

# 机器学习

## Machine Learning

北京航空航天大学计算机学院智能识别与图像处理实验室  
IRIP Lab, School of Computer Science and Engineering, Beihang University

黄 迪 刘庆杰

2018年秋季学期  
Fall 2018

# 课前回顾

# 主成分分析-算法

## ● 最大方差思想

使用较少的数据维度保留住较多的原数据特性

将 **$D$** 维数据集  $\{\mathbf{x}_n\}, n = 1, 2, \dots, N$  降为 **$M$** 维,  $M < D$

首先考虑  $M = 1$ , 定义这个空间的投影方向为 **$D$** 维向量  $\mathbf{u}_1$

出于方便且不失一般性, 令  $\mathbf{u}_1^T \mathbf{u}_1 = 1$

每个数据点  $\mathbf{x}_n$  在新空间中表示为  $\mathbf{u}_1^T \mathbf{x}_n$

样本均值在新空间中表示为  $\mathbf{u}_1^T \bar{\mathbf{x}}$ , 其中  $\bar{\mathbf{x}} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n$

投影后样本方差表示为  $\frac{1}{N} \sum_{n=1}^N \{\mathbf{u}_1^T \mathbf{x}_n - \mathbf{u}_1^T \bar{\mathbf{x}}\}^2 = \boxed{\mathbf{u}_1^T \mathbf{S} \mathbf{u}_1}$  最大

其中原样本方差  $\mathbf{S} = \frac{1}{N} \sum_{n=1}^N (\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^T$

# 主成分分析-算法

## ● 最大方差思想

使用较少的数据维度保留住较多的原数据特性

目标是**最大化**  $\mathbf{u}_1^T \mathbf{S} \mathbf{u}_1$ , *s.t.*  $\mathbf{u}_1^T \mathbf{u}_1 = 1$

利用拉格朗日乘子法  $\mathbf{u}_1^T \mathbf{S} \mathbf{u}_1 + \lambda_1(1 - \mathbf{u}_1^T \mathbf{u}_1)$

对  $\mathbf{u}_1$  求导置零得到  $\mathbf{S} \mathbf{u}_1 = \lambda_1 \mathbf{u}_1$

**$\mathbf{u}_1$ 是S的特征向量**

进一步得到  $\mathbf{u}_1^T \mathbf{S} \mathbf{u}_1 = \lambda_1$

**$\mathbf{u}_1$ 是S最大特征值对应的特征向量时  
方差取到极大值，称 $\mathbf{u}_1$ 为第一主成分**

# 主成分分析-算法

## ● 最大方差思想

使用较少的数据维度保留住较多的原数据特性

考虑更一般性的情况( $M > 1$ ), 新空间中数据方差最大的最佳投影方向由协方差矩阵 $S$ 的 $M$ 个特征向量 $u_1, \dots, u_M$ 定义, 其分别对应 $M$ 个最大的特征值 $\lambda_1, \dots, \lambda_M$

首先获得方差最大的1维, 生成该维的补空间;

继续在补空间中获得方差最大的1维, 生成新的补空间;

依次循环下去得到 $M$ 维的空间。

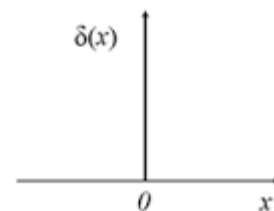
# 主成分分析-算法

## ● 最小均方误差思想

使原数据与降维后的数据(在原空间中的重建)的误差最小

定义一组正交的 $D$ 维基向量  $\{\mathbf{u}_i\}, i = 1, \dots, D$ , 满足

$$\mathbf{u}_i^T \mathbf{u}_j = \delta_{ij}$$



由于基是完全的, 每个数据点可以表示为基向量的线性组合

$$\mathbf{x}_n = \sum_{i=1}^D \alpha_{ni} \mathbf{u}_i$$

相当于进行了坐标变换

$$\{\mathbf{x}_{n1}, \dots, \mathbf{x}_{nD}\} \xrightarrow{\{\mathbf{u}_i\}} \{\alpha_{n1}, \dots, \alpha_{nD}\}$$



$$\alpha_{nj} = \mathbf{x}_n^T \mathbf{u}_j$$

# 主成分分析-算法

## ● 最小均方误差思想

使原数据与降维后的数据(在原空间中的重建)的误差最小

那么  $\mathbf{x}_n = \sum_{i=1}^D (\mathbf{x}_n^T \mathbf{u}_i) \mathbf{u}_i$

在 $M$ 维变量( $M < D$ )生成的空间中对其进行表示

$$\tilde{\mathbf{x}}_n = \sum_{i=1}^M z_{ni} \mathbf{u}_i + \sum_{i=M+1}^D b_i \mathbf{u}_i$$

独特的

共享的

目标最小化失真度  $J = \frac{1}{N} \sum_{n=1}^N \|\mathbf{x}_n - \tilde{\mathbf{x}}_n\|^2$

倒数置零得到  $z_{nj} = \mathbf{x}_n^T \mathbf{u}_j, j = 1, \dots, M$

$$b_j = \bar{\mathbf{x}}^T \mathbf{u}_j, j = M + 1, \dots, D$$

# 主成分分析-算法

## ● 最小均方误差思想

使原数据与降维后的数据(在原空间中的重建)的误差最小

$$\text{有 } \mathbf{x}_n - \tilde{\mathbf{x}}_n = \sum_{i=M+1}^D \{(\mathbf{x}_n - \bar{\mathbf{x}})^T \mathbf{u}_i\} \mathbf{u}_i$$

$$J = \frac{1}{N} \sum_{n=1}^N \sum_{i=M+1}^D (\mathbf{x}_n^T \mathbf{u}_i - \bar{\mathbf{x}}^T \mathbf{u}_i)^2 = \sum_{i=M+1}^D \mathbf{u}_i^T \mathbf{S} \mathbf{u}_i$$

$$\text{拉格朗日乘子法 } \tilde{J} = \sum_{i=M+1}^D \mathbf{u}_i^T \mathbf{S} \mathbf{u}_i + \sum_{i=M+1}^D \lambda_i (1 - \mathbf{u}_i^T \mathbf{u}_i)$$

求导得到  $\mathbf{S} \mathbf{u}_i = \lambda_i \mathbf{u}_i$

**$J$ 最小时取 $D-M$ 个最小的特征值**

对应失真度为  $J = \sum_{i=M+1}^D \lambda_i$

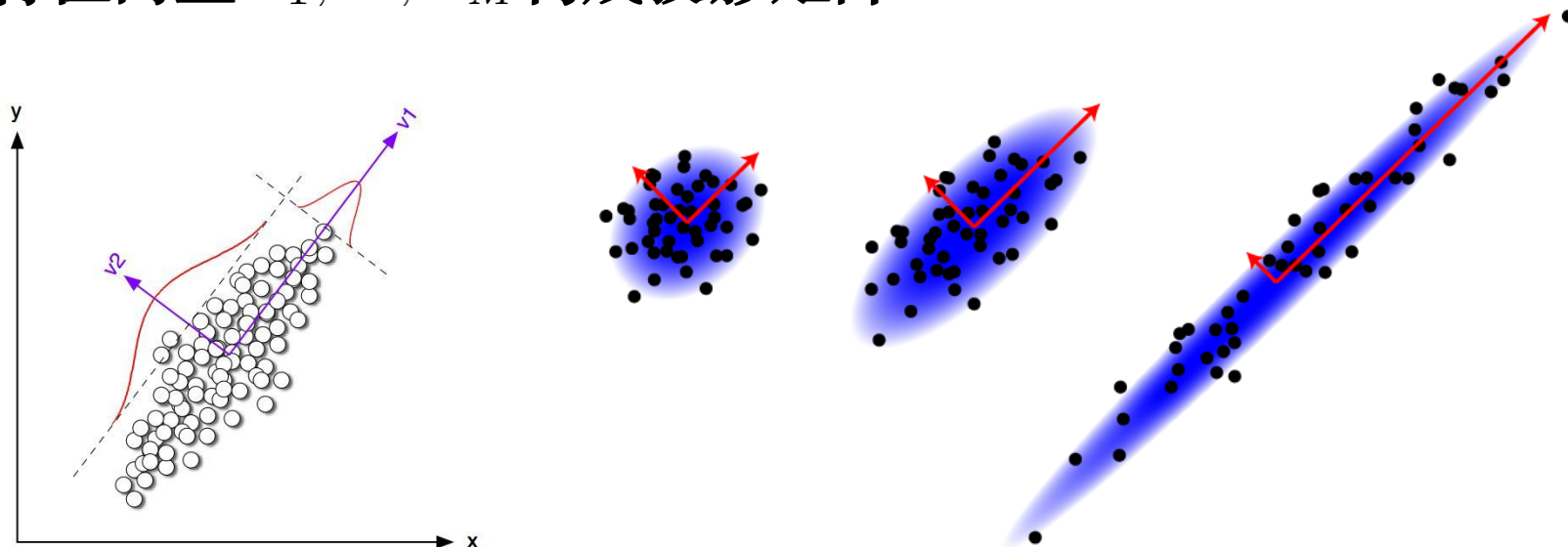
**主子空间对应 $M$ 个最大特征值**



# 主成分分析-算法

## ● 计算步骤

- ① 计算给定样本  $\{\mathbf{x}_n\}, n = 1, 2, \dots, N$  的均值  $\bar{\mathbf{x}}$  和协方差矩阵  $\mathbf{S}$ ;
- ② 计算  $\mathbf{S}$  的特征向量与特征值;
- ③ 将特征值从大到小排列, 前  $M$  个特征值  $\lambda_1, \dots, \lambda_M$  所对应的特征向量  $\mathbf{u}_1, \dots, \mathbf{u}_M$  构成投影矩阵。



# 主成分分析-应用

## ● 利用PCA处理高维数据

在实际应用中，样本维数可能很高，远大于样本的个数  
在人脸识别中，1000张人脸图像，每张图像 $100 \times 100$ 像素

$D$ 维空间， $N$ 个样本点， $N > D$

$\mathbf{X}$ 是 $N \times D$ 维的数据矩阵，其行向量为 $(\mathbf{x}_n - \bar{\mathbf{x}})^T$

$$\mathbf{S} = \frac{1}{N} \sum_{n=1}^N (\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^T \text{ 可以写为 } \mathbf{S} = N^{-1} \mathbf{X}^T \mathbf{X}$$

$S$ 维数?  $D \times D$ 维  $10000 \times 10000$

$$\frac{1}{N} \mathbf{X}^T \mathbf{X} \mathbf{u}_i = \lambda_i \mathbf{u}_i \longrightarrow \frac{1}{N} \mathbf{X} \mathbf{X}^T (\mathbf{X} \mathbf{u}_i) = \lambda_i (\mathbf{X} \mathbf{u}_i)$$

令  $\mathbf{v}_i = \mathbf{X} \mathbf{u}_i$ ，得到  $\boxed{\frac{1}{N} \mathbf{X} \mathbf{X}^T} \mathbf{v}_i = \lambda_i \mathbf{v}_i$   $N \times N$ 维

# 主成分分析-应用

- 利用PCA处理高维数据

在实际应用中，样本维数可能很高，远大于样本的个数  
在人脸识别中，1000张人脸图像，每张图像 $100 \times 100$ 像素

对  $\frac{1}{N} \mathbf{X} \mathbf{X}^T$  求的**特征值**  $\lambda_i$  和**特征向量**  $\mathbf{v}_i$

$$\frac{1}{N} \mathbf{X} \mathbf{X}^T \mathbf{v}_i = \lambda_i \mathbf{v}_i$$

$$\longrightarrow \frac{1}{N} \mathbf{X}^T \mathbf{X} (\mathbf{X}^T \mathbf{v}_i) = \lambda_i (\mathbf{X}^T \mathbf{v}_i) \quad \text{S的特征向量}$$

调整尺度  $\mathbf{u}_i \propto \mathbf{X}^T \mathbf{v}_i$  满足  $\|\mathbf{u}_i\| = 1$

$$\longrightarrow \mathbf{u}_i = \frac{1}{(N\lambda_i)^{1/2}} \mathbf{X}^T \mathbf{v}_i$$

**奇异值分解(Singular Value Decomposition, SVD)**

# 概率主成分分析

## ● PCA的概率表示

隐藏变量  $z$  以如下形式产生  $D$  维观测变量  $x$

$$x = Wz + \mu + \epsilon \rightarrow \text{高斯噪声}$$

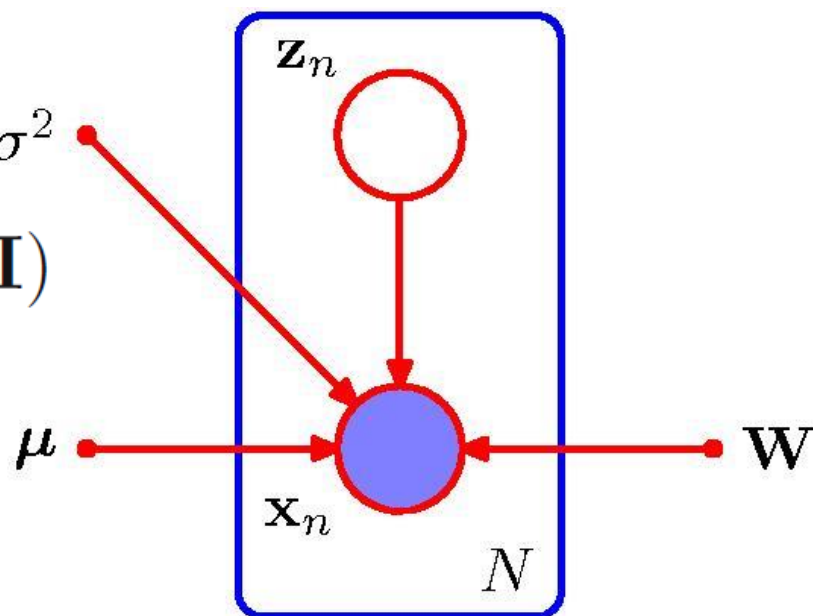
其中,  $z$  为  $M$  维的隐藏变量, 且满足高斯分布

$$p(z) = \mathcal{N}(z|0, I)$$

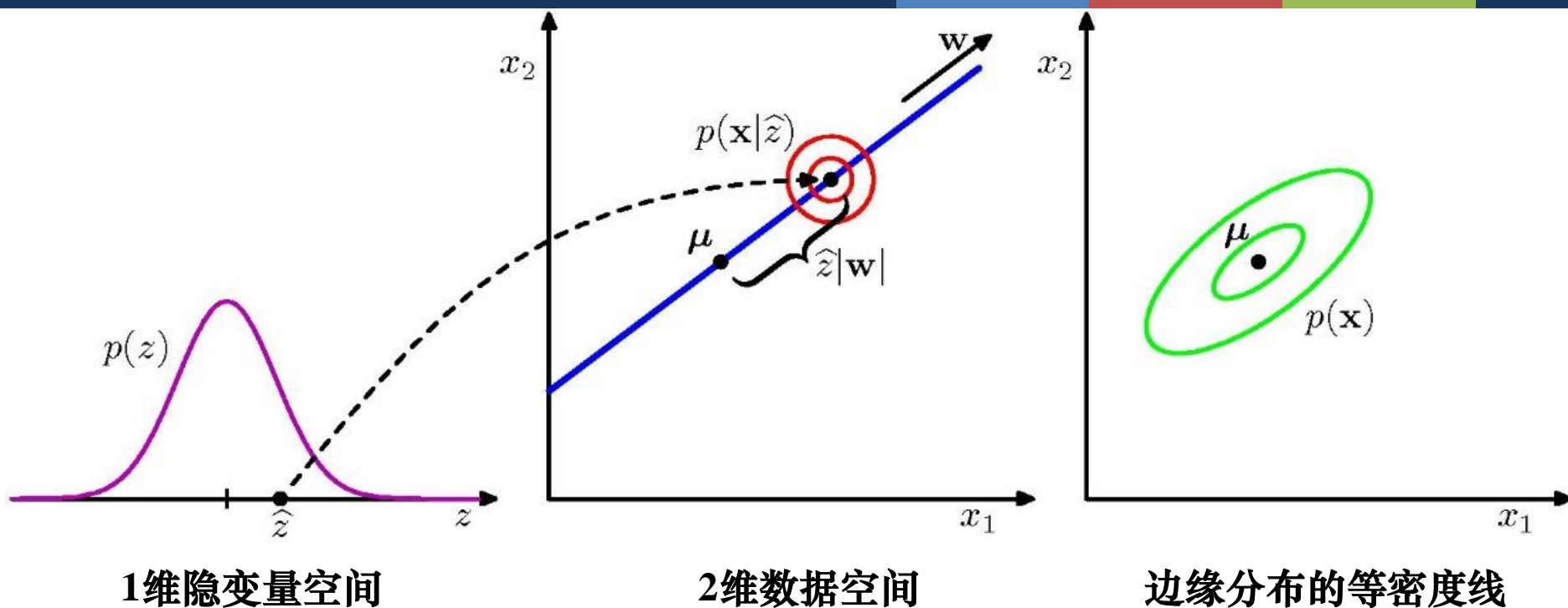
$x$  以  $z$  为条件的分布也满足高斯  $\sigma^2$

$$p(x|z) = \mathcal{N}(x|Wz + \mu, \sigma^2 I)$$

以有向图表示



# 概率主成分分析



从隐空间到数据空间的映射，与PCA的传统视角相反  
从数据空间到隐空间的映射，可以由贝叶斯定理得到

# 概率主成分分析

- PCA的概率表示

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I})$$

$$p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z} + \mu, \sigma^2\mathbf{I})$$

↓

$$p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z} = \mathcal{N}(\mathbf{x}|\mu, \mathbf{C})$$

$\mathbf{C} = \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I}$

$$\mathbb{E}[\mathbf{x}] = \mathbb{E}[\mathbf{W}\mathbf{z} + \mu + \epsilon] = \mu$$

$$\begin{aligned}\text{cov}[\mathbf{x}] &= \mathbb{E}[(\mathbf{W}\mathbf{z} + \epsilon)(\mathbf{W}\mathbf{z} + \epsilon)^T] \\ &= \mathbb{E}[\mathbf{W}\mathbf{z}\mathbf{z}^T\mathbf{W}^T] + \mathbb{E}[\epsilon\epsilon^T] = \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I}\end{aligned}$$

# 概率主成分分析

## ● PCA的概率表示


最大似然估计求解

给定  $\mathbf{X} = \{\mathbf{x}_n\}$ ，求其对数似然函数

$$\begin{aligned}\ln p(\mathbf{X}|\mu, \mathbf{W}, \sigma^2) &= \sum_{n=1}^N \ln p(\mathbf{x}_n|\mu, \mathbf{W}, \sigma^2) \\ &= -\frac{ND}{2} \ln(2\pi) - \frac{N}{2} \ln|\mathbf{C}| - \frac{1}{2} \sum_{n=1}^N (\mathbf{x}_n - \mu)^T \mathbf{C}^{-1} (\mathbf{x}_n - \mu)\end{aligned}$$

对  $\mu$  求导置零并代回 

$$= -\frac{N}{2} \{ D \ln(2\pi) + \ln|\mathbf{C}| + \text{Tr}(\mathbf{C}^{-1} \mathbf{S}) \}$$

 相关矩阵

  $\mathbf{W}_{\text{ML}} = \mathbf{U}_M (\mathbf{L}_M - \sigma^2 \mathbf{I})^{1/2} \mathbf{R}$

$$\sigma_{ML}^2 = \frac{1}{D-M} \sum_{i=M+1}^D \lambda_i$$

# 核主成分分析

## ● Kernel PCA

将主成分分析的线性假设一般化使之适应非线性数据

传统PCA:  $D$ 维样本  $\{\mathbf{x}_n\}, n = 1, 2, \dots, N, \sum_n \mathbf{x}_n = \mathbf{0}$

$$\mathbf{S}\mathbf{u}_i = \lambda_i \mathbf{u}_i \quad \mathbf{S} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n \mathbf{x}_n^T \quad \mathbf{u}_i^T \mathbf{u}_i = 1$$

核PCA: 非线性映射  $\phi(\mathbf{x}), \mathbf{x}_n \mapsto \phi(\mathbf{x}_n), \sum_n \phi(\mathbf{x}_n) = \mathbf{0}$

$$\mathbf{C}\mathbf{v}_i = \lambda_i \mathbf{v}_i \quad \mathbf{C} = \frac{1}{N} \sum_{n=1}^N \phi(\mathbf{x}_n) \phi(\mathbf{x}_n)^T$$

$$\longrightarrow \frac{1}{N} \sum_{n=1}^N \phi(\mathbf{x}_n) \{ \phi(\mathbf{x}_n)^T \mathbf{v}_i \} = \lambda_i \mathbf{v}_i$$

$$\longrightarrow \mathbf{v}_i = \sum_{n=1}^N a_{in} \phi(\mathbf{x}_n)$$



# 核主成分分析

## ● Kernel PCA

将主成分分析的线性假设一般化使之适应非线性数据

$$\text{核PCA: } \frac{1}{N} \sum_{n=1}^N \phi(\mathbf{x}_n) \phi(\mathbf{x}_n)^T \sum_{m=1}^N a_{im} \phi(\mathbf{x}_m) = \lambda_i \sum_{n=1}^N a_{in} \phi(\mathbf{x}_n)$$

$$k(\mathbf{x}_n, \mathbf{x}_m) = \phi(\mathbf{x}_n)^T \phi(\mathbf{x}_m)$$

$$\longrightarrow \frac{1}{N} \sum_{n=1}^N k(\mathbf{x}_l, \mathbf{x}_n) \sum_{m=1}^N a_{im} k(\mathbf{x}_n, \mathbf{x}_m) = \lambda_i \sum_{n=1}^N a_{in} k(\mathbf{x}_l, \mathbf{x}_n)$$

$$\longrightarrow \mathbf{K}^2 \mathbf{a}_i = \lambda_i N \mathbf{K} \mathbf{a}_i$$

$$\longrightarrow \mathbf{K} \mathbf{a}_i = \lambda_i N \mathbf{a}_i$$

# 等距映射

## ● ISO-Metric Mapping (ISOMAP)

保持数据点内在几何性质(测地距离)

对于给定数据  $X = \{x_1, x_2, \dots, x_N\}$ ，构造图  $G = \{V, E\}$

$V$ 是顶点集合， $E$ 是边的集合

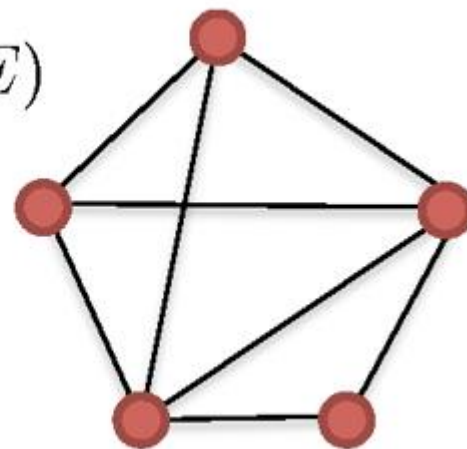
若  $d(i, j) = \text{dist}(x_i, x_j)$

$G = (V, E)$

小于某个值  $\in (\epsilon - \text{ISOMAP})$

或 $j$ 是 $i$ 的 $K$ 近邻( $K - \text{ISOMAP}$ )

则顶点 $i$ 与 $j$ 的边权值设为 $d(i, j)$ ，否则为0。



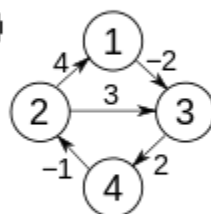
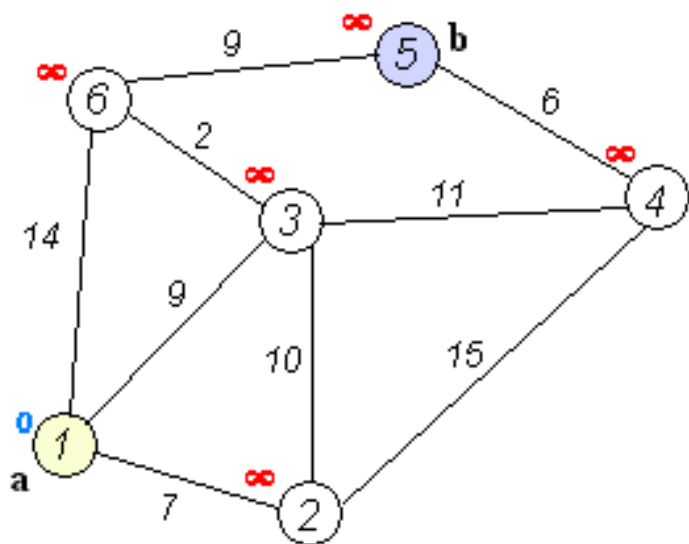
# 等距映射

## ● ISO-Metric Mapping (ISOMAP)

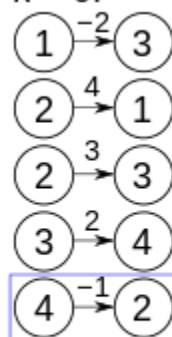
保持数据点内在几何性质(测地距离)

计算图  $G = \{V, E\}$  中任意两点间的最短距离，得到矩阵  $D_G(i, j)$

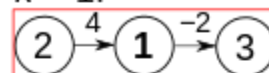
- Dijkstra最短路径算法
- Floyd-Warshall算法



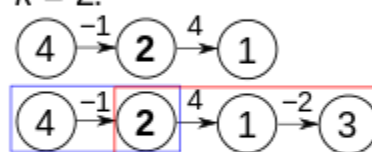
$k = 0$ :



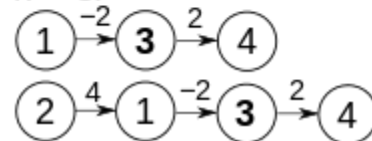
$k = 1$ :



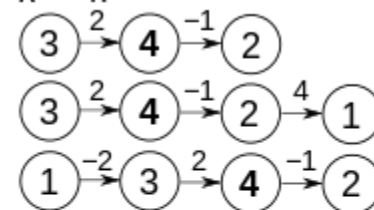
$k = 2$ :



$k = 3$ :



$k = 4$ :



# 等距映射

## ● ISO-Metric Mapping (ISOMAP)

保持数据点内在几何性质(测地距离)

令  $H = I_n - \frac{1}{N}11^T$  (中心矩阵, Centering Matrix)

并定义平方距离矩阵  $S(i, j) = D_G^2(i, j)$

求矩阵  $L = -\frac{1}{2}HSH$  的特征值与特征向量(按特征值降序排列),  $\lambda_p$  为第  $p$  个特征值,  $v_p$  为对应的特征向量。

降维矩阵为  $[\sqrt{\lambda_1}v_1, \dots, \sqrt{\lambda_d}v_d]_{N \times d}$

# 等距映射

## ● 计算步骤

### ①构造临近关系图

对每一个点，将它与指定半径邻域内所有点相连(或与指定个数最近邻相连)

### ②计算最短路径

计算临近关系图所有点对之间的最短路径，得到距离矩阵

### ③多尺度分析

将高维空间中的数据点投影到低维空间，使投影前后的距离矩阵相似度最大

# 等距映射

## ● ISOMAP优点和缺点

非线性

非迭代

全局最优

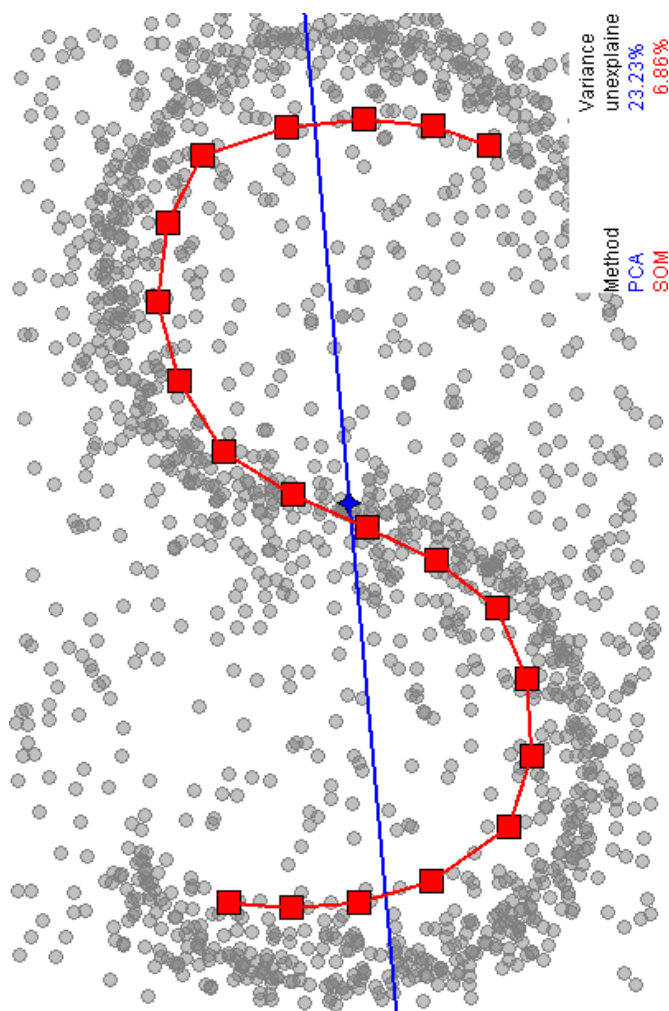
参数可调节

容易受噪声干扰

在大曲率区域存在短路现象

不适用于非凸参数空间

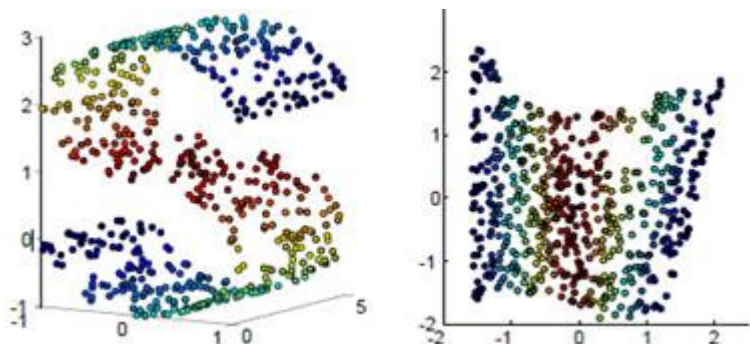
大样本训练速度慢



# 局部线性嵌入

## ● Local Linear Embedding (LLE)

保持数据点的原有流形结构



## Nonlinear Dimensionality Reduction by Locally Linear Embedding

Sam T. Roweis<sup>1</sup> and Lawrence K. Saul<sup>2</sup>

SCIENCE VOL 290 22 DECEMBER 2000

**前提假设：**采样数据所在的低维流形在局部是线性的，每个采样点可以用它的近邻点线性表示。

**学习目标：**在低维空间中保持每个邻域中的权值不变，即假设嵌入映射在局部是线性的条件下，最小化重构误差。

# 局部线性嵌入

## ● Local Linear Embedding (LLE)

保持数据点的原有流形结构

寻找每个样本点的 $K$ 近邻  $x_{ij} (j = 1, \dots, k)$

对每个点用 $K$ 个近邻进行重建，即求一组权值  $w_{ij}$ ,  $\sum_j w_{ij} = 1$

使  $\min \sum_i |x_i - \sum_j w_{ij} x_{ij}|^2$

求低维空间中的点集  $y_i$

使  $\min \sum_i |y_i - \sum_j w_{ij} y_{ij}|^2$



# 局部线性嵌入

## ● Local Linear Embedding (LLE)

保持数据点的原有流形结构

计算权值，首先构造局部协方差矩阵  $C^i$

$$C_{jk}^i = (x_i - x_j) \cdot (x_i - x_k)$$

然后，最小化  $\min \sum_i |x_i - \sum_j w_{ij} x_{ij}|^2$

$$s.t. \quad \sum_j w_{ij} = 1$$

最后可求得  $w_{ij} = \frac{\sum_k (C_{jk}^i)^{-1}}{\sum_{jk} (C_{jk}^i)^{-1}}$

# 局部线性嵌入

## ● Local Linear Embedding (LLE)

保持数据点的原有流形结构

计算低维数据，最小化  $\min \sum_i |y_i - \sum_j w_{ij} y_{ij}|^2$

可转化为最小化  $\min \sum_{ij} M_{ij} (y_i \cdot y_j)$

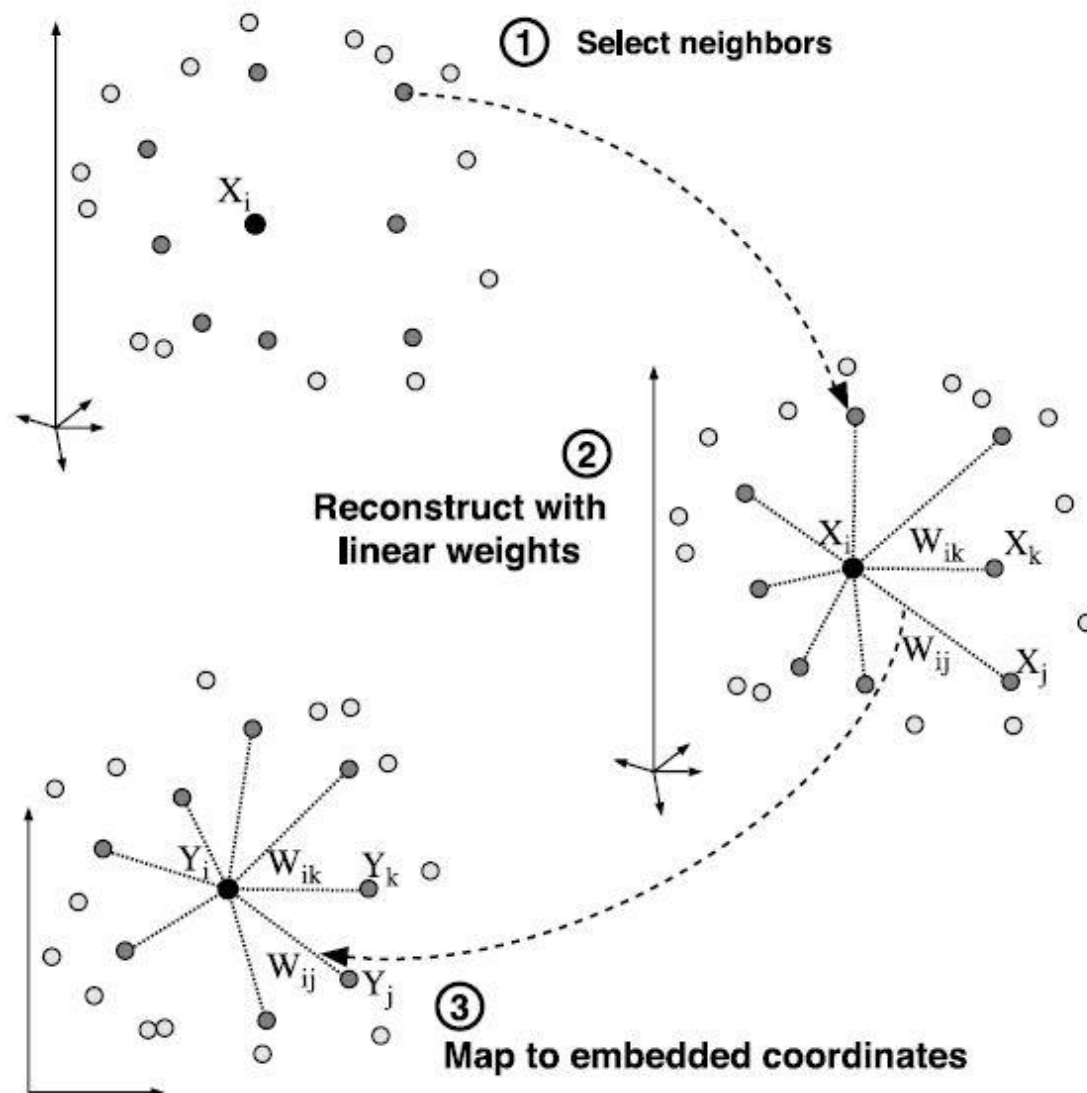
其中， $M = (I - W)^T (I - W)$

求解  $MY = \lambda Y$

取 $Y$ 为 $M$ 的最小 $d$ 个非零特征值所对应的特征向量，最终的输出结果即为 $N \times d$ 大小的矩阵

# 局部线性嵌入

## ● 计算步骤



# 局部线性嵌入

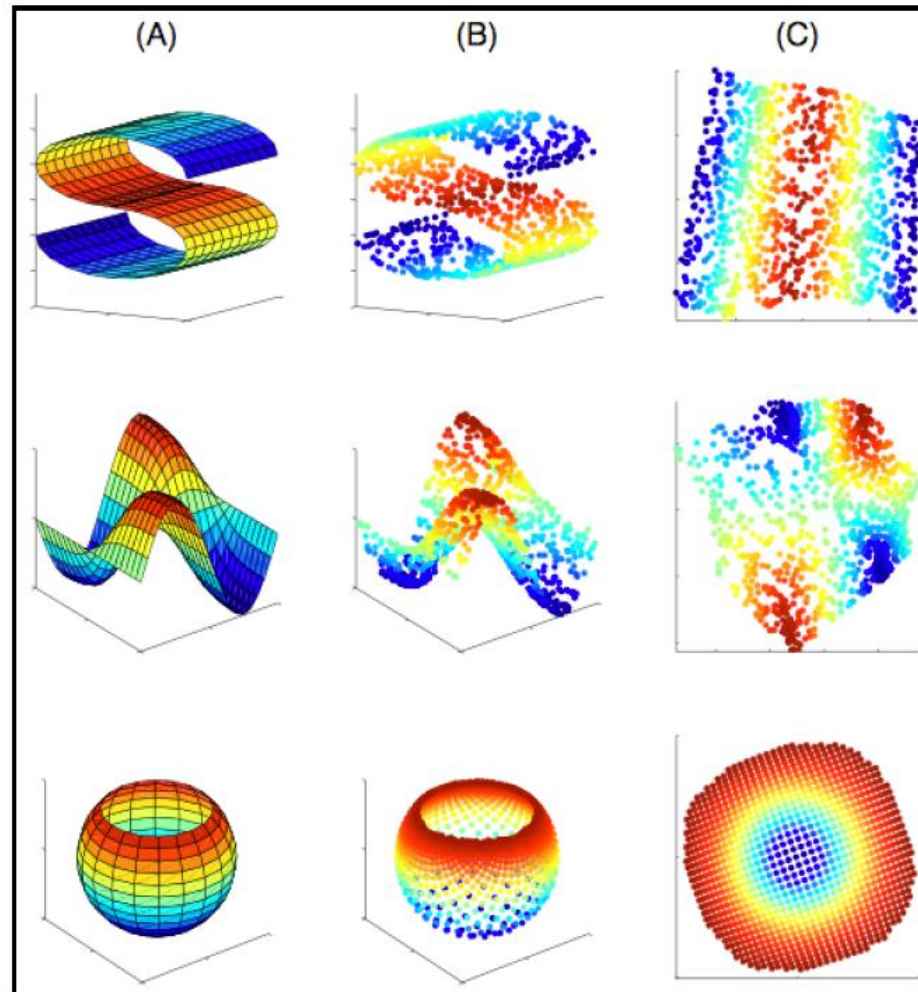
## ● Local Linear Embedding (LLE)

**Surfaces**

**N=1000**  
inputs

**k=8**  
nearest  
neighbors

**D=3**  
**d=2**  
dimensions



# 第10讲：采样方法

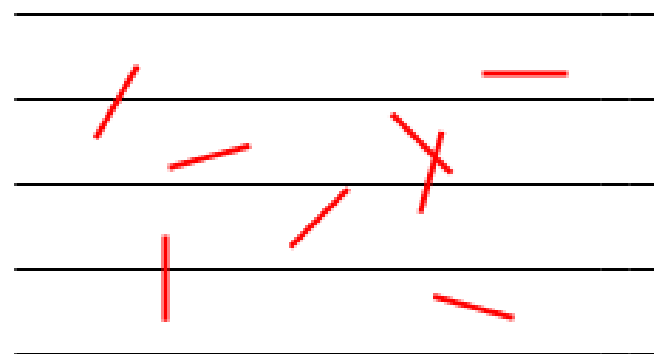
Chapter 10: Sampling Methods

# 布丰投针问题

- 1777年法国科学家布丰（Buffon）提出一种计算圆周率 $\pi$ 的方法—随机投针法。

1、在平面上画上间距为 $a$ 的平行线

2、取一根长度为 $l$  ( $l < a$ ) 的针，  
随机投掷于平面上，针与任一线相交的概率 $p$ 为



$$p = \frac{2l}{\pi a}$$

$\pi$ 的值可计算为：

$$\pi = \frac{2l}{ap} \approx \frac{2l}{a} \left( \frac{N}{n} \right)$$

$N$ 为总的投针次数， $n$ 为与平行线相交次数

试验者	时间	投掷次数	相交次数	圆周率估计值
Wolf	1850年	5000	2532	3.1596
Smith	1855年	3204	1218.5	3.1554
C.De Morgan	1860年	600	382.5	3.137
Fox	1884年	1030	489	3.1595
Lazzerini	1901年	3408	1808	3.1415929

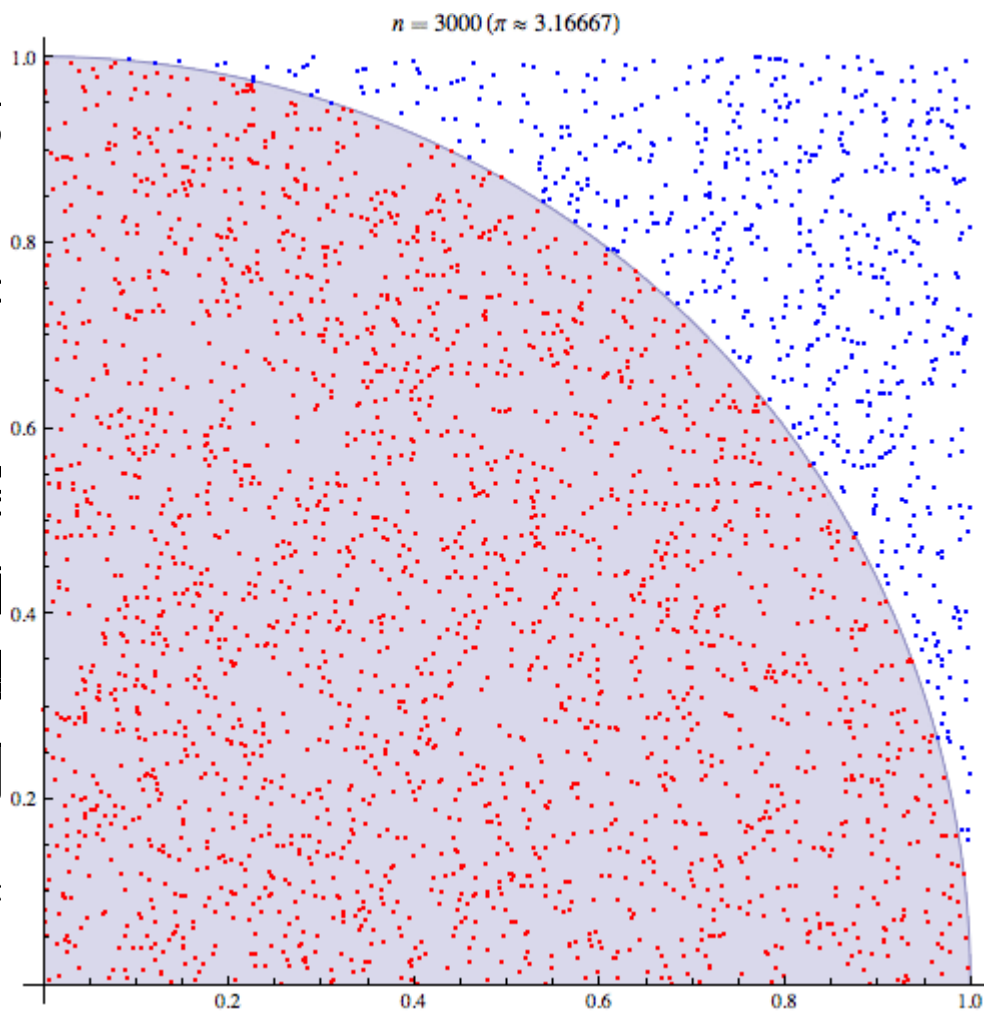
# 计算 $\pi$ - 另一个例子

- 正方形内部

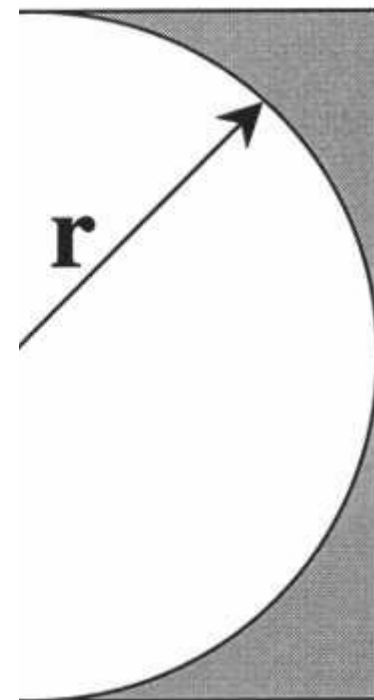
$$\frac{A_{\text{circle}}}{A_{\text{square}}} =$$

在正方形内随机生成  
个 $n$ 个点，计算每个点到原点的  
距离，判断点是否在圆内，统计为 $N_c$ ，则

$$\pi \approx 4 \frac{N_c}{n}$$



$\pi/4$

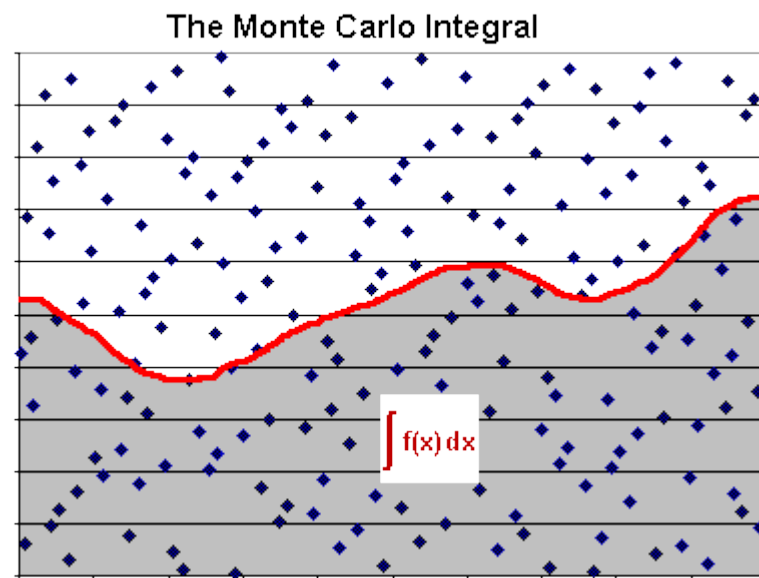


# 复杂函数定积分计算

- 上面的方法称为 **蒙特卡罗方法 (Monte Carlo method, Monte Carlo Simulation)** ,可以用于计算任一个函数的定积分。
- 如要求  $\int_a^b f(x) dx$  积分, 而  $f(x)$  积分的解析形式又很难求, 可通过数值方法求其近似值:

$$\begin{aligned}\int_a^b f(x) dx &= \int_a^b \frac{f(x)}{q(x)} q(x) dx \\ &\approx \frac{1}{N} \sum_{i=1}^N \frac{f(x_i)}{q(x_i)}\end{aligned}$$

其中  $q(x)$  是某种容易采样的分布,  $x_i$  是服从  $q(x)$  分布的随机样本



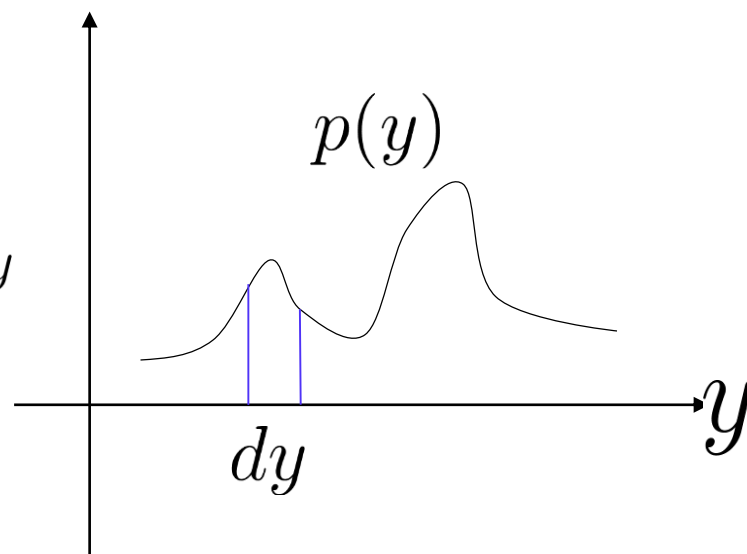
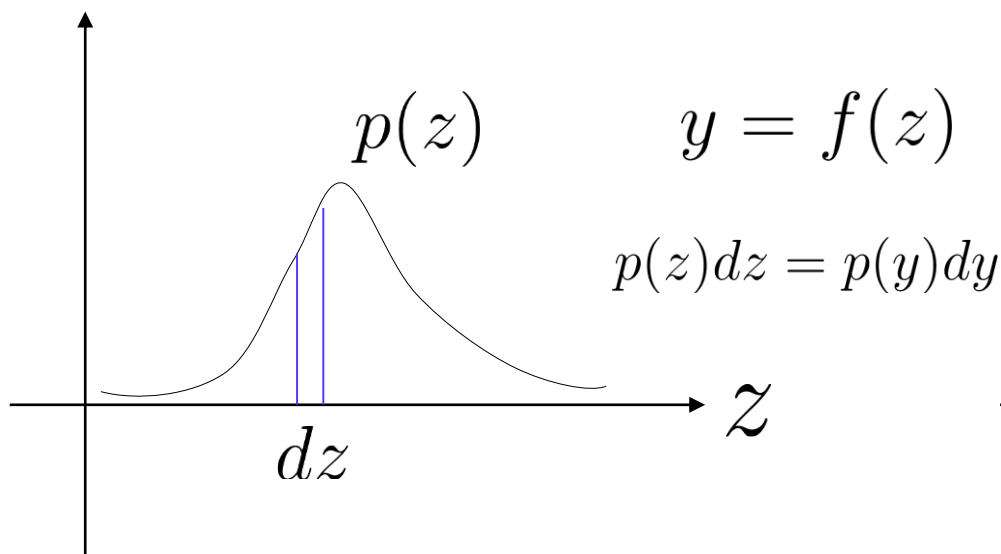


# 基本采样法 (Basic Sampling)

- 思想：从基本概率分布中产生新变量的分布

➤ 均匀分布 (Uniform distribution) :  $p(z) = 1 \quad z \in (0, 1)$

➤ 产生非均匀分布:  $p(y), \quad y = f(z)$



# 基本采样法 (Basic Sampling)

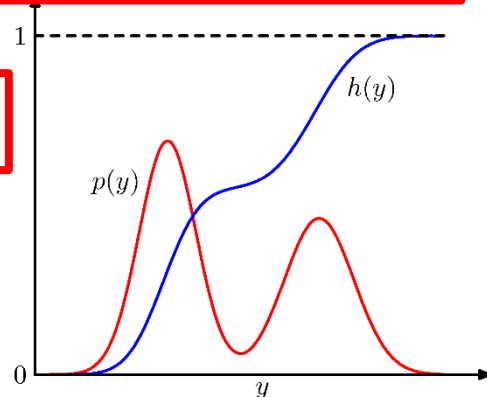
➤ 均匀分布:  $p(z) = 1 \quad z \in (0, 1)$

➤ 非均匀分布:  $p(y), \quad y = f(z)$

$$\left. \begin{array}{l} p(y) = p(z) \left| \frac{dz}{dy} \right| \\ p(z) = 1 \end{array} \right\} \rightarrow z = h(y) \equiv \int_{-\infty}^y p(\hat{y}) d\hat{y}$$

累积分布函数 (CDF)

$$\rightarrow y = h^{-1}(z)$$



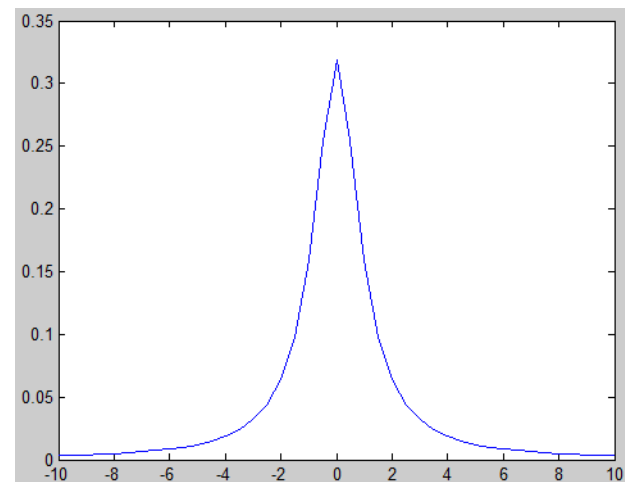
# 基本采样法 (Basic Sampling)

- 例子：标准柯西分布

已知  $z \sim U(0, 1)$

求  $y = f(z)$

使  $p(y) = \frac{1}{\pi} \frac{1}{1+y^2}$



➡ 
$$h(y) = \int_{-\infty}^y p(\hat{y}) d\hat{y} = \frac{1}{\pi} \arctan y + \frac{1}{2}$$

➡ 
$$y = h^{-1}(z) = \pi \tan(z - \frac{1}{2})$$

# 基本采样法 (Basic Sampling)

- **练习：指数分布**  $p(y) = \lambda \exp(-\lambda y) \quad y \in [0, \infty)$

求  $y = f(z)$

➡  $h(y) = \int_{-\infty}^y p(\hat{y}) d\hat{y} = 1 - \exp(-\lambda y)$

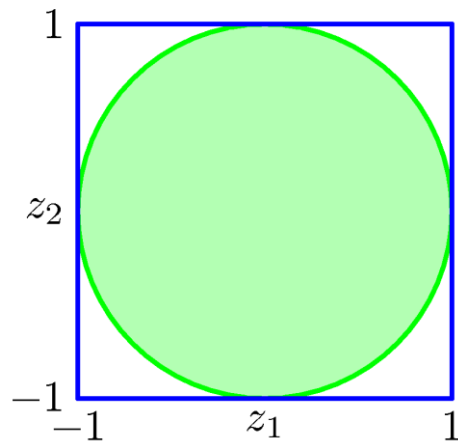
➡  $y = h^{-1}(z) = -\lambda^{-1} \ln(1 - z)$

- **多变量分布形式：**

$$p(y_1, \dots, y_M) = p(z_1, \dots, z_M) \left| \frac{\partial(z_1, \dots, z_M)}{\partial(y_1, \dots, y_M)} \right|$$

# 基本采样法 (Basic Sampling)

## ● 高斯分布:



$$z_1, z_2 \sim \text{unif} \begin{cases} y_1 = \sqrt{-2 \ln z_1} \cos(2\pi z_2) \\ y_2 = \sqrt{-2 \ln z_1} \sin(2\pi z_2) \end{cases}$$

$$p(z_1, z_2) = \frac{1}{\pi}, \quad (z_1 + z_2)^2 \leq 1$$

$$\begin{aligned} y_1 &= z_1 \left( \frac{-2 \ln z_1}{r^2} \right)^{1/2} \\ y_2 &= z_2 \left( \frac{-2 \ln z_2}{r^2} \right)^{1/2} \end{aligned} \quad r^2 = (z_1 + z_2)^2$$

$$p(y_1, y_2) = p(z_1, z_2) \left| \frac{\partial(z_1, z_2)}{\partial(y_1, y_2)} \right|$$

Box-Muller变换

$$= \left[ \frac{1}{\sqrt{2\pi}} \exp(-y_1^2/2) \right] \left[ \frac{1}{\sqrt{2\pi}} \exp(-y_2^2/2) \right]$$

# 基本采样法 (Basic Sampling)

- 高斯分布:

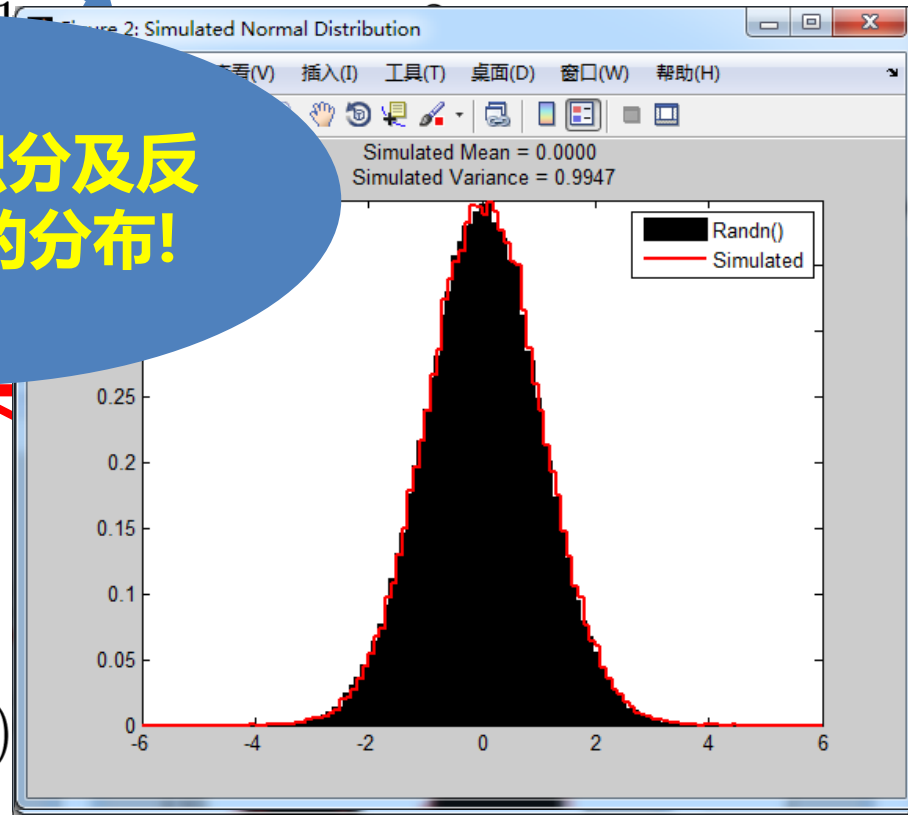
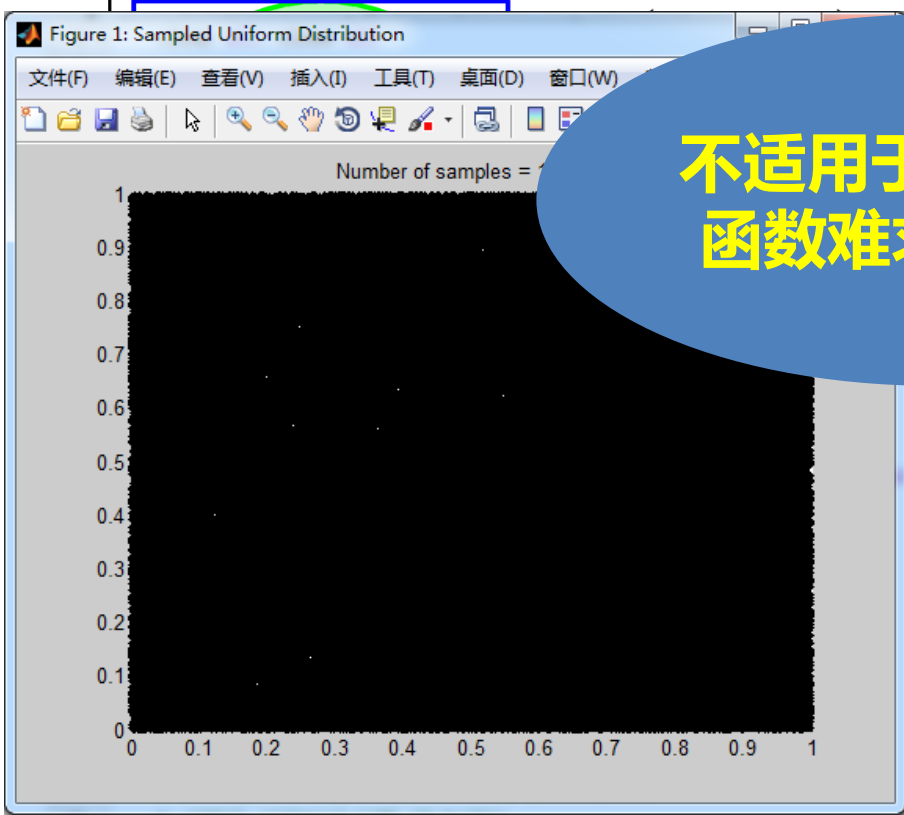
$z_1, z_2 \sim \text{Unif}(0, 1)$

$$y_1 = \sqrt{-2 \ln z_1} \cos(2\pi z_2)$$

$$y_2 = \sqrt{-2 \ln z_1} \sin(2\pi z_2)$$

不适用于积分及反函数难求的分布!

$\frac{z_2}{y_2} / 2)$



# 拒绝采样 (Rejection Sampling)

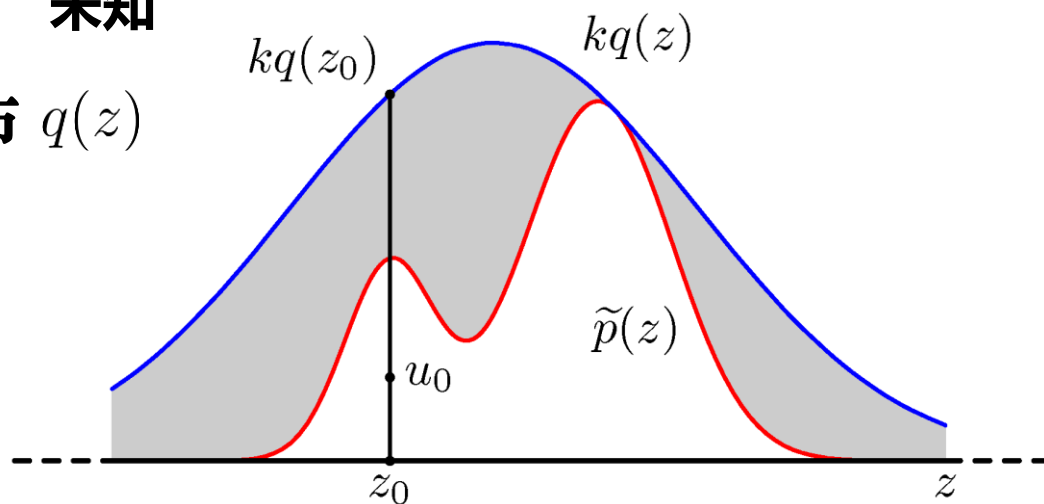
- 对一个很复杂的分布  $p(z)$  进行采样，但又无法给出其具体的解析形式，但是每个  $z$  可以计算其概率

$$p(z) = \frac{1}{Z_p} \tilde{p}(z)$$

已知  
未知

- 借助一个容易采样的分布  $q(z)$  去逼近  $\tilde{p}(z)$

$q(z)$  称为 **建议分布**  
**proposal distribution**

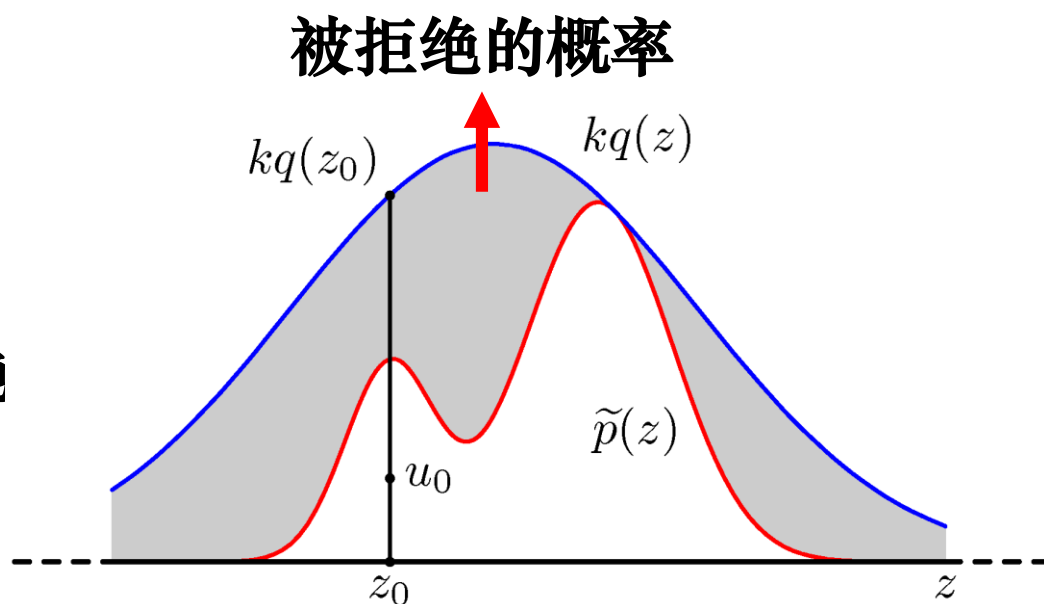


# 拒绝采样 (Rejection Sampling)

- 找到一个常数 $k$   $\tilde{p}(z) \leq kq(z)$   $\rightarrow$  对比函数 comparison function

- 采样步骤

- 1、从  $q(z)$  中产生  $z_0$
- 2、从  $[0, kq(z_0)]$  均匀分布中产生  $u_0$
- 3、如果  $u_0 > \tilde{p}(z_0)$  拒绝采样，否则接受采样



$$p(\text{accept}) = \int \{\tilde{p}(z)/kq(z)\}q(z)dz = \frac{1}{k} \int \tilde{p}(z)dz$$

拒绝采样又称为接受-拒绝采样 (Acceptance-Rejection Sampling)

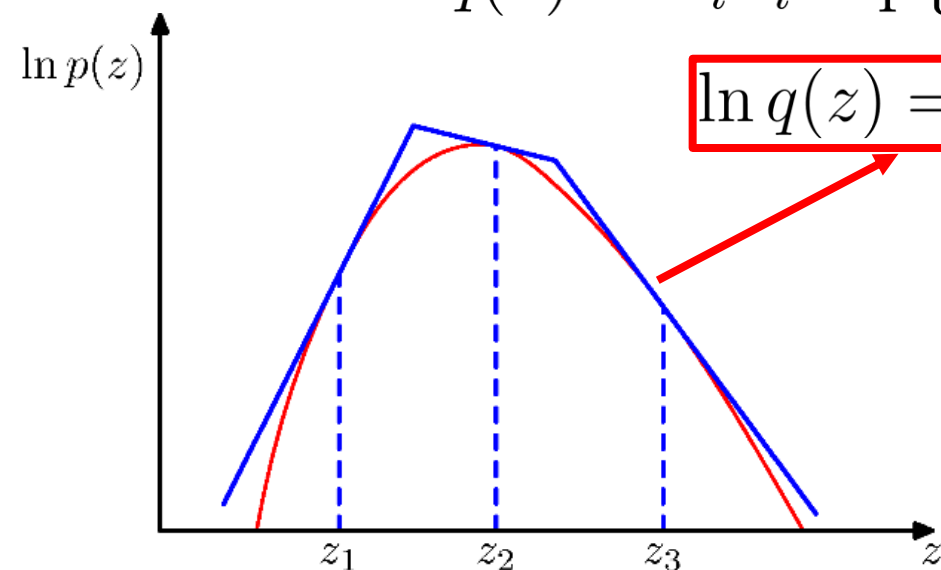


# 自适应拒绝采样 (Adaptive Rejection Sampling)

- 实际应用中，往往很难找到合适的  $q(z)$
- 特别地，当  $p(z)$  为log凸函数时，可采用ARS

$$q(z) = k_i \lambda_i \exp\{-\lambda_i(z - z_{i-1})\} \quad z_{i-1} < z \leq z_i$$

$$\ln q(z) = C - \lambda(z - z_{i-1}), \quad z_{i-1} < z \leq z_i$$



在log域执行拒绝采样

- 如果满足，接受
- 如果拒绝，重新逼近

# 重要性采样 (Importance Sampling)

- 对于某个变量  $z \sim p(z)$
- 一个关于  $z$  函数  $f(z)$ ，预测  $f(z)$  的关于  $z$  的期望：

$$\mathbb{E}(f) = \int f(z)p(z)dz$$

$p(z)$  很复杂，但可以估算每个出现  $z$  的概率  $p(z)$

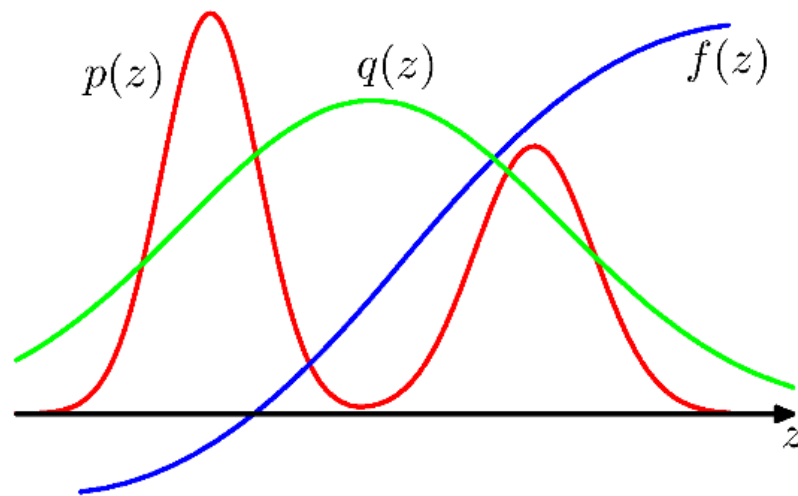
- 要估计  $\mathbb{E}(f) = \int f(z)p(z)dz$
- 可以按如下方式计算-将空间网络化

$$\mathbb{E}(f) \approx \sum_{l=1}^L p(z^{(l)})f(z^{(l)}) \longrightarrow \text{维数灾难}$$

# 重要性采样 (Importance Sampling)

- 与拒绝采样类似，借助一个容易采样的建议分布  $q(z)$

$$\begin{aligned}\mathbb{E}(f) &= \int f(z)p(z)dz \\ &= \int f(z)\frac{p(z)}{q(z)}q(z)dz \\ &\approx \frac{1}{L} \sum_{l=1}^L \boxed{\frac{p(z^{(l)})}{q(z^{(l)})}} f(z^{(l)})\end{aligned}$$



重要性权重

注:  $p(z) = \frac{1}{Z_p} \tilde{p}(z)$

# 重要性采样 (Importance Sampling)

- 同样地，我们也希望  $q(z)$  具有类似的性质，即

$$q(z) = \frac{1}{Z_q} \tilde{q}(z)$$


- 再看  $\mathbb{E}(f)$

$$\begin{aligned}\mathbb{E}(f) &= \int f(z)p(z)dz \\ &= \frac{Z_q}{Z_p} \int f(z) \frac{\tilde{p}(z)}{\tilde{q}(z)} q(z) dz \\ &\approx \boxed{\frac{Z_q}{Z_p}} \frac{1}{L} \sum_{l=1}^L \tilde{r}_l f(z^{(l)})\end{aligned}$$

$$\text{其中 } \tilde{r}_l = \frac{\tilde{p}(z^{(l)})}{\tilde{q}(z^{(l)})}$$

# 重要性采样 (Importance Sampling)

$$\begin{aligned}\frac{Z_p}{Z_q} &= \frac{1}{Z_q} \int \tilde{p}(z) dz \\ &= \int \frac{\tilde{p}(z)}{\tilde{q}(z)} q(z) dz \\ &\approx \frac{1}{L} \sum_{l=1}^L \tilde{r}_l\end{aligned}$$


$$\mathbb{E}(f) \approx \sum_{l=1}^L w_l f(z^{(l)})$$

$$w_l = \frac{\tilde{r}_l}{\sum_m \tilde{r}_m} = \frac{\tilde{p}(z^{(l)})/q(z^{(l)})}{\sum_m \tilde{p}(z^{(m)})/q(z^{(m)})}$$

# 马尔可夫蒙特卡罗方法 (Markov Chain Monte Carlo)

- 蒙特卡罗 (Monte Carlo) 蒙特卡洛坐落于欧洲地中海之滨、法国的东南方，有一个版图很小的国家**摩纳哥公国**。蒙特卡罗是世界著名的赌城，是摩纳哥的标志，**世界三大赌城之一**。富丽堂皇的蒙地卡罗赌场，建成于一八六三年，是一幢古色古香以及巍峨的宫殿式建筑物。1856年，摩纳哥亲王Charles三世为了解决财政危机，才在市区北边开设了第一家赌场。
- 蒙特卡罗方法于20世纪40年代美国在第二次世界大战中研制原子弹的“曼哈顿计划”计划的成员**S.M.乌拉姆**和**J.冯·诺伊曼**首先提出。数学家冯·诺伊曼用驰名世界的赌城—摩纳哥的Monte Carlo—来命名这种方法，为它蒙上了一层神秘色彩。在这之前，蒙特卡罗方法就已经存在。1777年，法国Buffon提出用投针实验的方法求圆周率 $\pi$ 。这被认为是蒙特卡罗方法的起源。



# 蒙特卡罗方法 (Monte Carlo Method)

注:

$$p(z) = \frac{1}{Z_p} \tilde{p}(z)$$

- 首先产生一个采样点  $z^{(\tau)}$
- 根据建议概率  $q(z|z^{(\tau)})$  产生新的采样点
- 依次类推, 产生马尔可夫链  $z^{(1)}, z^{(2)}, \dots$
- 要求  $q(z|z^{(\tau)})$  尽可能简单, 便于产生采样点;
- 有一个准则去决定是接受还是拒绝产生的采样点

# Metropolis采样法

- 建议概率:

$$q(\mathbf{z}_t | \mathbf{z}_{t-1})$$

- 接受概率:

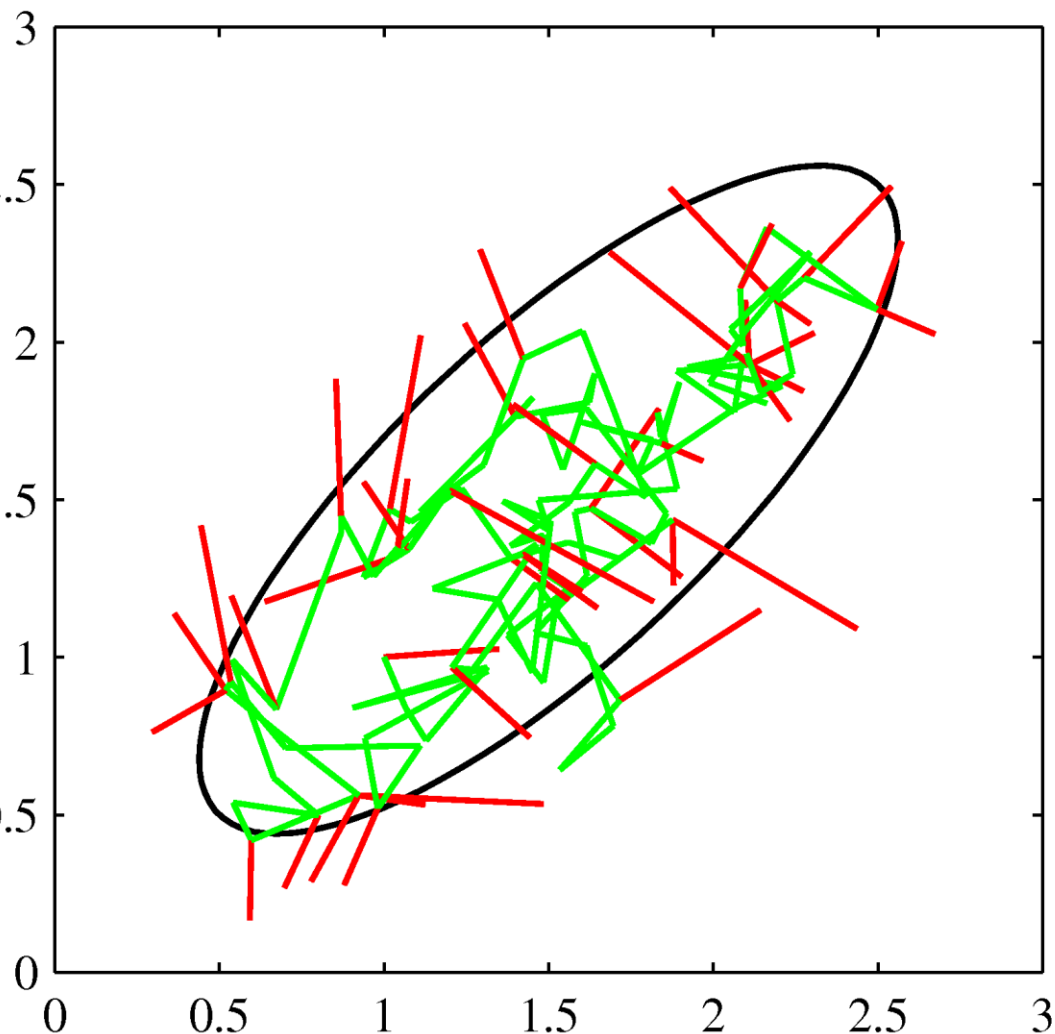
$$A(\mathbf{z}_t | \mathbf{z}_{t-1})$$

- 在(0,1)的均匀分布

- 如果  $u < A(\mathbf{z}_t | \mathbf{z}_{t-1})$

- 否则拒绝;

$\tau \rightarrow \infty$





# 马尔可夫链 (Markov Chain)

- 一阶马尔可夫链 (First Order Markov Chain)

$$p(z^{(m+1)} | z^{(1)}, \dots, z^{(m)}) = p(z^{(m+1)} | z^{(m)})$$

- 高阶马尔可夫 (High Order Markov Chain)

$$p(z^{(m+1)} | z^{(1)}, \dots, z^{(m)}) = p(z^{(m+1)} | z^{(m)}, \dots, z^{(m-n)})$$

- 转移概率 (Transition Probabilities)

$$T(z^{(m)}, z^{(m+1)}) \equiv p(z^{(m+1)} | z^{(m)})$$

- 转移概率矩阵

$$T = \begin{bmatrix} T(1,1), T(1,2), \dots, T(1,m) \\ T(2,1), T(2,2), \dots, T(2,m) \\ \vdots & \ddots & \vdots \\ T(m,1), T(m,2), \dots, T(m,m) \end{bmatrix}$$

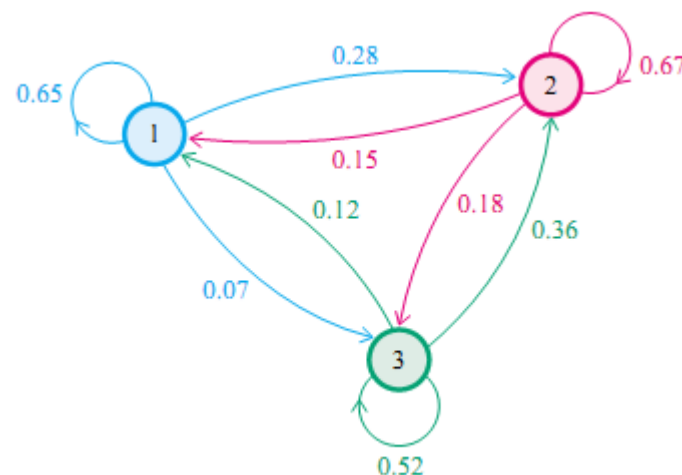
- 如果对所有的  $z^{(m)}$  都有相同的转移概率  $T_m$ ，则称为齐次马尔可夫 (Homogeneous Markov)

# 马尔可夫链 (Markov Chain)

## ● 例子:

社会学家经常把人按其经济状况分成3类：下层(lower-class)、中层(middle-class)、上层(upper-class)，我们用1,2,3 分别代表这三个阶层。社会学家们发现决定一个人的收入阶层的最重要的因素就是其父母的收入阶层。如果一个人的收入属于下层类别，那么他的孩子属于下层收入的概率是 0.65, 属于中层收入的概率是 0.28, 属于上层收入的概率是 0.07。事实上，从父代到子代，收入阶层的变化的转移概率如下

		子代		
	State	1	2	3
父代	1	0.65	0.28	0.07
	2	0.15	0.67	0.18
	3	0.12	0.36	0.52



# 马尔可夫链 (Markov Chain)

## ● 例子:

写成矩阵的形式:  $T = \begin{bmatrix} 0.65 & 0.28 & 0.07 \\ 0.15 & 0.67 & 0.18 \\ 0.12 & 0.36 & 0.52 \end{bmatrix}$

如果把当前这一段人处在下、中、上层的比例

那么他们子女所处阶层的分布比例将是:  $p_1$

他们孙子代各阶层的分布比例将是:  $p_2 = p_1 T$

第n代子孙各阶层的分布将是:  $p_n = p_{n-1} T$

假设初始概率分布为:  $p_0 = [0.21, 0.68, 0.11]$

计算后代各阶层的分布为:

第n代人	下层	中层	上层
0	0.210	0.680	0.110
1	0.252	0.554	0.194
2	0.270	0.512	0.218
3	0.278	0.497	0.225
4	0.282	0.490	0.226
5	0.285	0.489	0.225
6	0.286	0.489	0.225
7	0.286	0.489	0.225
8	0.289	0.488	0.225
9	0.286	0.489	0.225
10	0.286	0.489	0.225
...	...	...	...

# 马尔可夫链 (Markov Chain)

- 一个状态的边缘分布可以表示为

$$p(z^{(m+1)}) = \sum_{z^{(m)}} p(z^{(m+1)} | z^{(m)}) p(z^{(m)})$$

- 平稳性 (Stationary, 或不变性 Invariant)

$$p^*(z) = \sum_{z'} T(z', z) p^*(z')$$

- 细致平稳 (Detailed balance) - 充分条件

$$p^*(z) T(z, z') = p^*(z') T(z', z)$$

$$\sum_{z'} p^*(z') T(z', z) = \sum_{z'} p^*(z) T(z, z') = p^*(z) \sum_{z'} p(z' | z) = p^*(z)$$

# 马尔可夫链 (Markov Chain)

- 当  $m \rightarrow \infty$  马尔可夫链各状态趋于平稳, 即

$$p_m = p_{m-1}T \xrightarrow{m \rightarrow \infty} p = pT \quad p = [p(1), \dots, p(j), \dots]$$

平稳分布

- 同时

$$\lim_{m \rightarrow \infty} T^m = \begin{bmatrix} p(1), \dots, p(j), \dots \\ p(1), \dots, p(j), \dots \\ \dots, \dots \\ p(1), \dots, p(j), \dots \\ \dots, \dots \end{bmatrix} \quad \sum_j p(j) = 1$$

# Metropolis-Hastings 方法

- **思想**：对于需要采样的一分布  $p(z)$ ，构造一个转移矩阵为  $T$  的马尔可夫链，使它的平稳分布恰好为  $p(z)$
- 假设有一个转移矩阵  $Q(z, z') = q(z'|z)$ ， $q(z)$  为容易采样的分布
- 通常情况下，该转移矩阵难以满足细致平稳条件

$$p(z)q(z'|z) \neq p(z')q(z|z')$$

- 引入  $a(z, z')$  使

$$p(z)q(z'|z)a(z, z') = p(z')q(z|z')a(z', z)$$

其中：

$$\left. \begin{aligned} a(z, z') &= p(z')q(z|z') \\ a(z', z) &= p(z)q(z'|z) \end{aligned} \right\} \text{接受率}$$

# Metropolis-Hastings 方法

- 步骤:

- 1、初始化马尔可夫链状态  $z = z_0$
- 2、对  $\tau = 1, 2, \dots$  循环以下过程采样
  - 1) 第  $\tau$  个时刻马尔可夫链状态为  $z = z^{(\tau)}$  采样  $z^* = q(z|z^{(\tau)})$
  - 2) 从均匀分布中采样  $u \sim \text{uniform}[0, 1]$
  - 3) 如果  $u < a(z^{(\tau)}, z^*) = p(z^*)q(z^{(\tau)}|z^*)$  则接受  $z^{(\tau+1)} = z^*$
  - 4) 否则  $z^{(\tau+1)} = z^{(\tau)}$



如果  $a(z^{(\tau)}, z)$  过小，则采样效率较低！

# Metropolis-Hastings 方法

- 在细致平稳条件两边乘以因子 $C$

$$p(z)q(z'|z)a(z, z') \cdot C = p(z')q(z|z')a(z', z) \cdot C$$

**细致平稳条件并没有打破!!!**

- 同比例放大  $a(z, z')$ ,  $a(z', z)$  使最大的为1, 令

$$\begin{aligned} A(z, z') &= \min \left\{ 1, \frac{p(z)q(z'|z)}{p(z')q(z|z')} \right\} \\ &= \min \left\{ 1, \frac{\tilde{p}(z)q(z'|z)}{\tilde{p}(z')q(z|z')} \right\} \quad p(z) = \frac{1}{Z_p} \tilde{p}(z) \end{aligned}$$



# Metropolis-Hastings 方法

- 步骤:

- 1、初始化马尔可夫链状态  $z = z_0$
- 2、对  $\tau = 1, 2, \dots$  循环以下过程采样
  - 1) 第  $\tau$  个时刻马尔可夫链状态为  $z = z^{(\tau)}$  采样  $z^* = q(z|z^{(\tau)})$
  - 2) 从均匀分布中采样  $u \sim \text{uniform}[0, 1]$
  - 3) 如果  $u < A(z^{(\tau)}, z^*) = \min \left\{ 1, \frac{\tilde{p}(z^*)q(z'|z^*)}{\tilde{p}(z')q(z^*|z')} \right\}$  则接受  $z^{(\tau+1)} = z^*$
  - 4) 否则  $z^{(\tau+1)} = z^{(\tau)}$

# 吉布斯采样 (Gibbs Sampling)

- 一种特殊的M-H采样算法
- 针对多元分布进行采样  $p(\mathbf{z}) = p(z_1, \dots, z_M)$

**每次只改变一个维度上的值，保持其他维度不变**

$$p(z_1, z_2, z_3)$$

首先：初始化  $(z_1^{(0)}, z_2^{(0)}, z_3^{(0)})$

在第  $\tau$  步，假设已经产生了  $(z_1^{(\tau)}, z_2^{(\tau)}, z_3^{(\tau)})$

➡ 根据  $p(z_1 | z_2^{(\tau)}, z_3^{(\tau)})$  产生  $z_1^{(\tau+1)}$

➡ 根据  $p(z_2 | z_1^{(\tau+1)}, z_3^{(\tau)})$  产生  $z_2^{(\tau+1)}$

➡ 根据  $p(z_3 | z_1^{(\tau+1)}, z_2^{(\tau+1)})$  产生  $z_3^{(\tau+1)}$

# 与M-H的关系

- 建议概率：

$$q_k(\mathbf{z}^*|\mathbf{z}) = p(z_k^*|\mathbf{z}_{\setminus k})$$

- 接受概率：

$$A_k(z^*, z^{(\tau)}) = \min \left\{ 1, \frac{\tilde{p}(z^*)q_k(z^{(\tau)}|z^*)}{\tilde{p}(z^{(\tau)})q_k(z^*|z^{(\tau)})} \right\}$$

$$p(\mathbf{z}) = p(z_k|\mathbf{z}_{\setminus k})p(\mathbf{z}_{\setminus k}) \quad \mathbf{z}_{\setminus k}^* = \mathbf{z}_{\setminus k}$$

$$A(\mathbf{z}^*, \mathbf{z}) = \frac{p(\mathbf{z}^*)q_k(\mathbf{z}|\mathbf{z}^*)}{p(\mathbf{z})q_k(\mathbf{z}^*|\mathbf{z})} = \frac{p(z_k^*|\mathbf{z}_{\setminus k}^*)p(\mathbf{z}_{\setminus k}^*)p(z_k|\mathbf{z}_{\setminus k}^*)}{p(z_k|\mathbf{z}_{\setminus k})p(\mathbf{z}_{\setminus k})p(z_k^*|\mathbf{z}_{\setminus k})} = 1$$

# Slice Sampling

- Metropolis 方法的缺点：
  - 步长太短：走得太慢（可能随机散步）
  - 步长太长：拒绝率很好，效率较差；
- SLICE采样可以自适应调整步长
  - 将  $\mathcal{Z}$  空间扩展成  $(z, u)$  空间

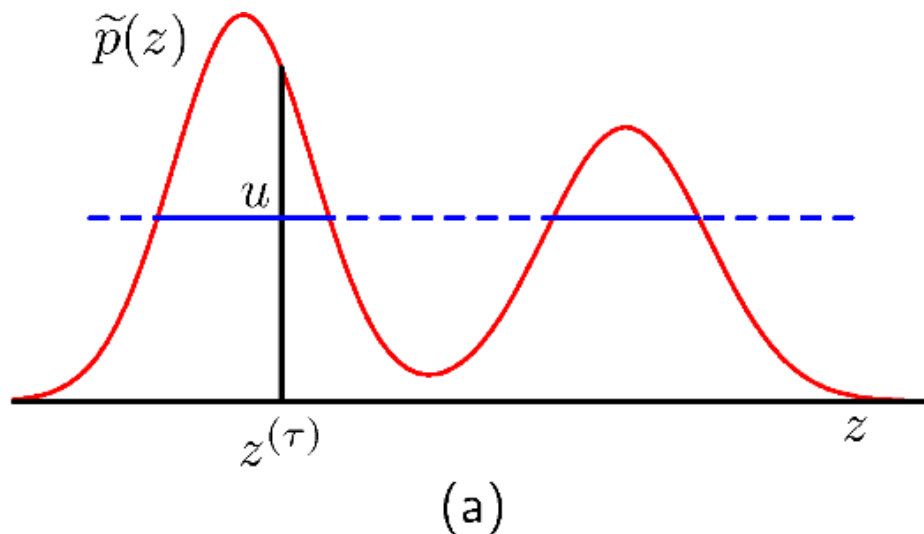
$$\hat{p}(z, u) = \begin{cases} 1/Z_p & \text{if } 0 \leq u \leq \tilde{p}(z) \\ 0 & \text{otherwise} \end{cases}$$

$$\int \hat{p}(z, u) du = \int_0^{\tilde{p}(z)} \frac{1}{Z_p} du = \frac{\tilde{p}(z)}{Z_p} = p(z)$$

# Slice Sampling

$$\hat{p}(z, u) = \begin{cases} 1/Z_p & \text{if } 0 \leq u \leq \tilde{p}(z) \\ 0 & \text{otherwise} \end{cases} \quad \int \hat{p}(z, u) du \int_0^{\tilde{p}(z)} \frac{1}{Z_p} du = \frac{\tilde{p}(z)}{Z_p} = p(z)$$

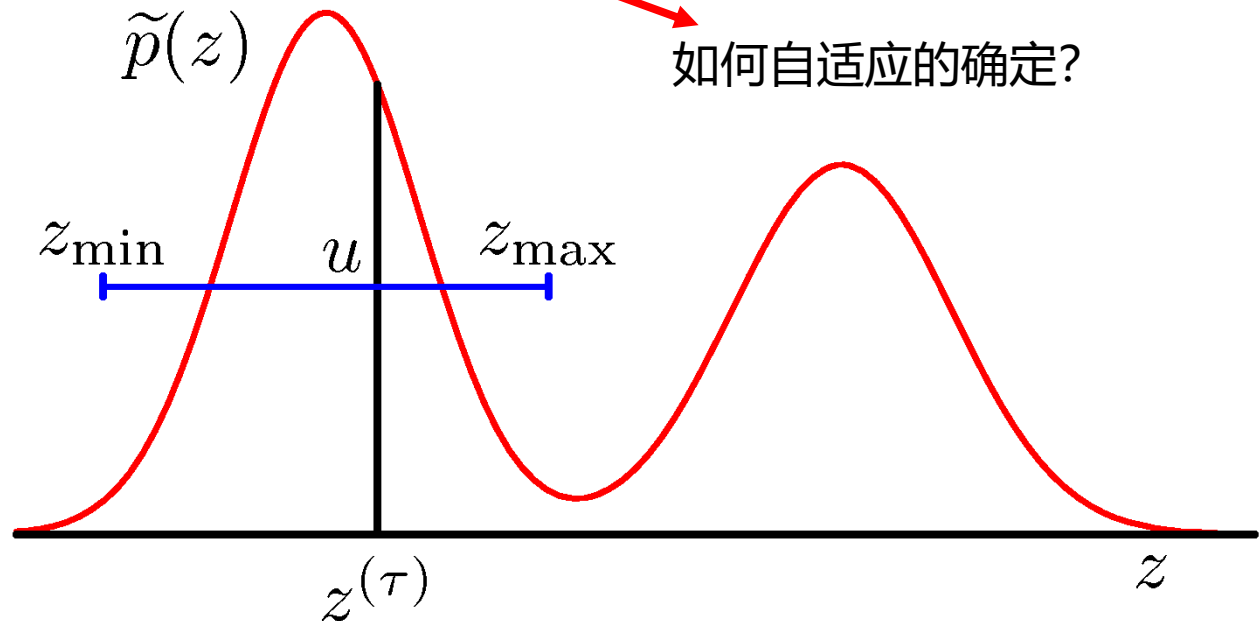
- 第一步：给定  $z$ ，在  $0 \leq u \leq \tilde{p}(z)$  范围内均匀分布产生  $u$
- 第二步：给定  $u$ ，在  $\{z : \tilde{p}(z) > u\}$  范围内均匀分布产生  $z$



# Slice Sampling

- 在实际应用中，很难确定范围

- 第一步：给定  $z$ ，在  $0 \leq u \leq \tilde{p}(z)$  范围内均匀分布产生  $u$
- 第二步：给定  $u$ ，在  $z_{\min} \leq z \leq z_{\max}$  范围内均匀分布产生  $z$



(b)