Big Data : 4 Big "V"

大数据有4个V.

- Volume : KB, MB, GB (10^9 bytes), TB, PB, EB (10^{18} bytes), ZB, YB — 体积大
- Variety : different sources from business to industry, different types — 种类多
- Value : redundant information contained in the data, need to retrieve useful information — 价值大
- Velocity : fast speed for information transfer — 速度快

data science

- Retrieve information from data with the help of computational power
- Transfer the information into knowledge
- Two perspectives of data sciences :
  - Study science with the help of data : bioinformatics, astrophysics, geosciences, etc.
  - Use scientific methods to exploit data : statistics, machine learning, data mining, pattern recognition, data base, etc.

Scientific Study of Data

- Grabbing data : business and industrial problem, professional areas — 获取数据
- Storing data : engineering problem, computer science, electronic engineering — 存储数据
- Analyzing data (**key problem**) : scientific problem, mathematics, statistics, computer science — 分析数据：关键问题

Data Analysis

- Ordinary data types :
  - Table : classical data (could be treated as matrix)
  - Set of points : mathematical description
  - Time series : text, audio, stock prices, DNA sequences, etc.
  - Image : 2D signal (or matrix equivalently, e.g., pixels), MRI, CT, supersonic imaging
  - video : 2D in space and 1D in time (another kind of time series)
  - Webpage and newspaper : time series with spatial structure
  - Network : relational data, graph (nodes and edges)
- Basic assumption : the data are generated from an underlying model, which is unknown in practice
  - Set of points : probability distribution — 概率密度分布
  - Time series : stochastic processes, e.g., Hidden Markov Model (HMM) — 随机过程
  - Image : random fields, e.g., Gibbs random fields
  - Network : graphical models, Beyesian models

点集 时间序列 图 网络

Difficulties

- Huge volume of data — 数据量大
- Extremely high dimensions — 极高维度
  - Curse of dimensionality : the model complexity and computational complexity become exponentially increasing with the growth of dimension — 模型复杂度、计算复杂度
  - Solutions :
    - Make use of prior information
    - Restrict to simple models
    - Make use of special structures, e.g., sparsity, low rank, smoothness
    - Dimensionality reduction, e.g., PCA, LDA, etc.

维度灾难

- Complex variety of data — 数据种类复杂
- Large noise : data are always contaminated with noises — 噪音大

Machine Learning

Definition

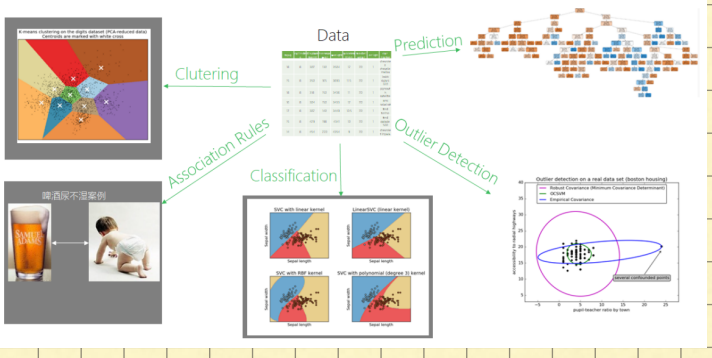
- Artificial Intelligence (AI) : learning from experiences (data), and improve the computer program adaptively
- Mathematics : Learning the underlying model from data, and generalize the model to adapt new data

> We define *machine learning* as a set of methods that can automatically detect patterns in data, and then use the uncovered patterns to predict future data, or to perform other kinds of decision making under uncertainty (such as planning how to collect more data!).
>
> — 《Machine Learning: A probabilistic perspective》

Related Areas

- Control theory : optimize the cost with optimal control parameters
- Information theory : entropy, optimal coding with best information
- Psycology : reference for machine learning algorithms
- Neuroscience : artificial neural network
- Biology : genetic algorithms
- Theory of Computing : study the computational complexity
- Statistics : large-sample limiting behavior, statistical learning theory
- Artificial Intelligence : symbolic computing
- Bayesian theory : conditionally probabilistic network

Applications



Classification: 分类
Regression: 回归
Clustering: 聚类
Prediction: 预测
Outlier Detection: 异常点检测
Association Rules: 关联规则

Supervised & unsupervised Learning

- Supervised learning : classification, regression
- Unsupervised learning : density estimation, clustering, dimensionality reduction
- Semi-supervised learning : with missing data, e.g., EM; self-supervised learning, learn the missing part of images, inpainting
- Reinforcement learning : play games, e.g., Go, StarCraft; robotics; auto-steering

监督学习：分类、回归
无监督学习
半监督学习
强化学习

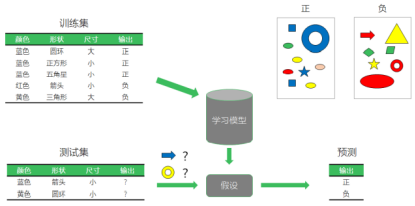
Supervised Learning 有监督学习

- Given labels of data : the labels could be symbols (spam or non-spam), integers (0 or 1), real numbers, etc.
- Training : find the optimal parameters (or model) to minimize the error between the prediction and target
- Classification : SVM, KNN, Desicion tree, etc.
- Regression : linear regression, CART, etc.

数据有标签

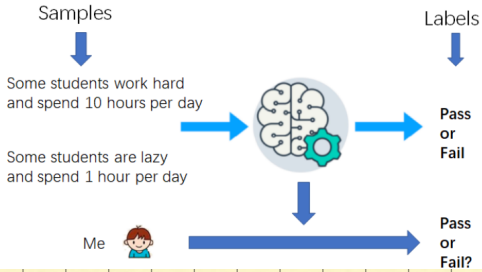
训练时要找到最优的参数/模型，使得预测与真实值误差最小。

分类和回归都是有监督学习



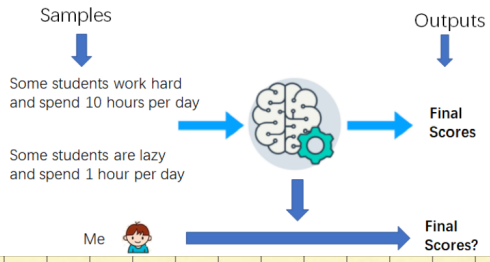
Classification

- Output is discrete 输出是离散的
- Examples : given the study hours, in-class performance, and final grades (Pass or Fail) of past students, can you predict the final grades of the current students based on their study hours and in-class performance ?
- Applications : Credit risk evaluation, clinical prediction of tumor, classification of protein functions, etc.



Regression

- Output is continuous 输出是连续的
- Examples : given the study hours, in-class performance, and final scores of past students, can you predict the final scores of the current students based on their study hours and in-class performance ?
- Applications : epidemiology, finance, investment analysis, etc.

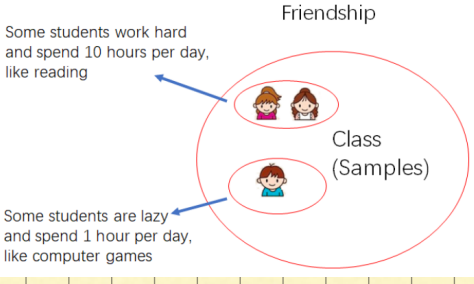


Unsupervised Learning 无监督学习

- No labels
- Optimize the parameters based on some natural rules, e.g., cohesion or divergence
- Clutering : K-Means, SOM

数据没有标签

基于一些基本规则来优化参数



Representation of Data

- Input space $\mathcal{X} = \{\text{All possible samples}\}$; $\mathbf{x} \in \mathcal{X}$ is an input vector, also called feature, predictor, independent variable, etc.; typically multi-dimensional; e.g., $\mathbf{x} \in \mathbb{R}^p$ is a weight vector or coding vector
- Output space $\mathcal{Y} = \{\text{All possible results}\}$; $y \in \mathcal{Y}$ is an output vector, also called response, dependent variable, etc.; typically one-dimensional; e.g., $y = 0$ or 1 for classification problems, $y \in \mathbb{R}$ for regression problems.
- For supervised learning, assume that $(\mathbf{x}, y) \sim \mathcal{P}$, a joint distribution on the sample space $\mathcal{X} \times \mathcal{Y}$

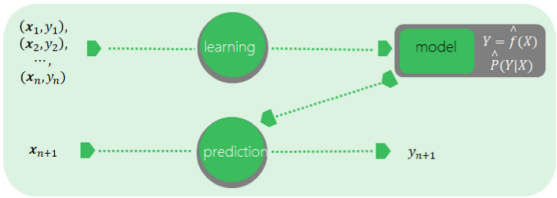
输入数据X，其元素一般为多维向量
也称为特征

输出数据Y，其元素一般为标量

对监督学习，一般假设有联合分布 $(X,Y)\sim P$，但P不可能知道。

Supervised Learning

- Goal : given $\mathbf{x}$, predict what is y; in deterministic settings, find the dependence relation $y = f(x)$; in probabilistic settings, find the conditional distribution $P(y|\mathbf{x})$ of y given $\mathbf{x}$
- Training dataset : $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n} \overset{i.i.d.}{\sim} \mathcal{P}$, used to learn an approximation $\hat{f}(x)$ or $\hat{P}(y|\mathbf{x})$
- Test dataset : $\{(\mathbf{x}_j, y_j)\}_{j=n+1}^{n+m} \overset{i.i.d.}{\sim} \mathcal{P}$, used to make a prediction $\hat{y}_j = \hat{f}(\mathbf{x}_j)$ or $\hat{y}_j = \arg\max_{y_j} \hat{P}(y_j|\mathbf{x}_j)$, and verify how accurate the approximation is



i.i.d.: independent and identically distributed
独立同分布

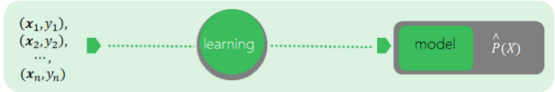
训练时通过 $\{(\vec{x}_i, y_i)\}_{i=1}^{n}$ 猜测出分布P

测试时通过 $\{(\vec{x}_i, y_i)\}_{i=n+1}^{m+n}$ 预测出结果并与真实值比较，验证准确性

学一定依赖于训练数据

Unsupervised Learning

- Goal : in probabilistic settings, find the distribution $P(\mathbf{x})$ of $\mathbf{x}$ and approximate it; there is no y
- Training dataset : $\{\mathbf{x}_i\}_{i=1}^{n} \overset{i.i.d.}{\sim} \mathcal{P}$, used to learn an approximation $\hat{P}(\mathbf{x})$; no test data in general



无输出，只有输入。
通过许多样本估计出样本总体分布
一般没有测试数据
对训练集的处理就是结果。

Learning Models

- Decision function (hypothesis) space : $\mathcal{F} = \{f_\theta | f_\theta = f_\theta(\mathbf{x}), \theta \in \Theta\}$ or $\mathcal{F} = \{P_\theta | P_\theta = P_\theta(y|\mathbf{x}), \theta \in \Theta\}$
- Loss function : a measure for the "goodness" of the prediction, $L(y, f(\mathbf{x}))$
  - 分类 0-1 loss : $L(y, f(\mathbf{x})) = I_{y \neq f(\mathbf{x})} = 1 - I_{y=f(\mathbf{x})}$ → 相等为0，无误差；不等为1，有误差
  - 回归 Square loss : $L(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$
  - 回归 Absolute loss : $L(y, f(\mathbf{x})) = |y - f(\mathbf{x})|$
  - 分类 Cross-entropy loss : $L(y, f(\mathbf{x})) = -y \log f(\mathbf{x}) - (1-y)\log(1 - f(\mathbf{x}))$ → 若 $y=f(x)$，则 $L(y,f(x))$ 最小
- Risk : in average sense, $R_{exp}(f) = E_P[L(y, f(\mathbf{x}))] = \int_{\mathcal{X}\times\mathcal{Y}} L(y, f(\mathbf{x}))P(\mathbf{x}, y)\mathrm{d}\mathbf{x}\mathrm{d}y$
- Target of learning : choose the best f^* to minimize $R_{exp}(f)$, $f^* = \min_f R_{exp}(f)$

决策函数空间（假设）：不确定的，取决于假设

损失函数：对每组样本点 $(\vec{x}, y)$ 的误差，衡量预测的好坏

平均误差：损失函数的期望
难直接求出，因为一般不知道P是多少
只能近似估计P，可用平均值来估计。
即 $R_{exp}(f) \approx \frac{1}{n}\sum_{i=1}^{n} L(y_i, f(x_i))$

学习的目标：
选出最优的 f^* 使得 $R_{exp}(f^*)$ 最小，即 $f^* = \min_f R_{exp}(f)$

Risk Minimization Strategy：依赖数据

- Empirical risk minimization (ERM) : given training set $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, $R_{emp}(f) = \frac{1}{n}\sum_{i=1}^{n} L(y_i, f(\mathbf{x}_i))$
  - By law of large number, $\lim_{n\to\infty} R_{emp}(f) = R_{exp}(f)$ → 由大数定律推出
  - Optimization problem : $\min_{f\in F} \frac{1}{n}\sum_{i=1}^{n} L(y_i, f(\mathbf{x}_i))$
- Structural risk minimization (SRM) : given training set $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ and a complexity functional $J = J(f)$, $R_{srm}(f) = \frac{1}{n}\sum_{i=1}^{n} L(y_i, f(\mathbf{x}_i)) + \lambda J(f)$ （模型复杂度描述）
  - $J(f)$ measures how complex the model f is, typically the degree of complexity
  - $\lambda \geqslant 0$ is a tradeoff between the empirical risk and model complexity
  - Optimization problem : $\min_{f\in F} \frac{1}{n}\sum_{i=1}^{n} L(y_i, f(\mathbf{x}_i)) + \lambda J(f)$

经验风险：用平均代替期望

结构风险

f：多项式 为元线性函数 分片函数

J(f)：次数 非零系数个数 越接近分片常数越好

入能平衡经验误差和模型复杂度

Algorithm

- Computational methods to solve the problem for f
- Numerical methods to solve the optimization problems
  - Gradient descent method, including coordinate descent, sequential minimal optimization (SMO), etc.
  - Newton's method and quasi-Newton's method
  - Combinatorial optimization
  - Genetic algorithms → 非凸问题
  - Monte Carlo methods → 配分以上的随机方法
  - ...

Model Assessment

Assume we have learned the model $y = \hat{f}(\mathbf{x})$, what is the error?

- Training error : $R_{emp}(\hat{f}) = \frac{1}{n}\sum_{i=1}^{n} L(y_i, \hat{f}(\mathbf{x}_i))$, tells the difficulty of learning problem
- Test error : $e_{test}(\hat{f}) = \frac{1}{m}\sum_{j=n+1}^{n+m} L(y_j, \hat{f}(\mathbf{x}_j))$, tells the capability of prediction; in particular, if 0-1 loss is used
  - Error rate : $e_{test}(\hat{f}) = \frac{1}{m}\sum_{j=n+1}^{n+m} I_{y_j \neq \hat{f}(\mathbf{x}_j)}$
  - Accuracy : $r_{test}(\hat{f}) = \frac{1}{m}\sum_{j=n+1}^{n+m} I_{y_j = \hat{f}(\mathbf{x}_j)}$
  - $e_{test} + r_{test} = 1$
- Generalization error : $R_{exp}(\hat{f}) = E_P[L(y, \hat{f}(\mathbf{x}))] = \int_{\mathcal{X}\times\mathcal{Y}} L(y, \hat{f}(\mathbf{x}))P(\mathbf{x}, y)\mathrm{d}\mathbf{x}\mathrm{d}y$（未知）, tells the capability for predicting unknown data from the same distribution, its upper bound M defines the generalization ability
  - As $n \to \infty$, $M \to 0$
  - As F becomes larger, M increases

训练误差：即最小值点

测试误差：将所取模型代入其他样本集中计算平均误差.

泛化误差

能估计出 Rexp 的上界

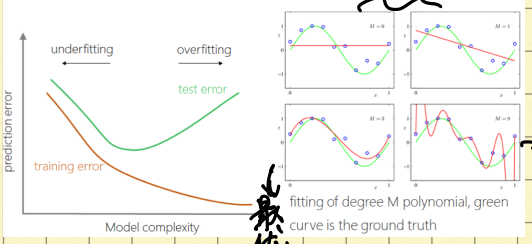
若能控制到上界取值较小也可.

对error而言，都是固定的最优函数 $\hat{f}(x)$

而risk是对抽象的一般任意函数.

Overfitting 过拟合

- Too many model parameters
- Better for training set, but worse for test set



欠拟合

过拟合

最优解

过多的模型参数.
对训练集拟合得好,但对测试集预测得差.

Model Selection

- Regularization : $\min_{f \in F} \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(\mathbf{x}_i)) + \underbrace{\lambda J(f)}_{\text{penalty}}$, choose λ to minimize empirical risk and model complexity simultaneously
- Cross-validation (CV) : split the training set into training subset and validation subset, use training set to train different models repeatedly, use validation set to select the best model with the smallest (validation) error
  - Simple CV : randomly split the data into two subsets
  - K-fold CV : randomly split the data into K disjoint subsets with the same size, treat the union of $K-1$ subsets as training set, the other one as validation set, do this repeatedly and select the best model with smallest mean (validation) error
  - Leave-one-out CV : $K = n$ in the previous case

1	2	3	4	5
Train	Train	Validation	Train	Train

正则化:
λ取决于更关心 risk 还是 model complexity

交叉验证:当没有 test set 时,对 training set 进行分割

简单CV:随机将数据分成两份

K折CV:随机将数据平分成K份,将其中K-1份作为training set,另一份作为test set.

loo CV:K取值n,即将每个样本都当作为一次测试数据

S_1	S_2	S_3

$S_1 \cup S_2$ training: $\min_f \frac{1}{|S_1 \cup S_2|} \sum_{(x_i, y_i) \in S_1 \cup S_2} L(y_i, f(x_i)) + \lambda J(f) \Rightarrow \hat{f}^{(-3)}(x, \lambda)$

S_3 validation: error: $\frac{1}{|S_3|} \sum_{(x_i, y_i) \in S_3} L(y_i, \hat{f}_{(S_1 \cup S_2)}(x_i, \lambda)) = e^{(-3)}(\lambda)$

以此类推,取出不同部分作为validation,其余training.

最终 $\hat{\lambda} = \min_{\lambda} \bar{e}(\lambda) = \min_{\lambda} \frac{1}{3}(e^{(-1)}(\lambda) + e^{(-2)}(\lambda) + e^{(-3)}(\lambda))$

test error

$\hat{\lambda}$: CV-selection

取 $\hat{f} = \min_f R_{SRM}(f) = \frac{1}{|S_1 \cup S_2 \cup S_3|} \sum L(y_i, f(x_i)) + \hat{\lambda} R(f)$