投资组合

1.投资组合收益率的计算

设投资组合中有 n 支股票，收益率分别为$r_1,r_2,\ldots,r_n$，相应的初始股票权重分别为$w_1,w_2,\ldots,w_n$，那么该投资组合的收益率为$r_p=\sum_{i=1}^{n}w_i r_i$。其中$\sum_{i=1}^{n}w_i=1$。

*证明：

设初始投入为C，在这 n 支股票上所投资的金额分别为$W_1,W_2,\ldots,W_n$，即满足$\sum_{i=1}^{n}W_i=C$。

当每支股票实现收益时，这 n 支股票对应的金额分别变化为$W_1(1+r_1),W_2(1+r_2),\ldots,W_n(1+r_n)$，那么现在投资组合的总价值为$C'=\sum_{i=1}^{n}W_i(1+r_i)$。

则$r_p=\frac{C'}{C}-1=\frac{\sum_{i=1}^{n}W_i(1+r_i)}{\sum_{i=1}^{n}W_i}-1=\sum_{i=1}^{n}w_i r_i$，其中$w_i=\frac{W_i}{\sum_{i=1}^{n}W_i}$，它是初始时刻在股票 i 上所投的金额占初始投资额的比例，而非持有的股票 i 的手数占组合总手数的比例。

注意点 1：

股票 i 占投资组合总金额的比例更新为：$w_i'=\frac{W_i(1+r_i)}{\sum_{i=1}^{n}W_i(1+r_i)}=\frac{W_i(1+r_i)}{C(1+r_p)}=w_i\frac{1+r_i}{1+r_p}$

因此，在计算换手率时，不应该用这里的w来计算。

注意点 2：

假设 n 支股票在 t 个交易日的日度收益率矩阵为$R_{nt}=(r_{ij})_{n\times t}$，$r_{ij}$为股票 i 在交易日 j 的收益率。

那么股票s_i在这 t 个交易日内整体的收益率$r_{s_i}=\prod_{j=1}^{t}(1+r_{ij})-1$，则投资组合（期间不调仓）在这 t 日内的收益率：$r_p=\sum_{i=1}^{n}w_i r_{s_i}=\sum_{i=1}^{n}w_i\left(\prod_{j=1}^{t}(1+r_{ij})-1\right)=\sum_{i=1}^{n}w_i\prod_{j=1}^{t}(1+r_{ij})-1$。

组合收益率的错误计算过程如下：

组合每日收益率$r_{d_j}=\sum_{i=1}^{n}w_i r_{ij}$，然后认为$r_p=\prod_{j=1}^{t}(1+r_{d_j})-1=\prod_{j=1}^{t}\left(1+\sum_{i=1}^{n}w_i r_{ij}\right)-1$。

错误原因在于，这种算法误以为组合中各股票的金额占比是恒定的，忽略了股价变动引起的每日股票金额占比的变动。

$\prod_{j=1}^{t}\left(1+\sum_{i=1}^{n}w_i r_{ij}\right)\geqslant\sum_{i=1}^{n}w_i\prod_{j=1}^{t}(1+r_{ij})$是否成立？其中$\sum_{i=1}^{n}w_i=1$，且$w_i$非负，$|r_{ij}|\leqslant 1$

2.换手率问题的解决思路

设 n 支股票在 t 个交易日的日度每股价格矩阵为$P=(P_{ij})_{n\times t}$，持仓股数（手数）矩阵为$Q=(Q_{ij})_{n\times t}$，资金分配额矩阵为$W=(W_{ij})_{n\times t}$，日度收益率矩阵为$R=(r_{ij})_{n\times t}$，日度资金分配权重矩阵为$w=(w_{ij})_{n\times t}$，则对任意股票s_i和交易日d_j均有$W_{ij}=P_{ij}\cdot Q_{ij}$；$w_{ij}=\frac{W_{ij}}{\sum_{i=1}^{n}W_{ij}}$。

以上各矩阵均记录的是发生最终变动后的状态（例如某天对股票的资金分配状态进行调整时，

先假定股票已经获得了当天的收益，然后再在这个资金状态上主动增减，W 记录的是主动增减后的资金状态）。

（1）不调仓（任意股票s_i每日的Q_{ij}均保持不变）

有$W_{i,j+\Delta t}=P_{i,j+\Delta t}\cdot Q_{ij}=\prod_{k=j}^{j+\Delta t-1}(1+r_{ik})W_{ij}$，收益$r_{s_i}=\frac{W_{i,j+\Delta t}}{W_{ij}}-1=\prod_{k=j}^{j+\Delta t-1}(1+r_{ik})-1$

组合收益$r_p=\frac{\sum_{i=1}^{n}W_{i,j+\Delta t}}{\sum_{i=1}^{n}W_{ij}}-1=\sum_{i=1}^{n}w_{ij}r_{s_i}$

（2）调仓但不考虑手续费

考察组合从d_j到d_{j+1}这相邻两天的变动：组合在d_j天的价值为$W_{pj}=\sum_{i=1}^{n}W_{ij}$，到$d_{j+1}$天时，价值变动为$W_{p,j+1}=\sum_{i=1}^{n}W_{i,j+1}=W_{pj}(1+r_{pj})$。注意这里的$r_{pj}$是根据$d_j$天的股票资金权重计算的，仍满足$r_{pj}=\sum_{i=1}^{n}w_{ij}r_{ij}$

假设在价值发生变动后开始调仓，且不额外引入新的资金投入，也不剩余任何闲置资金，而仅仅是利用现有资金$W_{p,j+1}$重新对各股票进行配置，则对股票s_i，其主动调仓的金额为：

$$\Delta W_{i,j+1}=W_{i,j+1}-P_{i,j+1}\cdot Q_{ij}=W_{i,j+1}-W_{ij}(1+r_{ij})$$

式中，第一项表示在d_{j+1}天股票s_i持有金额的目标值，第二项为在不调仓的情形下，d_{j+1}天股票s_i的自然持有金额。

调仓股数为$\Delta Q_{i,j+1}=\frac{\Delta W_{i,j+1}}{P_{i,j+1}}$；调仓完毕，有$W_{i,j+1}=P_{i,j+1}\cdot Q_{i,j+1}$，$Q_{i,j+1}=Q_{ij}+\Delta Q_{i,j+1}$

根据假设，有$\sum_{i=1}^{n}\Delta W_{i,j+1}=0$即$\sum_{i=1}^{n}[W_{i,j+1}-W_{ij}(1+r_{ij})]=0$，如此$W_{p,j+1}=W_{pj}(1+r_{pj})$

从资金权重的角度：

股票s_i在d_j天的资金权重为w_{ij}，到d_{j+1}天调仓前，资金权重变为$w_{ij}\frac{1+r_{ij}}{1+r_{pj}}$；调仓后，资金权重$w_{i,j+1}=\frac{W_{i,j+1}}{W_{p,j+1}}=\frac{\Delta W_{i,j+1}+W_{ij}(1+r_{ij})}{W_{pj}(1+r_{pj})}=\frac{\Delta W_{i,j+1}}{W_{pj}(1+r_{pj})}+w_{ij}\frac{1+r_{ij}}{1+r_{pj}}$。主动调仓导致的权重变化为：

$$\Delta w_{i,j+1}=w_{i,j+1}-w_{ij}\frac{1+r_{ij}}{1+r_{pj}}=\frac{\Delta W_{i,j+1}}{W_{pj}(1+r_{pj})}=\frac{P_{i,j+1}}{W_{pj}(1+r_{pj})}\Delta Q_{i,j+1}$$

在这种情形下，要计算组合在前后两天的收益率，$r_{pj}=\sum_{i=1}^{n}w_{ij}r_{ij}$仍成立，仅须保证$w_{ij}$已经是最终的目标资金分配权重。

如果要计算在 t 个交易日内组合的整体收益率：

$$r_p=\frac{W_{p,t+1}}{W_{p1}}-1=\prod_{j=1}^{t}(1+r_{pj})-1=\prod_{j=1}^{t}\left(1+\sum_{i=1}^{n}w_{ij}r_{ij}\right)-1$$

或者从单支股票出发仍有$r_p=\frac{W_{p,t+1}}{W_{p1}}-1=\frac{\sum_{i=1}^{n}W_{i,t+1}}{W_{p1}}-1=\sum_{i=1}^{n}w_{i1}r_{s_i}$，但这里：

$$r_{s_i}+1=\frac{W_{i,t+1}}{W_{i1}}=\frac{\Delta W_{i,t+1}+W_{it}(1+r_{it})}{W_{i1}}=\frac{\Delta W_{i,t+1}+\Delta W_{it}(1+r_{it})+W_{i,t-1}(1+r_{i,t-1})(1+r_{it})}{W_{i1}}$$

$$=\cdots=\frac{W_{i1}\prod_{j=1}^{t}(1+r_{ij})+\sum_{j=2}^{t+1}\Delta W_{ij}\prod_{k=j}^{t}(1+r_{ik})}{W_{i1}}=\prod_{j=1}^{t}(1+r_{ij})+\frac{\sum_{j=2}^{t+1}\Delta W_{ij}\prod_{k=j}^{t}(1+r_{ik})}{W_{i1}}$$

当对任意 j 有$\Delta W_{ij}=0$时，$r_{s_i}=\prod_{j=1}^{t}(1+r_{ij})-1$，此为不调仓情形下的公式。

（3）调仓且考虑手续费

设手续费率为α，即当主动换仓的金额为W时，组合总价值会损失掉αW；其余假设同上。

仍然考察组合从d_j到d_{j+1}这相邻两天的变动：

组合在d_j天的价值为W_{pj}，到d_{j+1}天时，组合的价值自然变动至$W_{pj}(1+r_{pj})$。假设d_{j+1}天的目标权重分配为$(w_{1,j+1},...,w_{n,j+1})$，为达到这个权重，在费用损失最小的情形下，投资组合中各股票的资金金额记为$(W_{1,j+1},...,W_{n,j+1})$，那么各股票主动调仓的资金变化情况如下：

$$\Delta W_{i,j+1}=W_{i,j+1}-W_{ij}(1+r_{ij})$$

在实际操作中，先对需要减仓的股票进行交易，而后用得到的资金购入需加仓的股票。因此，加仓股票的实际增加价值总和为减仓股票减少价值绝对值总和的$(1-\alpha)^2$：

$$-(1-\alpha)^2\sum_{\substack{i\\ \Delta W_{i,j+1}\leqslant 0}}\Delta W_{i,j+1}=\sum_{\substack{i\\ \Delta W_{i,j+1}>0}}\Delta W_{i,j+1}$$

买卖股票的手续费合计为：

$$fee=-\alpha\sum_{\substack{i\\ \Delta W_{i,j+1}\leqslant 0}}\Delta W_{i,j+1}-\alpha(1-\alpha)\sum_{\substack{i\\ \Delta W_{i,j+1}\leqslant 0}}\Delta W_{i,j+1}=\alpha(\alpha-2)\sum_{\substack{i\\ \Delta W_{i,j+1}\leqslant 0}}\Delta W_{i,j+1}$$

写为与$W_{i,j+1}$关联的式子：

$$fee=\alpha(\alpha-2)\sum_{\substack{i\\ \Delta W_{i,j+1}\leqslant 0}}[W_{i,j+1}-W_{ij}(1+r_{ij})]$$

其中，第一项为卖股票的手续费，第二项为买股票的手续费。交易费的存在将会导致组合在d_{j+1}天的实际价值削减为$W_{p,j+1}$。如下关系成立：

$$W_{p,j+1}=\sum_{i=1}^{n}W_{i,j+1}=\sum_{i=1}^{n}[W_{ij}(1+r_{ij})+\Delta W_{i,j+1}]=W_{pj}(1+r_{pj})-fee$$

根据权重的定义，对股票s_i有$w_{i,j+1}=\frac{W_{i,j+1}}{W_{p,j+1}}$。联立方程求解：

$$\begin{cases}W_{i,j+1}=w_{i,j+1}\cdot[W_{pj}(1+r_{pj})-fee],i=1,2,...,n\\ fee=\alpha(\alpha-2)\sum_{\substack{i\\ \Delta W_{i,j+1}\leqslant 0}}[W_{i,j+1}-W_{ij}(1+r_{ij})]\end{cases}$$

式中，$W_{1,j+1},...,W_{n,j+1}$和fee共 n+1 个未知数，以上有 n+1 个独立方程，故能求得唯一解。

Python 量化分析

1.股价影响因素

（1）公司层面；（2）行业层面；（3）市场层面；（4）心理；（5）宏观经济；（6）政治

2.金融分析

（1）基本面分析：a.宏观经济面：财政、货币政策；b.行业环境；c.公司财务信息、业绩报告

（2）技术面分析：技术指标（由市场交易信息得到的），eg：K 线，MA（均线：连续 xx 个交易日的收盘价均值，可加权），KDJ（随机指标），MACD（指数平滑移动平均线）

3.量化投资

（1）思想：使用机器学习等方法将投资理念投入实践中，机器比人工更能处理庞大的数据量，且能及时随市场变化而更新，模型的优化过程交给机器来实现。此外还能对选股策略进行**回测（基于历史）/模拟交易（基于当前的数据）**，验证策略是否合理。

（2）量化策略：输入（财务数据、行情数据、投资经验等等）➔策略（选什么股？什么时候买入卖出？资金如何在每支股票中分配？设置止盈止损的临界）➔输出（买入卖出信号，交易费用，实现的收益等等）

4.python 中的时间序列

设定 Series 的 index 为可识别的时间格式，欲对数据分周度、分月度进行计算，用 resample 函数

5.量化策略思路概览：

（1）止盈止损；

（2）双均线策略（解决择时买卖问题）

移动平均线常用的有 5 天、10 天的日均线（短线操作）；30 天、60 天的季均线（中期操作）；120 天和 240 天的年均线（长期操作）。

金叉：短期均线上穿长期均线，买入信号；死叉：短期均线下穿长期均线，卖出信号。

（3）因子选股策略（解决如何选股问题）

因子：选择股票的标准，如增长率、市值、市盈率、ROE

单因子选股策略：选取因子值最大或最小的前 N 支股票持仓，每隔一段时间调仓一次

多因子选股思想：①按某种顺序让不同的因子先后对股票进行筛选；②利用多个因子构造评分模型，选取评分最大的模型

（4）均值回归理论

认为股价波动以均线为中心，迟早会回归到均值水平。

偏离程度定义：(MA-P)/MA；可以通过数据训练得到用于择时买卖的偏离程度临界点，也可以将偏离程度视为一种因子进行选股（选偏离程度最高的 N 支股票进行调仓）。

（5）布林带策略（择时买卖）

布林带由三条线构成，上下两条线分别可视为价格的压力线和支撑线，中间线是价格平均线。

中间线：20 日均线

Up 线：20 日均线＋$k_1 \cdot SD$

Down 线：20 日均线－$k_2 \cdot SD$

其中 SD 为 20 日收益率的标准差；k_1、k_2 为比例系数，可以由训练得到

策略：股价突破压力线时清仓，跌破支撑线时全仓买入

（6）PEG 选股策略

彼得林奇：股票被合理定价时，市盈率 PE 与增长率 g 相等

PEG$\triangleq$PE/(100 $\cdot$ g)，值＜1，说明股价被低估；值＞1，股价被高估

PEG 估值法适用于成长型公司

策略：选 PEG 值最小的 N 支股票（但须剔除掉 PE 为负或 g 为负的股票）

（7）动量策略与反转策略

策略思想：选股票池中前一段时间内收益率最大 or 最小的 N 支股票

（8）羊驼交易法则

起始时随机买入 N 支股票，每天卖掉收益率最差的 M 支，再随机买入剩余股票池中的 M 支

XGBoost

1.原理：

（1）基学习器：决策树（以回归树为例；分类树的损失函数转为熵等指标）；

（2）加法模型：$\hat{y}_i^M=\sum_{j=1}^{M} f_j(x_i)=\hat{y}_i^{M-1}+f_M(x_i)$，$i$表示样本，$M$表示不同的树，$x_i$表示样本$i$的特征所构成的向量，$f_j(x_i)$表示第$j$棵树根据特征向量$x_i$所得到的预测值，$\hat{y}_i^M$表示$M$棵树对样本$i$的整体估计值；

（3）训练方式：**前向分步算法**（贪心策略，逐棵树优化）

在优化第m棵树时：$\hat{y}_i^m=\hat{y}_i^{m-1}+f_m(x_i)=\hat{y}_i^{m-1}+W_{m,q(x_i)}$，$\hat{y}_i^{m-1}$视为已知值，仅需增量优化$f_m(x_i)=W_{m,q(x_i)}$的值，$W_{m,q(x_i)}$表示回归树算法根据样本$i$的特征所得到的终节点估计值

目标函数：$Obj=\sum_{i=1}^{N} L\left(y_i,\hat{y}_i^m\right)+\sum_{j=1}^{m}\Omega(f_j)$，其中$L\left(y_i,\hat{y}_i^m\right)$表示损失函数（常用 MSE），$\Omega(f_j)$为正则项，代表子树的复杂度

优化目标：决策变量为当前这棵树终节点估计值向量$W_m=\left(W_{m,1},W_{m,2},...,W_{m,l_m}\right)$，最终求得$W_m^*=arg\min\sum_{i=1}^{N} L\left(y_i,\hat{y}_i^m\right)+\sum_{j=1}^{m}\Omega(f_j)$

$\Omega(f_j)=\gamma l_j+\frac{1}{2}\lambda\sum_{k=1}^{l_j}W_{j,k}^2$，超参数（自定义参数用以控制惩罚力度）：$\gamma$、$\lambda$；$l_j$：终结点个数

具体优化时，$W_m^*=arg\min\sum_{i=1}^{N} L\left(y_i,\overline{\hat{y}_i^{m-1}}+W_{m,q(x_i)}\right)+\overline{\sum_{j=1}^{m-1}\Omega(f_j)}+\gamma l_m+\frac{1}{2}\lambda\sum_{k=1}^{l_m}W_{m,k}^2$

$=arg\min\sum_{k=1}^{l_m}\left[\frac{1}{2}\lambda W_{m,k}^2+\sum_{i\in I_{m,k}} L\left(y_i,\overline{\hat{y}_i^{m-1}}+W_{m,k}\right)\right]+\gamma l_m$，其中$I_{m,k}$为树$m$落在终结点$k$的样本集合

经泰勒展开：$L\left(y_i,\overline{\hat{y}_i^{m-1}}+W_{m,k}\right)\approx L\left(y_i,\overline{\hat{y}_i^{m-1}}\right)+L'\left(y_i,\overline{\hat{y}_i^{m-1}}\right)W_{m,k}+\frac{1}{2}L''\left(y_i,\overline{\hat{y}_i^{m-1}}\right)W_{m,k}^2$，目的是把决策变量分离出来，简化运算，得到等价优化式：

$W_m^*\approx arg\min\sum_{k=1}^{l_m}\left[W_{m,k}\sum_{i\in I_{m,k}} L_i'+\frac{1}{2}W_{m,k}^2\left(\lambda+\sum_{i\in I_{m,k}} L_i''\right)\right]+\gamma l_m$。其中，$\sum_{i\in I_{m,k}} L_i'$表示属于终结点$k$的所有样本一阶梯度的加总；$\sum_{i\in I_{m,k}} L_i''$同理。

进一步，给定l_m时$W_m^*\approx arg\min\sum_{k=1}^{l_m}\left[W_{m,k}\sum_{i\in I_{m,k}} L_i'+\frac{1}{2}W_{m,k}^2\left(\lambda+\sum_{i\in I_{m,k}} L_i''\right)\right]+\gamma l_m$等价于：

$W_{m,k}^*\approx arg\min\left[W_{m,k}\sum_{i\in I_{m,k}} L_i'+\frac{1}{2}W_{m,k}^2\left(\lambda+\sum_{i\in I_{m,k}} L_i''\right)\right]$

判断$\lambda+\sum_{i\in I_{m,k}} L_i''$的符号：损失函数按含义而言应该是凸函数，二阶导为＋（才可能有最小值），而λ也为＋，故$\lambda+\sum_{i\in I_{m,k}} L_i''$为＋，$W_{m,k}^*\approx-\frac{\sum_{i\in I_{m,k}} L_i'}{\lambda+\sum_{i\in I_{m,k}} L_i''}$

（4）树的结构（贪心算法）：

每次节点分裂时，考察分裂前后目标函数最优值的信息增益（分类树中是基尼系数）$Gain = Obj^*_{前} - Obj^*_{后}$，使得$Gain$最大的划分方案即作为本次划分的最终依据。

$$Obj^* \approx \sum_{k=1}^{l_m}\left[W^*_{m,k}\sum_{i\in I_{m,k}} L'_i + \frac{1}{2}{W^*_{m,k}}^2\left(\lambda + \sum_{i\in I_{m,k}} L''_i\right)\right] + \gamma l_m \approx \gamma l_m - \frac{1}{2}\sum_{k=1}^{l_m}\frac{\left(\sum_{i\in I_{m,k}} L'_i\right)^2}{\lambda + \sum_{i\in I_{m,k}} L''_i}$$

树停止分裂的临界条件：①当次分裂可供选择的$Gain$的最大值$\leqslant 10^{-5}$；②终结点所含样本个数$\leqslant 1$；③限制终节点的个数等等

2.代码：

（1）精确贪心算法（得到的结果会比较精确，但排序尤其耗费时间）

Algorithm 1: Exact Greedy Algorithm for Split Finding

Input: I, instance set of current node

Input: d, feature dimension

$gain \leftarrow 0$

$G \leftarrow \sum_{i\in I} g_i,\ H \leftarrow \sum_{i\in I} h_i$

for $k = 1$ *to* m **do**

  $G_L \leftarrow 0,\ H_L \leftarrow 0$

  for j *in sorted*$(I,$ *by* $\mathbf{x}_{jk})$ **do**

    $G_L \leftarrow G_L + g_j,\ H_L \leftarrow H_L + h_j$

    $G_R \leftarrow G - G_L,\ H_R \leftarrow H - H_L$

    $score \leftarrow \max(score, \frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{G^2}{H+\lambda})$

  end

end

Output: Split with max score

时间复杂度：与两层循环与排序有关

（2）优化：近似算法（类似于随机森林思想）

①压缩特征数——列采样

a. 按树随机：每棵树随机选取一部分特征作为每次节点划分时的候选；b. 按层随机（更灵活）：树的每一层在划分时，随机选取

②压缩候选切分点数量（压缩每个特征的划分临界点/对每个特征的特征值进行分层抽样）：

a. 加权分位法（*）

优化式经泰勒展开后还等价于：$W^*_m = arg\min\sum_{i=1}^{N}\left(L'_i f_m(x_i) + \frac{1}{2}L''_i f_m^2(x_i)\right) + \Omega(f_m)$

利用配方法，等价于$arg\min\frac{1}{2}\sum_{i=1}^{N} L''_i\left(f_m(x_i) - \left(-\frac{L'_i}{L''_i}\right)\right)^2 + \Omega(f_m)$，即目标函数表示真实值为$-\frac{L'_i}{L''_i}$，权重为$L''_i$的平方损失。

在从原始特征值集合中抽选其中q个特征值来作为划分临界点时，这q个特征值的选取类似于分层抽样，规则如下（h是个体二阶导；使得划分后“h在不同划分区间是均匀的”）：

statistics of each training instances. We can define a rank functions $r_k : \mathbb{R} \rightarrow [0, +\infty)$ as

$$r_k(z) = \frac{1}{\sum_{(x,h)\in\mathcal{D}_k} h} \sum_{(x,h)\in\mathcal{D}_k, x<z} h, \tag{8}$$

which represents the proportion of instances whose feature value k is smaller than z. The goal is to find candidate split points $\{s_{k1}, s_{k2}, \cdots s_{kl}\}$, such that

$$|r_k(s_{k,j}) - r_k(s_{k,j+1})| < \epsilon, \quad s_{k1} = \min_i \mathbf{x}_{ik}, s_{kl} = \max_i \mathbf{x}_{ik}. \tag{9}$$

具体而言，加权分位法可以采取全局策略（一开始就定好每个特征的划分候选特征值，且在不同层级的划分中这些划分候选特征值不变）和局部策略（特征在每个层级的划分候选特征值会经过重新分层抽样）

3.细节

（1）缺失值的处理

缺失值不参与精确算法与优化算法中的排序、分位数计算过程，程序默认把特征存在缺失值的样本视为一个整体，比较该整体放在左侧叶节点还是右侧叶节点所得的$Gain$更大

（2）shrinkage

引入步长α：$\hat{y}_i^M = \hat{y}_i^{M-1} + \alpha \cdot f_M(x_i)$，如此防止过拟合。$\alpha$通常取 0.1。

（3）交叉验证

模型中的各处超参数可以通过最小化验证集上的误差来选取

信号与系统

1.信号

（1）含义：理解为自变量为时间 t 的函数，时间 t 可离散可连续

（2）自变量的变换

①时移（坐标轴上的左右平移）；②时间反转（变换前后关于 t=0 对称，如$x(\alpha t + \beta)$与$x(-\alpha t + \beta)$）；③尺度变换（将横轴压缩：$|\alpha| > 1$或拓展$|\alpha| < 1$）

（3）信号奇偶：与函数奇偶同理

（4）复指数：根据欧拉公式，$e^{i\theta} = \cos\theta + i\sin\theta$