基于 Markowitz 模型和遗传算法的统计性选股模型

作者：马世拓

院系：计算机科学与技术

班级：计算机 1901

学号：U201914900

摘要

股票的投资组合与评价一直是经济学领域一个经典问题，而对于此类问题有一些经典解法。本问题基于给定的 20 支股票进行了选股组合策略的建模与分析。

对于问题一给定的模型，这是一个评价问题。所衡量的要素大致分为两个，股票的收益和投资的风险。经过查阅文献，传统的层次分析法和模糊评价法并不是最适合本问题的解法。基于 Markowitz **模型**，我们使用统计学方法进行衡量，利用变异系数构建**改进夏普比率模型**进行评价标准的综合衡量。同时，为了形成对比，还使用了 TOPSIS **熵权模型**进行综合评估。最终得到的排名前五的股票分别为：迈瑞医疗，英科医疗，海天味业，绝味食品和美的集团。

问题二则是一个组合优化问题。对于这一问题，原始的优化目标有二：极小化风险和极大化收益，那么构造在不同投资策略下一个以每支股票的权重为系数的有关日收益率的随机变量，用均值描述其收益，方差描述其风险，构造一个**凸优化模型**，使用**遗传算法**进行求解。问题探究了购买股票种类数和遗传算法迭代次数对结果的影响。

问题三本质上是对问题一的一种补充，均值和标准差是 Markowitz **模型**中衡量收益和风险两个比较基本的统计量。这里从分布的观念，还用了偏度和波动率作为衡量统计量，构造“**偏度-波动率模型**”与问题一的统计结果进行了对比，分析大体相同，但在上汽、格力等股票上有一定程度的分歧，最后对分歧的原因作了简单探讨。

模型以统计方法和**凸优化模型**求解作为其数学基础，遵循基本的经济学理论，具有充分的可靠性；利用进化计算算法来程序设计得到结论非常方便，从而保证了求解结果的准确性和高效性。这一模型和相应建模手段也在实际经济学问题中有着重要应用。

关键词：Markowitz 模型，改进夏普比率模型，TOPSIS 熵权模型，他优化模型，遗传算法，偏度-波动率模型

一. 问题重述

股票的组合与风险评估一直是股市选择中的一个重要因素。而随着后疫情时代的到来，我国资本市场的发展和证券交易规模不断扩大，越来越多的资金投资于证券市场，与此同时市场价格的波动也十分剧烈，而波动作为证券市场中最本质的属性和特征。市场的波动对于人们风险收益的分析、股东权益最大化和监管层的有效监管都有着至关重要的作用，因此研究证券市场波动的规律性，分析引起市场波动的成因，是证券市场理论研究和实证分析的重要内容，也可以为投资者、监管者和上市公司等提供有迹可循的依据。对于题目给定的二十支股票的数据，要求解决以下问题：

1. 通过建立模型，从上述 20 支股票中，筛选出你认为最有潜力的 5 支股票。

2. 如果你有 1 亿元的初始资金，通过建立模型，请给出合理选股方案和投资组合方案。

3. 试给出合理的评价指标来评估投资策略的风险与预期年收益。

二. 问题分析

2.1 问题一的分析

对于问题一，这是一个股票的潜力筛选评价问题。评价股票通常是观察它的收益和风险，若使用层次分析法或模糊评价法，有一些指标的获取并不容易。因此，这里采取统计学策略，利用统计参量描述数据的收益和风险。

首先，对于股市分析而言，最重要的变量之一就是日收益率，它可以由收盘价得到。我们分连续性日收益率和离散性日收益率来处理。根据 Markowitz 模型提出的“均值-方差方法”，我们可以衡量日收益率的均值和方差，并建立改进后的夏普比率模型，使用变异系数作为评价标准选择最优的五支股票。

2.2 问题二的分析

问题二是一个典型的投资组合问题，这里将它抽象为一个优化问题来解决。从第一问的结果来看，均值为负数的股票会亏因此不买它们，最多只买均值为正数的股票。我们优化的策略就是在固定风险水平下收益期望最高，或者固定收益期望下风险最低，限制条件为所有股票的购买金额总和为 1 亿。使用统计均值描述期望，用协方差描述风险，这里由于股票的起止日期不同，故只截取了最近的一百天也就是前 100 条。

构造目标函数为一个分式函数，分子为方差而分母为期望，求在一个等式条件限制下的函数最小值。这一过程可以用遗传算法求解，寻找在购买多少支股票时风险最小，并搜索最优轮数。

2.3 问题三的分析

问题三可以说是对问题一的补充，从基本的统计学观点来看，均值和标准差已经可以描述期望收益和风险水平。但更进一步，若从统计到统计分布，我们可以将统计量改进为偏峰分布的偏度和市场的波动率，同样可以描述。

三. 模型假设

下面为我们的模型假设：

1. 假设从收盘价的动态变化过程可以衡量股票的基本情况

2. 投资者在考虑每一次投资选择时，其依据是某一持仓时间内的证券收益的概率分布。

3. 投资者是根据证券的期望收益率的方差或标准差估测证券组合的风险。

4. 投资者的决定仅仅是依据证券的风险和收益。

5. 在一定的风险水平上，投资者期望收益最大；相对应的是在一定的收益水平上，投资者希望风险最小。

假设 2-5 也正是 Markowitz 模型的假设，这一考虑从投资者心理考虑，然后假定了使用统计量，能更好地衡量基本水平。

四. 符号约定

下面为文中用到的一些变量符号：

符号	说明
μ	均值
σ	标准差
$E(X)$	数学期望
$D(X)$	方差
w_i	权值
r_t	日收益率，上标 c 则为连续，d 为离散
p_t	收盘价
S	夏普比率
$Skew$	偏度
p_{ij}	TOPSIS 矩阵的对应元素
e_j	熵值
CV	变异系数
Σ	协方差矩阵

（若出现不在表格中的符号，具体含义以文中解释为准）

五. 模型的建立与求解

5.1 模型一的建立与求解

问题一需要的模型是股票的评价模型。这里并不方便使用层次分析法或模糊评价策略，因为通常评价股票所考虑的变量并不完全能通过题给数据得到。那么可以考虑使用统计学方法衡量，文献[1]指出根据收盘价得到日收益率可以衡量股票的收益和风险情况。

5.1.1 求解日收益率和累计日收益率

日收益率可以有离散性日收益率和连续性日收益率两种，离散性日收益率被定义为收盘价的增量相比上一天的比值：

$$r_t^{(d)} = \frac{p_t - p_{t-1}}{p_{t-1}} \tag{1}$$

连续性日收益率定义为：

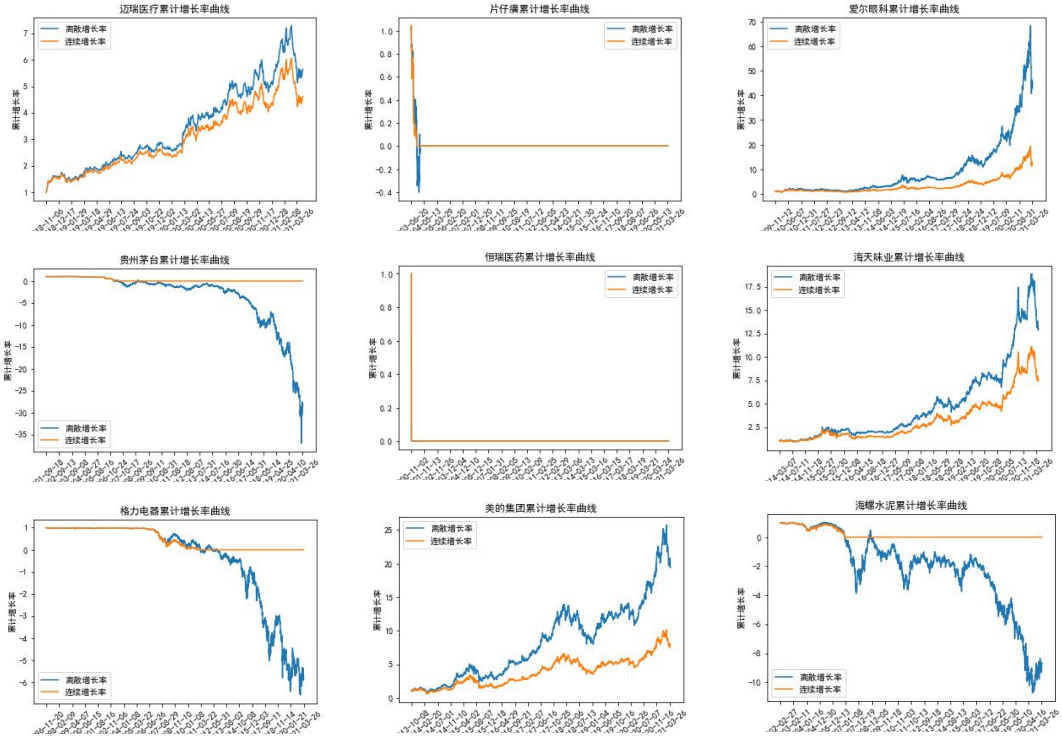
$$r_t^{(c)} = \ln \frac{p_t}{p_{t-1}} \tag{2}$$

当增长比较小的时候（即前后两天的收盘价接近时），根据极限理论可以得到二者基本相近。此外，有可能产生分母为 0 或者产生负数的对数这样的异常。对于这样的异常情况为简便起见，将收益率定义为-1。

累计日收益率被定义为这样一个连乘形式：

$$R = \prod_{t=1}^{n}(1 + r_t) \tag{3}$$

对题目给定的二十支股票求解累计日收益率，将其图像绘制如图 1 所示：



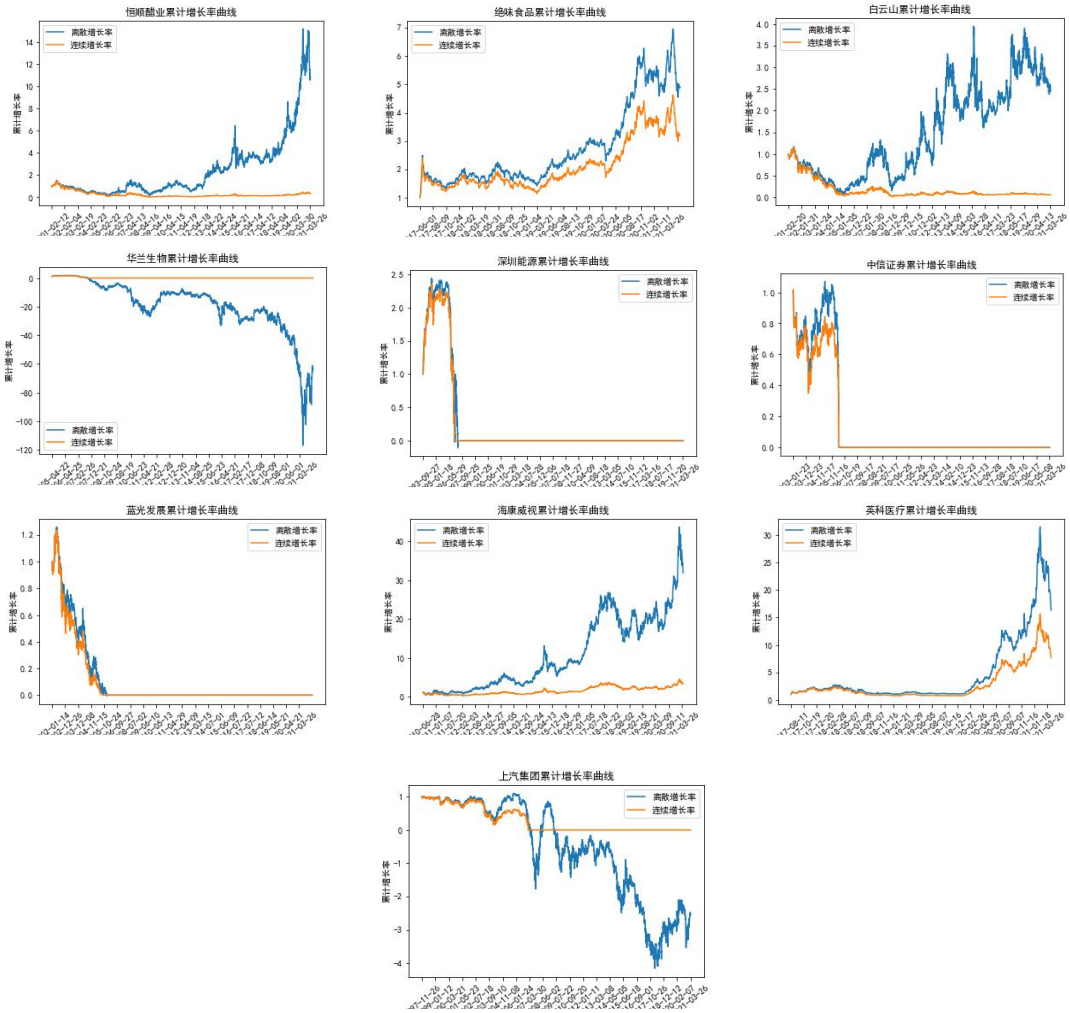


图 1.20 支股票的累计日收益率，蓝色表示离散性，黄色表示连续性

从图中可以看到对于连续性增长率而言很多都是-1，而这一操作本是针对少量异常情况做出的处理，但如果-1 太多会严重影响进一步的统计分析。故选择离散日收益率作为评价标准。另外可以发现，离散性收益率普遍会比连续性日收益率高，这是因为对数函数的增长速度是远弱于线性的，当增幅越大时线性增长和对数增长之间的差别就会越发明显。

5.1.2 日收益率的统计特征

为了更好地衡量累计日收益率，我们引入了几何平均值的概念。几何平均值基于这样一个事实：使用 50 元进行投资，第一天收盘价 100 元第二天收盘价 50 元，实际资产是没有增长的，但算术平均值结果为正数。这是因为收益实际上并非一个累加效应而是累乘效应，故几何平均日收益率被定义为：

$$\bar{r} = \sqrt[n]{\prod_{t=1}^{n}(1+r_t)} - 1 \tag{4}$$

为避免出现负数开偶次方根的情况，规定若负数开偶次方根，则为其相反数开偶次方根以后加上负号。

这些股票累计日收益率的统计特征如表 1 和图 2 的箱线图表示：

表 1. 累计日收益率的统计特征

股票	算术平均值	几何平均值	标准差
迈瑞医疗	0.003267	0.002912	0.026164
片仔癀	-0.000763	-1	0.248776
爱尔眼科	0.00187	0.001389	0.030747
贵州茅台	0.000454	-2.000713	0.710522
恒瑞医药	-0.009273	-1	0.244733
海天味业	0.001803	0.001477	0.025188
格力电器	0.006634	-2.000309	0.667769
美的集团	0.002224	0.001679	0.032845
海螺水泥	-0.001634	-2.000472	0.247581
恒顺醋业	0.001194	0.000493	0.037189
绝味食品	0.002119	0.001620	0.029676
白云山	0.001002	0.000192	0.040366
华兰生物	0.001308	-2.001037	0.075554
深圳能源	-0.001110	-1.000000	0.198421
中信证券	-0.006557	-1.000000	0.350147
蓝光发展	-0.002324	-1.000000	0.263102
海康威视	0.002235	0.001344	0.041599
英科医疗	0.004012	0.003127	0.041302
上汽集团	-0.002299	-2.000164	0.565073
蓝思科技	0.001795	0.001055	0.037946

箱线图如图 2：

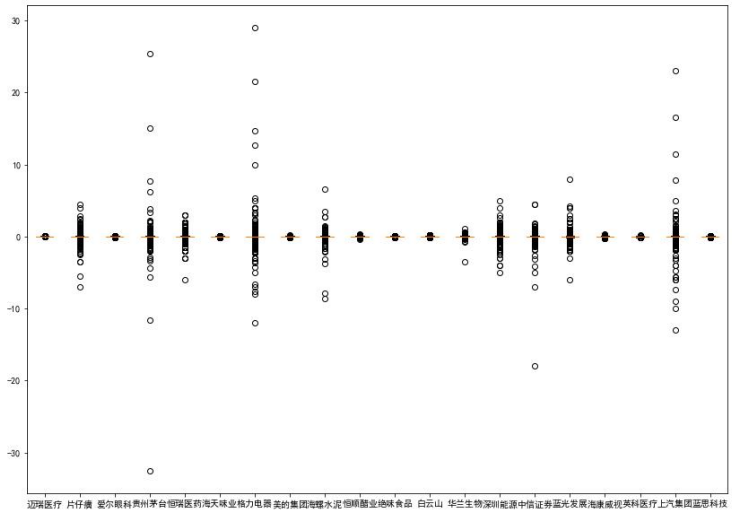


图 2. 日收益率的箱线图

可以看到，这些股票的日收益率都比较集中，但离群点也不少。由于标准差不大，均值都接近 0 所以箱体极窄。对于片仔癀、贵州茅台、格力电器等企业，离群点较多说明风险较大。

平均值反映了日收益率的平均水平，也可以说是一种期望。平均的收益率越高，可以说明基本盈利能力越强；而标准差可以反映市场的波动程度，描述的是风险水平，标准差越大则需要承担的风险越高。

从几何平均值的情况来看，这些股票中有 10 支几何平均值都为负数说明基本是以亏损居多，不建议投资；从标准差的情况来看，标准差越大则表示承担的风险也越大。

我们希望盈利越多，风险越小，那么为了综合衡量二者，就需要使用新的统计量来综合衡量。

5.1.3 Markowitz 模型的提出

Markowitz 模型于 1953 年被提出[2]，又被称为“均值-方差模型”。在期初，他购买一些证券，然后在期末卖出。那么在期初他要决定购买哪些证券以及资金在这些证券上如何分配，也就是说投资者需要在期初从所有可能的证券组合中选择一个最优的组合。这时投资者的决策目标有两个：尽可能高的收益率和尽可能低的不确定性风险。最好的目标应是使这两个相互制约的目标达到最佳平衡。由此建立起来的投资模型即为均值-方差模型。

经典的 Markowitz 模型的形式如下：

$$\begin{cases} \min\limits_{w} \ D\left(\sum\limits_{i=1}^{N} w_i r_i\right) \\ \max\limits_{w} \ E\left(\sum\limits_{i=1}^{N} w_i r_i\right) \\ \sum\limits_{i=1}^{N} w_i = 1 \end{cases} \tag{5}$$

5.1.4 基于变异系数的改进夏普比率模型

由于需要同时衡量风险和收益二者，我们使用日收益率的均值来表示收益，用标准差表示风险，分别对二者进行观察：

为了同时考虑二者，我们使用改进的夏普比率模型来描述。原始的夏普比率模型描述如下[3]：夏普比率就是衡量在每承担 1 个单位风险的情况下，所获得超越无风险收益率的超额回报是多少。夏普比率越高，说明在承担一定风险的情况下，所获得的超额回报越高。反之，如果夏普比率很小甚至为负，说明承担一定的风险所获的超额回报很小或者没有超额回报。公式为：

$$S = \frac{E(r) - r_f}{\sigma_r} \tag{6}$$

这里我们假定无风险利率为 0，问题将会得到进一步简化。

变异系数是指[4]概率分布离散程度的一个归一化量度，其定义为标准差与平均值之比。变异系数（coefficient of variation）只在平均值不为零时有定义，而且一般适用于平均值大于零的情况。变异系数也被称为标准离差率或单位风险。

变异系数的公式如下：

$$CV = \frac{\sigma}{\mu} \tag{7}$$

这一结果刚好是夏普比率的倒数（在无风险利率为 0 的情况下），而且具备统计学意义，因此将其作为新的统计学参量构建改进夏普比率模型。

将十支均值为正值的股票风险与收益散点图绘制如图 3 所示：

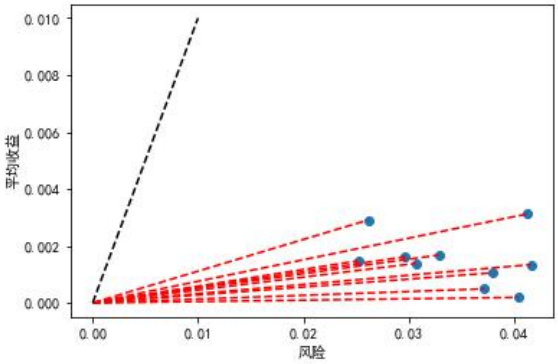


图 3. 股票风险与收益散点图

红色线段的斜率表示变异系数。一般而言，变异系数若为正数，越大则表明风险相对于收益越高，不建议选择。那么根据斜率，选出最优的五支股票为：迈瑞医疗，英科医疗，海天味业，绝味食品和美的集团。

其它所有股票的变异系数见附录。

5.1.5 基于 TOPSIS-熵权分析法的综合评价模型

为了与改进夏普比率模型进行对比，这里还使用了基于 TOPSIS-熵权分析法的模型。TOPSIS 方法是多目标决策分析中一种常用的有效方法，是一种逼近于理想解的排序法，又称为优劣解距离法。它根据有限个评价对象与理想化目标的接近程度进行排序，在现有的对象中进行相对优劣的评价[5]。

首先，对表一中的数据进行归一化处理，这里选择 min-max 归一化方法。

其次，对几何平均值和标准差利用熵权法确定权重。熵权法基于信息论，基于信息论的熵值法是根据各指标所含信息有序程度的差异性来确定指标权重的客观赋权方法，仅依赖于数据本身的离散程度。熵用于度量不确定性，指标的离散程度越大（不确定性越大）则熵值越大，表明指标值提供的信息量越多，则该指标的权重也应越大。其算法如下：

1. 对原始数据矩阵按列进行归一化处理
2. 计算熵值：

$$e_j = -\frac{\sum_{i=1}^{n} p_{ij} \ln p_{ij}}{\ln n} \tag{8}$$

3. 计算权重系数

$$w_j = \frac{1 - e_j}{\sum_{i=1}^{m} (1 - e_i)} \tag{9}$$

最终解得权重分别为 0.953 和 0.047。

最后，根据权重计算距离，评价与最优方案和最劣方案的接近程度并排序，

结果如表 2 所示：

表 2. TOPSIS 评价结果

股票	几何平均值	标准差	正理想解	负理想解	综合得分指数	排序
迈瑞医疗	0.610055022	0.231493545	0.033566885	0.604250575	0.94737227	2
片仔癀	0	0.26253184	0.6360844	0.02920856	0.04390330 6	7
爱尔眼科	0	0.226163158	0.63614349	0.023990526	0.03634190 2	12
贵州茅台	0.004848688	0.156517579	0.631579879	0.014797681	0.02289324 6	19
恒瑞医药	0	0.204632131	0.636198638	0.02090134	0.03180846 2	16
海天味业	0.292519742	0.172062361	0.347019848	0.289947778	0.45520018	4
格力电器	0	0.058953863	0.63696538	0	0	20
美的集团	0.313533144	0.226439652	0.325925092	0.311218377	0.48845886 7	3
海螺水泥	0	0.219774435	0.636158289	0.023073897	0.03500116 9	13
恒顺醋业	0	0.237471316	0.636120532	0.025612976	0.03870587 7	9
绝味食品	0	0.231383922	0.636132377	0.024739581	0.03743475 6	11
白云山	0	0.23142305	0.636132297	0.024745194	0.03744293 7	10
华兰生物	0	0.217682044	0.636163423	0.022773689	0.03456124 8	14
深圳能源	0	0.203414844	0.636202204	0.020726688	0.03155088 5	17
中信证券	0	0.206308616	0.636193806	0.021141875	0.03216298 1	15
蓝光发展	0	0.269404233	0.636078043	0.030194584	0.04531866 2	6
海康威视	0.175649502	0.274966787	0.462244155	0.176573436	0.27640665 9	5
英科医疗	0.642717496	0.294553837	0	0.63696538	1	1
上汽集团	0	0.180918477	0.636276728	0.017498999	0.02676605 8	18
蓝思科技	0	0.254590426	0.636093651	0.028069158	0.04226246 5	8

可以看到，综合评价的前五的股票分别为英科医疗，迈瑞医疗，美的集团，海天味业和海康威视，基本相近只是顺序不同。

5.2 模型二的建立与求解

股票投资组合是一个经典的组合优化问题，这一问题我们基于优化方法进行求解。类比线性判别模型的建立过程[6]，LDA 模型需要将数据进行降维映射使其类内散度低而类间散度高，基于多个类之间的均值和方差抽象出一个凸优化问题，那么这里同样可以利用均值和方差建立优化模型。

股票投资组合基于这样一个事实：不要把鸡蛋装在一个篮子里。若将自己的资产集中投资一支股票，承担风险和亏损水平往往也就越高。

这里为了同时衡量风险与收益，也采用类似的方法。

5.2.1 建立函数优化模型

构造随机变量 X，对于分配给若干支股票的资金，其加权收益为：

$$X(w)=w_1r_1+w_2r_2+\cdots+w_nr_n \tag{10}$$

对于股票收益的衡量，采用数学期望。由于 r 和 w 都是随机变量，采用主元方法，将期望表示为关于 w 的函数：

$$E=E(X)=\sum_{i=1}^{N}w_iE(r_i)=w^T\mu \tag{11}$$

对于股票风险的衡量，采用协方差矩阵描述。这里由于协方差矩阵是矩阵的形式，我们将风险也用矩阵形式表达：

$$V=D(X)=w^T\Sigma w \tag{12}$$

限制条件：

$$w_1+w_2+\cdots w_n=1 \tag{13}$$

类比广义瑞利商，构造这样一个关于各项投资金额的目标函数：分子为需要极小化的风险，分母为需要极大化的收益，实际上就是改进的变异系数。这样我们就可以同时考虑二者，用一个函数描述而非多目标规划。

凸优化问题可以表示为：

$$\min_{w}\ J(w)=\frac{w^T\Sigma w}{w^T\mu}$$

$$s.t.\begin{cases} w_i\geqslant 0,\ i=1,2,\cdots N \\ \sum_{i=1}^{N}w_i=1 \end{cases} \tag{14}$$

5.2.2 使用遗传算法求解优化模型

遗传算法是模拟达尔文生物进化论的自然选择和遗传学机理的生物进化过程的计算模型，是一种通过模拟自然进化过程搜索最优解的方法。其主要特点是直接对结构对象进行操作，不存在求导和函数连续性的限定；具有内在的隐并行性和更好的全局寻优能力；采用概率化的寻优方法，不需要确定的规则就能自动获取和指导优化的搜索空间，自适应地调整搜索方向。遗传算法以一种群体中的所有个体为对象，并利用随机化技术指导对一个被编码的参数空间进行高效搜索。其中，选择、交叉和变异构成了遗传算法的遗传操作；参数编码、初始群体

的设定、适应度函数的设计、遗传操作设计、控制参数设定五个要素组成了遗传算法的核心内容[7]。

遗传算法的流程图如图 4 所示：

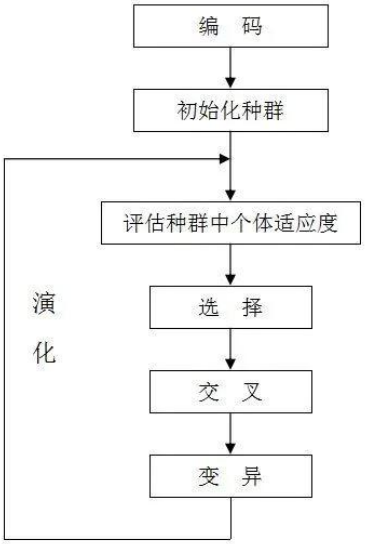


图 4. 遗传算法的流程图

分别选取 1-10 支均值为正值的股票进行选择，同样设置迭代次数为 1000 次，得到的投资金额和函数值如表 3 所示：

表 3. 投资权重与函数值关系

投资数	迈瑞	英科	海天	绝味	美的	爱尔	海康	蓝思	恒顺	白云山	最小函数值
1	1										8.67
2	0.789	0.211									5.61
3	0.282	0.062	0.656								2.86
4	0.026	0.227	0.552	0.195							2.69
5	0.339	0.012	0.426	0.106	0.118						2.52
6	0.015	0.249	0.095	0.178	0.338	0.125					2.31
7	0.051	0.051	0.176	0.366	0.226	0.08	0.05				1.17
8	0.107	0.005	0.110	0.079	0.155	0.226	0.189	0.13			1.07
9	0.04	0.11	0.05	0.00	0.04	0.08	0.08	0.08	0.463		0.9

		3	6	7	2	1	3	3	3		6
	10	0.08 7	0.04 3	0.19 7	0.11 5	0.00 2	0.20 8	0.17 3	0.13 2	0.000 1	0.04 2	0.9 3

可以看到，股票种类数越多则函数值越小，方案越优。但随着股票数增加，对目标函数效果提升越来越缓慢，如图 5 所示：

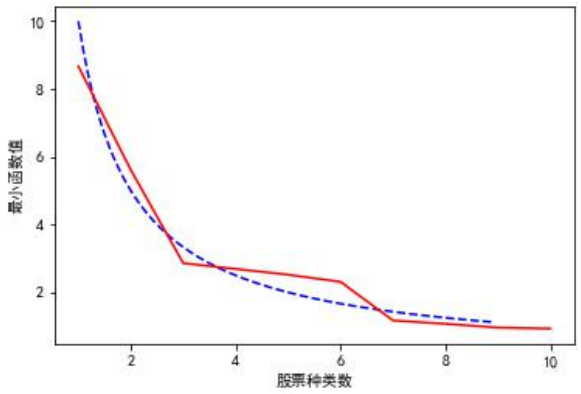


图 5. 最小函数值会随着股票种类数增加而减小，但效果变慢

接下来探讨迭代次数和函数值、时间的关系，经计算，迭代次数和函数值、时间关系如图 6 所示。可以看到，随着迭代次数增加，时间近乎是线性增长，而最小函数值处于一个波动状态。最后可以看到迭代次数 1000 为最合适的。

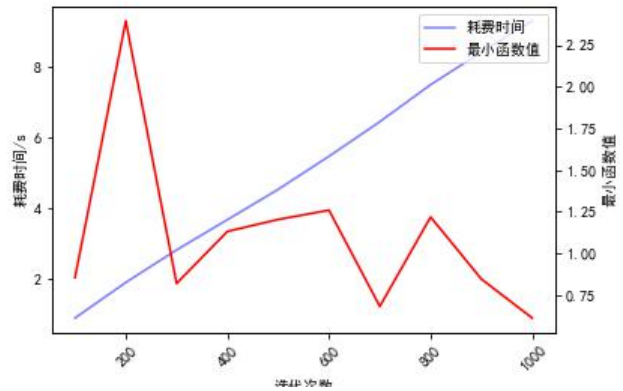


图 6. 迭代次数和耗费时间、最小函数值的关系

不同迭代次数下的投资金额绘制热力图如图 7

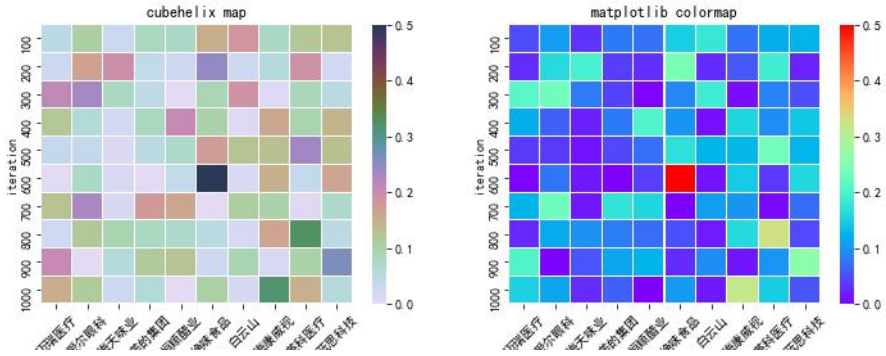


图 7. 不同股票的投资金额

5.3 模型三的建立与求解

5.3.1 基本的统计参量——均值，标准差

利用均值和标准差可以描述收益的平均水平与承担风险。前面的建模过程便使用了均值描述收益水平，标准差描述风险。

从模型一的数据中来看，离散性日收益率的几何平均值最能反映平均水准，而且很明显，当平均收益为负数时说明它基本上很容易处于亏损状态。一共有十支股票是基本亏损的，其中包括很多人热买的茅台等。

标准差是方差的算术平方根，这两个量都可以描述风险。从数据中可以看出，茅台、格力等股票的风险比较大，所以即使平均水平亏损但有可能盈利也会很大。只是要承担风险较高。

5.3.2 改进的统计参量——偏度，波动率

偏度是指随机变量概率分布的不对称性，是相对于平均值下对称程度的度量。偏度为零表示数值相对均匀的分布在平均值的两侧，但不一定意味着一定是对称分布[8]。若峰度为正，则说明更多样本分布在正值，平均盈利水平更高。这一统计量从样本分布的角度提出解释，比基本的统计量更进一步。

偏度的统计学定义为：

$$Skew = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^3}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2\right)^{\frac{3}{2}}} \tag{15}$$

利用程序设计方法可以解得偏度分别如图 8 所示：

```
迈瑞医疗日收益率偏度:  0.4248895669744356
片仔癀日收益率偏度:  -6.431497628748267
爱尔眼科日收益率偏度:  0.15010210745660468
贵州茅台日收益率偏度:  -9.440057860923055
恒瑞医药日收益率偏度:  -3.370198900139399
海天味业日收益率偏度:  0.22612945677556964
格力电器日收益率偏度:  22.072549251208187
美的集团日收益率偏度:  0.43280707844224425
海螺水泥日收益率偏度:  -12.556049314204362
恒顺醋业日收益率偏度:  -0.17131601148600473
绝味食品日收益率偏度:  0.43831089466364664
白云山日收益率偏度:  0.4579656629407939
华兰生物日收益率偏度:  -23.910527839693927
深圳能源日收益率偏度:  -2.0210244047821306
中信证券日收益率偏度:  -33.49329241302319
蓝光发展日收益率偏度:  6.100274576224784
海康威视日收益率偏度:  0.10683238495763804
英科医疗日收益率偏度:  0.35340070217112685
上汽集团日收益率偏度:  13.446747325133689
蓝思科技日收益率偏度:  0.252660197167851
```

图 8.20 支股票的收益率偏度

可以看到，我们通过均值筛选的股票和基于偏度的基本一致，除了上汽集团和格力电器。这可能是由于其一旦发生亏损则情况严重，也就是风险较高的缘故导致。

波动率定义为定义为这一变量在单位时间内连续复利回报率的标准差。当波动率被用于期权定价时，时间单位通常为一年，因此波动率就是一年连续复利回报率的标准差，但是当波动率用于风险控制时， 时间单位通常是一天，此时的波动率对应于每天连续复利回报率的标准差[9]。

分别取周期为 15 天、30 天和 45 天，波动率图像如图 9 所示

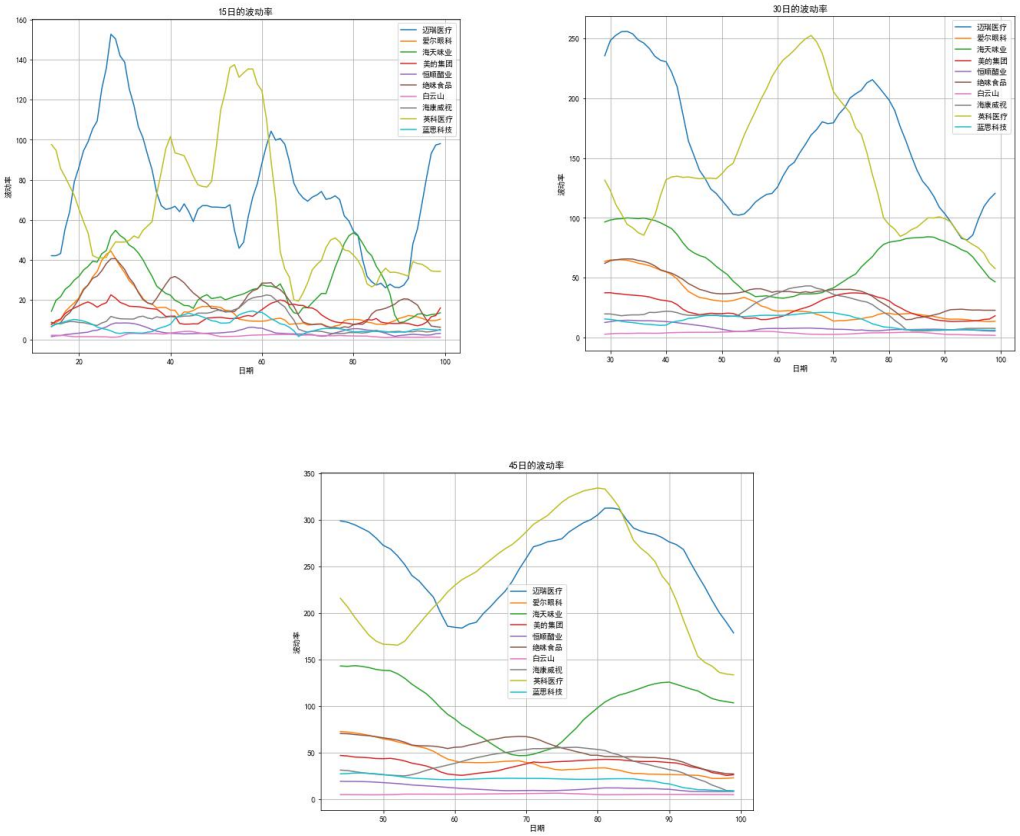


图 9. 不同周期下的波动率曲线

可以看到一个很有趣的现象，波动率越小的股票其变异系数反而越高。这是由于变异系数综合考虑了风险和收益两个因素，风险小的股票收益往往却更小，更难获取更好的收益，所以即使风险小也不容易盈利。

六. 模型的分析与改进

6.1 模型的优点

本模型的优点包括：

1. 从统计学角度出发建模，参照经济学概念的 Markowitz 模型，原理上虽然朴素但是能切实解决问题。
2. 在统计模型的同时还采取了 TOPSIS 模型进行对比，对结果进行了进一步讨论。
3. 采用优化模型+遗传算法，原理上有保证
4. 能够很容易地编程求解。

6.2 模型的缺点

而该模型也有一些缺点：

1. 未能充分考虑其它因素的影响，只是使用了日收益率这一个参量。
2. 由于市场具备一定的风险性和波动性，得到结果仍然有亏损可能。
3. 未能针对每个时间节点，利用时间序列预测方法逐一进行方案评估。

参考文献

[1]张宇,张世诺.浅析余额宝日收益率的波动研究[J].商场现代化,2021,{4}(11):123-125.
[2]姜玮怡.Markowitz均值-方差模型与RAROC模型在中国证券市场的实证研究[J]. 经济研究导刊, 2010, 000(002):103-106.
[3]余润, 陈汉军, 刘春章. 复夏普比率理论及其在中国的检验——一种考虑估计风险的投资组合业绩评价指标[J]. 技术经济与管理研究, 2003(01):56-57.
李超群,刘智慧,张玉洁.质心公式推导及其在求解积分中的应用[J].数学学习与研究,2014,{4}(01):69-70.
[4]王文森. 变异系数——一个衡量离散程度简单而有用的统计指标[J]. 中国统计, 2007, 2007(006):41-42.朱宝奇,苏煜.等效替换物理思想解决刚体定轴转动问题[J].物理通报,2021,{4}(S1):4-7+11.
[5]Chen C T . Extensions of the TOPSIS for group decision-making under fuzzy environment[J]. Fuzzy Sets & Systems, 2000, 114(1):1-9.
[6]周志华. 《机器学习》[J]. 中国民商, 2016, 03(No.21):93-93.
[7]葛继科, 邱玉辉, 吴春明,等. 遗传算法研究综述[J]. 計算機應用研究, 2008, 25(010):2911-2916.
[8]王学民. 偏度和峰度概念的认识误区[J]. 统计与决策, 2008, 000(012):145-146.
[9]石晓军, 陈殿左. 债权结构、波动率与信用风险——对中国上市公司的实证研究[J]. 财经研究, 2004(09):24-32.

附录

Environment: OS: Windows 10; CPU: Intel i7; GPU: NVIDIA GEFORCE 1650
Language:
Python 3.8.2 Jupyter notebook

In [1]:

```
import pandas as pd
data=[]
for i in range(1,21):
    dfi=pd.read_excel("附件：二十支股票重要参数.xlsx",sheet_name=i)
    data.append(dfi)
name=['迈瑞医疗','片仔癀','爱尔眼科','贵州茅台','恒瑞医药','海天味业','格力电器','美的集团','海螺水泥
      '华兰生物','深圳能源','中信证券','蓝光发展','海康威视','英科医疗','上汽集团','蓝思科技']
```

In [2]:

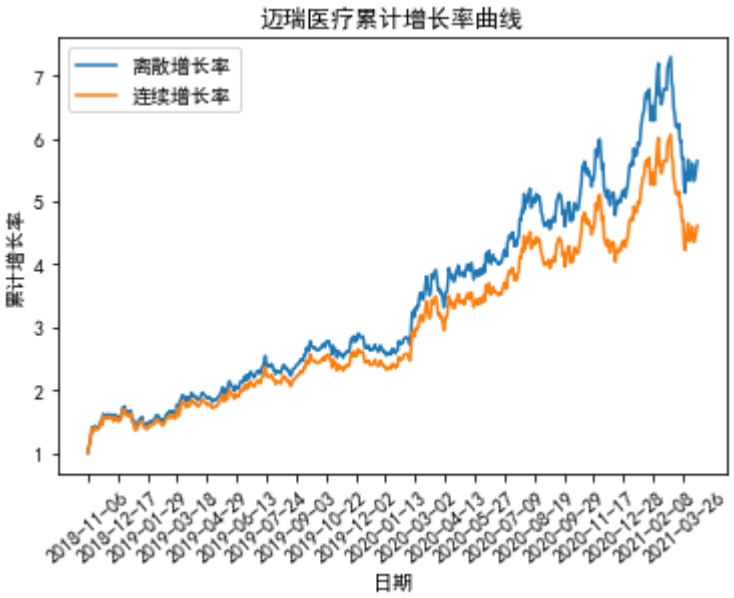
```
import numpy as np
import math
def daysy_dis(shouyi):
    daysyrate=np.zeros(len(shouyi)-1)
    for i in range(len(shouyi)-1):
        if shouyi[i+1]==0:
            daysyrate[i]=-1
        else:
            daysyrate[i]=(shouyi[i]-shouyi[i+1])/shouyi[i+1]
    return np.flipud(daysyrate)
def daysy_con(shouyi):
    daysyrate=np.zeros(len(shouyi)-1)
    for i in range(len(shouyi)-1):
        if shouyi[i]*shouyi[i+1]>0:
            daysyrate[i]=np.log(shouyi[i]/shouyi[i+1])
        else:
            daysyrate[i]=-1
    return np.flipud(daysyrate)
def mulsyrate(daysrate):
    mulsyrate=np.ones(len(daysrate))
    for i in range(len(daysrate)):
        for j in range(i):
            mulsyrate[i]*=(1+daysrate[j])
    return mulsyrate
def sim_ave(daysrate):
    return np.mean(daysrate)
def geo_ave(daysyrate):
    mulrate=mulsyrate(daysyrate)
    try:
        geo_ave=math.pow(mulrate[-1],1/(len(mulrate)))-1
    except:
        geo_ave=-math.pow(-mulrate[-1],1/(len(mulrate)))-1
    return geo_ave
```

In [3]:

```
rishouyilv_lisan=[]
rishouyilv_lianxu=[]
leijishouyilv_lisan=[]
leijishouyilv_lianxu=[]
for df in data:
    rishouyilv_lisan.append(daysy_dis(df['收盘价']))
    rishouyilv_lianxu.append(daysy_con(df['收盘价']))
for i in range(20):
    leijishouyilv_lisan.append(mulsyrate(rishouyilv_lisan[i]))
    leijishouyilv_lianxu.append(mulsyrate(rishouyilv_lianxu[i]))
```

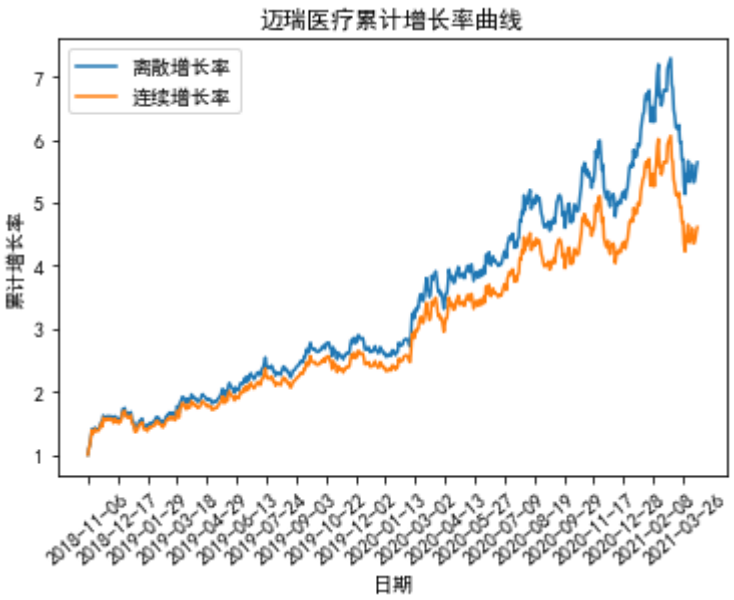
In [4]:

```
import matplotlib.pyplot as plt
plt.rcParams['font.sans-serif'] = ['SimHei'] # 用来正常显示中文标签
plt.rcParams['axes.unicode_minus'] = False # 用来正常显示负号
def plot_ljzzl(i):
    fig,ax=plt.subplots()
    xticks=range(0,len(data[i]['时间'][:-1]),len(data[i]['时间'][:-1])//20)
    xlabel=[data[i]['时间'][:-1][::-1][index] for index in xticks[::-1]]
    ax.set_xticks(xticks)
    ax.set_xticklabels(xlabel,rotation=40)
    plt.title(name[i]+"累计增长率曲线")
    plt.xlabel("日期")
    plt.ylabel("累计增长率")
    plt.plot(leijishouyilv_lisan[i])
    plt.plot(leijishouyilv_lianxu[i])
    plt.legend(['离散增长率','连续增长率'])
    plt.savefig("增长率/%d.png"%i)
    plt.show()
plot_ljzzl(0)
```



In [5]:

```
for i in range(i):
    plot_ljzzl(i)
```



In [6]:

```
for i in range(20):
    print(name[i]+"日收益率的算术平均值为:"+" %.4f,"%(np.mean(rishouyilv_lisan[i]))+" 几何平均值为:%
```

迈瑞医疗日收益率的算术平均值为: 0.0033, 几何平均值为:0.0029 ,方差为:0.0262
片仔癀日收益率的算术平均值为: -0.0008, 几何平均值为:-1.0000 ,方差为:0.2488
爱尔眼科日收益率的算术平均值为: 0.0019, 几何平均值为:0.0014 ,方差为:0.0307
贵州茅台日收益率的算术平均值为: 0.0005, 几何平均值为:-2.0007 ,方差为:0.7105
恒瑞医药日收益率的算术平均值为: -0.0093, 几何平均值为:-1.0000 ,方差为:0.2447
海天味业日收益率的算术平均值为: 0.0018, 几何平均值为:0.0015 ,方差为:0.0252
格力电器日收益率的算术平均值为: 0.0066, 几何平均值为:-2.0003 ,方差为:0.6678
美的集团日收益率的算术平均值为: 0.0022, 几何平均值为:0.0017 ,方差为:0.0328
海螺水泥日收益率的算术平均值为: -0.0016, 几何平均值为:-2.0005 ,方差为:0.2476
恒顺醋业日收益率的算术平均值为: 0.0012, 几何平均值为:0.0005 ,方差为:0.0372
绝味食品日收益率的算术平均值为: 0.0021, 几何平均值为:0.0016 ,方差为:0.0297
白云山日收益率的算术平均值为: 0.0010, 几何平均值为:0.0002 ,方差为:0.0404
华兰生物日收益率的算术平均值为: 0.0013, 几何平均值为:-2.0010 ,方差为:0.0756
深圳能源日收益率的算术平均值为: -0.0011, 几何平均值为:-1.0000 ,方差为:0.1984
中信证券日收益率的算术平均值为: -0.0066, 几何平均值为:-1.0000 ,方差为:0.3501
蓝光发展日收益率的算术平均值为: -0.0023, 几何平均值为:-1.0000 ,方差为:0.2631
海康威视日收益率的算术平均值为: 0.0022, 几何平均值为:0.0013 ,方差为:0.0416
英科医疗日收益率的算术平均值为: 0.0040, 几何平均值为:0.0031 ,方差为:0.0413
上汽集团日收益率的算术平均值为: -0.0023, 几何平均值为:-2.0002 ,方差为:0.5651
蓝思科技日收益率的算术平均值为: 0.0018, 几何平均值为:0.0011 ,方差为:0.0379

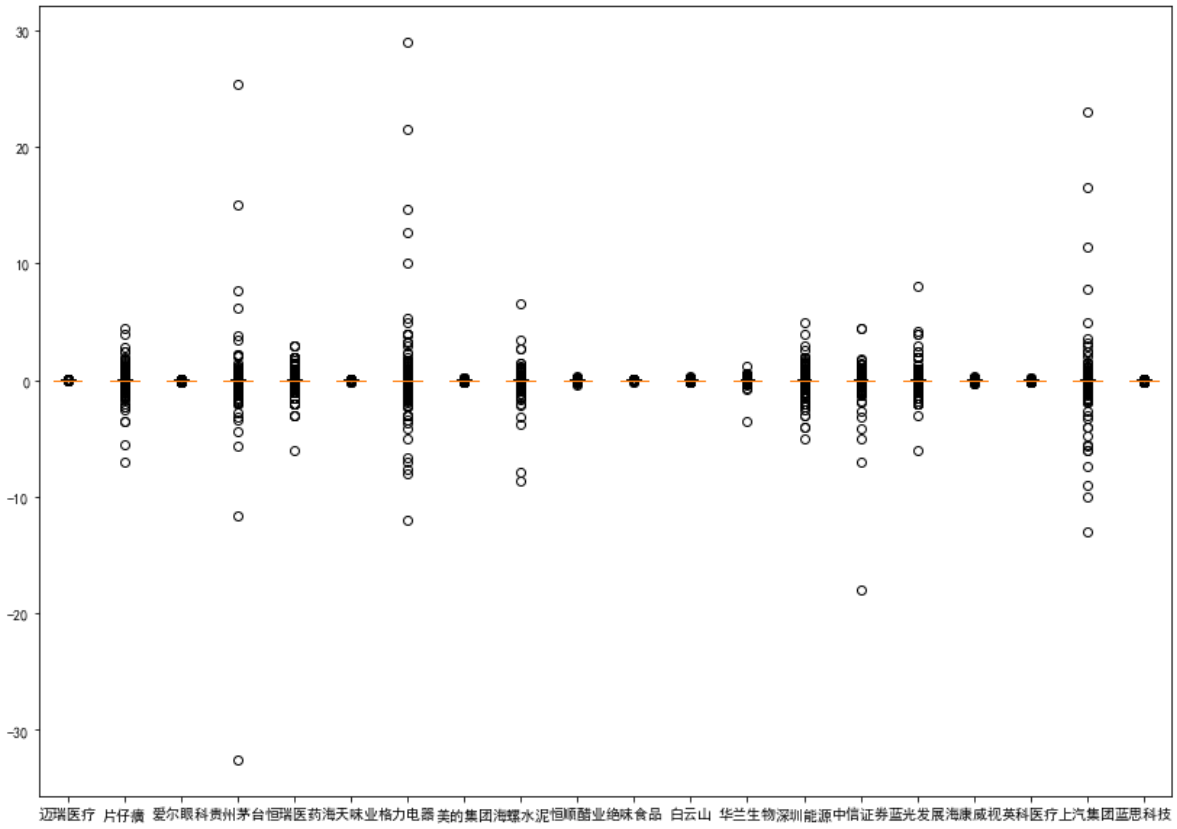
In [32]:

```
m=[]
for i in range(20):
    m.append(np.percentile(rishouyilv_lisan[i], (25, 50, 75), interpolation='midpoint'))
print(pd.DataFrame(m))
```

	0	1	2
0	-0.013011	0.002697	0.017724
1	-0.017974	0.000000	0.020101
2	-0.014121	0.000000	0.017316
3	-0.011306	0.000021	0.011984
4	-0.014728	-0.000000	0.015668
5	-0.011257	0.001293	0.013174
6	-0.006330	-0.000000	0.004514
7	-0.013917	0.001386	0.017337
8	-0.016622	-0.000000	0.016827
9	-0.015873	0.000000	0.018182
10	-0.015767	0.000000	0.017716
11	-0.018678	0.000000	0.017719
12	-0.014724	0.000000	0.016667
13	-0.016100	0.000000	0.015574
14	-0.015142	-0.000000	0.015796
15	-0.022021	0.000000	0.020627
16	-0.017742	0.000776	0.021053
17	-0.018846	0.002841	0.022552
18	-0.014164	-0.000000	0.013852
19	-0.018690	0.000000	0.019493

In [23]:

```
plt.figure(figsize=(14,10))
plt.boxplot(rishouyilv_lisan,labels=name,showbox=True,showcaps=True)
plt.savefig("箱线图.png")
plt.show()
```



In [8]:

```
def cv(zzl):
    return np.std(zzl)/geo_ave(zzl)
for i in range(20):
    print(name[i]+"日收益率的变异系数为:"+" %.4f"%cv(rishouyilv_lisan[i]))
```

迈瑞医疗日收益率的变异系数为: 8.9842
片仔癀日收益率的变异系数为: -0.2488
爱尔眼科日收益率的变异系数为: 22.1367
贵州茅台日收益率的变异系数为: -0.3551
恒瑞医药日收益率的变异系数为: -0.2447
海天味业日收益率的变异系数为: 17.0502
格力电器日收益率的变异系数为: -0.3338
美的集团日收益率的变异系数为: 19.5577
海螺水泥日收益率的变异系数为: -0.1238
恒顺醋业日收益率的变异系数为: 75.4878
绝味食品日收益率的变异系数为: 18.3146
白云山日收益率的变异系数为: 210.2056
华兰生物日收益率的变异系数为: -0.0378
深圳能源日收益率的变异系数为: -0.1984
中信证券日收益率的变异系数为: -0.3501
蓝光发展日收益率的变异系数为: -0.2631
海康威视日收益率的变异系数为: 30.9409
英科医疗日收益率的变异系数为: 13.2091
上汽集团日收益率的变异系数为: -0.2825
蓝思科技日收益率的变异系数为: 35.9714

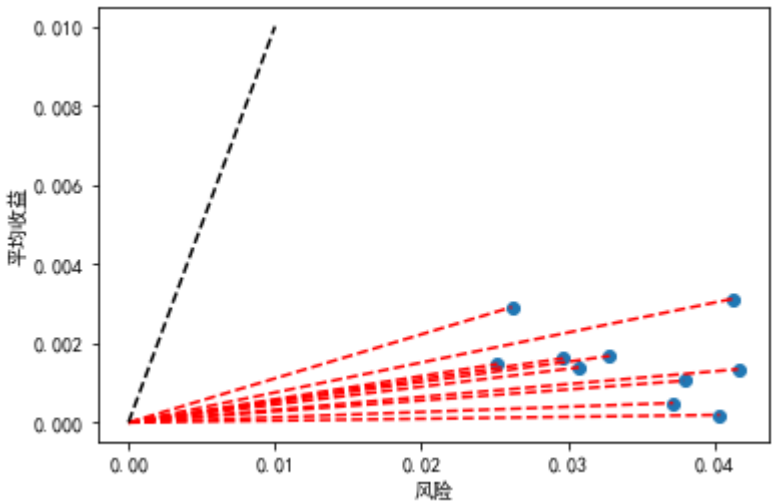
In [9]:

```
x=[]
y=[]
for i in range(20):
    if geo_ave(rishouyilv_lisan[i])>0:
        x.append(np.std(rishouyilv_lisan[i]))
        y.append(geo_ave(rishouyilv_lisan[i]))
        print(i)
```

```
0
2
5
7
9
10
11
16
17
19
```

In [10]:

```
plt.scatter(x,y)
for i in range(len(x)):
    plt.plot([0,x[i]],[0,y[i]],'r--')
plt.plot([0,0.01],[0,0.01],'k--')
plt.xlabel("风险")
plt.ylabel("平均收益")
plt.savefig("风险-收益散点图.png")
plt.show()
```



In [40]:

```
Newdf=[]
for i in [0,2,5,7,9,10,11,16,17,19]:
    Newdf.append(data[i]['收盘价'][0:100])
Newdf=pd.DataFrame(np.array(Newdf).T)
Newdf.columns=['迈瑞医疗','爱尔眼科','海天味业','美的集团','恒顺醋业','绝味食品','白云山','海康威视'

Newdf
```

Out[40]:

	迈瑞医疗	爱尔眼科	海天味业	美的集团	恒顺醋业	绝味食品	白云山	海康威视	英科医疗	蓝思科技
0	386.01	62.05	150.45	82.01	19.08	78.00	28.00	53.40	156.00	25.85
1	382.37	60.39	147.91	80.68	18.94	73.36	27.84	50.39	150.40	24.98
2	376.00	59.72	146.51	81.68	18.80	73.70	27.88	51.23	158.20	24.30
3	366.66	58.11	148.50	81.47	18.96	74.00	27.83	52.30	165.90	25.04
4	361.20	57.16	151.66	83.38	19.04	74.05	28.22	53.02	179.09	25.80
...	...	...	...	...	...	...	...	...	...	...
95	390.82	67.01	168.65	86.11	20.60	80.50	30.93	48.12	113.05	33.64
96	406.01	67.50	162.16	85.66	20.17	80.56	30.46	46.58	111.72	33.25
97	405.31	64.70	160.70	82.55	20.10	80.99	30.59	45.62	114.96	33.25
98	393.21	64.19	161.54	83.40	19.34	80.28	30.28	45.62	117.27	33.32
99	386.80	62.26	160.20	77.87	20.04	81.10	30.52	44.90	116.21	33.23

100 rows × 10 columns

In [81]:

```
Newdf['迈瑞医疗'].mean()/Newdf['迈瑞医疗'].std()
```

Out[81]:

8.672637419986167

In [78]:

```
newdf=Newdf
```

In [79]:

```
Sigma=np.cov(newdf.T)
mu=np.mean(newdf)
```

In [73]:

```
def J(w):
    fenmu=sum(w*mu)
    fenzi=0
    for i in range(10):
        for j in range(10):
            fenzi+=w[i]*w[j]*Sigma[i,j]
    return fenzi/fenmu
```

In [74]:

```
cons=[lambda w: sum(w)-1]
```

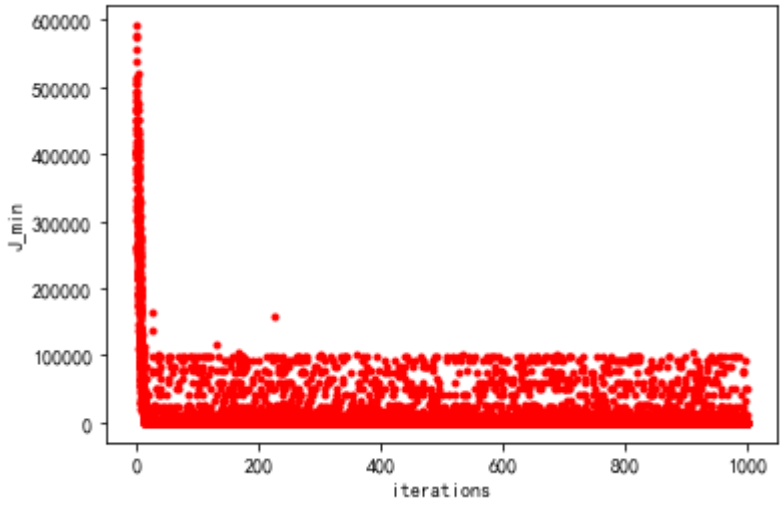
In [80]:

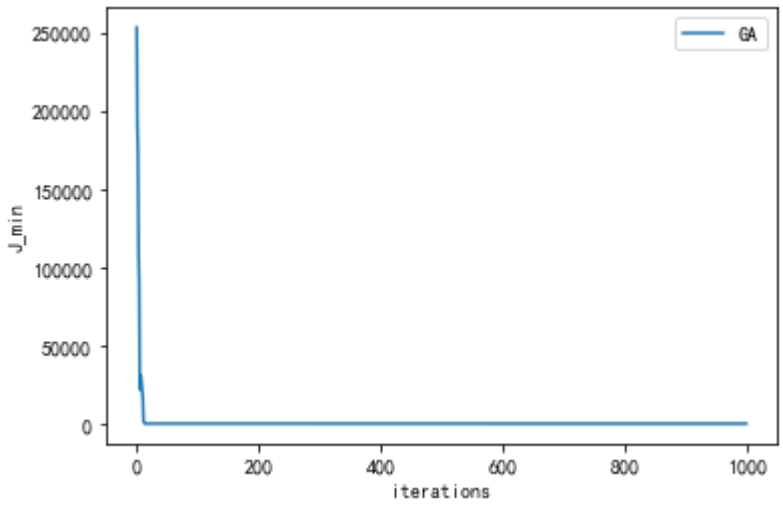
```
from sko import GA
import time
start_time=time.time()
ga=GA.GA(n_dim=10,func=J,constraint_eq=cons ,lb=[0]*10, ub=[1]*10,max_iter=1000)

general_best,func_general_best=ga.fit()
end_time=time.time()
print('best_x:',general_best)
print('best_y:',func_general_best)
print('time:',end_time-start_time)
```

```
best_x: [8.72027330e-02 4.32651665e-02 1.97157037e-01 1.14820487e-01
 1.61421309e-03 2.08212984e-01 1.73451792e-01 1.31983705e-01
 1.77919875e-04 4.21139623e-02]
best_y: [0.93011476]
time: 9.092033863067627
```

In [82]:

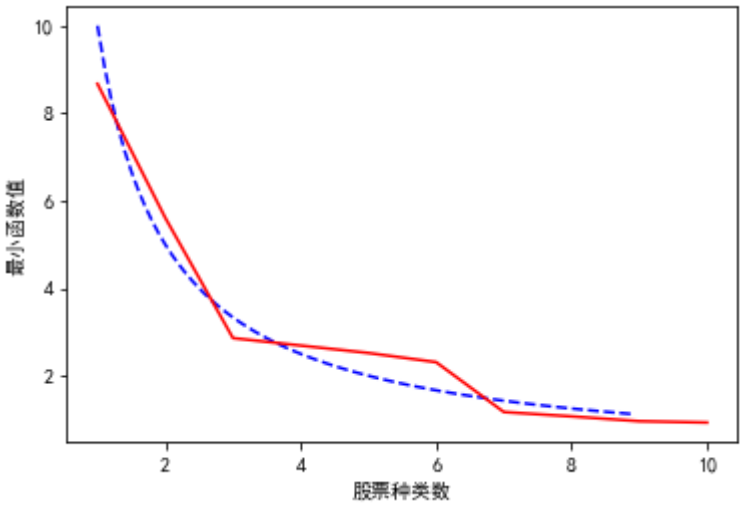
```
df_GA=pd.DataFrame(general_best,columns=None)
FitV_history_GA32 = pd.DataFrame(ga.all_history_Y)
plt.plot(FitV_history_GA32.index, FitV_history_GA32.values, '.', color='red')
plt.xlabel("iterations")
plt.ylabel("J_min")
plt.show()

plt_min1 = FitV_history_GA32.min(axis=1)
plt.plot(plt_min1.index, plt_min1, label='max')
plt.xlabel("iterations")
plt.ylabel("J_min")
plt.legend(['GA'])
plt.show()
```





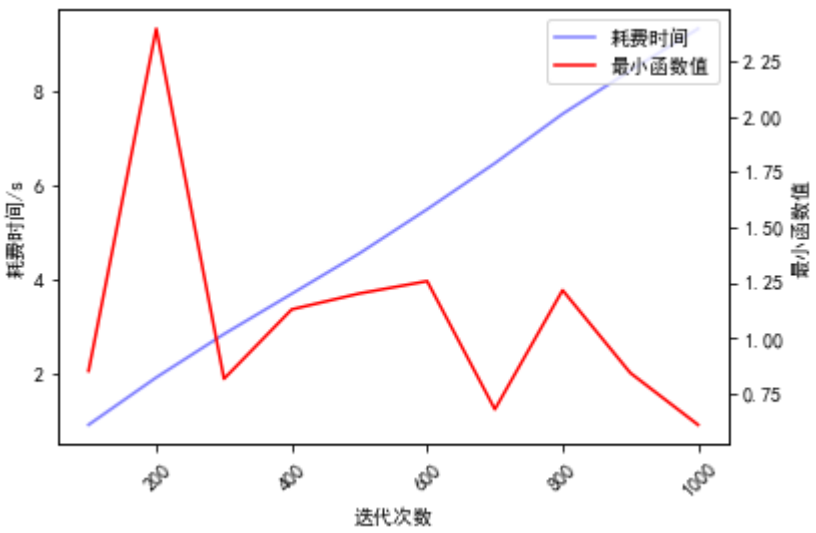
In [88]:

```
x1=np.arange(1,9,0.1)
x2=np.array([1,2,3,4,5,6,7,8,9,10])
y1=10/x1
y2=[8.67,5.61,2.86,2.69,2.52,2.31,1.17,1.07,0.96,0.93]
plt.plot(x1,y1,'b--')
plt.plot(x2,y2,'r-')
plt.xlabel("股票种类数")
plt.ylabel("最小函数值")
plt.savefig("函数值-股票.png")
plt.show()
```



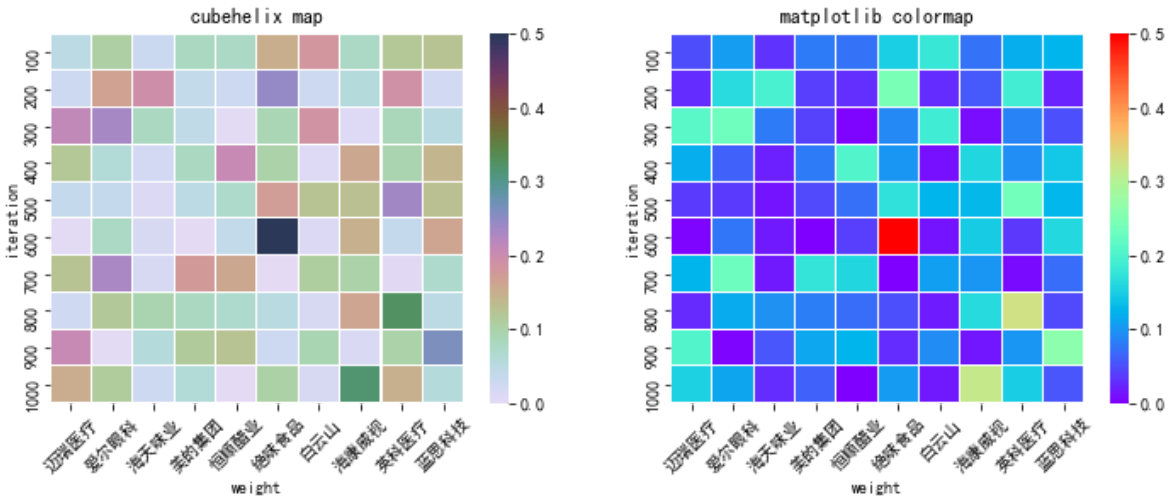
In [91]:

```
PreJ=[]
Time=[]
Weight=[]
for t in [100,200,300,400,500,600,700,800,900,1000]:
    start_time=time.time()
    ga=GA.GA(n_dim=10,func=J,constraint_eq=cons ,lb=[0]*10, ub=[1]*10,max_iter=t)

    general_best,func_general_best=ga.fit()
    end_time=time.time()
    Weight.append(general_best)
    PreJ.append(func_general_best)
    Time.append(end_time-start_time)

iteration=[100,200,300,400,500,600,700,800,900,1000]
fig, ax1 = plt.subplots()
plt.xticks(rotation=45)

ax1.plot(iteration, Time, color="blue", alpha=0.5, label="耗费时间")
ax1.set_xlabel("迭代次数")
ax1.set_ylabel("耗费时间/s")

ax2 = ax1.twinx()
ax2.plot(iteration, PreJ, color="red", label="最小函数值")
ax2.set_ylabel("最小函数值")

fig.legend(loc="upper right", bbox_to_anchor=(1, 1), bbox_transform=ax1.transAxes)
plt.savefig("迭代次数.png")
plt.show()
```



In [22]:

```
import seaborn as sns
f, (ax1,ax2) = plt.subplots(figsize = (12,4),ncols=2)

# cmap用cubehelix map颜色
cmap = sns.cubehelix_palette(start = 1.5, rot = 3, gamma=0.8, as_cmap = True)
pt =  pd.DataFrame(Weight)  # pt为数据框或者是协方差矩阵
sns.heatmap(pt, linewidths = 0.05, ax = ax1, vmax=0.5, vmin=0, cmap=cmap)
ax1.set_title('cubehelix map')
ax1.set_xlabel('weight')
ax1.set_xticklabels(['迈瑞医疗','爱尔眼科','海天味业','美的集团','恒顺醋业','绝味食品','白云山','海康
ax1.set_yticklabels(iteration)
ax1.set_ylabel('iteration')

# cmap用matplotlib colormap
sns.heatmap(pt, linewidths = 0.05, ax = ax2, vmax=0.5, vmin=0, cmap='rainbow')
# rainbow为 matplotlib 的colormap名称
ax2.set_title('matplotlib colormap')
ax2.set_xlabel('weight')
ax2.set_xticklabels(['迈瑞医疗','爱尔眼科','海天味业','美的集团','恒顺醋业','绝味食品','白云山','海康
ax2.set_yticklabels(iteration)
ax2.set_ylabel('iteration')

plt.savefig("热力图.png")
plt.show()
```



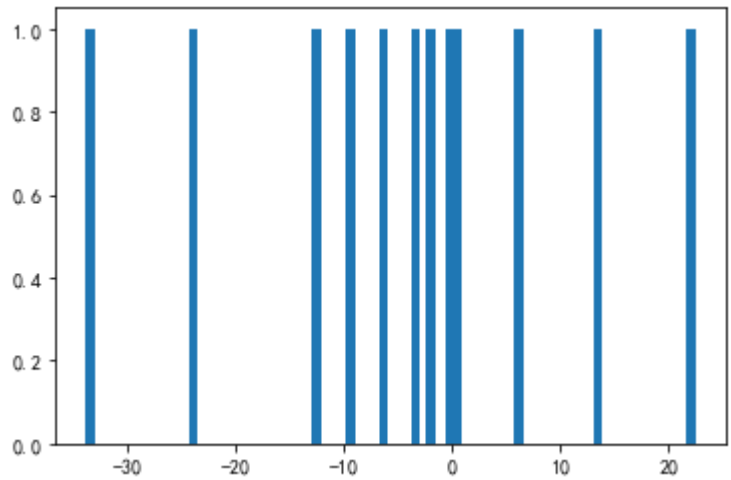
In [96]:

```
from scipy import stats
# 计算收益分布的偏度
skew=[]
for i in range(20):
    print(name[i]+"日收益率偏度: ",stats.skew(rishouyilv_lisan[i]))
    skew.append(stats.skew(rishouyilv_lisan[i]))
plt.bar(skew,1)
```

```
迈瑞医疗日收益率偏度:  0.4248895669744356
片仔癀日收益率偏度:  -6.431497628748267
爱尔眼科日收益率偏度:  0.15010210745660468
贵州茅台日收益率偏度:  -9.440057860923055
恒瑞医药日收益率偏度:  -3.370198900139399
海天味业日收益率偏度:  0.22612945677556964
格力电器日收益率偏度:  22.072549251208187
美的集团日收益率偏度:  0.43280707844224425
海螺水泥日收益率偏度:  -12.556049314204362
恒顺醋业日收益率偏度:  -0.17131601148600473
绝味食品日收益率偏度:  0.43831089466364664
白云山日收益率偏度:  0.4579656629407939
华兰生物日收益率偏度:  -23.910527839693927
深圳能源日收益率偏度:  -2.0210244047821306
中信证券日收益率偏度:  -33.49329241302319
蓝光发展日收益率偏度:  6.100274576224784
海康威视日收益率偏度:  0.10683238495763804
英科医疗日收益率偏度:  0.35340070217112685
上汽集团日收益率偏度:  13.446747325133689
蓝思科技日收益率偏度:  0.252660197167851
```

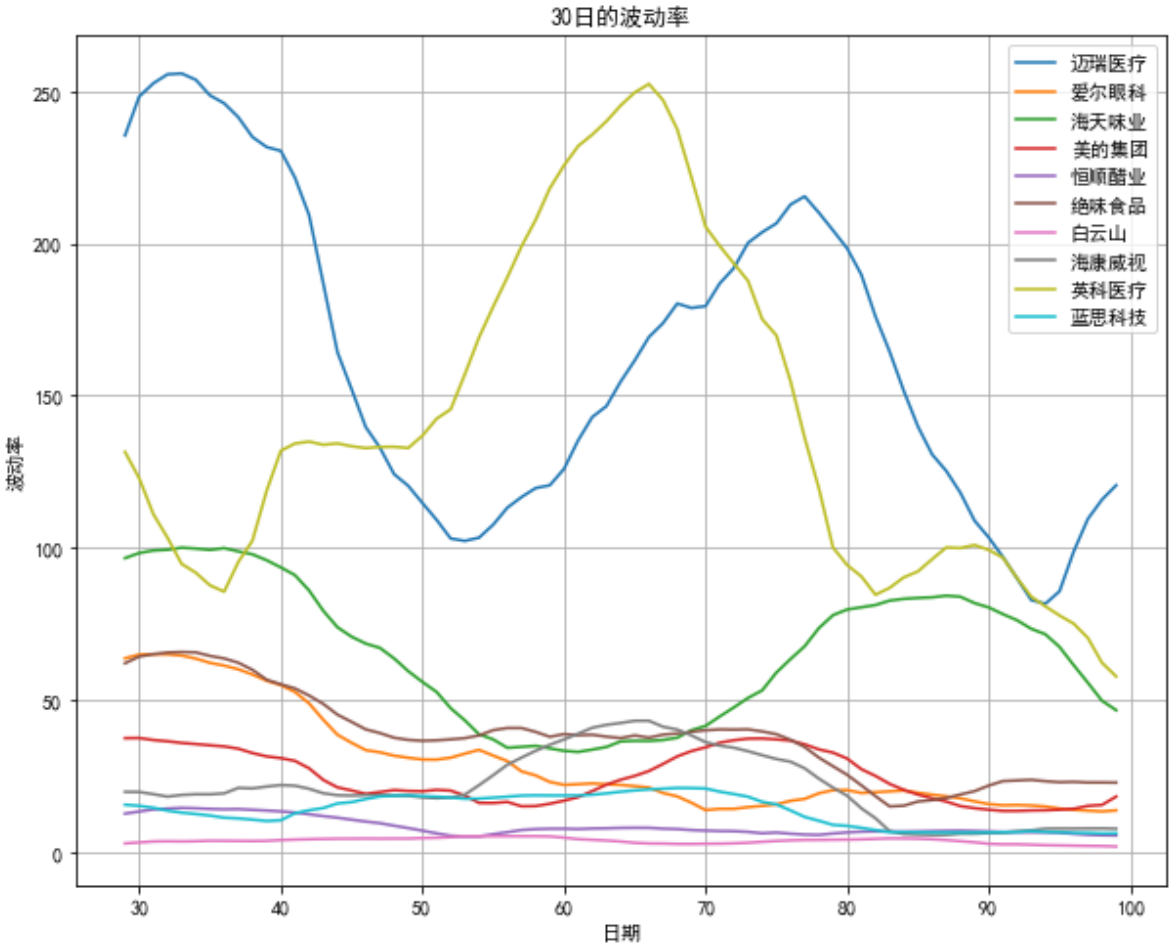
Out[96]:

```
<BarContainer object of 20 artists>
```



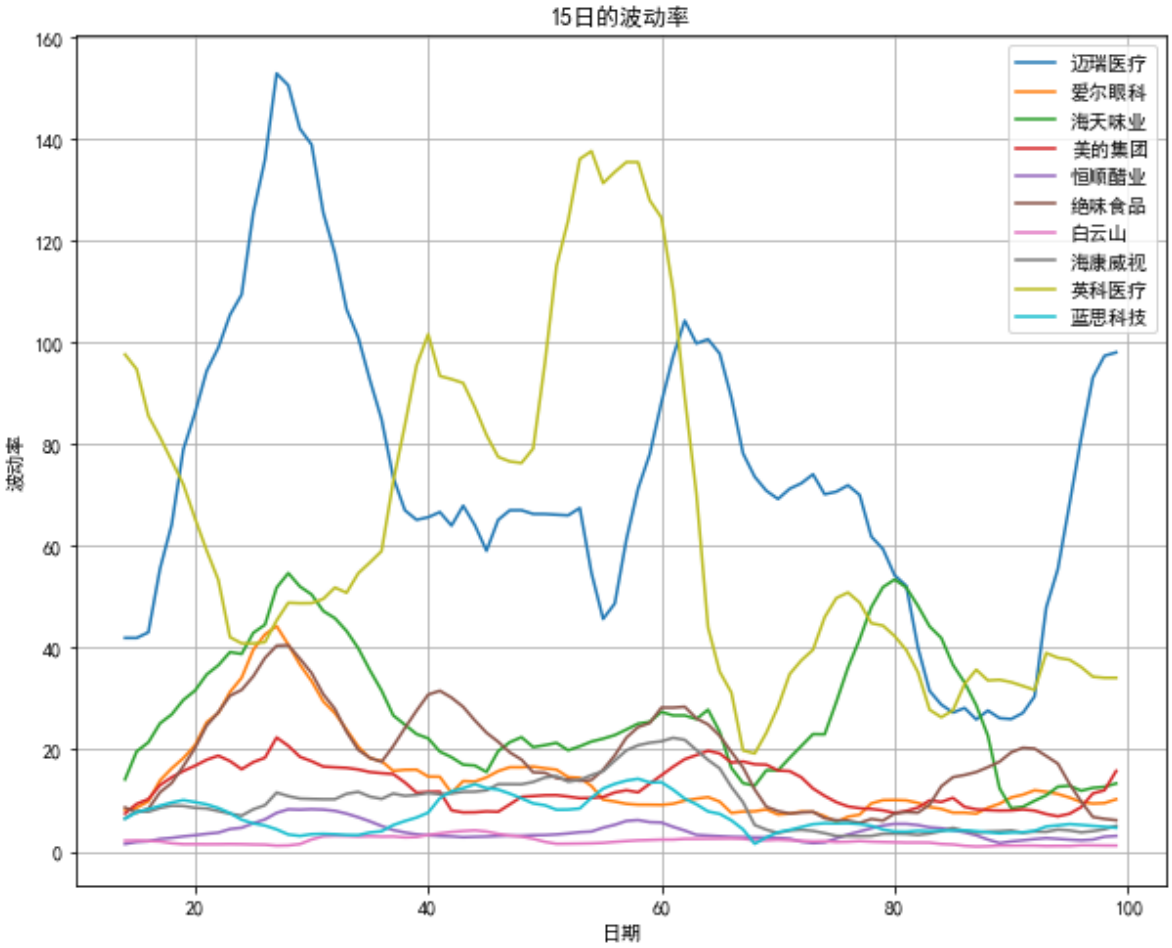
In [97]:

```
plt.figure(figsize=(10,8))
for i in ['迈瑞医疗','爱尔眼科','海天味业','美的集团','恒顺醋业','绝味食品','白云山','海康威视','英
    plt.plot(newdf[i].rolling(30).std()*np.sqrt(30))
plt.xlabel("日期")
plt.ylabel("波动率")
plt.title("30日的波动率")
plt.legend(['迈瑞医疗','爱尔眼科','海天味业','美的集团','恒顺醋业','绝味食品','白云山','海康威视','英
plt.grid()
plt.savefig("30日的波动率.png")
plt.show()
```



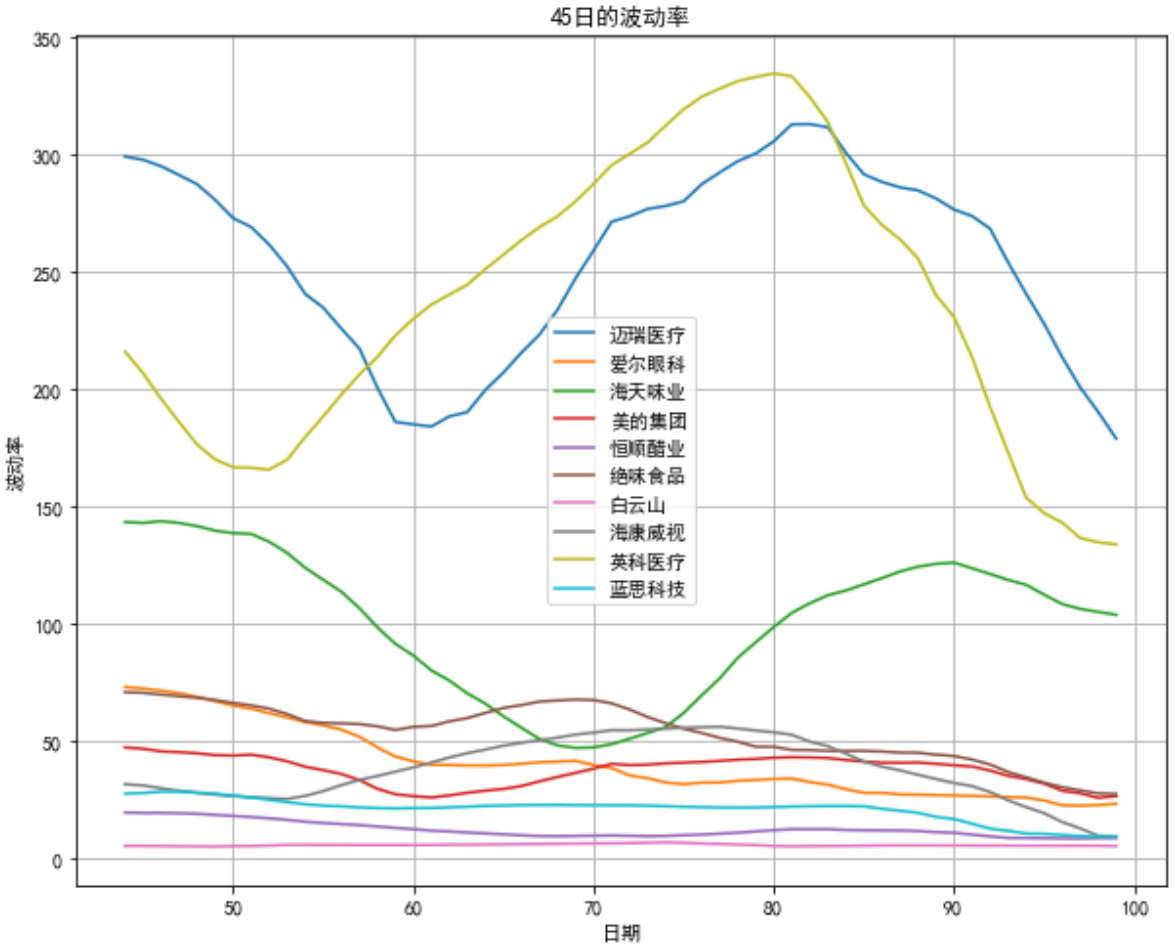
In [98]:

```
plt.figure(figsize=(10,8))
for i in ['迈瑞医疗','爱尔眼科','海天味业','美的集团','恒顺醋业','绝味食品','白云山','海康威视','英
    plt.plot(newdf[i].rolling(15).std()*np.sqrt(15))
plt.xlabel("日期")
plt.ylabel("波动率")
plt.title("15日的波动率")
plt.legend(['迈瑞医疗','爱尔眼科','海天味业','美的集团','恒顺醋业','绝味食品','白云山','海康威视','英
plt.grid()
plt.savefig("15日的波动率.png")
plt.show()
```



In [99]:

```
plt.figure(figsize=(10,8))
for i in ['迈瑞医疗','爱尔眼科','海天味业','美的集团','恒顺醋业','绝味食品','白云山','海康威视','英
    plt.plot(newdf[i].rolling(45).std()*np.sqrt(45))
plt.xlabel("日期")
plt.ylabel("波动率")
plt.title("45日的波动率")
plt.legend(['迈瑞医疗','爱尔眼科','海天味业','美的集团','恒顺醋业','绝味食品','白云山','海康威视','英
plt.grid()
plt.savefig("45日的波动率.png")
plt.show()
```



In [123]:

```
B=pd.DataFrame(m)
#B=(B-B.min())/(B.max()-B.min())
```

In [124]:

```
B=B[[1,2]]
```

In [125]:

```
def entropyWeight(data):
    data = np.array(data)
    # 归一化
    P = data / data.sum(axis=0)

    # 计算熵值
    E = np.nansum(-P * np.log(P) / np.log(len(data)), axis=0)

    # 计算权系数
    return (1 - E) / (1 - E).sum()

entropyWeight(B)
```

```
<ipython-input-125-2b26bee8194a>:7: RuntimeWarning: divide by zero encountered in lo
g
  E = np.nansum(-P * np.log(P) / np.log(len(data)), axis=0)
<ipython-input-125-2b26bee8194a>:7: RuntimeWarning: invalid value encountered in mul
tiply
  E = np.nansum(-P * np.log(P) / np.log(len(data)), axis=0)
```

Out[125]:

```
array([0.97941463, 0.02058537])
```

In [127]:

```
def topsis(data, weight=None):
    # 归一化
    data = data / np.sqrt((data ** 2).sum())

    # 最优最劣方案
    Z = pd.DataFrame([data.min(), data.max()], index=['负理想解', '正理想解'])

    # 距离
    weight = entropyWeight(data) if weight is None else np.array(weight)
    Result = data.copy()
    Result['正理想解'] = np.sqrt(((data - Z.loc['正理想解']) ** 2 * weight).sum(axis=1))
    Result['负理想解'] = np.sqrt(((data - Z.loc['负理想解']) ** 2 * weight).sum(axis=1))

    # 综合得分指数
    Result['综合得分指数'] = Result['负理想解'] / (Result['负理想解'] + Result['正理想解'])
    Result['排序'] = Result.rank(ascending=False)['综合得分指数']

    return Result, Z, weight

out=topsis(B)
pd.DataFrame(out[0]).to_csv("topsis.csv")
```

```
<ipython-input-125-2b26bee8194a>:7: RuntimeWarning: divide by zero encountered in lo
g
  E = np.nansum(-P * np.log(P) / np.log(len(data)), axis=0)
<ipython-input-125-2b26bee8194a>:7: RuntimeWarning: invalid value encountered in mul
tiply
  E = np.nansum(-P * np.log(P) / np.log(len(data)), axis=0)
```

In [128]:

```
out[1]
```

Out[128]:

	1	2
负理想解	0.000000	0.058954
正理想解	0.642717	0.294554

In [ ]:

```

```