Introduction

机器学习：algorithms that allow computers to **evolve behaviors** based on **empirical data**。要同时有数据和算法。数据本身价值很低，通过机器学习挖掘有用的部分。机器学习三驾马车：模型（如 SVM）、数据、算法。要求**独立同分布**：训练与测试在同一分布中采样得到。**有监督**：给定分类标签。**无监督**：无标签。

术语：给定**数据集** (data set) 中每条记录是**示例** (instance) 或**样本** (sample)，若有标签则为**样例** (example)，样例 y_i 张成空间为**标记空间** (label space)。反应对象某方面性质的事项是**属性** (attribute) 或**特征** (feature)，属性上取值为**属性值** (attribute value)，张成空间为**属性空间** (attribute space/sample space)，一个示例也是一个**特征向量** (feature vector)。学得数据的潜在规律为**假设** (hypothesis)，规律本身为**真相** (ground-truth)，学习算法为**学习器** (learner)。

基础：提取特征 → 算法模型 → 衡量标准。提取：将不同输入转化为特征向量。文本使用独热编码，图片使用像素值，语音识别使用 MFCC。

步骤：确定模型 → 训练模型 → 使用模型。

误差：泛化误差越小越好，经验误差（训练集误差）未必。

模型评估：留出法 (hold-out)、交叉验证 (cross-validation)、自助法 (bootstrap)。

留出法：直接划分 $D=S+T$，使用**分层采样**（即每类分别采样，用于类别多且复杂），重复多次取均值。

交叉验证：$D=D_1+\ldots D_k$，得到 k 个正确率取均值。通常 $k=10$。$*$ 若 k 为数据集大小则为留一法 (leave-one-out)。

自助法：令 $|D|=m$，则作放回采样 m 次得到 D' 为训练集，未采样数据为测试集，适用于数据集较小。

性能度量：衡量泛化能力。例如回归使用均方误差，分类任务使用错误率。对于分类任务，**精度** (accuracy)、**通过率** (sensitivity=TPR= $\frac{\text{TP}}{\text{TP+FN}}$)、**假阳率** (FPR= $\frac{\text{FP}}{\text{FP+TN}}$)，作 FPR(x)-TPR(y) 曲线得 **ROC 曲线**，曲线下面积 **AUC**，**查全率** (recall= $\frac{\text{TP}}{\text{TP+FN}}$)、**查准率** (precision= $\frac{\text{TP}}{\text{TP+FP}}$)、**Precision-Recall 图**（平衡点：查准率 = 查全率，根据平衡点判断优劣）、**F1-Score**($\text{F}_\beta=\frac{(1+\beta^2)\text{Prec}\times\text{Reca}}{\beta^2\times\text{Prec+Reca}}$)。判断是否有实质差别：使用假设检验。

Bias-Variance 分解：数据真实分布 $\mathcal{D}$，训练集 $S\sim\mathcal{D}^m$，每个元素 $\left(\boldsymbol{x}^{(i)},y^{(i)}\right)\in\mathbb{R}^n\times\mathbb{Y}$。则学习算法为 $\text{ALG}\in((\mathbb{R}^n\times\mathbb{Y})^m\to(\mathbb{R}^n\to\mathbb{Y}))$，模型为 $\hat{h}_S=\text{ALG}(S)$。令 h^* 为最优模型。则均方误差

$$\begin{aligned}\text{MSE}=&\mathbf{E}_{S,(\boldsymbol{x},y)\sim\mathcal{D}}\left[\left(y-\hat{h}_S(\boldsymbol{x})\right)^2\right]=\mathbf{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[(y-h^*(\boldsymbol{x}))^2\right]\\&+\left(h^*(\boldsymbol{x})-\mathbf{E}_S\left[\hat{h}_S(\boldsymbol{x})\right]\right)^2+\text{Var}_S\left(\hat{h}_S(\boldsymbol{x})\right)\end{aligned}$$

三项分别为固定误差，bias 和 variance。

监督学习

回归：

线性回归 $\hat{y}=\boldsymbol{\theta}^\top\boldsymbol{x}$，解为 $\boldsymbol{\theta}=\left(\boldsymbol{X}^\top\boldsymbol{X}\right)^{-1}\boldsymbol{X}^\top\boldsymbol{y}$ 其中 $\boldsymbol{X}=\left[\boldsymbol{x}^{(1)},\ldots,\boldsymbol{x}^{(n)}\right]^\top\in\mathbb{R}^{n\times d}$，几何解释为使预测值 $\hat{\boldsymbol{y}}$ 为真实值 $\boldsymbol{y}$ 在 $\boldsymbol{x}^{(i)}$ 张成平面上的投影，即 $\boldsymbol{X}^\top(\hat{\boldsymbol{y}}-\boldsymbol{y})=\boldsymbol{0}$。也可以用梯度下降。最小化均方误差实际上等价于在 $y^{(i)}=\boldsymbol{\theta}^\top\boldsymbol{x}^{(i)}+\epsilon_i$，$\epsilon_i\sim\mathcal{N}(0,\sigma)$ 的条件下最大似然。

岭回归 (ridge regression)：线性回归闭式解本身不稳定，容易奇异，因此可以改为最小化 $\frac{1}{2}\|\boldsymbol{X}\boldsymbol{\theta}-\boldsymbol{y}\|^2+\lambda\|\boldsymbol{\theta}\|^2$（等价于添加约束 $\|\boldsymbol{\theta}\|^2\le t$），变为 $\left(\boldsymbol{X}^\top\boldsymbol{X}+\lambda\boldsymbol{I}\right)\boldsymbol{\theta}=\boldsymbol{X}^\top\boldsymbol{y}$。

套索回归 (lasso regression)：L2 约束变为 L1 约束，**稀疏**。求解方法：梯度下降（使用次梯度）；二次规划（$\boldsymbol{\theta}=\boldsymbol{\theta}_+-\boldsymbol{\theta}_-$）；ISTA。

分类：

逻辑回归：二分类问题，$p(y)$ 有先验 Bernoulli 分布，$p(\boldsymbol{x}|y)$ 为 Gaussian，由此假设得线性分类器，最大化似然概率 $\ell(\boldsymbol{w})=\sum_{i=1}^m\left(1+\exp\left(-y^{(i)}\left(\boldsymbol{w}^\top\boldsymbol{x}^{(i)}+b\right)\right)\right)$。梯度下降求解。

K-NN：选择 k 和距离度量作为超参数。距离一般取 L-p norm。

Bayes 决策：$p(\boldsymbol{\theta}|\boldsymbol{x})=\frac{p(\boldsymbol{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\boldsymbol{x})}$ 其中表示后验概率怎样根据 $\boldsymbol{x}$ 的观测改变先验概率。不同 $\boldsymbol{\theta}$ 中决策：选取最大后验，决策边界随不同先验概率改变。**朴素贝叶斯**：特征之间彼此独立 $p(\boldsymbol{x}|\boldsymbol{\theta})=\prod p(x_i|\boldsymbol{\theta})$。

广义贝叶斯：$\{\boldsymbol{w}_j|j\in[c]\}$ 为自然状态的集合，$\{\alpha_i|i\in[a]\}$ 为可能行动集合，$\lambda(\alpha_i|\boldsymbol{w}_j)$ 为状态下采取特定行动的损失。则贝叶斯风险 $R(\alpha_i|\boldsymbol{x})=\sum_j\lambda(\alpha_i|\boldsymbol{w}_j)p(\boldsymbol{w}_j|\boldsymbol{x})$。例如二分类中 α_i 为“决策为 $\boldsymbol{w}_i$”，在给定 λ_{ij} 的情况下可以决定采用 α_1 还是 α_2。

$$\begin{aligned}R(\alpha_1|\boldsymbol{x})<R(\alpha_2|\boldsymbol{x})\Leftrightarrow&(\lambda_{21}-\lambda_{11})p(\boldsymbol{x}|\boldsymbol{w}_1)p(\boldsymbol{w}_1)>(\lambda_{12}-\lambda_{22})p(\boldsymbol{x}|\boldsymbol{w}_2)p(\boldsymbol{w}_2)\\\Leftrightarrow&\frac{p(\boldsymbol{x}|\boldsymbol{w}_1)}{p(\boldsymbol{x}|\boldsymbol{w}_2)}>\frac{\lambda_{12}-\lambda_{22}}{\lambda_{21}-\lambda_{11}}\frac{p(\boldsymbol{w}_2)}{p(\boldsymbol{w}_1)}\end{aligned}$$

对于 $\lambda_{ij}=1-\delta_{ij}$ 有 $R(\alpha_i|\boldsymbol{x})=1-p(\boldsymbol{w}_i|\boldsymbol{x})$，即最大化后验。贝叶斯需要知道 $p(\boldsymbol{w}_j)$ 先验和 $p(\boldsymbol{x}|\boldsymbol{w}_j)$，但是通常不知道。因此在假设 $p(\boldsymbol{x}|\boldsymbol{w}_j)=\mathcal{N}(\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j)$ 的基础上最大似然 $\max_{\boldsymbol{\theta}}p(\boldsymbol{x}|\boldsymbol{\theta})$。或者引入 $\boldsymbol{\theta}$ 的先验作最大后验估计 $\max_{\boldsymbol{\theta}}p(\boldsymbol{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})$，例如在 $p(x|\theta)\sim\mathcal{N}(\theta,\sigma^2)$ 和 $p(\theta)\sim\mathcal{N}(\mu_0,\sigma_0^2)$ 下，$p(\theta|S)\sim\mathcal{N}(\hat{\mu},\hat{\sigma}^2)$，其中

$$\hat{\mu}=\frac{n\sigma_0^2}{n\sigma_0^2+\sigma^2}\frac{\sum_i x_i}{n}+\frac{\sigma^2}{n\sigma_0^2+\sigma^2}\mu_0,\text{ and }\hat{\sigma}^2=\frac{\sigma_0^2\sigma^2}{n\sigma_0^2+\sigma^2}$$

线性判别分析 (Linear Discriminant Analysis)：映射到新的空间里使得类间距离大，类内距离小。$\boldsymbol{u}_i$ 为第 i 类均值，则类间距离 $d_b^2=\left|\boldsymbol{w}^\top(\boldsymbol{u}_1-\boldsymbol{u}_2)\right|^2$，类内距离 $d_w^2=\boldsymbol{w}^\top\left(\sum_i\boldsymbol{\Sigma}_i\right)\boldsymbol{w}$ 其中 $\boldsymbol{\Sigma}_i$ 为第 i 类协方差矩阵。目标为 $\max_{\boldsymbol{w}}\frac{d_b}{d_w}$。定义类内散射矩阵 $\boldsymbol{S}_w=\sum_i\boldsymbol{\Sigma}_i$，类间散射矩阵 $\boldsymbol{S}_b=(\boldsymbol{u}_1-\boldsymbol{u}_2)(\boldsymbol{u}_1-\boldsymbol{u}_2)^\top$，则最大化 $J(\boldsymbol{w})=\frac{\boldsymbol{w}^\top\boldsymbol{S}_b\boldsymbol{w}}{\boldsymbol{w}^\top\boldsymbol{S}_w\boldsymbol{w}}$，偏导 $\boldsymbol{S}_w^{-1}\boldsymbol{S}_b\boldsymbol{w}=J(\boldsymbol{w})\boldsymbol{w}$，对 $\boldsymbol{S}_w^{-1}\boldsymbol{S}_b$ 特征值分解取最大的特征向量。本质是降维，降维后使用 K-NN 算法，也可以保留更多特征向量从而保留更多信息，在对应维度上 K-NN。

多类 LDA：定义全局散度矩阵 $\boldsymbol{S}_T=\sum_i\left(\boldsymbol{x}^{(i)}-\boldsymbol{u}\right)\left(\boldsymbol{x}^{(i)}-\boldsymbol{u}\right)^\top$ 其中 $\boldsymbol{u}$ 为所有样本中心。则 $\boldsymbol{S}_T=\boldsymbol{S}_w+\sum_j N_j(\boldsymbol{u}_j-\boldsymbol{u})(\boldsymbol{u}_j-\boldsymbol{u})^\top$，后一项定义为 $\boldsymbol{S}_b$，仍然对 $\boldsymbol{S}_w^{-1}\boldsymbol{S}_b$ 特征值分解。注意到 $\boldsymbol{S}_b$ 的表达式意味着至多 rank 为 $M-1$，所以维数 $\le M-1$。

决策树：每个节点判断是否 $x_j\ge\text{thresh}$，选取原则为信息增益 $\text{IG}(X)=H(Y)-H(Y|X)$ 最大的特征 X（其中 Y 为标签）。

Def 熵：对于 $X\sim p(x)$，有 $H(X)=\mathbf{E}_p\left[-\log p(x)\right]=-\int_{\mathbb{R}}p(x)\log p(x)\mathrm{d}x$。

Def 条件熵：$H(Y|X)=-\int_{\mathbb{R}^2}p(x,y)\log p(y|x)\mathrm{d}y\mathrm{d}x$。

支持向量机：最大化几何间隔 $\gamma=\min\limits_{(\boldsymbol{x},y)\in\mathcal{D}}\frac{y}{\|\boldsymbol{w}\|}\left(\boldsymbol{w}^\top\boldsymbol{x}+b\right)$，对线性不可分数据引入松弛变量后

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}}\frac{1}{2}\|\boldsymbol{w}\|^2+C\sum_{i=1}^n\xi_i\text{ s.t. }\forall i,y^{(i)}\left(\boldsymbol{w}^\top\boldsymbol{x}^{(i)}+b\right)\ge1-\xi_i\text{ and }\boldsymbol{\xi}\ge\boldsymbol{0}$$

使用 Lagrange 对偶，Slater 条件成立，等价于优化

$$\begin{aligned}&\max_{\boldsymbol{\alpha}}\left(\sum_{i=1}^n\alpha_i-\frac{1}{2}\sum_{i=1}^n\sum_{j=1}^n\alpha_i\alpha_jy^{(i)}y^{(j)}\langle\boldsymbol{x}^{(i)},\boldsymbol{x}^{(j)}\rangle\right)\\&\text{s.t. }\boldsymbol{0}\le\boldsymbol{\alpha}\le C\boldsymbol{1}\text{ and }\sum_{i=1}^n\alpha_iy^{(i)}=0\end{aligned}$$

核函数：非线性，对应维数要足够大。**Mercer** 条件：若对于数据集有 $\boldsymbol{K}=\left[K\left(\boldsymbol{x}^{(i)},\boldsymbol{x}^{(j)}\right)\right]_{ij}$ 半正定（特征值非负），则可以作为核函数。

聚类/降维

无监督学习：没有标签，探索数据本身内在结构。聚类/降维。

相似度衡量：选定距离 $d(\boldsymbol{x},\boldsymbol{y})$ 后 $\text{aff}(\boldsymbol{x},\boldsymbol{y})=\exp\left(-\frac{d(\boldsymbol{x},\boldsymbol{y})}{2\sigma^2}\right)$。

K-Means 前辈：类别 $\{H_i|i\in[k]\}$，中心 $\boldsymbol{m}_i=\frac{1}{|H_i|}\sum\limits_{\boldsymbol{x}\in H_i}\boldsymbol{x}$，则损失函数为 $J=\sum\limits_{i\in[k]}\sum\limits_{\boldsymbol{x}\in H_i}\|\boldsymbol{x}-\boldsymbol{m}_i\|^2$。样本 $\hat{\boldsymbol{x}}$ 从 H_i 移动到 H_j 则 $\boldsymbol{m}_j^{\text{new}}=\boldsymbol{m}_j+\frac{\hat{\boldsymbol{x}}-\boldsymbol{m}_j}{|H_j|+1}$，$\boldsymbol{m}_i^{\text{new}}=\boldsymbol{m}_i+\frac{\boldsymbol{m}_i-\hat{\boldsymbol{x}}}{|H_i|-1}$，$J_j^{\text{new}}=J_j+\frac{|H_j|}{|H_j|+1}\|\hat{\boldsymbol{x}}-\boldsymbol{m}_j\|^2$，$J_i^{\text{new}}=J_i-\frac{|H_i|}{|H_i|-1}\|\hat{\boldsymbol{x}}-\boldsymbol{m}_i\|^2$。

K-Means：E-step 固定聚类中心，第 i 个样本分类为 $C_i=\arg\min_j\|\boldsymbol{x}_i-\boldsymbol{m}_j\|$，M-step 固定类别更新 $\boldsymbol{m}_j$。损失函数 $J=\sum_i\|\boldsymbol{x}_i-\boldsymbol{m}_{C_i}\|^2$，则 E-step 和 M-step 均确保 J 下降，且 J 有界，必然收敛。缺陷为所有数据明确分类，out-lier 影响很大，解决方法为 GMM。

层次聚类 (agglomerative clustering)：自底向上进行聚类，合并的过程可以记录为树状结构（使用 Huffman 树），由此灵活决定总类别数。类间相似性可以定义为平均/最大/最小/中心距离等。

高斯混合模型：K 个 Gaussian，概率密度为 $p(\boldsymbol{x})=\sum_{k=1}^K\pi_k\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)$，其中 $\sum_k\pi_k=1$。则对数似然 $\ell(\boldsymbol{\theta})=\sum_{n=1}^N\ln\left(\sum_{k=1}^K\pi_k\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)\right)$。引入隐变量 $\boldsymbol{z}$，则有 $p(z_k=1|\boldsymbol{x})=\frac{\pi_k\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)}{\sum_j\pi_j\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j)}$。对给定的 $\gamma(z_{nk})$ 优化 $\ell(\boldsymbol{\theta})$ 即有

$$\begin{aligned}&\pi_k^{\text{new}}=\frac{1}{N}\sum_{n=1}^N\gamma(z_{nk}),\boldsymbol{\mu}_k^{\text{new}}=\frac{\sum_{n=1}^N\gamma(z_{nk})\boldsymbol{x}_n}{\sum_{n=1}^N\gamma(z_{nk})},\\&\boldsymbol{\Sigma}_k^{\text{new}}=\sum_{n=1}^N\gamma(z_{nk})\left(\boldsymbol{x}_n-\boldsymbol{\mu}_k^{\text{new}}\right)\left(\boldsymbol{x}_n-\boldsymbol{\mu}_k^{\text{new}}\right)^\top/\sum_{n=1}^N\gamma(z_{nk})\end{aligned}$$

于是 E-step 为更新 $\gamma(z_{nk})$，M-step 为计算 π_k^{new}、$\boldsymbol{\mu}_k^{\text{new}}$、$\boldsymbol{\Sigma}_k^{\text{new}}$。无法处理非凸数据。

谱聚类：找中心转变为找邻居，转化为图割问题。先定义距离 $s_{ij}=\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2$ 或 $w_{ij}=\exp\left(-\frac{\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2}{2\sigma^2}\right)$，由此构图 (1)$W_{ij}=\begin{cases}0,&s_{ij}>\epsilon\\\epsilon,&s_{ij}\le\epsilon\end{cases}$，(2)$W_{ij}=s_{ij}$，(3)$W_{ij}=W_{ji}=\begin{cases}0,&\boldsymbol{x}_i\notin\text{KNN}(\boldsymbol{x}_j)\vee\boldsymbol{x}_j\notin\text{KNN}(\boldsymbol{x}_i)\\w_{ij},&\boldsymbol{x}_i\in\text{KNN}(\boldsymbol{x}_j)\wedge\boldsymbol{x}_j\in\text{KNN}(\boldsymbol{x}_i)\end{cases}$。则图割 $\text{Cut}(A,B)=$

$\sum_{i\in A}\sum_{j\in B}W_{ij}$，阶矩 $\boldsymbol{D}=\operatorname{diag}\left\{\sum_j W_{1j},...,\sum_j W_{nj}\right\}$，Laplace 矩阵 $\boldsymbol{L}=\boldsymbol{D}-\boldsymbol{W}$，定义 $\boldsymbol{x}$ 其中 x_j 表示是否属于类 A，则有 $\operatorname{cut}(A,\bar{A})=\boldsymbol{x}^\top\boldsymbol{L}\boldsymbol{x}$。

求解图割问题：定义 $\operatorname{vol}(A)=\sum_{i\in A,j\in A}W_{ij}$，则目标为最小化 $\operatorname{Ncut}(A_1,...,A_k)=\frac{1}{2}\sum_i\frac{\operatorname{cut}(A_i,\bar{A}_i)}{\operatorname{vol}(A_i)}$ 或 $\operatorname{Ratiocut}(A_1,...,A_k)=\frac{1}{2}\sum_i\frac{\operatorname{cut}(A_i,\bar{A}_i)}{|A_i|}$。

<u>Thm</u>： Laplace 矩阵满足对于任意 $\boldsymbol{f}\in\mathbb{R}^n$，$\boldsymbol{f}^\top\boldsymbol{L}\boldsymbol{f}=\frac{1}{2}\sum_{i,j}W_{ij}(f_i-f_j)^2$。

令 $\boldsymbol{f}\in\mathbb{R}^n$ 满足 $f_i=\begin{cases}\sqrt{\operatorname{vol}(\bar{A})/\operatorname{vol}(A)}, & \boldsymbol{x}_i\in A\\ -\sqrt{\operatorname{vol}(A)/\operatorname{vol}(\bar{A})}, & \boldsymbol{x}_i\notin A\end{cases}$，$\boldsymbol{f}^\top\boldsymbol{L}\boldsymbol{f}=\operatorname{Ncut}(A,\bar{A})\operatorname{vol}(V)$，因此优化问题转变为 $\min_A\boldsymbol{f}^\top\boldsymbol{L}\boldsymbol{f}$，为了进行松弛化，引入两个约束 $(\boldsymbol{D}\boldsymbol{f})^\top\mathbf{1}=0$ 以及 $\boldsymbol{f}^\top\boldsymbol{D}\boldsymbol{f}=\operatorname{vol}(V)$，即求解

$$\min_{\boldsymbol{f}}\boldsymbol{f}^\top\boldsymbol{L}\boldsymbol{f}\text{ s.t. }(\boldsymbol{D}\boldsymbol{f})^\top\mathbf{1}=0\text{ and }\boldsymbol{f}^\top\boldsymbol{D}\boldsymbol{f}=\operatorname{vol}(V)$$

引入 $\boldsymbol{g}=\boldsymbol{D}^{0.5}\boldsymbol{f}$ 则有

$$\min_{\boldsymbol{g}}\boldsymbol{g}^\top\boldsymbol{L}_{\text{sym}}\boldsymbol{g}\text{ s.t. }\boldsymbol{g}^\top(\boldsymbol{D}\mathbf{1})=0\text{ and }\boldsymbol{g}^\top\boldsymbol{g}=\operatorname{vol}(V)$$

其中 $\boldsymbol{L}_{\text{sym}}=\boldsymbol{D}^{-0.5}\boldsymbol{L}\boldsymbol{D}^{-0.5}$ 为规范化 Laplace 矩阵。求解只需要解最小 k 个特征值对应的特征向量 $\boldsymbol{g}_1,...,\boldsymbol{g}_k$，进而给出 $\boldsymbol{f}_i=\boldsymbol{D}^{-0.5}\boldsymbol{g}_i$。根据前述定义的 $\boldsymbol{f}$ 的含义，同一聚类中的 f 值应该相似，因此只要对 $\boldsymbol{F}=[\boldsymbol{f}_1,...,\boldsymbol{f}_k]\in\mathbb{R}^{n\times k}$ 作 K-Means 聚类即可。

PCA 降维：给定 $\{\boldsymbol{x}^{(i)}\in\mathbb{R}^m|i\in[n]\}$ 满足均值为 $\mathbf{0}$，希望找到正交的 $\boldsymbol{W}$ 作映射 $\boldsymbol{y}=\boldsymbol{W}^\top\boldsymbol{x}$ 使得 $\boldsymbol{\epsilon}=\boldsymbol{x}-\boldsymbol{W}\boldsymbol{y}=\boldsymbol{x}-\boldsymbol{W}\boldsymbol{W}^\top\boldsymbol{x}$ 最小。则对于数据集而言，

$$\min_{\boldsymbol{W}}\sum_i\boldsymbol{x}^{(i)\top}\boldsymbol{x}^{(i)}-\boldsymbol{x}^{(i)\top}\boldsymbol{W}\boldsymbol{W}^\top\boldsymbol{x}^{(i)}\Leftrightarrow\min_{\boldsymbol{W}}\operatorname{tr}\left(\boldsymbol{W}^\top\left(\sum_i\boldsymbol{x}^{(i)}\boldsymbol{x}^{(i)\top}\right)\boldsymbol{W}\right)$$

只需要令 $\boldsymbol{W}=[\boldsymbol{w}_1,...,\boldsymbol{w}_k]$ 中每一个向量满足 $\boldsymbol{w}_i^\top\left(\sum_i\boldsymbol{x}^{(i)}\boldsymbol{x}^{(i)\top}\right)\boldsymbol{w}_i$ 最小即可。即找最小特征值。保留的方差比例等于 $\sum_{i=1}^k\lambda_i/\sum_{i=1}^d\lambda_i$。

Multi-Dimensional Scaling：同样要求均值为 $\mathbf{0}$，保持降维后距离和原距离相同，则先计算得到 $\boldsymbol{T}=\tilde{\boldsymbol{X}}^\top\tilde{\boldsymbol{X}}$

$$t_{ij}=-\frac{1}{2}\left(d_{ij}^2-\tilde{\boldsymbol{x}}_i^2-\tilde{\boldsymbol{x}}_j^2\right)=-\frac{1}{2}\left(d_{ij}^2-\frac{1}{n}\sum_k d_{ik}^2-\frac{1}{n}\sum_k d_{jk}^2+\frac{1}{n^2}\sum_{k,l}d_{kl}^2\right)$$

再分解 $\boldsymbol{T}=\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\top$ 得到 $\boldsymbol{x}=\boldsymbol{U}\boldsymbol{\Lambda}^{0.5}$。

Isometric Feature Mapping：对非线性结构，距离不再合理，如"瑞士卷"。首先构造邻接图，KNN 或范围内全连接，然后用 Dijkstra 算出 Geodesic 测地距离，接着用 MDS 求解即可。

Locally Linear Embedding：假设为每个样本可以被邻居线性重建，即

$$\min_{\boldsymbol{W}}\sum_i\left\|\boldsymbol{x}^{(i)}-\sum_j W_{ij}\boldsymbol{x}^{(j)}\right\|^2\text{ s.t. }W_{ij}=0\text{ for non-neighbour }i,j\text{s and }\sum_j W_{ij}=0$$

求解 $\boldsymbol{W}$ 后在低维度求解

$$\min_{\boldsymbol{y}^{(i)}}\left\|\boldsymbol{y}^{(i)}-\sum_j W_{ij}\boldsymbol{y}^{(j)}\right\|^2\text{ s.t. }\sum_i\boldsymbol{y}^{(i)}=\mathbf{0}\text{ and }\sum_i\boldsymbol{y}^{(i)}\boldsymbol{y}^{(i)\top}=N\boldsymbol{I}$$

优化函数写为 $\operatorname{tr}\left(\boldsymbol{Y}(\boldsymbol{I}-\boldsymbol{W})^\top(\boldsymbol{I}-\boldsymbol{W})\boldsymbol{Y}^\top\right)$ 则同样变为求解 d 个最小特征值的特征向量。LLE 需要更多训练数据，计算量大，与 ISOMAP 一样没有显式映射函数，难以使用。

集成学习

单个算法能力不足/训练数据不够，因此训练很多弱分类器，数据复用。**Bagging**: 多次 Bootstrapping 得到 k 个分类器投票，使得 bias 不变但是 variance 变为 $1/k$。**随机森林**：对于 N 个 d 维样本，每次用 bootstrap 采样得到一个训练集训练一棵树。每次分裂都随机选择 $m\ll M$ 个特征，在这 m 个特征中寻找最大信息增益。

个体学习器：**同质/异质**。同质：存在强依赖（因此串行）/不存在强依赖（并行）。

Boosting：强学习算法（多项式时间学习，正确率很高）和弱学习算法（仅比随机猜测好）是等价的，因此只要找到弱学习算法就可以构造强的。先每个样本相同权重，每次迭代后对错误样本加大权重（重采样）。因此只降低 bias。

AdaBoost：对于二分类，初始化 $D_1(i)=\frac{1}{m}$ 为均等权重，对于当前权重求解最优分类器 $\hat{h}=\arg\min_h\epsilon_h=\arg\min_h\sum_{i=1}^m D_t(i)\times\mathbf{1}\left\{y^{(i)}\neq h(\boldsymbol{x}^{(i)})\right\}$，当前分类器的权重 $\alpha_t=\frac{1}{2}\ln\frac{1-\epsilon_{\hat{h}}}{\epsilon_{\hat{h}}}$，根据 $D_{t+1}(i)\propto D_t(i)\exp\left(-\alpha_t y^{(i)}h_i\left(\boldsymbol{x}^{(i)}\right)\right)$，最终结果输出为 $\hat{y}=\operatorname{sgn}\left(\sum_{t=1}^T\alpha_t h_t(\boldsymbol{x})\right)$。在每次权重更新后采用轮盘赌算法重新采样训练。若某次训练误差大于 $1/2$ 则直接抛弃。在这里多次训练后仍然能增大 margin 因此不会过拟合（bias-variance 不能解释）。

AdaBoost 的损失函数 $J=\sum_{i=1}^m\exp\left(-y^{(i)}f_t(\boldsymbol{x}^{(i)})\right)$，其中 f_t 为第 t 步的加权模型输出，$f_t\left(\boldsymbol{x}^{(i)}\right)=f_{t-1}\left(\boldsymbol{x}^{(i)}\right)+\alpha_t h_t\left(\boldsymbol{x}^{(i)}\right)$，则求导可得 $\alpha_t=\frac{1}{2}\ln\frac{1-\epsilon_t}{\epsilon_t}$。

Gradient Boosting Decision Tree：用于回归任务，输入 $\{(\boldsymbol{x}^{(i)},y^{(i)})\in\mathbb{R}^d\times\mathbb{R}|i\in[N]\}$，损失函数 $L(y,f(\boldsymbol{x}))$。初值取 $F_0(\boldsymbol{x})=\arg\min_c\sum_{i=1}^N L(y^{(i)},c)$，对于 $m\in[M]$ 重复使用 $\tilde{y}^{(i)}=-\left(\frac{\partial L\left(y^{(i)},F\left(\boldsymbol{x}^{(i)}\right)\right)}{\partial F\left(\boldsymbol{x}^{(i)}\right)}\right)_{F=F_{m-1}}$ 计算新的标签，对新的标签拟合新的回归树并为每个叶节点分配使得 L 最小的值，得到决策树 $h_m(\boldsymbol{x})$，更新 $F_m(\boldsymbol{x})=F_{m-1}(\boldsymbol{x})+h_m(\boldsymbol{x})$。因为回归中一般选择 L 为 MSE，所以新的标签为残差。

Extreme Gradient Boosting：引入正则化、二阶导数，因此提升效率。第 t 步

$$\text{obj}^t=\sum_{i=1}^N L\left(y^{(i)},\hat{y}_i^{t-1}+f_t\left(\boldsymbol{x}^{(i)}\right)\right)+\sum_{k=1}^t\Omega(f_k)$$

其中 $\Omega(f_t)=\gamma T+\frac{\lambda}{2}\sum_{j=1}^T w_j^2$ 表示第 t 个决策树的复杂程度，T 为总叶节点数，w_j 为第 j 个叶节点的值，I_j 为属于叶节点 j 的样本集合。前一项可以 Taylor 展开，令 g_i 为一阶导，h_i 为二阶导，则最小化

$$\text{obj}^t=\sum_{j=1}^T\left(\left(\sum_{i\in I_j}g_i\right)w_j+\frac{1}{2}\left(\sum_{i\in I_j}h_i+\lambda\right)w_j^2\right)+\gamma T$$

令 $G_j=\sum_{i\in I_j}g_i$，$H_j=\sum_{i\in I_j}h_i$，容易解得 $w_j^*=-\frac{G_j}{H_j+\lambda}$，$\text{obj}^{t*}=-\frac{1}{2}\sum_{j=1}^T\frac{G_j^2}{H_j+\lambda}+\gamma T$。

LightBGM：XGBoost 的优化。**互斥特征捆绑**：对于多个稀疏的特征进行捆绑，如特征 $x_1\sim x_3$ 均稀疏，样本 $\boldsymbol{x}$ 只在第 i 个特征有值 a，则捆绑后特征为 $10i+a$。**连续特征离散化**：直方图或排序 → 计算累积量。**类别特征最优分裂**：对于类别特征一般使用 one-hot，则为 one-vs-rest，效率低，改用 many-vs-many，如判断标准为 $X=1\vee X=2$。**并行优化**：原有优化为特征并行或数据并行，LightBGM 提出选举并行，即每个处理器处理一部分样本找 Top k 特征，再在中央加权找全局 Top k 特征；每个处理器接受到全局 Top k 后计算局部直方图，在中央合并为全局直方图，找到全局最优的分裂点。