随机过程大作业

——马氏链蒙特卡洛方法

姓名：刘可淳

学号：2015011105

院系：电子工程系

日期：2017/11/19

摘要

本文首先探讨了马氏链蒙特卡洛方法(Markov Chain Monte Carlo, MCMC)在估值仿真中的应用，研究了二维高斯相关系数的估计，实现Metropolis-Hastings(MH)算法，对给定的二维高斯分布进行随机采样，使用随机生成的样本估计二维高斯相关系数，并对比了不同建议分布的收敛速度和拒绝概率。此外，我还对比了MCMC方法和Gibbs采样在二维高斯相关系数估计问题上的异同。

随后，用基于RJ(Reversible Jump)-MCMC及模拟退火算法(Simulated Annealing algorithm, SA)的算法训练了径向基函数(Radial Basis Function, RBF)网络。接着用不同的模型选择准则实现了基于遍历的RBF网络的选择，包括AIC(Akaike Information Criterion)方法、BIC(Bayesian Information Criterion)方法、MDL(Minimum Description Length)方法、MAP（Maximum a posteriori）方法和HQC（Hannan-Quinn Criterion）方法，对比不同模型选择方法得到的模型参数及均方误差(MSE)，分析不同方法的性能。

关键词：马氏链蒙特卡洛；RBF网络；RJMCMC；模型选择；

目录

一、介绍

（一） 课题背景和研究现状

蒙特卡洛（MCMC）方法是一种计算方法，原理是通过大量随机样本，去了解一个系统，进而得到所要计算的值。Metropolis-Hastings 算法是 MCMC 方法中的一种，它有效构造了一个以 π 为平稳分布的马尔可夫链，在高维空间积分及优化问题上发挥了重要作用。

在机器学习中，RBF 网络是一种非常经典的网络结构，它能够逼近任意的非线性函数，可以处理系统内的难以解析的规律性，具有良好的泛化能力，并有很快的学习收敛速度，已成功应用于非线性函数逼近、时间序列分析、数据分类、模式识别、信息处理、图像处理、系统建模、控制和故障诊断等。

但在学习的过程中，随着模型复杂度的增加，可能会出现过拟合的情况，因此 RBF 网络需要与模型选择方法结合起来，经典的方法有 AIC[1][1]、BIC[1][3]、MDL[1][2]、MAP[1][4]和 HQC[1][8]等，这些也是最常用的方法。文献[1][6]的作者基于 RJ-MCMC[1][5][1][5]和模拟退火算法(SA)提出了一种新的 RBF 模型算法，实现了对 RBF 网络更好的学习。

（二） 研究内容及主要贡献

本工作的研究目标是针对现有的 RJMCMC+SA、AIC、BIC、MDL、MAP、HQC 等模型选择方法，结合 RBF 网络，对数据进行学习，并对比模型选择方法的性能。本文工作主要分为四个部分：

1. 二维高斯的相关系数估计：利用 MCMC 对给定的高斯分布的相关系数进行估计。分析采样次数和建议分布对估计值、收敛速度和拒绝概率的影响。实现 Gibbs 采样，比较 Gibbs 采样和 MH 算法在二维高斯的相关系数估计问题上的优劣；
2. 在理解 RJMCMC 的基础上，设计并实现一个 RJMCMC 的例子，实现基于 RJMCMC+SA 的 RBF 网络，通过给定数据集进行训练，并得到测试集的结果；
3. 实现基于 AIC、BIC、MDL、MAP 的 RBF 网络，比较不同模型选择方法的性能；
4. 实现了实验要求中未提到的 HQC 模型选择方法，进行模型选择并对比分析。

（三） 本文结构

在接下来的第二章中，将对 MCMC 进行介绍，并应用 MH 算法对给定的二维高斯分布进行相关系数估计并分析结果，此外还实现了 Gibbs 采样，并对比两种采样方法的不同；在第三章中，介绍 RJMCMC、模拟退火算法和 RBF 网络，设计并实现了一个 RJMCMC 实验，实现了基于 RJMCMC 和 SA 的模型选择方法的 RBF 模型；第四章实现了基于其他经典的模型选择方法的 RBF 模型，并进行了不同模型选择方法的比较和性能分析，此外，本章中还实现了实验要求中未提到的模型选择方法 HQC 并进行分析；最后，在第五章中对整篇论文进行总结。

二、二维高斯的相关系数估计

（一） 马尔科夫链及 MCMC 介绍

马尔科夫链（Markov Chain）的一般理论框架可以参见《Statistical Digital Signal Processing and Modeling》（Hayes,1996）和《Probability, Markov Chains, Queues, and Simulation》(William J.Stewart,2013)，此处直接应用其相关性质。

蒙特卡洛算法往往涉及模拟观测和期望估计，以及多元分布的归一化常数估值，其精髓在于使用随机数（或更常见的伪随机数）来解决很多计算问题。而 MCMC 是马氏链理论与蒙特卡洛算法结合的一个重要应用。

蒙特卡洛算法的一个重要应用是估算积分，如下例，计算积分:

$$\int_a^b g(t)dt$$

蒙特卡洛思想是是把该积分转化为求某一概率密度$f(x)$下的期望，从而积分估值问题转化为从已知目标概率密度$f(x)$中产生随机样本的问题。其基本步骤就是产生(伪)随机数，使之服从该概率分布$f(x)$，再求出期望。难点就在于$f(x)$的产生。当变量 X 是一维的情况时，这很容易做到。但如果变量 X 取值于R^n，直接产生符合某一分布的独立样本通常是很难的。这就引入了 MCMC 方法：

MCMC 方法由 Metropolis(Metropolis.et.a1.1953)奠定基石，他提出在以π为平稳分布的马氏链上产生相互依赖的样本，进而马氏链的值可以看成是从分布π中抽去样本，从而可以根据遍历定理对积分进行估计。之后由 Hastings 对其加以推广形成了 Metropolis-Hastings(MH)算法。换句话说，MCMC 方法是在蒙特卡洛产生随机样本时，样本根据一条马氏链的跳转概率来迭代更新。

（二） Metropolis-Hastings 算法

对于下述二维高斯分布：

$$N\left\{\begin{pmatrix}x_1\\x_2\end{pmatrix}\middle|\begin{pmatrix}5\\10\end{pmatrix}\quad\begin{pmatrix}1&-1\\-1&4\end{pmatrix}\right\}$$

我们有μ= $\begin{pmatrix}5\\10\end{pmatrix}$, Σ=$\begin{pmatrix}1 & -1\\-1 & 4\end{pmatrix}$, 可立得$\begin{pmatrix}x_1\\x_2\end{pmatrix}$理论相关系数ρ=$\sqrt{\frac{\sigma_{12}\times\sigma_{21}}{\sigma_{11}\times\sigma_{22}}}=-0.5$。(证明详见附录 A)

采用 MCMC 算法进行模拟，其核心是建立一个平稳分布为$p(x)$的马氏链来得到$p(x)$的样本分布。针对所给二元高斯分布，MH 的采样步骤可以总结为以下流程：

1. Set i = 1；
2. Generate initial value $X^{(1)}=\begin{pmatrix}x_1\\x_2\end{pmatrix}$;
3. for i=1: N
   - Generate a y(i) from a proposal distribution f(x); (Gauss、Uniform and Exponential is tested separately)
   - Evaluate the acceptance probability α = min(1，$\frac{f(y^{(i)})*p(X^{(i-1)}|y^{(i)})}{f(X^{(i-1)})*p(y^{(i)}|X^{(i-1)})}$);
   - Generate a u from a Uniform(0,1) distribution;
   - If u < α, accept the proposal ,set $X^{(i)}=y^{(i)}$;

     else set $X^{(i)}=X$ (i-1)。

可以证明，通过 MH 方法构造出来的链满足马氏性且以$f(x)$为平稳分布(详见附录 B)。

1. 实验方法

在上述流程中，取$p(x|y)$即跳转概率为全空间均匀跳转，则它在计算中可以被约去。分别选取采样次数 N 为 5000、20000、50000 次，建议分布选为参数固定的高斯分布、均匀分布和指数分布，这三种分布的均值和目标分布相同，比较这些情况下的估计结果、拒绝概率和收敛速度。拒绝概率=拒绝次数/迭代次数。此外，在迭代 50000 次的基础上，对比了以上一次采样点为中心点的高斯分布和均匀分布的估计结果和拒绝概率。

固定参数的建议分布分别是(matlab)：

i. 高斯分布：[5*randn(1) 10*randn(1)]
ii. 均匀分布：[unifrnd(0,10) unifrnd(5,15)]
iii. 指数分布：[1/5*exp(-1/5*x) 1/10*exp(-1/10*x)]

变化中心的建议分布分别是(matlab)：

i. 高斯分布：[normrnd(x,[1 1])]
ii. 均匀分布：[unifrnd(x-5,x+5)]

2. 实验结果及分析

表 1 不同采样次数和固定参数的建议分布下的相关系数估计

采样次数	5000			20000			50000		
	高斯分布	均匀分布	指数分布	高斯分布	均匀分布	指数分布	高斯分布	均匀分布	指数分布
估计值	-0.5282	-0.4700	-0.4869	-0.5110	-0.5043	-0.5117	-0.5063	-0.4930	-0.4991
收敛速度				12000	13000	8000	22000	12000	17000
拒绝概率	0.9776	0.7956	0.9344	0.9721	0.7967	0.9382	0.9749	0.7973	0.9399

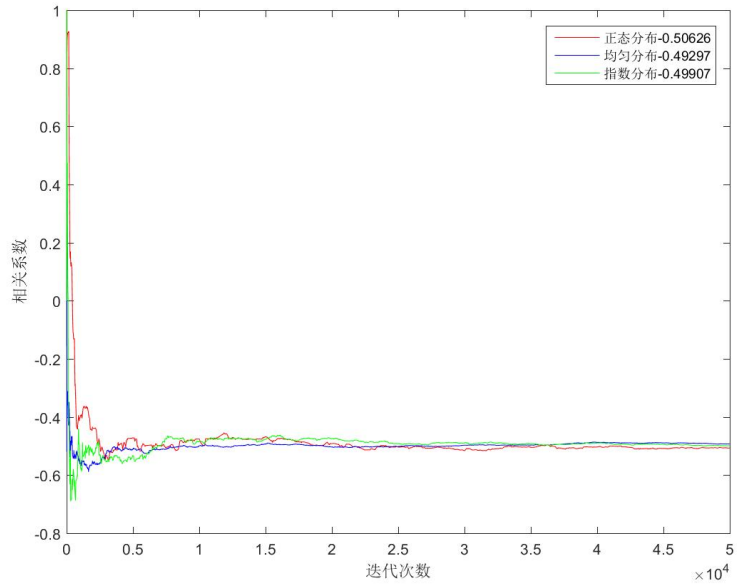
表 2 选取固定中心和变化中心的建议分布得到的结果

建议分布	固定中心		变化中心	
	高斯分布	均匀分布	高斯分布	均匀分布
估计值	-0.5063	-0.4930	-0.5041	-0.5025
拒绝概率	0.9749	0.7973	0.3946	0.8171

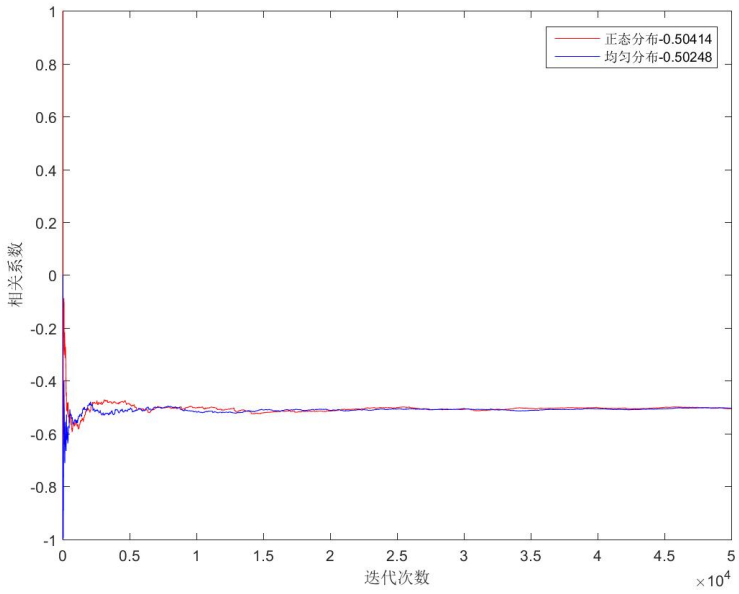
对$f(x)$每种分布而言，迭代次数增加，估计值会更加接近真实值，收敛速度由于结果样本小，不易估计，拒绝概率基本不变。

在相同的迭代次数下比较收敛速度。在 5000 次迭代时，从曲线中未看到明显的收敛趋势，故无法判定收敛速度；而在迭代次数较大时，不同建议分布的收敛速度无明显差异。

在相同的迭代次数下比较拒绝概率。三种分布的拒绝概率都较大，其中高斯分布最大，指数分布次之，均匀分布最小。分析得到可能是因为高斯分布与目标分布最为相似，且均值相同，因此最容易取到在目标分布里概率最高的点，而指数分布由于相同的原因也较容易取到目标分布里的概率最高的点，均匀分布则最不容易取到，从而拒绝概率相对低。



图片 1　50000 次迭代下固定中心点的建议分布的估计值曲线



图片 2　50000 次迭代下变化中心点的建议分布的估计值曲线

在迭代次数均为 50000 的情况下，两种不同类型的建议分布得到的估计值的误差接近，而在拒绝概率上不同类型的高斯分布有很大的区别，均匀分布则基本不变。

两种不同建议分布得到的估计值接近，说明估计的精度在较大迭代次数下与建议分布的选取关系不大。

变化中心点的高斯分布的拒绝概率比起固定中心点的小了很多，而均匀分布基本相同。分析原因可能是在固定点高斯分布的建议下，状态很容易在短时间内迅速跳转到概率最高的状态，而在变化中心点的高斯分布建议下，状态需要逐渐靠近概率最高的状态，因此拒绝概率较低；而均匀分布的拒绝概率差别不大，分析原因可能是因为变化中心点的均匀分布的范围取±5，这已经覆盖到了目标分布概率较大的区域，因此是否变化中心点差别不大。

（三） Gibbs 采样

在高维情况下，通常接受概率$\rho(X_n, Y)$并不高，导致 MH 算法的效率较低。Gibbs 算法在 MH 算法的基础上，从原始状态开始，每次随机对多维变量的一个维度进行采样，依相应变量的条件概率接受新的状态。这样可以有效提高算法效率，同时避免“维度灾难”。具体步骤如下：

1. Initialization: randomly set $\{x_i : i = 1, \dots, n\}$
2. Repeat Sampling, t = 0,1,2,……

i. $x_1^{(t+1)} \sim p(x_1 | x_2^{(t)}, x_3^{(t)}, \dots, x_n^{(t)})$

ii. $x_2^{(t+1)} \sim p(x_2 | x_1^{(t+1)}, x_3^{(t)}, \dots, x_n^{(t)})$

iii. …

iv. $x_j^{(t+1)} \sim p(x_j | x_1^{(t+1)}, \dots, x_{j-1}^{(t+1)}, x_{j+1}^{(t)}, \dots, x_n^{(t)})$

v. …

vi. $x_n^{(t+1)} \sim p(x_n | x_1^{(t+1)}, x_2^{(t+1)}, \dots, x_{n-1}^{(t)})$

该算法在各维度采样中，可以将每维度轮换采样换成随机采样，每步采样实现了高维空间$\mathcal{R}^n$中任意两点的转移。

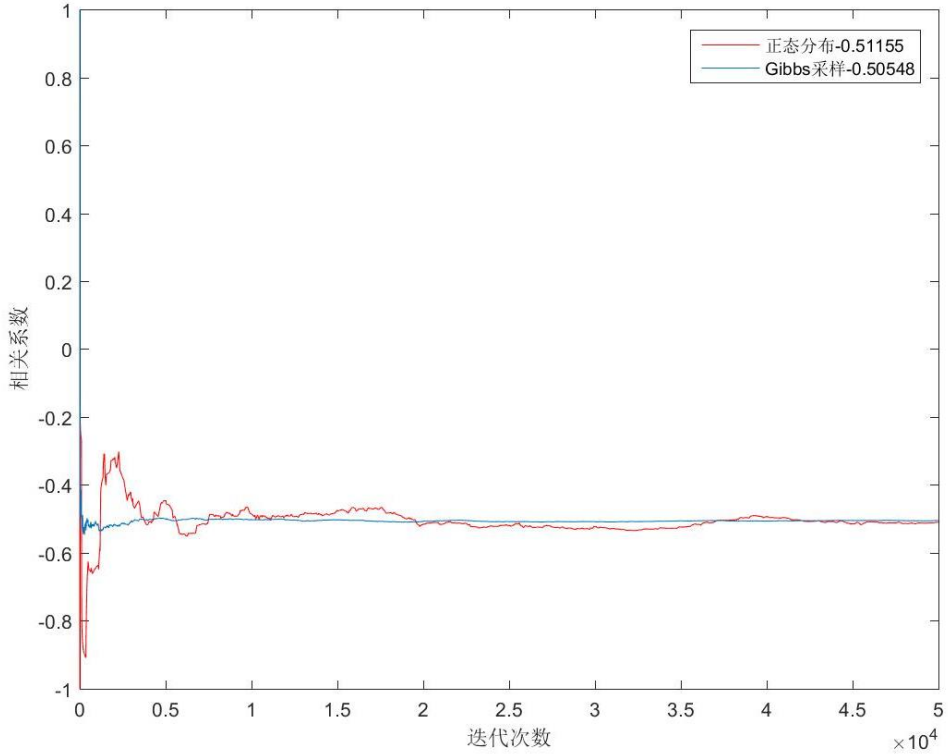
1. 实验方法

按照上述步骤实现了二维高斯的 Gibbs 采样。迭代次数为 50000 次。

2. 实验结果及分析

表 3　迭代 50000 次 Gibbs 采样和 MH 采样的结果对比

采样方法	Gibbs	MH （建议分布为固定中心的高斯）
估计值	-0.50548	-0.51155
收敛速度	300	约 20000



图片 3　Gibbs 采样和 MH 采样对二维高斯相关系数估计值曲线

在迭代次数相同的情况下，可以看出 Gibbs 采样比起 MH 采样更接近真实值，且 Gibbs 采样的收敛速度明显快于 MH 采样。这说明了在高维情况下，Gibbs 采样确实提高了算法效率。

三、基于 RJMCMC+SA 的 RBF 模型

（一） RJ-MCMC 介绍

与 MCMC 相似，RJMCMC 的目的也是为了产生一个以Π为平稳分布的马氏链，不同的是，MCMC 适用于变量个数即参数空间的维度是确定的问题中，而对于参数空间维度不定的问题，只能用 RJMCMC 来解决。

RJMCMC 可以在不同参数空间中跳转以产生样本，一个可以应用 RJMCMC 的例子是，在一张图片中，有 k 个物体（$k \leqslant k_{max}$），每个物体需要 n 个参数 θ 来描述，由于 k 是不确定的，所以参数空间的维度不确定，那么就需要 RJMCMC 来进行推断确定 k 的大小。

根据贝叶斯推断，已知样本点 y，后延概率=先验概率*调整因子，即

$$p(k,\theta^k|y) \propto p(y|k,\theta^k)p(k)p(\theta^k|k)$$

为了方便，记$x=(k,\theta^k)$，给定 k，x的全空间记为C_k，则$C_k=\{k\}\times\mathcal{R}^{n_k}$，记$C=\bigcup_{k\in\varkappa}C_k$，与 MCMC 相似，在 RJMCMC 中同样需要一个接受率 α，以满足细致平稳方程（证明详见[1][5]）。

$$\alpha_m(x,x') \approx \min\left(1, \frac{\pi(dx')q_m(x',dx)}{\pi(dx)q_m(x,dx')}\right)$$

其中，$q_m(x',dx)$表示类型 m 的从 x'跳转到 x 的概率，满足条件$0 < \sum_m q_m(x,\mathrm{C}) \leq 1$。

为了实现 RJMCMC，需要有 birth、death 和 update 三种跳转，birth 实现了从低维空间到高维空间的跳转，death 实现了从高维空间向低维空间的跳转，而 update 实现了在同一参数空间中的状态改变。具体到前文所述的例子中，birth 对应的是图片中物体个数的增加，death 对应的是图片中物体个数的减少，update 对应的则是物体参数的更新。显然，当 k=0 时，不存在 death 和 update 两种跳转，当 k=kmax 时，不存在 birth 这种跳转。

通过实现 RJMCMC，可以解决回归中的变量选择、非嵌套回归模型、不同数目参数的贝叶斯模型选择、多维变点问题、图像分割和物体识别等问题。

1. 实验设计

利用 RJMCMC 实现用 Gauss 函数拟合一元三次多项式，具体步骤如下：

1. Initialization: set $(k^{(0)}, \mu^{(0)}) \epsilon \Theta$
2. Iteration i.
   - Sample u~$\mathcal{U}(0,1)$
   - If (u $\leq b_{k^{(i)}}$)

     -then "birth" move
     -else if (u $\leq b_{k^{(i)}} + d_{k^{(i)}}$) then "death" move
     -else update the Gauss centres
     End if.
3. i ← i + 1 and go to 2.
4. Compute the coefficients $\alpha_{1:m}$ by least squares (optimal in this case):

$$\hat{\alpha}_{1:m,i} = [D'(\mu_{1:k}, x)D(\mu_{1:k}, x)]^{-1}D'(\mu_{1:k}, x)y_{1:N,i}$$

其中矩阵 D 有如下形式：

$$\begin{bmatrix} \varphi(x_1,\mu_1)\ldots & \varphi(x_1,\mu_k) \\ \varphi(x_2,\mu_1)\ldots & \varphi(x_2,\mu_k) \\ \vdots \quad \ldots & \vdots \\ \varphi(x_N,\mu_1)\ldots & \varphi(x_N,\mu_k) \end{bmatrix}$$

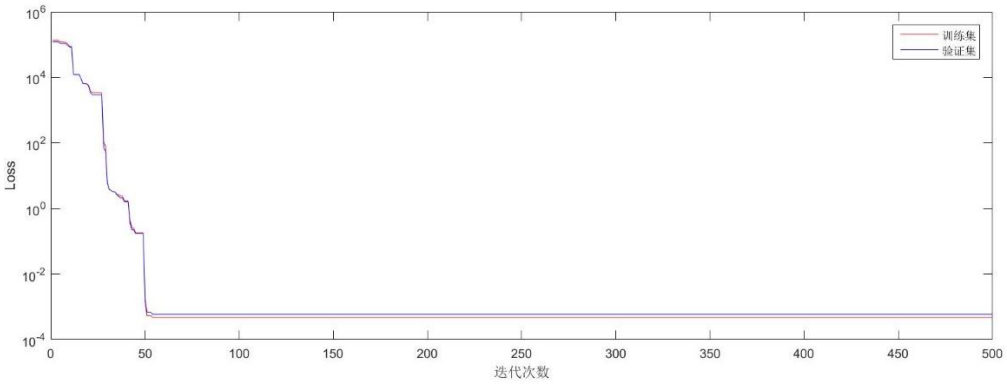
当 k=0 时，$b_{0^{(i)}} = 1$；当k = kmax时，$b_{kmax^{(i)}} = 0, d_{kmax^{(i)}} = 0.5$；其他情况下$b_{k^{(i)}} = d_{k^{(i)}} = 1/3$。

迭代次数取 2000 次，kmax为 100 个，随机取 1000 个满足$\mathrm{y} = x^3$的点，前 80%为训练集，后 20%为验证集，Loss 用 mse 衡量。

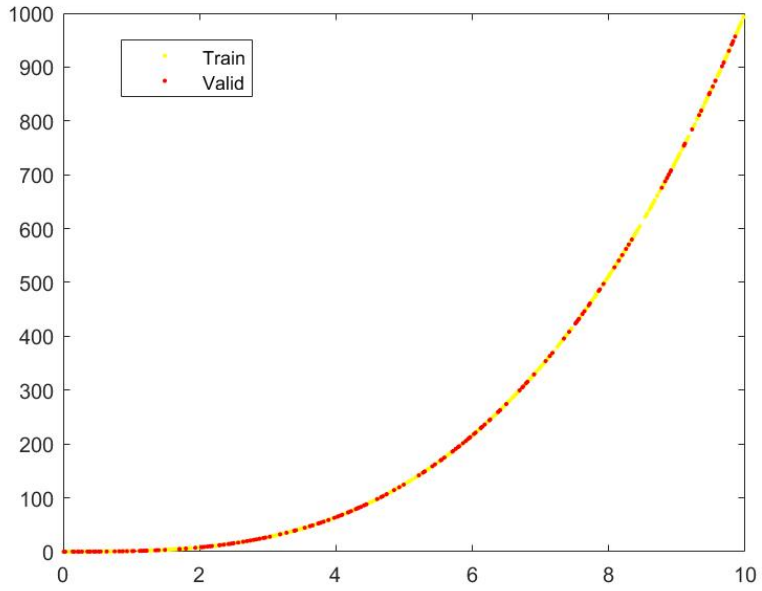
2. 实验结果

表 4　RJMCMC 高斯函数拟合三次方函数结果

数据集	训练集	验证集
函数个数 k	14	
Loss	4.7356e-04	5.9392e-04
收敛速度	55	55



图片 4　RJMCMC 拟合效果（Loss 曲线）



图片 5　训练数据和拟合结果

从 Loss 看，验证集上的 Loss 是 10^-4 量级，足够小，同时 Loss 曲线收敛速度相同，此外，将训练集和验证结果 plot 出来可以看出验证结果的确在$y = x^3$这个函数上，说明拟合足够精确。

（二） 模拟退火算法（SA）介绍

模拟退火是一种通用概率算法，用来在固定时间内寻求在一个大的搜寻空间内找到的最优解。算法先以搜寻空间内一个任意点作起始：每一步先选择一个“邻居”，然后再计算从现有位置到达“邻居”的概率。具体实现步骤如下：

1. 初始化：生成一个可行的解作为当前解输入迭代过程，并定义一个足够大的数值作为初始温度。
2. 迭代过程：
   a) 由一个产生函数从当前解产生一个位于解空间的新解；为便于后续的计算和接受，减少算法耗时，通常选择由当前新解经过简单地变换即可产生新解的方法，如对构成新解的全部或部分元素进行置换、互换等，注意到产生新解的变换方法决定了当前新解的邻域结构，因而对冷却进度表的选取有一定的影响。
   b) 计算与新解所对应的目标函数差。因为目标函数差仅由变换部分产生，所以目标函数差的计算最好按增量计算。事实表明，对大多数应用而言，这是计算目标函数差的最快方法。
   c) 判断新解是否被接受，判断的依据是一个接受准则，最常用的接受准则是 Metropolis 准则：若 $\Delta t'<0$ 则接受 S′作为新的当前解 S，否则以概率 $\exp(-\Delta t'/T)$ 接受 S′作为新的当前 S。
   d) 当新解被确定接受时，用新解代替当前解，这只需将当前解中对应于产生新解时的变换部分予以实现，同时修正目标函数值即可。此时，当前解实现了一次迭代。可在此基础上开始下一轮试验。而当新解被判定为舍弃时，则在原当前解的基础上继续下一轮试验。
3. 退火准则：在某个温度状态 T 下，当一定数量的迭代操作完成后，降低温度 T，在新的温度状态下执行下一个批次的迭代操作。

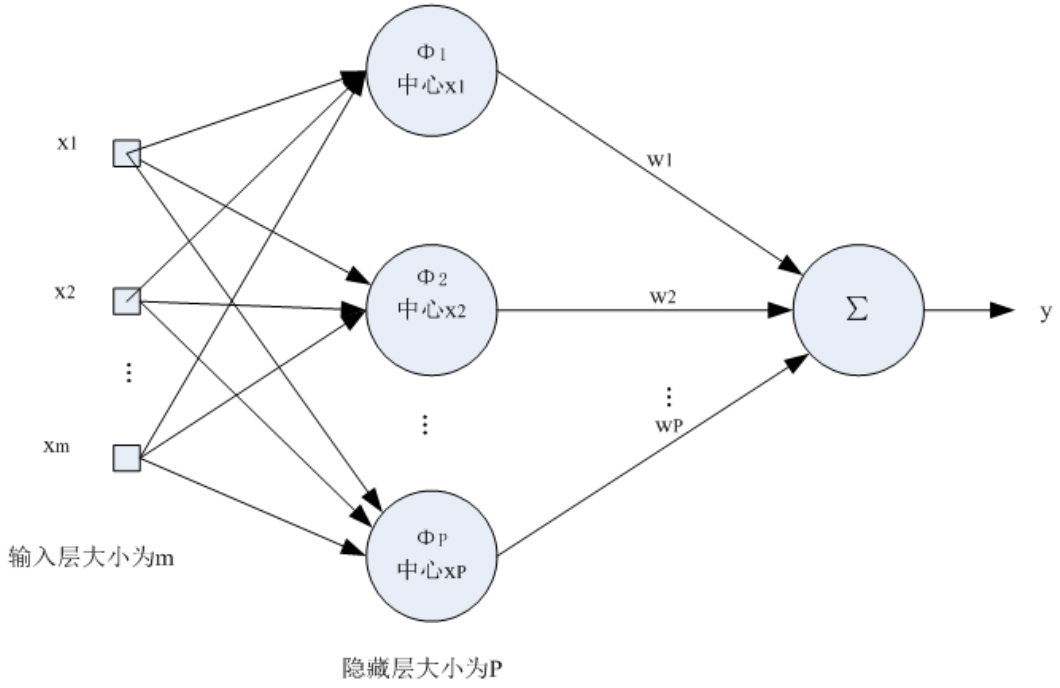
4. 停止准则：温度 T 降至某最低值时，完成给定数量迭代中无法接受新解，停止迭代，接受当前寻找的最优解为最终解。

（三） RBF 网络介绍

径向基函数（Radial Basis Function,RBF）是一个取值仅仅依赖于离原点距离的实值函数，也就是 Φ(x)=Φ(‖x‖)，或者还可以是到任意一点 c 的距离，c 点称为中心点，也就是 Φ(x,c)=Φ(‖x-c‖)。任意一个满足 Φ(x)=Φ(‖x‖)特性的函数 Φ 都叫做径向基函数。常用的径向基函数有：

1. Gauss（高斯）函数：$\varphi(r) = \exp(-\frac{r^2}{2\sigma^2})$
2. Reflected Sigmoid（反常 S 型）函数：$\varphi(r) = \frac{1}{1+\exp(\frac{r^2}{\sigma^2})}$
3. Inverse multiquadrics（拟多二次）函数：$\varphi(r) = \frac{1}{\sqrt{r^2+\sigma^2}}$

RBF 网络是一种三层静态前向网络，分别为输入层、隐含层和输出层。



图片 6 RBF 网络

它可以表示成

$$y_{N\times c} = D(\mu_{k\times d}, x_{k\times d})\alpha_{(1+d+k)\times c} + n_{N\times c}$$

其中 y 为观测样本，N 为样本点数，c 为每个样本点的维度；x 为 N 个观测样本点对应的变量，d 为每个变量的维度；矩阵 D 有如下形式

$$\begin{bmatrix} 1 & x_{1,1} & \cdots & x_{1,d} & \varphi(x_1,\mu_1) & \dots & \varphi(x_1,\mu_k) \\ 1 & x_{2,1} & \cdots & x_{2,d} & \varphi(x_2,\mu_1) & \dots & \varphi(x_2,\mu_k) \\ \vdots & & \vdots & & & \vdots & \\ 1 & x_{N,1} & \cdots & x_{N,d} & \varphi(x_N,\mu_1) & \dots & \varphi(x_N,\mu_k) \end{bmatrix}$$

$\varphi(x,\mu)$表示 RBF 函数，k 为 RBF 函数的个数，μ为函数的中心；α为加权系数，n是服从高斯分布的白噪声，$n_i \sim \mathcal{N}(0,\sigma_i^2), i = 1,2,\dots,c$。RBF 模型的参数集合可用$\mathcal{M}_s = (\mu,\alpha,\sigma^2,k)$表示。

（四） RJMCMC+SA

Christophe Andrieu, Nando de Freitas, Arnaud Doucet[1][6][1][6]提出一种基于 RJMCMC+SA 的 RBF 模型选择方法。具体实现步骤如下：

1. Initialization: set $(k^{(0)},\theta^{(0)})\epsilon\Theta$
2. Iteration i.
   • Sample $u\sim\mathcal{U}(0,1)$ and set the temperature with a cooling schedule.

   • If ($u \le b_{k^{(i)}}$)

   -then “birth” move
   -else if ($u \le b_{k^{(i)}} + d_{k^{(i)}}$) then “death” move
   -else if ($u \le b_{k^{(i)}} + d_{k^{(i)}} + s_{k^{(i)}}$) then “split” move
   -else if ($u \le b_{k^{(i)}} + d_{k^{(i)}} + s_{k^{(i)}} + m_{k^{(i)}}$)
   then “merge” move
   -else update the RBF centres
   End if.
3. $i \leftarrow i+1$ and go to 2.
4. Compute the coefficients $\alpha_{1:m}$ by least squares (optimal in this case):

$$\hat{\alpha}_{1:m,i} = [D'(\mu_{1:k},x)D(\mu_{1:k},x)]^{-1}D'(\mu_{1:k},x)y_{1:N,i}$$

其中$b_{k^{(i)}}, d_{k^{(i)}}, s_{k^{(i)}}, m_{k^{(i)}}, u_{k^{(i)}}$分别表示对于第 i 维数据，在 k 个 RBF 函数时，birth、death、split、merge 和 update 的系数。split 和 merge 是作者在 RJMCMC 的基础上新增的 move，split 表示将其中的一个中心点分裂成两个距离不远的中心点，merge 表示将两个距离不远的中心点合并成一个中心点，相应的 RBF 函数个数 k 会有变化（具体过程见[1][6]）。

1. 实验方法

按照文献中说明的步骤，实现了 RJMCMC+SA 的 RBF 模型。其中，数据集的前 80%划分为训练集，后 20%划分为验证集，惩罚项用 MDL 准则，loss 用 mse 来衡量。实现步骤中的细节如下：

1. 初始化：函数个数 k=0; 函数个数上限 kmax=200; 迭代次数 iteration=3000；初始温度 T=100；
2. 迭代：
   随机产生一个[0,1]区间的数 u，判断即将进行的 move；
   i. birth：在[x_{min}-3,x_{max}+3]中均匀随机生成一个新的中心点，添加进 mu 中，计算接受概率 Abirth，更新；

ii. death：随机取一个现有的中心点，从 mu 中删去，计算接受概率 Adeath，更新；

iii. split：随机取一个中心点 mu_i 并在[0,1]之间随机生成一个分裂长度 u，得到新的两个中心点 mu_i-u 和 mu_i+u，计算接受概率 Asplit，更新；

iv. merge：随机取一个中心点 mu_i 并在中心点空间中找到一个离 mu_i 最近的中心点，将两个点融合成它们的中点，计算接受概率 Amerge，更新；

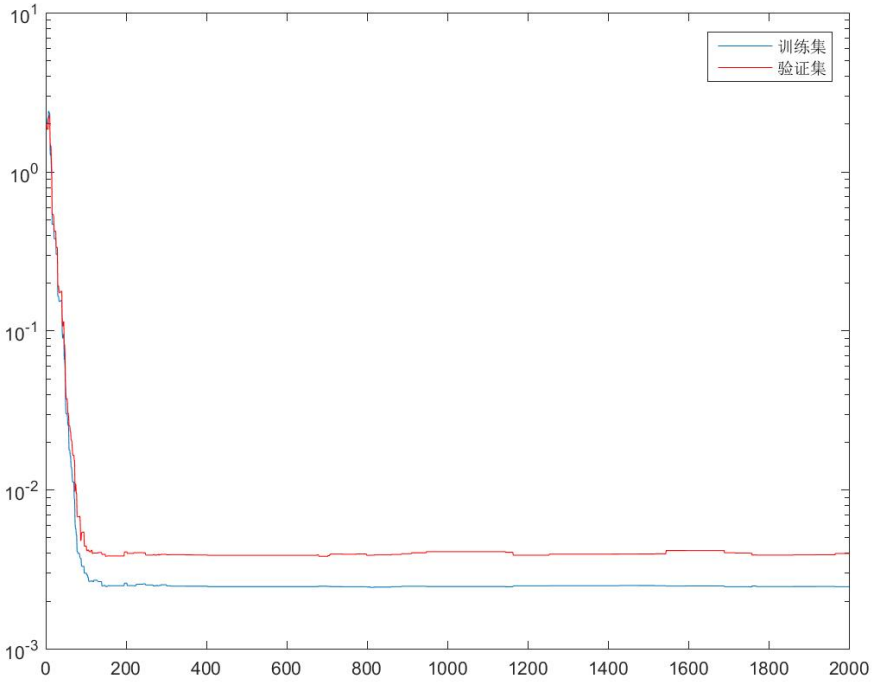
v. update：随机选取一个中心点，以它为中心用高斯分布随机出一个新的中心点来替换原有中心点，计算接受概率 Aupdate，更新；

需要注意的是在 birth 步骤中，产生新的中心点的范围，最开始实验时我用的范围是[x_{min},x_{max}]，但拟合效果一般，data2 的验证集 Loss 始终在 10^2 量级。经过修改把范围拓宽后 Loss 降了一个数量级。

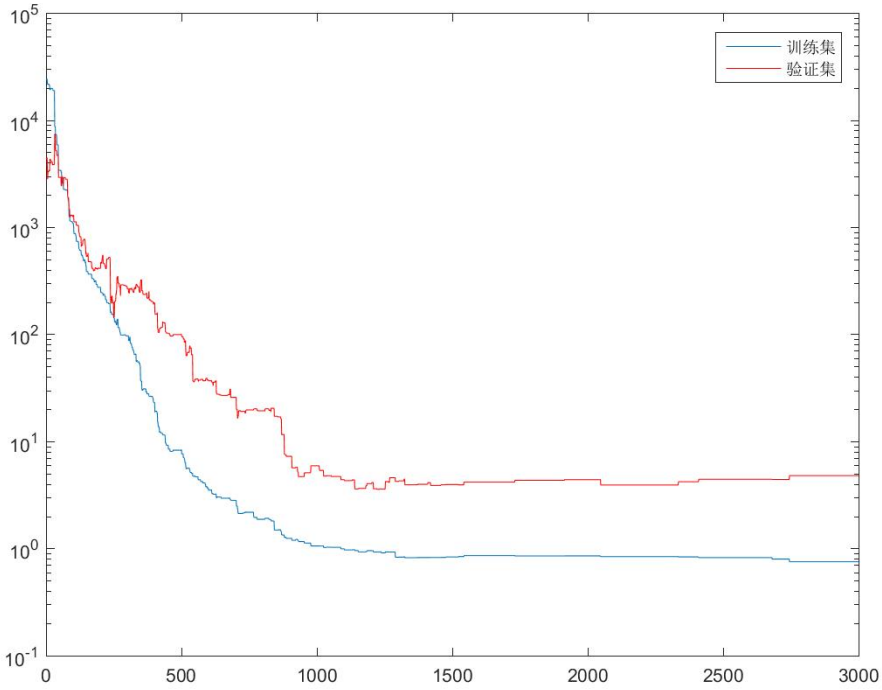
2. 实验结果

表 5 RJMCMC+SA 的 RBF 拟合结果

数据集	Data1		Data2	
	测试集	验证集	测试集	验证集
RBF 函数个数	34		108	
Loss	0.0024	0.0038	0.7519	3.5642
收敛速率	140	140	1300	1300



图片 7 data1 的 loss（对数坐标）



图片 8　data2 的 loss（对数坐标）

四、不同模型选择方法的 RBF 模型

（一） 模型选择方法介绍

随着模型复杂度增加，训练误差波动降低，平均训练误差降低趋向于 0，而测试误差波动上升，平均测试误差先降低后升高。这个现象说明训练误差不能代替测试误差来作为模型选择和评价的手段。复杂的模型可能在训练集上拟合的很好，但是面对新的测试集，预测误差不降反升，发生了所谓的 “过拟合” 现象，所以面对不同的模型，需要一个模型评价的准则来进行模型选择。常见的模型评价准则有三种角度：重复抽样与预测稳定性角度、似然与模型复杂度角度、VC 维与风险上界控制角度，典型的例子如下：

i. 重复抽样与预测稳定性角度：CV、GCV、Boostrap
ii. 似然与模型复杂度角度：AIC[1][1]、BIC[1][3]、MDL[1][2]、MAP[1][4]、HQC[1][8]
iii. VC 维与风险上界控制角度：SRM

在本次实验的所有实现中，均使用了交叉验证(CV)法，也就是将数据集划分成训练集和验证集。接下来具体介绍从似然与模型复杂度角度出发的五种模型选择方法。

1. AIC 准则

Akaike 信息准则（AIC）是给定数据集的统计模型的相对质量的估计量。鉴

于数据模型的集合，AIC 估计每个模型相对于其他模型的质量。因此，AIC 为模型选择提供了一种手段。

AIC 建立在信息论的基础上：它提供了一个给定模型用于表示生成数据的过程时丢失的相对信息的估计。在这样做的时候，它处理的是模型的合适性和模型的复杂性之间的平衡。

AIC 在测试无效假设的意义上不提供对模型的测试。它没有提到模型的绝对质量，只是相对于其他模型的质量。因此，如果所有候选模型都不合适，AIC 将不会给出任何警告。

假设我们有一些数据的统计模型。令ξ是模型中估计参数的数量。假设$\hat{L}^2$是模型的极大似然。那么模型的 AIC 值如下：

$$\text{AIC} = -2log\hat{L}^2 + 2\xi$$

在 RBF 模型中，$\xi = k(c+1) + c(1+d)$，且极大似然$\hat{L}^2 = (\widehat{\sigma_k}^2)^{-N/2}$，其中$\widehat{\sigma_k}^2$为残差平方和。整理 AIC 得到：

$$\text{AIC}:\mathcal{M}_s = \arg\min_{(\mathcal{M}_k:k\in\mathcal{Z}_q)}\left\{\frac{N}{2}log\widehat{\sigma_k}^2 + \xi\right\}$$

当两个模型之间存在较大差异时，差异主要体现在似然函数项，当似然函数差异不显著时，上式第二项，即模型复杂度则起作用，从而参数个数少的模型是较好的选择。

一般而言，当模型复杂度提高（k 增大）时，似然函数 L 也会增大，从而使 AIC 变小，但是 k 过大时，似然函数增速减缓，导致 AIC 增大，模型过于复杂容易造成过拟合现象。目标是选取 AIC 最小的模型，AIC 不仅要提高模型拟合度（极大似然），而且引入了惩罚项，使模型参数尽可能少，有助于降低过拟合的可能性。

2. BIC 准则

Bayesian Information Criterion 贝叶斯信息准则(BIC)与 AIC 相似，用于模型选择，1978 年由 Schwarz 提出。训练模型时，增加参数数量，也就是增加模型复杂度，会增大似然函数，但是也会导致过拟合现象。针对过拟合问题，AIC 和 BIC 均引入了与模型参数个数相关的惩罚项，而 BIC 的惩罚项比 AIC 的大，考虑了样本数量，样本数量过多时，可有效防止模型精度过高造成的模型复杂度过高。模型 BIC 值如下：

$$\text{BIC} = -2log\hat{L}^2 + \log n * \xi$$

其中 n 为样本数量，其余符号与 AIC 中符号含义相同，整理 BIC 得到：

$$\text{BIC}:\mathcal{M}_s = \arg\min_{(\mathcal{M}_k:k\in\mathcal{Z}_q)}\left\{\frac{N}{2}log\widehat{\sigma_k}^2 + \frac{\xi}{2}\log N + \left(\frac{\xi}{2}+1\right)\log(\xi+2) + O(\log\log n)\right\}$$

上式是 BIC 的原始形式，但在一般应用中并不使用上式。假设样本数量 N 趋于无穷且样本之间相互独立，则 BIC 可以化简成和 MDL 一样的形式：

$$\text{BIC}:\mathcal{M}_s = \arg\min_{(\mathcal{M}_k:k\in\mathcal{Z}_q)}\left\{\frac{N}{2}log\widehat{\sigma_k}^2 + \frac{\xi}{2}\log N\right\}$$

3. MDL 准则

最小描述长度（MDL）原则是奥卡姆剃刀的一种形式化，其中针对给定数据集的最佳假设（模型及其参数）是导致数据最佳压缩的假设。 MDL 由 Jorma Rissanen 于 1978 年提出，其目的是为了根据信息论中的基本概念来解释极大后验假设(MAP)。这是信息论和计算学习理论中的一个重要概念。

MDL 基本原理是对于一组给定的实例数据 D，如果要对其进行保存，为了节省存储空间，一般采用某种模型对其进行编码压缩，然后再保存压缩后的数据。同时，为了以后正确恢复这些实例数据，将所用的模型也保存起来。所以需要保存的数据长度(比特数) 等于这些实例数据进行编码压缩后的长度加上保存模型所需的数据长度，将该数据长度称为总描述长度。最小描述长度(MDL) 准则就是要求选择总描述长度最小的模型。

如果将贝叶斯网络作为对实例数据进行压缩编码的模型， MDL 准则就可以用于贝叶斯网络的学习该度量被视为网络结构的描述长度和在给定结构下样本数据集的描述长度之和。一方面，用于描述网络结构的编码位随模型复杂度的增加而增加；另一方面，对数据集描述的编码位随模型复杂度的增加而下降。因此，贝叶斯网络的 MDL 总是力求在模型精度和模型复杂度之间找到平衡。模型 MDL 值如下：

$$\text{MDL:}\mathcal{M}_s = \arg\min_{(\mathcal{M}_k:k\in Z_q)}\left\{\frac{N}{2}log\widehat{\sigma_k}^2 + \frac{\xi}{2}\log N\right\}$$

4. MAP 准则

Maximum a posteriori(MAP)最大后验准则是 Petar M djuric 提出的，它是基于贝叶斯推断中的最大后验概率而形成的，作者提出的目的是为了解决在 AIC 准则和 MDL 准则中对参数权重的忽略，该准则对大样本有很高的依赖性。

模型 MAP 的值为：

$$\text{MAP:}\mathcal{M}_s = \arg\max_{(\mathcal{M}_k:k\in Z_q)}\left\{\frac{f(y|\mathcal{M}_k)*p(\mathcal{M}_k)}{f(y)}\right\}$$

在模型等概率即$p(\mathcal{M}_k) = 1/q$的假设下，由于$f(y)$与$\mathcal{M}_k$无关，所以 MAP 整理得到（推导过程见[1][4]）：

$$\text{MAP:}\mathcal{M}_s = \arg\max_{(\mathcal{M}_k:k\in Z_q)}\{f(y|\mathcal{M}_k)\} = \arg\max_{(\mathcal{M}_k:k\in Z_q)}\left\{\int_{\psi_k} f(y|\psi,\mathcal{M}_k)f(\psi|\mathcal{M}_k)d\psi\right\}$$

$$= \arg\min_{(\mathcal{M}_k:k\in Z_q)}\left\{-\log f\left(y|\hat{\psi},\mathcal{M}_k\right) + \frac{1}{2}\log|\widehat{\mathcal{H}_k}|\right\}$$

其中$\widehat{\mathcal{H}_k}$是$-\log f\left(y|\hat{\psi},\mathcal{M}_k\right)$对$\psi$的 Hessian 矩阵，$\psi$是参数空间。

在以高斯函数为核函数用 RBF 网络学习的过程中，$\mathcal{H}_k$就是 D 矩阵（图片 6 RBF 网络）。因此 MAP 整理得到：

$$\text{MAP:}\mathcal{M}_s = \arg\min_{(\mathcal{M}_k:k\in Z_q)}\left\{\frac{N}{2}log\widehat{\sigma_k}^2 + \frac{1}{2}\log|H_k^T H_k|\right\}$$

5. HQC 准则

在统计上，Hannan-Quinn 信息准则（HQC）是模型选择的标准。它是 Akaike 信息准则（AIC）和贝叶斯信息准则（BIC）的替代方案。模型 HQC 的值为：

$$\mathrm{HQC} = -2log\hat{L}^2 + \xi \log\log n$$

其中 L 为最大似然，其余参数同上。整理 HQC 得到：

$$\mathrm{HQC}: \mathcal{M}_s = \arg \min_{(\mathcal{M}_k: k \in \mathcal{Z}_q)} \left\{ \frac{N}{2} log\widehat{\sigma_k}^2 + \frac{\xi}{2} \log\log n \right\}$$

Claeskens 和 Hjort 曾指出，HQC 和 BIC 一样，但与 AIC 的渐近效率不同。然而，由于很小的因素，它忽略了最优估计率。他们进一步指出，无论使用哪种方法来微调标准，在实践中都比在$\log\log n$这个术语中更为重要，因为即使对于非常大的 n，后一个数字也是很小的。然而，与 AIC 不同的是，$\log\log n$这个术语确保了 HQC 的一致性。从迭代对数的规律可以看出，任何一致性强的方法都必须至少忽略一个$\log\log n$因子的效率，所以在这个意义上，HQC 是渐近非常好的表现。Van der Pas 和 Grünwald 证明，在许多情况下，基于修正贝叶斯估计的模型选择，在许多情况下表现为像 HQC 一样渐近，同时保留了贝叶斯方法的优点，如使用先验等。

（二） 实验

1. 实验方法

根据 BIC[1][2]论文中的方法，RBF 函数中心点的选取是通过遍历得到的。由于在本问题中，模型变量有 k+1 个，即 k 个中心点、1 个函数个数，且这四种准则考察的变量只有 k、$\widehat{\sigma_k}^2$和 D 矩阵，所以在我的代码实现中，我通过遍历 k 和多次随机采样选取中心点$\mu_{1:k}$，得到了k_{max}个模型，以此为数据集，应用到 4 个不同的模型选择方法上，并对比实验结果。代码流程如下：

1. Initialization: set iter = 5000, Nmax = 200
2. Iteration:
   for k = 1:k_{max}
      for i =1:iter //Train
      - Generate $\mu_{1:k}$ from uniform distribution
      - Evaluate loss on train set
      - Update the model
      - Evaluate loss on validation set
3. Choose Model by applying different criteria

在具体的实现中，$k_{max} = 200$，迭代次数$\mathrm{iter} = 5000$，以 mse 作为模型更新的判定标准，数据的前 80%作为训练集，后 20%作为验证集。

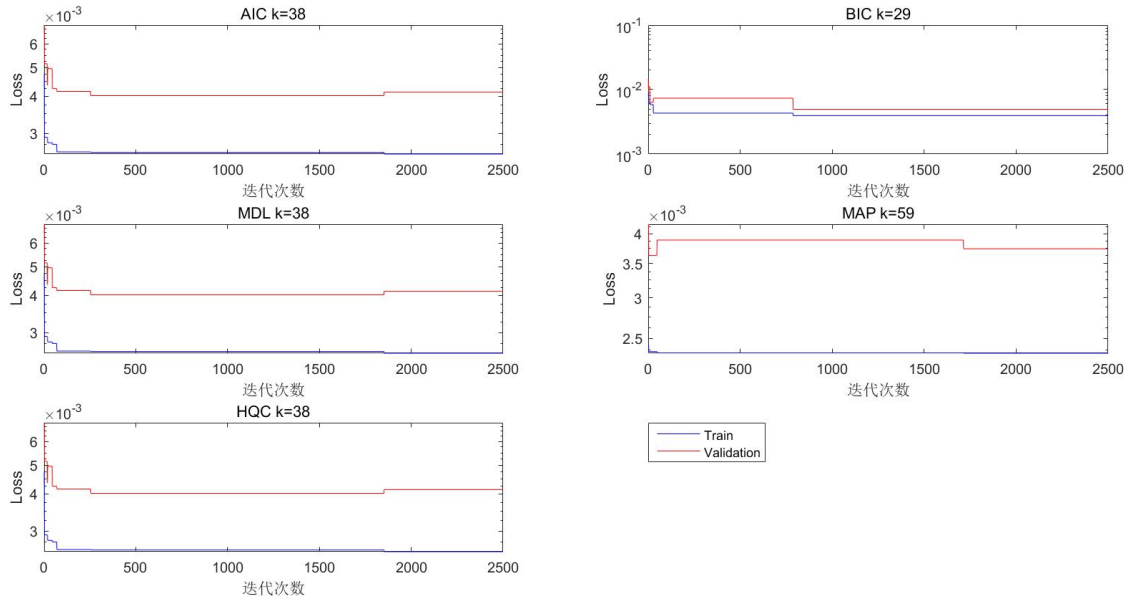
2. 实验结果

表 6 data1 在不同模型选择方法下的结果

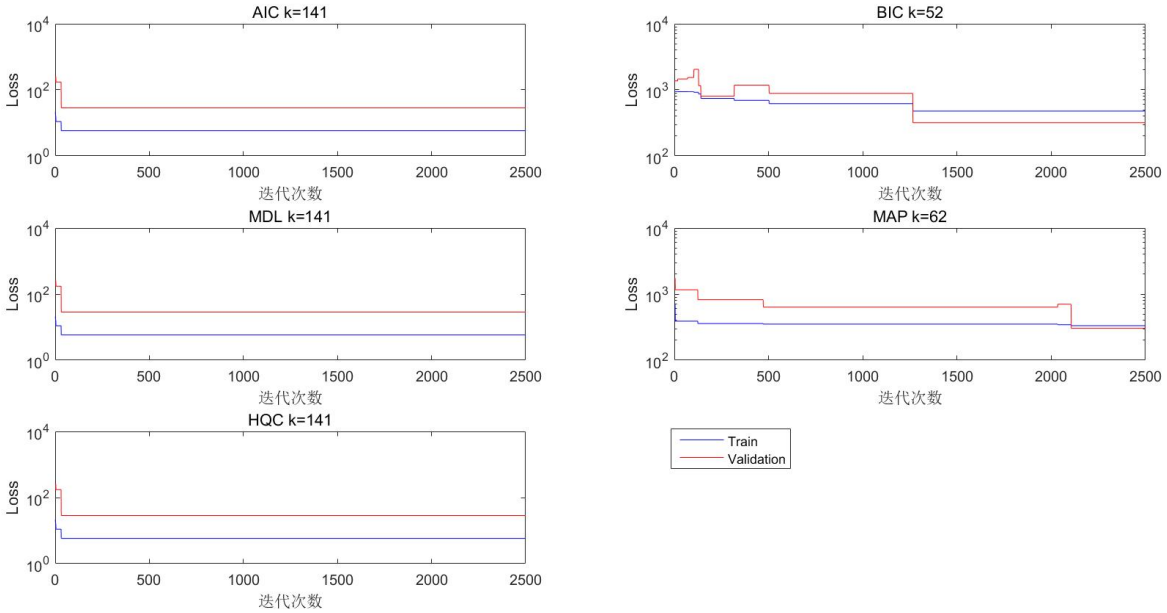
模型选择方法	AIC	BIC	MDL	MAP	HQC	RJMCMC+SA
RBF 函数个数	38	29	38	59	38	34
训练集 Loss	0.0025	0.0039	0.0025	0.0023	0.0025	0.0024
验证集 Loss	0.0037	0.0049	0.0037	0.0036	0.0037	0.0038
收敛速度	300	800	300	100	300	140

表 7 data2 在不同模型选择方法下的结果

模型选择方法	AIC	BIC	MDL	MAP	HQC	RJMCMC+SA
RBF 函数个数	141	52	141	62	141	108
训练集 Loss	5.8100	477.8513	5.8100	323.7846	5.8100	0.7519
验证集 Loss	28.5411	318.5826	28.5411	303.3851	28.5411	3.5642
收敛速度	200	1300	200	2100	200	1300



图片 9 Data1 在不同模型选择准则下的结果



图片 10　Data2 在不同模型选择准则下的结果

3. 结果分析

i.　AIC/BIC/MDL/MAP/HQC~RJMCMC+SA

由 表 6　data1 在不同模型选择方法下的结果 可以看出，在 data1 的拟合问题上，AIC、BIC、MDL、MAP 和 HQC 五种模型选择方法和 RJMCMC+SA 的结果在 Loss 和函数个数 k 上相似，且 Loss 都是 e-03 量级，即拟合结果较好。

由 表 7　data2 在不同模型选择方法下的结果 的结果可以看出，选择结果的整体趋势和 data1 相近。但五种经典的模型选择方法应用到的模型集的核函数中心产生方式与 RJMCMC+SA 的不同，前者是在 x_{min}~x_{max} 中均匀随机产生，后者则是在 x_{min}-3~x_{max}+3 中均匀随机产生，考虑到在 RJMCMC+SA 的实验中，两种不同的中心产生方式对结果有一定影响，认为这个因素不可忽略。受迭代次数、核函数中心选取方式不同的影响，AIC、BIC、MDL、MAP 和 HQC 五种模型选择方法的 Loss 都比 RJMCMC+SA 的 Loss 大，其中 BIC 和 MAP 的 Loss 较大。

原本我设想的收敛速度应有明显差别，即 RJMCMC+SA 的收敛速度快于其他模型，但从结果中无法得出收敛速度有明显差异的结论，分析原因可能是模型采样次数只有 1 次、迭代次数较少、中心点生成方法不同等因素的影响，并不能直接从结果中得到算法效率的相关结论。

ii.　AIC/BIC/MDL/MAP/HQC

在本次实验的两个数据集上的结果中，AIC、MDL 和 HQC 得到的结果是一样的，而 BIC、HQC 则都不同。从$\mathcal{M}_s$上来看，AIC、MDL 和 HQC 的$\mathcal{M}_s$中的似然项相同、惩罚项相似，惩罚项的系数最多只差了$\log N \approx 7$倍，而另外两个准则中，BIC 有一个$\mathrm{k}\log k$项，MAP 有一个$\log|H_k^T H_k|$项，相差较大。因此在这个问题下这三种选择方法挑选出来的模型结果相同是可以接受的。

在两个结果中，BIC 的 k 都是最小的，其次是 MAP，再然后是 AIC、MDL 和 HQC。这说明这些准则惩罚力度不同，BIC 最严格，惩罚项与 k 成线性关系的 AIC、MDL 和 HQC 惩罚力度较小，而 MAP 中对于不同的 observation matrix

的惩罚力度不同，不能一概而论。

五、总结

MCMC 在数值计算问题中有非常重要的应用，其中 RJMCMC 可以视为 MCMC 方法的一个新开端，它为贝叶斯模型选择提供了强大的工具。在深度学习领域非常经典的 RBF 模型利用 RJMCMC 方法，可以有效提高学习效率。针对二维高斯的相关系数估计及不同模型选择方法结合 RBF 网络的学习结果，本文主要得出以下结论：

- MCMC 方法在对二维高斯相关系数进行估计的时候，对于给定的二维高斯分布，在误差允许范围内准确估计出相关系数。增加抽样次数可以使估计值更加接近真实值。在建议分布的均值与给定分布的均值相同的前提下，建议分布$f(x)$为固定中心点的高斯分布时会比变化中心点的高斯分布时有更高的拒绝概率，说明在迭代过程中能更快地选取到概率较高的样本点。采用 50000 次抽样，采用 Gibbs 采样比 MH 采样更接近真实值，且收敛速度更快，说明在高维问题中，Gibbs 采样往往是比 MH 采样效率更高的方法。
- 对比用 RJMCMC+SA 和遍历两种方法进行 RBF 网络学习的效果，RJMCMC+SA 在时间复杂度上远胜于遍历，且 Loss 优于遍历得到的结果，这与中心点产生的方法和迭代次数都有关系。
- 由于计算资源和时间的限制，在进行模型选择方法对比时，事先跑出来的模型集来不及调参，所以结果较为粗糙，可能这也是选择结果与 RJMCMC+SA 的选择结果相比误差较大的原因，不过这也说明了 RJMCMC+SA 的效率较高。
- 对比 AIC\BIC\MDL\MAP\HQC 五种模型选择方法，这五种模型选择方法都是从最大似然和模型复杂度的权衡角度出发，在选择是加入了衡量模型复杂度的惩罚项，其中引入了$\mathrm{k}\log k$的 BIC 的惩罚项力度最大，相应的选出来的模型的复杂度较低，Loss 也较大，而惩罚项是 k 的线性函数的 AIC/MDL/HQC 的惩罚力度较小，所以模型复杂度较高，Loss 较低，而 MAP 准则对于不同的模型参数的惩罚力度不同，可以认为 MAP 准则更智能地权衡了最大似然和模型复杂度之间的关系。
- 在本实验中应用的模型选择方法都是基于极大似然和模型复杂度的关系角度，由于时间有限，未能尝试其他角度的模型选择方法。

致谢

衷心感谢欧志坚老师在随机过程这门课上对我的指导，让我收获了非常多的知识。感谢生医系的廖筑秀学长借我服务器资源，帮助我在较短时间内跑出结果；感谢电子系的马可、孙铭和赵雅娟同学，他们的讲解大大加深了我对 RJMCMC 的理解。

感谢罗姆楼深夜亮起的灯和永不断电的电源。

参考文献

[1] H. Akaike, “A new look at the statistical model identification,” *IEEE Transactions on Automatic Control*, vol. 19, no. 6, pp. 716–723, 1974.

[2] G. Schwarz, “Estimating the dimension of a model,” *Annals of Statistics*, vol. 6, no. 2, pp. págs. 15–18, 1978.

[3] J. Rissanen, “Stochastic complexity,” *Journal of the Royal Statistical Society*, vol. 49, no. 3, pp. 223–239, 1987.

[4] P. M. Djuric, “Asymptotic map criteria for model selection,” *IEEE Transactions on Signal Processing*, vol. 46, no. 10, pp. 2726–2735, 2002.

[5] P. J. Green, “Reversible jump markov chain monte carlo computation and bayesian model determination,” *Biometrika*, vol. 82, no. 4, pp. 711–732, 1995.

[6] C. Andrieu, N. D. Freitas, and A. Doucet, “Reversible jump mcmc simulated annealing for neural networks,” in *Sixteenth Conference on Uncertainty in Artificial Intelligenc*e, pp. 11–18, 2000.

[7] Michael J. D. Powell *(1977).* "Restart procedures for the conjugate gradient method" *(PDF).* Mathematical Programming. *Springer. 12 (1): 241–254.* doi:10.1007/bf01593790.

[8] Hannan, E. J., and B. G. Quinn (1979), "The Determination of the order of an autoregression", Journal of the Royal Statistical Society, Series B, 41: 190–195.

附录 A

正态分布（*Normal distribution*）又名高斯分布（*Gaussian distribution*），是一个在数学、物理及工程等领域都非常重要的概率分布，在统计学的许多方面有着重大的影响力。

若随机变量X服从一个数学期望为μ、标准方差为σ^2的高斯分布，记为：$X\sim N(\mu\ ,\sigma^2)$,则其概率密度函数为

$$f(x)=\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

累积分布函数是指随机变量X小于或等于x的概率，一维高斯分布函数用密度函数表示为：

$$F(x;\mu,\sigma)=\frac{1}{\sigma\sqrt{2\pi}}\int_{-\infty}^{x}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)dx.$$

若随机变量X是二维分布，则其概率密度表示为：

$$f(x,y)=\frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}}e^{\left\{\frac{-1}{2(1-\rho^2)}\left[\frac{(x-\mu_1)2}{\sigma_1^2}-2\rho\frac{(x-\mu_1)(y-\mu_2)}{\sigma_1\sigma_2}+\frac{(y-\mu_2)2}{\sigma_2^2}\right]\right\}}$$

其中$\mu_1\ ,\mu_2\ ,\sigma_1\ ,\sigma_2\ ,\rho$ 都是常数，且$\sigma_1>0\ ,\sigma_2>0\ ,-1<\ \rho<1$ 我们称$(x\ ,y)$服从参数为 $\mu_1\ ,\mu_2\ ,\sigma_1\ ,\sigma_2\ ,\rho$ 的二维正态分布，记为$(x\ ,y)\sim N(\mu_1\ ,\mu_2\ ,{\sigma_1}^2\ ,{\sigma_2}^2\ ,\rho)$.

用期望值向量μ和协方差矩阵Σ表示，其中

$$\mu=\begin{pmatrix}\mu_1\\ \mu_2\end{pmatrix}\ ,\Sigma=\begin{pmatrix}\sigma(x\ ,x) & \sigma(x\ ,y)\\ \sigma(y\ ,x) & \sigma(y\ ,y)\end{pmatrix}=\begin{pmatrix}{\sigma_1}^2 & \rho\sigma_1\sigma_2\\ \rho\sigma_2\sigma_1 & {\sigma_2}^2\end{pmatrix}$$

记为$X=\begin{pmatrix}x_1\\ x_2\end{pmatrix}\sim N\left\{\begin{pmatrix}\mu_1\\ \mu_2\end{pmatrix}\ ,\ \begin{pmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{pmatrix}\right\}$,故可立得变量间相关系

$$\rho=\sqrt{\frac{\Sigma_{12}\times\Sigma_{21}}{\Sigma_{11}\times\Sigma_{22}}}$$

附录 B

对于一条马氏链，其可能达到一个平稳分布p^*，需满足$p^* = p^*F$；也就是说平稳分布的条件是马氏链是不可约非周期的。当马氏链是周期的它会在状态之间以一个确定的方式循环。是平稳分布的一个充分条件是下面的细致平衡条件成立：

$$F(j,k)p_j{}^* = \sum_{i,j} F(k,j)p_k{}^* \qquad (*)$$

(*)式被称为细致平衡条件。要证明通过 MH 方法构造出来的链满足马氏性且以$p(x)$为平稳分布，只需要证明 Metroplis-Hastings 方法的转移核满足细致平衡条件即可。首先，*MH*方法构造出来的链满足马氏性是显然的，因为$X^{(t)}$的产生仅依赖于$X^{(t-1)}$。其次，对于满足细致平衡方程的证明，不失一般性，以二元函数为例说明，分三种情况：

1、$p(x)f(x\ ,y) = p(y)f(y\ ,x)$，则由(2)式可得$\alpha(x\ ,y) = \alpha(y\ ,x) = 1$；

又∵ $p(x\ ,y)$定义为$p(x \to y) = \alpha(x\ ,y)f(x\ ,y)$，加上假设条件$f(x\ ,y)\ p(x) = f(y\ ,x)\ p(y)$，立得$f(x\ ,y)p(x) = f(y\ ,x)p(y)$

(*)式细致平衡条件满足；

2、$p(x)f(x\ ,y) > p(y)f(y\ ,x)$，则由(2)式可得$\alpha(x\ ,y) = \frac{p(y)f(y,x)}{p(x)f(x,y)}$，$\alpha(y\ ,x) = 1$

∴ $f(x\ ,y)\ p(x) = f(x\ ,y)\alpha(x\ ,y)\ p(x) = f(x\ ,y)\frac{p(y)f(y,x)}{p(x)f(x,y)}\ p(x) = f(y\ ,x)\ p(y)$，

(*)式细致平衡条件满足；

3、$p(x)f(x\ ,y) < p(y)f(y\ ,x)$，则由(2)式可得$\alpha(x\ ,y) = 1$，$\alpha(y\ ,x) = \frac{p(x)f(x,y)}{p(y)f(y,x)}$

∴$f(y\ ,x)p(y) = f(y\ ,x)\alpha(y\ ,x)\ p(y) = f(y\ ,x)\frac{p(x)f(x,y)}{p(y)f(y,x)}\ p(y) = f(x\ ,y)\ p(x)$,

(*)式细致平衡条件满足，证毕。

附录 C

matlab 文件清单

1. MH、Gibbs

MH 算法估计二维高斯相关系数	MH.m
Gibbs 采样估计二维高斯相关系数	Gibbs.m

2. RJMCMC 实验设计

RJMCMC 实验运行函数	rjmcmc_exp.m
birth 函数	birth_exp.m
death 函数	death_exp.m
update 函数	update_exp.m

3. RJMCMC+SA

RJMCMC+SA 的 RBF 网络运行函数	RJMCMC_SA.m
birth 函数	birth.m
death 函数	death.m
merge 函数	merge.m
split 函数	split.m
update 函数	update1.m

4. AIC\BIC\MDL\MAP\HQC

遍历求解模型参数运行函数	Model1.m
产生中心点	update_mu.m
用不同准则选择模型	ChooseModel.m