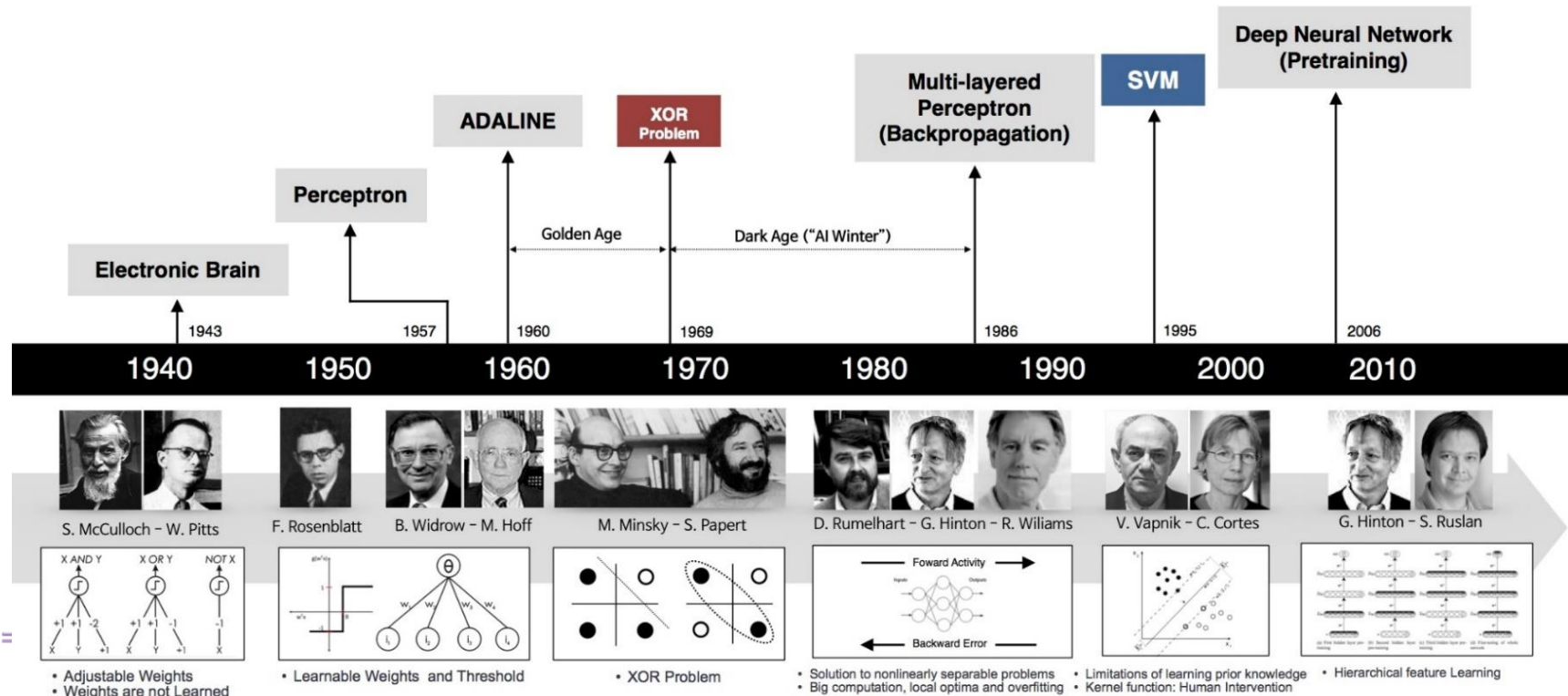
- 파이썬
- 아나콘다



# 인공지능과 딥러닝 개요

# AI와 딥러닝 역사

- 1940년대 부터 시작한 분야
  - 두 번의 흑한기를 지냄
    - AI의 첫번째 암흑기 1969-1980
    - AI의 두번째 암흑기 1987-1993
- 2010년 이후 여러 문제 해결
  - 최고의 전성기를 누림



# 왜 지금 딥러닝이 인기?

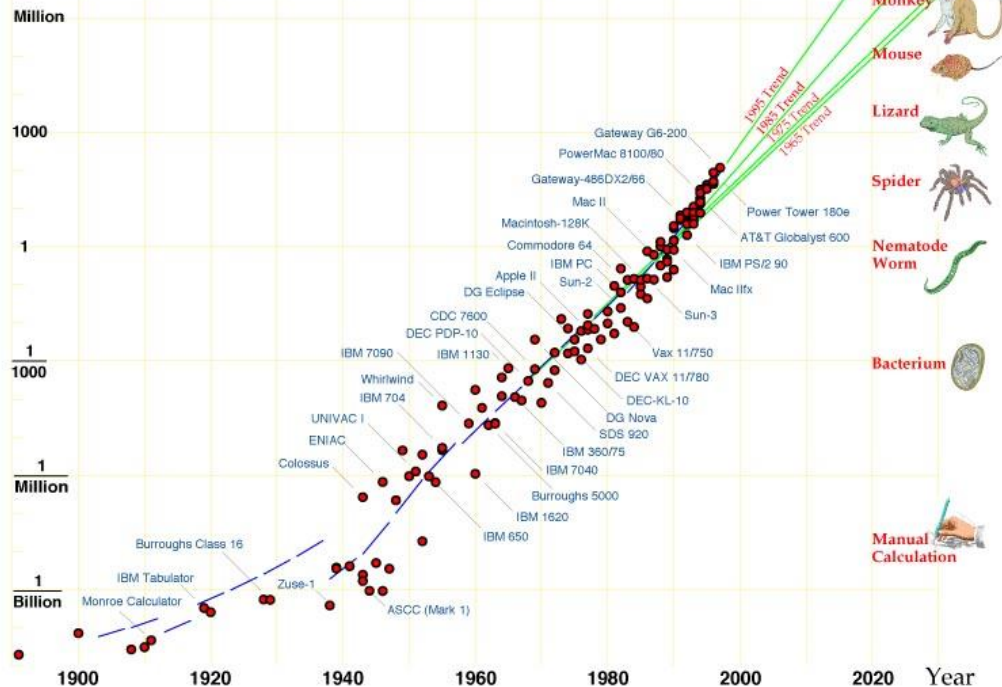
- 딥러닝의 문제가 해결되고 있는 과정
  - 빅데이터, 계산 속도, 알고리즘

## BIG DATA



Evolution of Computer Power/Cost

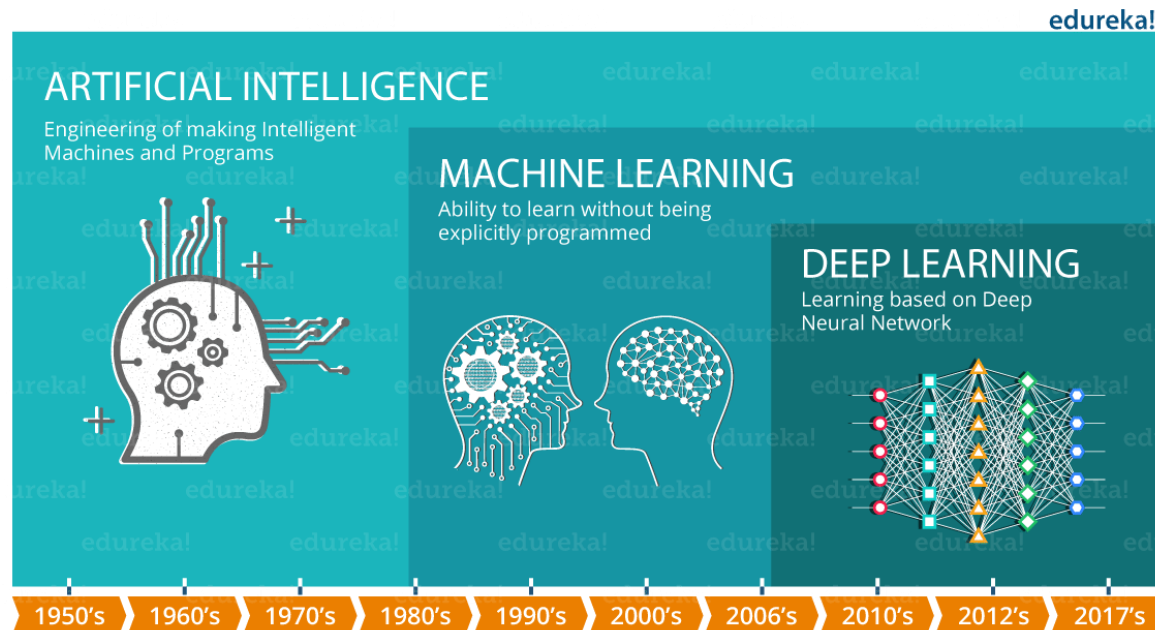
MIPS per \$1000 (1997 Dollars)



# 인공지능과 머신러닝, 딥러닝

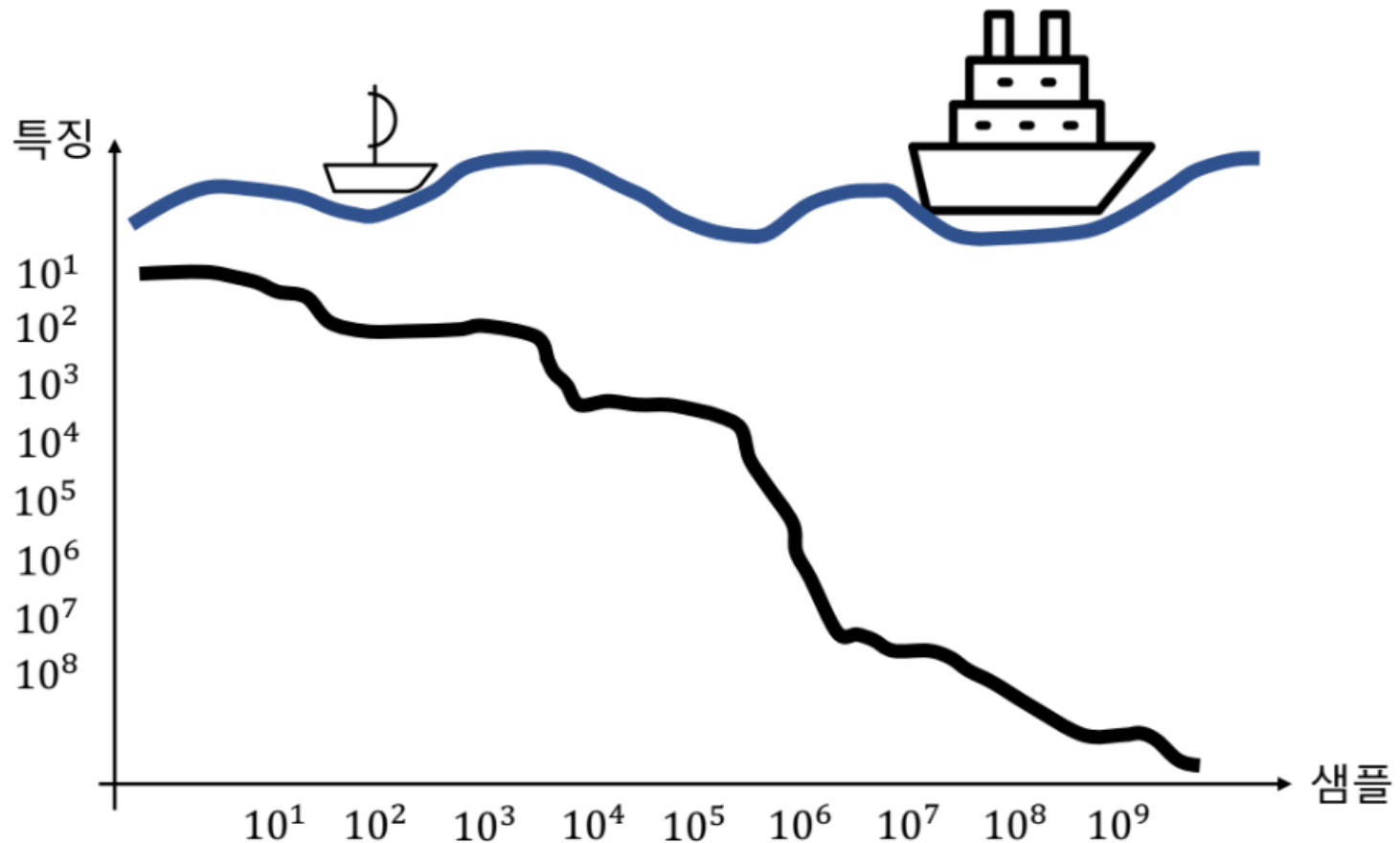
## • 인공지능(AI: Artificial Intelligence)

- 컴퓨터가 인간처럼 지적 능력을 갖게 하거나 행동하도록 하는 모든 기술
- 머신러닝(machine learning)
  - 머신러닝은 기계가 스스로 학습할 수 있도록 하는 인공지능의 한 연구 분야
  - SVM(Support Vector Machine): 수학적 방식의 학습 알고리즘
  - 딥러닝
    - 다중 계층의 신경망 모델을 사용하는 머신러닝의 일종.



# 기계학습과 딥러닝

- 특징과 데이터가 많을수록 딥러닝에 적합

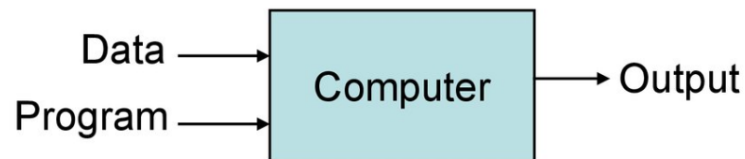


# 머신러닝

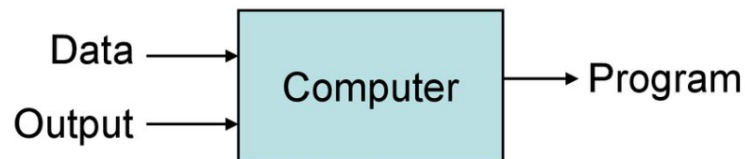
- 기계학습이라고도 부르는 머신러닝(machine learning)

- 주어진 데이터를 기반으로 기계가 스스로 학습하여 성능을 향상시키거나 최적의 해답을 찾기 위한 학습 지능 방법
  - 스스로 데이터를 반복적으로 학습하여 기술을 터득하는 방식
  - 명시적으로 프로그래밍(explicit programmng)을 하지 않아도 컴퓨터가 학습을 할 수 있도록 해주는 인공지능의 한 형태
  - 더 많은 데이터가 유입되면, 컴퓨터는 더 많이 학습을 하고, 시간이 흐르면서 더 스마트 해져서 작업을 수행하는 능력과 정확도가 향상

## Traditional Programming



## Machine Learning



# 머신러닝 분류 개요

- 머신러닝은 지도학습과 자율학습, 그리고 강화학습으로 분류
  - 지도학습(supervised learning)
    - 올바른 입력과 출력의 쌍으로 구성된 정답의 훈련 데이터(labeled data)로부터 입출력 간의 함수를 학습시키는 방법
      - - k-최근접 이웃 (k-Nearest Neighbors)
      - - 선형 회귀 (Linear Regression)
      - - 로지스틱 회귀 (Logistic Regression)
      - - 서포트 벡터 머신 (Support Vector Machines (SVM))
      - - 결정 트리 (Decision Tree)와 랜덤 포레스트 (Random Forests)
      - - 신경망 (Neural networks)
  - 자율학습(unsupervised learning)
    - 정답이 없는 훈련 데이터(unlabeled data)를 사용하여 데이터 내에 숨어있는 어떤 관계를 찾아내는 방법
  - 강화학습(reinforcement learning)
    - 잘한 행동에 대해 보상을 주고 잘못된 행동에 대해 벌을 주는 경험을 통해 지식을 학습하는 방법
      - 딥마닝의 알파고
        - » 자동 게임분야



# 비지도 학습과 지도 학습

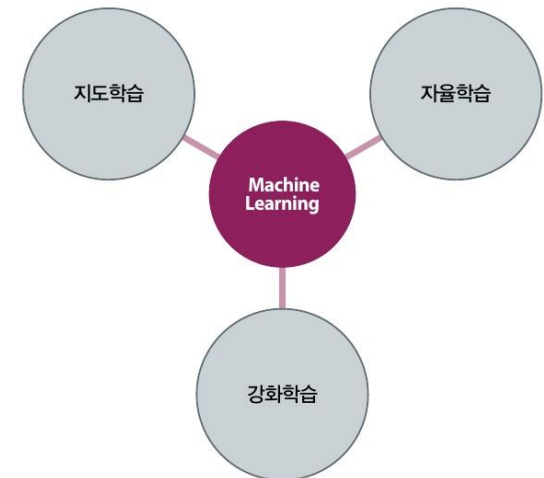
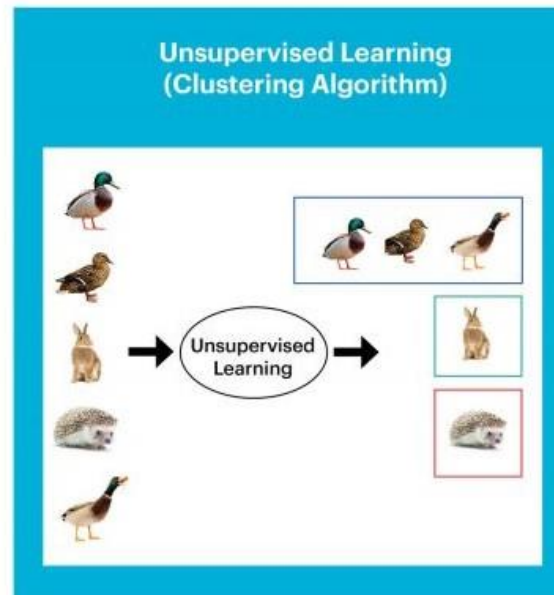
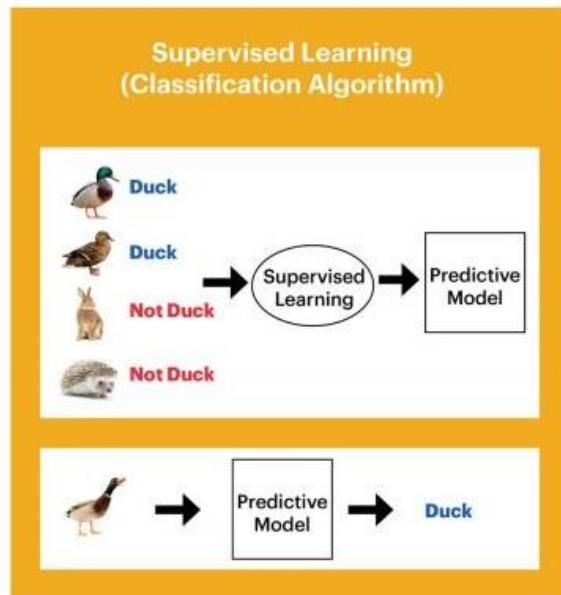
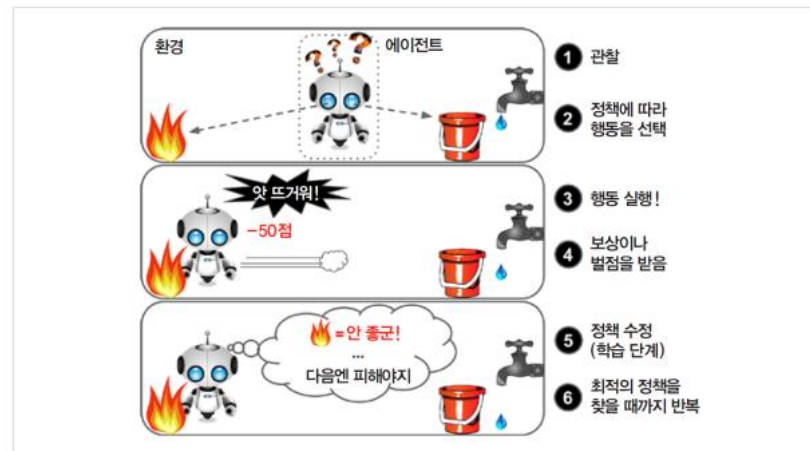


그림 10.26 ▶ 머신러닝 분야

## • 자율학습

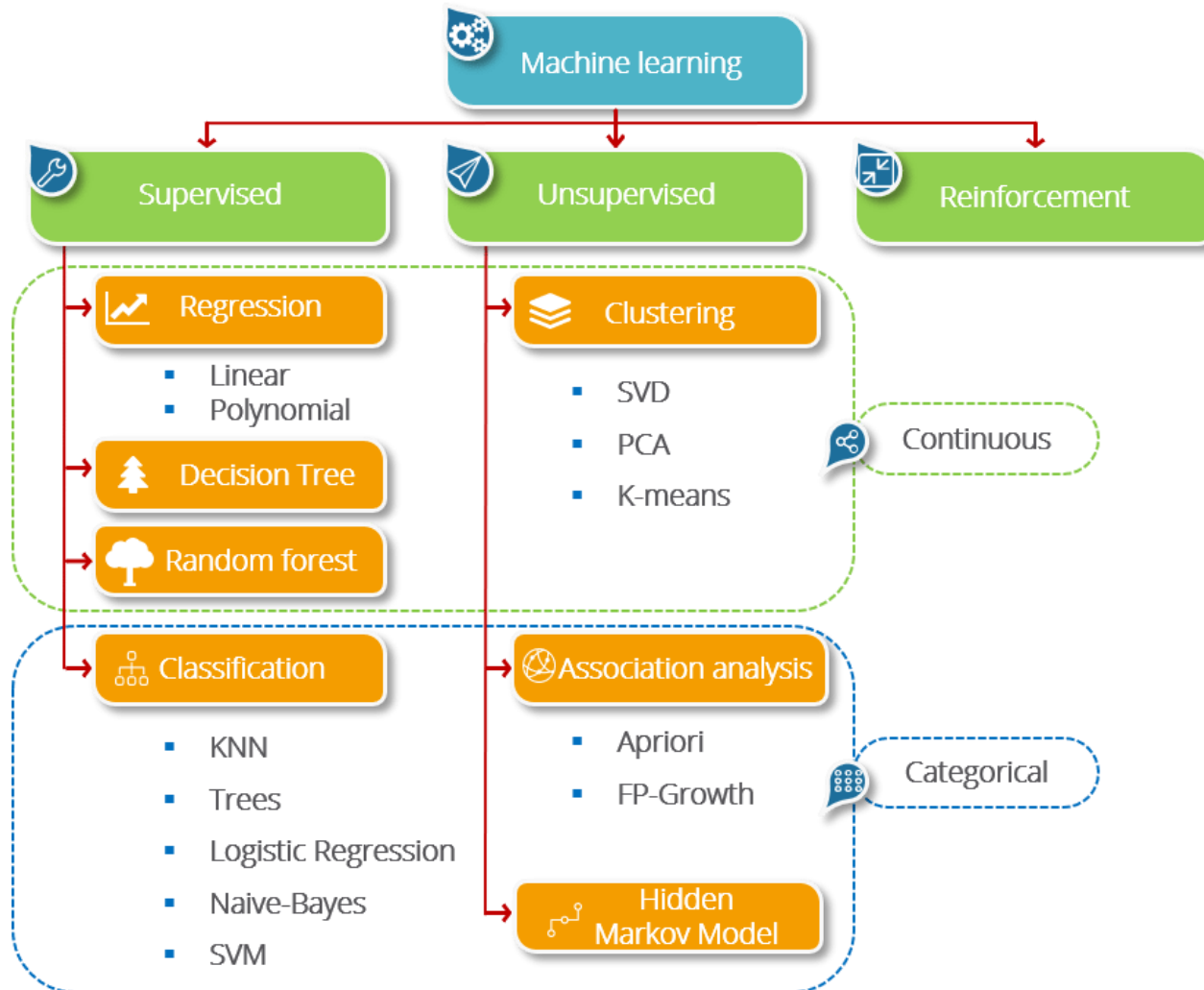
- 군집 (Clustering)
  - - k-평균 (k-Means)
  - - 계층 군집 분석 (Hierarchical Cluster Analysis (HCA))
  - - 기댓값 최대화 (Expectation Maximization)
- 시각화 (Visualization)와 차원 축소(Dimensionality reduction)
  - - 주성분 분석 (Principal Component Analysis (PCA))
  - - 커널 (kernel PCA)
  - - 지역적 선형 임베딩 (Locally-Linear Embedding (LLE))
- 연관 규칙 학습 (Association rule learning)
  - - 어프라이어리 (Apriori)
  - - 이클렛 (Eclat)

## • 강화학습



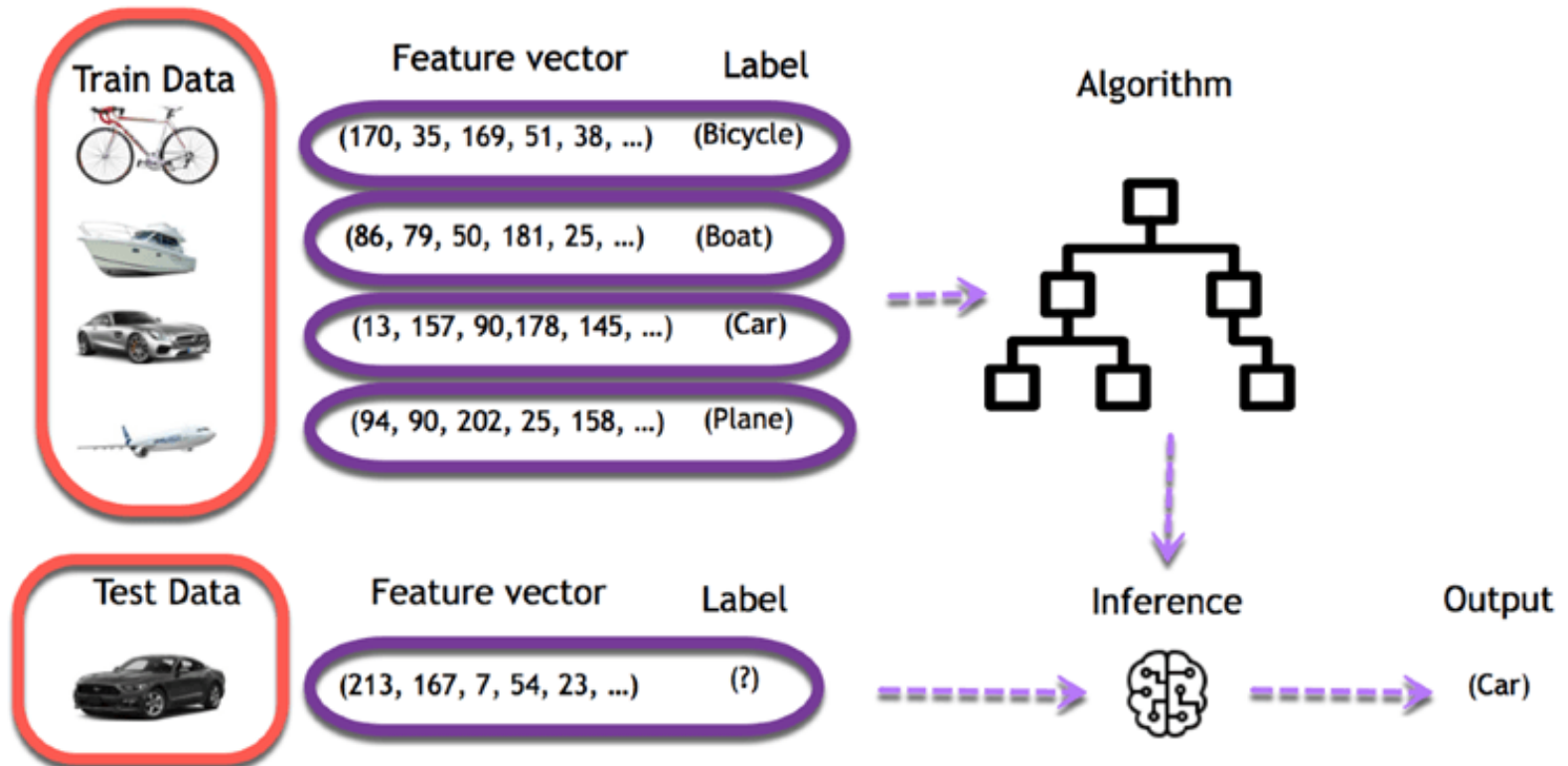
(그림 출처 : Hands-On Machine Learning 도서 - 한빛미디어)

# 머신 러닝 분류



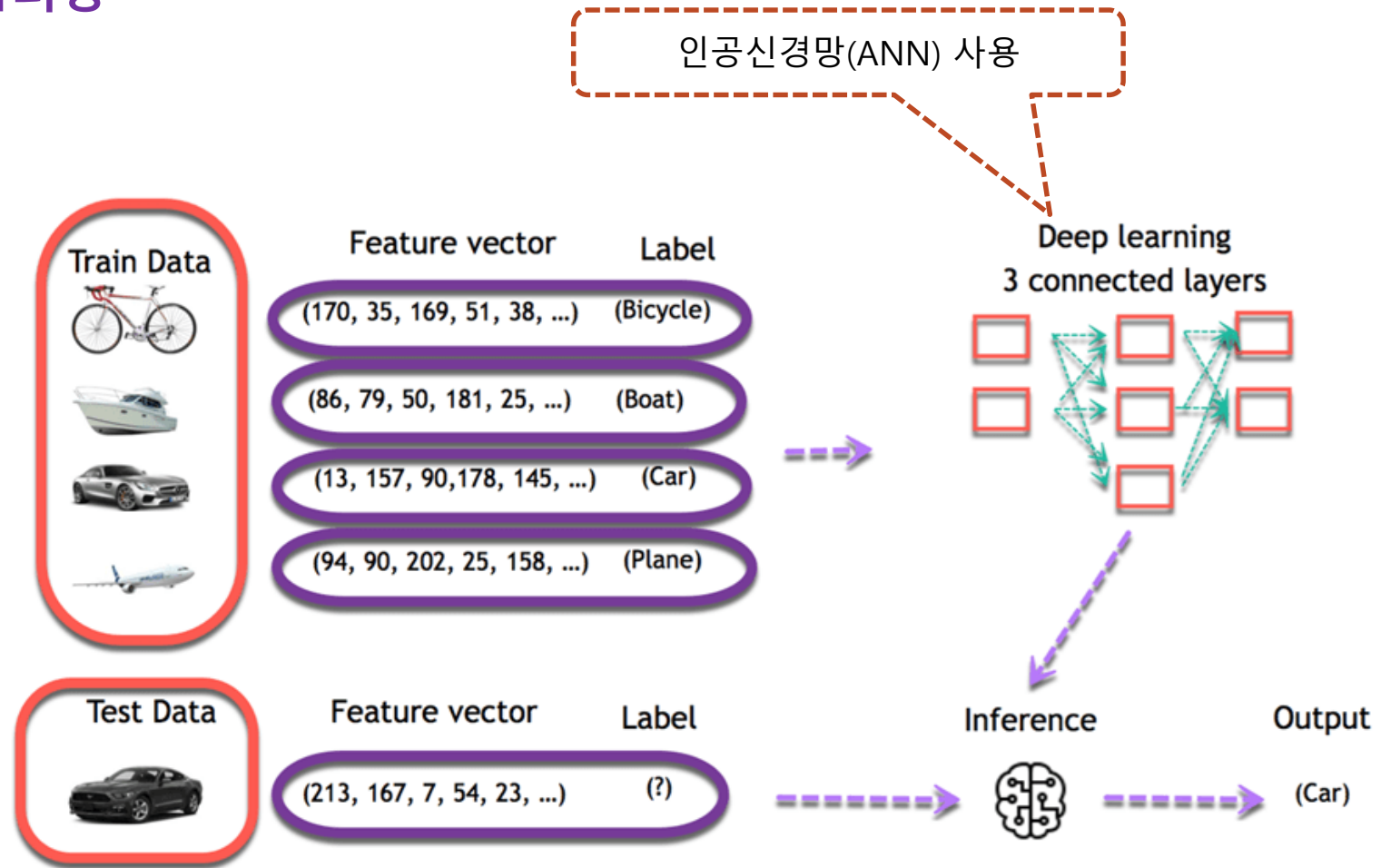
# 머신러닝과 딥러닝의 분류(1)

## 머신러닝



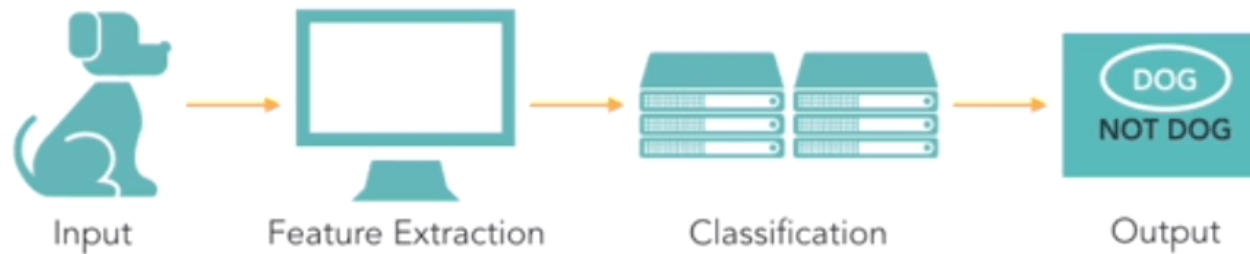
# 머신러닝과 딥러닝의 분류(2)

## • 딥러닝



# 머신러닝과 딥러닝의 차이

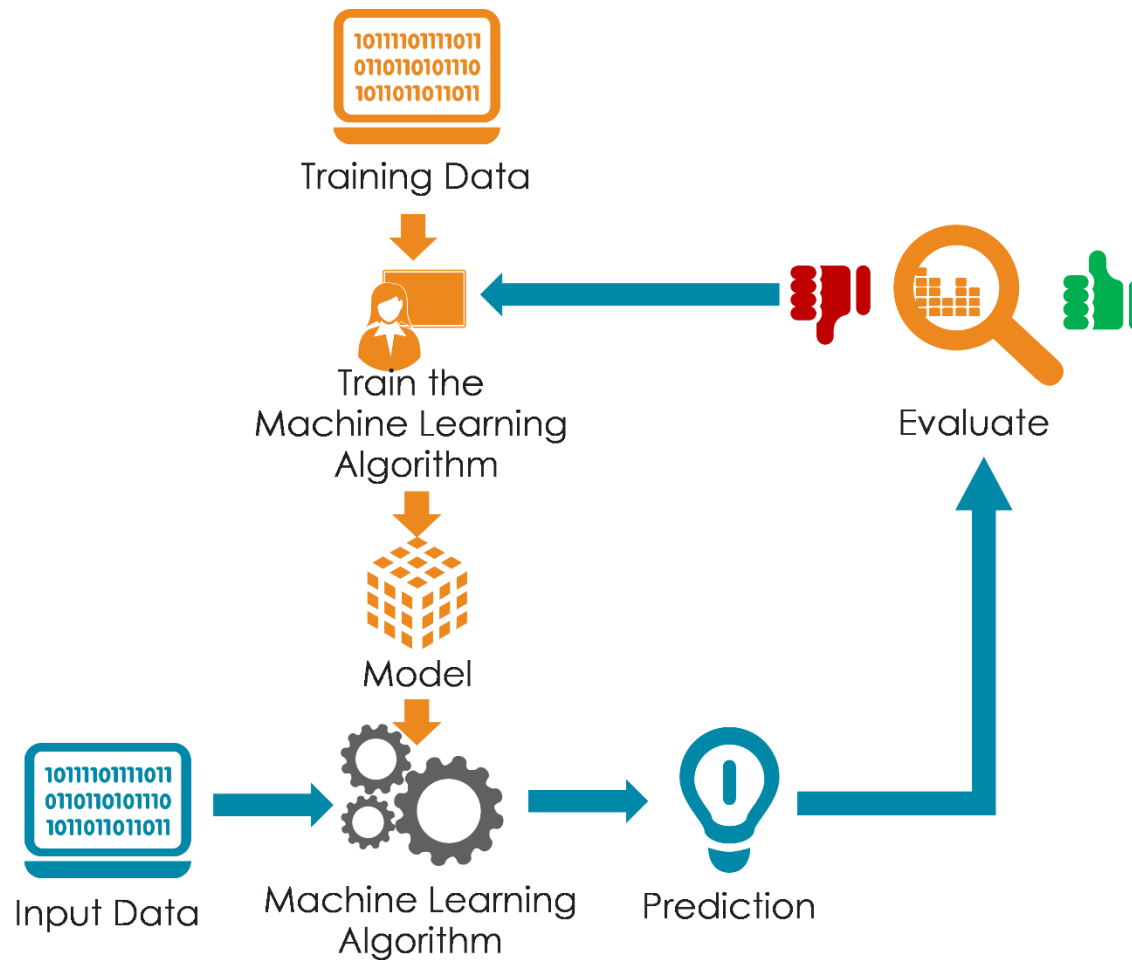
## TRADITIONAL MACHINE LEARNING



## DEEP LEARNING



# 머신러닝 과정



# 머신러닝과 딥러닝 비교

## 머신러닝과 딥러닝의 차이점

|             | 기계 학습  | 딥러닝   |
|-------------|--|---|
| 데이터의<br>존성  | 중소형 데이터 세트에서 탁월한 성능  | 큰 데이터 세트에서 뛰어난 성능                                 |
| 하드웨어<br>의존성 | 저가형 머신에서 작업하십시오.   | GPU가있는 강력한 기계가 필요합니다. DL은 상당한<br>양의 행렬 곱셈을 수행합니다. |
| 기능 공학       | 데이터를 나타내는 기능을 이해해야 함   | 데이터를 나타내는 최고의 기능을 이해할 필요가 없<br>습니다                |
| 실행 시간       | 몇 분에서 몇 시간   | 최대 몇 주. 신경망은 상당한 수의 가중치를 계산해<br>야합니다.             |
| 통역 성        | 일부 알고리즘은 해석하기 쉽고 (물류, 의사 결정 트리) 일부<br>는 거의 불가능합니다 (SVM, XGBoost) | 불가능한 어려움  |



# 딥러닝 개요

- **심층학습이라고도 부르는 딥러닝(deep learning)**
  - 인공지능망(ANN: Artificial Neural Network) 사용
    - 인간의 신경세포인 뉴런(neuron)을 모방하여 만든 가상의 신경으로 뇌와 유사한 방식으로 입력되는 정보를 학습하고 판별하는 신경 모델
  - 다양한 데이터에서 다중 계층인 심층신경망(deep neural network)을 사용
    - 학습 성능을 높이는 고유 특징들만 스스로 추출하여 학습하는 알고리즘
    - 입력값에 대해 여러 단계의 심층신경망을 거쳐 자율적으로 사고 및 결론 도출

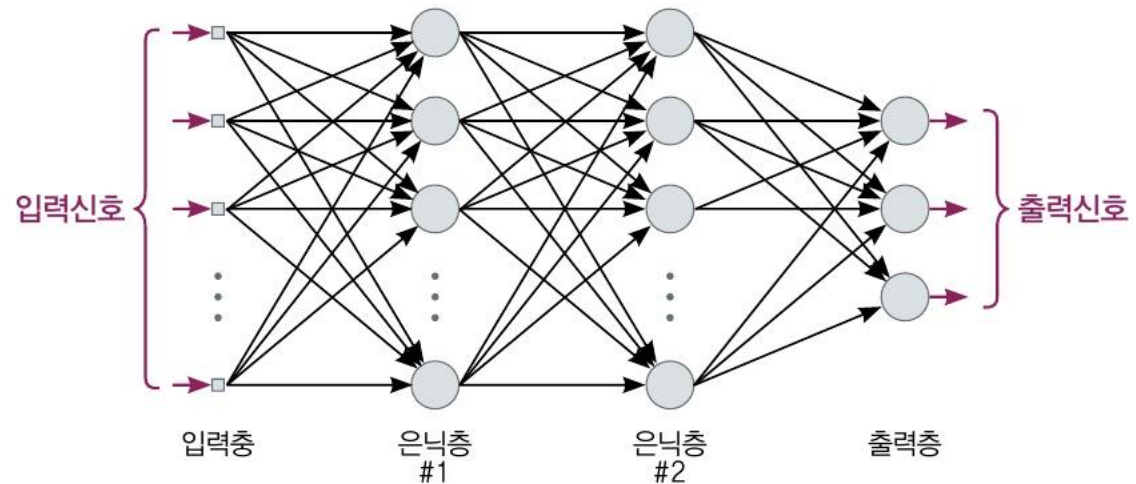
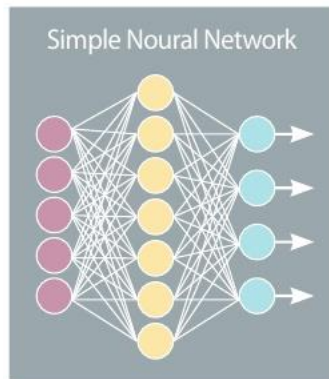


그림 10.28 ▶ 단일계층과 딥러닝의 다중계층 신경망

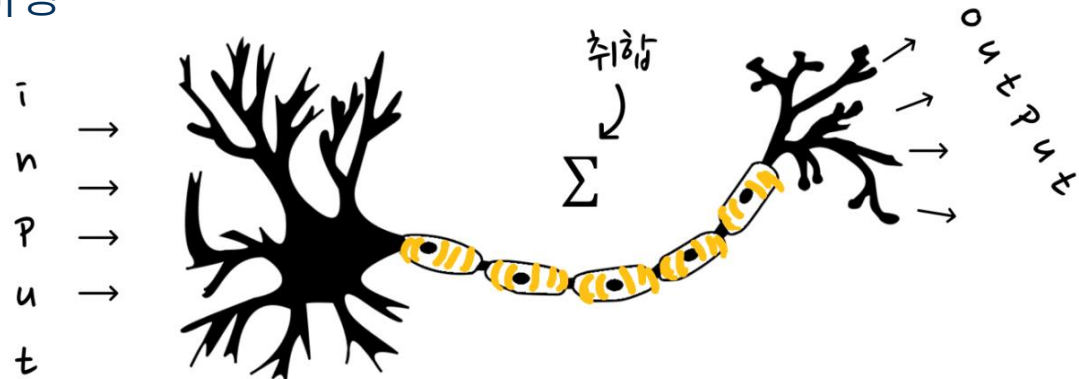
# 인공신경망에서 시작

## • 세계 최초의 인공신경망을 제안

- 1957년 심리학자인 프랭크 로젠블랫(Frank Rosenblatt) 코넬대 교수
- 퍼셉트론(perceptron)
  - 다수의 신호(input)를 입력 받아서 하나의 신호(output)를 출력
  - 입력층(input layer)과 출력층(output layer),
  - 중간층의 은닉층(hidden layer)인 여러 개의 층으로 연결하여 하나의 신경망을 구성
- 신경망에서는 방대한 양의 데이터를 신경망으로 유입
  - 데이터를 정확하게 구분하도록 시스템을 학습시켜 원하는 결과를 얻어냄

## • 현재, 발전해 여러 분야에서 활용

- 항공기나 드론의 자율비행
- 자동차의 자율주행
- 필체 인식
- 음성인식에 이용
- 언어 번역



# 인공신경망 개념

- 실제 뉴런과 인공 신경망
- 입력층과 출력층
  - 은닉층

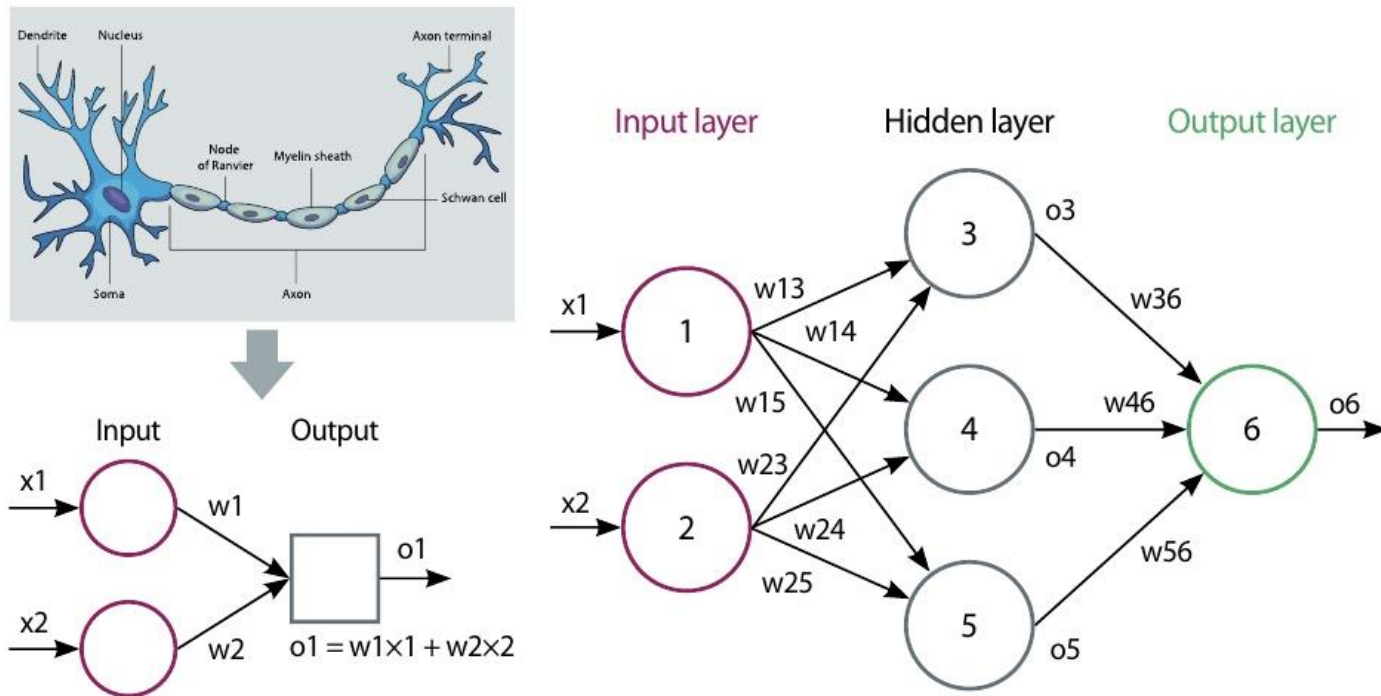


그림 10.27 ▶ 뉴런과 퍼셉트론, 신경망 개념

# 딥러닝 활용

과거 수개월 소요되었던 딥러닝이 몇 분~수시간 만에 처리가 가능

## • 이세돌을 이긴 알파고

- 다중 계층의 신경망 구조로 반복 계산에는 많은 계산 능력이 필요하고 이를 고성능의 컴퓨터로 해결
- 부동산숫점 계산에 탁월한 GPU(Graphic Processing Unit)와 분산처리가 가능한 클라우드 컴퓨팅 사용
- 스마트폰, 자동차, 스피커, 냉장고, TV 등 모든 주변 기기들에 인공지능이 더해져 지능화되고 있음

## • 다양한 분야에서 활용

- 인간과 대화하는 지능형 에이전트와 실시간 채팅이 가능한 챗봇(chatbot)
  - 음성인식과 자연어처리, 자동번역 등의 분야
  - 애플의 시리, 삼성의 빅스비, IBM 의 왓슨, 구글 나우, 마이크로소프트의 코타나, 아마존의 알렉사와 대시 등
- 얼굴을 비롯한 생체인식, 사물 인식, 자동차 번호판 인식 등 다양한 인식 분야
- X-ray 사진 판독과 각종 진단 등의 의료분야
- 드론의 자율비행이나 자동차의 자율주행 분야
- 주식이나 펀드, 환율, 일기예보 등의 예측 분야
- 음악의 작곡과 그림을 그리는 회화, 소설을 쓰는 분야 등에도 활용

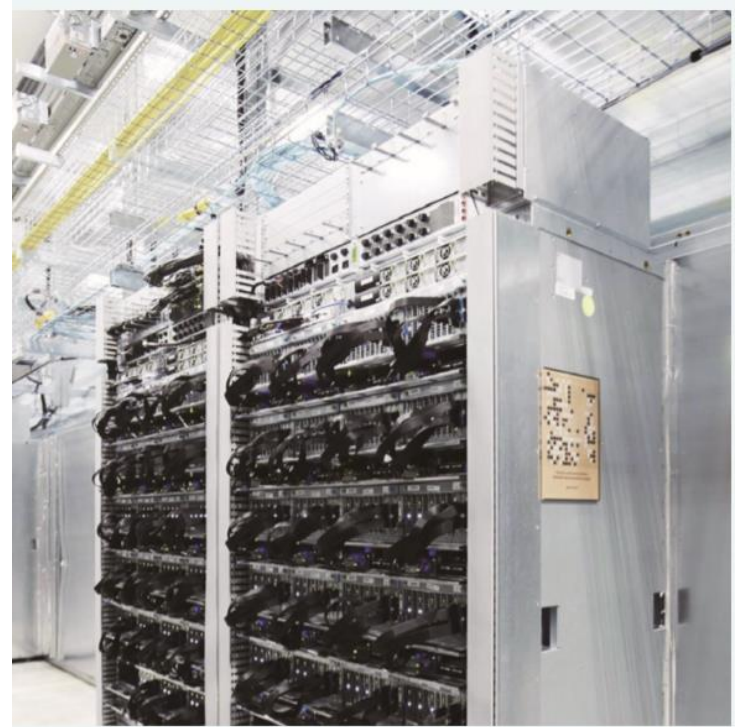
# 구글 딥마인드

- **딥마인드(DeepMind)**

- 원래 데미스 하사비스(Demis Hassabis)가 2010년 창업한 영국의 벤처 기업
- 2014년에 구글에 4억달러에 인수

- **2016년의 알파고**

- 구글의 딥마인드에서 개발한 인공지능 바둑 프로그램
  - 머신러닝의 강화학습과 신경망의 딥러닝이 적용
- 인터넷상에 있는 3000만 건의 기보 데이터를 기반으로 1차적으로 학습
  - 다시 컴퓨터끼리 대국을 시켜 경험을 반복 학습하는 방식으로 알파고의 기력을 향상
- 딥마인드의 알파고는 2017년 말에 바둑 프로그램의 역할을 종료



# 그래픽처리 장치 GPU의 인기

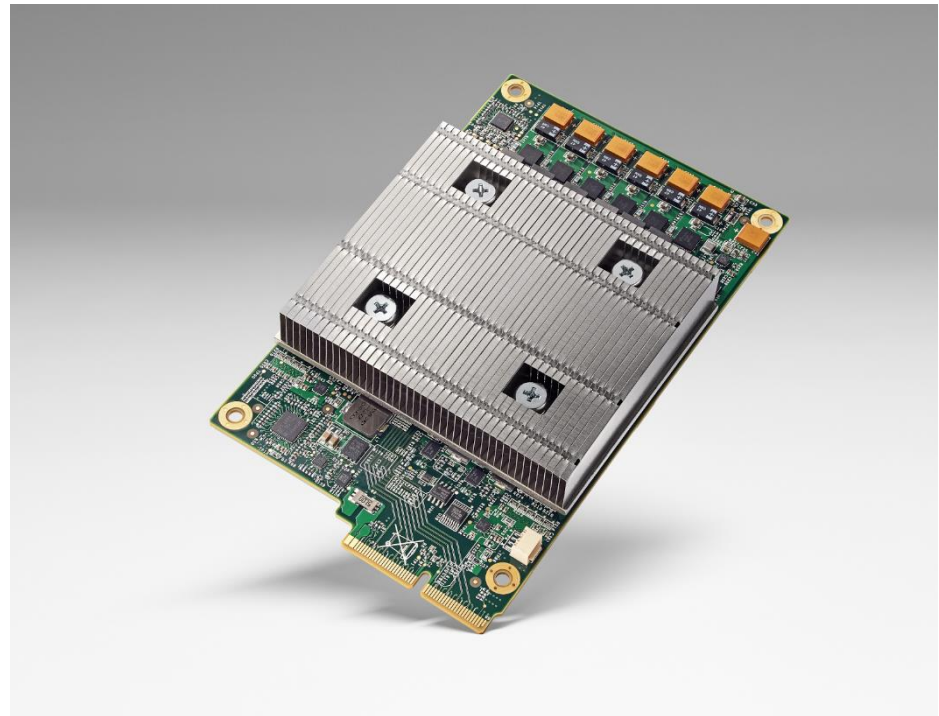
- **그래픽 처리 장치 GPU(Graphics Processing Unit)**
  - 그래픽 연산 처리를 하는 전용 프로세서
  - GPU란 용어는 1999년 엔비디아(Nvidia)에서 처음 사용
- **GPGPU(General Purpose Graphic Processing Unit)**
  - 일반 CPU 프로세서를 돕는 보조프로세서(coprocessor)로서의 GPU
  - 중앙 처리 장치(CPU)가 맡았던 응용 프로그램들의 계산에 GPU를 사용하는 기술
    - GPU 컴퓨팅이란 GPGPU를 연산에 참여
    - 고속의 병렬처리로 대량의 행렬과 벡터를 다루는 데 뛰어난 성능을 발휘
  - 딥러닝의 심층신경망에서 빅데이터를 처리하기 위해 대량의 행렬과 벡터를 사용
    - GPU 사용이 매우 효과적
  - 12개 GPU가 2,000개의 CPU와 비슷한 계산 능력



# 구글의 TPU

- 구글은 2016년

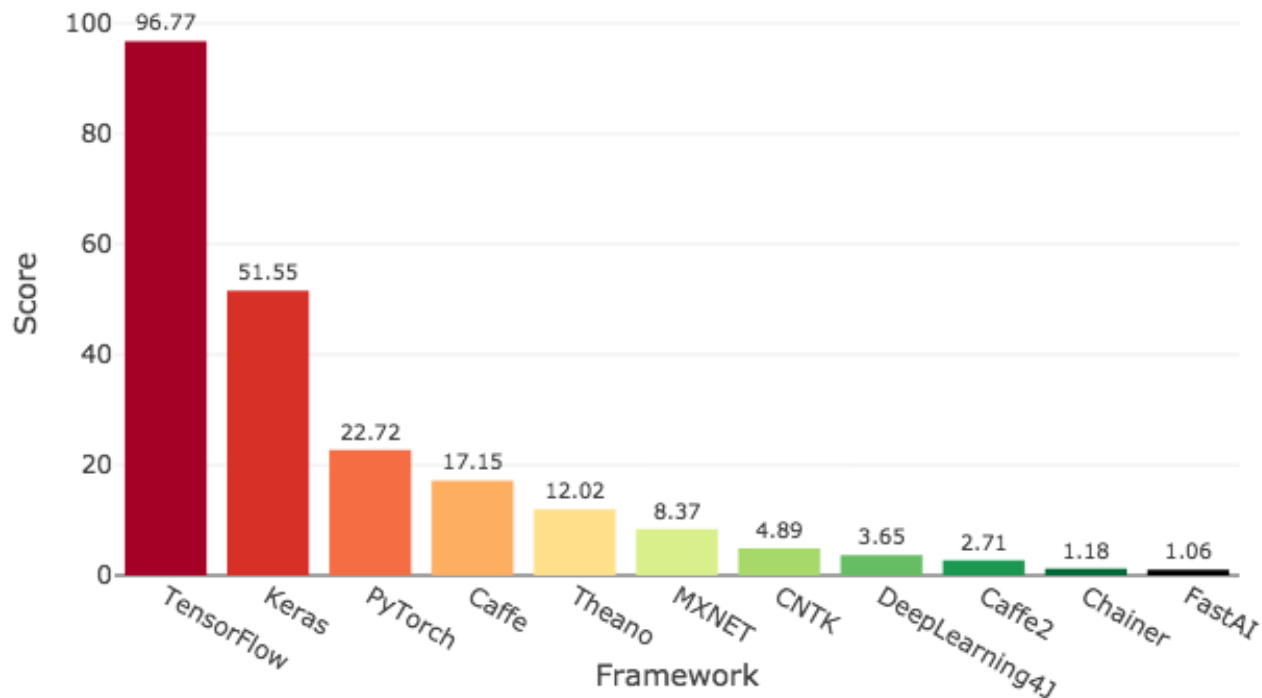
- 텐서 처리 장치(Tensor Processing Unit)를 발표
- 텐서란 벡터.행렬 을 의미
- TPU는 데이터 분석 및 딥러닝용 칩으로서 벡터.행렬연산의 병렬처리에 특화
- 텐서플로(TensorFlow)
  - TPU를 위한 소프트웨어



# 딥러닝 라이브러리(플랫폼)

- 다양한 라이브러리 활용
  - 파이썬 가장 적합한 언어

Deep Learning Framework Power Scores 2018

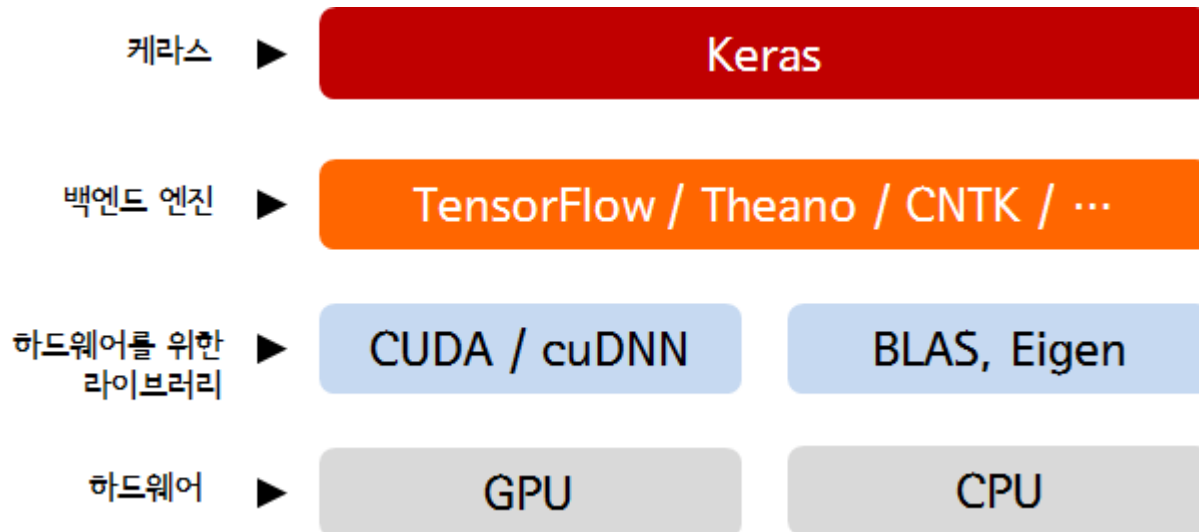




# 케라스의 특징

## • Tensorflow의 고수준 API

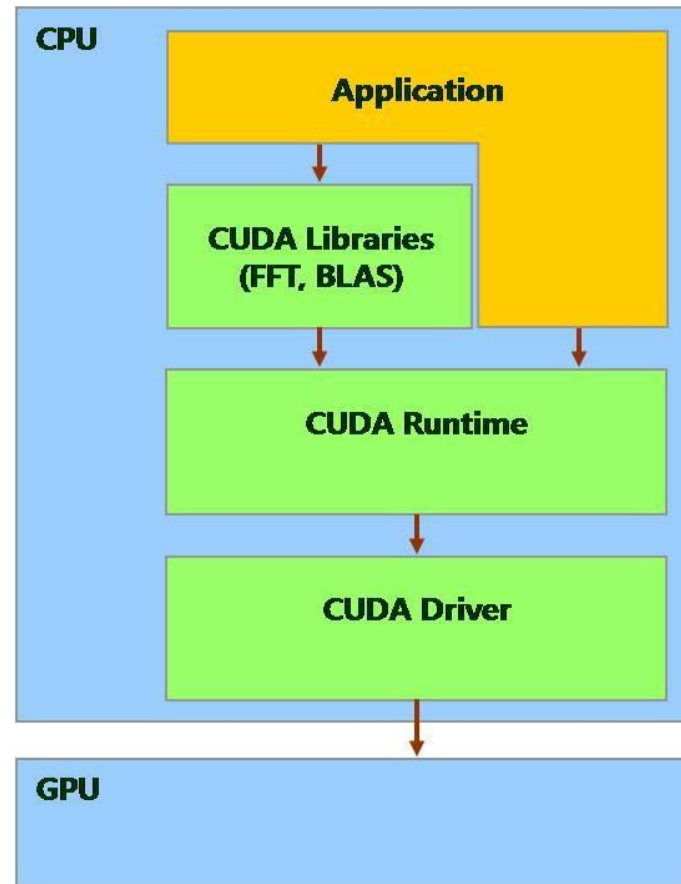
- 동일한 코드로 CPU와 GPU에서 실행 가능
- 사용하기 쉬운 API를 가지고 있어 딥러닝 모델의 프로토타입을 빠르게 생성



Basic Linear Algebra Subprograms  
 Compute Unified Device Architecture  
 CUDA Deep Neural Network library

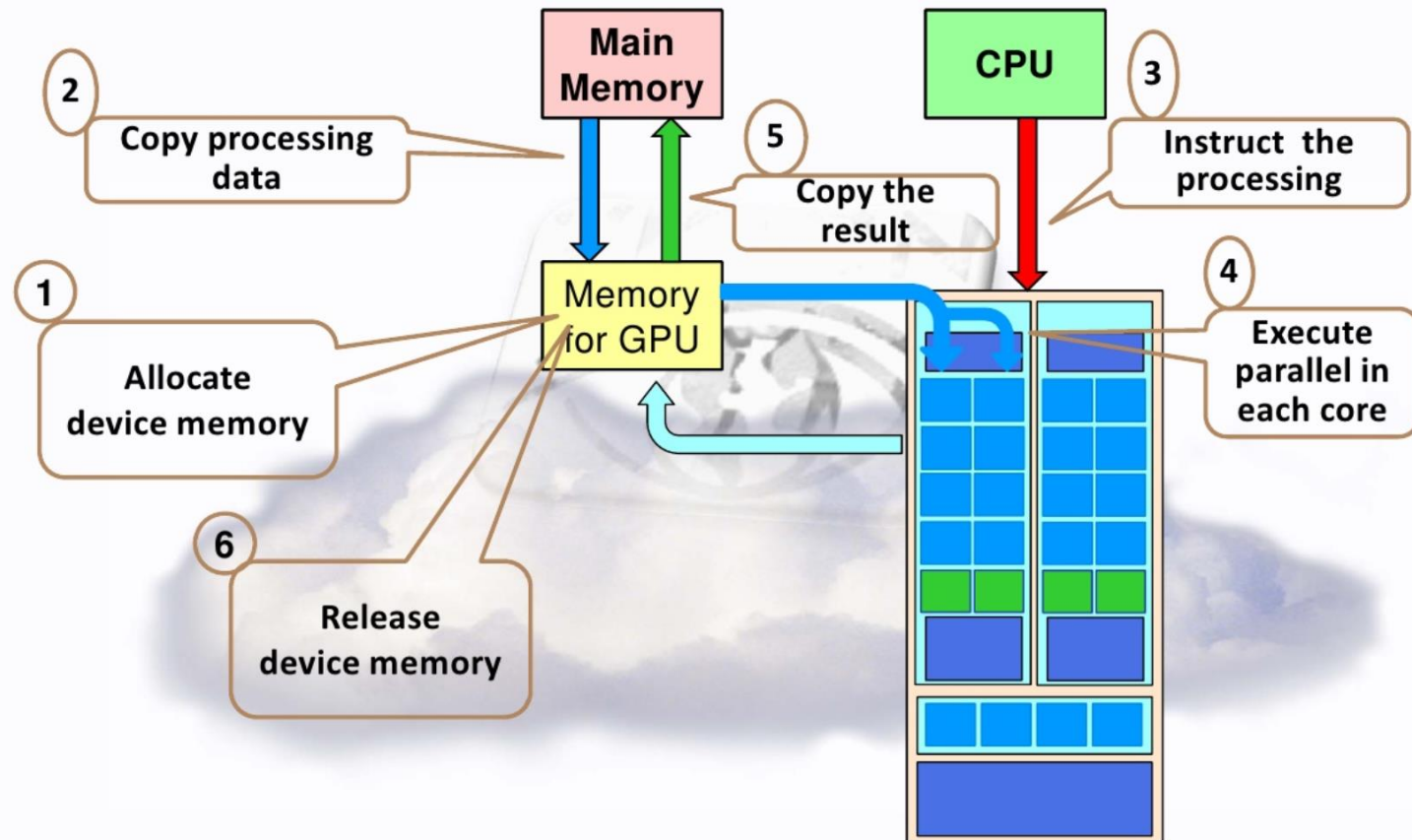
# CUDA

- Compute Unified Device Architecture의 약자



# GPU 활용

## Processing Flow on CUDA



# 딥러닝 라이브러리와 GPU

