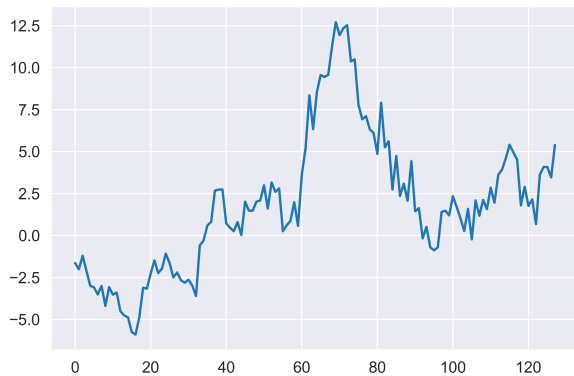


时间序列的图像表示和ARIMA模型的参数确定

数据生成阶段的改进

- 之前 p , d , q 完全混合, d 若非0, 时间序列存在趋势项, 可能会导致图像编码的时候无法准确的衡量两个时间点的距离 (或者相似度)
 - 修正: p 和 q 的取值范围: 0, 1, 2, 3, 4, 5; 由于 d 很好区分, 在实验阶段 d 恒定为0, 保持序列的平稳。
- 在 d 为0时候按照`auto.arima`生成的序列仍然是非平稳的, 尤其是 p 和 q 阶数比较高的时候

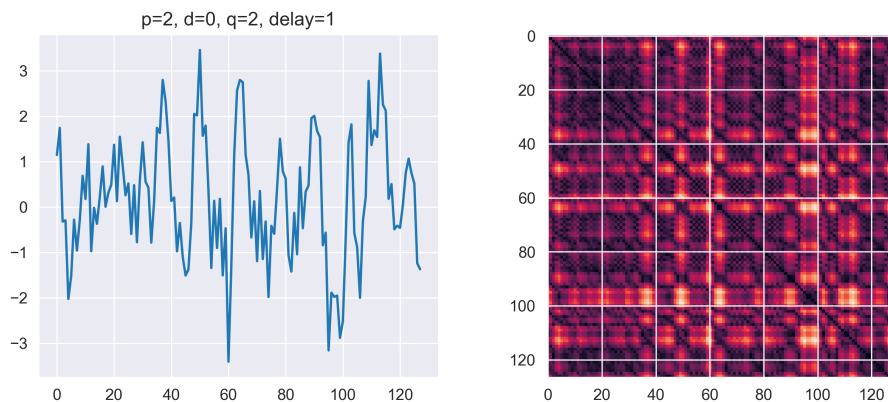
如下图: `order=(2, 0, 4)`, `ar=(0.3, 0.6)`, `ma=(0.6, 0.5, 0.3, 0.6)`, 平稳性检验 p 值: 0.23



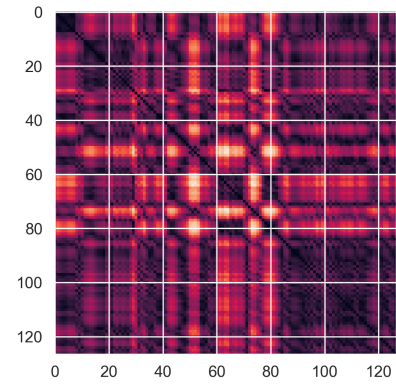
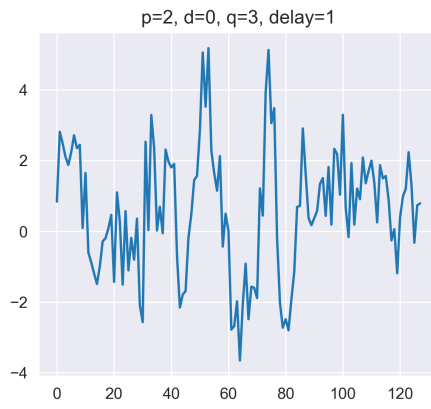
- 增加: 更加严格的系数设定和平稳性检验, 保证AR部分的平稳, 生成时间序列再做一次平稳性检验, 丢弃不通过检验的数据

图像编码阶段的分析

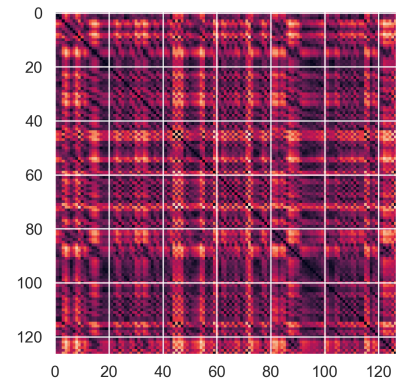
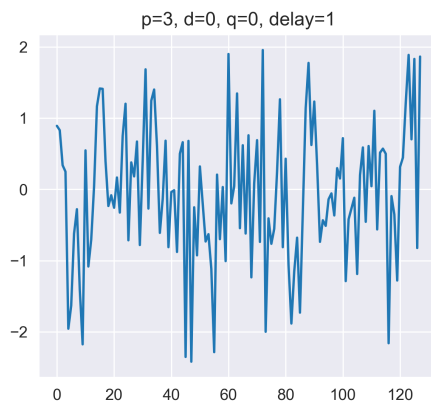
- 发现了之前的一个bug: Recurrence plot的维数设置为2, 才能揉合进未来时间点的信息, 捕捉相同的升降趋势, 之前默认设置为1, 导致一个点的坐标只是用自己的取值代表的, 没有考虑邻近的点
- 修改之后不同时间序列的图像编码:
 - RP反映了不同时间点的相似关系, 黑色区域对应的取值更小, 表明两个时间点的相似度更高



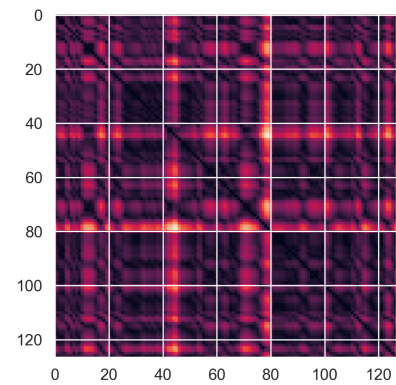
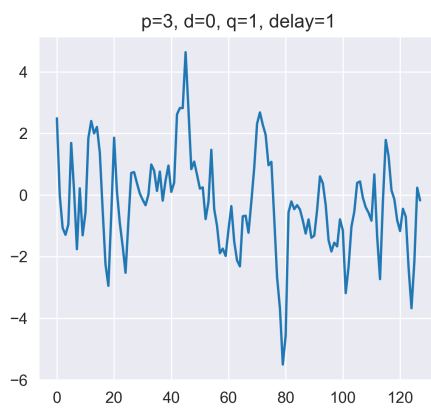
- 极端波动区域与其他时间段相距较远, 对应RP的颜色偏亮



- 取值波动较大的序列的相似格子会很少

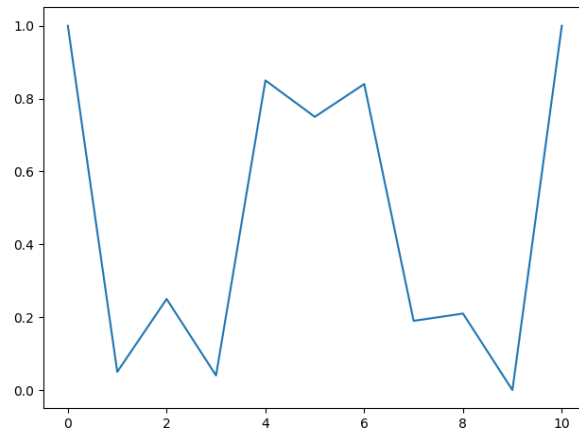


- 取值平缓的序列对应跟多黑色区域

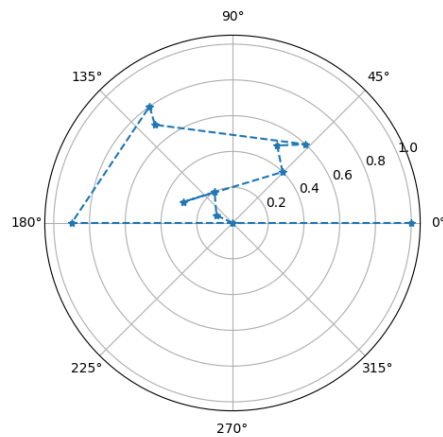


- NOTE: recurrence plot 中的距离度量方式或许可以改成余弦距离
- 第二种编码方式: GAF

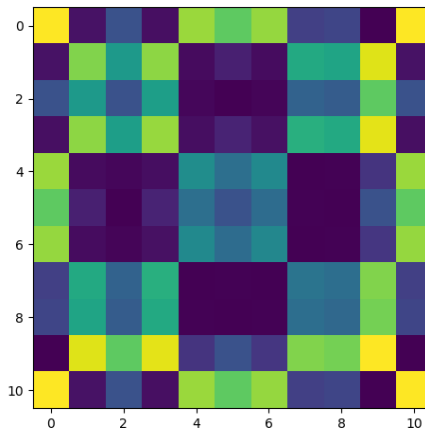
○



○



○

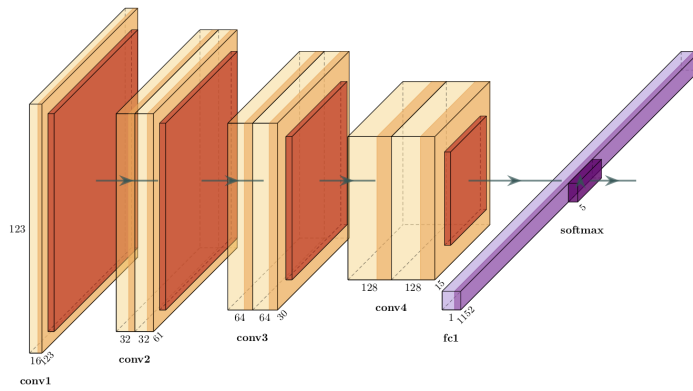


用 $\cos(\theta_1 + \theta_2)$ 度量两点的距离，相当于加了惩罚项的内积

- 不同的编码方式形成不同的通道
 - 尝试只用RP作为编码方式，取不同的滞后阶数形成的图片作为不同的通道，由于p, q的最大值都是5，因此构造了5个通道的图像数据作为特征进行分类，
 - 使用RP, GAF, MTF三种编码作为3通道图像作为训练数据进行分类
 - 第二种方法的效果比第一种好。

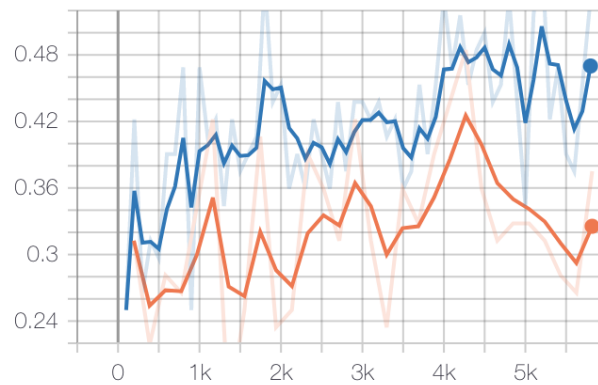
模型训练

- 之前一直尝试使用预训练的模型 (VGG) + finetune 进行分类，效果并不好
- 改进：考虑到任务与传统的图像分类的任务并不一样，因此因此从头到尾训练一个cnn model，模型结构如下：



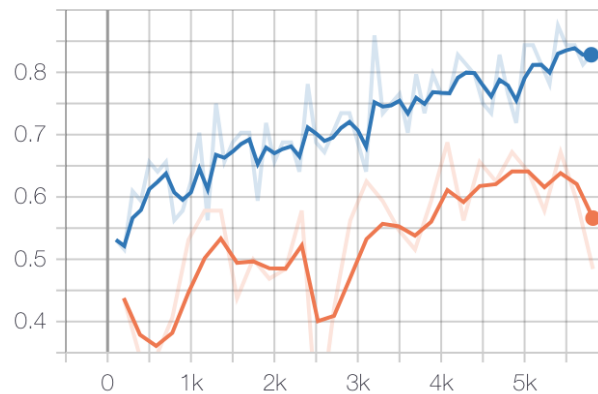
- 准确率
 - $p(0.3 \sim 0.35)$:

accuracy



- $q(0.5 \sim 0.64)$:

accuracy

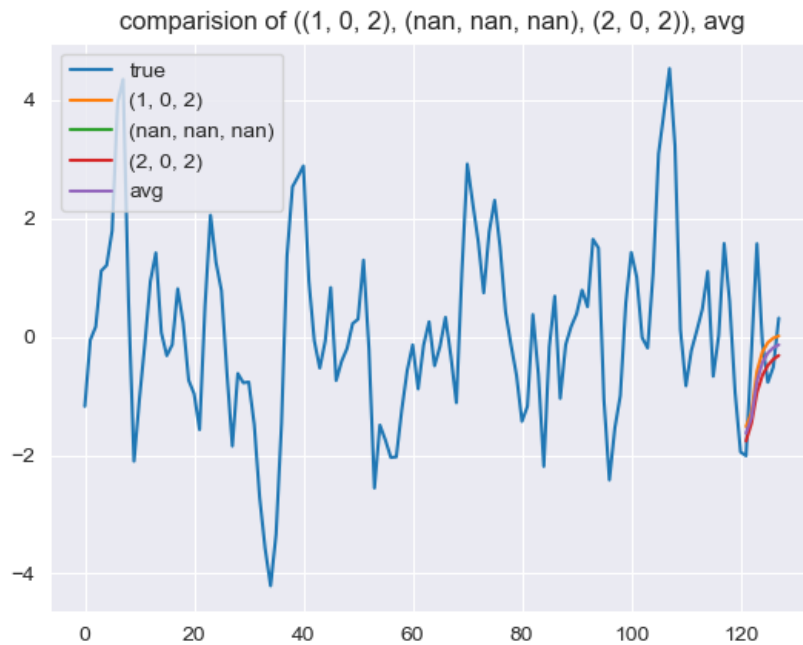


- p 很低, q 还行, 似乎都很低, 为了对比, 使用auto arima定阶在测试集上的准确率如下
 1. accuracy of p : 0.2671387
 2. accuracy of q : 0.2944963
- NOTE: 或许可以尝试不去区分 p 和 q 而是区分 p 和 q 的combination, 这样就不用训练两个模型, 直接进行25类的多分类

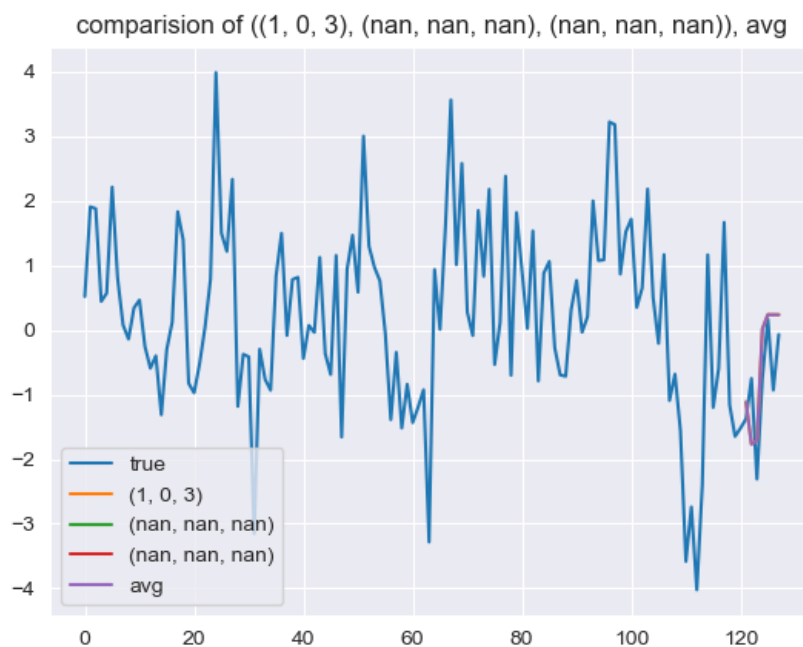
预测和model average

- 流程: 利用训练好的模型对测试集的时间序列的 p 和 q 进行预测, 得到 p 和 q 的预测类别概率分布, 取 top_n 的类别进行组合可以得到不同的ARIMA模型的 p , q 参数的阶数组合, 以及组合对应的概率, 用不同的参数组合可以拟合不同的模型, 对应参数组合的概率可以作为不同模型的权重进行model_average
- 示例: 给定一条时间序列, 预测得到:

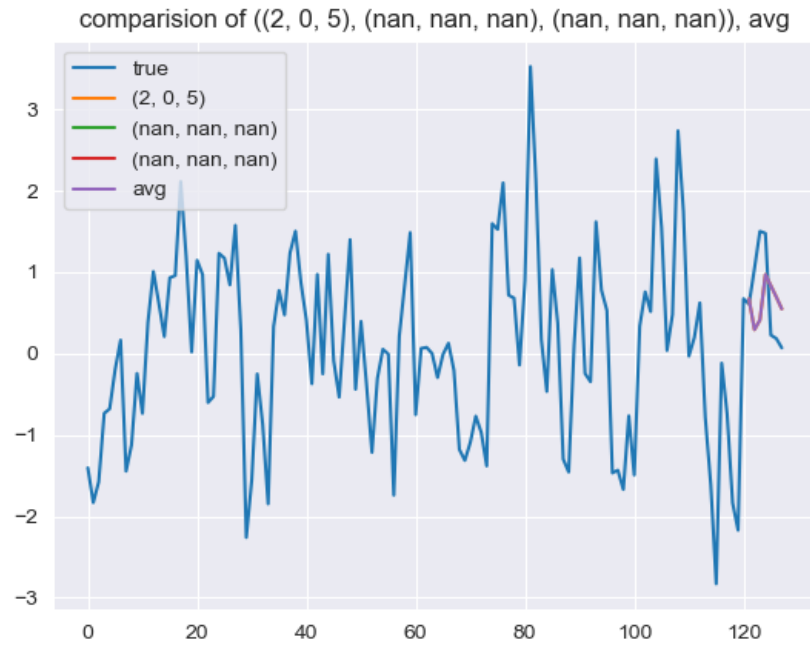
- p的不同类别的top_3分布为{2: 0.6, 1: 0.2, 3: 0.1, ...}
 - p的不同类别的top_3分布为{0: 0.4, 1: 0.3, 3: 0.2, ...}
 - 组合参数: (2, 0), (2, 1), (2, 3), (1, 0), (1, 1), (1, 3), (3, 0), (3, 1), (3, 3)
 - 计算组合参数的权重, 取top_3, 如(2, 0), (2, 3), (1, 0)
 - 使用不同的参数进行ARIMA建模, 得到单个模型的预测结果
 - 用参数对应的权重进行average, 得到最终的预测结果
- 测试集表现 & bad case 分析
 - true order: (3, 0, 2)
 -



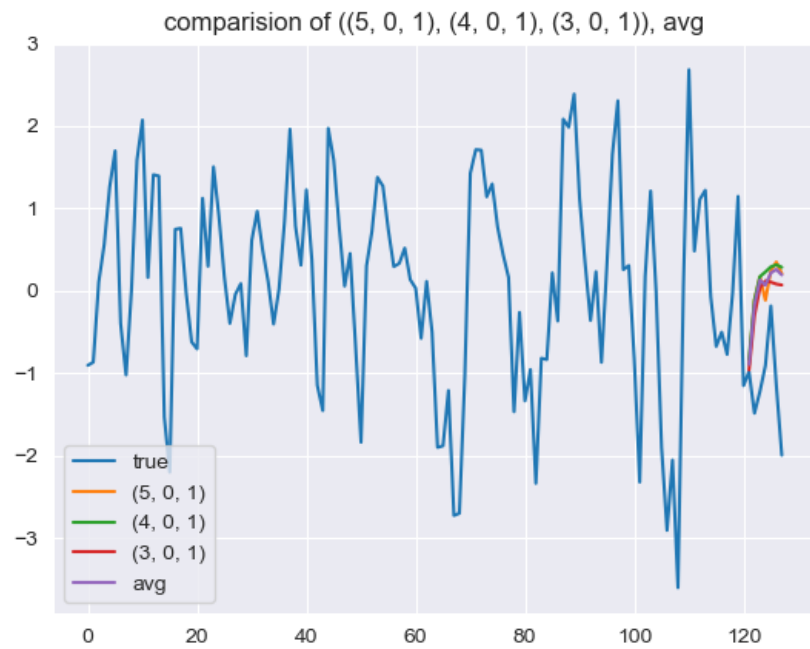
- true order: (1, 0, 3), 对阶数完全分类正确的case预测表现更好, 优化分类模型仍然是关键, model average必须建立在单个模型较为准确的情况下进行
 -



- true order: (1, 0, 5)
-

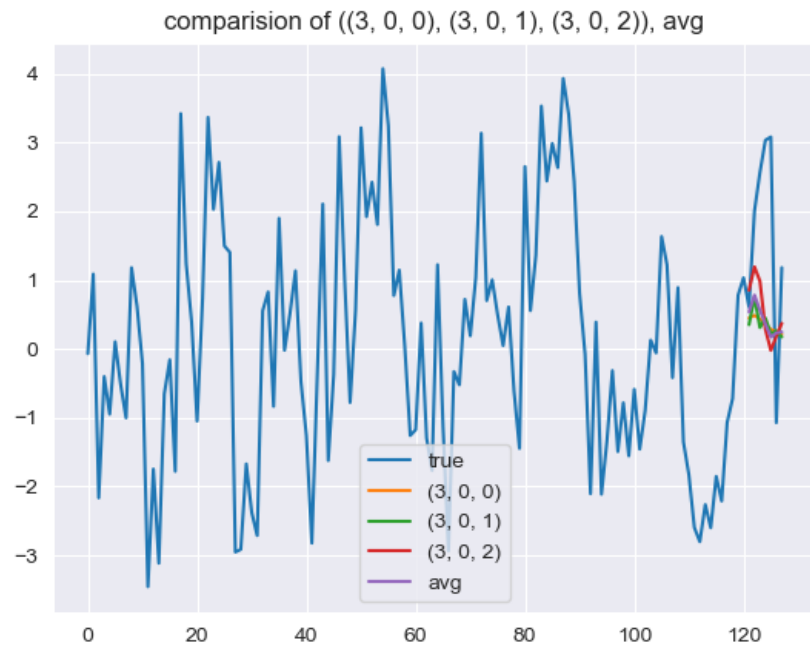


- true order: (1, 0, 3)
-



- true order: (0, 0, 5)

o



o true order: $(0, 0, 0)$, 生成数据中存在的这类p, d, q全为0的数据, 这类数据的预测偏离更大

o

