MBA Lecture Note

Introduction to Big Data and Artificial Intelligence

Lecturer: Jinkyu Kim
Hanyang University, Ph.D. student

Contents

Ⅰ. Big Data Revolution: Trend

Ⅱ. Terminology Description

Ⅲ. Alternative Datasets

IV. Machine Learning Techniques

Ⅴ. Supervised Learning

VI. Unsupervised Learning

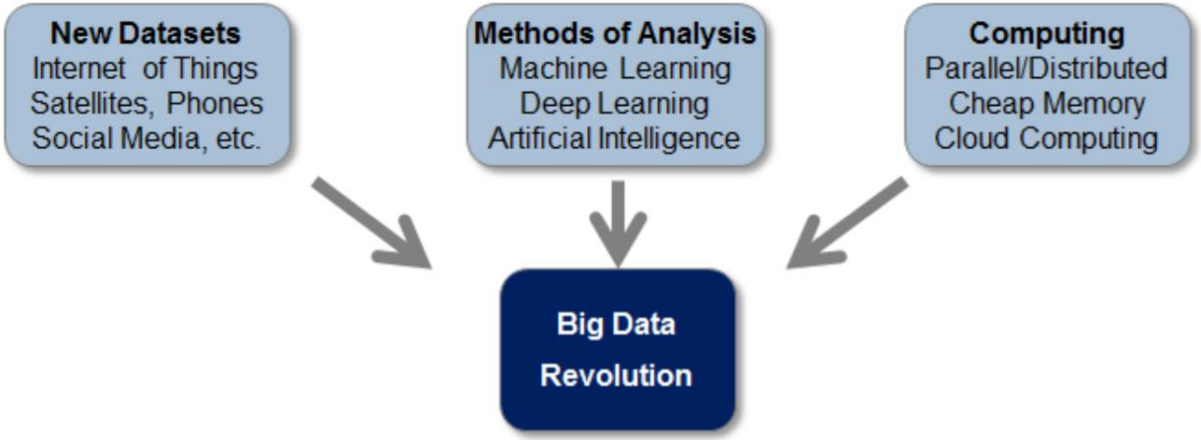
VII. Deep Learning

VIII. Application

00. Epilogue

1. Big Data Revolution: Trend

Figure 1: Factors leading to Big Data Revolution



New Datasets
Internet of Things
Satellites, Phones
Social Media, etc.

Methods of Analysis
Machine Learning
Deep Learning
Artificial Intelligence

Computing
Parallel/Distributed
Cheap Memory
Cloud Computing

Big Data Revolution

Source: J.P.Morgan Macro QDS.

□ Exponential increase in amount of data available

□ Increase in computing power and data storage capacity, at reduced cost

□ Advancement in Machine Learning methods to analyze complex datasets

1. Big Data Revolution: Trend

□ Amount of data

- 90% of the data in the world today: created in the past two years

- Expected: 4.4 zettabytes (2015) => 44 zettabytes (2020)
  cf) 1 zettabyte = 1,000,000,000,000 Gb

- Further Acceleration:
  IoT (Internet of Things), smart phones, cost reduction in satellite techs

1. Big Data Revolution: Trend

□ Computing power & Storage "Cloud Computing" – Physical Ability

- Remote, shared access to resources

  ex) A single web search on Google: coordinated across ~1000 computers
  ex) Software-as-a-service (SaaS), Platform-as-a-service (PaaS), etc.

- Diminished the barriers to large-scale data processing and analytics

1. Big Data Revolution: Trend

□ Machine Learning methods – Software Ability

- Part of Statistics and Computer Science

- Enable analysis of large and unstructured datasets

- Classical Machine Learning, Deep Learning, Reinforcement Learning

- Ex) Amazon Echo, Google Home, Apple Siri, …

"Just 0.5% of the data produced is currently being analyzed [Regalado (2013)]"
- Compelling reason for learning about new datasets and ML toolkits.

2. Terminology Description

□ “Big” Data

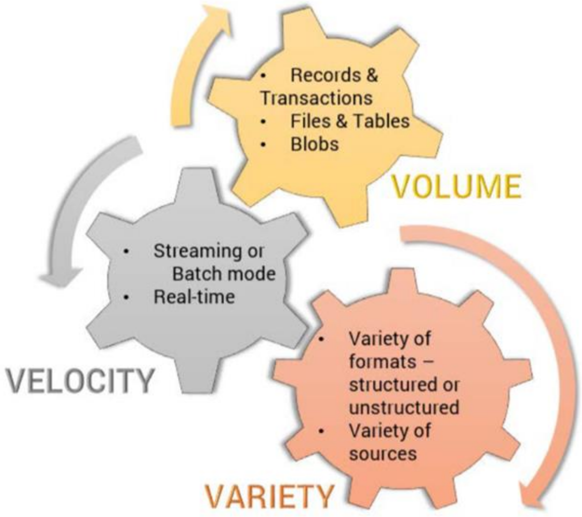
□ Machine Learning

□ Artificial Intelligence

2. Terminology Description

□ “Big” Data: What is Big?

Figure 2: Features of Big Data



Source: J.P.Morgan Macro QDS.

- Volume: the size of data is huge
- Velocity: the speed with which data is sent or received
- Variety: variety of formats (structured, semi-structured, unstructured)

2. Terminology Description

□ Machine Learning (ML)

- Goal: enable computers to learn from their experience in certain tasks

- Enable the machine to improve performance as their experience grows

- Category: Supervised Learning, Unsupervised Learning, Deep and Reinforcement Learning

2. Terminology Description

□ Artificial Intelligence (AI)

- Broader scheme of enabling machines with human-like intelligence

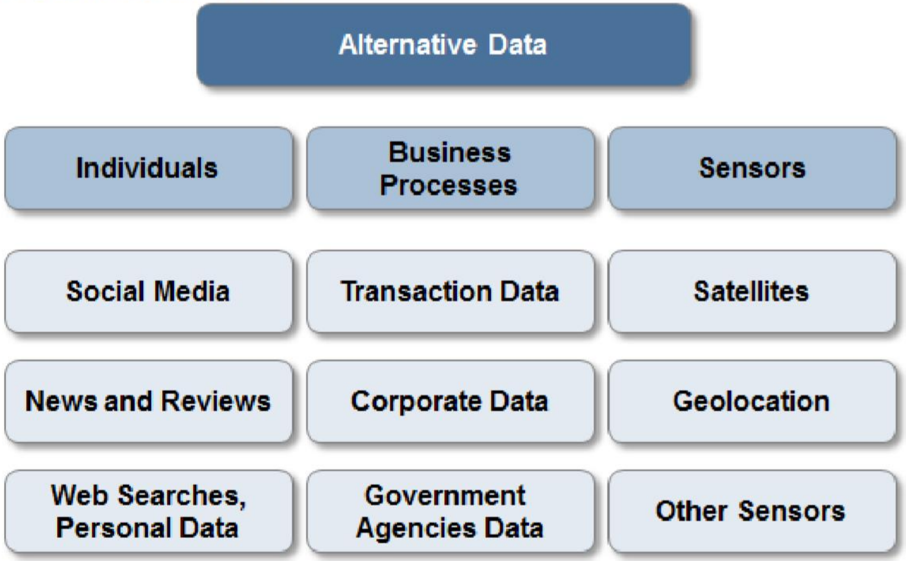
- At the first stage: hardcoding a large number of rules "Symbolic-AI"

- Machine Learning is another attempt to achieve AI

- Machine Learning & Deep Learning so far represent the most serious attempt at achieving AI (image&pattern recognition, language trans., etc.)

3. Classification of Alternative Data Sets

Figure 3: Classification of Big/Alternative Data sources



Alternative Data

Individuals	Business Processes	Sensors
Social Media	Transaction Data	Satellites
News and Reviews	Corporate Data	Geolocation
Web Searches, Personal Data	Government Agencies Data	Other Sensors

Source: J.P.Morgan Macro QDS.

□ Data generated by individuals

□ Data generated by Business Processes

□ Data generated by sensors

3. Classification of Alternative Data Sets

□ Data generated by individuals

- Mostly recorded through textual mediums (unstructured & distributed across multiple platforms)

□ Possible sources

- Social media: Twitter, Facebook, LinkedIn, etc.
- Specialized sites: Business-reviewing website Yelp, E-commerce groups like Amazon, Mobile App analytics companies like App annie
- Web searches, and personalized data such as Google Search trends, etc.

3. Classification of Alternative Data Sets

□ Data generated by Business Processes

- produced by corporations and public entities (often highly structured)

□ Examples

- transaction records such as credit card data
- banking records
- supermarket scanner data
- supply chain data

3. Classification of Alternative Data Sets

□ Data generated by sensors

- mechanically collected data through sensors embedded in various devices (typically unstructured & size is massive)

□ Examples

- satellite imaging
- geolocation data
- cameras fixed at a location (ex. CCTV)
- weather and pollution data
- IoT (Internet of Things)

4. Classification of Machine Learning Techniques

□ Necessity of New Methods

- Large and less structured data: hard to handle with spreadsheet program

- New methods of automatically analyzing unstructured data are needed

- ML yielded some spectacular results
  (image & pattern recognition, natural language processing, auto-driving)

"What is the application of ML in finance,
and how do these methods differ from each other?"

4. Classification of Machine Learning Techniques

□ First stage: Symbolic Artificial Intelligence – not a ML

- We can instruct a computer to perform certain tasks based on rules.

- Why don't we give machines a large number of rules so that it appears to be "intelligent"? (This is a so-called "Symbolic AI")

- However, it will freeze the first time it encounters a situation that does not exactly match a set of pre-programmed rules.

- This is not a Machine Learning. It's just making a lot of rules.

=> Not a big success

4. Classification of Machine Learning Techniques

□ Machine Learning

- the computer is given an input (set of variables and datasets) and output (consequences of input variables)

- The machine then finds or "learn" a rule that link input and output

- ML has 3 important sub-categories, i.e., supervised learning, unsupervised learning, deep learning

4. Classification of Machine Learning Techniques

□ Supervised Learning

- answer sheet
- trying to find a rule, an "equation" that we can use to predict a variable.

□ Unsupervised Learning

- no answer sheet
- uncovering the structure of data

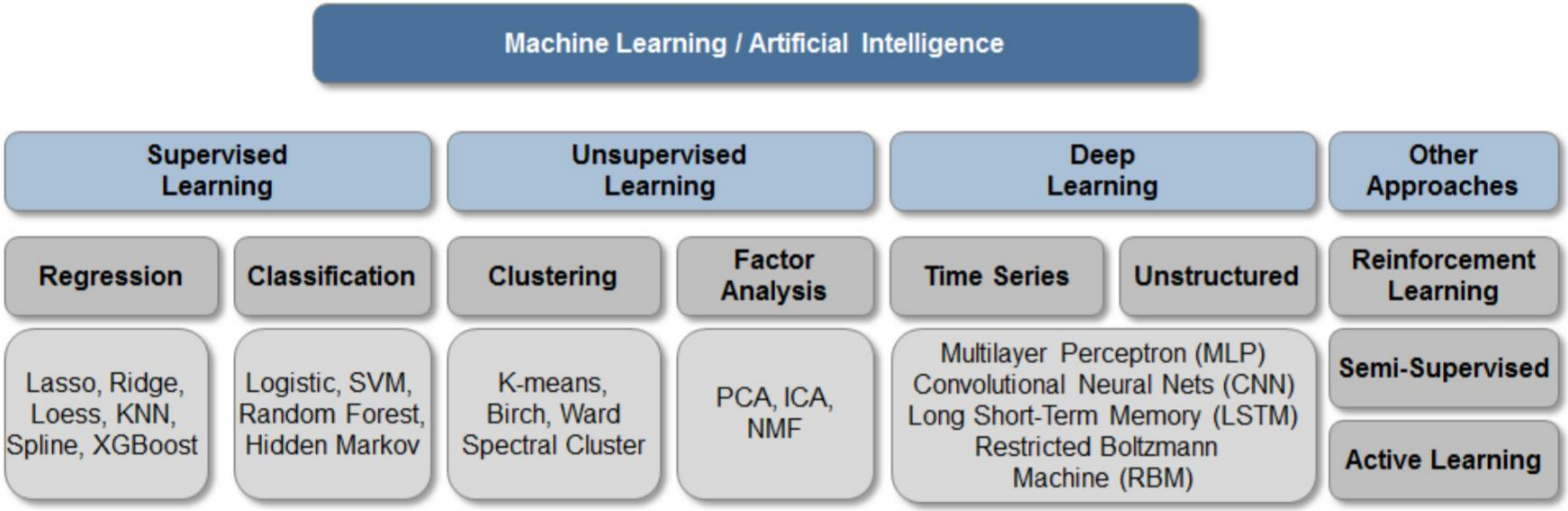
□ Deep Learning

- similar to how people learn
- ex) A child will learn concept of face by identifying some simple features like eyes, nose, mouth, etc. From simple features one builds a more complex concepts such as face.

4. Classification of Machine Learning Techniques

□ Classification of ML techniques

Figure 7: Classification of Machine Learning techniques



Machine Learning / Artificial Intelligence

Supervised Learning
- Regression: Lasso, Ridge, Loess, KNN, Spline, XGBoost
- Classification: Logistic, SVM, Random Forest, Hidden Markov

Unsupervised Learning
- Clustering: K-means, Birch, Ward Spectral Cluster
- Factor Analysis: PCA, ICA, NMF

Deep Learning
- Time Series
- Unstructured
- Multilayer Perceptron (MLP) Convolutional Neural Nets (CNN) Long Short-Term Memory (LSTM) Restricted Boltzmann Machine (RBM)

Other Approaches
- Reinforcement Learning
- Semi-Supervised
- Active Learning

Source: J.P.Morgan Macro QDS

5. Supervised Learning

□ Supervised Learning

- Algorithm is provided with historical data (both input and output variables) and is trying to find the relationship that has the best predictive power for out of the sample data

□ Regression and Classification

- Regressions: try to predict output variables based on a number of input variables (ex. predict asset returns)

- Classification: attempt to group or classify output into categories (ex. decide buy/sell actions)

5. Supervised Learning-Regression

□ Linear Regression

- Even a simple linear regression can be thought of as a Supervised Machine Learning method.

- Sometimes, it is too simplistic to uncover the true relationship.

$$y = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \varepsilon.$$

OLS: Minimize Historical Sum of $\left(y - \left(\beta_0 + \sum_{i=1}^{n} \beta_i x_i\right)\right)^2.$

5. Supervised Learning-Regression

□ Penalized Regression

$$y = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \varepsilon.$$

OLS: Minimize Historical Sum of $\left(y - \left(\beta_0 + \sum_{i=1}^{n} \beta_i x_i\right)\right)^2$.

Lasso: Minimize Historical Sum of $\left(y - \left(\beta_0 + \sum_{i=1}^{n} \beta_i x_i\right)\right)^2 + \alpha \sum_{i=1}^{n} |\beta_i|$,

Ridge: Minimize Historical Sum of $\left(y - \left(\beta_0 + \sum_{i=1}^{n} \beta_i x_i\right)\right)^2 + \alpha \sum_{i=1}^{n} \beta_i^{\,2}$,

Elastic Net: Minimize Historical Sum of $\left(y - \left(\beta_0 + \sum_{i=1}^{n} \beta_i x_i\right)\right)^2 + \alpha_1 \sum_{i=1}^{n} |\beta_i| + \alpha_2 \sum_{i=1}^{n} |\beta_i|^2$.

- the optimizer will set unnecessary and very large betas to zero

5. Supervised Learning-Regression

□ K-Nearest Neighbor (KNN)

- Given a new datapoint, we search through historical data and identify a number of K of similar instances
- Once we have located K nearest neighbors, we can average the output variable y for this subset and use that as our prediction

□ LOESS

- Using data for K-nearest neighbors, for each new point we fit a linear regression based on the K nearest neighbors, and predict the output using those coefficients

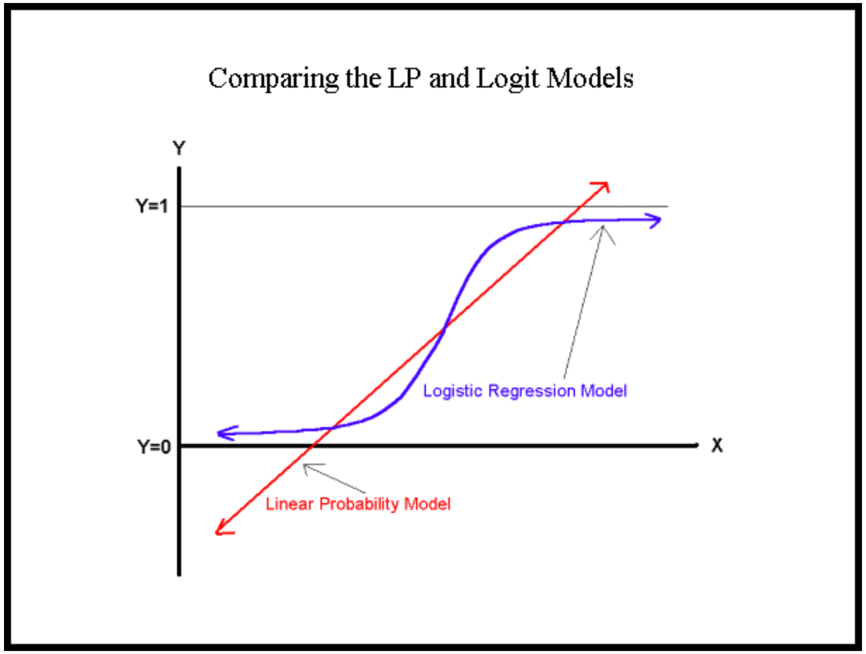
=> Simple way to extrapolate analysis on past similar events

5. Supervised Learning-Classification

□ Logistic Regression

- produces output as a binary decision such as “buy” or “sell”

- forecast the probability of an event given historical sample data



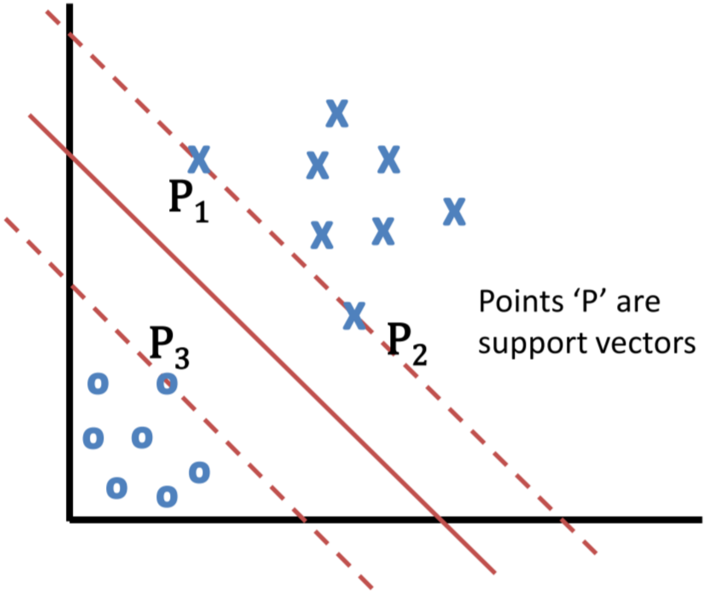
5. Supervised Learning-Classification

□ Support Vector Machine (SVM)

- the most popular classification algorithm

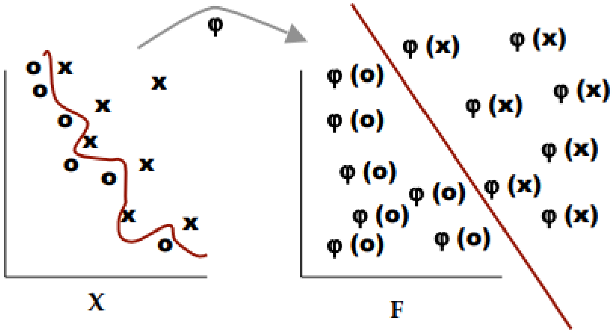
- enhance with new derived features / axis-transform / higher dimension

Figure 58: Simplified illustration of SVM classification



Points ‘P’ are support vectors

Source: J.P.Morgan Macro QDS



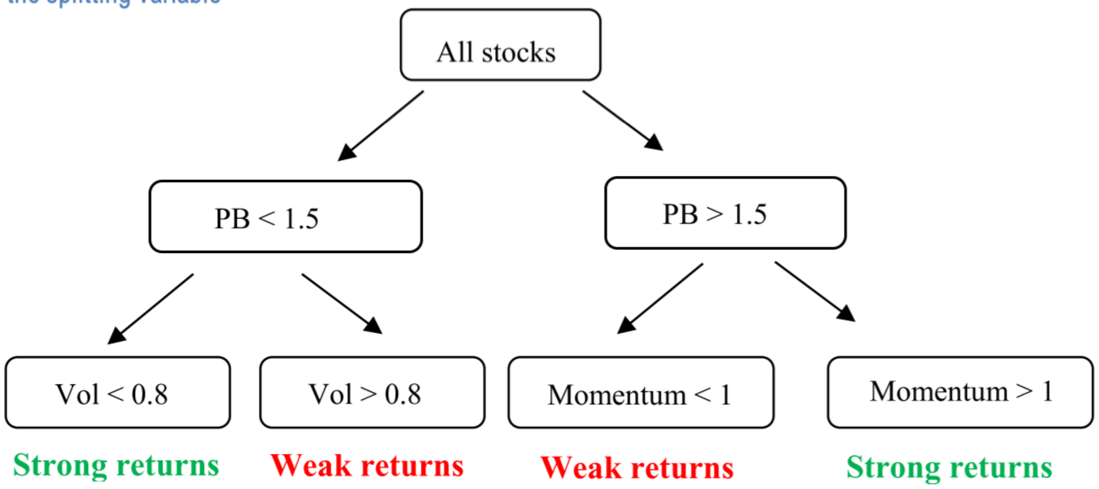
5. Supervised Learning-Classification

□ Decision Tree

- Essentially, flow charts used commonly in business management

- To fit a decision tree, the algorithm usually looks for the best variable and the best splitting value among all possibilities, so that a particular loss function is minimized

Figure 63: Example of decision tree to classify whether future stock returns are good or poor. In each node, we split the data into two subsets depending on the value of the splitting variable



Source: J.P.Morgan Macro QDS, FactSet

5. Supervised Learning-Classification

□ Random Forests

- A single decision tree is capture the structure of the in-sample data very well, but it tends to work poorly out-of-sample (overfitting)

- Bagging: draw many random subsets of the original data

- Fit a decision tree on each subset

- Average the predictions of all decision trees

=> Reducing the variances of the prediction

6. Unsupervised Learning

□ Unsupervised Learning

- Identify relationships between variables and their common drivers
- In unsupervised learning, the machine is simply given the entire data sets and it does not have a notion of what are independent and what are the dependent variables

□ Clustering and Factor analysis

- Clustering: splitting a dataset into smaller groups based on similarity
- Factor Analysis: identify the main drivers of the data

6. Unsupervised Learning-Clustering

□ Clustering

- In the supervised learning, the algorithm tries to minimize the sum of squared errors, i.e., (predicted value – actual value)^2.

- Now, we have no answer sheet. We don't know what actual values are.

- In this situation, how can we compute errors?

6. Unsupervised Learning-Clustering

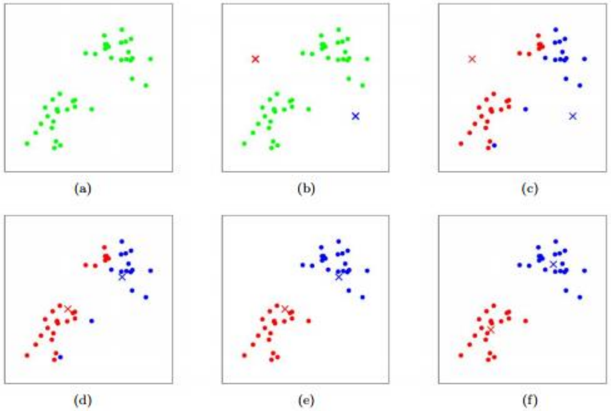
□ K-means clustering

- simplest clustering algorithm

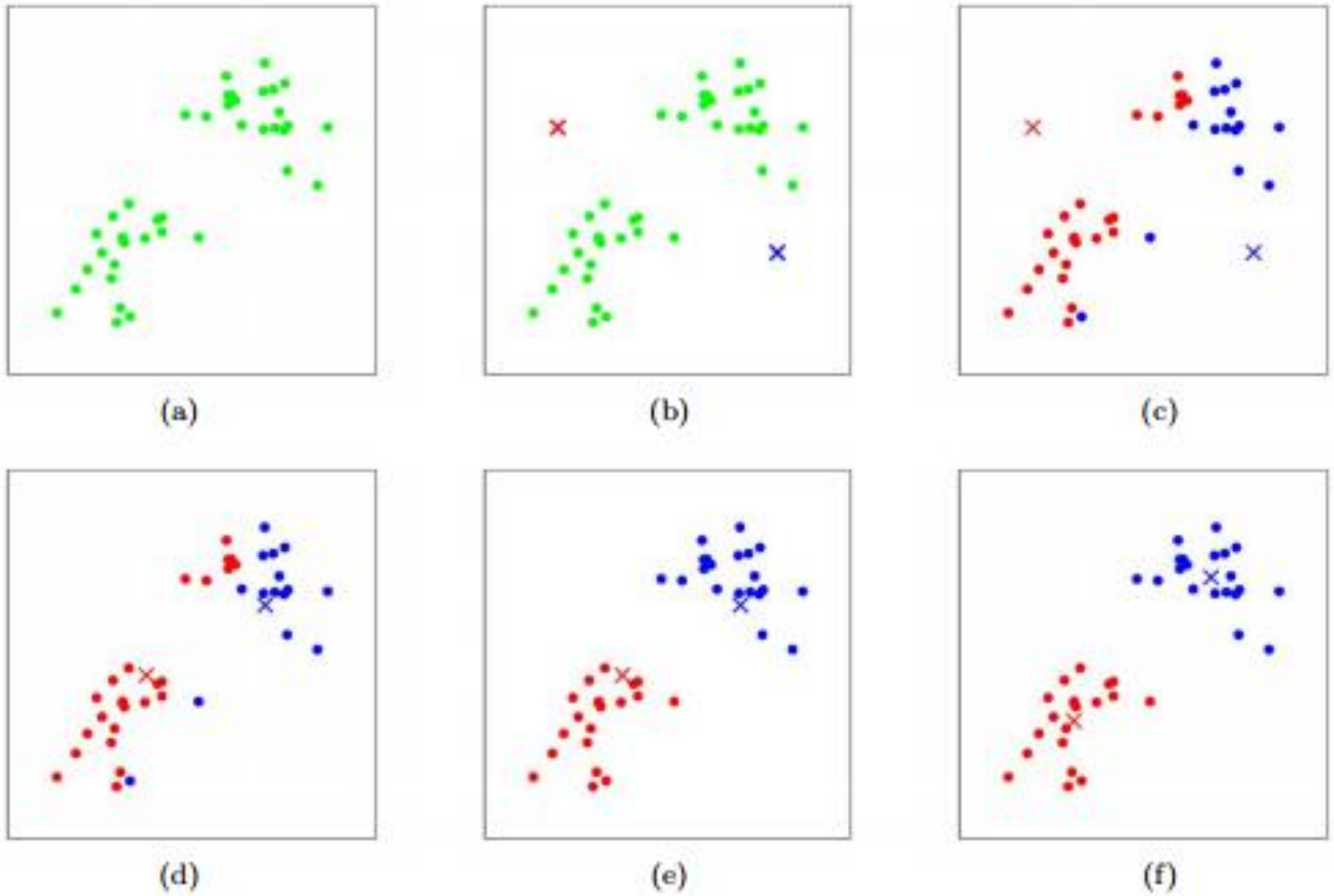
- algorithm starts by initially marking random points as centroids

- it iteratively does a two-step calculations

  1. maps points to closest centroids
  2. redefines the centroid as the mean of the points mapped to it



6. Unsupervised Learning-Clustering



(a)

(b)

(c)

(d)

(e)

(f)

6. Unsupervised Learning-Factor Analysis

□ Factor Analysis

- Goal: identify the main drivers of the data or identify the best representation of data

□ Principal Component Analysis

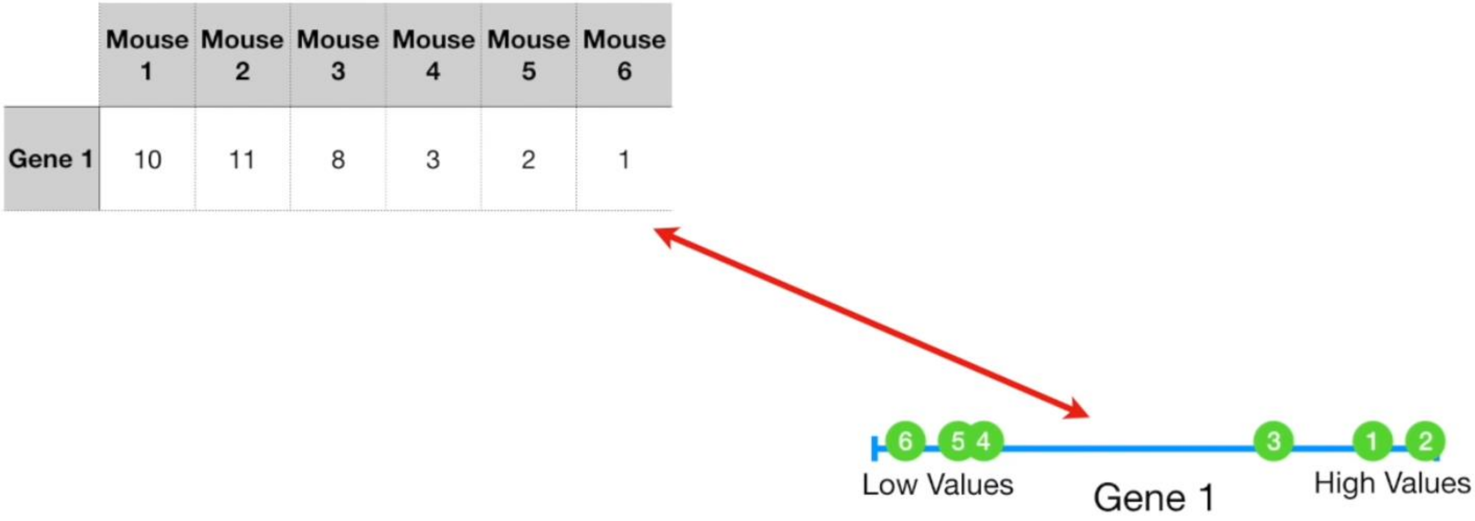
- a very well-known method of factor analysis
- originally a statistical tool for dimensionality reduction

6. Unsupervised Learning-Factor Analysis

□ Principal Component Analysis - Intuition

- Assume we want to categorize observations based on similarities

- If we have only one characteristic, that characteristic is the only reference so that we can compare each group based on that reference

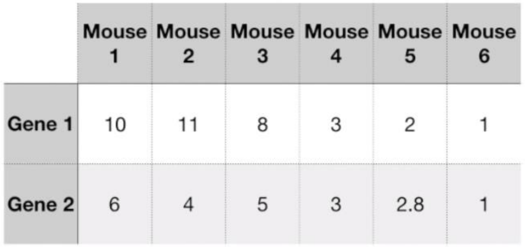


	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	10	11	8	3	2	1

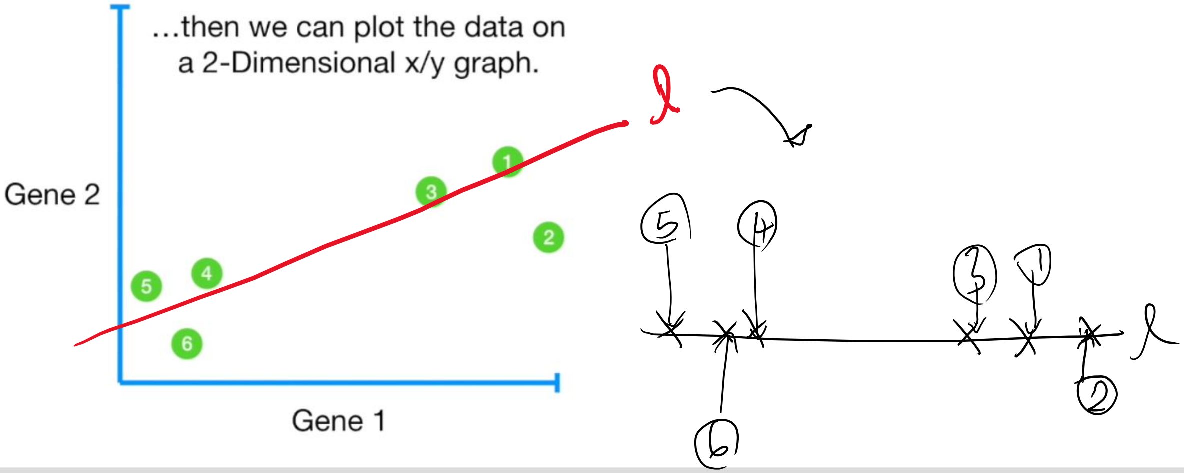
6. Unsupervised Learning-Factor Analysis

□ Principal Component Analysis - Intuition

- If we have two characteristics, we can compare it with two axes.
- KEY IDEA: also, we can compare it with the single reference line “l”.
- This single line is roughly the concept of “Principal Component” (PC)
- We can compare two characteristics with just one PC. But, what if there are n characteristics?



	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	10	11	8	3	2	1
Gene 2	6	4	5	3	2.8	1



6. Unsupervised Learning-Factor Analysis

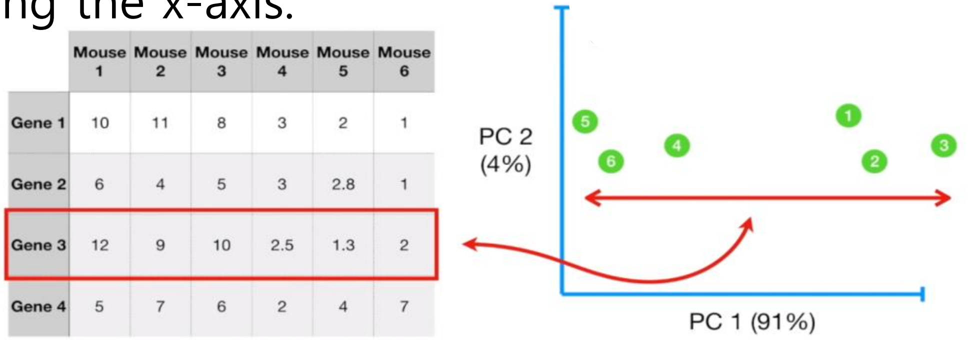
□ Principal Component Analysis - Intuition

- PCA is the technique that extract the most valuable characteristics or the combination of characteristics for clustering the data. (high variation)

- That is, PC can be thought of as the most important part of the variation that explains the variation of characteristics of entire samples

- Once we finish the extraction of PC, we can compare observations based on the extracted PC.

- For example, PCA might tell us that Gene3 is responsible for separating samples along the x-axis.



	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	10	11	8	3	2	1
Gene 2	6	4	5	3	2.8	1
Gene 3	12	9	10	2.5	1.3	2
Gene 4	5	7	6	2	4	7

7. Deep Learning

□ Deep Learning

- analyzes data in multiple layers of learning

- simpler concepts => combining => complex and abstract

□ Deep Learning vs. Classical Machine Learning

- Classical machine learning perform well on numerical prediction and analysis tasks

- But not at “intuitive” things (ex. recognizing images, analyzing texts)

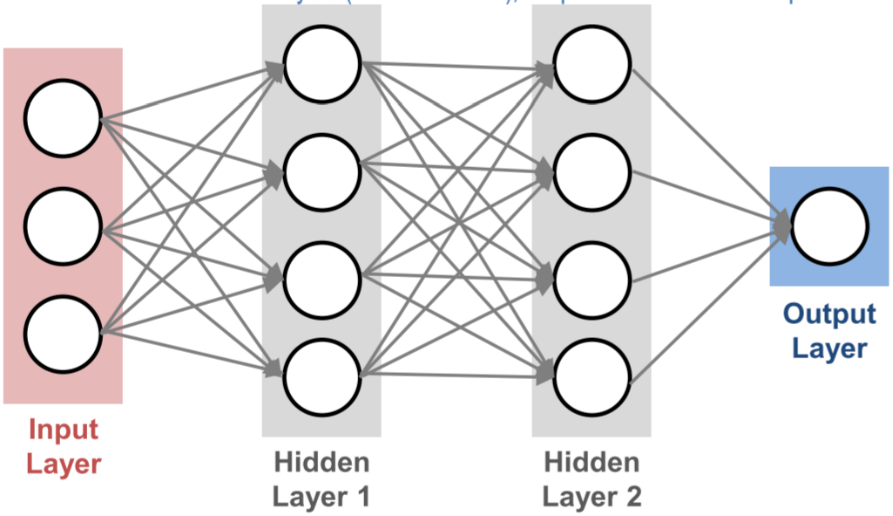
- Goal of Deep Learning: perform tasks that are difficult for people to define, but easy to perform

7. Deep Learning

□ How it works

- Neuron: a simple calculator that computes **first** the weighted average of inputs given to it and **then** output the result if the value exceeds a pre-determined threshold

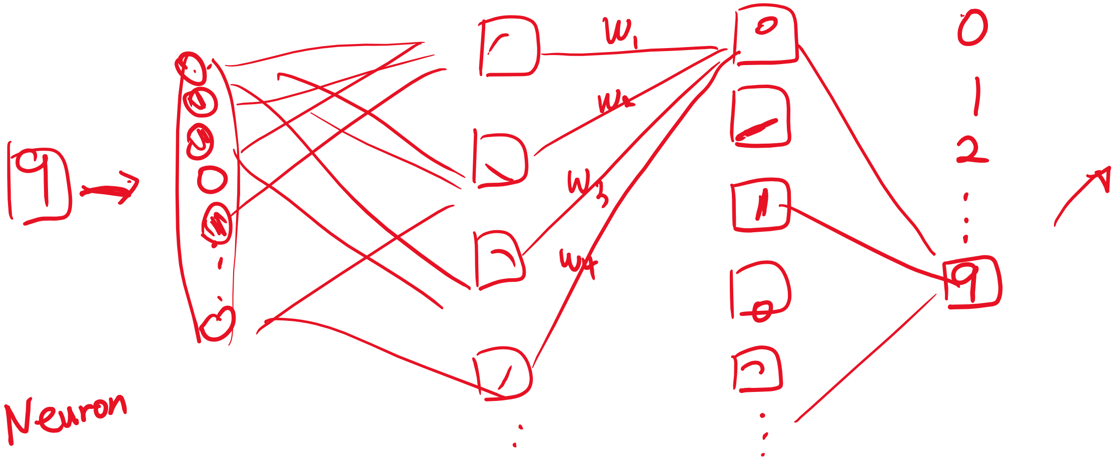
Figure 85: Example of a Neural Network with 2 hidden layers (4 neurons each), 3 input variables and 1 output variable



Source: J.P.Morgan Macro QDS

7. Deep Learning

□ Simple illustration : Hand-written number recognition



$\min \| x_i - y_i \|$

Each Neuron

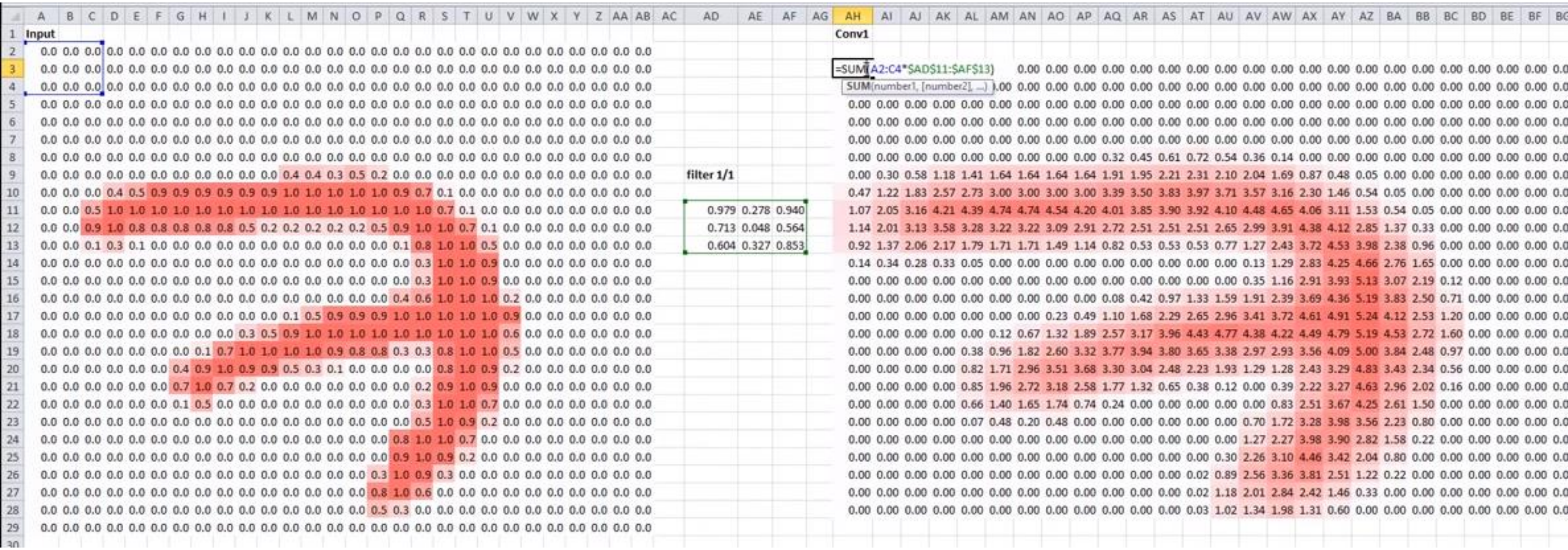
$$\max\left[\underset{\uparrow \text{sigmoid}}{\sigma}\left(w_1 a_1 + w_2 a_2 + \cdots + w_n a_n - \underset{\uparrow \text{bias}}{10}\right), 0\right] \Leftrightarrow a^{(1)} = \sigma\left(W a^{(0)} + b\right)$$

- Learning = Finding the appropriate weights and biases, i.e., W & b
- MLP (Multilayer Perceptron)

7. Deep Learning

□ CNN (Convolutional Neural Network): Extension1

- Let's use convolutional filter as a layer to detect the edge of images well



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB
1	Input																											
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.4	0.4	0.3	0.5	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.4	0.5	0.9	0.9	0.9	0.9	0.9	0.9	1.0	1.0	1.0	1.0	1.0	0.9	0.7	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11	0.0	0.0	0.5	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.7	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
12	0.0	0.0	0.9	1.0	0.8	0.8	0.8	0.8	0.8	0.5	0.2	0.2	0.2	0.2	0.2	0.5	0.9	1.0	1.0	0.7	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
13	0.0	0.0	0.1	0.3	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.8	1.0	1.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.3	1.0	1.0	0.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0
15	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.3	1.0	1.0	0.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0
16	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.4	0.6	1.0	1.0	1.0	0.2	0.0	0.0	0.0	0.0	0.0	0.0
17	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.5	0.9	0.9	0.9	1.0	1.0	1.0	1.0	1.0	0.9	0.0	0.0	0.0	0.0	0.0	0.0
18	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.3	0.5	0.9	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.6	0.0	0.0	0.0	0.0	0.0	0.0
19	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.7	1.0	1.0	1.0	1.0	0.9	0.8	0.8	0.3	0.3	0.8	1.0	1.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0
20	0.0	0.0	0.0	0.0	0.0	0.0	0.4	0.9	1.0	0.9	0.9	0.5	0.3	0.1	0.0	0.0	0.0	0.0	0.8	1.0	0.9	0.2	0.0	0.0	0.0	0.0	0.0	0.0
21	0.0	0.0	0.0	0.0	0.0	0.0	0.7	1.0	0.7	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.9	1.0	0.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0
22	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.3	1.0	1.0	0.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0
23	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	1.0	0.9	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0
24	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	1.0	1.0	0.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	1.0	0.9	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
26	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.3	1.0	0.9	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
27	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	1.0	0.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
28	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
29	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

	AD	AE	AF
9	filter 1/1		
11	0.979	0.278	0.940
12	0.713	0.048	0.564
13	0.604	0.327	0.853

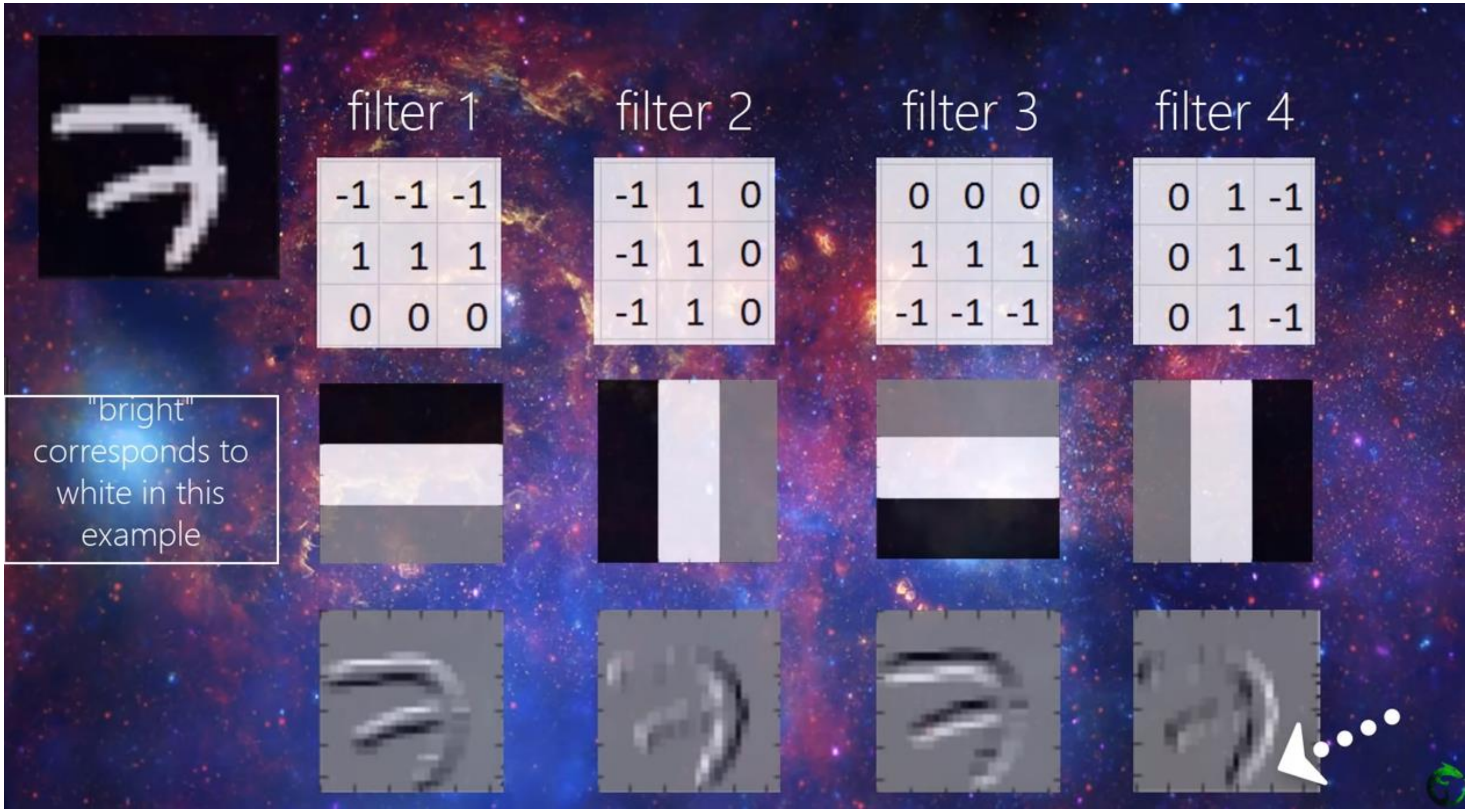
	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU
1	Conv1													
3	=SUM(A2:C4*$AD$11:$AF$13)						0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	SUM(number1, [number2], ...)					.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.32	0.45	0.61	0.72	0.54
9	0.00	0.30	0.58	1.18	1.41	1.64	1.64	1.64	1.64	1.91	1.95	2.21	2.31	2.10
10	0.47	1.22	1.83	2.57	2.73	3.00	3.00	3.00	3.00	3.39	3.50	3.83	3.97	3.71
11	1.07	2.05	3.16	4.21	4.39	4.74	4.74	4.54	4.20	4.01	3.85	3.90	3.92	4.10
12	1.14	2.01	3.13	3.58	3.28	3.22	3.22	3.09	2.91	2.72	2.51	2.51	2.51	2.65
13	0.92	1.37	2.06	2.17	1.79	1.71	1.71	1.49	1.14	0.82	0.53	0.53	0.53	0.77
14	0.14	0.34	0.28	0.33	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
16	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.42	0.97	1.33	1.59
17	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.23	0.49	1.10	1.68	2.29	2.65	2.96
18	0.00	0.00	0.00	0.00	0.00	0.12	0.67	1.32	1.89	2.57	3.17	3.96	4.43	4.77
19	0.00	0.00	0.00	0.00	0.38	0.96	1.82	2.60	3.32	3.77	3.94	3.80	3.65	3.38
20	0.00	0.00	0.00	0.00	0.82	1.71	2.96	3.51	3.68	3.30	3.04	2.48	2.23	1.93
21	0.00	0.00	0.00	0.00	0.85	1.96	2.72	3.18	2.58	1.77	1.32	0.65	0.38	0.12
22	0.00	0.00	0.00	0.00	0.66	1.40	1.65	1.74	0.74	0.24	0.00	0.00	0.00	0.00
23	0.00	0.00	0.00	0.00	0.07	0.48	0.20	0.48	0.00	0.00	0.00	0.00	0.00	0.00
24	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.30
26	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.89
27	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	1.18
28	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	1.02

	AV	AW	AX	AY	AZ	BA	BB	BC	BD	BE	BF	BG
3	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0
4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0
5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0
6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0
7	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0
8	0.36	0.14	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0
9	2.04	1.69	0.87	0.48	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.0
10	3.57	3.16	2.30	1.46	0.54	0.05	0.00	0.00	0.00	0.00	0.00	0.0
11	4.48	4.65	4.06	3.11	1.53	0.54	0.05	0.00	0.00	0.00	0.00	0.0
12	2.99	3.91	4.38	4.12	2.85	1.37	0.33	0.00	0.00	0.00	0.00	0.0
13	1.27	2.43	3.72	4.53	3.98	2.38	0.96	0.00	0.00	0.00	0.00	0.0
14	0.13	1.29	2.83	4.25	4.66	2.76	1.65	0.00	0.00	0.00	0.00	0.0
15	0.35	1.16	2.91	3.93	5.13	3.07	2.19	0.12	0.00	0.00	0.00	0.0
16	1.91	2.39	3.69	4.36	5.19	3.83	2.50	0.71	0.00	0.00	0.00	0.0
17	3.41	3.72	4.61	4.91	5.24	4.12	2.53	1.20	0.00	0.00	0.00	0.0
18	4.38	4.22	4.49	4.79	5.19	4.53	2.72	1.60	0.00	0.00	0.00	0.0
19	2.97	2.93	3.56	4.09	5.00	3.84	2.48	0.97	0.00	0.00	0.00	0.0
20	1.29	1.28	2.43	3.29	4.83	3.43	2.34	0.56	0.00	0.00	0.00	0.0
21	0.00	0.39	2.22	3.27	4.63	2.96	2.02	0.16	0.00	0.00	0.00	0.0
22	0.00	0.83	2.51	3.67	4.25	2.61	1.50	0.00	0.00	0.00	0.00	0.0
23	0.70	1.72	3.28	3.98	3.56	2.23	0.80	0.00	0.00	0.00	0.00	0.0
24	1.27	2.27	3.98	3.90	2.82	1.58	0.22	0.00	0.00	0.00	0.00	0.0
25	2.26	3.10	4.46	3.42	2.04	0.80	0.00	0.00	0.00	0.00	0.00	0.0
26	2.56	3.36	3.81	2.51	1.22	0.22	0.00	0.00	0.00	0.00	0.00	0.0
27	2.01	2.84	2.42	1.46	0.33	0.00	0.00	0.00	0.00	0.00	0.00	0.0
28	1.34	1.98	1.31	0.60	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0

7. Deep Learning

□ CNN (Convolutional Neural Network): Extension1

- One simple convolution layer



"bright" corresponds to white in this example

filter 1

-1	-1	-1
1	1	1
0	0	0

filter 2

-1	1	0
-1	1	0
-1	1	0

filter 3

0	0	0
1	1	1
-1	-1	-1

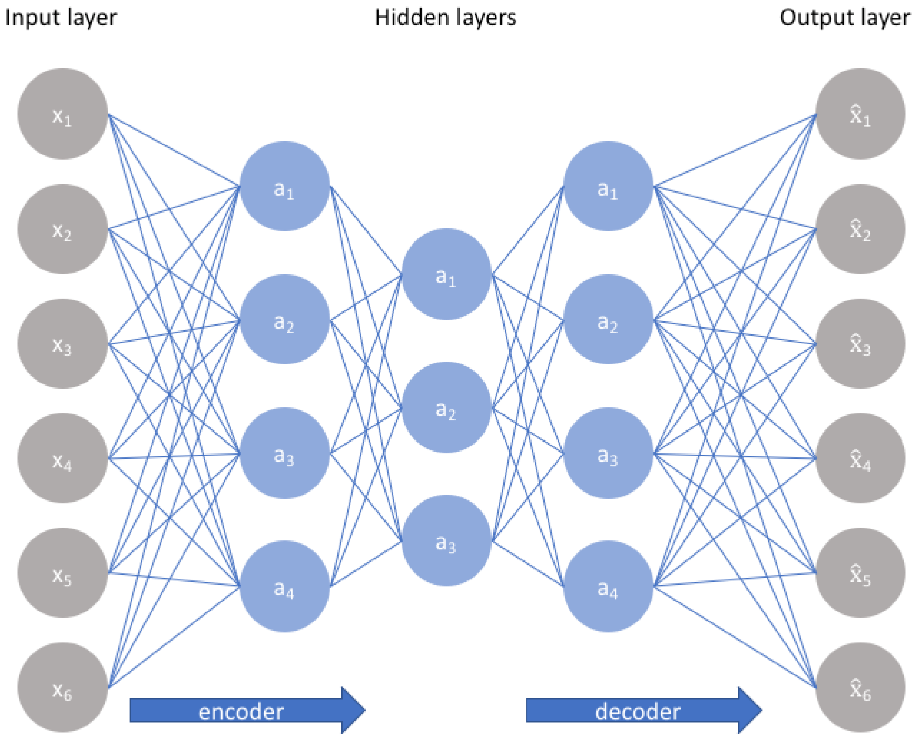
filter 4

0	1	-1
0	1	-1
0	1	-1

7. Deep Learning

□ Auto-encoder: Extension2

- Why not reverse-engineer the extracted feature and reconstruct the original input and compare it with the “real” original one?

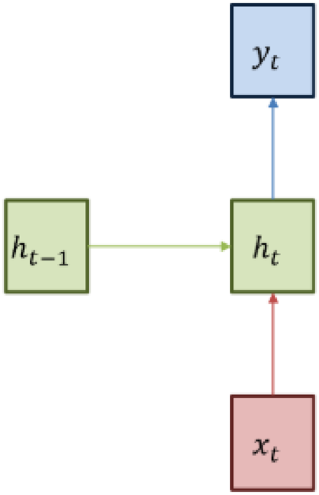


7. Deep Learning

□ RNN (Recurrent Neural Network): Extension3

- Why not incorporate historical contexts to the learning method? ex) predicting consecutive letter of word? h,e,l,l,o

- Vanishing gradient problem: cannot deal with long term memory (structure of the product, characteristics of tanh)



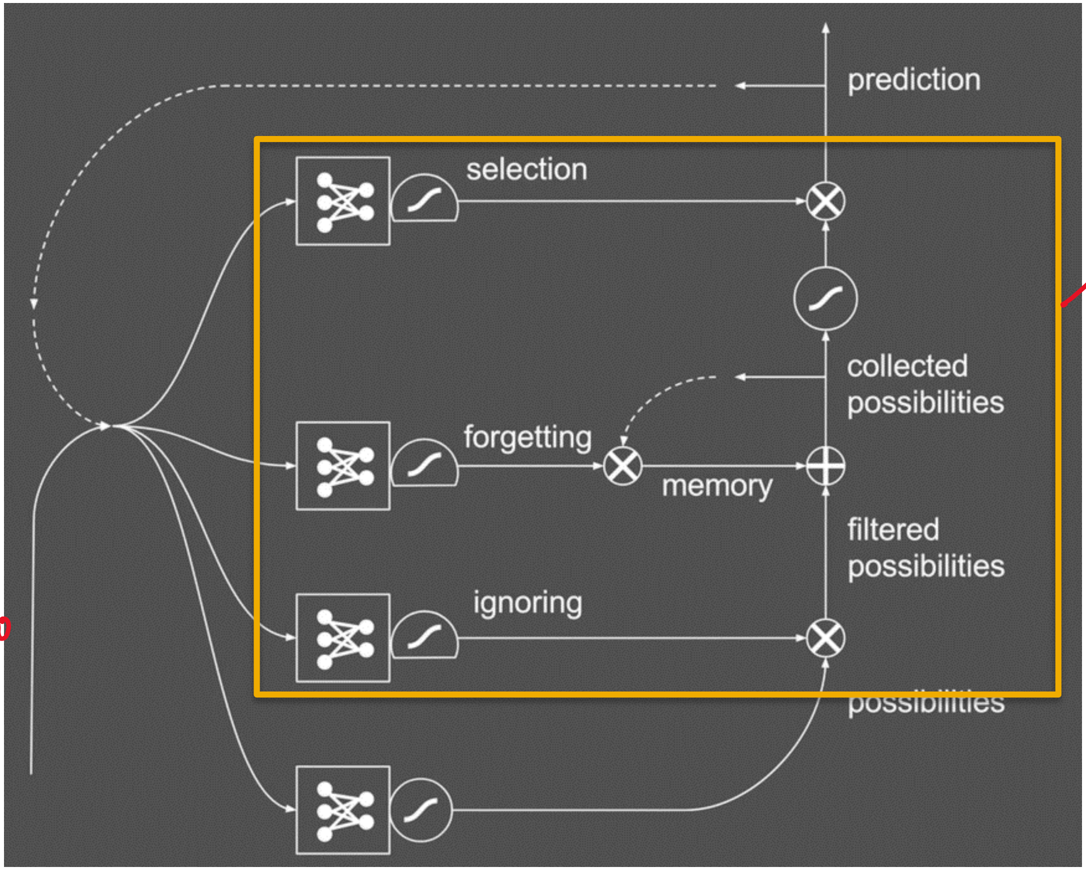
$$y_t = W_{hy} h_t + b_y$$

$$h_t = \tanh(W_{hh} h_{t-1} + W_{xh} x_t + b_h)$$

7. Deep Learning

□ LSTM (Long-Short Term Memory): Extension4

- Add special boxes to the RNN to adjust vanishing gradient problem



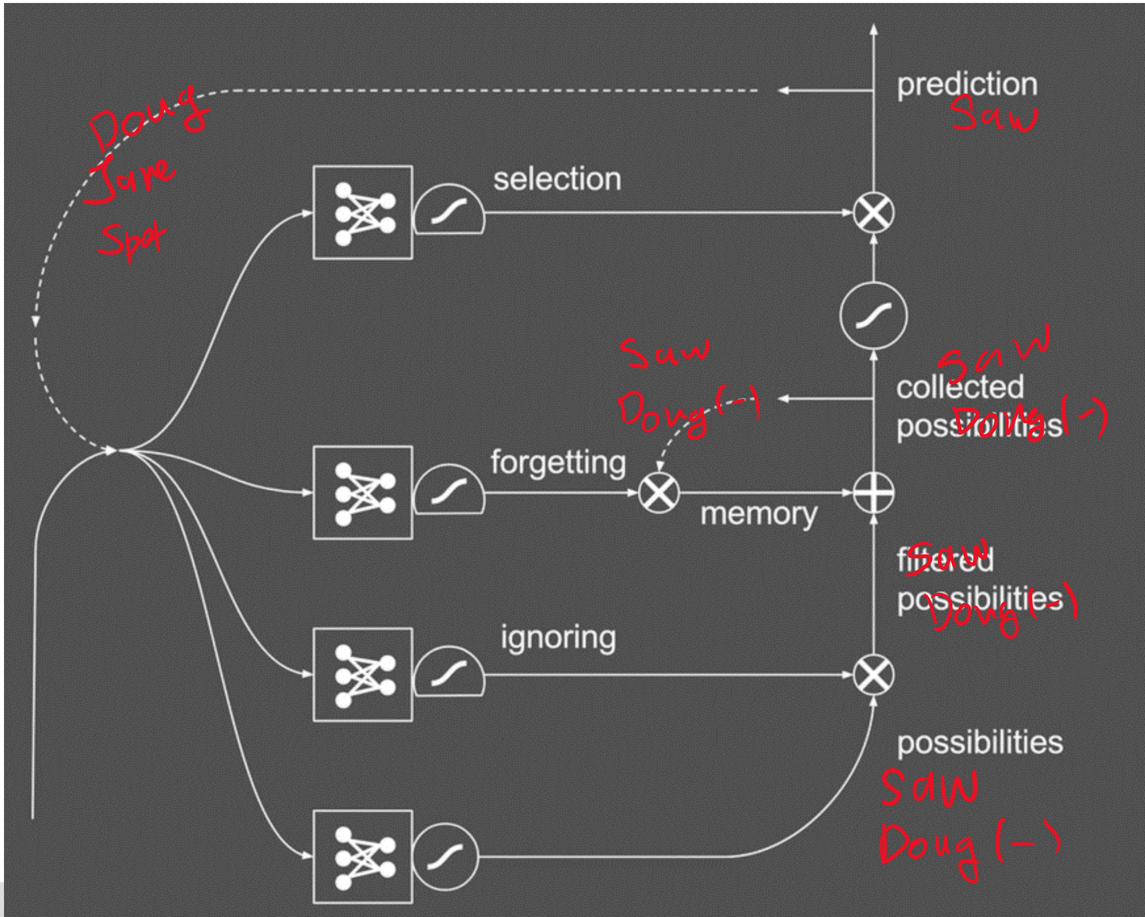
→ LSTM unit

New Info

7. Deep Learning

□ LSTM Example: Let's make a children's book

- component: Jane, Spot, Doug, saw, .
- Goal:
  Jane saw Spot.
  Spot saw Doug.
  Doug saw Spot. …
- Possible mistakes:
  Jane saw Jane.
  Doug saw Spot saw …
  Doug.
- Current stage:
  Jane saw Spot.
  Doug …



7. Deep Learning

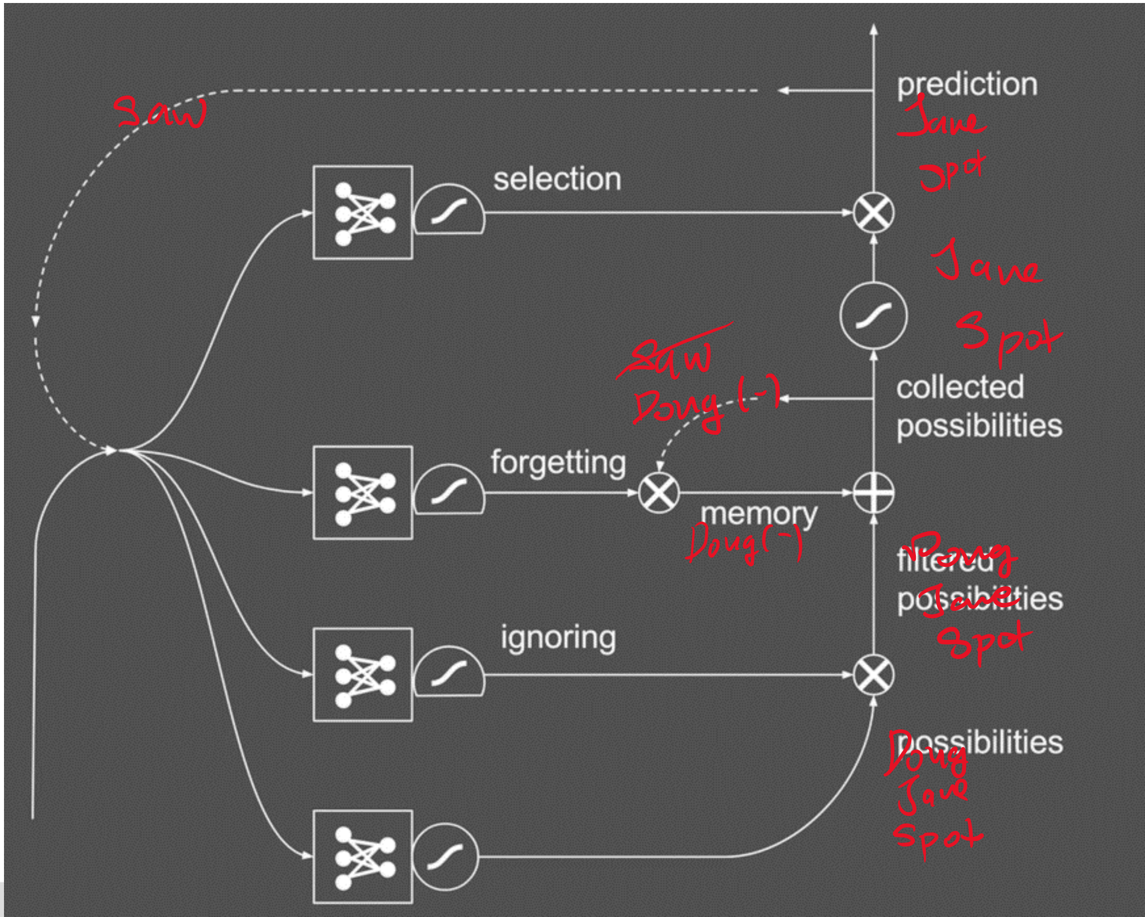
□ LSTM Example: Let's make a children's book

- component: Jane, Spot, Doug, saw, .

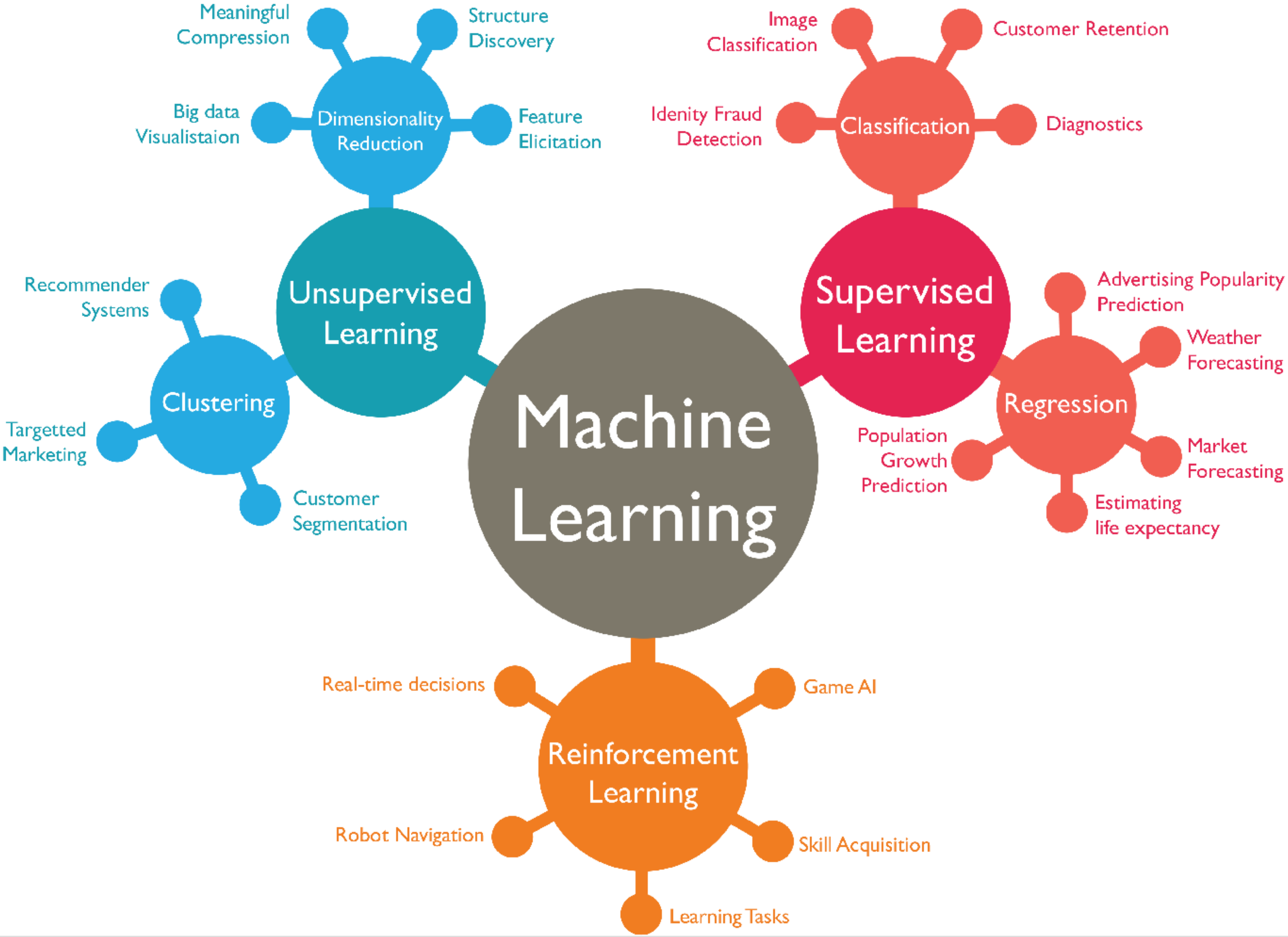
- Goal:
  Jane saw Spot.
  Spot saw Doug.
  Doug saw Spot. …

- Possible mistakes:
  Jane saw Jane.
  Doug saw Spot saw …
  Doug.

- Current stage:
  Jane saw Spot.
  Doug saw …

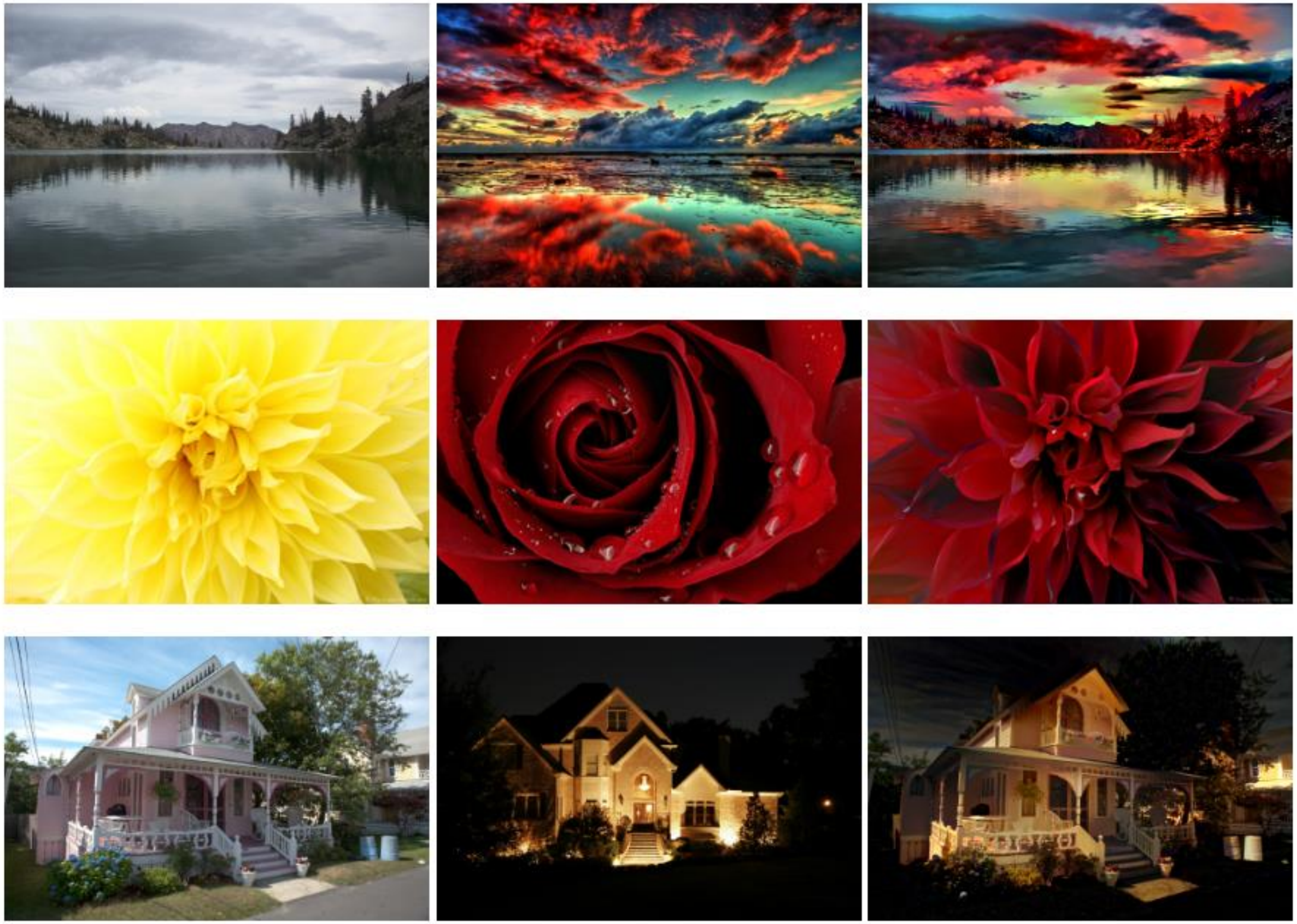


8. Application: General



8. Application

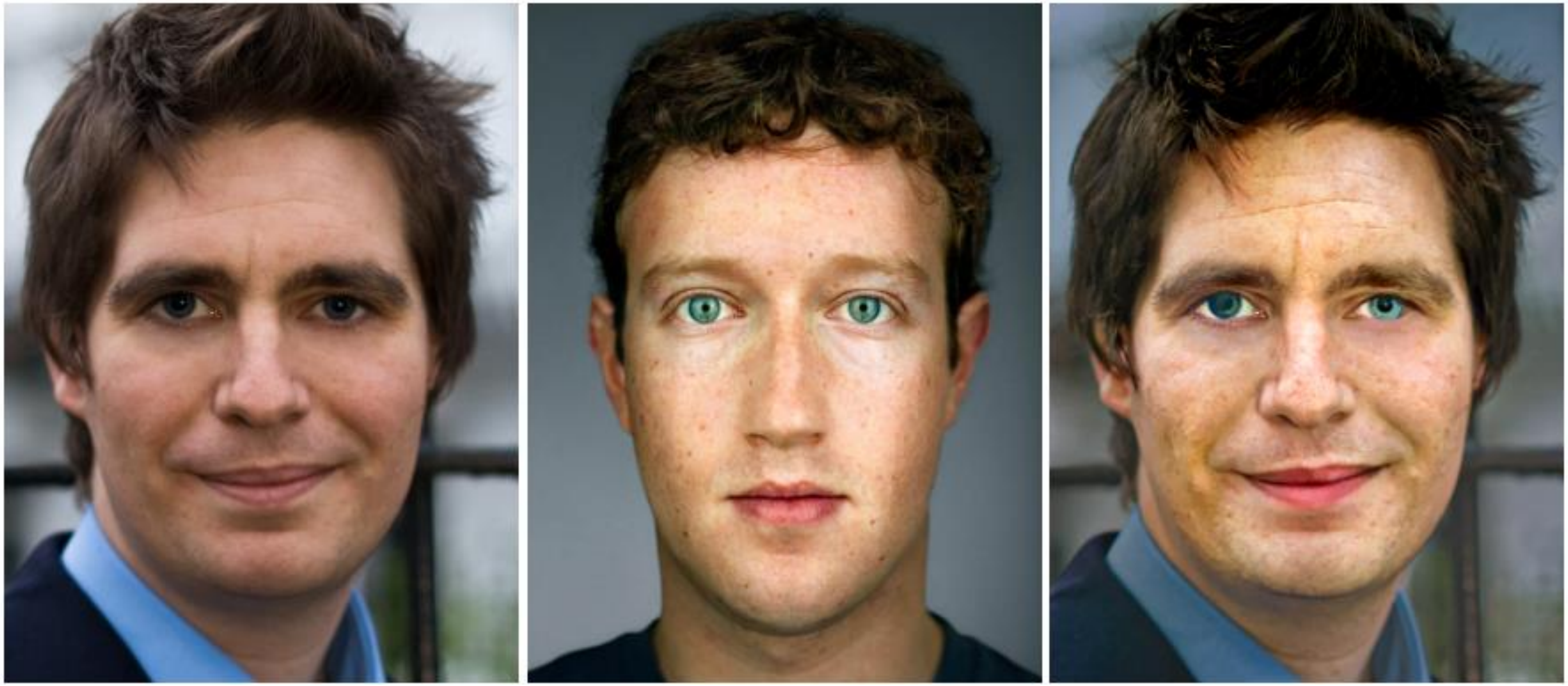
□ deep-photo-styletransfer: input, style, output



8. Application

□ deep-photo-styletransfer: input, style, output

- how to map style to the original contents?
- extension of CNN



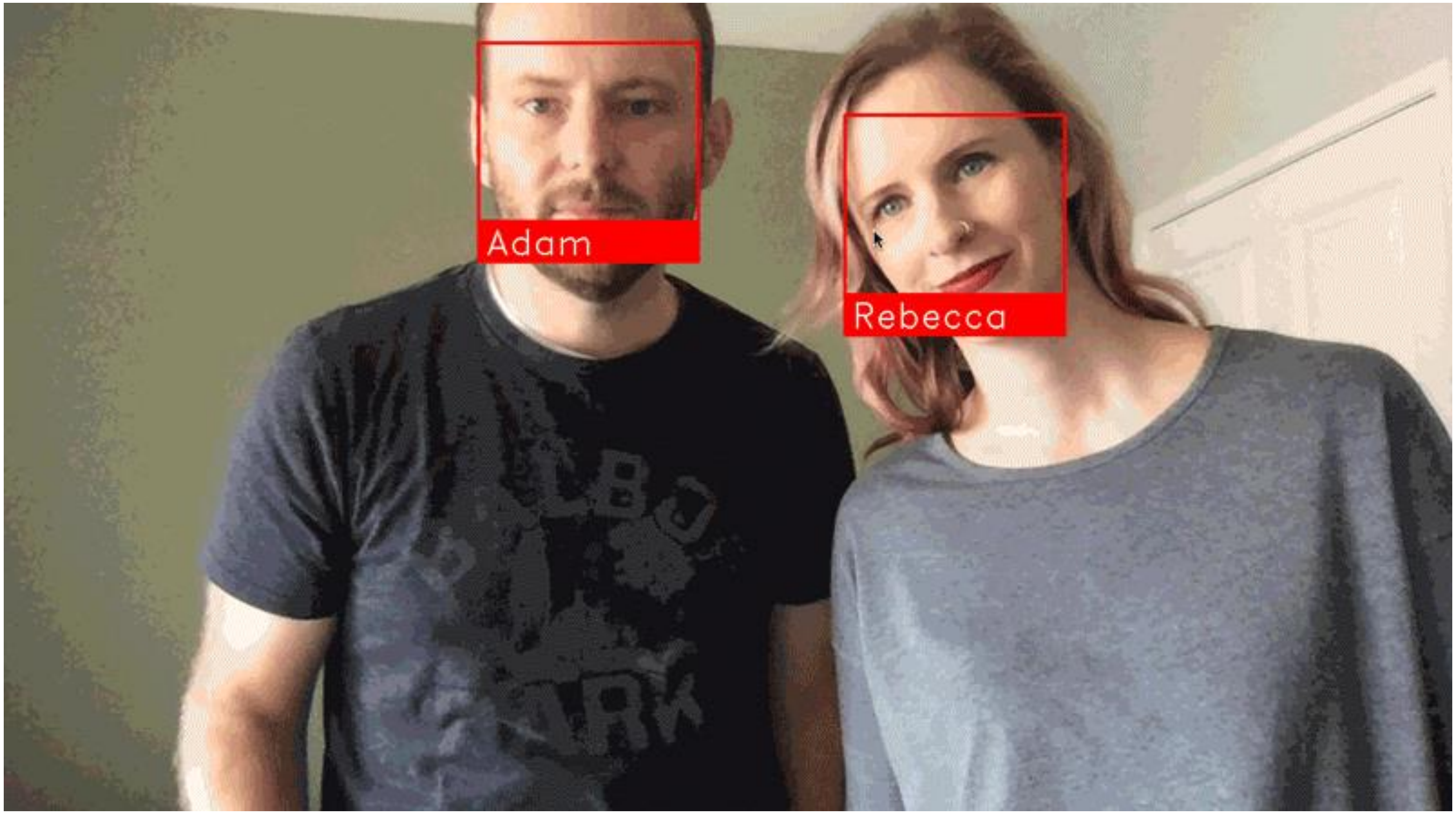
8. Application

□ deep-photo-styletransfer extension

Freiburger Münsterplatz

8. Application

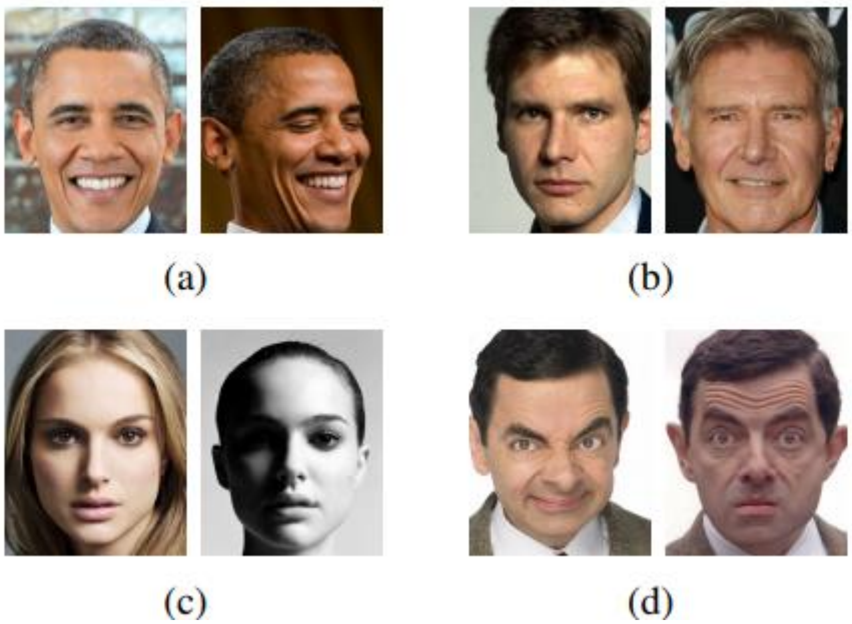
□ Face recognition



8. Application

□ Face recognition

- Overcome this!



8. Application

□ Face recognition

- Face detection: find the position of the faces
- Face alignment: scale and crop face images to compare with references
- Face representation: pixel values of face images->feature vector
- Face matching: compare the similarity score
- This time again, CNN

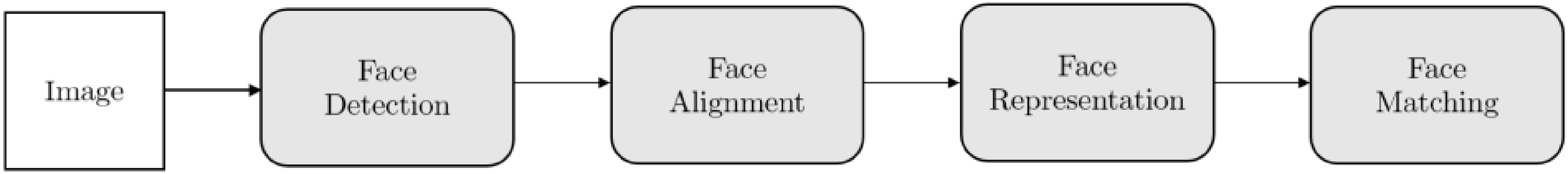


Fig. 2: Face recognition building blocks.

8. Application

□ Magenta: Art and Music composer

- RNN

8. Application

□ Google: Neutral Networks and ‘Machine that Dream’

Selection of images created by Google’s neural network

- These days, it’s probably easier to list areas that Google isn’t working on with machine learning.

8. Application

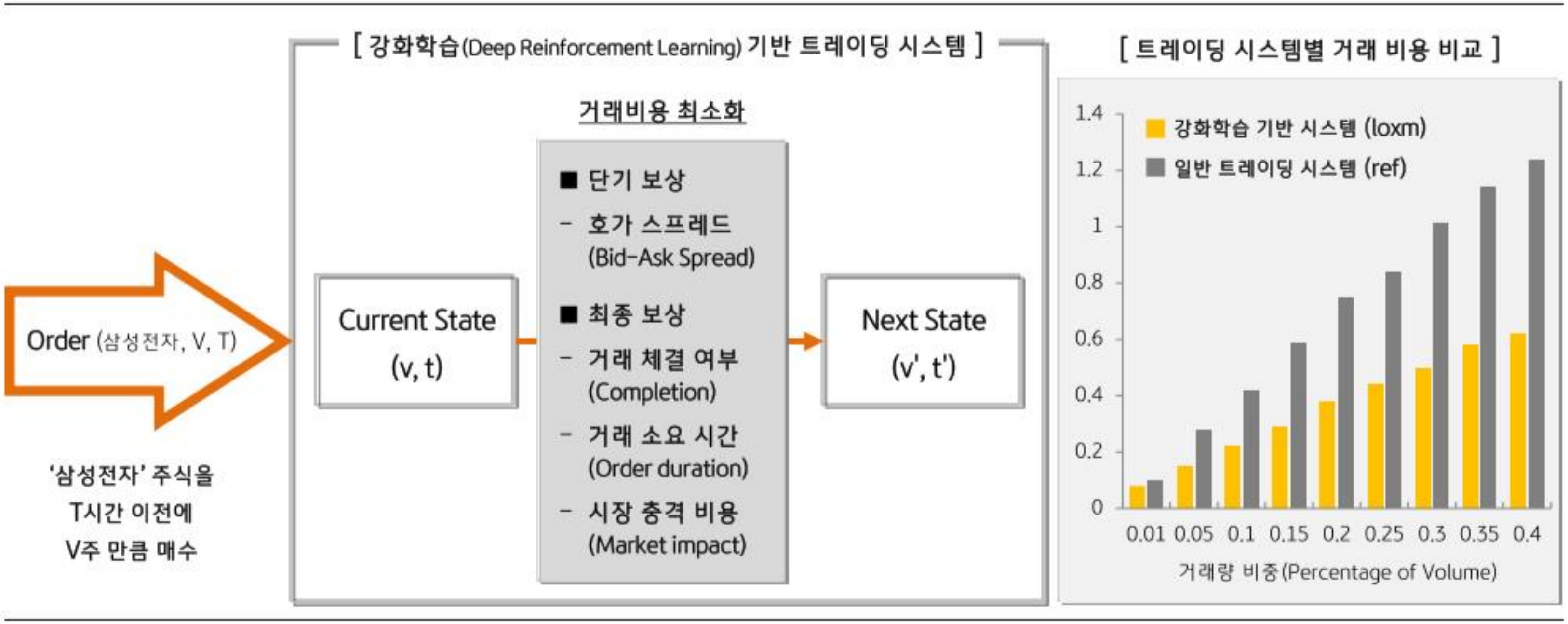
□ Deep Voice Conversion: Neural Voice Cloning

8. Application

□ LOXM: JP Morgan Trading system

- minimize the cost function with the inputs as bid-ask spread, the success of transactions, duration, market impact costs

- great performance compared to the benchmark system


[강화학습(Deep Reinforcement Learning) 기반 트레이딩 시스템]
Order (삼성전자, V, T)
'삼성전자' 주식을 T시간 이전에 V주 만큼 매수
거래비용 최소화
Current State (v, t)
■ 단기 보상
- 호가 스프레드 (Bid-Ask Spread)
■ 최종 보상
- 거래 체결 여부 (Completion)
- 거래 소요 시간 (Order duration)
- 시장 충격 비용 (Market impact)
Next State (v', t')
[트레이딩 시스템별 거래 비용 비교]
강화학습 기반 시스템 (loxm)
일반 트레이딩 시스템 (ref)
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0.01
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
거래량 비중(Percentage of Volume)

자료: 'Active Learning in Trading Algorithms(JPMorgan, 2016)' 및 KB경영연구소

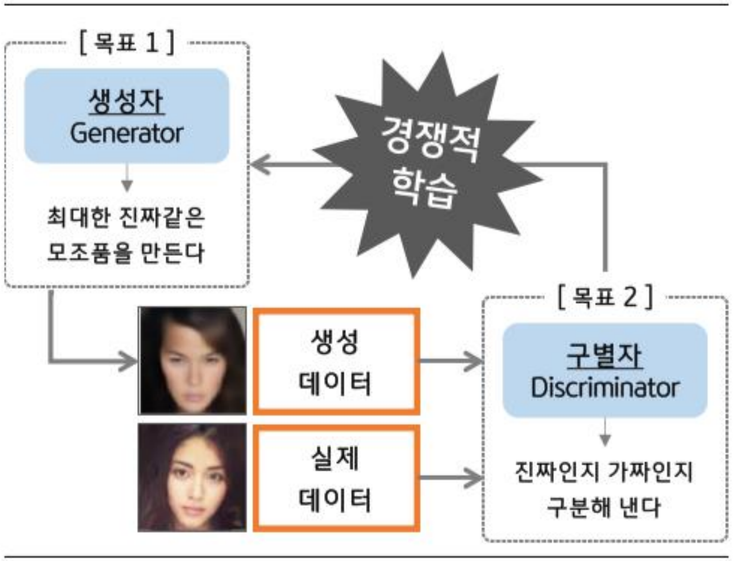
8. Application

□ Fraud Detection

- Making a competition between two model: Generator and Discriminator

- Generator generates fraud-like data
- Discriminator discriminates real one with the generated one

- Good choice when not enough data



자료: Tech M 및 KB경영연구소

8. Application

□ Credit Rating System

- need to know “why”

- make a small perturbation to the specific variable so that figure out which factor has the most influence for categorization

- FICO: Credit rating company “Reasons Reporter”

- Abnormality detection and Credit Rating

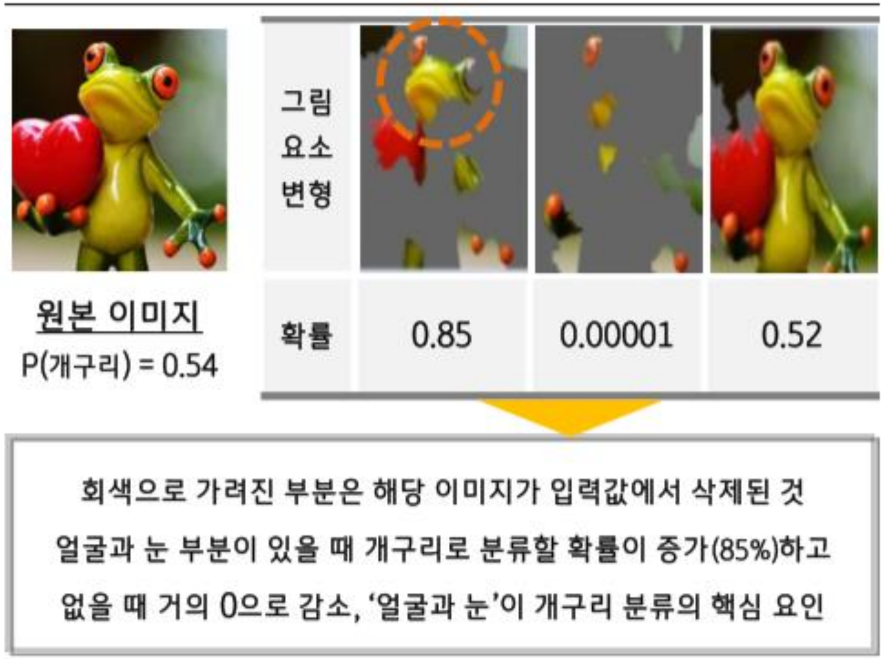


그림 요소 변형			
확률	0.85	0.00001	0.52

회색으로 가려진 부분은 해당 이미지가 입력값에서 삭제된 것
얼굴과 눈 부분이 있을 때 개구리로 분류할 확률이 증가(85%)하고
없을 때 거의 0으로 감소, ‘얼굴과 눈’이 개구리 분류의 핵심 요인

자료: Marco Tulio Ribeiro et al.(O’REILLY, 2016)

00. Epilogue

THANK YOU

FOR LISTENING