Introduction to Machine Learning

Binchi Zhang
LUD Lab, Xi'an Jiaotong University
2022/1/17

What is Machine Learning

- Traditional Programming:

Input: data, program

Output: result

- Machine Learning:

Input: data, result

Output: program

What is Machine Learning

Definition: 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.

--By Tom M. Mitchell

Three basic concepts:

- Task
- Metric
- Data

Sample Applications

- Bot Detection:

T: Determine whether an account is a bot.

P: Percentage of accounts correctly classified.

E: A social network data set with node labels.

- AlphaGo:

T: Play chess.

P: Percentage of games winning against people.

E: Playing practice games against itself.

Types of Learning

- Supervised learning

Given training data & desired output

- Unsupervised learning

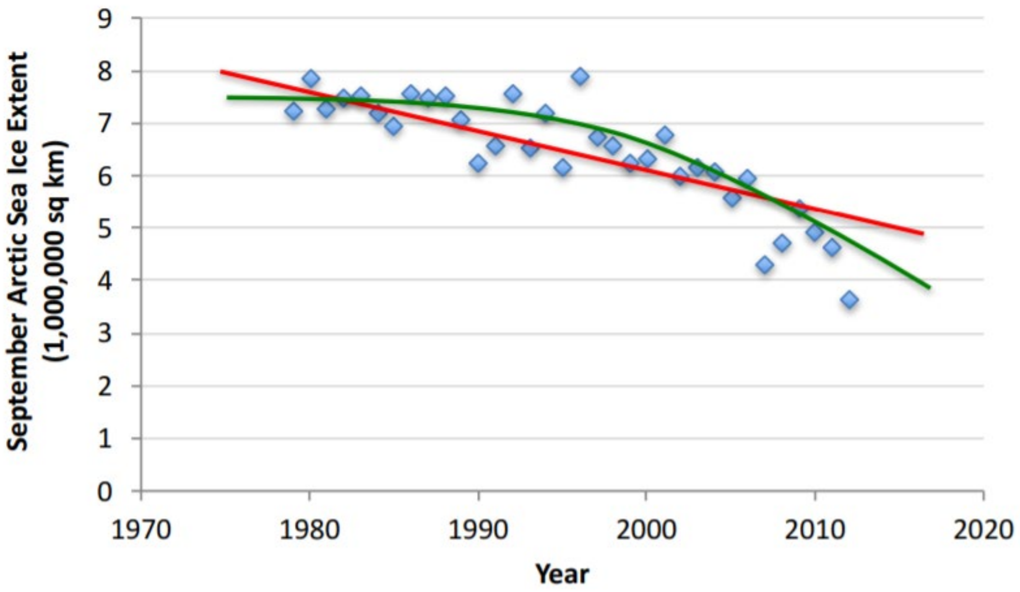
Given training data only

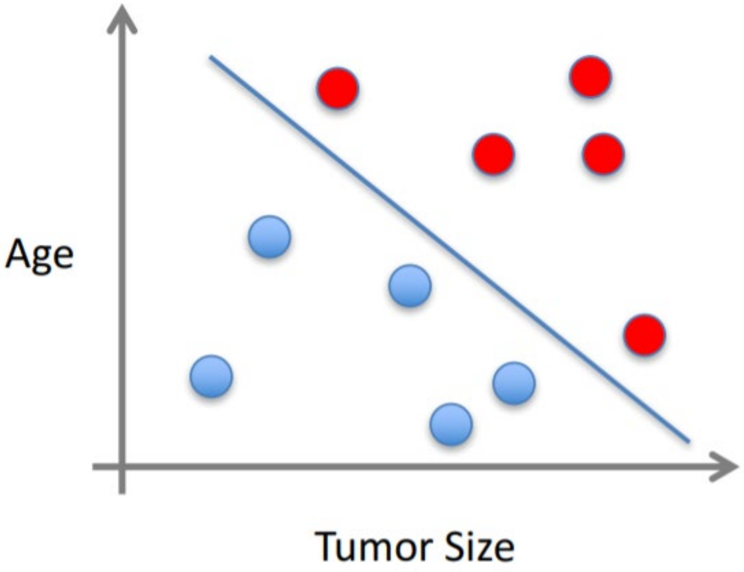
- Reinforcement learning

Rewards from action sequences

Supervised Learning

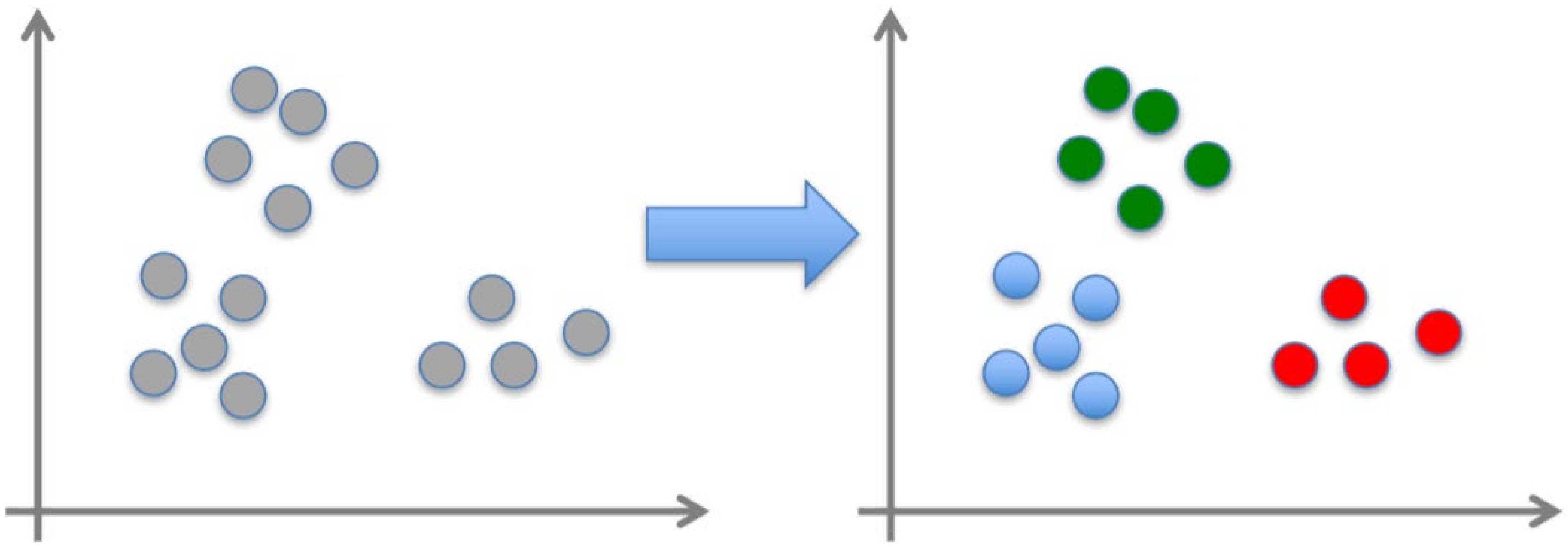
- Feature vector x
- Training set $\{(x_1, y_1), \ldots, (x_n, y_n)\}$
- Output y: regression/classification problem





Unsupervised Learning

- Training set $\{x_1, \ldots, x_n\}$
- Output hidden structure e.g. clustering



Model

To learn a function $y = f(x)$

Hypothesis space $F = \{f | Y = f(X)\}$

Types:

- Numerical Models: Linear Regression, Neural Network, Support Vector Machine…
- Probabilistic Graphical Models: Naïve Bayes, Hidden Markov models, Bayesian networks…
- Symbolic Models: Decision Trees…

Loss Function

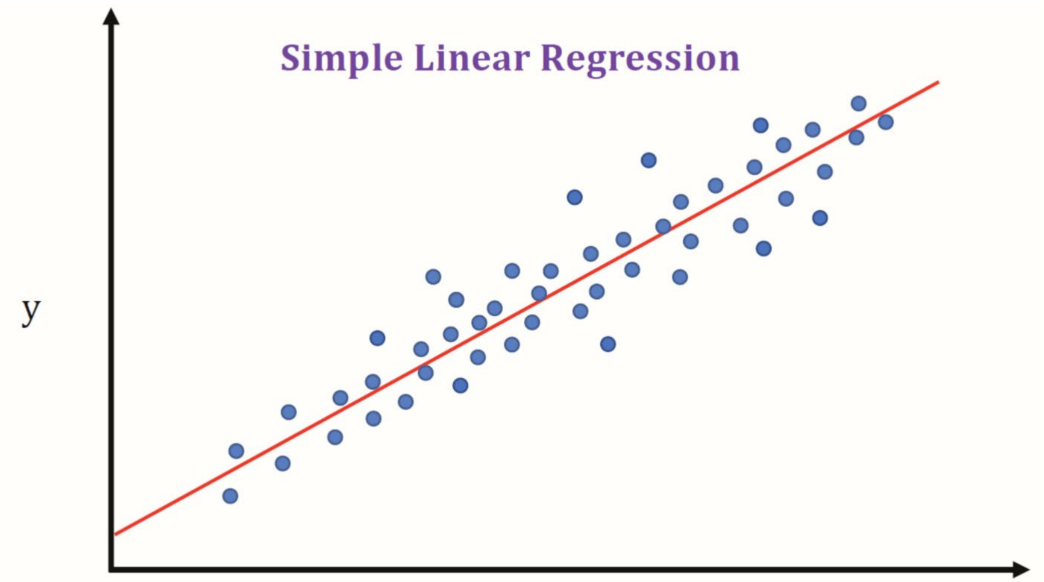
The error between prediction and actual value.

- 0-1 loss function: $L\big(Y, f(X)\big) = \begin{cases} 1, Y \neq f(X) \\ 0, Y = f(X) \end{cases}$
- Quadratic loss function: $L\big(Y, f(X)\big) = (Y - f(X))^2$
- Cross-entropy loss function: $L\big(Y, P(Y|X)\big) = -logP(Y|X)$

...

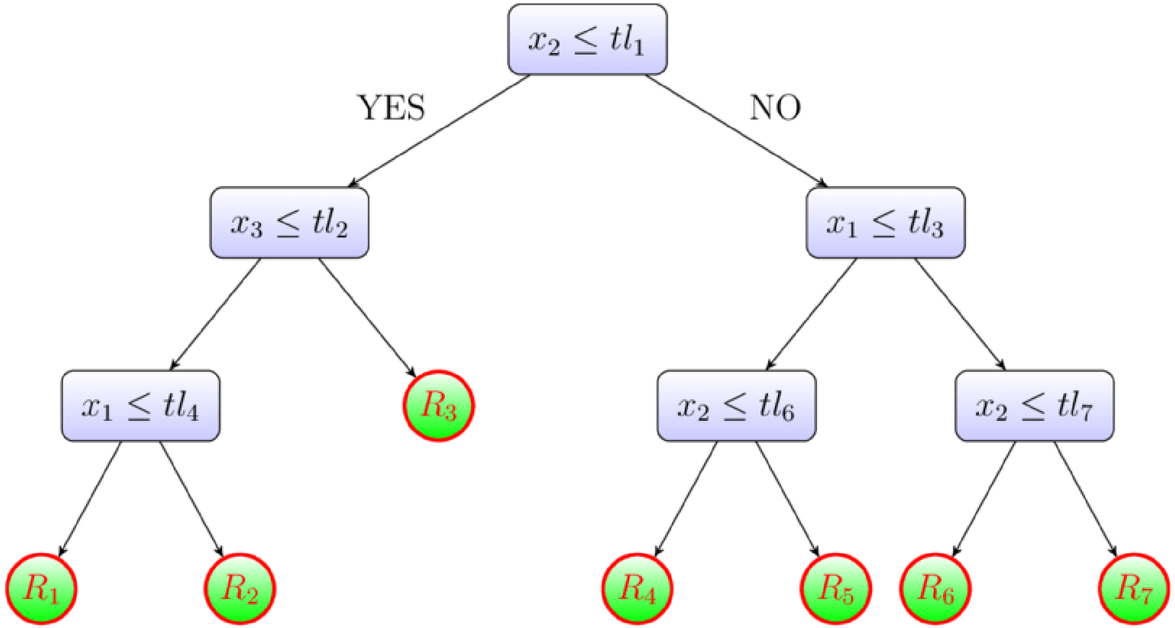
Linear Regression

- Regression model
- $f(x) = \theta_0 + \theta_1 x$
- $L(X, Y) = \sum_{i=1}^{n}(y_i - \theta_1 x_i - \theta_0)^2$



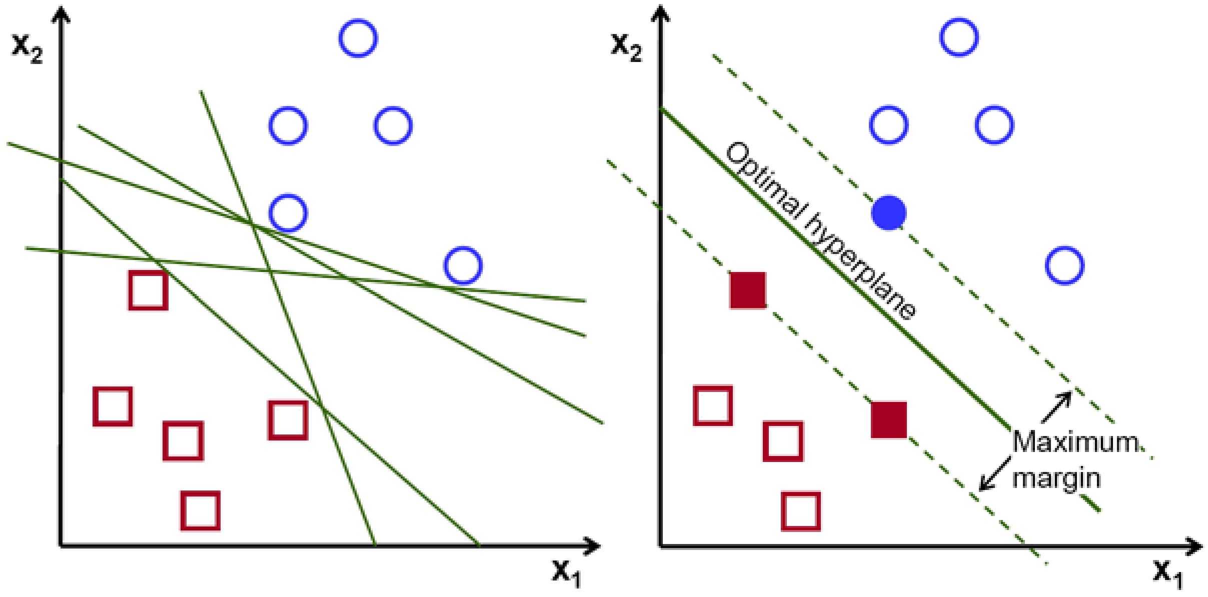
Decision Tree

- Classification model
- Internal nodes: attributes
- Leaf nodes: classes



Support Vector Machine

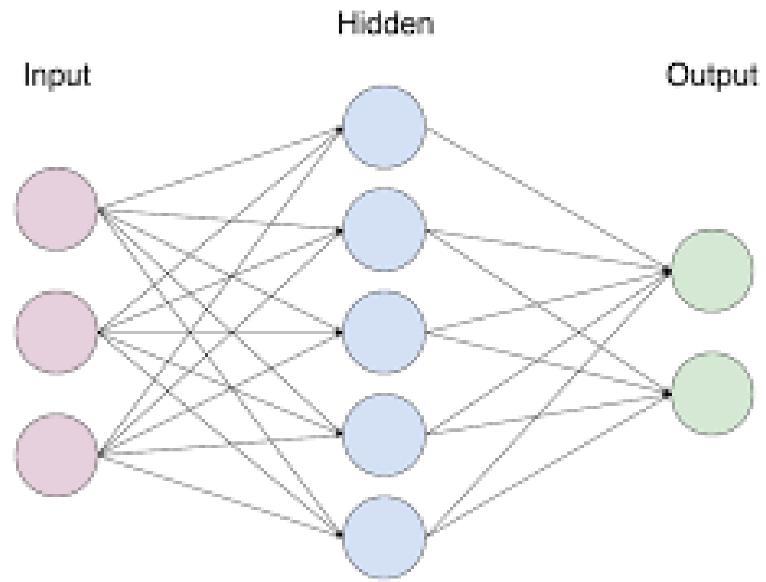
- Binary classification model
- Hyperplane: $w^T x + b = 0$, w: support vector
- $\min \frac{1}{2}\|w\|^2, y_i(w^T x_i + b) \geq 1$

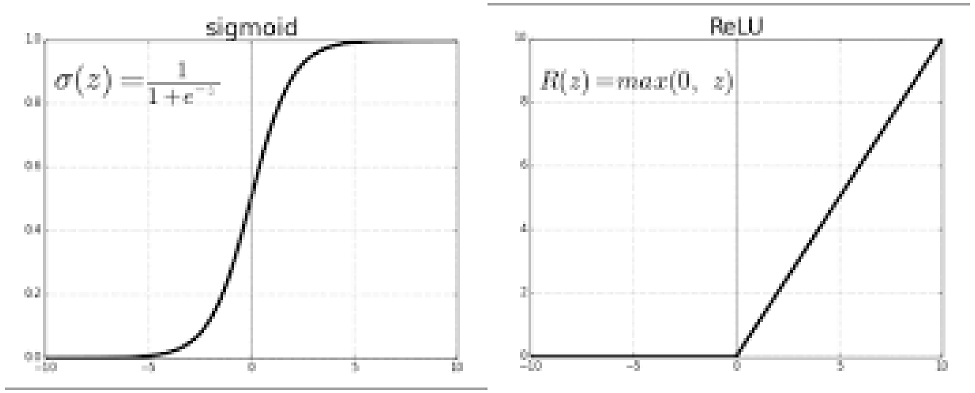


- Soft margin
- Kernel method

Basic Neural Network

- Mapping and activation
- $h^{(l+1)} = g(\Theta^{(l)} h^{(l)})$
- Activation function: Sigmoid, ReLU





Parameter

- **Model parameter**: trainable, hypothesis space; e.g.
- Weights of neural network
- Support vector
- Coefficients of linear regression

- **Hyperparameter**: manually set; e.g.
- Learning rate
- Batch size

Training & Test Distribution

- We generally assume that the training and test examples are independently drawn from the same overall distribution of data i.e. training set and test set are i.i.d
- To minimize the loss on unknown test set, we should minimize its estimator – empirical loss on train set:

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i))$$

To solve this optimization problem, we should…

Optimization

- Gradient descent based methods: SGD, mini batch GD; momentum, Nesterov, Adam

$$\theta := \theta - \alpha \frac{dJ}{d\theta}$$

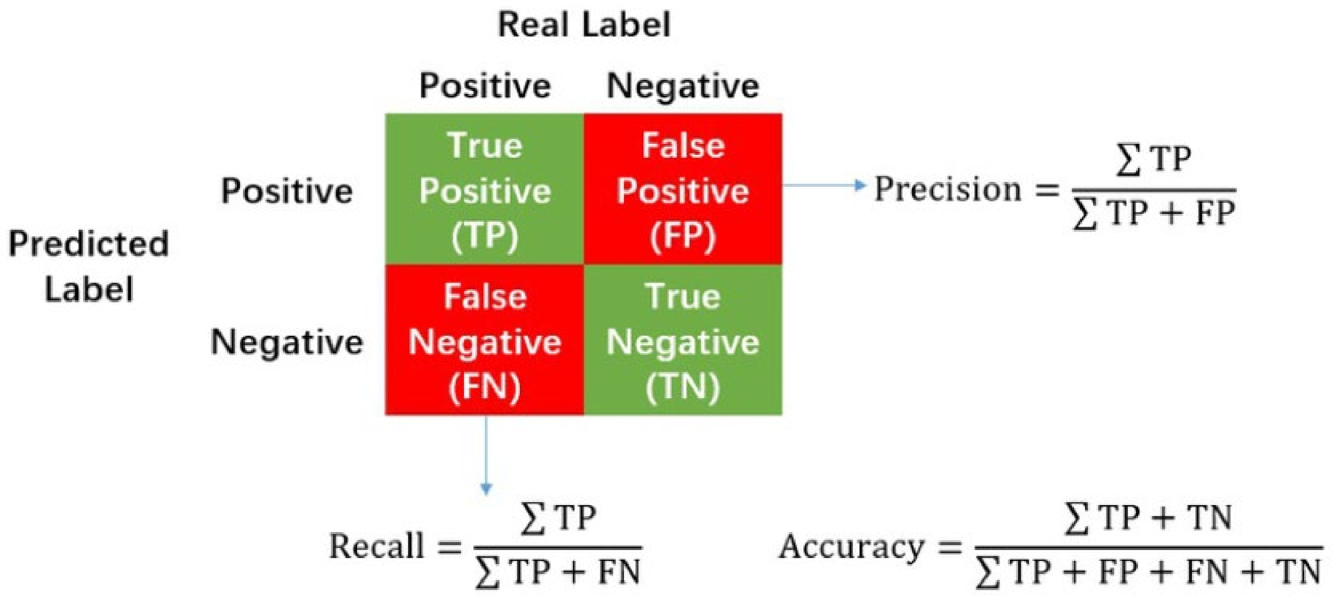
- Newton method

$$J(\theta) \approx J(\theta_0) + J'(\theta_0)(\theta - \theta_0) + \frac{1}{2}J''(\theta_0)(\theta - \theta_0)^2$$

- Coordinate descent method

Evaluation

- Metrics: measure model performance on tasks
- Classification: accuracy, precision, recall, f1-score
- Regression: MSE, MAE



Evaluation

- Dataset split: train, validation, test
- Underfitting & overfitting
- Regularization: $\lambda \sum ||\theta||_1$, $\lambda \sum ||\theta||_2^2$