预测任务报告

LW

目录

1 任务分析

题目中给出了 70 天的数据，每个数据文件中有 32 个数值型特征和类别信息，其中同类别的信息是按时间排列的，不同类别的信息则没有确定的时间先后关系；同时样本外数据包含了样本内数据没有出现过的类别信息，如 test_example.csv 中的 C039，这给预测带来了以下挑战：

- 不确定性较大：样本内数据并没没有包含完整的标签信息
- 信噪比较低：金融场景下数据常见的特点
- 时序信息不足

从经验和直觉出发，以上第一、第二点较好理解，同时处理起来也更为灵活：个人的假设是类别信息类似于股票代码，这些可以不作为特征加入预测模型，对于没有出现的类别，也可以采用 KNN 等聚类计算 top-K 向量的方式来估计其相关性较高的已知类别。

而第三点带来了较大挑战：首先测试集并没有明确指出数据属于哪天，在预测时，我们仅拥有 32 个数值特征信息和一个类别标签；其次，时序模型要确保输入的时序合法性，不能用未来的数据预测现在，尽管给出的训练集中完整文件是按天划分成 70 分，但不同类别间的数据却是没有相对时序关系保证的，这导致一些需要时间信息的模型（如 transformer 中 position encoding 等) 无法应用在该预测中。

为了较好的完成该预测任务，我们**选择了迭代的方式，通过使用简单到复杂的模型，来逐步探索解决上述挑战，从而改进预测的精度**；在流程上，我们会先进行数据分析，考虑数据分布的特点和可能使用的模型，再从简单的线性回归等回归模型入手分析预测效果，再考虑使用树模型提高预测稳定性，以及用多层感知机模型来进一步加入非线性性结构的刻画；对于时序性和不确定性，我们考虑使用高斯过程回归来提高模型的预测能力，最后总结并分析可行的改进方向。

2 数据分析

数据探查是建立模型前的重要工作，除去一些常见的数据分析，这里我们先指出对所给数据集进行方差分析的结果。方差分析（ANOVA）通过比较样本内（组内）和样本间（组间）的变异来确定不同组之间是否存在显著差异。具体计算上，总平方和 SST 是所有观测值与总均值差的平方和，组间平方和 SSB 是各组均值与总均值差的平方和，组内平方和 SSW 是每个组内观测值与该组均值差的平方和。F 值 = 组内均方差 MSB / 组间均方差 MSW，如果 F 大于 F 分布临界值则拒绝 H0，可以认为该因子引起的差异显著。ANOVA 结果如下图所示：

	sum_sq	df	F	PR(>F)
C	7.867794	205.0	0.823417	0.96938
Residual	46358.453841	994603.0	NaN	NaN

图 1 类别特征的分析

	sum_sq	df	F	PR(>F)
F1	0.341107	1.0	6.889994	8.670464e-03
F2	1.631969	1.0	32.963991	9.439680e-09
F3	0.154898	1.0	3.128760	7.692803e-02
F5	3.486929	1.0	70.432139	4.880540e-17
F6	13.958845	1.0	281.953363	4.117169e-63
F8	0.574371	1.0	11.601666	6.594200e-04
F9	2.934027	1.0	59.264119	1.402650e-14
F10	1.702912	1.0	34.396946	4.521072e-09
F11	18.196093	1.0	367.541115	1.220757e-81
F12	0.286912	1.0	5.795301	1.607249e-02
F14	0.436600	1.0	8.818832	2.982719e-03
F18	0.455943	1.0	9.209543	2.408744e-03
F19	0.263733	1.0	5.327111	2.100003e-02
F20	0.814418	1.0	16.450354	5.001279e-05
F21	0.734903	1.0	14.844244	1.169023e-04
F22	2.458079	1.0	49.650492	1.859749e-12
F24	0.760441	1.0	15.360082	8.895796e-05
F25	1.154066	1.0	23.310862	1.382059e-06
F26	1.000063	1.0	20.200186	6.989471e-06
F28	0.287054	1.0	5.798171	1.604627e-02
F29	1.110730	1.0	22.435522	2.178834e-06
F31	2.334675	1.0	47.157868	6.621810e-12
F32	1.001985	1.0	20.239004	6.849125e-06
Residual	2604.101811	52600.0	NaN	NaN

图 2 其他特征的分析

如图可知，C 的 P 值较大，大于常见的显著性水平 0.05 或 0.01，说明分类变量 C 对因变量 Y 的影响不显著。再对其他的变量进行方差分析，可以得出：这说明 F2, F5, F6, F9, F10, F11, F14, F18, F20, F21, F22, F24, F25, F26, F29, F31, F32 的显著性较强，其他变量的显著性一般。

（本节代码在 ANOVA.py）

3 回归分析

本节中，我们通过简单的线性回归等模型来建立初步的预测模型，通过交叉验证来衡量其预测效果。我们按具体模型来展开讨论：

3.1 多元线性回归

使用 one-hot 对分类变量 c 进行编码。首先对于这一问题，以 data0 为例，分析其数据分布。以 IQR 作为衡量极端值的方式，四分位点分别记为 Qi, 则 IQR = Q3 –Q1，位于 Q3+1.5IQR 或 Q1-1.5IQR 之外的数据可以认为是极端值。本数据结构中极端值较多，且不存在缺失值。

对分类变量和特征变量进行多元线性回归，作为初步的统计分析。统计结果表明，回归 R2 约为 0.04，且决断 R2 很小。这说明直接进行多元线性回归的回归效果一般。更进一步地讲，使用 Omnibus 检验模型残差正态性，检验结果高于临界值，表明残差显著偏离正态分布。使用 Durbin-Watson 检验残差的自相关性，检验结果为 1.779 ，接近 2，表明残差中没有显著的自相关性。条件数为 68.3，表明存在一些多重共线性的问题，但并不是非常严重。

由于残差偏离正态分布，因此直接使用多元线性回归的效果一般。此时可以使用 Box-Cox 变换，这是一个对于因变量的一个变换方式，使数据更接近正态分布，从而提高模型的准确度和可靠性，提高预测精度，并减少过拟合的风险。

使用 Chow 检验，评估不同数据集是否服从相同的回归方程。以 data0 到 date10 为例，将数据集按照时间顺序分割成两部分，分别拟合整体模型与各自模型。得到 Chow 检验 F 统计量: 为 0.7901877，p 值为 0.9928，未拒绝原假设，数据集可以合并研究。

3.2 含有交叉项的回归以及 RFE

由于简单线性回归中存在欠拟合的情况，此时考虑加入交叉项以增加特征数。具体而言，加入 F1 到 F32 的两两交叉项，此时需要考虑对特征进行筛查与选择，RFE 是一种递归特征消除法，递归地构建模型并在每一轮训练过程中去掉最不重要的特征，来找到最佳的特征子集。以含交叉项的回归方程作为基模型，利用回归系数的 t 检验和回归系数来评估特征重要性，每次去掉对模型性能影响最小的一个或多个特征。

3.3 Lasso 回归

Lasso 回归相比于 RFE，都可以进行特征数的压缩，但 RFE 的计算开销相对较大，且需要通过手动选择或者交叉验证的方法选择最终的特征数。Lasso 回归的计算更加便捷，其中需要调整参数 alpha。以 alpha = 0.006 为例，最终选择的特征为: ['F2_F9' 'F2_F16' 'F5_F14' 'F8_F27' 'F9_F22' 'F9_F27' 'F10_F14' 'F11_F32','F14_F25' 'F20_F32']，且经过验证，此时各个特征的 t 检验结果均显著。

作为初步尝试，我们取了 10 天数据随机选取 20% 作为验证集，得到的 rmse 结果为: linear: 0.23034 ，lasso 0.23015，rfe 0.23019

（本节代码在 rg.py 中）

4 树模型预测及比较

相比于相对简单的线性回归与 lasso 回归，随机森林可以通过构建多个决策树并结合其输出结果来进行回归，通过多数决策来提高模型的稳定性和准确性，同时减少过拟合的风险。在本例中，通过训练 100 棵决策树，得到最终的决断 R2 为 0.05，相较于其他回归方式有着更强的预测能力与泛化能力。(这部分代码也在 rg.py 中)

lgbm 模型由于其高效可并行和内存占用少，也是在偏中高频要求性能较高时常用的预测模型，这里我们使用 lgbm 也完成了一个回归模型并对其进行了调参数，为了公平起见且消除潜在的时序不合理现象，我们选择 70 天的数据中前 60 天作为训练集，后 10 天作为验证集，得出的 rmse 是 lgbm 更为优秀。注意这里由于逻辑上我们假设可能的测试集是在日期

靠后的（假设就是 60 天之后），并且做出了按照日期先后顺序的划分来评价模型表现，而树模型实际上在该任务中并不考虑时序信息，如果采用随机划分训练集和验证集的方式，我们的 rmse 可以达到 0.21 附近稍高一点的数值。

这部分内容和代码我们写在 Jupyter Notebook 中以方便直观展示和调用 predict 函数（homework.ipynb），同时我们也使用加权线性回归处理异方差，并在相同的集合上进行比较；当然，这只是一个比较的角度，不同模型取得最佳表现的场景不同，在程序包里也提供了 mod.py，rf_predict.py（随机森林），以及 fwls_prediction.py 进行单独调参，优化和预测。

5 非线性性与神经网络模型

我们也考虑了利用神经网络来建立回归模型做预测，这里面有两个难点：

- 神经网络特别是深度神经网络如 CNN 等需要对模型结构等超参进行设计和优化，涉及到的调参和训练时间较长
- 由于预测中的时序不确定问题，以及测试集不带有时间戳，该场景给使用一些模型比如 LSTM 等带来较多不便，也会影响趋势、上下文等信息的捕获

考虑到上述两个制约因素，我们决定采用多层感知机来进行神经网络回归，来增加模型的非线性结构和处理复杂数据的能力，这部分代码在 mlp.py 中。

6 时序性与高斯过程回归

考虑到本预测任务的不确定性，相比于复杂的深度神经网络回归，在这一周时间内我们优先考虑使用高斯过程回归，取得了比上述模型更好的效果，这部分代码在 test_gpfy.py 中，参数为 param_new_3000.pkl。

高斯过程是一种核方法，对测试样本的预测依赖于测试样本和所有训练样本之间的关系，这种关系由样本之间的核函数值决定。一般而言，时间上比较接近的样本之间的相似性较高，对应的核函数值较大。于是可以认为，与测试样本时间上比较接近的训练样本对预测结果的影响较大。在本次预测任务中，由于测试集没有显示的给出时间，预测的核心

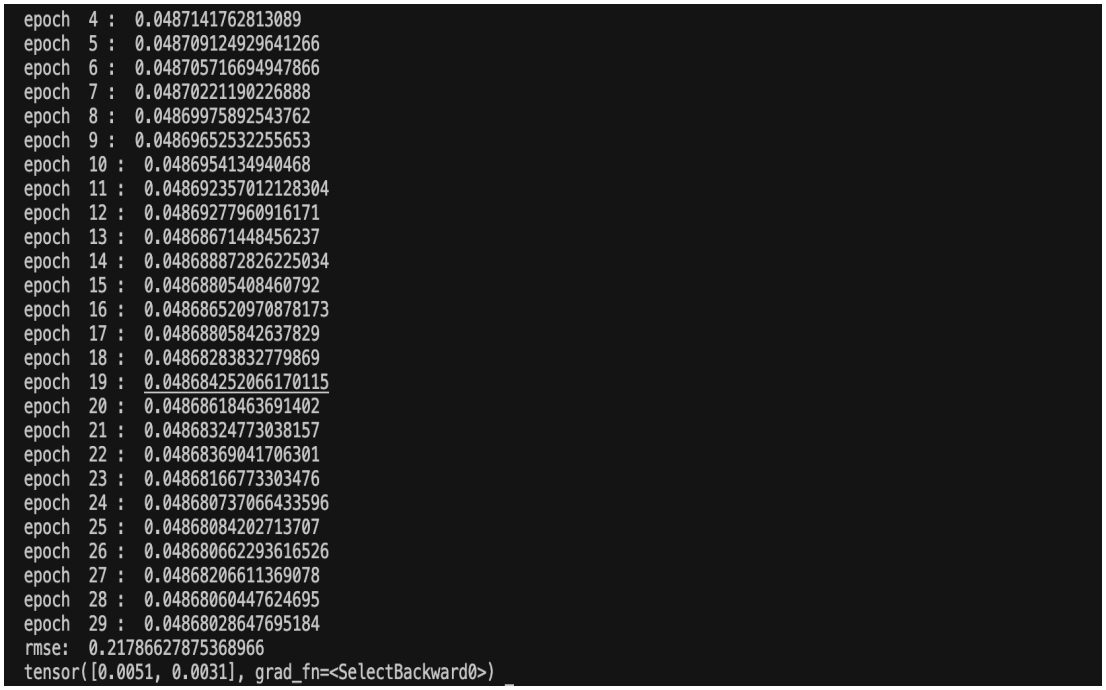

图 3 多层感知机的结果

依据是数值特征，所以我们希望利用高斯过程的特点（协方差矩阵是 X 的函数）去捕获训练数据中，蕴含在数值特征中的时序信息，从而提高预测的精度。

由于百万数据上实现一般的高斯过程需要很大的存储空间，且计算很耗时，这里我们参考了论文 [1]，利用特殊的高斯过程来构建回归模型，并借助 github 上代码仓库 GPFY[2]。由于时间原因，这里我们使用了 70 天全部训练数据集进行训练，然后在测试集上给出结果，迭代了 3000 次，训练了 14 小时，所得 rmse 为 0.20953966254804757

这部分算是我们提升和改进的重点，目前尝试不同参数也有取得更好效果，认为该模型会比前述模型表现都更优秀，由于时间原因就没有展开公式和论文应用思路。

7 总结与展望

在本预测任务中，我们从问题的难点入手，逐步递进的考虑模型的选择与迭代，在最小化 rmse 的目标下提高预测的精度，考虑了多种模型和解决预测难点的思路，由于时间限制，我们还期望以下方向来优化预测结果：

- 模型融合：通过结合多个模型的优势来提高预测精度和鲁棒性，在该预测任务中，可

以使用加权平均融合，更为精细的方式还有待研究。

- 模型结构：设计更加复杂精细的模型结构，如深度神经网络等来学习数据中蕴含的模式；特别的，有更多时间进行高斯过程回归的迭代也可以得到更好参数和效果。

参考文献

[1] Stefanos Eleftheriadis, Dominic Richards, and James Hensman. Sparse gaussian processes with spherical harmonic features revisited. *arXiv preprint arXiv:2303.15948*, 2023.

[2] Eleftheriadis Stefanos. Gpfy. `https://github.com/stefanosele/GPfY`, 2024.