交通大数据

超参数优化

- 刘志远教授
- zhiyuanl@seu.edu.cn

超参数优化

❑ 学习目标

- 了解机器学习中超参数的特点
- 掌握超参数调节技巧

超参数优化

❑ 超参数

考虑我们学习过的一些机器学习模型中的参数：

A类参数	B类参数
支持向量机中核函数的参数	线性回归的截距
决策树叶节点的最少数据量	决策树分叉位置
深度学习的学习率	支持向量机中分类界面的系数
神经网络的层数	

- A类参数通常在模型训练前由人为指定；
- 不同于B类参数，A类参数的值无法从训练数据中估计得到，而是提前给定，这类参数被称为**超参数**
- 对于每一组给定的超参数，需要利用数据训练得到具体的参数的最优值

超参数优化

❑ 超参数

超参数是影响模型性能的一个重要维度。对于一个复杂的模型，其往往包含多个超参数，各个超参数的取值及其组合是一个非常复杂的难题（这主要是因为一组超参数的效果进行评估需要进行一次完整的模型训练）。

目前常用的超参数调整方法有

（1）网格搜索（grid search）

（2）随机搜索（random search）

（3）贝叶斯优化（Bayesian Optimization）

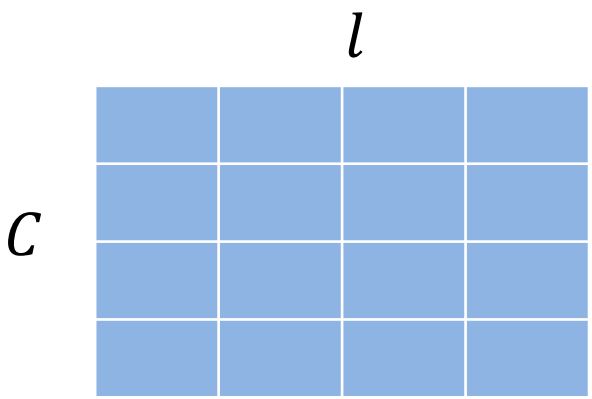
超参数优化

❑ 网格搜索（grid search）

SVM模型存在两个超参数：

- RBF核函数带宽（length scale）：l
- 软间隔超参数：c

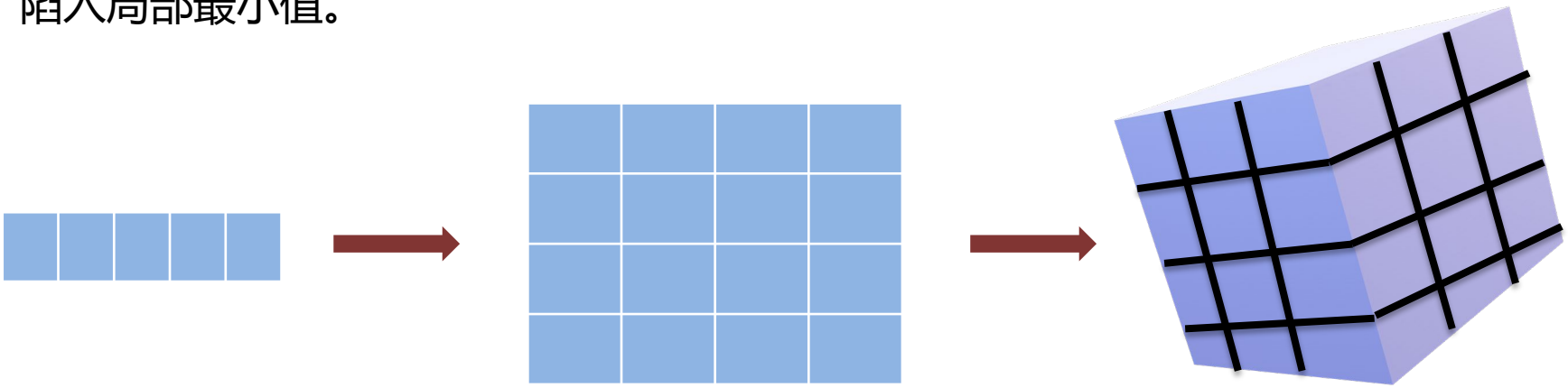
顾名思义，网格搜索是像网格一样用有限个取值来近似所有超参数的取值空间，之后逐一计算“网格”中的每个超参数组合，最终通过比较来确定最优解。这里的网格是通过一定的步长（间距）来计算而得的，这个步长和超参数的备选取值是提前人为确定的。



超参数优化

❑ 网格搜索（grid search）

网格搜索需要枚举各个维度，因此当超参数变多之后，潜在的组合方案（即网格数）会非常多，容易陷入“维数灾难”（curse of dimensionality）导致计算时间过长。在实际操作中，如果需要调参的超参数比较多，一般都会固定多数参数，分步对 1~2 个超参数进行调优，这样能够减少计算时间，但是缺点是难以自动化进行，而且容易陷入局部最小值。



一维　二维　三维

超参数优化

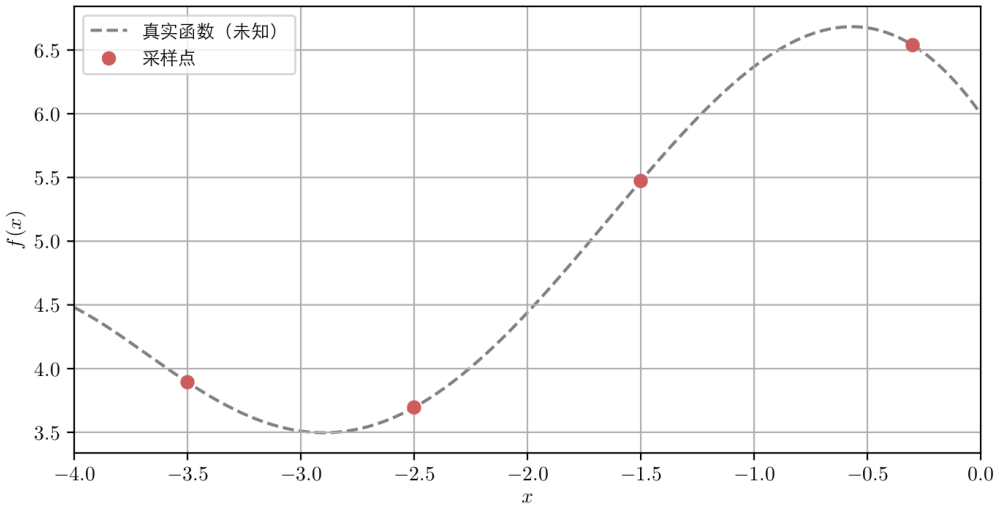
❑ 随机搜索（random search）

在网格搜索的基础上，随机搜索不采用穷举的计算形式，而是在可能取值中随机选择参数方案， 除了网格化的离散取值集合，随机搜索还可以直接从连续空间中随机取值。随机搜索的抽样方法可以根据参数的分布来合理选定。

与网格搜索相比，随机搜索的计算量可以得到有效控制，但更加无法保障所求得的解的质量。网格搜索与随机搜索都有比较好的并行计算潜质（embarrassingly parallel），是非常适合在多线程的服务器平台上处理的任务。

超参数优化

❑ 贝叶斯优化



将调超参问题作以下数学定义：

用x表示参数方案，其取值空间为χ，以$y = f(x)$表示模型表现（通常以模型训练完之后的损失函数最优值来表示）。我们试图找到一组最优的参数方案，这套方案使$f(x)$取最优值，若以损失函数值表征模型表现，那么该问题可以表述为：

$$\min_{x \in \chi} f(x)$$

对于一组参数x_i，我们需要基于x_i将模型完整训练一遍才能得到对应于x_i的模型表现。然而这种训练往往是耗时费力的，显然不可能通过枚举法试验全部参数方案找到最优方案，我们可以用贝叶斯优化来解决这一问题

超参数优化

❑ 贝叶斯优化（Bayesian Optimization）

考虑到$f(x)$非常复杂，难以写出准确表达式或求得导数，将$f(x)$视作黑箱，仅仅关注其输入和输出。基于已有的n组参数$[x_1, x_2, \dots, x_n]$(可以随机生成)，进行模型训练，获得各组参数对应的目标函数值$[y_1, y_2, \dots, y_n]$，贝叶斯优化的思想可以由两个关键步骤来阐述：

1. 基于已观测到的n个点的输入和输出信息，构造原函数$f(x)$的逼近函数$\overline{f}(x)$
2. 基于$\overline{f}(x)$，利用一套规则求潜在的最优参数值x'，训练模型计算$f(x')$，更新已知点的信息和逼近函数

通过上述两个步骤的迭代使$\overline{f}(x)$不断逼近 $f(x)$，最终利用逼近函数求最优化问题

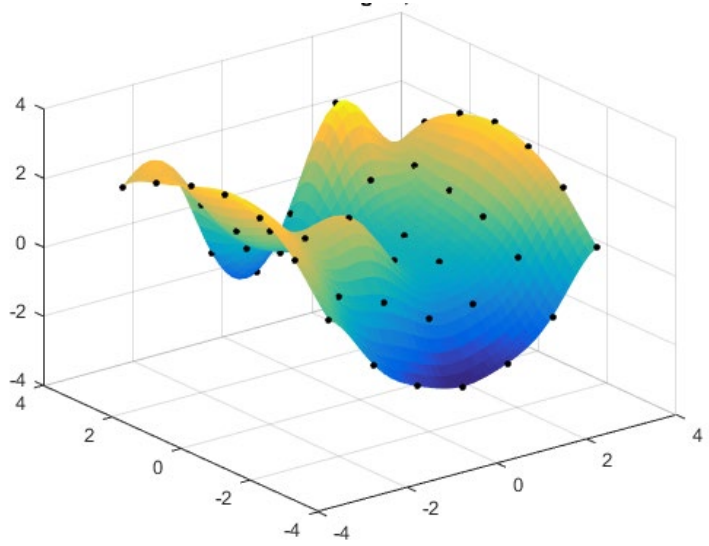
这种优化思想实际上就是**代理模型**（surrogate model 或 Metamodel）求解优化问题的的思想，这里对代理模型进行简要的介绍

代理模型（Surrogate model/Metamodel）

在实际工程中求解一些优化问题所需的计算量与成本是极高的，例如飞行器设计问题、车辆碰撞优化等问题，飞行器与车辆参数需要结合实验结果进行确定，一方面，实验环境复杂，参数优化问题高度非线性；另一方面，实验的金钱成本与时间成本非常高，对全部可行的参数进行遍历实验是难以接受的。代理模型（surrogate model）常被应用于这一类优化问题。

代理模型（Surrogate model/Metamodel）

代理模型实际上是对复杂目标函数的近似表达（approximation），在优化过程中借助该近似模型实现对目标函数的优化



以已知点拟合曲面

A metamodel (or surrogate model) is an analytical approximation of the objective function. Metamodel optimization methods iterate over two main steps. First, the metamodel is fitted based on a set of simulated observations. Second, it is used to perform optimization and derive a trial point (in this paper the term point refers to a given decision vector value x). The performance of the trial point can be evaluated by the simulator, which leads to new observations. As new observations become available the accuracy of the metamodel can be improved (step 1), leading ultimately to better trial points (step 2).

Osorio C, Bierlaire M. A simulation-based optimization framework for urban transportation problems[J]. Operations Research, 2013, 61(6): 1333-1345.

代理模型（Surrogate model/Metamodel）

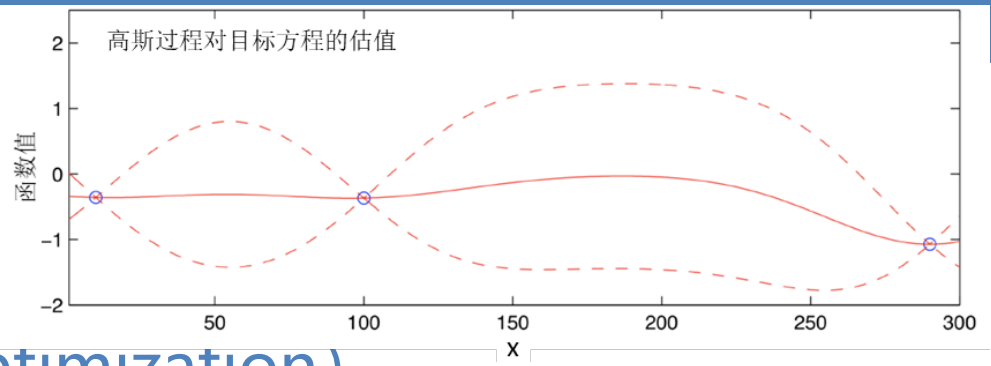
根据代理模型是否依据具体问题进行构建，代理模型可以分为Physical metamodel 与 Functional metamodel

Physical Metamodel：与具体问题相结合的代理模型，一般是对复杂问题的简化表达，例如以静态交通分配模型作为动态仿真模型的代理模型

Functional Metamodel：不与具体问题结合的代理模型，可以用于处理各种问题，例如高斯过程模型，多项式模型等

贝叶斯优化方法选用高斯过程（Gaussian process， GP）模型

超参数优化

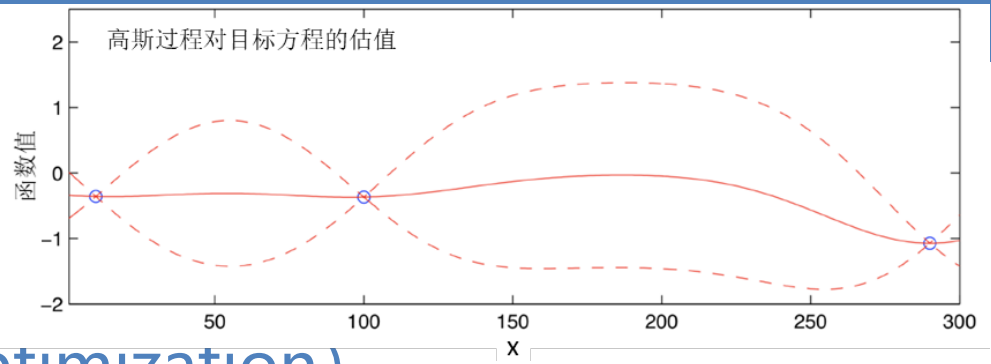


❑ 贝叶斯优化（Bayesian Optimization）

第一步：基于已观测到的n个点的输入和输出信息，构造逼近函数$\bar{f}(x)$

$f(x)$在可行域内的具体取值虽然是未知的，但是已有的n个数据点为我们了解$f(x)$提供了重要依据。在$[x_1, x_2, \ldots, x_n]$处，我们已精确地知道了$f(x)$的取值，根据这n个点，可以构造一个平滑的曲线$\bar{f}(x)$来近似$f(x)$。进一步，我们可以**假设**：可行域内离这n个点$[x_1, x_2, \ldots, x_n]$距离近的点，其函数值我们也“更有把握”知道，即$\bar{f}(x)$的值更准。如图 1 所示，实线是$\bar{f}(x)$曲线，三个圆圈是目前已经检测过的三个值$[x_1, x_2, x_3]$；虚线是我们对于$f(x)$的估值下限和上限，代表了对$f(x)$的估值误差，可见，离检测过的点$[x_1, x_2, x_3]$越远，估值误差就越大。这是贝叶斯优化方法中最重要的一个假设。↵

超参数优化



❑ 贝叶斯优化（Bayesian Optimization）

第一步：基于已观测到的n个点的输入和输出信息，构造逼近函数$\bar{f}(x)$

为了方便讲解，我们称上面的假设为**假设 1**。为满足这种假设，通常情况下[2]，我们可以构造一种工具，以满足：两个样本i, j间的距离$\| x_i - x_j \|^2$越近，函数值$\bar{f}(x)$相差越小。在高斯过程中，满足这一要求的工具是**核函数**。↵

最常用的一个核函数为**高斯核函数**，也称为径向基函数（Radial Basis Function, RBF），其基本形式如下，其中σ和l是高斯核的超参数。↵

$$\kappa(x_i, x_j) = \sigma_k^2 \exp\left(-\frac{\| x_i - x_j \|_2^2}{2l^2}\right) \qquad (1)$$

由公式（1）可见，x_i和x_j的距离越近，核函数的取值越大；x_i和x_j的距离越远，核函数的取值越接近 0。因此，可以利用核函数来定义x_i和x_j的相关性，相关性越高，则二者的函数值$\bar{f}(x)$越接近。↵

超参数优化

❑ 贝叶斯优化（Bayesian Optimization）

第一步：基于已观测到的n个点的输入和输出信息，构造逼近函数$\overline{f}(x)$

在贝叶斯优化中，进一步选用高斯过程来完刻画各变量之间的关系。高斯过程假设给定的观测点 $y = [y_1, y_2, \dots, y_n]^T$ 服从n维联合高斯分布，也就是：

$$y \sim N(\boldsymbol{\mu_y}, \boldsymbol{K_{yy}})$$

其中$\boldsymbol{\mu_y} \epsilon \mathbb{R}^{\boldsymbol{n}}$，一般而言我们将$\boldsymbol{\mu_y}$设置为一个零向量，这主要是因为$\boldsymbol{\mu_y}$的值不影响高斯过程建模，$\boldsymbol{\mu_y}$对于最终求最优值的问题也不会产生任何影响。$\boldsymbol{K_{yy}}$是协方差矩阵。

$$\boldsymbol{K_{yy}} = \begin{bmatrix} Cov(y_1, y_1) & \cdots & Cov(y_1, y_n) \\ \vdots & \ddots & \vdots \\ Cov(y_n, y_1) & \cdots & Cov(y_n, y_n) \end{bmatrix}$$

超参数优化

❑ 贝叶斯优化（Bayesian Optimization）

第一步：基于已观测到的n个点的输入和输出信息，构造逼近函数$\overline{f}(x)$

$$\boldsymbol{K}_{yy} = \begin{bmatrix} Cov(y_1, y_1) & \cdots & Cov(y_1, y_n) \\ \vdots & \ddots & \vdots \\ Cov(y_n, y_1) & \cdots & Cov(y_n, y_n) \end{bmatrix}$$

通过径向基（RBF）核函数对相关性进行度量：

$$Cov(y_i, y_j) = \exp\left(-\frac{\|x_i - x_j\|_2^2}{2l^2}\right)$$

- 采用径向基函数中对相关性进行度量，蕴含了一个假设，即：两个参数在空间中越接近，他们对应的模型表现就越接近，这一假设在处理连续函数时是合理的
- 径向基核函数的参数l可以通过最大似然估计获得

❑ 贝叶斯优化（Bayesian Optimization）

第一步：基于已观测到的n个点的输入和输出信息，构造逼近函数$\overline{f}(x)$

我们记未知观测值点x'（即没有进行模型训练得到模型表现值$f(x')$）处的函数值为y',那么根据高斯过程的假设，$[y_1, y_2, \dots, y_n, y']^T$服从$n+1$维联合高斯分布，根据联合高斯分布，有：

$$\begin{bmatrix} \boldsymbol{y} \\ y' \end{bmatrix} \sim N\left(\begin{bmatrix} \boldsymbol{\mu_y} \\ \boldsymbol{\mu_{y'}} \end{bmatrix}, \begin{bmatrix} \boldsymbol{K_{yy}} & \boldsymbol{K_{yy'}} \\ \boldsymbol{K_{y'y}} & \boldsymbol{K_{y'y'}} \end{bmatrix}\right)$$

根据联合高斯分布公式我们可以整理得到y'的后验分布：

$$y'|x', \boldsymbol{x}, \boldsymbol{y} \sim N(K_{yy'}^T K_{yy}^{-1}(y - \mu_y) + \mu_{y'}, K_{y'y'} - K_{yy'}^T K_{yy}^{-1} K_{yy'})$$

其中μ_y 与$\mu_{y'}$为零向量，$K_{yy'} = \begin{bmatrix} Cov(y_1, y') \\ \vdots \\ Cov(y_n, y') \end{bmatrix}$, $\boldsymbol{K_{yy}} = \begin{bmatrix} Cov(y_1, y_1) & \cdots & Cov(y_1, y_n) \\ \vdots & \ddots & \vdots \\ Cov(y_n, y_1) & \cdots & Cov(y_n, y_n) \end{bmatrix}$

至此，逼近函数$\overline{\boldsymbol{f}}(\boldsymbol{x})$已经由高斯过程给出

❑ 贝叶斯优化（Bayesian Optimization）

第一步：基于已观测到的n个点的输入和输出信息，构造逼近函数$\overline{f}(x)$

根据联合高斯分布公式我们可以整理得到y'的后验分布：

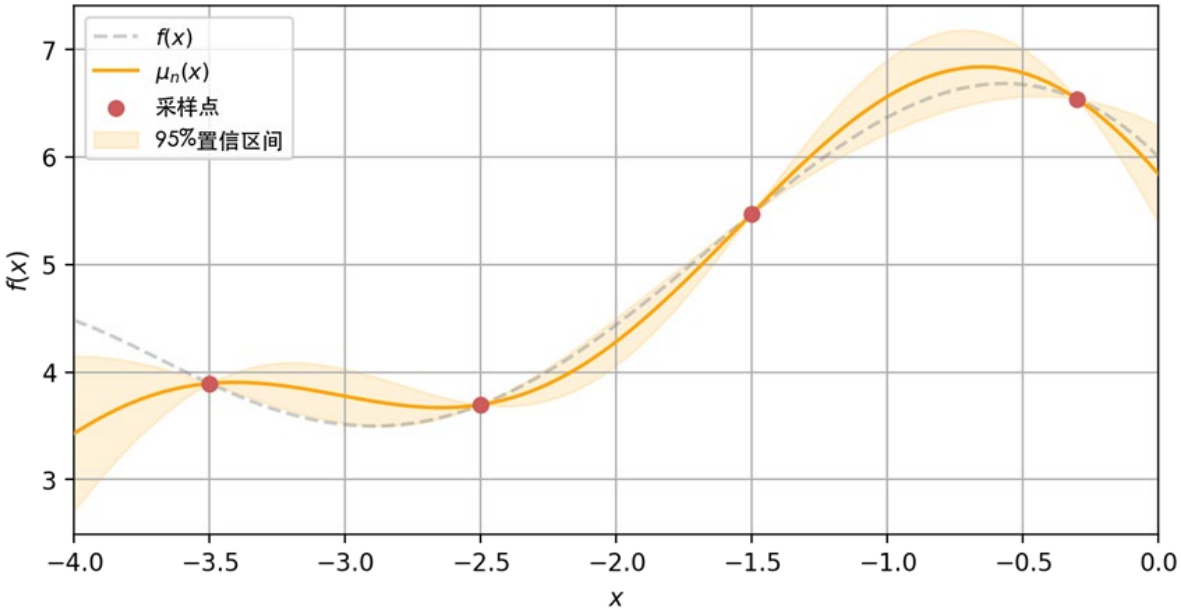
$$y'|x', \boldsymbol{x}, \boldsymbol{y} \sim N(K_{yy'}^T K_{yy}^{-1}(y-\mu_y)+\mu_{y'}, K_{y'y'}-K_{yy'}^T K_{yy}^{-1} K_{yy'})$$

我们可以记：$\mu_n(x')=K_{yy'}^T K_{yy}^{-1}(y-\mu_y)$, $\sigma_n(x')=K_{y'y'}-K_{yy'}^T K_{yy}^{-1} K_{yy'}$那么$\mu_n$表示逼近函数在$x'$处的**估计值**，$\sigma_n$表示**估计误差**

❑ 贝叶斯优化（Bayesian Optimization）

第一步：基于已观测到的n个点的输入和输出信息，构造逼近函数$\overline{f}(x)$

例：假设原函数为$f(x)=0.03x^5+0.2x^4-0.1x^3-2.4x^2-2.5x+6$ 将初始采样点选取为 $\boldsymbol{x}=[x_1,x_2,x_3,x_4]$=$[-3.5,-2.5,-1.5,-0.3]$可以利用高斯过程估计得到如下曲线：



超参数优化

❑ 贝叶斯优化（Bayesian Optimization）

第二步：基于$\bar{f}(x)$，利用一套规则来求潜在的最优参数值x'

（1）最优参数值应该是使得目标函数取极值（最大或最小，根据问题而定）的x值，然而我们并不知道目标函数的形式，无法直接优化，因此我们可以通过逼近函数$\bar{f}(x)$来求最优的x

（2）逼近函数必然与原函数是不同的，我们除了找逼近函数的最优x，还要考虑到逼近函数估计不准的区域是否会有潜在的最优点。也就是说，要综合考虑估计值（exploitation）与估计误差（exploration）

在贝叶斯优化中，我们通过求解采集函数（acquisition function）找到潜在的最优参数值。这里介绍最为常见的两类采集函数：Upper Confidence Bound（UCB）和Expected Improvement

超参数优化

❑ 贝叶斯优化（Bayesian Optimization）

第二步：基于$\bar{f}(x)$，利用一套规则来求潜在的最优参数值x'

➢ **Upper Confidence Bound（UCB）**

UCB采集函数的思想非常直接，将估计值$\mu_n(x')$与估计误差$\sigma_n(x')$进行加权，求：

$$\max_x \mu_n(x) + k\sigma_n(x)$$

一般而言，k取为常数，也可根据迭代次数逐渐减小。由于优化问题非凸，需要用多起点牛顿法或多起点梯度下降法进行求解。

超参数优化

❑ 贝叶斯优化（Bayesian Optimization）

第二步：基于$\bar{f}(x)$，利用一套规则来求潜在的最优参数值x'

➢ **Expected Improvement（EI）**

EI的思想是，找到x_{n+1}，其对应的目标函数值$y_{n+1} = f(x_{n+1})$, 将y_{n+1}加入$y = [y_1, y_2, \dots, y_n]^T$ 后所有 目标函数值 y 中 **最大值的提升量的期望最高**，也就是求解：

$$\max_x E[(f(x) - f_n^*), 0]$$

经过化简，EI采集函数可以被整理为：

$$\max_x (f_n^* - \mu_n(x))\Phi\left(\frac{f_n^* - \mu_n(x)}{\sigma_n(x)}\right) + \sigma_n(x)\phi\left(\frac{f_n^* - \mu_n(x)}{\sigma_n(x)}\right)$$

其中Φ为标准正态分布的累积概率密度函数；ϕ为标准正态分布的概率密度函数

❑ 贝叶斯优化（Bayesian Optimization）

综上所述，贝叶斯优化的算法流程可以总结为以下步骤：

输入：样本集 $D_{1:n}=\{(x_1,y_1),(x_2,y_2),\ldots,(x_n,y_n)\}=(\boldsymbol{x},\boldsymbol{y})$，核函数 $\kappa(x_i,x_j)$，采集函数 $H(x)$

步骤一：基于样本集 $D_{1:n}$ 构建高斯过程，得到后验分布 $N(\mu_n,\sigma_n^2)$，获得原函数的逼近曲线 $\bar{f}(x)$。进一步构建采集函数 $H(x)$

步骤二：最大化采集函数 $H(x)$，得到下一个评估点：

$x_{n+1}=\arg\max_{x'\in X} H(x'\mid D_{1:n})$，此时 $x_{n+1}=\arg\max_{x'\in X}\mu_n(x')+\beta_{n+1}^{1/2}\sigma_n(x')$；

步骤三：评估目标函数值 $y_{n+1}=f(x_{n+1})$；

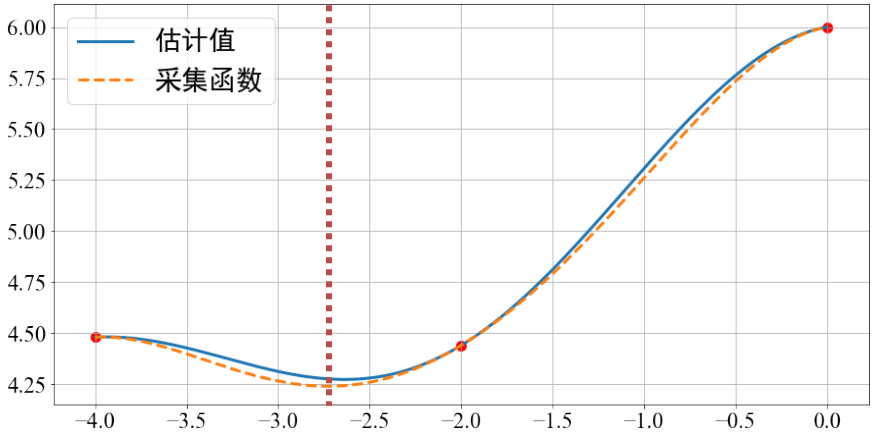
步骤四：更新样本集 $D_{1:n+1}=D_{1:n}\cup\{(x_{n+1},y_{n+1})\}$，并且更新高斯过程模型 $g(x)$，获得新的后验分布 $N(\mu_n,\sigma_n^2)$；

步骤五：令 $n=n+1$，重复步骤二至四。

❑ 贝叶斯优化（Bayesian Optimization）

例：利用贝叶斯优化算法求解$f(x) = 0.03x^5 + 0.2x^4 - 0.1x^3 - 2.4x^2 - 2.5x + 6$在$[-4,0]$内的极小值

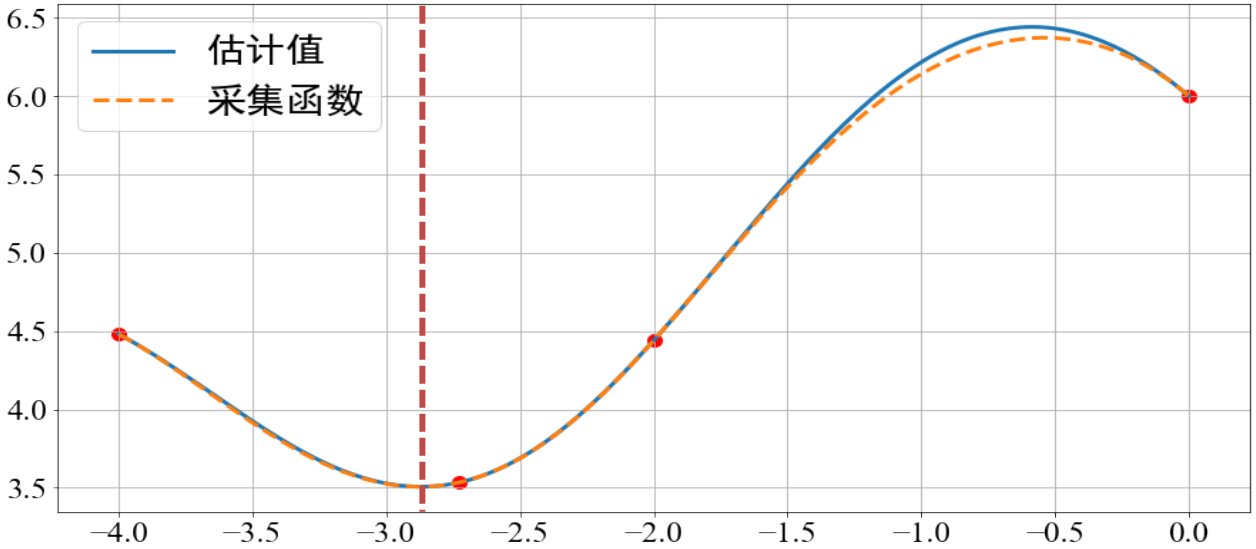
（1）随机选取采样点$\boldsymbol{x} = [-4, -2, -0]$，分别将采样点带入$f(x)$，得到目标函数值$\boldsymbol{y} = [4.48，4.4，6]$，将$\boldsymbol{x}$，$\boldsymbol{y}$作为输入，用高斯过程拟合$\boldsymbol{y} = f(\boldsymbol{x})$，得到如下曲线：



根据高斯过程拟合所得曲线，在$[-4,0]$内搜索采集函数的最小值，得到$x = -2.73$时取得，当$x = -2.73$时，目标函数值为$y = 3.53$，更新采样点集：$\boldsymbol{x} = [-4, -2, -2.73, -0]$, $\boldsymbol{y} = [4.48，4.4，3.53, 6]$

❑ 贝叶斯优化（Bayesian Optimization）

（2）新增采样点后的采样点集： $\boldsymbol{x} = [-4, -2, -2.73, -0]$, $\boldsymbol{y} = [4.48, 4.4, 3.53, 6]$，将$\boldsymbol{x}$，$\boldsymbol{y}$作为输入，用高斯过程拟合$\boldsymbol{y} = f(\boldsymbol{x})$，得到如下曲线：



根据高斯过程拟合所得曲线，在$[-4,0]$内搜索采集函数的最小值，得到$x = -2.8$时取得，当$x = -2.8$时，计算目标函数值，更新采样点集

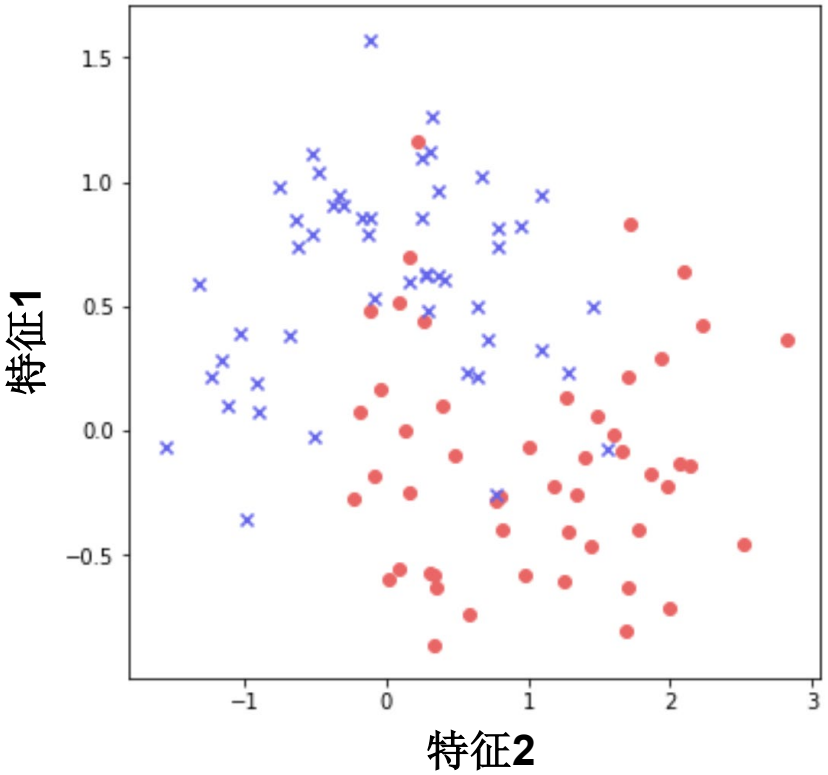
重复上述过程，直到满足算法精度，停止新增采样点

案例分析：调节支持向量机超参数

SVM模型存在两个超参数：

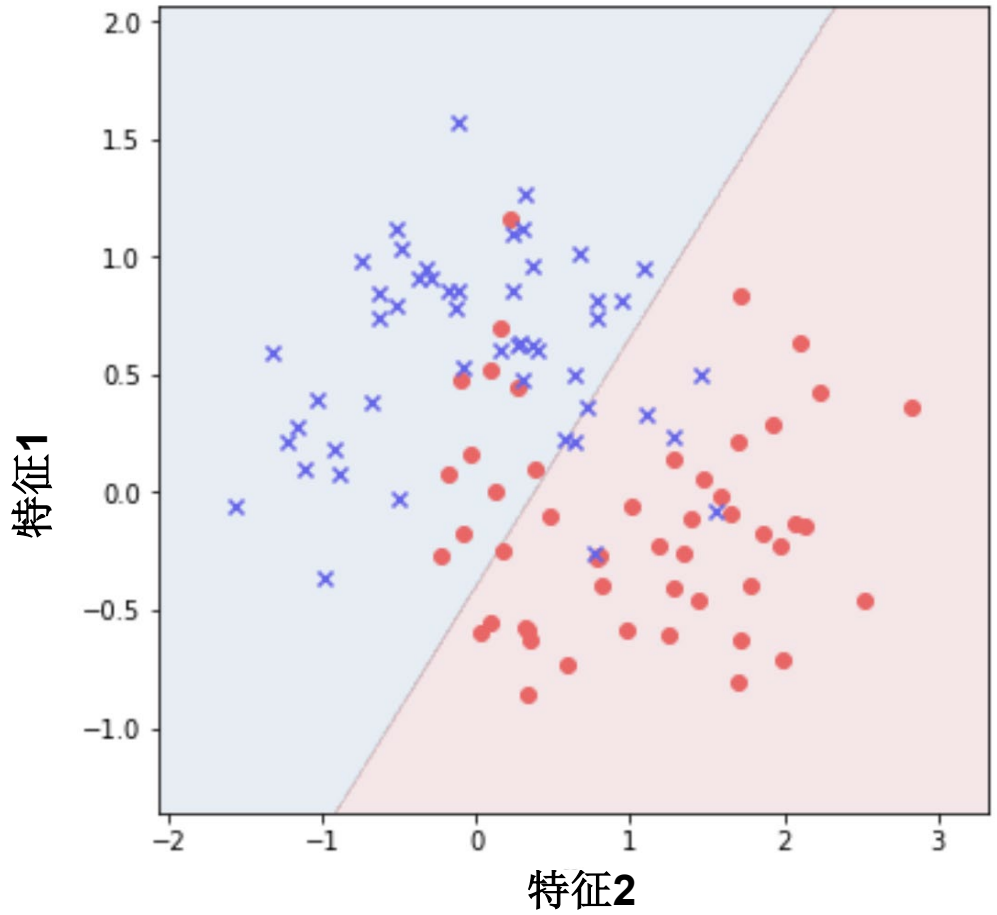
- RBF核函数带宽（length scale）：$\boldsymbol{l}$
- 软间隔超参数：$\boldsymbol{C}$

尝试根据以下二维样本，找到最优SVM超参数组合



案例分析：调节支持向量机超参数

随意指定一组参数，$C = 1$，$l = 0.01$，分类效果如下所示



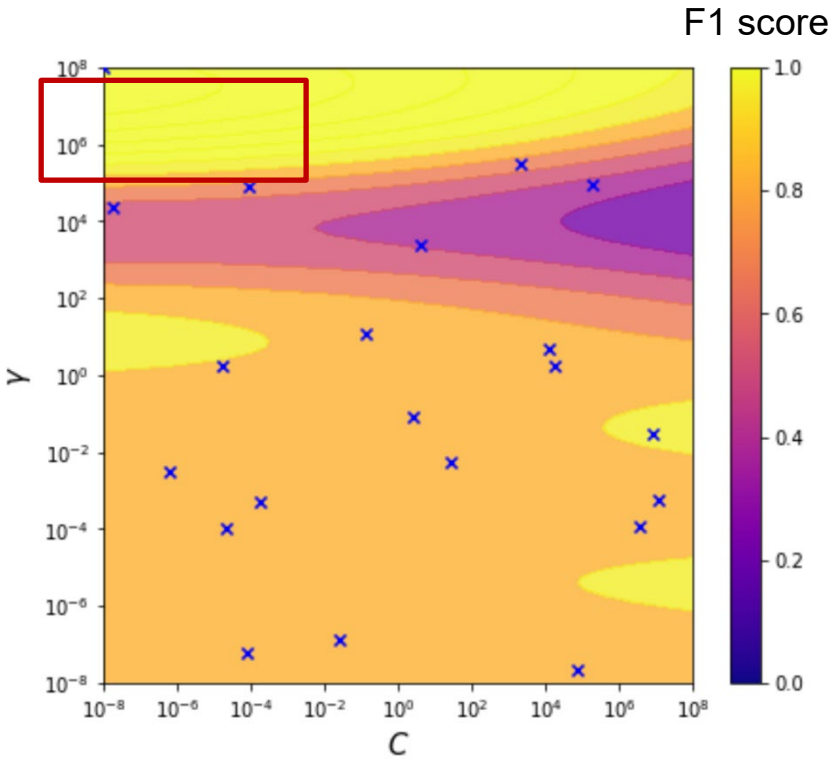
当随意指定参数时，分类效果很不理想，存在很多分类错误的样本

考虑采用贝叶斯优化调节超参数

❑ 案例分析：调节支持向量机超参数

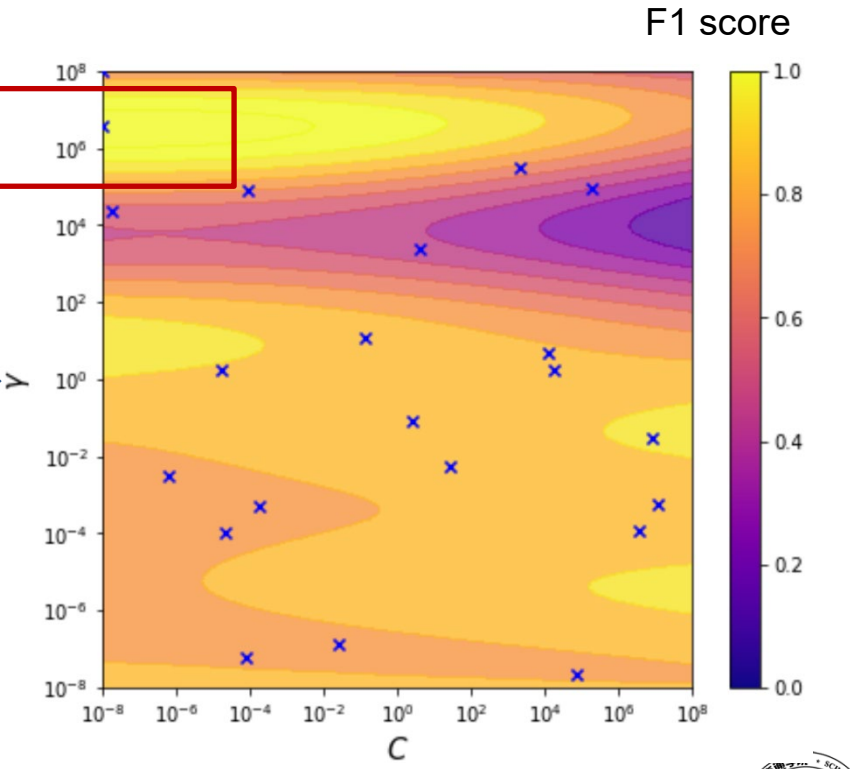
目标函数：n折交叉验证F1分数均值

变量：RBF核函数带宽，软间隔超参数



高斯过程估计曲面（20 个采样点）

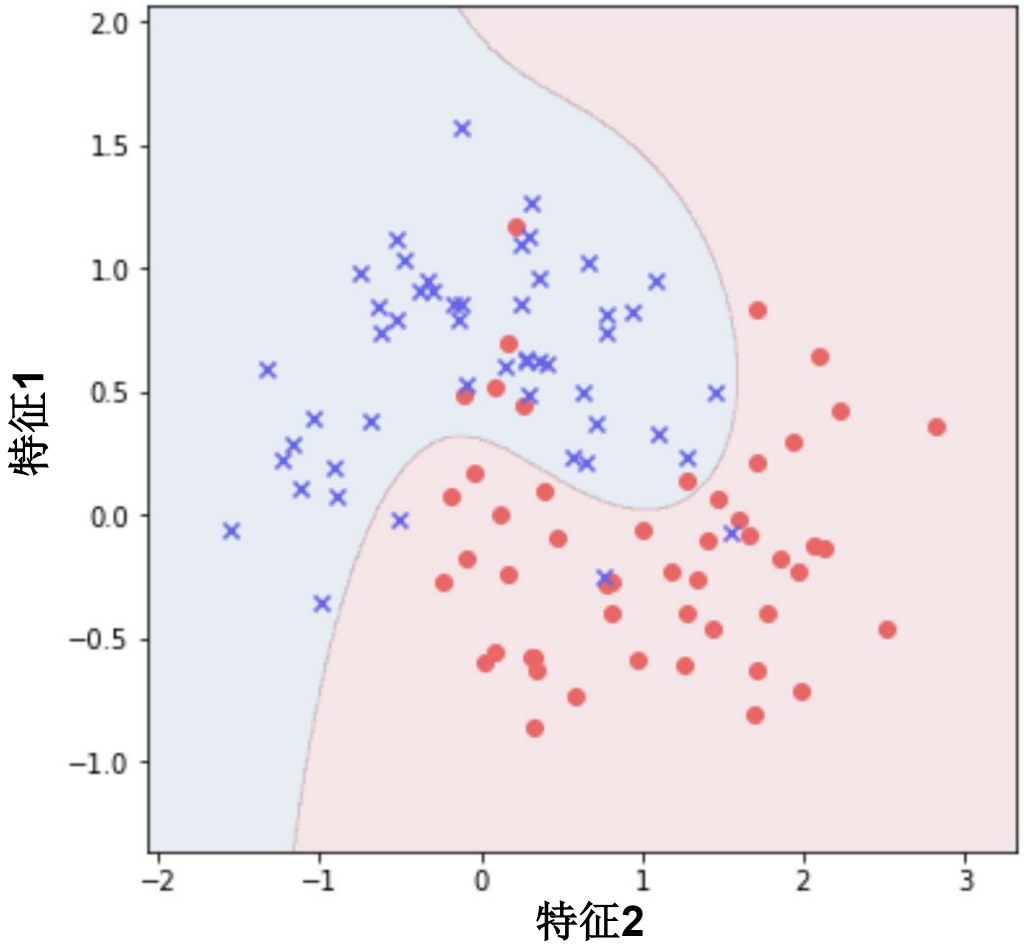
新增一个点



高斯过程估计曲面（21 个采样点）

❑ 案例分析：调节支持向量机超参数

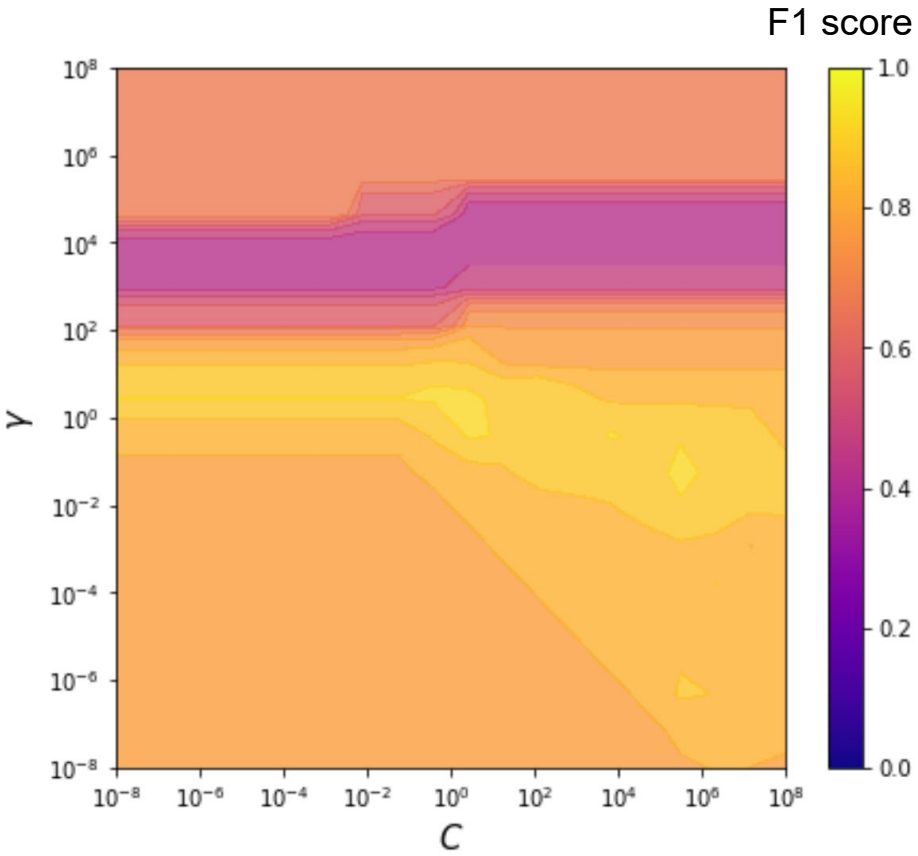
经过50轮迭代，得到最优参数： $C = 1.46 * 10^4$，$l = 2.29 * 10^{-1}$，分类效果如下：



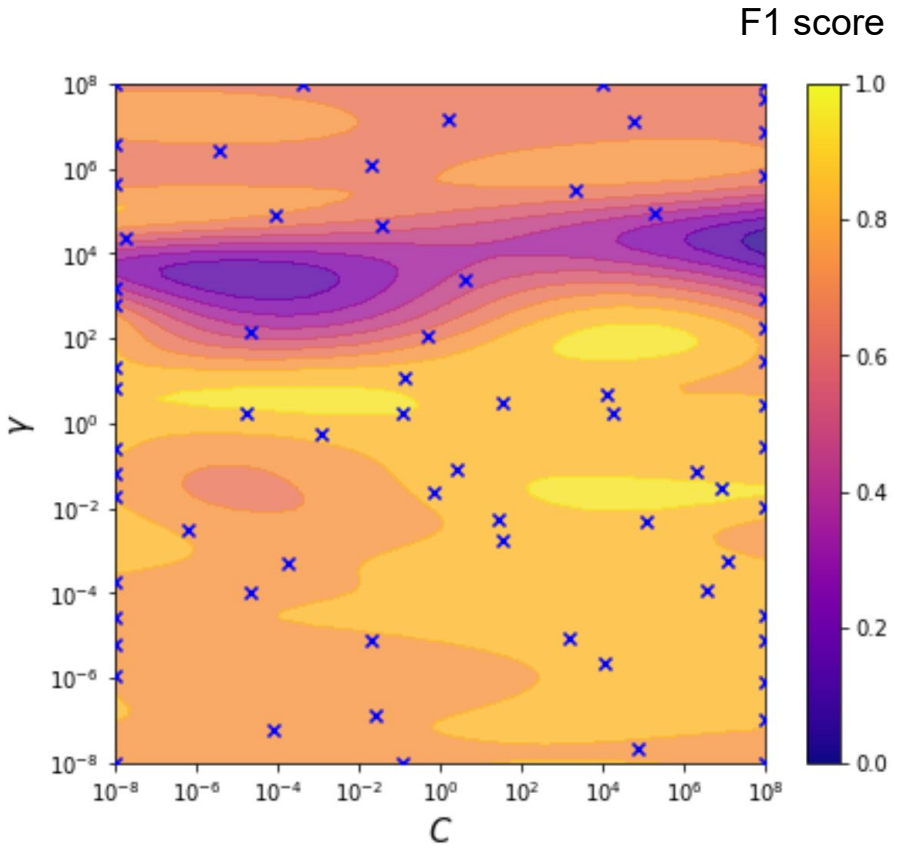
可以看出分类效果有了明显提升

❑ 案例分析：调节支持向量机超参数

以划分网格枚举的方法绘制真实超参数曲面，与高斯过程最终估计曲面对比



真实超参数曲面



由高斯过程估计
所得超参数曲面