Lecture8 机器学习概念

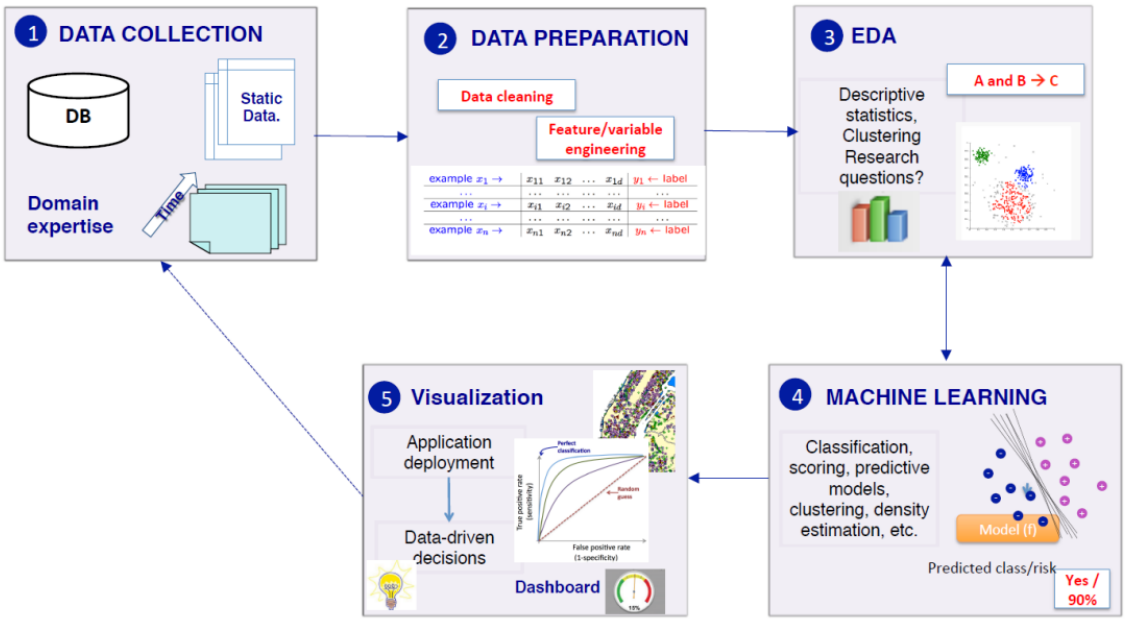
1. 机器学习介绍

数据类型

数据有不同的大小和风格（类型）

- 文本、数字、点击流、图、表、图片、交易、视频...

数据科学的流程



机器学习的应用

- 垃圾邮件过滤
- 信用卡诈骗检测
- 支票、邮政编码上的数字识别
- 人脸检测
- MRI 图像分析
- 推荐系统
- 搜索引擎
- 笔迹识别
- 场景分类

机器学习与统计

统计	机器学习
假设检验 实验设计 方差分析 Anova 线性回归 Linear regression 逻辑回归 Logistic regression GLM（广义拉格朗日乘子） PCA（主成分分析）	SVM（支持向量机） 神经网络 决策树 归纳法 Rule induction 聚类 Clustering 关联规则 Association rules 特征选择 Feature selection 可视化 图模型 遗传算法

机器学习的定义

> “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. "
>
> —— Tom Mitchell. Machine Learning 1997.

2. 监督与非监督 Supervised vs. Unsupervised

给定：训练数据 $(x_1, y_1), \dots (x_n, y_n)$

- $x_i \in \mathbb{R}^d$
  - x_i 代表样本 i，它是一个 d 维的数据
- y_i 是标签
  - y_i 代表样本 i 的标签

example $x_1 \rightarrow$	x_{11}	x_{12}	$\dots$	x_{1d}	$y_1 \leftarrow$ label
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
example $x_i \rightarrow$	x_{i1}	x_{i2}	$\dots$	x_{id}	$y_i \leftarrow$ label
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
example $x_n \rightarrow$	x_{n1}	x_{n2}	$\dots$	x_{nd}	$y_n \leftarrow$ label

fruit	length	width	weight	label
fruit 1	165	38	172	Banana
fruit 2	218	39	230	Banana
fruit 3	76	80	145	Orange
fruit 4	145	35	150	Banana
fruit 5	90	88	160	Orange
...				
fruit n	...	...	...	...

- 监督学习
  - 通过**没有标签**的数据学习一个模型
- 非监督学习
  - 通过**有标签**的数据学习一个模型

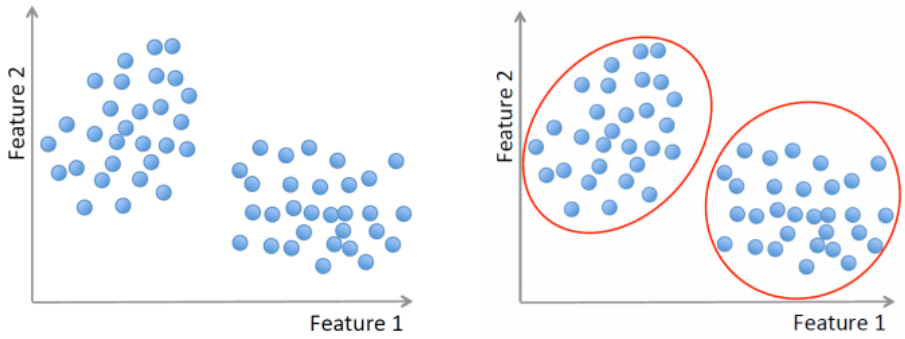
非监督学习

训练数据：样本 x，没有标签 y

$$x_1, \ldots, x_n, x_i \in X \subset \mathbb{R}^d$$

- x_i：样本
- X：样本数据集
- $\mathbb{R}^d$：d 维的实数集（特征）

聚类 Clustering / Segmentation



$$f : \mathbb{R}^d \to \{C_1, \ldots C_k\}$$

- f：聚类函数 $f(x_i)$
- C_i：聚类结果

例如，按照不同维度的参数对水果种类进行聚类

聚类方法

- K-means
- 高斯混合 Gaussian mixtures
- 层次聚类 Hierarchical clustering
- 谱聚类 Spectral clustering

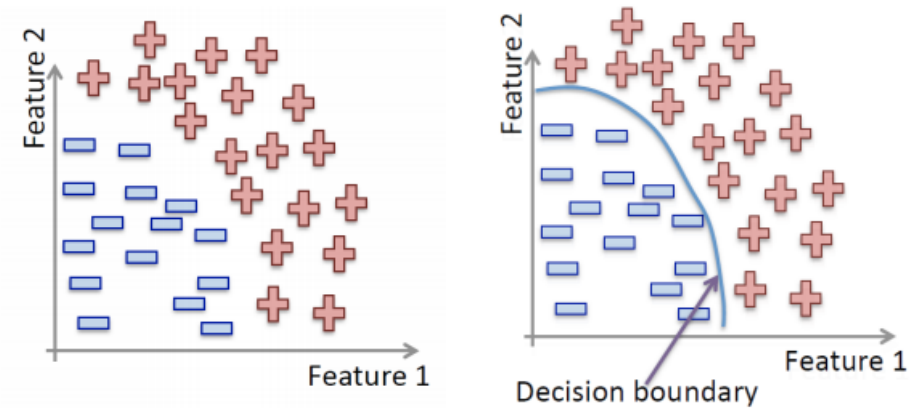
监督学习

训练数据：样本 x，带有标签 y

$$(x_1, y_1), \ldots, (x_n, y_n), x_i \in \mathbb{R}^d$$

- x_i：样本
- y_i：样本的标签
- $\mathbb{R}^d$：d 维的实数集（特征）

分类 Classification



- y 是**离散**的，在这里为了简化，$y \in \{-1, +1\}$

$$f : \mathbb{R}^d \to \{-1, +1\}$$

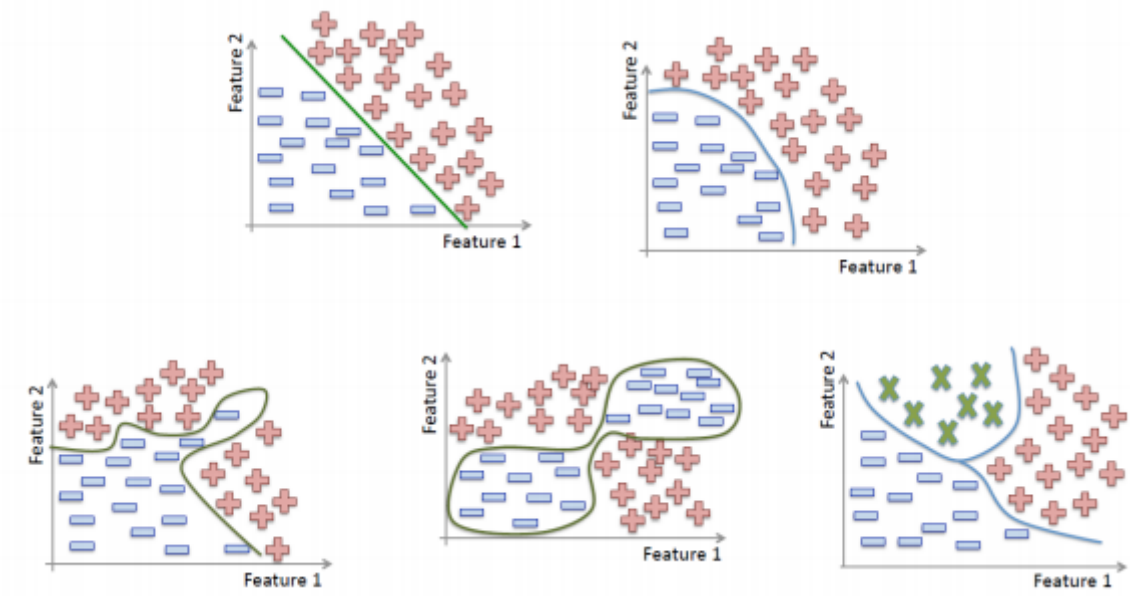
- f：分类函数 $f(x_i)$
- $\{-1, +1\}$：分类结果

例如：判断水果是香蕉 / 橘子

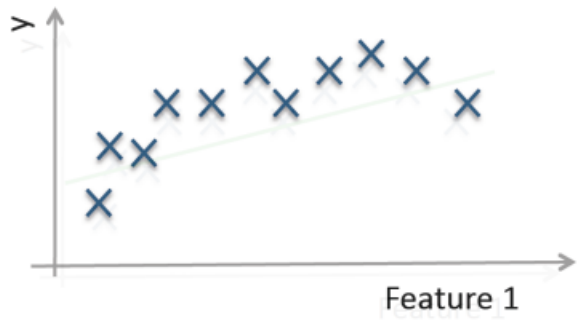
分类方法

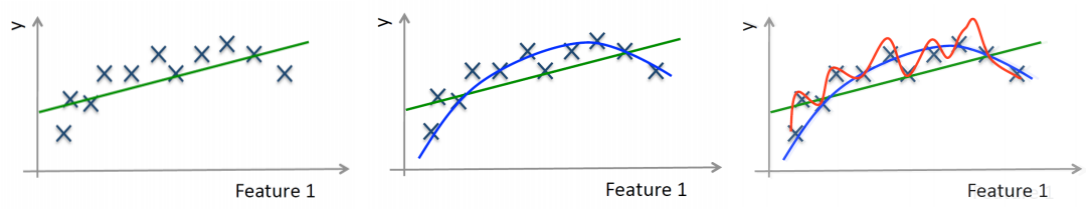
- 支持向量机 Support Vector Machines
- 神经网络 Neural Networks
- 决策树 Decision Tree
- K-近邻 K-nearest neighbors
- 朴素贝叶斯 Naive Bayes

分类效果不同



回归 Regression





- y 是**连续的实数**，$y \in \mathbb{R}$

$$f : \mathbb{R}^d \to \mathbb{R}$$

- f：回归函数 $f(x_i)$，也叫做**回归器 regressor**

例如：通过水果的长、宽，预测水果的重量

3. 分类方法 - K 近邻

介绍

- 并不是所有的机器学习算法都需要建立一个模型
- KNN (K- Nearest Neighbors) K 近邻
- 核心想法：**样本之间具有相似性**
- 假设：两个相似的样本有相同的标签
- 假设所有的样本都是在空间 $\mathbb{R}^d$ 的 d 维的样本，

算法

KNN 使用标准的**欧几里得距离**来计算样本的邻近程度

给定两个样本 x_i 和 x_j，它们的距离 $d(x_i, x_j)$ 为

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{d} (x_{ik} - x_{jk})^2}$$

- x_{ik} 是样本 x_i 的第 k 个维度

训练算法

- 将每个被训练的样本 (x, y) 加入训练集 D
  - $x \in \mathbb{R}^d, y \in \{-1, +1\}$

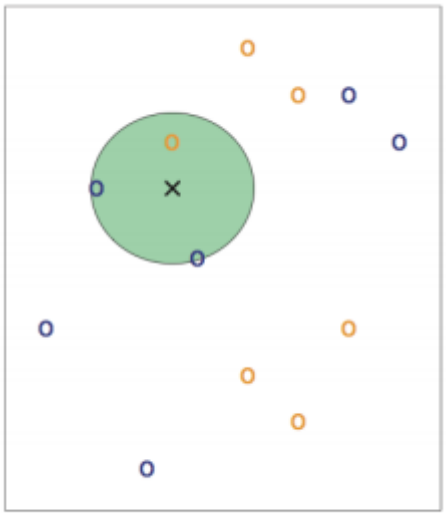
分类算法

给定一个待分类的样本 x_q，假设 $N_k(x_q)$ 是前 k 个距离 x_q 最近的样本，那么对于 x_q 的分类标签 $\hat{y_q}$ 为

$$\hat{y_q} = sign(\sum_{x_i \in N_k(x_q)} y_i)$$

- 即，参考 $N_k(x_q)$ 中的所有样本，它们共同投票选出分类标签

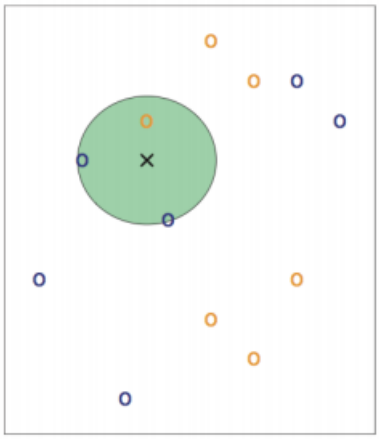
示例

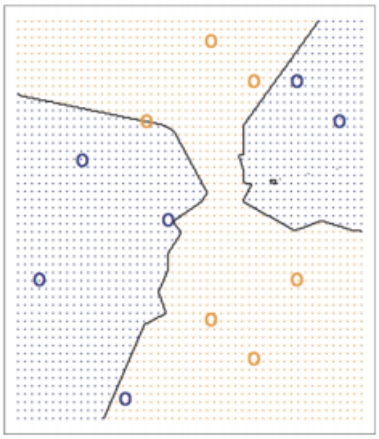


3-NN

$K = 3$，即让 3 个最近的邻居进行投票

分类效果大致如下





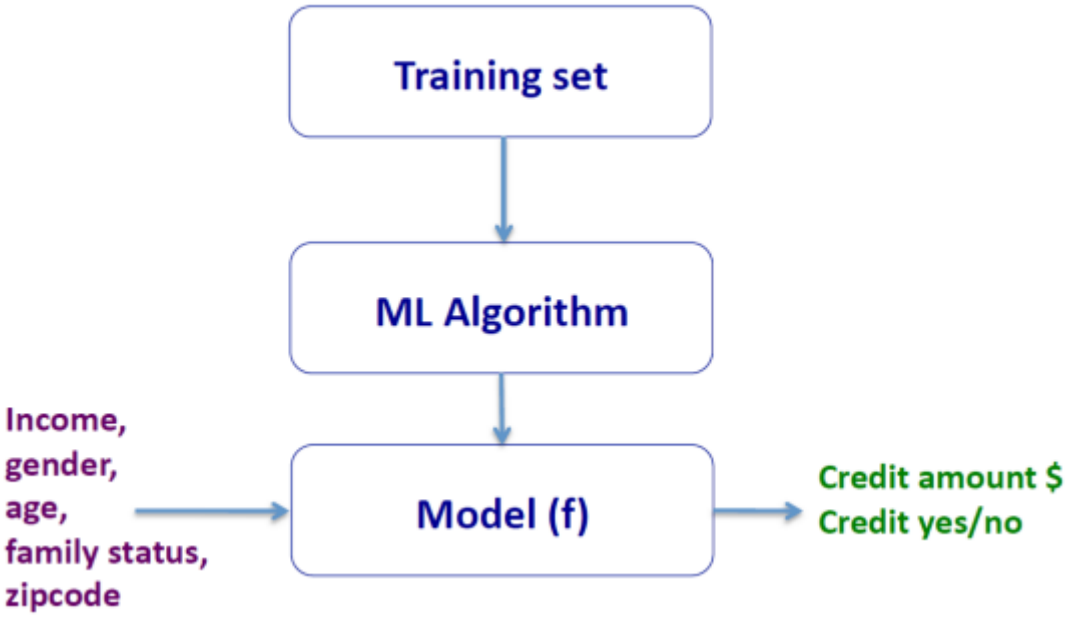
优缺点

- 优点
  - 容易实现
  - 效果很好
  - 不需要建立模型，做出假设，调试参数等
  - 可以很容易地扩展新的样本
- 缺点
  - 存储数据集所需要的空间很大
  - 计算时间大，如果训练集 D 中有 n 个样本，每个样本的维度为 d，那么计算时间需要 $O(n\times d)$
  - 维度灾难：高维度的数据距离差异不大，很难再用距离刻画相似性

应用

- 信息检索
- 大型数据库中使用最近邻
- 手写体字符分类
- 推荐系统（像你这样的用户可能喜欢类似的电影）
- 乳腺癌的诊断
- 医疗数据挖掘（类似的患者症状）
- 一般的模式识别

4. 训练和测试 Train & Test



- 训练 从上到下：数据 → 算法 → 模型
- 测试 从左到右：新数据 → 测试 → 预测结果

误差 Error

我们用 E^{train} 表示**训练误差 in-sample error / training error / empirical error**

$$E^{train}(f)=\sum_{i=1}^{n}loss(y_i,f(x_i))$$

- y_i：样本的实际标签
- $f(x_i)$：通过模型预测的样本的预测标签
- 整个训练误差就是训练的每个样本的实际与预测之间的误差

我们的目标是尽可能优化使得 $E^{train}(f)$ 更小

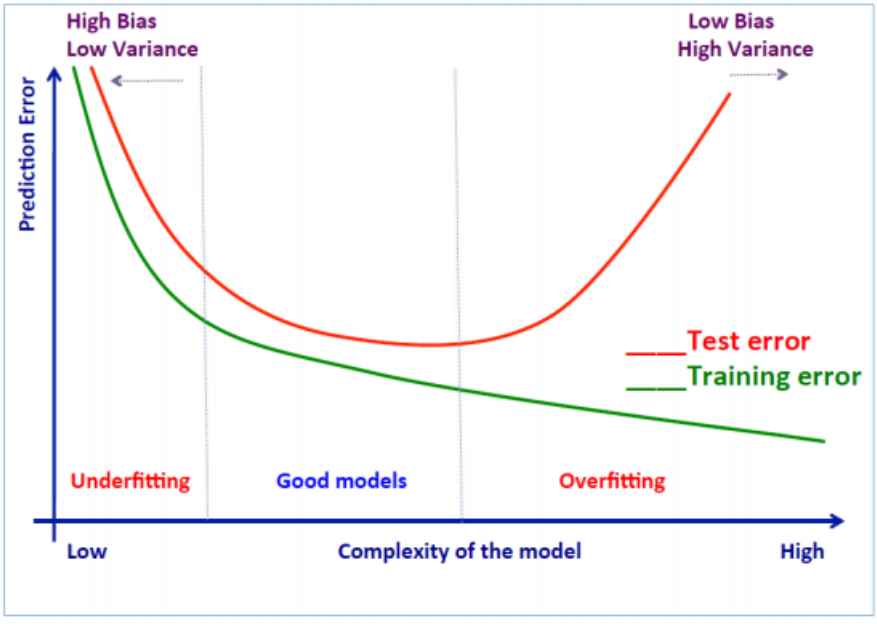
同时，我们也希望**测试误差 out-sample error / test error / true error** $E^{test}(f)$ 尽可能小

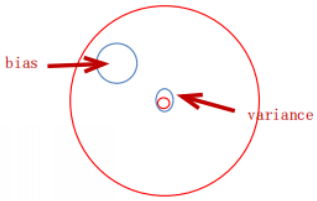
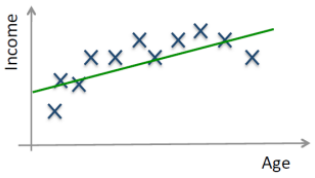
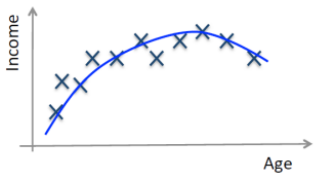
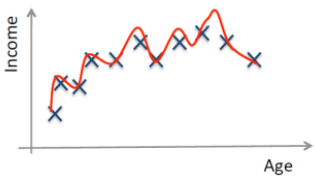
训练集，验证集和测试集

示例：将数据随机分成 3 份

- 训练集：60%
  - 训练集是用来学习一个模型（如分类模型）的一组样本
- 验证集：20%
  - 验证集是一组不能用于学习模型的示例，但可以帮助优化模型参数（例如，K-NN 中选择 K 的值）验证集可以用来控制过拟合
- 测试集：20%
  - 测试集用于评估最终模型的性能，并提供测试误差的估计
  - 永远不要以任何方式使用测试集来进一步调整参数或修改模型

过拟合与欠拟合 Overfitting/underfitting



欠拟合	好的模型	过拟合
高偏差 High bias		高方差 High variance
		

避免过拟合

通常，使用更简单的模型

- 减少特征的数量或者进行特征选择
- 进行模型选择
- 使用正则化（保留特征，但通过设置较小的参数值来降低它们的重要性）
- 通过交叉验证来估计测试误差

正则化 Regulation

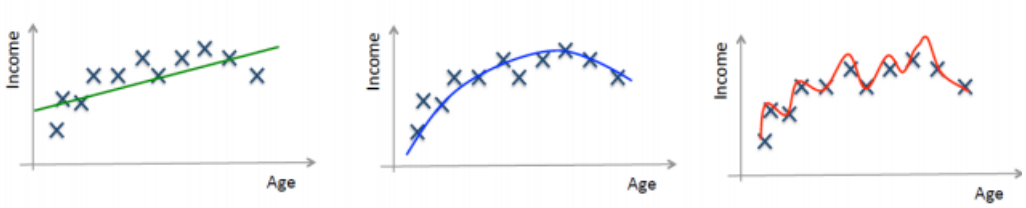
我们将误差重新定义为，我们希望最小化的是

- Classification term + C × Regularization term

$$\sum_{i=1}^{n} loss(y_i, f(x_i)) + C \times R(f)$$

- C：平衡的常数
- $R(f)$：正则化的函数，可能表示模型的复杂度或者别的

在上面的分类中，我们可以发现训练出来的三个模型分别是

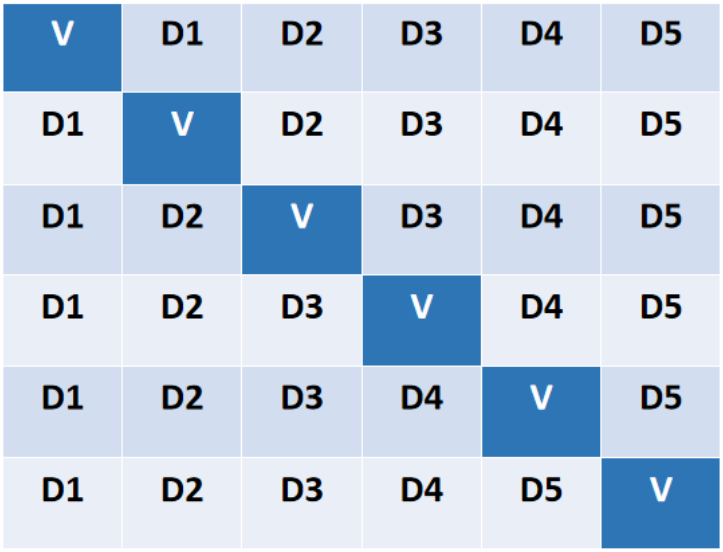


$$f(x) = \lambda_0 + \lambda_1 x \dots (1)$$
$$f(x) = \lambda_0 + \lambda_1 x + \lambda_2 x^2 \dots (2)$$
$$f(x) = \lambda_0 + + \lambda_1 x + \lambda_2 x^2 + \lambda_3 x^3 + \lambda_4 x^4 \dots (3)$$

使用正则化后的误差，尽可能避免使用过于高阶的多项式

K 重交叉验证

一种利用训练数据估计测试误差的方法



算法

给定一个机器学习的算法 A 和一个训练集 D

1. 将数据集 D 分成 n 个相同大小的子集 $D_1, \ldots, D_k$
2. for j = 1 to k:
3. 使用算法 A 训练所有的 D_i $(i \in 1, \ldots, k, i \neq j)$，获得模型 f_j
4. 将模型 f_j 用于测试没有训练的子集 D_j 来计算训练误差 E^{Dj}
5. 计算所有交叉模型的误差和 $\sum_{j=1}^{k}(E^{D_j})$

混淆矩阵 Confusion Matrix

→真实标签 ↓预测标签	Positive	Negative
Positive	True Positive（TP）	False Positive（FP）
Negative	False Negative（FN）	True Negative（TN）

定义	公式	说明
Accuracy	$\frac{(TP+TN)}{TP+TN+FP+FN}$	正确预测的百分比
Precision	$\frac{TP}{TP+FP}$	正确的阳性预测的百分比
Sensitivity / Recall	$\frac{TP}{TP+FN}$	被预测为阳性的占真实的阳性的百分比
Specificity	$\frac{TN}{TN+FP}$	被预测为阴性的占真实的阴性的百分比