量化投资 Using R

第十章综合案例研究 - - Fama1973

罗智超 (ROKIA.ORG)

Contents

背景

资料下载

- 数据下载地址：http://pan.baidu.com/s/1pKJXkSV
- 文献下载地址：Fama & MacBeth(1973)
- 代码下载地址：主程序 相关函数

选择理由

- 股票交易数据，具有普遍代表性，且数据获取容易
- 论文中模型逻辑复杂，绕来绕去，理不出关系
- 涉及多张表的数据动态生成与交叉关联应用，你中有我，我中有你
- 涉及移动回归，回归中使用的数据时间窗是滚动的
- 涉及分组回归且需提取、计算回归系数及残差的标准差
- 要将 n 个步骤的回归结果汇总到一个表，且需体现不同的回归 id
- 如何将上面问题用 100 行以内的 Base R 代码解决

问题描述

- 老师给了一个数据集和一篇论文（英文），说："同学：请用论文中的模型和原始数据集把论文中的计算结果跑出来，给你一周时间。"
- 于是，你用了 5 天的时间终于把论文里面的内容看明白了，假设其中要求你做以下工作。

问题描述 (Cont'd)

- 这碎碎的一抹青翠，好似乱坠了嫔妃的眼，平摊于日下，甚是沁人心脾。提神醒脑可是极好的！若忍心炙烤煎熬，蔫萎而焦灼，岂不是辜负了？

问题描述 (Cont'd)

- 组合收益率对组合前一期 beta 值和组合前一期回归残差的标准差进行回归，并进行 t 检验。

问题描述 (Cont'd)

- 根据提供的源数据完成以下工作

问题描述 (Cont'd)

- 第一步：对每个 period 的 Portfolio formation period(pplist) 的所有股票进行回归（RET ~ EWRTED)(reg.pp)，将回归系数从小到大排列，根据回归系数大小，将所有股票等分为 20 组。
- 第二步：对每个 period 的在第一步分组的股票进行跟踪。在 Testing period(**tplist**) 周期中**逐月**对每组股票进行回归分析（AVG_RET_bygroup ~ BETA+SD_resid) (reg.3)。
- 2.1 计算因变量：提取 pplist 每组股票的代码清单，然后根据代码清单在 **tplist** 中提取对应期间 (**逐月**) 的股票数据。分组计算 RET 的均值 AVG_RET_bygroup。

	Periods				
	1	2	3	4	5
Portfolio formation period ...	1926–29	1927–33	1931–37	1935–41	1939–45
Initial estimation period	1930–34	1934–38	1938–42	1942–46	1946–50
Testing period	1935–38	1939–42	1943–46	1947–50	1951–54
No. of securities available	710	779	804	908	1,011
No. of securities meeting data requirement	435	576	607	704	751

Figure 1: 分析对象

- 2.2 计算自变量：提取 pplist 每组股票的代码清单，然后根据代码清单在 **iplist** 中提取对应期间（**开始 5 年**）的股票数据。分股票进行回归（RET ~ EWRTED）(reg.2)，将回归结果的 beta 以及 sd_resi 分组计算均值。
- 第三步，将所有 period 的第一步、第二步的工作中计算的回归结果汇总到一个数据集。

五分钟思考

- 如何得到上面的计算过程（其实这个过程就是作者的研究思路）
- 如何将上面的计算过程转换成代码设计思路

问题实现

- 一团的乱、一堆的坑
- 框架设计最重要
- 步步为营、稳扎稳打

绕坑

- 设计思路决定后面所有的工作
- 切忌拿到问题马上下笔，先利用碎片时间思考下思路

绕坑 (Cont'd)

- 构建主框架

```
pplist<-list(c(1926:1929),c(1927:1933),c(1931:1937),
             c(1936:1941),c(1939:1945),c(1943:1949),
             c(1947:1953),c(1951:1957),c(1955:1961))
iplist<-list(c(1930:1934),c(1934:1938),c(1938:1942),
             c(1942:1946),c(1946:1950),c(1950:1954),
             c(1954:1958),c(1958:1962),c(1962:1966))
```

```
tplist<-list(c(1935:1938),c(1939:1942),c(1943:1946),
             c(1947:1950),c(1951:1954),c(1955:1958),
             c(1959:1962),c(1963:1966),c(1967:1968))
```

绕坑 (Cont'd)

- 主体框架

```
#Step one:creat all portfolio groups
for (i in 1:length(pplist))
  reg.pp(pplist[[i]])
#return df.step1
#Step two:regression
for (i in 1:length(tplist))
  ml.tp<-monthlist(tplist[[i]])
    for (j in 1:length(ml.tp))
    reg.3(i,j)
#return  df.final
```

绕坑 (Cont'd)

- 将 sas 数据集导入到 r

```
#stat transfer
#haven::read_sas
#foreign::read.sas
#sas7bdat::read.sas7dbat
```

- 参考《数据分析》课件第四章

绕坑 (Cont'd)

- 导入到 R 后发现，DATE： - 12417...

```
df$DATE<-as.Date(df$DATE, origin="1970-01-01")
# 自己挖了一个坑，跳进去却浑然不知道
```

绕坑 (Cont'd)

- pplist iplist tplist 同步问题

```
calcultY<-function(i,j){
  a<-pplist[[i]]
  p<-paste(a[1],a[length(a)],sep="~")
  ds.x<-subset(df2,YEAR==substr(ml.tp[j],1,4) &
               MONTH==as.numeric(substr(ml.tp[j],5,6)))
  ds.y<-subset(df.step1,pp==p,select=c(PERMNO,group))
  df.xy<-merge(ds.x,ds.y,by="PERMNO",all.x=TRUE)
#...
}
```

绕坑 (Cont'd)

- 分组回归
- 参考分组回归的几种方法

```
library(lme4)
lm.1<-lmList(RET ~ EWRETD | PERMNO , data=df.t)
beta<-coef(lm.1)
# 注意要判断每组 obs 的数量是否满足回归方程最少要求
```

绕坑 (Cont'd)

- 如何提取分组回归的 beta 系数和残差的标准差

```
beta<-coef(lm.1)
sd_resid<-lapply(1:length(lm.2),
                 function(x)
                   sd(lm.2[[x]]$residuals) )
# 如何将提出出来的结果合并到一个数据集
   sd_resid<-do.call(rbind,sd_resid)
   sd_resid<-data.frame(sd_resid,labels(lm.2))
```

绕坑 (Cont'd)

- 如何将回归系数排序后等分 20 组

```
 x<-seq(from=0,to=nrow(beta),
        by=ceiling(nrow(beta)/20))[2:20]
  x<-c(x,nrow(beta))
  beta$group<-cut(1:nrow(beta),
                  c(0,x),labels = paste0("g",1:20))
  beta$pp<-paste(timeRange[1],
              timeRange[length(timeRange)],sep="~")
```

绕坑 (Cont'd)

- 如何逐月滚动 (roll) 提取横截面数据进行回归

```
 ml.ip<-monthlist(iplist[[i]])
  st<-as.Date(paste0(ml.ip[j],"01"),"%Y%m%d")
  et<-st+months(60)
  ds.x<-subset(df2,DATE>=st & DATE<=et)
```

绕坑 (Cont'd)

- 如何从多张表中提取所需的回归数据

```
 a<-pplist[[i]]
  p<-paste(a[1],a[length(a)],sep="~")
  ds.x<-subset(df2,YEAR==substr(ml.tp[j],1,4) &
              MONTH==as.numeric(substr(ml.tp[j],5,6)))
```

```
  ds.y<-subset(df.step1,pp==p,select=c(PERMNO,group))
  df.xy<-merge(ds.x,ds.y,by="PERMNO",all.x=TRUE)
```

绕坑 (Cont'd)

- 如何标记存储回归结果 id

```
beta$pp<-paste(timeRange[1],
               timeRange[length(timeRange)],sep="~")
```

绕坑 (Cont'd)

- 如何将所有循环的结果保存在一个数据集中

```
    df.final[[tpid]]<<-res.lm3
```

绕坑 (Cont'd)

- 如何处理报错问题（必须考虑到主要可能的错误）

```
#First type check
clear.reg<-function(df,n){
  #make sure each regression has at least n obs
  d<-df %>% group_by(PERMNO)%>% summarise(nr=n())
  c<-subset(d,nr>=n,select=c(PERMNO,nr))
  df<-merge(df,c,by="PERMNO")
}
# Second type check
  if (nrow(y)==0 | nrow(x) == 0 )
    {return(print(paste0(ml.tp[j],
   "test without data!"))) }
```

总结

- 这是一个综合的练习
- 需要扎实的基本功训练
- 思路的上层建筑决定代码效率
- 独立协同完成
- 不要放过每一个细节
- 极致的工匠精神是我们都欠缺的
- 独立完成后，分析问题解决问题的水平会达到一个新的水平