提纲

- **计算机视觉**
  - 图像形成
  - 简单图像特征
  - 图像分类
  - 物体检测
  - 三维世界

引言

- 大多数使用视觉的智能体采用**被动传感**，不需要主动发出光就能看到景象
- **主动传感**涉及雷达或超声波等信号的发射，以及对反射进行感知
- **特征**是通过对图像进行计算而获得的一串数字，用来表示图像
- 计算机视觉的两个核心问题
  - **重建**，智能体从一幅或一组图像中建立一个关于世界的模型，
  - **识别**，智能体根据视觉信息和其他信息对所见到的物体进行辨识

图像形成

NANJING UNIVERSITY

无透镜成像：针孔照相机

- 图像传感器收集场景中物体散射的光，并构建二维图像
- 我们将整个图像平面称为一个传感器，但同时每个像素都是一个单独的微小传感器
- 聚焦图像：确保到达传感器的所有光子都来自世界中某个物 体上大致相同位置的点

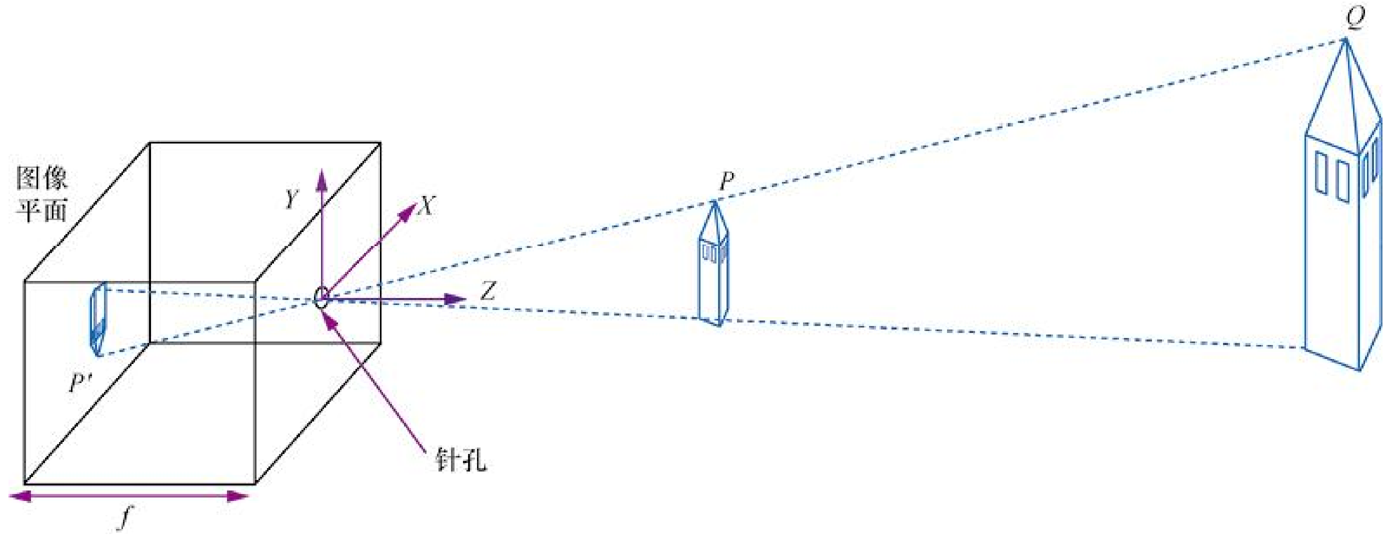
针孔照相机:

- 针孔照相机由盒子前部的针孔开口*O*和盒子后部的图像平面组成
- 这个开口也称作光圈，从针孔到图像平面的距离被称为**焦距**
- 只要物体在传感器的时间窗口内只移动一小段距离，我们也可以用针孔照相机来获得运动物体的聚焦图像
- 运动物体的图像会散焦，这种效应也称作**运动模糊**

图像形成

透视投影（perspective projection）

- 针孔照相机示意图



- 考虑场景中的点P，其坐标为(X, Y, Z)。P投影到图像平面上的点P′，其坐标为(x, y, z)。 若令f为焦距，即从针孔到像平面的距离，那么由相似三角形的性质可得

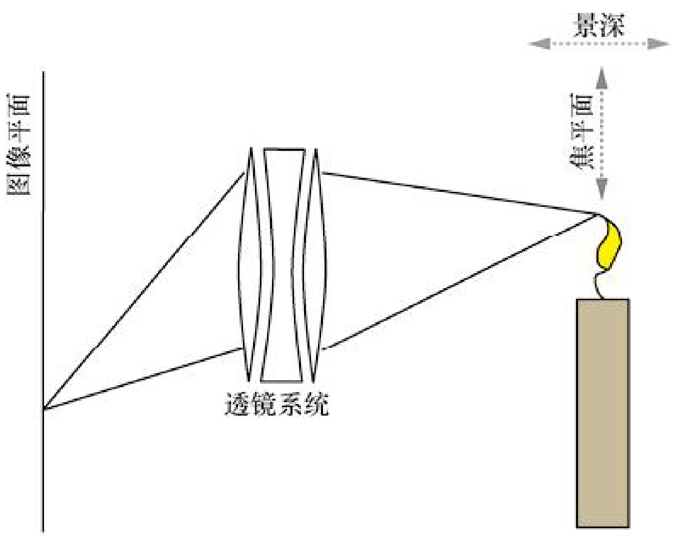
$$\frac{-x}{f}=\frac{X}{Z},\frac{-y}{f}=\frac{Y}{Z} \quad \Rightarrow \quad x=\frac{-fX}{Z},\ y=\frac{-fY}{Z}$$

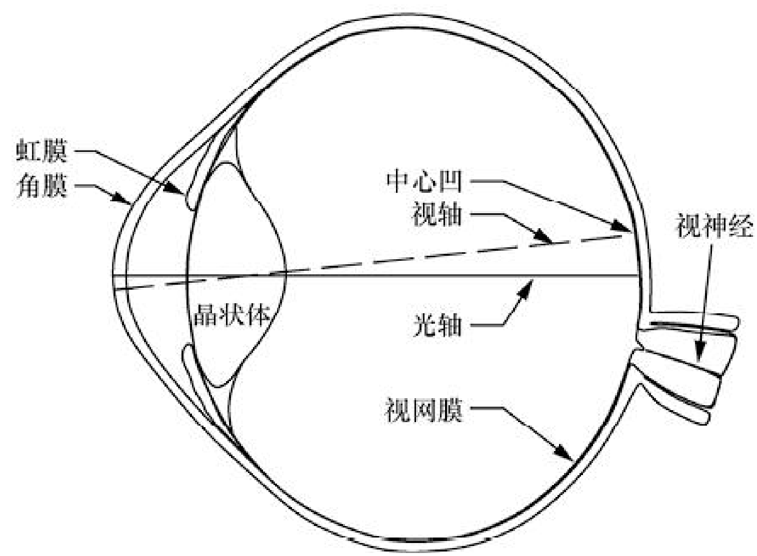
图像形成

NANJING UNIVERSITY

透镜系统

- 带透镜的照相机（或眼睛）将捕捉到所有照射到透镜上任何地方的光并将所有光聚焦到图像平面上的一个点
- 透镜的设计限制了它们只能聚焦距离透镜深度范围在Z以内的点上的光
- 这个范围的中心（聚焦最清晰的位 置）称作**焦平面**（focal plane）
- 聚焦能够保持足够清晰的深度范围称为**景深**（depth of field）
- 照相机的镜头光圈（开口）越大，景深越小





图像形成

光线与明暗

- 图像中像素的亮度是关于投影到该像素的场景表面切片的亮度的函数

- 不明确性的产生是因为有3个因素影响了从物体上的一个点到达图像的光量:
  - 环境光（ambient light）的总光强
  - 该点处于向光面还是阴影中
  - 从该点反射（reflect）的光的总量

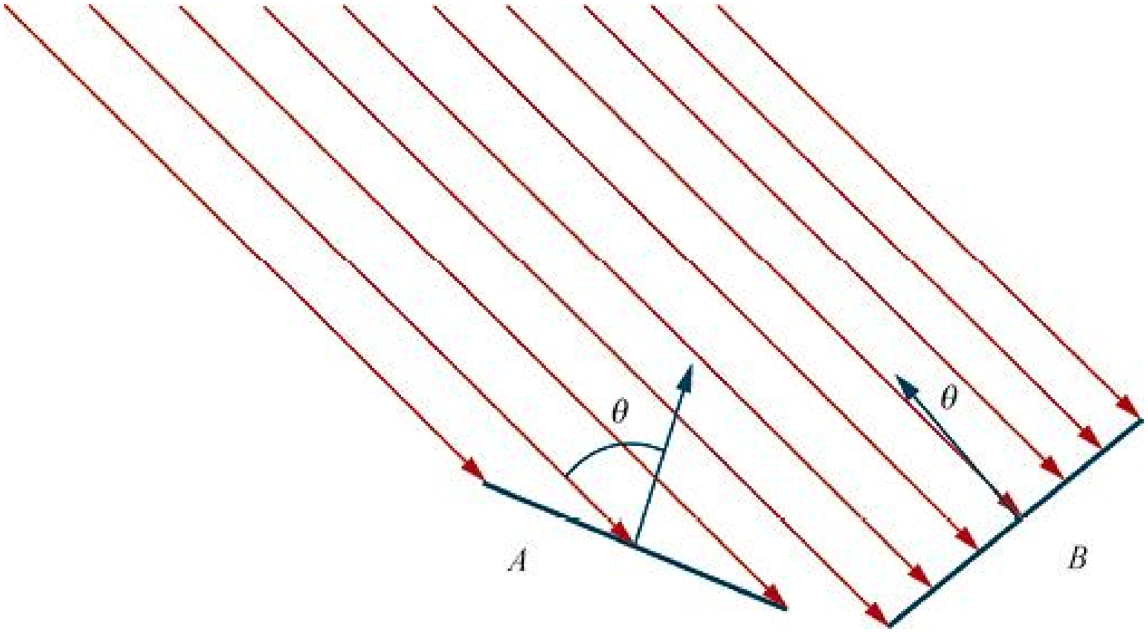
- **漫反射**将光均匀地散射到离开表面的各个方向上，因此漫反射表面的亮度不依赖于观察的方向

- **镜面反射**使得入射光以一定角度离开该表面，其方向由光到达的方向决定。

- 太阳是外界照明的主要来源，它的所有光线都从一个已知的方向平行地传播过来。
  - 采用**远点光源**来对这种行为进行建模

图像形成

光线与明暗



两个由远点光源照亮的表面切片，其中点光源的光线由带箭头的射线表示。

图像形成

光线与明暗

- 由远点光源模型进行照明的漫反射表面切片将反射它收集到的光的一部分，具体的比例由**漫反射系数**给出。其范围通常是0.05 ～0.95。**兰伯特余弦定律**表明，漫反射切片的亮度由下式给出：

$$I = \rho I_0 \cos\theta$$

  I_0为光源的光强，θ是光源方向和表面法线之间的夹角，ρ为漫反射系数。

- 各种照明的效果

图像形成

颜色

- **三原色原理**
  - 通过混合适量的3个原色（primary），来达到任何光谱能量密度下的视觉效果，无论它有多么复杂。

- **三原色**
  - 任何两种颜色的任意混合色都不会与第三种颜色相同.

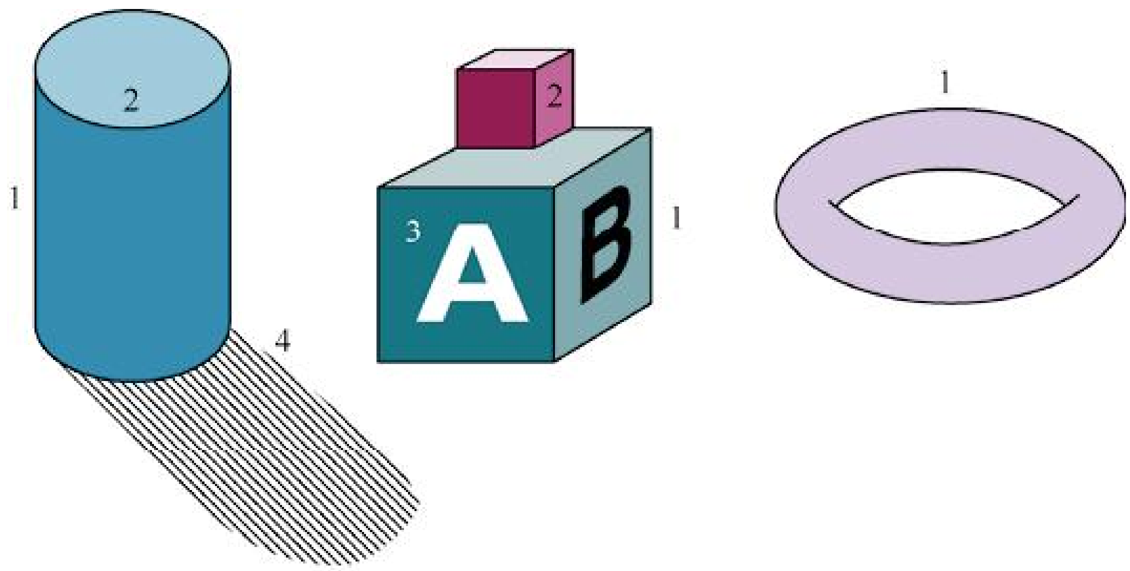
- 常见的选择是红原色、绿原色以及蓝 原色，简称RGB

- 大多数计算机视觉应用将一个表面建模为具有3种不同（RGB）漫反射系数，并将光源建模为具有3种（RGB）强度的模型。在此基础上，将兰伯特余弦定律应用于每个像素，以获得红、绿和蓝的像素值。

简单图像特征

边缘

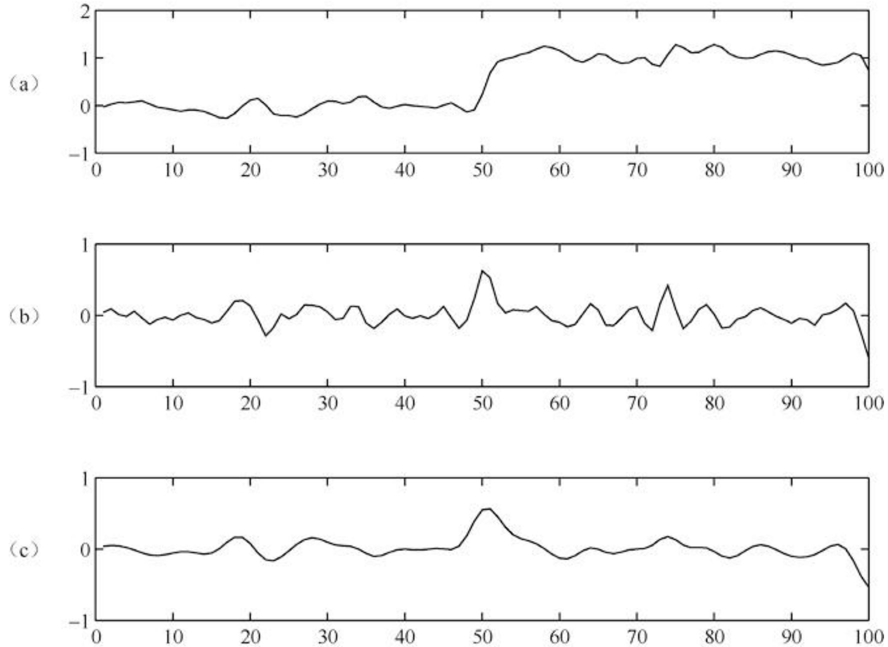
- 边缘是图像平面中的直线或曲线，图像亮度在此处发生了“显著”的变化。



不同类型的边缘：（1）深度不连续；（2）表面方向不连续；（3）反射不连续；（4）照明不连续（阴影）

边缘

- 对图像进行微分，然后寻找导数I'(x)较大的位置来对图像边缘进行辨别
    - 仅根据光强差异会因噪声而导致识别边缘时出现错误
    - **噪声：**像素值的变化与边缘无关



（a）一个跨过边缘的一维截面上的光强$I(x)$。（b）强度的导数$I'(x)$。此函数中的较大值对应于边缘，但函数中存在噪声。（c）经过平滑处理后的光强的导数。

简单图像特征

边缘

- 使用周围像素来控制噪声，从而进行平滑处理
- 使用附近像素的加权和作为 对像素“真实”值的预测，其中距离最近的像素的权重最大
- **高斯滤波器：**

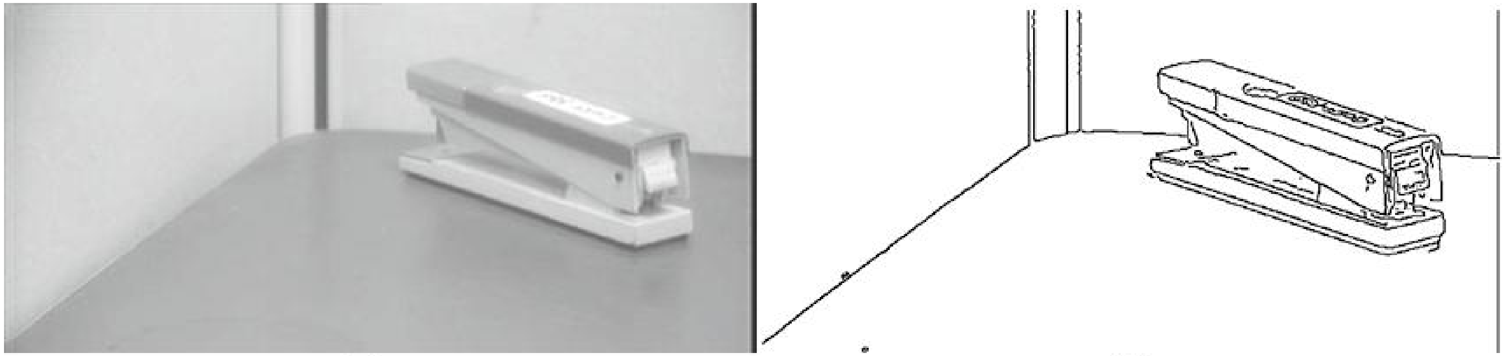
$$G_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/(2\sigma^2)} \quad （一维情况）$$

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/(2\sigma^2)} \quad （二维情况）$$

- 应用高斯滤波器意味着将光强$I(x_0, y_0)$替换为 $I(x, y)G_\sigma(d)$ 在所有像素 (x, y) 上的和, 其中 d 表示从 (x_0, y_0) 到 (x, y) 的距离。
- 平滑函数可以表示为图像与高斯核的卷积，进而可以将平滑和边缘检测融合到一个操作中，因为卷积的导数等于两函数之一的导数与另一函数进行卷积。

简单图像特征

边缘



（a）　　　　（b）

（a）一个订书机的照片。（b）从（a）中计算出的边缘

简单图像特征

纹理

- 在计算视觉中，**纹理**是指表面上可以被视觉感知到的图案。通常来说，这些图案大致上是规则的。
- 常见的粗糙的纹理模型是元素的模式重复，有时也称为**纹理元素**或**纹元**
- 纹理是图像切片的性质，而不是一个孤立像素的特性
- 纹理表示已经在两个重要任务中发挥了较大作用：物体识别和图像块匹配
- 纹理表示的一个基本构造方法
  - 给定一个图像切片，计算该切片中每个像素的梯度方向
  - 然后利用关于方向的直方图对该切片进行表征

光流

- **光流:** 当照相机与场景中的一个或多个物体之间发生相对运动时，图像中产生的视运动。它描述了观察者和场景之间的相对运动而导致的图像中特征的运动方向和速度。

- 相似性度量：**差值平方和(SSD)**

$$\mathrm{SSD}(D_x, D_y) = \sum_{(x,y)} (I(x,y,t) - I(x+D_x, y+D_y, t+D_t))^2 .$$

- 其中, (x, y) 表示以为(x_0, y_0)中心的像素块中的像素位置。寻找 (D_x, D_y)使SSD最小化。 (x_0, y_0)处的光流 $(v_x, v_y) = (D_x/D_t, D_y/D_t)$.

- 场景中应该存在一些纹理，从而使得图像中像素之间存在亮度的显著差异。

简单图像特征

光流

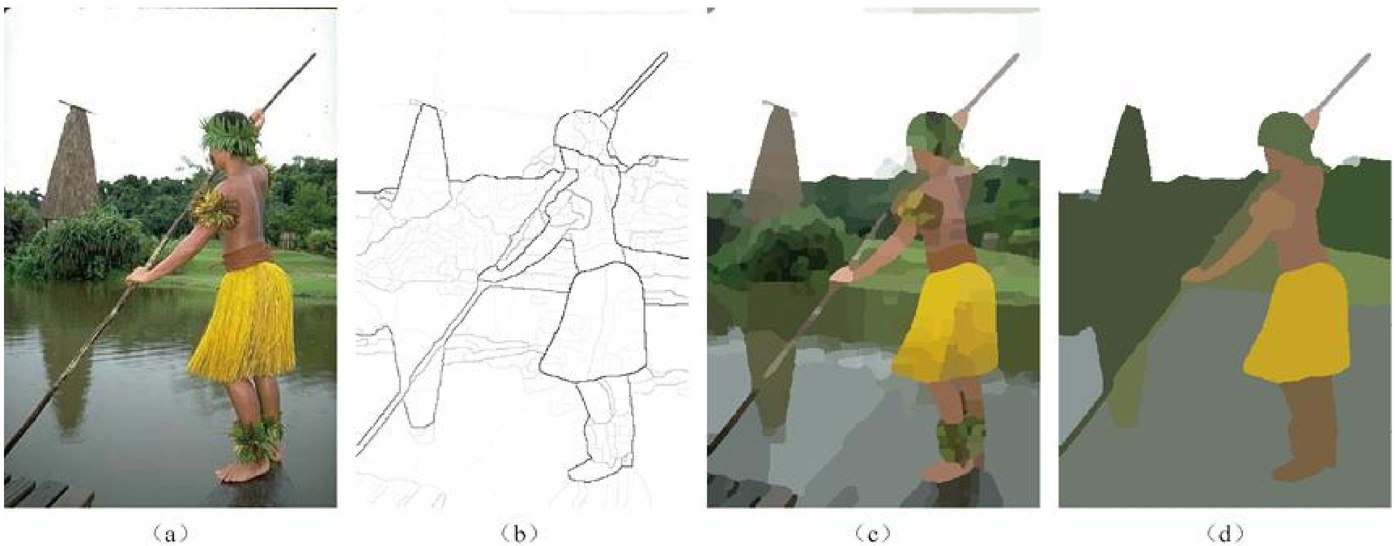
一个视频序列的两帧，以及与从一帧到另一帧的位移相对应的光流场。
注意由箭头方向刻画的网球拍和右腿的动作。

简单图像特征

自然图像分割

- **分割**是将一幅图像分解成若干组相似像素集的过程

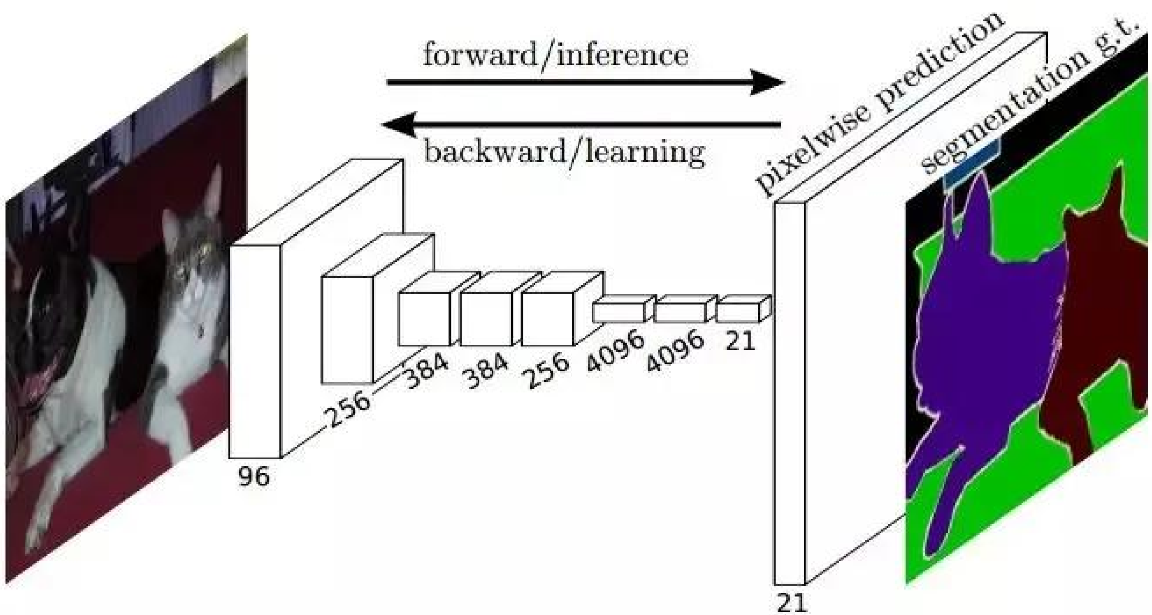
- 两种方法：检测边界或将像素聚类成多个区域
  - 分类问题
  - 像素聚类问题



（a）原始图像。（b）图像的边界轮廓。（c）通过对图像精细划分得到的各个分割区域。（d）通过对图像进行较粗糙的分割得到的各个分割区域。

简单图像特征

自然图像分割



Fully Convolutional Networks for Semantic Segmentation (CVPR 2015)

图像分类

- 现代的计算机视觉系统通过外观（例如颜色和纹理，而不通过几何性质）对图像进行分类。

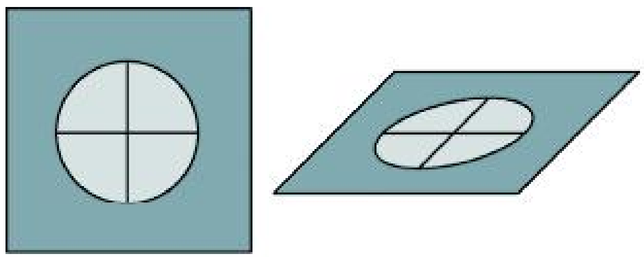
- 存在两个难点：

- 同一类别的不同实例可能看起来不同，也就是存在类内差异

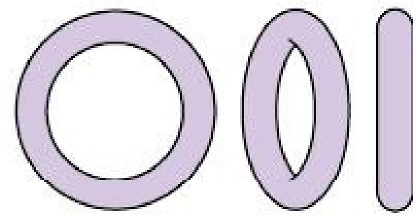
- 不同的时刻看起来可能不同，取决于以下几种效应
  - 光照
  - 透视收缩
  - 视角
  - 遮挡
  - 变形

- 现代方法通过使用卷积神经网络从大量的训练数据中学习表示和分类器来处理这些问题。在训练集足够丰富的情况下，分类器在训练中会多次看到任何一个重要的效应， 因此可以根据具体效应进行调整。

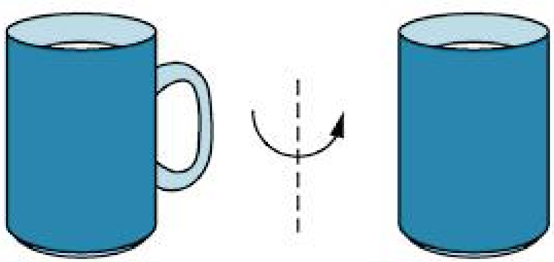
图像分类



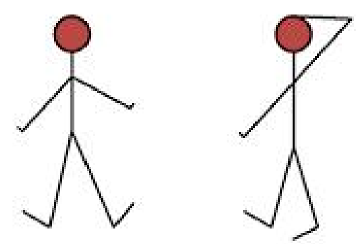
透视收缩



视向



遮挡



变形

产生外观变化的重要因素，它们可使同一物体的不同图像看起来不同。

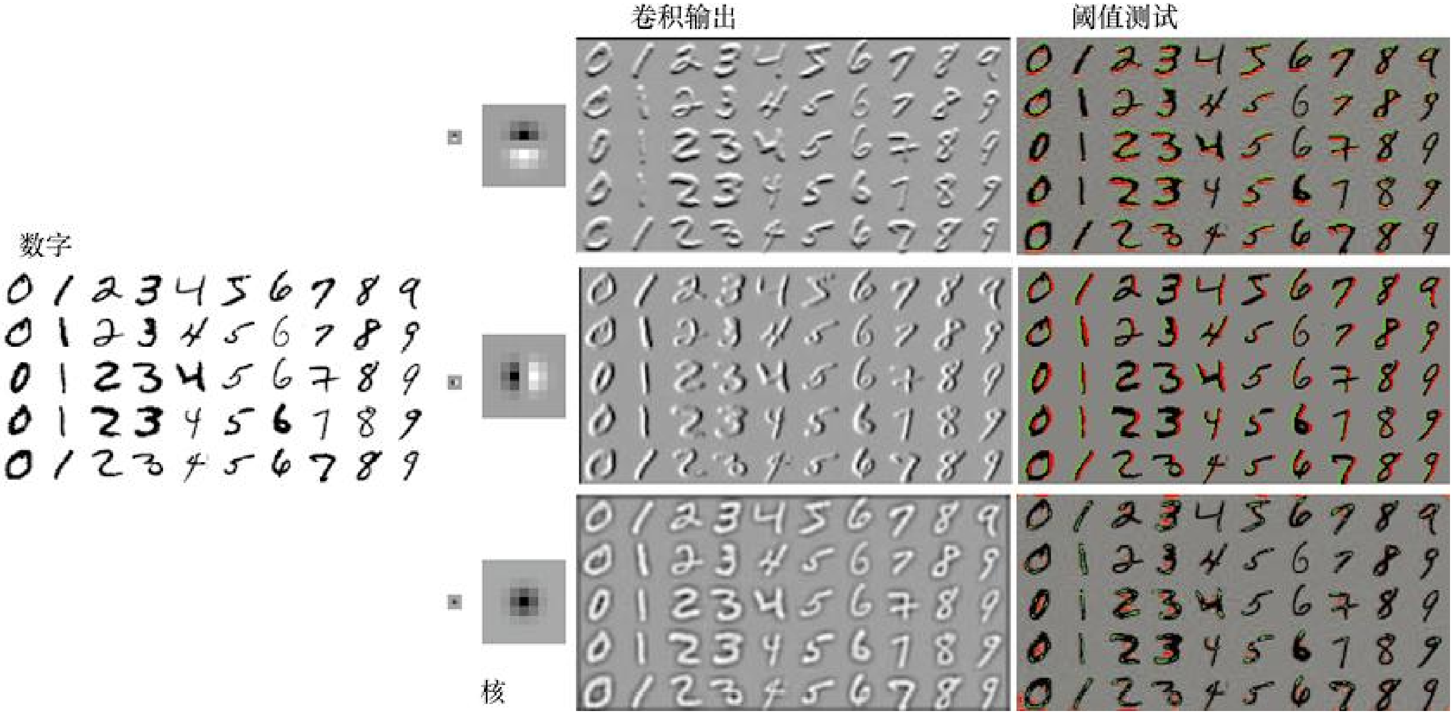
图像分类

基于卷积神经网络的图像分类

- 卷积神经网络（CNN）是非常成功的图像分类器。在有足够的训练数据和较好的训练技巧情况下，CNN产生了非常成功的分类系统
- 可以拍摄一幅关于数字的图像，并在不改变数字本身的情况下进行一些小的更改
- 局部模式可以提供相当多的信息
- 局部模式之间的空间关系也包含较多信息
- 卷积与ReLU 激活函数的复合看作一个局部模式检测器
  - 卷积将度量图像的每个局部窗口与核模式的相似程度
  - ReLU激活函数将低分窗口置为零，并突出高分窗口
- 多个卷积核的卷积可以找到多种模式
- 可以通过将新的一层应用于第一层的输出来检测复合模式

图像分类

基于卷积神经网络的图像分类



最左侧是MNIST数据集中的一些图像。中间图的左侧为3个卷积核。它们以实际大小（图中的小方块）给出，并放大以显示其内容：中度灰色的值为0，浅色表示正值，深色表示负值。中间图的右中侧给出了将左侧这些核应用于图像的结果。最右侧给出了响应大于阈值（绿色）与小于阈值（红色）的像素。

图像分类

基于卷积神经网络的图像分类

- **数据集增强：**对训练样本进行复制并稍加修改

- 将图像随机地移动、旋转或稍微拉伸， 或者将像素的色彩随机地进行少量调整

- 也可以把数据集增强的方法用于测试过程，而不是训练过程。

- 基于CNN 的分类器擅长忽略那些没有区分力的模式

- **环境或者上下文（context）：**物体上的模式可能是有区分力的

  - 例如一个猫玩具、一个带小铃铛的项圈或者一盘猫粮实际上可能有助于我们判断出所观察的物体是猫

物体检测

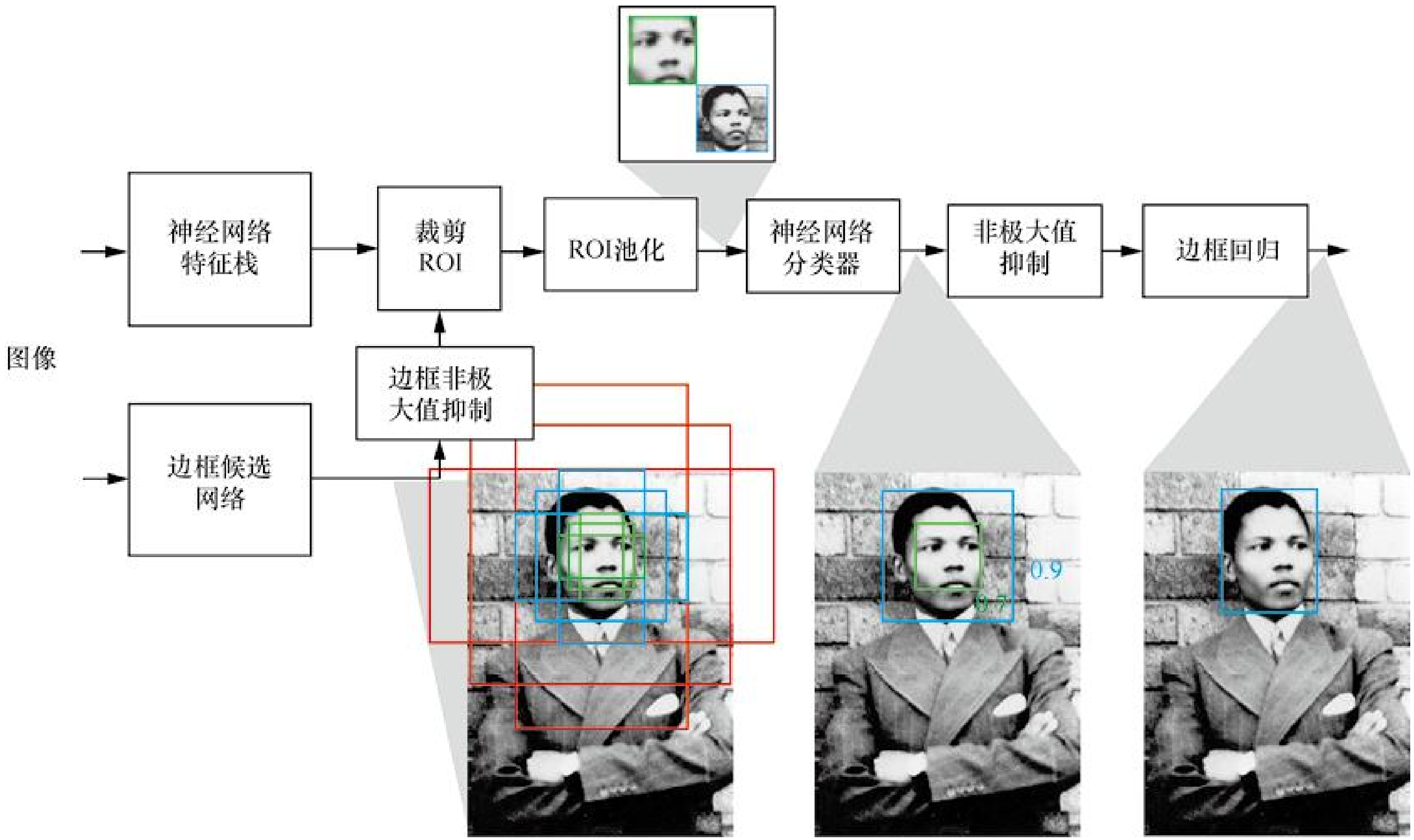
- **物体检测**器在一幅图像中寻找多个物体，分别判断每个物体属于什么类别，并通过在物体周围添加一个边框来反映出每个物体的位置。

- 构建一个物体检测器:
  - 可以通过在较大的图像上观察一个小的滑动窗口（一个矩形）
  - 在每个检测点上，我们使用CNN分类器对窗口中观测到的内容进行分类

- 细节:
  - 确定窗口的形状
  - 为窗口构建一个分类器
  - 决定要查看哪些窗口
  - 选择要报告的窗口
  - 利用这些窗口反映物体的精确位置

- 一个能找到包含物体的区域的网络称为**区域候选网络**（regional proposal network，RPN）
  - *Faster RCNN* 将大量边界框集合编码为固定大小的映射

物体检测



Faster RCNN使用两个网络：　一个网络用于计算候选图像框（称为“锚框”）的物体检测得分；　第二个网络是一个特征栈，用于计算适合分类的图像表示。

三维世界

- 在三维世界中，有两张关于同一物体的图片通常比只有一张要好：

- 如果你从不同的视角拍摄了同一场景的两幅图像，并且你对这两部摄像机了解得足够多，那么你可以通过计算第一个视图中的点对应第二个视图中的哪个点，并应用一些几何知识，来构建一个三维模型

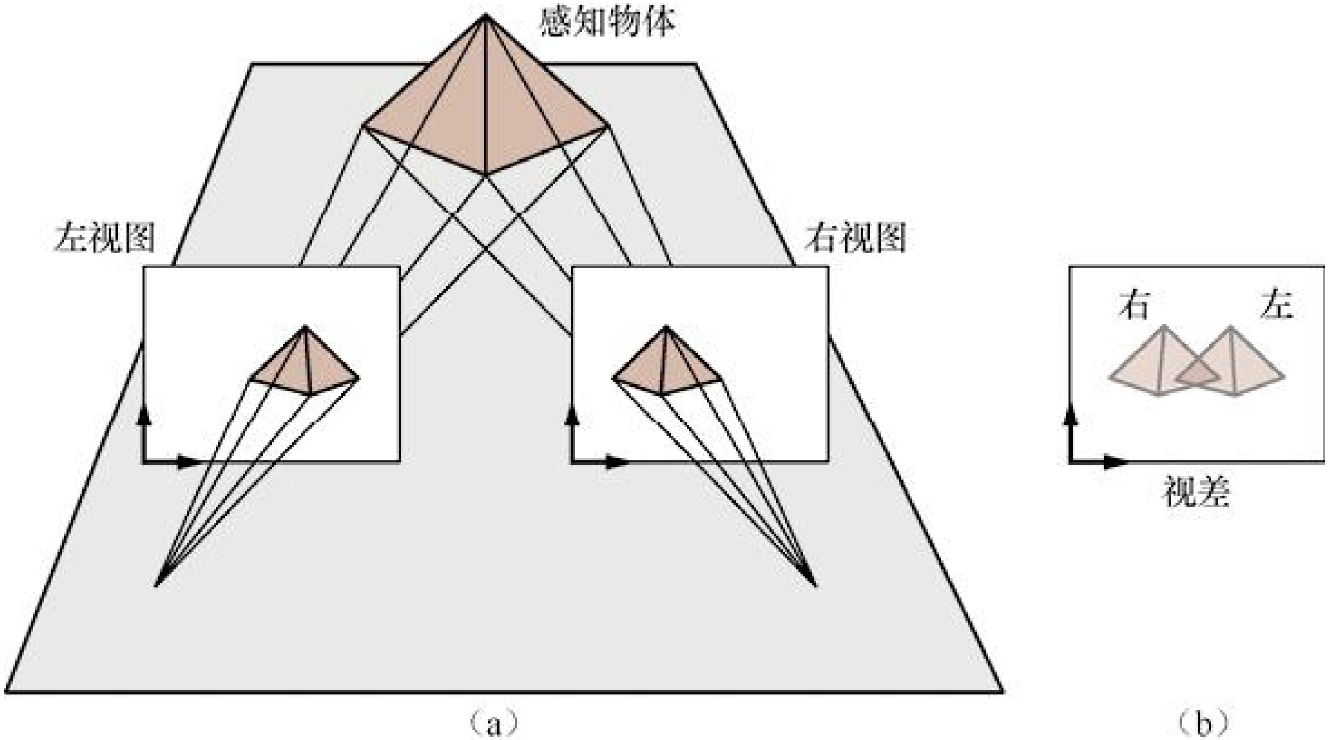
- 如果你有两个包含足够多点的视图，并且你知道第一个视图中的点对应第二个视 图中的哪个点，那么你不需要对摄像机了解太多就可以构建出该三维模型

- 关键的问题在于建立第一个视图中的点与第二个视图中的点的对应关系

- 通常有两种方法来获得一个场景的多个视图
  - 安置两部摄像机
  - 移动摄像机

三维世界

双目立体视觉



将摄像机按平行于图像平面的方式进行平移会导致图像特征在摄像机平面中发生移动。（a）位置上的差异是对物体深度的暗示。（b）如果我们对左右两幅图像进行叠加，我们将观察到视差。

三维世界

单个视图的三维线索

- 如果图片中有证据表明一个物体遮挡了另一个物体，那么遮挡另一个物体的物体将离眼睛更近。
- 纹理也是三维结构的重要线索。尽管纹理元素在场景中的物体上的分布可能是均匀的，但在透视图中，由于透视收缩，纹理看起来将是不均匀的。
- 明暗是三维形状的一个线索，如果一个表面的法线指向光源，则该表面会更亮；如果它背向光源，那么该表面会较暗。
- 物体之间的空间关系是另一个重要线索。

计算机视觉的应用

理解人类行为

从单一的图像中重建人类模型。**最左图**为一张图片，**中左图**为原图与重建出的身体叠加的图片，**中右图**为重建出的身体的另一个视图，**最右图**是重建出的身体的另一个不同视图。

计算机视觉的应用

南京大学
NANJING UNIVERSITY

理解人类行为

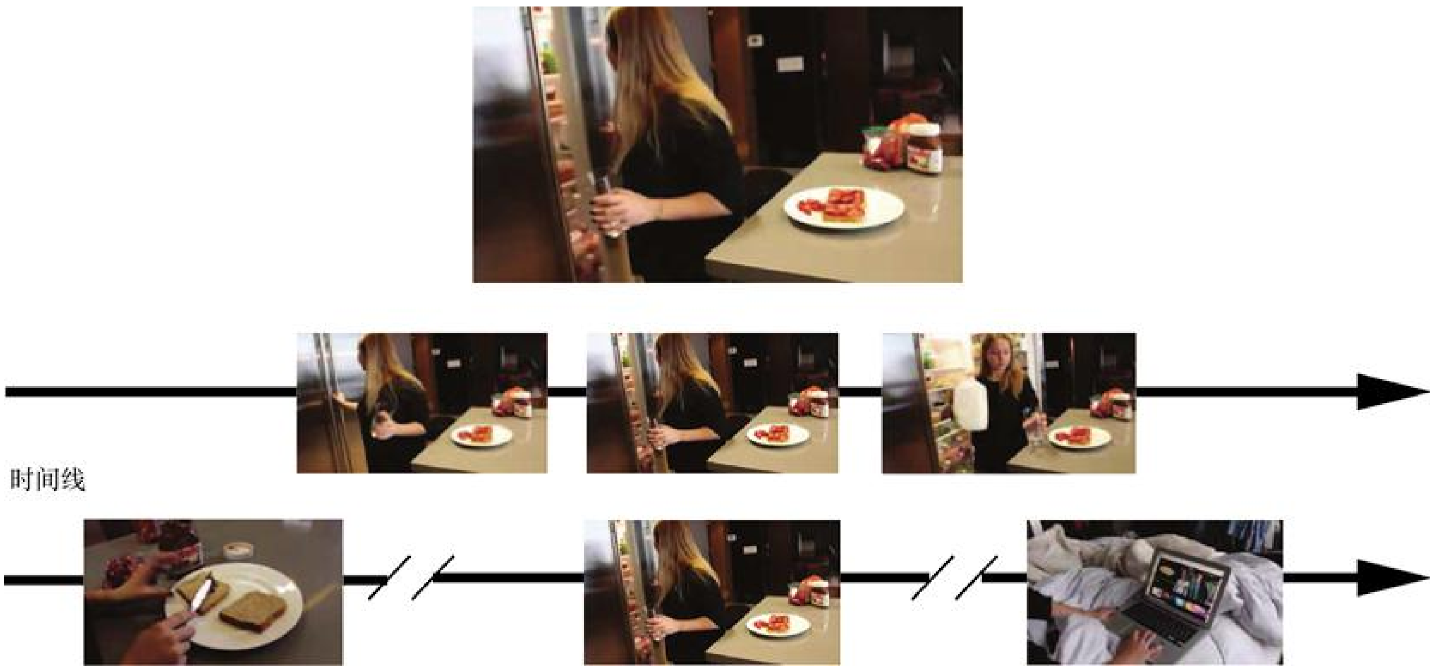
打开冰箱

从冰箱里
拿东西

同一个动作看起来很不一样，不同的动作看起来很相似。这些例子是来自一个数据集中的自然动作。上面3幅图表示标签为“打开冰箱”的样本，有的是特写，有的是远处拍摄。下面3幅图表示标签为“从冰箱里拿东西”的样本。

计算机视觉的应用

理解人类行为



我们所说的动作取决于时间尺度。对于最上面的单幅图像，最好的描述是“打开冰箱”。但是，如果你看完了一段视频短片（由中间一行图像表示），关于这个动作的最佳描述就是“从冰箱里拿牛奶”。如果你看完了一段较长的视频（由最下面一行图像表示），关于这个动作的最佳描述是“准备点心”。

计算机视觉的应用

匹配图片与文字

A baby eating a piece of food in his mouth | A young boy eating a piece of cake | A small bird is perched on a branch | A small brown bear is sitting in the grass

自动图像标题系统给出了一些好的结果和一些失败的结果。左边的两个标题很好地描述了各 自的图像，尽管 “eating … in his mouth”是一个不流畅的表达，这是早期标题系统所使用的循环神经网 络语言模型的一个相当典型的特点。根据右边的两个标题，我们认为标题系统似乎不了解松鼠，所以 从环境猜测该动物；它也没有意识到这两只松鼠在吃东西。

计算机视觉的应用

匹配图片与文字

Q. What is the cat wearing?
A. Hat

Q. What is the weather like?
A. Rainy

Q. What surface is this?
A. Clay

Q. What toppings are on the pizza?
A. Mushrooms

Q. How many holes are in the pizza?
A. 8

Q. What letter is on the racket?
A. w

Q. What color is the right front leg?
A. Brown

Q. Why is the sign bent?
A. It's not

视觉问答系统产生关于图像的自然语言问题的答案。**顶部：**该系统对有关图像的一些相当棘手的问题给出了非常合适的答案。**底部：**不太令人满意的答案。例如，系统被要求猜测比萨饼上的洞的个数，但系统并不知道什么算洞，而且洞本身很难计数。类似地，系统认为猫的腿的颜色是棕色，这是因为图片背景是棕色的，并且系统不能正确定位猫的腿。

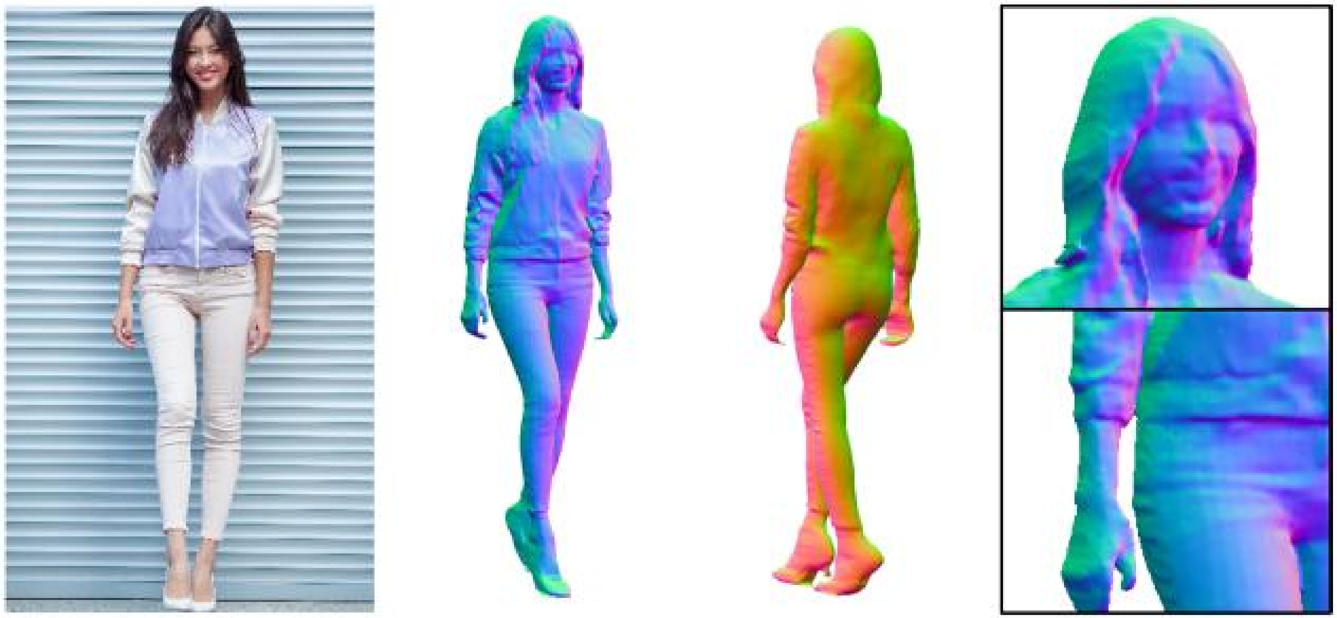
计算机视觉的应用

多视图重建

Representing Scenes as Neural Radiance Fields for View Synthesis
(ECCV 2020)

计算机视觉的应用

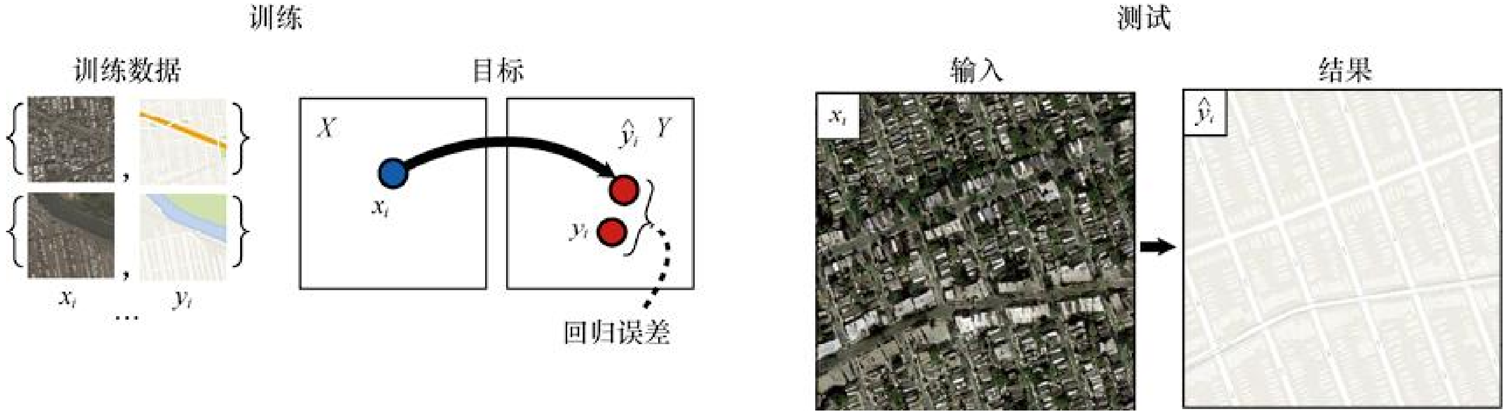
单视图中的几何



PIFuHD: Multi-Level Pixel-Aligned Implicit Function for High-Resolution 3D Human Digitization (CVPR 2020)

计算机视觉的应用

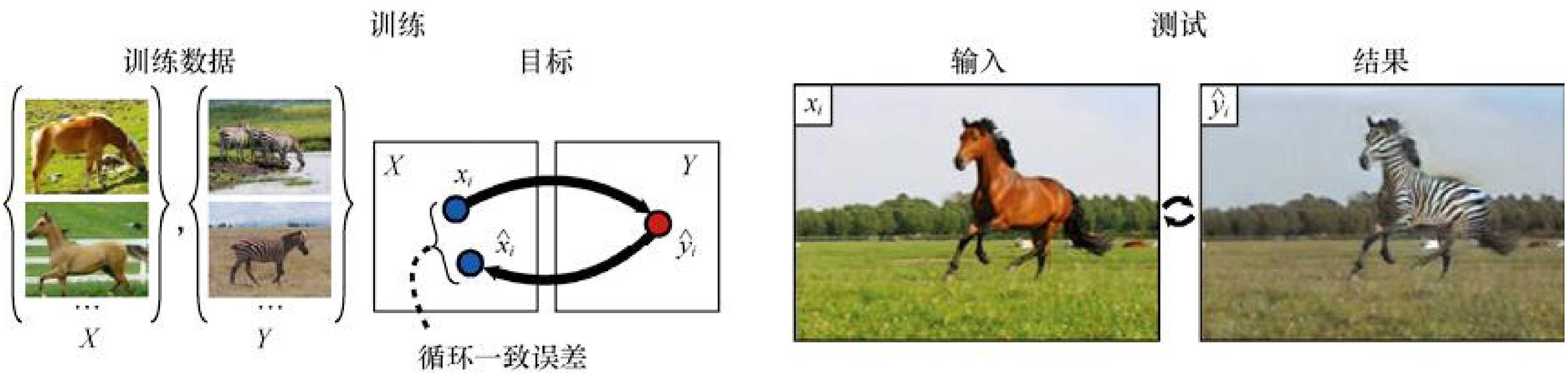
生成图片



成对图像的转换，其中输入由航空影像和相应的道路图组成，我们的目标是训练一个从航空影像生成道路图的网络（该系统还可以学习从道路图生成航空影像。）网络通过比较 （X型样本xi的输出）和Y型的正确输出yi进行训练。在测试时，网络必须从新的X型输入中生成新的Y型图像。

计算机视觉的应用

生成图片



未配对图像转换：给定两组图像（X型是马，Y型是斑马），但没有对应的配对，我们要学习将马转换成斑马。该方法训练两个预测器：一个将X型映射为Y型，另一个将Y型映射为X型。如果第一个网络将马xi映射为斑马，那么第二个网络应当把 映射回原始的xi。两个网络利用xi和 i之间的差进行训练。从Y型到X型再回到Y型的循环必须是封闭的。这样的网络可以成功地对图像进行丰富的变换。

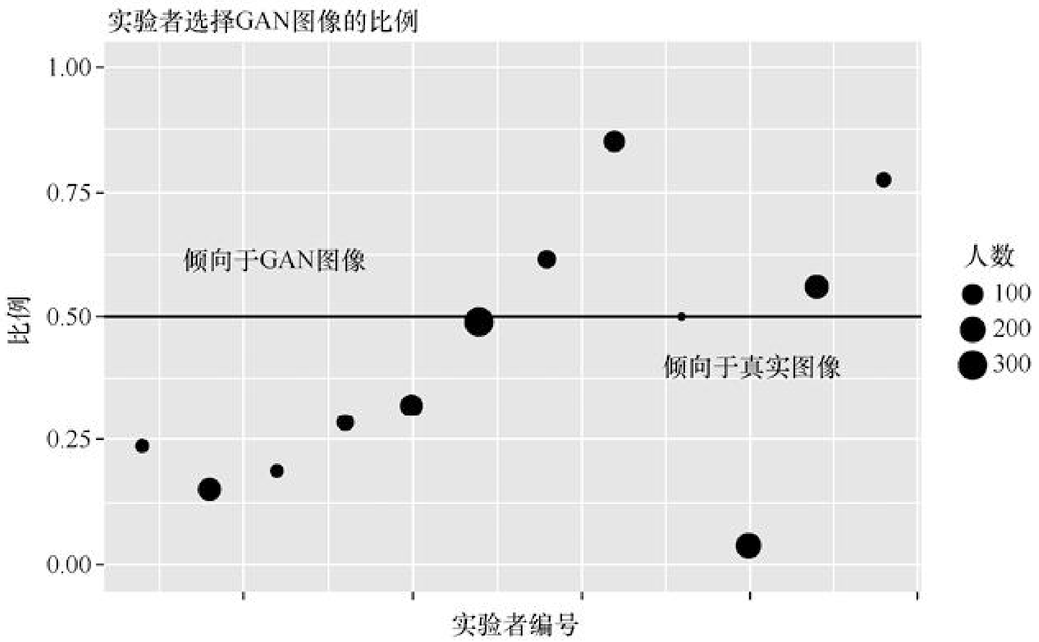
计算机视觉的应用

生成图片

风格转换：将内容为猫的照片与抽象绘画的风格相结合，生成经过抽象风格渲染的猫的新图像（右图）。

计算机视觉的应用

生成图片



GAN生成了肺部X射线图像。左图的一对图像为一张真实的X射线图和一张由GAN产生的X射线图。右图为一项测试的结果，它要求放射科医生在看到左边所示的一对X射线图后，判断哪一张X射线图是真实的。平均来说，他们的选择正确率为61%，这比任意猜测的结果要好一些。

小结

- 图像的表示蕴含着边缘、纹理、光流和区域等信息。这些信息为我们提供了有关物体边界以及图像之间对应关系的线索。
- 利用卷积神经网络可以得到精确的图像分类器，它从大量数据中去学习对分类任务有用的特征。
- 图像分类器可以转换成物体检测器。一个分类器对图像中框内的内容进行评分， 另一个分类器则判断该框中是否包含物体，以及该物体是什么。
- 当我们有同一个场景的多个视图时，我们可以重建场景的三维结构以及视图之间 的关系。在许多情况下，我们也可以从单个视图中重建三维几何结构。
- 计算机视觉方法的应用非常广泛。

重要知识点回顾

知识点回顾

- **第三章：通过搜索进行问题求解**
  - 无信息搜索策略
  - 有信息（启发式）搜索策略

无信息搜索策略

无信息搜索算法不提供有关某个状态与目标状态的接近程度的任何线索：

- **广度优先搜索 Breadth-first search**
- **一致代价搜索 Uniform-cost search**
- **深度优先搜索 Depth-first search**

无信息搜索策略

广度优先搜索

- 优先扩展最浅的未被扩展的节点
- 边界（frontier）可以实现为一个FIFO队列，即, 新节点（总是比其父节点更深）进入队列的队尾，而旧节点，即比新节点浅的节点，首先被扩展。

无信息搜索策略

一致代价搜索（Dijkstra算法）

- 优先扩展代价最小的未被扩展的节点
- 边界（frontier）可以实现为一个按路径代价排序的队列，最浅层的优先

无信息搜索策略

深度优先搜索

- 优先扩展最深的未被扩展的节点
- 边界（frontier）可以实现为一个LIFO队列，即后进先出。

有信息（启发式）搜索策略

- **有信息搜索（informed search）策略**使用关于目标位置的特定领域线索来比无信息搜索策略更有效地找到解。

- 线索以**启发式函数（heuristic function）**的形式出现，记为h(n)：

  h(n) = 从节点n的状态到目标状态的最小代价路径的代价估计值

例如，在寻径问题中，我们可以通过计算地图上两点之间的直线距离来估计从当前状态到目标的距离。

有信息（启发式）搜索策略

贪心最佳优先搜索（greedy best-first search）

- 首先扩展h(n)值最小的节点，即看起来最接近目标的节点，因为这样可能可以更快找到解。
- 评价函数f(n) = h(n)

有信息（启发式）搜索策略

A*搜索

- 主要思想：避免扩展代价已经很高的路径
- 评价函数f(n) = g(n) + h(n)

g(n) = 从初始节点到节点n的路径代价
h(n) = 从节点n的状态到目标状态的最小代价路径的代价估计值
f (n) = 经过n到一个目标状态的最优路径的代价估计值

- 对于**可容许的启发式（admissible heuristic）**函数，A*搜索是代价最优的，即

$$h(n) \le h^*(n)$$

这里 h*(n) 经过节点n到目标状态的真实代价.（h(n) ≥ 0, 对于任意目标G，h(G) = 0.)

知识点回顾

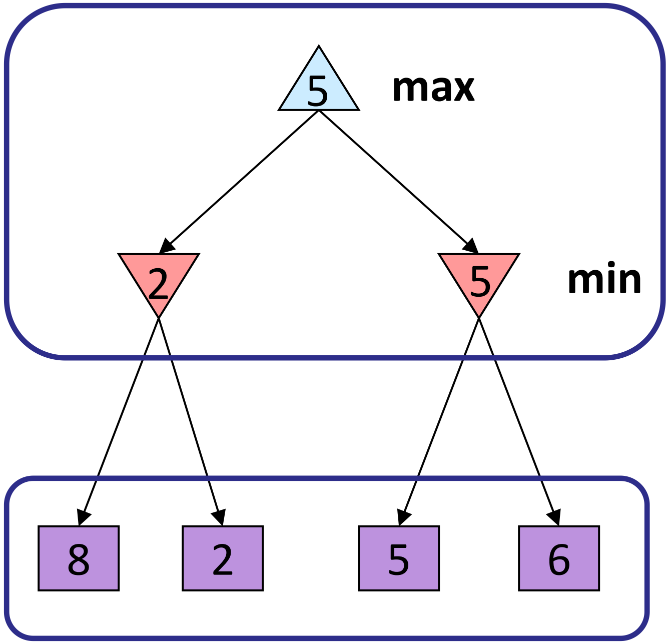
- **第五章：对抗搜索和博弈**
  - 极小化极大搜索
  - 评价函数
  - α-β剪枝
  - 期望最大搜索

极小化极大搜索

- 确定性的，零和游戏：
  - 井字棋, 象棋，围棋
  - 一个玩家最大化结果
  - 另一个玩家最小化结果

- 极小化极大搜索：
  - 状态空间搜索树
  - 玩家交替进行操作
  - 计算每个节点的**极小化极大值**:
    针对对手所能获得的最佳效用值

极小化极大值:
递归计算



终止状态值:
效用值

评价函数

- 在深度受限搜索中评价函数给出非终止状态的期望效用的估计值

$$\text{UTILITY}(loss, p) \leqslant \text{EVAL}(s, p) \leqslant \text{UTILITY}(win, p)$$

- 理想的函数：返回位置的实际的极小化极大值
- 在实际中：特征的线性加权，例如：

$f_1(s)$ = (白棋皇后数 – 黑棋皇后数)

$$\text{EVAL}(s) = w_1 f_1(s) + w_2 f_2(s) + \cdots + w_n f_n(s) = \sum_{i=1}^{n} w_i f_i(s)$$

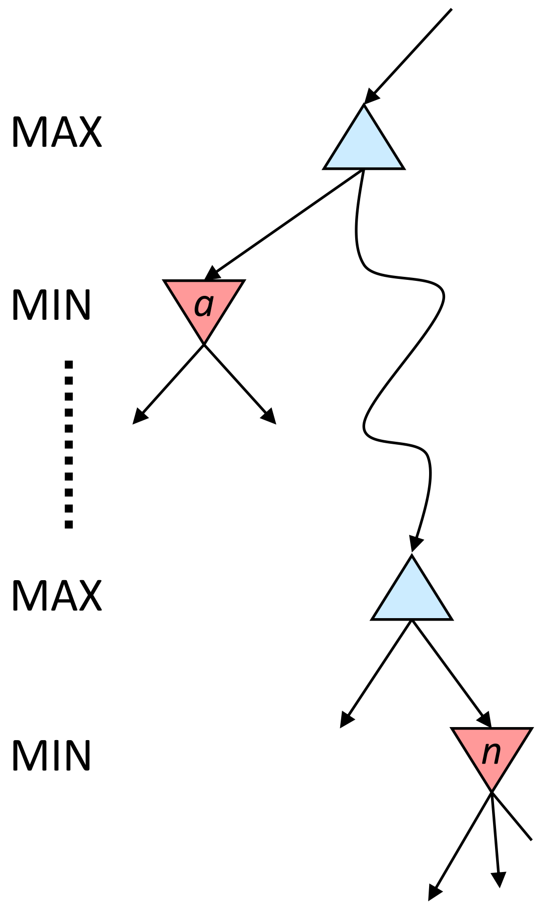
- **一个好的评价函数：**
  首先，计算时间不能太长！ （加快搜索速度）
  其次，评价函数应与实际的获胜机会密切相关。

博弈中的优化决策

α-β剪枝（alpha-beta pruning）

α-β剪枝的一般情况：

- 计算某一节点n的MIN值；
- 开始遍历n的后继节点；
- n节点的值是随着后继节点的遍历而下降的；
- 令 α 为MAX当前能够获得的最大值；
- 如果n的值变得比a小，它的值便不会被MAX采用，所以我们停止考虑n的其他后继节点。

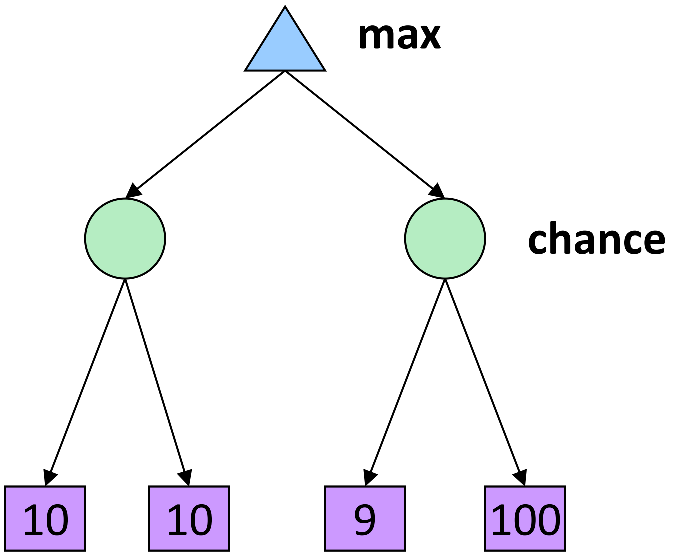


期望最大(expectimax)搜索

- 随机性产生的原因
  - 显式的随机性：游戏中的掷色子。
  - 不可预测的对手：吃豆人中幽灵的随机响应。
  - 动作可能存在一定的失败概率：当机器人移动时，轮胎可能会打滑。

- 这时的状态值应该反应平均情况，即期望最大(expectimax), 而不是最坏情况，即极小化极大(minimax)。

- **期望最大(expectimax)搜索:** 计算最佳博弈策略下的平均分
  - Max 节点与极小化极大搜索中的一样。
  - 机会节点类似Min节点，但是结果状态不确定。
  - 计算它们的期望效用值，即，对后继状态值使用对应的概率进行加权平均。



知识点回顾

- **第七章：逻辑智能体**
  - 逻辑推断
  - 命题逻辑：语法
  - 命题逻辑：语义
  - 命题定理证明

逻辑推断

前述的例子不仅阐明了什么是蕴含，还展示了如何用蕴含的定义来推导出结论，即进行**逻辑推断**。如果将 KB 的所有推论的集合比作干草堆而将 α 比做一根针，那么**蕴含正如草堆中的针一样，而推断就像找到这根针的过程。**如果一个推断算法 i 可以从 KB 中推导出 α，则记为

$$KB \vdash_i \alpha$$

可靠性：推断算法 i 是可靠的，如果

对于任何 $KB \vdash_i \alpha$, 那么 $KB \models \alpha$ 也为真

完备性：推断算法 i 是完备的，如果

对于任何 $KB \models \alpha$, 那么 $KB \vdash_i \alpha$ 也为真

命题逻辑：语法

命题逻辑的**语法**定义合法的语句。**原子语句（atomic sentence）**由单个**命题符号（proposition symbol）**构成。使用括号和被称作**逻辑联结词（logicalconnective）**的运算符可以将简单语句构造成**复合语句（complex sentence）**。常用的联结词有5个：

- $\neg$（非）。类似$\neg W_{1,3}$这样的语句称为$W1,3$的**否定**。一个**文字**要么是原子语句，即**正文字**，要么是原子语句的否定，即**负文字**。

- $\wedge$（与）。主要联结词是$\wedge$的语句称为**合取式**，例如$W_{1,3} \wedge P_{3,1}$，其各部分称为**合取子句**。（$\wedge$看起来像是“And”中的“A”。）

- $\vee$（或）。主要联结词是$\vee$的语句称为**析取式**，例如$(W_{1,3} \wedge P_{3,1}) \vee W_{2,2}$，其各部分为**析取子句**，本例中分别为$(W_{1,3} \wedge P_{3,1})$和$W2,2$。

- $\Rightarrow$（蕴涵）。如$(W_{1,3} \wedge P_{3,1}) \Rightarrow W_{2,2}$这样的语句称为**蕴涵式**（implication）或条件式，其**前提**（premise）或**前件（antecedent）**是$(W_{1,3} \wedge P_{3,1})$，其**结论**（conclusion）或**后件**（consequent）是$\neg W_{2,2}$。蕴涵式也被称为**规则**（rule）或**if-then**声明。有时，蕴涵符号在一些书籍中写作$\supset$或$\rightarrow$。

- $\Leftrightarrow$（当且仅当）。语句$W_{1,3} \Leftrightarrow \neg W_{2,2}$是**双向蕴涵式**（biconditional）。

命题逻辑：语义

语义定义了用于判定特定模型中语句真值的规则。命题逻辑中，模型就是对每个命题符号设定真值，即真（true）或假（false）。命题逻辑的语义必须指定在给定模型下如何计算任一语句的真值。**这是以递归的方式实现的。**所有语句都是由原子语句和5个联结词构建的。

- $\neg P$为真，当且仅当在m中P为假。
- $P \wedge Q$为真，当且仅当在m中P和Q都为真。
- $P \vee Q$为真，当且仅当在m中P或Q中至少一个为真。
- $P \Rightarrow Q$为真，除非在m中P为真而Q为假。
- $P \Leftrightarrow Q$为真，当且仅当在m中P和Q都为真或都为假。

命题定理证明：逻辑等价

逻辑等价（logical equivalence）：如果两个语句在相同的模型集合中都为真，则这两个语句逻辑等价，可以写作： $\alpha \equiv \beta$
等价的另一种定义为任意两条语句是等价的，当且仅当它们互相蕴含：

$\alpha \equiv \beta$ 当且仅当 $\alpha \models \beta$ 且 $\beta \models \alpha$

标准的逻辑等价：

$(\alpha \wedge \beta) \equiv (\beta \wedge \alpha)$ $\wedge$的交换律

$(\alpha \vee \beta) \equiv (\beta \vee \alpha)$ $\vee$的交换律

$((\alpha \wedge \beta) \wedge \gamma) \equiv (\alpha \wedge (\beta \wedge \gamma))$ $\wedge$的结合律

$((\alpha \vee \beta) \vee \gamma) \equiv (\alpha \vee (\beta \vee \gamma))$ $\vee$的结合律

$\neg(\neg\alpha) \equiv \alpha$ 双重否定律

$(\alpha \Rightarrow \beta) \equiv (\neg\beta \Rightarrow \neg\alpha)$ 假言易位

$(\alpha \Rightarrow \beta) \equiv (\neg\alpha \vee \beta)$ 蕴涵消去

$(\alpha \Leftrightarrow \beta) \equiv ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha))$ 等价消去

$\neg(\alpha \wedge \beta) \equiv (\neg\alpha \vee \neg\beta)$ 德摩根律

$\neg(\alpha \vee \beta) \equiv (\neg\alpha \wedge \neg\beta)$ 德摩根律

$(\alpha \wedge (\beta \vee \gamma)) \equiv ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma))$ $\wedge$对$\vee$的分配律

$(\alpha \vee (\beta \wedge \gamma)) \equiv ((\alpha \vee \beta) \wedge (\alpha \vee \gamma))$ $\vee$对$\wedge$的分配律

命题定理证明：有效性和可满足性

- **有效性（validity）**：如果一条语句在所有模型中都为真，则这条语句是有效的。

  例如 $True, A \vee \neg A, A \Rightarrow A, (A \wedge (A \Rightarrow B)) \Rightarrow B$

  蕴含的定义可以推导出演绎定理:

  $KB \models a$ 当且仅当 $(KB \Rightarrow a)$ 是有效的

命题定理证明：有效性和可满足性

- **有效性（validity）**：如果一条语句在所有模型中都为真，则这条语句是有效的。

  例如 $True, A \lor \neg A, A \Rightarrow A, (A \land (A \Rightarrow B)) \Rightarrow B$

  蕴含的定义可以推导出演绎定理:

  $KB \models \alpha$ 当且仅当（$KB \Rightarrow \alpha$）是有效的

- **可满足性（satisfiability）**：如果一条语句在某些模型中为真或能够被满足，则这条语句是可满足的。

  如果一条语句没有模型使其为真，则这条语句是不可满足的。
  例如, $A \land \neg A$

  可满足可以推导出归谬法:

  $KB \models \alpha$ 当且仅当（$KB \land \neg \alpha$）是不可满足的

- **单调性**：它表明蕴含的语句集只能随着信息被加入知识库而增长。

  如果 $KB \models \alpha$，则 $KB \land \beta \models \alpha$

命题定理证明：推断与证明

- **肯定前件**（Modus Ponens，mode that affirms的拉丁语），写作

$$\frac{\alpha \Rightarrow \beta, \alpha}{\beta}$$

- **合取消去**（and-elimination），即可以从一个合取式推导出任一合取子句：

$$\frac{\alpha \wedge \beta}{\alpha}$$

- 所有逻辑等价都可以用作推断规则。例如，等价消去可以产生两条推断规则：

$$\frac{\alpha \Leftrightarrow \beta}{(\alpha \Rightarrow \beta)(\beta \Rightarrow \alpha)} \text{ 和 } \frac{(\alpha \Rightarrow \beta)(\beta \Rightarrow \alpha)}{\alpha \Leftrightarrow \beta}$$

命题定理证明：通过归结证明

单元归结（unit resolution）规则

$$\frac{\ell_1 \vee \cdots \vee \ell_k, \qquad m}{\ell_1 \vee \cdots \vee \ell_{i-1} \vee \ell_{i+1} \vee \cdots \vee \ell_k}$$

单元归结规则可以推广为**全归结规则**

$$\frac{\ell_1 \vee \cdots \vee \ell_k, \qquad m_1 \vee \cdots \vee m_n}{\ell_1 \vee \cdots \vee \ell_{i-1} \vee \ell_{i+1} \vee \cdots \vee \ell_k \vee m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \vee \cdots \vee m_n}$$

归结使用两个子句并产生一个新的子句，该新子句包含除一对互补文字以外的原始子句的所有文字，例如

$$\frac{P_{1,1} \vee P_{3,1}, \qquad \neg P_{1,1} \vee \neg P_{2,2}}{P_{3,1} \vee \neg P_{2,2}}$$

一次只能归结一对互补文字，例如

$$\frac{P \vee \neg Q \vee R, \qquad \neg P \vee Q}{\neg Q \vee Q \vee R}$$

命题定理证明：通过归结证明

形式为子句合取式的语句被称为**合取范式（conjunctive normal form）或CNF**。把语句转换为CNF的过程：

$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$

1. 消去 $\Leftrightarrow$, 替换 $\alpha \Leftrightarrow \beta$ 为 $(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$.

   $(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1})$

2. 消去 $\Rightarrow$,替换 $\alpha \Rightarrow \beta$ 为 $\neg\alpha \vee \beta$.

   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg(P_{1,2} \vee P_{2,1}) \vee B_{1,1})$

3. 将 $\neg$ 内移，通过使用德摩根律和双重否定律:

   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge ((\neg P_{1,2} \wedge \neg P_{2,1}) \vee B_{1,1})$

4. 应用分配律:

   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg P_{1,2} \vee B_{1,1}) \wedge (\neg P_{2,1} \vee B_{1,1})$

提纲

- **第十二章：不确定性的量化**
  - 贝叶斯法则
  - 朴素贝叶斯模型

贝叶斯法则及其应用

- 乘积法则推可以写成两种形式：

$$P(a \wedge b) = P(a \mid b)P(b) \quad 和 \quad P(a \wedge b) = P(b \mid a)P(a)\ .$$

- 联立两式右侧，除以 $P(a)$, 我们可以得到**贝叶斯法则**：

$$P(b|a) = \frac{P(a|b)P(b)}{P(a)}\ .$$

- 通常，我们把一些未知**原因**（*cause*）的**结果**（*effect*）视为证据，并想要确定这个原因。在这种情况下，贝叶斯法则变为了：

$$P(cause|effect) = \frac{P(effect\ |cause)P(cause)}{P(effect)}$$

- 条件概率 $P(effect|cause)$ 量化**因果**方向上的关系，而 $P(cause|effect)$ 描述**诊断**方向上的关系。

朴素贝叶斯模型

- 单个原因直接影响许多结果，给定原因时，所有这些结果都是条件独立的。此时，完全联合分布可以写作：

$$P(Cause, Effect_1, \ldots, Effect_n) = P(Cause) \prod_i P(Effect_i | Cause)$$

- 这样的概率分布叫作**朴素贝叶斯**（naive Bayes）模型。“朴素”是因为它经常（作为一种简化假设）用于在给定原因变量时，“结果”变量不是严格独立的情况。

- 考虑观测到的结果 $E=e$,剩余结果变量Y是未观测的：

$$\mathbf{P}(Cause \mid \mathbf{e}) = \alpha \sum_{\mathbf{y}} \mathbf{P}(Cause, \mathbf{e}, \mathbf{y})$$

- 根据朴素贝叶斯，我们有

$$\begin{aligned}\mathbf{P}(Cause \mid \mathbf{e}) &= \alpha \sum_{\mathbf{y}} \mathbf{P}(Cause)\mathbf{P}(\mathbf{y} \mid Cause)\left(\prod_j \mathbf{P}(e_j \mid Cause)\right) \\ &= \alpha \mathbf{P}(Cause)\left(\prod_j \mathbf{P}(e_j \mid Cause)\right)\sum_{\mathbf{y}} \mathbf{P}(\mathbf{y} \mid Cause) \\ &= \alpha \mathbf{P}(Cause)\prod_j \mathbf{P}(e_j \mid Cause)\end{aligned}$$

提纲

- **第十三章：概率推理**
  - 贝叶斯网络中的精确推断：枚举法
  - 贝叶斯网络中的精确推断：变量消元法
  - 贝叶斯网络中的近似推理：直接采样
  - 贝叶斯网络中的近似推理：拒绝采样
  - 贝叶斯网络中的近似推理：似然加权

贝叶斯网络中的精确推断

NANJING UNIVERSITY

通过枚举进行推断

- 一般形式:
  - 证据变量: $E_1 \dots E_k = e_1 \dots e_k$
  - 查询变量: Q
  - 隐藏变量: $H_1 \dots H_r$

  $\left.\begin{array}{l} \\ \\ \\ \end{array}\right\} \begin{array}{l} X_1, X_2, \dots X_n \\ \text{所有变量} \end{array}$

- 目标概率: $P(Q|e_1 \dots e_k)$

- **步骤1:** 在联合概率分布表中选择和证据变量一致的条目。

- **步骤2:** 对隐藏变量H求和消元来获得查询和证据变量的联合概率。

$$P(Q, e_1 \dots e_k) = \sum_{h_1 \dots h_r} \underbrace{P(Q, h_1 \dots h_r, e_1 \dots e_k)}_{X_1, X_2, \dots X_n}$$

- **步骤3:** 归一化。

$$P(Q|e_1 \cdots e_k) = \frac{1}{Z} P(Q, e_1 \cdots e_k) \qquad Z = \sum_q P(Q, e_1 \cdots e_k)$$

贝叶斯网络中的精确推断

变量消元法

- 查询语句: $P(Q|E_1 = e_1, \ldots E_k = e_k)$

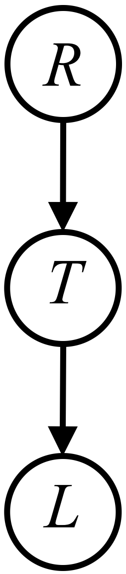
- 从初始因子开始:
  - 局部的 CPTs (会由证据实例化)

- 如果还有隐藏变量未被处理，则循环:
  - 选择一个隐藏变量 H
  - 连接所有含有隐藏变量 H 的因子
  - 求和消元 H

- 连接所有剩余的因子并归一化

贝叶斯网络中的精确推断

变量消元法



$$P(L) = ?$$

- 枚举法

$$= \underbrace{\sum_t \underbrace{\sum_r \underbrace{P(L|t) \underbrace{P(r)P(t|r)}_{\text{连接 r}}}_{\text{连接 t}}}_{\text{消元 r}}}_{\text{消元 t}}$$

- 变量消元法

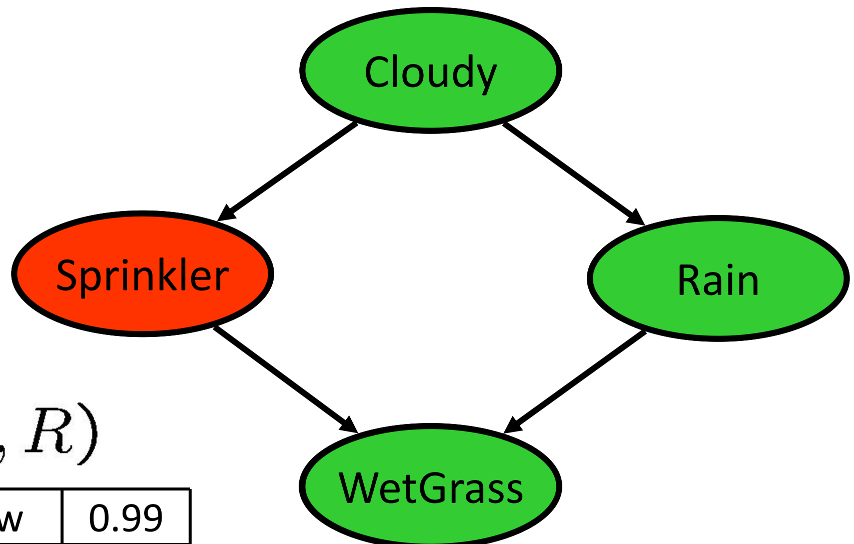
$$= \underbrace{\sum_t \underbrace{P(L|t) \underbrace{\sum_r \underbrace{P(r)P(t|r)}_{\text{连接 r}}}_{\text{消元 r}}}_{\text{连接 t}}}_{\text{消元 t}}$$

贝叶斯网络中的近似推断

直接采样

$P(C)$

+c	0.5
-c	0.5



$P(S|C)$

+c	+s	0.1
	-s	0.9
-c	+s	0.5
	-s	0.5

$P(R|C)$

+c	+r	0.8
	-r	0.2
-c	+r	0.2
	-r	0.8

$P(W|S,R)$

+s	+r	+w	0.99
		-w	0.01
	-r	+w	0.90
		-w	0.10
-s	+r	+w	0.90
		-w	0.10
	-r	+w	0.01
		-w	0.99

样本:

+c, -s, +r, +w

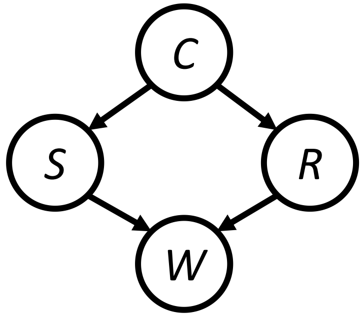
-c, +s, -r, +w

…

贝叶斯网络中的近似推断

拒绝采样

- 假设我们想要计算 P(C | +s)
  - 使用直接采样的方法生成样本
  - 忽略 (拒绝) 所有与证据变量不一致的样本，在这例子中 S = +s
  - 计算剩余样本中C = +c 或者-c 的个数
  - 与目标条件概率是一致的(即, 当样本足够多，会逼近真实概率值)



+c, -s, +r, +w

+c, +s, +r, +w

-c, +s, +r, -w

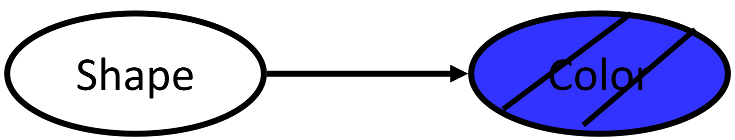
+c, -s, +r, +w

-c, -s, -r, +w

贝叶斯网络中的近似推断

似然加权

- 思路：固定证据变量的取值，对其他变量进行采样
- 问题：样本分布与真实分布不一致
- 解决方案：给定父节点，使用证据变量的概率进行加权



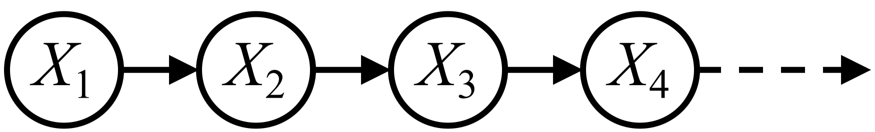
pyramid, blue
pyramid, blue
sphere, blue
cube, blue
sphere, blue

提纲

- **第十四章：时间上的概率推理**
  - 马尔可夫模型
  - 隐马尔可夫模型

马尔可夫模型

- $\mathbf{X}_t$: 表示在时刻 t 的状态变量
- 初始状态概率分布：随机变量**X**的先验概率分布
- 转移模型：给定前一状态的值时，最新状态变量的概率分布: $\mathbf{P}(\mathbf{X}_t \mid \mathbf{X}_{t-1})$
- 稳态假设: 转移概率在任何时刻都相同。



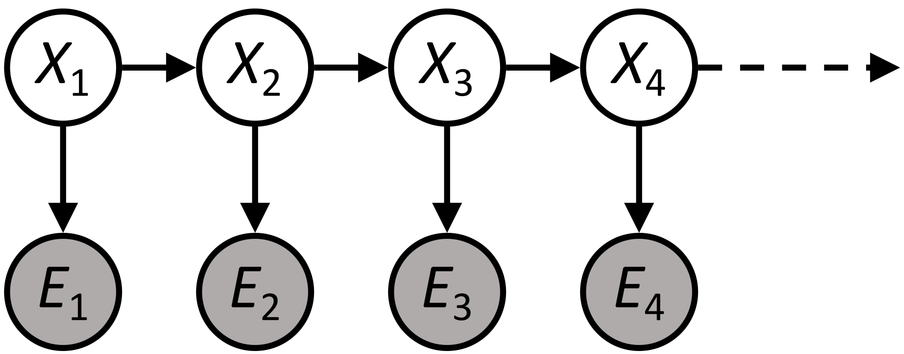
$P(X_t) = ?$

$P(X_1)$

$P(X_t|X_{t-1})$

隐马尔可夫模型

- 隐马尔可夫模型的定义:
  - 初始概率: $P(X_1)$
  - 转移概率: $P(X_t \mid X_{t-1})$
  - 输出概率: $P(E_t \mid X_t)$



提纲

- **第十九章：样例学习**
  - 决策树学习
  - 线性回归与分类
  - 非参数模型：最近邻模型
  - 集成学习：自适应提升法

决策树学习

- 决策树（*decision tree*）将属性值向量映射到单个输出值（即“决策”）。
  - 执行一系列测试来实现其决策，它从根节点出发，沿着适当的分支，直到到达叶节点为止。
  - 树中的每个内部节点对应于一个输入属性的测试
  - 该节点的分支用该属性的所有可能值进行标记
  - 叶节点指定了函数要返回的值

- 布尔型的决策树等价于如下形式的逻辑语句：

$$Output \Leftrightarrow (Path_1 \vee Path_2 \vee \cdots)$$

线性回归与分类

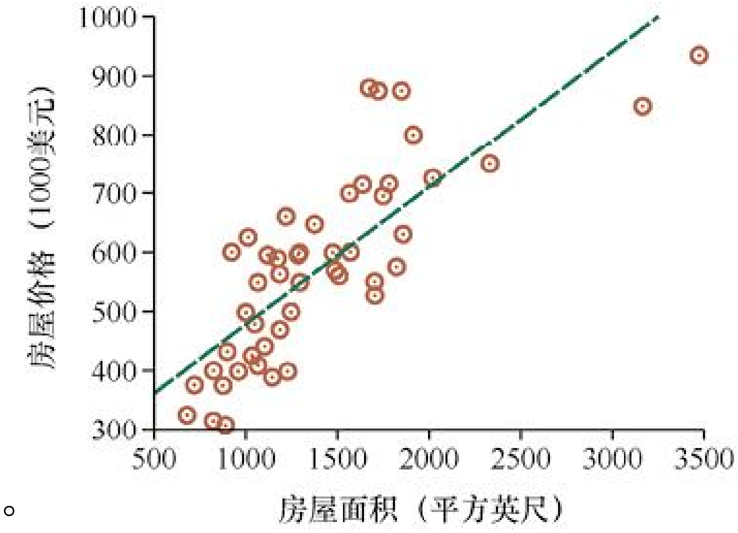
单变量线性回归

输入 x 和输出 y

$$y = w_1 x + w_0$$

线性函数

$$h_{\mathbf{w}}(x) = w_1 x + w_0$$



（a）

线性回归：找到最匹配这些数据的线性函数 $h_{\mathbf{w}}$

找到对应的权重值 (w_0, w_1) 使得其经验损失最小。

平方误差损失函数, L_2, 对所有训练样本求和：

$$Loss(h_{\mathbf{w}}) = \sum_{j=1}^{N} L_2(y_j, h_{\mathbf{w}}(x_j)) = \sum_{j=1}^{N} (y_j - h_{\mathbf{w}}(x_j))^2 = \sum_{j=1}^{N} (y_j - (w_1 x_j + w_0))^2 .$$

$$\frac{\partial}{\partial w_0} \sum_{j=1}^{N} (y_j - (w_1 x_j + w_0))^2 = 0 \text{ and } \frac{\partial}{\partial w_1} \sum_{j=1}^{N} (y_j - (w_1 x_j + w_0))^2 = 0 .$$

$$w_1 = \frac{N(\sum x_j y_j) - (\sum x_j)(\sum y_j)}{N(\sum x_j^2) - (\sum x_j)^2}; \quad w_0 = (\sum y_j - w_1(\sum x_j))/N .$$

非参数模型

最近邻模型

k-近邻

- 给定待查询的 $\mathbf{x}_q$, 寻找最接近$\mathbf{x}_q$的k个样例。
- 为了实现分类,我们寻找$\mathbf{x}_q$的一组邻居，并以占比最大的输出值为分类结果。例如 $k = 3$ 并且输出值为*Yes*, *No*, *Yes* , 则分类结果为 *Yes*.
- 为了实现回归，我们可以取k个邻居的平均值或中位数，也可以在最近邻邻居上求解一个线性回归问题。
- 闵可夫斯基距离（Minkowski distance）或 L^p范数

$$L^p(\mathbf{x}_j, \mathbf{x}_q) = \left(\sum_i |x_{j,i} - x_{q,i}|^p\right)^{1/p}.$$

- p =2, 欧几里得距离
- p =1, 曼哈顿距离
- 对于布尔属性值，汉明距离
- **马氏距离:** 考虑维度之间的协方差

$$D_M(x, y) = \sqrt{(x-y)^T \Sigma^{-1} (x-y)}$$

自适应提升法

- **加权训练集：**给每个样例赋予一个权重$w_j \geq 0$，该权重描述了样例在训练过程中应计数的次数。

- 从所有样例具有相等的权重开始，根据该训练集，训练第一个假设h_1.

- 我们希望下一个假设能在被分类错误的样例中表现得更好，因此我们将增加它们的权重，同时减小正确分类的样例的权重。

- 基于这个重新进行加权得到的训练集，我们训练得到假设h_2。这一过程将以这种方式不断进行，直到生成K个假设。

- 类似于贪心算法，即不会回退，一旦算法选择了某个假设h_i ，它就永远不会抛弃该选择，而是会添加新的假设：

$$h(\mathbf{x}) = \sum_{i=1}^{K} z_i h_i(\mathbf{x})$$

提纲

- **第二十章：概率模型学习**
  - 贝叶斯学习
  - 最大后验学习
  - 最大似然学习
  - 最大似然参数学习：离散模型
  - 贝叶斯参数学习

贝叶斯学习

- 将学习看作假设空间中概率分布的贝叶斯更新：

  H 是假设变量, 值为 $h_1, h_2, \ldots$, 先验分布 $\mathrm{P}(H)$

  第j个观测 d_j给出了随机变量的输出 D_j
  训练数据 $\mathrm{d} = d_1, \ldots, d_N$

- 给定到目前为止的数据, 每一个假设有一个后验分布:

  $P(h_i|\mathrm{d}) = \alpha P(\mathrm{d}|h_i)P(h_i)$

  这里 $P(\mathrm{d}|h_i)$ 被称为**似然**

- 预测为在假设上的概率加权平均:

  $\mathrm{P}(X|\mathrm{d}) = \Sigma_i\ \mathrm{P}(X|\mathrm{d}, h_i)P(h_i|\mathrm{d}) = \Sigma_i\ \mathrm{P}(X|h_i)P(h_i|\mathrm{d})$

最大后验学习

- 在假设空间上求和通常是非常困难的 $\Sigma_i\ \mathrm{P}(X|h_i)P(h_i|\mathrm{d})$
- **最大后验** (MAP)学习：选择 h_{MAP} 来最大化 $P(h_i|\mathrm{d})$

即, 最大化 $P(\mathrm{d}|h_i)P(h_i)$ 或 $\log P(\mathrm{d}|h_i) + \log P(h_i)$

$$\boldsymbol{P}(X \mid \boldsymbol{d}) \approx \boldsymbol{P}(X \mid h_{\mathrm{MAP}})$$

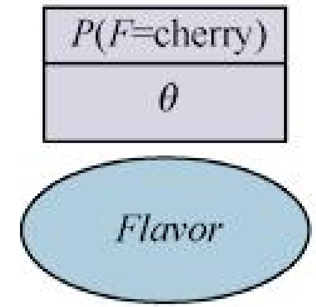
- $-\log_2 P(\boldsymbol{d} \mid h_i) - \log_2 P(h_i)$负对数项可以被看作
  - 给定假设编码数据所需的比特数 + 编码假设所需的比特数
  - 这是最小描述长度(MDL)学习的基本思想

最大似然学习

- 当数据集很大时，假设的先验分布就不那么重要了，因为来自数据的证据足够强大，足以淹没假设的先验分布。
- 最大似然 (ML) 学习：选择 h_{ML} 来最大化 $P(\mathrm{d}|h_i)$
- 即, 简单的获取对数据的最佳拟合；对于假设空间具有均匀先验分布，等同于最大后验学习 (例如所有的假设都同样复杂)
- 最大似然学习是“标准”的（非贝叶斯）统计学习方法

完全数据学习

最大似然参数学习：离散模型



- 可能含有樱桃味和酸橙味糖果的糖果袋，其中糖果口味的比例完全未知。
- 参数θ表示樱桃味糖果所占的比例，其对应的假设为h_θ
- 如果我们假设所有的比例有相同的先验可能性，那么采用最大似然估计是合理的
- 现在假设我们已经打开了N颗糖果，其中有c颗为樱桃味，则该特定数据集的似然为：

$$P(\mathbf{d} \mid h_\theta) = \prod_{j=1}^{N} P(d_j \mid h_\theta) = \theta^c \cdot (1-\theta)^\ell$$

- 最大似然假设所需的参数即为使得上式最大化的参数 。由于log函数是单调函数， 我们可以最大化对数似然来简化计算：

$$L(\mathbf{d} \mid h_\theta) = \log P(\mathbf{d} \mid h_\theta) = \sum_{j=1}^{N} \log P(d_j \mid h_\theta) = c \log\theta + \ell \log(1-\theta)$$

- 对上式关于θ进行求导，并令导数为0可得：

$$\frac{\mathrm{d}L(\mathbf{d} \mid h_\theta)}{\mathrm{d}\theta} = \frac{c}{\theta} - \frac{\ell}{1-\theta} = 0 \qquad \Rightarrow \qquad \theta = \frac{c}{c+\ell} = \frac{c}{N}$$

完全数据学习

NANJING UNIVERSITY

贝叶斯参数学习

- 基于贝叶斯方法的参数学习过程从一个关于假设的先验分布开始，随着新数据出现而不断更新该分布。

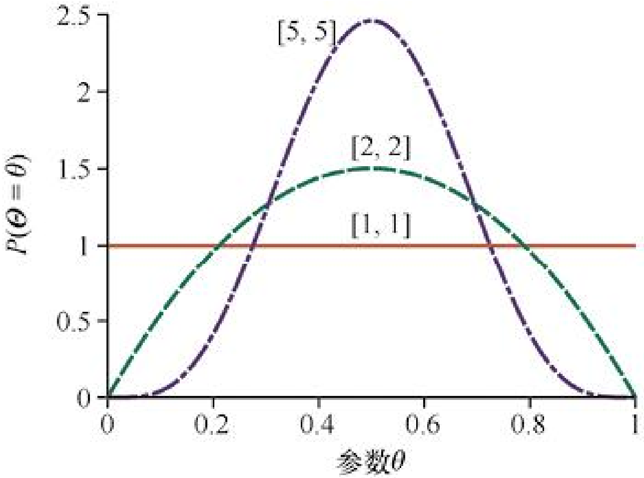
- 从贝叶斯角度看，随机变量 $\boldsymbol{\Theta}$定义了假设空间，θ 是 $\boldsymbol{\Theta}$ 的一个未知值。

- 假设先验是先验分布 **P(Θ)**。因此, P($\boldsymbol{\Theta}$ = θ) 是糖果袋中含有比例θ的樱桃味糖果的先验概率。

- $P(\theta) = Uniform[0,1](\theta)$, 均匀分布是 β分布的一个特例.

- β分布由两个超参数 a 和 b 定义：

$$\text{beta}[a, b](\theta) = \alpha\, \theta^{a-1}(1-\theta)^{b-1}\,,$$



完全数据学习

贝叶斯参数学习

- 假设我们观测到了一颗樱桃味的糖果，那么我们有

$$\begin{aligned} P(\theta \mid D_1 = cherry) &= \alpha\, P(D_1 = cherry \mid \theta) P(\theta) \\ &= \alpha'\, \theta \cdot \mathrm{beta}[a,b](\theta) = \alpha'\, \theta \cdot \theta^{a-1}(1-\theta)^{b-1} \\ &= \alpha'\, \theta^{a}(1-\theta)^{b-1} = \mathrm{beta}[a+1,b](\theta)\,. \end{aligned}$$