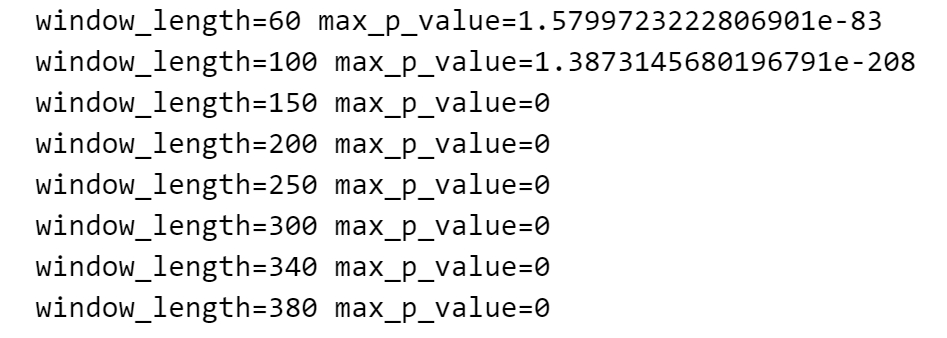
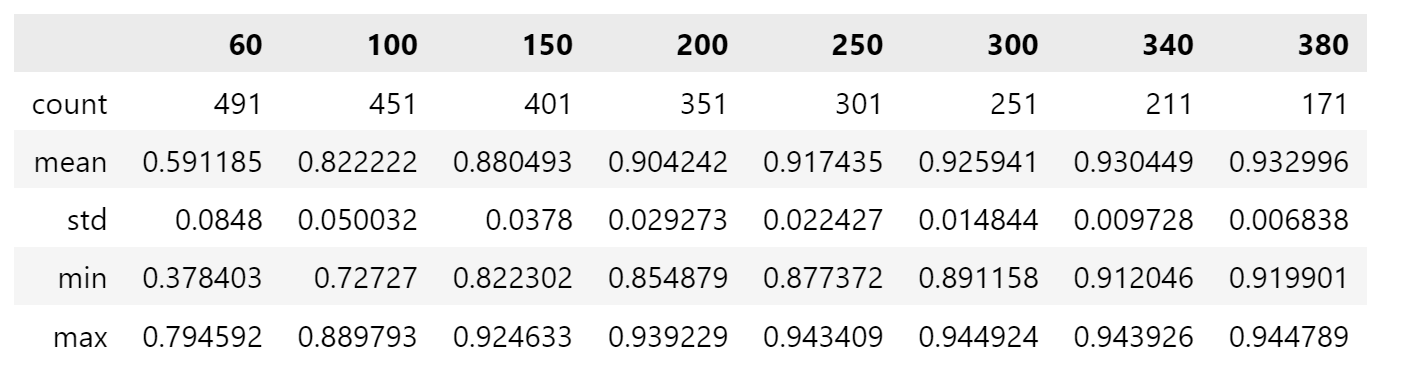
1. **确定window length**

window\_lengths = np.array([60,100,150,200,250,300,340,380])

bartlett球状检验，p值接近0，拒绝原假设，说明变量之间有相关关系。rolling测试取max\_p\_value，结果如下：

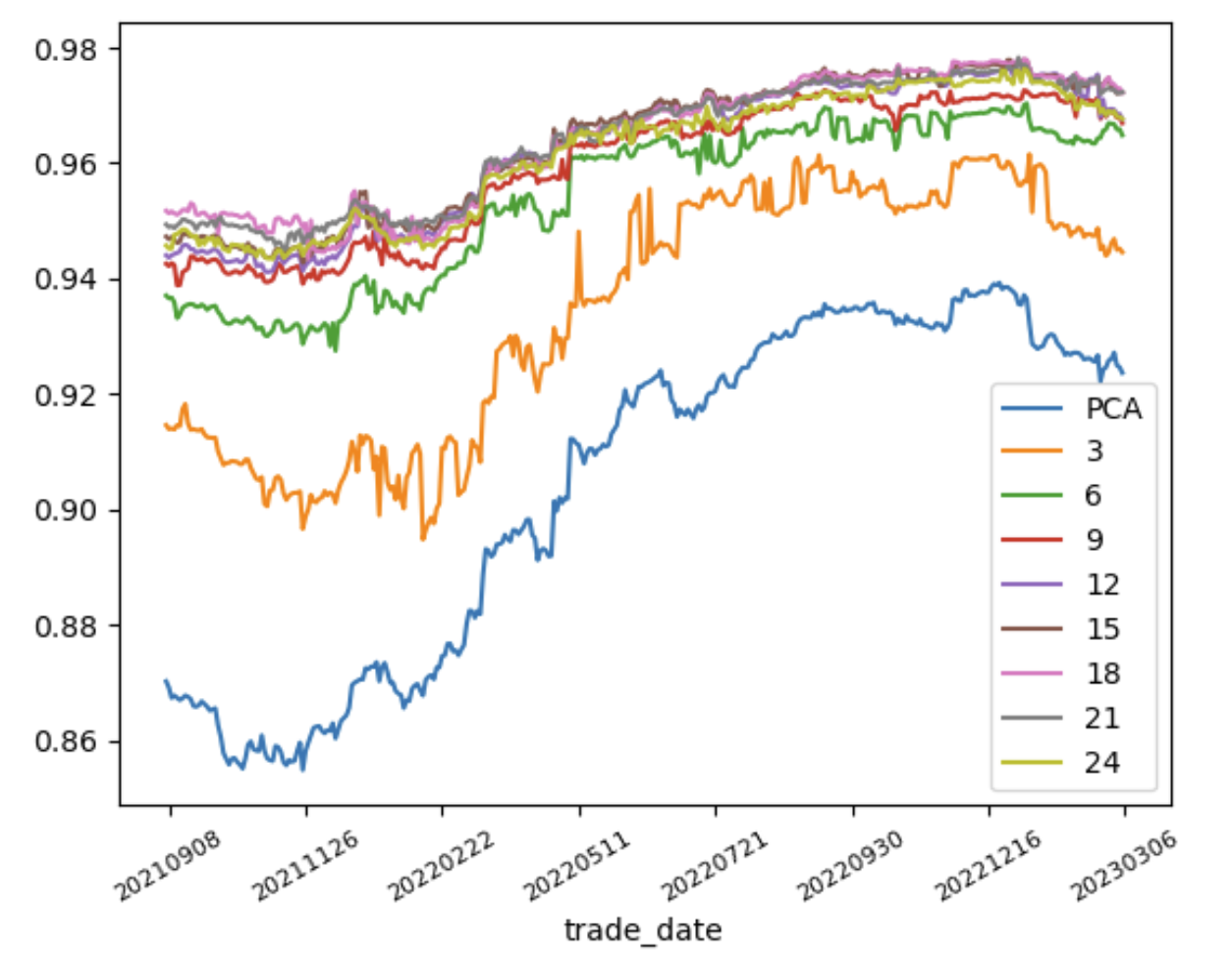
都很接近0 区分效果差

kmo检验，统计量越接近1，变量间的相关性越强，偏相关性越弱，主成分分析的效果越好。rolling统计结果如下：

window length < 200时 ，mean小于0.9；window过长对数据要求变大，加上时间跨度过长 股票与市场的相关关系 与 当前的相关关系 会有偏差。综合考虑，取window length = 200

1. **确定nb\_cluster**

kmo检验法：预设nb\_clusters = [3,6,9,12,15,18,21,24]

在window length = 200 的条件下rolling计算kmo统计量

HPCA分层之后明显优于PCA，数据线性相关性更强，更适合做主成分分析。nb\_cluster到9之后优化效果就不显著了，到20之后效果开始下滑。nb\_cluster取值可以考虑9~20

变化度：今日某股票所在簇的成员股票与昨日不同（增、减、改），变化度加一 （这里不能反映一次变化的程度，如果需要可以在算法上优化）

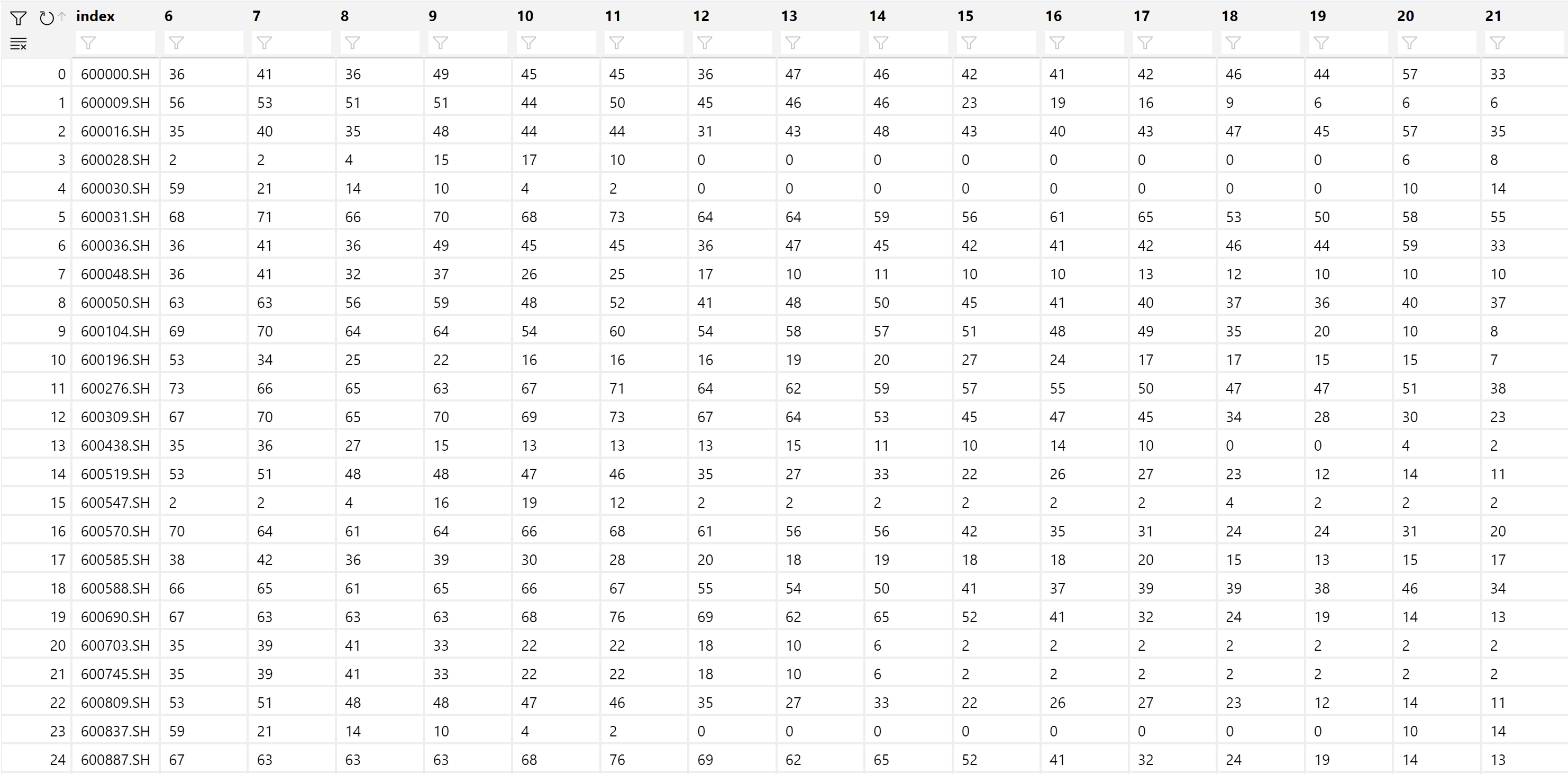
window length = 200;nb\_cluster取6~21；做200次rolling实验，统计每支股票的变化度，测试结果入下：

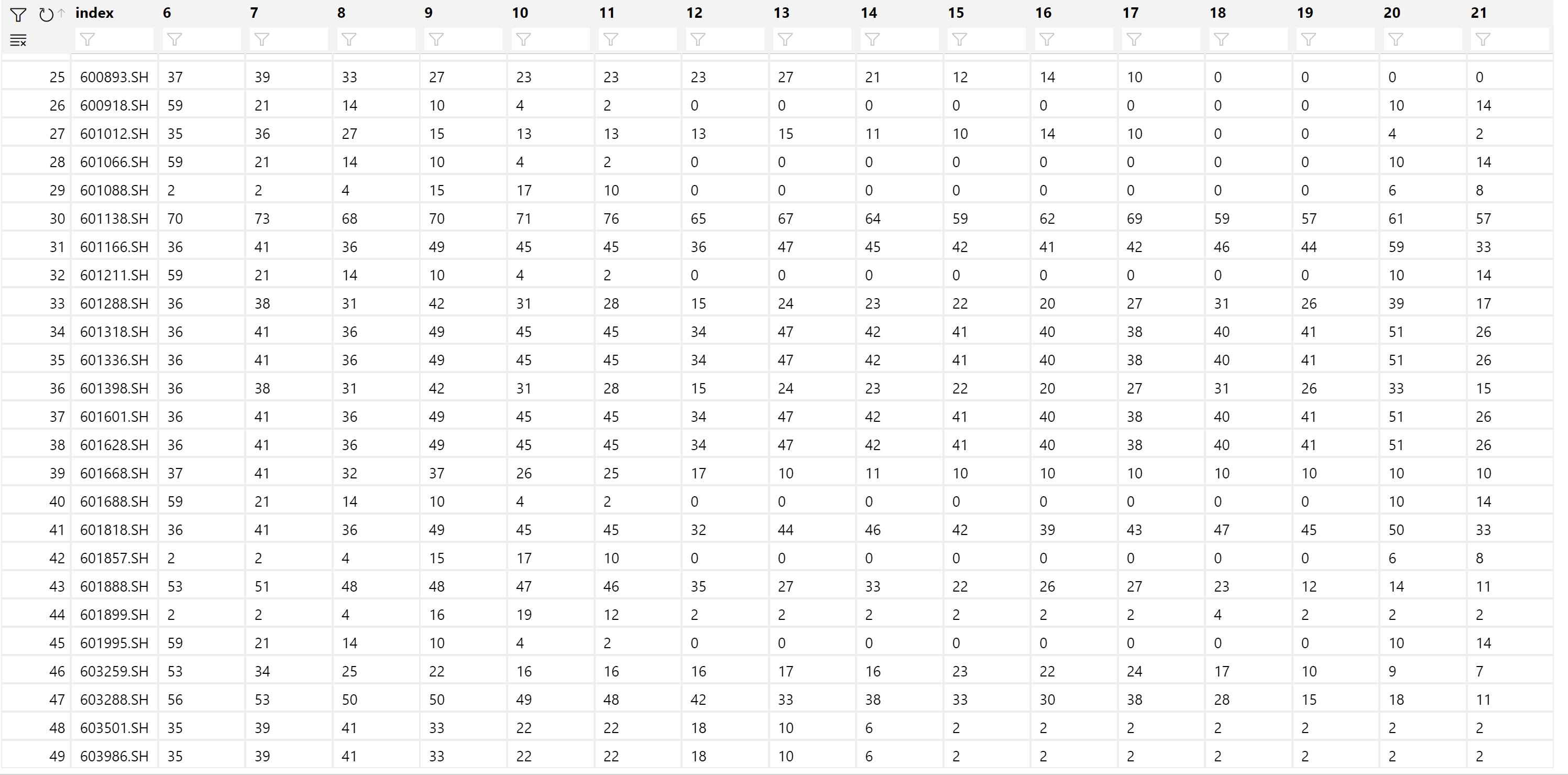


希望变化度更小一点，nb\_cluster=19时 25%分位数、50%分位数表现极优。

综合考虑，取nb\_cluster=19

变化度详细实验结果：



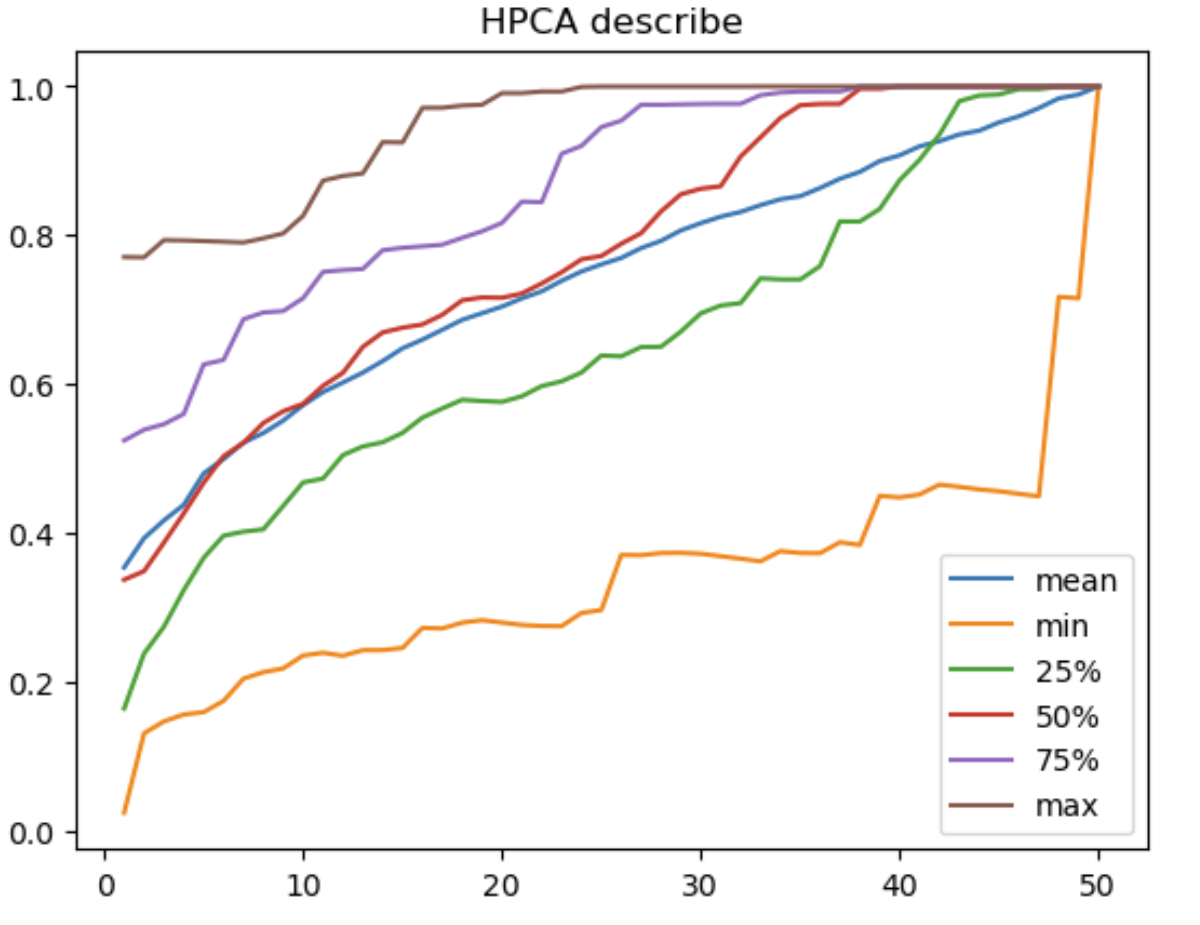
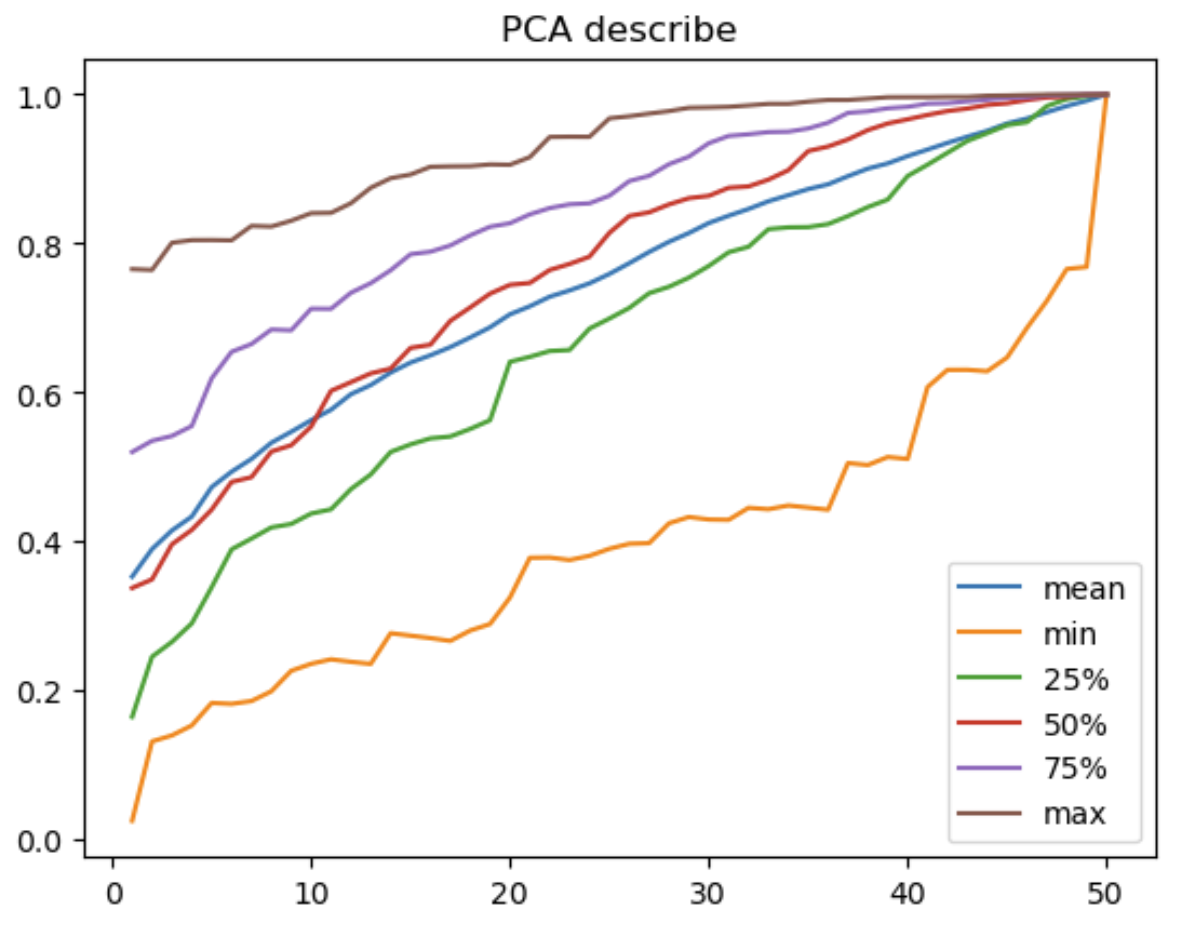


1. **确定n\_component**

ret除以标准差，右乘特征向量，得到特征组合收益率F

F和ret做多元线性回归，得到rsquared\_adj作为检验数，衡量数据适合做多元线性回归。（这里用rsquared\_adj改进了rsquared，仍然变量越多，r^2越大）

rsquared\_adj作为value；股票名作为index；n\_component作为col 画pd.T.describe()图像



上面呈现（20220512-20230306）的结果，其他时间段的实验也类似，结果是稳定的。

在这组实验中PCA和HPCA的表现差不多，都需要选取较多的主成分

然后就在想：选取较多数量的主成分的劣势在哪里？一般做PCA是希望得到低维数据以方便观察分析预测。但在当前的模型中，主成分通过一些数学变换后 作为自变量输入到回归模型中，属于中间过程，似乎不用分析预测等操作。（当然较多数量的主成分会花更多的时间去计算，不过这似乎不重要，不知道老师对主成分数量选多选少有什么看法）

当然rsquared\_adj检验标准过于单一，可能会导致较低数量的主成分的优势被掩盖

1. **阈值选取 （选股方法）**

zscores是降维+回归后的结果数据 根据结果数据按策略选股

阈值选取—直接影响选股数量；对权重影响不大，因为所选股票的zscores一般差别不大