2022.5.19

基于特征的LSTM组合结果

特征选取与参考集/测试集特征计算方式类似于3.14汇报文件，具体如下：

数据：取最后28期作为测试集，倒数29-56期作为训练组合模型的参考集；首先计算参考集阶段的时间序列特征，即计算时间序列从有正数据的第一期至参考集第一期、至参考集第二期、.....、至参考集第二十八期的间序列的特征，所涉及的24个特征如下：

- 间断数据特征：ADI、CV2、零值比例
- 熵（使用 `tsfeatures` 的 `entropy()` 计算）
- 基于 STL 的特征（使用 `tsfeatures::stl_features()` 计算）
- 基于自相关的特征 （使用 `tsfeatures::acf_features()` 计算，数据的频率设为7，即考虑7天的季节相关）
- 滞后1-7日与滞后1-28日内的均值与标准差

对于测试集，由于不知道时间序列的真值，以上特征不可直接计算，采用如下方法估计特征：将测试集28期的所有方法的分位数预测结果（0.01-0.99）都进行简单平均，之后利用这28期的分位数情况去抽样获得100条测试期的可能数据（取整以保证是一个需求数据），利用这100个时间序列数据与之前的真值获得测试期的时间序列特征估计（100条时间序列特征结果取均值）。

LSTM的网络结构也同3.14汇报，为一个LSTM层接一个线性层。关于损失函数的设定，考虑以下3种：

- 类似于FFORMA，即基准模型损失的加权平均
- 基准模型加权组合后的平均损失
- 考虑引入截距：将网络输出增加一项用于表示截距，对网络输出进行 $tanh()$ 变换以使输出的系数与截距结果限定在[-1,1]。最终使用组合结果的损失。

其平均损失结果及与回归方法、平均方法的对比如下，可以看到使用组合后损失其结果更准确；引入截距后貌似比损失平均的结果要好一些，比组合后损失互有好坏，整体差距不大；其在两端相对表现更差，中间较好。——也就是说，引入截距后在LSTM体系里没有特别突出的改进。

（注：LSTM输出权重结果存随机性，故每次实验结果不能完全一致，但前三位系数基本不会变化）

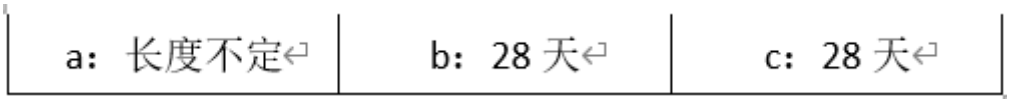
这里额外展示了两种“作弊”情况的平均结果，用以观察方法与上限的距离：逐点最优指在测试集某条序列某个时点在该点预测最优的结果，逐序列指训练集在某序列28期平均结果最优的方法结果。

分位数	0.01	0.025	0.165	0.25	0.5	0.75	0.835	0.975	0.99
面板回归（11/10模型）	**0.0143**	**0.0356**	**0.2150**	**0.3064**	**0.4890**	**0.4892**	**0.4204**	0.1340	0.0695
加权平均（11模型）	0.0155	0.0376	0.2210	0.3145	0.5035	0.5003	0.4269	0.1342	0.0696
简单平均（11模型）	0.0171	0.0389	0.2208	0.3147	0.5046	0.5010	0.4270	**0.1337**	**0.0694**
LSTM组合（损失平均）	0.0143	0.0356	0.2230	0.3161	0.4921	0.4907	0.4254	0.1448	0.0737
LSTM组合（组合损失）	0.0143	0.0357	0.2158	0.3066	0.4915	0.4908	0.4237	0.1363	0.0724
LSTM组合（带截距）	0.0149	0.0371	0.2170	0.3071	0.4923	0.4903	0.4204	0.1375	0.0723
LSTM组合（带截距，滞后特征）	0.0161	0.0376	0.2172	0.3071	0.4975	0.4894	0.4206	0.1336	0.0699
池内逐点最优方法均值	0.0122	0.0291	0.1514	0.2021	0.2644	0.2184	0.1803	0.0568	0.0276
池内逐序列最优方法均值	0.0134	0.0333	0.1975	0.2801	0.4395	0.4217	0.3529	0.0953	0.0423

模型权重的变化情况：以0.975分位数在测试集上的结果为例，可以看到除了在预测horizon初期的时候有所变化，后逐渐收敛不变。有如下几种可能原因：

- 测试集上的特征是通过预测数据的重复抽取估计的，而基准预测倾向于长期收敛，使得特征不变；
- 每次只新引入1期数据使得特征变化不会太大；
- 28期对于评估预测权重分配并不够长。

对此问题进一步思考：首先可以都采用滞后数据的特征以解决测试集上的输入问题；但是仅有28天refer set上的预测值来估计接下来28天的时变权重分配，是否足够支持？（当然，如果增加refer set的预测值，这使得计算量增加很多，因为基准预测生成的工作量非常大）



下面是无/有截距的系数结果，引入截距后并放松系数限制后对分配结果变化还是很明显的。

分位数	0.01	0.025	0.165	0.25	0.5	0.75	0.835	0.975	0.99
quantGAM	0.0001	0.0004	0.0072	0.0184	0.0269	0.0246	0.0406	0.0810	0.0855
VZ	0.0816	0.2303	0.0022	0.0059	0.0116	0.0121	0.0350	0.1623	0.1962
WSS	0.8592	0.5926	0.1664	0.2503	0.0115	0.0085	0.0120	0.0364	0.0628
poisson_static	0.0000	0.0000	0.0005	0.0021	0.0084	0.0078	0.0146	0.0360	0.0501
poisson_damped	0.0000	0.0000	0.0007	0.0077	0.0437	0.0623	0.2732	0.0775	0.0872
poisson_undamped	0.0000	0.0000	0.1278	0.1452	0.7155	0.7573	0.4352	0.1480	0.0954
nb_static	0.0070	0.0687	0.0052	0.0087	0.0138	0.0070	0.0144	0.0418	0.0691
nb_damped	0.0005	0.0006	0.0330	0.0445	0.0168	0.0098	0.0263	0.1185	0.0847
nb_undamped	0.0008	0.0079	0.2631	0.1662	0.0289	0.0180	0.0323	0.1023	0.0811
LSTM	0.0108	0.0933	0.3606	0.3149	0.0630	0.0296	0.0363	0.0568	0.0797
lgb	0.0400	0.0063	0.0333	0.0361	0.0598	0.0629	0.0802	0.1393	0.1081

分位数	0.01	0.025	0.165	0.25	0.5	0.75	0.835	0.975	0.99
quantGAM	0.0478	0.0209	0.0178	0.0243	0.0326	0.0664	0.0602	0.0548	0.1584
VZ	0.2757	0.1925	0.0290	0.0426	0.0405	0.1146	0.0017	0.1477	0.0011
WSS	0.0305	0.0830	0.1716	0.0907	0.0288	0.0616	0.0958	-0.0136	0.2998
poisson_static	-0.0073	-0.0098	-0.0147	-0.0330	-0.0273	-0.1040	-0.0417	0.1352	-0.0280
poisson_damped	0.0135	-0.0180	0.0051	0.0620	0.1575	0.0679	0.1946	0.1010	0.0123
poisson_undamped	0.0340	-0.0186	0.0722	0.0892	0.3965	0.3162	0.2061	-0.0532	0.0956
nb_static	0.2079	0.2753	0.0995	-0.0165	0.0339	-0.1070	0.0814	0.1605	0.0001
nb_damped	0.0407	0.0771	0.0776	0.0337	0.0138	0.0870	0.0969	0.0033	0.0378
nb_undamped	0.0656	0.0954	0.1705	0.1184	0.0838	0.1460	0.0923	0.1791	0.1627
LSTM	0.1717	0.2705	0.2769	0.2148	0.1401	0.0952	0.1212	0.0569	0.0151
lgb	0.0103	-0.0558	0.1482	0.2003	0.1719	0.1349	0.1442	0.1769	0.1558
截距	-0.0038	-0.0171	-0.0015	-0.0023	0.0158	-0.0091	0.0376	0.0959	-0.1514

接下来会尝试使用利用深度学习方法构造特征而非人为选取，以此支持元学习需要。目前来看，LSTM方法可能还有一定改进空间。

整数输出?

考虑到间断数据中离散分布的分位数问题，理论上分位数应该是个整数，才能保证其是在整数分布列下的结果。

基准方法除LGB外都能保持较高比例的整数输出，这与输出方式有关。如果考虑大多数基准预测的方法，可以简单取近似整数值来作为分位数预测结果。但是由于研究任务是考察分位数组合预测的效果，所以基准方法可以视作中间结果，不将绝对整数值作为训练？

但是，作为最终预测，组合预测可能需要考虑化为整数输出，以便在较多较密的分位数估计中找到分布列。目前来看，貌似是结果取近似与做计数数据分位数回归两种方法，其它方法有待考虑。