模式识别

评估方法
PCA方法

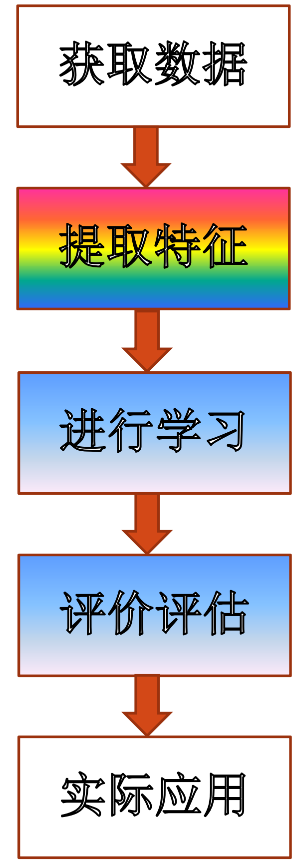
张振宇
南京大学智科院
2025

评估方法简介

Introducing evaluation method

细化(refined)的框架

- ✓ 机器学习 $f: \mathcal{X} \mapsto \mathcal{Y}$
  1. 与领域无关的特征变换和特征抽取
     - Normalization, PCA, FLD, …
  2. 针对不同数据特点的不同学习算法
     - SVM, Decision Tree, imbalanced learning, HMM, DTW, graphical model, deep learning, pLSA, …
  3. 机器学习方法常见分类、策略
- ✓ 针对不同问题的评价准则 (evaluation criterion)



评价准则—泛化和测试误差

- 暂时只考虑分类问题的评价
- 假设$(\boldsymbol{x}, y) \sim p(\boldsymbol{x}, y)$
  - 泛化误差generalization error：$\mathrm{E}_{(\boldsymbol{x},y)\sim p(\boldsymbol{x},y)}[f(\boldsymbol{x}) \neq \boldsymbol{y}]$
    - 通常无法实际计算
  - 根本假设：训练集D_{train}和测试集D_{test}都是服从真实数据分布$p(\boldsymbol{x}, y)$的，或者，他们的样例是从$p(\boldsymbol{x}, y)$中取样（sample）的
  - 测试误差(testing error)

$$err = \frac{1}{n}\sum_{i=1}^{n} \mathbb{I}(f(\boldsymbol{x}_i) \neq y_i), \qquad \boldsymbol{x}_i \in D_{test}$$

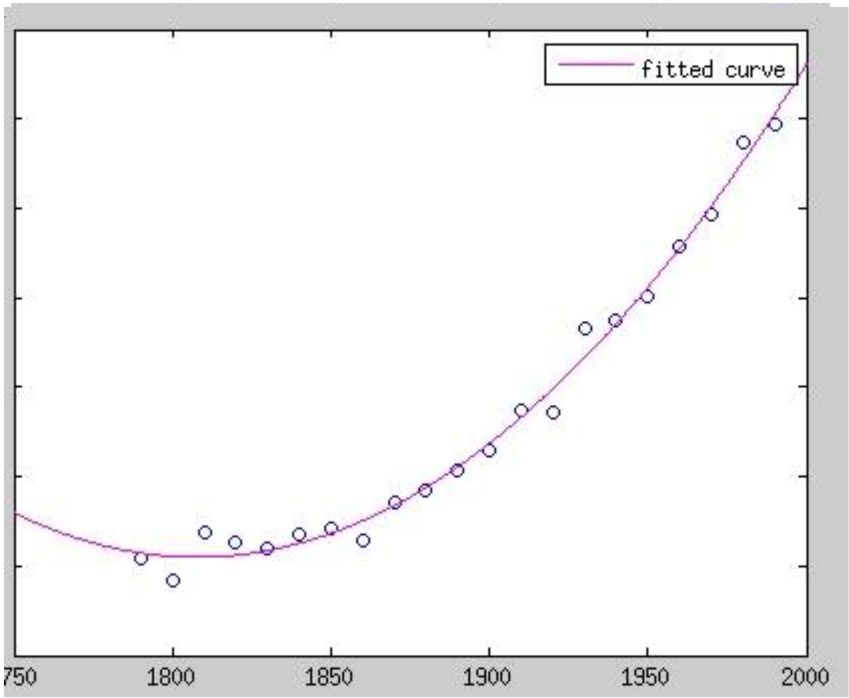
  - 准确率(accuracy)：$acc = 1 - err$

一种常见的学习框架

- 代价最小化cost minimization
  - 错误是最常考虑的代价，所以现在我们可以说学习的目标是在训练集上获得最小的代价
- $\min_f \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(f(\boldsymbol{x}_i) \neq y_i)$， $\boldsymbol{x}_i \in D_{train}$
  - 难以优化 – 怎样求解?
  - 一种方法是：把不连续的指示函数(indicator function)换成性质相似，但好优化的函数
  - 如，$(f(\boldsymbol{x}_i) - y_i)^2$
- 学习这种**思路**：形式化、简化、优化
  - Formalization, simplification, optimization

过拟合和欠拟合

✓ Overfitting & underfitting，以回归(regression)为例



- 以一阶多项式拟合（直线）
- 学习模型的复杂性小于数据的复杂性
- 称为欠拟合 underfitting

正则化regularization

- ✓ 通常难以精确估计学习模型、数据的复杂性
  - 往往选用较复杂的学习模型
  - 训练集误差通常小于测试集误差(需要两者不相交)
- ✓ 那么如何降低overfitting的可能性呢？
  - 正则化regularization

- ✓ 进一步阅读：
  - 正则化如何能降低模型的复杂性？ PRML以及ESL(The Elements of Statistical Learning)

如果没有测试集

- ✓ 例如，总的数据量比较小（如医学图像）
  - 如何评估？
- ✓ 交叉验证cross validation
  1. 将训练集分为大小大致相等的N部分
  2. for $i = 1:N$
     1. 取第i部分的数据为测试集
     2. 取所有其余（一共$N-1$个部分）的数据为训练集
     3. 学习模型并评估/测试得到错误率为err_i
  3. 交叉验证得到的错误率为$\frac{1}{N}\sum_{i=1}^{N} err_i$
  - 称为N倍交叉验证N-fold CV （常用N=5 or 10)
- ✓ 可能需要进行多次试验（后面会讲）

数据、代价的不平衡性imbalance

- 例如，两类问题中，一类数据远比另一类数据多
  - 如，体检中阴性和阳性
  - 男女比例
  - 或在一类犯错的代价远高于另一类

  - 不平衡学习(imbalance learning)
  - 代价敏感学习(cost-sensitive learning)

  - 进一步阅读：周志华教授主页和论文 http://cs.nju.edu.cn/zhouzh/zhouzh.files/publication/publication.htm

评价不平衡时的准则(1)

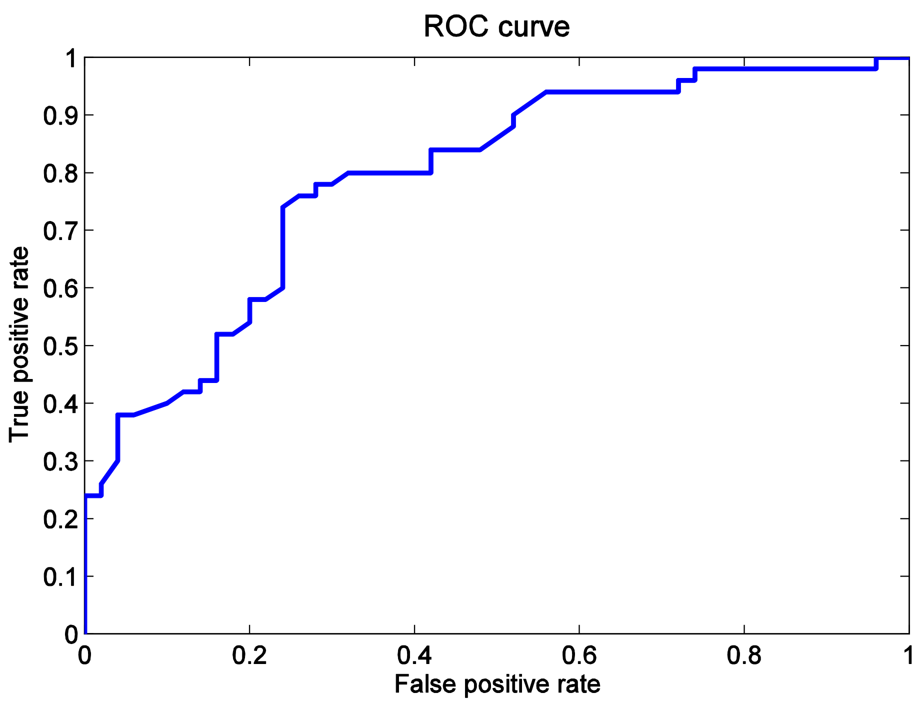
	预测为positive	预测为negative
真实值为positive	True positive (真阳性)	False negative(伪阴性)
真实值为negative	False positive(伪阳性)	True negative(真阴性)

- ✓ TP、TN、FP、FN：标记四种情况的样例数目
- ✓ TOTAL：总数TP+TN+FP+FN
  - 正样本数目：P＝TP+FN，负样本数目：N＝FP+TN
- ✓ False positive rate: FPR ＝ FP / N
- ✓ False negative rate: FNR ＝ FN / P
- ✓ True positive rate: TPR ＝TP / P
- ✓ Accuracy：ACC ＝ (TP+TN) / TOTAL

评价不平衡时的准则(2)

✓ AUC-ROC (Area Under the ROC Curve)

- ROC – Receiver operating characteristic



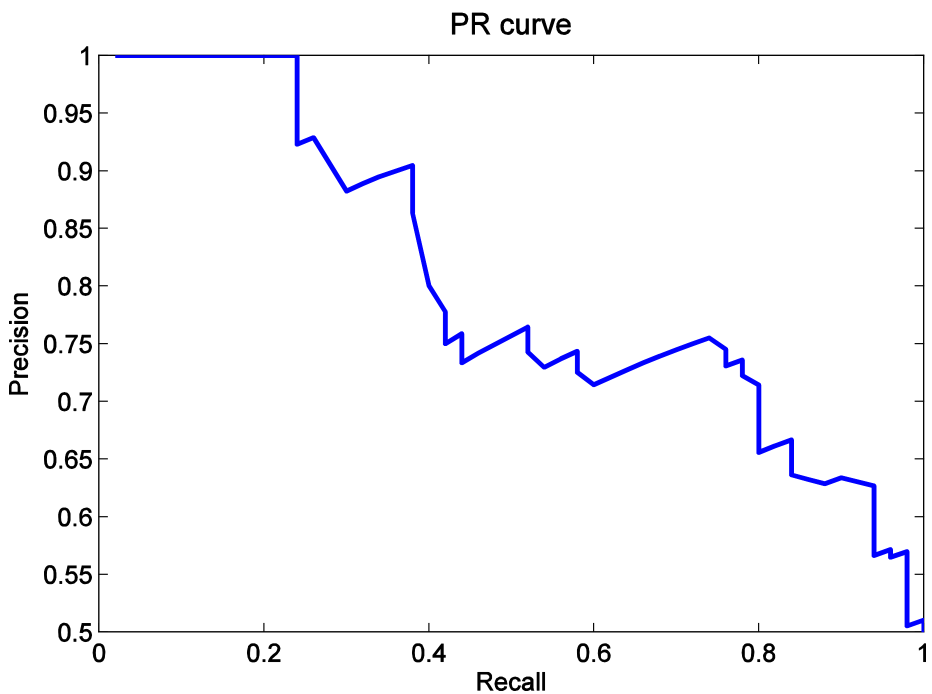
- Y轴： TPR
- X轴： FPR
- 其值为面积
- 为什么？
- 对角线是？
- 非减

评价不平衡时的准则(3)

- Precision（查准率）: PRE = TP / (TP + FP)
- Recall（查全率）: REC = TP / P (和TPR一样）
- F1 score: Precision和Recall的调和平均(harmonic mean)
  - 调和平均: $\left(\frac{x^{-1}+y^{-1}}{2}\right)^{-1} = \frac{2xy}{x+y}$
  - F1 = 2TP / (2TP+FP+FN)
  - 为什么？

评价不平衡时的准则(4)

✓ AUC-PR (Area Under the Precision-Recall Curve)



- Y轴： Precision
- X轴： Recall
- 其值为面积
- 为什么？
- 单调吗？

进一步阅读: [The Relationship Between Precision-Recall and ROC Curves]

能100%准确吗:Bayes框架的回答(1)

- ✓ $f: \mathcal{X} \mapsto \mathcal{Y}$, 一个数据对(data pair): $(\boldsymbol{x}, y)$
  - 假设注重于分类： $\mathcal{Y} = \{1,2, \dots, m\}$
  - 先验概率prior probability： $p(y = i)$
    - 在没有看到任何数据时，怎么分类？
  - 后验概率posterior probability: $p(y = i|\boldsymbol{x})$
    - 看到数据$\boldsymbol{x}$后，得到更多的信息，可以对分类有更好的估计
  - 类条件概率class conditional probability: $p(\boldsymbol{x}|y = i)$
    - 数据的综合分布$p(\boldsymbol{x})$和每个类别内部的分布$p(\boldsymbol{x}|y = i)$不一样
- ✓ 贝叶斯定理Bayes' theorem

$$p(y = i|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|y = i)p(y = i)}{p(\boldsymbol{x})} = \frac{\text{条件} \times \text{先验}}{\text{数据}}$$

代价矩阵

- 目前常见的为$\begin{pmatrix}\lambda_{11} & \lambda_{12}\\ \lambda_{21} & \lambda_{22}\end{pmatrix}=\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}$
  - λ_{ij}: 当真实值为i、模型预测为j时付的代价
  - 0-1代价：即分类正确代价为0，分类错误代价为1
  - 但是，根据实际情况，可以给λ_{ij}设置任何值
    - 代价不平衡学习

- 对代价的计算：$\mathrm{E}_{(\boldsymbol{x},y)}\left[\lambda_{y,f(\boldsymbol{x})}\right]$
  - 当使用0-1代价时，和错误率一致

能100%准确吗:Bayes框架的回答(2)

- ✓ 贝叶斯决策规则Bayes decision rule:
  - 选择代价最小的类别输出

$$\underset{y}{\operatorname{argmin}}\, \mathrm{E}_{(\boldsymbol{x},y)}\left[\lambda_{y,f(\boldsymbol{x})}\right]$$

  - 贝叶斯风险Bayes risk：使用贝叶斯决策规则的风险
  - 其是理论上我们能得到的最好的结果，记为R^*
- ✓ 在使用0-1风险时，风险和错误率等价
  - 所以，R^*是我们理论上能得到的最小误差
  - $1-R^*$是理论上最高的准确率！

- ✓ 自学：DHS2.1 DHS2.2（包括似然比规则likelihood ratio rule）

真实值Groundtruth

- ✓ 大多数时候，是手工标注的(manual annotation)
  - 或者人也不知道确切的答案
  - 有时候疲劳或其他因素会导致标注的错误
  - 很耗时、昂贵
- ✓ 真实值的形式
  - 分类：一个离散的类别
  - 回归regression：一个连续的值
  - 结构structured output：例如，输出一个句子的分词结果“一个/句子/的/分词/结果”

错误从哪里来—以回归为例？

✓ 真实（但未知）的函数$F(\boldsymbol{x})$

- 用由其产生的数据集D来学习，即$y = F(\boldsymbol{x})$没有误差
- 回归的代价函数是欧几里得距离

✓ $\mathrm{E}_D\left[\left(f(\boldsymbol{x};D) - F(\boldsymbol{x})\right)^2\right] = \left(\mathrm{E}_D[f(\boldsymbol{x};D)] - F(\boldsymbol{x})\right)^2 + \mathrm{E}_D[(f(\boldsymbol{x};D) - \mathrm{E}_D[f(\boldsymbol{x};D)])^2]$

- $\boldsymbol{x}$和$F(\boldsymbol{x})$不包含随机性，只有D出现时才取期望
- 简写为$E[(f-F)^2] = (F - Ef)^2 + E[(f - Ef)^2]$
- DHS 376页的处理（或翻译）有问题

偏置-方差分解

- Bias-variance decomposition
  - $\mathrm{E}[(f-F)^2] = (F-Ef)^2 + \mathrm{E}[(f-Ef)^2]$
  - $\mathrm{E}[F-Ef]$ -- 偏置bias
    - 当训练集取样有差异时，其值不变
  - $\mathrm{E}[(f-Ef)^2] = \mathrm{Var}_D(f(\boldsymbol{x};D))$方差
    - 当训练集取样有差异时，会带来预测的差异（误差不同）
- 误差=偏置^2+方差
- 当考虑到$y=F(\boldsymbol{x})$有误差是（白噪声）
  - 误差=偏置^2+方差+噪声
  - 估计误差时，如没有测试集，需多次平均
- 进一步阅读：分类时候的分解(DHS9.3.2)

对分解的解读

- ✓ 偏置与数据无关，是由模型（的复杂度）决定的
  - 例如，线性分类器（1阶多项式）的偏置大
  - 但是，7阶多项式的复杂度高，偏置小
- ✓ 但是，方差$\mathrm{Var}_D(f(\boldsymbol{x}; D))$和抽样得到的训练集以及模型两者都有关系
  - 例如，高阶多项式的方差大
- ✓ 怎么减少误差？
  - 对于噪音，机器学习没有办法—高质量的数据获取！
  - 减少偏置和方差
    - 如集成方法(ensemble methods)

统计测试

- 可以决定我们对于评估的结果是否有信心
  - 自学教材第4.5节

  - 目标是理解统计测试的基本思想

  - 知识要点
    - 怎么理解置信度（第4.5.2节最后部分）
    - T检验的实际操作流程

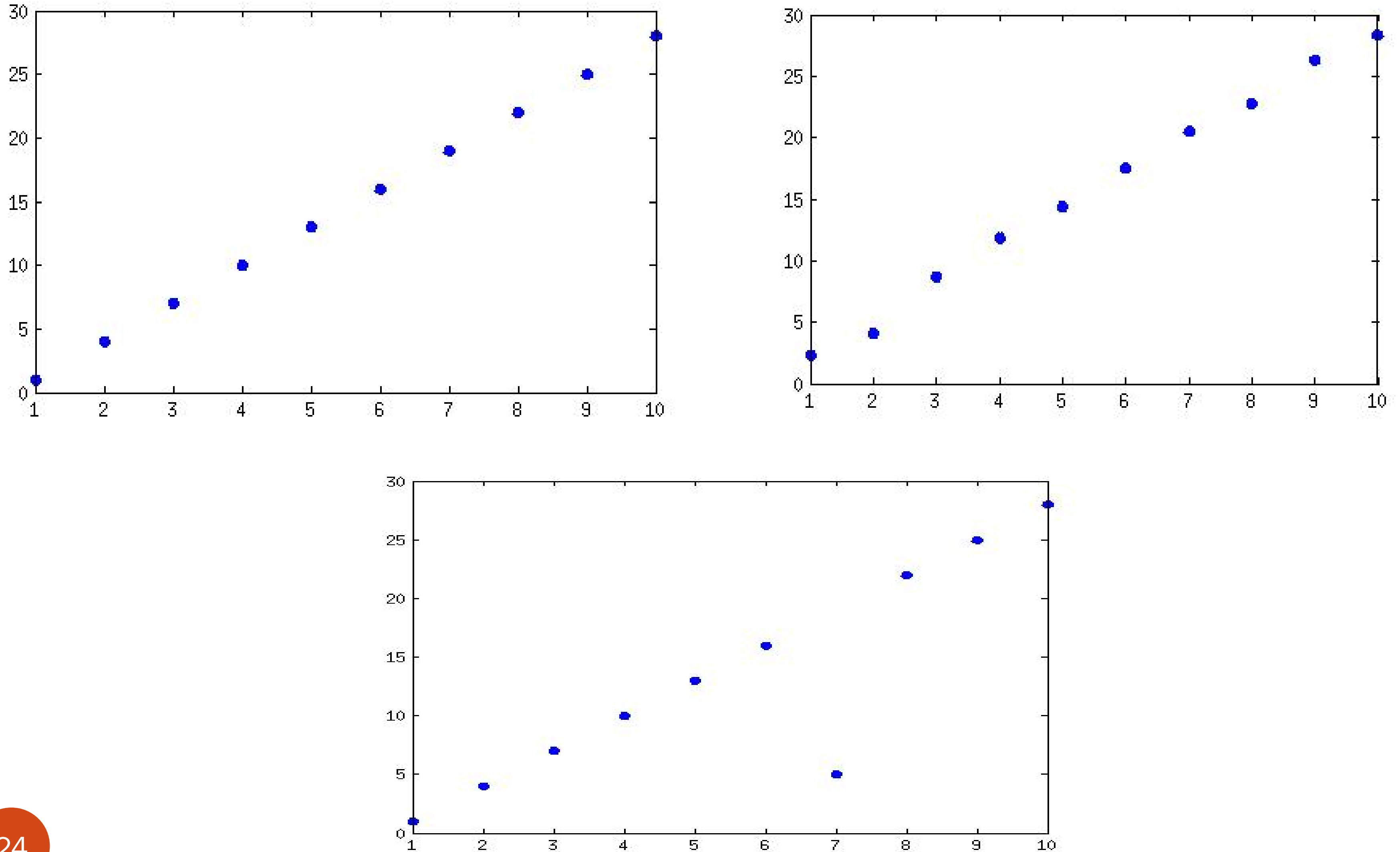
    - 因假设“概率与数理统计”是前置课程,第4.5节只是一个简要回顾

目标

- ✓ 理解PCA的含义、目的、适用范围
- ✓ 熟记PCA的各个步骤，能实际应用PCA
- ✓ 了解PCA的各种相关解释，理解其含义
- ✓ 提高目标
  - 理解相关推导，能自主独立完成推导
  - 进一步能通过独立阅读理解更多PCA的含义、限制、解释等，并能应用到学习、研究中遇到的问题中去

PCA基础

你的数据是多少维的？



常见的数据特点

✓ 数据各维度之间不是互相独立的

- 数据的内在维度(intrinsic dimensionality)通常远低于其表面维度
- 因此，需要降低数据维度(dimensionality reduction)
- PCA在降维方法中（可能）是最常用的一种

这是谁？

$$96 \times 108 = 10368?$$

Starting point: 零阶表示

- Zero-dimensional representation
- 不允许使用任何维度，如何最佳表示$\boldsymbol{x}$?
- 寻找某个固定(constant)的$\boldsymbol{m}$，使得

$$J_1(\boldsymbol{m}) = \min_{\boldsymbol{m}} \sum_{i=1}^{n} \|\boldsymbol{x}_i - \boldsymbol{m}\|^2$$

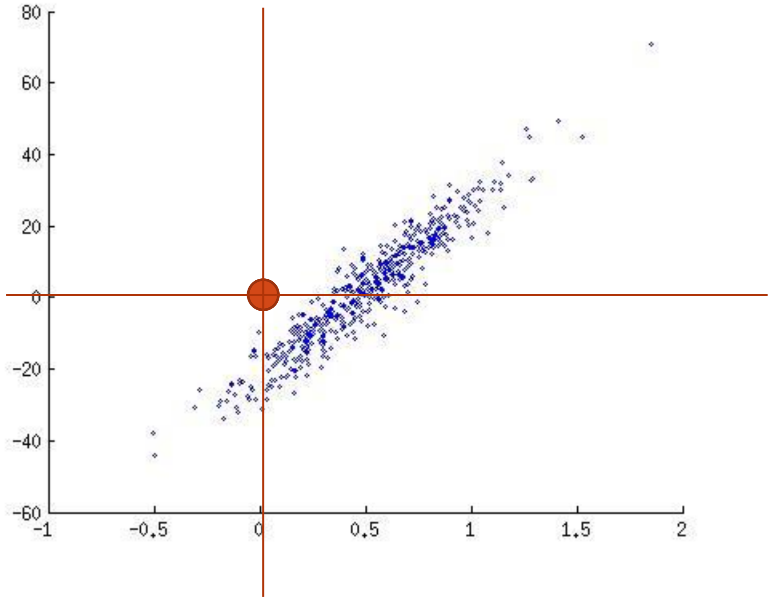
- 最优解：(证明？)

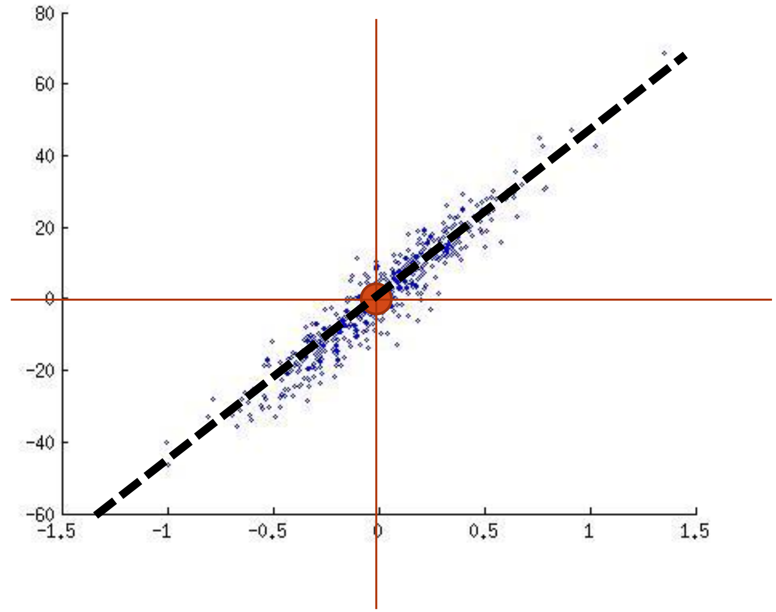
$$\boldsymbol{m}^* = \operatorname{argmin}_{\boldsymbol{m}} \sum_{i=1}^{n} \|\boldsymbol{x}_i - \boldsymbol{m}\|^2 = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i$$

一维表示：数据维度间的线性关系

- ✓ 如同前面的例子
  - 数据是d维
  - 但是内在维度可能是m维的，$m < d$或者$m \ll d$
  - PCA用线性关系来降低维度
- ✓ $\boldsymbol{x} \in \mathbb{R}^d$: 原来的高维数据（随机变量）
  - 训练样本：$\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_n$
- ✓ 假设$m = 1$, 用原数据的单个线性组合来表示
  - $y_i = \boldsymbol{w}^T \boldsymbol{x}_i + b$
  - $y_1, y_2, \dots, y_n$ -- 新的数据/特征(features)
  - 如何寻找最佳的$\boldsymbol{w}$？如何找到最佳的b？

Idea：选择什么方向？为什么？





✓什么方向最优？

形式化formalization：最大化方差

- 方差是衡量新特征包含信息多少的度量
  - 有时也称为能量energy
- 优化目标函数 $J_2(\boldsymbol{w}) = \frac{1}{n}\sum_{i=1}^{n}\left\|\boldsymbol{w}^T(\boldsymbol{x}_i - \overline{\boldsymbol{x}})\right\|^2$
- 发现问题了吗?
  - $J_2(\boldsymbol{w})$可以是无穷大或者无穷小！
  - 最常用的解决办法：加上限制条件 $\|\boldsymbol{w}\|^2 = \boldsymbol{w}^T\boldsymbol{w} = 1$

$$\underset{\boldsymbol{w}}{\operatorname{argmax}} \frac{1}{n}\sum_{i=1}^{n}\left\|(\boldsymbol{x}_i - \overline{\boldsymbol{x}})^T\boldsymbol{w}\right\|^2$$

$$\text{s.t.} \qquad \boldsymbol{w}^T\boldsymbol{w} = 1$$

  - s.t. –subject to,表示约束条件constraint(s)

简化simplification 变换transformation

$$\left\|(\boldsymbol{x}_i-\overline{\boldsymbol{x}})^T\boldsymbol{w}\right\|^2=\left((\boldsymbol{x}_i-\overline{\boldsymbol{x}})^T\boldsymbol{w}\right)^T\left((\boldsymbol{x}_i-\overline{\boldsymbol{x}})^T\boldsymbol{w}\right)$$
$$=\boldsymbol{w}^T(\boldsymbol{x}_i-\overline{\boldsymbol{x}})(\boldsymbol{x}_i-\overline{\boldsymbol{x}})^T\boldsymbol{w}$$

$$\frac{1}{n}\sum_{i=1}^{n}\left\|(\boldsymbol{x}_i-\overline{\boldsymbol{x}})^T\boldsymbol{w}\right\|^2=\boldsymbol{w}^T\sum_{i=1}^{n}\frac{1}{n}(\boldsymbol{x}_i-\overline{\boldsymbol{x}})(\boldsymbol{x}_i-\overline{\boldsymbol{x}})^T\boldsymbol{w}$$
$$=\boldsymbol{w}^T\mathrm{Cov}(\boldsymbol{x})\boldsymbol{w}$$

优化optimization

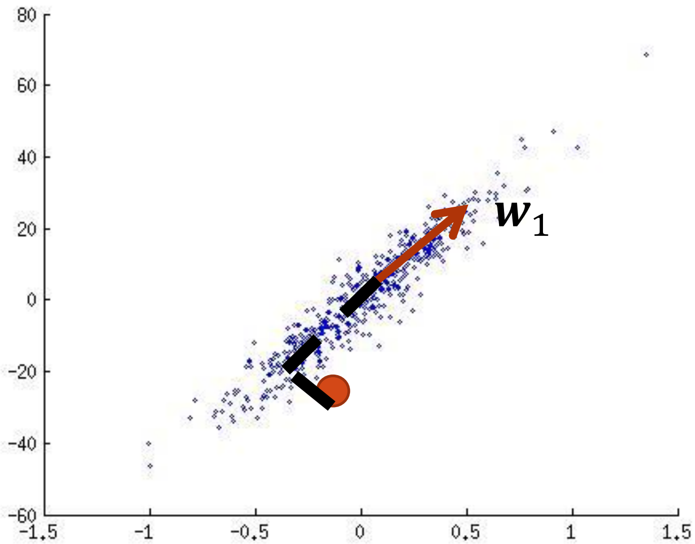
- ✓ 拉格朗日乘子法 Lagrange multipliers
  - 将有约束的优化问题转化为无约束的优化问题
- ✓ Lagrangian 拉格朗日函数

$$f(\boldsymbol{w},\lambda)=\boldsymbol{w}^T Cov(\boldsymbol{x})\boldsymbol{w}-\lambda(\boldsymbol{w}^T\boldsymbol{w}-1)$$

- ✓ λ：拉格朗日乘子Lagrange multiplier
- ✓ 最优的必要条件：$\frac{\partial f}{\partial \boldsymbol{w}}=\mathbf{0}$，$\frac{\partial f}{\partial \lambda}=0$
- ✓ $\frac{\partial f}{\partial \boldsymbol{w}}=2\mathrm{Cov}(\boldsymbol{x})\boldsymbol{w}-2\lambda\boldsymbol{w}=\mathbf{0}$
  - 我们这里的前提条件是什么？
  - 应该想到用哪一个公式？
- ✓ $\mathrm{Cov}(\boldsymbol{x})\boldsymbol{w}=\lambda\boldsymbol{w}$, $\boldsymbol{w}^T\boldsymbol{w}=1$!

选哪个特征向量？

- ✓ $J_2(\boldsymbol{w}) = \boldsymbol{w}^T \mathrm{Cov}(\boldsymbol{x})\boldsymbol{w} =?$
- ✓ $\mathrm{Cov}(\boldsymbol{x})$是半正定的（如何证明？）
- ✓ 选取λ_1(即最大特征值)对应的特征向量$\boldsymbol{\xi}_1$为$\boldsymbol{w}_1$
  - 为什么？约束条件满足了吗？
- ✓ 怎样用$\boldsymbol{w}_1$来近似$\boldsymbol{x}$？
  - 投影！
  - $\boldsymbol{x} \approx \overline{\boldsymbol{x}} + (\boldsymbol{w}_1^T(\boldsymbol{x} - \overline{\boldsymbol{x}}))\boldsymbol{w}_1$
  - 所以，$y_i = \boldsymbol{w}_1^T(\boldsymbol{x} - \overline{\boldsymbol{x}})$
  - 那么，$b =?$



J_1和J_2的等价关系

✓ 若干向量

- $\boldsymbol{x}_i$：降维之前的向量
- $\boldsymbol{w}_1^T(\boldsymbol{x}_i-\overline{\boldsymbol{x}})\boldsymbol{w}_1=y_i\boldsymbol{w}_1$：降维之后的向量
- $\widehat{\boldsymbol{x}}$：在原空间中重建的向量
- 目前的重建关系：$\widehat{\boldsymbol{x}}_i\approx\overline{\boldsymbol{x}}+y_i\boldsymbol{w}_1$

✓ J_1的目的是使得$\widehat{\boldsymbol{x}}_i$和$\boldsymbol{x}_i$尽可能相差小($\overline{\boldsymbol{x}}$固定为均值)

- $J_1(\boldsymbol{w},\boldsymbol{a})=\sum_{i=1}^{n}\frac{1}{n}\|\boldsymbol{x}_i-(\overline{\boldsymbol{x}}+a_i\boldsymbol{w})\|^2$
- $\boldsymbol{w}$：投影方向，a_i：投影系数

✓ 最小化J_1得到的a_i和$\boldsymbol{w}$与J_2得到的结果完全一致！

- 试着去证明!

如果需要更多投影方向？

- What if we need $\boldsymbol{w}_2, \boldsymbol{w}_3, \ldots$
  - 新的投影方向需要继续保持“能量”
  - 但是需要限制
  - $\boldsymbol{w}_2 \perp \boldsymbol{w}_1$， $\boldsymbol{w}_3 \perp \boldsymbol{w}_2$， $\boldsymbol{w}_3 \perp \boldsymbol{w}_1$， …
- 在上述限制条件下
  - $\boldsymbol{w}_2 = \boldsymbol{\xi}_2$， $\boldsymbol{w}_3 = \boldsymbol{\xi}_3$， …
  - 重建系数: $\boldsymbol{w}_j^T(\boldsymbol{x} - \overline{\boldsymbol{x}})$
- 总之，

$$\boldsymbol{x} \approx \overline{\boldsymbol{x}} + (\boldsymbol{w}_1^T(\boldsymbol{x} - \overline{\boldsymbol{x}}))\boldsymbol{w}_1 + (\boldsymbol{w}_2^T(\boldsymbol{x} - \overline{\boldsymbol{x}}))\boldsymbol{w}_2 + \cdots$$

重建和原数据的关系

- ✓ 假设$n > d$，即数据比维数多
  - 进一步假设$\mathrm{Cov}(\boldsymbol{x})$可逆
  - 如果$n < d$，那么情况如何、还能做PCA变换吗？
- ✓ $\mathrm{Cov}(\boldsymbol{x})$是$d \times d$矩阵，有$d$个互相垂直的特征向量$\boldsymbol{\xi}_i$
  - 重建会有d个互相垂直的投影方向$\boldsymbol{w}_i$

$$\forall \boldsymbol{x}, \qquad \boldsymbol{x} = \bar{\boldsymbol{x}} + \sum_{i=1}^{d} \left(\boldsymbol{w}_i^T (\boldsymbol{x} - \bar{\boldsymbol{x}})\right) \boldsymbol{w}_i$$

  - 将$\boldsymbol{w}_i$拼成矩阵形式$W = [\boldsymbol{w}_1 \;\; \boldsymbol{w}_2 \;\; \cdots \boldsymbol{w}_d]$ $(d \times d)$
  - 投影系数是$W^T(\boldsymbol{x} - \bar{\boldsymbol{x}})$，投影方向是$W$
  - $\boldsymbol{x} = \bar{\boldsymbol{x}} + WW^T(\boldsymbol{x} - \bar{\boldsymbol{x}})$ （为什么？）
  - 重建是完全精确的（没有误差），为什么？

降维

- ✓ 很多时候，有些投影方向是噪声
  - 需要扔掉一些方向
  - 扔掉哪些？扔掉多少?
- ✓ 去掉特征值最小的那些
- ✓ 通常保持90%的能量

$$\frac{\lambda_1 + \lambda_2 + \cdots + \lambda_T}{\lambda_1 + \lambda_2 + \cdots + \lambda_d} > 0.9$$

  - 寻找第一个T，使得上面的不等式成立

降维的损失

- ✓ 现在$\widehat{W} = [\boldsymbol{w}_1 \ \ \boldsymbol{w}_2 \ \ \cdots \boldsymbol{w}_T]$ $(d \times T)$
- ✓ $\boldsymbol{x} - \widehat{\boldsymbol{x}} = \sum_{j=T+1}^{d} (\boldsymbol{w}_j^T (\boldsymbol{x} - \overline{\boldsymbol{x}}))\boldsymbol{w}_j = \sum_{j=T+1}^{d} \boldsymbol{e}_j$
  - 这个误差多大？
  - $\boldsymbol{e}_j^T \boldsymbol{e}_k = 0$ 如果 $j \neq k$ (为什么？)
  - $\mathrm{E}\left(\|\boldsymbol{x} - \widehat{\boldsymbol{x}}\|^2\right) = \sum_{j=T+1}^{d} \mathrm{E}\left(\left\|\boldsymbol{e}_j\right\|^2\right)$
  - $\mathrm{E}\left(\left\|\boldsymbol{e}_j\right\|^2\right) = \lambda_j$
- ✓ 这样降维保证平均（期望）重建误差最小
  - 直接优化重建误差J_1得到同样的结果

小结：PCA变换的步骤

- ✓ 训练样本：$\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$
- ✓ 计算得到$\overline{\boldsymbol{x}}$和$\mathrm{Cov}(\boldsymbol{x})$
- ✓ 求得$\mathrm{Cov}(\boldsymbol{x})$的特征值和特征向量
  - Matlab， R， octave， …
- ✓ 根据特征值选定T
- ✓ 根据T的值确定矩阵$\widehat{W}$

- ✓ 对任何数据$\boldsymbol{x}$，其新的经过PCA变换得到的特征是

$$\boldsymbol{y} = \widehat{W}^T(\boldsymbol{x} - \overline{\boldsymbol{x}})$$

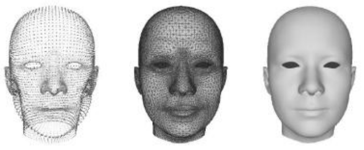
重建则为 $\boldsymbol{x} \approx \hat{\boldsymbol{x}} = \overline{\boldsymbol{x}} + \widehat{W}\boldsymbol{y}$

应用示例——3D人脸基底制作

01

模板构建

$$S = (X_1, Y_1, Z_1, X_2, ..., Y_n, Z_n)^T \in \Re^{3n}$$
$$T = (R_1, G_1, B_1, R_2, ..., G_n, B_n)^T \in \Re^{3n}$$

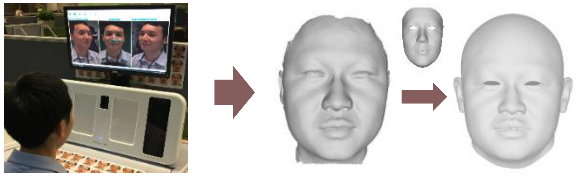


(a) 三维点表示 (b) 三角网格 (c) 平滑面表示

02

数据集制作

$$\mathbf{S}_{mod} = \sum_{i=1}^{m} a_i \mathbf{S}_i\,, \quad \mathbf{T}_{mod} = \sum_{i=1}^{m} b_i \mathbf{T}_i\,, \quad \sum_{i=1}^{m} a_i = \sum_{i=1}^{m} b_i = 1.$$

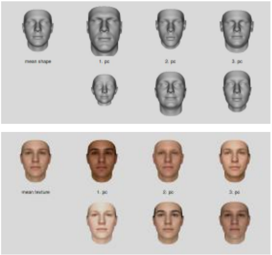


采集数据 模板注册

03

基底制作

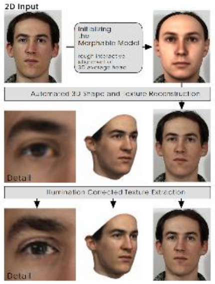
$$\Delta S_i = S_i - \overline{S} \text{ and } \Delta T_i = T_i - \overline{T}.$$
$$S_{model} = \overline{S} + \sum_{i=1}^{m-1} \alpha_i s_i\,, \quad T_{model} = \overline{T} + \sum_{i=1}^{m-1} \beta_i t_i$$



04

优化求解

$$\mathbf{I}_{model}(x, y) = (I_{r,mod}(x, y), I_{g,mod}(x, y), I_{b,mod}(x, y))^T$$
$$E_I = \sum_{x,y} \|\mathbf{I}_{input}(x, y) - \mathbf{I}_{model}(x, y)\|^2.$$



几何系数 α

纹理系数 β

人脸姿态 r, t

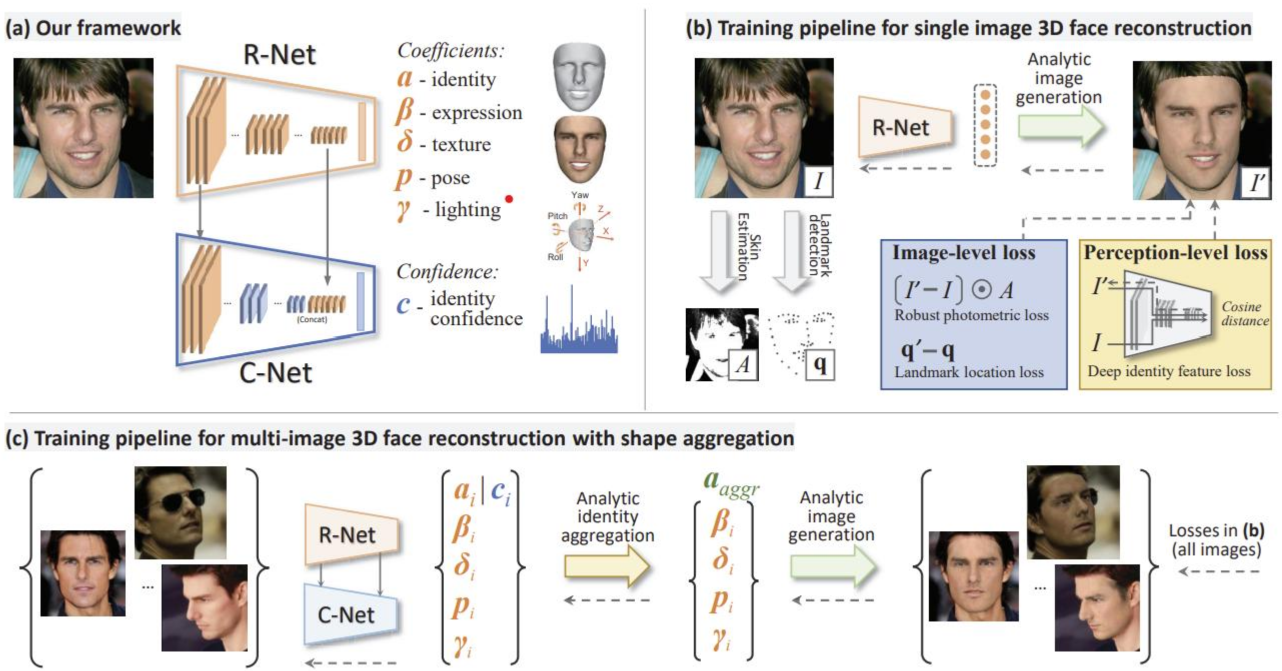
光照 l

应用示例——3D人脸基底制作、求解

基于3DMM的人脸重建本质都是求解基底系数

$$\Delta S_i = S_i - \overline{S} \text{ and } \Delta T_i = T_i - \overline{T}.$$

$$S_{model} = \overline{S} + \sum_{i=1}^{m-1} \alpha_i s_i\,,\quad T_{model} = \overline{T} + \sum_{i=1}^{m-1} \beta_i t_i$$



- Deng Y, Yang J, Xu S, et al. Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. 2019: 0-0.