강의 시작

연합 학습 이해 및 실습

홍 성 은 연구원

sungkenh@gmail.com
강원대학교 컴퓨터공학과 데이터지능연구실(AI 신약개발 지원센터 외부연구원)

목차

CONTENTS

목차

CONTENTS

이 강의를 마치면

- 1. 딥러닝과 신경망의 기초를 이해할 수 있다.
- 2. Pytorch를 사용한 딥러닝 코드를 작성할 수 있다.
- 3. 연합 학습의 프로세스를 이해하고 기본 코드를 작성할 수 있다.
- 4. 연합 학습의 가중치 융합 알고리즘을 이해하고 코드로 작성할 수 있다.
- 5. 이 강의에서 제공되는 모든 코드를 활용할 수 있다.

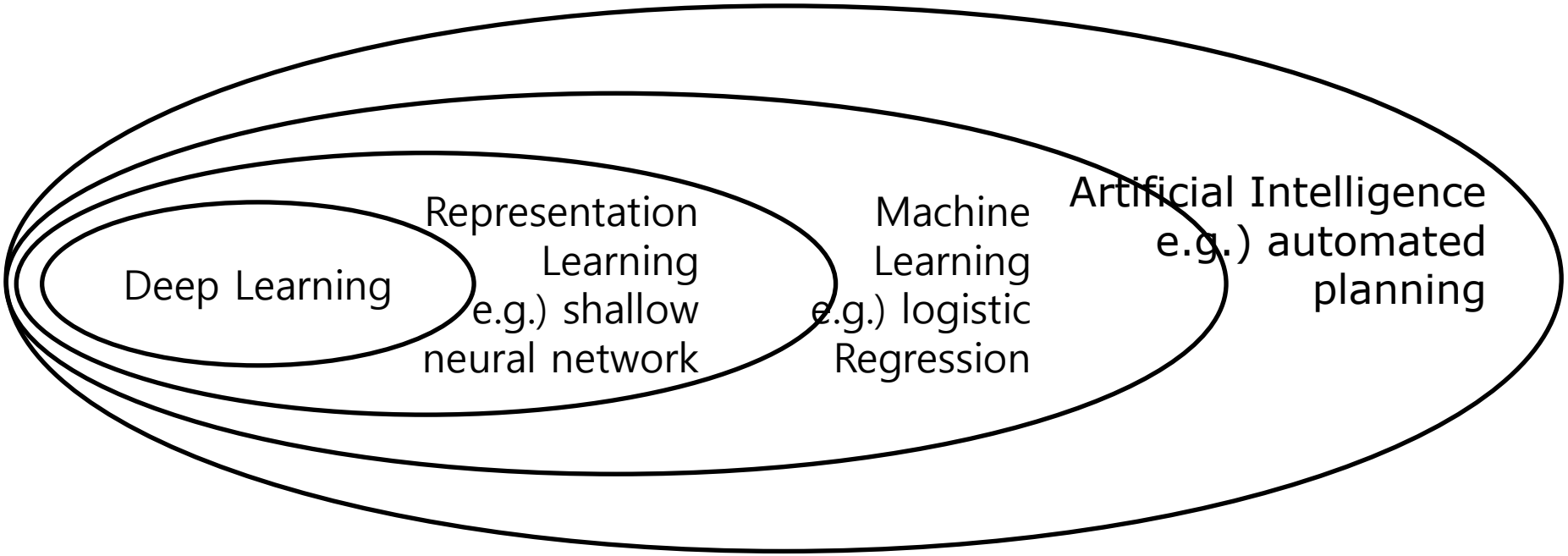
CONTENTS

Reference

- https://github.com/heartcored98/Standalone-DeepLearning
- https://github.com/OpenMined/PySyft/tree/dev/packages/syft/examples
- https://github.com/Gharibim/federated_learning_course

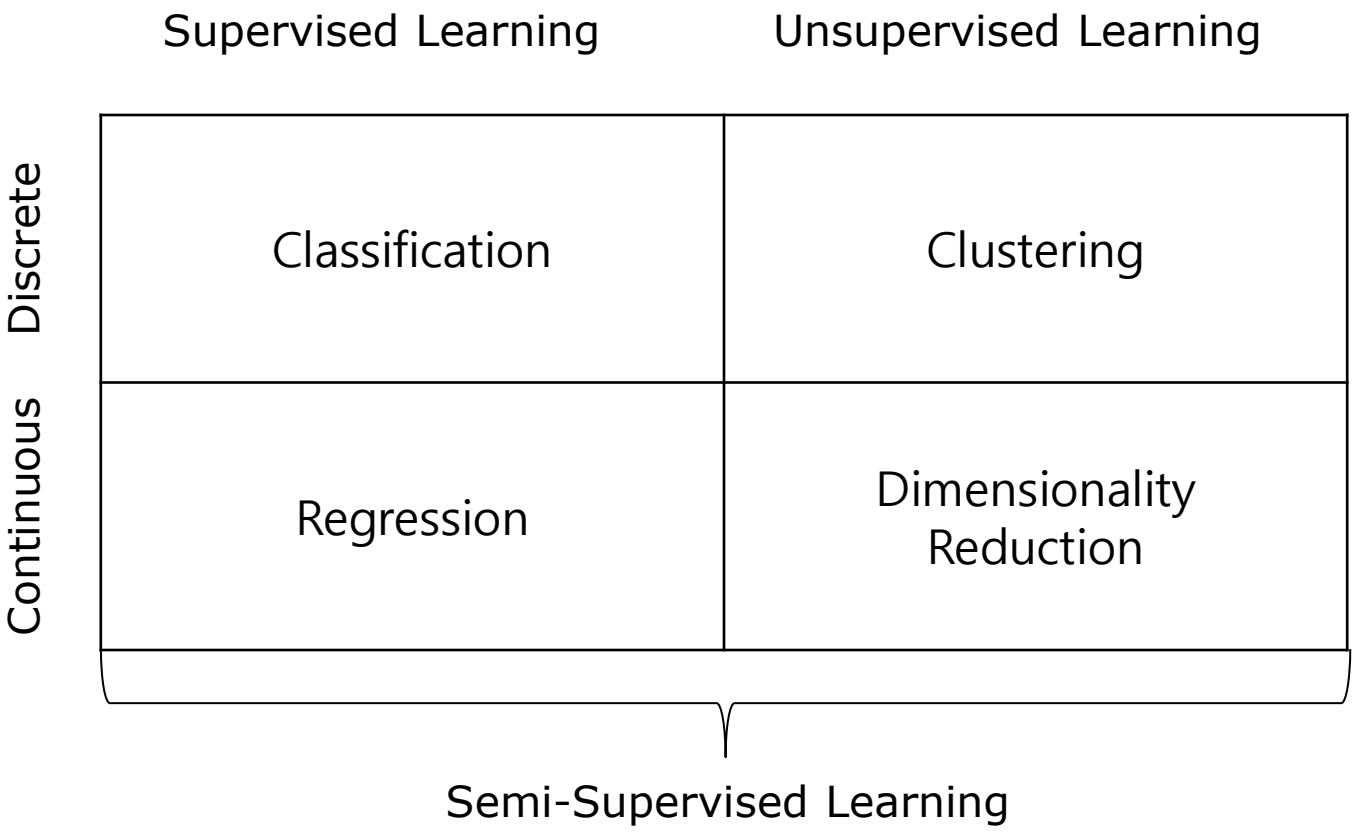
CONTENTS

딥러닝과 신경망

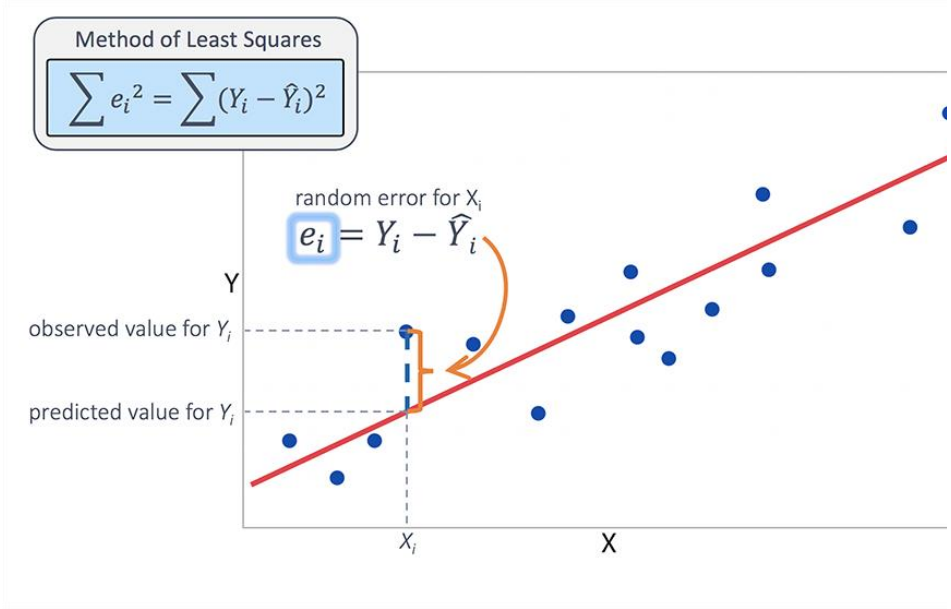

Deep Learning
Representation Learning
e.g.) shallow neural network
Machine Learning
e.g.) logistic Regression
Artificial Intelligence
e.g.) automated planning

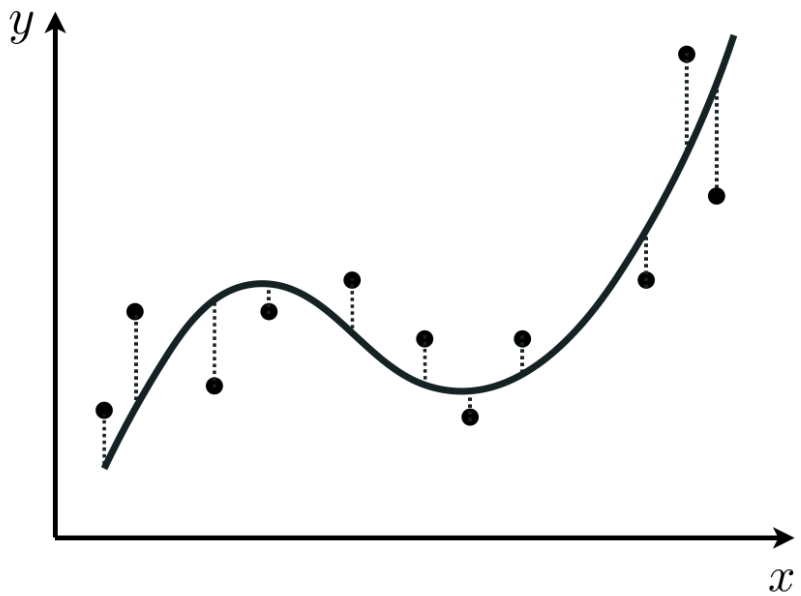
딥러닝과 신경망

- 문제 분류



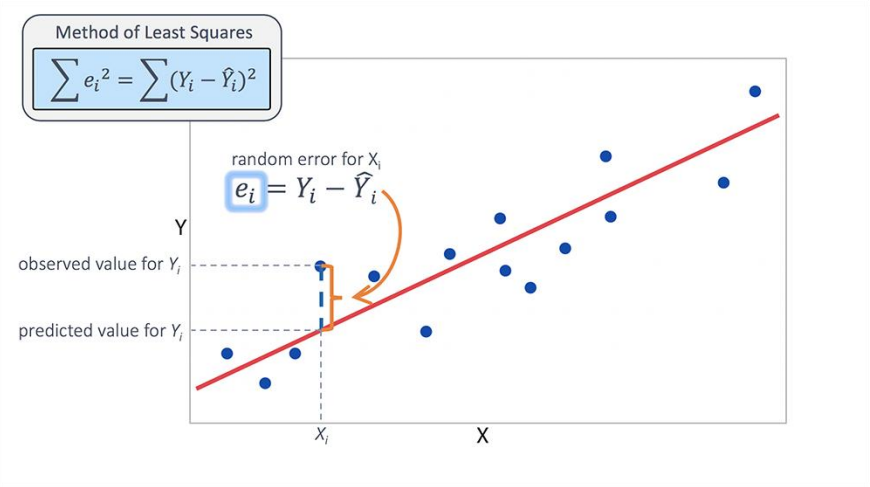
- 회귀 문제
  - 학습 데이터 x로 부터 y를 예측하는 함수 f(x)를 찾는 과정으로 x와 y는 모두 연속적인 수치 값





딥러닝과 신경망

- 회귀 문제와 손실함수
  - MAE(Mean absolute Error): 원본 값과 예측 값에 대한 절대 오류의 평균
  - MSE(Mean Squared Error): 원본 값과 예측 값에 대한 오류 제곱의 평균
  - RMSE(Root MSE): MSE의 제곱근
  - R-squared: 원본 값과 예측 값을 비교하여 회귀모델이 얼마나 잘 원본 값을 나타내는지 [0,1]



$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}|$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2$$

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2}$$

$$R^2 = 1 - \frac{\sum(y_i - \hat{y})^2}{\sum(y_i - \bar{y})^2}$$

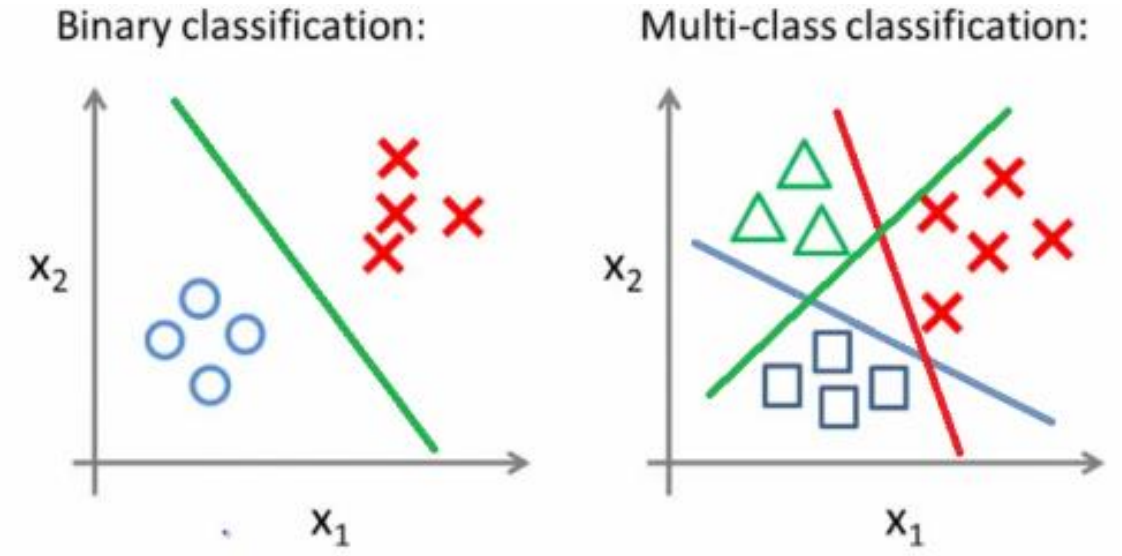
Where,

$\hat{y}$ – *predicted value of y*

$\bar{y}$ – *mean value of y*

딥러닝과 신경망

- 분류 문제
  - 새로운 데이터가 어떤 카테고리 집합에 속하는지 판단하는 것

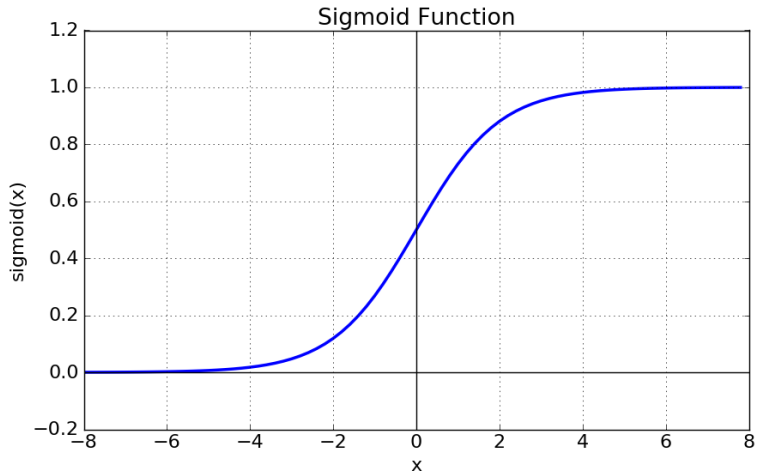


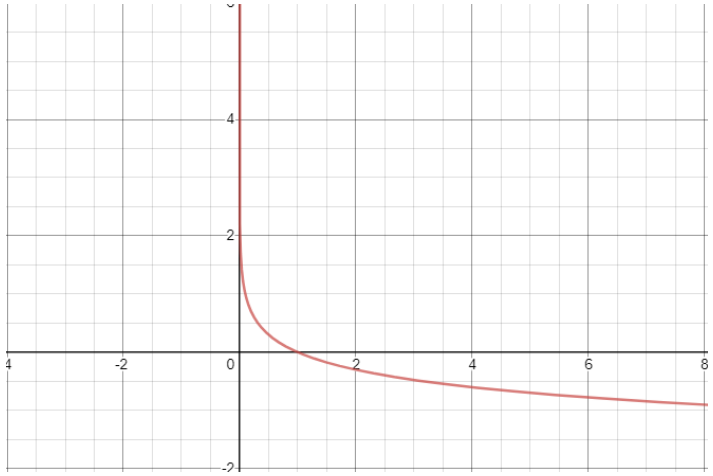
딥러닝과 신경망

- 분류에 사용되는 손실함수
  - BCE(Binary Cross Entropy): 이진 분류기를 훈련 시 사용하는 함수로 손실함수는 예측 값과 실제 값이 같으면 0이 되는 특성을 갖고 있어야 합니다. 예측 값과 실제 값이 모두 1로 같을 때 손실함수 값이 0이 되어야함

$$L = -\frac{1}{N}\sum_{i=1}^{N} t_i \log(y_i) + (1 - t_i)\log(1 - y_i)$$

$$\text{if } y_i = 1, t_i = 1, L = 0$$
$$\text{if } y_i = 0, t_i = 1, L = \infty$$





  - CCE(Categorical Cross Entropy)

$$L = -\frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{C} t_i \log(y_{ij})$$

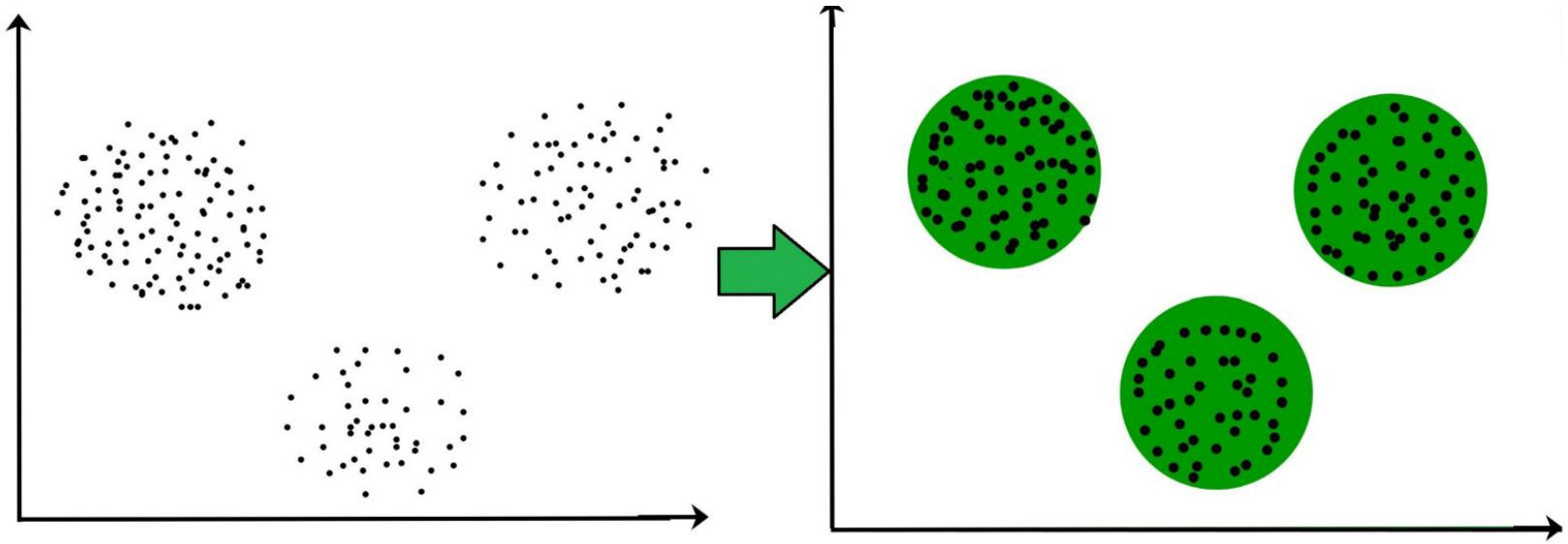
$$p_j = \frac{e^{x_j}}{\sum_{k=1}^{K} e^{x_k}}$$

$$= \frac{e^{x_j}}{e^{x_1} + e^{x_2} + \cdots + e^{x_K}} \quad for\ j = 1, \ldots, K$$

...(공식1: softmax 함수)

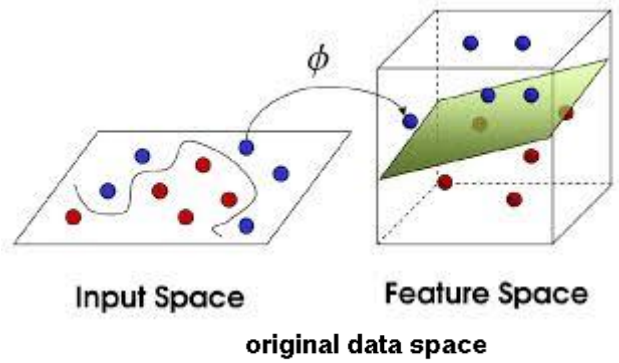
딥러닝과 신경망

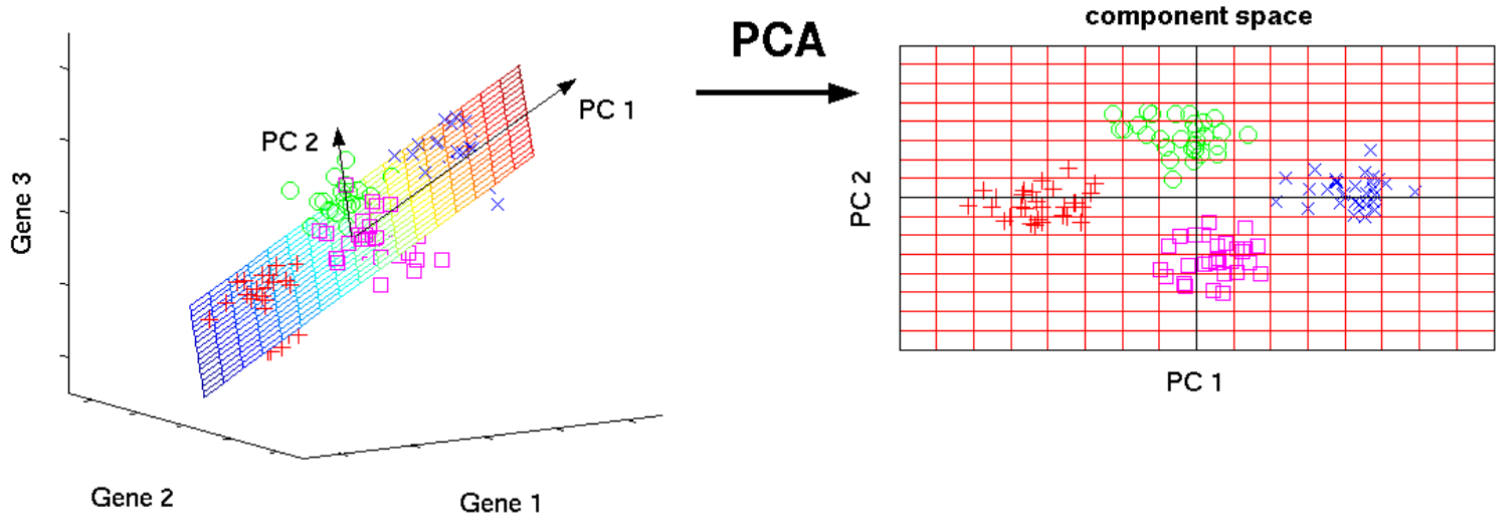
- 클러스터링
  - K개의 그룹에 속하는 유사한 샘플을 찾는 과정



딥러닝과 신경망

- 차원 축소
  - 입력 데이터의 차원의 저주를 피하기 위해 차원을 축소하는 방법

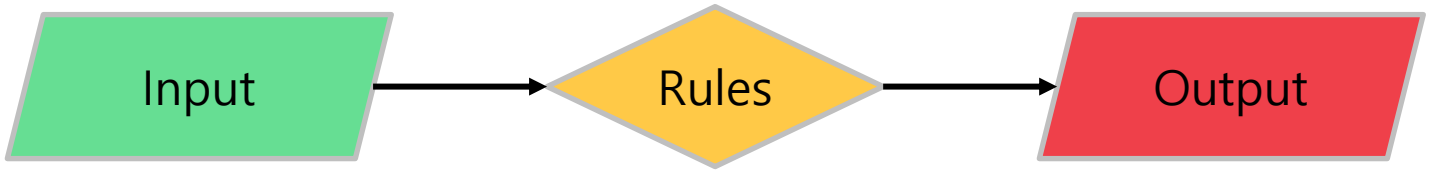




Reduce unnecessary representation axis

딥러닝과 신경망

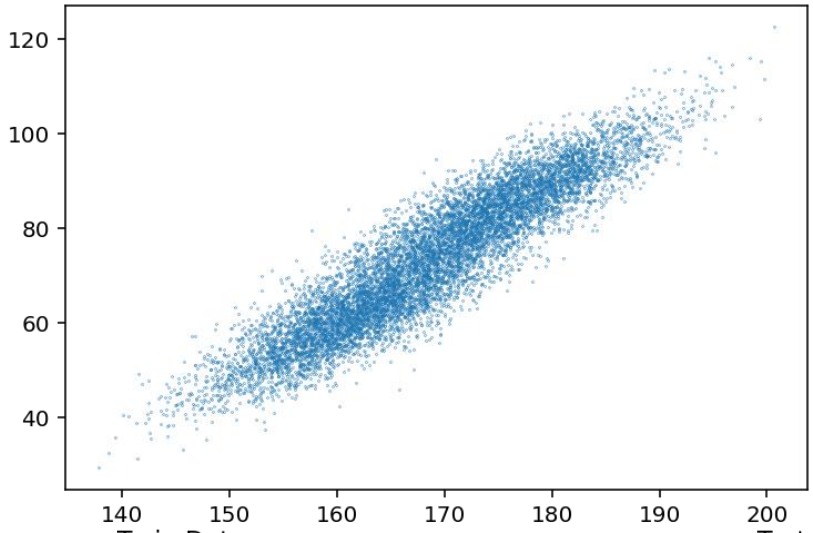
- 일반적인 프로그래밍 패러다임



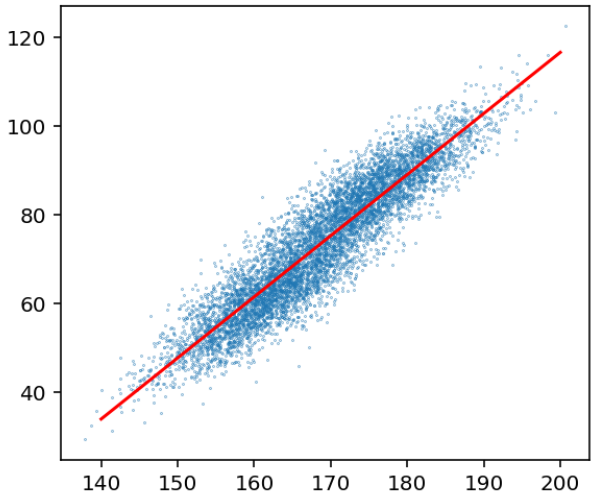
- 기계학습 프로그래밍 패러다임

딥러닝과 신경망

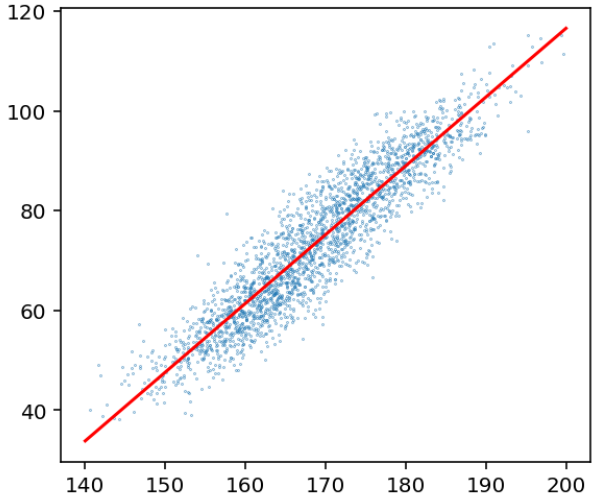
- 키와 몸무게 예시



Train Data



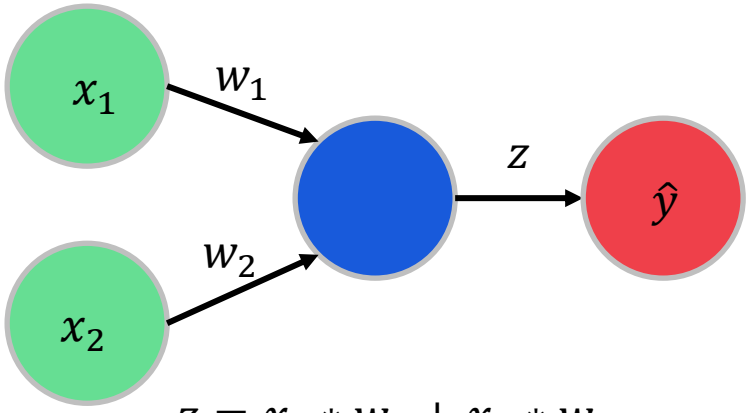
Test Data



딥러닝과 신경망

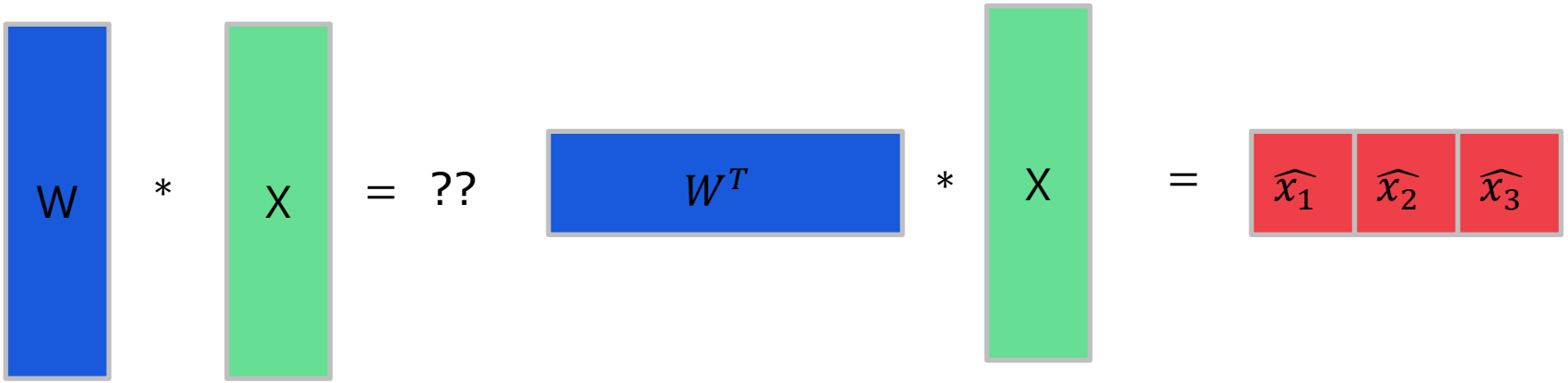
- 키와 몸무게 예시

	Gender	Height	Weight
0	Male	73.847017	241.893563
1	Male	68.781904	162.310473
2	Male	74.110105	212.740856
3	Male	71.730978	220.042470
4	Male	69.881796	206.349801



$$z = x_1 * w_1 + x_2 * w_2$$

$$\sum_{i=1}^{n} x_i w_i = \sum_{i=1}^{n} WX+b$$

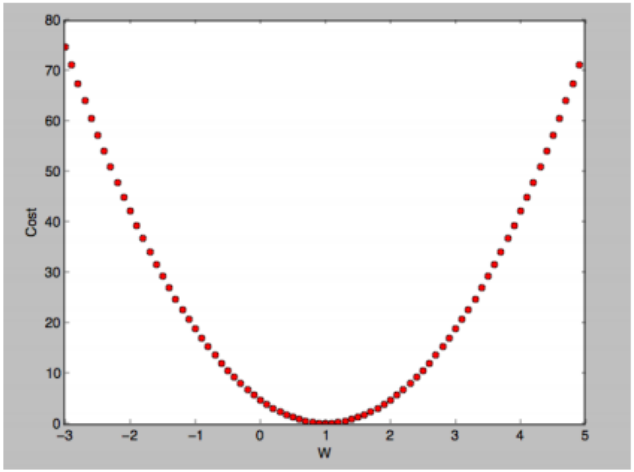


딥러닝과 신경망

- 경사하강법
  - 역전파가 해결한 문제는 파라미터가 매우 많고 층이 여러 개 쌓였을때 가중치(w)와 편향(b)를 학습시키기 어려웠다는 것인데, 앞에서 배운 역전파 알고리즘으로 각 layer에서 기울기를 구하고 그 값을 이용하여 Gradient descent 방법으로 가중치와 편향을 업데이트 시키면서 해결한 것

How to minimize cost?

$$cost(W) = \frac{1}{m}\sum_{i=1}^{m}(Wx^{(i)} - y^{(i)})^2$$



$$\mathrm{W} := \mathrm{W} - \alpha \frac{\partial}{\partial \mathrm{W}} \mathrm{cost(W)}$$

$$\mathrm{Cost(W,b)} = \frac{1}{2\mathrm{m}}\sum_{i=1}^{\mathrm{m}}(W\mathrm{x}^{(\mathrm{i})} - \mathrm{y}^{(\mathrm{i})})^2$$

딥러닝과 신경망

Hypothesis
Model

$$\mathrm{H(x)} = \mathrm{Wx}$$

Cost
Loss

$$\mathrm{Cost(W,b)} = \frac{1}{\mathrm{m}}\sum_{i=1}^{\mathrm{m}}(W\mathrm{x}^{(\mathrm{i})} - \mathrm{y}^{(\mathrm{i})})^2$$

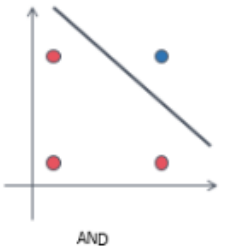
$$\mathrm{Cost(W,b)} = \frac{1}{2\mathrm{m}}\sum_{i=1}^{\mathrm{m}}(W\mathrm{x}^{(\mathrm{i})} - \mathrm{y}^{(\mathrm{i})})^2$$

Optimization

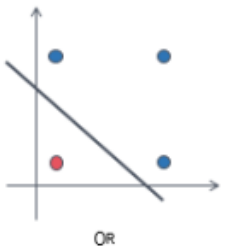
$$\mathrm{W} := \mathrm{W} - \alpha\frac{\partial}{\partial \mathrm{W}}\mathrm{cost(W)}$$

딥러닝과 신경망

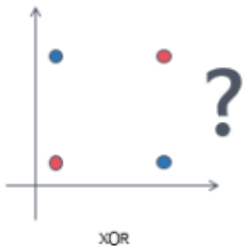
- 선형 분류기



AND		
Input_A	Input_B	Output
0	0	0
0	1	0
1	0	0
1	1	1

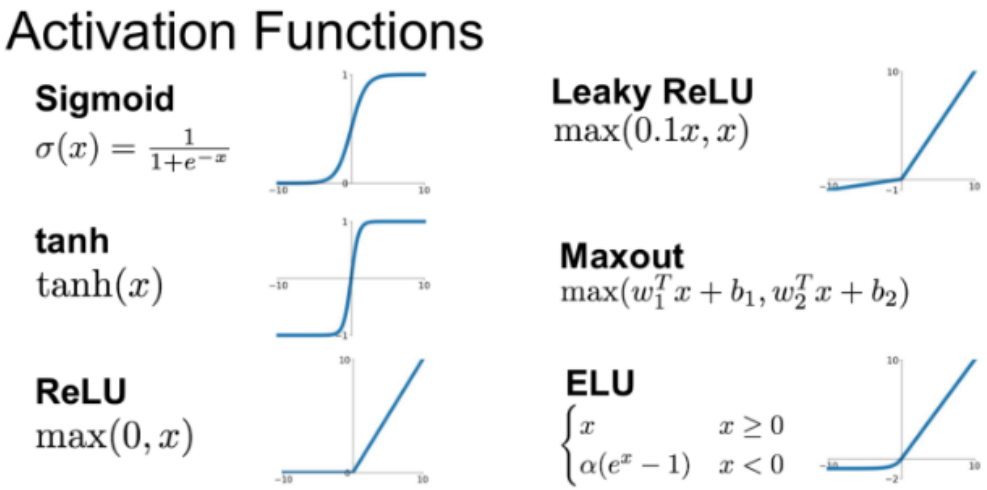


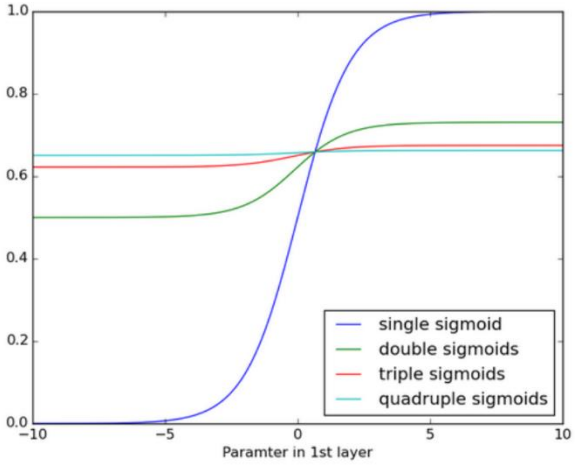
OR		
Input_A	Input_B	Output
0	0	0
0	1	1
1	0	1
1	1	1



XOR		
Input_A	Input_B	Output
0	0	0
0	1	1
1	0	1
1	1	0

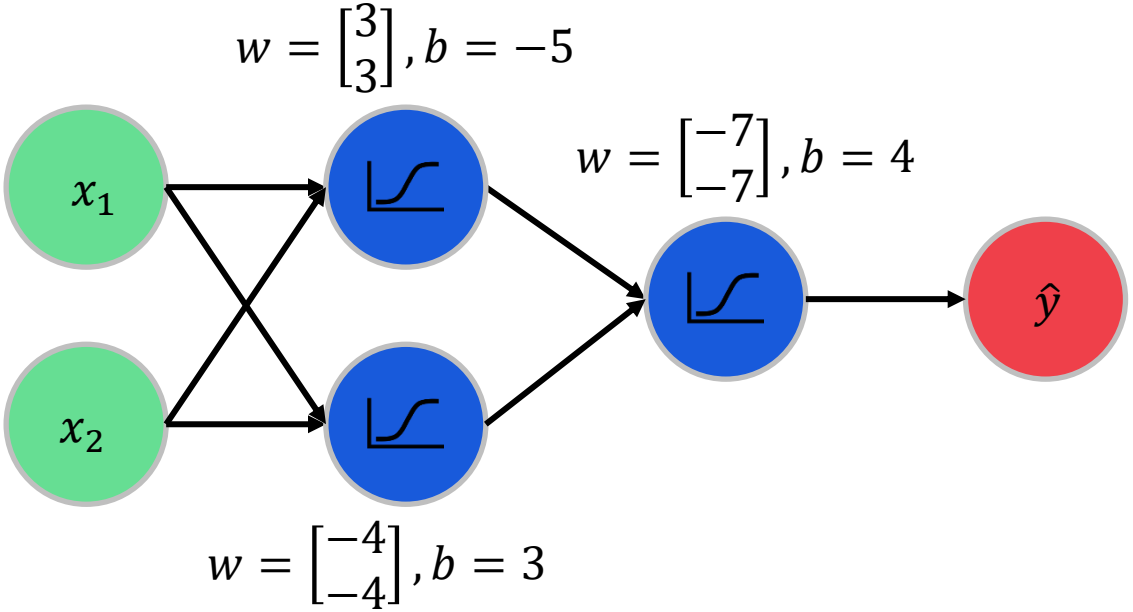
- 활성화 함수





딥러닝과 신경망

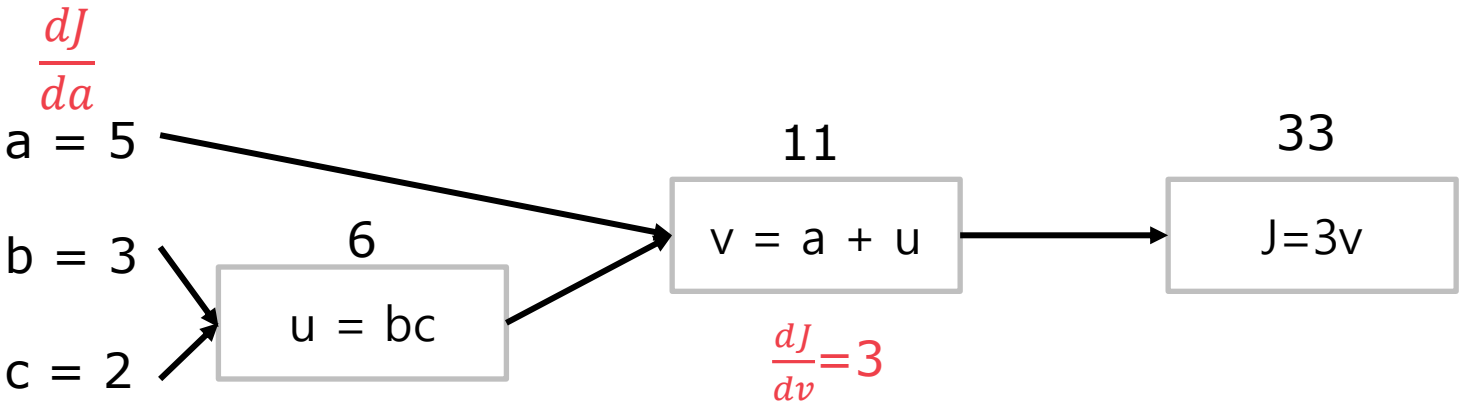
- XOR문제 해결



- Linear한 문제만 해결할 수 있었지만 활성화 함수를 사용하여 Non-linear 문제도 해결 가능
- 비선형 문제를 이렇게 MLP로 해결할 수 있었지만 MLP의 레이어를 쌓을 수록 weight와 bias를 학습시키는 것이 어려워짐

딥러닝과 신경망

- 역전파
  - 역전파 알고리즘이 등장하면서, 여러 층을 쌓은 신경망 모델의 학습이 가능해짐
  - 출력 값에 대한 입력 값의 기울기(미분 값)을 출력층에서부터 계산하여 거꾸로 전파 시키는 것



$\frac{dJ}{dv}=3$

$\frac{dJ}{da}=3 = \frac{dJ}{dv} * \frac{dv}{da}$

$\frac{dv}{da}=1$

자세한 설명: https://www.youtube.com/watch?v=nJyUyKN-XBQ&list=PLkDaE6sCZn6Ec-XTbcX1uRg2_u4xOEky0&index=15&ab_channel=DeepLearningAI

딥러닝과 신경망

- CNN(Convolutional Neural Network)

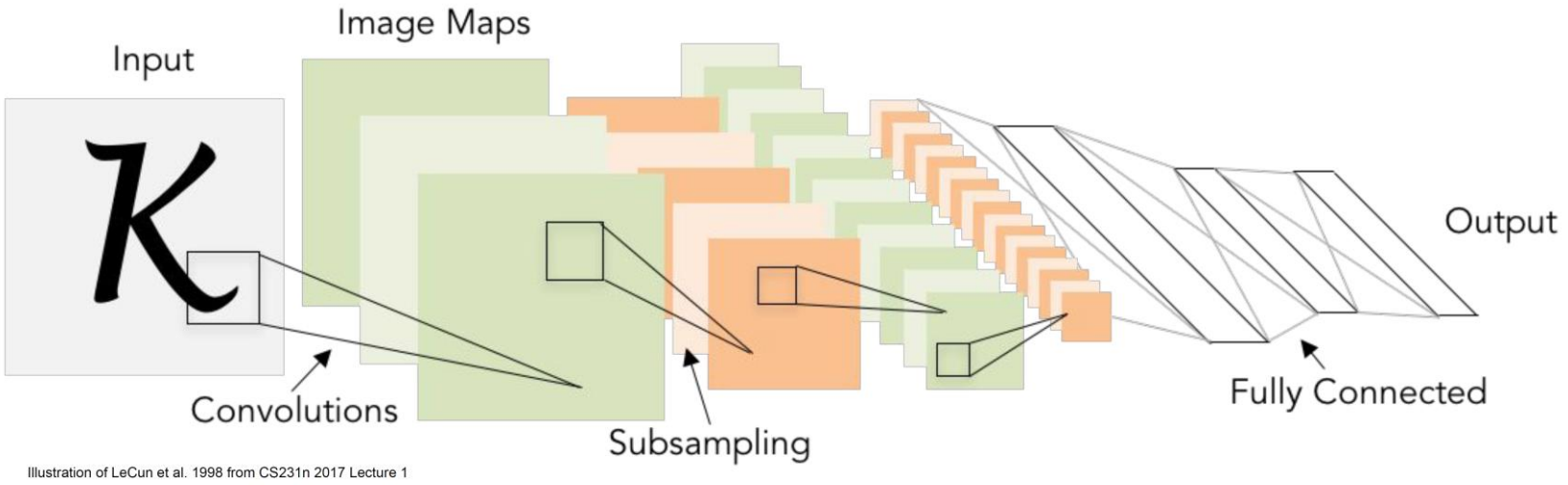
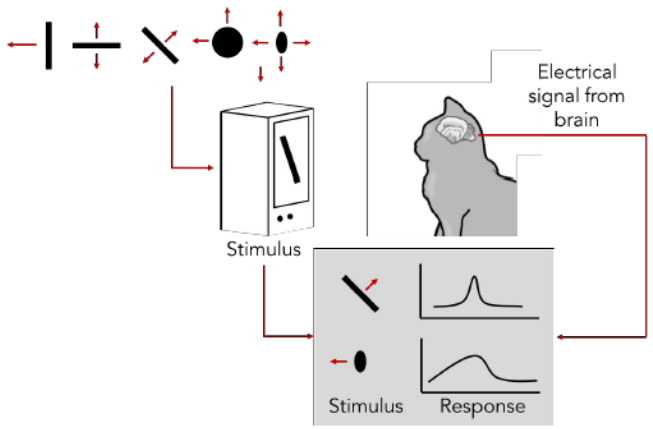


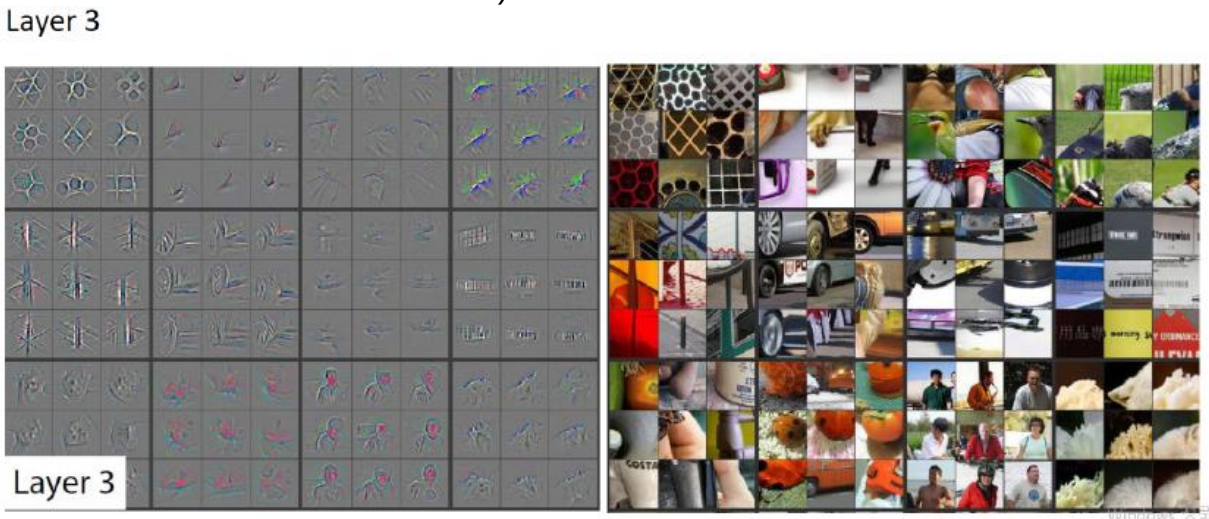
Illustration of LeCun et al. 1998 from CS231n 2017 Lecture 1

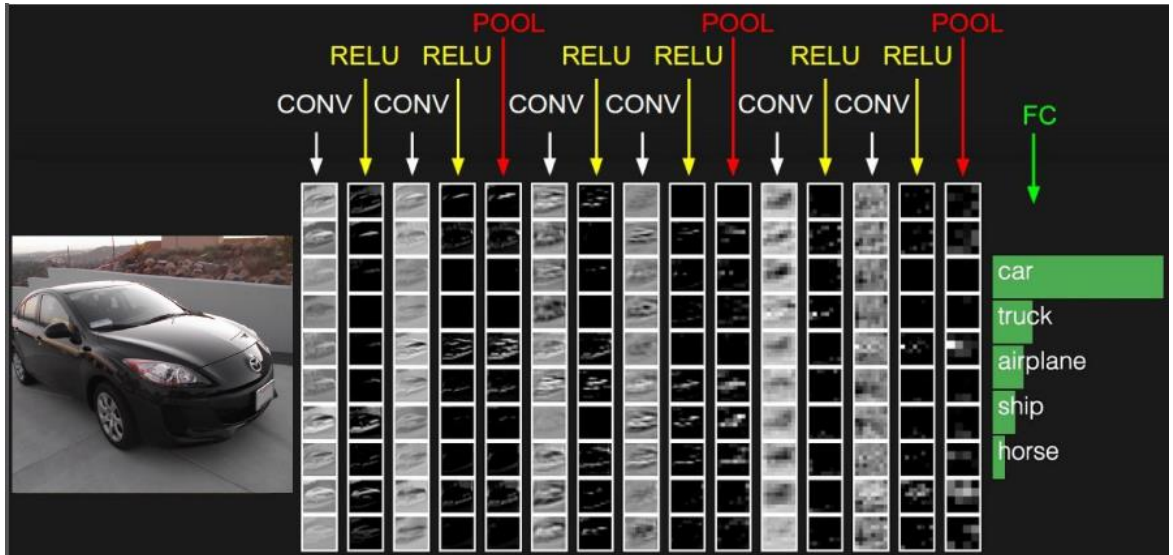


Cat image by CNX OpenStax is licensed under CC BY 4.0; changes made

딥러닝과 신경망

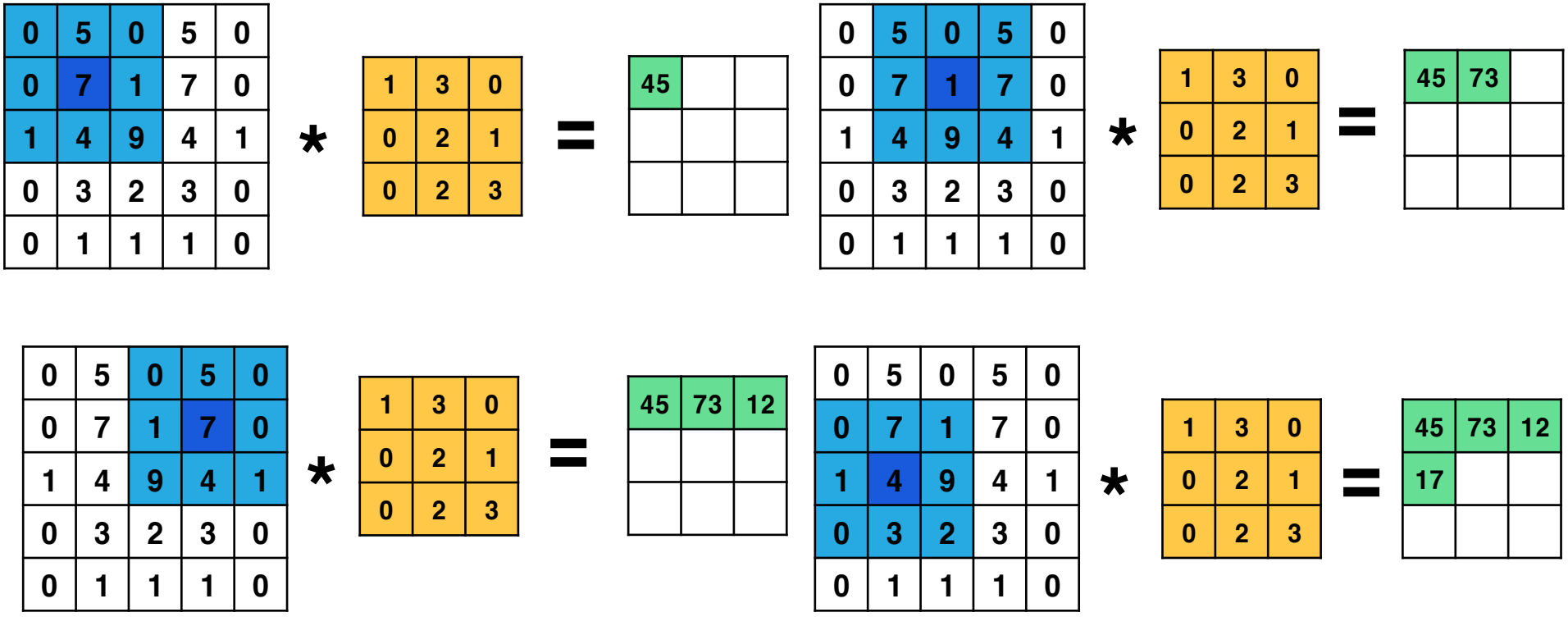
- CNN(Convolutional Neural Network)





딥러닝과 신경망

- CNN(Convolutional Neural Network)
  - 5x5 NxN (image) 3x3 FxF filter output 3x3 output size = input size – filter size +1
  - output size = (N-F)/stride+1 stride 1 => (5-3)/1 +1 = 3 stride 2 => (5-3)/2+1 =2



딥러닝과 신경망

- CNN(Convolutional Neural Network)
  - Zero padding
  - Input size : WxH, Number of filters : K, Filter size : F, Stride size: S, Zero padding : P
  - Input data: $W\_1 \times H\_1 \times D\_1$ $W\_2=(W\_1-F+2P)/S+1$, $H\_2=(H\_1-F+2P)/S+1$, $D\_2=K$

0	0	0	0	0	0	0
0	0	5	0	5	0	0
0	0	7	1	7	0	0
0	1	4	9	4	1	0
0	0	3	2	3	0	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

*

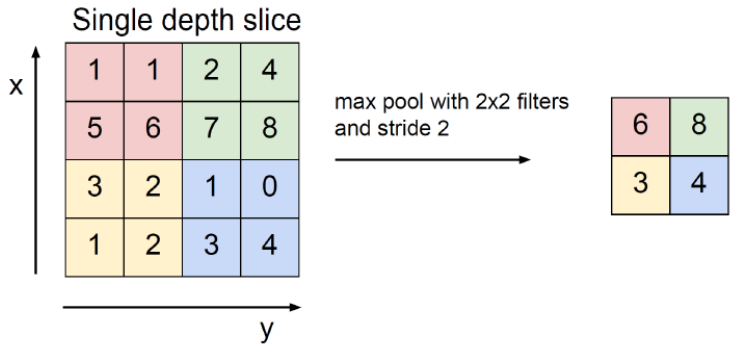
1	3	0
0	2	1
0	2	3

=

26	27	28	24	0
21	65	44	40	7
15	50	45	37	9
9	26	43	29	7
1	12	12	13	3

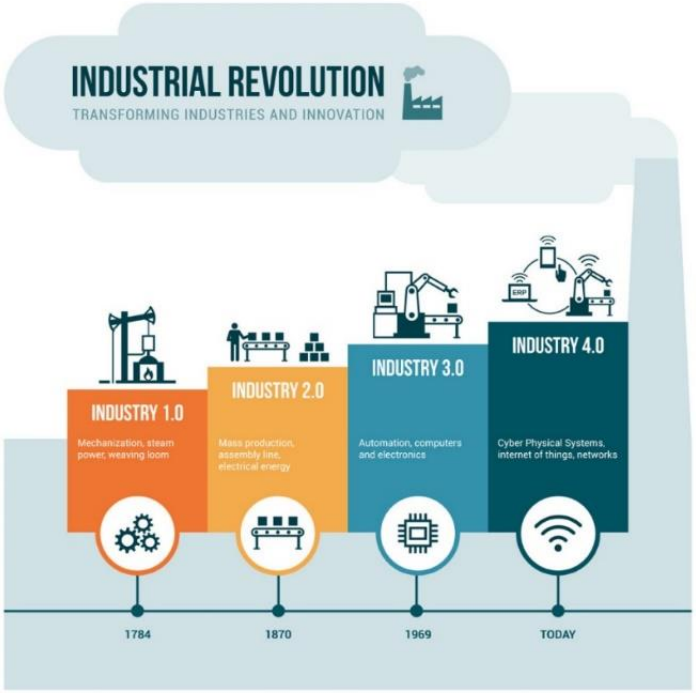
MAX POOLING

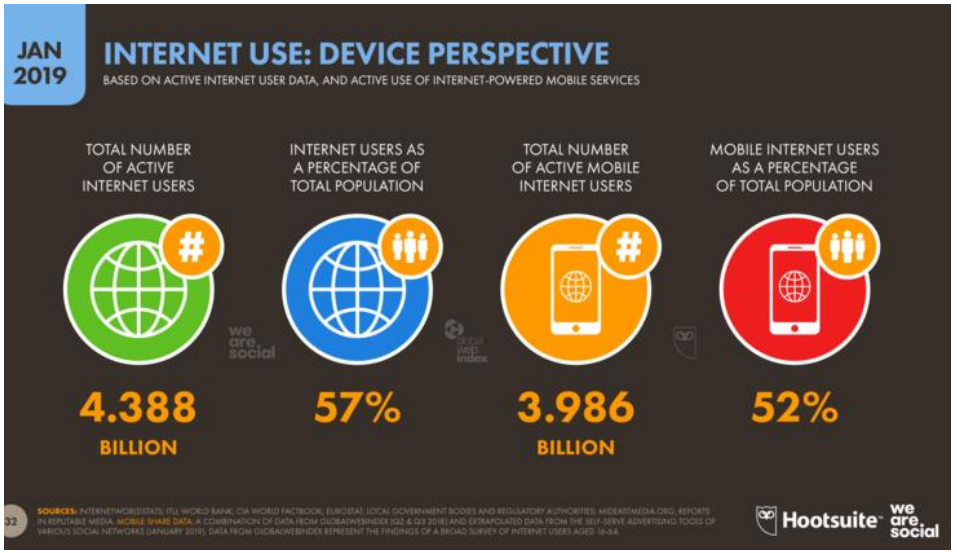
  - pooling



연합학습 -등장배경-

- 인터넷과 스마트폰의 엄청난 보급률로 인한 초 연결 시대로 수십억 명의 사람들이 계속해서 웹에 연결하면서 엄청난 데이터가 발생되고 축적됨
- 4차 산업 혁명은 인공지능, 빅데이터 등 디지털 기술로 촉발되는 초연결 기반의 지능화 혁명[1]

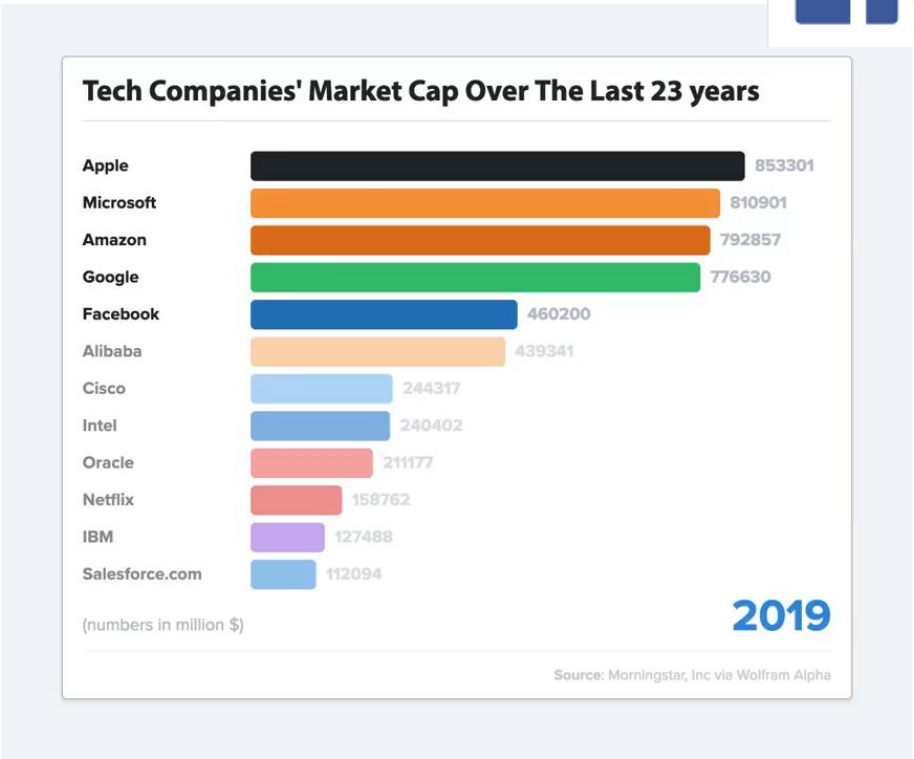




[1] https://www.4th-ir.go.kr/4ir/list
https://nadiahanin.wordpress.com/the-fourth-industrial-revolution/

연합학습 -등장배경-

- 분산형 데이터의 증가가 생성한 새로운 요구
  - 개인 맞춤형 서비스
  - 데이터 통합, 공유로 새로운 비즈니스 기회와 통찰력을 발견하는 것
- 데이터 통합에서 발생하는 다양한 문제
  - Centralized Learning(이해관계, 효율성, 분석시간, 비용 등 많은 제약 발생
- Market cap으로 살펴보는 산업 변화



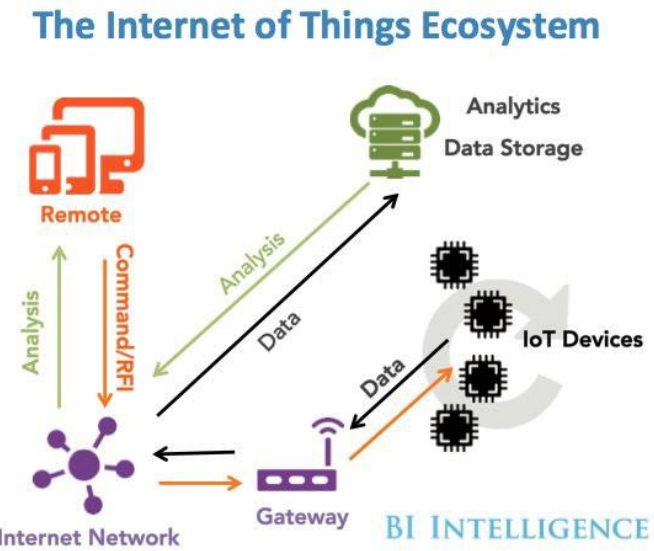
연합학습 -등장배경-

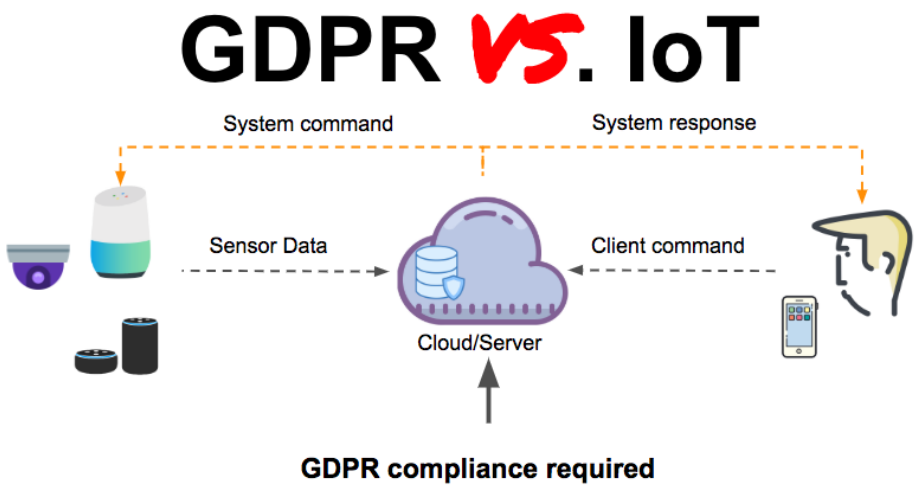
- 웹에 데이터가 돌아다니면서 개인정보 유출 사건들이 발생
- 2017년, 의료기록, 신용카드 혹은 금융 데이터, 개인정보가 하루 1000만 개 꼴로 유출됐으며, 초단위로 환산하면 1초당 122개 수준
- 의료기록, 신용카드, 금융, 개인정보의 유출사건에서 많은 사람들이 피해를 보면서 정보보호에 대한 제도적, 기술적 개선 필요성 증대
- 이에 따라, 다양한 기술 연구 및 법률 및 제도가 수립됨

http://www.datanet.co.kr/news/articleView.html?idxno=115123

연합학습 -등장배경-

- 유럽의 일반 정보보호법(GDPR)이 2018년 5월 발효, 한국 개인정보보호, 정보통신망법 등
- 데이터 기반 산업군에서 데이터의 수집, 처리, 관리하는 기존 방식 수정 필요
- de-identification은 데이터에 포함된 개인정보를 비식별화하는 방법으로 기존에 많이 사용되었음
- 이러한 정보보호법은 민감한 개인정보의 유출 및 공유를 어렵게 하는 제약으로 작용함
- 이에 따라, **데이터 공유 및 집계를 하지 않는 인공지능 모델의 학습 방법론인 연합 학습**이 등장하게 됨





https://resources.experfy.com/iot/internet-of-everything-the-iot-market-is-projected-to-expand-12x-from-2017-2023/

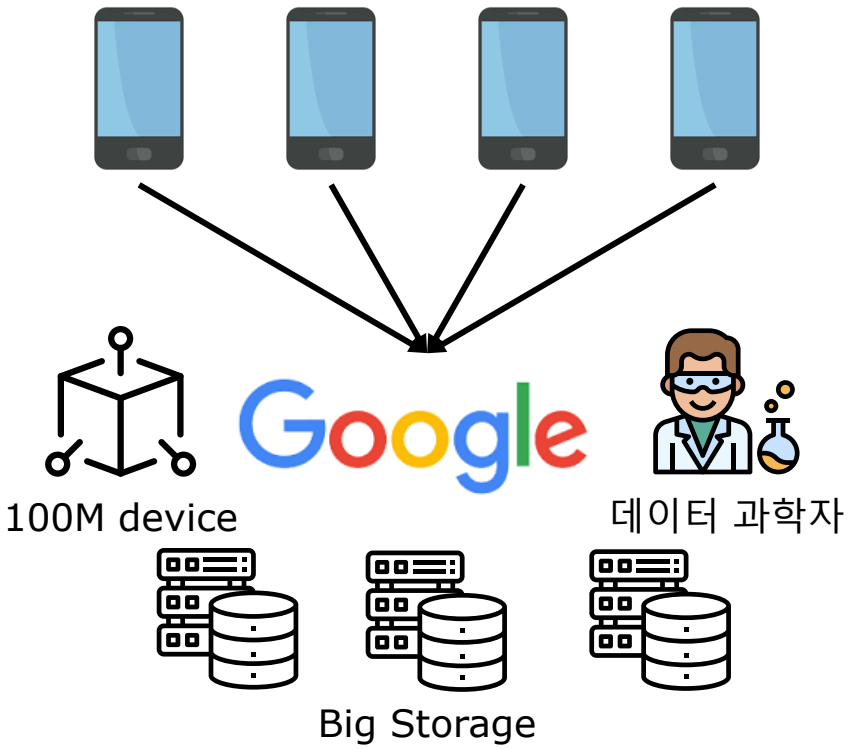
연합학습 -등장배경-

- 문제 상황
  - 빅 데이터

100M device

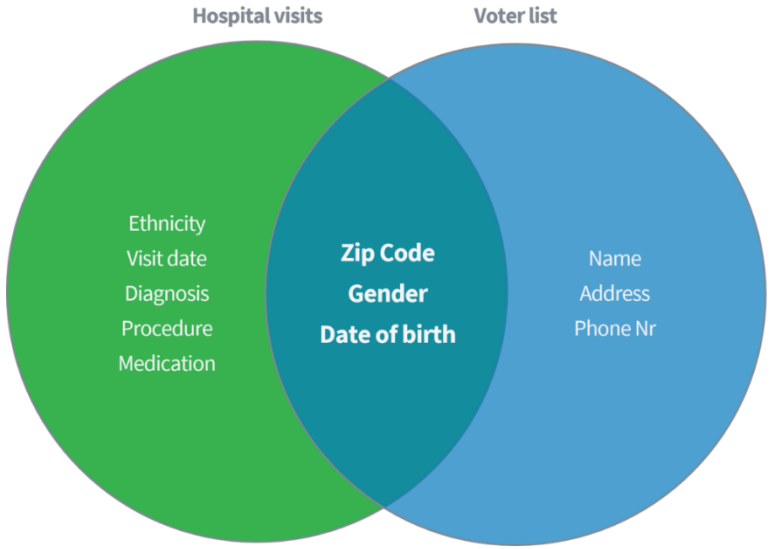
1. Data size

2. Privacy



연합학습 -등장배경-

- 문제 상황
  - 프라이버시
  - 비식별화(제거, 변환, 해쉬)
  - 연결 공격(Linkage attack)



구분	지역 코드	연령	성별	질병
1	13053	28	남	전립선염
2	13068	21	남	전립선염
3	13068	29	여	고혈압
4	13053	23	남	고혈압

구분	이름	지역코드	연령	성별
1	김민준	13053	28	남
2	박지훈	13068	21	남
3	이지민	13068	29	여
4	최현우	13053	23	남

유사 식별자	미국인구 중 식별된 비율 (248 million)
5-digit ZIP, gender, date of birth	87%
place, gender, date of birth	53%
country, gender, date of birth	18%

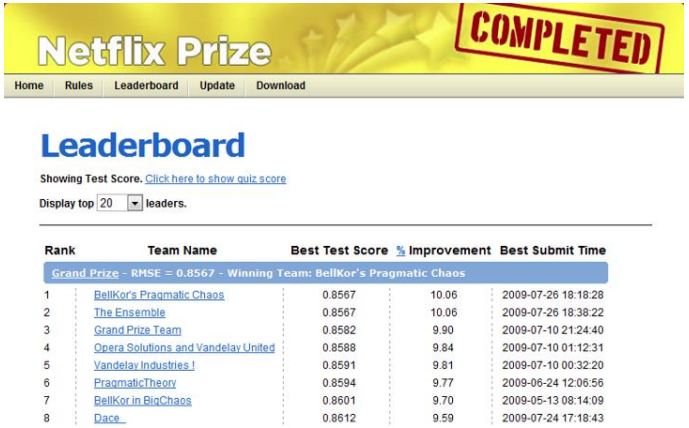
https://itwiki.kr/w/K-%EC%9D%B5%EB%AA%85%EC%84%B1
https://www.syntho.ai/5-examples-why-removing-names-fails-as-anonymization/

연합학습 -등장배경-

- 문제 상황
  - 프라이버시
  - 연결 공격(Linkage attack)
  - 이런 예제들은 익명화가 데이터의 보안성을 높여주지않는 다는 것을 증명하였음

NETFLIX

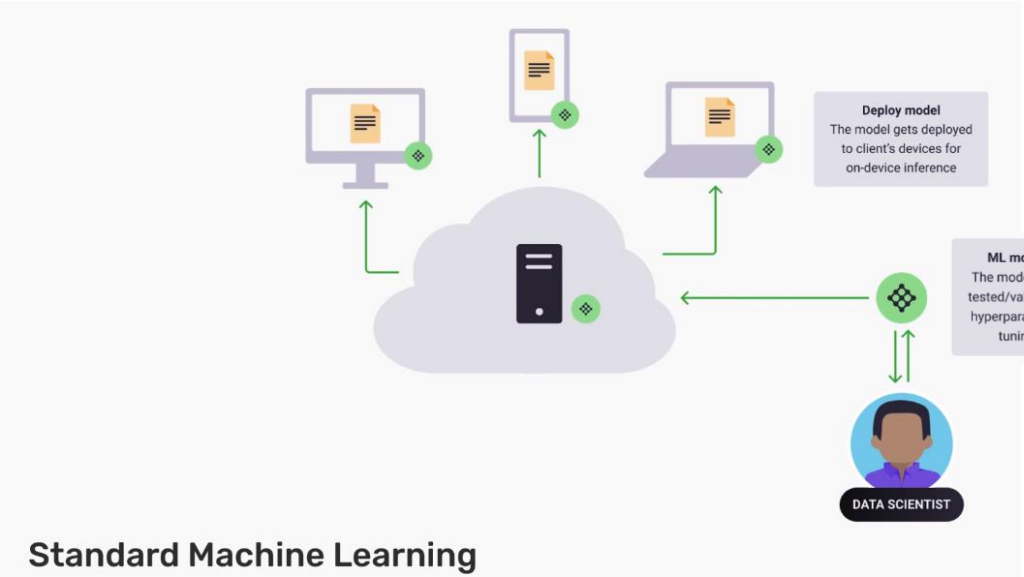
UserID	영화	평가일	등급
123456789	미션 임파서블	2008 년 10 월 12 일	4



https://www.syntho.ai/5-examples-why-removing-names-fails-as-anonymization/

기계 학습 기존 학습법

- 기계학습은 인공지능의 하위 분야로 데이터에서 패턴과 특징을 찾아 내기위해 학습된 알고리즘을 말함 (새로운 데이터에 대한 의사결정과 예측을 수행할 수 있음)
- 표준 기계학습에서 데이터는 수집되고 선별되어 하나(중앙 저장소) 또는 여러 개의 클라이언트(클라우드)에 저장됨
- 데이터 과학자는 머신러닝을 하기위해 저장소에 접근하고, 기계학습 알고리즘을 활용하여 데이터에 대한 AI 모델을 훈련 시킴
- **데이터가 클라이언트에서 중앙 서버로 이동한다는 것을 기억**



출처:
https://www.youtube.com/watch?v=49XVqOCBtXI&ab_channel=OpenMined

연합 학습

- 전통적인 기계학습 방법론에는 크게 4가지 한계가 있음

중앙집중식 데이터

표준 기계학습 접근법은 모든 훈련 데이터를 한데 모아서 사용

개인정보 침해

데이터 소유자들은 개발자와 중앙 머신 소유자에게 민감한 정보가 노출될 수 밖에 없음

민감한 정보의 부족

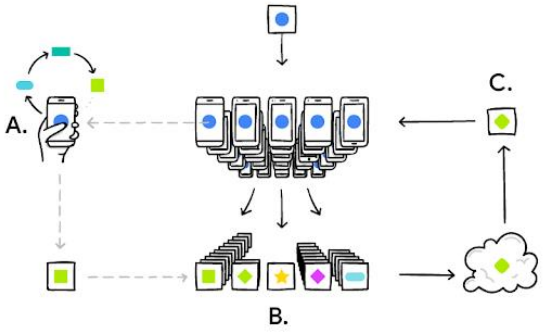
개인정보보호를 위해, 소유자가 유용하지만 민감한 데이터를 항상 공유하지 않으면 발생

계산 비용

수백만개 이상의 데이터를 학습하는 중앙 기계는 높은 계산을 할 수 있는 리소스가 필요

연합학습

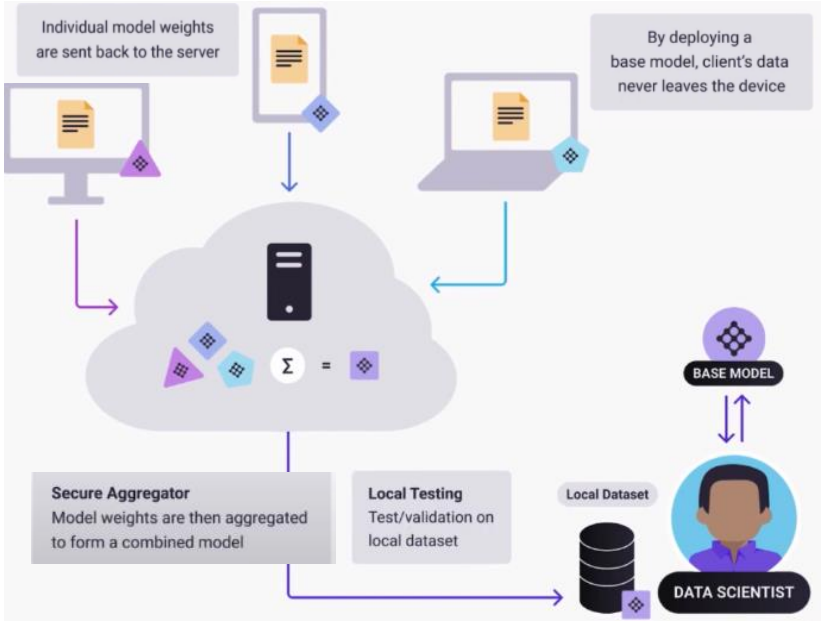
- 2017년 구글의 한 연구팀에서는 Federated Learning(연합 학습)을 제안함
- 연합학습이란 기기나 기관 등에 분산된 데이터를 직접 공유하지 않으면서, 협력하여 AI 모델을 학습할 수 있는 분산형 학습법임
- 연합 학습은 개별 장치에서 발생하는 데이터를 학습하기 위해 사용하는 어플리케이션(모델)을 제공하고 개별 장치에서 학습하면서 가중치만을 서로 공유함
- 초기 연구에서 데이터를 모두 사용한 모델과 대략 5%의 성능 차이를 보임
- 하지만, 데이터 수집없이, 외부 전송없이 가중치를 공유하여 AI모델을 학습할 수 있음
- GDPR, 개인정보보호 시대에 AI 모델을 학습하기 위한 최적의 방법으로 각광받고 있음
- 연합학습의 장점
  - 데이터는 소유자의 장치를 벗어나지 않음
  - 민감한 개인정보가 보호됨
  - 데이터 소유자는 데이터 공유가 더 편리해짐으로써 더 나은 AI 모델 구축을 가능하게 함
  - 모델 학습이 여러 장치에 분산되므로 표준 기계학습 방법론 보다 적은 계산 비용이 필요함



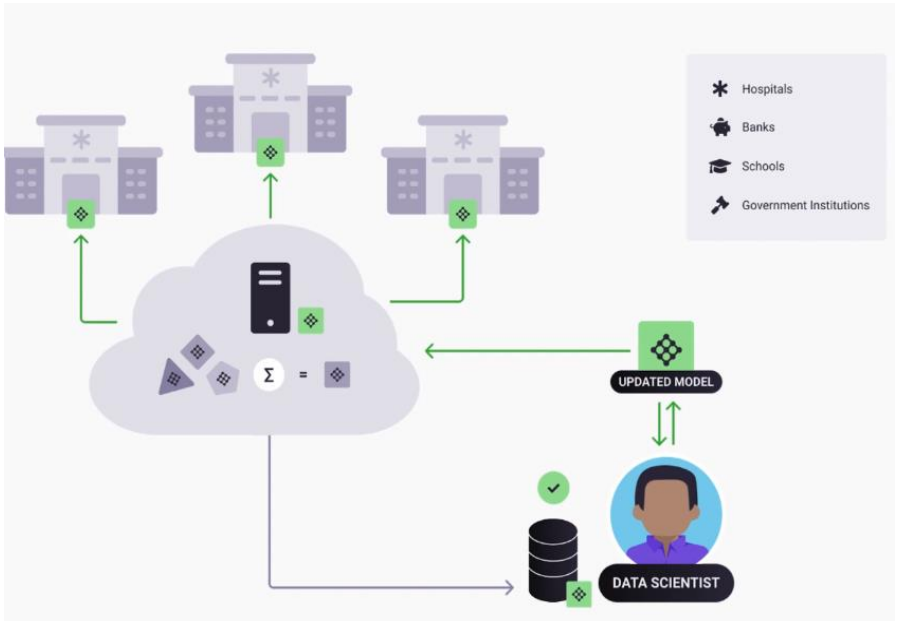
연합학습

- 연합학습은 2가지 종류로 구분 가능함

Cross-device Federated Learning



Cross-silo Federated Learning



연합학습

- 연합 학습과 분산학습

〈 표-1 〉 연합학습과 분산학습 비교

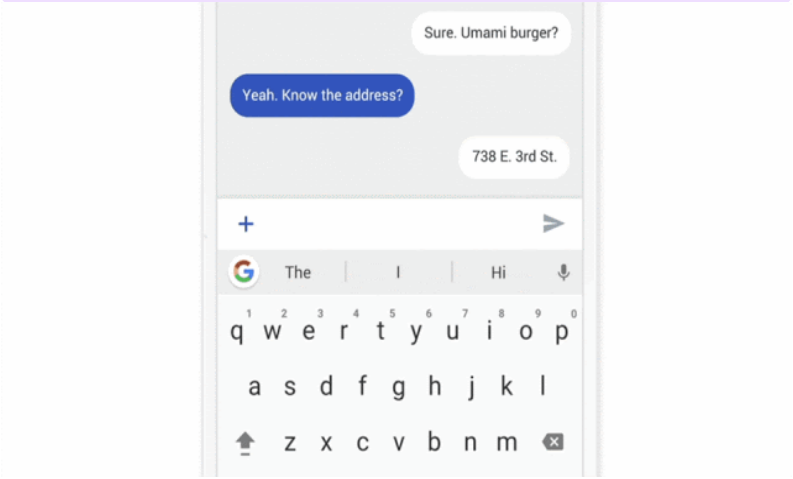
구분	분산학습 (Datacenter)	연합학습	
		Cross-silo	Cross-device
참여 환경	- 클라이언트가 하나의 클러스터나 데이터센터 내에 있는 컴퓨터 노드들로 구성	- 서로 다른 기업, 조직 (의료, 금융 등)단위에서 연합학습 클라이언트로 참여	- 매우 많은 수의 모바일 기기, IoT 기기 단위로 구성된 클라이언트가 연합학습에 참여
데이터 분포	- 데이터는 중앙에 저장 - 어떤 클라이언트라도 다른 클라이언트의 데이터를 읽을 수 있음	- **데이터가 로컬에서 생성되고, 분산되어 존재** - 각 클라이언트는 자신이 데이터를 저장하고, 다른 클라이언트의 데이터에는 접근할 수 없음 - Non-IID(Independent & Identically Distributed) Data	
통합 조정	- 중앙에서 통합하고 관리	- **중앙서버에서 전체 학습을 조정하고 관리** - 그러나, 중앙서버는 원본 데이터(raw data)를 볼 수 없음	
통신	- 하나의 데이터센터/클러스터 내의 모든 클라이언트는 연결됨	- Hub-and-spoke 형태로 구성 - Hub는 전체 학습을 조정(원본 데이터 없이)하고, 각 클라이언트에 연결	
데이터 가용성	- 모든 클라이언트는 거의 항상 가용		- 일부 클라이언트는 특정 순간(예: 주간)에만 가용
규모	- 1~1,000개 클라이언트	- 2~100개 클라이언트	- 대규모 병렬로 10^{10}개의 클라이언트까지 가능
주요 병목점	- 데이터센터 내 연산 능력 (고속통신망 가정)	- 연산 능력과 통신 속도 모두	- 통신 속도(wi-fi 또는 느린 통신환경에서 작동)
주소 지정	- 클라이언트는 고유 ID 또는 이름을 지정		- 클라이언트 ID 없음(직접 인덱싱할 수 없음)
상태 유지	- 모든 클라이언트는 매 라운드에 참여하고 상태를 전달		- 클라이언트가 작업에 한 번만 참여할 수 있음(매 라운드에 새로운 클라이언트 참여 가능)
신뢰성	- 클라이언트 참여 실패 가능성이 적음		- 매 라운드에서 클라이언트 5%이상 실패가능(배터리, 통신 불안정 등)
데이터 분할	- 클라이언트 간 임의 분할/재분할 가능	- 수평(horizontal) 또는 수직(vertical)적[8] 고정 분할	- 수평(horizontal)적 고정 분할

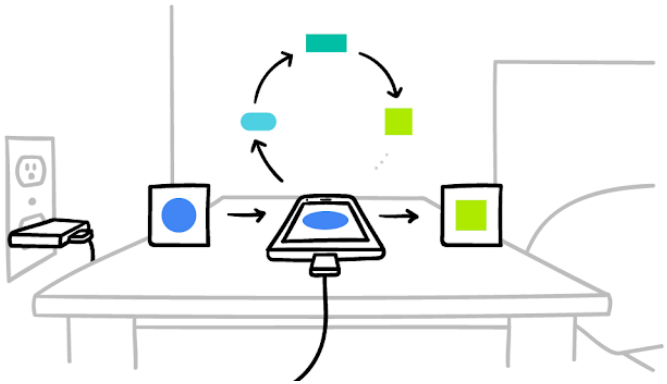
* 출처 : Peter Kairouz et al.(2019) 참고하여 재작성

출처: 연합학습 기술 동향 및 산업적 시사점, ETRI 2020.06

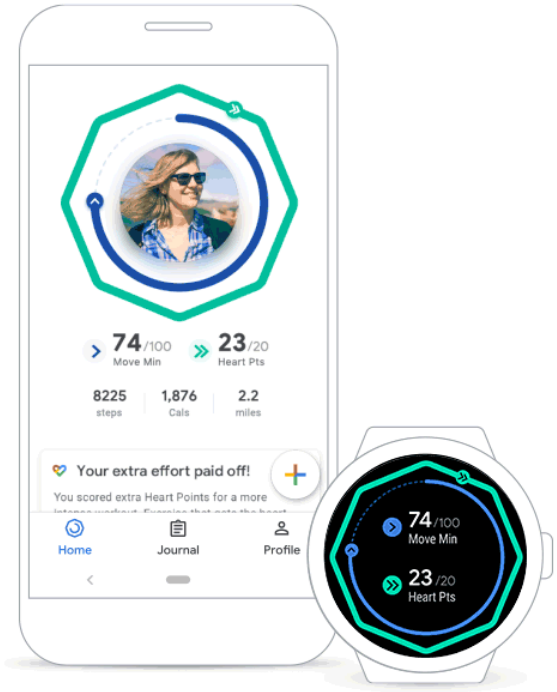
연합학습 사례

Gboard on Android





피트니스 앱



연합학습 사례

- 추가 사례
  - 자율 주행
  - 헬스케어 모니터링
  - 질병의 진단, 예후 예상
  - 신용카드 사기 탐지
  - 개인화된 추천 시스템
  - 개인정보를 보호받는 감시 시스템
  - 소셜 미디어의 감성 분석

관련연구

- 2021년 4월 Qi Dou, Tiffany Y. So의 CT스캔 사진에서 COVID-19 폐 이상을 감지하는 연합학습 연구가 게시되었음
- 7개의 다국적 센터에서 132 명의 환자 데이터를 사용했으며, 교육 및 테스트를 위해 홍콩의 3개 병원과 모델 일반화 가능성을 검증하기 위해 중국 본토와 독일에서 4개의 외부 독립 데이터 세트를 사용하였음

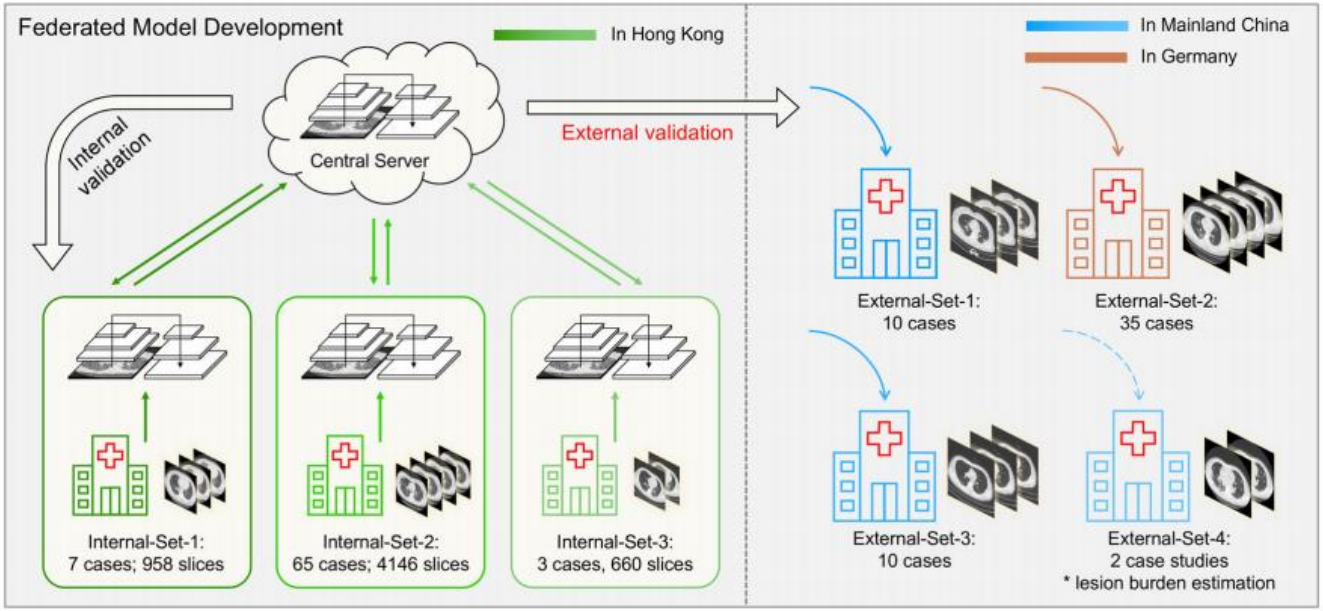
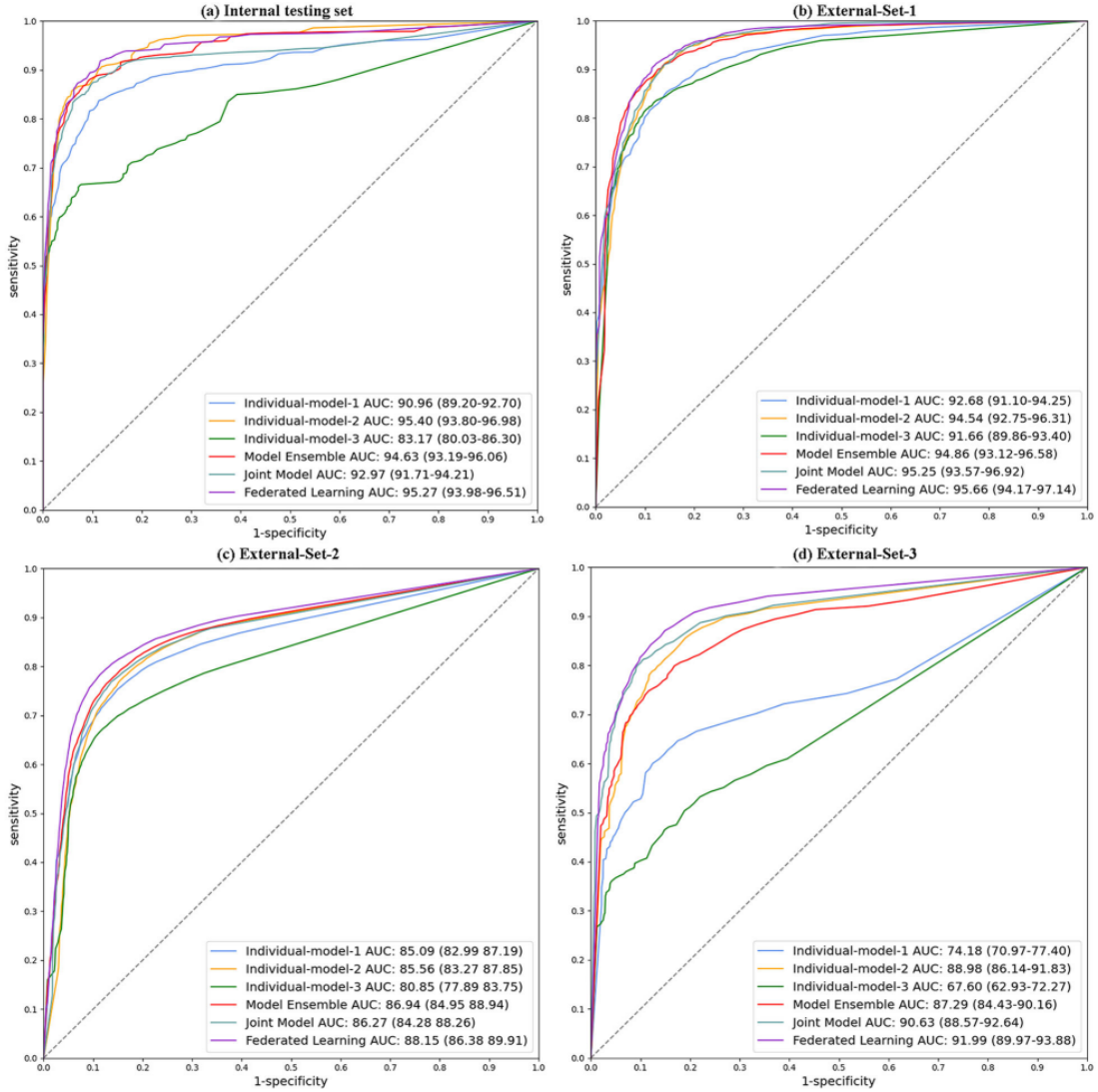
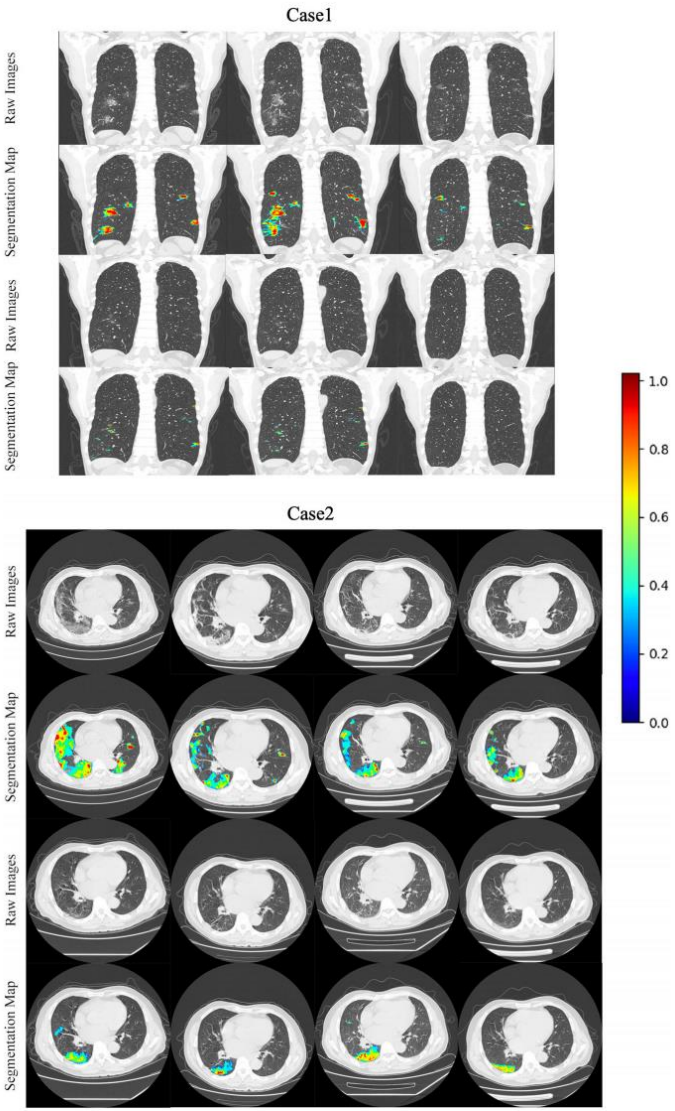


Fig. 1 Overview of our AI scheme to develop a privacy-preserving CNN-based model for detecting CT abnormalities in COVID-19 patients with a multinational validation study. A privacy-preserving AI system was developed with CT data from three hospitals in Hong Kong using federated learning, and then the generalizability was validated on external cohorts from Mainland China and Germany.

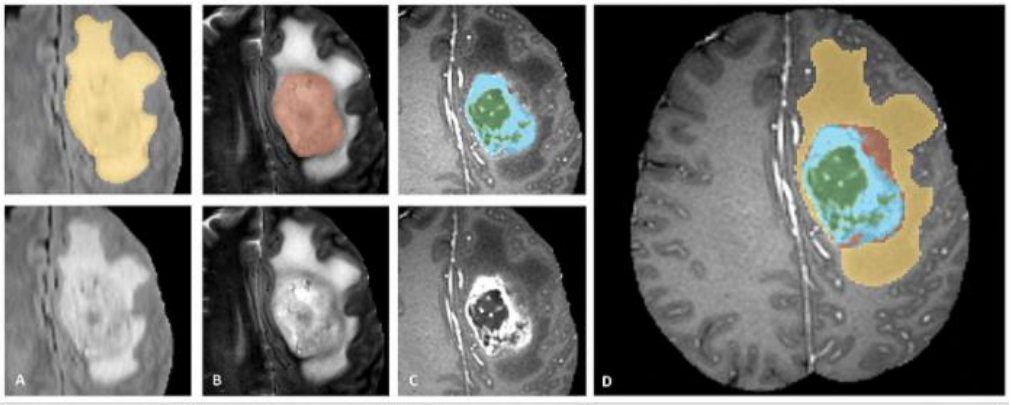
관련연구

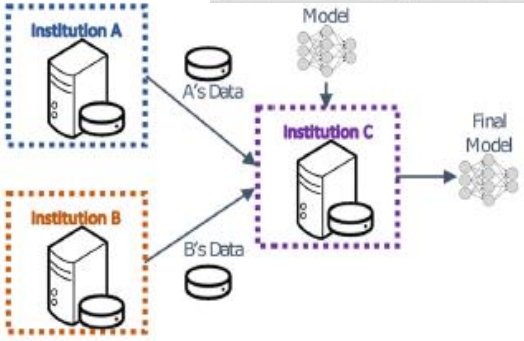

(a) Internal testing set
sensitivity
1-specificity
Individual-model-1 AUC: 90.96 (89.20-92.70)
Individual-model-2 AUC: 95.40 (93.80-96.98)
Individual-model-3 AUC: 83.17 (80.03-86.30)
Model Ensemble AUC: 94.63 (93.19-96.06)
Joint Model AUC: 92.97 (91.71-94.21)
Federated Learning AUC: 95.27 (93.98-96.51)
(b) External-Set-1
sensitivity
1-specificity
Individual-model-1 AUC: 92.68 (91.10-94.25)
Individual-model-2 AUC: 94.54 (92.75-96.31)
Individual-model-3 AUC: 91.66 (89.86-93.40)
Model Ensemble AUC: 94.86 (93.12-96.58)
Joint Model AUC: 95.25 (93.57-96.92)
Federated Learning AUC: 95.66 (94.17-97.14)
(c) External-Set-2
sensitivity
1-specificity
Individual-model-1 AUC: 85.09 (82.99 87.19)
Individual-model-2 AUC: 85.56 (83.27 87.85)
Individual-model-3 AUC: 80.85 (77.89 83.75)
Model Ensemble AUC: 86.94 (84.95 88.94)
Joint Model AUC: 86.27 (84.28 88.26)
Federated Learning AUC: 88.15 (86.38 89.91)
(d) External-Set-3
sensitivity
1-specificity
Individual-model-1 AUC: 74.18 (70.97-77.40)
Individual-model-2 AUC: 88.98 (86.14-91.83)
Individual-model-3 AUC: 67.60 (62.93-72.27)
Model Ensemble AUC: 87.29 (84.43-90.16)
Joint Model AUC: 90.63 (88.57-92.64)
Federated Learning AUC: 91.99 (89.97-93.88)

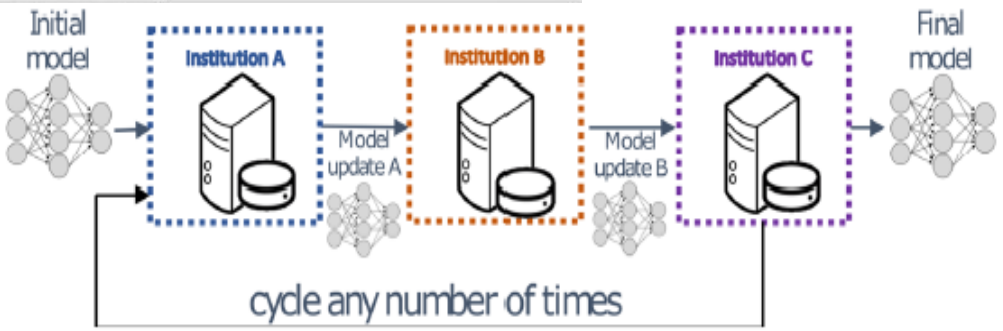

Case1
Raw Images
Segmentation Map
Raw Images
Segmentation Map
Case2
Raw Images
Segmentation Map
Raw Images
Segmentation Map
1.0
0.8
0.6
0.4
0.2
0.0

관련연구

- 2020년 7월 네이처지에 발표된 Micah J의 논문에 따르면 뇌암을 예측하는 뇌스캔 MRI 이미지 데이터를 사용하여 CDS 방법과 CIIL 방법의 비교 연구를 수행하였음
- 뇌스캔 MRI 데이터(BraTS 2017)은 10개의 기관, 3개의 멀티 모달 클래스로 구성되어 있으며, 240x240 크기의 이미지를 분류한다.
- 아래 그림의 d방식 CIIL을 사용하여 학습을 수행하였을 때, 높은 성능을 보였음





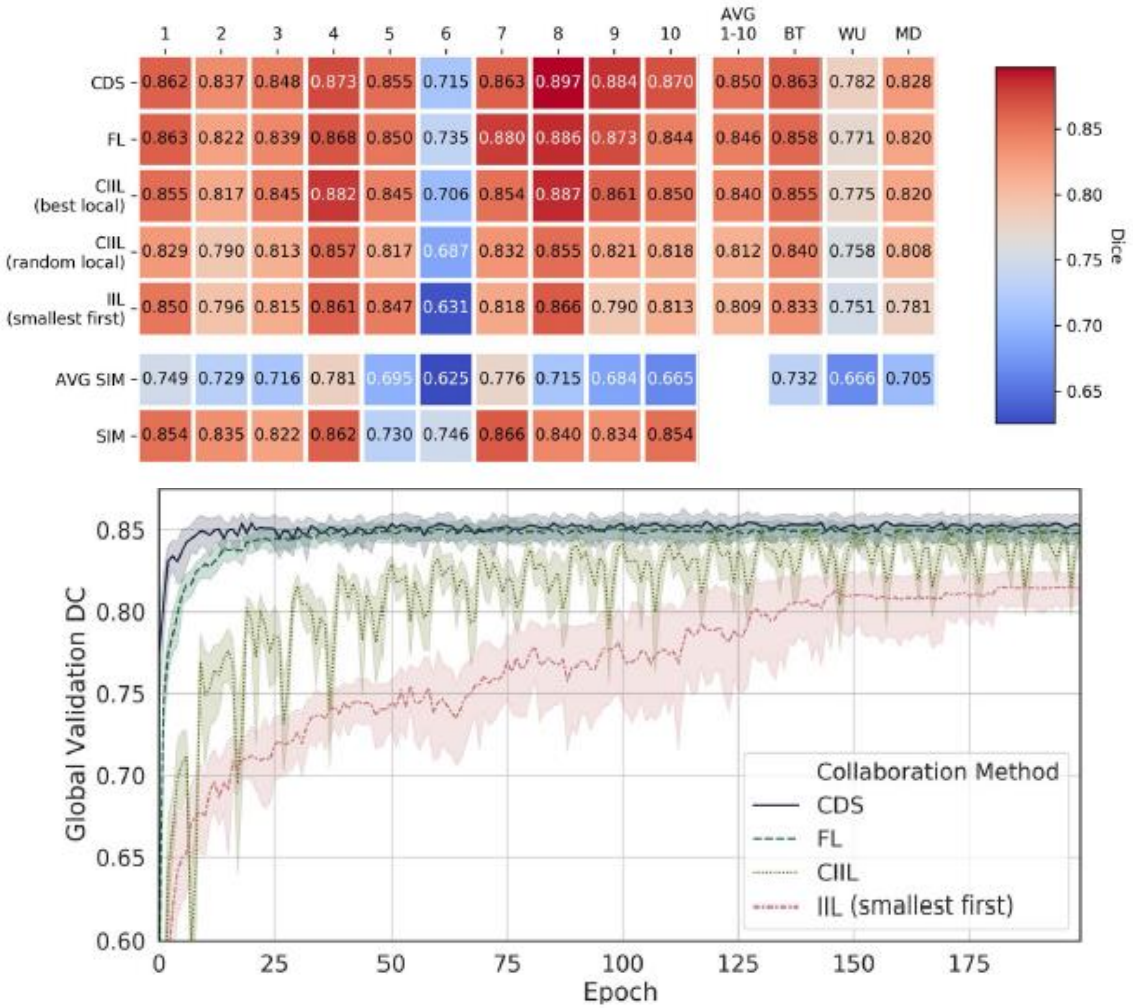


(a) Collaborative Learning through Centralized Data Sharing (d) Data-private Collaborative Learning using Cyclic Institutional Incremental Learning

https://www.nature.com/articles/s41598-020-69250-1.pdf

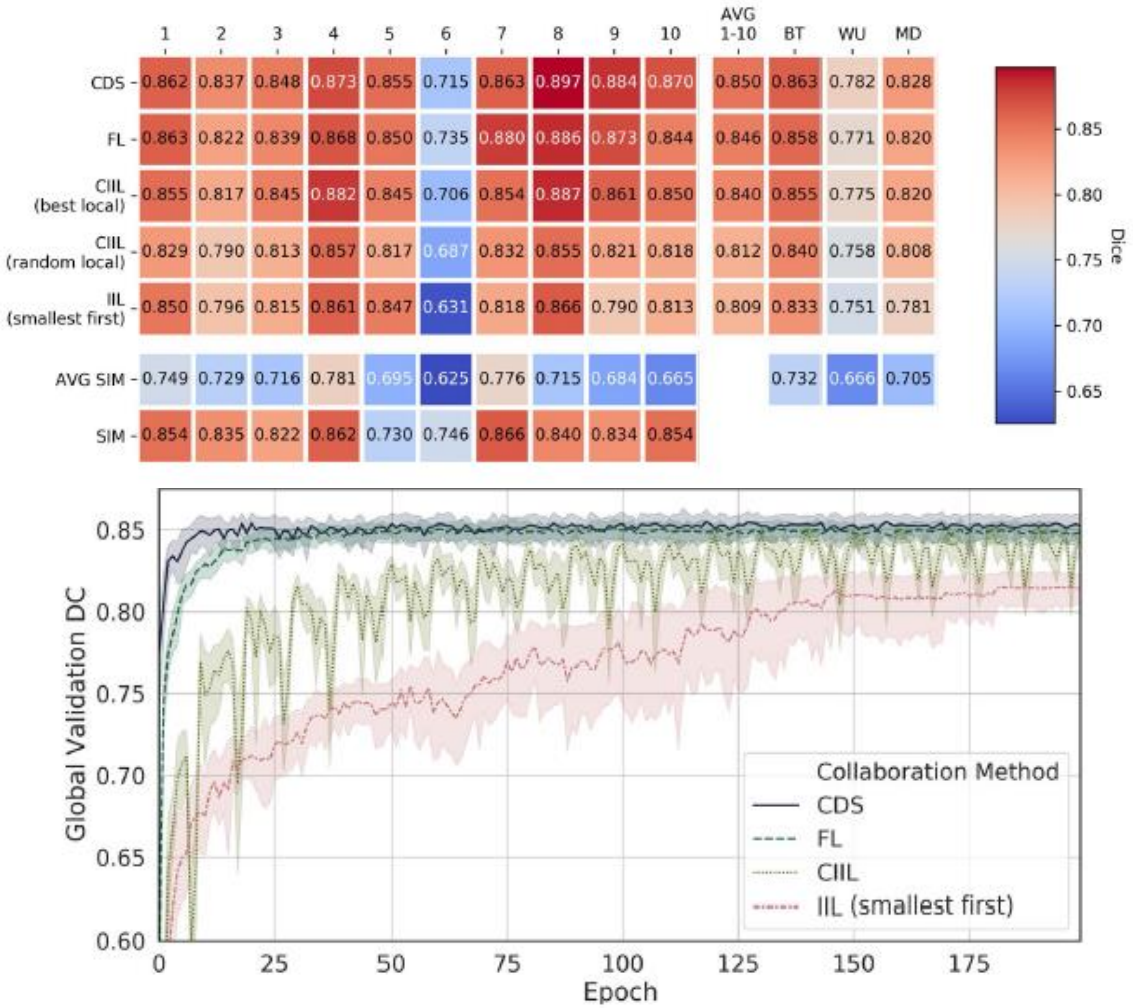
관련연구

- CDS(중앙 집중형 모델)의 성능에 비해 CDS: acc 85%, CIIL: acc 84%의 정확도를 보이면서 기존 정확도에 비해 99%의 정확도 목표를 달성하였다는 연구 결과가 있음



관련연구

- CDS(중앙 집중형 모델)의 성능에 비해 CDS: acc 85%, CIIL: acc 84%의 정확도를 보이면서 기존 정확도에 비해 99%의 정확도 목표를 달성하였다는 연구 결과가 있음



관련연구

- 2021년 5월 게재된 Yiqiang Chen, Jindong Wang의 FedHealth연구는 스마트폰에 있는 데이터(행동 감지)를 연합 학습하는 시나리오를 보여주는 연구로 연합 학습에 전이학습을 추가하여 고성능의 Activity 인식 모델을 달성하였음
- 이 연구는 웨어러블 헬스케어 분야의 최초의 연합 전이 학습 프레임워크를 보여줬다는 점, 고성능의 모델을 달성했다는 점에 의미가 있음
- 연구의 한계는 웨어러블 디바이스 데이터가 과거 기계학습에 이용되었던 Human Activity 공개 데이터셋을 사용했기에 실제 상황에 적용되는지 검증되지 않았다는 것

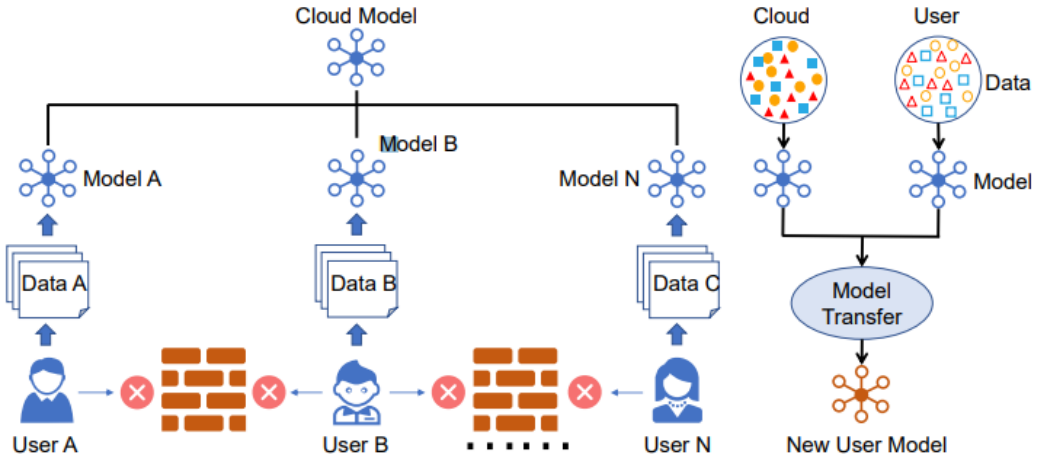


Figure 2: Overview of the FedHealth framework. "User" represents organizations

Table 2: Classification accuracy (%) of the test subject

Subject	KNN	SVM	RF	NoFed	FedHealth
P1	83.8	81.9	87.5	94.5	**98.8**
P2	86.5	96.9	93.3	94.5	**98.8**
P3	92.2	97.2	88.9	93.4	**100.0**
P4	83.1	95.9	91.0	95.5	**99.4**
P5	90.5	98.6	91.6	92.6	**100.0**
AVG	87.2	94.1	90.5	94.1	**99.4**

연합학습과 신약개발

- MELLODDY(Machine Learning Ledger Orchestration for Drug Discovery)
  - 2019년 6월 3일 부터 2022년까지 수행되는 유럽의 프로젝트로 10개의 제약회사(10억개 이상의 데이터 제공)와 7개의 데이터 과학 회사, AI 회사, 공공 연구 그룹 및 대학 파트너들이 데이터 처리 및 분석을 지원하는 프로젝트로 IMI(**Innovative Medicines Initiative**)로 부터 2천만$의 자금 지원을 받았음
  - 제약 회사의 화학 라이브러리에 기계학습 방법을 사용하여 약물 발견 및 개발을 위한 유용한 화합물을 예측하는 정확도 높은 모델을 생성하는 플랫폼을 개발하는 것을 목표로 함
  - 제약회사는 회사들이 보유한 화합물 라이브러리 데이터를 제공하고, owkin에서는 블록체인 기술을 제공함으로써 개인정보 보호 기능을 달성하며, Nvidia에서는 연합 학습 프레임워크 기술을 제공하여 플랫폼을 개발하는 중
  - 제약회사의 보유 데이터 이외에도 ChEMBL DB v25 공공 화합물 라이브러리 데이터를 사용하여 180만개의 화합물 정보를 추가 사용함

제약 파트너

AMGEN, astellas, AstraZeneca, BAYER, Boehringer Ingelheim, gsk, janssen, MERCK, NOVARTIS, SERVIER

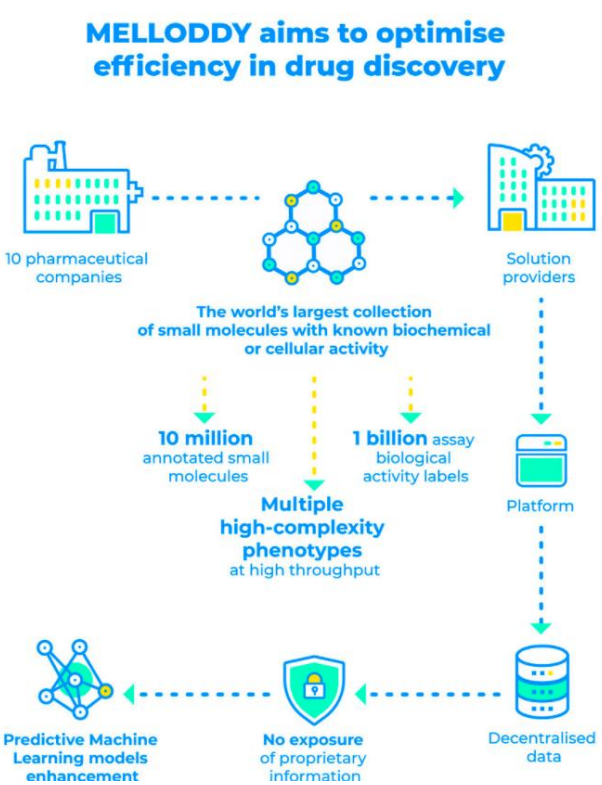
공공 파트너

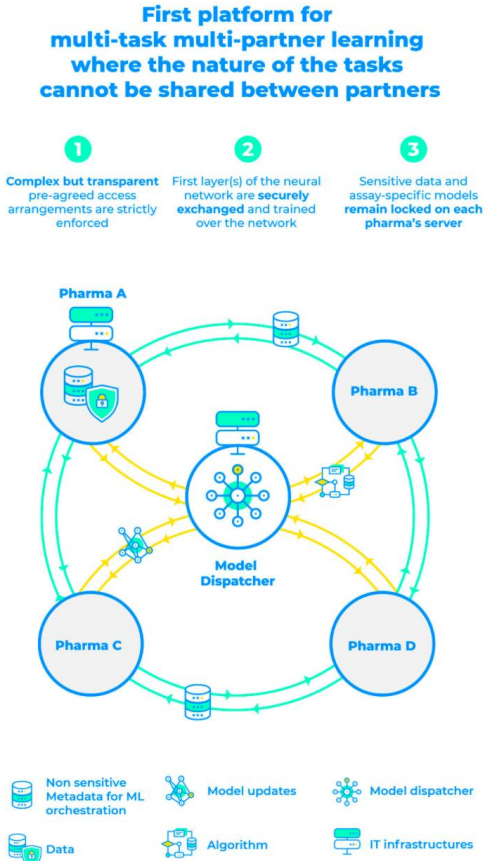
IKTOS, KUBERMATIC, KU LEUVEN, NVIDIA, OWKIN, Substra Foundation

출처: https://www.melloddy.eu/infographic

연합학습과 신약개발

- MELLODDY(Machine Learning Ledger Orchestration for Drug Discovery)
  - 약물 발견 예측 모델을 훈련하고 평가할 수 있는 연합학습 프레임워크를 구축하는 것이 목표
  - 멀티태스킹 예측 기계 학습 알고리즘을 포함한 'MELLODDY' 파이프라인에서 컨소시엄들의 방대한 데이터로 신약 개발에 가장 효과적인 후보물질을 도출할 수 있음

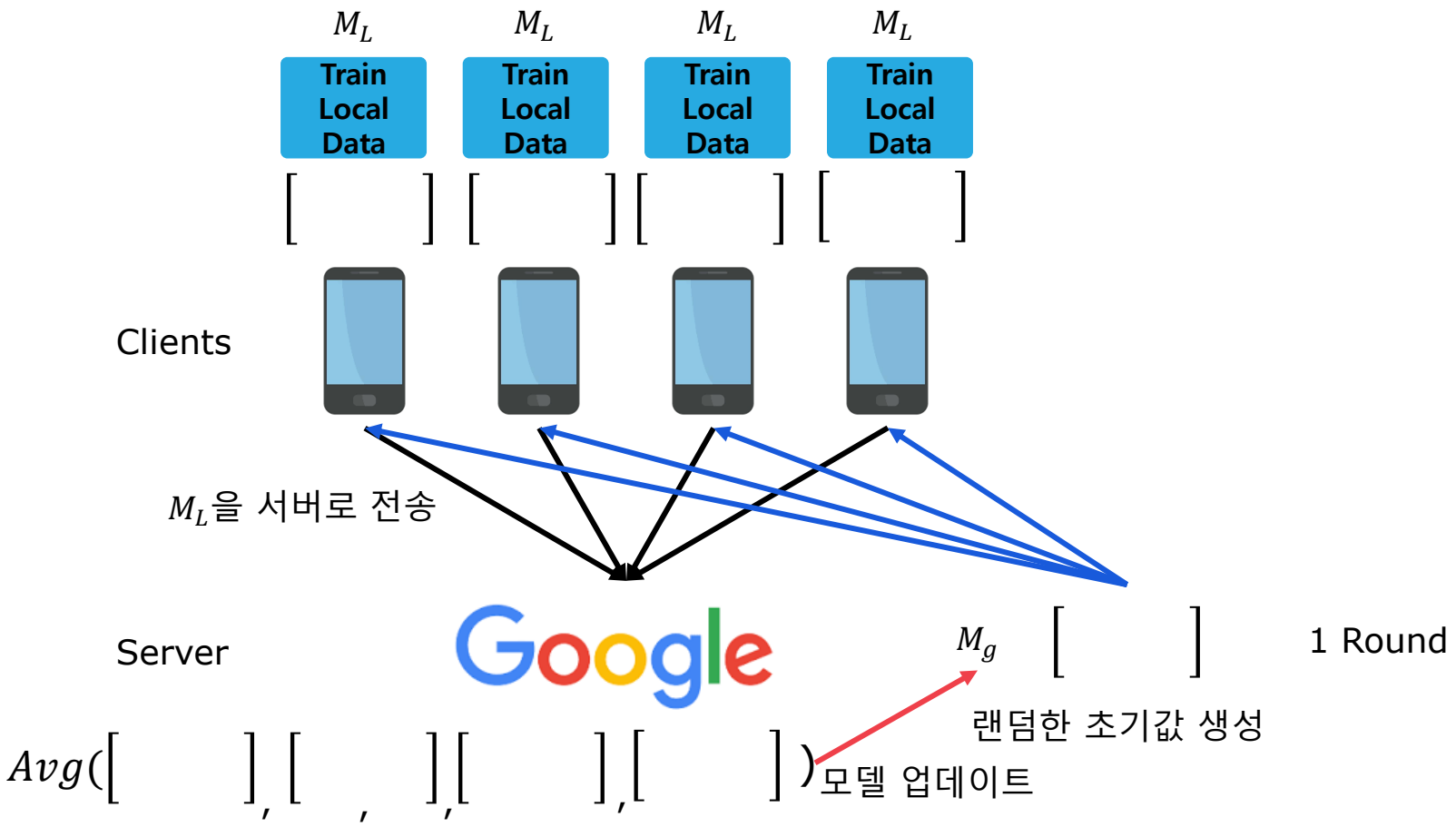




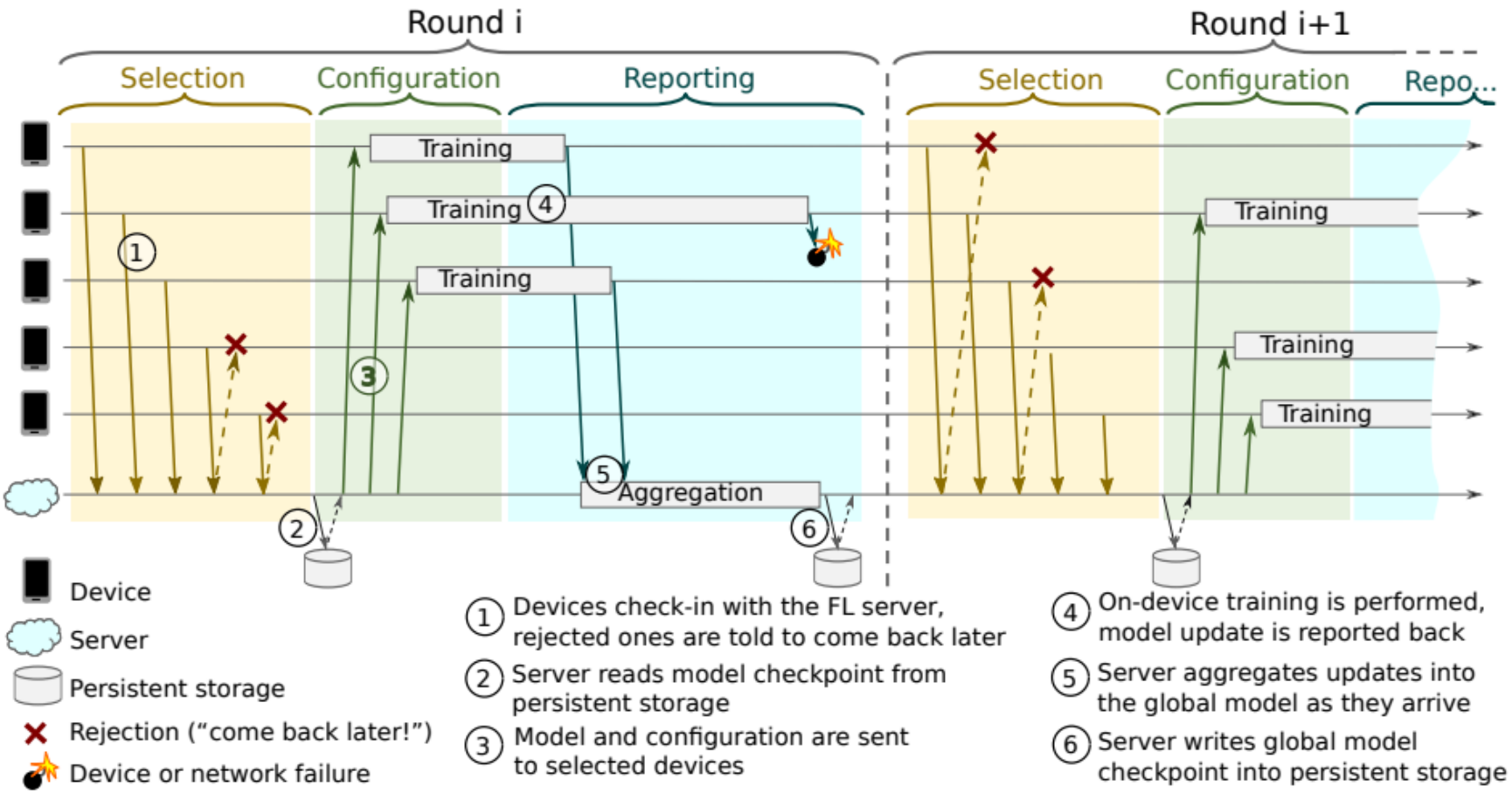
출처: https://www.melloddy.eu/infographic

연합학습 프로세스

- 모델(Mg)을 훈련하는데 분산되어 저장된 데이터의 프라이버시를 침해하지 않고, 즉 데이터를 하나로 모으지 않고 학습하는 것을 위한 프로세스



연합학습 프로세스


Round i
Round i+1
Selection
Configuration
Reporting
Selection
Configuration
Repo...
Training
Training
Training
Training
Training
Training
Aggregation
1
2
3
4
5
6
Device
Server
Persistent storage
Rejection (“come back later!”)
Device or network failure
1 Devices check-in with the FL server, rejected ones are told to come back later
2 Server reads model checkpoint from persistent storage
3 Model and configuration are sent to selected devices
4 On-device training is performed, model update is reported back
5 Server aggregates updates into the global model as they arrive
6 Server writes global model checkpoint into persistent storage

연합학습

- 분류 문제 기준으로 생각해보면,

$$MAX\left(\frac{1}{n}\sum_{i=1}^{n} P(y_i|x_i,w)\right) \qquad MIN\left(\frac{1}{n}\sum_{i=1}^{n} -\log(P(y_i|x_i,w))\right)$$

$$f_i(w) = loss(x_i, y_i, w) \qquad \min_{w\in\mathbb{R}} f(w), \quad f(w) = \frac{1}{n}\sum_{i=1}^{n} f_i(w)$$

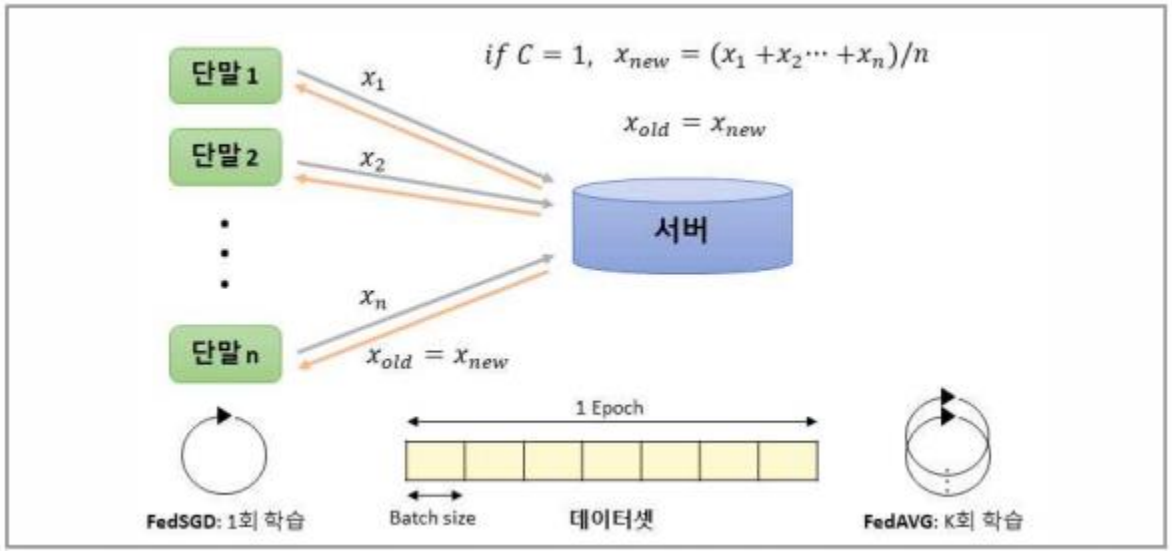
$$f(w) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(w) \qquad F_k(w) = \frac{1}{n_k}\sum_{i\in P_k} f_i(w)$$

- 여기서 K는 client의 수, F_k는 각 클라이언트별 Loss를 의미하며, P_k는 클라이언트 k가 가진 데이터들을 의미함
- $g(k)$ 는 K클라이언트의 그라디언트를 의미함 (FedSGD)

$$w_{t+1} = w_t - \eta\Delta f(w_t) = w_t - \eta\sum_{k=1}^{K} \frac{n_k}{n} g(k)$$

연합학습 알고리즘

- FedSGD
  - 각 단말은 한번 학습한 파라미터 값(gradient, w)를 서버로 전달, 서버는 모든 클라이언트의 파라미터의 평균을 계산하여 글로벌 파라미터를 업데이트하고 다시 모든 단말로 전달, 이 과정은 파라미터 수렴 조건이 만족될때까지 반복(C는 참여할 단말의 수를 결정)
- FEDAVG
  - 각 단말이 일정한 횟수 K만큼 반복적으로 학습을 수행한 후의 파라미터 값을 서버 로 전달하는 방식



출처: 연합학습 기술 동향 및 산업적 시사점, ETRI 2020.06

연합학습 알고리즘

- FedSGD, FEDAVG

Algorithm 1 `FederatedAveraging`. The K clients ar
indexed by k; B is the local minibatch size, E is the numbe
of local epochs, and η is the learning rate.

Server executes:
initialize w_0
for each round $t = 1, 2, \dots$ **do**
$m \leftarrow \max(C \cdot K, 1)$
$S_t \leftarrow$ (random set of m clients)
for each client $k \in S_t$ **in parallel do**
$w_{t+1}^k \leftarrow \text{ClientUpdate}(k, w_t)$
$w_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} w_{t+1}^k$

ClientUpdate(k, w): *// Run on client k*
$\mathcal{B} \leftarrow$ (split $\mathcal{P}_k$ into batches of size B)
for each local epoch i from 1 to E **do**
for batch $b \in \mathcal{B}$ **do**
$w \leftarrow w - \eta \nabla \ell(w; b)$
return w to server

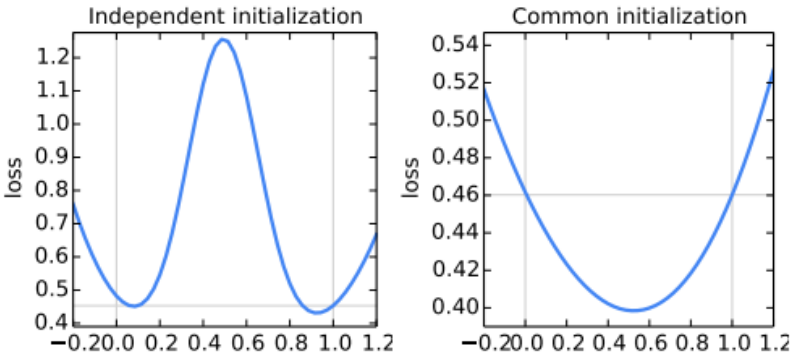


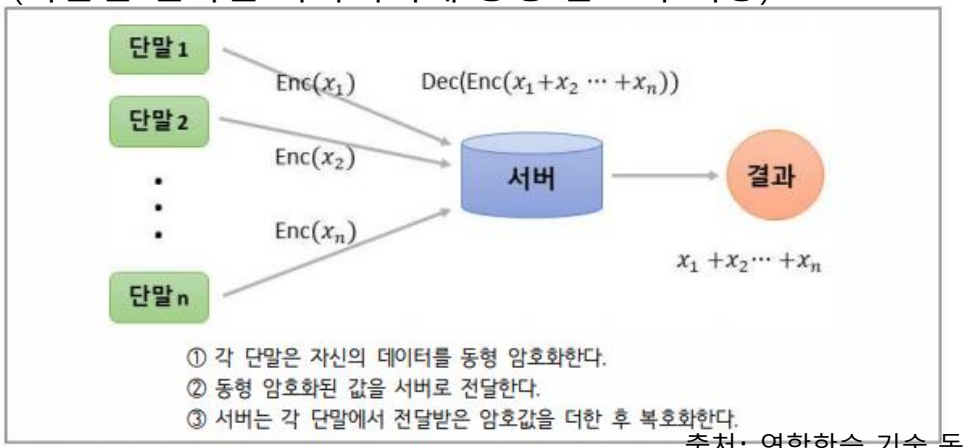
Table 2: Number of communication rounds to reach a target accuracy for `FedAvg`, versus `FedSGD` (first row, $E = 1$ and $B = \infty$). The u column gives $u = En/(KB)$, the expected number of updates per round.

MNIST CNN, 99% ACCURACY					
CNN	E	B	u	IID	NON-IID
FEDSGD	1	∞	1	626	483
FEDAVG	5	∞	5	179 (3.5×)	1000 (0.5×)
FEDAVG	1	50	12	65 (9.6×)	600 (0.8×)
FEDAVG	20	∞	20	234 (2.7×)	672 (0.7×)
FEDAVG	1	10	60	34 (18.4×)	350 (1.4×)
FEDAVG	5	50	60	29 (21.6×)	334 (1.4×)
FEDAVG	20	50	240	32 (19.6×)	426 (1.1×)
FEDAVG	5	10	300	20 (31.3×)	229 (2.1×)
FEDAVG	20	10	1200	18 (34.8×)	173 (2.8×)
SHAKESPEARE LSTM, 54% ACCURACY					
LSTM	E	B	u	IID	NON-IID
FEDSGD	1	∞	1.0	2488	3906
FEDAVG	1	50	1.5	1635 (1.5×)	549 (7.1×)
FEDAVG	5	∞	5.0	613 (4.1×)	597 (6.5×)
FEDAVG	1	10	7.4	460 (5.4×)	164 (23.8×)
FEDAVG	5	50	7.4	401 (6.2×)	152 (25.7×)
FEDAVG	5	10	37.1	192 (13.0×)	41 (95.3×)

출처: https://arxiv.org/pdf/1602.05629.pdf

연합학습 데이터 보호

- 차등정보보호
  - 2016년 Dework가 제안한 기술로 차등정보보호는 원본 데이터에 수학적 노이즈를 추가하여 프라이버시 노출 위험을 낮추는 기술임
  - 하나의 개인정보가 전체 자료에 추가로 포함될 때 증가하는 노출 위험을 '차등정보보호'라고 정의하고 이를 수학적으로 측정하는 방법을 제안
  - 차등정보보호 기술은 FedSDG, FedAVG 등 연합학습 알고리즘을 사용하여 단말에서의 학습 결과인 파라미터에 노이즈를 추가해 프라이버시 노출을 방지함
- 동형암호
  - 동형암호는 암호화된 데이터를 복호화 없이도 연산할 수 있는 암호기술
  - 원본 데이터를 암호화한 상태에서 각종 연산을 했을 때, 그 결과가 암호화하지 않은 상태의 연산 결과와 동일하게 나오는 암호 알고리즘
  - FedSDG, FedAVG 등 연합학습 알고리즘은 동형암호를 추가 사용함으로써 보안성 을 한층 강화할 수 있음 (학습한 결과인 파라미터에 동형 암호화 적용)



출처: 연합학습 기술 동향 및 산업적 시사점, ETRI 2020.06

연합학습 주요 이슈

- 학습 모델 파라미터 업데이트 공격
  - 클라이언트의 업데이트 정보에 손상을 가하거나 파라미터 값을 편향되도록 공격
- 학습 데이터 공격
  - 데이터 자체를 오염시키는 공격
- 회피 공격
  - 미세한 노이즈를 학습 데이터에 추가하여 데이터를 교란하는 방법

구분	주요 내용
통신 비용	- 로컬 단말과 중앙서버 간의 통신은 연합학습을 위해 구성된 전체 네트워크(연합 네트워크)에서 중요한 병목 현상을 유발 - 실제로 연합 네트워크는 수백만 대의 스마트폰으로 구성되는 경우, 네트워크 통신 속도는 로컬 컴퓨터보다 수십 배 느릴 수 있음 - 글로벌 모델을 업데이트하고 단말과 서버 간 데이터 전달 과정에서 통신 효율적인 방법을 개발해야 함 - 이때, 고려해야 할 두 가지 요소는 단말과 서버간 통신 횟수와 회당 전송되는 데이터 크기를 줄이는 것
시스템 이질성	- 연합학습에 참여한 로컬 단말(장치)는 저장공간, 계산 및 통신 성능, 배터리 수준 등이 매우 다양함 - 각 단말의 시스템과 네트워크의 제약으로 인해서 특정 시간에 참여할 수 있는 단말의 수가 신뢰할 수 없거나 학습 과정에서 사라질 수 있음 - 따라서 참여한 단말의 수의 변화, 이기종 하드웨어 등에 강인한 연합학습 메커니즘이 개발되어야 함
통계적 이질성	- 연합학습은 참여한 단말이 수집한 데이터가 독립적이고 동일한 확률 분포(IID: independent and identically distributed)라고 가정함 - 그러나 스마트폰 사용자의 사용 언어, 데이터 연결점의 개수 등이 다를 수 있어 모델 생성 및 분석 과정에서 복잡성이 매우 높아질 수 있음 - 각 단말에서 수집하는 데이터의 통계적 이질성을 다룰 수 있는 연합학습 방법 연구가 필요함
프라이버시 문제	- 연합학습의 프라이버시 보호를 강화하기 위해서 차등정보보호, 동형암호, 안전한 다자간 계산 등의 알고리즘을 추가 사용하고 있으나, 이것은 학습 모델의 성능 저하나 시스템의 효율성을 낮추는 문제가 발생함 - 이론적으로나 경험에 비추어 프라이버시 강화와 시스템 효율성 간 최적의 균형점을 찾는 방안 모색 필요

출처: 연합학습 기술 동향 및 산업적 시사점, ETRI 2020.06

설치법

- Pysyft(syft=0.2.9버전 환경)
  - conda create -n pysyft python=3.7.10
  - conda activate pysyft
  - conda install jupyter notebook==5.7.8 tornado==4.5.3
  - conda install pytorch==1.4.0 torchvision==0.5.0 -c pytorch
  - pip install syft==0.2.9
  - Pip install ipykernel
  - Pip install ipynb
  - python3 -m ipykernel install --user --name pysyft --display-name "pysyft"
  - conda install pytorch==1.6.0 torchvision==0.7.0 cpuonly -c pytorch
- Pysyft2(syft=0.5.0버전 환경)
  - Conda create –n pysyft2 python=3.9.5 –y
  - Conda activate pysyft2
  - Pip install syft
  - Pip install ipykernel
  - python3 -m ipykernel install --user --name pysyft2 --display-name "pysyft2"
  - pip install jupyter #notebook에러 날때
  - Pip install ipynb

Q & A

Q & A