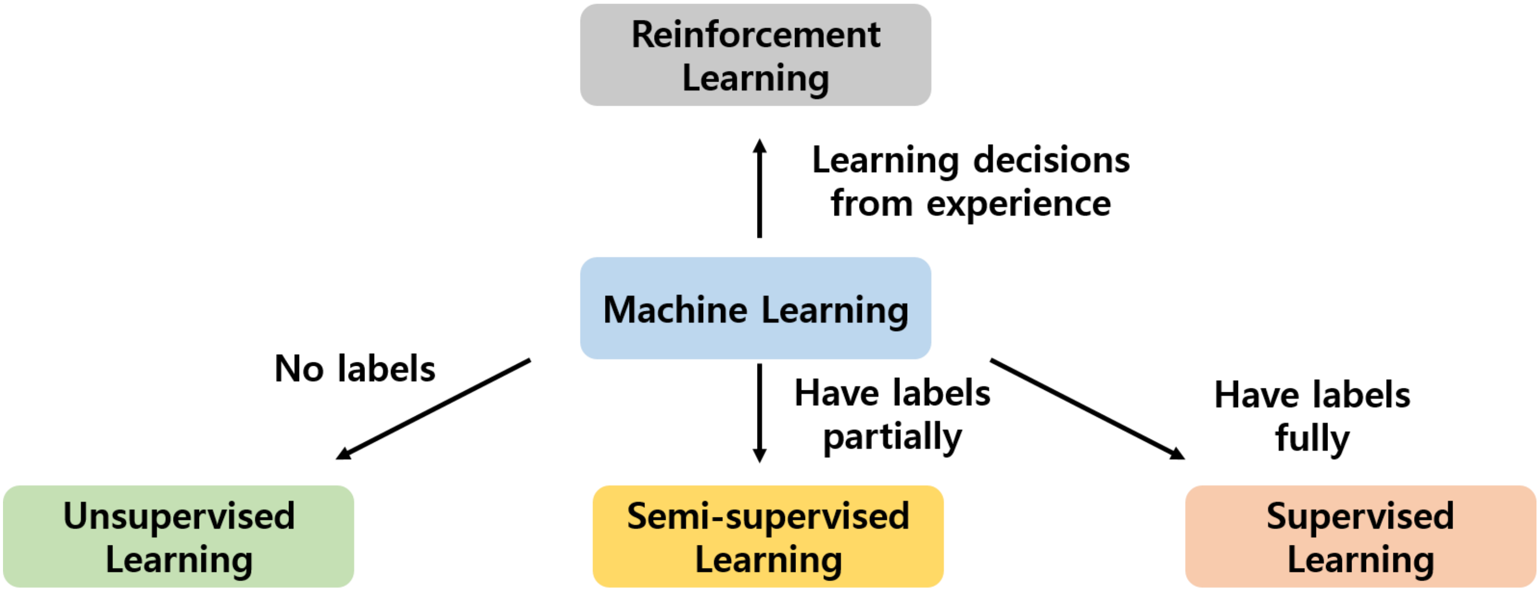
Learning algorithms

Seongok Ryu
KAIST Chemistry

Contents

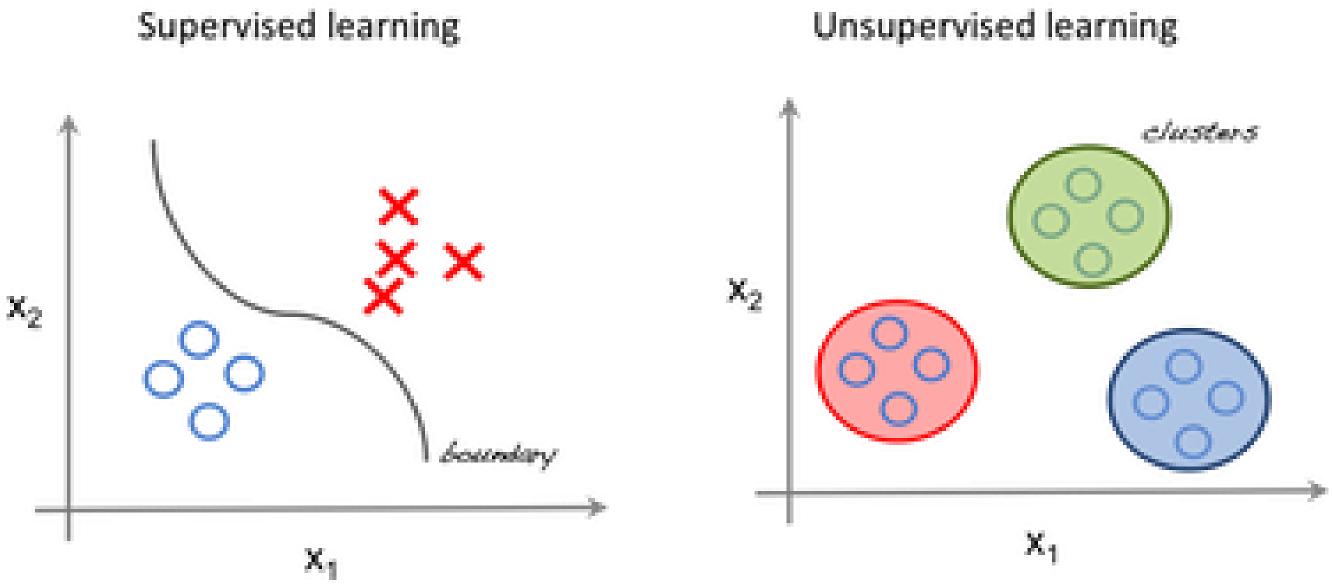
- A survey on data collection for machine learning
- Supervised learning
- Unsupervised learning
- Beyond supervised learning: label-efficient algorithms

Brief illustration of learning algorithms



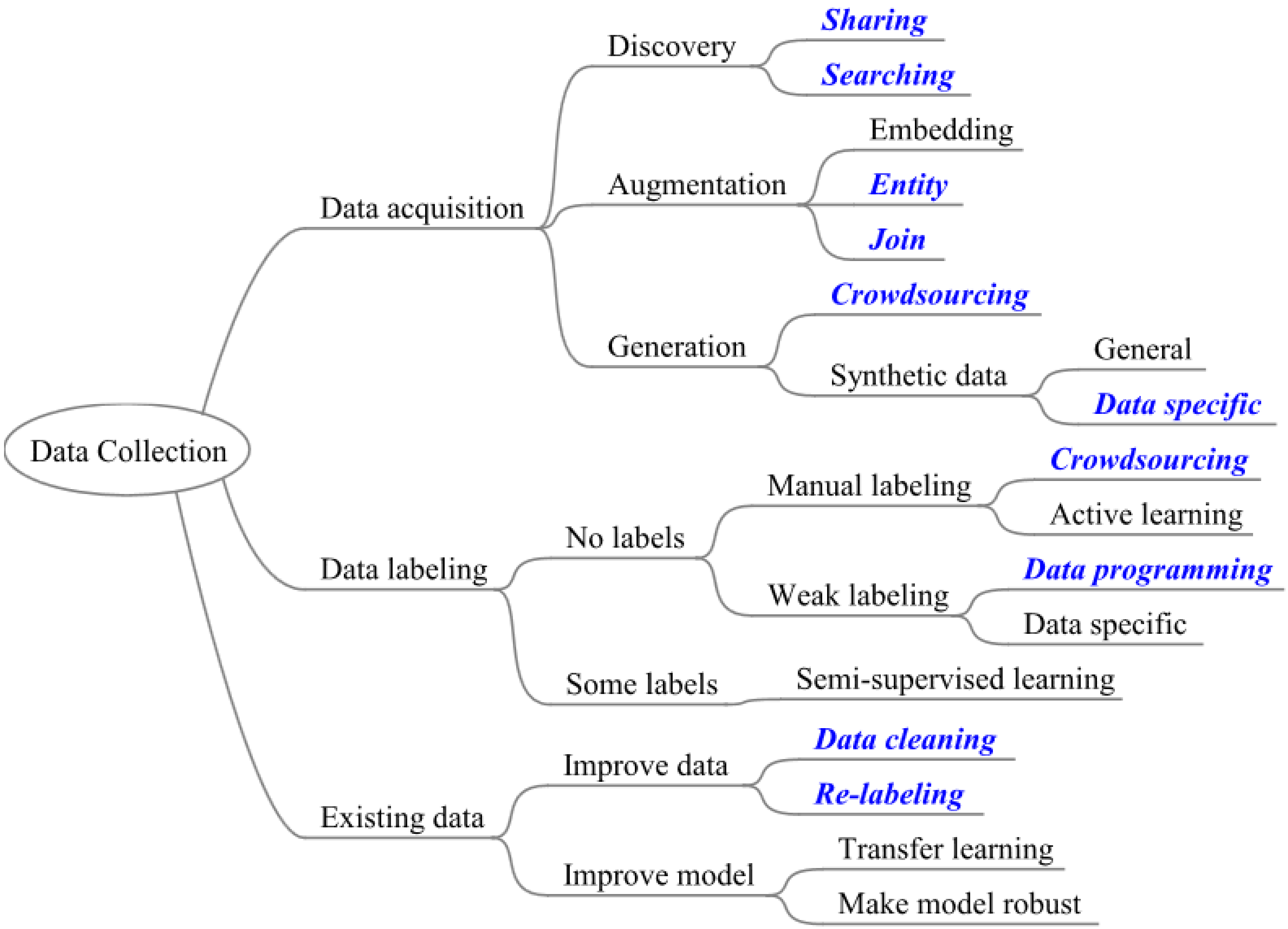
- Reinforcement learning: to learn a decision (policy) from experience by interacting between an environment
- Supervised learning: to learn a hypothesis that best explains the input-label relationship
- Unsupervised learning: to learn features, which are also called as representations or latent variables
- Semi-supervised learning: obtaining a hypothesis by using both labeled and unlabeled data

Brief illustration of learning algorithms



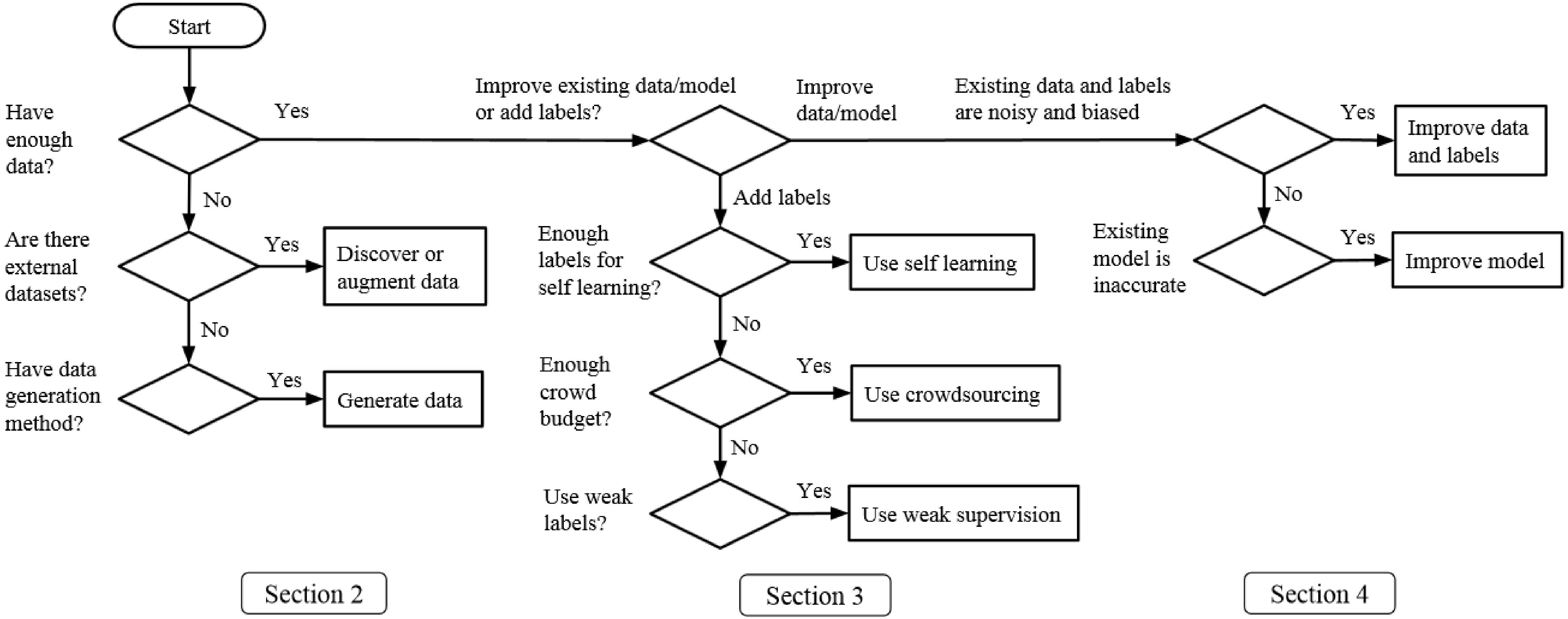
- Reinforcement learning: to learn a decision (policy) from experience by interacting between an environment
- Supervised learning: to learn a hypothesis that best explains the input-label relationship
- Unsupervised learning: to learn features, which are also called as representations or latent variables
- Semi-supervised learning: obtaining a hypothesis by using both labeled and unlabeled data

A survey on data collection for machine learning



Roh, Yuji, Geon Heo, and Steven Euijong Whang.
"A Survey on Data Collection for Machine Learning: a Big Data-AI Integration Perspective."

A survey on data collection for machine learning



Roh, Yuji, Geon Heo, and Steven Euijong Whang.
"A Survey on Data Collection for Machine Learning: a Big Data-AI Integration Perspective."

A survey on data collection for machine learning

Public chemistry database

- For drug discovery

PubChem

ChEMBL

ZINC

DUD•E

A Database of Useful Decoys: Enhanced

- For quantum chemistry

  QM7/8/9/… @ http://quantum-machine.org/datasets/

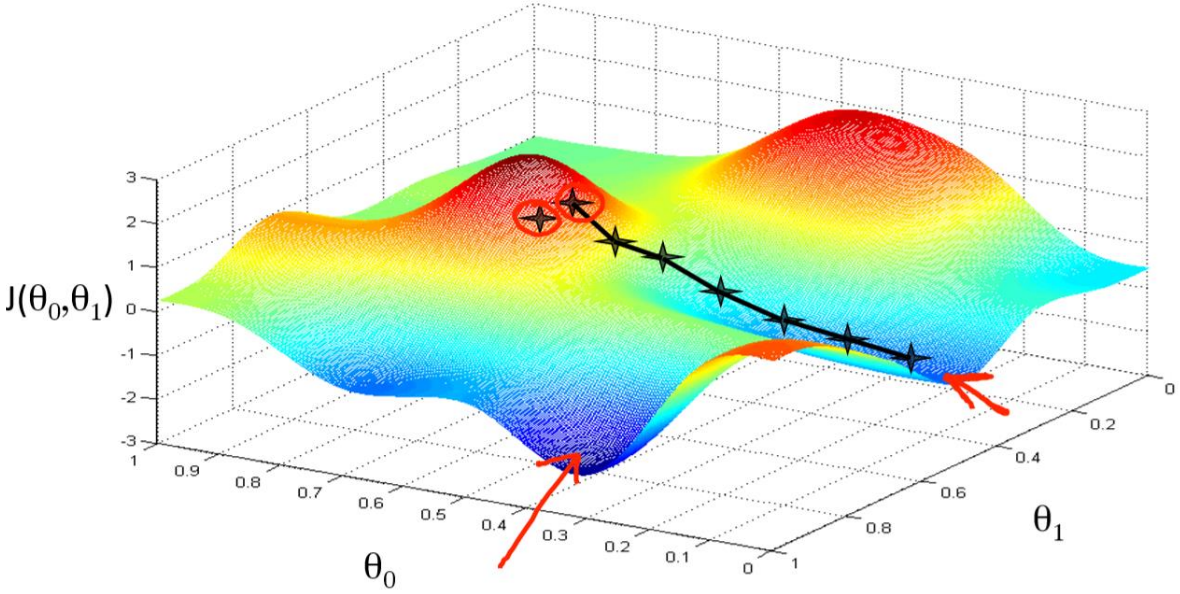
- Please find and note more databases ☺

Supervised learning

- The goal of supervised learning is to find a hypothesis $h: X \rightarrow Y$ that maps given inputs X to labels Y.
- Following ERM principle, the risk of hypothesis h is defined as the expectation of loss $L(h(x_i), y_i)$ for (empirical) training samples (x_i, y_i), where $X = (x_1, \dots, x_n)$ and $Y = (y_1, \dots, y_n)$

$$R_{emp}(h) = \frac{1}{n}\sum_{i=1}^{n} L(h(x_i), y_i)$$

- Minimizing the empirical risk can be solved by applying optimization algorithms.



Supervised learning

- When h is given by a conditional probability distribution $p(y|x)$ and the loss function is the negative log-likelihood, $L(\hat{y}, y) \coloneqq -\log p(y|x)$, then ERM becomes equivalent to maximum likelihood estimation.
- For regression tasks, ones usually assumes the form of likelihood as $p(y|x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{|h(x)-y|^2}{2\sigma^2}\right)$.

  Then, our minimization objective is given by the L2-norm:

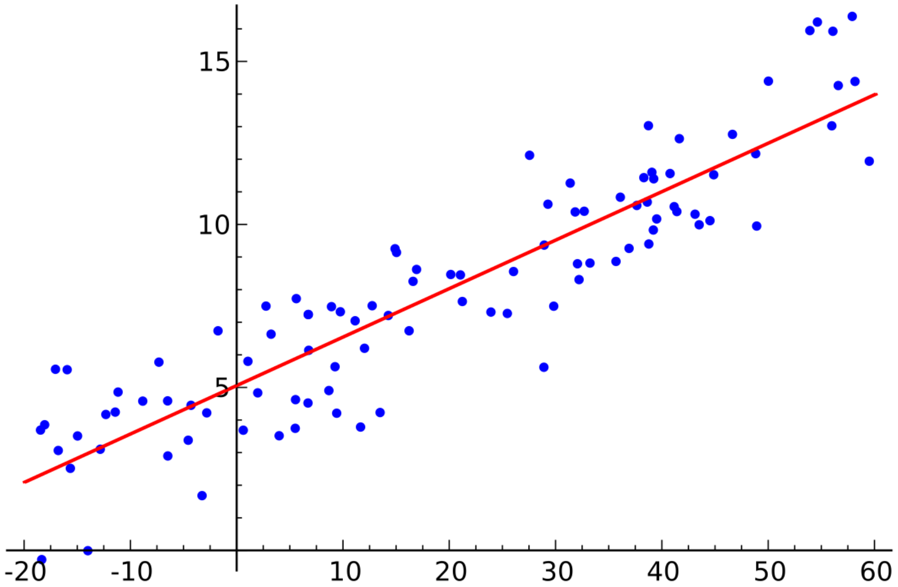
$$R_{emp}(h) = \frac{1}{n}\sum_{i=1}^{n} |h(x_i) - y_i|^2$$

- For classification tasks, we usually let our minimization objective as a cross-entropy loss:

$$R_{emp}(h) = -\frac{1}{n}\sum_{i=1}^{n} y_i \log h(x_i) + (1 - y_i)\log(1 - h(x_i))$$

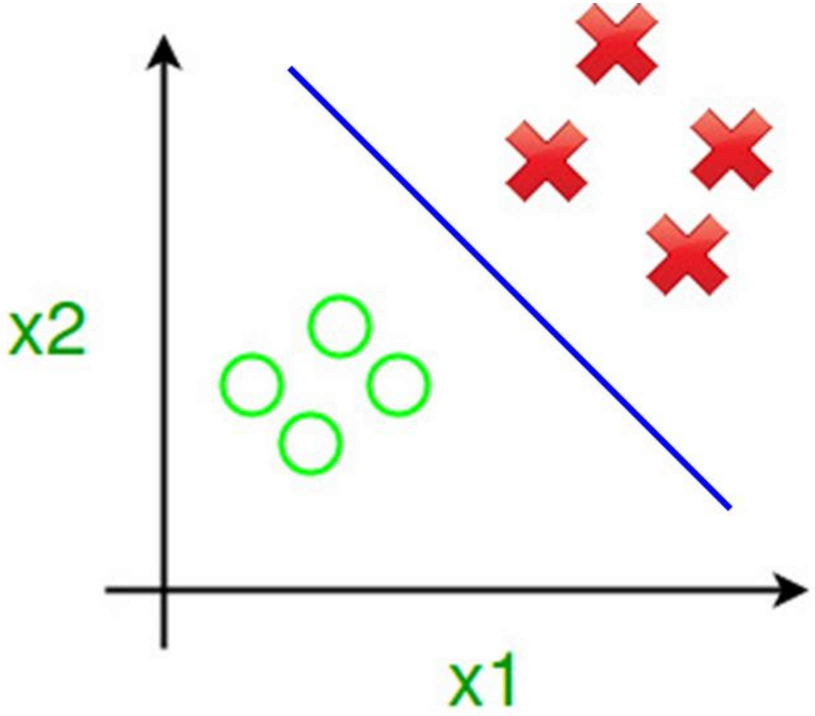
Supervised learning

Regression tasks



$$R_{emp}(h) = \frac{1}{n}\sum_{i=1}^{n} |h(x_i) - y_i|^2$$

Classification tasks



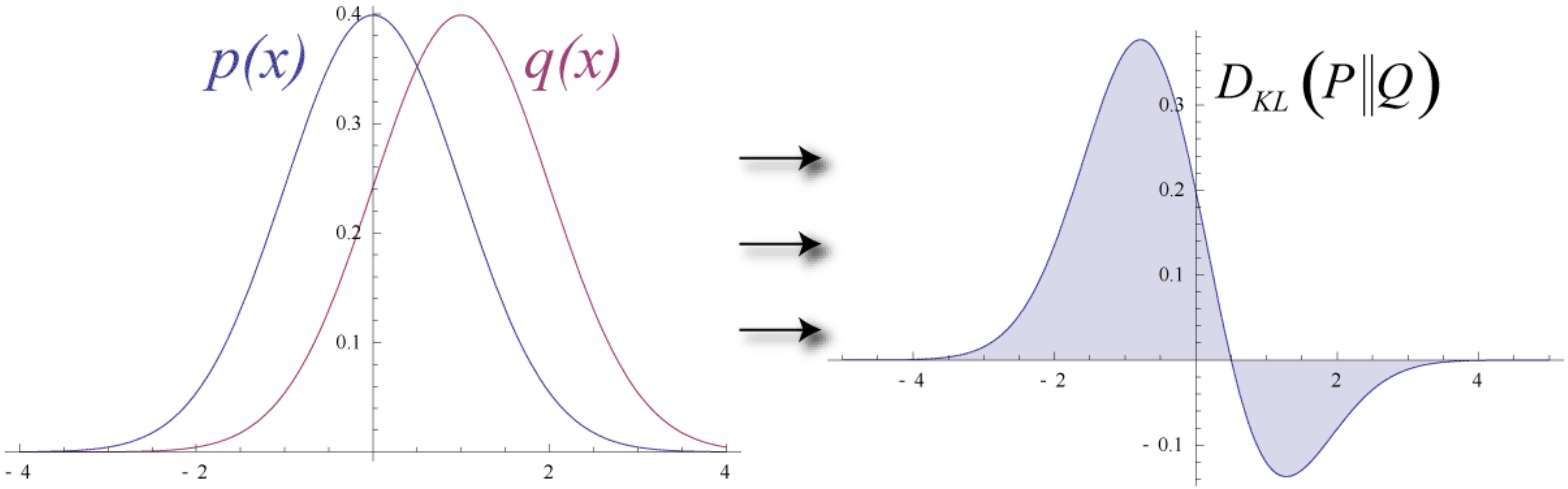
$$R_{emp}(h) = -\frac{1}{n}\sum_{i=1}^{n} y_i \log h(x_i) + (1 - y_i)\log(1 - h(x_i))$$

Supervised learning

- We can understand minimizing the cross-entropy loss makes the distribution of (empirical) labels and predictive labels become similar to each other.
- First, let’s investigate the meaning of Kullback-Leibler (KL) divergence

$$\mathrm{KL}(p||q) = \int p(x) \log \frac{p(x)}{q(x)} dx$$

Minimizing KL-divergence between the two distributions $p(x)$ and $q(x)$ makes those two distributions become similar.



Supervised learning

- For classification problems, ones aim to obtain predictive distribution which can explain given data distribution.
- Therefore, minimizing the KL-divergence between predictive and true distribution, $p_{pred}(y|x)$ and $p_{data}(y|x)$, can be an objective function of a classification problem:

$$\mathrm{KL}(p_{data}||p_{pred}) = \int p_{data}(y|x) \log \frac{p_{pred}(y|x)}{p_{data}(y|x)} dx$$

$$= \int p_{data}(y|x) \log p_{data}(y|x)\, dx \; - \int p_{data}(y|x) \log p_{pred}\,(y|x) dx$$

$$= -\int p_{data}(y|x) \log p_{pred}\,(y|x) dx$$

- For binary classification problems, we can rewrite this as summation of the divergence values for each sample

$$\mathrm{KL}(p_{true}||p_{data}) = \; -\sum_{i=1}^{n} y_i \log h(x_i) + (1 - y_i) \log(1 - h(x_i))$$

Supervised learning

- Note on KL-divergence

Minimising

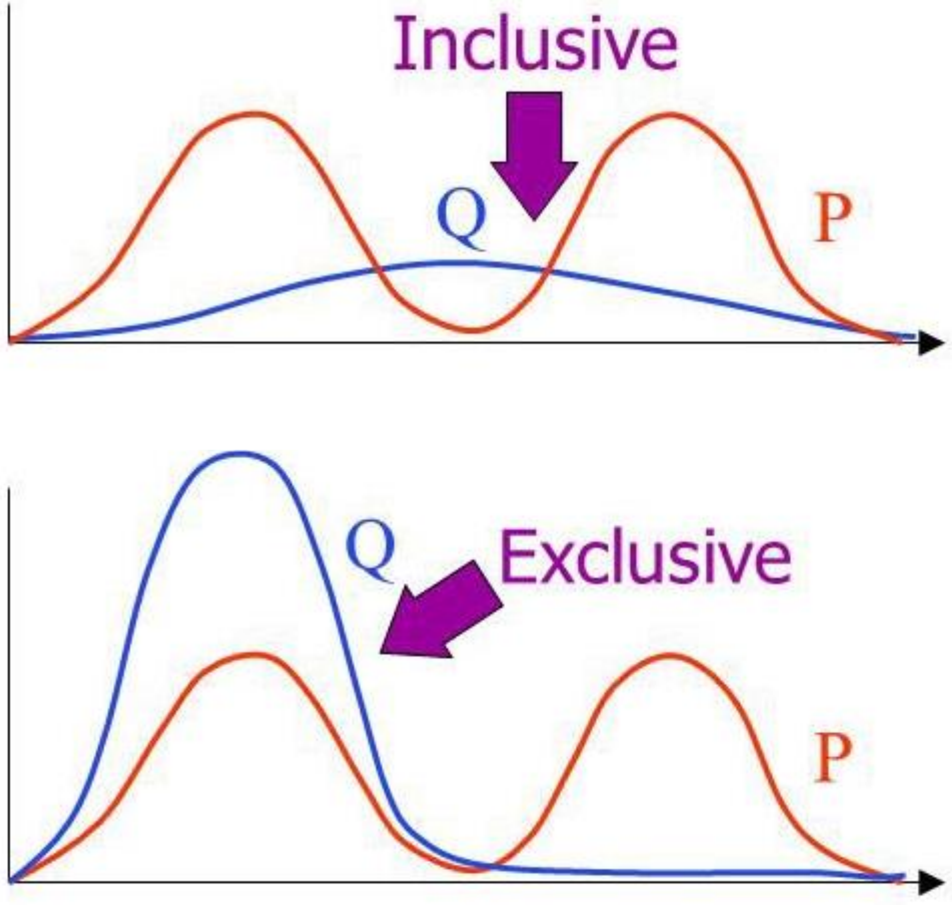
$\mathrm{KL}(P||Q)$

$$=\sum_{H} P(H|V)\ln\frac{P(H|V)}{Q(H)}$$

Minimising

$\mathrm{KL}(Q||P)$

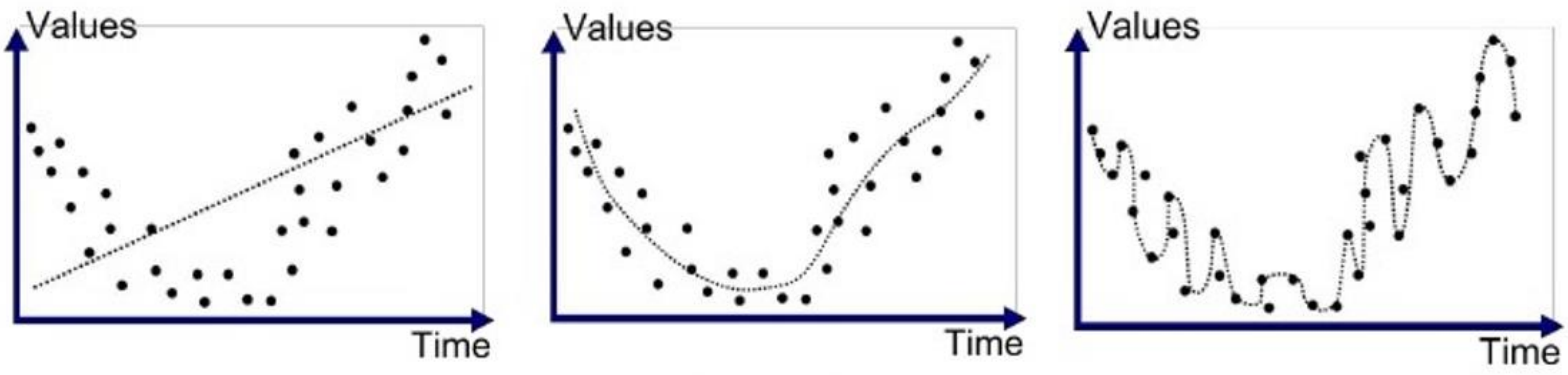
$$=\sum_{H} Q(H)\ln\frac{Q(H)}{P(H|V)}$$



See the chapter 21.2.2 of “Machine Learning: A Probabilistic Perspective” authored by Kevin Murphy

Supervised learning

- Minimizing the empirical risk usually causes *over-fitting* problem.



Underfitted

Good Fit/Robust

Overfitted

- Since we do not have true data distribution but have empirical data distribution, it is unavoidable to meet the generalization error, which is defined by the difference between the expected and empirical error.
- See the Wikipedia ‘Generalization error’, https://en.wikipedia.org/wiki/Generalization_error

Supervised learning

- Sometimes, we usually use the term ‘*bias-variance trade-off*’ to describe over-fitting problems.
- Let us consider the function with noise $y = f(x) + \epsilon$, where the noise term follows $\epsilon \sim \mathcal{N}(0, \sigma^2)$.
- Our goal is to find the hypothesis $\hat{f}(x)$, which approximates the true hypothesis $f(x)$ as well as possible.

  The expectation of empirical risk can be decomposed into bias and variance as follows:

$$\mathbb{E}\left[\left(y - \hat{f}\right)^2\right] = \mathbb{E}\left[y^2 - 2y\hat{f} + \hat{f}^2\right]$$

$$= \mathbb{E}[y^2] + \mathbb{E}\left[\hat{f}^2\right] - \mathbb{E}\left[2y\hat{f}\right]$$

$$= \mathrm{Var}[y] + \mathbb{E}[y]^2 + \mathrm{Var}\left[\hat{f}\right] + \mathbb{E}\left[\hat{f}\right]^2 - \mathbb{E}\left[2y\hat{f}\right]$$

$$= \mathrm{Var}[y] + \mathrm{Var}\left[\hat{f}\right] + \left(f^2 - \mathbb{E}\left[2y\hat{f}\right] + \mathbb{E}\left[\hat{f}\right]^2\right)$$

$$= \mathrm{Var}[y] + \mathrm{Var}\left[\hat{f}\right] + \left(f - \mathbb{E}\left[\hat{f}\right]\right)^2$$

$$= \mathrm{Var}[y] + \mathrm{Var}\left[\hat{f}\right] + \mathbb{E}\left[f - \hat{f}\right]^2$$

$$= \sigma^2 + \mathrm{variance} + \mathrm{bias}^2$$

* $\mathrm{Var}[X] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$

* $\mathbb{E}[y] = \mathbb{E}[f + \epsilon] = \mathbb{E}[f]$

* $\mathbb{E}[f] = f$

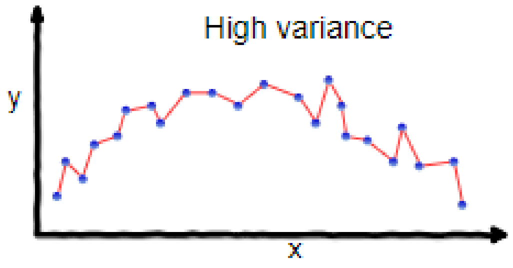
Supervised learning

- Bias-variance tradeoff:

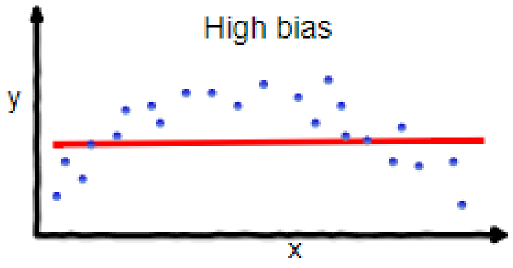
$$\mathbb{E}\left[\left(y-\hat{f}\right)^2\right] = \mathrm{Var}[y] + \mathrm{Var}\left[\hat{f}\right] + \mathbb{E}\left[f-\hat{f}\right]^2$$

$$= \sigma^2 + \text{variance} + \text{bias}^2$$

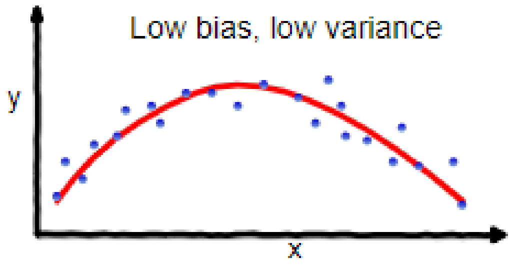
- The first term σ^2 is the *irreducible error*, inherent in our data, and also called as *aleatoric uncertainty*.
- The *variance* of our hypothesis can be understood as *how much the hypothesis will move around its mean*.
- The *bias* term can be thought of as the error caused by the simplifying assumptions built into the method. For example, when approximating a non-linear function $f(x)$ with a learning method for linear models, there will be error in the estimates $\hat{f}(x)$ due to too simple assumption.



overfitting



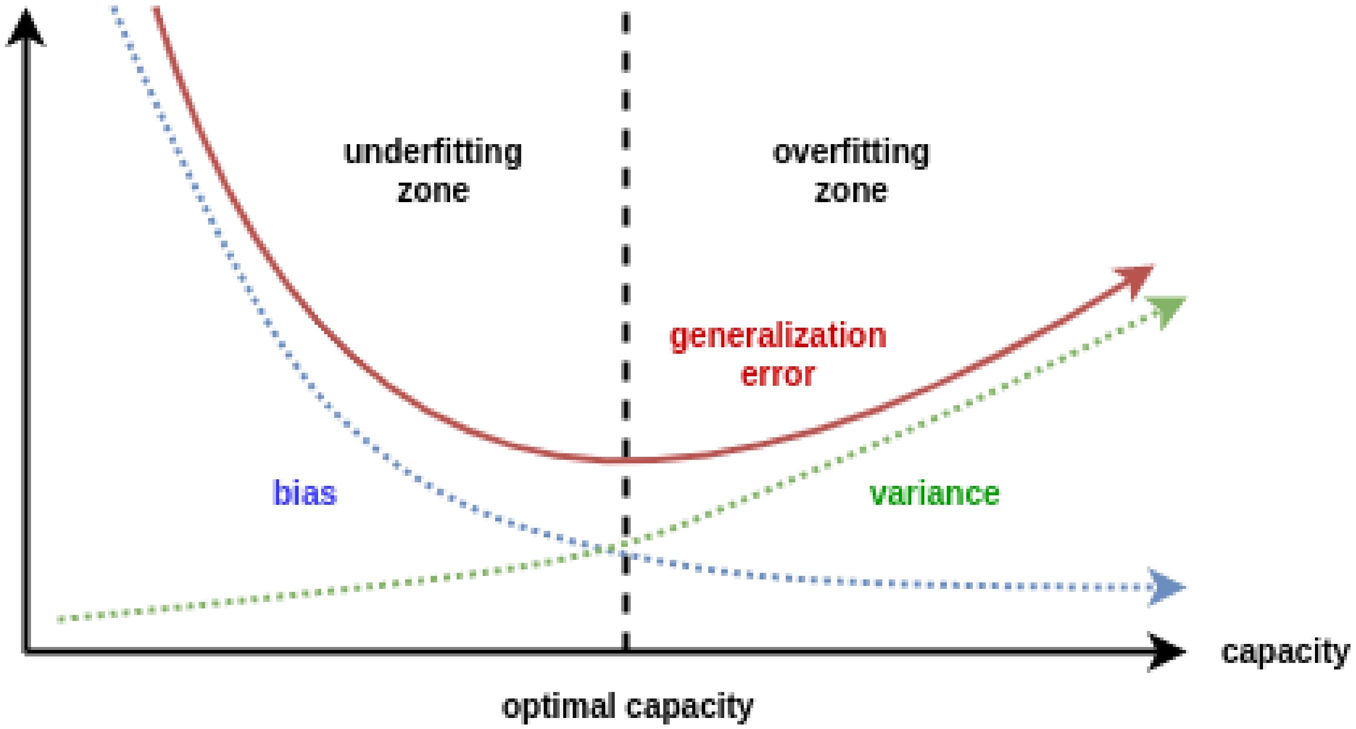
underfitting

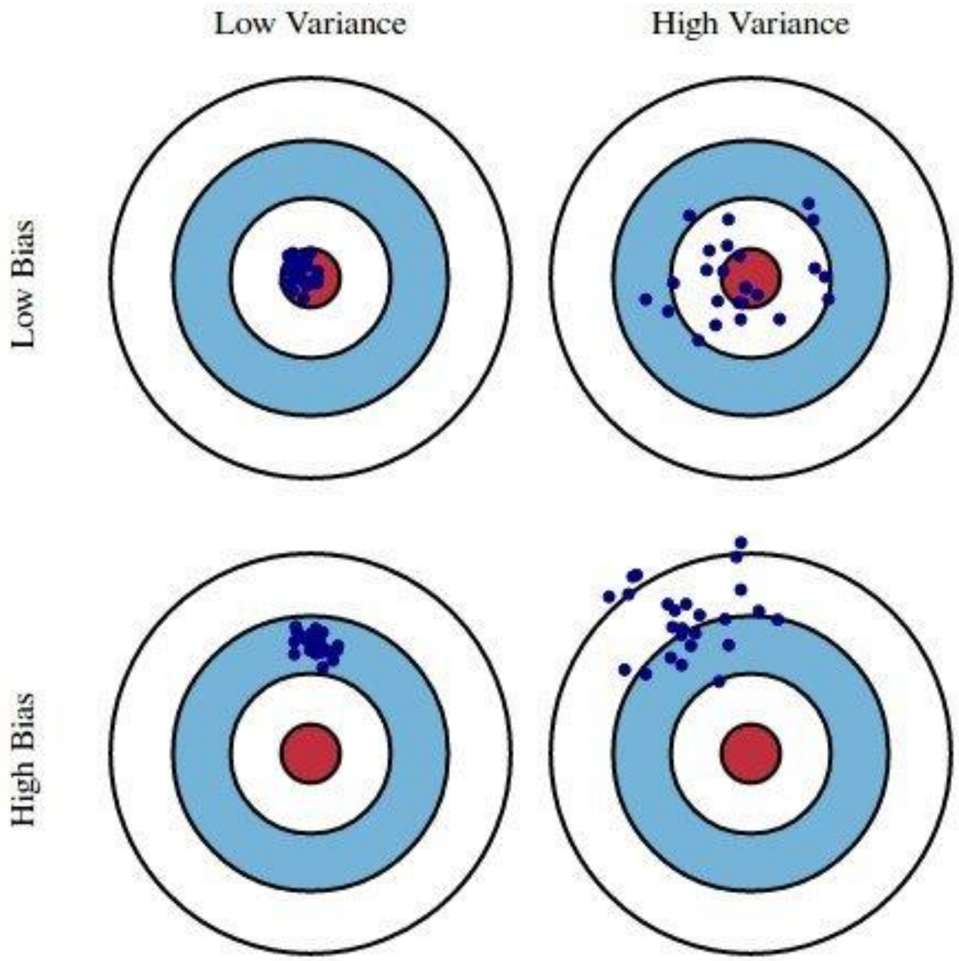


Good balance

Supervised learning

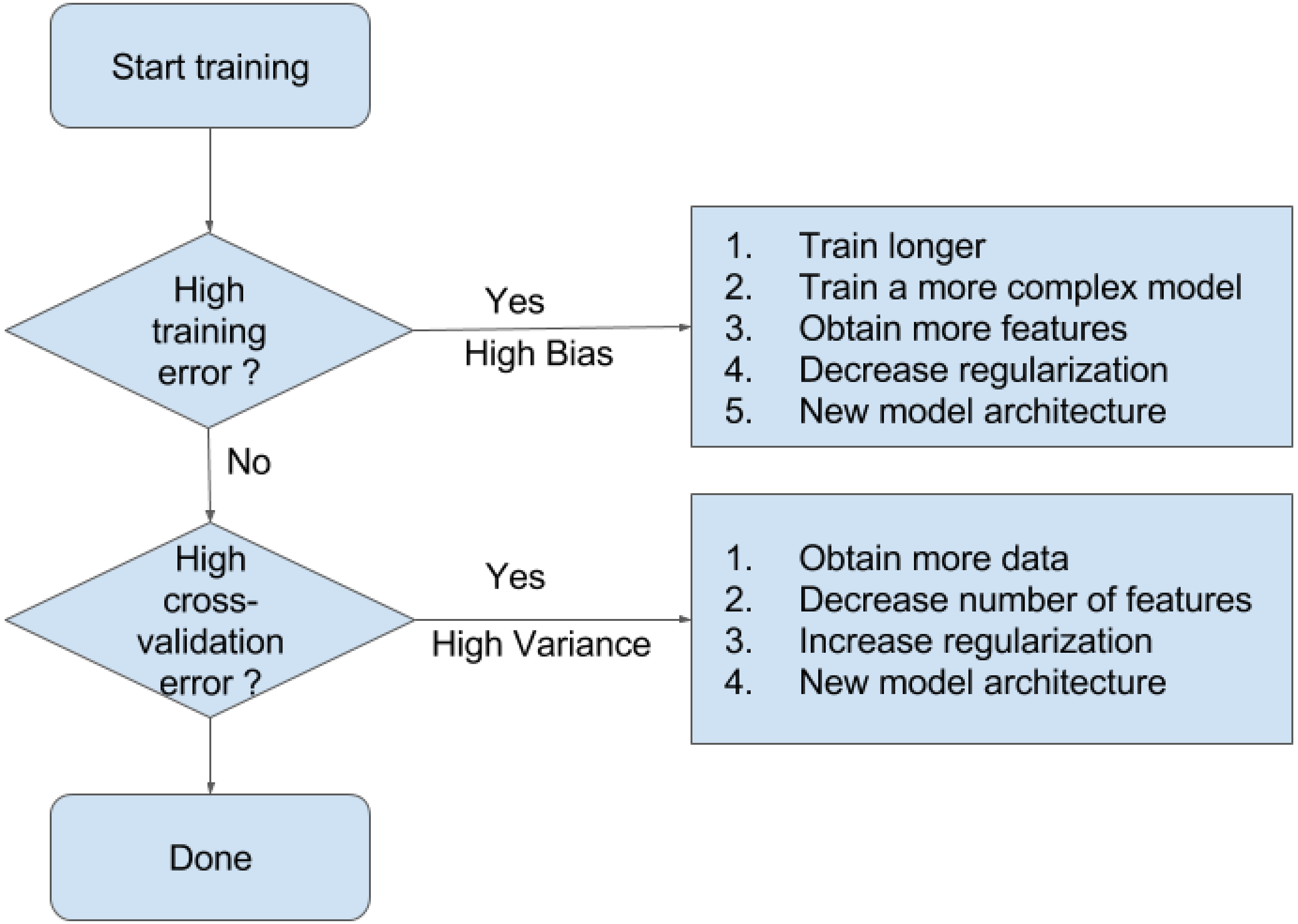
- Bias-variance tradeoff





Supervised learning

- Strategy to construct models with supervised-learning



1. Train longer
2. Train a more complex model
3. Obtain more features
4. Decrease regularization
5. New model architecture

1. Obtain more data
2. Decrease number of features
3. Increase regularization
4. New model architecture

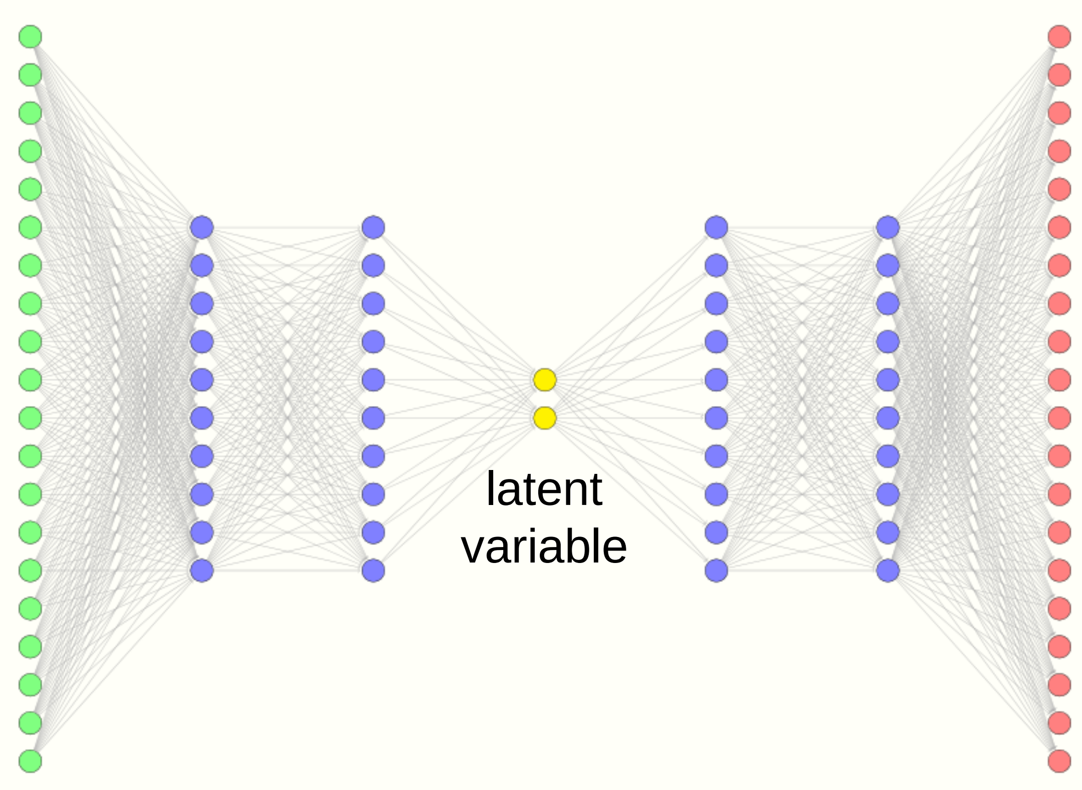
Unsupervised learning

- *Unsupervised learning* is a type of learning algorithms that find previously unknown patterns in examples without pre-existing labels.
- In statistics, latent variables are variables that are not directly observed but are rather inferred (through a mathematical model) from other variables that are observed, and mathematical models that aim to explain observed variables in terms of latent variables are called as *latent variable models*.
- For example, raw images represented by 32x32x3 pixels can have total $256^{32\times32\times3}$ bits.
- Therefore, finding latent variables (representations, hidden variables, patterns) in a low-dimensional space help us to solve given problem by reducing the dimensionality of data.
- Ex) Hidden Markov models, Variational autoencoders, Generative adversarial networks, …
- We will investigate very detail of unsupervised learning at the chapter ‘Generative models’.

Unsupervised learning

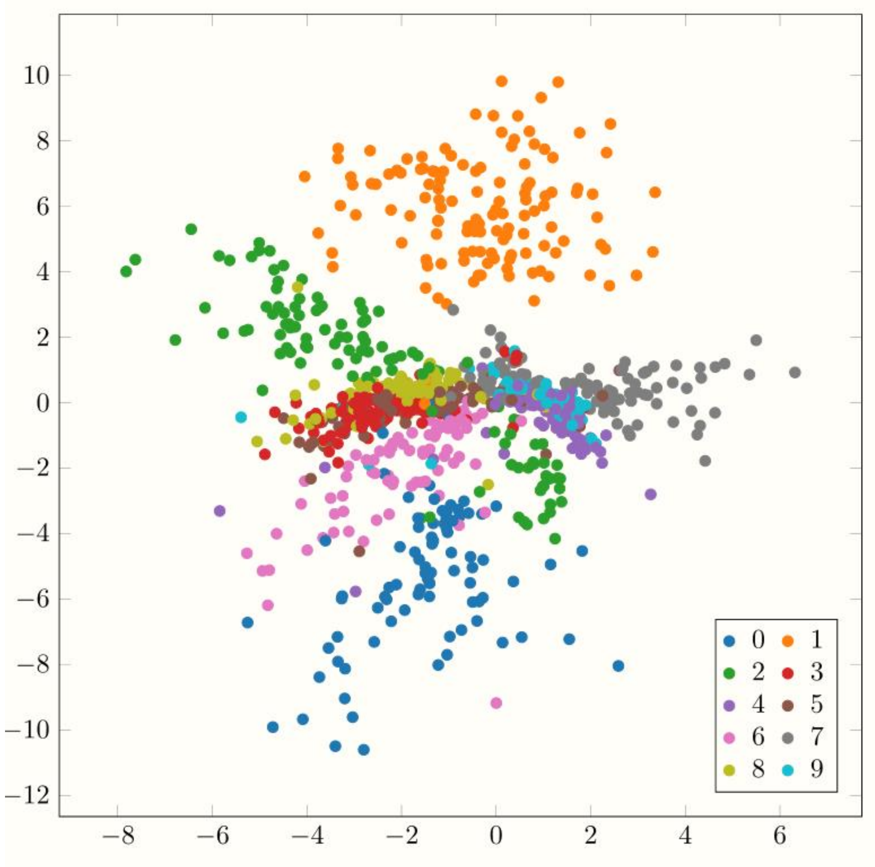
- Example: Autoencoder for a dimensionality reduction

the architecture of autoencoder



observable
variable

two dimensional latent space



https://gertjanvandenburg.com/blog/autoencoder/

Beyond supervised learning

- Securing labels is often very expensive and laborious, which makes training neural networks be troublesome.
- For example, annotating toxicity labels for hundreds of thousands of molecules will necessitate large scale bio-assay experiments.
- Data-efficient learning algorithms have actively studied in machine learning communities.
  - ✓ ***Semi-supervised learning*** utilizes unlabeled data as well as labeled data to train models. Unlabeled data is used to regularize models by forcing consistent predictions on both labeled and unlabeled data.
  - ✓ ***Transfer learning*** is a learning paradigm that transfer pre-trained knowledge to train other models.
  - ✓ ***Active learning*** communicates with external environment with so-called the acquisition function, which suggests data samples to be acquired. It aims to train models as small amount of labels as possible by acquiring most desired samples.
- We will investigate more in later chapters.