

# Intro

## Machine Learning

Microsoft Student Partner

백진헌

- 1. Concept of Machine Learning**
- 2. Supervised Learning**
- 3. Unsupervised Learning**

## Concept of Machine Learning

### ✓ Machine Learning?

**“ Field of study that gives computers the ability to learn without being explicitly programmed.” , Arthur Samuel (1959)**

## Concept of Machine Learning

### ✓ Machine Learning?

**“ 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.” , Tom Mitchell (1997)**

## Concept of Machine Learning

✓ **Machine Learning?**

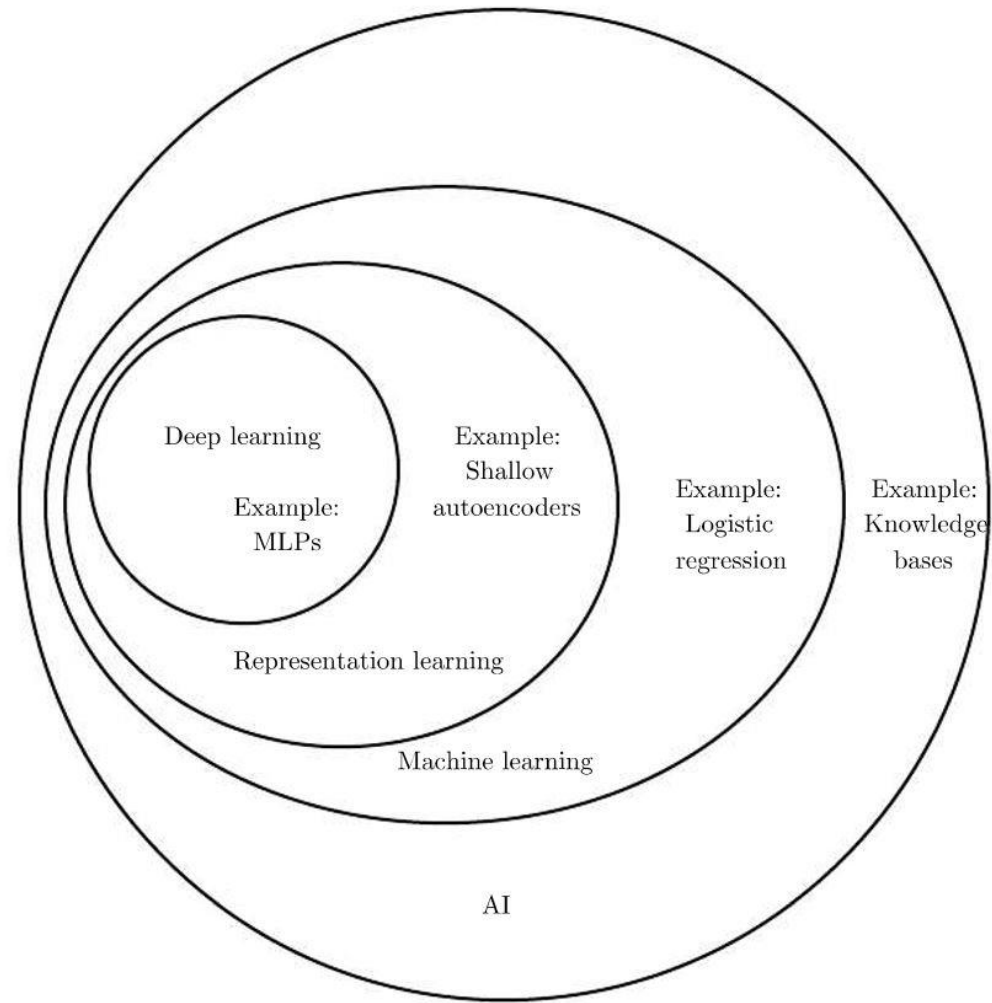
**Ex) Spam mail filter,**

**E(Experience): Watching you label emails as spam or not spam.**

**T(Task): Classifying emails as spam or not spam.**

**P(Performance): The number (or fraction) of emails correctly classified as spam / not spam.**

# Concept of Machine Learning



# Concept of Machine Learning

✓ AI

**“ How to get this informal knowledge into a computer !”**

## Concept of Machine Learning

✓ AI

**“ How to get this informal knowledge into a computer !”**

**Knowledge base approach**



## Concept of Machine Learning

✓ AI

**None of these projects has led to a major success.**

# Concept of Machine Learning

## ✓ Machine Learning

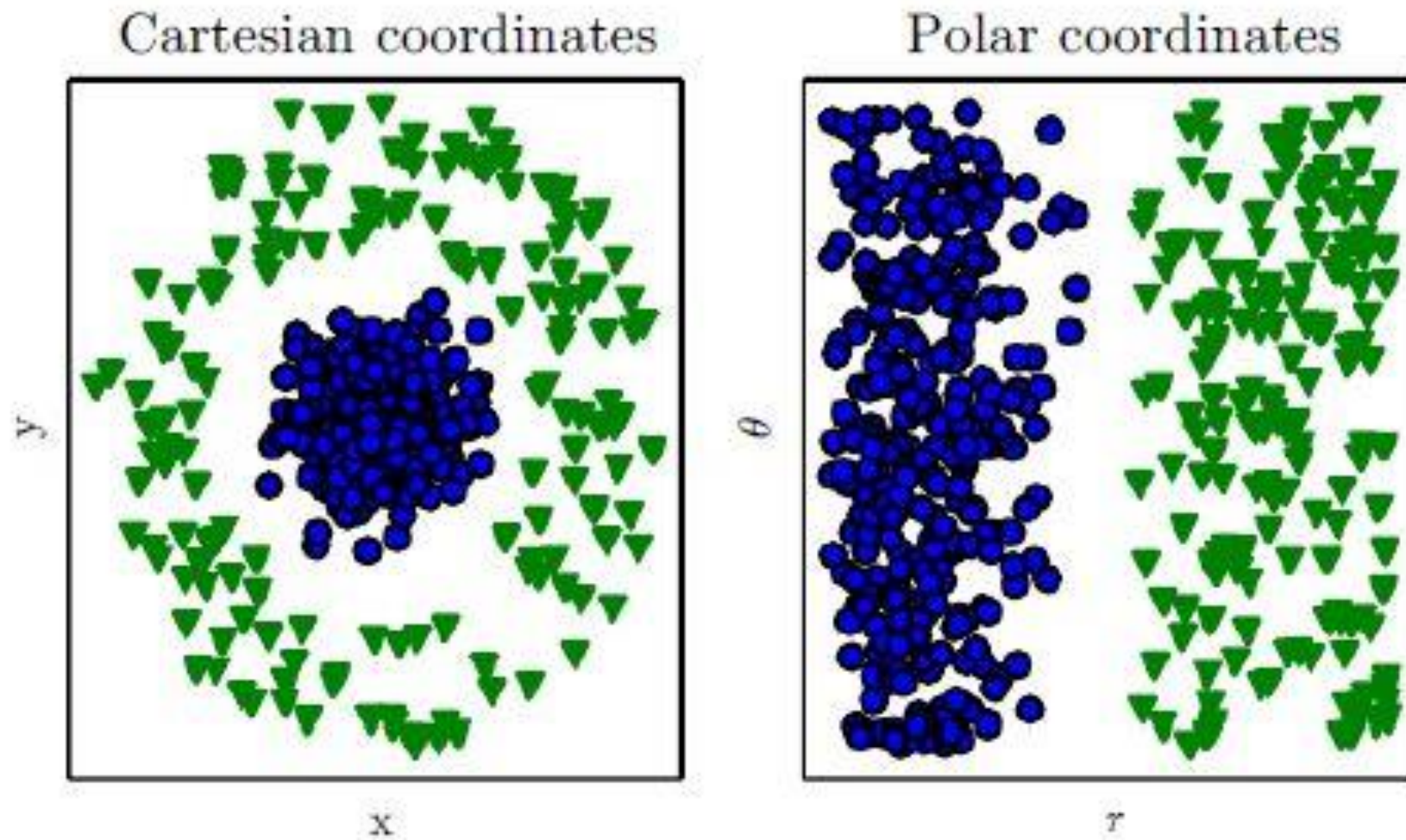
**“Extract patterns from raw data.”**

## Concept of Machine Learning

### ✓ Machine Learning

“Difficult to know what **features** should be extracted.”

# Concept of Machine Learning



## Concept of Machine Learning

### ✓ Representation Learning

**“Discover not only the mapping from representation to output  
but also the representation itself.”**

## Concept of Machine Learning

### ✓ Representation Learning - Autoencoder

**Autoencoder: encoder + decoder**

**Encoder:** input data into a different representation

**Decode:** new representation back into the original format

## Concept of Machine Learning

### ✓ Representation Learning

**“What if difficult to obtain a representation”**

## Concept of Machine Learning

### ✓ Deep Learning

**“Build complex concepts out of simpler concepts(representation)”**

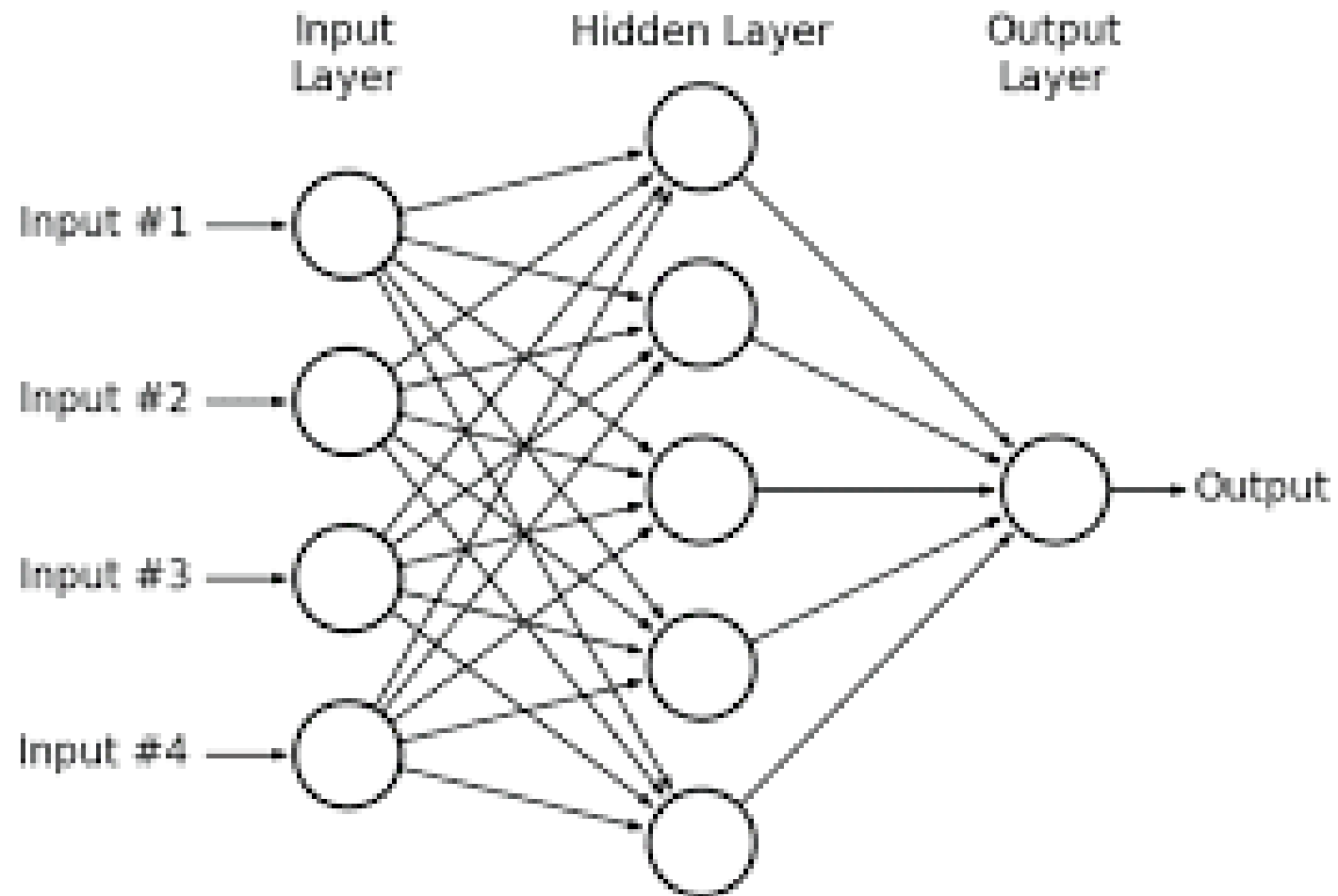


## Concept of Machine Learning

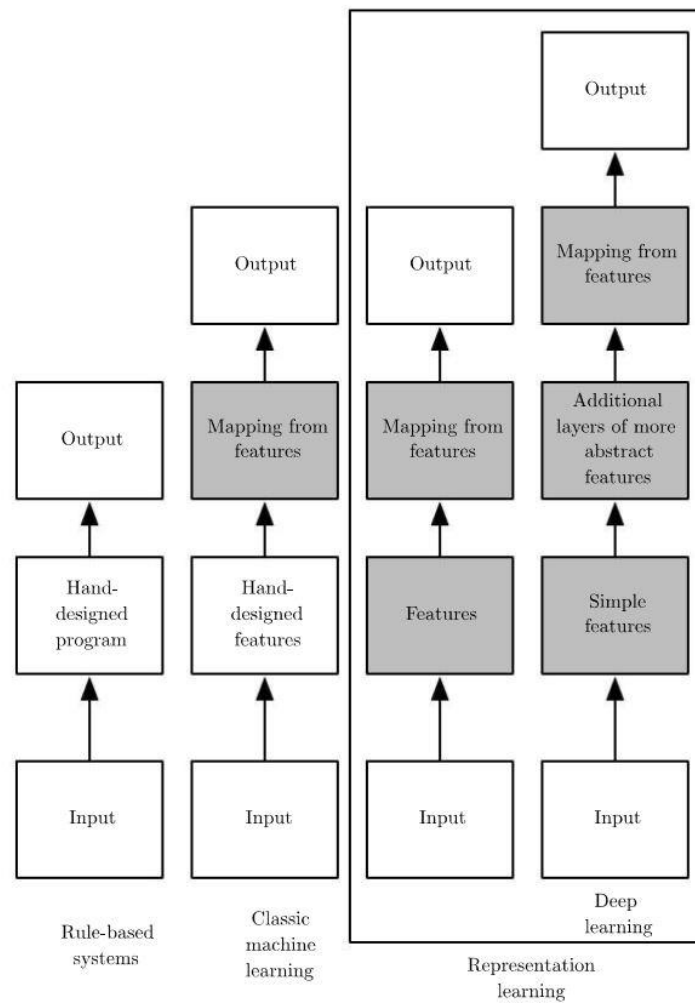
✓ **Deep Learning** - **MLP** (multilayer perceptron)

**“Each function provides a new representation”**

# Concept of Machine Learning



# Concept of Machine Learning



## ✓ Supervised Learning

: Experience a dataset containing features, but each example is also associated with a label or target.

## ✓ Unsupervised Learning

: Experience a dataset containing many features, then learn useful properties of the structure of this dataset.

# **Supervised Learning**

**✓ Supervised Learning**

**: Regression (Continuous Value)**

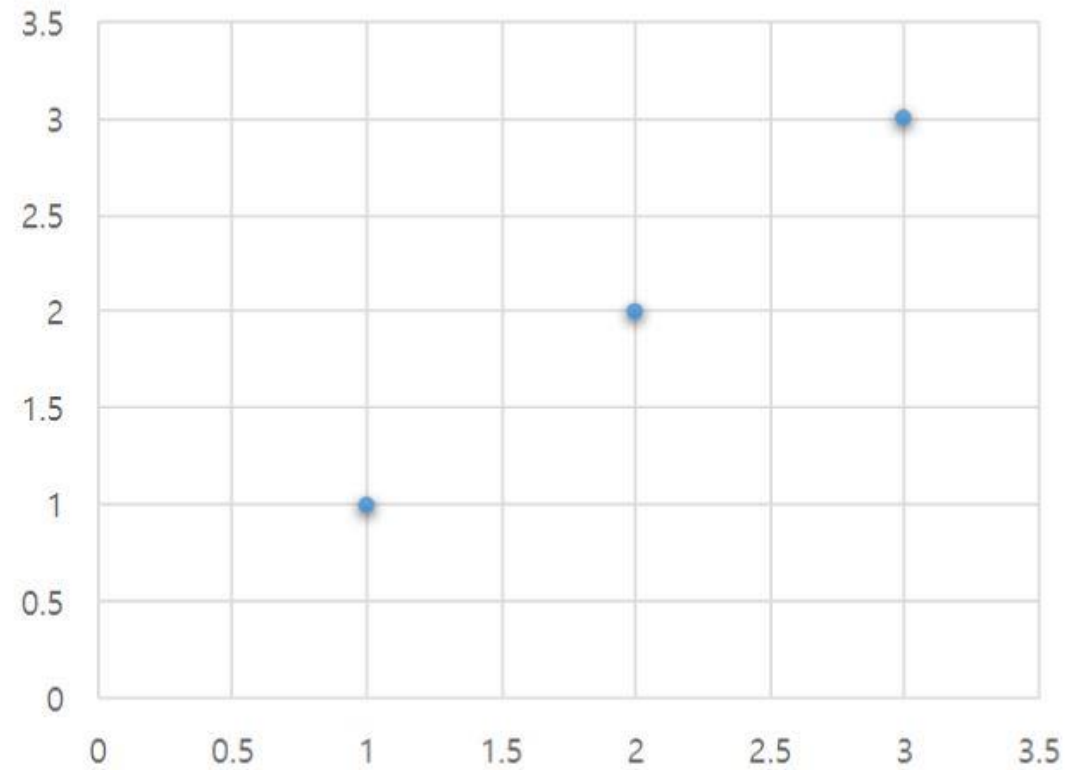
**: Classification (Discrete Value)**

**In Deep Learning? CNN / RNN**

# Supervised Learning

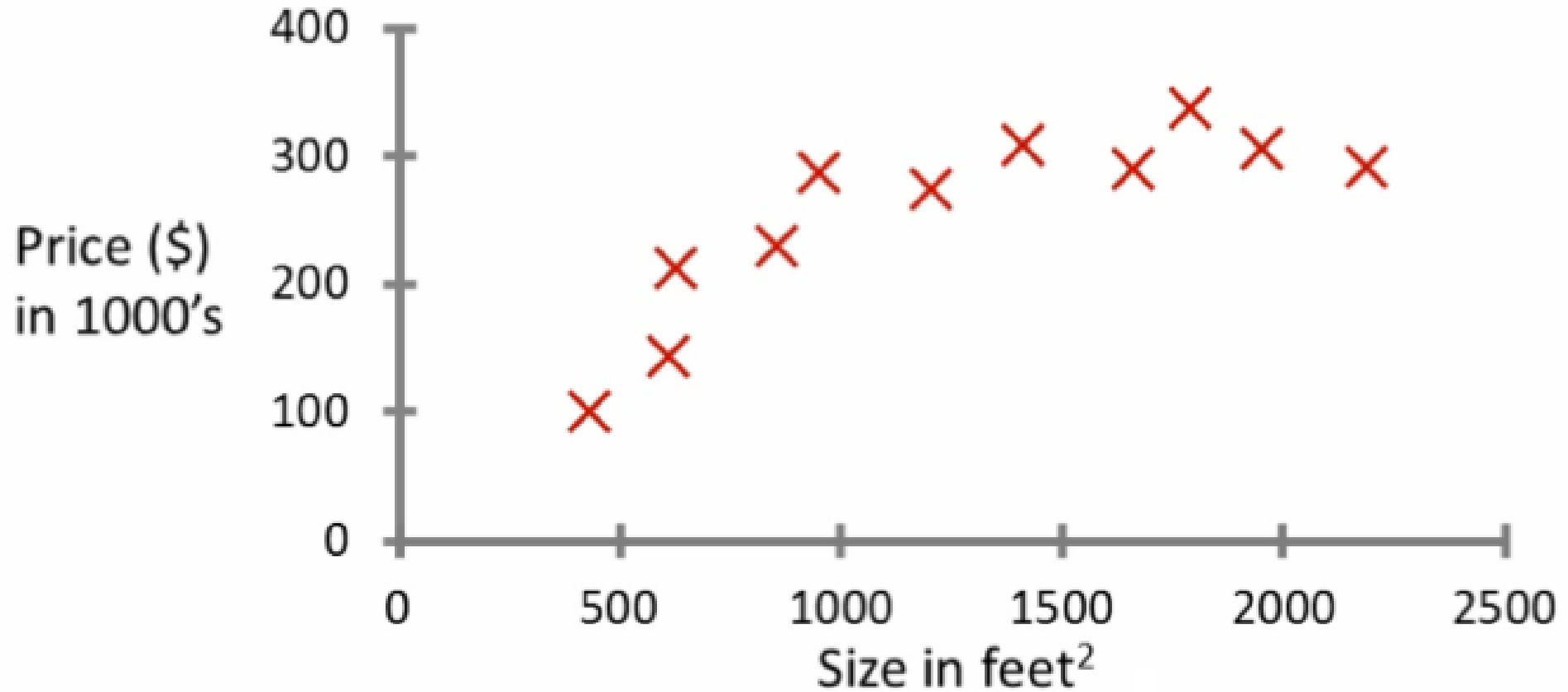
## ✓ Supervised Learning - Regression

x	y
1	1
2	2
3	3



## Supervised Learning

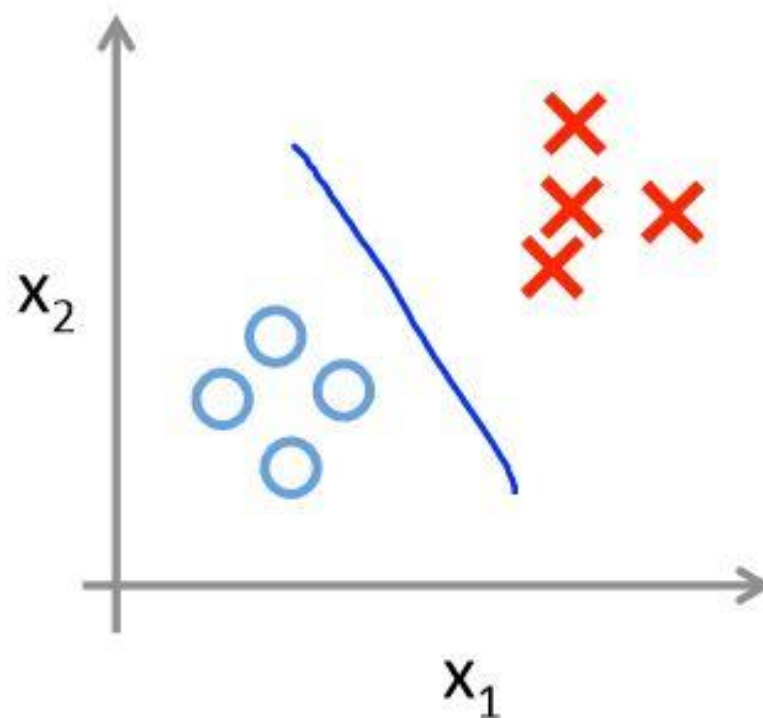
### Housing price prediction.



# Supervised Learning

## ✓ Supervised Learning - Classification

Binary classification:





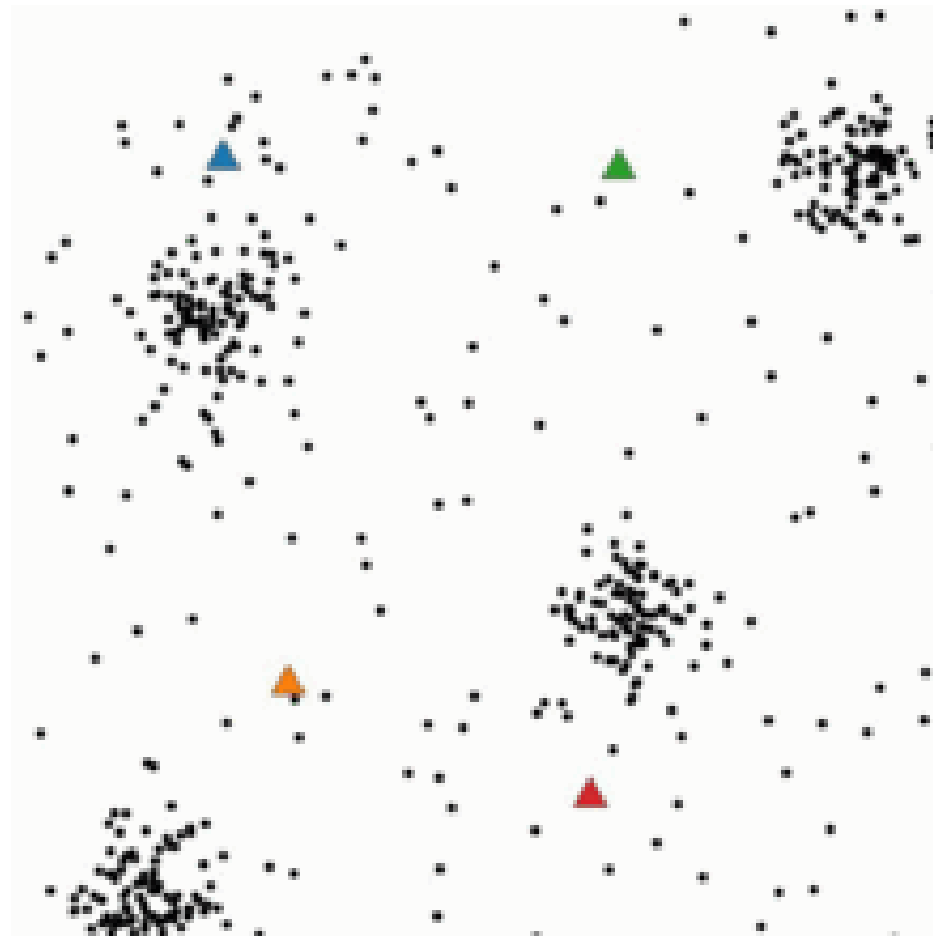
# Unsupervised Learning

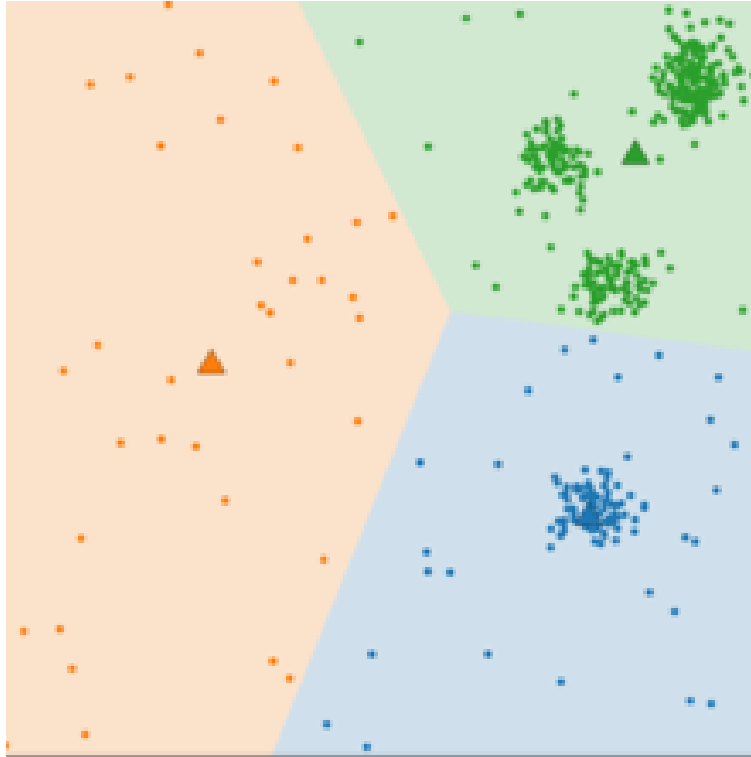
✓ Unsupervised Learning

: Clustering

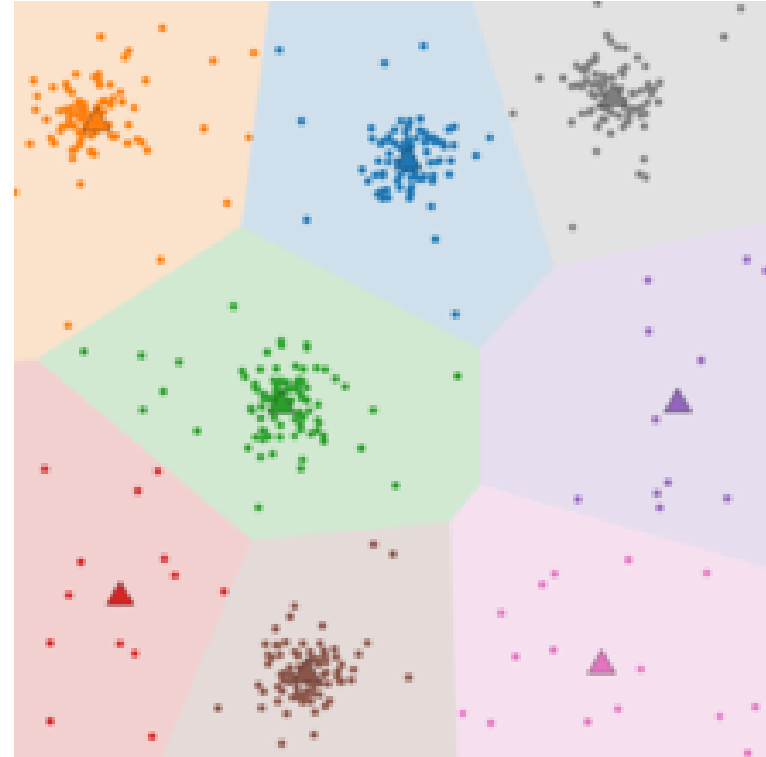
In Deep Learning? AutoEncoders

## ✓ Unsupervised Learning - Clustering



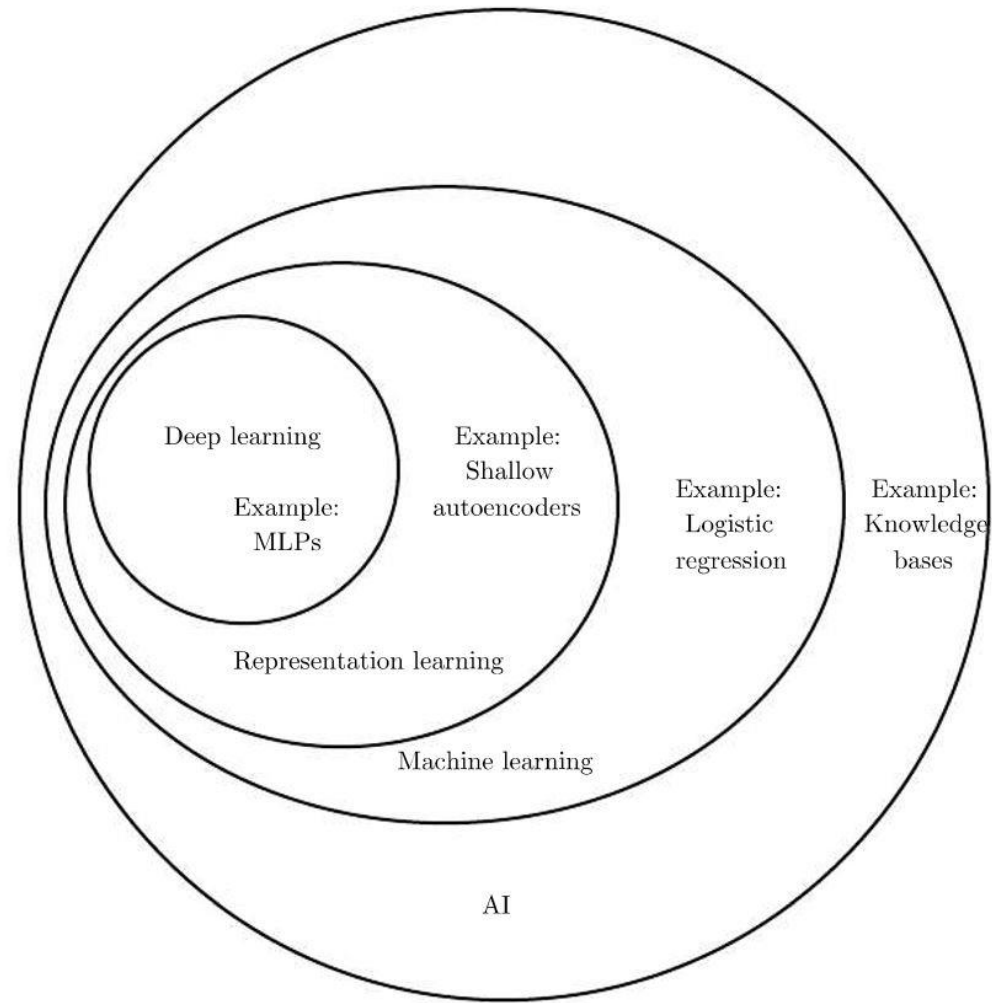


$K = 3$



$K = 8$

# Warm Up



## Warm Up

✓ **Supervised Learning**

✓ **Unsupervised Learning**

# Supervised Learning & HOL

Machine Learning

Microsoft Student Partner

백진헌

## 1. Linear Regression

(HOL with Azure Machine Learning)

## 2. Neural Network

(HOL with Azure Notebook)

# Supervised Learning

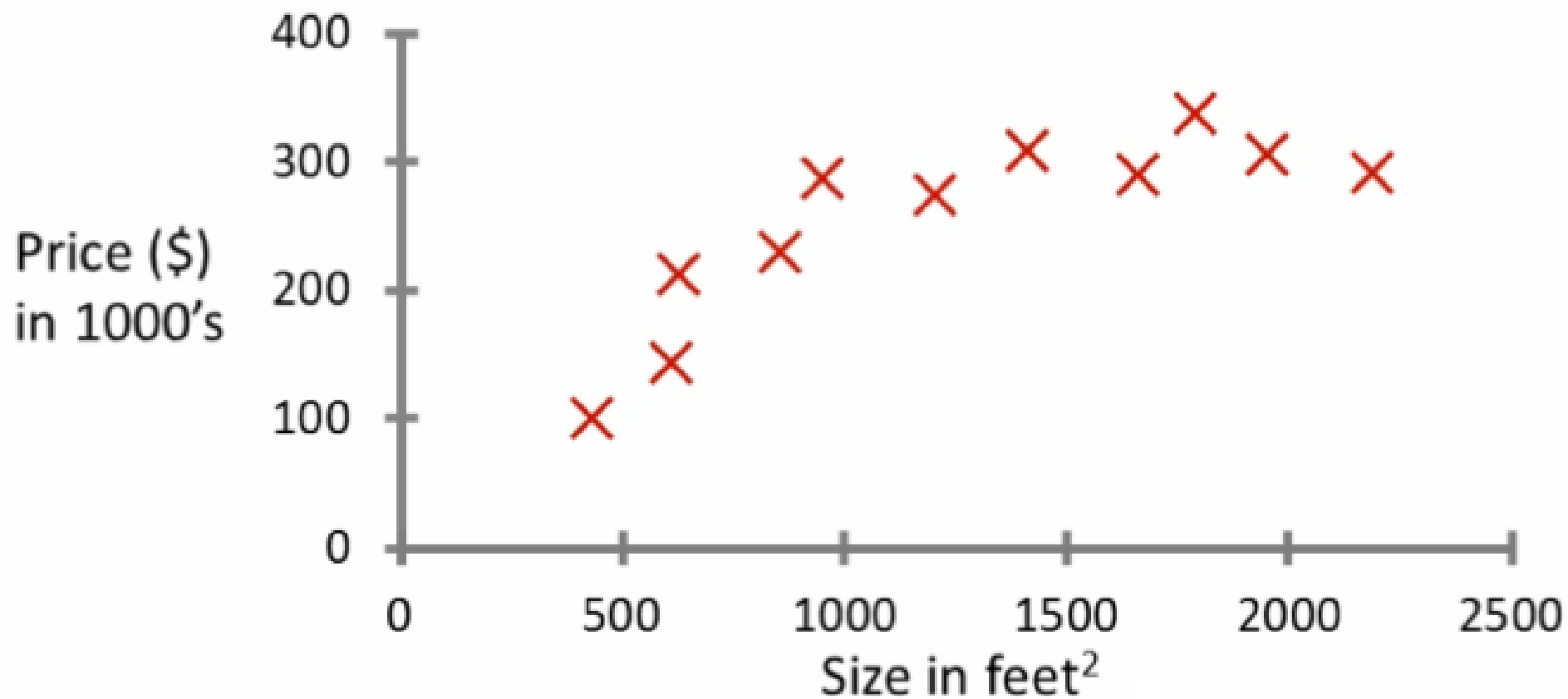
## [Machine Learning]

✓ Regression  
: Linear Regression



## Supervised Learning

### Housing price prediction.



# Supervised Learning

$$Y = W * X + b$$

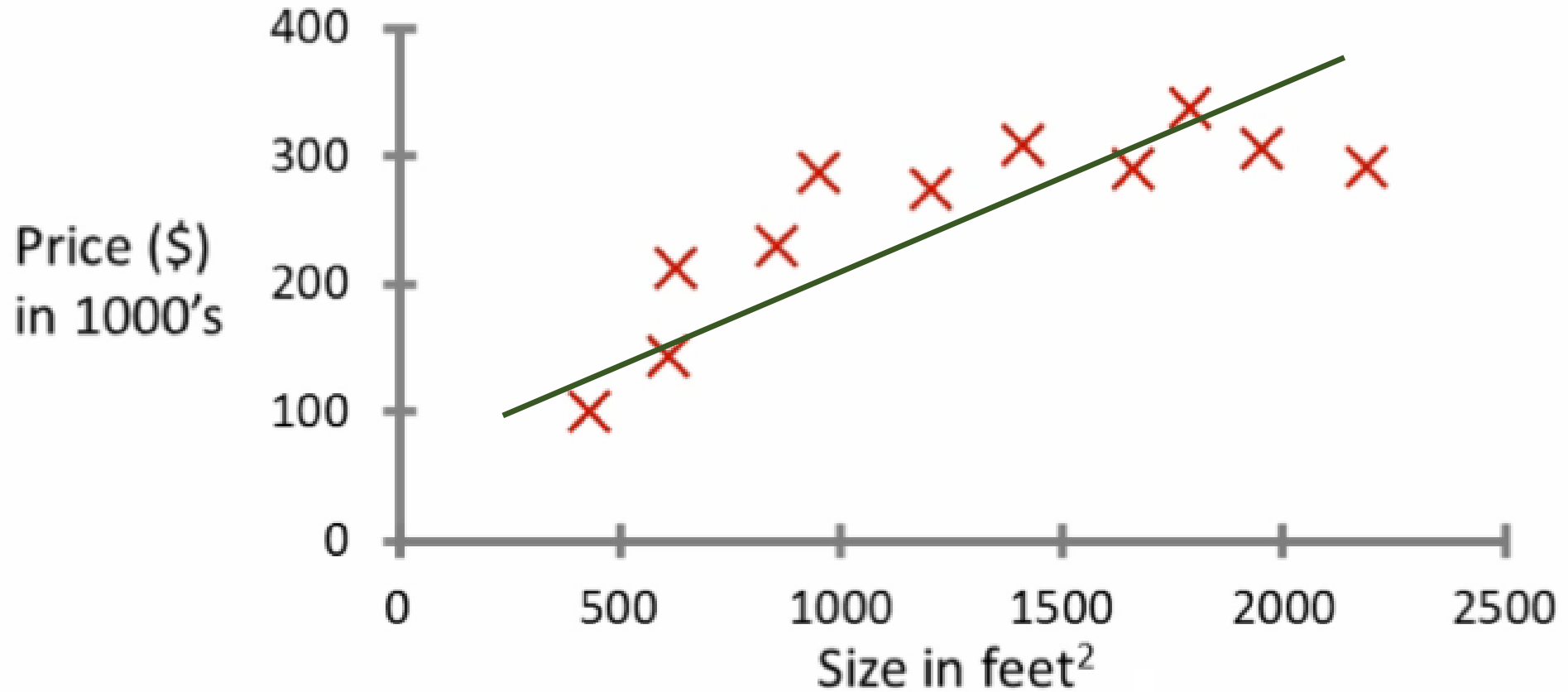
**X** : Y를 예측할 때 필요한 유의미한 피처 (집의 크기)

**W, b** : Y를 예측할 때 사용되는 각 피처에 대한 무게 값

**Y** : 학습된 모델을 통해 예측한 값

# Supervised Learning

## Housing price prediction.



$$Y = (W1*X1) + (W2*X2) + \dots + (Wn*Xn) + b$$

**X** : Y를 예측할 때 필요한 유의미한 피처 (집의 크기, 집의 층 수 등)

**W, b** : Y를 예측할 때 사용되는 각 피처에 대한 무게 값

**Y** : 학습된 모델을 통해 예측한 값

$$Y = (W1*X1) + (W2*X2) + \dots + (Wn*Xn) + b$$

그럼 우리가 학습을 통해 계산해야 하는 것은?

$$Y = (W_1 * X_1) + (W_2 * X_2) + \dots + (W_n * X_n) + b$$

계산해야 하는  $W_{(\text{weight})}$  는 어떻게 계산할까 ?

# Supervised Learning

$$Y_{\text{(predict)}} = (W1 * X1) + (W2 * X2) + \dots + (Wn * Xn) + b$$

$Y_{\text{predict}}$  : 학습된 모델을 통해 예측한 값

$Y$  : 실제 값 (Label)

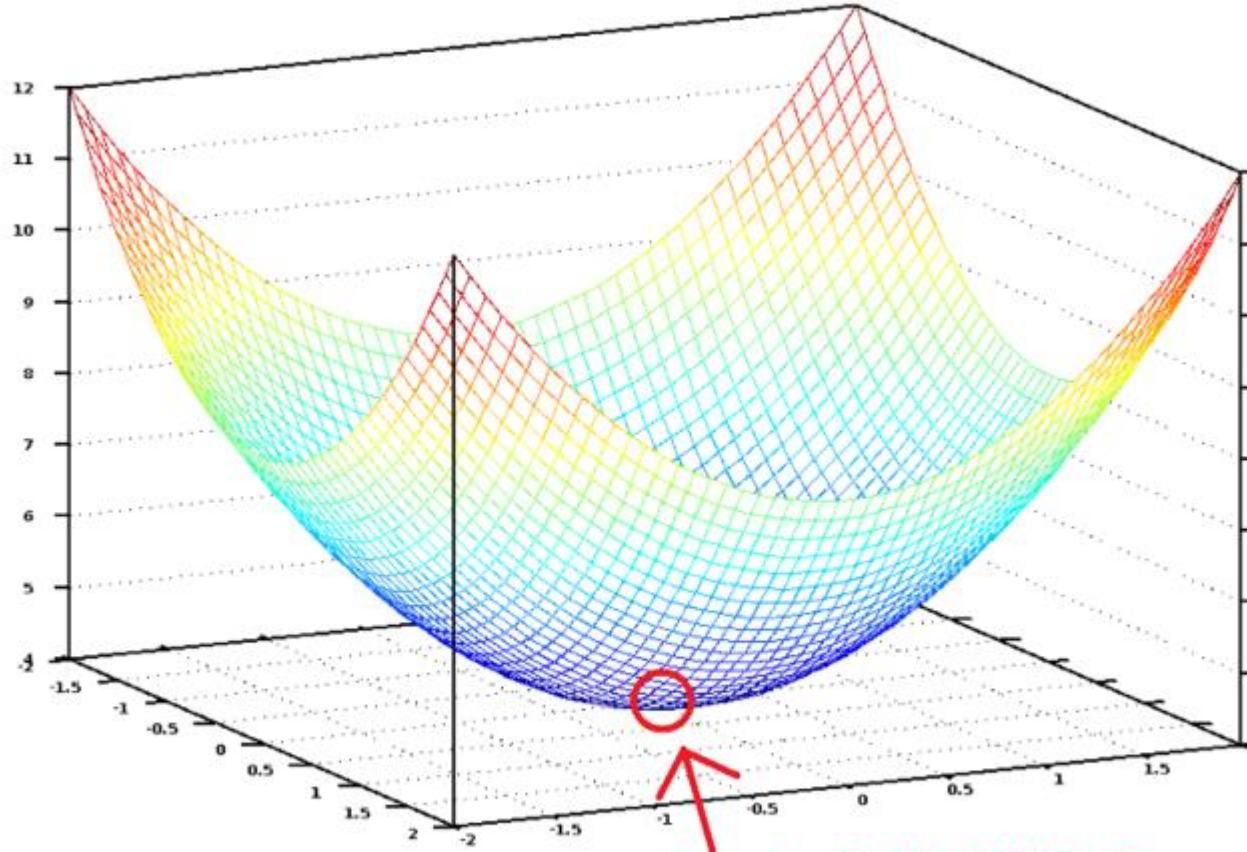
$$Y_{(\text{predict})} = (W1*X1) + (W2*X2) + \dots + (Wn*Xn) + b$$

$$\text{Cost Function} : (Y - Y_{(\text{predict})})^2$$



# Supervised Learning

## ✓ Graph of Cost Function



*cost function이 최소화되는 곳  
= 가장 효율적인 회귀식*

## Supervised Learning

✓ **Gradient, Gradient Descent**

# Supervised Learning

## ✓ Derivative

$$f'(x) > 0$$

: positive direction -> increase

$$f'(x) < 0$$

: negative direction -> increase

# Supervised Learning

## ✓ Partial Derivative

Suppose  $f: \mathbb{R}^n \rightarrow \mathbb{R}$

The *partial* derivative of  $f(x_1, \dots, x_n)$  at  $x_i$

$$\frac{\partial}{\partial x_i} f(x_1, \dots, x_n) = \lim_{h \rightarrow 0} \frac{f(x_1, \dots, x_i + h, \dots, x_n) - f(x_1, \dots, x_i, \dots, x_n)}{h}$$

# Supervised Learning

## ✓ Partial Derivative

Suppose  $f: \mathbb{R}^n \rightarrow \mathbb{R}$

The *partial* derivative of  $f(x_1, \dots, x_n)$  at  $x_i$

$$\frac{\partial}{\partial x_i} f(x_1, \dots, x_n) = \lim_{h \rightarrow 0} \frac{f(x_1, \dots, x_i + h, \dots, x_n) - f(x_1, \dots, x_i, \dots, x_n)}{h}$$

$\frac{\partial f}{\partial x_i}$  방향으로 가면,  $x_i$  축에서 증가하는 방향으로 향한다.

## ✓ Gradient

Suppose  $f: \mathbb{R}^n \rightarrow \mathbb{R}$

Gradient  $\nabla f(x)$

$$\nabla f = \left[ \frac{\partial f}{\partial x_1} \quad \cdots \quad \frac{\partial f}{\partial x_n} \right]$$

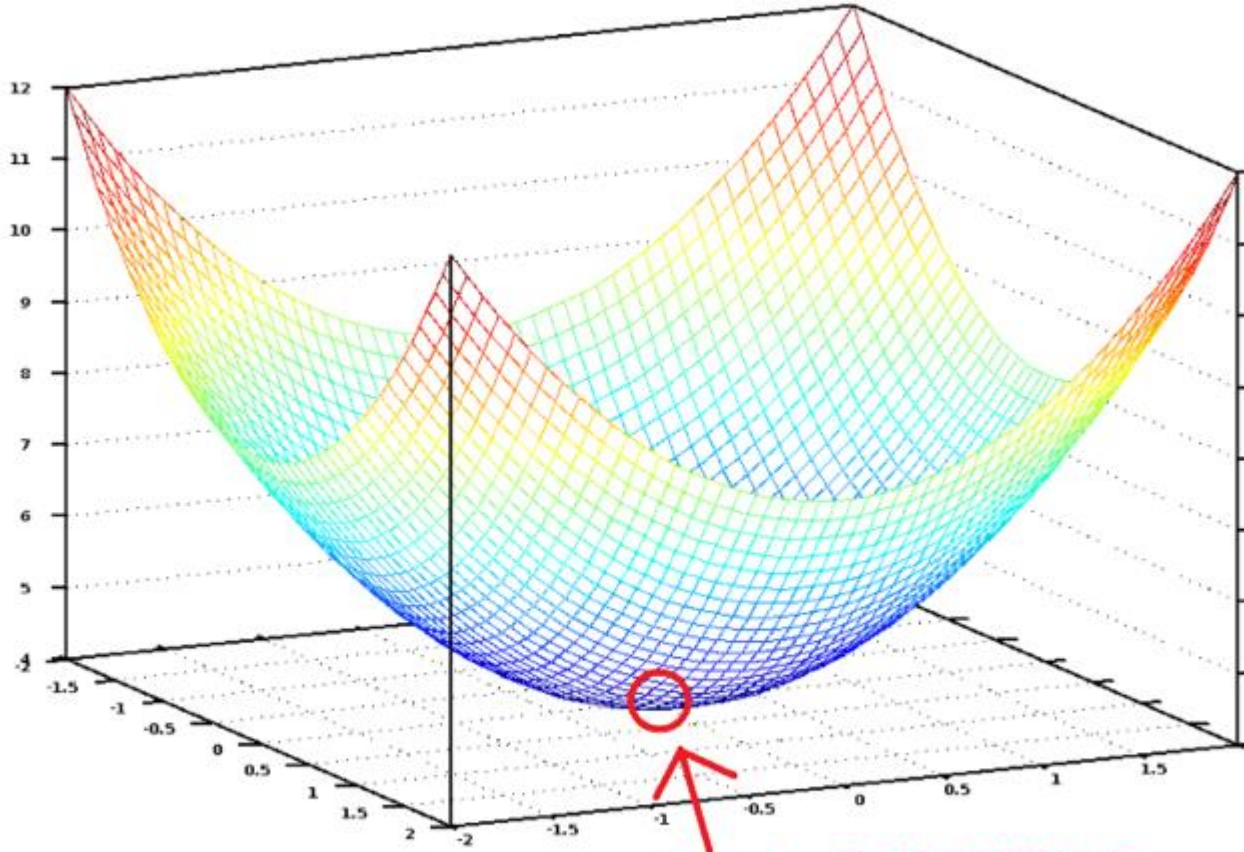
- A vector containing the partial derivatives of the function  $f$  at certain point  $x$
- Direction of **steepest ascent** (i.e., maximum increase of the function  $f$ ) from  $x$ 
  - Thus,  $-\nabla f$  is the direction of steepest descent (maximum decrease)

# Supervised Learning

$$Y_{(\text{predict})} = (W1*X1) + (W2*X2) + \dots + (Wn*Xn) + b$$

$$\text{Cost Function} : (Y - Y_{(\text{predict})})^2$$

## ✓ Gradient Descent



*cost function이 최소화되는 곳*

*= 가장 효율적인 회귀식*

## Supervised Learning

### ✓ Stochastic Gradient Descent

**“Gradient Descent is computationally expensive”**



# Supervised Learning

✓HOL



Automobile Price Data

## Supervised Learning

✓HOL

**“Azure Machine Learning Studio”**

**(<https://studio.azureml.net/>)**

# Supervised Learning

✓HOL

1) New

2) EXPERIMENT

3) Blank Experiment

# Supervised Learning

검색창: Automobile price data 선택

The screenshot displays the Microsoft Azure Machine Learning Studio interface. The top navigation bar shows the title 'Microsoft Azure Machine Learning Studio' and the user 'Jinheon Baek-Free-Works...'. The main workspace is titled 'Linear Regression' and contains a single data asset named 'Automobile price data (Raw)' with a circled '1' below it. A 'Mini Map' in the bottom-left corner provides a smaller view of the workspace. The left sidebar shows a search bar with 'auto' and a list of 'Saved Datasets' including 'Automobile price data (...)' and 'MPG data for various au...'. The right sidebar shows the 'Properties' tab for the selected dataset, listing details: SUBMITTED BY (Microsoft C...), SIZE (25.8 KB), FORMAT (GenericCSV), and CREATED ON (4/9/2015 7...). A 'Quick Help' section at the bottom right notes that a 'Missing Value Scrubber module required' for certain data analysis tasks. The bottom toolbar includes icons for '+ NEW', 'RUN HISTORY', 'SAVE', 'SAVE AS', 'DISCARD CHANGES', 'RUN', 'SET UP WEB SERVICE', and 'PUBLISH TO GALLERY'.

# Supervised Learning

검색창: Select Columns in Dataset 선택  
Automobile price data → Select Columns ... 이어주기

The screenshot displays the Microsoft Azure Machine Learning Studio interface. The main workspace is titled "Linear Regression" and shows a workflow diagram. The first node is "Automobile price data (Raw)", which is connected to a second node, "Select Columns in Dataset". The "Select Columns in Dataset" node has a red error icon, indicating a problem. The left sidebar contains a search bar and a list of modules categorized under "Data Transformation", "Feature Selection", "Statistical Functions", and "Text Analytics". The right sidebar shows the "Properties" pane with sections for "Experiment Properties", "Summary", and "Description". The bottom of the interface features a toolbar with icons for "NEW", "RUN HISTORY", "SAVE", "SAVE AS", "DISCARD CHANGES", "RUN", "SET UP WEB SERVICE", and "PUBLISH TO GALLERY".

# Supervised Learning

Select Columns in Dataset 클릭 (오른쪽 메뉴 확인)

Launch column selector 클릭

With rules → All columns → Exclude → normalized-losses (column names)

The screenshot displays the Microsoft Azure Machine Learning Studio interface. The main workspace shows a workflow for a Linear Regression model. The first step is 'Automobile price data (Raw)', which feeds into the 'Select Columns in Dataset' step, marked with a circled '1'. The left sidebar contains a 'select' search bar and a list of operations under 'Data Transformation' and 'Feature Selection'. The right sidebar shows the 'Properties' panel for the 'Select Columns in Dataset' step, with the 'Selected columns' set to 'All columns' and 'Exclude column names' set to 'normalized-losses'. A 'Launch column selector' button is visible. The bottom status bar includes icons for 'NEW', 'RUN HISTORY', 'SAVE', 'SAVE AS', 'DISCARD CHANGES', 'RUN', 'SET UP WEB SERVICE', and 'PUBLISH TO GALLERY'.

# Supervised Learning

검색창: Split Data

Select Columns in Dataset → Split Data (연결)

Split Data 오른쪽 메뉴 → Fraction of rows ... (0.66 설정)

Microsoft Azure Machine Learning Studio

Linear Regression

split

Automobile price data (Raw)

Select Columns in Dataset

Split Data

1 2

Mini Map

Properties

Split Data

Splitting mode: Split Rows

Fraction of rows in the first set: 0.66

Randomized split: ☒

Random seed: 0

Stratified split: False

Quick Help

Split the rows of a dataset into two distinct sets (more help...)

+ NEW

RUN HISTORY

SAVE

SAVE AS

DISCARD CHANGES

RUN

SET UP WEB SERVICE

PUBLISH TO GALLERY

# Supervised Learning

검색창: Linear Regression

Linear Regression 오른쪽 메뉴 ->

- 1) Solution Method: Online Gradient Descent
- 2) Number of training epochs: 30

The screenshot displays the Microsoft Azure Machine Learning Studio interface. The main workspace shows a workflow for Linear Regression. The workflow starts with a dataset named 'Automobile price data (Raw)', which is processed by a 'Select Columns in Dataset' node, followed by a 'Split Data' node. The 'Linear Regression' node is highlighted with a blue border and a circled '1'. A 'Mini Map' at the bottom left provides a smaller overview of the workflow. The left sidebar contains a search bar and a list of categories: Trained Models, Feature Selection, Machine Learning, Statistical Functions, and Deprecated. The 'Machine Learning' category is expanded, showing 'Initialize Model' and 'Regression'. The 'Regression' category is further expanded, showing 'Bayesian Linear Regr...' and 'Linear Regression'. The right sidebar shows the 'Properties' panel for the 'Linear Regression' node. The 'Solution method' is set to 'Online Gradient Descent', the 'Create trainer mode' is 'Single Parameter', the 'Learning rate' is '0.1', the 'Number of training epochs' is '30', and the 'L2 regularization weight' is '0.001'. Other options like 'Normalize features', 'Average final hypothe...', 'Decrease learning rate', 'Random number seed', and 'Allow unknown categ...' are also visible. The bottom status bar includes icons for 'NEW', 'RUN HISTORY', 'SAVE', 'SAVE AS', 'DISCARD CHANGES', 'RUN', 'SET UP WEB SERVICE', and 'PUBLISH TO GALLERY'.



# Supervised Learning

검색창: Train Model

Linear Regression → Train Model (왼쪽 연결)

Split Data → Train Model (오른쪽 연결)

The screenshot displays the Microsoft Azure Machine Learning Studio interface. The left sidebar contains a search bar with the text "train" and a list of modules categorized under "Machine Learning". The "Train" category is expanded, showing modules like "Sweep Clustering", "Train Anomaly Detect...", "Train Clustering Model", "Train Matchbox Reco...", "Train Model", and "Tune Model Hyperpa...". The "Train Model" module is highlighted. The main workspace shows a workflow diagram titled "Linear Regression". The workflow starts with "Automobile price data (Raw)", which connects to "Select Columns in Dataset". This module then connects to "Split Data". The "Split Data" module has two outputs: one connects to "Linear Regression" and the other to "Train Model". The "Train Model" module is highlighted with a red circle and the number "1". A "Mini Map" in the bottom left corner provides a overview of the workflow. The right sidebar shows the "Properties" panel for the "Train Model" module, with the "Label column" property set to "Selected columns: Launch the selector tool to make a selection". The bottom status bar includes icons for "NEW", "RUN HISTORY", "SAVE", "SAVE AS", "DISCARD CHANGES", "RUN", "SET UP WEB SERVICE", and "PUBLISH TO GALLERY".

# Supervised Learning

Train Model 오른쪽 메뉴 → Launch column selector  
With Rules → Include → column names → price

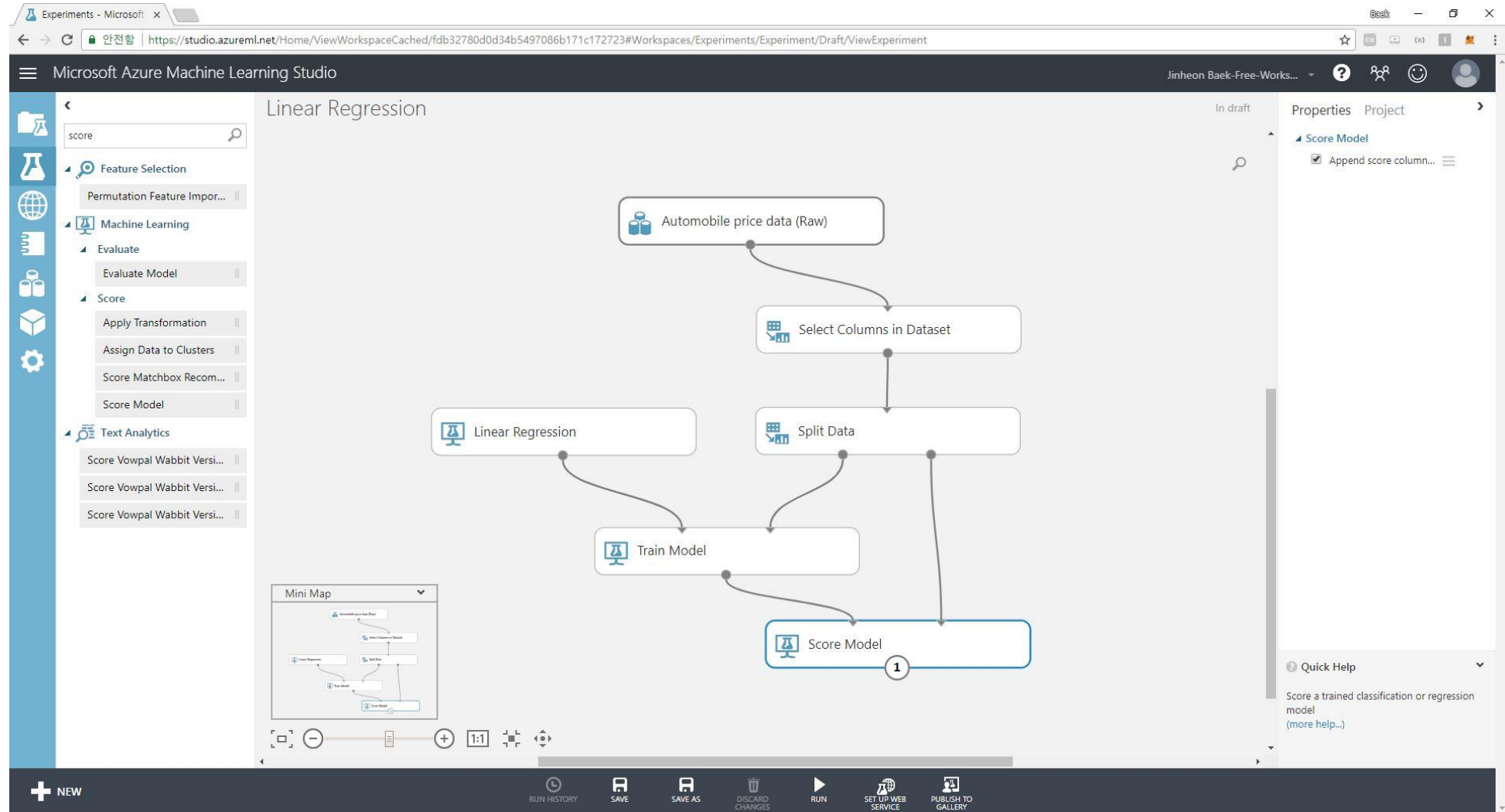
The screenshot displays the Microsoft Azure Machine Learning Studio interface. The main workspace shows a 'Linear Regression' experiment with the dataset 'Automobile price data (Raw)'. A 'Select a single column' dialog box is open, showing the 'WITH RULES' tab. The dialog has a dropdown menu set to 'Include', another dropdown set to 'column names', and a text input field containing 'price'. The right-hand pane shows the 'Train Model' button and a 'Launch column selector' button. The bottom status bar includes icons for 'NEW', 'RUN HISTORY', 'SAVE', 'SAVE AS', 'DISCARD CHANGES', 'RUN', 'SET UP WEB SERVICE', and 'PUBLISH TO GALLERY'.

# Supervised Learning

검색창: Score Model

Train Model → Score Model (왼쪽 연결)

Split Data → Score Model (오른쪽 연결)

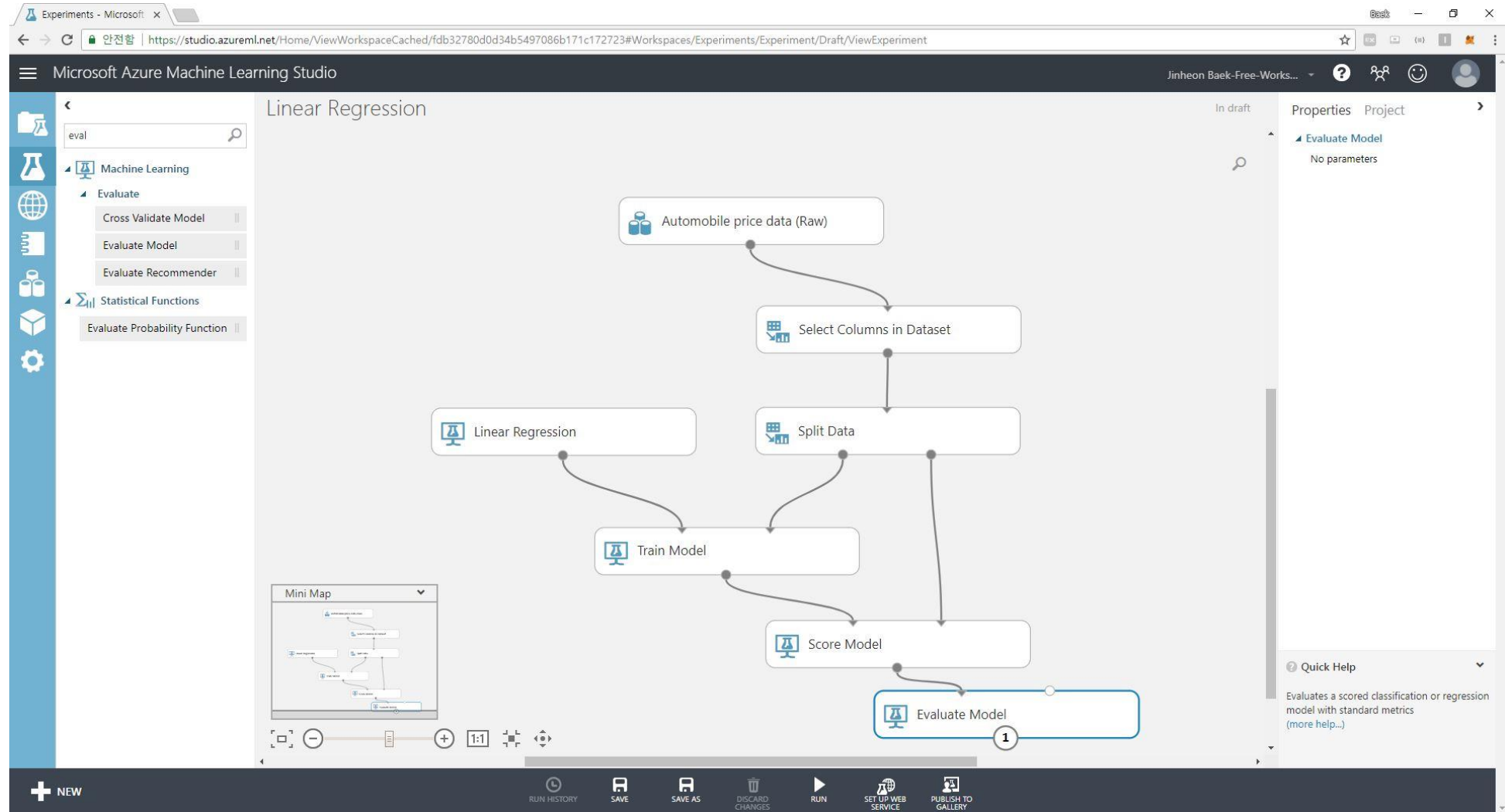


# Supervised Learning

검색창: Evaluate Model

Score Model → Evaluate Model (왼쪽 연결)

하단 Run 실행 후 결과 보기 !



# Supervised Learning

## [Deep Learning]

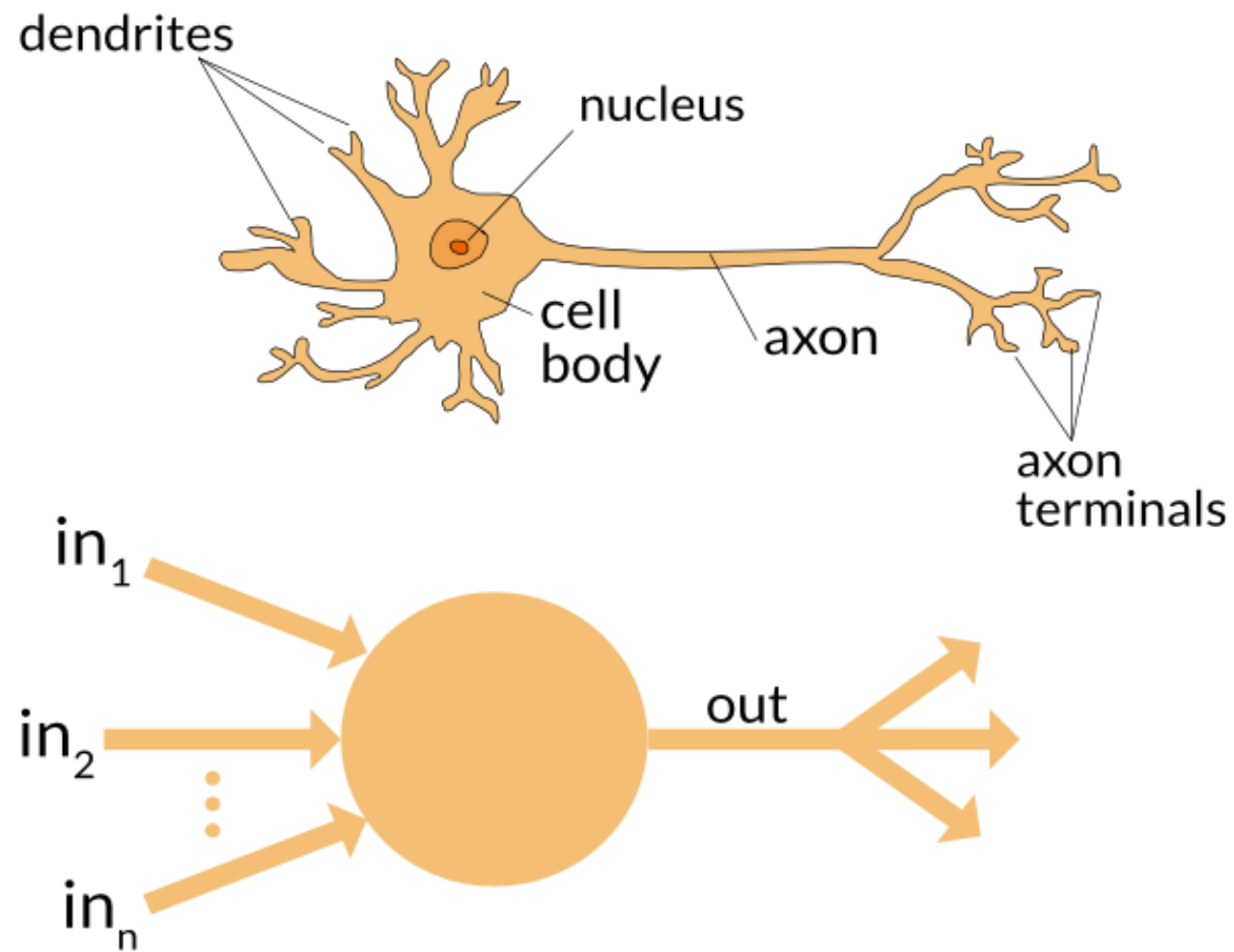
✓ Neural Network  
: Simple MLP (Multi-Layer Perceptron)

Note)

이건 mnist 로 하자!!!!

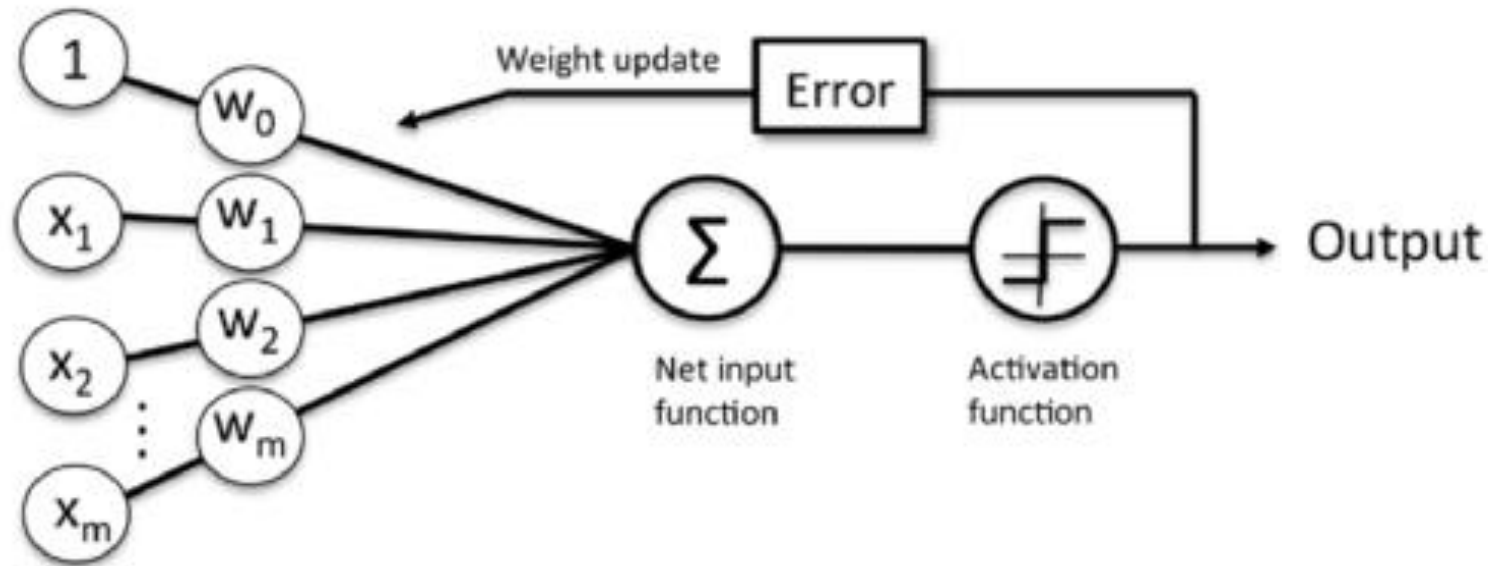
젠장, Neural Network 설명도 해야해...

# Supervised Learning



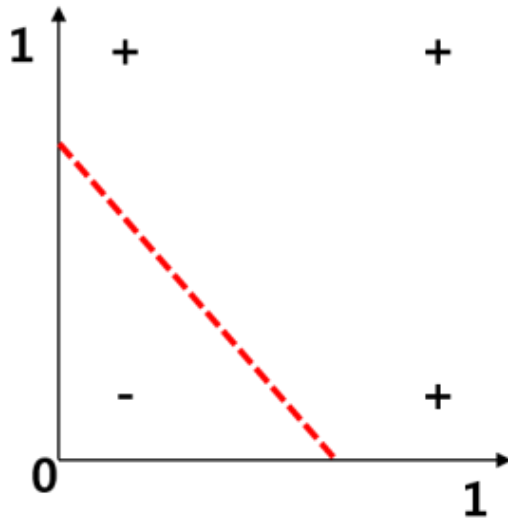
# Supervised Learning

## Single Layer Neural Network



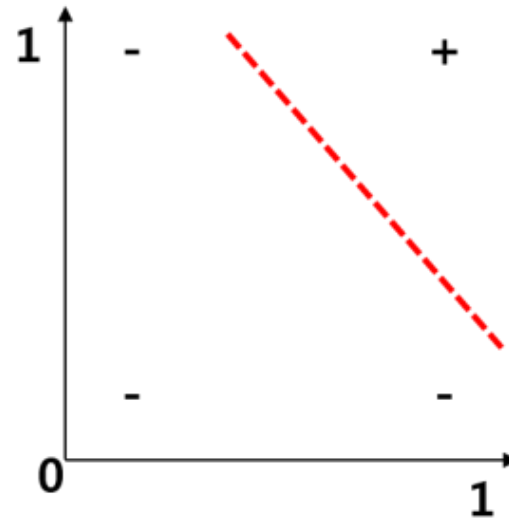
# Supervised Learning

OR



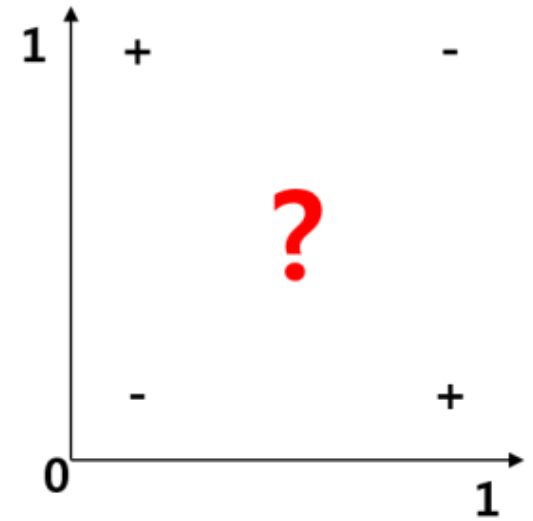
$x_1$	$x_2$	$y$
0	0	0
0	1	1
1	0	1
1	1	1

AND



$x_1$	$x_2$	$y$
0	0	0
0	1	0
1	0	0
1	1	1

XOR

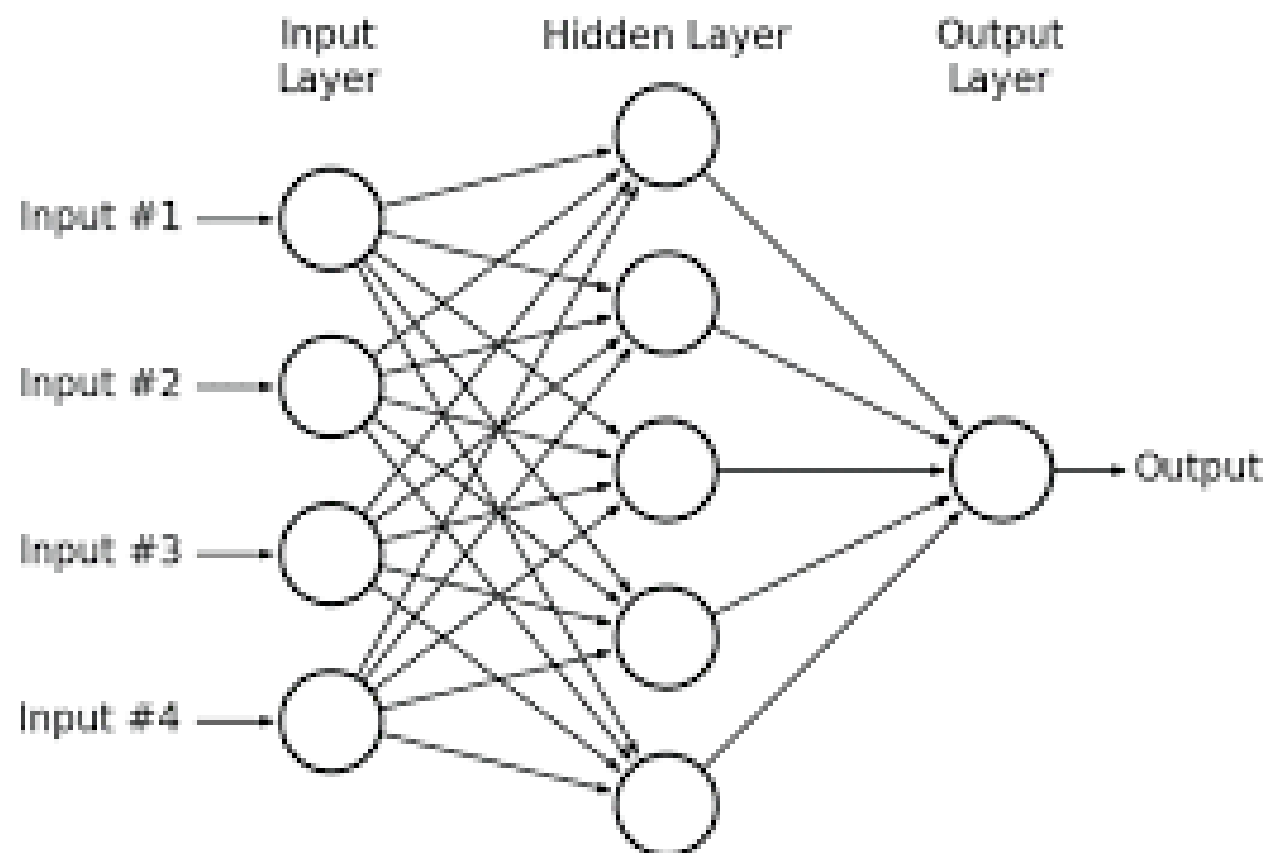


$x_1$	$x_2$	$y$
0	0	0
0	1	1
1	0	1
1	1	0

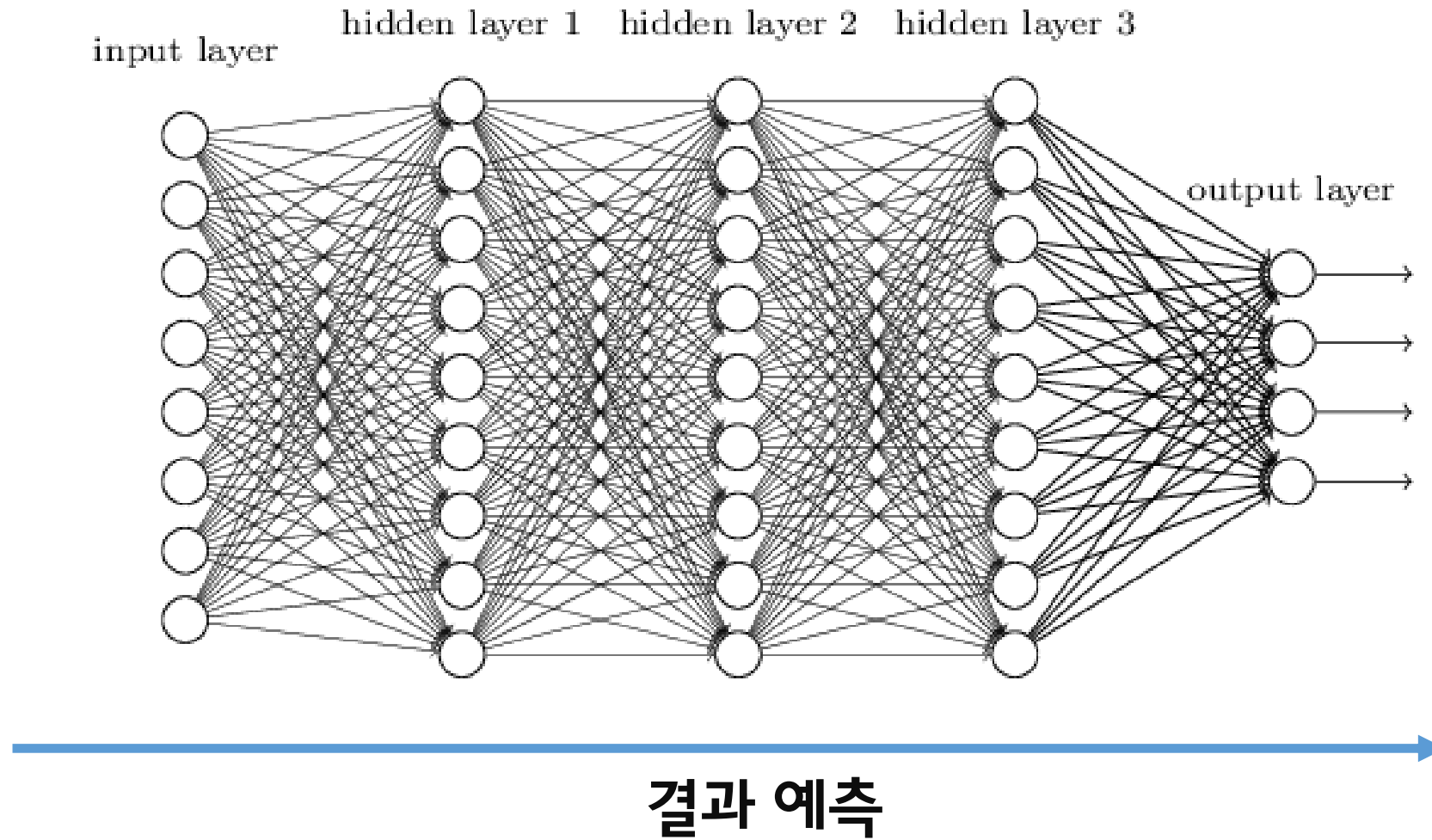


# Supervised Learning

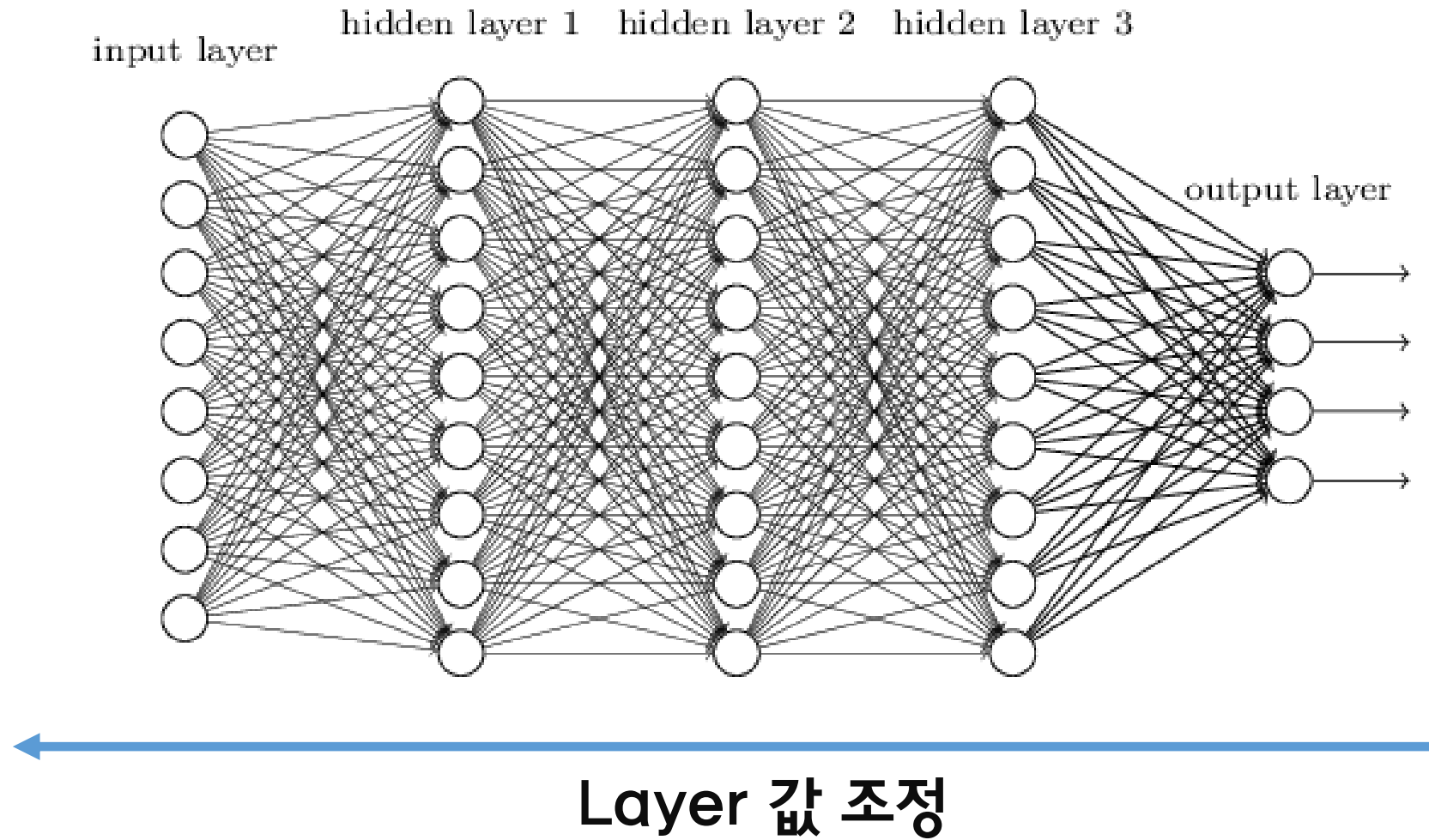
## Multi-Layer Neural Network



# Neural-network



# Neural-network



✓HOL

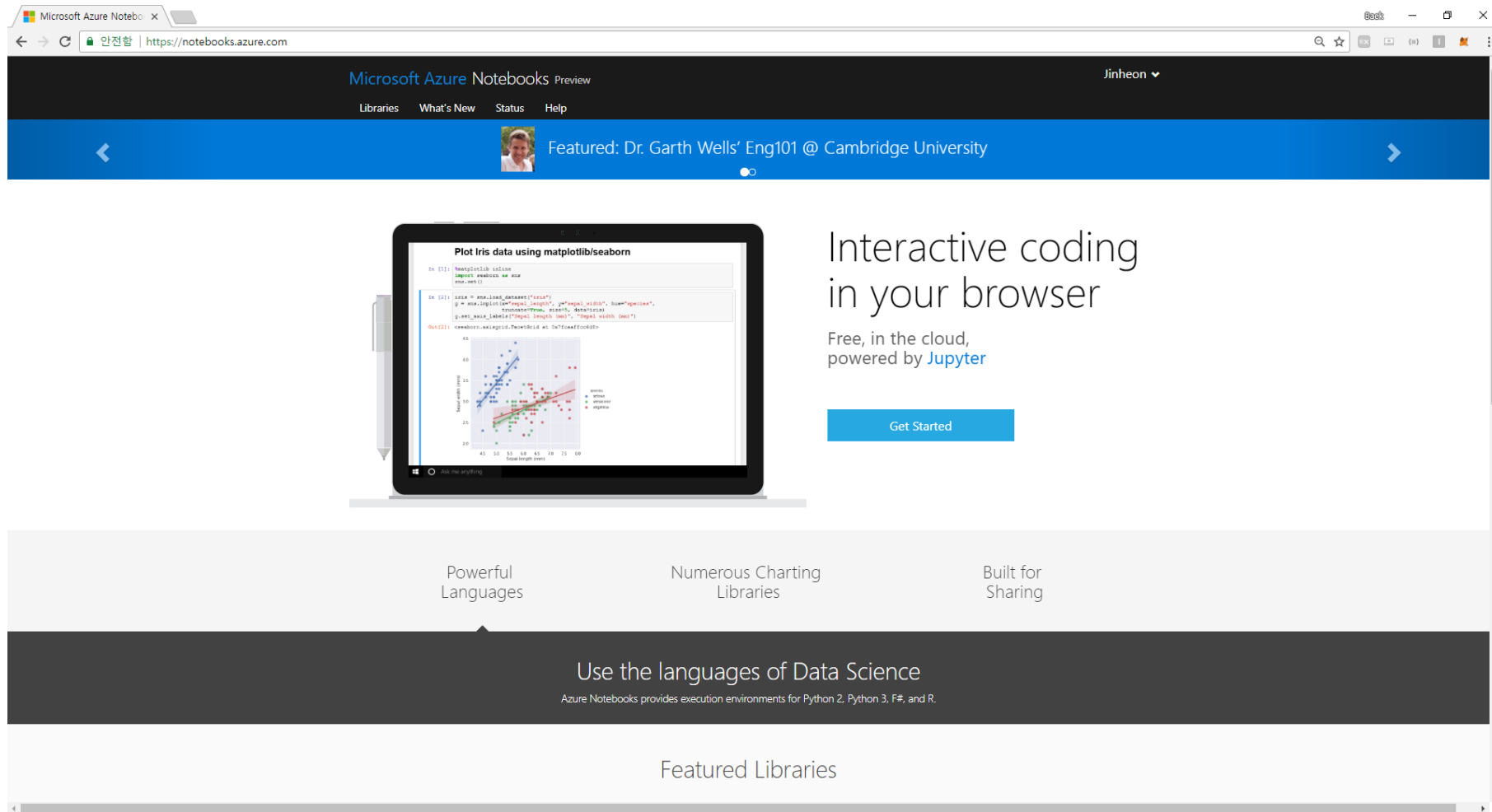
**“MNIST Data Set ?”**



28 x 28, 필기체 숫자 ( 0 ~ 9 )

# Neural Network

## ✓ Azure Notebook (<https://notebooks.azure.com>)



The screenshot shows the Microsoft Azure Notebooks website. The browser address bar displays "https://notebooks.azure.com". The page header includes "Microsoft Azure Notebooks Preview" and a user profile "Jinheon". A blue banner features a profile picture and the text "Featured: Dr. Garth Wells' Eng101 @ Cambridge University". The main content area shows a laptop displaying a scatter plot titled "Plot Iris data using matplotlib/seaborn". To the right of the laptop, the text "Interactive coding in your browser" is displayed, followed by "Free, in the cloud, powered by Jupyter" and a "Get Started" button. Below this, three features are listed: "Powerful Languages", "Numerous Charting Libraries", and "Built for Sharing". A dark grey bar contains the text "Use the languages of Data Science" and "Azure Notebooks provides execution environments for Python 2, Python 3, F#, and R." At the bottom, a section titled "Featured Libraries" is partially visible.

Microsoft Azure Notebooks Preview Jinheon

Libraries What's New Status Help

Featured: Dr. Garth Wells' Eng101 @ Cambridge University

Plot Iris data using matplotlib/seaborn

```
In [1]: %matplotlib inline
import pandas as pd
from sklearn import datasets

In [2]: iris = datasets.load_iris(data_path='iris.csv')
g = sns.grelplot(iris['sepal_length'], iris['petal_length'], hue='species',
               palette='magma', style='points', markers='o')
g.set_xlabel('sepal_length (cm)')
g.set_ylabel('petal_length (cm)')
Out[2]: <matplotlib.axes._subplots.AxesSubplot at 0x7f0e0f0e0f00>
```

Interactive coding in your browser

Free, in the cloud, powered by Jupyter

Get Started

Powerful Languages Numerous Charting Libraries Built for Sharing

Use the languages of Data Science

Azure Notebooks provides execution environments for Python 2, Python 3, F#, and R.

Featured Libraries

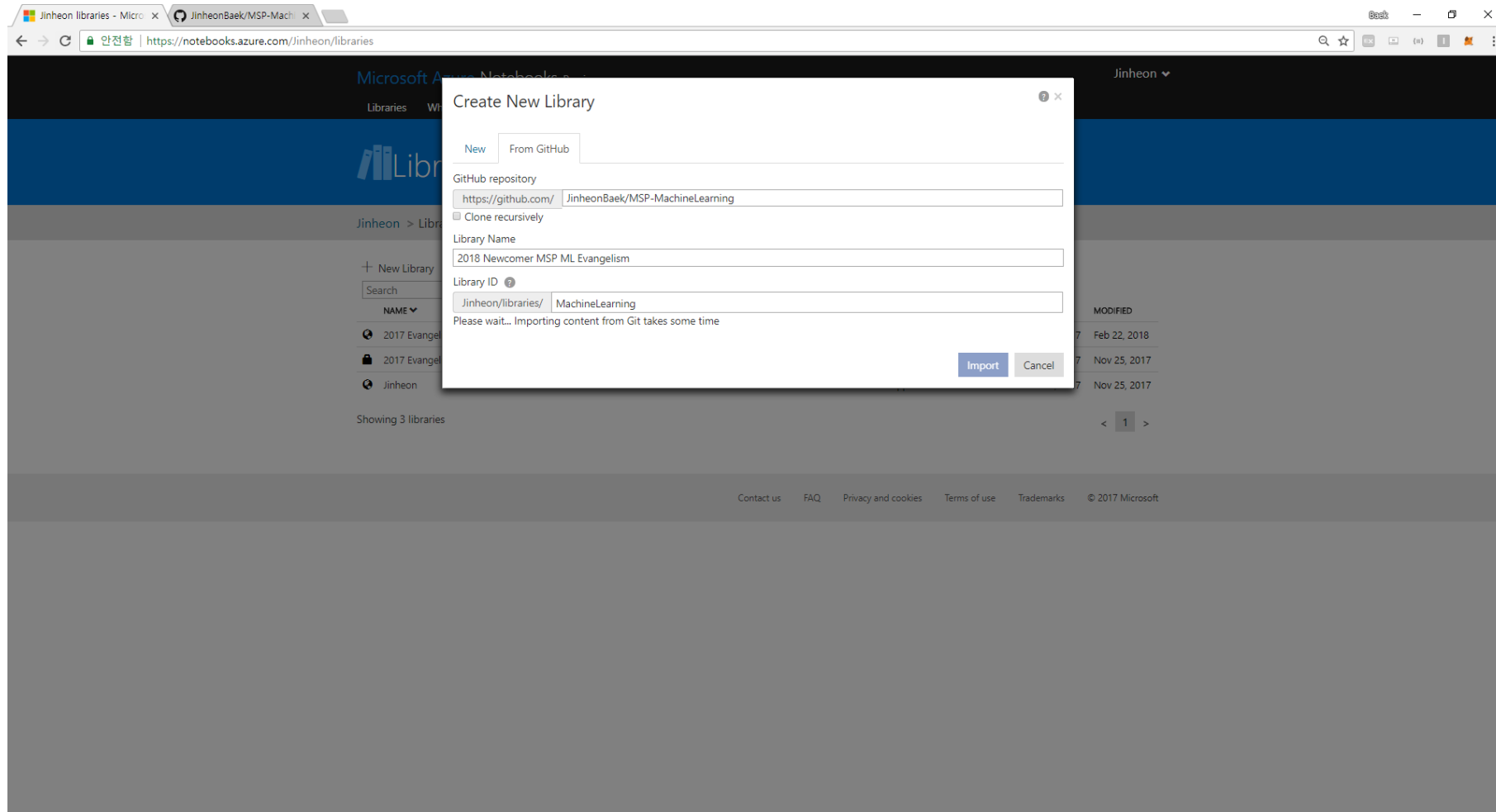
## Neural Network

✓ Azure Notebook (<https://notebooks.azure.com>)

1. Logic
2. Libraries 메뉴
3. New Library

GitHub repository: JinheonBaek/MSP-MachineLearning  
Library Name (예시): 2018 Newcomer MSP ML Evangelism  
Library ID (예시): MachineLearning

✓ Azure Notebook (<https://notebooks.azure.com>)





### ✓HOL - Challenge

**“Testing score is above 0.98”**

# Fitting

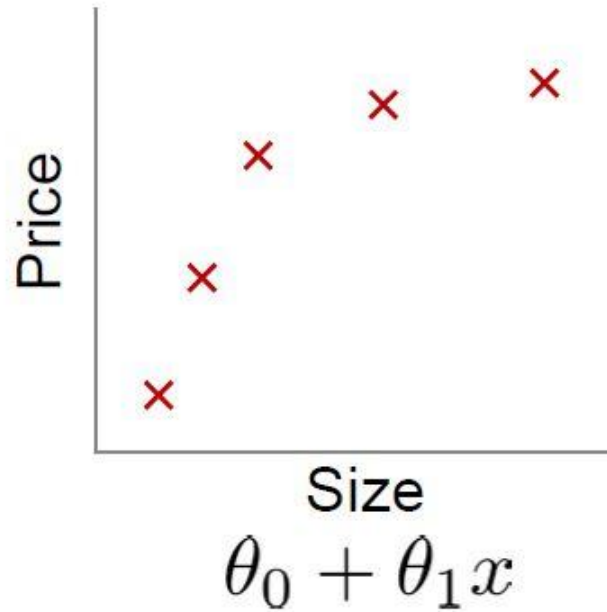
## Machine Learning

Microsoft Student Partner

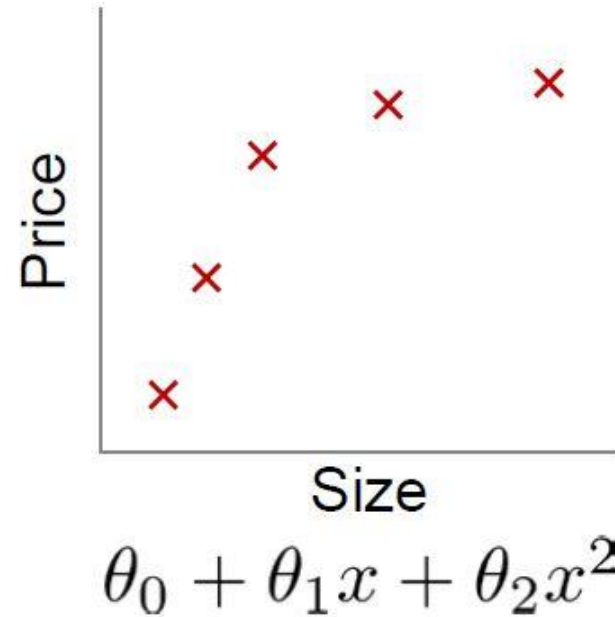
백진헌

# Fitting

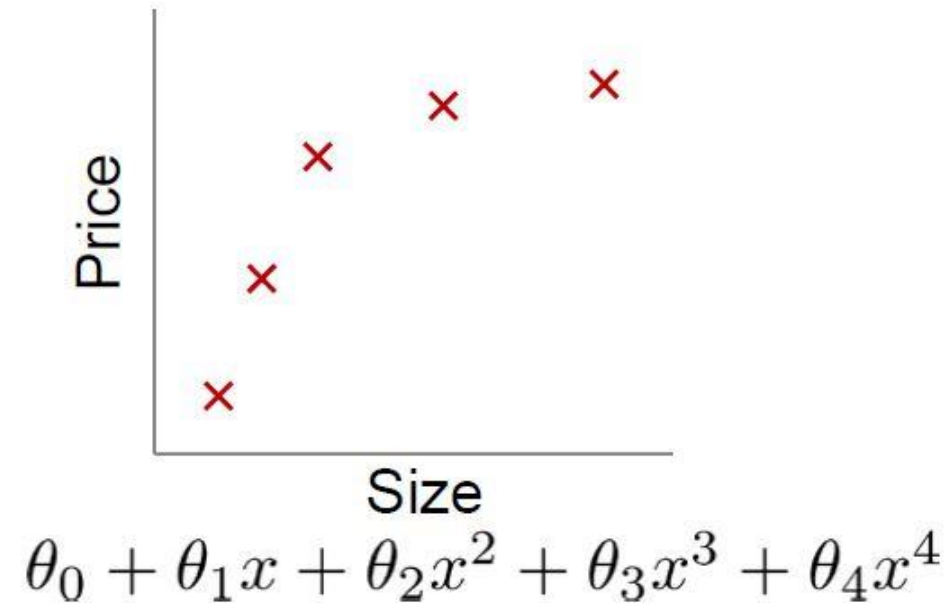
## ✓Fitting



High bias  
(underfit)



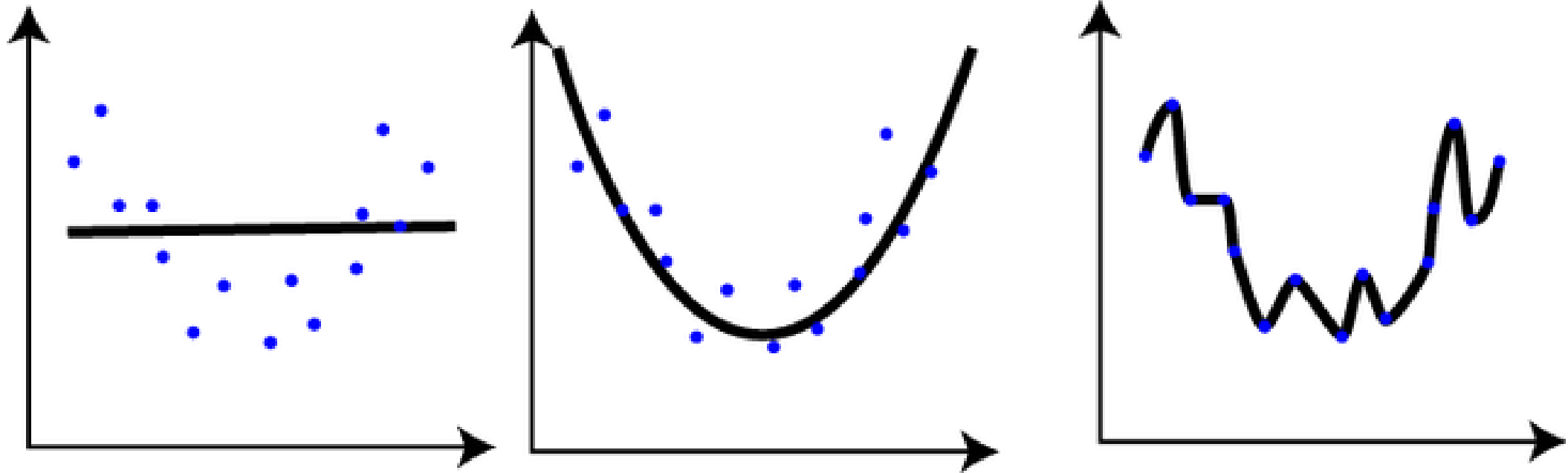
“Just right”



High variance  
(overfit)

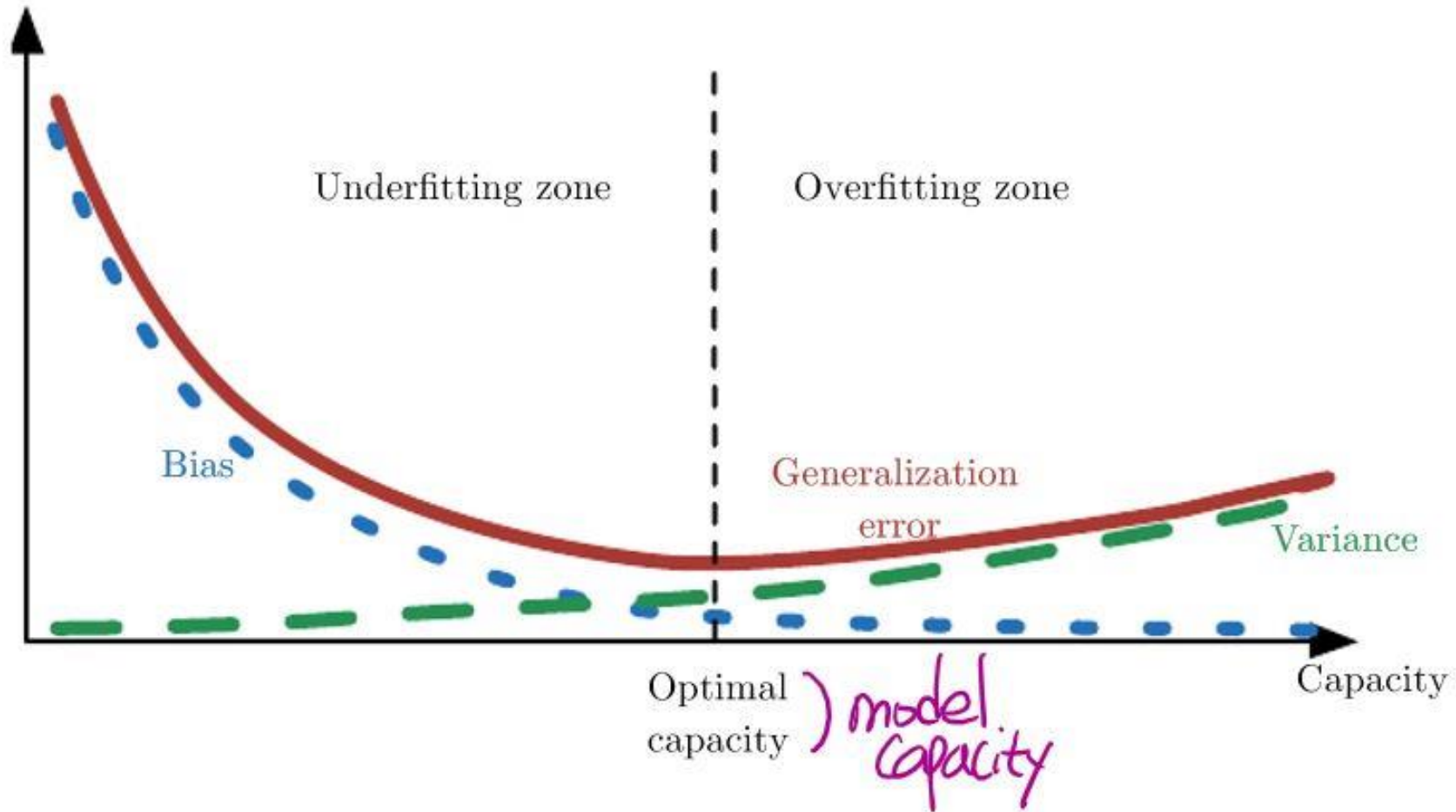
# Fitting

## ✓Fitting



# Fitting

✓ Fitting



## Fitting

✓ Fitting HOL

“ Connect Four ”

# Model

## Machine Learning

Microsoft Student Partner

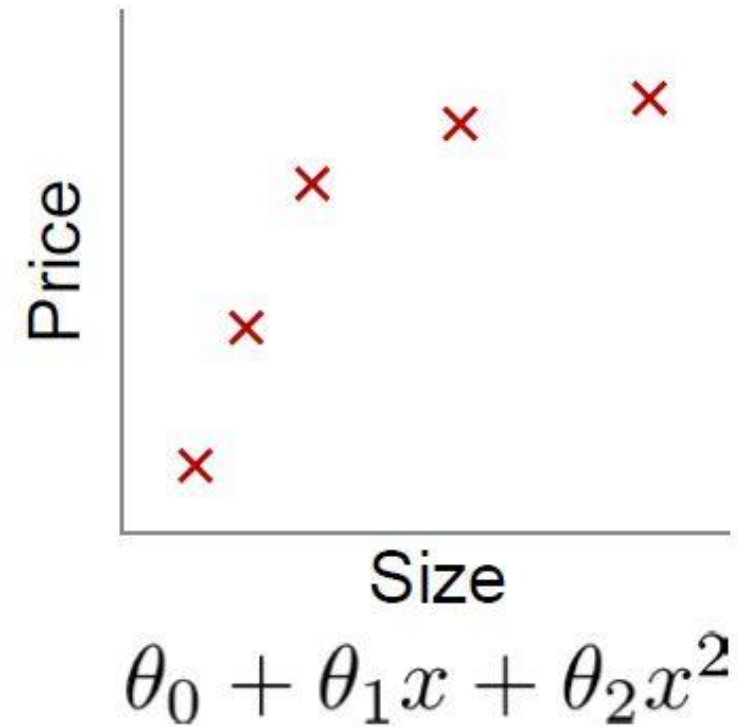
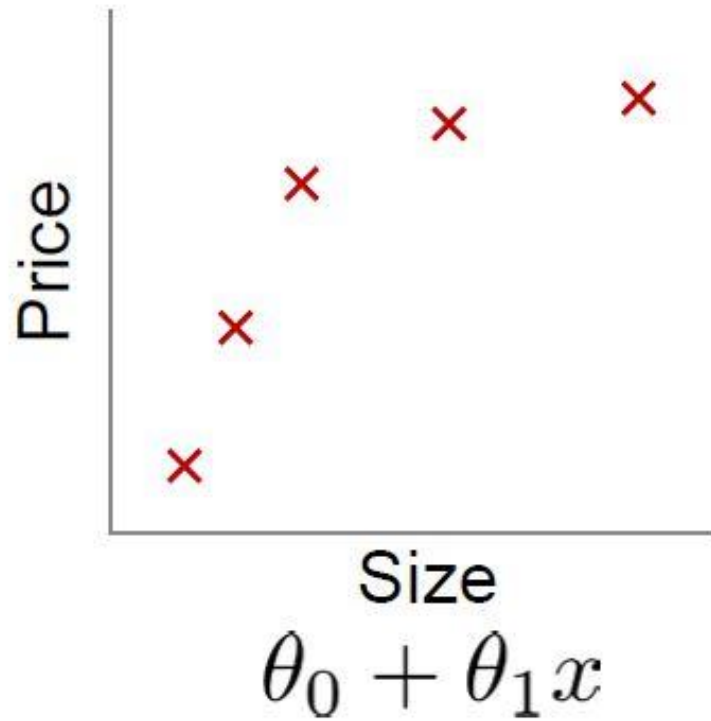
백진헌

## Models

- ✓ **Linear vs Nonlinear**
- ✓ **Black box vs Descriptive**
- ✓ **First-principle vs Data-driven**



## ✓ Linear vs Nonlinear



### ✓ **Blackbox vs Descriptive Model**

#### **Linear Regression**

**: Models are descriptive, because one can see which variables are weighed heaviest.**

#### **Neural Network**

**: Models are generally opaque.**

# Models

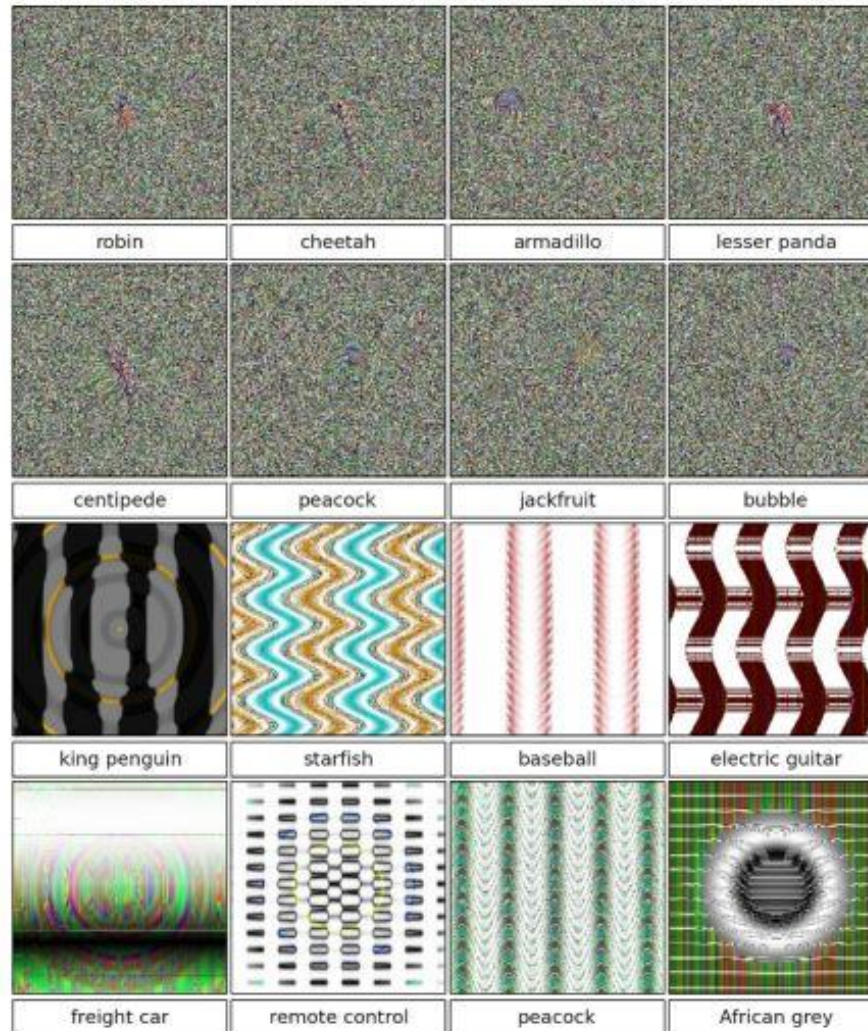
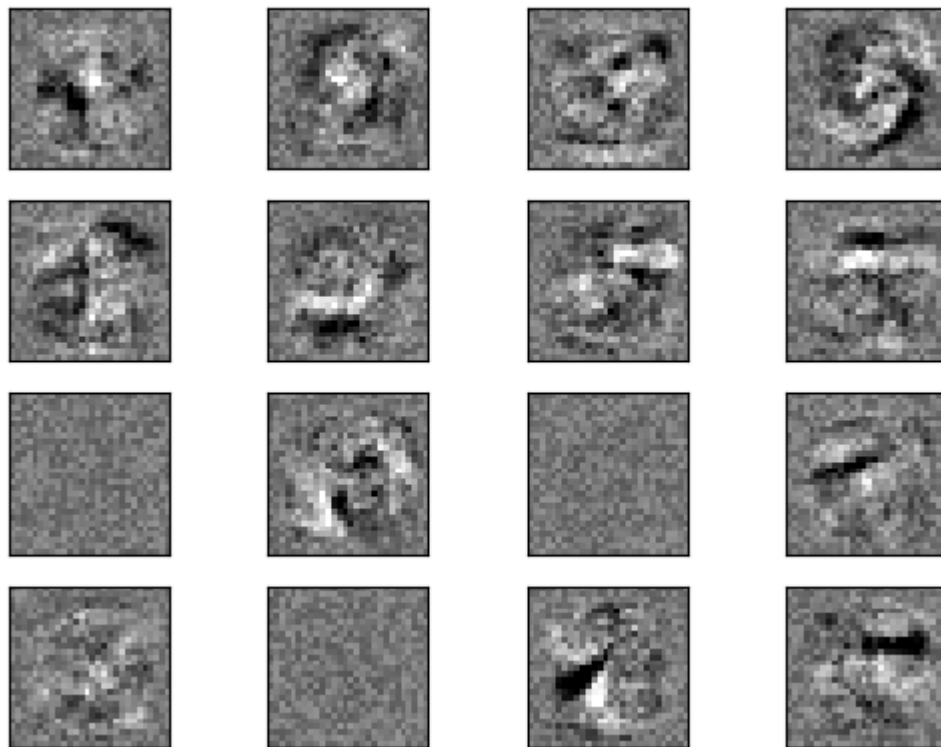


Figure 1. Evolved images that are unrecognizable to humans, but that state-of-the-art DNNs trained on ImageNet believe with  $\geq 99.6\%$  certainty to be a familiar object. This result highlights differences between how DNNs and humans recognize objects. Images are either directly (*top*) or indirectly (*bottom*) encoded.

# Models



### ✓ First-principle vs Data-driven

#### **First-principle**

**: Models based on a theoretical explanation.**

#### **Data-driven**

**: Models based on observed data correlations.**

## References

- ✓ **Deep Learning Book - MIT**
- ✓ **Machine Learning - Coursera**
- ✓ **Korea Univ. Cose432 (Data Science)**

# | Q & A

**Thanks**