

基于 SVM 的非线性函数曲线拟合

孙宁

重庆大学无线通信技术实验室

支持向量机(Support Vector Machine, SVM)是一种常见的判别方法。在机器学习领域,是一个有监督的学习模型,通常用来进行模式识别、分类以及回归分析。它是机构风险最小化算法,就是在训练误差和模型复杂度之间寻求平衡,防止过拟合,从而达到真实误差的最小化。在训练样本足够多的情况下,我们可以利用 SVM 很好地拟合非线性函数。

支持向量机的目的在于求得最优的即几何间隔最大的超平面,在样本数据是线性可分的时候,这里的间隔最大化又叫硬间隔最大化(训练数据近似可分的话就叫软间隔)。关于 SVM 的原理这里不再赘述,主要介绍一下如何在 MATLAB 平台上使用 SVM 做非线性函数的曲线拟合。

- 所用函数介绍:

MATLAB 自带的 SVM 函数只能用于分类,我们在 MATLAB 平台上使用 SVM 进行回归分析时,需要下载一个 libsvm 的工具包,主要使用的函数为 `svmtrain()` 和 `svmpredict()`。

`svmtrain()`实现对训练数据集的训练,获得 SVM 模型。用法:

```
model= svmtrain(train_label, train_data, ['options']);
```

其中, `train_label` 表示训练集的标签, `train_data` 表示训练集的数据, `options` 可选项,根据需求对其进行设置,如果用回归的话,其中的 `-s` 参数值应为 3。

`svmpredict()`是对测试样本进行预测,用法为:

```
[predict_label,accuracy,dec_values]=svmpredict(heart_scale_label,heart_scale_in  
st, model);
```

`svmpredict` 函数返回三个值, `predict_label`, 是训练集预测得到的 label 向量。第二个输出是 `accuracy`, 是一个 3 维的向量,分别是:分类准确率(分类问题中用到的参数指标)、平均平方误差(MSE (mean squared error)) (回归问题中用到的

参数指标)、平方相关系数(squared correlation coefficient) (回归问题中用到的参数指标)。第三个输出是个矩阵,包含着决策值或者是概率估计。当训练数据有 k 类时,决策值矩阵是一个 n 行 $k*(k-1)/2$ 列的矩阵 (n 为测试数据集个数, k 为类别数),而每一行的输出是 $k*(k-1)/2$ 个二分类器的结果。当'-b 1'被指定时,概率估计矩阵是一个 n 行 k 类的矩阵 (n 为测试数据集个数, k 为类别数),每一行的输出是该测试数据属于每个类的概率。

• 仿真分析

我们首先对二维场景进行仿真,当训练样本量分别为 11 个和 101 个时,仿真结果如下图所示:

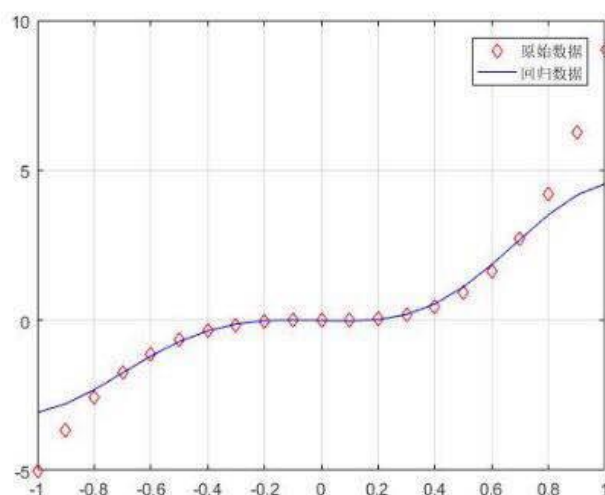


图 1 训练样本为 11 时的二维拟合效果

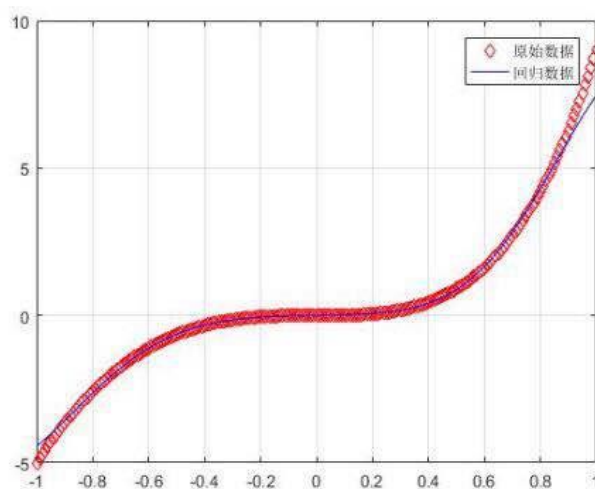


图 2 训练样本为 101 时的二维拟合效果

由上图可以看出，当训练样本过少时，拟合的效果比较差，当增加训练样本的数目时，预测的结果与实际函数图像更接近，拟合效果变好。

当对三维的函数进行预测时，我们对训练集的个数分别设置为 1000 个，5000 个和 10000 个，对比拟合效果，比较误差：

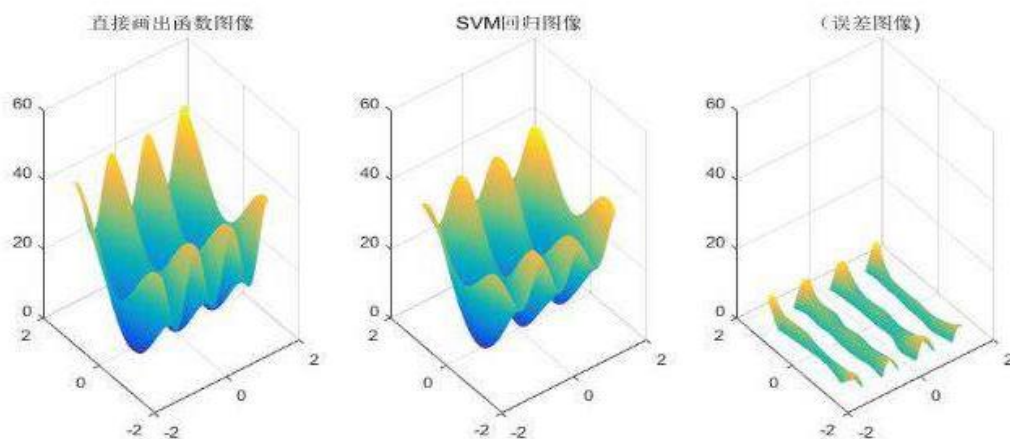


图 3 1000 个训练样本  重庆无线通信技术实验室

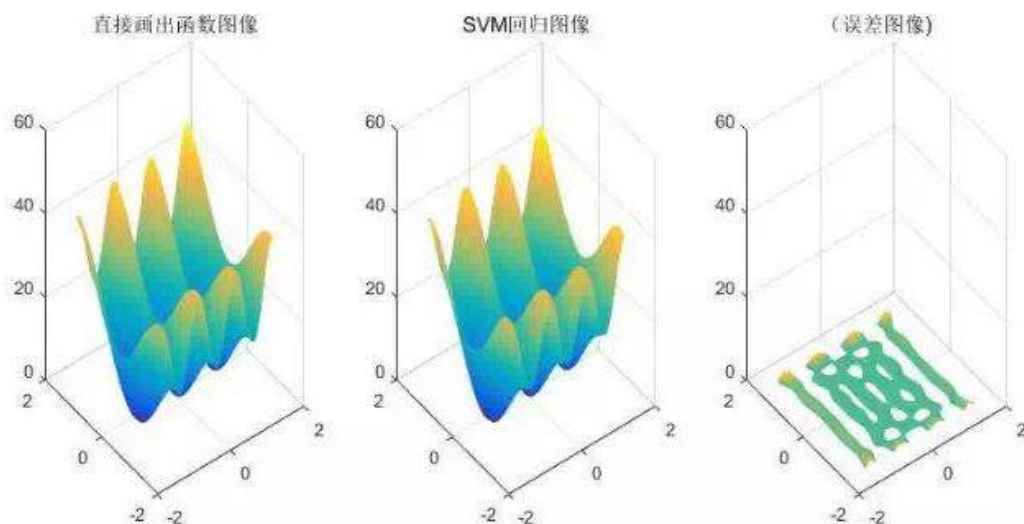


图 4 5000 个训练样本  重庆无线通信技术实验室

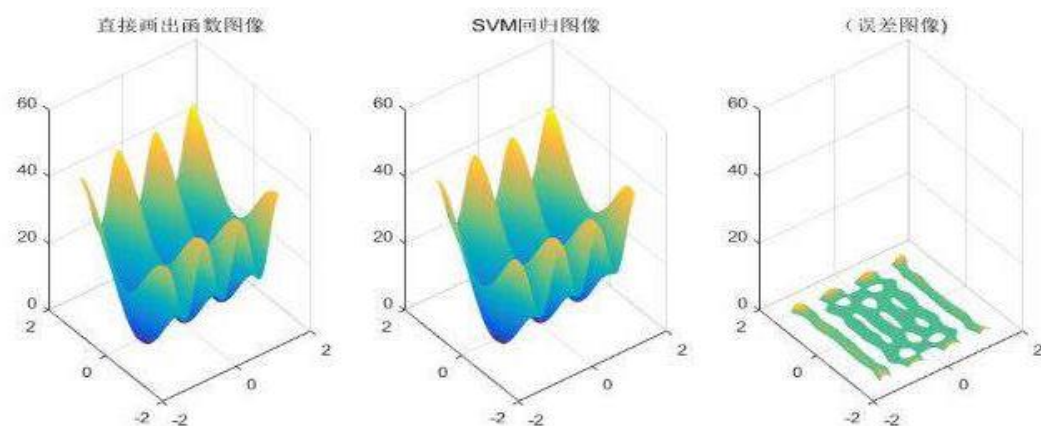


图 5 10000 个训练样本  重庆无线通信技术实验室

从上图可以看出，随着训练样本数的增加，拟合效果越好，误差也越小。在训练样本数达到一定量的时候，SVM 算法对曲线的拟合误差将非常小，但同时，训练样本数的增加，也将导致程序运行时间加长。

MATLAB 代码：

1、二维函数曲线拟合：

```
close all;
clear;
clc;
format compact;

%%生成待回归的数据
x=(-1:0.01:1)';
y = x.^5 + 2*x.^4 + 6*x.^3;
figure(1);
plot(x,y,'rd');
hold on;

% 训练模型
model = svmtrain(y,x,'-s 3 -t 2 -c 2.2 -g 2.8 -p 0.01');

%利用建立的模型看其在训练集合上的回归效果
```

```
[py,mse,devalue]= svmpredict(y,x,model);

%画图, 预测图
plot(x,py,'b-');
legend('原始数据','回归数据');
grid on;

2、三维函数曲线拟合:
close all;
clear;
clc;
format compact;
%画出真实函数图像
interval =0.01; %步长
[x1, x2]=meshgrid(-1.5:interval:1.5);
F = 20 + x1.^3- 10*cos(2*pi*x1) + x2.^3 - 10*cos(2*x2);
subplot(1, 3,1)
mesh(x1, x2,F);
zlim([0, 60] )
title('直接画出函数图像')

%%训练样本
x =rand(10000,2);
x = (x -0.5)*1.5 *2;
x3 = x(:,1);
x4 = x(:,2);
T = 20 + x3.^3- 10*cos(2*pi*x3) + x4.^3 - 10*cos(2*x4);

%建立 SVM 模型
model =svmtrain(T,x,'-s 3 -t 2 -c 2.2 -g 2.8 -p 0.01 -h 0');

%%产生测试样本
interval=0.01 ;
[i, j]=meshgrid( - 1.5:interval: 1.5);
row= size( i);
tx1 = i(:);
tx2 = j(:);
tx=[tx1,tx2];
%Ty=20+tx1.^3-10*cos(2*pi*tx1)+tx2.^3-10*cos(2*pi*tx2);
Y = tx1 + tx2;

%%进行预测,主要是预测 P
[P,mse,devalue]= svmpredict(Y,tx,model);
```

```
%%画图
v = reshape(P,row);
subplot(1, 3,2)
mesh(i,j,v);
zlim([0,60])
title('SVM 回归图像')

%%误差图像
subplot(1, 3,3)
mesh(x1, x2,F-v);
zlim( [ 0,60])
title ('误差图像')
set(gcf,'position', [300, 250, 900, 400])
```

本文原载微信公众号“重庆无线通信技术实验室”。

本文作者：孙宁，重庆大学无线通信技术硕士研究生，主研方向为人工智能算法及其在无线通信中的应用。



重庆无线通信技术实验室