### 案例 4：基于 SVM-PSO 模型的中小企业信用评分预测

* **问题背景**：中小企业是国民经济的重要组成部分，但融资难问题一直制约其发展。银行等金融机构需要对中小企业的信用风险进行评估，以决定是否放贷及放贷额度。中小企业信用评分受企业财务状况、经营年限、行业前景、负责人信用记录等多因素影响，且样本数据相对较少。
* **问题描述**：某商业银行计划构建中小企业信用评分模型，根据企业的各项指标预测其信用等级（分为 AAA、AA、A、BBB、BB、B、CCC 共 7 个等级）。要求模型能在小样本、高维特征情况下准确预测，帮助银行有效控制信贷风险。
* **数据情况**：提供该银行过去 5 年的中小企业贷款客户数据，每条数据包含企业的财务指标（资产负债率、净利润率、流动比率等 20 项）、经营数据（成立年限、员工人数、年销售额等）、行业类别、负责人信用记录及最终信用等级。数据量约 2000 条，部分企业存在财务数据不完整的情况。

### 案例 4：SVM-PSO 模型中小企业信用评分预测代码

|  |
| --- |
| import pandas as pd  import numpy as np  from sklearn.preprocessing import LabelEncoder, StandardScaler  from sklearn.model\_selection import train\_test\_split  from sklearn.svm import SVC  from sklearn.metrics import classification\_report, accuracy\_score  from pyswarm import pso  import joblib  # 数据加载与预处理  data = pd.read\_csv('small\_business\_credit.csv')  data = data.dropna()  # 信用等级编码  le = LabelEncoder()  data['credit\_rating'] = le.fit\_transform(data['credit\_rating'])  # 特征与目标变量  X = data.drop('credit\_rating', axis=1)  y = data['credit\_rating']  # 数据标准化  scaler = StandardScaler()  X\_scaled = scaler.fit\_transform(X)  # 划分训练集和测试集  X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_scaled, y, test\_size=0.2, random\_state=42)  # 定义SVM参数优化目标函数  def svm\_objective(params):  C = 10 \*\*params[0] # 惩罚因子，取指数避免负数值  gamma = 10\*\* params[1] # 核函数参数    svm = SVC(C=C, gamma=gamma, kernel='rbf', random\_state=42)  svm.fit(X\_train, y\_train)  y\_pred = svm.predict(X\_train)  return 1 - accuracy\_score(y\_train, y\_pred) # 最小化错误率  # 参数范围 (-5, 5) 对应 (1e-5, 1e5)  lb = [-5, -5]  ub = [5, 5]  # PSO优化  best\_params, \_ = pso(svm\_objective, lb, ub, swarmsize=20, maxiter=50, debug=True)  best\_C = 10 \*\* best\_params[0]  best\_gamma = 10 \*\* best\_params[1]  # 最优SVM模型训练与预测  best\_svm = SVC(C=best\_C, gamma=best\_gamma, kernel='rbf', random\_state=42)  best\_svm.fit(X\_train, y\_train)  y\_pred = best\_svm.predict(X\_test)  # 评估模型  print(f'最优参数: C={best\_C}, gamma={best\_gamma}')  print(f'准确率: {accuracy\_score(y\_test, y\_pred)}')  print(classification\_report(y\_test, y\_pred))  # 保存模型  joblib.dump(best\_svm, 'svm\_pso\_credit.pkl')  joblib.dump(scaler, 'scaler\_credit.pkl')  joblib.dump(le, 'label\_encoder\_credit.pkl') |