### ****案例名称：零售企业销售额预测与定价策略优化****

**案例应用场景：**连锁超市销量预测与动态定价

**案例所属专业：**经济、管理、金融、大数据类专业  
**适用课程：**商业数据分析、管理科学等  
**案例侧重点：**时间序列预测、回归分析、价格弹性模型  
**核心目标：**

1.掌握用AI预测销量并优化定价的方法

2.理解数据驱动决策的商业价值

### ****一、案例涉及的知识点****

**1.数据处理**

·缺失值处理（均值填充）

·特征工程（节假日标志、促销活动编码）

**2.算法模型**

·线性回归（分析价格-销量关系）

·SARIMA时间序列模型（季节性销量预测）

**3.商业分析**

·价格弹性系数计算

·利润最大化模型（利润 = (价格 - 成本) × 销量）

### ****二、案例涉及的技能点****

·数据可视化：绘制销量趋势图、价格-销量散点图

·模型调参：调整SARIMA的(p,d,q)(P,D,Q)参数

·商业决策：根据预测结果制定定价策略

### ****三、代码实现****（完整可运行，需安装pandas、statsmodels库）

**1. 数据预处理与可视化**

import pandas as pd

import matplotlib.pyplot as plt

# 加载示例数据集（模拟超市销售数据）

data = pd.read\_csv("sales\_data.csv") # 字段：日期, 销量, 价格, 是否促销, 节假日

data['日期'] = pd.to\_datetime(data['日期'])

# 可视化销量趋势

plt.figure(figsize=(12,4))

plt.plot(data['日期'], data['销量'], label='实际销量')

plt.title("近两年销量趋势分析")

plt.xlabel("日期")

plt.ylabel("销量（件）")

plt.grid(True)

plt.show()

#### **2. 价格弹性回归分析**

from sklearn.linear\_model import LinearRegression

# 构建特征：价格、促销、节假日

X = data[['价格', '是否促销', '节假日']]

y = data['销量']

# 训练线性回归模型

model = LinearRegression()

model.fit(X, y)

# 输出价格弹性系数（经济学意义）

price\_coef = model.coef\_[0]

print(f"价格弹性系数：{price\_coef:.2f}（价格每上涨1元，销量减少约{abs(price\_coef):.0f}件）")

1. **时间序列销量预测（SARIMA）**

from statsmodels.tsa.statespace.sarimax import SARIMAX

# 按周聚合数据

weekly\_data = data.resample('W', on='日期').sum()

# 训练SARIMA模型（参数需根据数据调整）

model\_sarima = SARIMAX(weekly\_data['销量'],

order=(1,1,1), # (p,d,q)

seasonal\_order=(1,1,1,52)) # (P,D,Q,周期)

results = model\_sarima.fit()

# 预测未来4周销量

forecast = results.get\_forecast(steps=4)

print("未来四周销量预测：")

print(forecast.predicted\_mean.round())

1. **定价策略优化**

import numpy as np

# 定义利润计算函数（假设成本为15元/件）

def calc\_profit(price):

predicted\_sales = model.predict([[price, 0, 0]])[0] # 假设无促销/节假日

return (price - 15) \* predicted\_sales

# 寻找最优价格（网格搜索）

prices = np.linspace(20, 40, 50)

profits = [calc\_profit(p) for p in prices]

optimal\_price = prices[np.argmax(profits)]

print(f"理论最优定价：{optimal\_price:.1f}元（预期利润{max(profits):.0f}元）")

### ****四、教学实施建议****

**1.数据集准备**

·使用公开数据集（如Walmart Sales Data）

·或生成模拟数据（代码见附录）

**2.课堂讨论**

·如何平衡预测准确性与商业可行性？

·价格弹性模型未考虑哪些现实因素（如竞争对手定价）？

**3.扩展任务**

·加入天气数据（温度、降雨量）提升预测精度

·用蒙特卡洛模拟评估策略风险

### ****附录：模拟数据生成代码****

import pandas as pd

import numpy as np

# 生成24个月日期

dates = pd.date\_range(start='2022-01-01', end='2023-12-31', freq='D')

n = len(dates)

# 生成模拟特征

np.random.seed(42)

data = pd.DataFrame({

'日期': dates,

'价格': np.random.normal(30, 5, n).clip(20,40), # 价格在20-40元波动

'是否促销': np.random.choice([0,1], n, p=[0.8,0.2]),

'节假日': np.where(dates.month.isin([1,2,5,10]), 1, 0), # 假设1/2/5/10月为旺季

})

# 生成销量（带季节性和价格影响）

base\_sales = 100 + 20\*np.sin(2\*np.pi\*dates.dayofyear/365) # 季节性波动

data['销量'] = (base\_sales

- 3\*data['价格'] # 价格弹性

+ 50\*data['是否促销']

+ 30\*data['节假日']

+ np.random.normal(0, 10, n) # 随机噪声

).astype(int).clip(0)

data.to\_csv("sales\_data.csv", index=False)

**案例特点**

**1.专业贴合：直接关联企业核心决策问题（定价、库存）**

**2.难度分级：**

·基础任务：运行现有代码分析结果

·进阶任务：添加新特征优化模型

**3.商业思维：强调从数据到决策的完整链条**

通过此案例，学生可在2-4课时内掌握AI在商业预测中的基础应用，并为后续学习更复杂的决策模型（如强化学习）奠定基础。