ETRI Electronics and Telecommunications Research Institute

Machine Learning 101

A Tour of Machine Learning Algorithms

31 March 2017

Sung-Soo Kim

sungsoo@etri.re.kr

Smart Data Research Group

ETRI

I’m not a “*Machine Learning*” (ML) guy!

But…

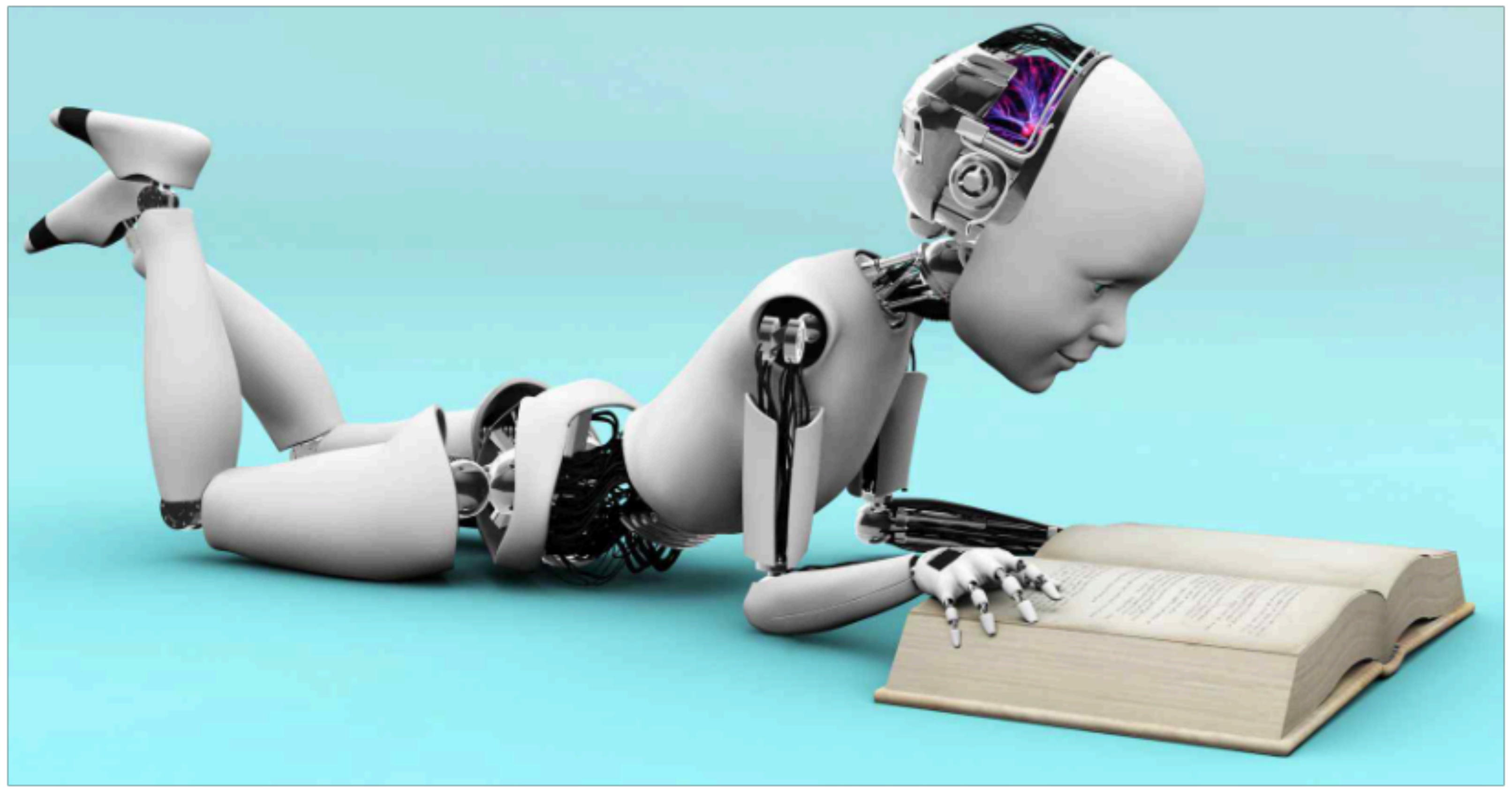
Maybe, I will be a ML guy!

Machine Learning in Robotics


Machine Learning

Machine Learning's Successes

- *Computer Vision*
  - Image inpainting/denoising, segmentation
  - object recognition/detection, scene understanding
- *Information Retrieval / NLP*
  - Text, audio, and image retrieval
  - Parsing, machine translation, text analysis
- *Speech Processing*
  - Speech recognition, voice identification
- *Robotics*
  - Autonomous car driving, planning, control
- *Computational Biology*
- *Cognitive Science*



Training/evaluating deep neural networks

Technique leading to many high-profile AI advances in recent years

Speech recognition/natural language processing

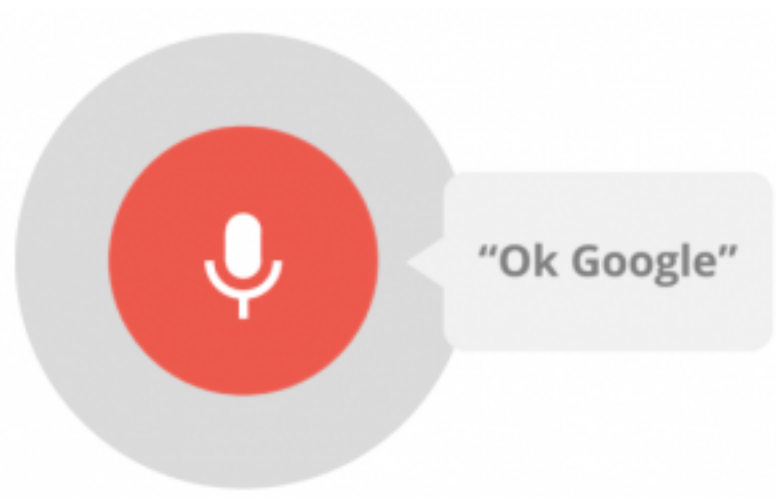
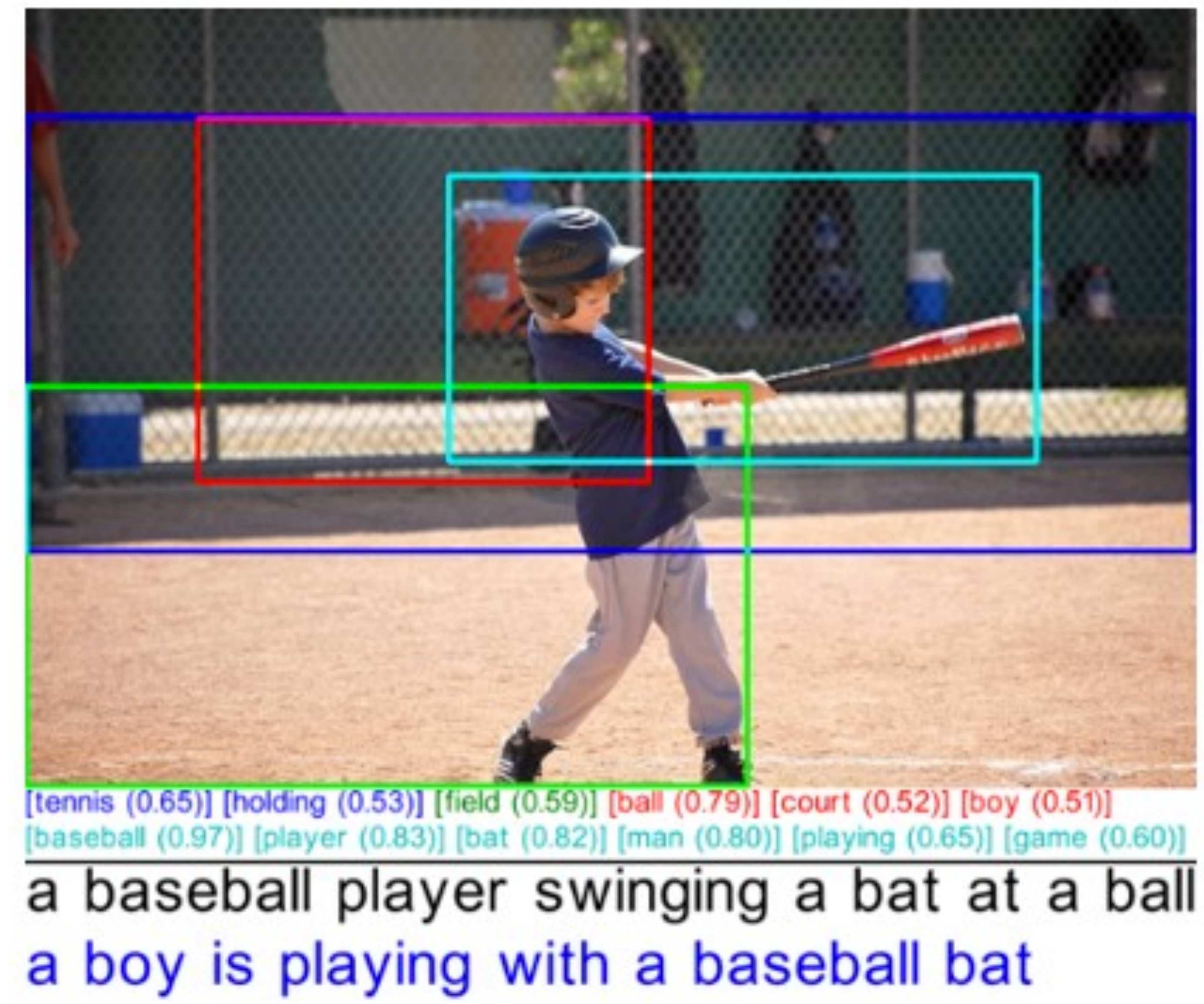
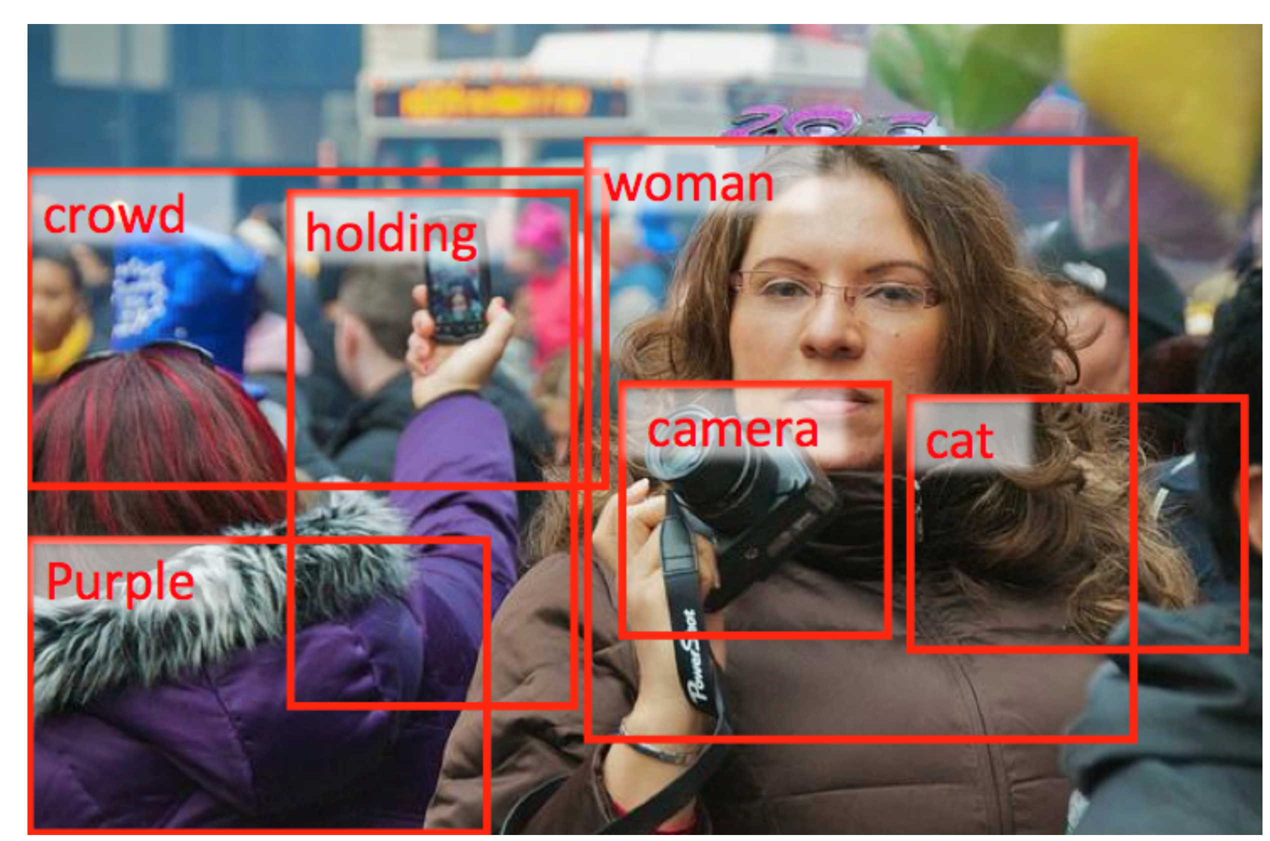


Image interpretation and understanding

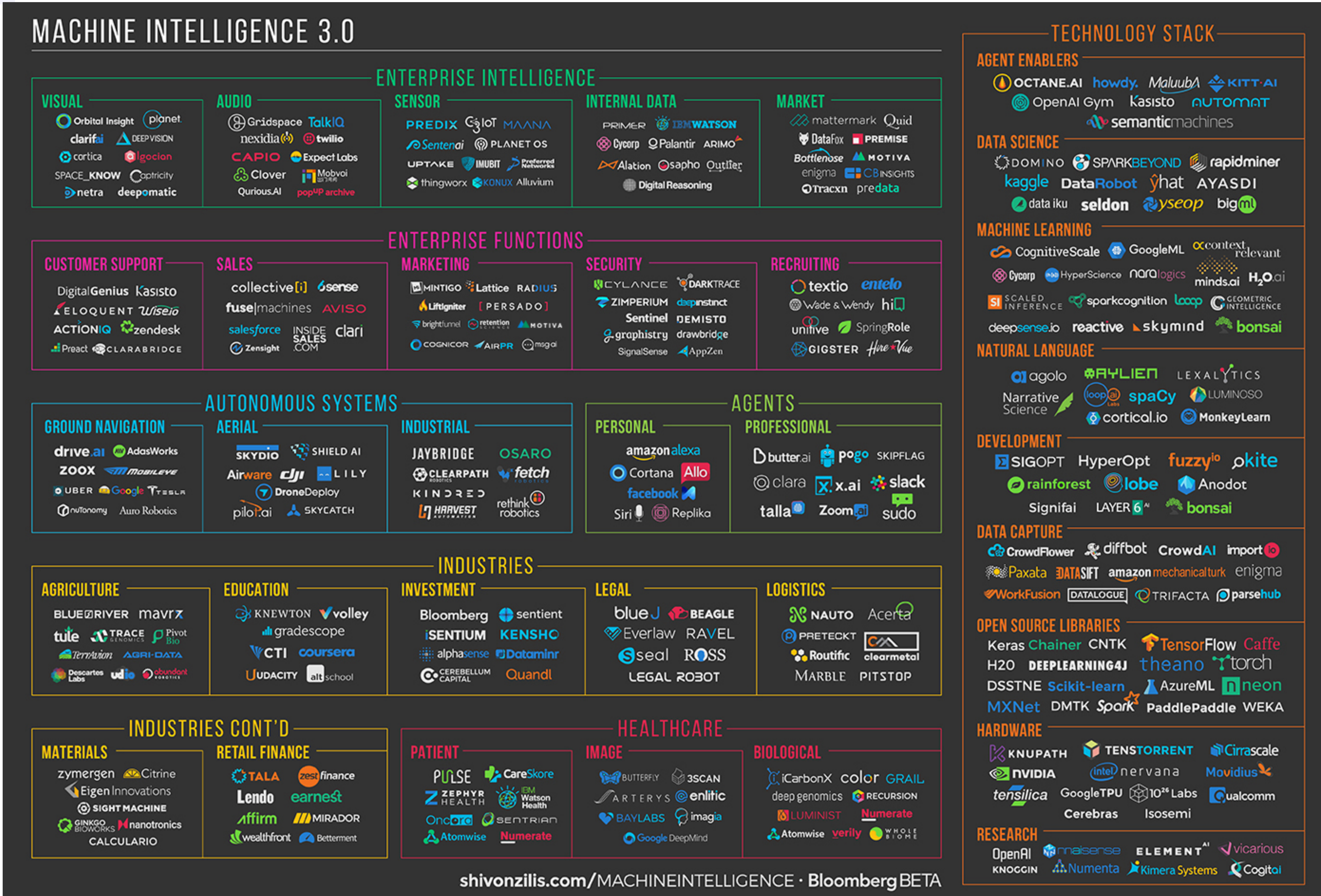




Top 10 Trends 2015 ~ 2017 [Gartner]

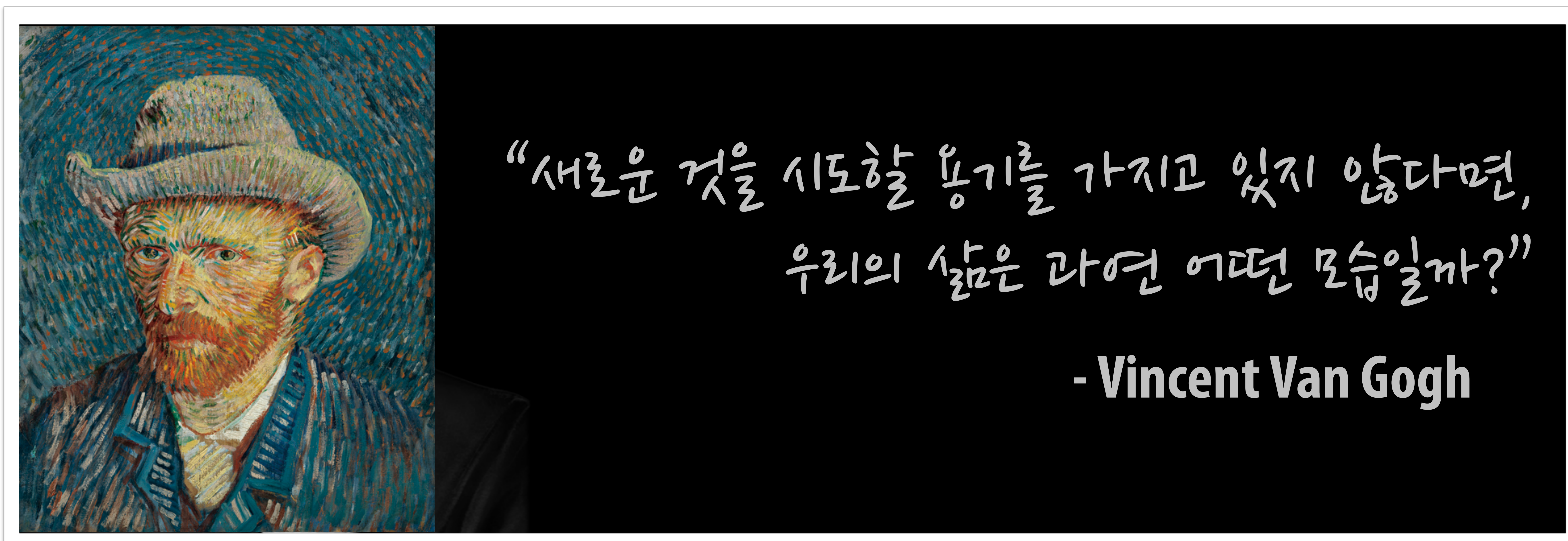
2015	2016	2017
Merging the Real World and Virtual World	**Digital Mesh**	**Intelligent**
Computing Everywhere	Device Mesh	Artificial Intelligence and Advanced Machine Learning
The Internet of Things	Continuous & Ambient UX	Intelligent Apps
3D Printing	3D Printing Materials	Intelligent Things
Intelligence Everywhere	**Smart Machines**	**Digital**
Advanced, Pervasive and Invisible Analytics	Information of Everything	Virtual Reality and Augmented Reality
Context-Rich Systems	Advanced Machine Learning	Digital Twins
Smart Machines	Autonomous Agents & Things	Blockchains and Distributed Ledgers
The New IT Reality Emerges	**New IT Reality**	**Mesh**
Cloud/Client Computing	Adaptive Security Architecture	Conversational Systems
Software-Defined Applications and Infrastructure	Advanced Systems Architecture	Digital Technology Platforms
Web-Scale IT	Mesh App & Service Architecture	Mesh App and Service Architecture
Risk-Based Security and Self-protection	IoT Architecture & Platforms	Adaptive Security Architecture

Machine Intelligence 3.0



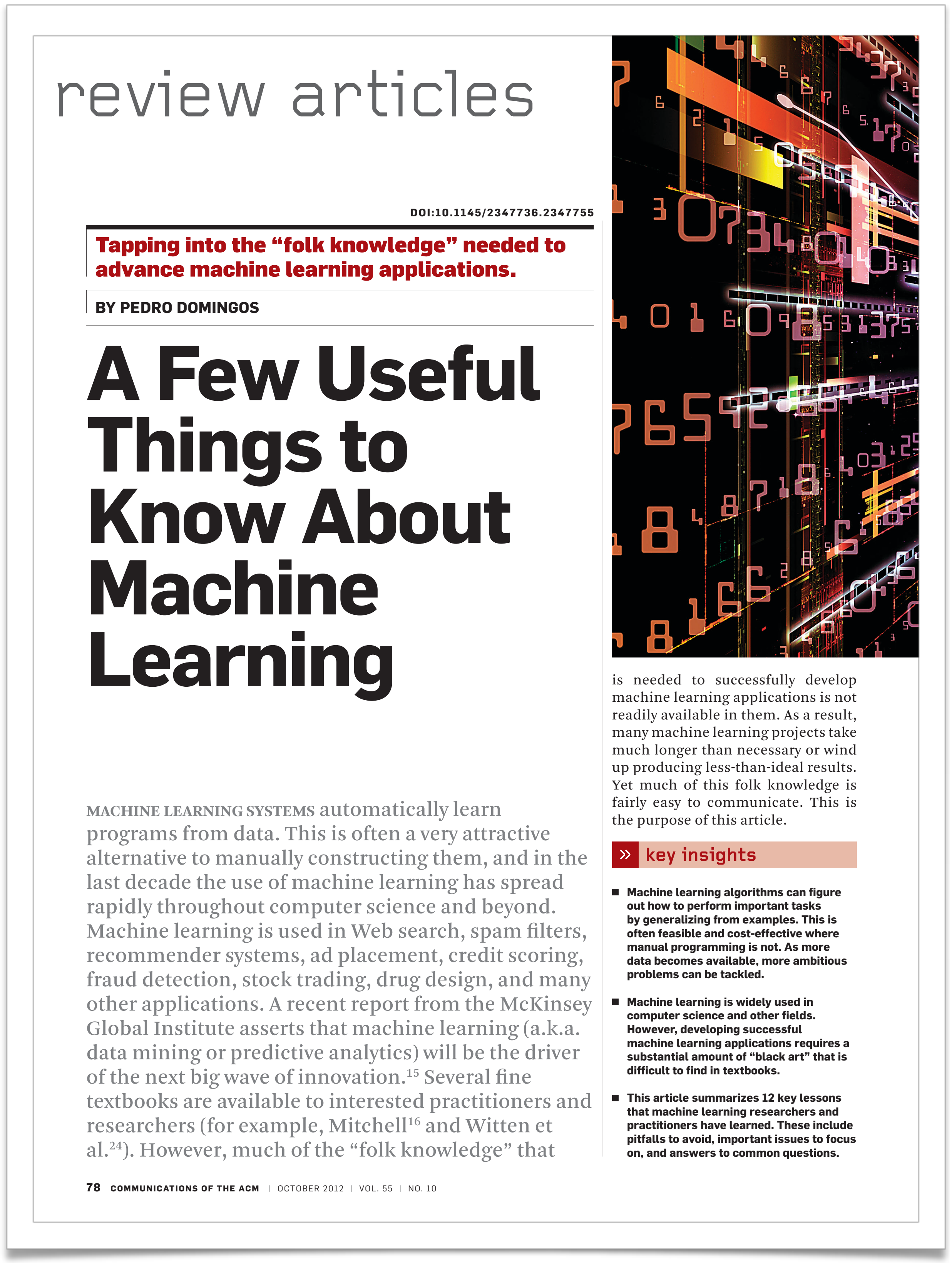
Outline

- What is Machine Learning?
- A Fast Tour of Machine Learning Algorithms
- A Few Useful Things to Know About Machine Learning
- Summary



References and Slide Credits

- M. I. Jordan and T. M. Mitchell, *Machine learning: Trends, perspectives, and prospects*, Science, Vol. 349 Issue 6245, July 2015.
- Predo Domingos, *A Few Useful Things to Know About Machine Learning*, Communications of the ACM, Vol. 55 No. 10, Oct. 2012.
- T. M. Mitchell, *Machine learning*, McGraw-Hill, 1997.



Main Takeaways

- **Takeaway #1**
  - **Machine learning is *not monolithic*.**

- **Takeaway #2**
  - **Deep learning has emerged as a technique with *strong advantages*, but also has important *drawbacks* as well.**

- **Takeaway #3**
  - **Machine learning will have *major implications* for what products will look like going forward.**

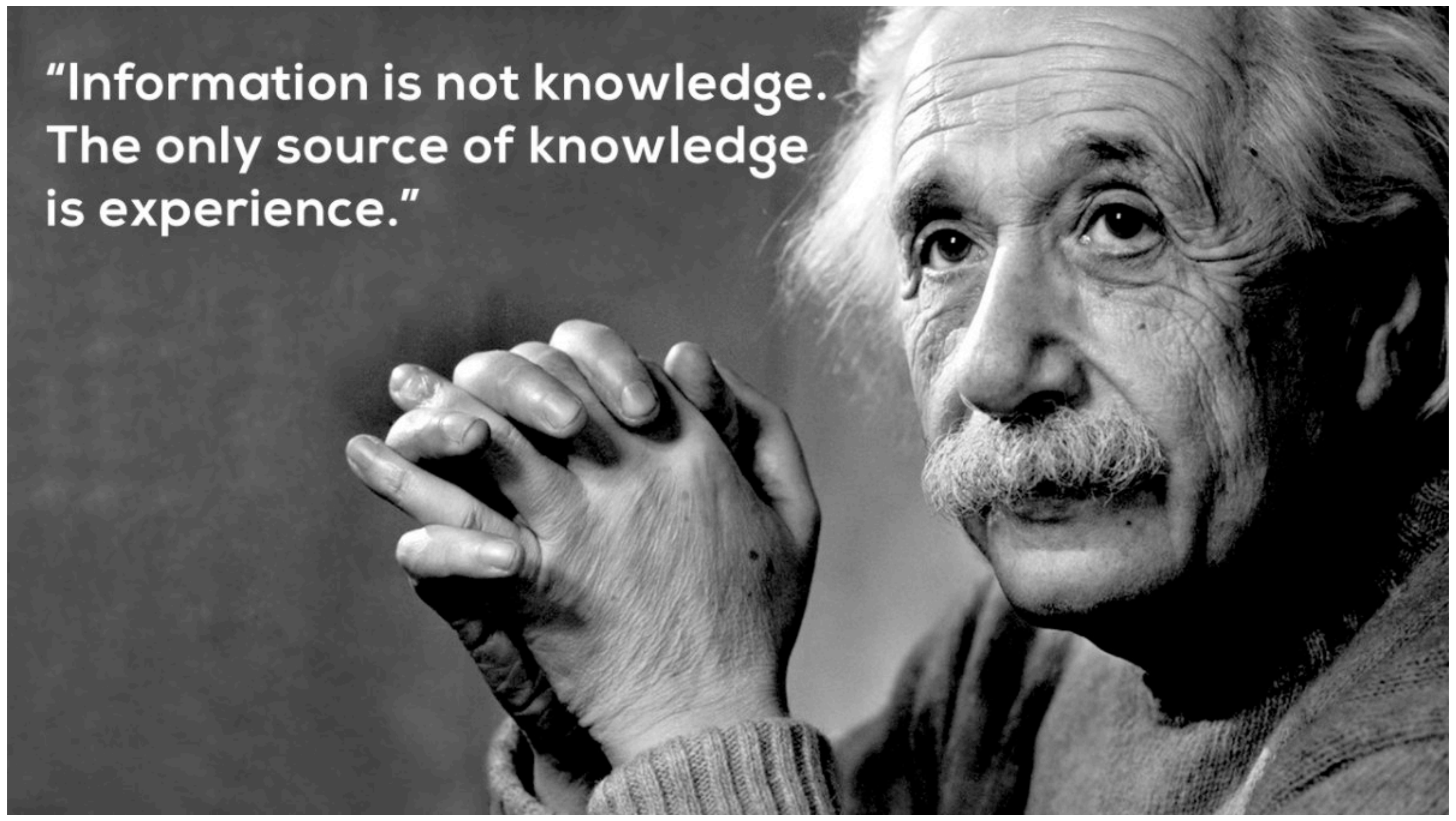
- **Takeaway #4**
  - **There are lot of "*folk knowledge*" that is needed to successfully develop machine learning applications, which is *not* readily available in ML textbooks.**

Fundamental Question

- Where does *knowledge* come from?

Evolution

Experience

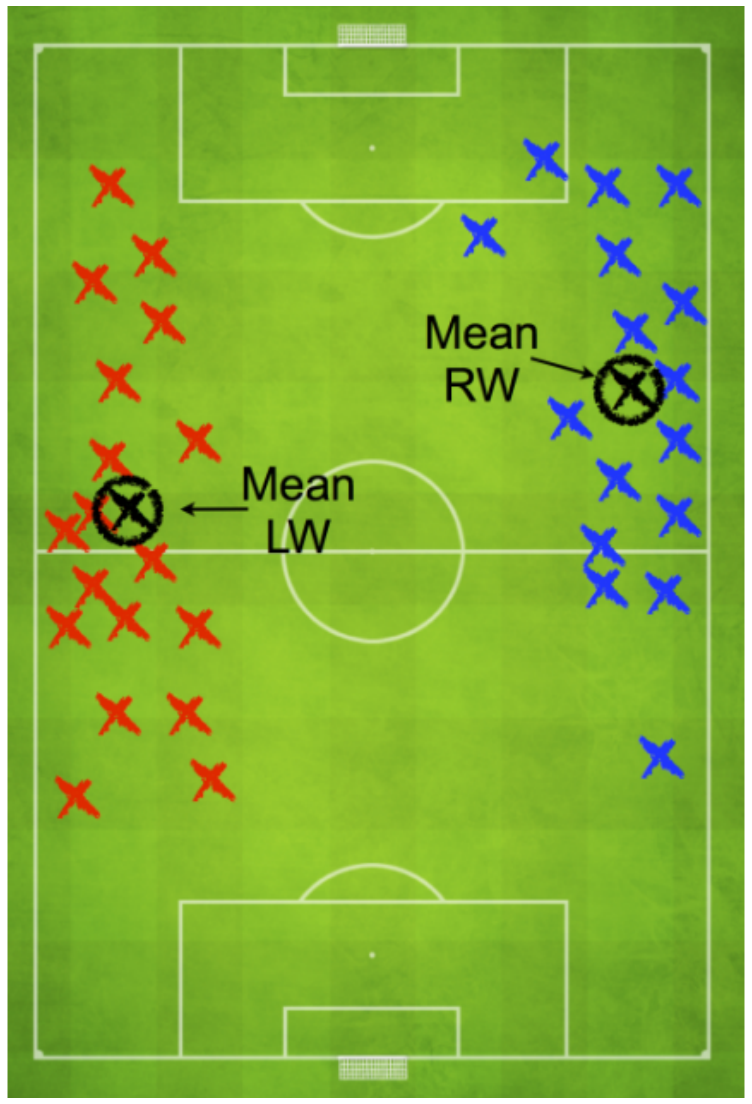


Culture

Computer

What is Machine Learning?

- Limitations of *explicit* programming
  - Spam filter: many rules
  - Automatic driving: too many rules
- Machine Learning
  - "Field of study that gives computers the ability to learn *without being explicitly programmed*" - Arthur Samuel (1959)
  - "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*" - T. Michell (1997)



Example: A program for soccer tactics

T: Win the game

P: Goals

E: (x) Player's movements
(y) Evaluation

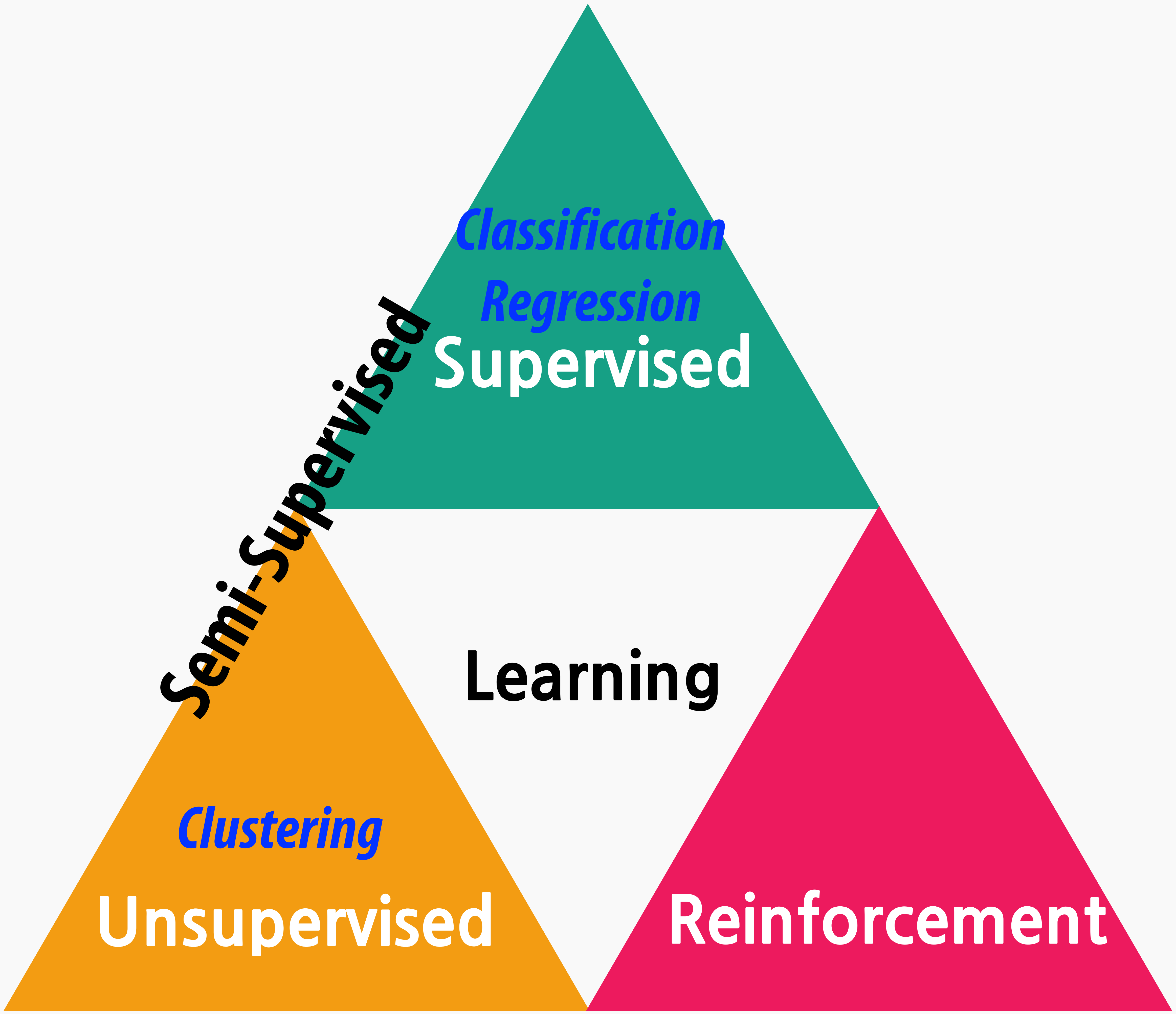
Research Fields

- Self-improving Systems
- Knowledge Discovery
- Data-Driven SW Design
- Automatic Programming

Machine Learning in Robotics

Three Types of Machine Learning

머신러닝의
세가지 타입



Supervised Learning

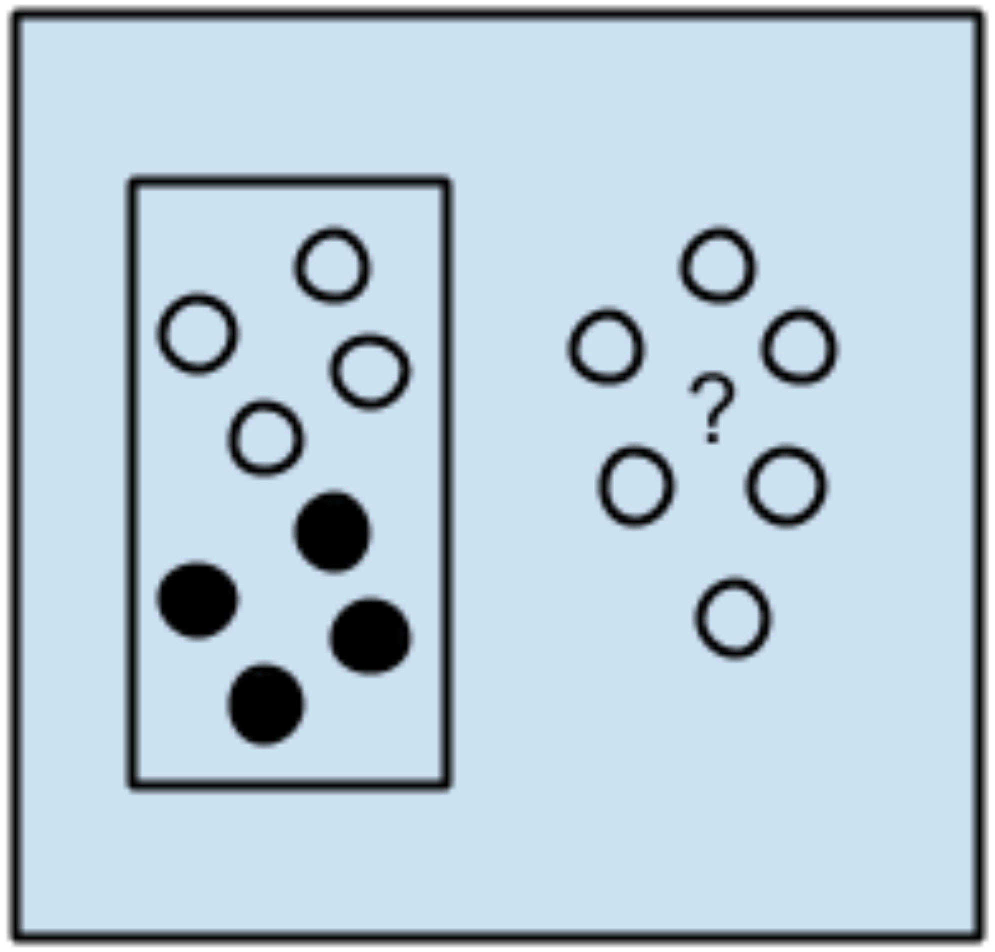
Supervised Learning

Input data is called training data and has a known label or result such as spam/not-spam or a stock price at a time.

A model is prepared through a training process in which it is required to make predictions and is corrected when those predictions are wrong. The training process continues until the model achieves a desired level of accuracy on the training data.

Example problems are classification and regression.

Example algorithms include Logistic Regression and the Back Propagation Neural Network.



Supervised Learning Algorithms

Unsupervised Learning

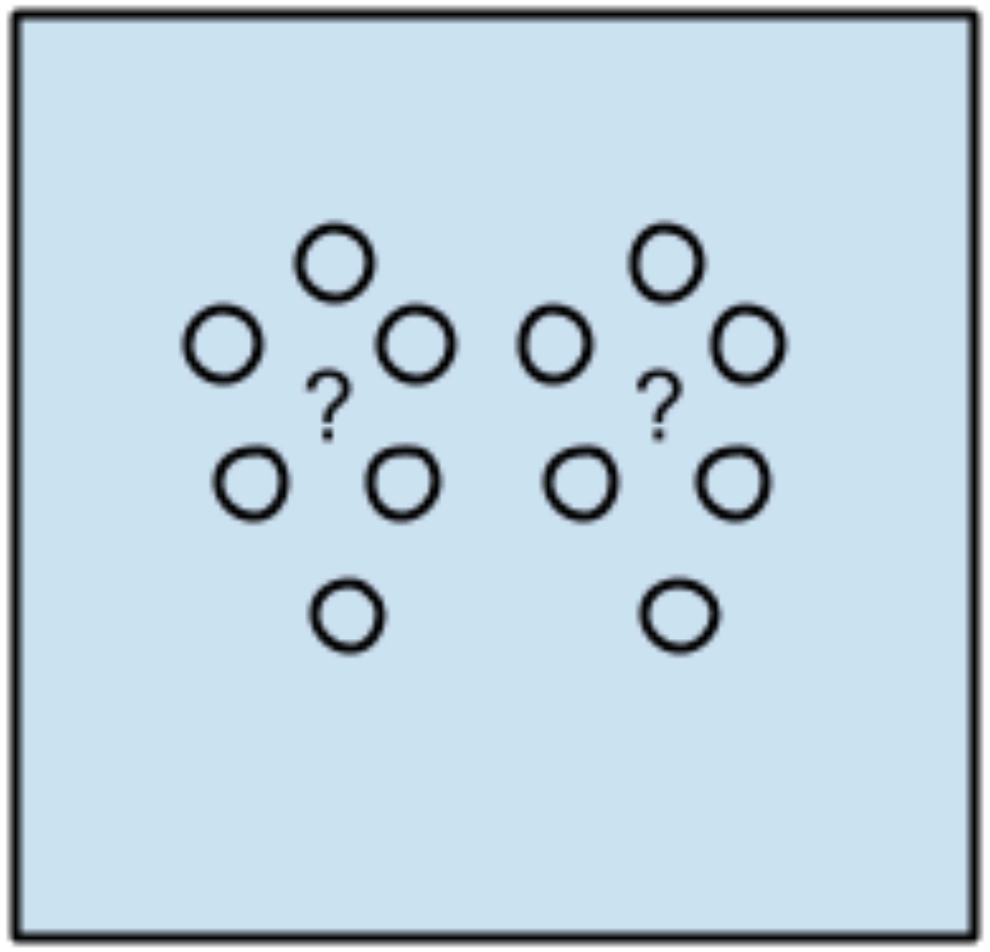
Unsupervised Learning

Input data is not labeled and does not have a known result.

A model is prepared by deducing structures present in the input data. This may be to extract general rules. It may be through a mathematical process to systematically reduce redundancy, or it may be to organize data by similarity.

Example problems are clustering, dimensionality reduction and association rule learning.

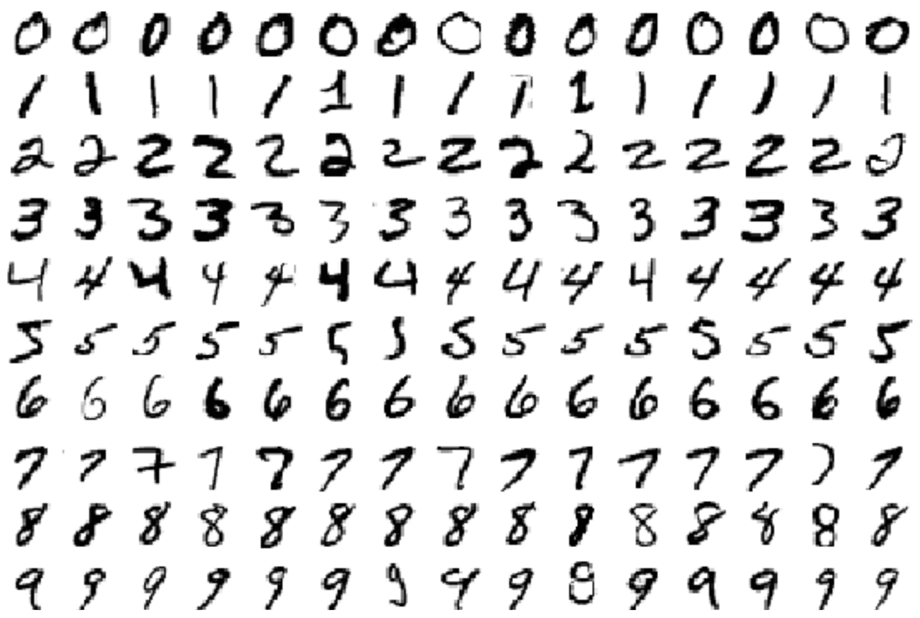
Example algorithms include: the Apriori algorithm and k-Means.



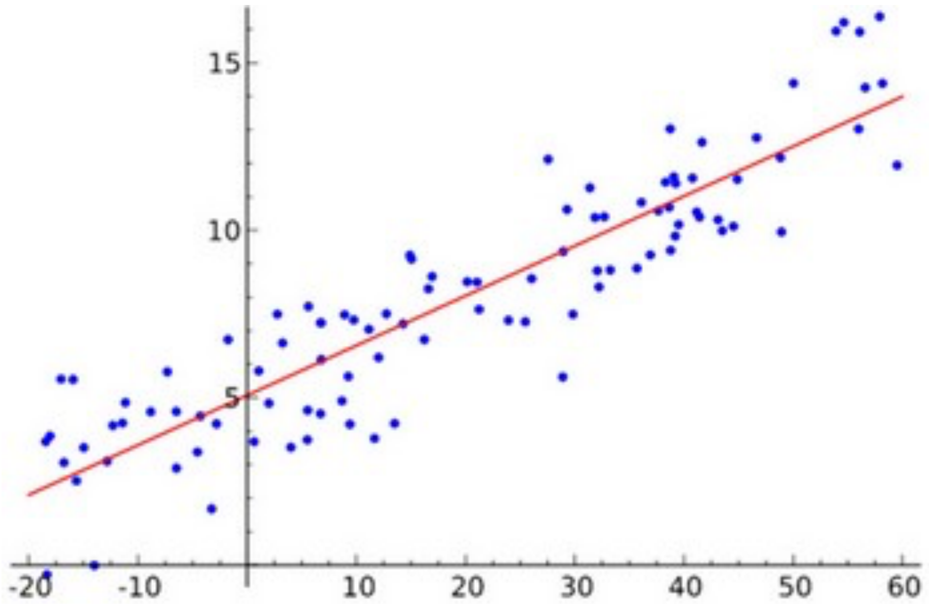
Unsupervised Learning Algorithms

What is Machine Learning?

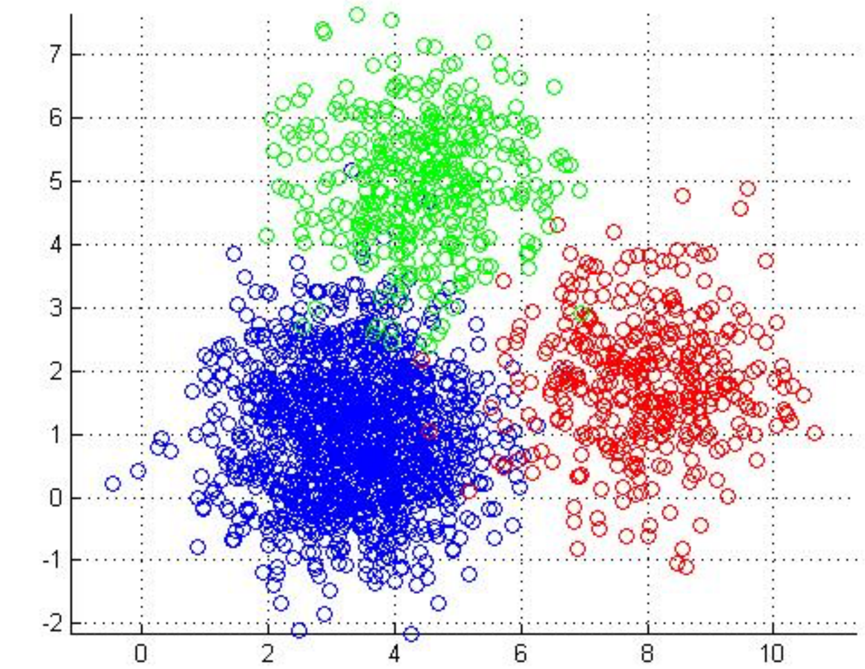
Classification



Regression

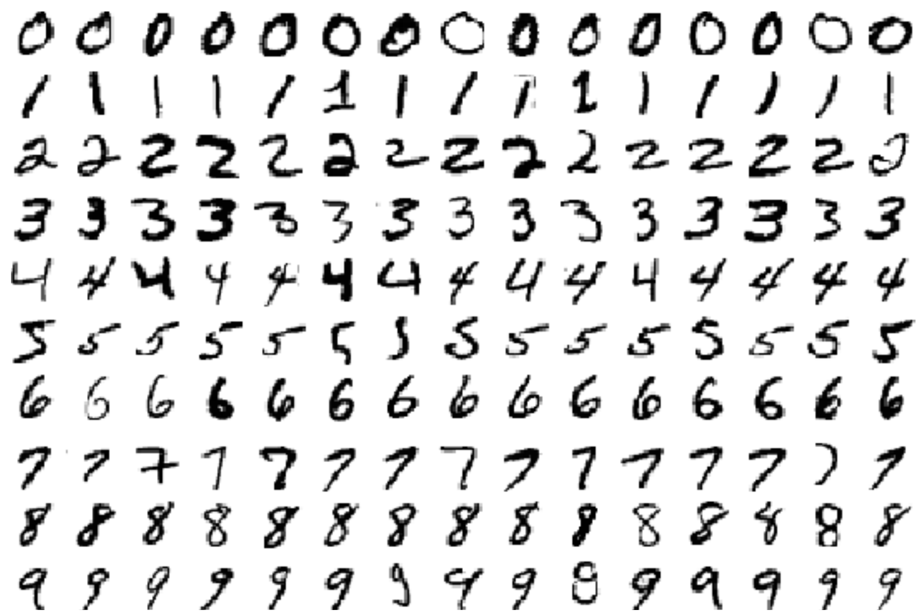
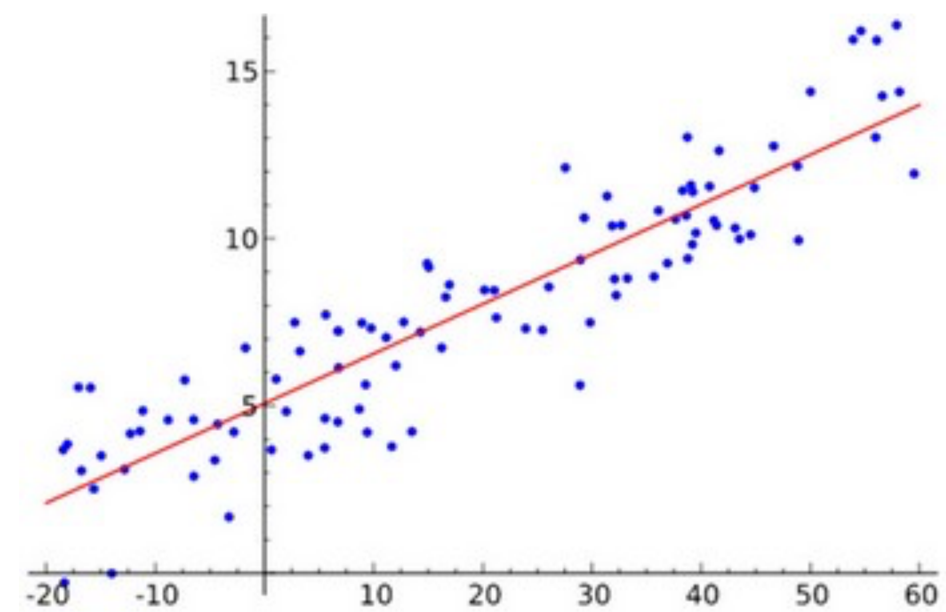
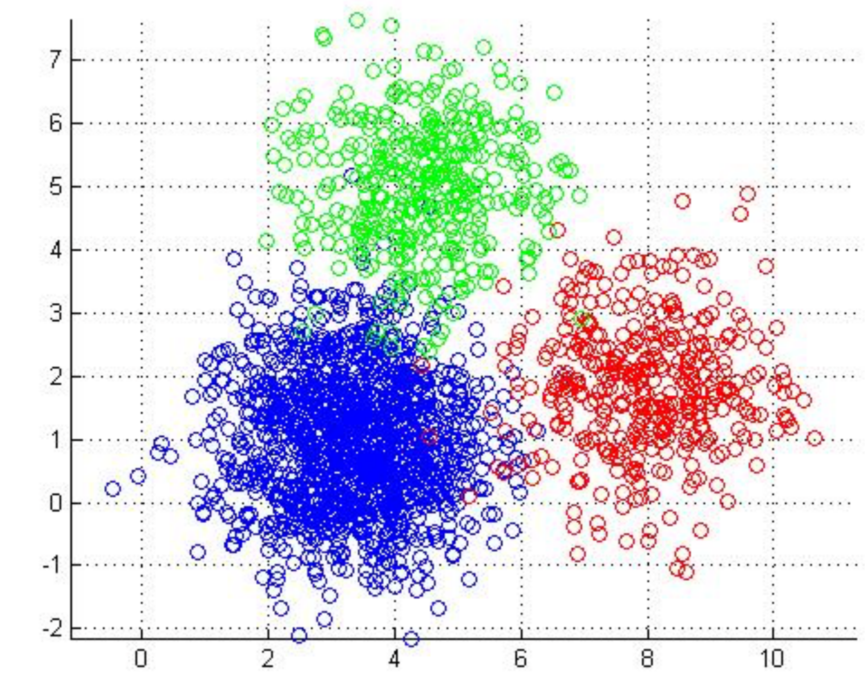


Clustering



"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***" - T. Michell (1997)

Supervised Learning, Unsupervised Learning

	Classification	Regression	Clustering
			
Task	discrete target values x: pixels (28*28) y: 0, 1, 2, 3, . . . , 9	real target values $x \in (-20, 60)$ $y \in \mathbb{R}$	no target values $x \in (0, 10) \times (-2, 7)$
Performance	0-1 loss function $L(\hat{y}, y) = I(\hat{y} \neq y)$	L2 loss function $L(f, \hat{f}) = \|\|f - \hat{f}\|\|_2^2$	$L(\hat{y}, y) = I(\hat{y} \neq y)$ $L(f, \hat{f}) = \|\|f - \hat{f}\|\|_2^2$
Experience	labeled data (pixels) ➤ (number)	labeled data (x) ➤ (y)	unlabeled data (x_1, x_2)

Semi-Supervised Learning

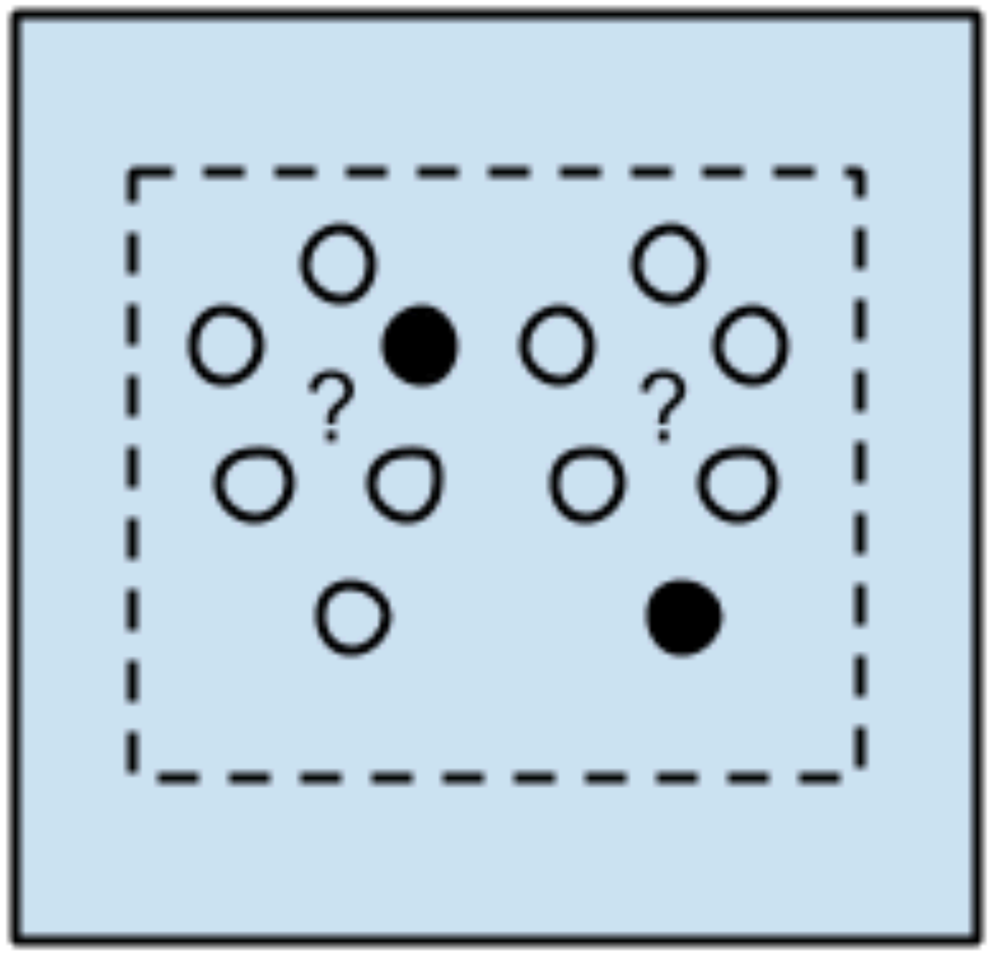
Semi-Supervised Learning

Input data is a mixture of labeled and unlabelled examples.

There is a desired prediction problem but the model must learn the structures to organize the data as well as make predictions.

Example problems are classification and regression.

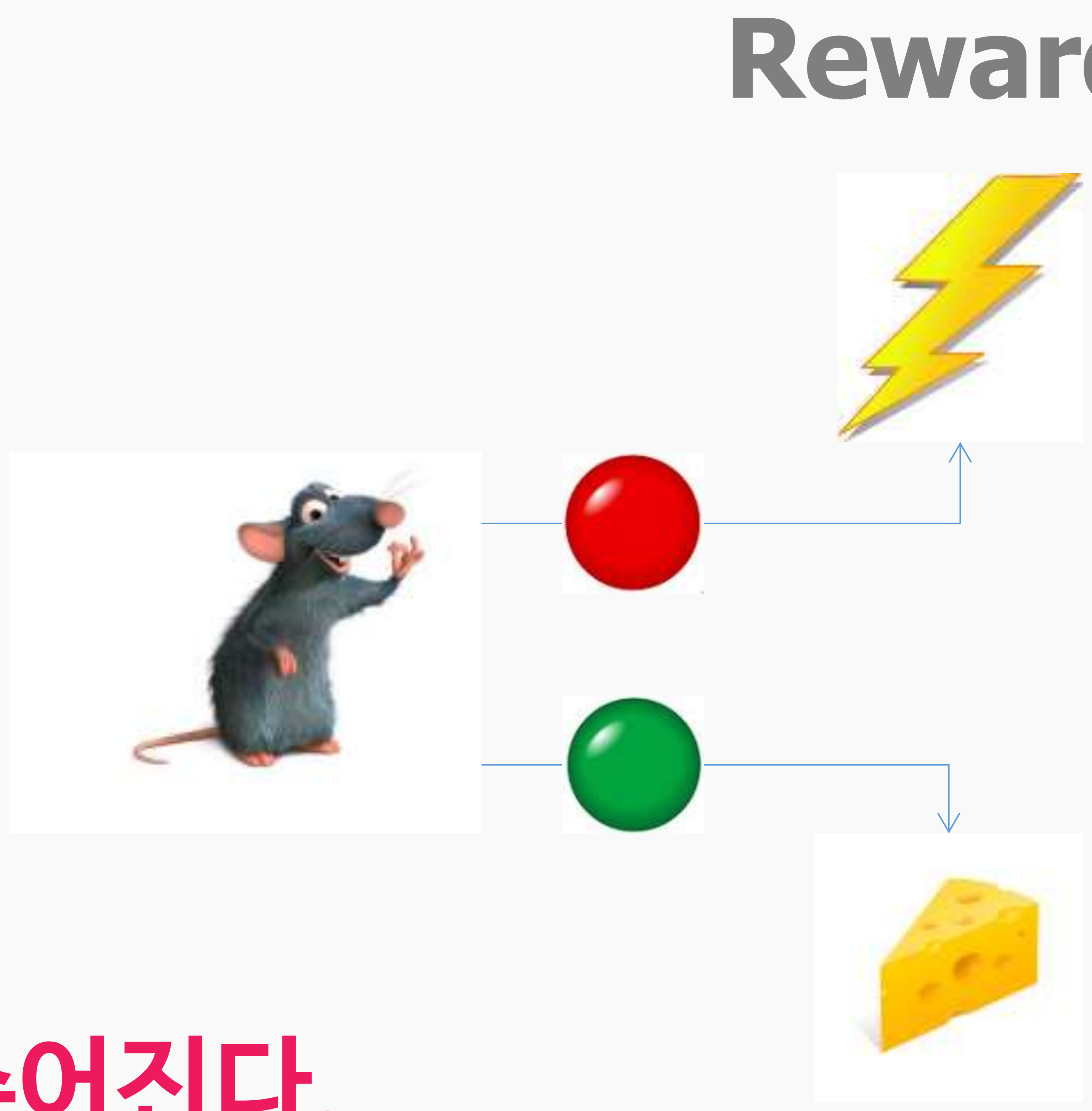
Example algorithms are extensions to other flexible methods that make assumptions about how to model the unlabeled data.



Semi-supervised Learning Algorithms

Reinforcement Learning

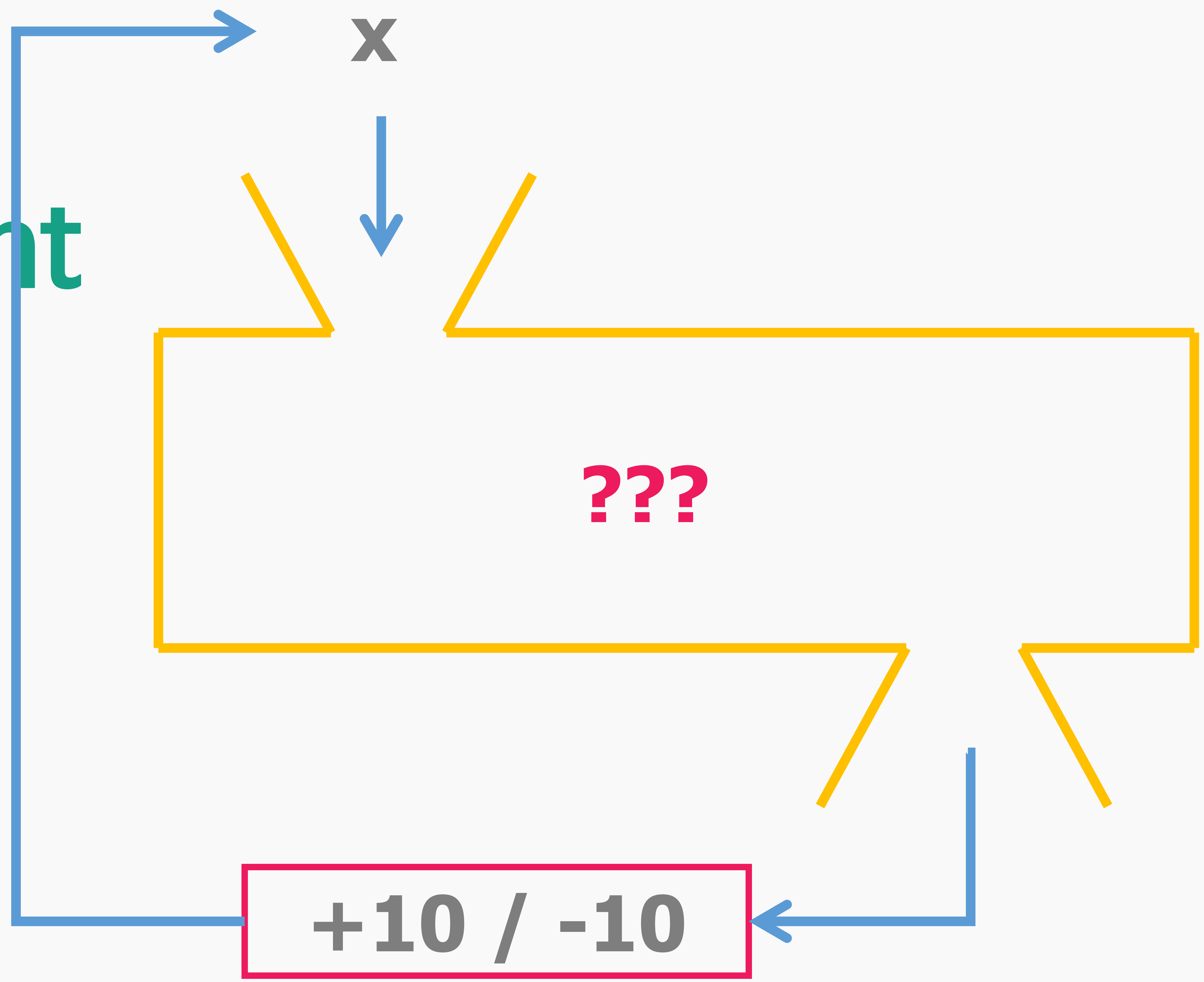
Reinforcement Learning



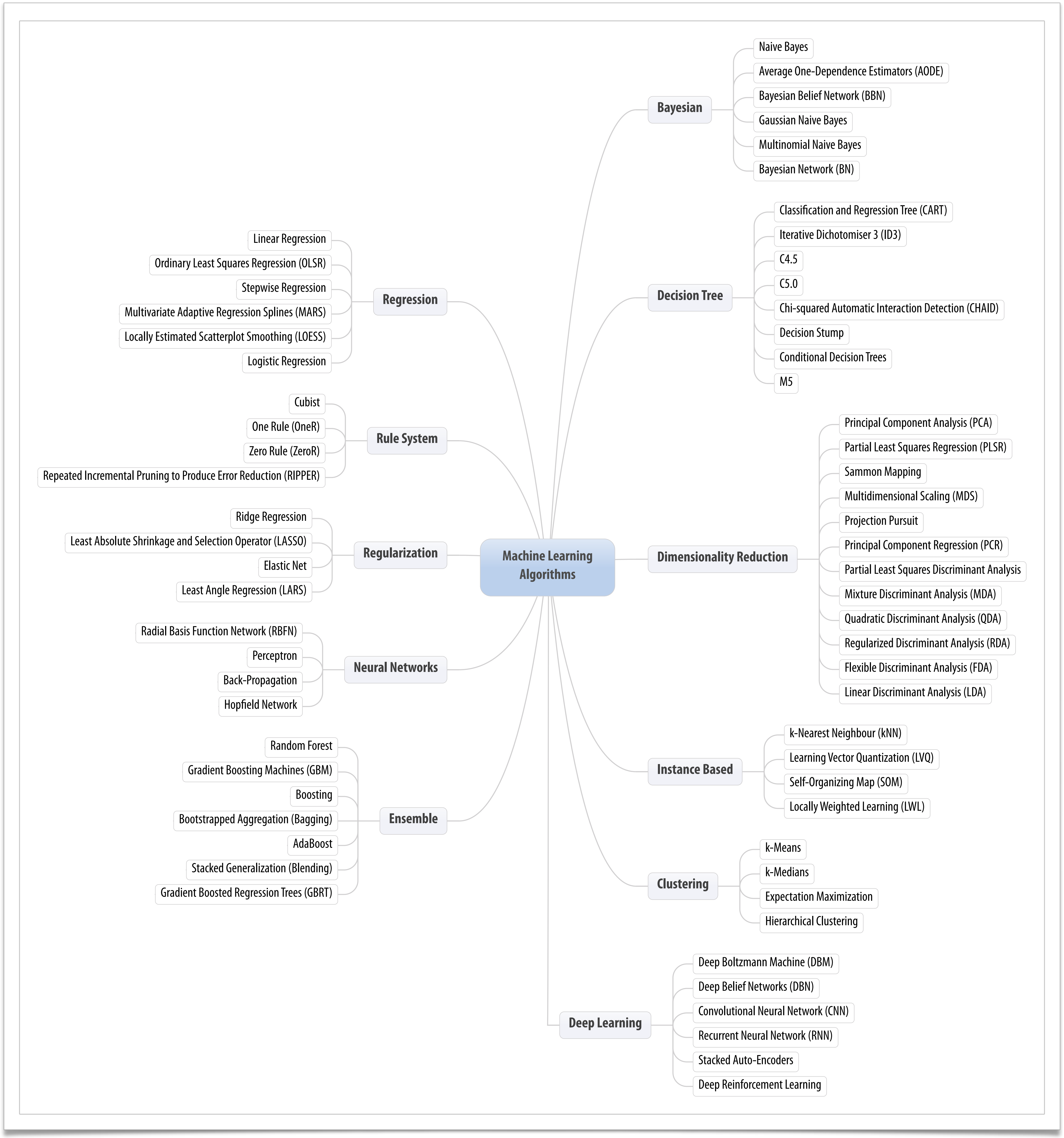
- (정답이 아닌) reward가 주어진다.

Reinforcement Learning

Reinforcement Learning



A Tour of Machine Learning Algorithms

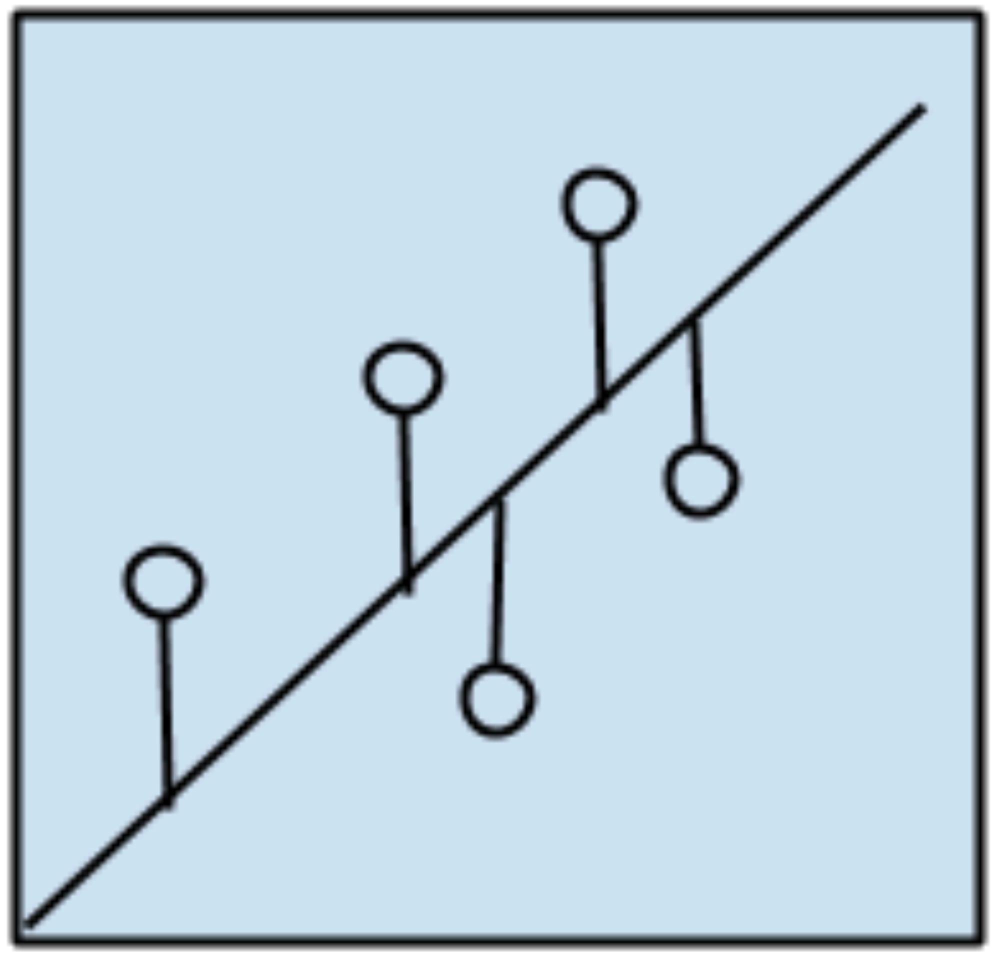


Regression Algorithms

Regression Algorithms

Regression is concerned with modeling the relationship between variables that is iteratively refined using a measure of error in the predictions made by the model.

Regression methods are a workhorse of statistics and have been co-opted into statistical machine learning. This may be confusing because we can use regression to refer to the class of problem and the class of algorithm. Really, regression is a process.



Regression Algorithms

The most popular regression algorithms are:

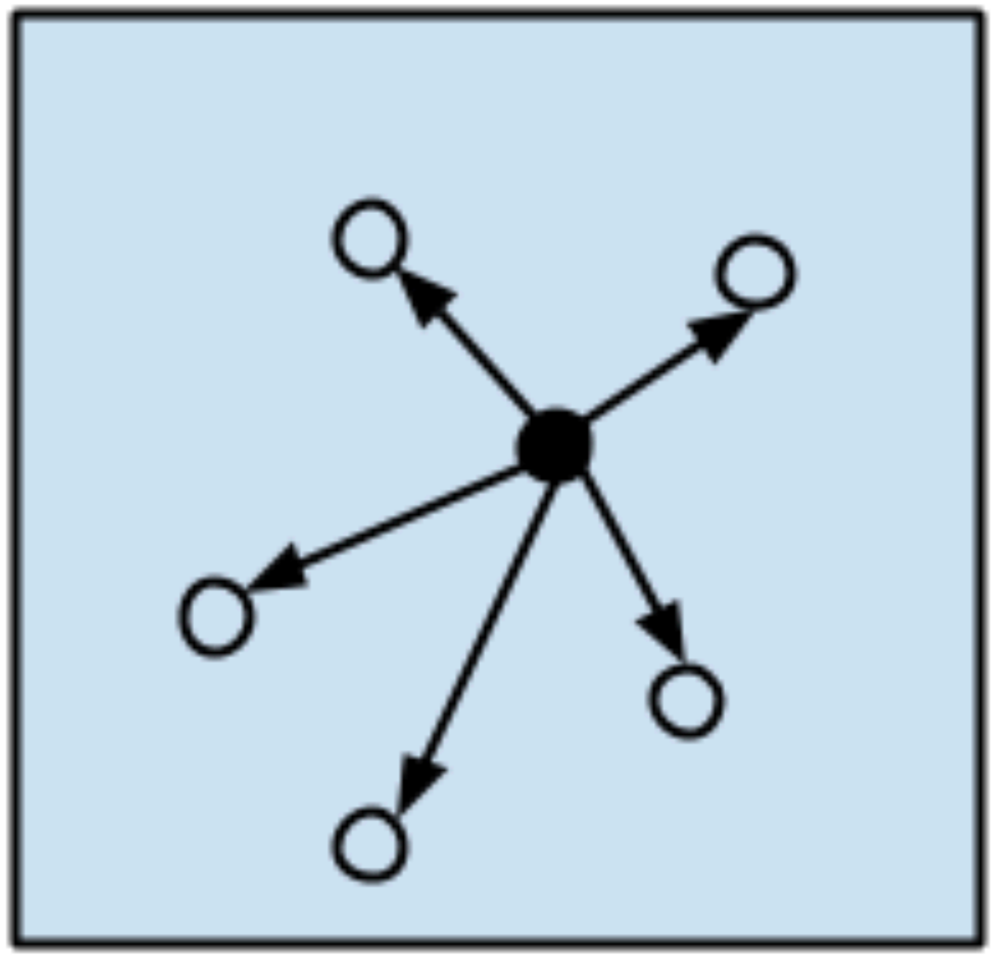
- Ordinary Least Squares Regression (OLSR)
- Linear Regression
- Logistic Regression
- Stepwise Regression
- Multivariate Adaptive Regression Splines (MARS)
- Locally Estimated Scatterplot Smoothing (LOESS)

Instance-Based Algorithms

Instance-based Algorithms

Instance-based learning model is a decision problem with instances or examples of training data that are deemed important or required to the model.

Such methods typically build up a database of example data and compare new data to the database using a similarity measure in order to find the best match and make a prediction. For this reason, instance-based methods are also called winner-take-all methods and memory-based learning. Focus is put on the representation of the stored instances and similarity measures used between instances.



Instance-based Algorithms

The most popular instance-based algorithms are:

- k-Nearest Neighbor (kNN)
- Learning Vector Quantization (LVQ)
- Self-Organizing Map (SOM)
- Locally Weighted Learning (LWL)

Regularization Algorithms

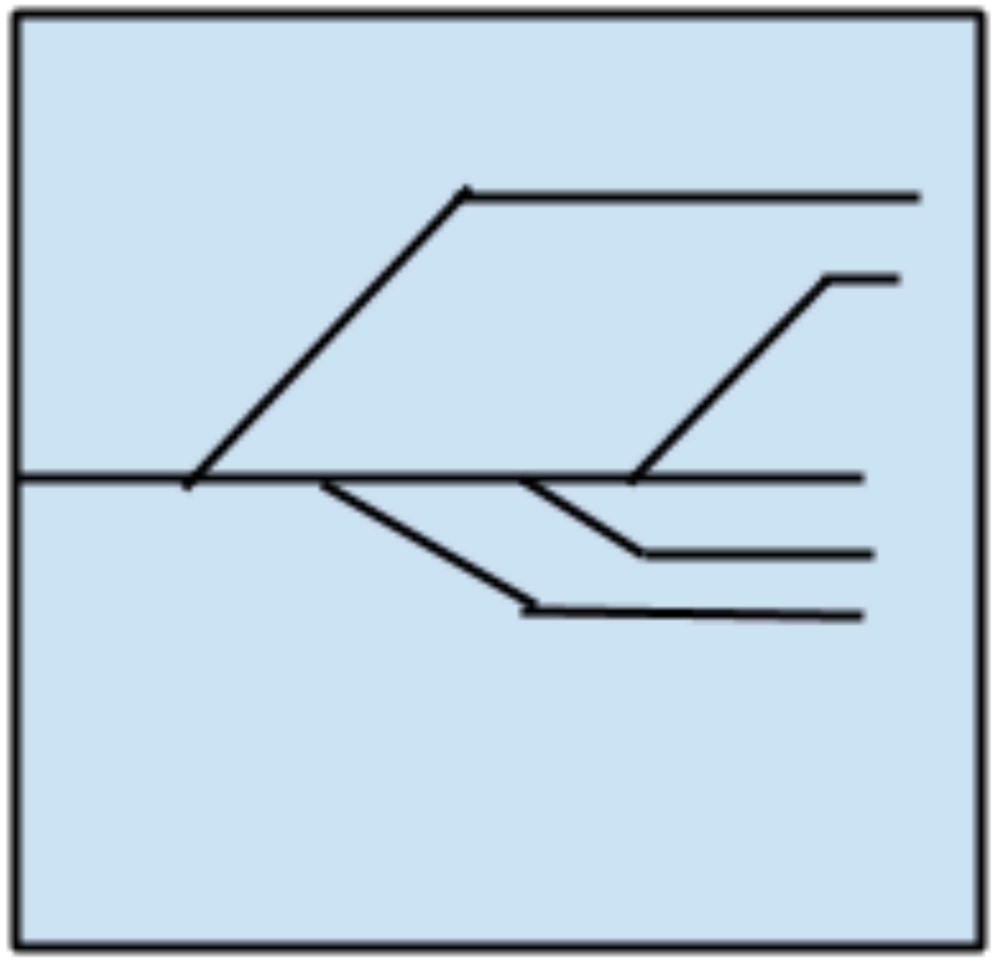
Regularization Algorithms

An extension made to another method (typically regression methods) that penalizes models based on their complexity, favoring simpler models that are also better at generalizing.

I have listed regularization algorithms separately here because they are popular, powerful and generally simple modifications made to other methods.

The most popular regularization algorithms are:

- Ridge Regression
- Least Absolute Shrinkage and Selection Operator (LASSO)
- Elastic Net
- Least-Angle Regression (LARS)



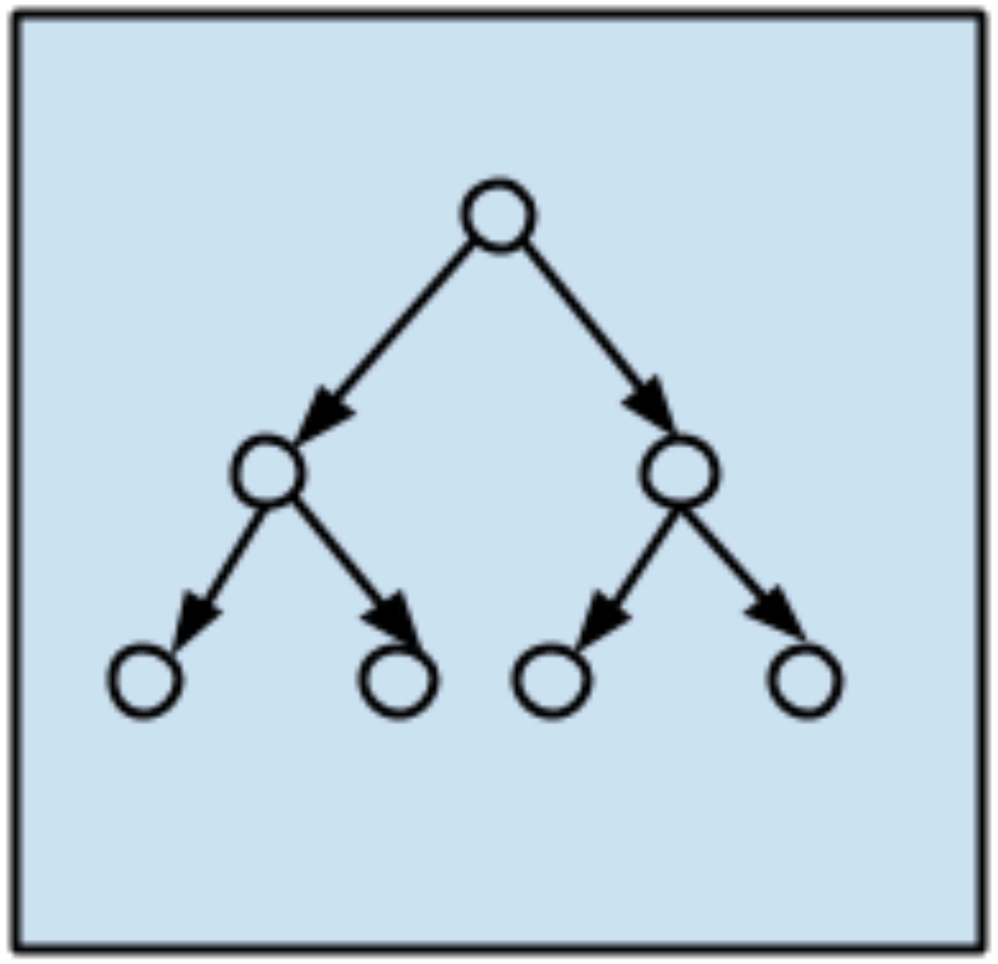
Regularization Algorithms

Decision Tree Algorithms

Decision Tree Algorithms

Decision tree methods construct a model of decisions made based on actual values of attributes in the data.

Decisions fork in tree structures until a prediction decision is made for a given record. Decision trees are trained on data for classification and regression problems. Decision trees are often fast and accurate and a big favorite in machine learning.



Decision Tree Algorithms

The most popular decision tree algorithms are:

- Classification and Regression Tree (CART)
- Iterative Dichotomiser 3 (ID3)
- C4.5 and C5.0 (different versions of a powerful approach)
- Chi-squared Automatic Interaction Detection (CHAID)
- Decision Stump
- M5
- Conditional Decision Trees

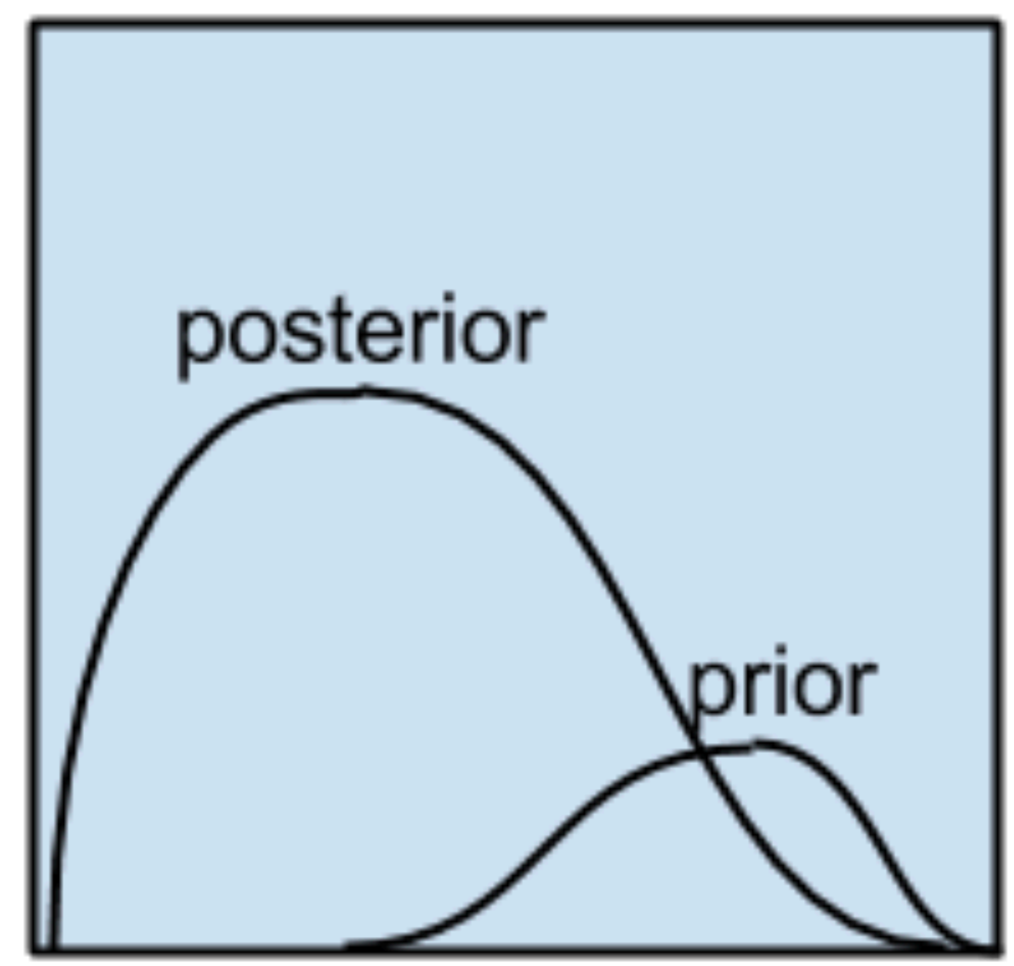
Bayesian Algorithms

Bayesian Algorithms

Bayesian methods are those that explicitly apply Bayes' Theorem for problems such as classification and regression.

The most popular Bayesian algorithms are:

- Naive Bayes
- Gaussian Naive Bayes
- Multinomial Naive Bayes
- Averaged One-Dependence Estimators (AODE)
- Bayesian Belief Network (BBN)
- Bayesian Network (BN)



Bayesian Algorithms

See Appendix for Bayes' Theorem

Clustering Algorithms

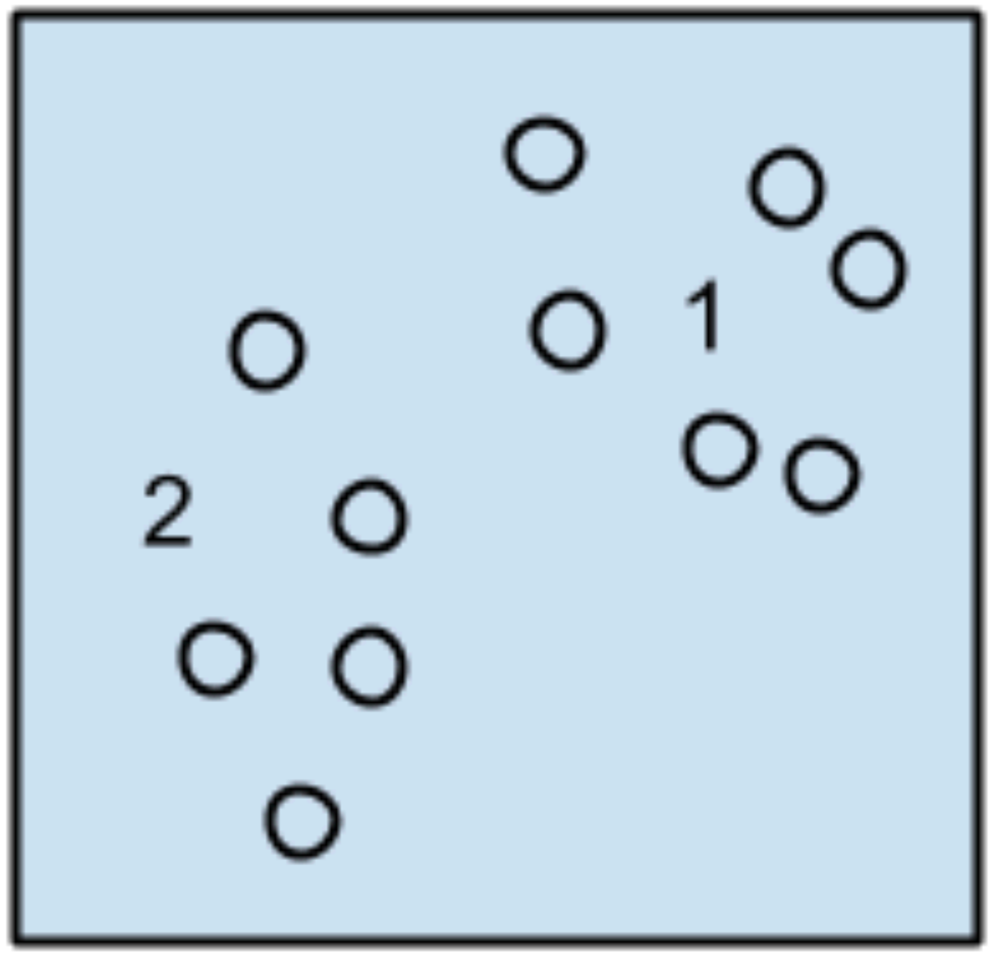
Clustering Algorithms

Clustering, like regression, describes the class of problem and the class of methods.

Clustering methods are typically organized by the modeling approaches such as centroid-based and hierarchal. All methods are concerned with using the inherent structures in the data to best organize the data into groups of maximum commonality.

The most popular clustering algorithms are:

- k-Means
- k-Medians
- Expectation Maximisation (EM)
- Hierarchical Clustering



Clustering Algorithms

Association Rule Learning Algorithms

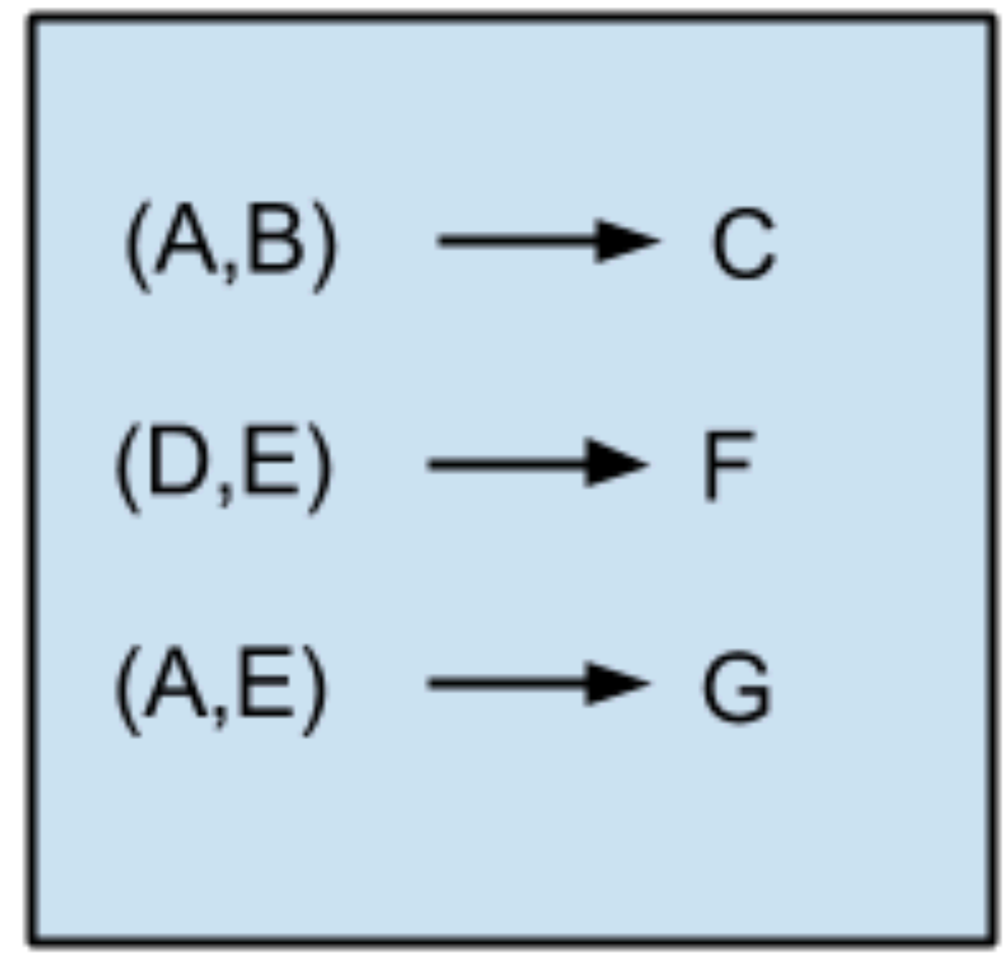
Association Rule Learning Algorithms

Association rule learning methods extract rules that best explain observed relationships between variables in data.

These rules can discover important and commercially useful associations in large multidimensional datasets that can be exploited by an organization.

The most popular association rule learning algorithms are:

- Apriori algorithm
- Eclat algorithm



Association Rule Learning Algorithms

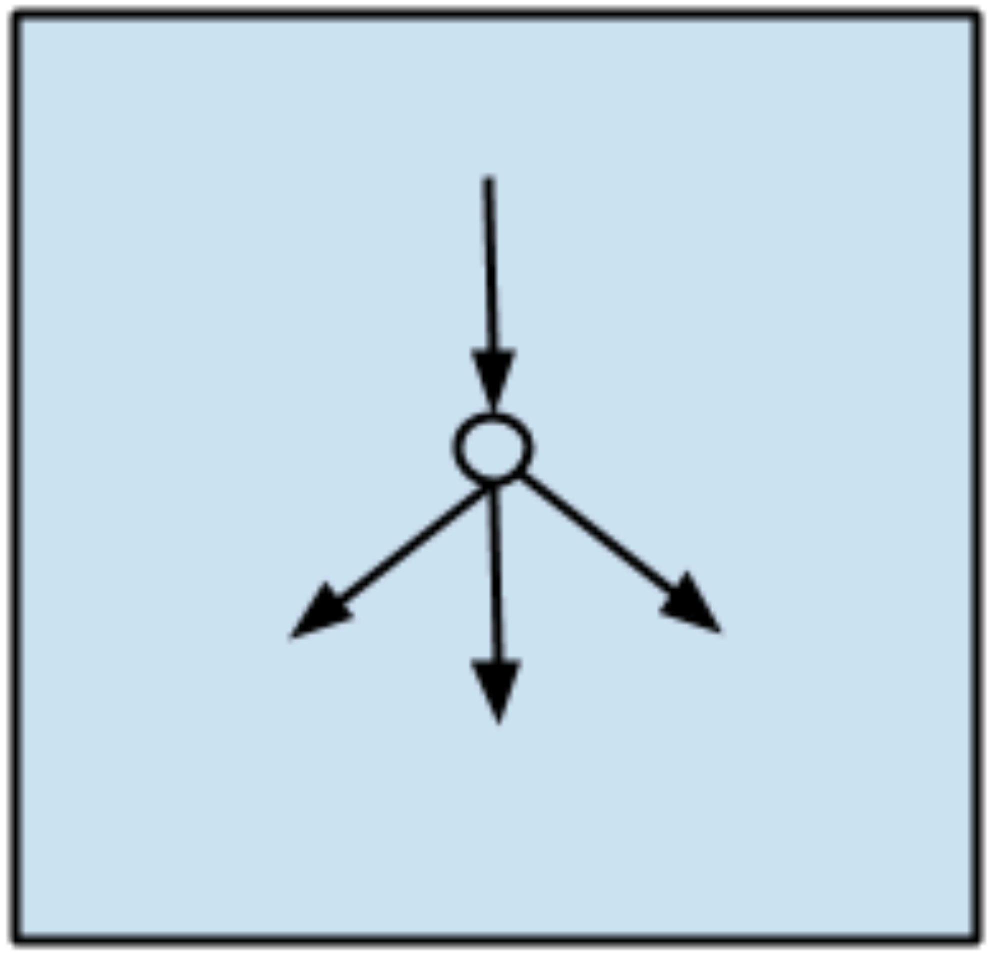
Artificial Neural Network Algorithms

Artificial Neural Network Algorithms

Artificial Neural Networks are models that are inspired by the structure and/or function of biological neural networks.

They are a class of pattern matching that are commonly used for regression and classification problems but are really an enormous subfield comprised of hundreds of algorithms and variations for all manner of problem types.

Note that I have separated out Deep Learning from neural networks because of the massive growth and popularity in the field. Here we are concerned with the more classical methods.



Artificial Neural Network Algorithms

The most popular artificial neural network algorithms are:

- Perceptron
- Back-Propagation
- Hopfield Network
- Radial Basis Function Network (RBFN)

Deep Learning Algorithms

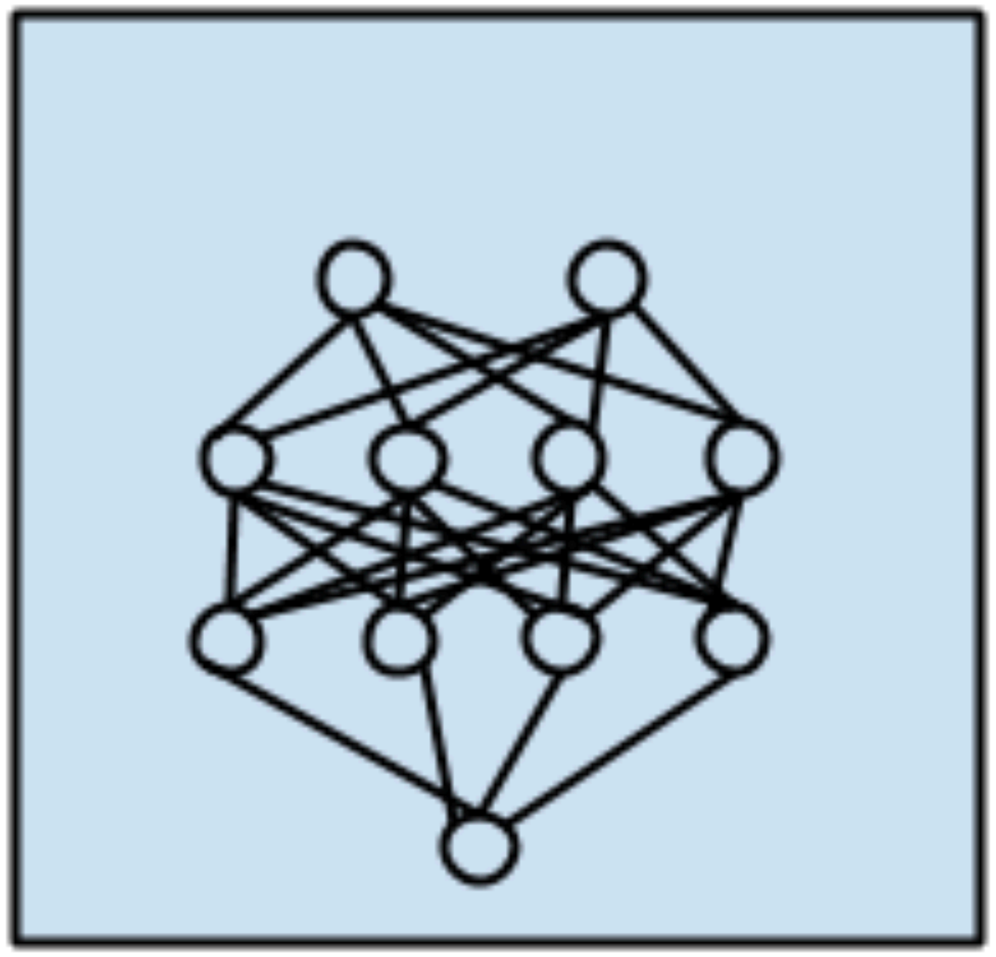
Deep Learning Algorithms

Deep Learning methods are a modern update to Artificial Neural Networks that exploit abundant cheap computation.

They are concerned with building much larger and more complex neural networks and, as commented on above, many methods are concerned with semi-supervised learning problems where large datasets contain very little labeled data.

The most popular deep learning algorithms are:

- Deep Boltzmann Machine (DBM)
- Deep Belief Networks (DBN)
- Convolutional Neural Network (CNN)
- Stacked Auto-Encoders



Deep Learning Algorithms

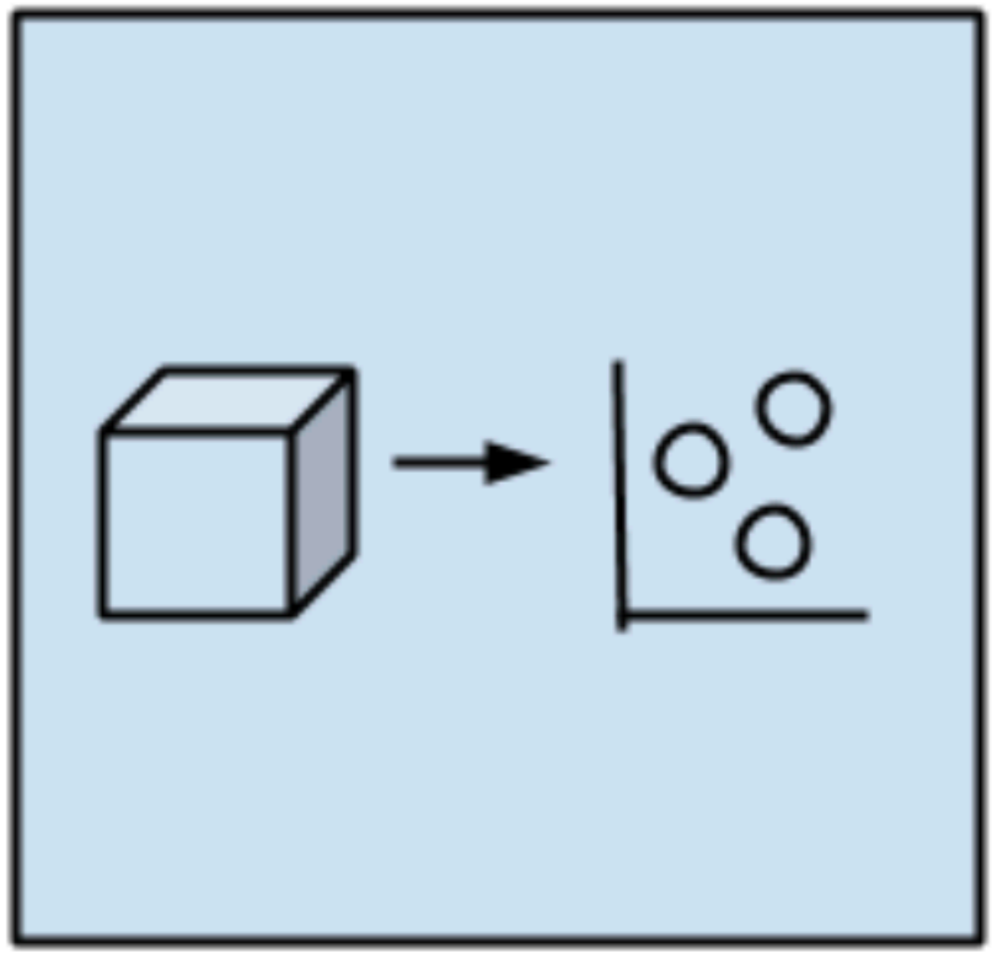
Dimensionality Reduction Algorithms

Dimensionality Reduction Algorithms

Like clustering methods, dimensionality reduction seek and exploit the inherent structure in the data, but in this case in an unsupervised manner or order to summarize or describe data using less information.

This can be useful to visualize dimensional data or to simplify data which can then be used in a supervised learning method. Many of these methods can be adapted for use in classification and regression.

- Principal Component Analysis (PCA)
- Principal Component Regression (PCR)
- Partial Least Squares Regression (PLSR)
- Sammon Mapping
- Multidimensional Scaling (MDS)
- Projection Pursuit
- Linear Discriminant Analysis (LDA)
- Mixture Discriminant Analysis (MDA)
- Quadratic Discriminant Analysis (QDA)
- Flexible Discriminant Analysis (FDA)



Dimensional Reduction Algorithms

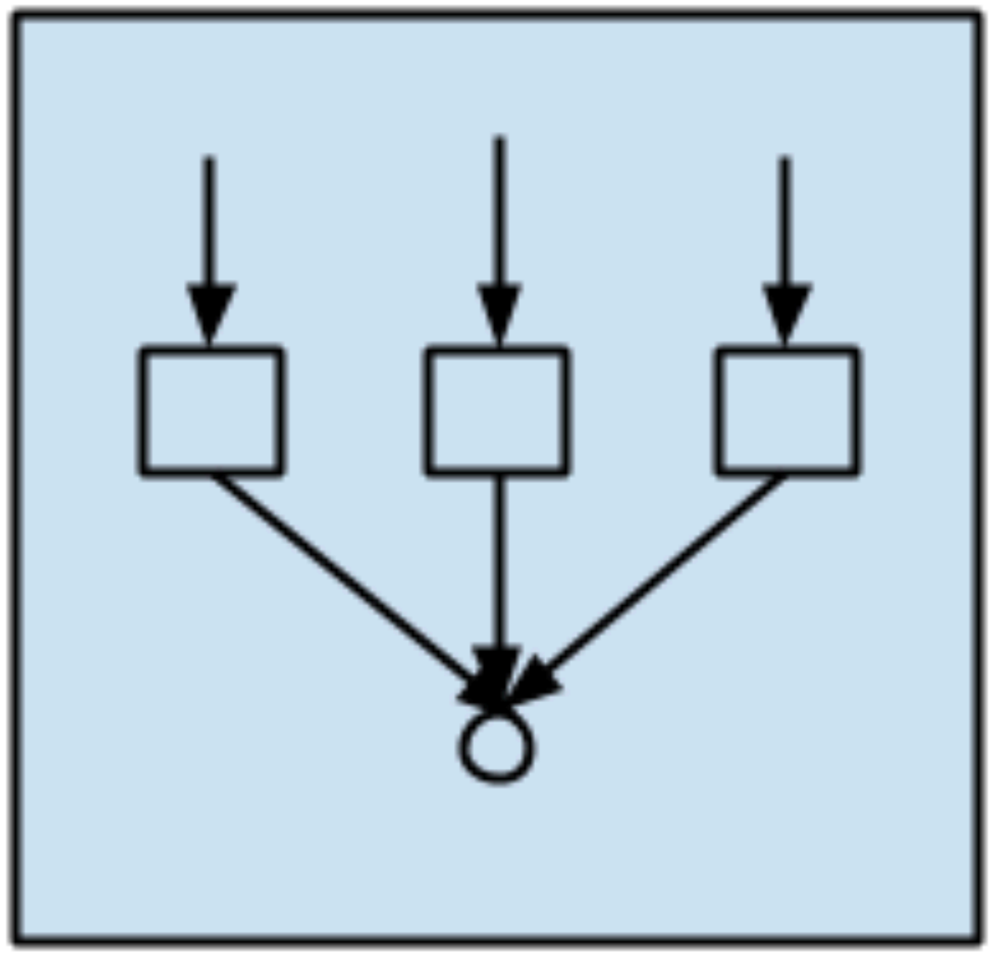
Ensemble Algorithms

Ensemble Algorithms

Ensemble methods are models composed of multiple weaker models that are independently trained and whose predictions are combined in some way to make the overall prediction.

Much effort is put into what types of weak learners to combine and the ways in which to combine them. This is a very powerful class of techniques and as such is very popular.

- Boosting
- Bootstrapped Aggregation (Bagging)
- AdaBoost
- Stacked Generalization (blending)
- Gradient Boosting Machines (GBM)
- Gradient Boosted Regression Trees (GBRT)
- Random Forest



Ensemble Algorithms

The End of the Tour!

review articles

DOI:10.1145/2347736.2347755

Tapping into the "folk knowledge" needed to advance machine learning applications.

BY PEDRO DOMINGOS

A Few Useful Things to Know About Machine Learning

MACHINE LEARNING SYSTEMS automatically learn programs from data. This is often a very attractive alternative to manually constructing them, and in the last decade the use of machine learning has spread rapidly throughout computer science and beyond. Machine learning is used in Web search, spam filters, recommender systems, ad placement, credit scoring, fraud detection, stock trading, drug design, and many other applications. A recent report from the McKinsey Global Institute asserts that machine learning (a.k.a. data mining or predictive analytics) will be the driver of the next big wave of innovation.[15] Several fine textbooks are available to interested practitioners and researchers (for example, Mitchell[16] and Witten et al.[24]). However, much of the "folk knowledge" that is needed to successfully develop machine learning applications is not readily available in them. As a result, many machine learning projects take much longer than necessary or wind up producing less-than-ideal results. Yet much of this folk knowledge is fairly easy to communicate. This is the purpose of this article.

key insights

- **Machine learning algorithms can figure out how to perform important tasks by generalizing from examples. This is often feasible and cost-effective where manual programming is not. As more data becomes available, more ambitious problems can be tackled.**
- **Machine learning is widely used in computer science and other fields. However, developing successful machine learning applications requires a substantial amount of "black art" that is difficult to find in textbooks.**
- **This article summarizes 12 key lessons that machine learning researchers and practitioners have learned. These include pitfalls to avoid, important issues to focus on, and answers to common questions.**

Introduction

- Machine learning algorithms can figure out how to perform important tasks by *generalizing from examples*.
- As *more data* becomes available, more ambitious problems can be tackled.
- This paper contains lot of "*folk knowledge*" that is needed to successfully develop machine learning applications, which is *not* readily available in ML textbooks.

Learning = R + E + O

- Learning = Representation + Evaluation + Optimization
  - ***Representation***: algorithm, implementation
  - ***Evaluation***: metric selection, results based on concrete data
  - ***Optimization***: from off-the-shelf do custom settings

Table 1. The three components of learning algorithms.

Representation	Evaluation	Optimization
Instances	Accuracy/Error rate	Combinatorial optimization
K-nearest neighbor	Precision and recall	Greedy search
Support vector machines	Squared error	Beam search
Hyperplanes	Likelihood	Branch-and-bound
Naive Bayes	Posterior probability	Continuous optimization
Logistic regression	Information gain	Unconstrained
Decision trees	K-L divergence	Gradient descent
Sets of rules	Cost/Utility	Conjugate gradient
Propositional rules	Margin	Quasi-Newton methods
Logic programs		Constrained
Neural networks		Linear programming
Graphical models		Quadratic programming
Bayesian networks		
Conditional random fields		

It's Generalization That Counts

- ***Distinguish*** **between *training data* and *test data*!**
- **To use as many data as possible use *cross-validation***
- **Principle #1 "not just good ML algorithms, but more data!"**

Overfitting Has Many Faces

- Bias: 잘못된 것을 계속 학습하는 경향
- Variance: 실제 데이터와 상관없이 아무거나 배워버리는 경향
- **The Bias-Variance Decomposition**
  - *Bias* – e.g., linear predictors problem (*underfitting*)
  - *Variance* – e.g., decision trees problem (*overfitting*)
- **Cross-validation, regularization**

Figure 2. Naïve Bayes can outperform a state-of-the-art rule learner (C4.5rules) even when the true classifier is a set of rules.

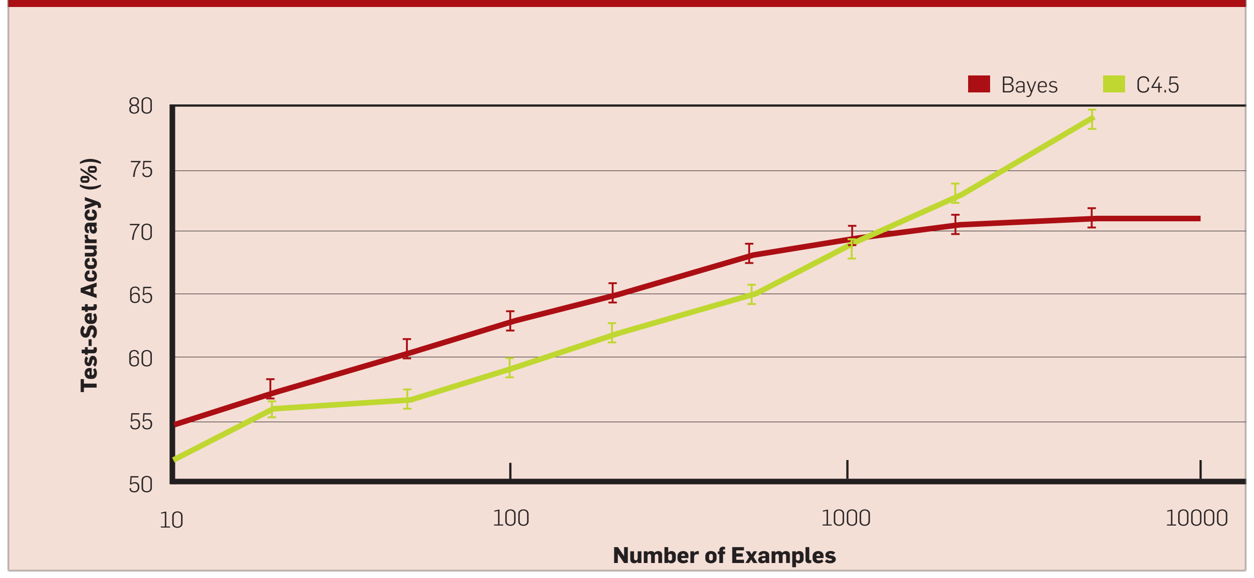
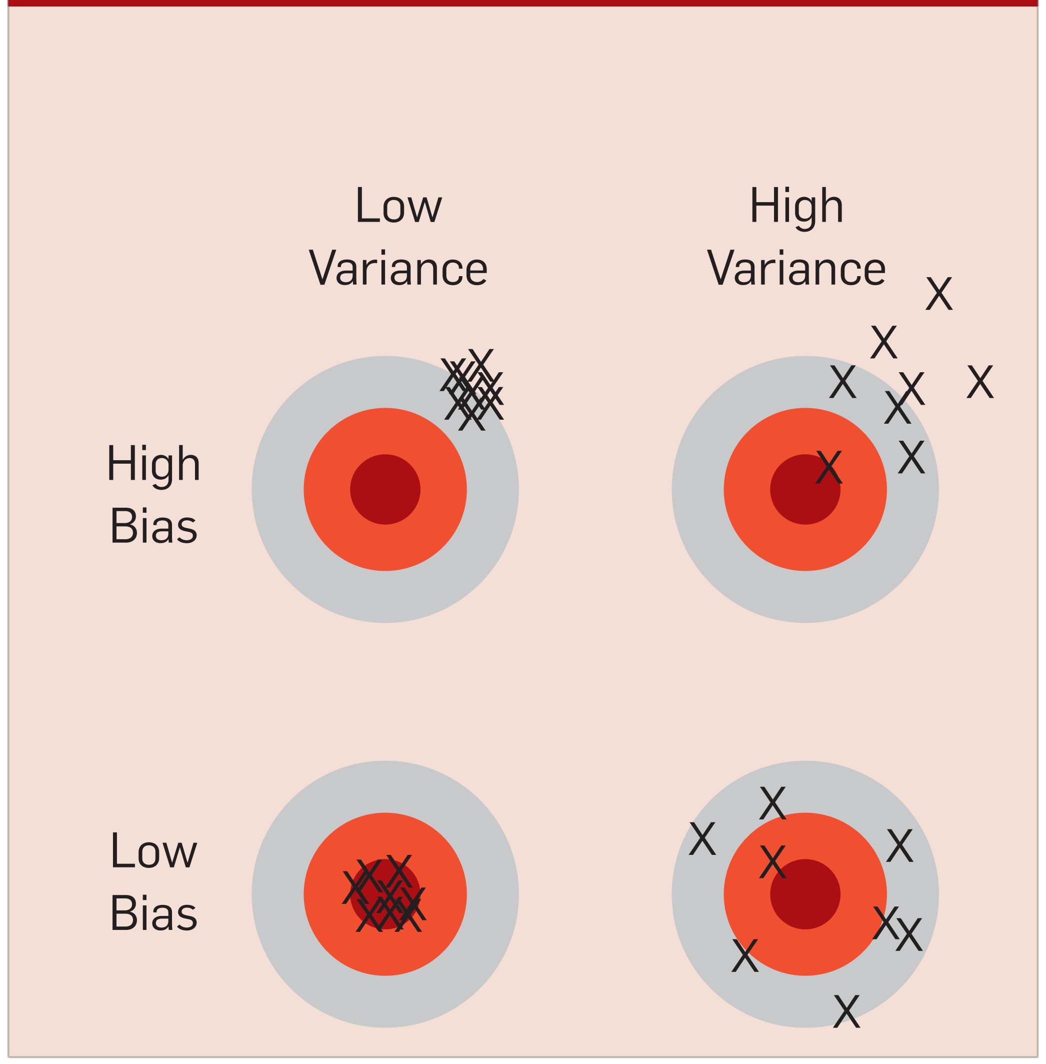
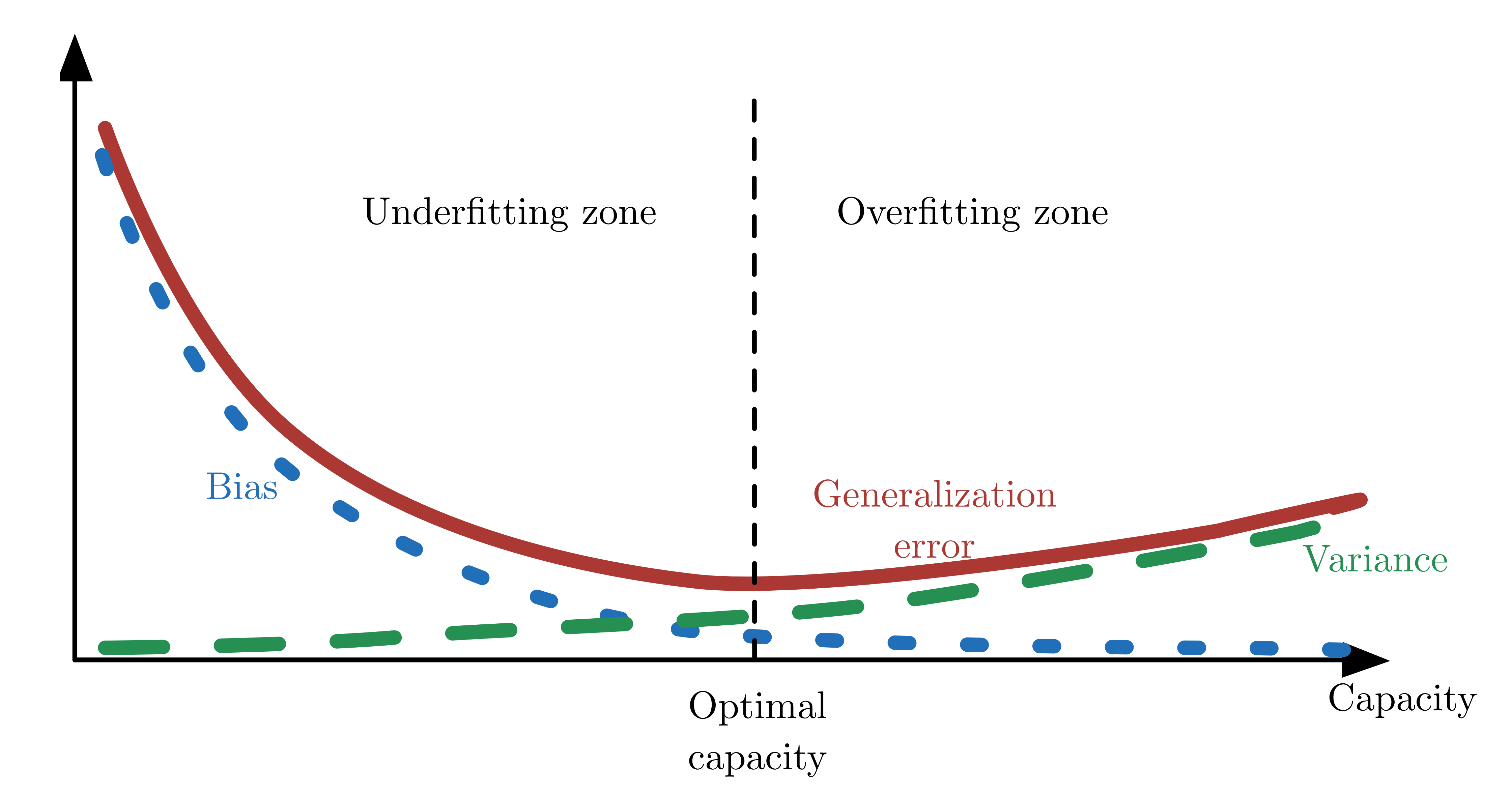


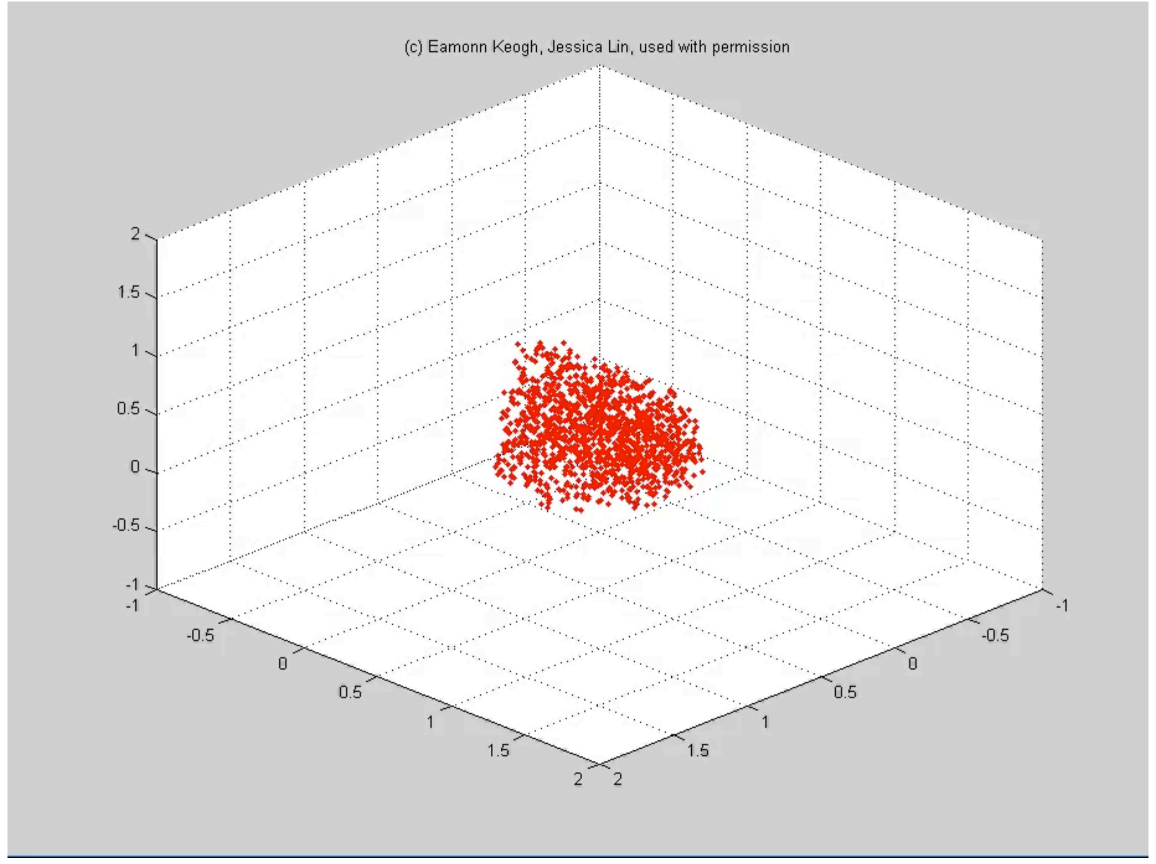
Figure 1. Bias and variance in dart-throwing.

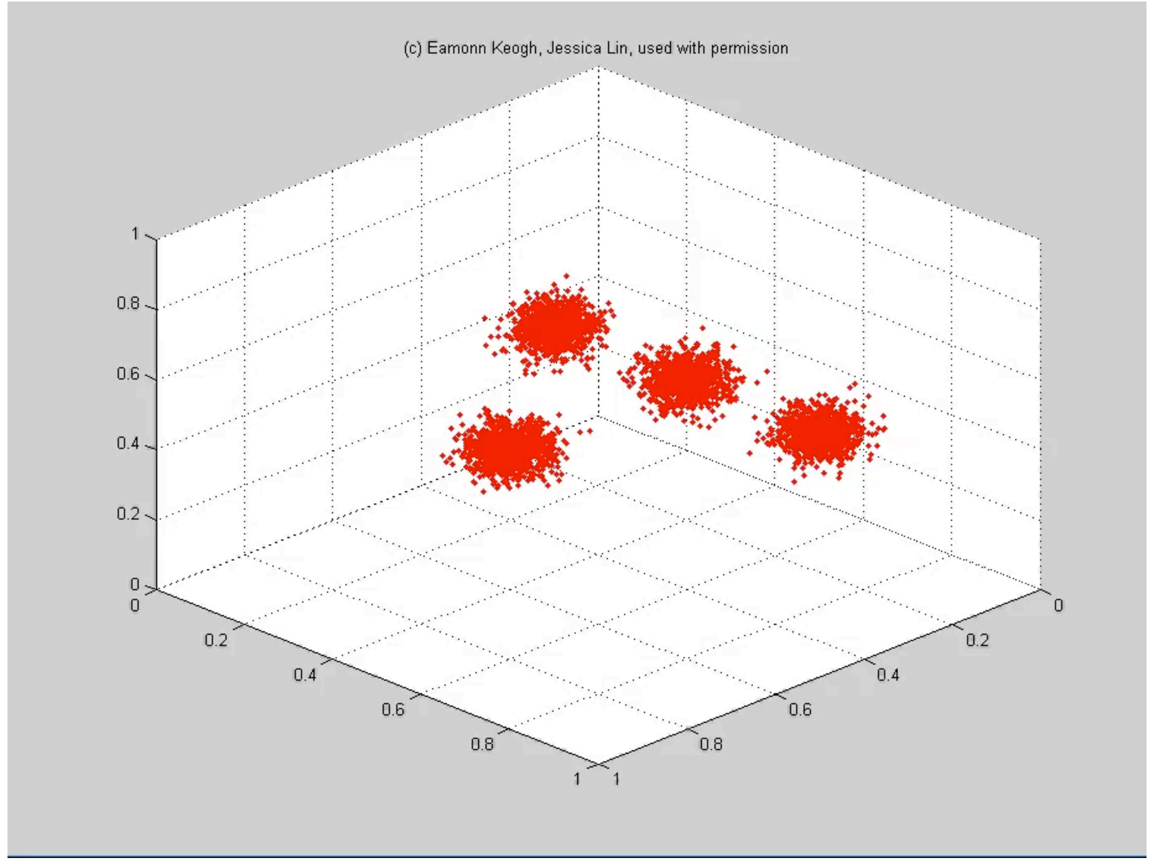




Intuition Fails in High Dimensions

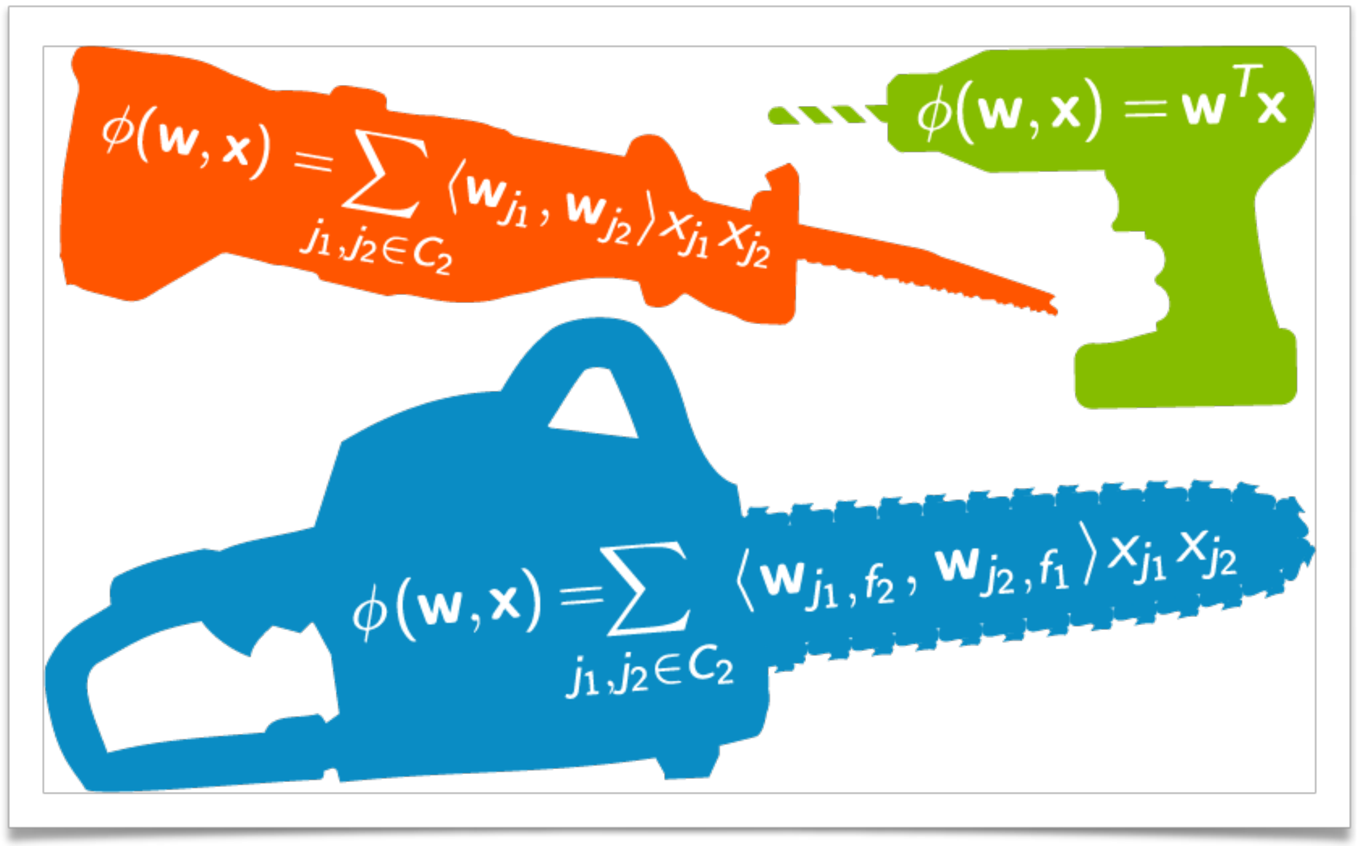
- *Curse of dimensionality* [Bellman, 1957]
  - Correct generalizing becomes *exponentially harder* as the *dimensionality (number of features)* of the examples grows
  - With a fixed number of training instances, the predictive power *reduces* as the dimensionality increases; *Hughes phenomenon* (named after Gordon F. Hughes)
  - Add new attributes only if they bring *new information* (do not use highly correlating features)
- *Blessing of non-uniformity*
  - Non-uniform training examples [e.g., handwritten digit recognition]
  - Dimensionality reduction techniques [*Note: the *dangers* of dimensionality reduction]

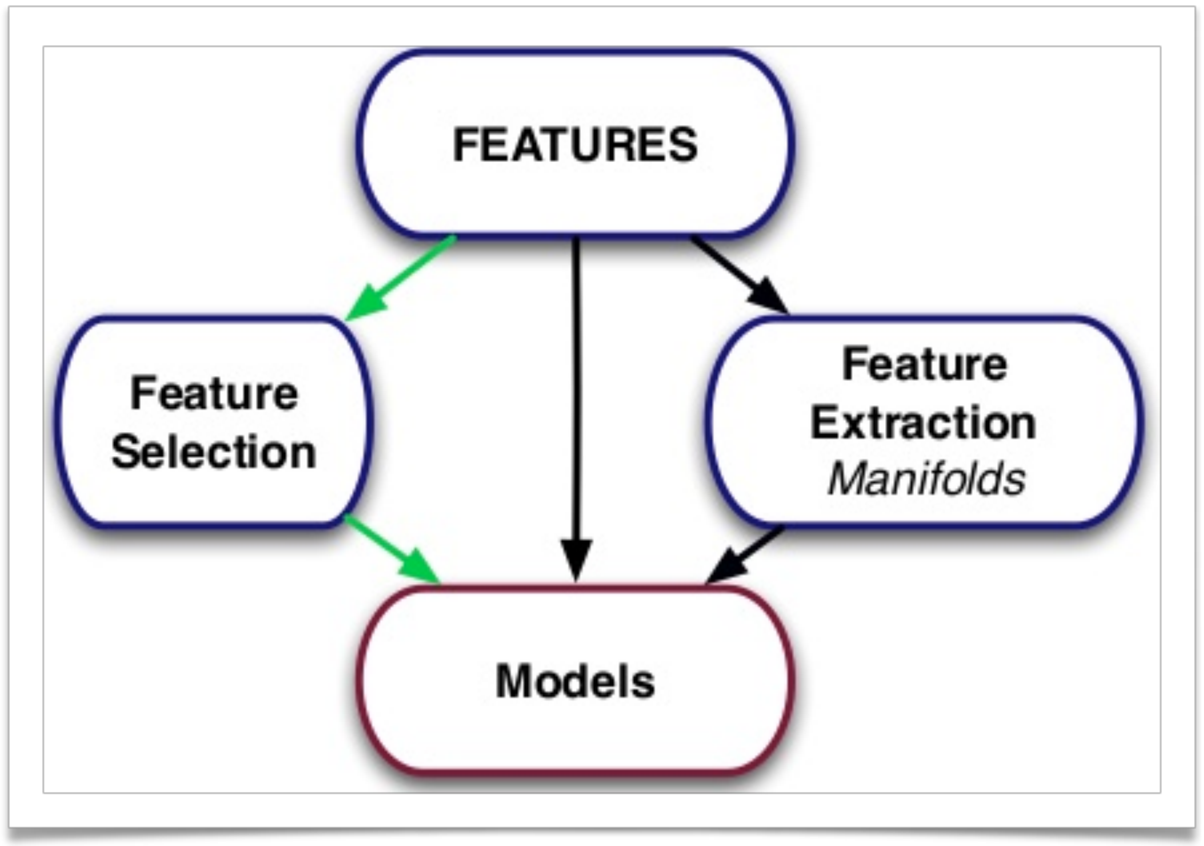




Feature Engineering is the Key

- The ***most important factor*** of ML project is the ***features*** used.
  - With many independent features that each ***correlate well*** with the predicted class, learning is easy.
  - On the other hand, if the class is a ***very complex function*** of the features, you may ***not*** be able to learn it.
- ML is often the quickest part, but that's because we've already mastered it pretty well!
  - Feature engineering is more difficult because it's ***domain-specific***, while learners can be largely general-purpose.

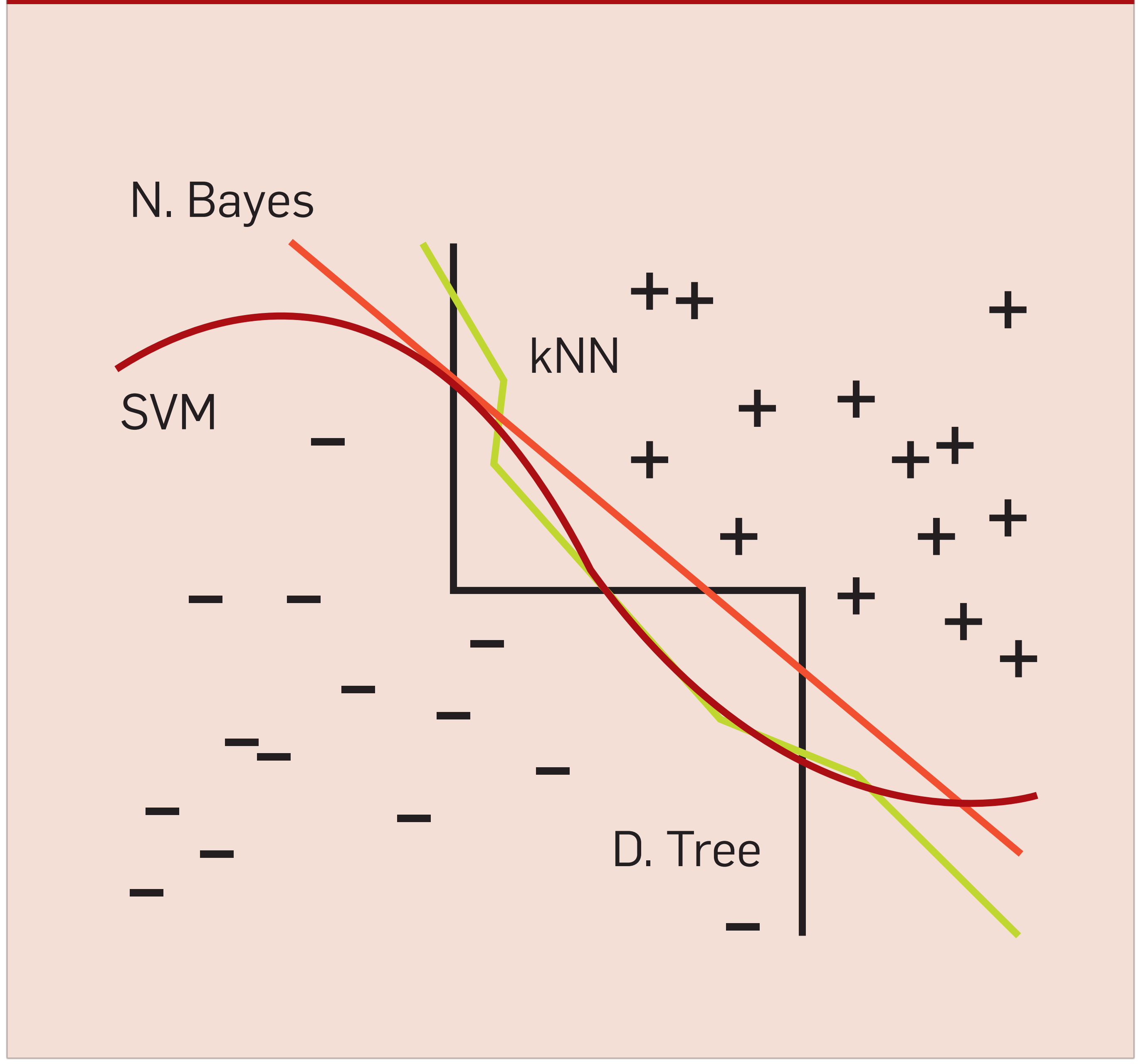




More Data Beats a Cleverer Algorithm

- ML researchers are mainly concerned with creation of *better learning algorithm*, but pragmatically the ***quickest path to success*** is often to just *get more data*.
- This does bring up ***another problem***, however: *scalability*.
  - In most of computer science, the 2 main limited resources are *time* and *memory*. In ML, there is a 3rd one: *training data*.
  - Bottleneck changed from *data* to *time*

Figure 3. Very different frontiers can yield similar predictions. (+ and – are training examples of two classes.)



Learn Many Models, Not Just One

- **Ensemble Methods**

  - ***Bagging*** **- the simplest technique, which simply generate random variations of the training set by** ***resampling*****, learn a classifier on each, and combine the results by voting. This works because it greatly** ***reduces variance*** **while only slightly increasing bias.**

  - ***Boosting*** **- training examples have** ***weights*****, and these are varied so that each new classifier focuses on the examples the previous ones tended to get wrong.**

  - ***Stacking*** **- the outputs of individual classifiers become the inputs of a** ***"higher-level" learner*** **that figures out how best to combine them.**

Simplicity Does Not Imply Accuracy

- **Occam's razor** (오컴의 면도날)
  - **Principle of Economy** (경제성의 원리), **Principle of Parsimony** (사고 절약의 원리)
  - **"Given two equally accurate theories, choose the one that is *less complex*"**
  - **The explanation of any phenomenon should make as *few assumptions* as possible**

- **No free lunch theorem**
  - **Paper: "No Free Lunch Theorems for Optimization", *IEEE Transactions on Evolutionary Computation*, 1997.**
  - 특정한 문제에 최적화된 알고리즘은 다른 문제에 대해서는 그렇지 않다는 것을 수학적으로 증명한 정리
  - ML을 적용하는 데 있어 종종 범하는 실수 중 하나는 하나의 ML 프로그램에서 많은 것을 바란다는 것
  - "머신러닝은 마법이 아니다!"

Correlation Does Not Imply Causation

- 단지 동시에 두 가지 결과가 함께 나왔다고 해서 인과관계가 있다고 가정하지 마라!

- ML 알고리즘에서의 training을 관측가능한 변수(observable data)에 의해 수행되기 때문에, 인과관계(causual relationship)을 생성해주는 예측변수(predictive variable)들이 드러나지 않는다!

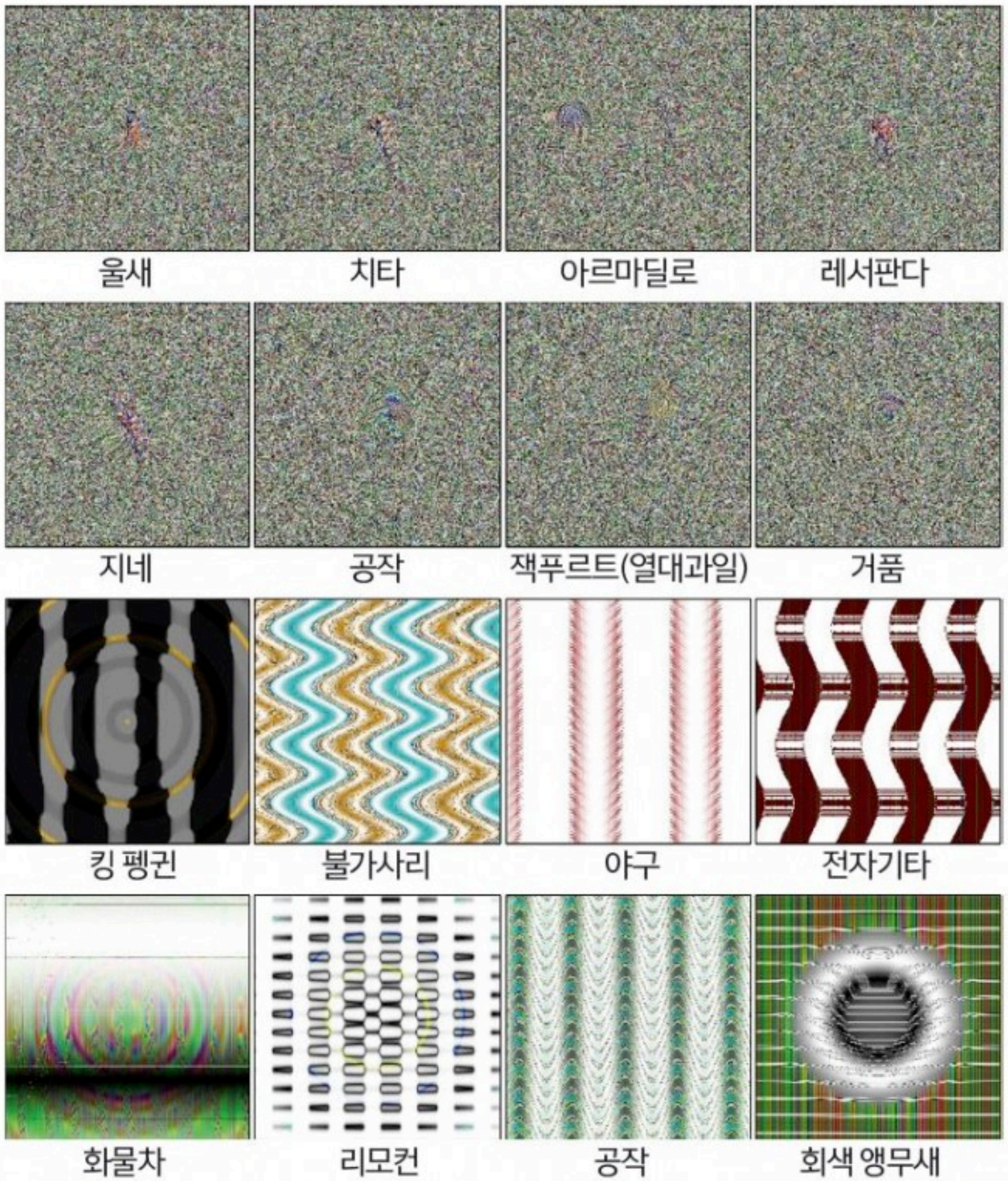
- **Experimental data** [인과관계(causality)를 파악할 수 있는 데이터]

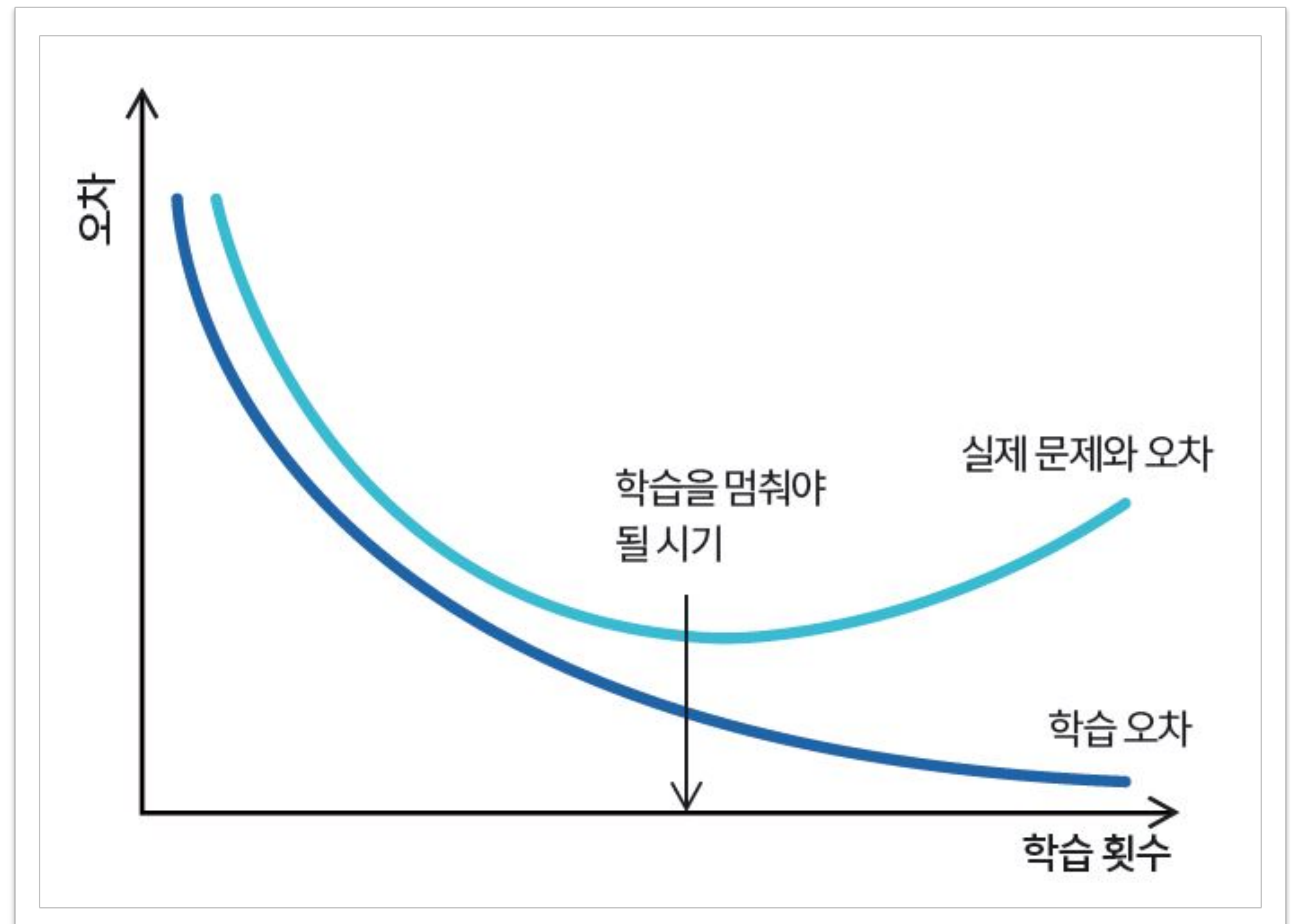
- 만약 이러한 데이터(experimental data)를 얻을 수 있다면, 최대한 많이 확보하라!

Major Drawbacks in Deep Learning

- There are ***drawbacks*** to keep in mind when using deep learning.

'심층 신경망은 쉽게 바보가 될 수 있다', CVPR 2015

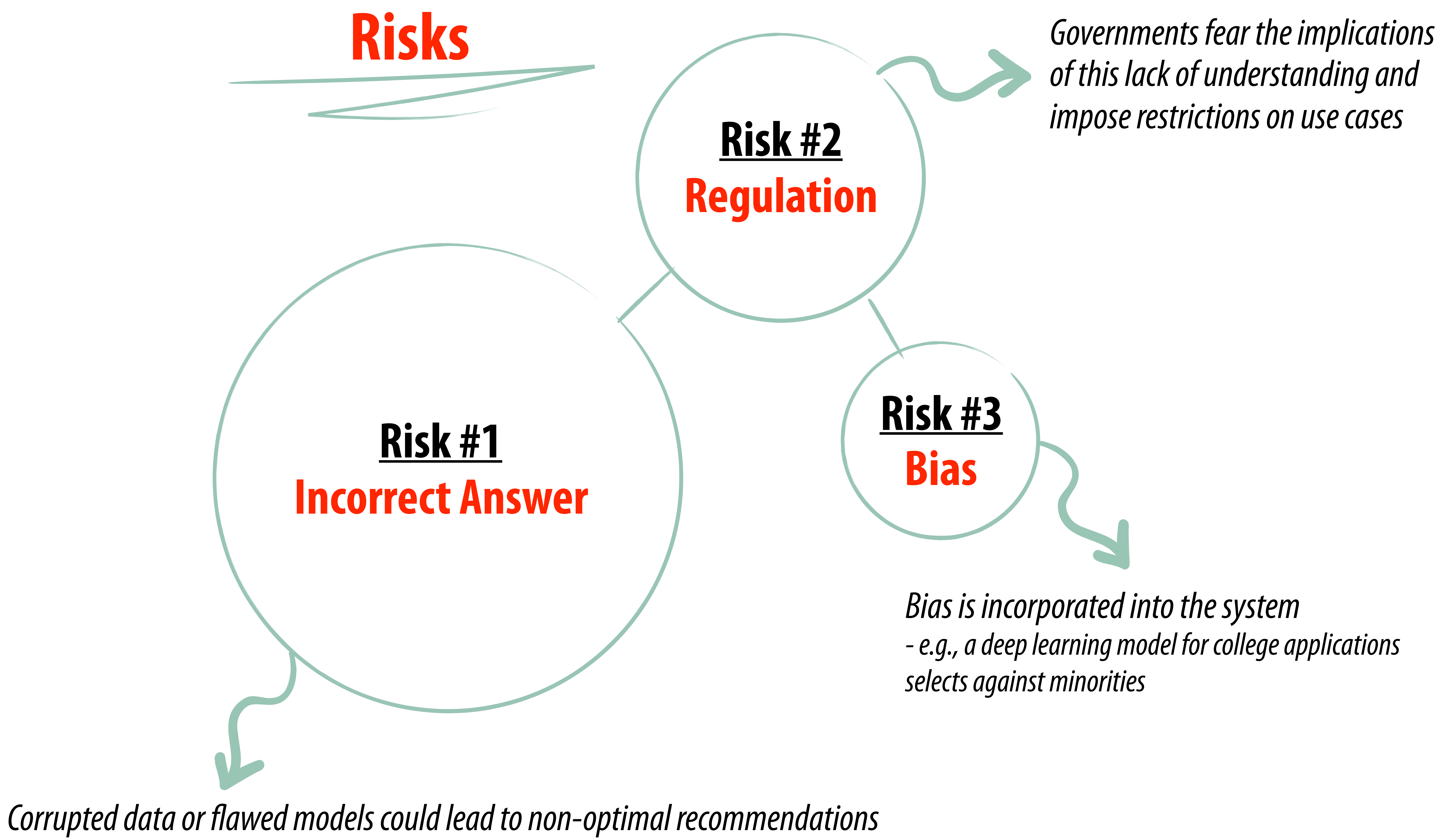




Overfitting!

Major Drawbacks in Deep Learning

- There are *drawbacks* to keep in mind when using deep learning.



Summary

- <u>A Tour of ML Algorithms</u>: Machine learning is ***not monolithic***.
  - Supervised, Semi-Supervised, Unsupervised, Reinforcement Learning
  - Machine learning algorithms

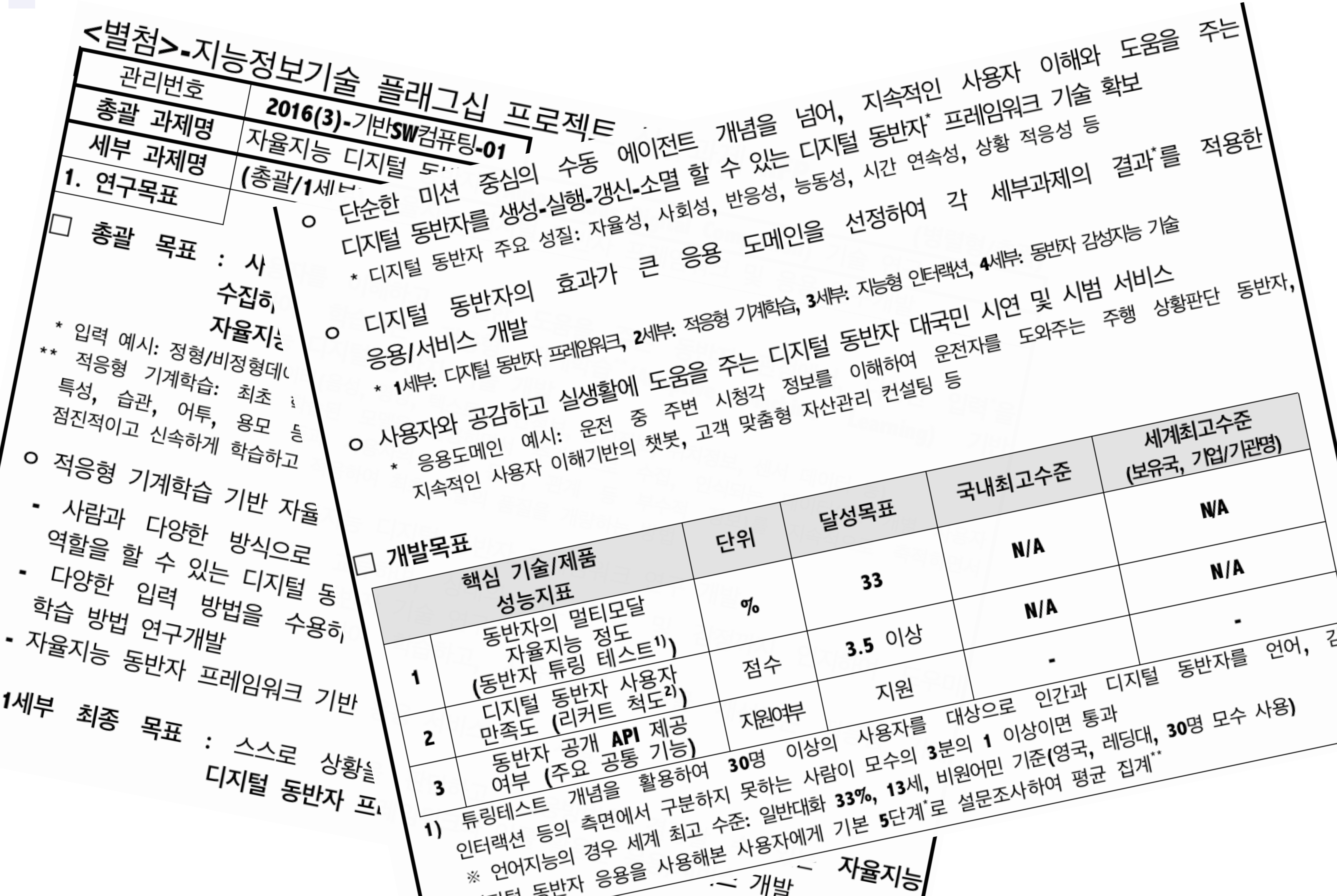
- "*Folk knowledge*" in machine learning
  - training vs. test data, cross-validation, more data, many models (ensemble), curse of dimensionality, occam's razor, no-free lunch theorem

- Deep learning has emerged as a technique with *strong advantages*, but also has important *drawbacks* as well.
  - Incorrect answer, bias, overfitting, regularization


one more thing
or two...


her
A SPIKE JONZE LOVE STORY

Enjoy the failure!


<별첨>-지능정보기술 플래그십 프로젝트

관리번호	2016(3)-기반SW컴퓨팅-01
총괄 과제명	자율지능 디지털 동
세부 과제명	(총괄/1세부
1. 연구목표	

□ 총괄 목표 : 사

수집하

자율지능

* 입력 예시: 정형/비정형데

** 적응형 기계학습: 최초

특성, 습관, 어투, 용모 등

점진적이고 신속하게 학습하고

o 적응형 기계학습 기반 자율

- 사람과 다양한 방식으로 역할을 할 수 있는 디지털 동
- 다양한 입력 방법을 수용하 학습 방법 연구개발
- 자율지능 동반자 프레임워크 기반

1세부 최종 목표 : 스스로 상황을

디지털 동반자 프

o 단순한 미션 중심의 수동 에이전트 개념을 넘어, 지속적인 사용자 이해와 도움을 주는 디지털 동반자를 생성-실행-갱신-소멸 할 수 있는 디지털 동반자* 프레임워크 기술 확보

* 디지털 동반자 주요 성질: 자율성, 사회성, 반응성, 능동성, 시간 연속성, 상황 적응성 등

o 디지털 동반자의 효과가 큰 응용 도메인을 선정하여 각 세부과제의 결과*를 적용한 응용/서비스 개발

* 1세부: 디지털 동반자 프레임워크, 2세부: 적응형 기계학습, 3세부: 지능형 인터랙션, 4세부: 동반자 감성지능 기술

o 사용자와 공감하고 실생활에 도움을 주는 디지털 동반자 대국민 시연 및 시범 서비스

* 응용도메인 예시: 운전 중 주변 시청각 정보를 이해하여 운전자를 도와주는 주행 상황판단 동반자, 지속적인 사용자 이해기반의 챗봇, 고객 맞춤형 자산관리 컨설팅 등

□ 개발목표

	핵심 기술/제품 성능지표	단위	달성목표	국내최고수준	세계최고수준 (보유국, 기업/기관명)
1	동반자의 멀티모달 자율지능 정도 (동반자 튜링 테스트1))	%	33	N/A	N/A
2	디지털 동반자 사용자 만족도 (리커트 척도2))	점수	3.5 이상	N/A	N/A
3	동반자 공개 API 제공 여부 (주요 공통 기능)	지원여부	지원	-	-

1) 튜링테스트 개념을 활용하여 30명 이상의 사용자를 대상으로 인간과 디지털 동반자를 언어, 감

인터랙션 등의 측면에서 구분하지 못하는 사람이 모수의 3분의 1 이상이면 통과

※ 언어지능의 경우 세계 최고 수준: 일반대화 33%, 13세, 비원어민 기준(영국, 레딩대, 30명 모수 사용)

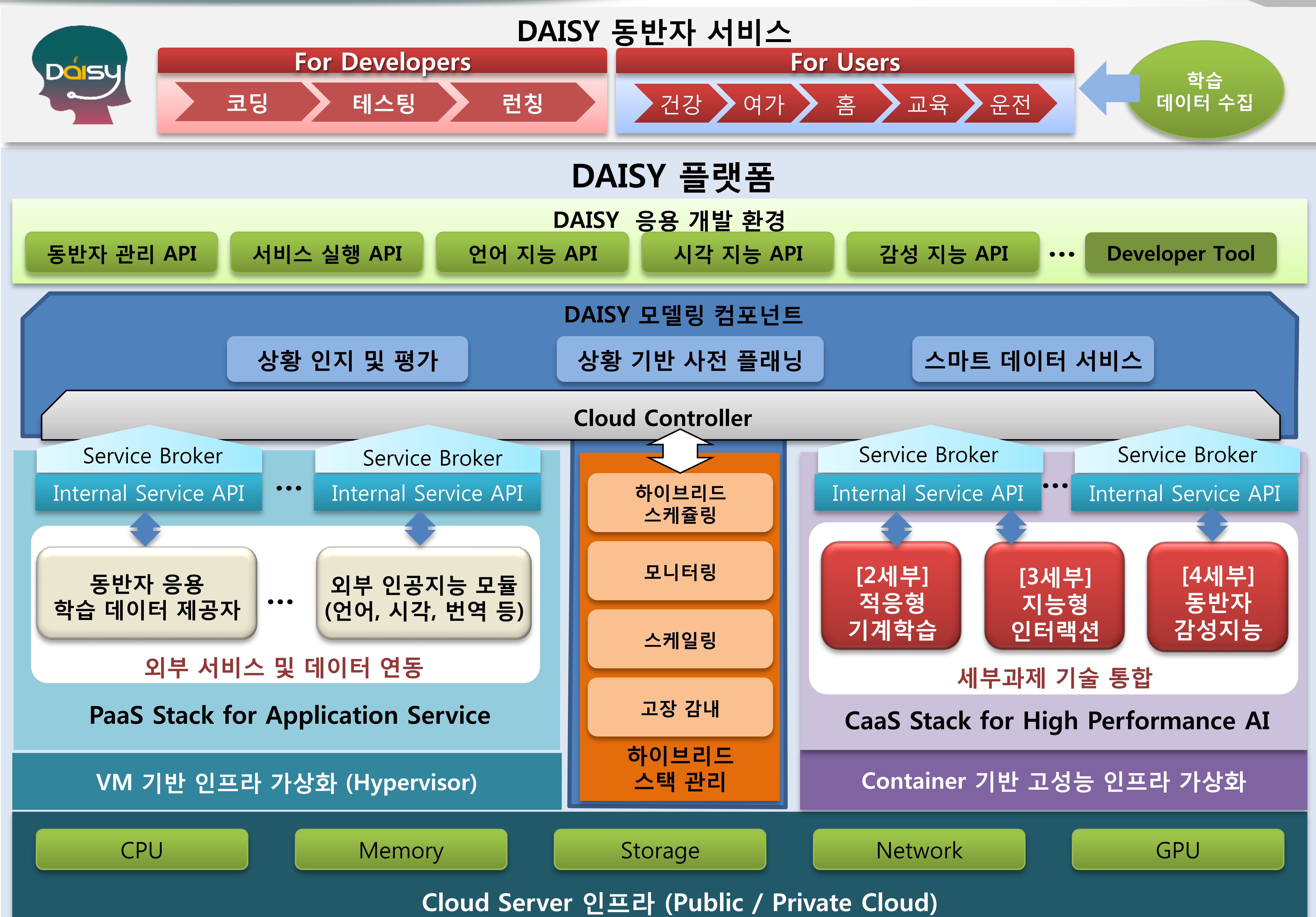
지털 동반자 응용을 사용해본 사용자에게 기본 5단계*로 설문조사하여 평균 집계**

개발 자율지능

Software Architecture

연구개발목표의 명확성

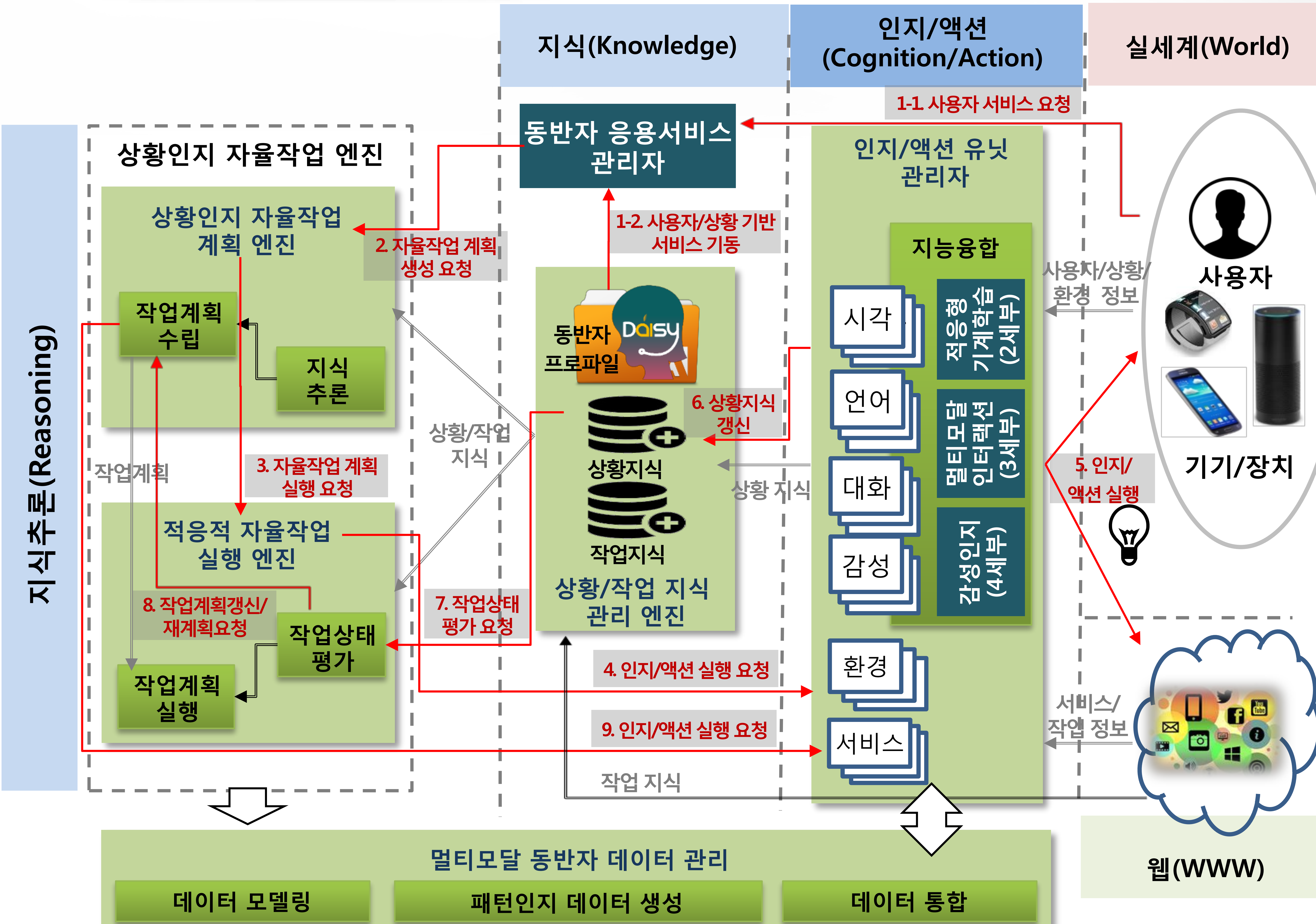
2.2 시스템 동작 메커니즘 – DAISY 플랫폼의 소프트웨어 구조



Container : 특정 응용을 위한 경량 가상화 기술, CaaS (Container as a Service): Container 기반의 가상화를 제공하는 Cloud Service

연구개발목표의 명확성

2.2 시스템 동작 메커니즘 – DAISY 인공지능 동작 메커니즘



One of the TF Team Meeting Logs

- 스마트 데이터 작성본 검토
  - 검토의견
    - 범례: 의견 [의견 제시자]
    - **작성 배경**
      - RFP의 연구내용 "자율지능 디지털 동반자를 위한 학습 및 시험 데이터 확보"을 디지털 동반자 기계학습을 위한 학습 데이터(training data)와 시험 데이터(test data)를 확보하는 것으로 해석하고 작성한 것임 [김성수]
      - 이러한 관점(training data/test data)에서, 데이터 정의(모델링), 통합, 신규데이터 생성, 세 분류로 나누어서 중점연구내용을 작성한 것임 [김성수]
      - 세부 내용에 대해서는 깊게 고민할 시간이 없었던 관계로 보강이 필요할 것임 [김성수]
    - **문제 정의 (Challenging Problems)**
      - 첫 꼭지 작성된 내용의 순서를 "서비스 도메인별 개념 모델링 및 사용자 인터랙션 유형별 메타 데이터 정의" 로 수정하면 좋겠음 [조준면 박사]
      - 첫 번째와 두 번째 꼭지의 내용이 도전적인 내용인지 표현이 되지 않았음. 도전적인 문제라는 것을 강조하는 것이 필요해 보임. 세 번째 꼭지는 그대로 진행해도 괜찮아 보임 [이창은 책임]
      - 멀티모달 표현 및 적응형 기계학습에 필요한 사용자 컨텍스트 데이터에 대한 강조가 필요함 [조영조 박사]
      - 의견 반영 [김성수]
        - 스마트 데이터 모델링, 통합, 처리 기술
          1. 멀티모달 기반 적응형 기계학습용 데이터 모델 부재
             → 멀티모달리티 대응 **인터랙션 유형별 메타데이터 정의**가 **사용자 컨텍스트 데이터** 모델링 요구
          2. 공유/보유 데이터의 데이터들간 이질성, 복잡성, 불완전성 산재 및 실시간 대응성 결여
             → 데이터 정제를 통한 **고품질 데이터 확보/통합과 실시간 개방형 데이터 연계/제공**이 필요
          3. 대규모 학습데이터 확보를 위한 신규 데이터 생성에 대한 요구 및 신규 데이터 생성에 따른 변경 대응력 부족 → **보유 데이터를 활용한** 신규데이터 생성 및 하향 호환성 유지
    - **중점추진 연구내용 (개요 슬라이드)**
      - RFP 내용에서 기존 RFP 큰 타이틀(꼭지)로 추가/수정이 필요함 [이창은 책임]
      - 윗 부분 연구내용에서 연구내용의 도전적인 내용(데이터 복잡성, 이질성 등)을 표현[해결해야 할 도전과제]해 줬으면 좋겠음 [김원영 박사]
        - 기존 그림을 삭제하고 그 부분에 해당 키워드 중심으로 추가하겠음 [김성수]
      - 스마트 데이터 관리 기술이 전체 시스템 구성도 상에 반영될 수 있도록 그림으로 데이터 연계(제공)되는 큰 그림이 그럴 필요가 있음 [김원영 박사, 이창은 책임]
        - 스마트 데이터 부분에 대한 별도 그림 작성 필요 [김원영 박사, 이창은 책임]
        - 추후 전체 그림에 반영될 수 있도록 큰 그림(데이터 연계, 제공 부분) 작성이 필요 [김원영 박사, 이창은 책임]
      - "주요 이슈 사항"에 대한 의견
        - 이슈사항이 문제점으로 보이지 않음. 해결해야 하는 이슈가 되는 내용으로 강조해야 함 [조영조 박사]
        - 데이터 규모/형식의 복잡성으로 기인한 문제들을 요약하여 핵심 도전문제를 정리하면 좋을 듯 [김원영 박사]
        - 데이터 연계를 제공하는 부분 (<u>복잡성, 다양성, 실시간성 등을 강조</u>**하여 정리** 필요) [이창은 책임]
        - 예를 들어, 현재 데이터 규모로는 대국민 서비스를 지원할 수준이 아님. 따라서, 도처에 산재해 있는 다양한 형식의 데이터를 제공하기 위해서 해결해야 할 주요한 문제점(도전과제)을 정리하면 좋겠음 [조영조 박사]
        - 이슈사항 첫 꼭지는 사용자 컨텍스트와 멀티 모달 데이터를 강조 포인트로 부각하면 좋겠음 [이창은 책임, 조영조 박사]
        - 두번째 꼭지, 다수의 데이터 소스 (공개/기업/공공기관)에 존재하는 데이터에 대한 접근 및 외부 제공 방법에 대한 고민이 필요함 [이창은 책임]
    - **중점추진 연구내용 (창의성 슬라이드)**
      - 데이터 통합 기술과 신규 데이터 생성 기술이 어떤 차이가 있는가? 직관적으로 차이를 이해 할 수 있도록 수정이 필요해 보임 [김원영 박사]
        - 데이터 통합에서의 이슈는 데이터 품질과 관련되며,
          - 불일치/불완전 데이터 삭제 등
        - 신규 데이터 생성은 학습 데이터의 규모를 확장하기 위한 방법임 [김성수]
      - 창의성 부분의 기술꼭지 타이틀이 단순히 연구꼭지를 기술하는 것이 아니라, 창의적인 내용을 반영하는 제목으로 수정이 필요 [이창은 책임]
        - 연구내용과 부합해서 작성한 것임 [김성수]
        - 해당 세부내용에 대한 창의적인 접근을 표현할 수 있는 내용으로 수정하겠음 [김성수]
      - 세부 내용에도 창의성을 강조할 수 있는 문구로 수정이 필요함. 예) 핵심 특허출원 예정 등 [이창은 책임]
        - 예를 들어, 깁스 샘플링을 적용하는 이유(차별성, 창의성)등을 수식하는 문구가 필요함 [이창은 책임]
      - 기술 내용에 대해서는 기존 RFP를 반영하여 잘 작성된 것 같음. 추가적으로 **<u>각 기술에 대한 표현(형용사 등)에 대해 부각될 수 있도록 수정</u>**이 필요해 보임 [조준면 박사]
    - **데이터 확보 방안**
      - 작성된 두 페이지를 한 페이지로 통합(첫 슬라이드 포맷으로)하고, 기업체 위주로 재구성 할 것 [이창은 책임, 조영조 박사]
    - **SW 품질관리계획**
      - "SPICE Level 3 인증 획득~~" 부분을 바로 아래 내용(sub item)을 요약한 제목으로 수정이 필요함 [이창은 책임]
        - "효과적인 연구개발 협업을 위한 시스템 구축 및 운용"으로 수정 [김성수]
    - **과제관리계획**
      - 엑소브레인, 딥뷰 과제등과 연계 표현이된 HPC 포맷을 활용하여 정리 [이창은 책임]
      - 해당 파일은 이메일로 보내 주겠음 [김원영 박사]
        - 메일 수신하는 대로 정리 하겠음 [김성수]
    - **기대효과**
      - 과제 제안서의 요약문을 정리할 것, 요약문을 이용해서 정리가 안된 듯함. 강조되는 내용이 없어 보임 [조영조 박사]
      - 강조해야 할 키워드들이 제대로 반영이 안되어 있음. 키워드(기술, 정량적으로 강조)으로 처리해 주기 바람 [조영조 박사]

One of the TF Team Meeting Logs

범례 의견 [의견 제시자]

작성 배경

- RFP의 연구내용 "자율지능 디지털 동반자를 위한 학습 및 시험 데이터 확보"을 디지털 동반자 기계학습을 위한 학습 데이터(training data)와 시험 데이터(test data)를 확보하는 것으로 해석하고 작성한 것임 [김성수]
- 이러한 관점(training data/test data)에서, 데이터 정의(모델링), 통합, 신규데이터 생성, 세 분류로 나누어서 중점연구내용을 작성한 것임 [김성수]
- 세부 내용에 대해서는 깊게 고민할 시간이 없었던 관계로 보강이 필요할 것임 [김성수]

문제 정의 (Challenging Problems)

- 첫 꼭지 작성된 내용의 순서를 "서비스 도메인별 개념 모델링 및 사용자 인터랙션 유형별 메타 데이터 정의" 로 수정하면 좋겠음 [조준면 박사]
- 첫 번째와 두 번째 꼭지의 내용이 도전적인 내용인지 표현이 되지 않았음. 도전적인 문제라는 것을 강조하는 것이 필요해 보임. 세 번째 꼭지는 그대로 진행해도 괜찮아 보임 [이창은 책임]
- 멀티모달 표현 및 적응형 기계학습에 필요한 사용자 컨텍스트 데이터에 대한 강조가 필요함 [조영조 박사]
- 스마트 데이터 모델링, 통합, 처리 기술
  1. 멀티모달 기반 적응형 기계학습용 데이터 모델 부재
  → 멀티모달리티 대응 **인터랙션 유형별 메타데이터 정의**가 **사용자 컨텍스트 데이터** 모델링 요구
  2. 공유/보유 데이터의 데이터들간 이질성, 복잡성, 불완전성 산재 및 실시간 대응성 결여
  → 데이터 정제를 통한 **고품질 데이터 확보/통합과 실시간 개방형 데이터 연계/제공**이 필요

범례 의견 [의견 제시자]

작성 배경

- RFP의 연구내용 "자율지능 디지털 동반자를 위한 학습 및 시험 데이터 확보"을 디지털 동반자 기계학습을 위한 학습 데이터(training data)와 시험 데이터(test data)를 확보하는 것으로 해석하고 작성한 것임 [김성수]
- 이러한 관점(training data/test data)에서, 데이터 정의(모델링), 통합, 신규데이터 생성, 세 분류로 나누어서 중점연구내용을 작성한 것임 [김성수]
- 세부 내용에 대해서는 깊게 고민할 시간이 없었던 관계로 보강이 필요할 것임 [김성수]

중점추진 연구내용 (창의성 슬라이드)

- 세부 내용에도 창의성을 강조할 수 있는 문구로 수정이 필요함. 예) 핵심 특허출원 예정 등 [이창은 책임]
  - 예를 들어, 깁스 샘플링을 적용하는 이유(차별성, 창의성)등을 수식하는 문구가 필요함 [이창은 책임]
- 기술 내용에 대해서는 기존 RFP를 반영하여 잘 작성된 것 같음. 추가적으로 각 기술에 대한 표현(형용사 등)에 대해 부각될 수 있도록 수정이 필요해 보임 [조준면 박사]

데이터 확보 방안

- 작성된 두 페이지를 한 페이지로 통합(첫 슬라이드 포맷으로)하고, 기업체 위주로 재구성 할 것 [이창은 책임, 조영조 박사]

SW 품질관리계획

- "SPICE Level 3 인증 획득~~" 부분을 바로 아래 내용(sub item)을 요약한 제목으로 수정이 필요함 [이창은 책임]
  - "효과적인 연구개발 협업을 위한 시스템 구축 및 운용"으로 수정 [김성수]

과제관리계획

- 엑소브레인, 딥뷰 과제등과 연계 표현이된 HPC 포맷을 활용하여 정리 [이창은 책임]
- 해당 파일은 이메일로 보내 주겠음 [김원영 박사]
  - 메일 수신하는 대로 정리 하겠음 [김성수]

기대효과

- 과제 제안서의 요약문을 정리할 것, 요약문을 이용해서 정리가 안된 듯함. 강조되는 내용이 없어 보임 [조영조 박사]
- 강조해야 할 키워드들이 제대로 반영이 안되어 있음. 키워드(기술, 정량적으로 강조)으로 처리해 주기 바람 [조영조 박사]

DataSets for Autonomous Digital Companion

Click stream
User activity
Generated content
Purchases
Clicks
Likes
Sensor data
. . .

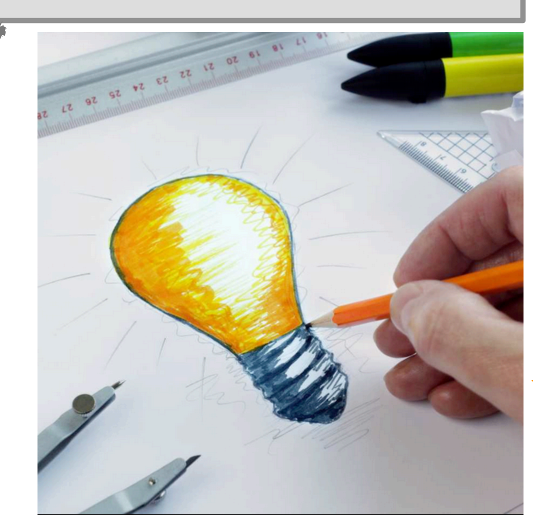
More Data

Better Analytics

Object Storage
Databases
Data Warehouse
Streaming analytics
BI
Hadoop
Spark/Presto
Elastic search

Machine Learning
Artificial Intelligence

Better Products



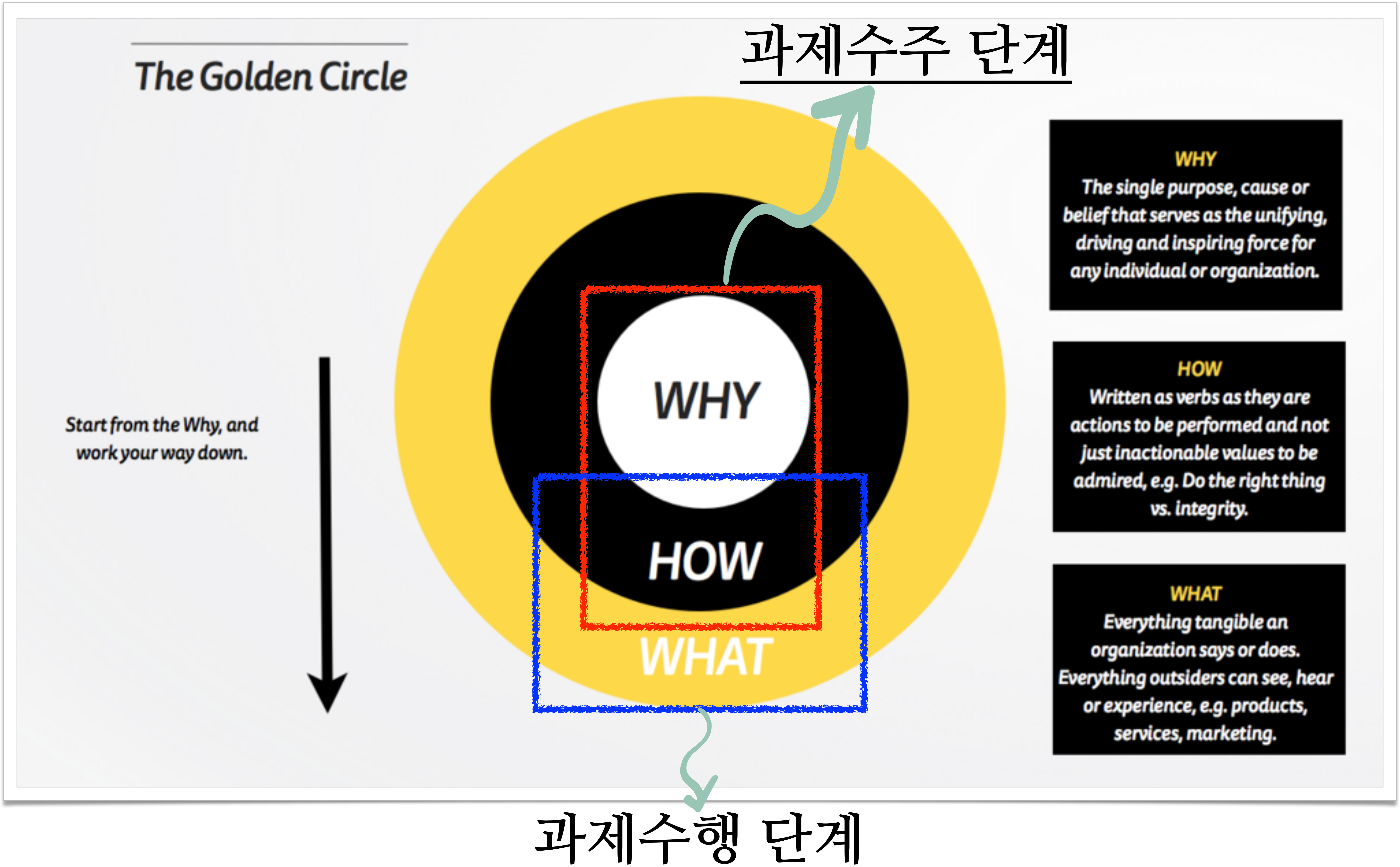
More Users

자율지능 디지털 동반자를 위한 학습 및 시험 데이터 확보

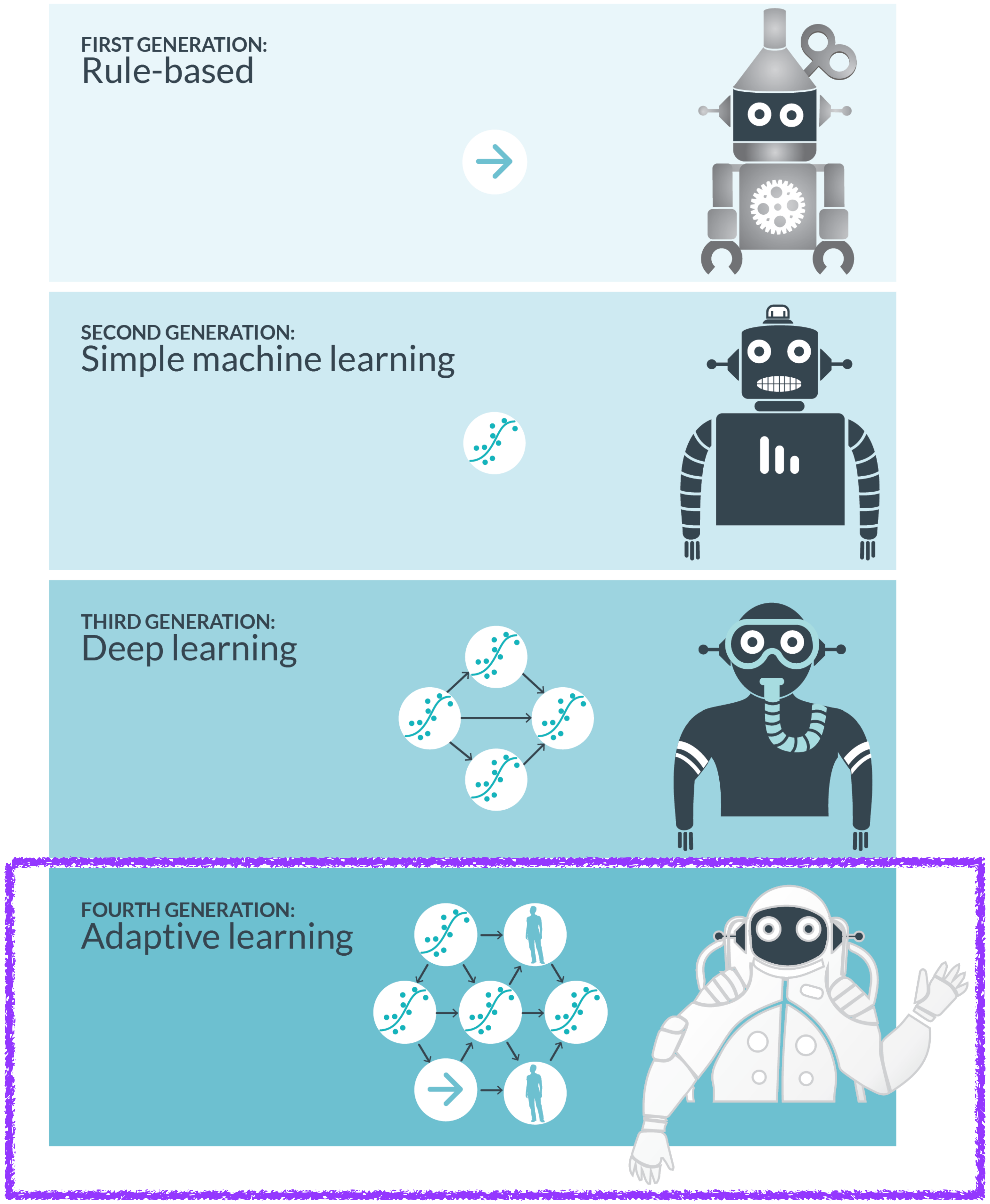
From My Perspective

The Golden Circle

'진행 단계별 (과제수주, 과제수행) 전략이 핵심이다!'

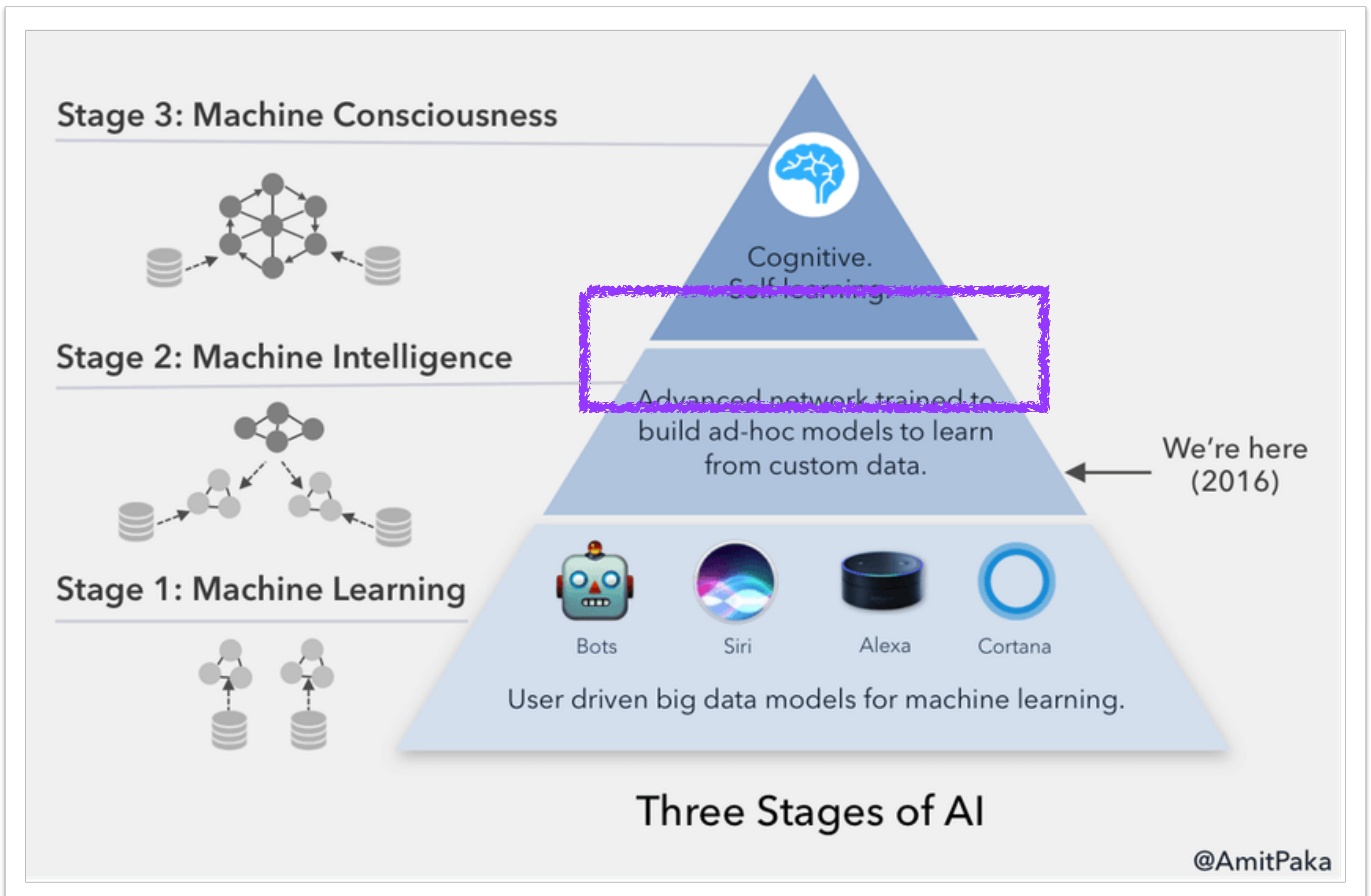


Four Generations in Machine Learning



Three Stages of Artificial Intelligence

- **First Stage:** Artificial *Narrow* Intelligence
- **Second Stage:** Artificial *General* Intelligence
- **Third Stage:** Artificial *Super* Intelligence



Modeling Multimodal Data and Dynamics

별첨 5. 세부 연구내용의 창의성 –WP4. 스마트 데이터 관리 기술

"3V가 3V를 만날때": 3V 빅데이터를 이용한 3V 멀티모달 데이터 모델링

빅데이터
Volume
Variety
Velocity

Toward Natural Interaction with DAISY!

인터랙션 메타데이터 모델링

VERBAL VOCAL VISUAL

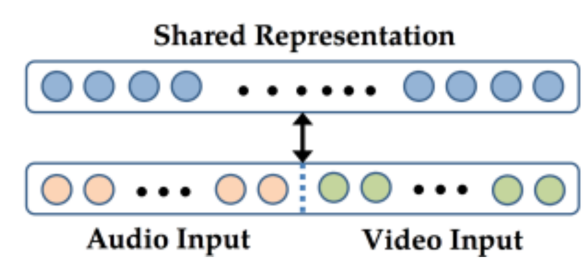
멀티모달 데이터
Verbal
Vocal
Visual

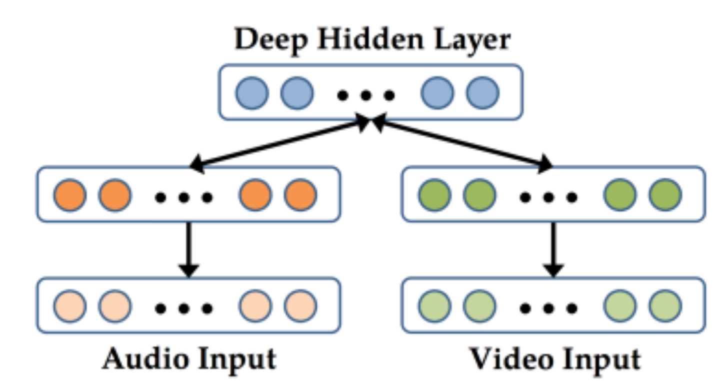
사용자
5W context
Preferences

사용자 컨텍스트 데이터

사용자 컨텍스트 데이터 모델링

Modeling Multimodal Dynamics!





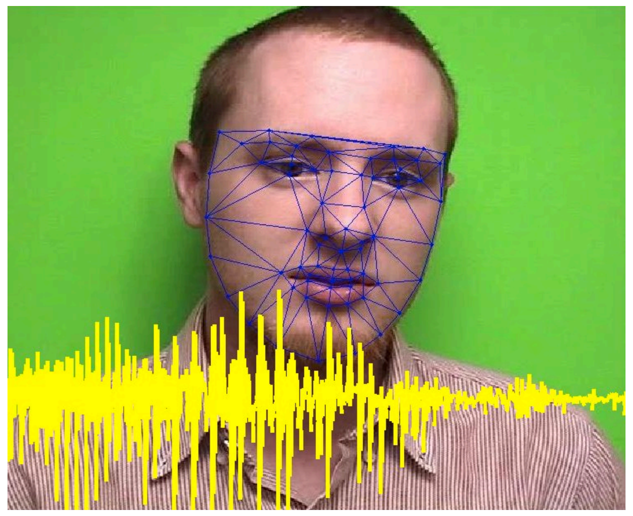
멀티모달 데이터를 이용한 기계학습 접근법

	Feature Learning	Supervised Training	Testing
Classic Deep Learning	Audio	Audio	Audio
	Video	Video	Video
Multimodal Fusion	A + V	A + V	A + V
Cross Modality Learning	A + V	Video	Video
	A + V	Audio	Audio
Shared Representation Learning	A + V	Audio	Video
	A + V	Video	Audio

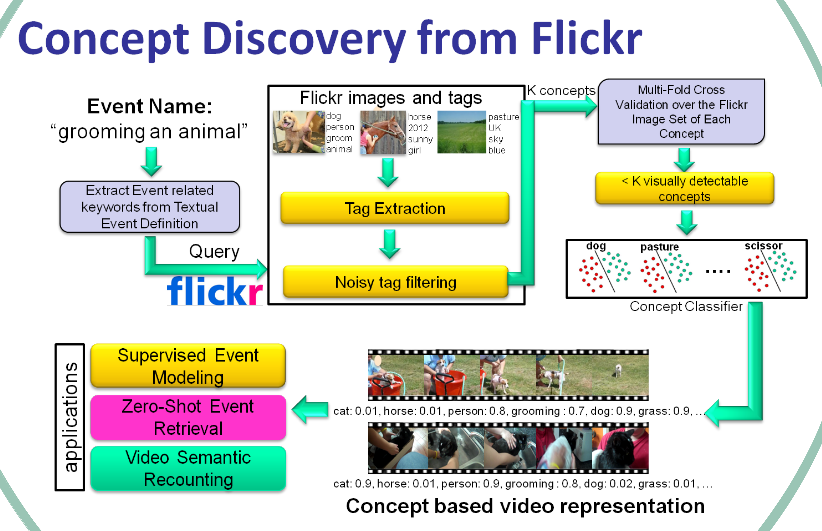
하이브리드 멀티모달/Personalized 데이터 모델링 기술

- 멀티모달 기계학습용 인터랙션 메타데이터 모델링
- 적응형 기계학습을 위한 사용자 컨텍스트(context) 데이터 모델링
- 응용서비스 도메인별 개념 모델링(concept modeling)
- 스마트 데이터 구축을 위한 애널리틱스 베이스 테이블(ABT) 모델링

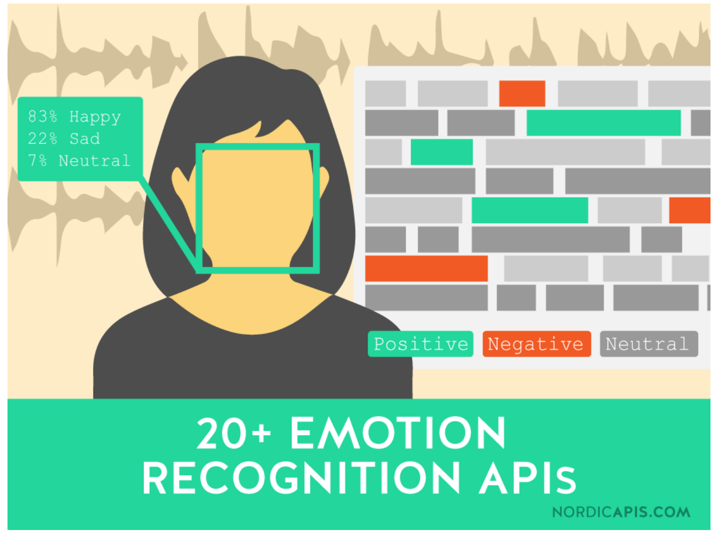
Multimodal Machine Learning



Audio-visual speech recognition

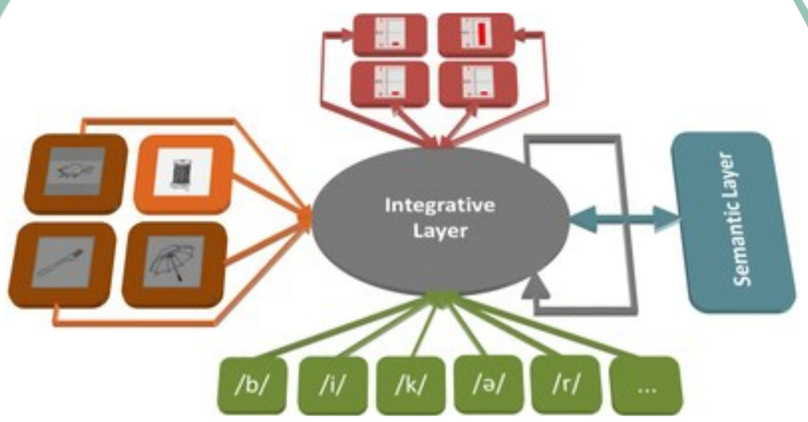


Cross-media event retrieval and tagging



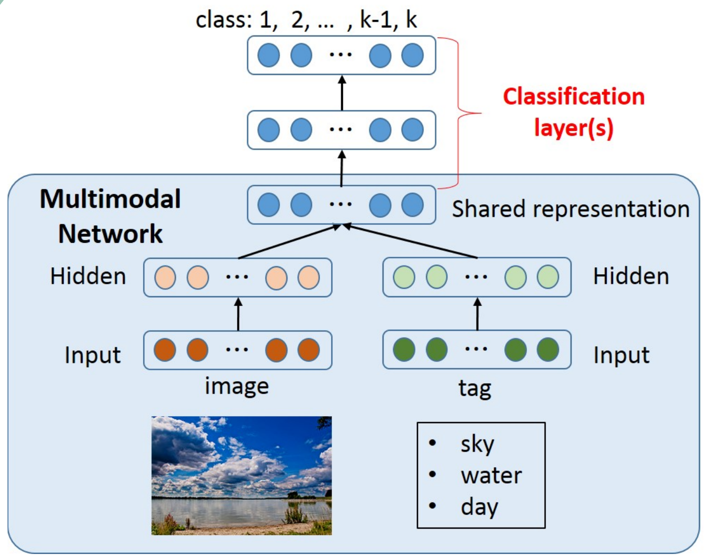
Audio-visual emotion recognition

Research Challenges



Multimodal language processing

Image and video captioning



Multimodal deep learning

Continuing to Learn.... Keep Learning!

Raise your standards as high as you can live with, avoid wasting your time on routine problems, and always try to work as closely as possible at the boundary of your abilities.

Do this because it is the only way of discovering how that boundary should be moved forward.

- Edsger W. Dijkstra (1930-2002)

рахмат
danke
謝謝
ngiyabonga
teşekkür ederim
spas
Баярлалаа
mersi
kia ora
barka
welalin
tack
mahalo
tapadh leat
спасибо
faafetai lava
vinaka
blagodaram
dank je
misaotra
matondo
paldies
grazzi
спасибі
kiitos
dankie
nanni
nandri
dhanyavad
thank you
gracias
хвала
asante
manana
bedankt
hvala
mauruuru
köszönöm
bayarlalaa
gràçie
akun
dankon
aĉiū
obrigada
murakoze
tenki
enkosi
djiere dieuf
tau
mochchakkeram
дякую
mamnun
chokrane
dziękuję
chnorakaloutioun
gratias ago
gràcies
sulpáy
go raibh maith agat
sobodi
dēkuji
sagolun
sukriya
kop khun krap
taiku
obrigado
arigatô
takk
dakujem
trugarez
mèsi
didi madloba
najis tuke
ありがとう
tanemirt
grazie
terima kasih
rahmet
ευχαριστώ
diolch
dhanyavadagalu
shukriya
mercê
мерси
kam sah hamnida
rahmat
তোমাকে ধন্যবাদ
감사합니다
xièxie
merci

Appendix

Probability: Frequentists vs. Bayesians

- **Frequentist Interpretation**
  - **Probabilities represent *long run frequencies* of events** $Prob(x) = \frac{occurrence(x)}{N}$
  - **Example. # of coin flips : *head* or *tail*?**
- **Bayesian Interpretation**
  - **Probability is used to quantify our *uncertainty* about something; hence it is fundamentally related to *information* rather than repeated trials.**
  - **The probability that certain events happen *cannot be computed* by a numeric frequency because we cannot repeat them**
  - **Bayesian approach focuses on computing *how uncertain* (or *certain*) the event may happen by means of *collected observation data* (*evidences*)**
  - **Bayes' Theorem**

$$p(\mathbf{w}|D) = \frac{p(D|\mathbf{w})p(\mathbf{w})}{p(D)}$$

$$p(\mathbf{w}|D) = \frac{p(D|\mathbf{w})}{p(D)} \cdot p(\mathbf{w})$$

(in a form of recurrence formula)

- ***w*: Target subject of uncertainty**
- *D*: Observed data
- *p*(***w***|*D*): **posterior** belief
  (updated belief after an observed data set is given)
- *p*(*D*|***w***): **likelihood**
- *p*(***w***): **prior** belief about an event ***w***
- *p*(*D*): **evidence**

Bayesian Probability

- **A sample example: "*guessing a gender of a person*"**
  - S#1. "A person is shown blinded"

$$p(female) = \frac{1}{2}$$

  - S#2. "The person is taller than 170cm!"

- Likelihood: $p(D|female) = \frac{1}{50}$
- Evidence: $p(D) = \frac{1}{10}$
- Posterior: $p(female|D) = \frac{1}{10}$

  - S#3. "The person's hair is longer than 30cm!"

- Likelihood: $p(D|female) = \frac{3}{5}$
- Evidence: $p(D) = \frac{1}{3}$
- Posterior: $p(female|D) = \frac{9}{50}$

  - S#4. "The person is wearing a skirt!"

- Likelihood: $p(D|female) = \frac{3}{4}$
- Evidence: $p(D) = \frac{1}{3}$
- Posterior: $p(female|D) = \frac{81}{200}$