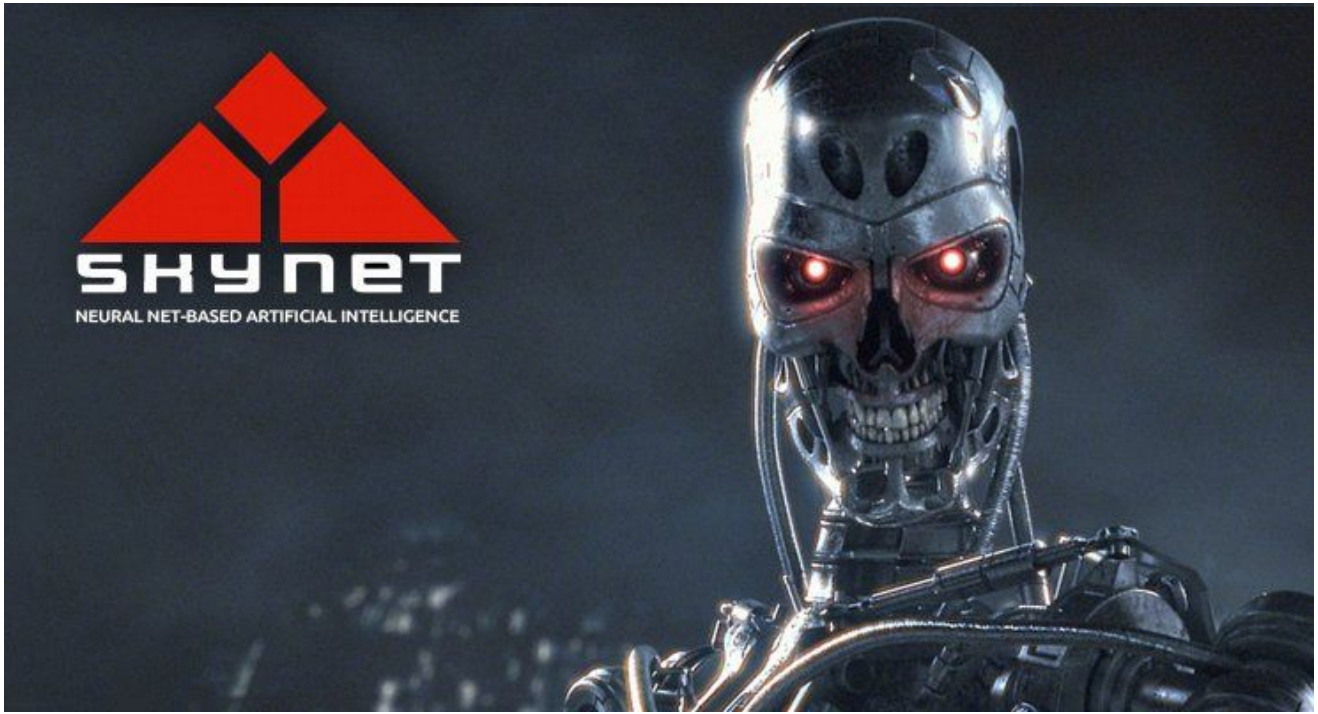
https://goo.gl/6qFTEB

한눈에 보는 머신 러닝

박 종 민

1. 머신러닝



https://minnov8.com/2015/01/31/minnov8-gang-298-welcome-to-skynet/

1. 머신러닝

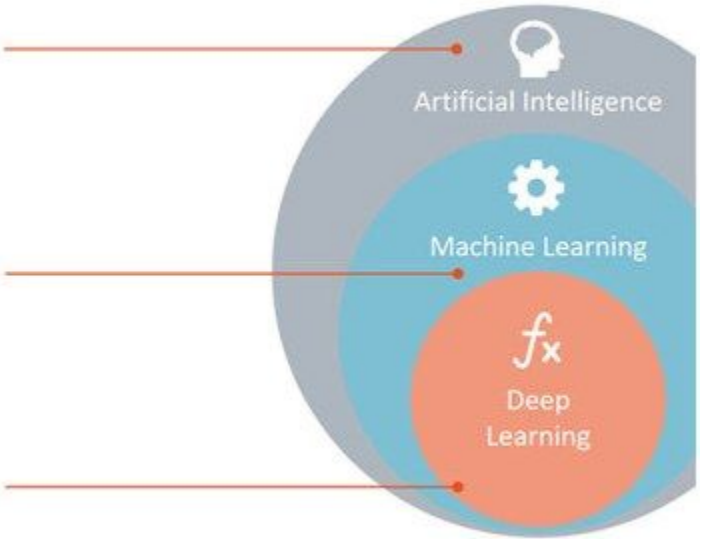
https://memegenerator.net/instance/72049399/matrix-architect-i-am-the-architect

1. 머신러닝

Artificial Intelligence
Any technique which enables computers to mimic human behavior.

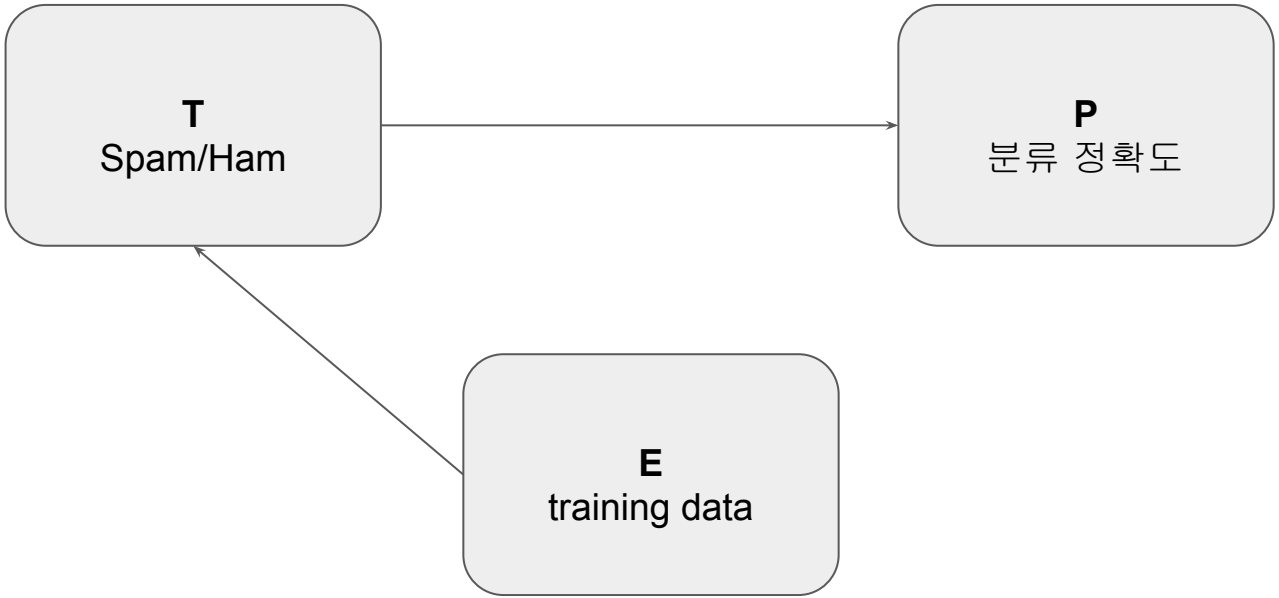
Machine Learning
Subset of AI techniques which use statistical methods to enable machines to improve with experiences.

Deep Learning
Subset of ML which make the computation of multi-layer neural networks feasible.



https://www.kdnuggets.com/2017/07/rapidminer-ai-machine-learning-deep-learning.html

1. 머신러닝



1. 머신러닝

1. 기존 솔루션은 많은 수동 조정과 규칙이 필요 -> 하나의 머신러닝 모델이 코드를 간단히 더 잘 수행할 수 있음
2. 전통적인 방법으로 해결 불가능 -> 머신러닝은 찾을 수 있음
3. 유동적인 환경
4. 복잡한 문제와 대량의 데이터에서 통찰 얻기

2. 머신러닝 시스템

1. Supervised Learning
2. Unsupervised Learning
3. Semi-supervised Learning
4. Reinforcement Learning
5. Batch Learning
6. Online Learning
7. Instance-based Learning
8. Model-based Learning

2-1. Supervised Learning

- 알고리즘에서 사용하는 training data에 **label**(=class)라는 답이 있음
- **Classification** and **Regression**
- k-Nearest Neighbors
- Linear Regression
- Logistic Regression
- SVM
- Decision Tree, Random Forests
- Neural Network

2-2. Unsupervised Learning

- training data에 **label**이 없음
- Clustering
  - k-Means
  - HCA
- Visualization and Dimensionality Reduction
  - PCA
  - t-SNE
- Anomaly Detection
- Association Rule learning
  - Apriori

2-3. Semisupervised Learning

- label이 있는 데이터는 조금, 대부분은 label이 없음
- Supervised Learning과 Unsupervised Learning의 조합
- 예: Facebook, Google Photos의 얼굴 태깅

출처: 남희석 Facebook

2-4. Reinforcement Learning

- **Environment**을 관찰하여 동작하는 **Agent**의 **Action**을 통해 **Reward**를 주는데 가장 큰 보상을 얻기 위한 **Policy**를 찾기 위해 스스로 학습하는 것
- 예: AlphaGo, 쿠키런 AI
  - 알파고(네이처지) https://goo.gl/MeFZmM
  - 쿠키런 AI(DEVIEW2016) https://www.slideshare.net/carpedm20/ai-67616630

2-5. Batch Learning

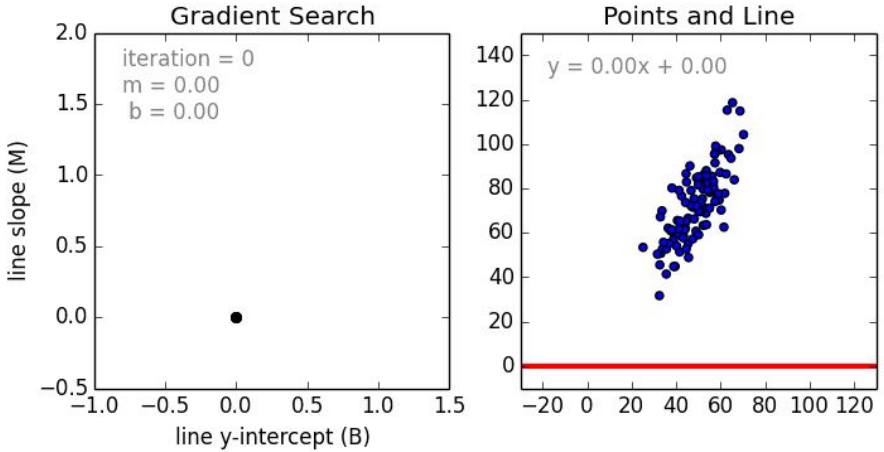
- 가지고 있는 데이터를 모두 사용하여 훈련시키는 방법
  - Offline Learning
  - 학습한 것을 적용만 할뿐, 추가 데이터에 대해 점진적으로 학습할 수 없음
  - 새로운 데이터에 대해 학습하려면, 이전 데이터를 포함한 전체 데이터를 사용하여 처음부터 다시 훈련
  - 많은 리소스를 필요, 데이터가 아주 크다면 유지하는 것이 불가능
  - 리소스가 제한된 시스템에서는 사용 불가 또는 심각한 문제를 발생

2-6. Online Learning

- 데이터를 순차적으로 하나 또는 mini-batch 단위로 훈련시킴
  - **CAUTION**: 보통 오프라인에서 진행, incremental learning 개념으로 생각
  - 단계별 학습이 빠르고, 리소스가 적게 필요함
  - 새로운 데이터를 학습할 때 전체 데이터가 아닌 새로운 것만 학습하면 됨
- **Out-of-core Learning** (외부 메모리 학습)
  - 컴퓨터 한 대의 메인 메모리에 들어갈 수 없는 아주 큰 데이터셋을 학습, 일부를 읽고 학습 반복
- **Learning Rate**: 변화하는 데이터에 얼마나 빠르게 적응하는가
  - High: 새로운 데이터에 빠르게 적응하지만 예전 데이터를 금방 잊어버림
  - Low: 새로운 데이터의 특이점에 적응이 느림
- 문제점: 새로운 데이터로 나쁜 데이터가 주입되었을 때, 성능이 점진적으로 감소
  - Microsoft 트위터 챗봇 인종차별: https://www.theverge.com/2016/3/24/11297050/tay-microsoft-chatbot-racist

2-7. Instance-based and Model-based Learning

- 사례 기반 학습
  - 시스템이 사례를 기억함으로써 학습, **similarity** 측정하여 새로운 데이터에 일반화
- 모델 기반 학습
  - 샘플의 모델을 만들어 예측에 사용하는 것
  - http://nbviewer.jupyter.org/github/rickiepark/handson-ml/blob/master/01_the_machine_learning_landscape.ipynb
  - https://github.com/llSourcell/linear_regression_live



3. 머신러닝이 풀어야할 과제 (데이터)

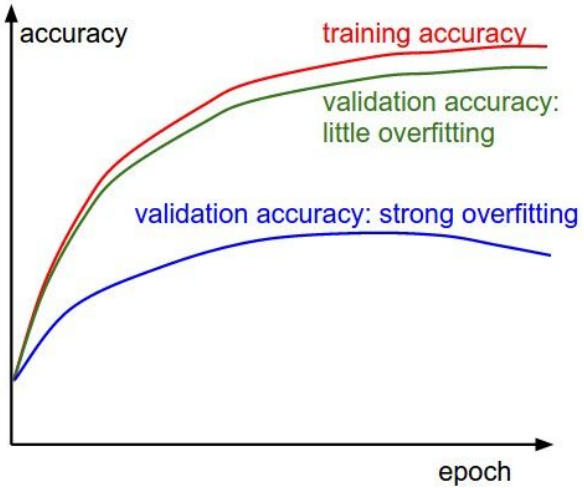
1. Not Enough ~~Minerals~~ Training Data
    a. Scaling to Very Very Large Corpora for Natural Language Disambiguation
    b. The Unreasonable Effectiveness of Data
    c. 충분한 데이터가 있으면 문제 해결이 쉽다. 그러나 작은 규모의 데이터가 여전히 매우 흔하고, 추가로 데이터를 모으는 것은 어렵고, 비용이 많이 발생하므로 알고리즘을 무시하지 말아야함
2. Without Representation
    a. 일반화를 하기 위해서는 샘플이 데이터 전체를 잘 대표해야 함
    b. 샘플이 작으면, sampling noise가 생김
    c. 샘플이 크더라도, sampling bias가 발생

3. 머신러닝이 풀어야할 과제 (데이터)

1. 낮은 품질의 데이터
    a. 일부 샘플이 outlier를 가진다면, 무시하거나 수동으로 고치는 것이 좋음
    b. 일부 샘플에 feature가 빠져있다면 feature 무시, 해당 샘플 무시, 값을 채울지, 모델을 분리 등을 결정
2. 관련 없는 특성
    a. 관련이 없는 특성으로 훈련을 한다면, 당연히 결과가 좋지 않을 것
    b. Feature Engineering: 훈련에 사용할 좋은 특성을 찾는 것
    c. Feature Selection: 가지고 있는 특성 중 훈련에 가장 유용한 특성을 선택
    d. Feature Extraction: 특성을 결합하여 더 유용한 특성을 만듦

3. 머신러닝이 풀어야할 과제 (알고리즘)

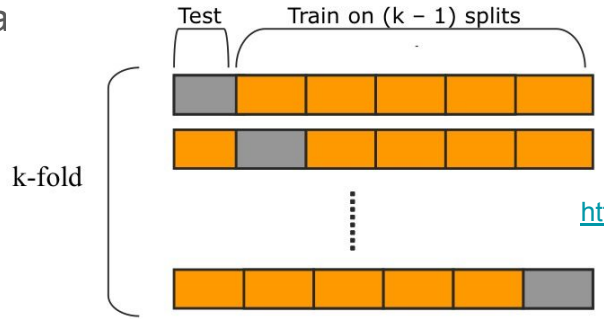
1. Overfitting
   a. 모델이 training data에 너무 잘 맞지만 일반성이 떨어진다는 뜻
   b. 파라미터 수가 적은 모델을 선택, 훈련 데이터 특성을 줄임, 모델에 제약을 가해 단순화
   c. 더 많은 training data
   d. outlier 나 anomaly 제거
2. Underfitting
   a. 모델이 너무 단순해서 제대로 학습하지 못할 때 발생
   b. 파라미터가 더 많은 모델을 선택
   c. 더 좋은 feature를 제공 (feature engineering)
   d. 모델의 제약을 줄임



http://cs231n.github.io/neural-networks-3/

4. 머신러닝 테스트와 검증

1. Training Set and Test Set
   a. 모델이 얼마나 잘 일반화 되는지 알기 위한 방법으로, 일반화 오차를 통해 모델의 정확도를 파악
   b. Training Error가 낮지만, Generalization Error가 높다면 Overfitting
2. Validation Set
   a. 모델과 하이퍼파라미터가 테스트 세트에 최적화 되어 생기는 문제를 해결
   b. Training Set와 Validation Set를 비교하면서 훈련하고, 마지막 단 한번 Test Set으로 모델 평가
3. Cross-Validation Method
   a. Tra



많은 양의 데이터를 뺏기지 않기 위해 사용

http://qingkaikong.blogspot.kr/2017/02/machine-learning-9-more-on-artificial.html

5. 연습문제 1

1. 머신러닝을 어떻게 정의할 수 있나요?
2. 머신러닝이 도움을 줄 수 있는 문제 유형 네 가지를 말해보세요.
3. 레이블된 훈련 세트란 무엇인가요?
4. 가장 널리 사용되는 지도 학습 작업 두 가지는 무엇인가요?
5. 보편적인 비지도 학습 작업 네 가지는 무엇인가요?
6. 사전 정보가 없는 여러 지형에서 로봇을 걸어가게 하려면 어떤 종류의 머신러닝 알고리즘을 사용할 수 있나요?

5. 연습문제 2

1. 고객을 여러 그룹으로 분할하려면 어떤 알고리즘을 사용해야 하나요?
2. 스팸 감지의 문제는 지도 학습과 비지도 학습 중 어떤 문제로 볼 수 있나요?
3. 온라인 학습 시스템이 무엇인가요?
4. 외부 메모리 학습이 무엇인가요?
5. 예측을 하기 위해 유사도 측정에 의존하는 학습 알고리즘은 무엇인가요?
6. 모델 파라미터와 학습 알고리즘의 하이퍼파라미터 사이에는 어떤 차이가 있나요?
7. 모델 기반 알고리즘을 찾는 것은 무엇인가요? 성공을 위해 이 알고리즘을 사용하는 가장 일반적인 전략은 무엇인가요? 예측은 어떻게 만드나요?

5. 연습문제 3

1. 머신러닝의 주요 도전 과제는 무엇인가요?
2. 모델이 훈련 데이터에서의 성능은 좋지만 새로운 샘플에서의 일반화 성능이 나쁘다면 어떤 문제가 있는 건가요? 가능한 해결책 세 가지는 무엇인가요?
3. 테스트 세트가 무엇이고 왜 사용해야 하나요?
4. 검증 세트의 목적은 무엇인가요?
5. 테스트 세트를 사용해 하이퍼파라미터를 튜닝하면 어떤 문제가 생기나요?
6. 교차 검증이 무엇이고, 왜 하나의 검증 세트보다 선호하나요?

6. 마무리

- 공짜 점심 없음 이론 (No Free Lunch)
  - 데이터에 관해 완벽하게 어떤 가정도 하지 않으면 한 모델을 다른 모델보다 선호할 근거가 없음
  - **경험하기 전**에 더 잘 맞을 것이라고 보장할 수 없음
  - 어떤 모델이 최선인지 아는 유일한 방법은 모든 모델을 **평가**
  - The Lack of A Priori Distinctions Between Learning Algorithms, D.Wolperts, 1996
- 참고자료
  - Hands-on Machine Learning with Scikit-Learn & Tensorflow
  - Introduction to Machine Learning with Python
  - https://tensorflow.blog/핸즈온-머신러닝/