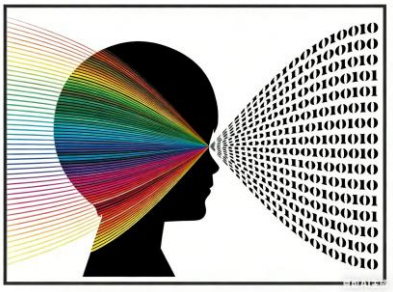


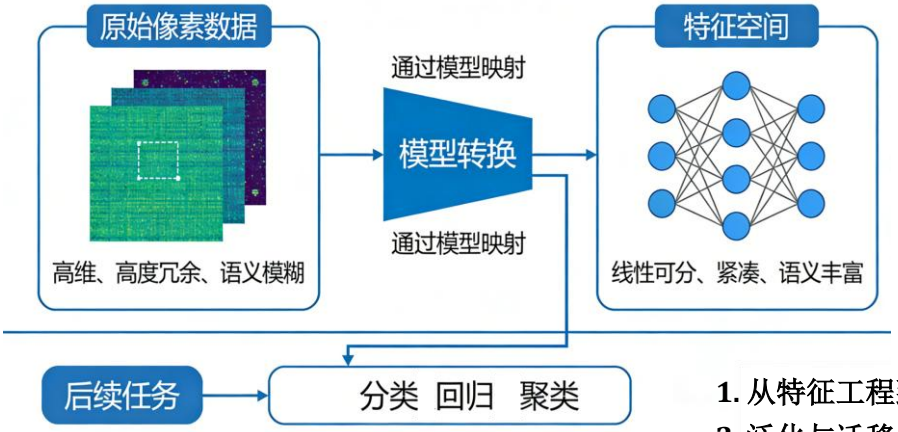
计算机视觉

第4章 视觉特征学习

福州大学 陈飞
chenfei314@fzu.edu.cn



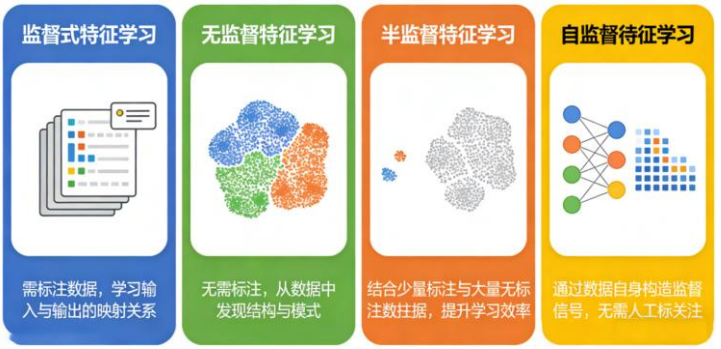
视觉特征学习



1. 从特征工程到特征学习
2. 泛化与迁移
3. 特征空间的度量学习

本章内容

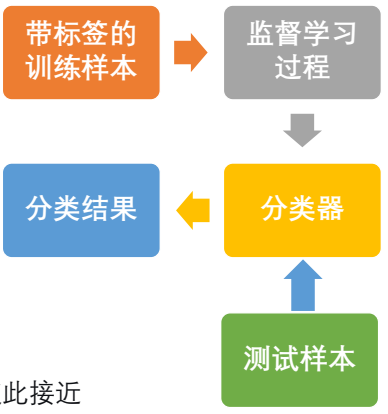
- ❑ 监督式特征学习
- ❑ 无监督特征学习
- ❑ 半监督特征学习
- ❑ 自监督特征学习



4.1监督式特征学习

• **监督学习：**有标签数据、直接反馈、预测结果/未来

- 输入空间：$\mathcal{X} \subseteq \mathbb{R}^D$（原始数据，如 D 维像素）
- 标签空间：$\mathcal{Y} = \{1, 2, \ldots, C\}$（$C$ 个类别）
- 训练数据集：$\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$，其中 $\mathbf{x}_i \in \mathcal{X}$，$y_i \in \mathcal{Y}$



监督式特征学习的目标是学习一个**特征映射函数**（编码器）：

$$f_\theta : \mathcal{X} \to \mathbb{R}^d$$

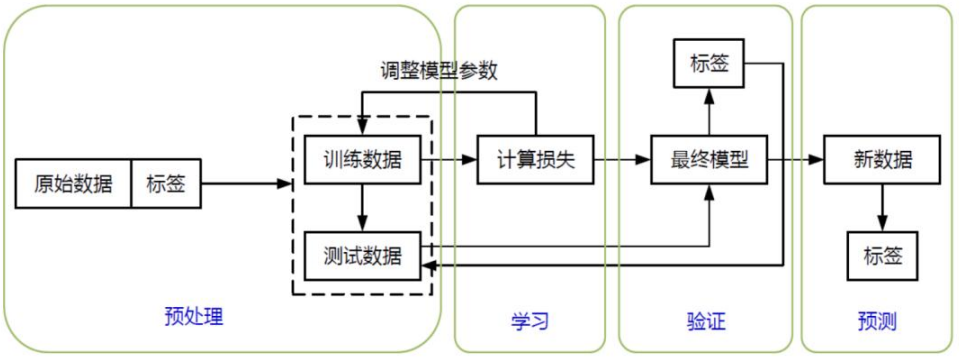
其中 $d \ll D$（特征维度远小于原始维度），且 $f_\theta(\mathbf{x})$ 应具备：

•**判别性**：不同类别的特征易于区分；**紧凑性**：同类样本的特征彼此接近

监督式特征学习

优化问题:

$$\min_{\theta} \mathcal{L}_{task}\left(\{(f_\theta(\mathbf{x}_i), y_i)\}\right) + \lambda \cdot \Omega(\theta)$$



基于距离的识别

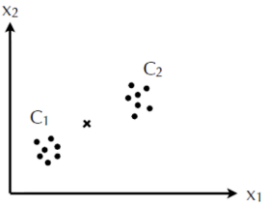
- 基于距离的识别：把测试模式和已有模式之间的距离作为判断准则。该技术是最常见的模式识别技术，是其它高级识别技术的基础。
- 判别公式

$$x \in C_i, \quad \text{if } d(x, C_i) \le d(x, C_j), \quad \forall j \ne i$$

x：测试模式

C：物体类

d：距离



- 类的表达：最基于距离的识别中，使用最简单的一种关于类的知识表达，即所有训练模式的集合。

$$C_i = \{y_{i,1}, y_{i,2}, \cdots, y_{i,N_i}\}$$

点对点的距离测量

欧式距离：

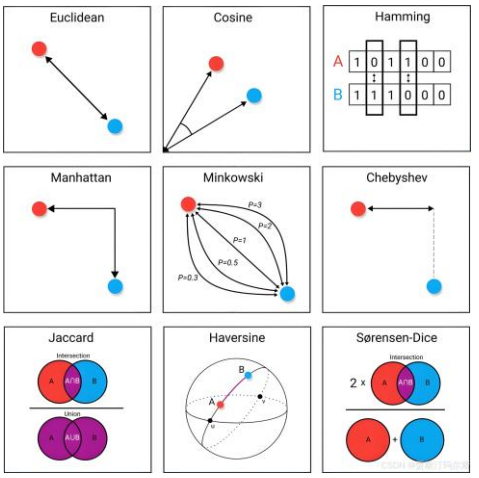
$$d(x,z_i)=(x-z_i)^T(x-z_i)=\sum_m (x_m-z_{i,m})^2$$

Manhattan距离：

$$d(x,z_i)=|x-z_i|=\sum_m |x_m-z_{i,m}|$$

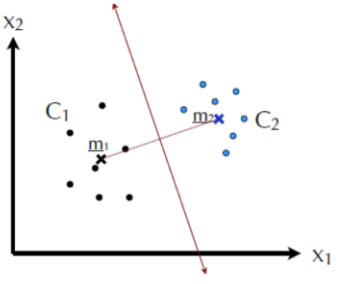
距离度量标准(Distance Metric)：

1. 同一性: $d(x,z)=0,\ \text{iff } x=z$
2. 非负性： $d(x,z)\geq 0$
3. 对称性 $d(x,z)=d(z,x)$
4. 三角不等式：$d(x,z)\leq d(x,y)+d(y,z)$

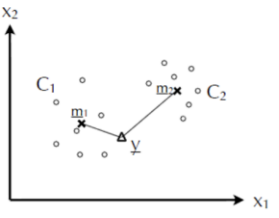


MED分类器

- MED分类器：最小欧式距离分类器(Minimum Euclidean Distance Classifier)
- 对于2个类而言，MED分类器的决策边界是一个超平面，该平面垂直且二分连接两个类原型的线。



$$x\in C_1,\quad \text{if } d(x,C_1)<d(x,C_2)$$
$$(x-z_1)^T(x-z_1)<(x-z_2)^T(x-z_2)$$



点对概率分布的距离测量

- 如果把一个类中含有的所有模式看作一个概率分布，则可以计算该类所含模式的统计量，依据该类的统计量来计算距离。
- **Mahalanobis（马哈拉诺比斯）距离**：

$$d_M(x,C_i)=(x-\mu_i)^T S_i^{-1}(x-\mu_i)$$

μ_i：类的均值

S_i：类的协方差

该距离不仅考虑了类的均值对于距离测量的影响，还加入了类的方差对于距离测量的影响。

•**自动归一化**：内部包含了每个维度的方差信息，自动将不同尺度的特征拉到同一量级比较。
去相关性：通过协方差矩阵的逆，将数据投影到一个各维度独立、方差为1的空间中，消除了特征间的相关性影响。

点对统计分布的距离测量

- **Mahalanobis距离**的属性

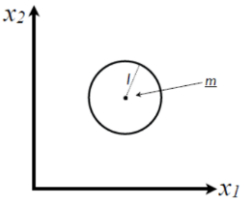
当S=I时：等于欧式距离：

$$d_M(x,C_i)=(x-\mu_i)^T S_i^{-1}(x-\mu_i)=(x-\mu_i)^T(x-\mu_i)$$

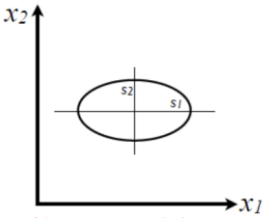
当S是对角矩阵时：

$$d_M(x,C_i)=(x-\mu_i)^T\begin{bmatrix}\frac{1}{s_1^2} & \cdots & 0\\ \vdots & \cdots & \vdots\\ 0 & \cdots & \frac{1}{s_m^2}\end{bmatrix}(x-\mu_i)$$

$$=\sum_m\left(\frac{x_i-\mu_i}{s_i}\right)^2$$



等距图是一个球面

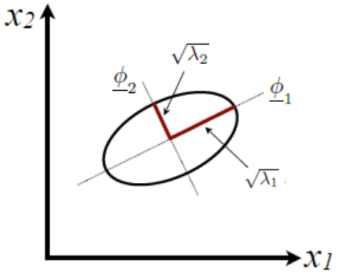


等距图是一个超椭圆面

点对统计分布的距离测量

- **Mahalanobis距离**的属性

当S是任意值时：等距图是一个有方向的超椭圆面点



等距图是一个有方向的超椭圆面

ϕ：S的特征向量

λ：S的特征值

1.中心化：将数据点平移到原点。

2.旋转（由**特征向量**决定）：消除特征之间的相关性。

3.缩放（由**特征值**决定）：让不同方向上的方差变得相同（归一化）。

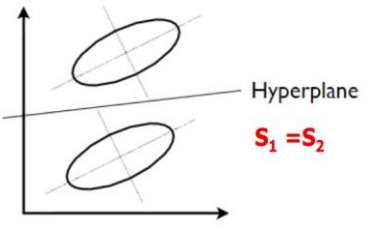
MICD分类器

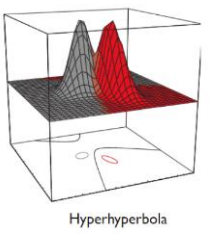
- MICD分类器：**Minimum Intra-class Distance Classifier**，基于Mahalanobis距离的分类器。

$$x \in C_1, \quad \text{iff } d_M(x, C_1) < d_M(x, C_2)$$

$$\text{iff } (x-\mu_1)^T S_1^{-1}(x-\mu_1) < (x-\mu_2)^T S_2^{-1}(x-\mu_2)$$

- 对于2个类而言，如果S是任意值，则MICD决策边界是一个超抛物面或者超双曲面。

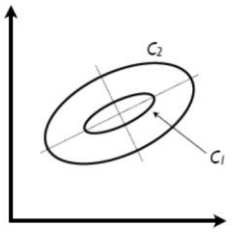


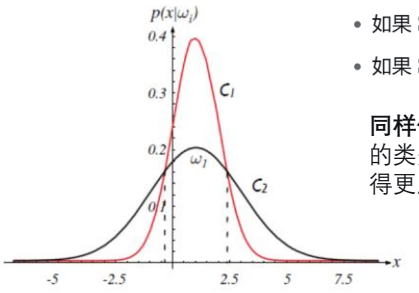


MICD分类器：缺点

- 如下情况，可以看出MICD分类器的一个缺陷。

$\mu_1 = \mu_2,\ S_1 \neq S_2$





$$D_c^2(\mathbf{x}) = (\mathbf{x} - \boldsymbol{\mu}_c)^T \mathbf{S}_c^{-1} (\mathbf{x} - \boldsymbol{\mu}_c)$$

- 如果 $\mathbf{S}_c$ **大**（数据分散），则 $\mathbf{S}_c^{-1}$ **小**
- 如果 $\mathbf{S}_c$ **小**（数据紧凑），则 $\mathbf{S}_c^{-1}$ **大**

同样偏离均值的距离，在方差大的类别中，马氏距离会被“打折”得更厉害。

MICD: $d_M^2(\underline{x}, C_1) > d_M^2(\underline{x}, C_2) \forall \underline{x}$

MICD分类器会选择方差较大的类

方差大的类别更容易被选中

- **类别A**：数据分布紧凑，协方差很小（类内紧密）
- **类别B**：数据分布分散，协方差很大（类内松散）

MICD分类器

假设一维特征，两类数据分布：

类别	均值 μ	方差 σ^2	标准差 σ
类别A	0	1	1
类别B	5	100	10

现在有一个待测样本 $\mathbf{x} = 3$:

计算马氏距离:

- 到类别A: $D_A = |3 - 0|/1 = 3$
- 到类别B: $D_B = |3 - 5|/10 = 0.2$

结果：虽然3离类别A的中心（0）更近（绝对距离3），离类别B的中心（5）更远（绝对距离2），但由于类别B的方差极大，马氏距离仅为0.2，**分类器判定它属于类别B**。

MAP分类器

- **MAP分类器(Maximum A Posterior Classifier)**：如果把一个类看作是一个概率分布，则判断准则可以设计如下：
- 给定一个测试模式，如果某一个类对于该测试模式的后验概率(posterior)最大，则表示该模式属于这个类。

$$x \in C_i, \quad \text{if } p(C_i \mid x) > p(C_j \mid x), \ \forall j \neq i$$

$$p(C_i \mid x): \text{后验概率}$$

- 根据**贝叶斯规则(Bayes' rule)**，后验概率是由先验概率(prior)和观测似然(observation likelihood)计算得到。

$$p(C_i \mid x) = \frac{p(x \mid C_i)p(C_i)}{p(x)}$$

$$\text{其中：} \ p(x) = \sum_j p(x \mid C_j)p(C_j)$$

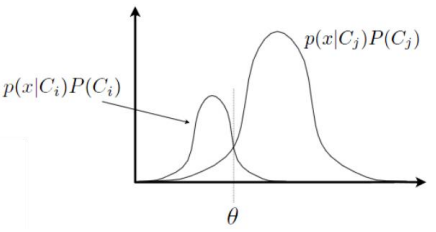
MAP分类器

- MAP分类器可以进一步表达为：

$$x \in C_i, \quad \text{if } p(x \mid C_i)p(C_i) > p(x \mid C_j)p(C_j), \ \forall j \neq i$$

- 对于2个类而言，决策边界位于：

$$p(x \mid C_1)p(C_1) = p(x \mid C_2)p(C_2)$$



- 使用MAP分类器时，**需要事先知道每个类的先验概率和每个类的观测似然函数**。这两类知识需要通过学习算法得到。

MAP分类器

- 如果观测似然函数是高斯函数：(以单变量为例)

$$p(x \mid C_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left[-\frac{1}{2}\left(\frac{x-\mu_i}{\sigma_i}\right)^2\right]$$

- 判断准则公式两端取对数：

$$p(\underline{x}|C_i)P(C_i) \underset{C_j}{\overset{C_i}{\gtrless}} p(\underline{x}|C_j)P(C_j)$$

$$log\left(\frac{p(\underline{x}|C_i)}{p(\underline{x}|C_j)}\right) \underset{C_j}{\overset{C_i}{\gtrless}} log\left(\frac{P(C_j)}{P(C_i)}\right)$$

17

MAP分类器

$$\left(\frac{x-\mu_j}{\sigma_j}\right)^2 - \left(\frac{x-\mu_i}{\sigma_i}\right)^2 \underset{C_j}{\overset{C_i}{\gtrless}} 2log\left(\frac{P(C_j)\sigma_i}{P(C_i)\sigma_j}\right)$$

- 在其它条件相同时，MAP分类器偏向于先验概率大的类。
- 在其它条件相同时，MAP分类器偏向于方差小的类，即紧密的类。
- 决策边界：

$$\left(\frac{1}{\sigma_j^2} - \frac{1}{\sigma_i^2}\right)x^2 - 2\left(\frac{\mu_j}{\sigma_j^2} - \frac{\mu_i}{\sigma_i^2}\right)x + \frac{\mu_j^2}{\sigma_j^2} - \frac{\mu_i^2}{\sigma_i^2} - 2log\left(\frac{P(C_j)\sigma_i}{P(C_i)\sigma_j}\right) = 0$$

18

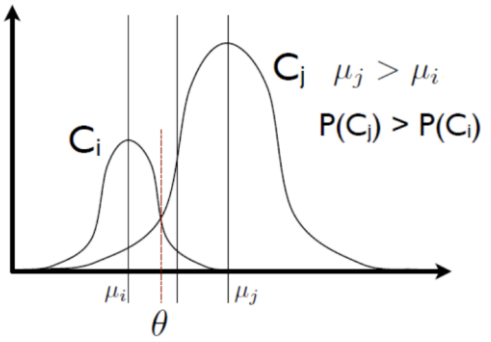
MAP分类器

- 两个类方差相同时：

$$\theta = \frac{\mu_i + \mu_j}{2} + \frac{\sigma^2}{\mu_i - \mu_j} log\left(\frac{P(C_j)}{P(C_i)}\right)$$

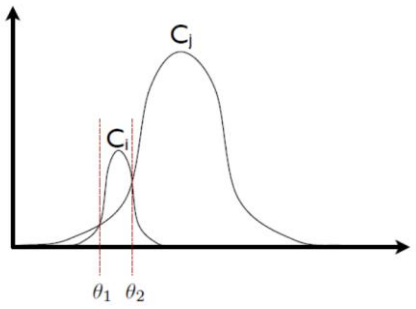
$$\theta_{MED} = \theta_{MICD} = \frac{\mu_i + \mu_j}{2}$$

- 两个类方差相同时，MAP分类器决策边界向具有较小先验概率的类偏移。

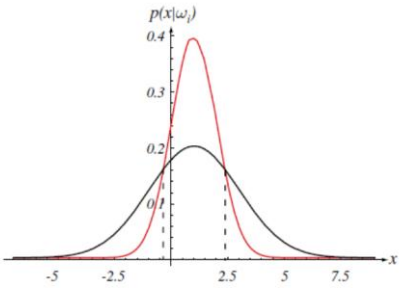


MAP分类器

- 两个类方差不相同时：决策边界有两个解。



- 在其它条件相同时，MAP分类器偏向于方差小的类。



$\mu_1 = \mu_2, \quad S_1 \neq S_2$

MAP分类器会选择方差较小的类

MAP分类器

- **类别A**：$\mu_A = (0,0)$，$\mathbf{S}_A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$（紧凑）
- **类别B**：$\mu_B = (5,5)$，$\mathbf{S}_B = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$（分散）

待测样本 $\mathbf{x} = (2,2)$：

马氏距离：

- $D_A^2 = 2^2 + 2^2 = 8$
- $D_B^2 = \frac{(-3)^2}{100} + \frac{(-3)^2}{100} = 0.09 + 0.09 = 0.18$

行列式项（二维时 $\log|\mathbf{S}| = \log(\sigma_1^2\sigma_2^2)$）：

- $\log|\mathbf{S}_A| = \log(1 \times 1) = 0$
- $\log|\mathbf{S}_B| = \log(100 \times 100) = \log(10000) \approx 9.21$

MAP判别值（等先验）：

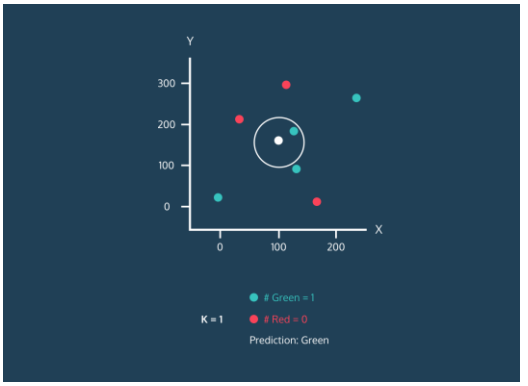
- $\delta_A = -\frac{1}{2} \times 0 - \frac{1}{2} \times 8 = -4$
- $\delta_B = -\frac{1}{2} \times 9.21 - \frac{1}{2} \times 0.18 = -4.605 - 0.09 = -4.695$

结果：$\delta_A > \delta_B$，判为A（紧凑类）。

虽然样本在绝对距离上离B中心更近（(2，2) 到 (5，5) 欧氏距离约4.24），但MAP选择了紧凑的A。

k近邻算法

- 工作原理：存在一个样本数据集合，并且样本集中每个数据都存在标签，
- 输入没有标签的新数据后，将新的数据的每个特征与样本集中数据对应的特征进行比较
- 然后算法提取样本最相似数据(最近邻)的分类标签。（只选择样本数据集中前k个最相似的数据）



距离度量 $|AB| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$

k-近邻算法步骤

1.计算已知类别数据集中的点与当前点之间的距离；

2.按照距离递增次序排序；

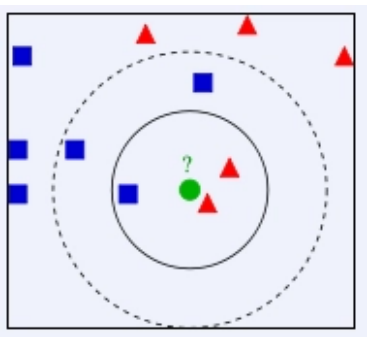
3.选取与当前点距离最小的k个点；

4.确定前k个点所在类别的出现频率；

5.返回前k个点所出现频率最高的类别作为当前点的预测分类。

k近邻(k-nearest neighbor, k-NN)

- 如果K=3，绿色圆点的最近的3个邻居是2个红色小三角形和1个蓝色小正方形，少数从属于多数，基于统计的方法，判定绿色的这个待分类点属于红色的三角形一类。
- 如果K=5，绿色圆点的最近的5个邻居是2个红色三角形和3个蓝色的正方形，还是少数从属于多数，基于统计的方法，判定绿色的这个待分类点属于蓝色的正方形一类。



k近邻方法

K 值	效果	风险
太小 (如 K=1)	模型复杂，决策边界曲折	容易**过拟合**，对噪声敏感
太大 (如 K=N)	模型简单，决策边界平滑	容易**欠拟合**，忽略局部特征
适中	平衡偏差和方差	✅ 推荐

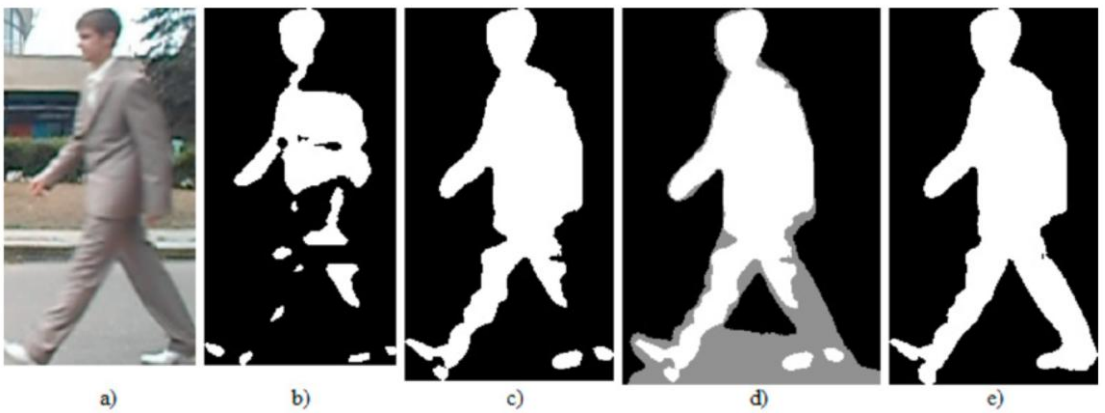
K 通常取奇数（避免投票平局）

K < √N（N 为样本总数）

常用范围：$3 \le K \le 10$

交叉验证：尝试多个 K 值，选择准确率最高的

k近邻：图像分割



Different parameters KNN-based Background/Foreground segmentation algorithm results: (a) original image; (b) History = 500, Dist2Threshold = 400, τ = 0.5; (c) History = 200, Dist2Threshold = 30; (d) with shadow; (e) τ = 0.75.

4.2无监督视觉特征学习

• 无监督学习：无标签/目标、无反馈、寻找数据中隐藏的结构

•**摆脱标注依赖**：人工标注成本高昂，且难以覆盖长尾、开放世界的概念。

•**利用海量数据**：互联网上的图像和视频几乎是无限的，无监督学习是挖掘这些数据的有效手段。

•**提升泛化能力**：通过无监督预训练，模型往往能学到更鲁棒的特征，在下游任务（如检测、分割）中表现优于有监督预训练（在数据量充足的情况下）。

27

常见方法

三大范式：生成式、对比式 和 掩码式。

（1）生成式：K-means算法、混合高斯模型、概率密度估计、**自编码器、变分自编码器、生成对抗网络**

（2）对比式：SimCLR、MoCo、SwAV

（3）掩码式：MAE、BeiT

评价指标：

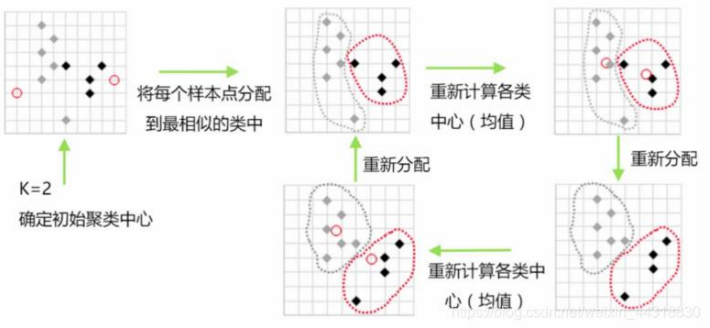
（1）线性评估：固定预训练好的骨干网络参数，只在其顶部训练一个线性分类器。这是最常用的标准，用于检验特征的**线性可分性**。

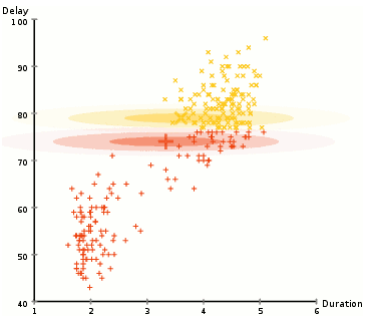
（2）微调评估：将预训练模型作为初始化，在下游任务（目标检测、语义分割、实例分割）上对整个网络进行全量微调，观察其收敛速度和最终精度。这更能反映特征在实际任务中的迁移能力。

28

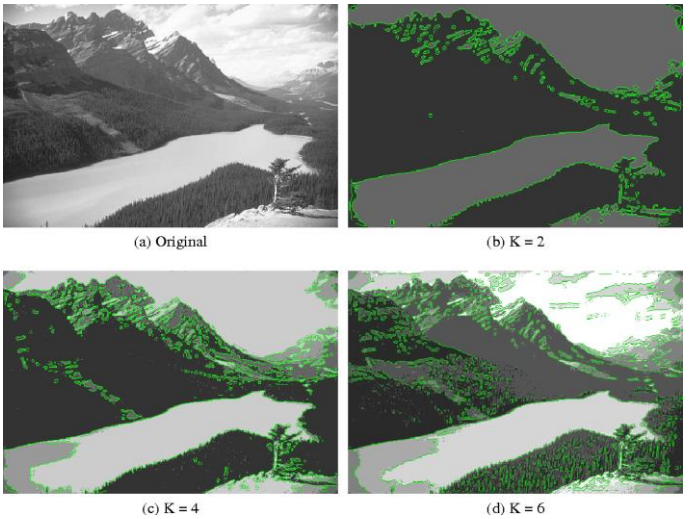
K-means算法

基于划分的聚类算法，它以 k 为参数，把 n 个数据对象分成 k 个簇，使簇内具有较高的相似度，而簇间的相似度较低。



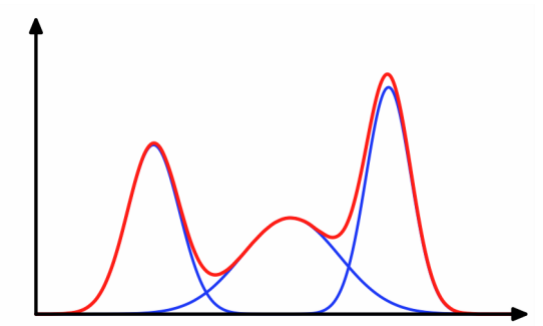


K-means 图像分割



高斯混合模型（GMM）

- 混合模型是一个可以用来表示在总体分布（distribution）中含有K个子分布的概率模型，换句话说，混合模型表示了观测数据在总体中的概率分布，它是一个由K个子分布组成的混合分布。混合模型不要求观测数据提供关于子分布的信息，来计算观测数据在总体分布中的概率。



31

单高斯模型

- 当样本数据 X 是一维数据（Univariate）时，高斯分布遵从下方概率密度函数（Probability Density Function）：

$$P(x|\theta) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(x-\mu)^2}{2\sigma^2})$$

其中 μ 为数据均值（期望），σ 为数据标准差（Standard deviation）。

- 当样本数据 X 是多维数据（Multivariate）时，高斯分布遵从下方概率密度函数：

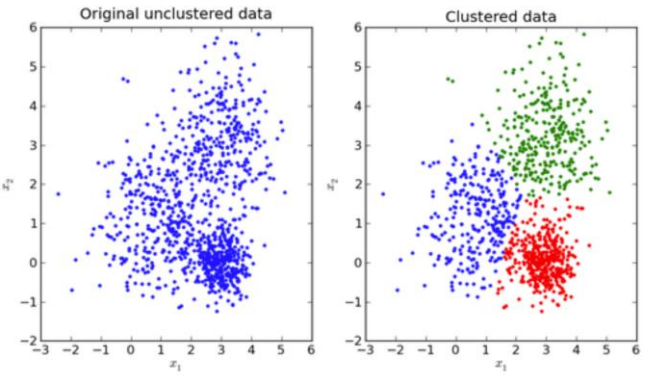
$$P(x|\theta) = \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma|^{\frac{1}{2}}} exp(-\frac{(x-\mu)^T \Sigma^{-1} (x-\mu)}{2})$$

其中，μ 为数据均值（期望），Σ 为协方差（Covariance），D 为数据维度。

32

高斯混合模型

- 高斯混合模型可以看作是由 K 个单高斯模型组合而成的模型，这 K 个子模型是混合模型的隐变量（Hidden variable）。
- 一般来说，一个混合模型可以使用任何概率分布，这里使用高斯混合模型是因为高斯分布具备很好的数学性质以及良好的计算性能。



图中每个点都由 K 个子模型中的某一个生成

高斯混合模型

- x_j 表示第 j 个观测数据，$j = 1, 2, \ldots, N$
- K 是混合模型中子高斯模型的数量，$k = 1, 2, \ldots, K$
- α_k 是观测数据属于第 k 个子模型的概率，$\alpha_k \geq 0$，$\sum_{k=1}^{K} \alpha_k = 1$
- $\phi(x|\theta_k)$ 是第 k 个子模型的高斯分布密度函数，$\theta_k = (\mu_k, \sigma_k^2)$。其展开形式与上面介绍的单高斯模型相同
- γ_{jk} 表示第 j 个观测数据属于第 k 个子模型的概率

高斯混合模型的概率分布为：

$$P(x|\theta) = \sum_{k=1}^{K} \alpha_k \phi(x|\theta_k)$$

对于这个模型而言，参数 $\theta = (\tilde{\mu_k}, \tilde{\sigma_k}, \tilde{\alpha_k})$，也就是每个子模型的期望、方差（或协方差）、在混合模型中发生的概率。

模型参数学习

- 对于单高斯模型，我们可以用最大似然法（Maximum likelihood）估算参数 的值，

$$\theta = argmax_\theta L(\theta)$$

- 这里我们假设了每个数据点都是独立的（Independent），似然函数由概率密度函数（PDF）给出。

$$L(\theta) = \prod_{j=1}^{N} P(x_j|\theta)$$

- 由于每个点发生的概率都很小，乘积会变得极其小，不利于计算和观察，因此通常我们用 Maximum Log-Likelihood 来计算（因为 Log 函数具备单调性，不会改变极值的位置，同时在 0-1 之间输入值很小的变化可以引起输出值相对较大的变动）：

$$logL(\theta) = \sum_{j=1}^{N} logP(x_j|\theta)$$

模型参数学习

- 对于高斯混合模型，Log-Likelihood 函数是：

$$logL(\theta) = \sum_{j=1}^{N} logP(x_j|\theta) = \sum_{j=1}^{N} log(\sum_{k=1}^{K} \alpha_k \phi(x|\theta_k))$$

- 如何计算高斯混合模型的参数呢？这里我们无法像单高斯模型那样使用最大似然法来求导求得使 likelihood 最大的参数，因为对于每个观测数据点来说，事先并不知道它是属于哪个子分布的（hidden variable），因此 log 里面还有求和，对于每个子模型都有未知的 $\alpha_k, \mu_k, \sigma_k$，直接求导无法计算。**需要通过迭代的方法求解**。

EM 算法

- EM 算法是一种迭代算法，1977 年由 Dempster 等人总结提出，用于含有隐变量（Hidden variable）的概率模型参数的最大似然估计。
- 每次迭代包含两个步骤：

1. E-step：求期望 $E(\gamma_{jk}|X,\theta)$ for all $j=1,2,\ldots,N$
2. M-step：求极大，计算新一轮迭代的模型参数

- 这里不具体介绍一般性的 EM 算法（通过 Jensen 不等式得出似然函数的下界 Lower bound，通过极大化下界做到极大化似然函数），只介绍怎么在高斯混合模型里应用从来推算出模型参数。

EM 算法

- E-step：依据当前参数，计算每个数据 j 来自子模型 k 的可能性

$$\gamma_{jk}=\frac{\alpha_k\phi(x_j|\theta_k)}{\sum_{k=1}^{K}\alpha_k\phi(x_j|\theta_k)},j=1,2,\ldots,N;k=1,2,\ldots,K$$

- M-step：计算新一轮迭代的模型参数

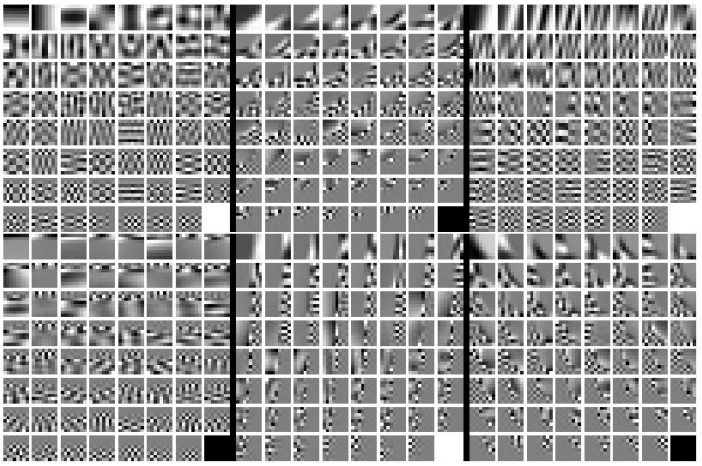
$$\mu_k=\frac{\sum_j^N(\gamma_{jk}x_j)}{\sum_j^N\gamma_{jk}},k=1,2,\ldots,K$$

$$\Sigma_k=\frac{\sum_j^N\gamma_{jk}(x_j-\mu_k)(x_j-\mu_k)^T}{\sum_j^N\gamma_{jk}},k=1,2,\ldots,K$$ （用这一轮更新后的 μ_k ）

$$\alpha_k=\frac{\sum_{j=1}^N\gamma_{jk}}{N},k=1,2,\ldots,K$$

- 重复计算 E-step 和 M-step 直至收敛 （$||\theta_{i+1}-\theta_i||<\varepsilon$，$\varepsilon$ 是一个很小的正数，表示经过一次迭代之后参数变化非常小）

自然图像块学习



从学习得到的高斯混合模型（GMM）中随机选取的6个协方差矩阵的特征向量，按特征值从大到小排序。

注意其结构的丰富性——有些特征向量看起来像PCA（主成分分析）的分量，而另一些则建模了纹理边界、边缘以及其他不同方向的结构。

D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. ICCV, 2011

基于自然图像块学习的图像去噪

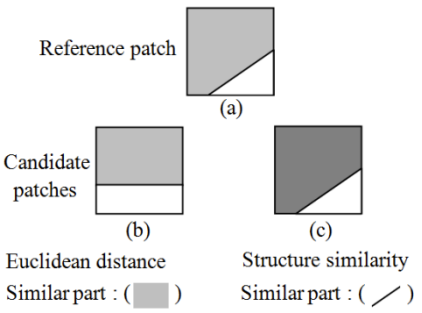


Table 1. Mahalanobis distance (MD) vs. Euclidean distance (ED).

σ_{noise}	Smooth		Structural		Textural	
	MD	ED	MD	ED	MD	ED
$\sigma = 35$	6.2	**8.1**	**36.4**	23.0	**65.6**	57.3
$\sigma = 55$	4.2	**7.2**	**24.5**	13.4	**53.1**	36.5
$\sigma = 75$	3.4	**6.9**	**17.0**	10.0	**43.0**	24.3

由于图像块空间并非像欧几里得空间那样的球形，因此使用由图像块协方差矩阵表征的马氏距离进行图像块相似性度量可能是更优的选择。

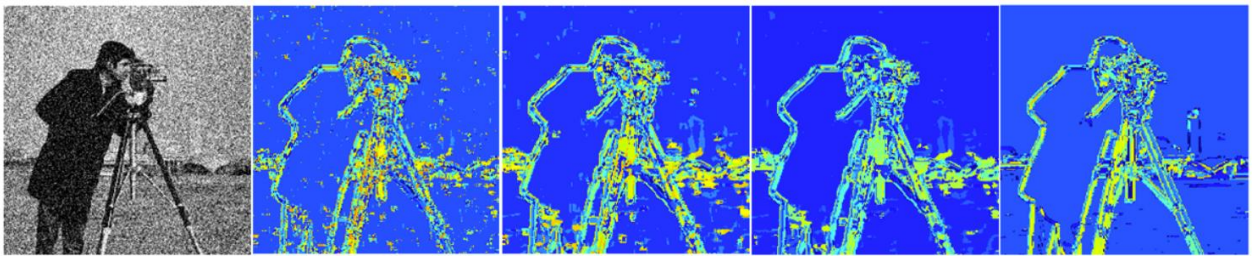
Chen, F., Zhang, L., & Yu, H. (2015). External Patch Prior Guided Internal Clustering for Image Denoising. *ICCV*, 2015.

基于自然图像块学习的图像去噪

Chen, F., Zhang, L., & Yu, H. (2015). External Patch Prior Guided Internal Clustering for Image Denoising. *ICCV*, 2015.

41

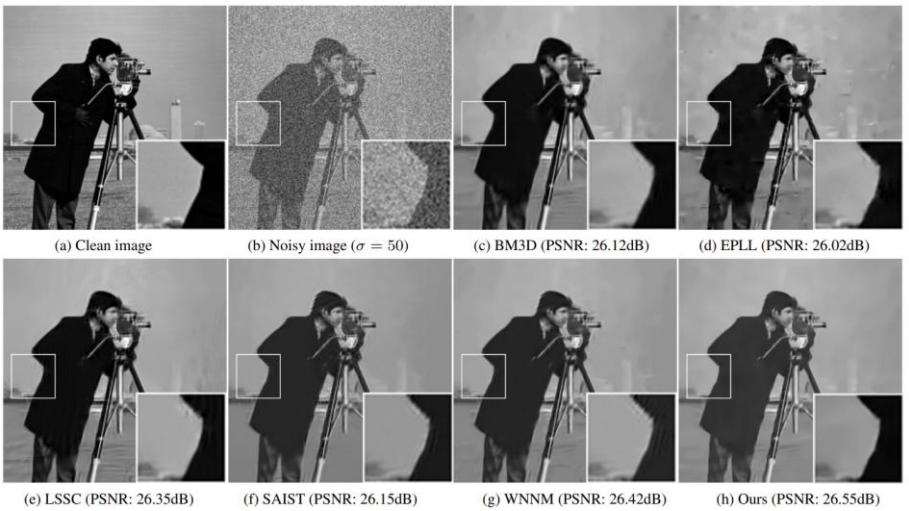
基于自然图像块学习的图像去噪



Chen, F., Zhang, L., & Yu, H. (2015). External Patch Prior Guided Internal Clustering for Image Denoising. *ICCV*, 2015.

42

基于自然图像块学习的图像去噪



(a) Clean image (b) Noisy image ($\sigma = 50$) (c) BM3D (PSNR: 26.12dB) (d) EPLL (PSNR: 26.02dB)

(e) LSSC (PSNR: 26.35dB) (f) SAIST (PSNR: 26.15dB) (g) WNNM (PSNR: 26.42dB) (h) Ours (PSNR: 26.55dB)

核密度估计

- 密度估计(Density Estimation)：如果概率分布形式未知，可以通过无参数(non-parametric)技术来实现概率密度估计。
- 直方图技术：将特征空间分为一系列的格子(bins），根据训练模式的特征值，累积相应的格子，最终得到所有训练模式的统计值。

- 直方图技术的问题：

1. 如何确定格子的数目。
2. 如何划分特征空间。

这两个问题是相互关联的，通过解决了一个问题，另一个问题也随之解决。人为确定或者根据训练模式自适应确定。

概率密度估计

- 核(kernel)技术：给定一个核函数(kernel function），给定训练模式的概率密度可以估计如下：

$$\hat{p}(x)=\frac{1}{N}\sum_{i=1}^{N}K_h(x-x_i)=\frac{1}{Nh}\sum_{i=1}^{N}K(\frac{x-x_i}{h})$$

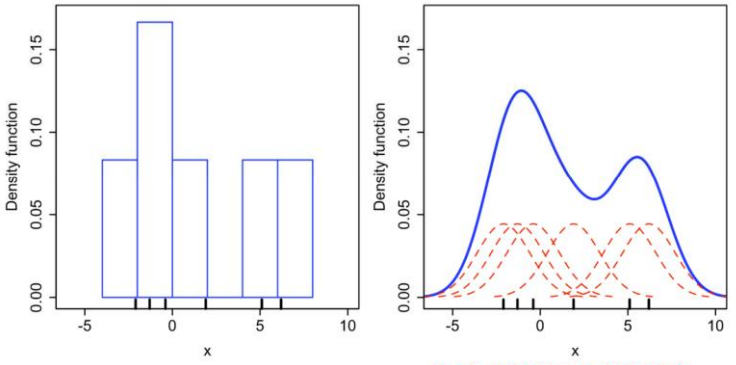
- $K(\cdot)$ 核函数，必须是对称的函数
- h 带宽，决定了特征空间分割的细化程度

$$K_h(\cdot):\text{尺度化的核函数，}K_h(x)=\frac{1}{h}K\left(\frac{x}{h}\right)$$

- 核函数可以是均匀分布函数、三角分布函数、高斯分布函数等该技术也称为Parzen Windows。

概率密度估计

- 核技术和直方图技术的对比：核技术的估计结果更加平滑。

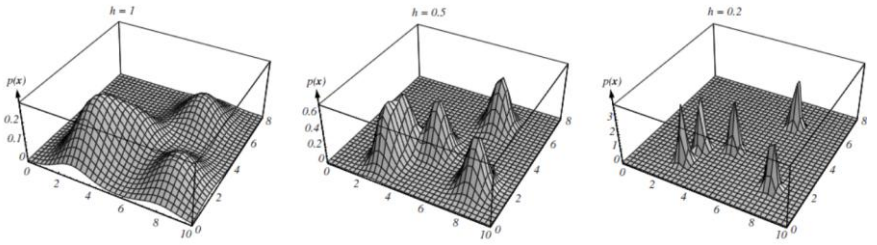


训练模式个数为N=6

红色曲线代表6个高斯核，
蓝色曲线是估计的分布

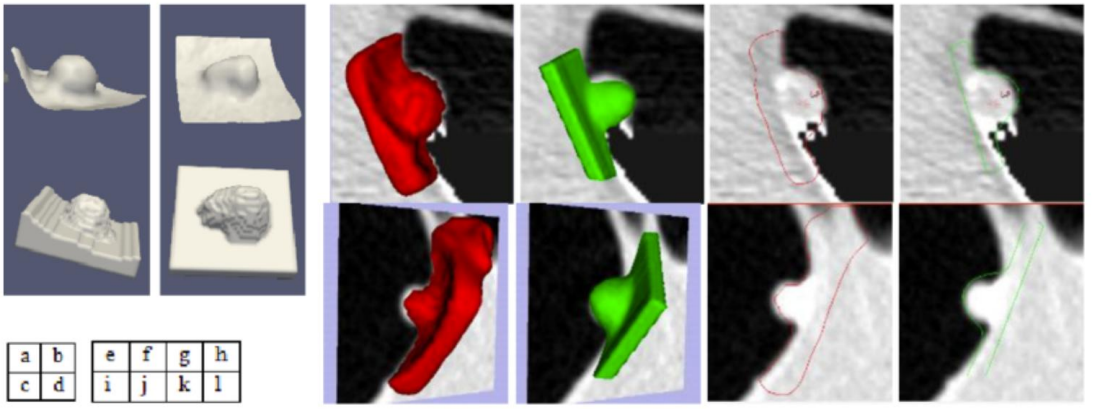
概率密度估计

- 核技术中的带宽选取原则：
- Generalization (泛化能力)：因为给定的训练模式对于实际的概率分布而言，数量是很少的、是稀疏的，所以要求根据这些训练模式估计出来的概率分布既能够符合这些训练模式之间的关系，同时也要有一定预测能力，即也能估计未看见的模式。



47

形状先验：目标分割



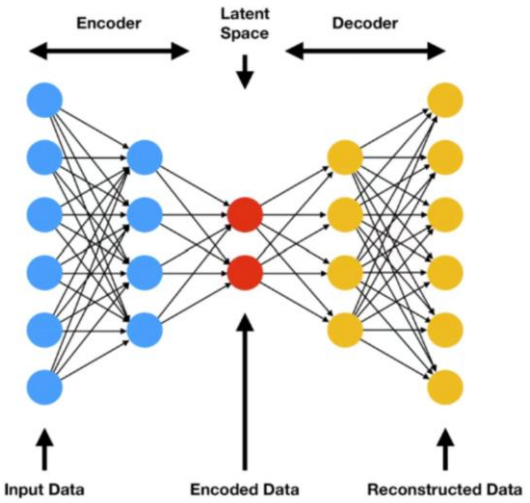
a	b
c	d

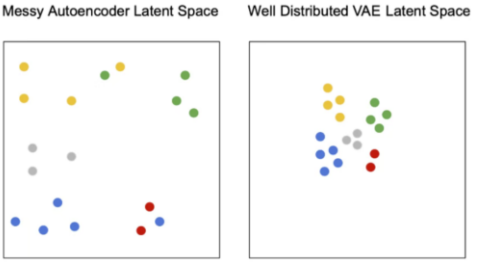
e	f	g	h
i	j	k	l

Left: construction of a shape-prior model (a-d). Right: segmentation results (e-l). The red surface (e, i) and curve (g, k) represent the zero-level active contour, and the green surface (f, j) and curve (h, l) represent the deformed and transformed shape-prior model.

48

自编码器（Auto-encoder)

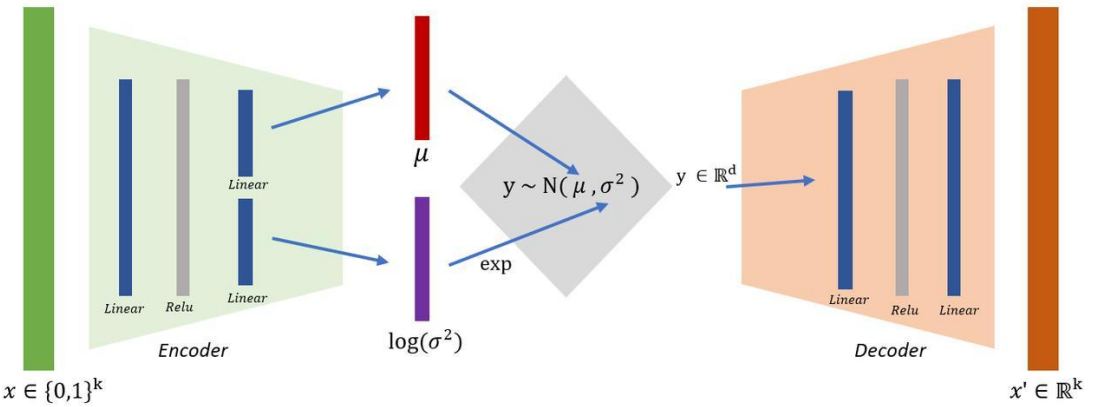




AE 与 VAE 隐空间对比

语义相似的数据点彼此相邻，而语义不同的点彼此远离

变分自编码器（Variational Auto-encoder, VAE)



VAE 的核心目标**不是压缩数据**，而是**学习数据的潜在概率分布，从而生成新样本**。

变分自编码器（Variational Auto-encoder, VAE）

$$\mathcal{L}(\phi,\theta) = -\mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right] + D_{KL}\left(q_\phi(z|x)\|p(z)\right)$$

第一项：重构损失

鼓励解码器能根据 z 还原出 x。

对于图像，常用均方误差或交叉熵。

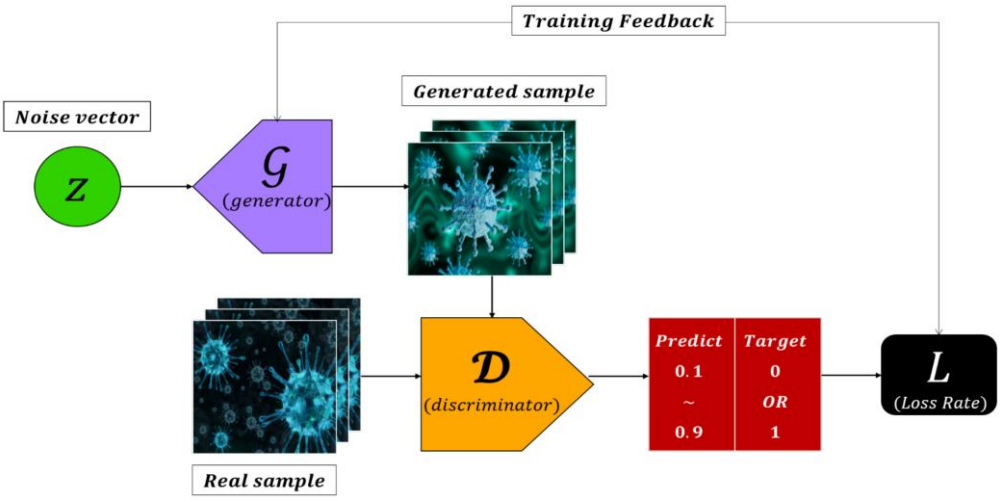
第二项：KL 散度

让编码器输出的分布 $q_\phi(z|x)$ 接近先验分布 $p(z)$（通常取标准正态分布 $\mathcal{N}(0,I)$）。

这是一个正则项，防止隐变量空间过度分散，并促使隐变量具有连续、可插值的特性。

	自编码器	VAE
输出	固定编码向量	编码向量的分布
隐空间	可能不连续、不规则	连续、平滑，可采样
生成能力	不能生成新样本	能从先验采样生成新样本
损失	重构误差	重构误差 + KL 正则
适用性	降维、特征提取	生成模型、隐空间操控

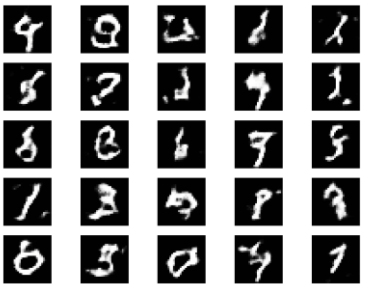
生成对抗网络



生成对抗网络

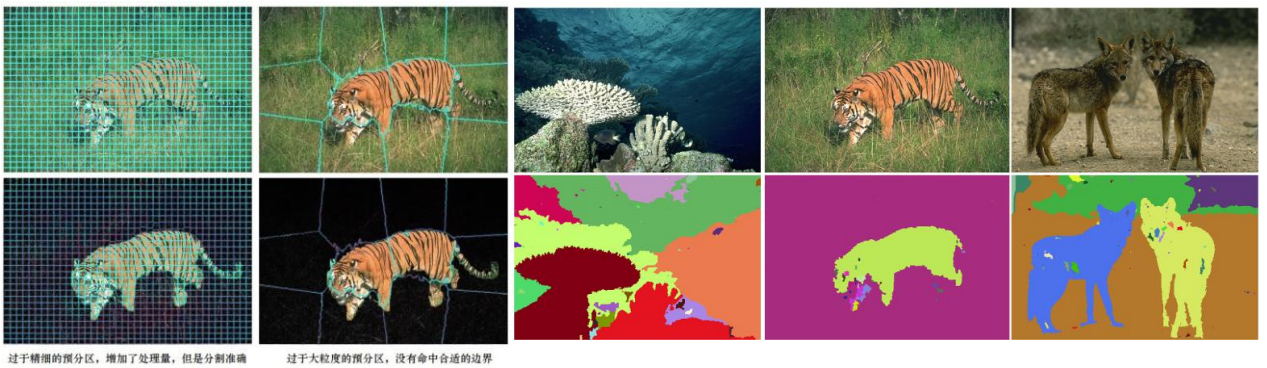
$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z)))]$$

样本类型	判别器理想输出	对目标函数的贡献
真实样本 x	$D(x) \to 1$	$\log(1) = 0$ （最大化）
生成样本 $G(z)$	$D(G(z)) \to 0$	$\log(1-0) = \log(1) =$



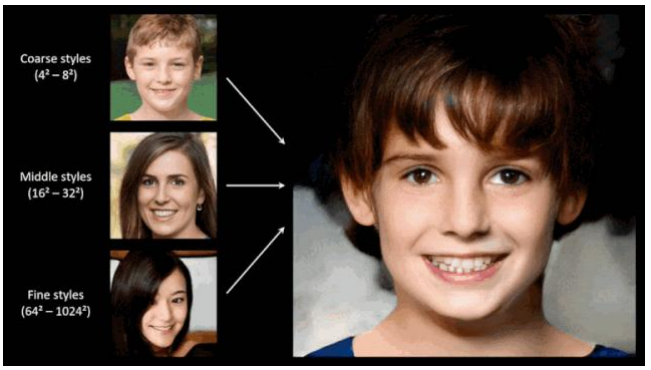
53

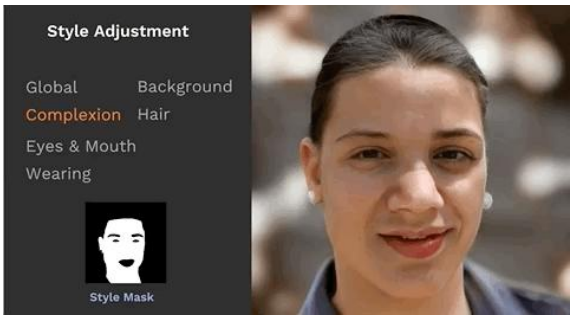
无监督图像分割



过于精细的预分区，增加了处理量，但是分割准确　　过于大粒度的预分区，没有命中合适的边界

54



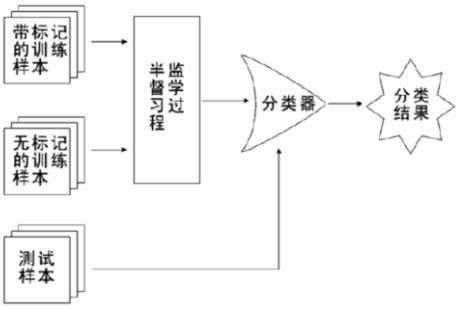


讨论

- 题目：有人认为“无监督学习是人工智能的‘暗物质’，虽然难以评估，但蕴含着真正的智能”；也有人认为“没有标签数据的监督学习只是‘花架子’，无法落地产生精确的商业价值”。请围绕以下争议展开辩论：
- “在解决实际工业问题时，无监督学习是否只是监督学习的‘辅助工具’（如用于降维、数据预处理），还是具备独立解决核心业务问题的‘主力军’地位？”

4.3半监督特征学习

- 生成模型中的半监督学习
- 基于平滑的半监督方法
- 伪标签”(Pseudo-label)
- Noisy Student
- 一致性正则化
- π-model



例如，大量医学影像，医生把每张片子上的每个病灶都标注出来再进行学习，是不可能的，能否只标注一部分，并且还能利用未标注的部分？

57

半监督学习

- 半监督学习就是一部分数据有标签，一部分数据没有标签，而通常情况下，无标签的数据数量远远大于有标签的数据。
- 有标签数据有R个，无标签数据有U个，那么上面就分别表示有标签数据和无标签数据。

有标签数据 $\{(x^r, \hat{y}^r)\}_{r=1}^{R}$

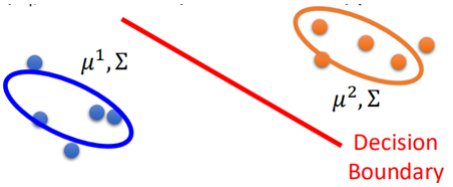
无标签数据 $\{x^u\}_{u=R}^{R+U}$

半监督通常是要基于一些假设，然后进行建模的，那么**半监督学习的效果好不好，就是假设的是否合理**。

58

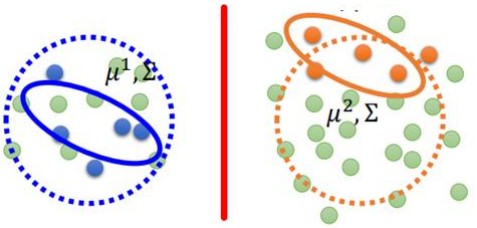
生成模型中的半监督学习

- 在监督学习中，概率生成模型用来分类的方法：假设数据集服从高斯分布，然后利用最大似然估计估算出样本分布的参数，然后对未知样本进行分类。



- 那么在半监督学习中，我们不仅有带有标签的数据，还会有大量的没有标签的数据，如图所示：

图中绿色的点是无标签数据，那么这些无标签的数据就会迫使原来的分类边界进行移动，变成竖线。



生成模型中的半监督学习

- **理论上**，在监督学习中，我们最大化目标函数：

$$logL(\theta) = \sum_{(x^r, \hat{y}^r)} logP_\theta(x^r|\hat{y}^r)$$

- 在半监督学习中，由于多了一部分无标签样本，因此**目标函数**变为：

$$logL(\theta) = \sum_{(x^r, \hat{y}^r)} logP_\theta(x^r|\hat{y}^r) + \sum_{x^u} logP_\theta(x^u)$$

- 但是我们并不知道无标签数据来自那一个class，就无法估测上面的概率，但是x^u可能来自于C1也可能来自于C2，那么：

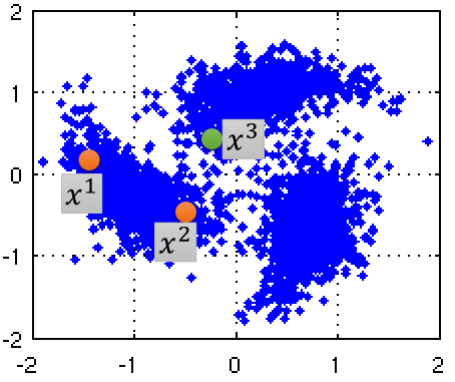
$$P_\theta(x^u) = P_\theta(x^u|C_1)P(C_1) + P_\theta(x^u|C_2)P(C_2)$$

- 无法直接进行求解，要利用EM算法进行迭代求解。

基于平滑的半监督方法

- 基于平滑的假设的思想就是：当x1和x2通过一个高密度区域相连，那么x1和x2就是相似的。

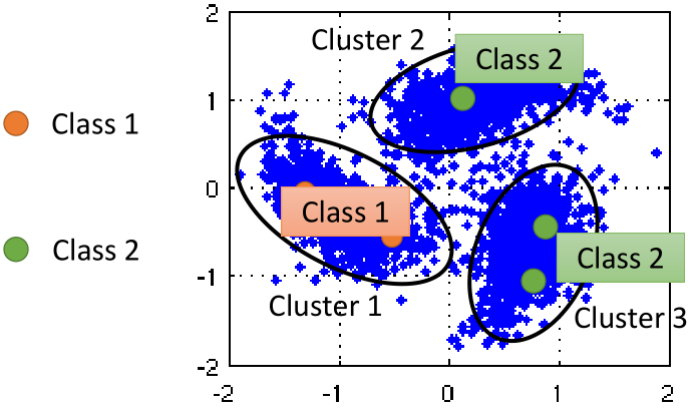
x1和x2从距离计算上来看相聚较远，而x2和x3相聚更近，然而x1和x2中间有一块高密度区域相连，那么就认为x1和x2更相似，而x2和x3则不相似。



61

聚类后再分类

- 这种方法比较直观，就是将所有的数据进行聚类，然后根据unlabeled data所属的簇中labeled data所属的类即为unlabeled data的类。



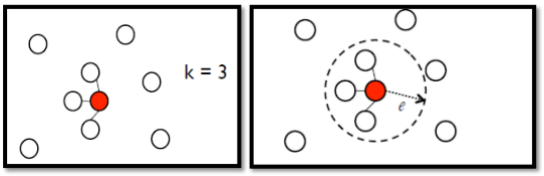
62

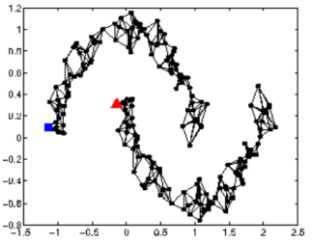
图方法

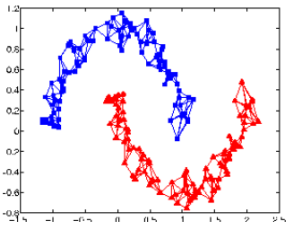
首先定义xi和xj之间的相似度

$$s(x^i, x^j) = exp\left(-\gamma \|x^i - x^j\|^2\right)$$

边的连接可以采用K-nearest neighbor或者e-neighborhood：







图拉普拉斯正则

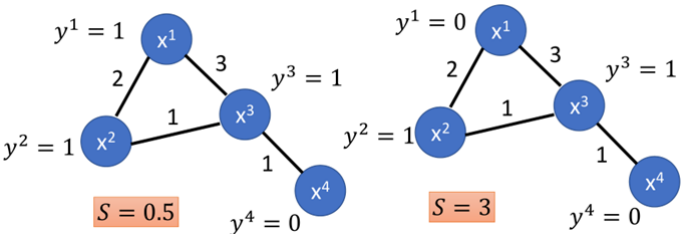
$$S = \frac{1}{2}\sum_{i,j} w_{i,j}(y^i - y^j)^2$$

Smaller means smoother

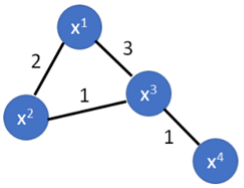
For all data (no matter labelled or not)

- S的值越小，表示图越光滑，越好，如下两个图，来计算一下哪一个图更光滑：

图的边就是通过RBF计算得到的权重的大小，y即为分类结果，那么显然左边的比右边的分类结果更合理，因为S更小。



图拉普拉斯正则



$$W=\begin{bmatrix}0&2&3&0\\2&0&1&0\\3&1&0&1\\0&0&1&0\end{bmatrix}\qquad D=\begin{bmatrix}5&0&0&0\\0&3&0&0\\0&0&5&0\\0&0&0&1\end{bmatrix}$$

图拉普拉斯矩阵 $L=D-W$

$$S=\frac{1}{2}\sum_{i,j}w_{i,j}\left(y^i-y^j\right)^2=\boldsymbol{y}^T L\boldsymbol{y}$$

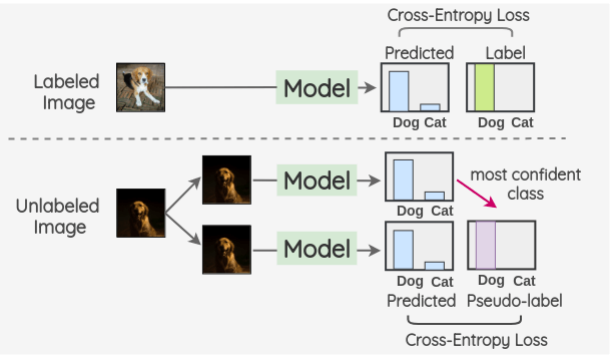
那么在进行训练时，由**原本的labeled data使得损失最小，还要使得越smooth越好，即S越小越好**，即损失函数变成了：

$$L=\sum_{x^r}C(y^r,\hat{y}^r)+\lambda S$$

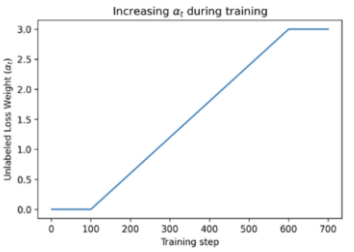
As a regularization term

伪标签”(Pseudo-label)

- 在有标签图像上使用交叉熵损失训练一个模型，利用该模型对无标签图像进行预测，并使用最大置信度类别作为伪标签，然后通过计算预测结果和伪标签之间的交叉熵损失来训练模型。



$$L=L_{labeled}+\alpha_t * L_{unlabeled}$$



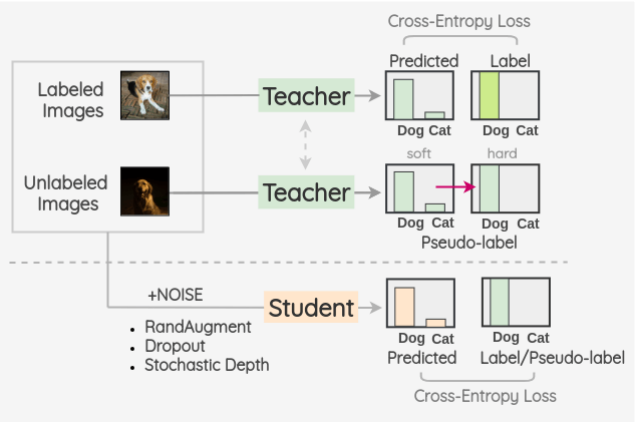
Amit Chaudhary. “Semi-Supervised Learning in Computer Vision.” https://amitness.com/2020/07/semi-supervised-learning (2020)

错误的伪标签会累积误差，导致模型退化

Noisy Student

- 训练两个独立的模型，即“教师模型”和“学生模型”。
- 在有标签图像上训练一个教师模型，用来给无标签图像打伪标签。
- 将有标签图像和伪标签图像混在一起，使用RandAugment、Dropout、Stochastic Depth添加噪声，训练一个学生模型。
- 学生模型训练好以后，将其作为新的教师模型，重复上述过程若干次。

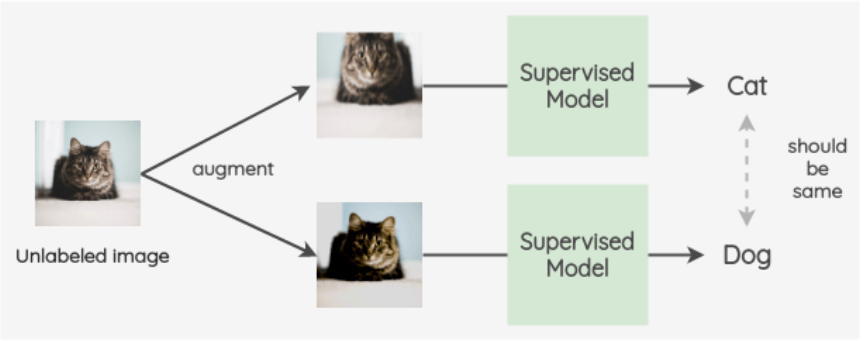
迭代式的自训练方法，
整个过程计算量巨大且耗时！



Xie et al. proposed a semi-supervised method inspired by Knowledge Distillation called “Noisy Student” in 2019.

一致性正则化(Consistency Regularization)

- 一致性正则化(Consistency Regularization)
- 通过要求模型对扰动前后的输入产生一致的预测，来利用未标注数据。



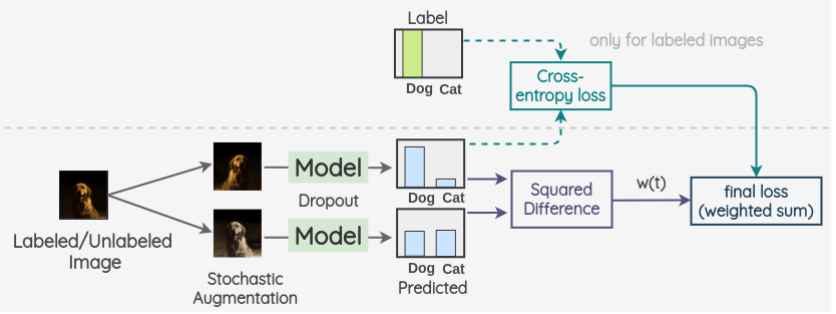
收敛缓慢、超参数敏感！

一致性正则化假设标注数据和未标注数据来自**相同的分布**。当未标注数据中包含标注数据中未出现的类别（**开放集半监督学习**）或存在领域漂移时，方法会失效。

π-model

- 对同一个未标注样本施加两次不同的随机数据增强（或噪声扰动），强迫模型对这两个增强版本的预测结果保持一致。

计算有监督损失（交叉熵）：$\mathcal{L}_s = \text{CrossEntropy}(f_\theta(\tilde{x}), y)$



高度依赖于**数据增强的质量**。

- 增强太弱 → 模型学不到有意义的不变性
- 增强太强 → 破坏语义，强制一致性反而有害
- 跨领域泛化差

This model was proposed by Laine et al. in a conference paper at ICLR 2017.

69

4.4自监督特征学习

- 在自然语言处理领域，以Mask Language Model(BERT) 和 Generative Pretraining(GPT) 为代表的自监督预训练模型，取得了突破性的进展。
- 要在深度神经网络中进行有监督学习，需要足够的标注数据。但是人工标注数据是一个既耗时又昂贵的过程。还有一些领域，例如医疗领域，获取足够的数据本身就很困难。因此，当前有监督学习范式的一个主要瓶颈是如何获得大量的标注数据。

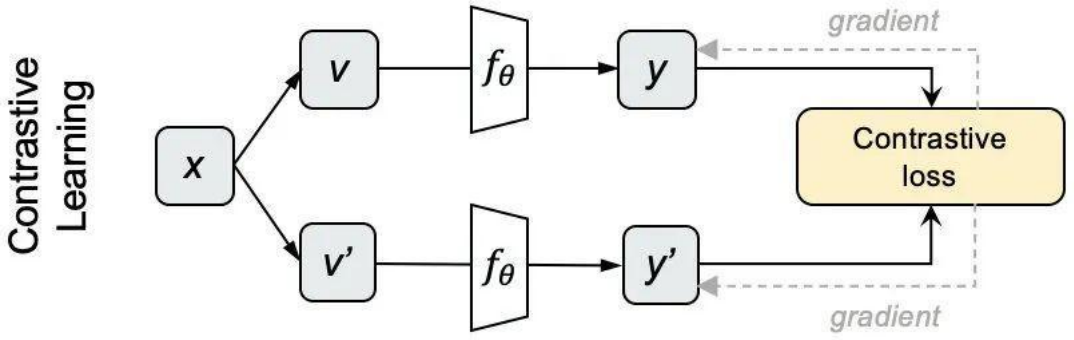
70

自监督学习方法

1.根据所有待预测部分之外的信息预测任意一部分信息。
2.根据过去预测未来。
3.根据过去最近的情况预测未来。
4.根据现在预测过去。
5.根据底层信息预测顶层信息。
6.根据可见的信息预测不可见的信息。
7.假设有一部分输入数据未知，并且对其进行预测。

SimCLR

SimCLR（**Simple Framework for Contrastive Learning of Visual Representations**）是谷歌大脑在2020年提出的自监督对比学习框架。它通过**拉近同一图片的不同增强视图、推远不同图片的视图**来学习视觉表示，在ImageNet上首次实现了自监督学习超越监督学习的效果。



图像着色

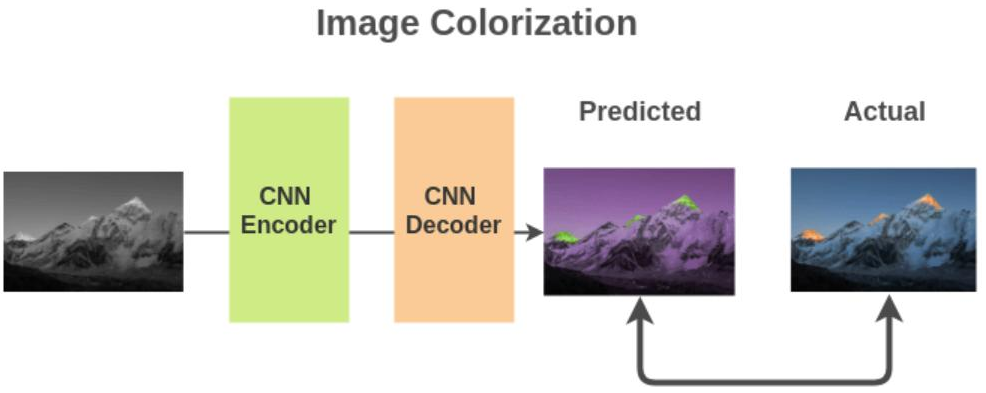
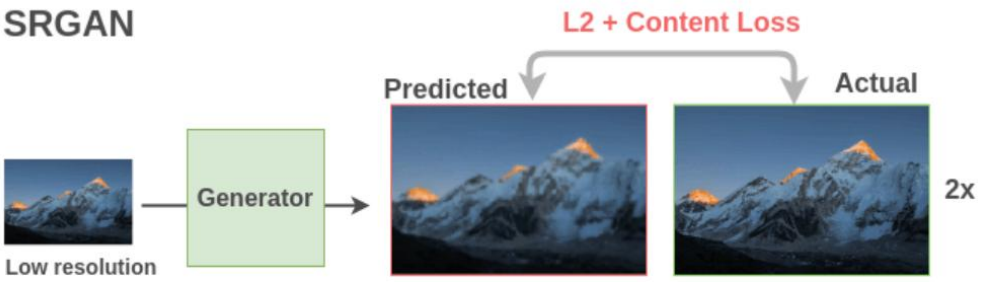
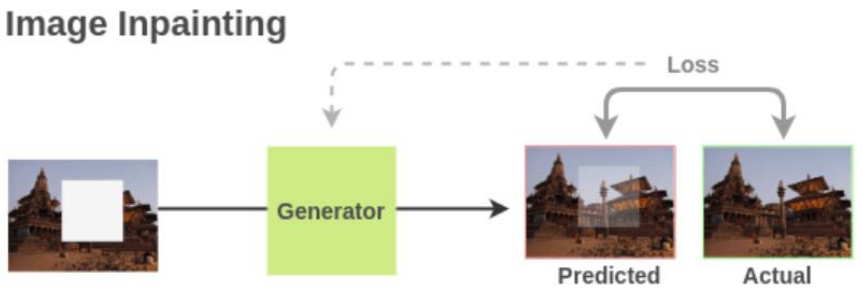


Image Colorization
CNN Encoder
CNN Decoder
Predicted
Actual

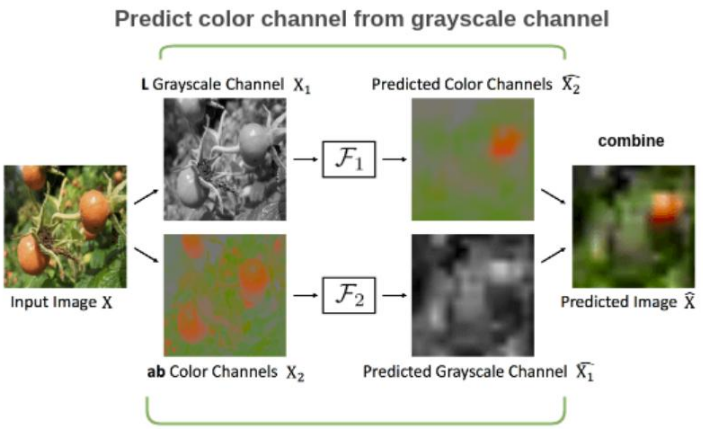
图像超分辨


SRGAN
L2 + Content Loss
Predicted
Actual
Generator
Low resolution
2x

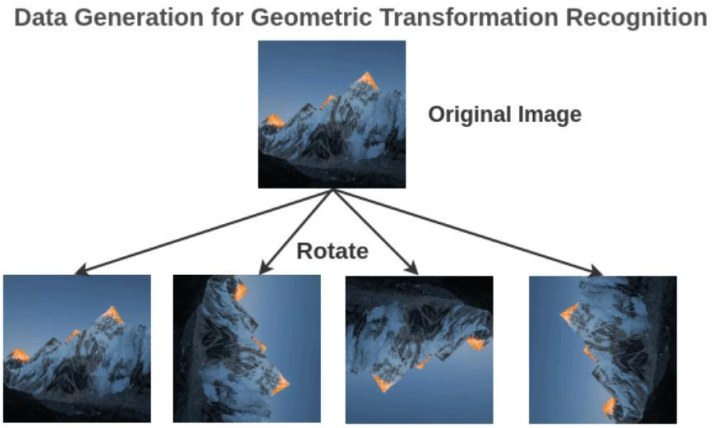
图像修补



Cross-Channel 预测

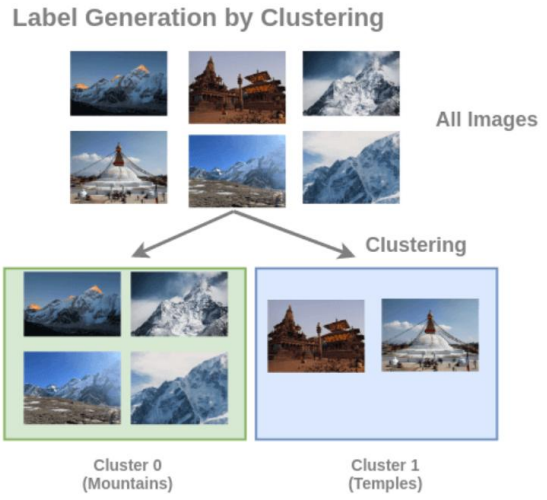


几何变换识别



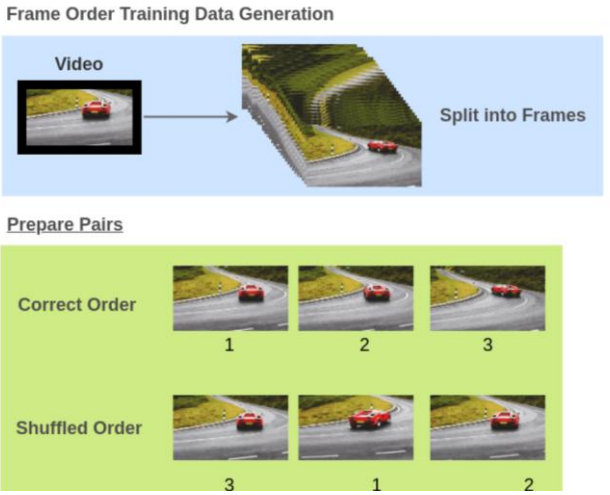
77

图像聚类



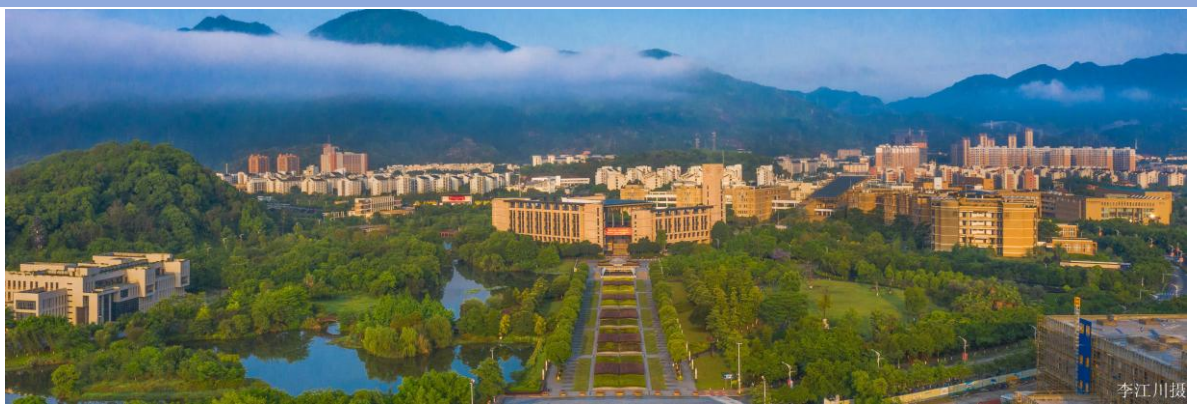
78

视频帧序



讨论

- **假设你是一家初创公司的首席数据科学家，公司刚拿到以下三个业务需求，预算只够先启动一个。你会选择先做哪一个？请说明理由，并阐述该任务属于监督学习还是无监督学习，以及具体的算法选型。**
- A需求（电商）： 想把用户分成不同的群体，以便针对不同群体发放不同风格的优惠券，但目前没有任何用户的标签数据。
- B需求（银行）： 希望建立一个系统，能在信用卡交易发生的瞬间，判断出该笔交易是否为“盗刷”。
- C需求（医疗）： 拥有海量的患者病历文本（非结构化数据），想探索是否存在某种尚未被医学界明确定义的、具有相似症状的新型疾病亚型。



Thank you!