

# Introduction to Machine Learning

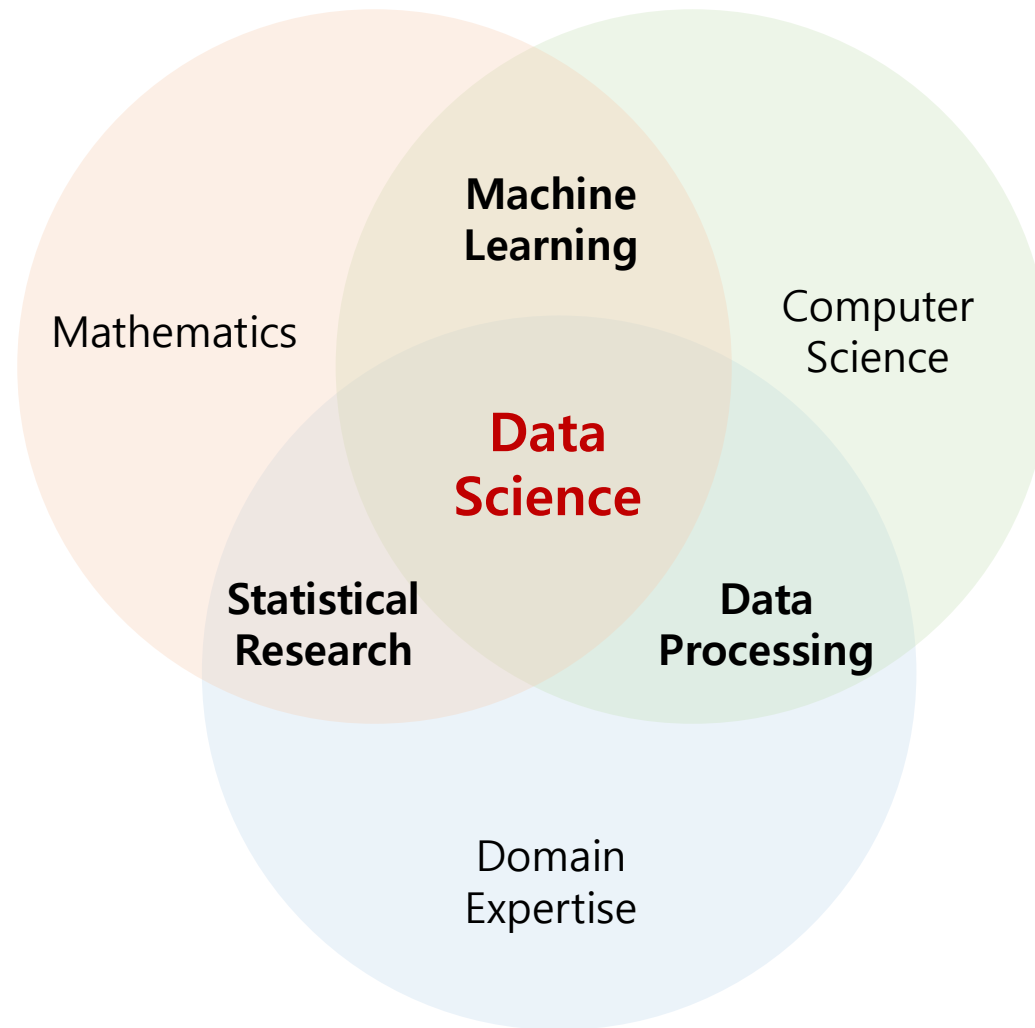
Hyunjoong Kim

[soy.lovit@gmail.com](mailto:soy.lovit@gmail.com)

[github.com/lovit](https://github.com/lovit)

# Data science

---



# Myths of data science

---

- You have to know how to **code**
- It's all about the tools
- Data Science requires a **deep understanding of statistics, math** and statistical methods
- Data Science is just a **buzzword**
- AI will replace the Data Scientist
- A Master's **degree** in Data Science = Data Scientist

# Myths of data science

---

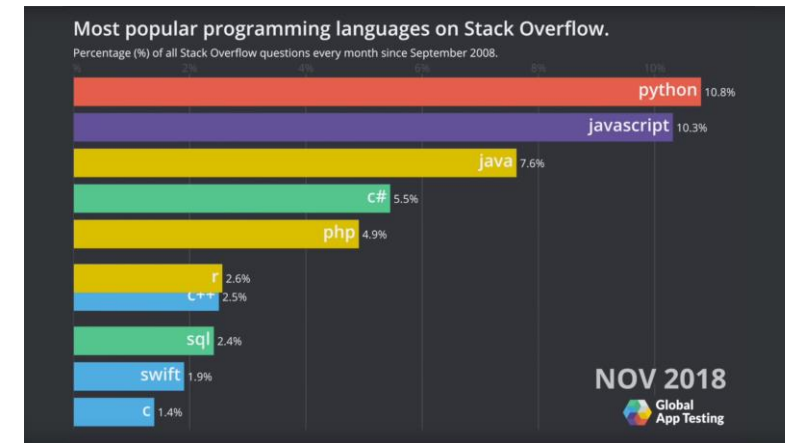
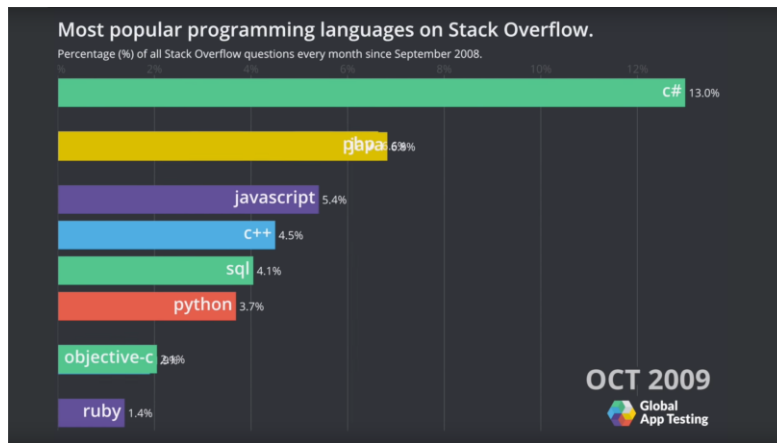
- **Complex models** are better than simple models
- Data Science requires **massive computing power**
- **More data** = more accuracy
  - Garbage in, garbage out
- They think that large organization **already has enough data** to use for analysis

# Python

---

A **programming language** is a formal **language**,  
which comprises a **set of instructions** that produce various kinds of output.  
Programming languages are used in computer programming to implement algorithms.

**Python** is an **interpreted, high-level**, general-purpose programming language



# Python

---

- Python 은 데이터 분석 / 머신러닝 분야의 발전과 함께 성장중입니다.
  - numpy, scipy, scikit-learn
  - pandas
  - theano, tensorflow, keras, pytorch
  - 하나의 언어를 중심으로 데이터 분석 / 머신러닝을 위한 기능들이 모이고 있습니다.
- Python 은 사용이 쉬운 언어입니다. 하지만 언어이기 때문에 익숙해지기 위해서는 반복적인 연습이 필요합니다.

# 수업 목표

---

- 데이터 분석을 잘 수행하기 위해서는
  - 데이터 수집 계획을 세울 수 있어야 합니다.
  - 데이터를 **정제하고 처리**할 수 있는 수준의 프로그래밍을 수행해야 합니다.
  - 머신러닝 **모델의 작동 원리를 이해**하고, **제 기능에 맞게 활용**해야 합니다.

# 수업 목표

---

- 데이터를 정제하고 처리할 수 있는 수준의 프로그래밍을 수행해야 합니다.
  - 데이터 분석의 90% 는 전처리, 10% 는 모델링이라는 농담도 있습니다.
  - 머신러닝 모델의 학습 부분보다 데이터 처리의 코드가 훨씬 깁니다.
  - **데이터 처리** → 모델 학습 → 학습 결과 탐색에 필요한 프로그래밍을 익혀봅시다.
    - Pandas, Bokeh, Seaborn, Numpy >> scikit-learn
- 모든 준비를 마치고 분석을 시작할 수는 없습니다. 필요를 느끼지 못하면 기억도 잘 나지 않습니다. **실전에서** 프로그래밍을 **시작**해봅시다.



# 수업 목표

---

- 언어에 익숙해지려면 **반복적인 연습**이 필요합니다.
  - 수업의 전반부에 Python, Bokeh, Numpy, Scikit-learn 에 대한 코드 설명이 많습니다.
  - Logistic regression 부분까지 이들의 사용법을 알아보고, 중반부 이후에 알고리즘에 집중합니다.
- 프로그래밍을 잘하려면 **좋은 코드**를 많이 읽고, **자신의 문제**를 직접 해결하는 경험을 쌓아야 합니다.

# 수업 목표

---

- 알고리즘을 학습하는 것과 “잘 이용하는 것”은 다릅니다.
  - 알고리즘도 단점들이 존재합니다.
  - 각 알고리즘의 한계를 알고, 상황에 맞게 적절히 선택합니다.
- 복잡한 모델 이전에, **기본이 되는 모델부터 정확히 이해**해야 합니다.
  - 복잡한 모델의 이해를 위한 기본기입니다.

# 수업 구성

---

- 이 수업의 대상은 파이썬을 이용하여 데이터 분석을 시작하시는 분입니다.
  - 실습 코드를 함께 살펴보는 시간을 많이 가질 예정입니다.

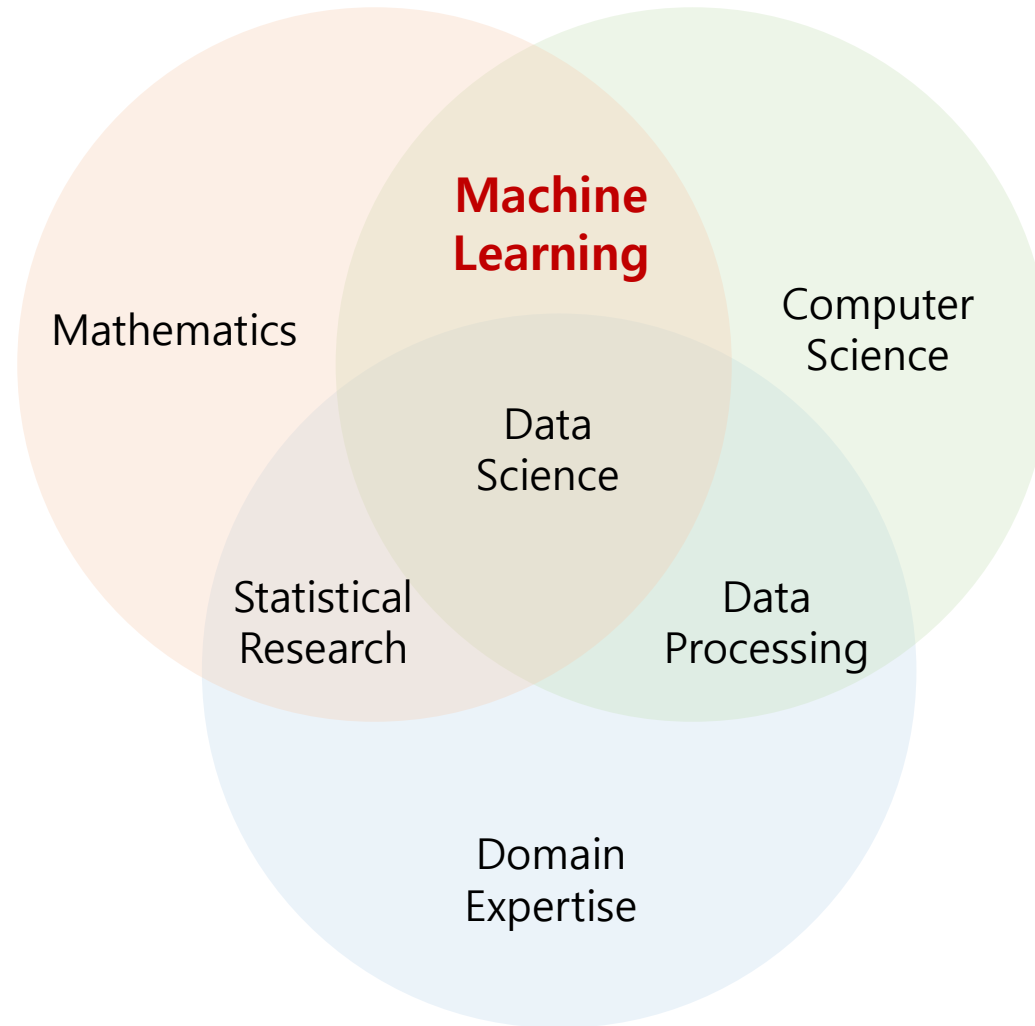
# 수업 구성

---

- 이 수업의 실습은 공용서버에서 이뤄집니다.
  - 하지만 각자의 실험 환경 (노트북, 외부서버)에서도 수업 실습을 할 수 있도록 코드를 배포하고 있습니다.
- 연습은 각자의 실험 환경에서 진행하시기를 권장합니다.
  - 자신의 분석 환경 준비도 연습이 필요합니다.

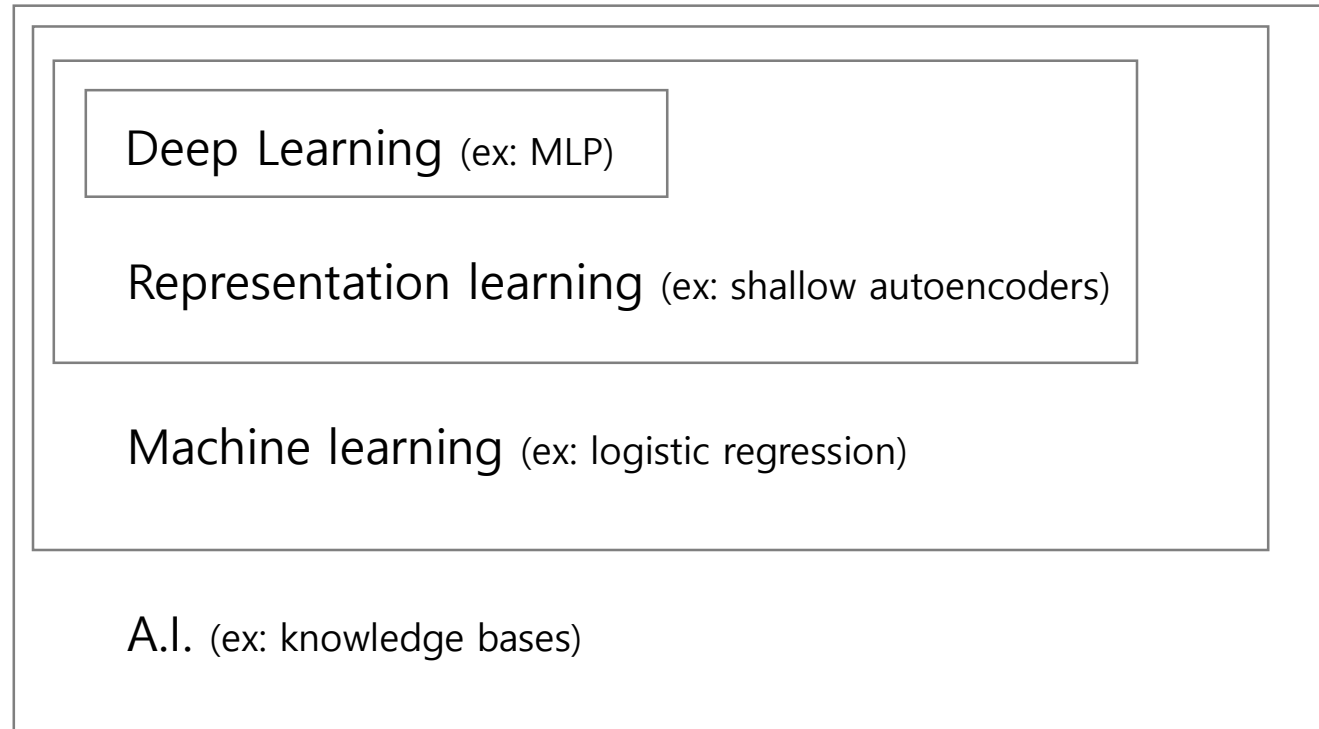
# Machine Learning

---



# Machine Learning

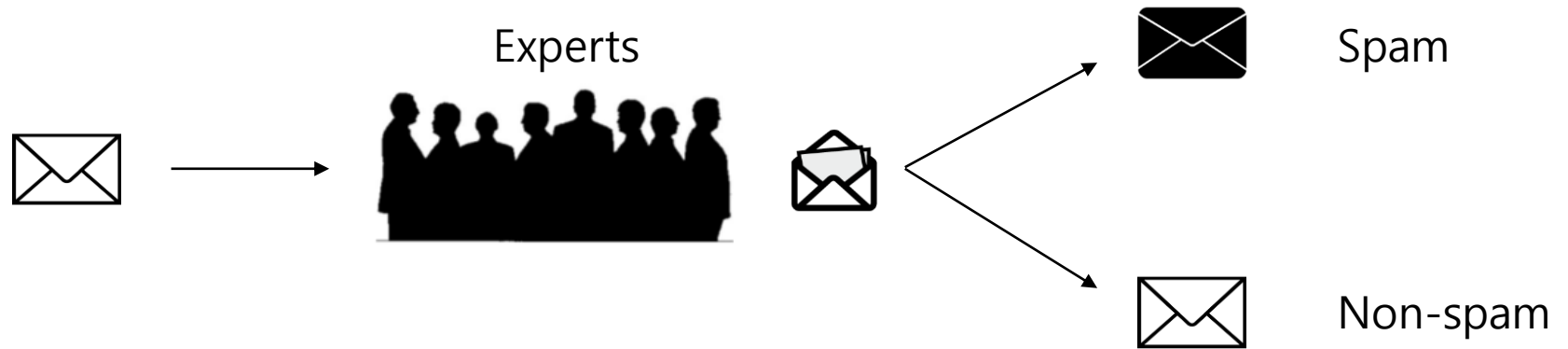
---



# Spam Filtering

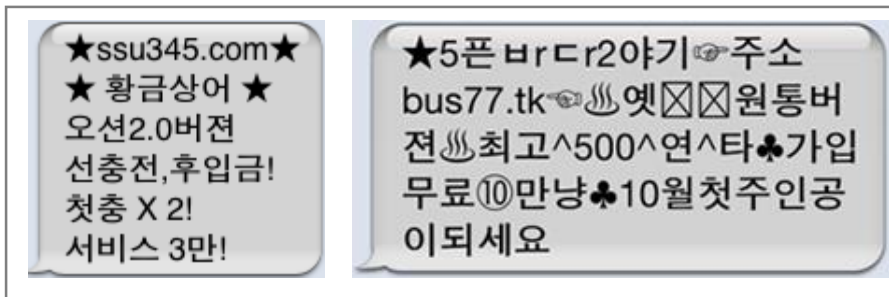
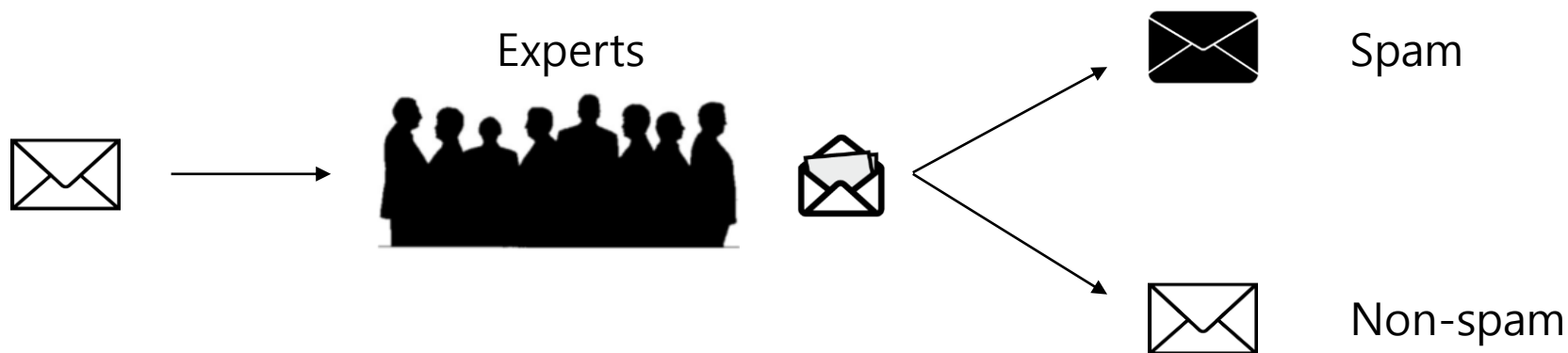
---

- 사람이 메일을 spam / non-spam 으로 분류한다면?



# Spam Filtering

- 사람이 메일을 spam / non-spam 으로 분류한다면? **분류 규칙**을 세웁니다

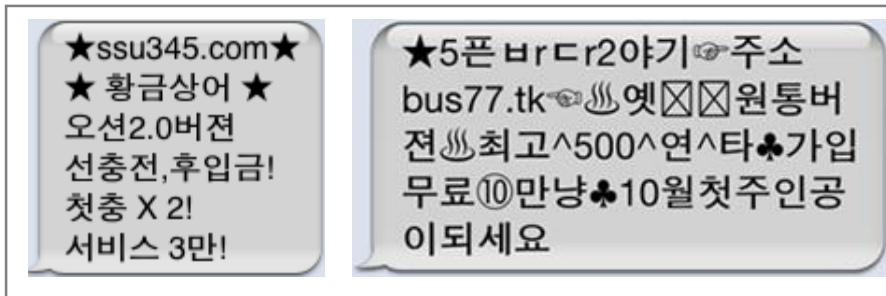
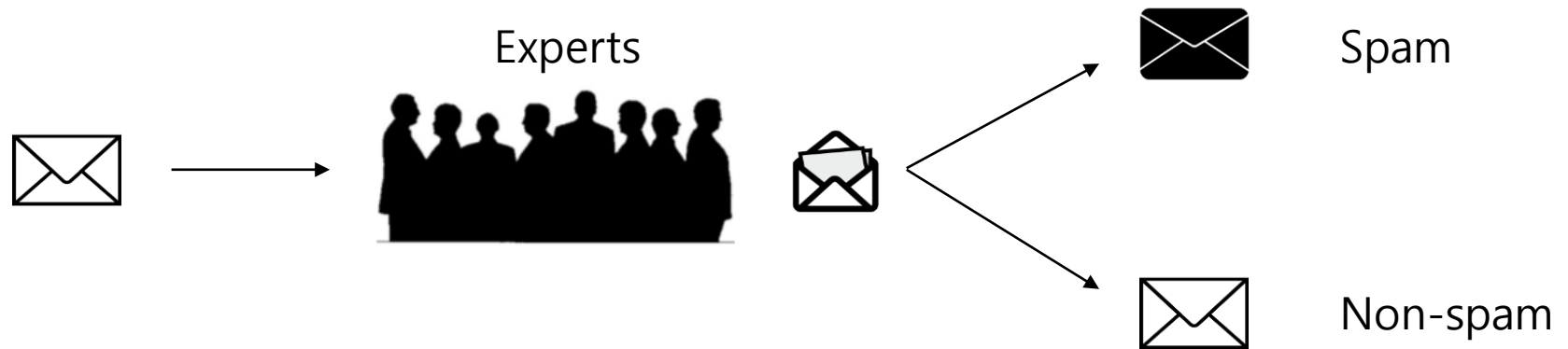


- Rules 1. 특수 문자 4개 이상  
Rules 2. "바다이야기", "오션2.0" 등의 단어 포함 유무  
Rules 3. {"충전", "입금", "무료", "가입"} 중 2개 이상의 단어가 포함



# Spam Filtering

- 사람이 메일을 spam / non-spam 으로 분류한다면?



Rules 1. 특수 문자 4개 이상

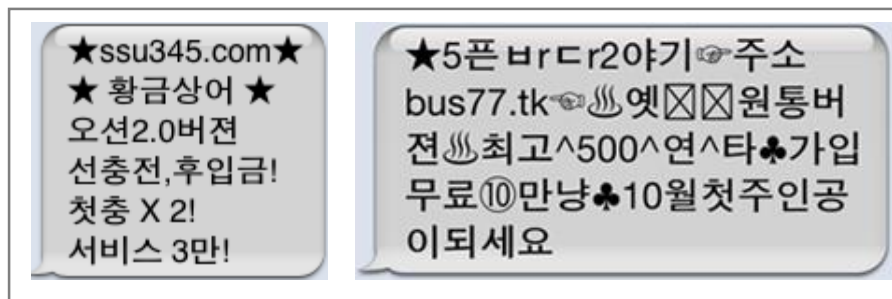
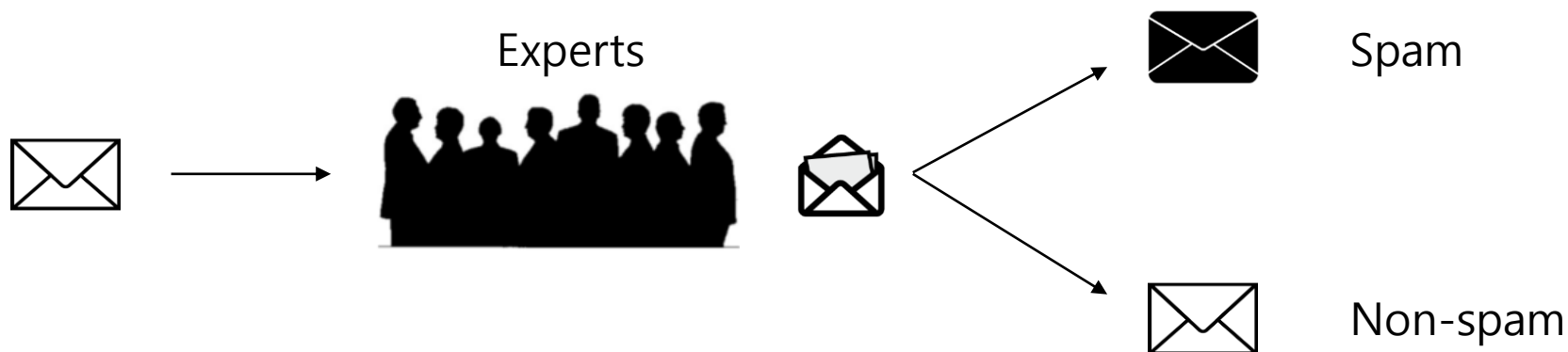
→ "수고했어!! ☺♥~♥☺(☺>-<-)و" 는 스팸으로 분류

Rules 2. "바다이야기", "오션2.0" 등의 단어 포함 유무

Rules 3. {"충전", "입금", "무료", "가입"} 중 2개 이상의 단어가 포함

# Spam Filtering

- 사람이 메일을 spam / non-spam 으로 분류한다면?



Rules 1. 특수 문자 4개 이상

Rules 2. "바다이야기", "오션2.0" 등의 단어 포함 유무  
→ "바다이야기" 로 회피

Rules 3. {"충전", "입금", "무료", "가입"} 중 2개 이상의 단어가 포함

# Spam Filtering

---

- 규칙 기반으로 스팸을 구분할 경우 (Expert system)
  - 규칙이 충분하지 않거나
  - 규칙을 유지보수 하기 어렵거나
  - 오분류가 일어나기 쉽습니다.
- 사람에 의해 관리되는 규칙 집합은 정확도가 낮은 규칙들로 이뤄진, 미완성된 규칙 집합일 가능성이 높습니다.

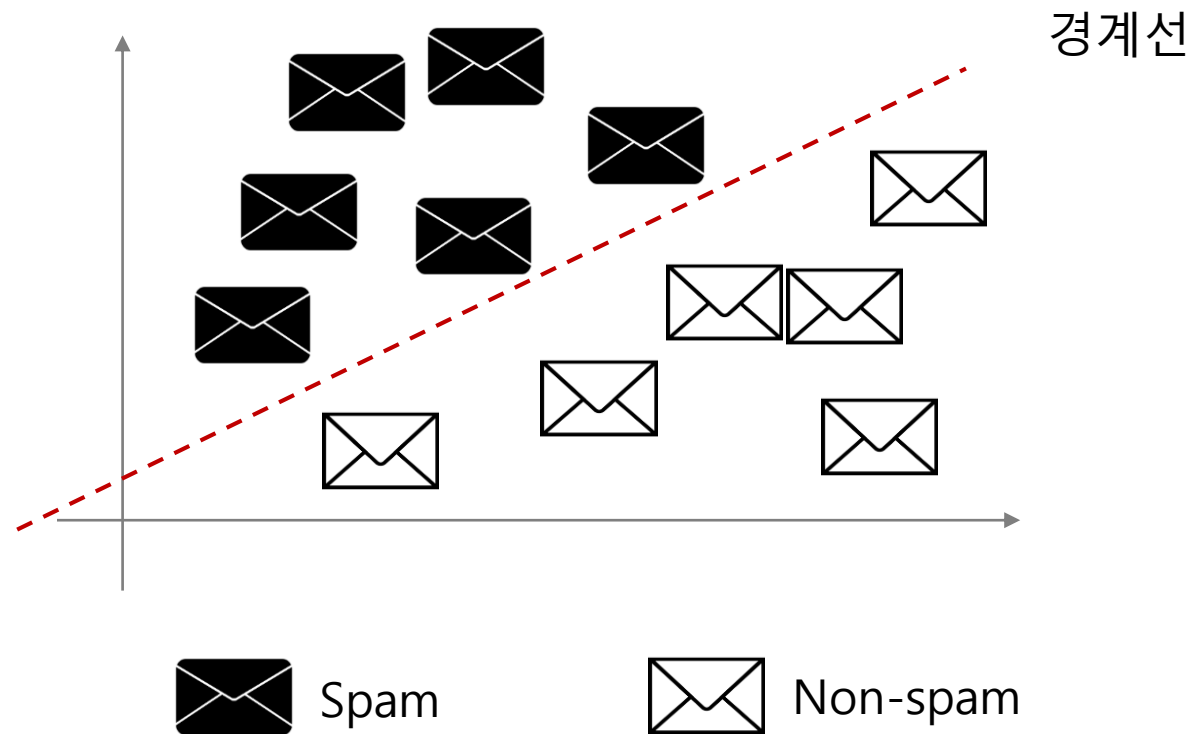
# Spam Filtering

---

- 머신러닝 알고리즘도 규칙을 기반으로 스팸을 분류합니다.
  - 데이터를 기반으로 "스팸을 구분하는데 필요한 정확하고 안정적인 규칙"들을 학습합니다.
  - 사용하는 모델이 복잡하면 학습할 수 있는 규칙의 다양성이 증가합니다.

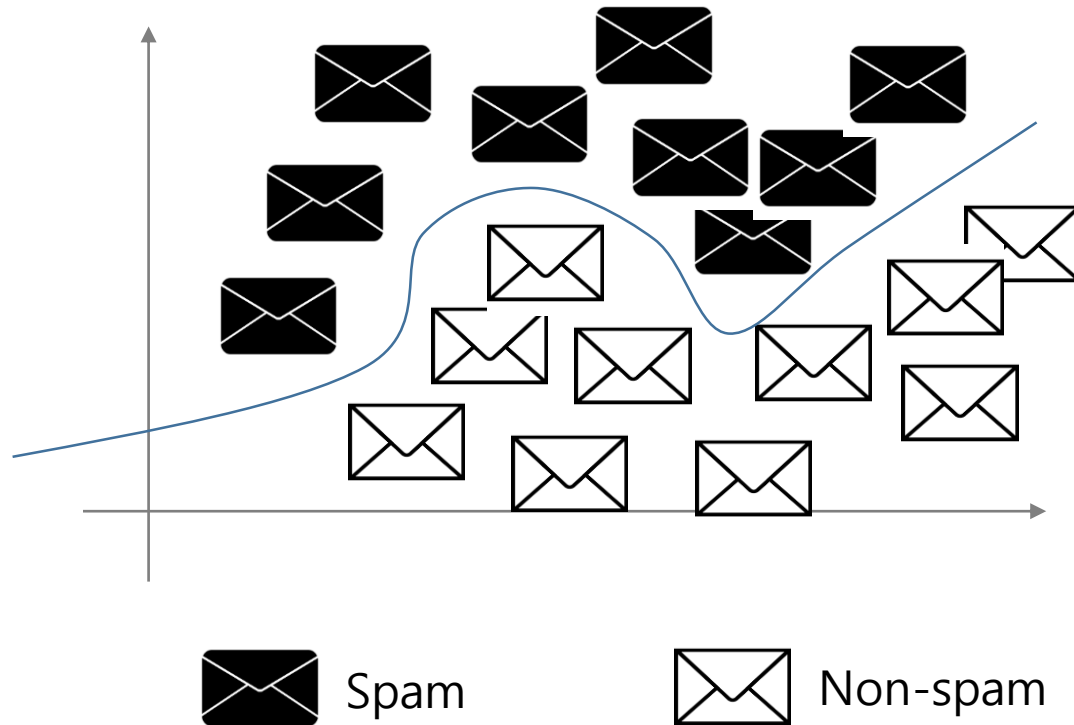
# Spam Filtering

- 많은 머신 러닝 알고리즘은 벡터 공간에서 작동하도록 설계되었습니다.
  - 데이터를 벡터로 표현한 뒤, 스팸을 구분하는 경계면을 학습합니다.  
(linear classifiers)



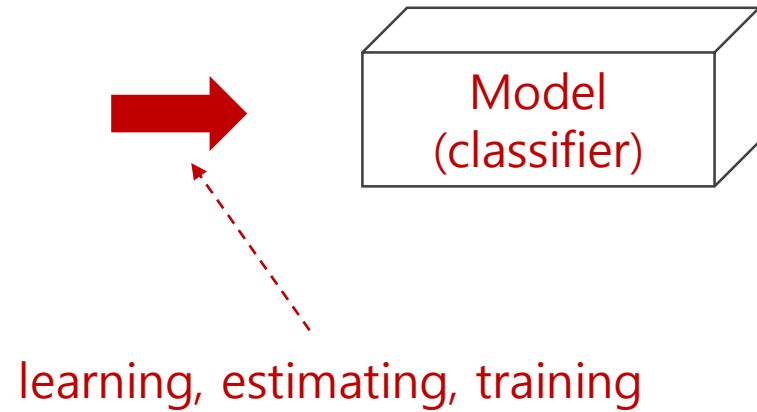
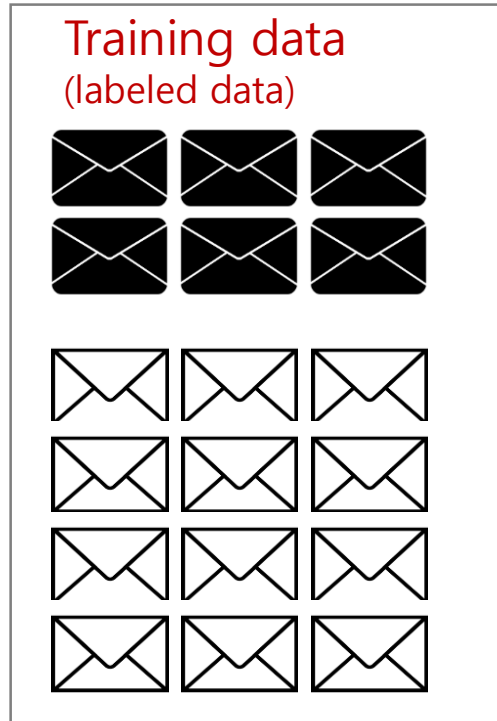
# Spam Filtering

- 많은 머신 러닝 알고리즘은 벡터 공간에서 작동하도록 설계되었습니다.
  - 데이터의 분포가 복잡하면 이를 분류할 수 있는 복잡한 경계면이 필요합니다.  
(non-linear classifiers)

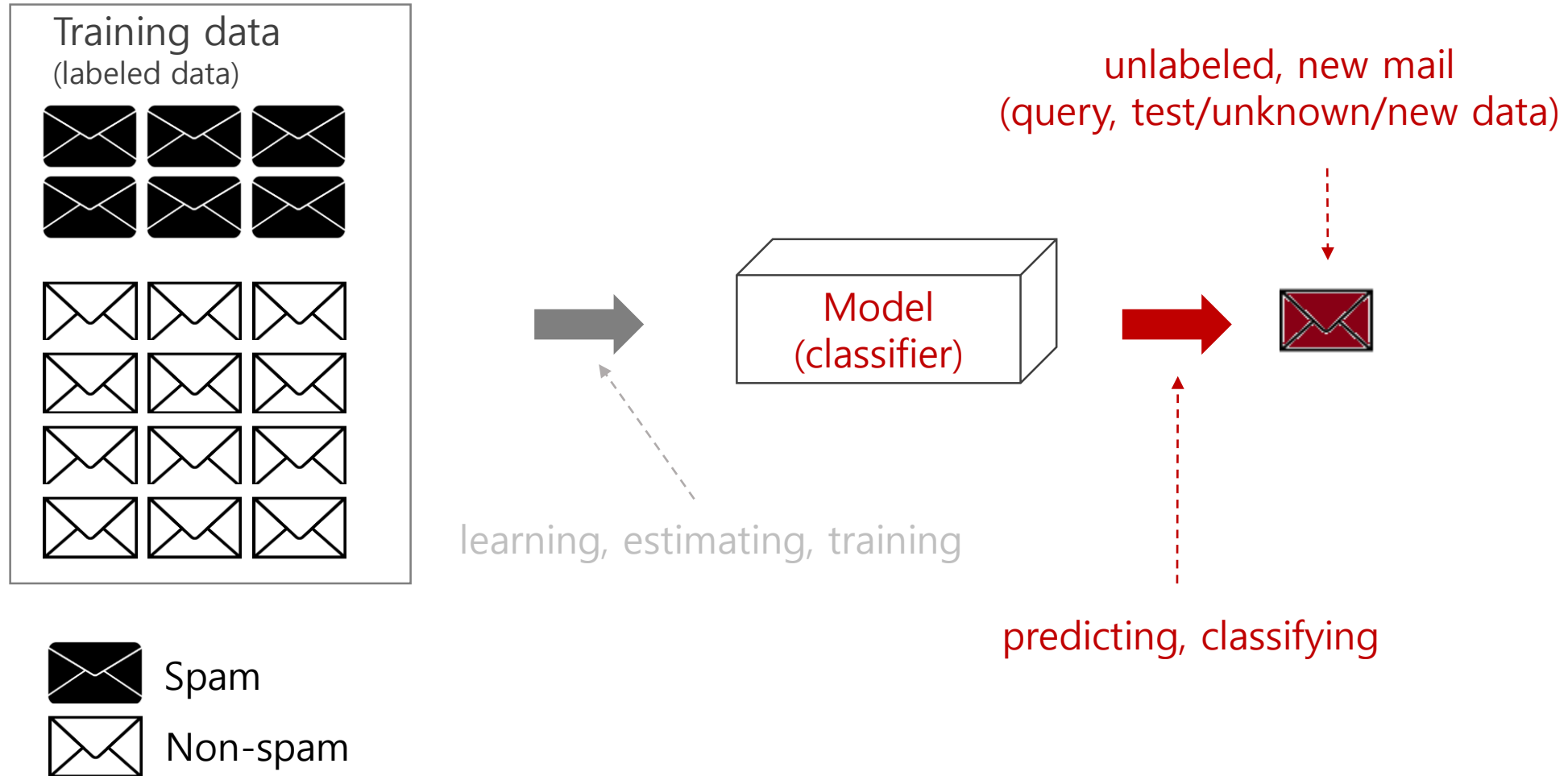


# Spam Filtering

---

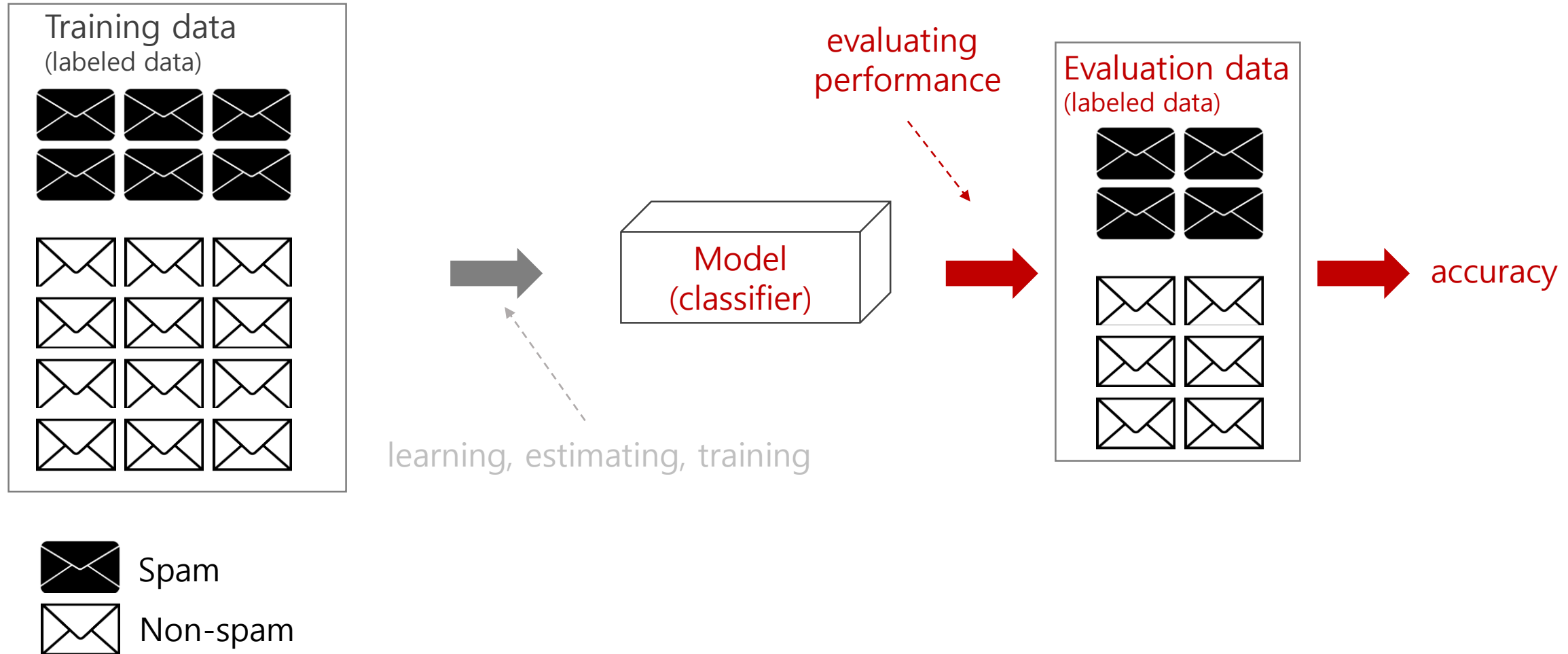


# Spam Filtering





# Spam Filtering



# 학습 데이터의 종류에 따른 머신 러닝의 분류

---

- 지도학습 (Supervised learning)
  - Labeled data 를 바탕으로  $X \rightarrow y$  의 관계를 학습하는 모델
  - (예시) 입력된 사진의 종류를 개/고양이로 분류
- 비지도학습 (Unsupervised learning)
  - Unlabeled data 를 바탕으로 패턴을 학습하는 모델
  - (예시) 벡터 공간의 거리를 바탕으로 비슷한 데이터를 하나의 그룹으로 묶음
- 강화학습 (Reinforcement learning)
  - Labeled data 가 주어지지 않았지만, 더 적합한 상황에 대한 기준이 있는 경우
  - (예시) AlphaGo, 로봇의 보행을 위한 로봇통제 방법 학습<sup>[1]</sup>

# 지도학습 (Supervised learning)

---

- Classification 은  $y$  가 명목변수일 때  $X$  로부터  $y$  를 구분합니다.
  - 제품의 불량/정상 구분
  - 메일의 스팸 구분
  - 문서의 종류 분류
- Regression 은  $y$  가 연속형 변수일 때  $X$  로부터  $y$  의 값을 예측합니다.
  - 내일의 주가지수 예측
  - 다음달 매출액 예측

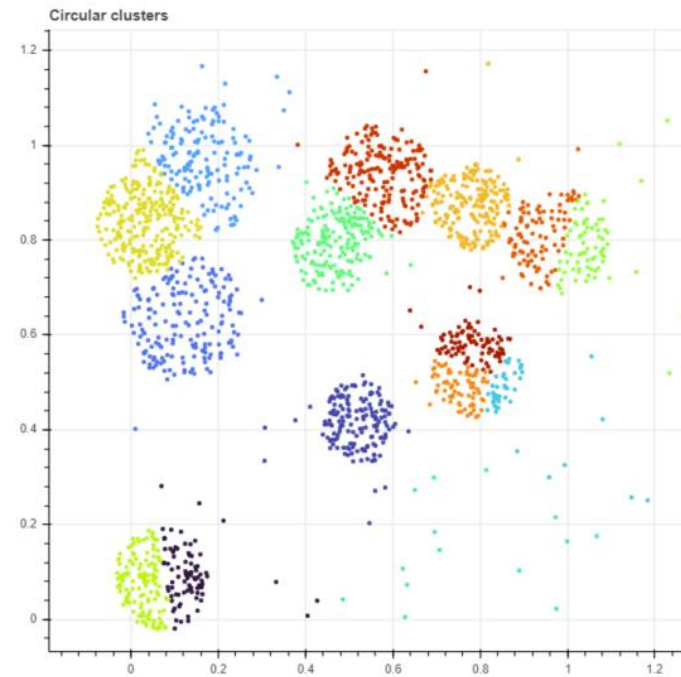
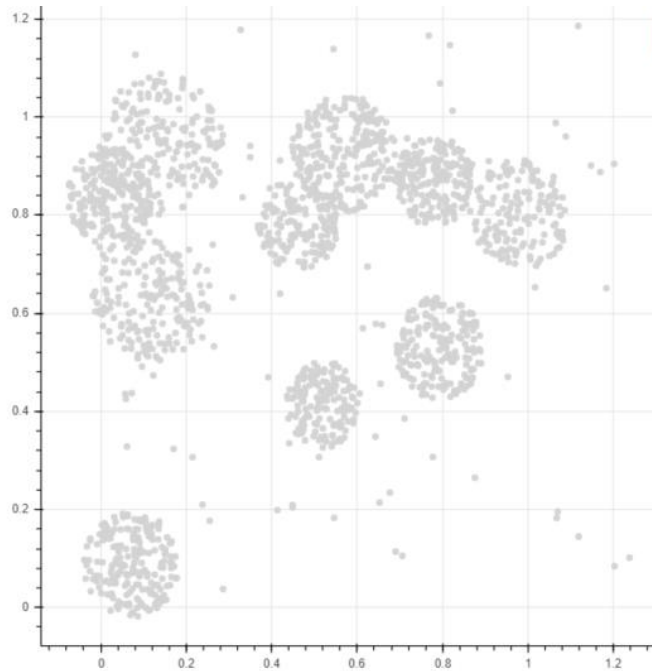
## 비지도학습 (Unsupervised learning)

---

- Clustering 은 거리가 비슷한 데이터를 하나의 그룹으로 묶습니다.
- Association rules mining 은 연관성이 높은 아이템 집합을 탐색합니다.
  - (장바구니 분석, 추천)
- Density estimation 은 데이터 공간의 밀도를 학습합니다.
  - Anomaly detection 에 이용되기도 합니다.

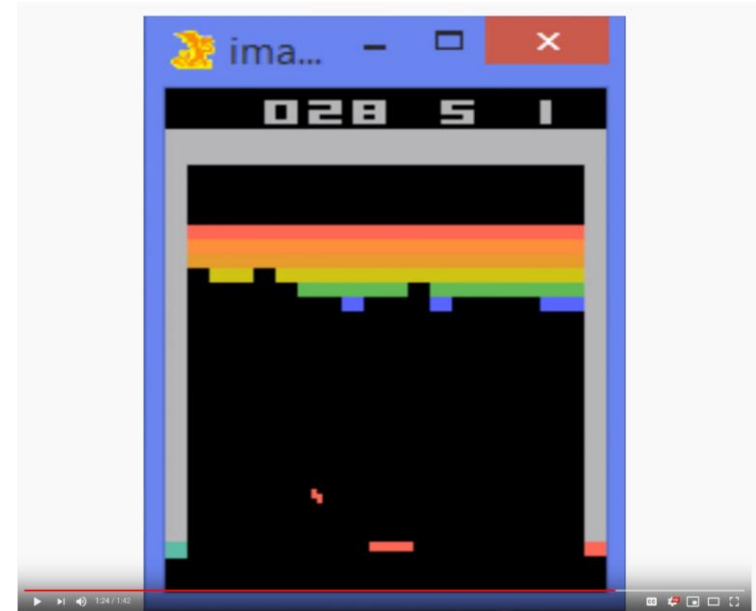
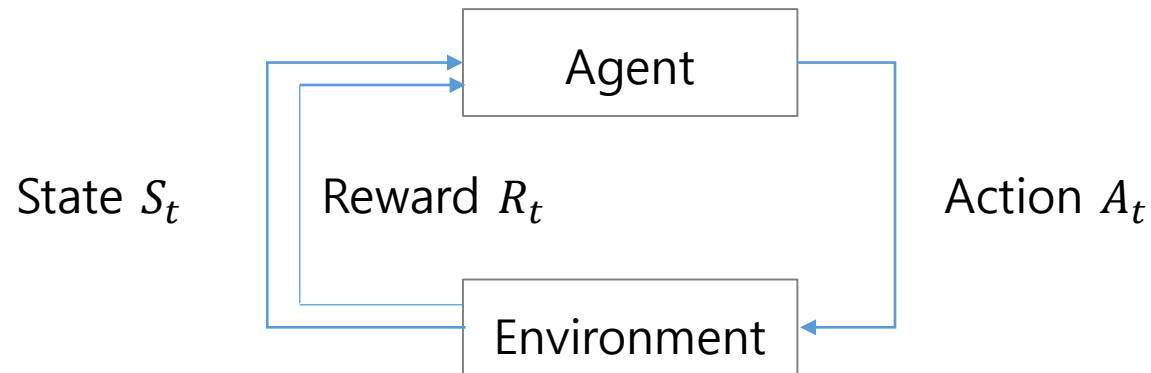
# 비지도학습 (Unsupervised learning)

- 클러스터링은 거리가 비슷한 데이터를 하나의 집합으로 묶습니다.



# 강화학습 (Reinforcement learning)

- Google DeepMind's Deep Q-learning playing Atari Breakout



# Steps in Learning Workflow

Understand the business problem  
(Defining the problem)

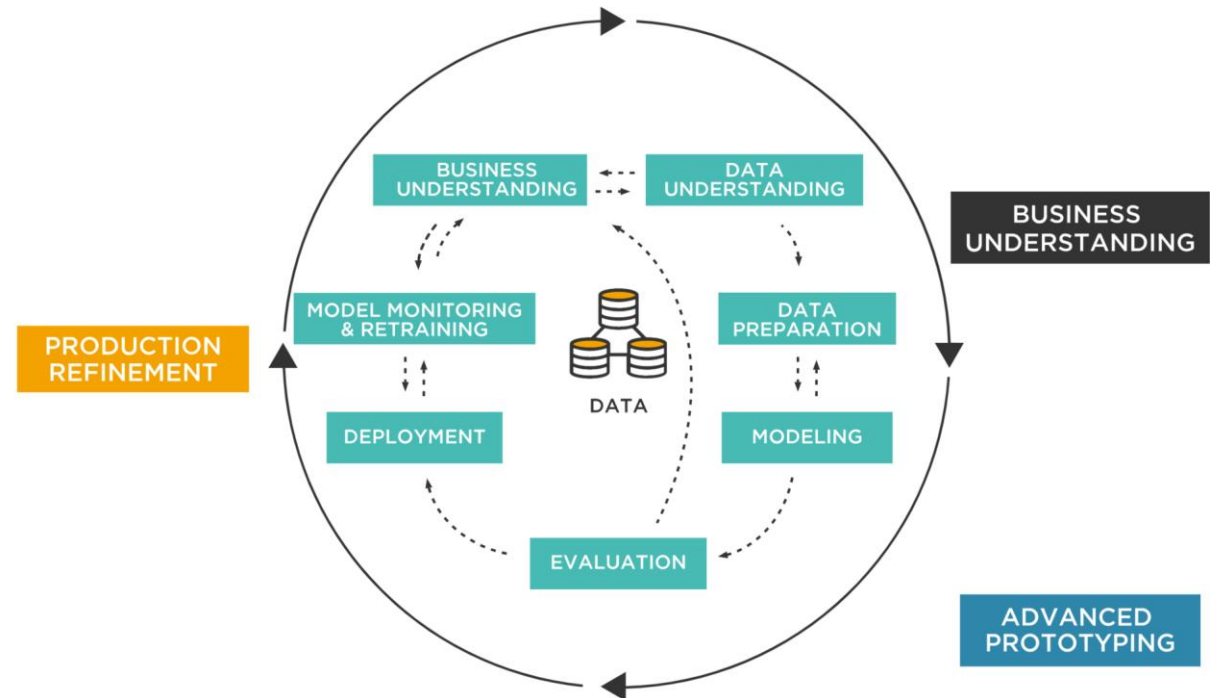
Gather and integrate the raw data

Explore, transform, clean, and  
prepare the data

Create and select models  
based on the data

Test, tune, and deploy the models

Monitor, test, refresh, and  
govern the models





# Step 1. Defining the problem

---

"machine learning" coined by Tom M. Mitchell,

"A computer program is said to **learn** from **experience E** with respect to some class of **tasks T** and **performance measure P** if its performance at tasks in T, as measured by P, improves with experience E."

학습데이터













알고리즘으로 표현된 문제

과업의 성능 평가 지표

# Step 1. Defining the problem

- 추천시스템의 Experience, Task, Performance 는 무엇일까요?

FOR YOU

 이마트몰 [가락시장] 방울토마토 1.3kg/팩 8,180 원	 이마트몰 테팔 세프 딜라이트 인덕션 프라이팬 22cm★테팔 순 결제금액 3만원이상 구매 27,160 원 38,800 원	 이마트몰 매직캔 리필 14,16,20L 7,900 원	 이마트몰 롯데 티코 밀크 초코 510ml 3,500 원	 신세계몰 내셔널지오그래픽 N194UDW940 벨루가 덕 다운 점퍼 CARBON BLACK 287,280 원 319,200 원	 새벽배송 대화 양면엠보싱 크린장갑 50매 1,395 원 1,550 원
 구매할 때 되지 않았나요?	 #자주구매 상품이 할인중입니다.	 오늘 10개 이상 팔린 인기 상품입니다.	 오늘 10개 이상 팔린 인기 상품입니다.	 오늘 100개 이상 팔린 인기 상품입니다.	 오늘 100개 이상 팔린 인기 상품입니다.

## Step 1. Defining the problem

---

- 문제를 명확히 정의할수록 데이터 수집, 모델 수립이 쉽습니다.
- 문제 중심으로 분석을 시작하는 것이 좋습니다.
  - 데이터는 있으나 무엇을 할 수 있는지 / 해야 하는지 고민하기도 합니다.
- 문제 설계는 과업의 목적과 성능 측정 기준의 명확한 정의를 포함합니다.

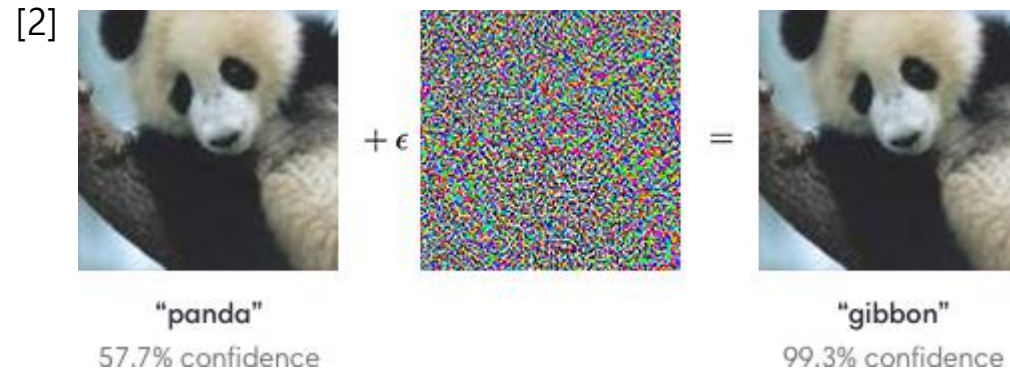
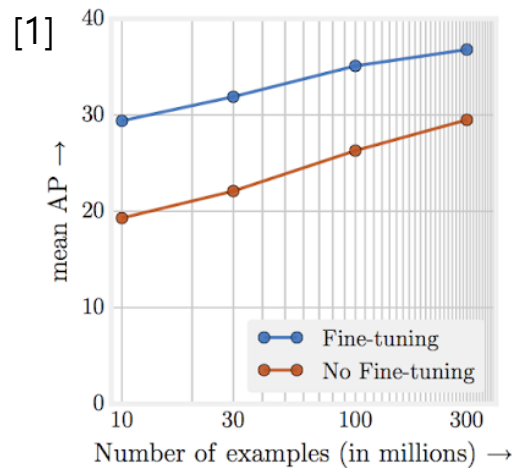
## Step 2. Gather and integrate the raw data

---

- 문제를 해결하는데 도움이 되는 데이터를 수집해야 합니다.
  - 추천시스템은 고객의 구매내역만 있어도 문제 설계가 가능합니다.
  - 스팸 분류를 위해서는 반드시 메일의 스팸 레이블이 필요합니다.  
메일만 모아서는 분류 모델을 학습할 수 없습니다.

## Step 2. Gather and integrate the raw data

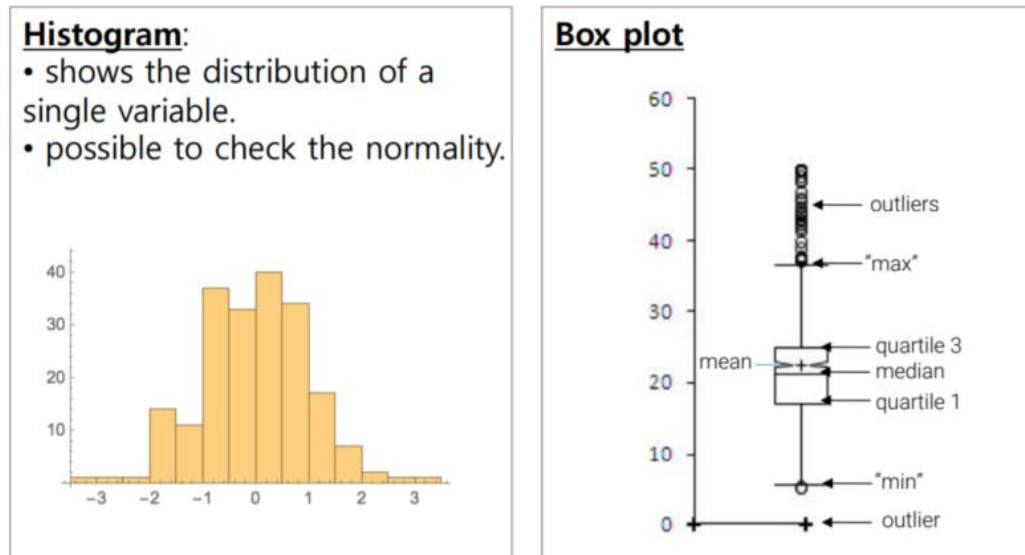
- 양질의 데이터가 더 많이 이용될 경우 성능은 증가할 수 있습니다.
  - Garbage in, garbage out. 저질의 데이터는 성능을 저하할 수도 있습니다.
  - 심지어 머신러닝 모델의 공격도 가능합니다.



## Step 3. Explore, transform, clean, and prepare the data

---

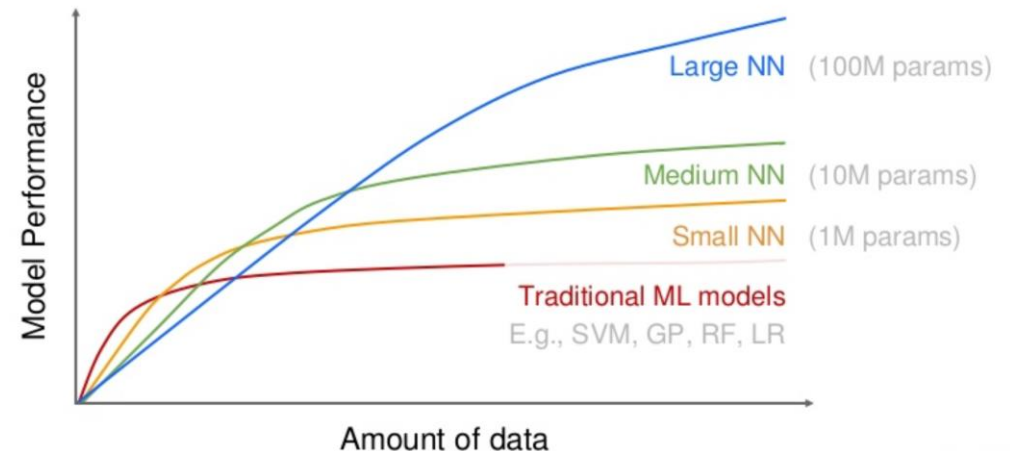
- 수집된 데이터에는 여러 종류의 노이즈가 포함될 수 있습니다.
  - Outliers, missing values 는 모델의 학습을 방해할 수 있습니다.
  - Garbage in, garbage out, again
  - 주로 전처리 90% 는 이 과정을 의미합니다.



## Step 4. Create and select models based on the data

- 문제와 데이터에 적합한 모델을 선택해야 합니다.
  - 간단한 문제 해결을 위하여 복잡한 모델을 이용할 경우, 오히려 과적합이 발생하기도 합니다.
  - 이를 위해서도 성능 평가 기준의 설계가 중요합니다.

- Better Representation Learning Helps.
- Capacity is Crucial.



## Step 5, 6. Deploy & monitoring

---

- 예상한 수준의 성능이 보이는지 평가를 한 뒤에 모델을 배포합니다.
- 배포된 뒤에는 예상한 성능이 유지되는지 모니터링 합니다.
  - 모델이 적용될 데이터가 변한다면 모델은 새로운 패턴을 제대로 인식하지 못할 수 있습니다. 모델의 업데이트가 필요한 시점이 있습니다.



Defining the problem

입력된 메일을 Spam / Non-spam 으로 구분한다.

Gather and integrate  
the raw data

사용자들이 Spam 으로 구분한 메일들을 수집한다.  
정확한 구분을 위하여 일부 메일은 수작업으로 정답을 부여한다.

Explore, transform, clean, and  
prepare the data

메일의 내용, 특수 문자 개수, 특정 표현 등의 지식들을 벡터로 표현한다.

Create and select models  
based on the data

메일 분류는 특정 표현 유무가 가장 큰 힌트이므로 선형 분류모델을 이용한다.  
확률을 함께 출력하는 Logistic Regression 을 선택한다.

Test, tune, and  
deploy the models

평가데이터를 바탕으로 성능을 확인하고, 기준에 합격하면 이를 배포한다.

Monitor, test, refresh,  
and govern the models

실제 서비스에서 배포된 시스템에 대한 오류가 보고되는지 모니터링한다.

# Topics in this class

# Topics

---

## 1. Introduction & Python basic

- 머신러닝의 개념을 알아보고, 실습을 통하여 seaborn, bokeh, pandas, numpy 의 사용법을 익힙니다.
- 실습 코드는 1일에 소화할 양이 아니며, 위 네 패키지의 튜토리얼입니다. 필요한 내용들은 각 일차별 내용에 포함되어 있으니, 자세한 튜토리얼을 보고 싶을 때 발췌하면 좋습니다.

## 2. Linear Regression

## 3. Logistic Regression

- 지도학습 기법의 기본 모델인 선형회귀와 로지스틱 분류모델을 통하여 머신러닝의 기본 개념 및 scikit-learn 의 사용방법을 공부합니다.
- Python, Seaborn, Numpy 의 기본적인 사용법을 공부합니다.

## 4. Feature extraction and preprocessing

- Pandas 를 이용한 테이블 병합, 데이터 탐색을 공부합니다.
- 텍스트 데이터의 벡터화 방법을 간단히 알아봅니다.

# Topics

---

## 5. Feed forward neural network

- 뉴럴 네트워크를 통하여 비선형 모델의 원리를 알아봅니다.
- 이미지 데이터를 핸들링하는 방법도 살펴봅니다.

## 6. Support Vector Machine

## 7. Decision Tree

## 8. Tree based Ensembles

- Kaggle 에서 좋은 성능을 보여주는 Random Forest, XGBoost 등의 앙상블 기반 모델들을 알아봅니다.

## 9. Nearest Neighbor methods

- Collaborative Filtering 을 통한 추천 모델을 알아봅니다.

## 10. Clustering