# Machine Learning Review

## Tuesday 8h00 – 8h45

Géraldine Conti, Matthew Vowels, Bern Winter School on Machine Learning 2023, Muerren

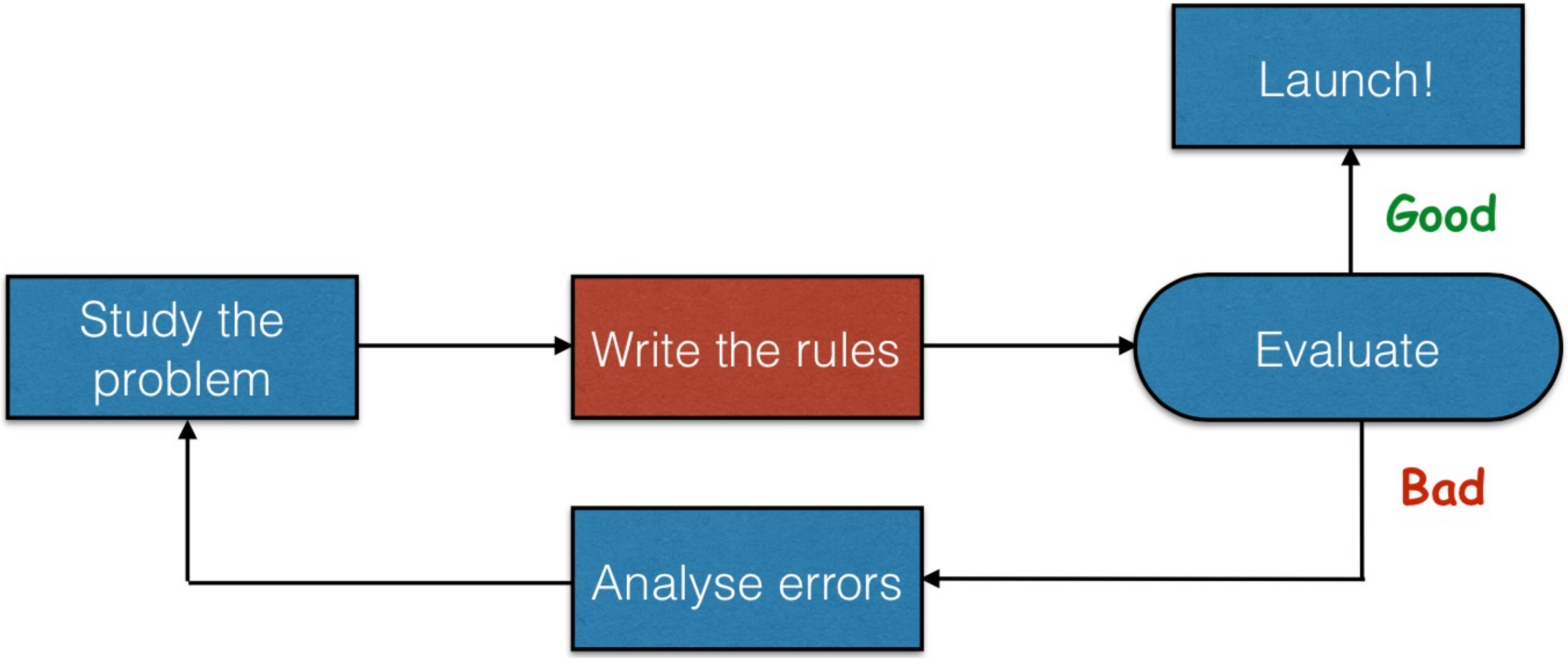
# Who am I?

- Engineering PhD @ CVSSP, Surrey UK
- Psych PhD UNIL (waiting to defend)
- Post-doc Sheffield, Post-doc UNIL
- Affiliate member of AI @ Surrey and machine learning CVSSP, University of Surrey
- Collaborator for causal discovery at EPFL
- Research interests:
  - Causal Inference
  - Causal Discovery
  - Semiparametrics
  - Deep Latent Variable Modeling
  - Disentanglement
  - Computer Vision
  - Multimodal Fusion
  - Psychology

# Useful Resources

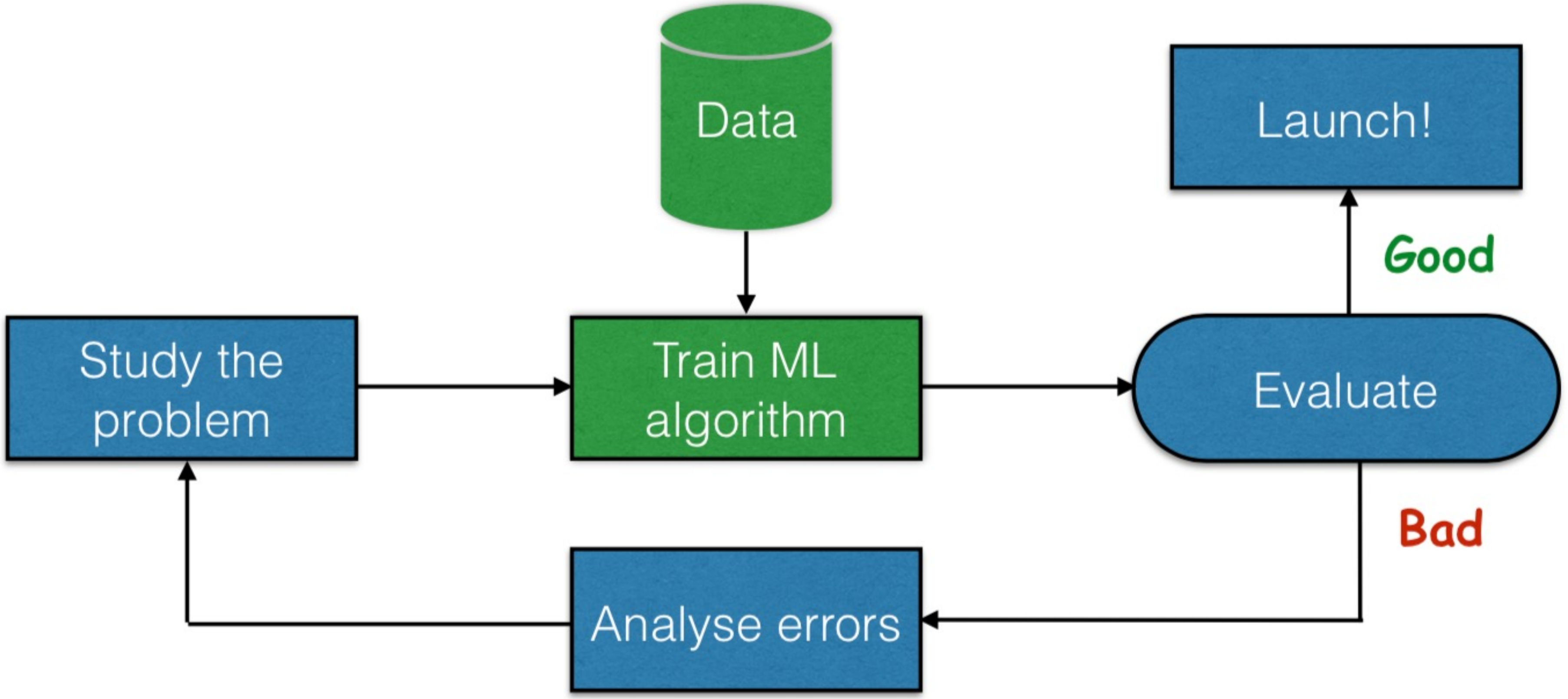
- **Deep Learning book (Goodfellow, Bengio, Courville)**
- **Lightning AI course with Sebastian Raschka**
  **https://lightning.ai/pages/courses/deep-learning-fundamentals/**
- **Machine Learning @ Stanford (Prof Andrew Ng)**
  **https://www.coursera.org/specializations/deep-learning**

# Traditional approach (Software 1.0)



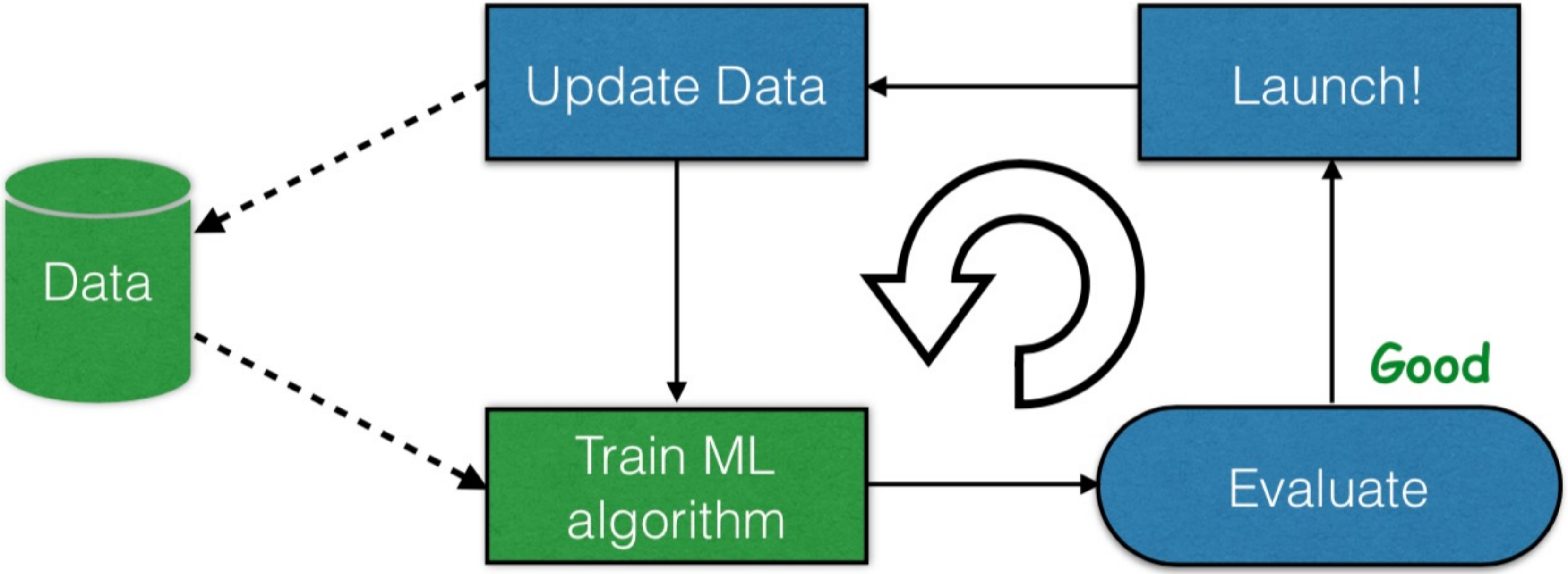

*List of all the knowledge and formal rules*

# Machine Learning approach (Software 2.0)



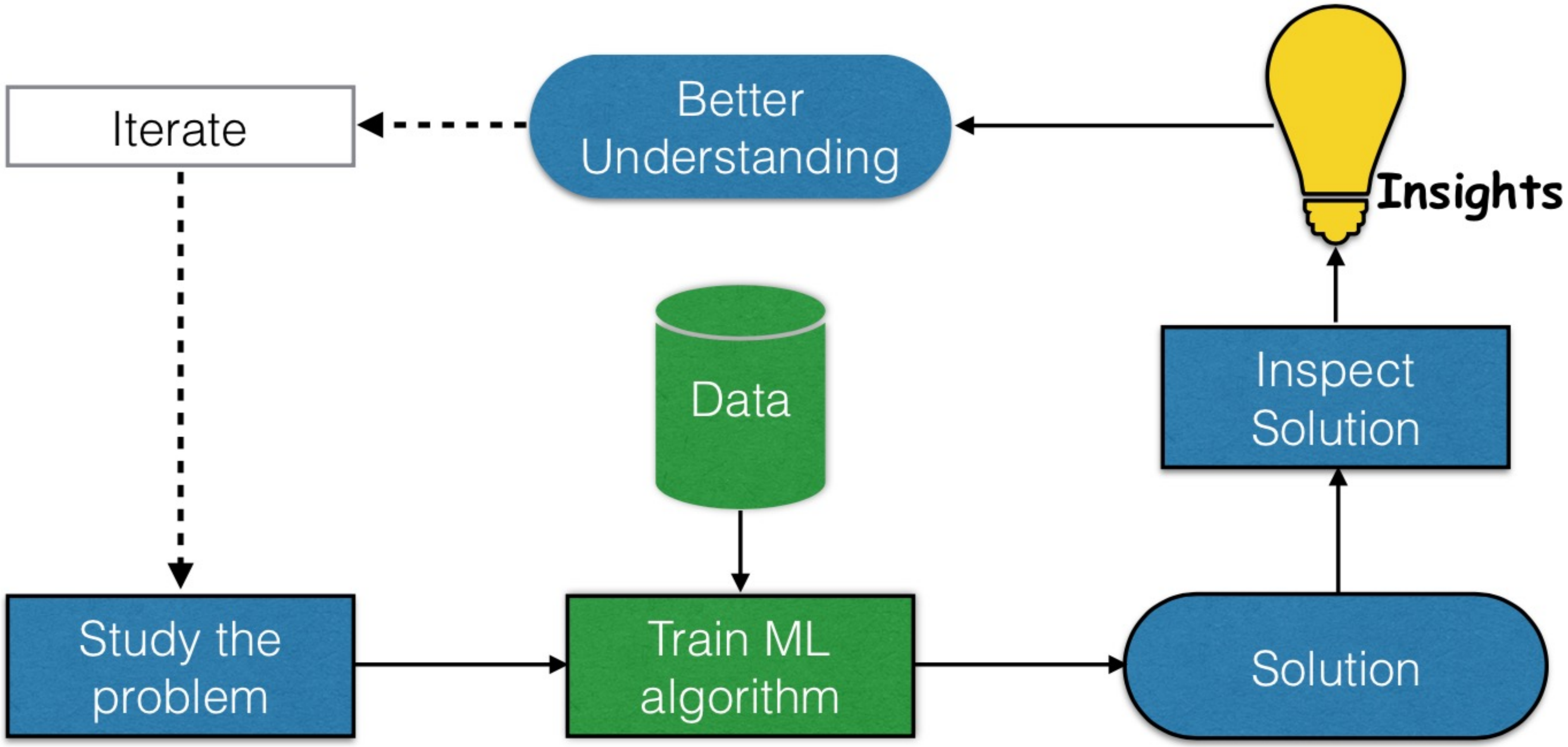

*Learning from examples*

# Machine Learning approach (Software 2.0)




*Adapting to change*

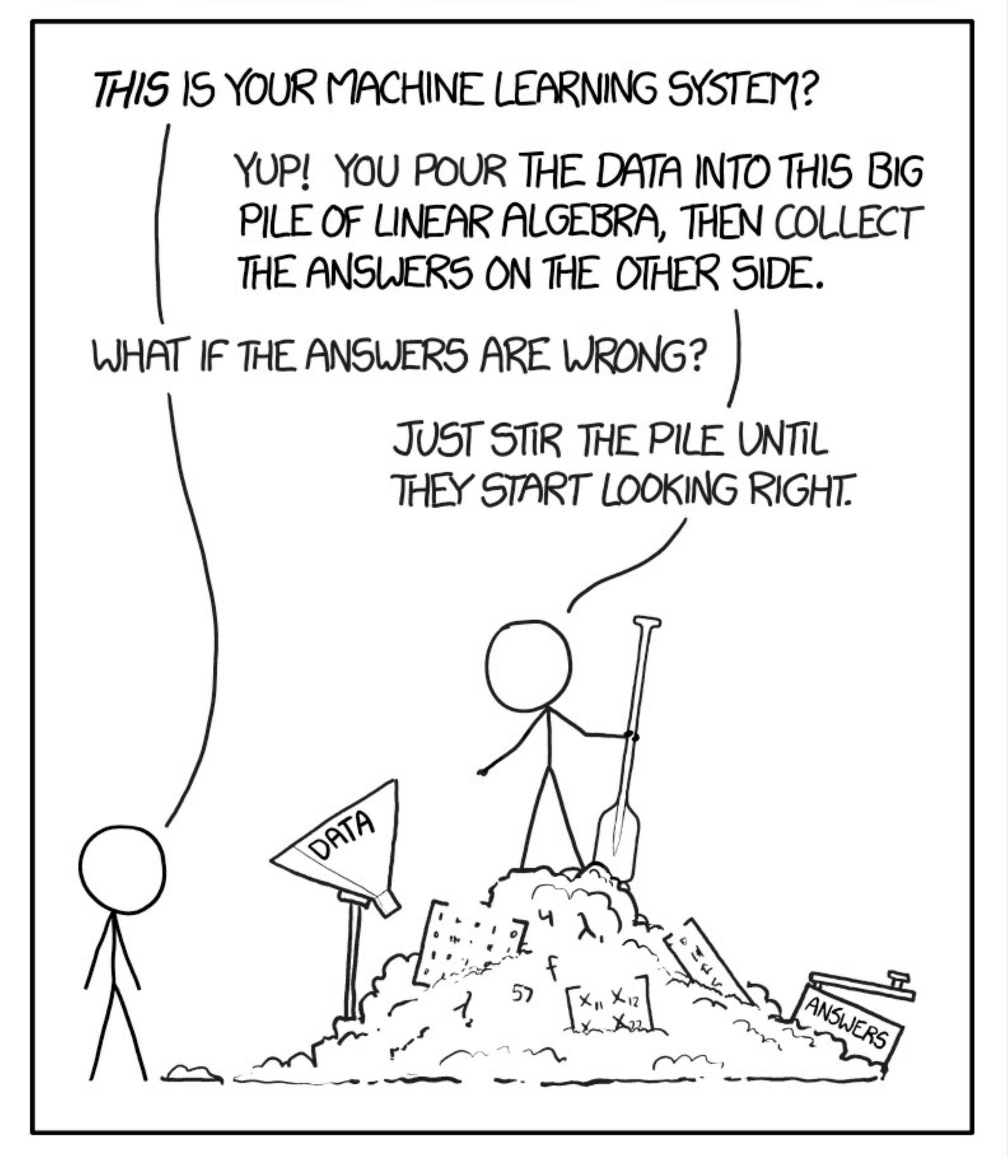
# Machine Learning approach (Software 2.0)




*Help Humans learn*

# What is Machine Learning ?

*"Can machines do what we (as thinking entities) can do?"* (Turing)



# Let’s warm up !

- Can you list **2 examples** of Machine Learning use in society ?
  1.
  2.
- Is Machine Learning used in your company/institution ? If yes, can you cite **1 use case** ?

- Do you know any “technical terms” about Machine Learning (algorithm names, …) ? If yes, can you list **up to 3 keywords** ?

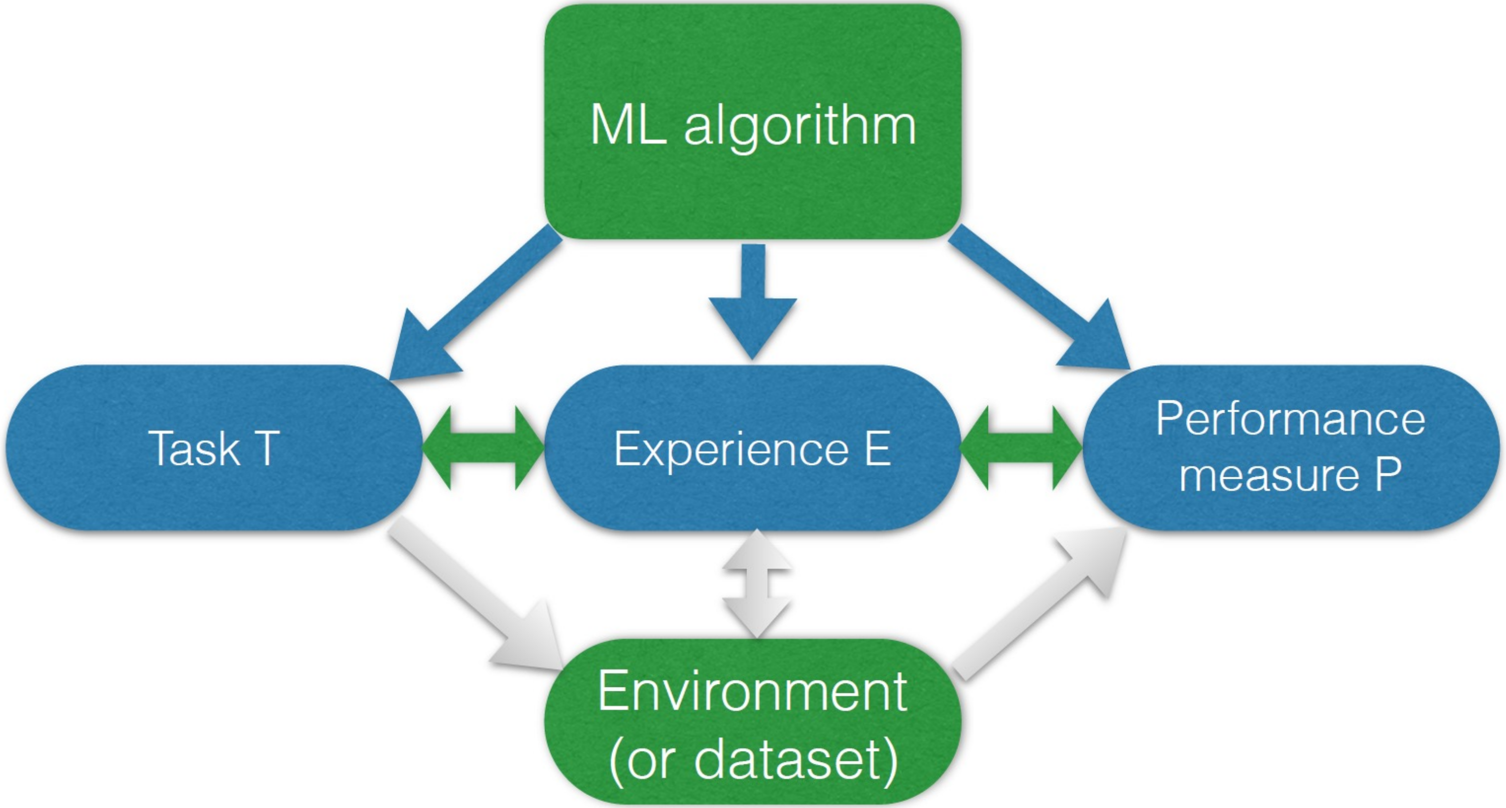
- How do you **feel** about Machine Learning and AI in general ?

# Definition… in words

*« 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.»*

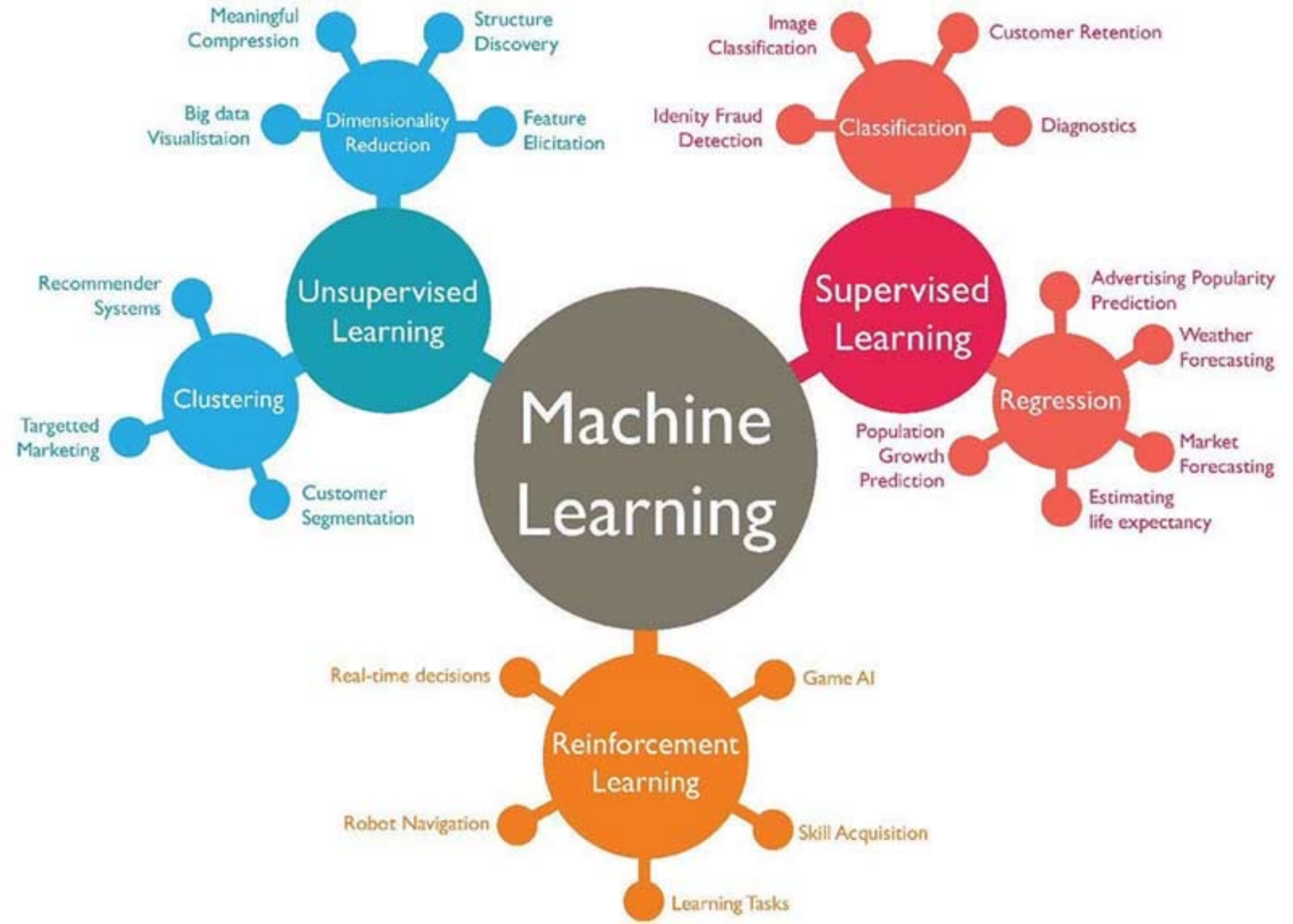
Tom M. Mitchell (1997)

# Definition… schematically



# Experience E

What data to use to solve the task

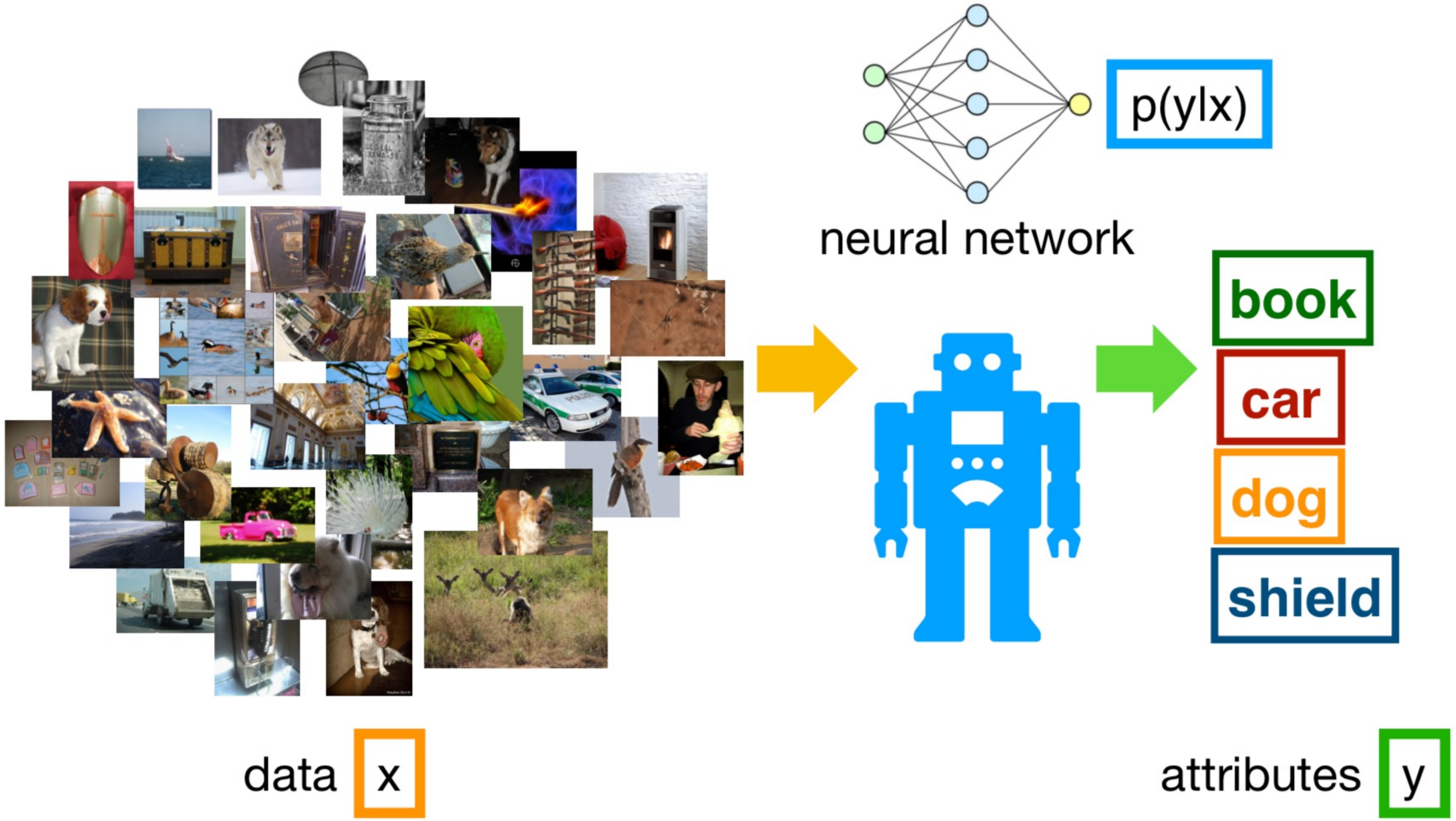



**Learning Pillars :** How much information is given to the ML algorithm

# Learning Pillars

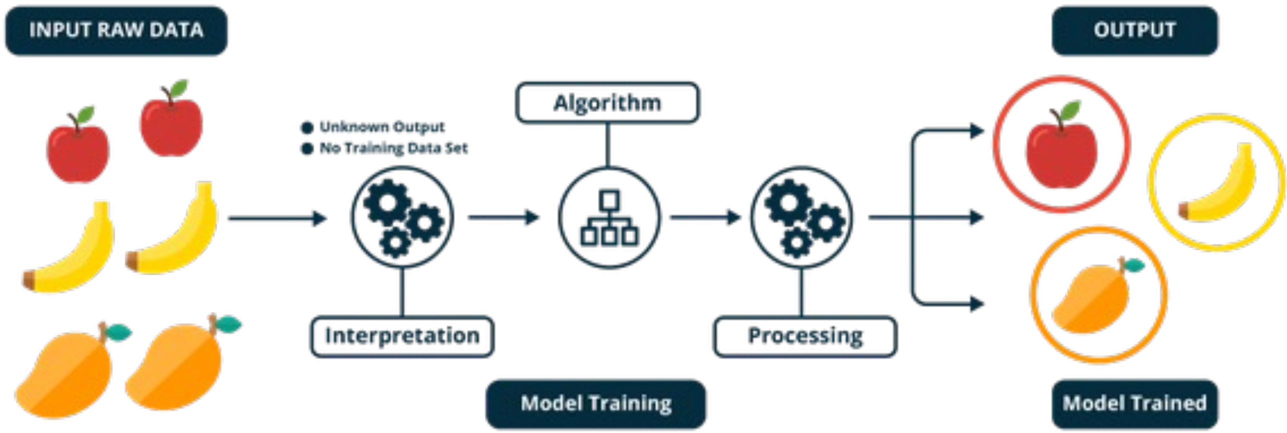
# Supervised Learning

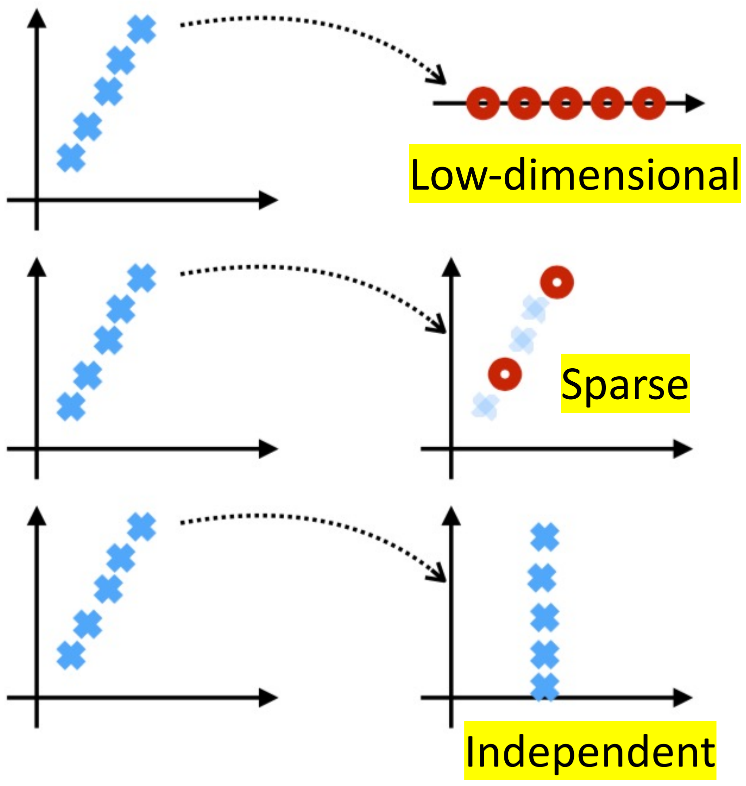
- Prediction of an output **y** given an input **x**



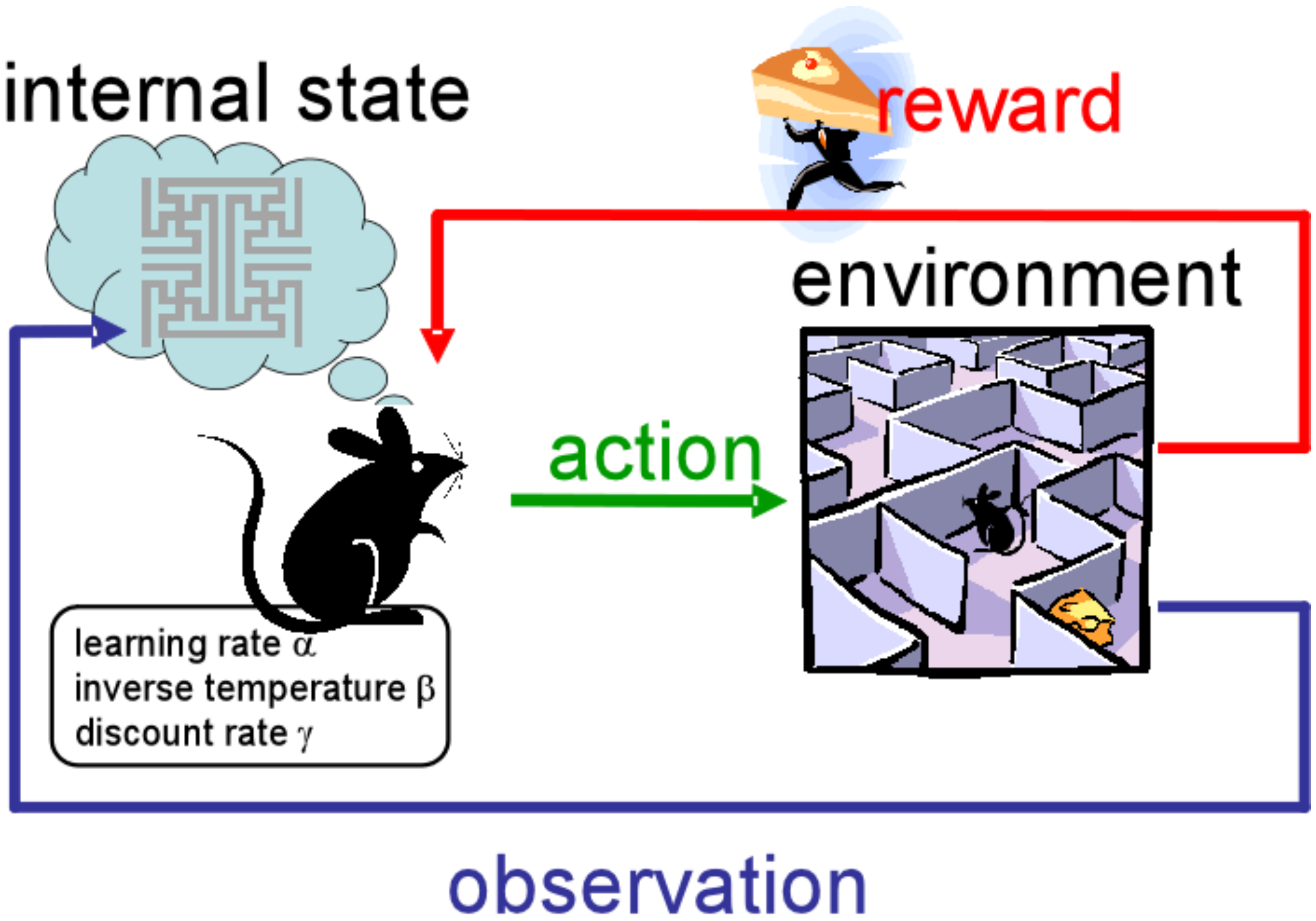
# Unsupervised Learning

- Find a *suitable data representation*
    - Preserving all task-relevant information
    - Simpler than the original data and easier to use



# Reinforcement Learning



# Data assumption

- IID (independent and identically distributed)

1) Come from the *same distribution*

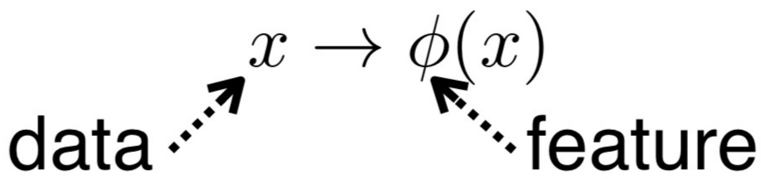
$$p_{x^{(i)}}(x) = p_{x^{(j)}}(x)$$

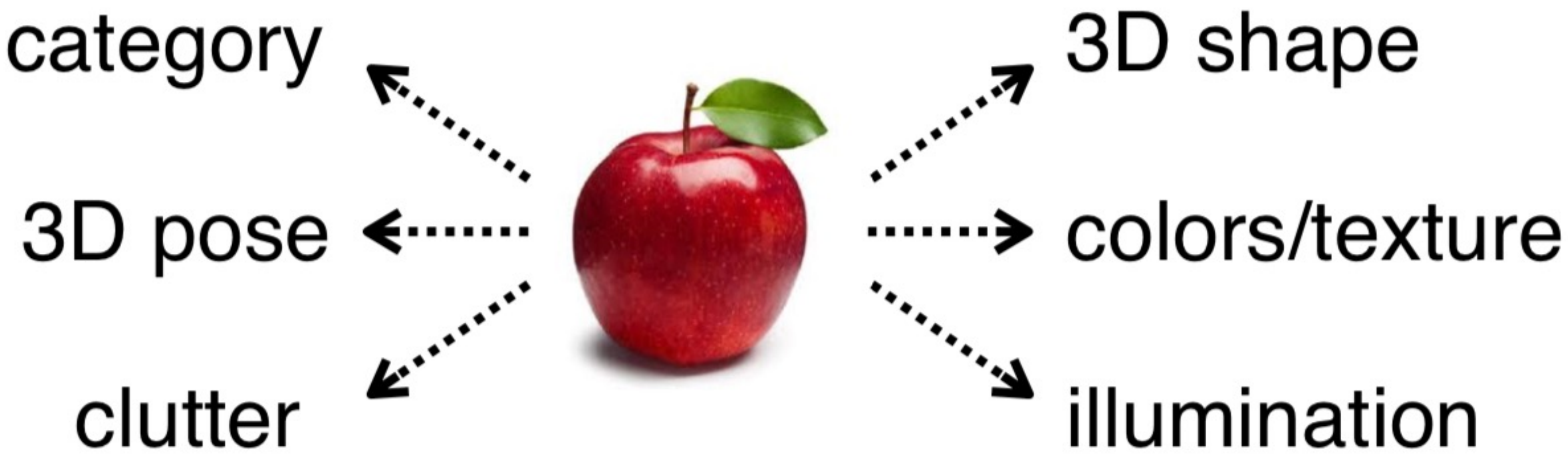
2) Are *independent*

$$p\left(x^{(1)}, \ldots, x^{(m)}\right) = \prod_{i=1}^{m} p\left(x^{(i)}\right)$$

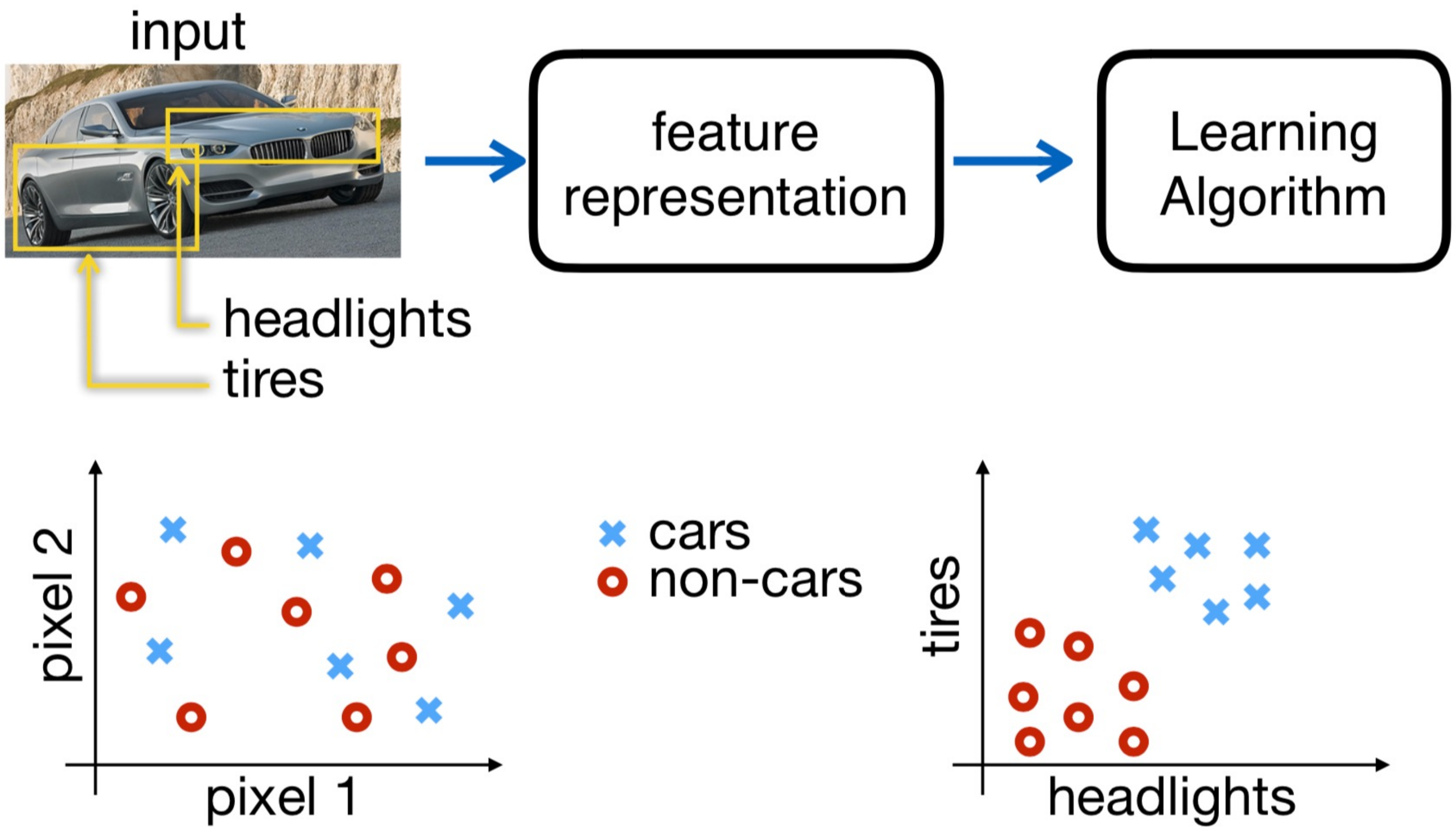
# Features

- Data often encoded into more focused relevant information (features or internal representation) to simplify the decision



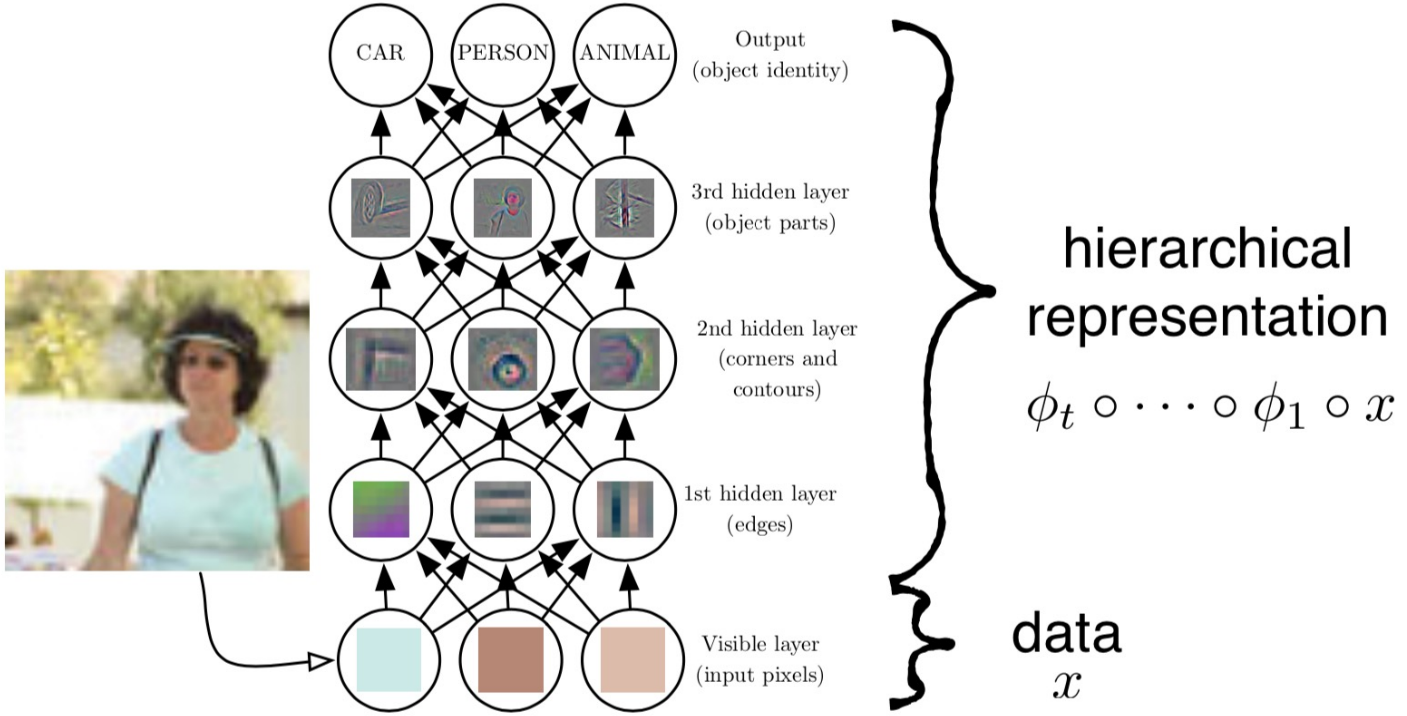



# Features Example :Image classification



# Deep Learning

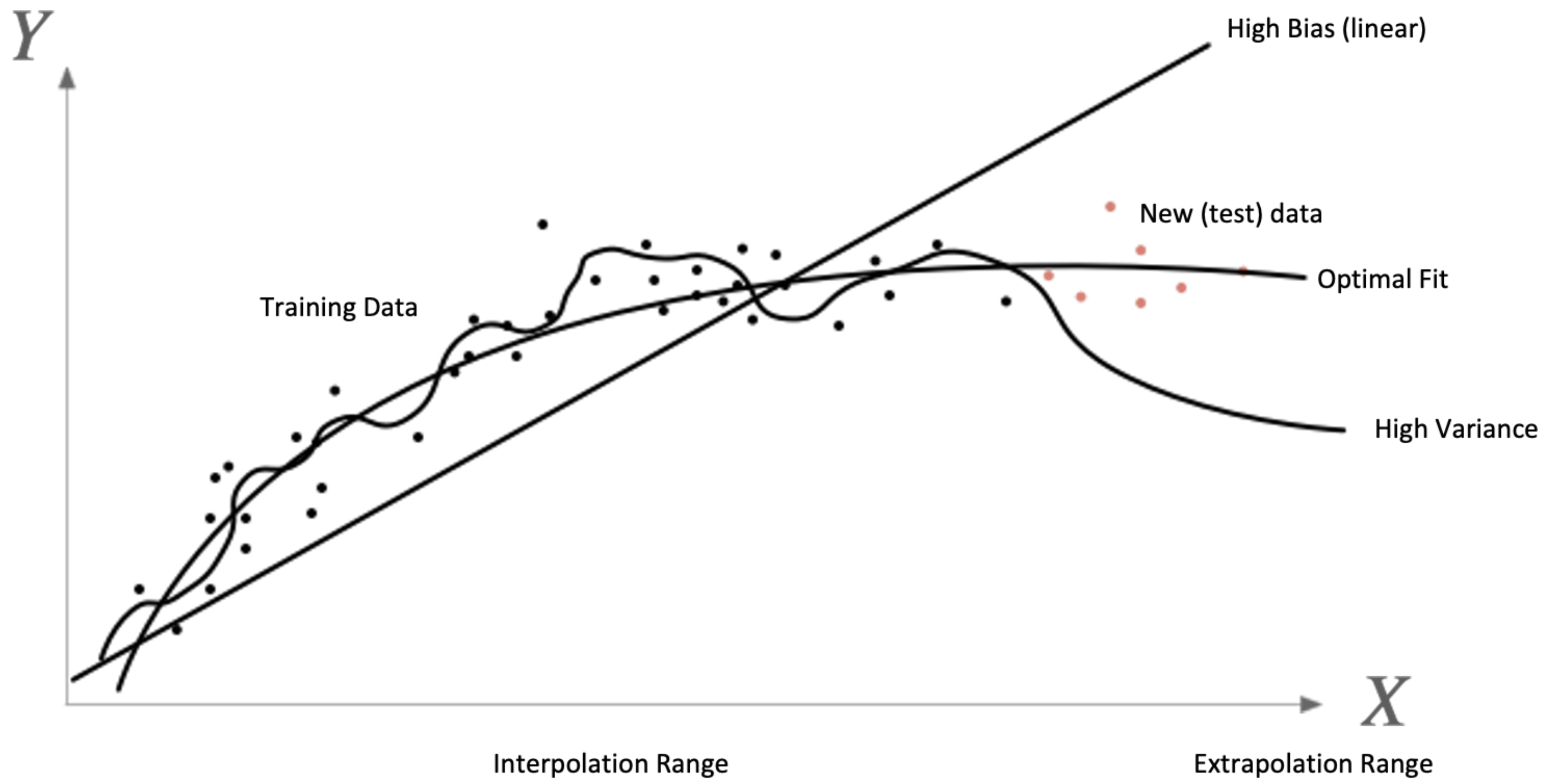
*« Build a machine that can learn from experience and understand the world as a hierarchy of concepts »*



# Training/Validation/Test sets

- Separate the data into 2(3) sets
  - Training set for training
  - Development / Validation set to find the best parameters
  - (Test set to estimate the performance)
- Separation depends on size of the dataset
- Make sure no algorithmic decisions are being made using data which are also being used to test the algorithm

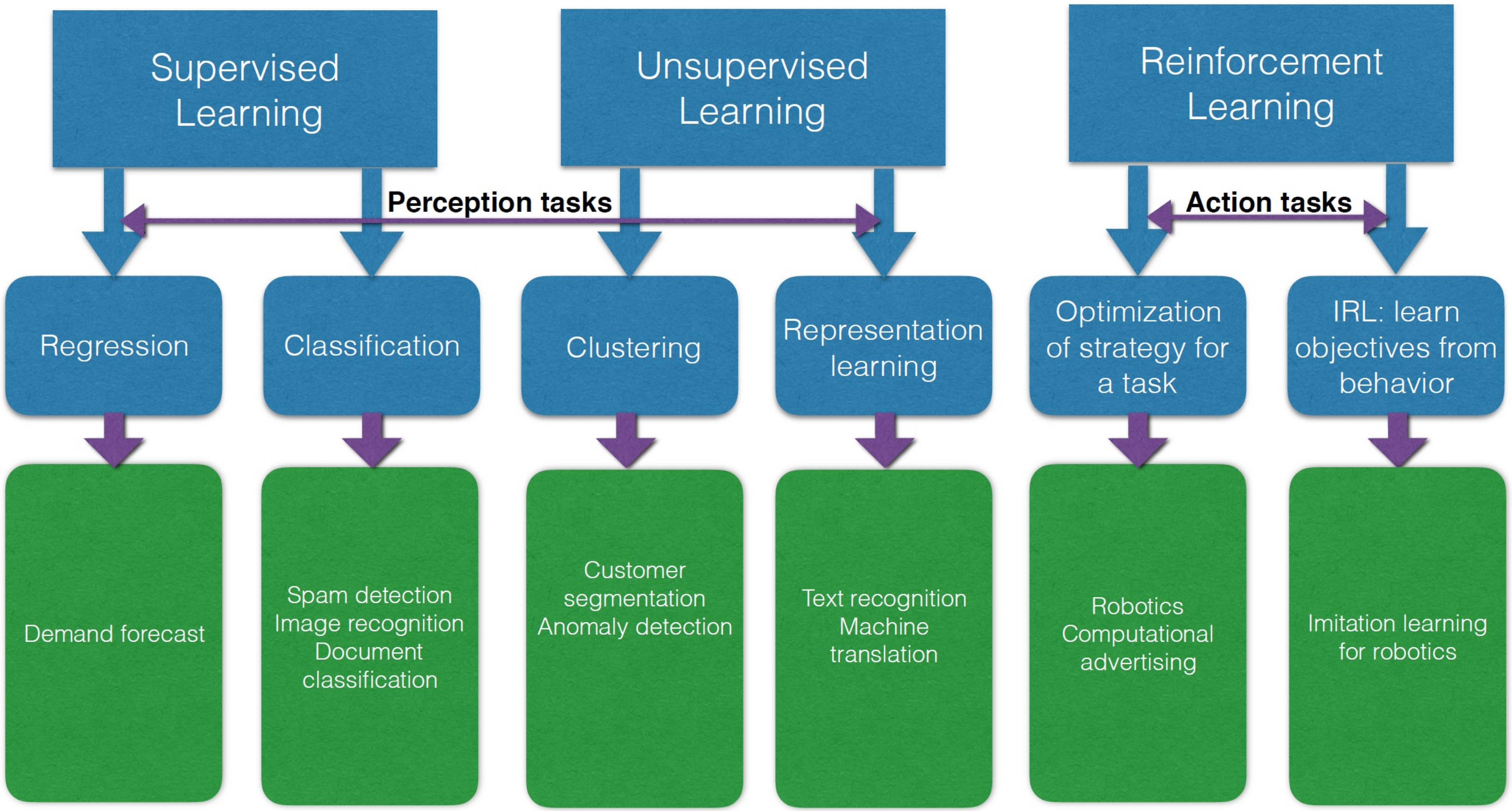
# Training/Validation/Test sets



# Task T

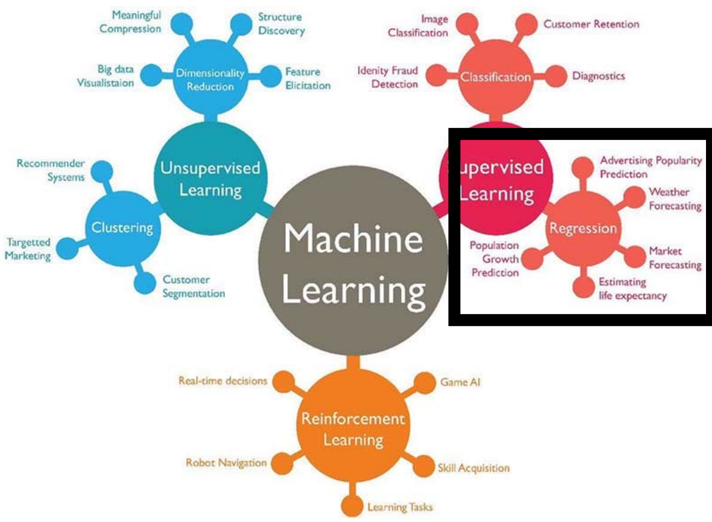
Find the function f that satisfies f(x) = y using the ***training set***

# Problem Types



# Regression

Predict results within *a continuous output*



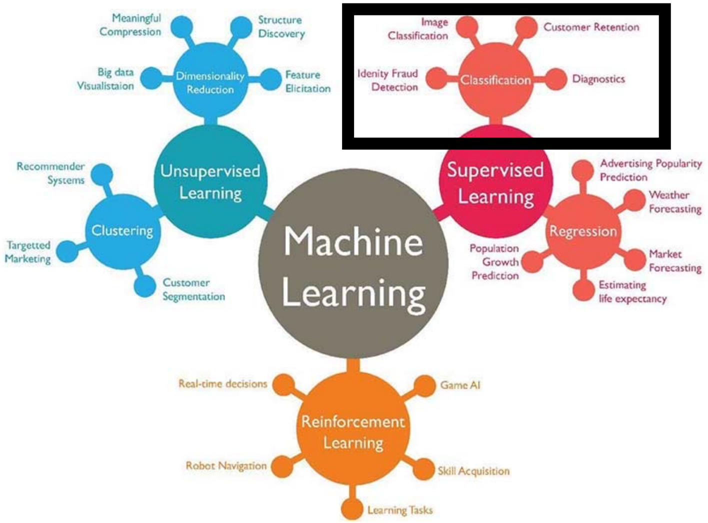

$82000

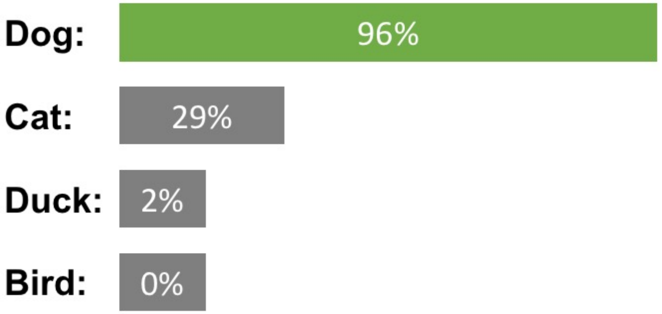
$55500

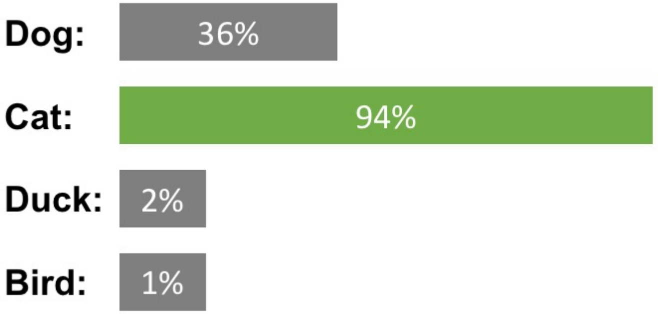
???

# Classification

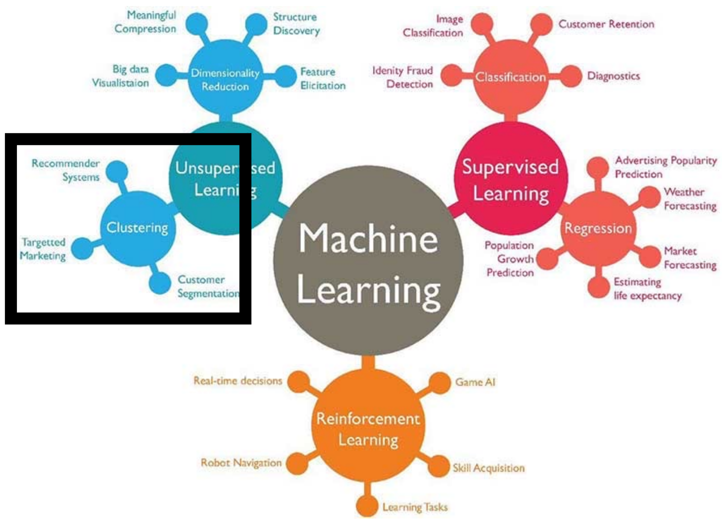
- categorize new inputs as belonging to one of a set of categories → Predict results within *a discrete output (categories)*



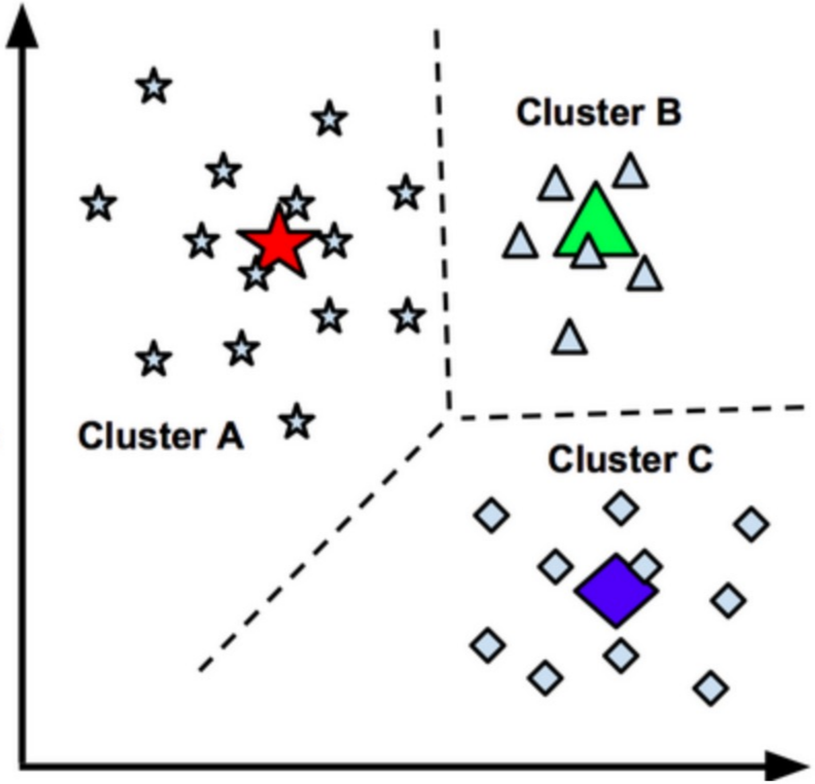



# Clustering

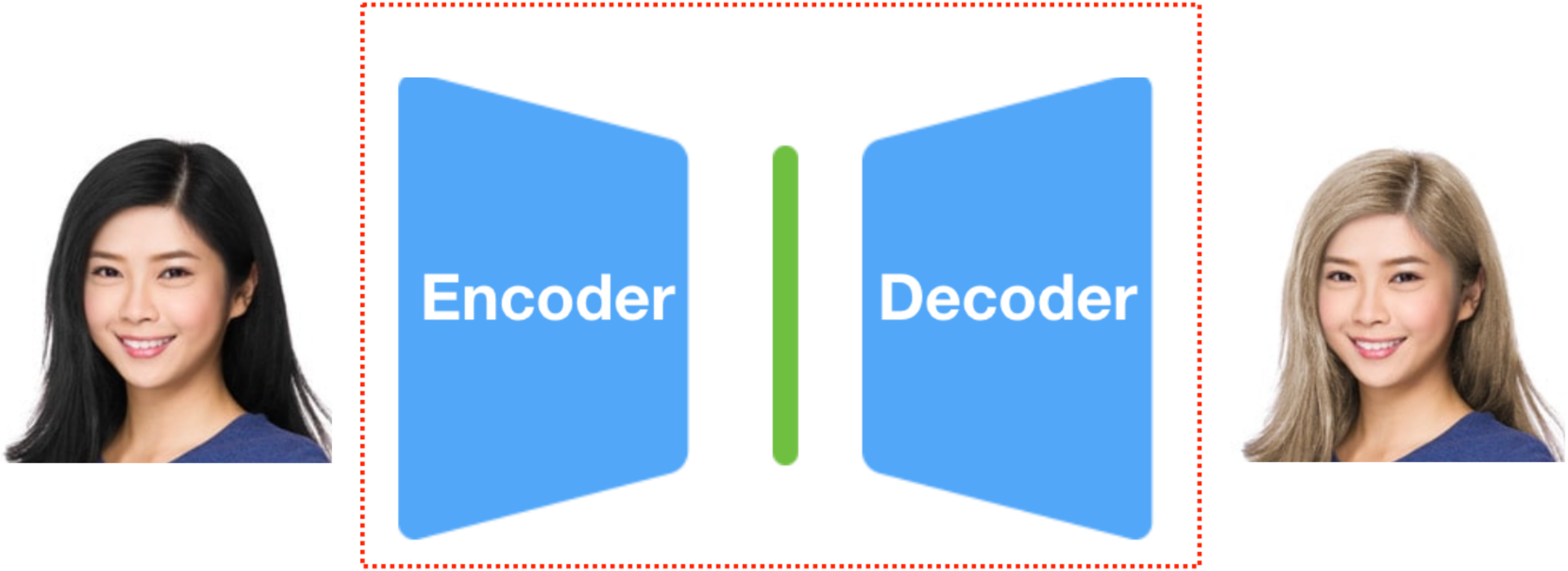



- create a set of categories, for which individual data instances have a set of common or similar characteristics.



# Data Generation

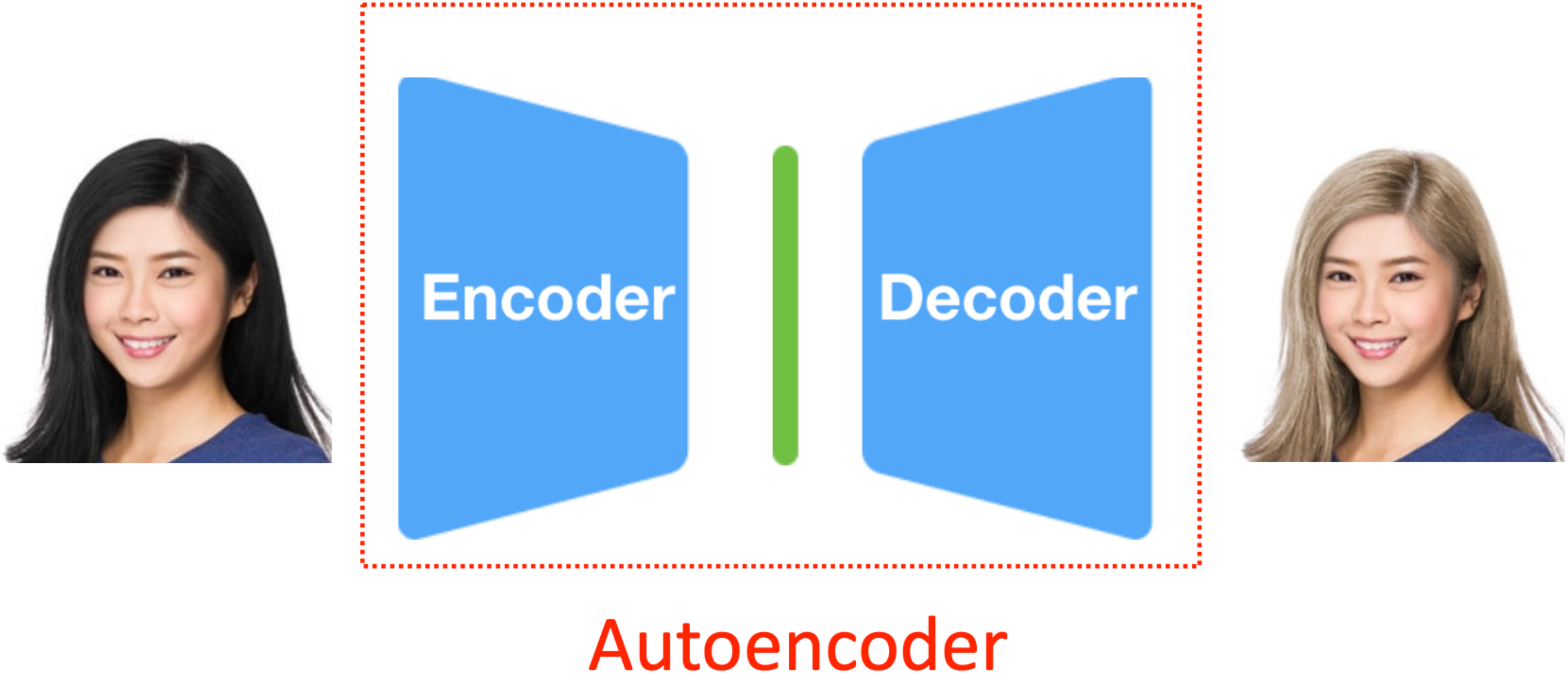
- generate appropriately novel data



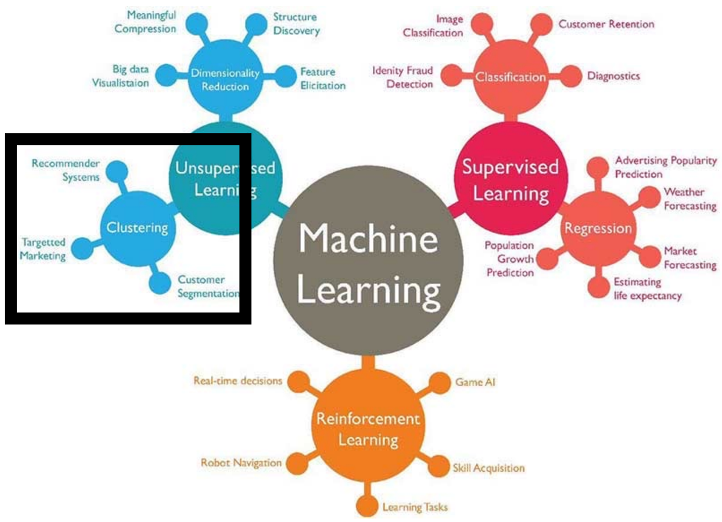

Autoencoder

# Data Generation

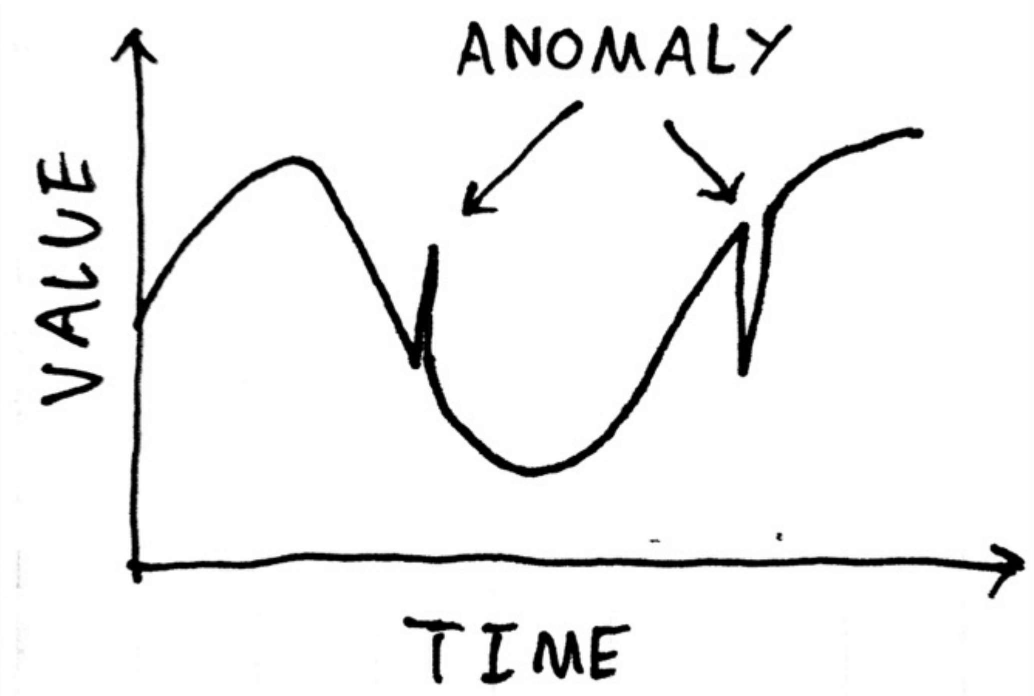
- generate appropriately novel data

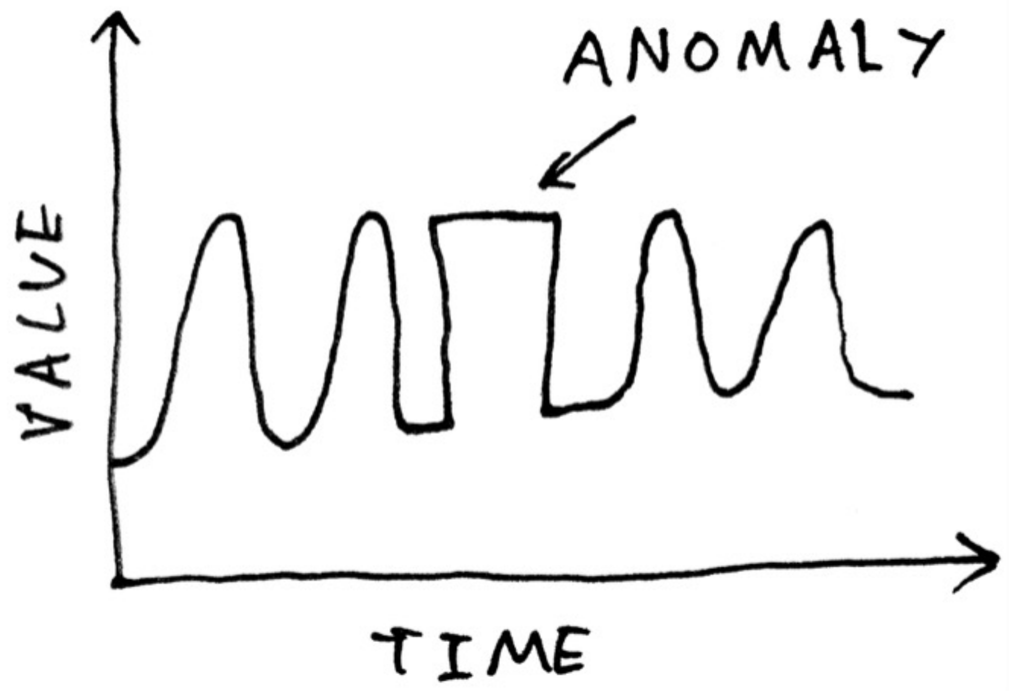


# Anomaly Detection

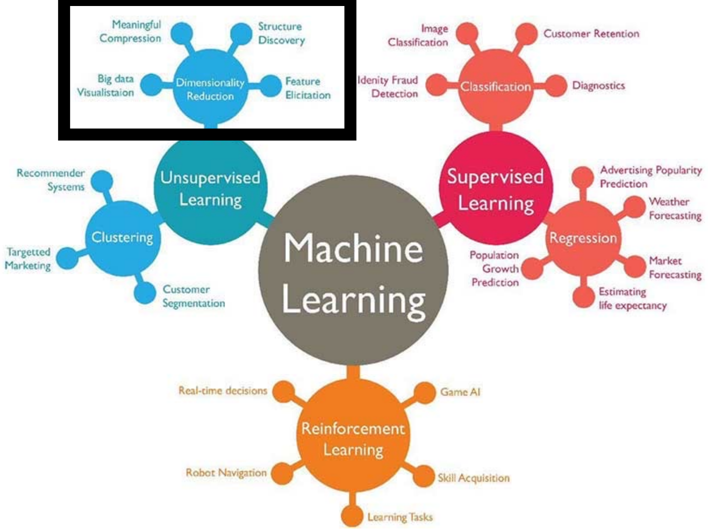



- determine whether specific inputs are out of the ordinary.



# Representation Learning



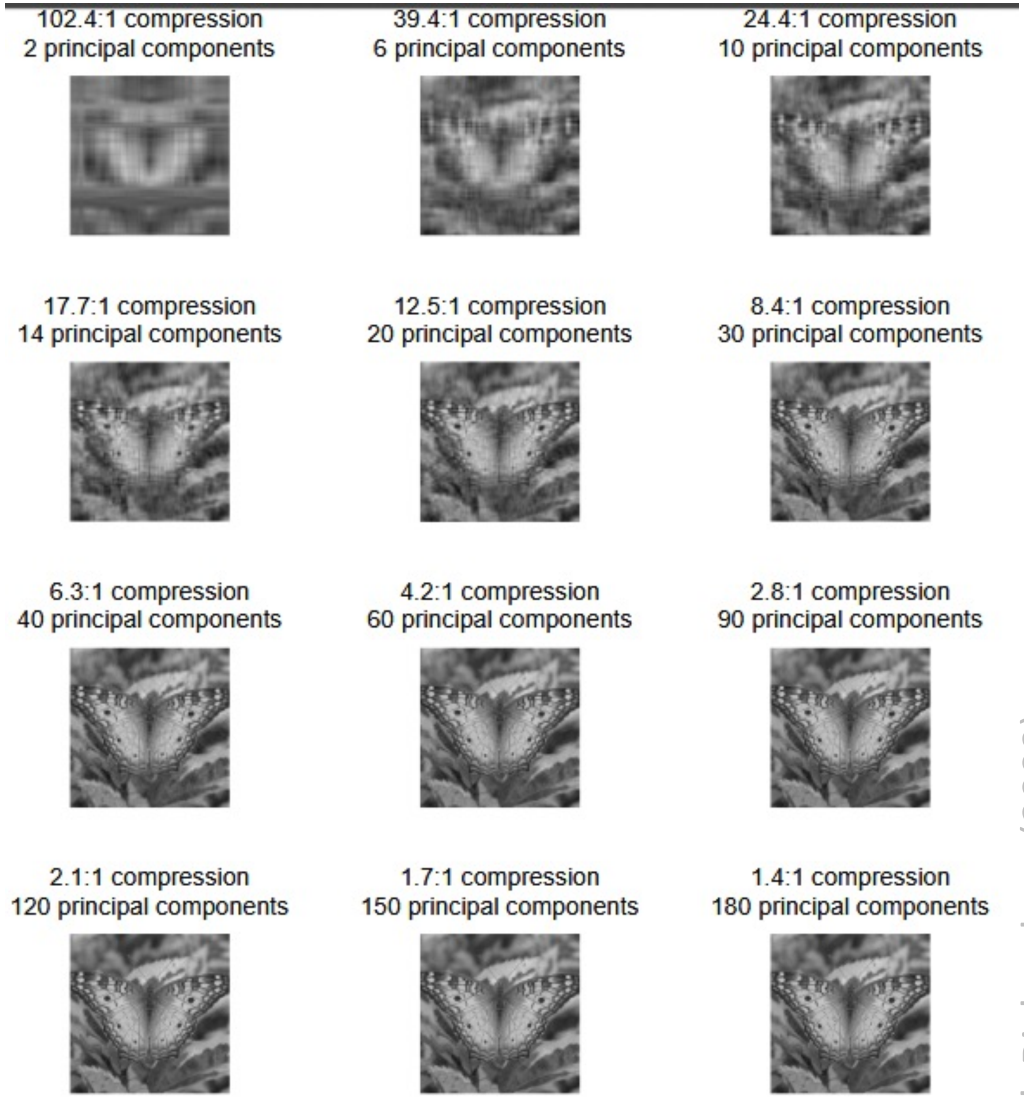
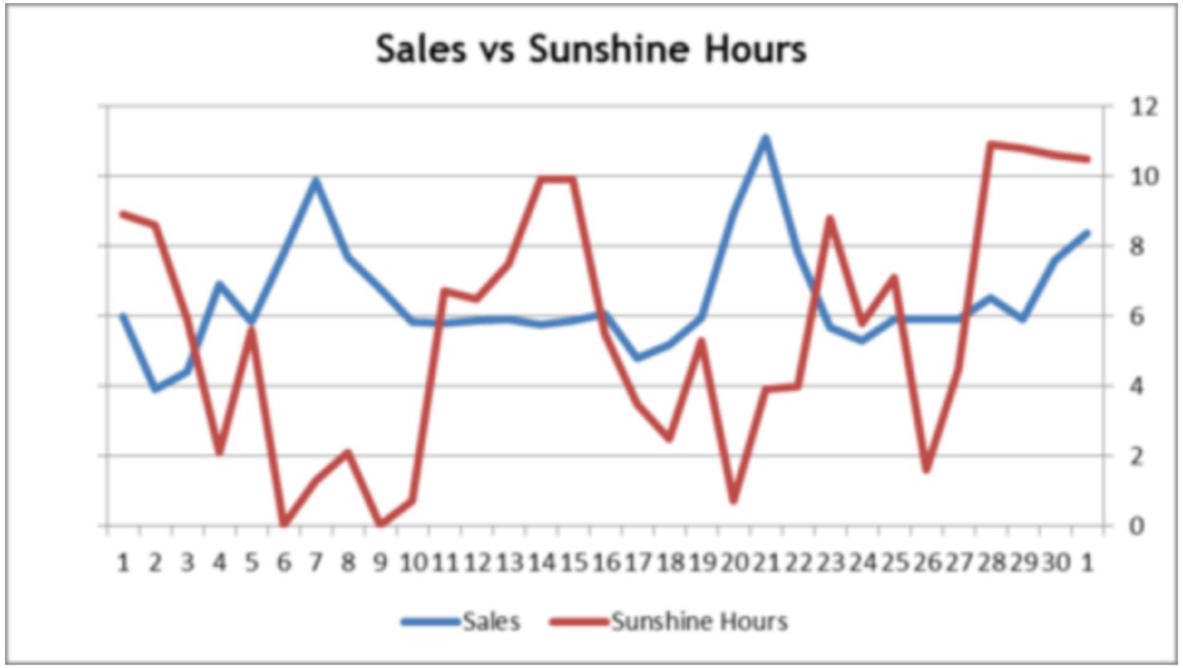




Figure 10: The visual effect of retaining principal components

Mark Richardson (2009)

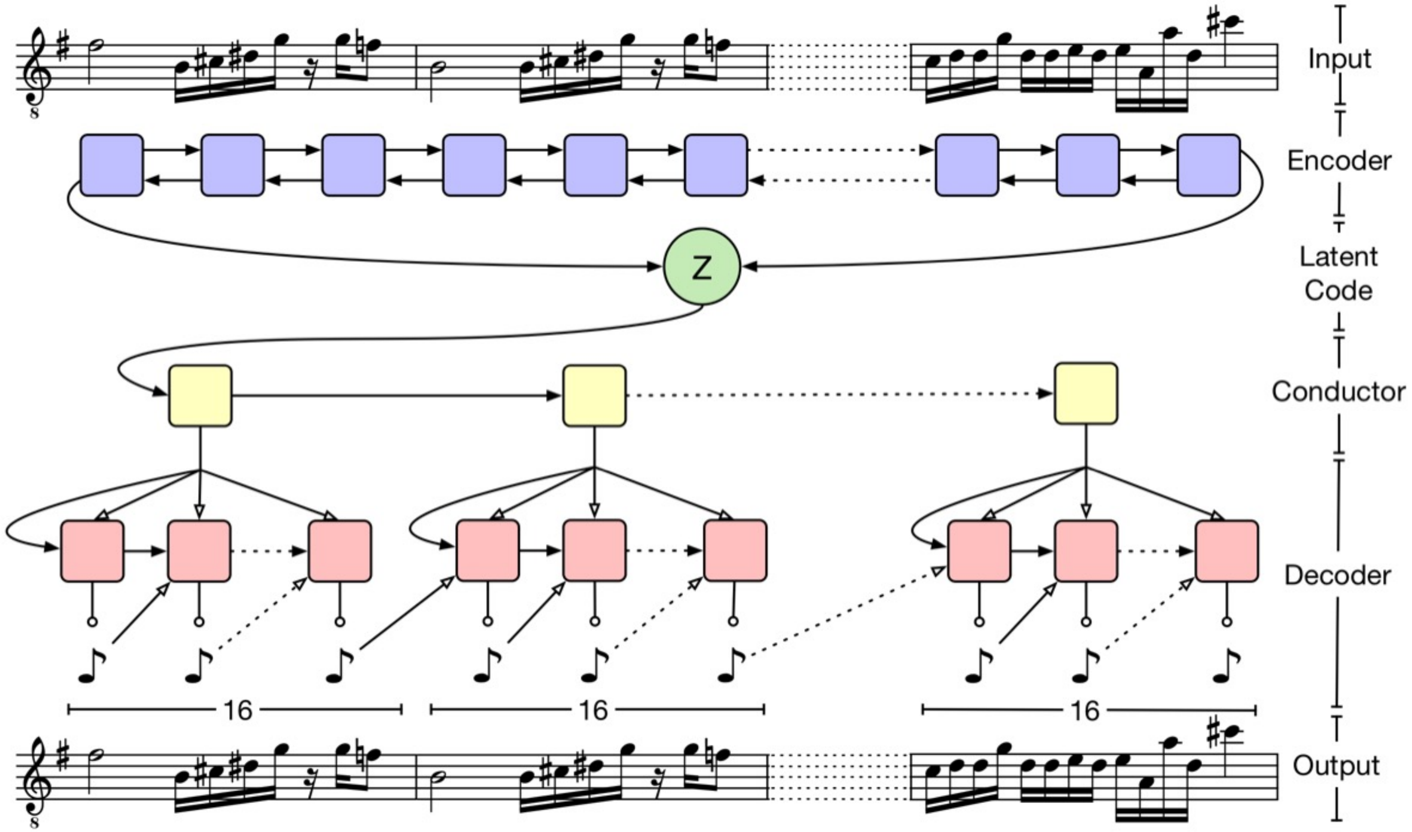
# Continuous Estimation

- estimate the next numeric value in a sequence (*prediction* for time series data)



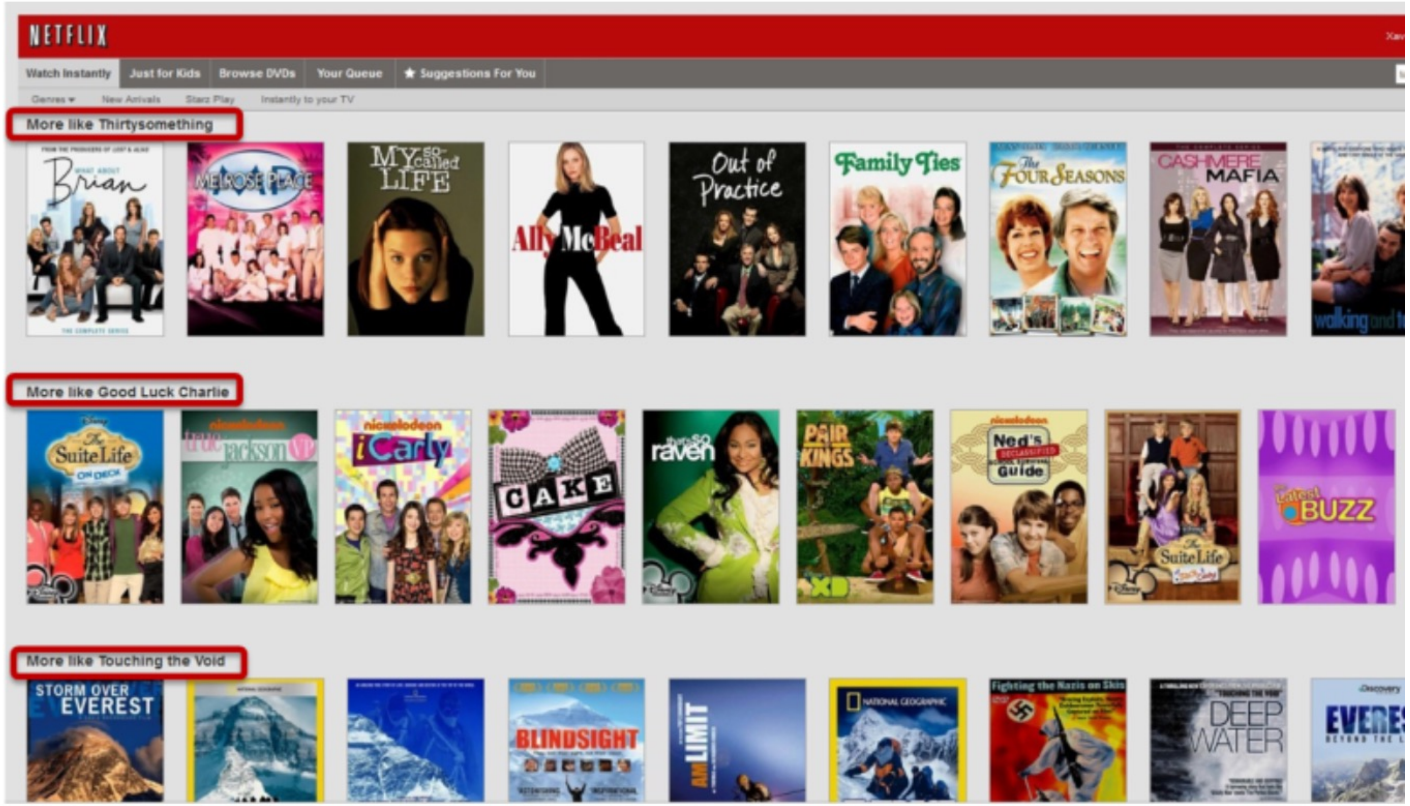
# Data Generation

- generate appropriately novel data



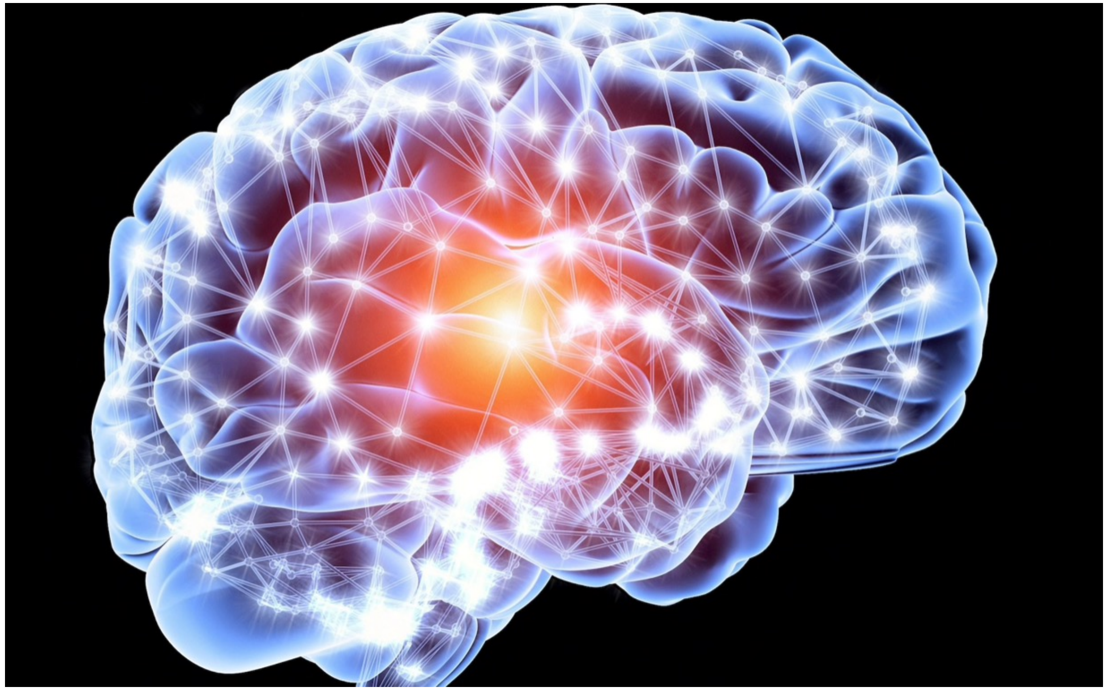
# Ranking

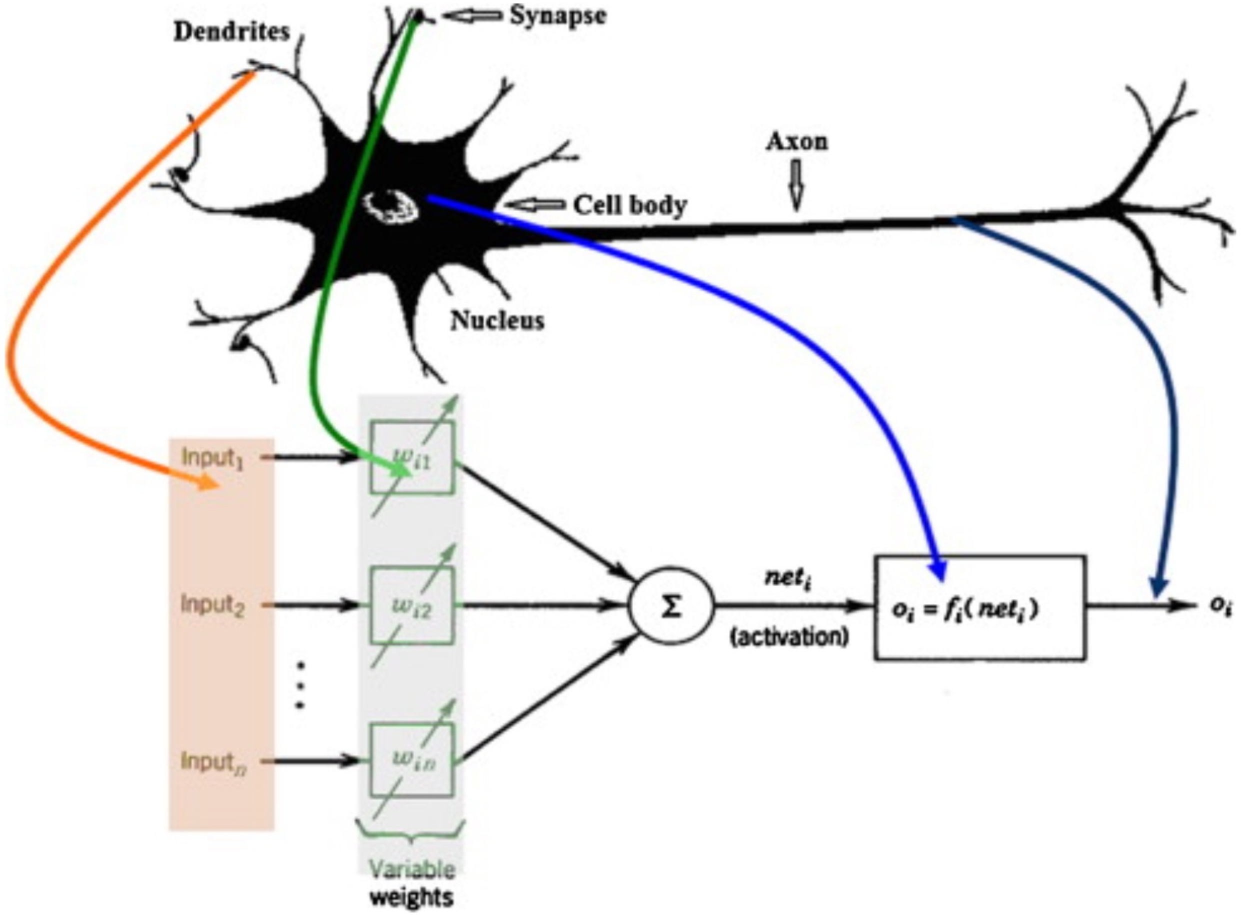
- Used in information retrieval problems
- Used in recommendation systems



# Neural Network (NN)

- Learning algorithm inspired by *how the brain works*





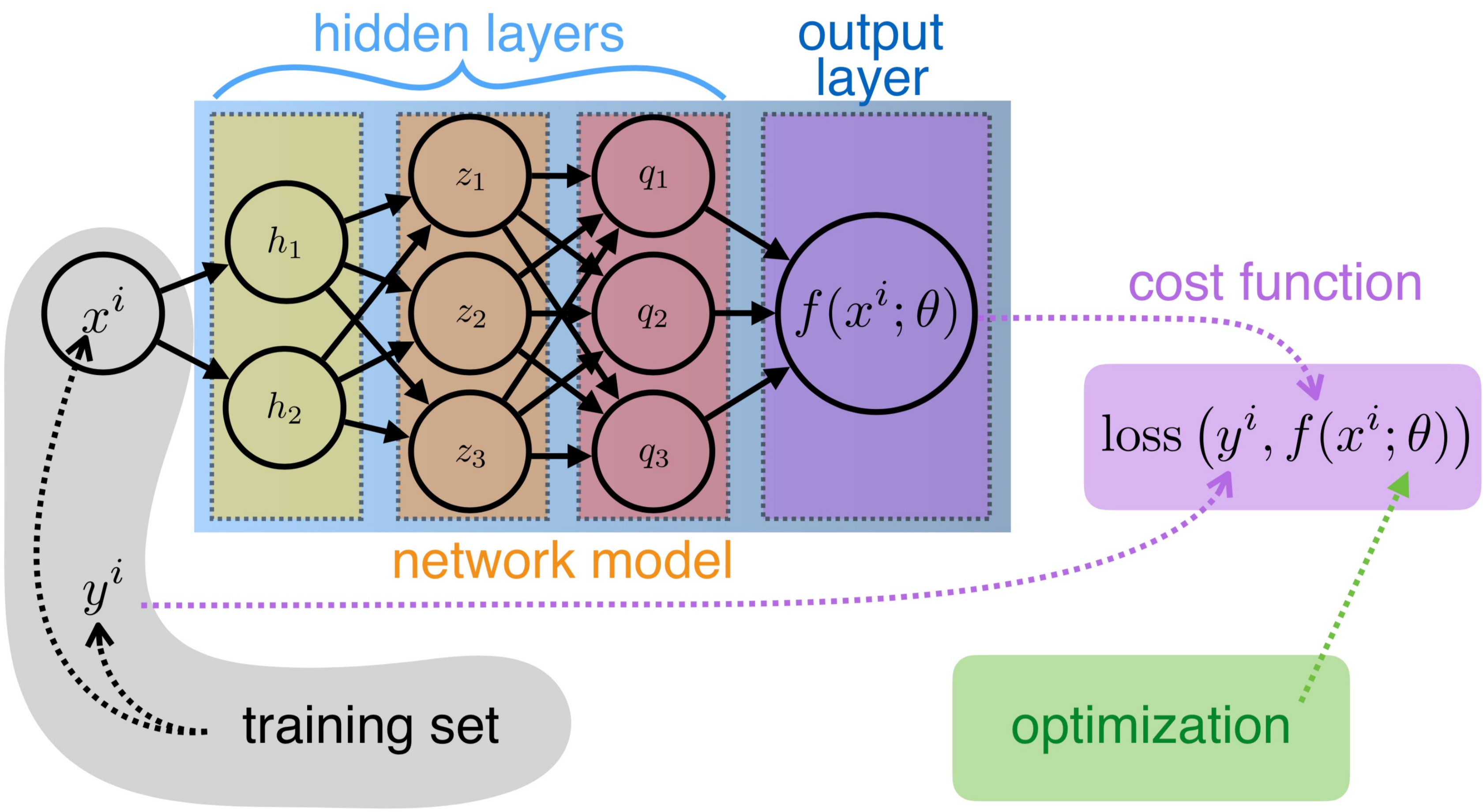
# Deploying a Neural Network

Given a task (in terms of **I/O mappings**), we need :

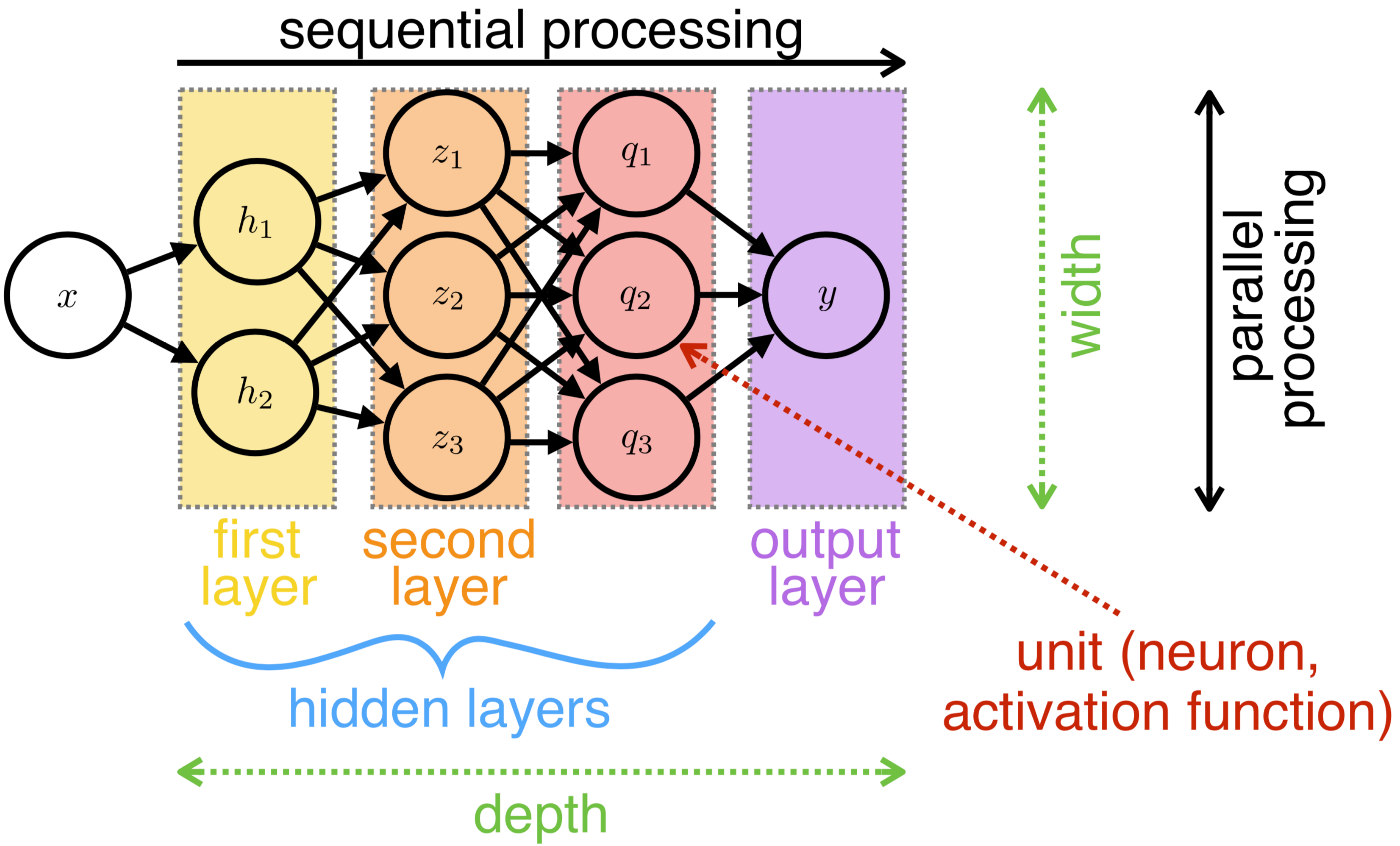
1) **Network model**

2) **Cost function (/objective/loss function)**

3) **Optimization**

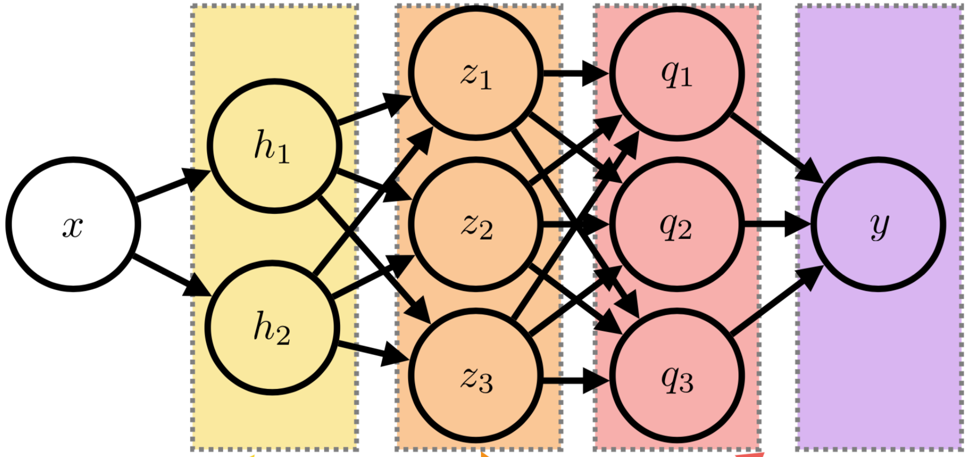


# Network model



# Activation functions

*Different types of activation functions for the hidden layers and the output layer*




$$y = f_4(q_1, q_2, q_3)$$

$$\boxed{y = f_4(f_{3,1}(f_{2,1}(f_{1,1}(x), f_{1,2}(x)), \ldots), \ldots)}$$

Hierarchical representation

$$h_1 = f_{1,1}(x)$$
$$h_2 = f_{1,2}(x)$$

$$z_1 = f_{2,1}(h_1, h_2)$$
$$z_2 = f_{2,2}(h_1, h_2)$$
$$z_3 = f_{2,3}(h_1, h_2)$$

$$q_1 = f_{3,1}(z_1, z_2, z_3)$$
$$q_2 = f_{3,2}(z_1, z_2, z_3)$$
$$q_3 = f_{3,3}(z_1, z_2, z_3)$$

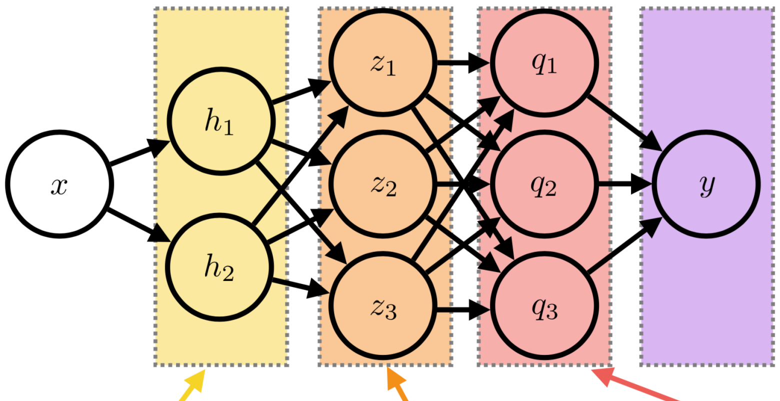
Fully connected

$$\boxed{f_{2,2}(h_1, h_2) = w_1 h_1 + w_2 h_2 + b_{2,2}}$$

Weights w and bias b parameters to optimize

# Activation functions

*Different types of activation functions for the hidden layers and the output layer*




$$y = f_4(q_1, q_2, q_3)$$

$$h_1 = f_{1,1}(x)$$
$$h_2 = f_{1,2}(x)$$

$$z_1 = f_{2,1}(h_1, h_2)$$
$$z_2 = f_{2,2}(h_1, h_2)$$
$$z_3 = f_{2,3}(h_1, h_2)$$

$$q_1 = f_{3,1}(z_1, z_2, z_3)$$
$$q_2 = f_{3,2}(z_1, z_2, z_3)$$
$$q_3 = f_{3,3}(z_1, z_2, z_3)$$

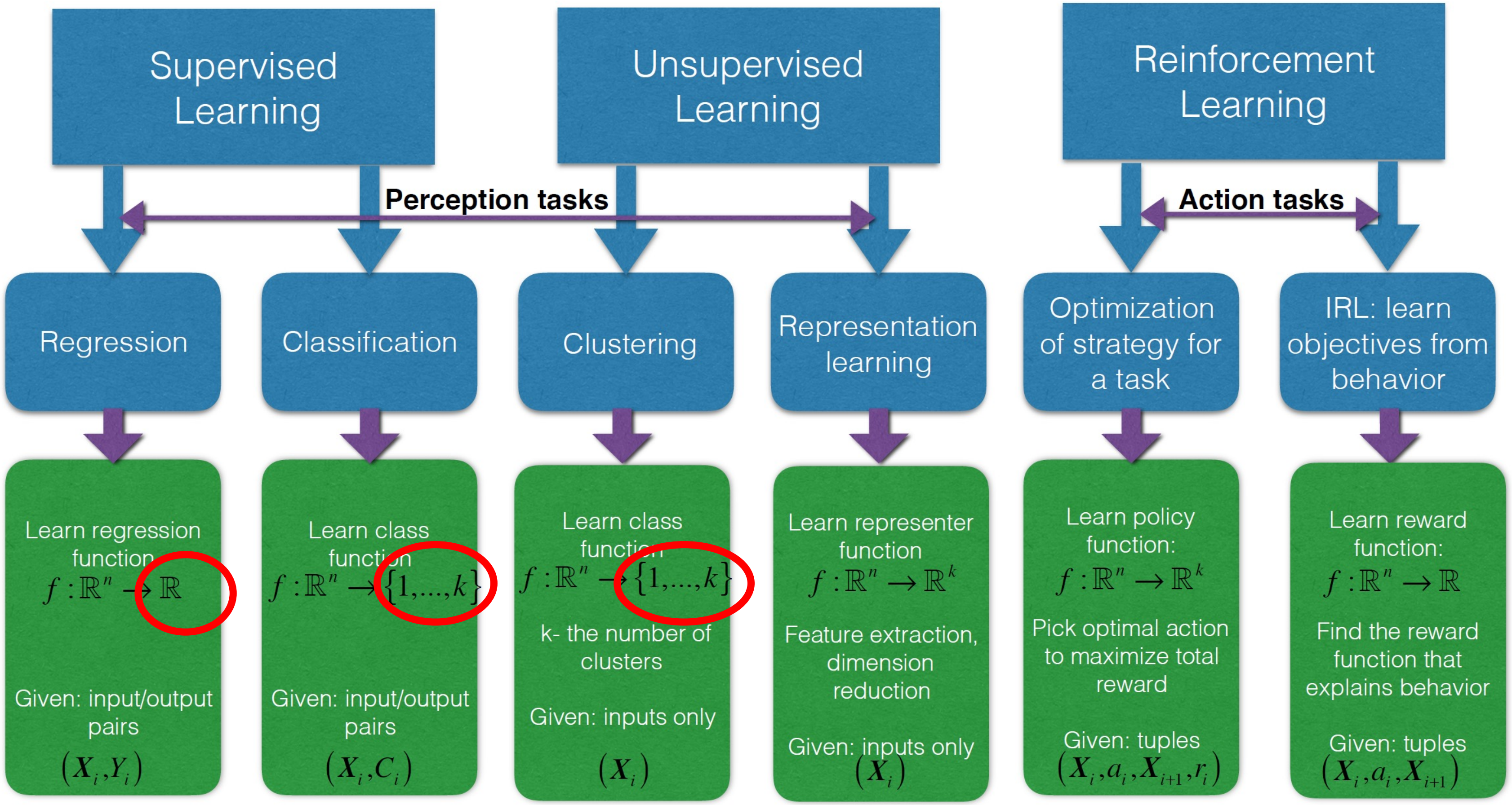
**(3,1)** **(3,<u>2</u>)** **(<u>2</u>,1)**

$$Z = W_z H$$

Fully connected

$$\boxed{f_{2,2}(h_1, h_2) = w_1 h_1 + w_2 h_2 + b_{2,2}}$$

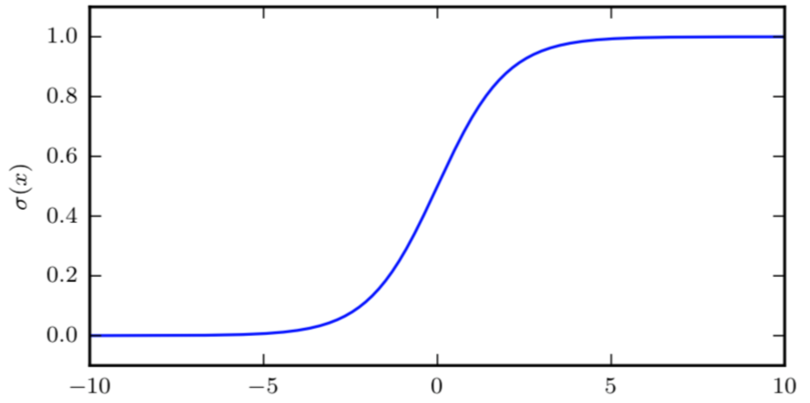
# Neural Network Outputs



# Output layer : activation functions

## 1) **Classification** : probability vector

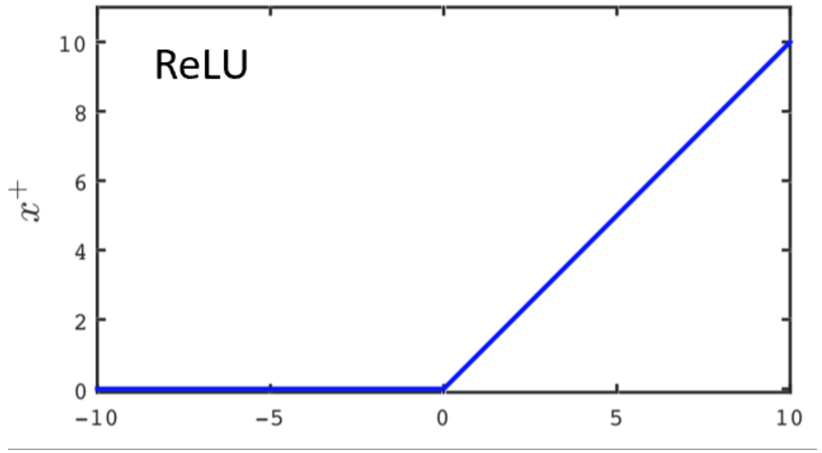
- Sigmoid (binary class)
- Softmax (multiple class)




$$Z = \sigma(W_z H)$$

## 2) **Regression** : mean estimate

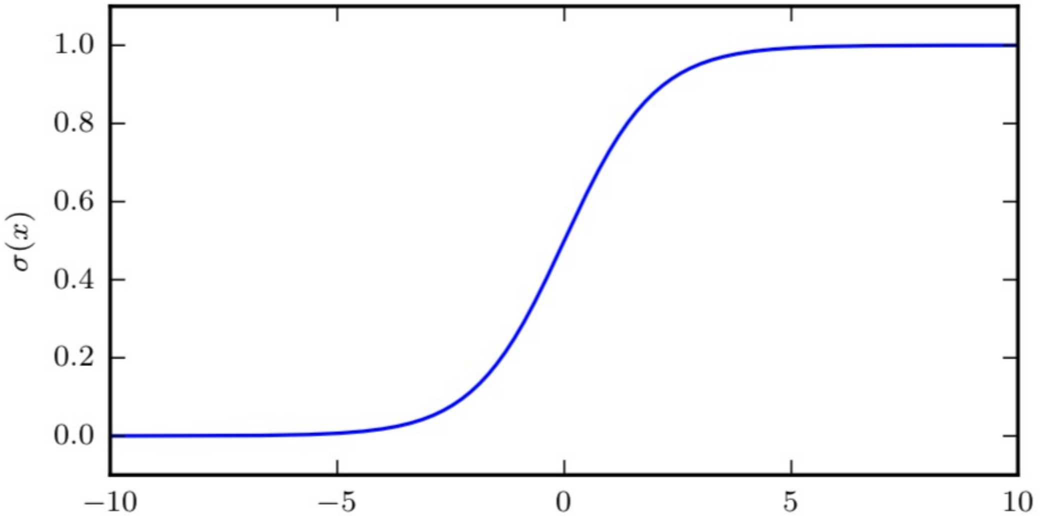
- ReLU
- Softplus
- Smoothed max
- Generalization of ReLU (leaky ReLU,…)



# Sigmoid and softmax

**Sigmoid** (*two-class* classifier) :

$$\boxed{\sigma(x) = \frac{1}{1+e^{-x}}}$$




**Softmax** (*multi-class* classifier) :

$$\boxed{\mathrm{softmax}(z)_i = \frac{\exp(z_i)}{\sum_j \exp(z_j)}}$$

# ReLU, softplus and smoothed max

**Softplus** (smooth approx. of ReLU) :

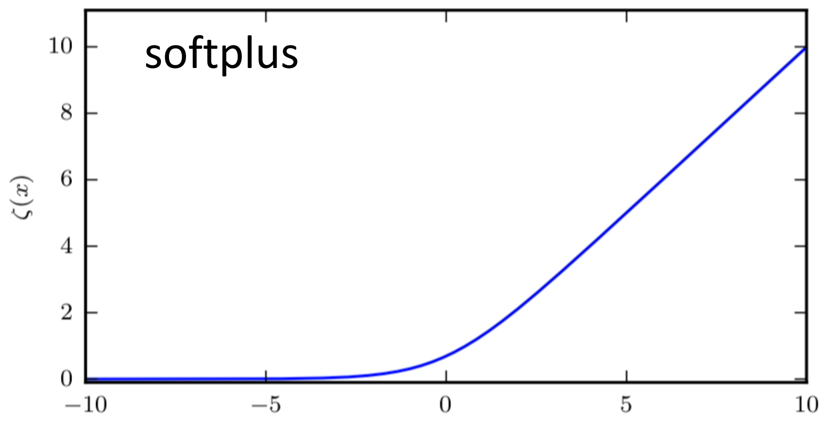
$$\zeta(x) = \log(1 + \exp(x))$$

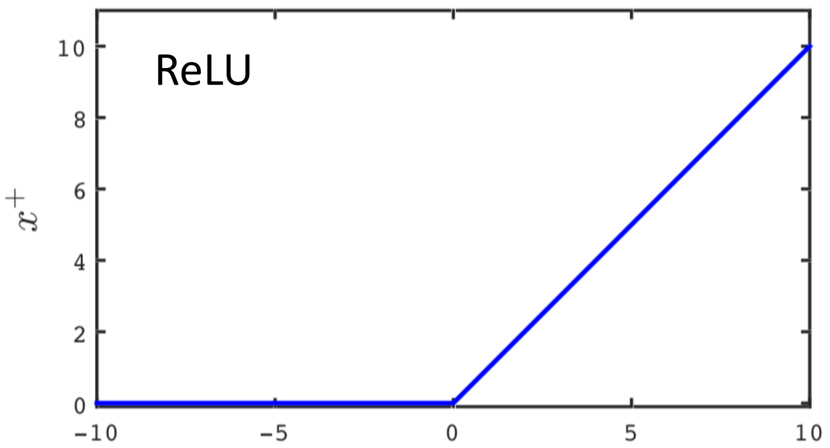
**Smoothed max** (*extension* of softplus) :

$$\zeta(x) = \log \sum_j \exp(x_i)$$

**ReLU** (Rectified Linear Unit) :

$$x^+ = \max(0, x)$$



# Let’s start playing !

Tutorial 1: 9h-9h45

Introduction to Tensorflow