模式识别与机器学习

期末复习

说明

- 专业硕士班级不考察“概率图模型”相关章节(P40)
- 学术硕士班级仅了解不做考试要求:“特征工程”“测试和评价”“项目开发”等相关章节(P5，P11-13，P28-29)

模式识别与机器学习

- 模式识别 vs. 机器学习：一体两面，不做区分

- 模式识别：利用计算机对物理对象进行分类，在风险最低的条件下，使识别的结果尽量与真实情况相符合
  - 在特征空间和解释空间之间找到一种映射关系：$y = f(\boldsymbol{x})$

- 机器学习：利用大量的训练数据，获得产生数据的模式f并进行预测

风险和错误率

■错误率：$P(error) = \int P(error|\boldsymbol{x})p(\boldsymbol{x})d\boldsymbol{x}$

- 无需对$\boldsymbol{x}$积分，对每个$\boldsymbol{x}$都求错误率最小的决策，就是求对所有$\boldsymbol{x}$平均错误率最小的决策
- $P(error|\boldsymbol{x}) = 1 - P(Y = y|\boldsymbol{x})$

■风险：平均损失

■损失L_{cy}：将本应属于y类的模式判别成属于c类的代价

$$R(\hat{y}(\boldsymbol{x}) = c|\boldsymbol{x}) = \sum_{y=1}^{C} L_{cy}P(Y = y|\boldsymbol{x})$$

其他评价指标

■回归：RMSE、R2分数、…

$$\mathrm{RMSE}\ (\hat{\boldsymbol{y}}, \boldsymbol{y}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2} \qquad R^2(\hat{\boldsymbol{y}}, \boldsymbol{y}) = 1 - \frac{SS_{res}\ (\boldsymbol{y}, \hat{\boldsymbol{y}})}{SS_{tot}(\boldsymbol{y})} = 1 - \frac{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}{\frac{1}{N}\sum_{i=1}^{N}(\bar{y} - y_i)^2}$$

■分类：正确率、ROC/AUC、P/R，F1、…

$$\mathrm{Accuracy}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \frac{1}{N}\sum_{i=0}^{\mathrm{N}} I(\hat{y}_i = y_i) \qquad P = \frac{TP}{\widehat{N}_+},\ R = \frac{TP}{N_+},\ F1 = \frac{2(P \times R)}{P + R}$$

■聚类：轮廓系数、…

$$SI = \frac{1}{N}\sum_{i=1}^{N} s(i)\,, s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

$a(i)$：样本点i到与其所属簇中其它点的平均距离

$\overline{d(i, C_k)}$：样本点i到其他簇$C_k(\boldsymbol{x}_i \notin C_k)$内所有点的平均距离

$b(i)$：所有$\overline{d(i, C_k)}$的最小值

机器学习的3个方面

- 函数族$\mathcal{F}$ ： $f \in \mathcal{F}$
- 目标函数$J(f)$ ：度量f的好坏
- 优化算法： $f^* = \arg\min_f J(f)$

1. 函数族

- 线性模型： $f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$
- 多项式
- 核方法（由核函数决定）
- 决策树（分段常数）
- 神经网络（由网络结构决定）
- 集成学习：多棵决策树的加权平均
  - 随机森林
  - GBDT
- 概率图模型：利用条件独立假设，简化概率计算

- 参数模型：线性模型、多项式、神经网络、概率图模型
- 非参数模型：核方法、决策树、集成学习

2. 目标函数

- $J(f) = L(f) + \lambda R(f)$

- 损失函数$L(f)$：函数f与训练数据的拟合程度
  - 负log似然损失：L2/L1损失、交叉熵损失
  - 合页损失（SVM）、ε不敏感损失（SVR）
  - 聚类损失：K均值、谱聚类
  - 降维损失：结构保持、重构
- 正则项$R(f)$：函数f自身的复杂程度
  - L2正则、L1正则
  - K-Lipschitz 连续（WGAN）
- 正则参数λ
  - λ越小，模型越复杂，训练误差越小，偏差小、方差大
  - λ越大，模型越简单，训练误差越大，偏差大、方差小

模型的其他复杂度参数

- 线性模型：特征的数目
- 多项式：多项式阶数
- 核方法：核函数超参数（RBF核的核函数宽度）
- 决策树（树的最大深度、叶子结点数目、叶子结点代表的训练样本数目、...）
- 神经网络
  - 网络的连接方式：全连接、局部连接、...
  - 网络的层数
  - 每层的神经元的数目

3. 优化算法

- $f^* = \arg\min_f J(f)$
- 梯度下降/上升
  - 梯度：
    常用损失函数的梯度计算
    批处理梯度下降、随机梯度下降、小批量梯度下降
    动量法
  - 学习率
    学习率的影响
    自适应学习率
- 坐标下降/上升：每次选择一个或一部分参数更新
  - K均值聚类
  - EM
  - Lasso、SMO

前处理：特征工程

- 数据探索性分析
- 特征编码
- 特征预处理
- 特征选择

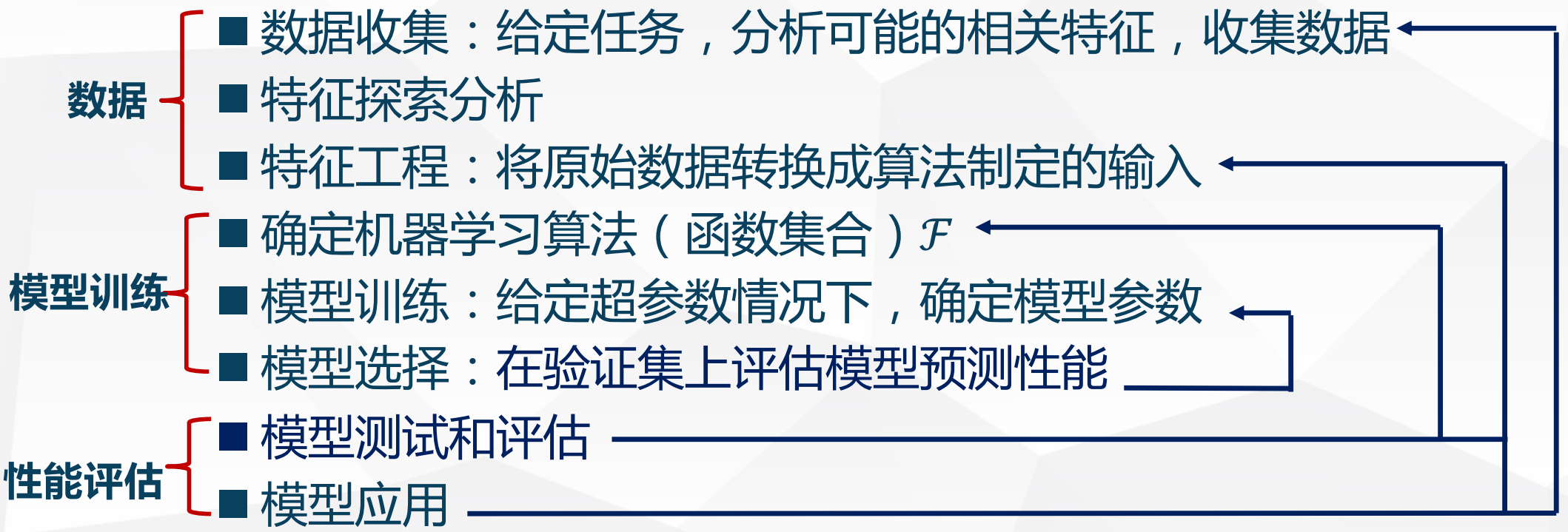
- 没有免费的午餐
  - 特征工程与机器学习模型一起考虑

后处理：模型选择和模型评估

- 训练集、验证集、测试集划分
- 训练集和验证集
  - 留出法
  - 交叉验证
- 训练误差、验证误差、测试误差
- 统计学习理论
  - 奥卡姆剃刀原理

机器学习项目的开发过程

$$y = f(\boldsymbol{x})$$



■生产式分类器：$P(y=c|\boldsymbol{x})=\frac{p(\boldsymbol{x}|y=c)P(y=c)}{\sum_{c'}p(\boldsymbol{x}|y=c')P(y=c')}$

■类先验$P(y=c)$

- 两类： $y\sim\text{Bernoulli}(\theta)$
- 多类 ：$y\sim\text{Multinoulli}(\boldsymbol{\theta})$

■类条件$p(\boldsymbol{x}|y=c)$

- **高斯判别分析**： $p(\boldsymbol{x}|y=c)=N(\boldsymbol{\mu}_c,\boldsymbol{\Sigma}_c)$
- 朴素贝叶斯： $p(\boldsymbol{x}|y=c)=\prod_{j=1}^{D}p(x_j|y=c)$
- 深度生成模型：学习从简单分布（高斯）到$p(\boldsymbol{x}|y=c)$的映射（神经网络）

高斯判别分析

- $P(Y=c|\boldsymbol{x})=\dfrac{p(\boldsymbol{x}|Y=c)P(Y=c)}{\sum_{c'}p(\boldsymbol{x}|Y=c')P(Y=c')}$

$$p(\boldsymbol{x}|Y=c)=\frac{1}{(2\pi)^{\frac{D}{2}}|\boldsymbol{\Sigma}_c|^{\frac{1}{2}}}\exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_c)^{\mathrm{T}}\boldsymbol{\Sigma}_c^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_c)\right)$$

- 类别c的判别函数： $f_c(\boldsymbol{x})=-\frac{1}{2}\ln(|\boldsymbol{\Sigma}_c|)-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_c)^{\mathrm{T}}\boldsymbol{\Sigma}_c^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_c)+\ln P(Y=c)$
- 当$\boldsymbol{\Sigma}_1=\boldsymbol{\Sigma}_2=\boldsymbol{\Sigma}$时，

$$f_1(\boldsymbol{x})-f_2(\boldsymbol{x})=-\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}(\boldsymbol{\Sigma}^{-1}-\boldsymbol{\Sigma}^{-1})\boldsymbol{x}+\left(\boldsymbol{\mu}_1^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}-\boldsymbol{\mu}_2^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\right)\boldsymbol{x}+b+\ln\frac{P(Y=1)}{P(Y=2)}$$

$$=(\boldsymbol{\mu}_1^{T}-\boldsymbol{\mu}_2^{T})\boldsymbol{\Sigma}^{-1}\boldsymbol{x}+b+\ln\frac{P(Y=1)}{P(Y=2)}$$

$$b=-\frac{1}{2}\boldsymbol{\mu}_1^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_1+\frac{1}{2}\boldsymbol{\mu}_2^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_2$$

- 函数族：线性函数 $f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$
- 目标函数&优化算法

名称	目标函数	优化算法
Fisher判别分析	$J(\boldsymbol{w}) = \dfrac{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_b\boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_w\boldsymbol{w}}$	解析解 $\boldsymbol{w} = \boldsymbol{S}_w^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$
感知器	$J(\boldsymbol{w}) = \sum_{\boldsymbol{x}_i \in \mathcal{M}} -y_i f(\boldsymbol{x}_i)$	随机梯度下降

- 两类Fisher判别的判别面为：$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} = \dfrac{1}{2}\boldsymbol{w}^{\mathrm{T}}(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2)$
  - 投影后的类中心连线的垂直平分线为判别面 $(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^{\mathrm{T}}\boldsymbol{S}_w^{-1}\boldsymbol{x} = \dfrac{1}{2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^{\mathrm{T}}\boldsymbol{S}_w^{-1}(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2)$

第3章 判别式分类器

- 决策树：分段常数函数
- 损失函数：
  - 分类：Gini指数
  - 回归：L2损失
- 正则：
  - 叶子结点数目
- 优化算法：贪心
  - 精确穷举搜索
  - 直方图近似搜索

■函数族：线性函数 $\hat{y} = f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$

■损失函数

- L2损失： $L(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$
- Huber损失：$L_\delta(\hat{y}, y) = \begin{cases} \frac{1}{2}(\hat{y} - y)^2 & |r| \le \delta \\ \delta|\hat{y} - y| - \frac{1}{2}\delta^2 & otherwise \end{cases}$

■优化算法

- 解析法
- 梯度下降
- 坐标下降（了解）

$$\frac{\partial L}{\partial \boldsymbol{w}} = \frac{\partial L}{\partial f}\frac{\partial f}{\partial \boldsymbol{w}} = (\hat{y} - y)\boldsymbol{x}$$

第6章 Logistic回归

- 函数族：线性函数

$$\mu(\boldsymbol{x}) = \sigma\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}\right)$$

$$p(y|\boldsymbol{x};\mu) = \mu(\boldsymbol{x})^{y}\left(1-\mu(\boldsymbol{x})\right)^{(1-y)}$$

- 损失函数
  - 交叉熵损失/负log似然损失

$$L(\mu(\boldsymbol{x}), y) = -y\ln\left(\mu(\boldsymbol{x})\right) - (1-y)\ln\left(1-\mu(\boldsymbol{x})\right)$$

- 优化算法
  - 梯度下降
  - 牛顿法（了解）

$$\frac{\partial L}{\partial \boldsymbol{w}} = \frac{\partial L}{\partial \mu}\frac{\partial \mu}{\partial \boldsymbol{w}} = \left(-\frac{y}{\mu(\boldsymbol{x})} + \frac{1-y}{1-\mu(\boldsymbol{x})}\right)\mu(\boldsymbol{x})\left(1-\mu(\boldsymbol{x})\right)\frac{\partial\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}\right)}{\partial \boldsymbol{w}}$$

$$= (\hat{y} - y)\boldsymbol{x}$$

- 多类分类Logistic回归（softmax分类器）：了解

第7章 SVM

- 分类器的间隔：$\frac{2}{\|\boldsymbol{w}\|_2}$
- 线性SVM
  - 硬间隔/软间隔
  - 支持向量
  - 合页损失
  - 原问题 vs. **对偶问题**
- 核化SVM
  - 常用核函数及其复杂度参数
- SVR

■硬线性SVM

- 原问题

$$\min \frac{1}{2}\|\boldsymbol{w}\|_2^2$$

$$\text{s.t. } y_i\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b\right) \geq 1, \qquad i = 1,2,\dots,N$$

- **对偶问题**

$$\max\left(\sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\,\alpha_j y_i y_j \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j\right)$$

$$\text{s.t.} \sum_{i=1}^{N}\alpha_i\, y_i = 0$$

$$0 \leq \alpha_i, \qquad i = 1,2,\dots,N$$

■硬线性SVM

- **对偶问题**

$$\max\left(\sum_{i=1}^{N}\alpha_i-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\,\alpha_j y_i y_j \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j\right)$$

$$\text{s.t.}\sum_{i=1}^{N}\alpha_i\,y_i=0$$

$$0\le\alpha_i,\qquad i=1,2,\dots,N$$

- **支持向量：** $\alpha_i\neq 0$

- **对偶解 → 原问题解**

$$\boldsymbol{w}=\sum_{i=1}^{N}\alpha_i\,y_i\boldsymbol{x}_i$$

用任意一个支持向量即可求得b： $b=y_i-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i$

合页损失：$\xi = L_{Hinge}(y, \hat{y}) = \begin{cases} 0 & y\hat{y} \geq 1 \\ 1 - y\hat{y} & otherwise \end{cases}$

■软间隔SVM

• 原问题 $J(\boldsymbol{w}, b, C) = C\sum_{i=1}^{N} \xi_i + \frac{1}{2}\|\boldsymbol{w}\|_2^2$

$$\text{s.t. } y_i\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b\right) \geq 1 - \xi_i, \qquad i = 1,2,\dots,N$$

$$\xi_i \geq 0, \quad i = 1,2,\dots,N$$

•对偶问题 $\max\left(\sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i\,\alpha_j y_i y_j \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j\right)$

$$\text{s.t.} \sum_{i=1}^{N} \alpha_i\, y_i = 0$$

$$0 \leq \alpha_i \leq C, \qquad i = 1,2,\dots,N$$

第4章 特征选择和提取

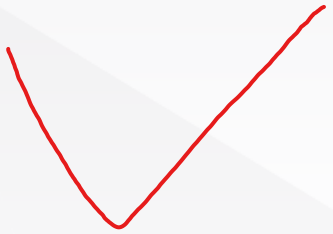
■数据预处理

- 常用数值型特征的预处理/编码方案：log变换、量化
- 常用类别型特征的编码方案：**独热编码**、计数编码、hash编码、嵌入编码
- 特征缩放：**标准化**、正规化

■特征提取：重点掌握 **PCA**

■特征选择：了解

PCA



■模型训练：给定训练样本$\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N)$

- 1. 样本中心化： $\overline{\boldsymbol{x}} = \frac{1}{N}\sum_{j=1}^{N} \boldsymbol{x}_j$ ，$\boldsymbol{x}_i = \boldsymbol{x}_i - \overline{\boldsymbol{x}}$
- 2. 计算$\boldsymbol{S} = \boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}$
- 3. 对$\boldsymbol{S}$做**特征值分解**
- 4. D'最大特征值对应的特征向量: $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots \boldsymbol{w}_{D'}$（ D' 的选择）

输出: $\boldsymbol{W} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots \boldsymbol{w}_{D'})$

■推理/测试：对新的样本$\boldsymbol{x}$，其对应的低维表示为： $\boldsymbol{z} = \boldsymbol{W}^{\mathrm{T}}(\boldsymbol{x} - \overline{\boldsymbol{x}})$

■重构：给定$\boldsymbol{z}$，重构结果为： $\hat{\boldsymbol{x}} = \boldsymbol{W}\boldsymbol{z} + \overline{\boldsymbol{x}}$

第8章 统计学习基础

- 经验风险
- 期望风险
  - 结构风险：经验风险 +正则
- 训练误差、验证误差、测试误差
  - 交叉验证
- 过拟合、欠拟合

- 泛化误差分解：偏差-方差分解
- 泛化误差上界
  - 训练误差$E_{\text{train}}(f^*)$、训练样本数目N、VC维d_{vc}

$$E_{\text{train}}(f^*) - \Omega(d_{\text{vc}}, N, \delta) \leq E_{\text{test}}(f^*) \leq E_{\text{train}}(f^*) + \Omega(d_{\text{vc}}, N, \delta)$$

$$\Omega(d_{\text{vc}}, N, \delta) = \sqrt{\frac{8}{N} \ln\left(4\frac{(2N)^{d_{\text{vc}}} + 1}{\delta}\right)}$$

训练集、验证集、测试集

■验证集：选择模型

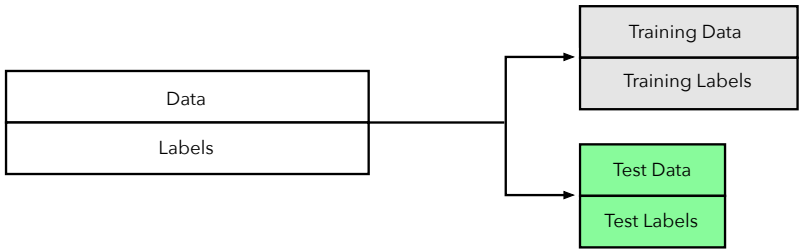
- 用样本外误差，估计测试误差
- 验证误差是真实误差的无偏估计，两者的差距与验证集的大小成反比
- 用不同超参数在验证集上的性能做模型选择（确定最佳超参数）
- 确定超参数后，用全体训练数据（训练集+验证集）训练模型参数
- 当训练数据集较小时，可采用交叉验证方式得到验证集

■测试集：评估模型

- 测试集是从总体选出来的部分样本，与训练集不重合，模拟没有见过但未来可能遇到的数据
- 用测试误差估计真实误差

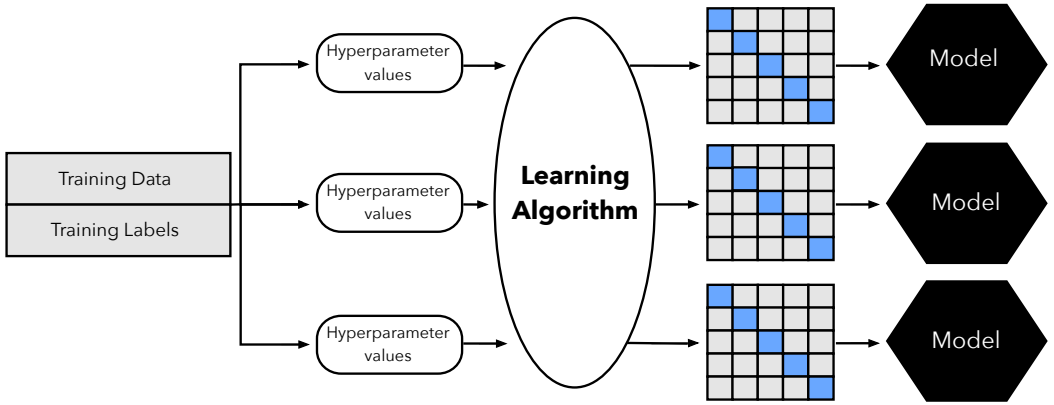
交叉验证

1



留出一部分数据作为测试集

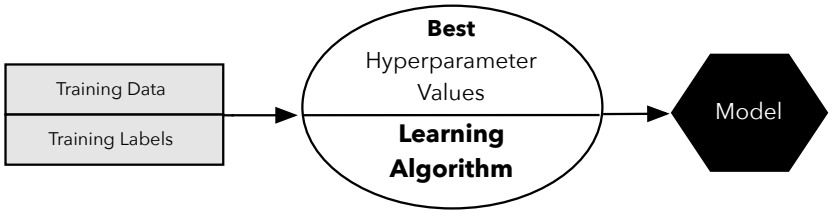
2



对训练集采用K折交叉验证：根据每折的训练数据，训练不同超参数对应的学习算法，得到每个超参数每折交叉验证对应的模型，并在每折的验证集上评估每个超参数对应的模型的性能

交叉验证

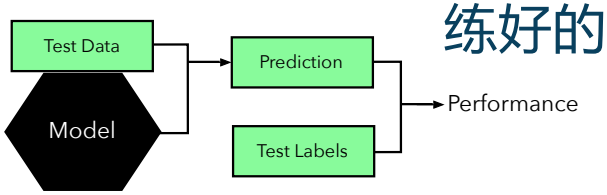
3



对最佳超参数，在所有训练数据上训练模型

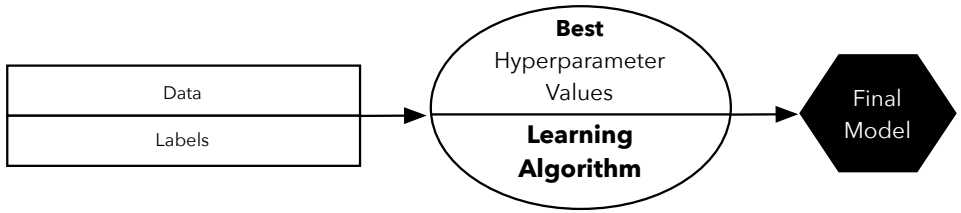
4

对最佳超参数上用所有训练数据训练好的模型在测试集进行性能评估



5

对最佳超参数，在所有数据上训练模型



第9章 集成学习

- **Bagging**：降低方差、偏差不变
  - 基学习器类型相同：通常较复杂，偏差较小，方差较大
  - 通过对样本和特征进行随机采样得到不同的训练集训练各基学习器
  - 模型融合：多个基学习器结果的平均/投票
  - 多个基学习器可并行训练
- **Boosting**：降低偏差、方差不变
  - 基学习器类型相同：通常较简单，偏差较大，方差较小
  - 每次迭代通过改变样本的权重（ AdaBoost ）或改变标签（GBM）得到不同的训集训练各基学习器、样本和特征可随机采样
  - 模型融合：多个基学习器结果的加权平均
  - 多个基学习器顺序训练，不能并行
- Stacking：了解
  - 基学习器类型可以不同

第10章 聚类

- 聚类性能评价指标：了解
  - 簇内距离越小越好，簇间距离越大越好
- 聚类算法
  - **K均值聚类**
  - 高斯混合模型
    - EM算法：了解
  - 层次聚类：了解
  - 基于密度的聚类：DBSCAN
  - 基于图的聚类：了解

K均值聚类

■目标函数： $J = \sum_{i=1}^{N}\sum_{k=1}^{K} r_{i,k}\,||\boldsymbol{x}_i - \boldsymbol{\mu}_k||^2$

■优化算法：坐标下降

- 初始化：随机选择K个簇中心$\boldsymbol{\mu}_k$
- 固定$\boldsymbol{\mu}_k$，最小化$J(r_{i,k})$

$$z_i = \text{argmin}_{k'}\text{dist}(\boldsymbol{x}_i, \boldsymbol{\mu}_{k'})$$

$$\begin{cases} r_{i,k} = 1, k = z_i \\ r_{i,k} = 0, k \neq z_i \end{cases}$$

- 固定$r_{i,k}$，最小化$J(\boldsymbol{\mu}_k)$

$$\boldsymbol{\mu}_k = \frac{\sum_i r_{i,k}\boldsymbol{x}_i}{\sum_i r_{i,k}}$$

第11章 降维

■线性降维

- PCA
- 多维尺度缩放（MDS）：了解

■非线性降维：至少掌握**1种算法**

- 重构残差最小：核化PCA、**自编码器**
- 全局距离保持：ISOMAP
- 邻域距离保持：Laplacian Eigenmaps、T-NSE，UMAP

第12章 半监督学习

- 半监督学习的三**个基本假设**
  - 高密度区域平滑假设
  - 聚类假设/低密度分隔假设
  - 流形假设

- 半监督学习算法：**至少熟悉一种算法**
  - 自我训练
  - 多视角学习
  - 生成模型
  - 半监督SVM
  - 基于图的算法
  - 半监督聚类

第13章 深度学习

■神经元的结构

- **激活函数**：**非线性**、梯度消失、计算简单
- ReLU：最常用
- sigmoid：主要用于门控函数
- tanh：主要用于RNN

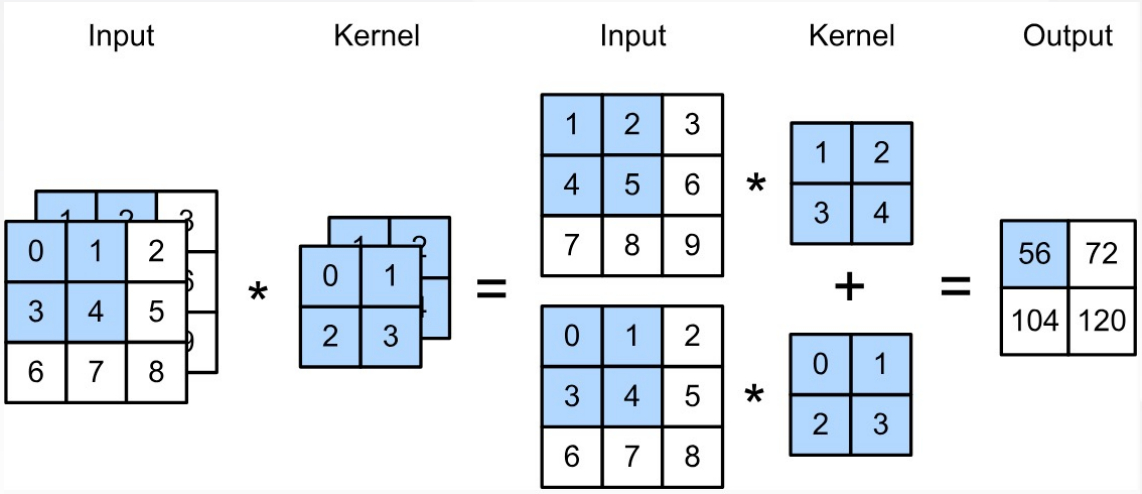
■神经网络结构

- 全连接
- **卷积**：局部连接、权值共享
- 循环：历史信息压缩、梯度消失、梯度爆炸
- **跳跃连接**：缓减梯度消失
- **Transformer**

卷积

■根据数据特点设计的网络结构：局部连接、权值共享

- 平移不变、模式局部相关



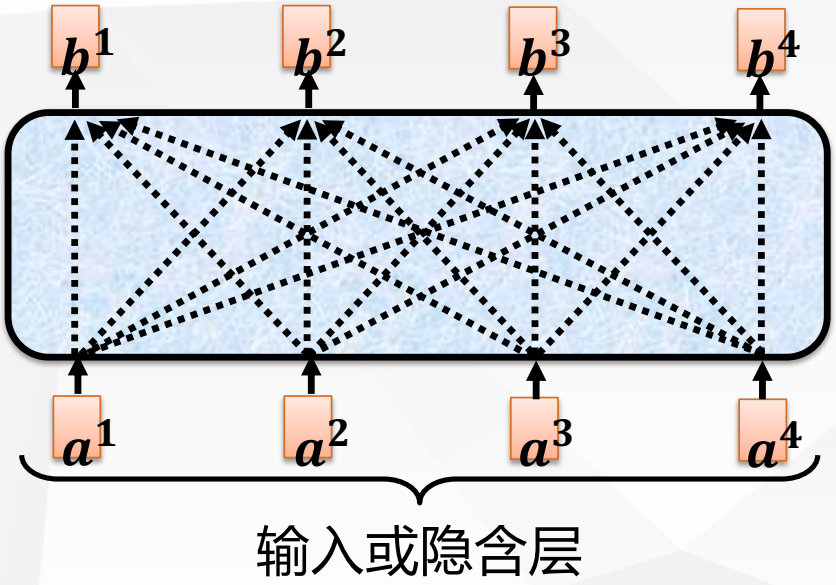
$$(1\times1+2\times2+4\times3+5\times4)$$
$$+(0\times0+1\times1+3\times2+4\times3)$$
$$=56$$

- 输入：$\boldsymbol{X}$: $c_i\times n_h\times n_w$，输出：$\boldsymbol{Y}$: $c_o\times m_h\times m_w$，卷积核大小为$k_h\times k_w$，则卷积核的参数数目为（不考虑偏置项b）：

$$c_o\times c_i\times k_h\times k_w$$

Transformer

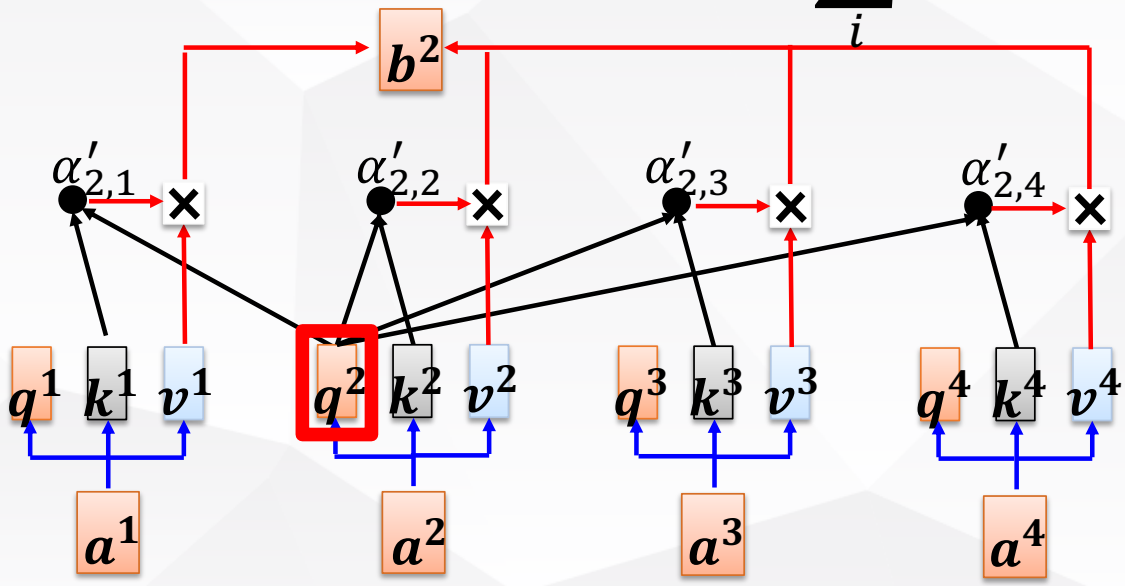
■自注意力



输入或隐含层

$$Q = IW^q$$
$$K = IW^k$$
$$V = IW^v$$

$$b^2 = \sum_i \alpha'_{2,i} v^i$$

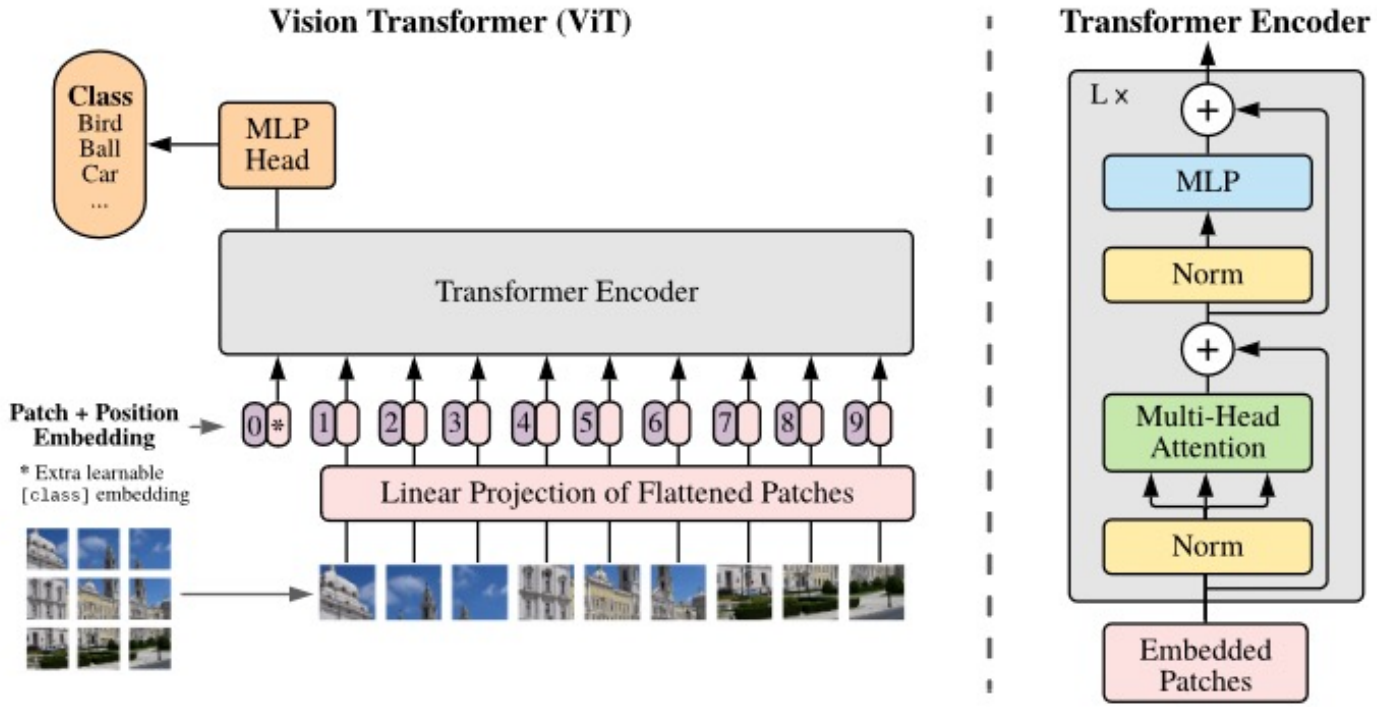


$$\text{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V}$$

$$\text{MultiHead}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \text{concat}(\text{head}_1, \text{head}_2, \ldots, \text{head}_H)\boldsymbol{W}^o$$

例：ViT（ Visual Transformer ）

■网络结构



■模型训练

- 损失函数：交叉熵损失（图片分类）
- 优化算法：Adam

第13章 深度学习

- 神经网络模型的训练：梯度下降
  - **梯度消失**与梯度爆炸
  - 梯度计算：反向传播、批处理梯度下降、随机梯度下降、小批量梯度下降
  - 自适应的梯度下降：动量法、自适应学习率调整
  - 参数初始化
    - 小的随机数：方差的确定
    - **预训练模型**
    - **BN**
- 神经网络抗过拟合
  - 及早停止
  - 正则
  - 数据增广
  - **Dropout**