1问题分析和数据处理

1.1xx概念（文科描述） 投资效用的概念 分析问题 最后要求得投资组合效用最大化

1.2数据处理

这里先给出对数收益率的概念和公式

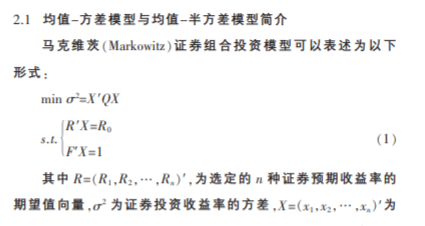
利用matlab对附件2中的数据进行处理，得到2019年对数收益率的数据。并且对每一种股票的对数收益率进行统计，将平均值作为期望对数收益率进行分析。

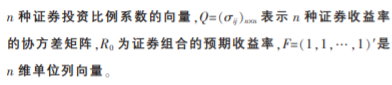
2模型建立

2.1均值-半方差模型

2.1.1 Markowitz的均值-方差模型

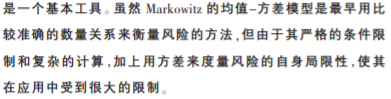
先是一段简单的文字介绍 然后把这个表述描述一下



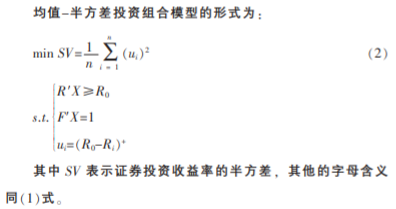


2.1.2 模型的改进——均值-半方差模型

把这段话圆滑地换个说法说一下（像这种类似的东西可以百度找一些相似的说法参照修改）



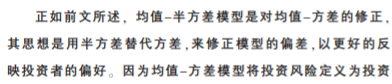
均值-半方差模型将位于投资期望线之上和之下的收益率分开计算，既规避了风险又确保投资者的最大收益避免了均值－方差模型造成的过滤掉可能的超额收益机会，所以更有利于投资者进行投资。

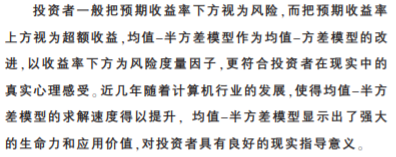


均值－方差模型将投资风险定义为投资收益的不确定性，而半方差模型则将投资风险定义为可能的损失。

2.1.3 模型评价

换个说法

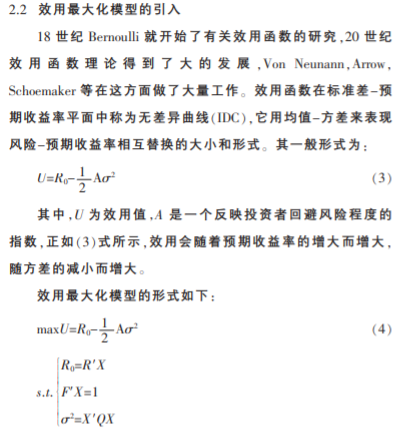


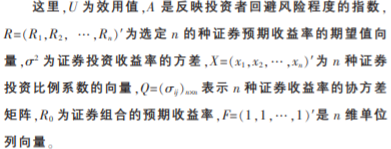


2.2 基于均值-半方差模型的效用最大化模型

2.2.1普遍的效用最大化模型

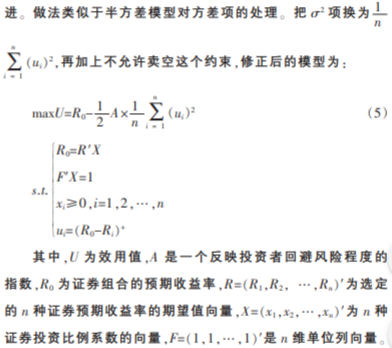
换个说法 这里参考百度也行 不要再用文章里的说法 用太多就过头了





2.2.2 基于均值-半方差模型的修正

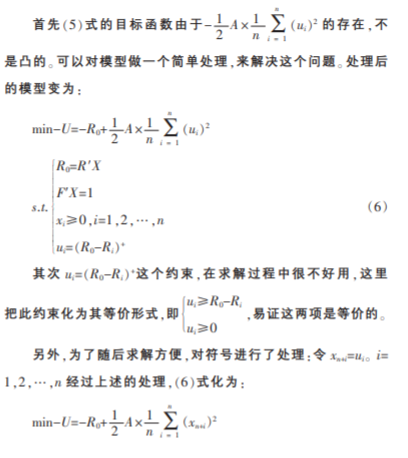
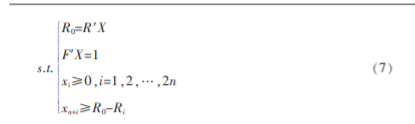
由于均值-半方差模型采用了半方差代替了方差，所以在形式上出现了偏差，并且在投资风险上定义不同，因此有必要进行修正。



3模型求解

3.1 转化为非线性二次规划模型

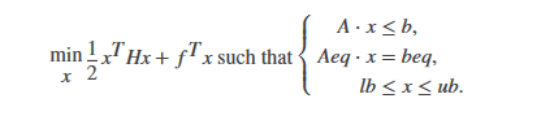
这里换个说法描述

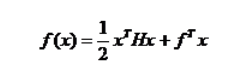
至此将改进后的效用最大化模型转化为线性约束的非线性二次规划模型。x1，x2，…,x57 组成的一维向量就是我们要求的投资方案。

3.2 quadprog() 函数求解二次规划模型

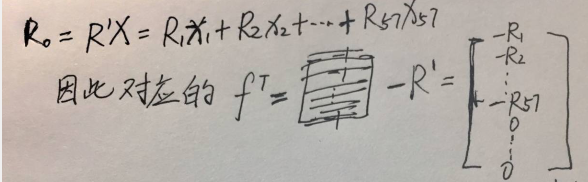
对于二次规划模型有很多的解法，比如…（这里你们百度一下）由于我们的求解目标是线性约束的非线性二次规划模型，所以选择使用matlab的quadprog() 函数来求解这个问题。

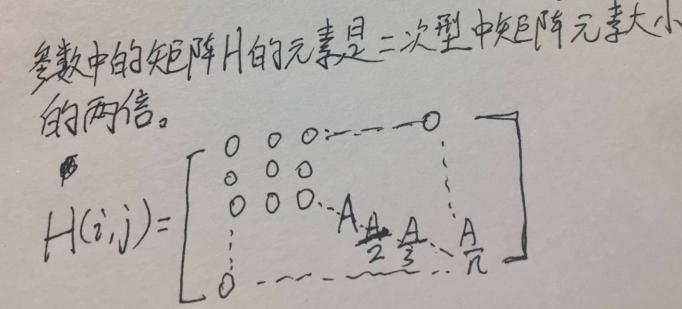


首先将目标函数的形式转化为

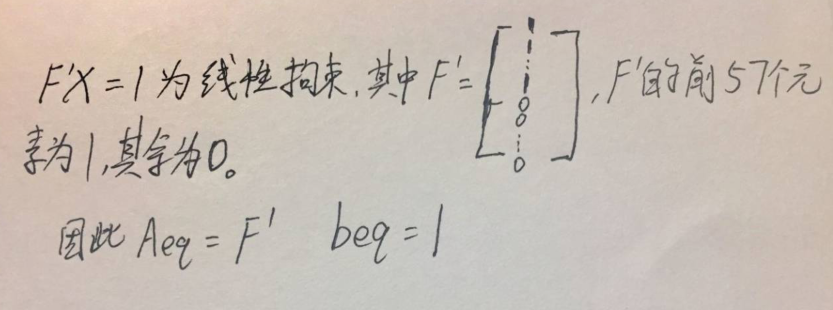


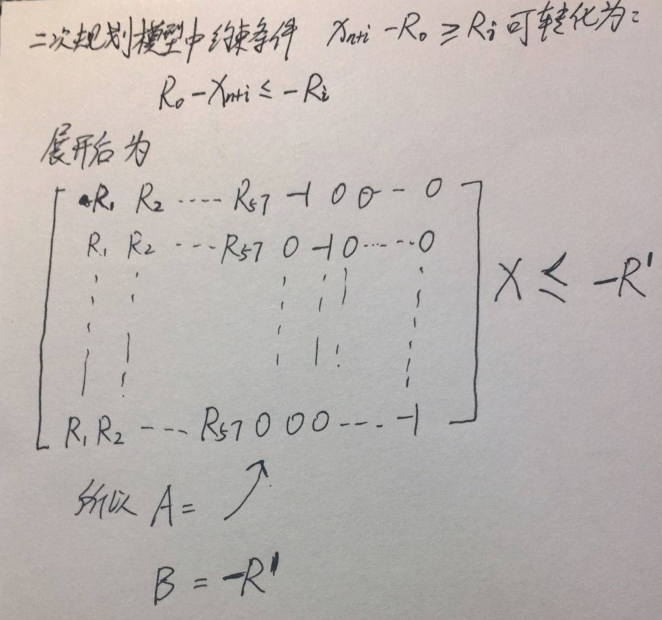
显然 R’向量也要拓展到114个元素，从58个元素开始用0填充。





约束条件要转化为的形式：





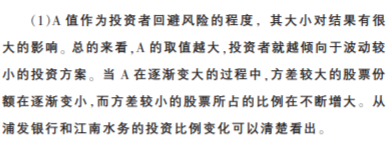
至此所有的参数已经转化完毕。

3.3 模型结果分析

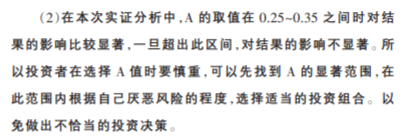
输入R’和A参数后带入quadprog()后得出如下表格

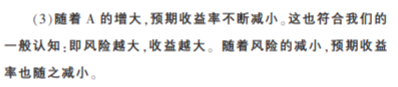
|  |  |  |
| --- | --- | --- |
| A值 | 最大效用 | 是否满足投资所有股份 |
| …… | …… | …… |
| 17.2 | 0.0026 | 否 |
| 17.3 | 0.0025 | 否 |
| 17.4 | 0.0025 | 是 |
| 17.5 | 0.0025 | 是 |
| …… | …… | …… |
| 22.3 | 0.0016 | 是 |
| 22.4 | 0.0016 | 否 |
| 22.5 | 0.0016 | 否 |
| …… |  |  |

这里描述一下A值的作用



A值选取17.3~22.3时对结果影响显著，超过此区间对结果影响不显著。（然后把下面这段话改写）





3.4题目最优解

可以看出A值选取17.3时配置方案满足所有的股份都有，并且效用最大

此时分配方案如下：

你们弄一个漂亮的表格 文字再加多点

|  |  |
| --- | --- |
| 1 | 0.0120% |
| 2 | 0.0102% |
| 3 | 0.0172% |
| 4 | 0.0183% |
| 5 | 0.0212% |
| 6 | 0.0173% |
| 7 | 0.0128% |
| 8 | 0.0113% |
| 9 | 0.0095% |
| 10 | 0.0152% |
| 11 | 0.0165% |
| 12 | 0.0300% |
| 13 | 0.0201% |
| 14 | 0.1074% |
| 15 | 0.0124% |
| 16 | 0.0085% |
| 17 | 0.0130% |
| 18 | 0.0239% |
| 19 | 0.0629% |
| 20 | 0.0093% |
| 21 | 0.0359% |
| 22 | 0.1555% |
| 23 | 0.0185% |
| 24 | 0.0306% |
| 25 | 0.0141% |
| 26 | 0.0319% |
| 27 | 0.0165% |
| 28 | 0.0167% |
| 29 | 0.0081% |
| 30 | 0.0058% |
| 31 | 0.0117% |
| 32 | 0.0136% |
| 33 | 0.0173% |
| 34 | 0.0123% |
| 35 | 0.0158% |
| 36 | 0.0131% |
| 37 | 0.0475% |
| 38 | 0.0076% |
| 39 | 0.1428% |
| 40 | 0.0373% |
| 41 | 98.6741% |
| 42 | 0.0051% |
| 43 | 0.0083% |
| 44 | 0.0126% |
| 45 | 0.0108% |
| 46 | 0.0241% |
| 47 | 0.0055% |
| 48 | 0.0319% |
| 49 | 0.0281% |
| 50 | 0.0154% |
| 51 | 0.0075% |
| 52 | 0.0133% |
| 53 | 0.0078% |
| 54 | 0.0165% |
| 55 | 0.0104% |
| 56 | 0.0128% |
| 57 | 0.0142% |