

# Week\_01 python, 머신러닝, 딥러닝

## 강사소개

김민준 | 서울대학교 자유전공학부 휴학 중

영어교육 스타트업 0xFLOW CTO

010-5511-4898 | [lakiu@naver.com](mailto:lakiu@naver.com)

## 강의목표

python을 통해 프로그래밍을 익히고 기초적인 데이터 처리 및 코드 수정을 할 수 있다.

데이터과학과 머신러닝의 기초를 이해하고, Azure 클라우드 서비스를 통해 실습한다.

Keras와 Jupyter, AWS 등을 이용해 딥러닝 모델을 설계하고 학습시킬 수 있다.

## 강의구성

기초 python+ 머신러닝 + 딥러닝 + 약간의 수학을 이론 및 실습을 통해 학습함.

교양서적에서 얻을 수 있는 피상적인 이해를 넘어 실전적인 기술과 지식을 갖추 수 있게 함

구현한 머신러닝, 딥러닝 모델을 Github에 올려 포트폴리오로 활용할 수 있도록 지도할 예정.

## 강의내용

1주차: python, 머신러닝, 딥러닝 소개

2주차: 처음 배우는 python, 기초 통계학

3주차: 회귀분석, 분류, ROC Curve

4주차: 군집분석, 추천, 모델의 평가

5주차: 내 생애 첫 딥러닝

6주차: CNN을 이용한 이미지 인식, 분류

7주차: RNN을 이용한 텍스트, 시계열 분석

8주차: GAN을 이용한 이미지 생성

## python이란?

귀도 반 로섬이 만든 프로그래밍 언어

문법이 간결하고 직관적인 코딩이 가능해 인기

간단한 프로그램부터 인공지능, 딥러닝 등에 이르기까지 다양한 곳에서 활용되고 있음

## 프로그래밍의 기본

변수를 이용해 데이터를 계산하고 다루기

많은 변수를 다양한 자료구조를 활용해 저장하기

코드의 흐름을 조건문, 반복문 등을 통해 제어

자주 쓰이는 기능을 함수로 만들어 재사용하기

다른 사람이 만든 라이브러리 등을 가져와 쓰기

## 머신러닝이란?

일반적인 프로그램은 주어진 데이터를 프로그래머가 작성한 방식 그대로 처리함

머신러닝은 입력된 데이터와 정답 등을 비교하며 오차를 계산하고, 더 정확한 답을 내기 위해 내부의 작동방식 및 변수 등을 스스로 조정함

머신러닝에는 양질의 데이터를 다수 확보하는 것이 가장 중요하며, 최근에는 데이터마ining, 빅데이터 기술의 발전으로 데이터를 얻기 쉬워짐

## 빅데이터

3V를 만족하는 데이터로

Volume: 기존의 톨로 분석하기 힘들 정도로 많은 규모의 데이터이며

Velocity: 데이터 입출력에 요구되는 속도가 높고

Variety: 데이터의 종류가 다양하다

대표적인 예로 SNS에 올라오는 글들이 있다.



## 딥러닝이란?

머신러닝의 한 갈래로, 데이터 처리와 학습을 사람과 동물의 뇌를 인공적으로 구현한 인공신경망(Artificial Neural Network)를 사용하는 방식

머신러닝은 작동방식의 기본적인 틀과 학습규칙을 사람이 일일이 다 정해줘야 하지만 딥러닝은 충분한 데이터가 있으면 스스로 규칙을 찾고 학습함

한편 그만큼 계산에 필요한 컴퓨팅 자원이 기하급수적으로 증가하고, 개발자가 잘 모르는 문제에도 적용을 할 수 있지만 반대로 딥러닝이 왜 그 문제를 풀 수 있는지에 대한 답을 알기도 힘들다

따라서 과거에는 이러한 단점이 너무 커 딥러닝이 제대로 학습을 할 수 없었고 학계에서 거의 매장됐으나, 데이터가 확보되고 GPGPU를 사용해 그래픽카드를 계산에 사용하는 기술이 보편화되면서 딥러닝이 다시 쓰이게 될 환경이 갖춰졌다



## 딥러닝, 머신러닝 설계

### 1. 목표 설정

딥러닝, 머신러닝을 통해 해결할 문제와 목표를 설정한다

### 2. 입력과 출력 정의

어떤 데이터를 입력으로 주고, 이에 대해 어떤 출력이 나와야 할지를 정한다. 이에 따라 모형의 큰 틀이 정해지게 된다

### 3. 데이터 준비

데이터를 미리 가공하여 학습에 적합한 형태로 만든다. 수치 데이터는 단위를 통일하거나 범위를 조정하고, 문자 데이터는 전처리를 통해 가공한다

### 4. 학습 데이터와 검증 데이터의 분리

학습에 쓰이는 데이터는 모형이 정답을 알고 그에 맞게 학습을 하므로, 학습 데이터를 잘 맞춘다고 해서 모형이 정말로 학습을 잘 했다는 보장이 없다. 따라서 독립된 검증 데이터가 필요하다.

### 5. 알고리즘, 딥러닝 레이어 설계

어떤 머신러닝 알고리즘을 사용할지, 그리고 딥러닝은 어떤 레이어를 어떤 방식으로 구성할지 등등 모형의 실제 처리방식을 설계한다

### 6. 학습

설계한 모형을 학습시킨다. 데이터가 크고 모형이 복잡할수록 연산량이 기하급수적으로 늘어나므로, GPGPU를 지원하는 하드웨어와 라이브러리를 쓰거나 AWS, Azure 등의 클라우드 서비스를 쓴다

### 7. 재설계

학습을 하며 나타나는 모형의 특성을 보고 부족한 부분을 찾아 보완한다. 기준점이 되는 모형, 즉 베이스라인을 설정하고 이보다 더 나은 모형을 만들 수 있도록 재설계한다.

### 8. 검증

학습이 끝나면 검증 데이터를 통해 모형의 성능을 평가한다. 목표 설정에서 계획한 것을 달성했는지 평가하고, 어떤 부분에서 오류와 한계가 나타나는지 분석한다.

### 9. 배포 및 실사

학습이 끝난 모형을 분석, 예측에 활용하거나 혹은 웹, 장치 등에 배포한다. 학습을 시키는 것은 높은 연산능력이 필요하지만, 학습이 끝난 모형을 사용하는 것은 비교적 적은 연산능력으로도 가능하므로 스마트폰, 임베디드 하드웨어 등에도 배포가 가능하다.



## 필요한 능력과 기술

분야에 대한 지식 (Domain Knowledge)

프로그래밍 능력, 라이브러리에 대한 이해와 경험

수학 (통계학, 미적분, 선형대수학 등)

머신러닝, 딥러닝 기술에 대한 이해

기술 트렌드를 따라가고 이해할 수 있는 능력