

第五届中国R语言会议

R与金融投资分析的框架

数据、模型、策略及报告

邓一硕

Web:<http://yishuo.org>

Email:dengyishuo@163.com

前言

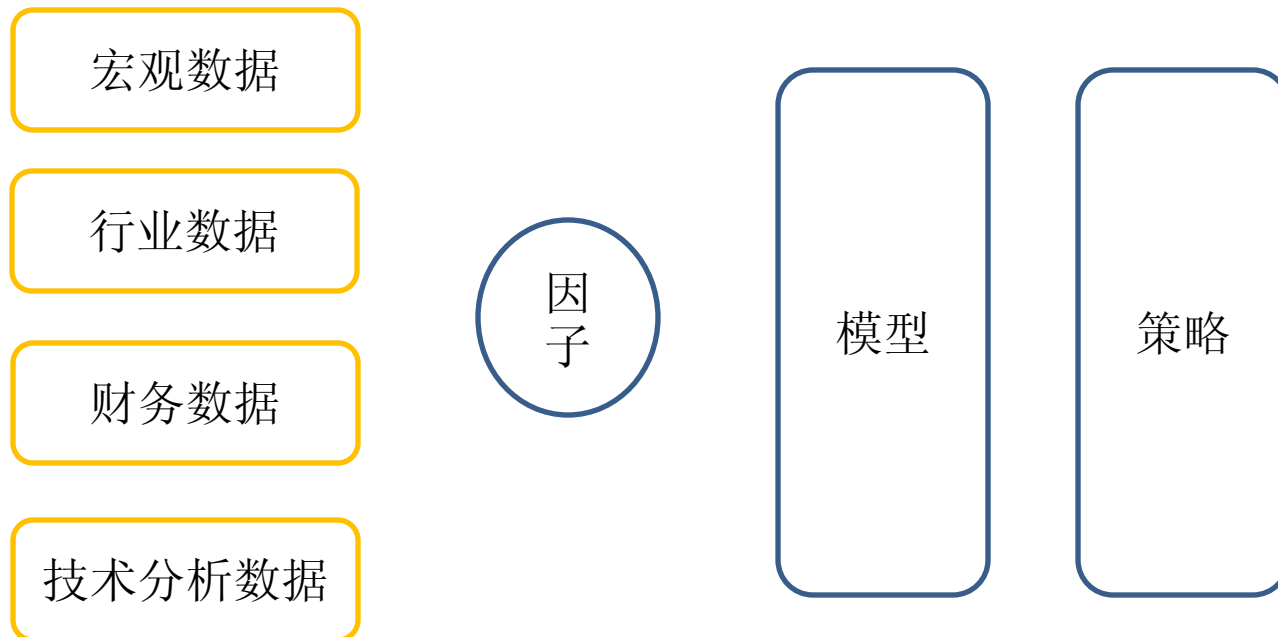
要学会安抚自己那颗略带恐惧的心！

——彼得·林奇

CAPM模型

- 理论基础： $R_i = \alpha + \beta R_m$
- Alpha来源：基本面→财务建模
→行业分析
- Beta来源：股票与指数相关性→回归分析
- Rm来源：指数波动→股指建模
→宏观建模

一个框架



数据篇

数据，数据，没有数据的推理是罪恶！

——福尔·摩斯

数据的重要性

- 数据和信息是金融分析的灵魂
- 模型与策略都是基于数据得出
- 模型好坏的判断标准便是未来的数据是否符合模型的预测结果
- 与策略相比，客户更关心的是数据

金融数据处理与R

- R是处理金融数据的利器
- 数据的导入与导出 ([R Data Import/Export](#))
 - 可以处理多种类型的数据
- 数据重整与清洗
- 数据的可视化 (graphics,lattice,ggplot2,ggobi)
- 针对数据进行建模
 - 提供了线性回归、机器学习等各种模型的函数

模型篇

All models are wrong but some are useful.
——George E. P. Box

模型的作用

- 模型是数据的统帅
- 好的模型往往可以将繁杂的数据提炼成一条简单的规律。比如一条回归线便可以刻画一群杂乱的点。
- 好的模型可以稳定地预测或者指导未来。

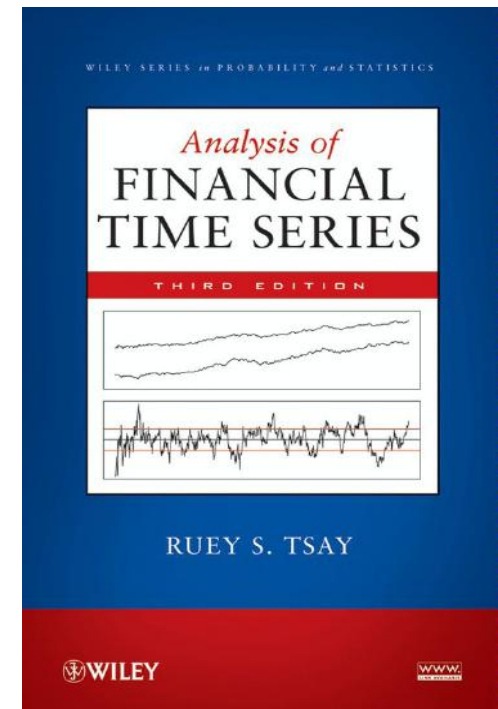
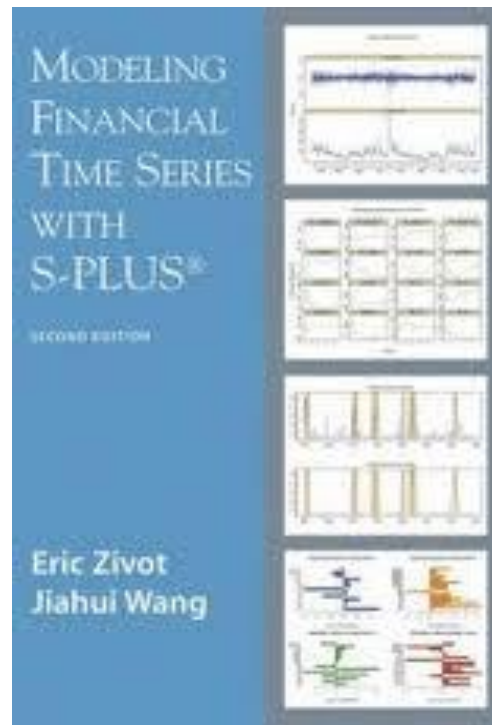
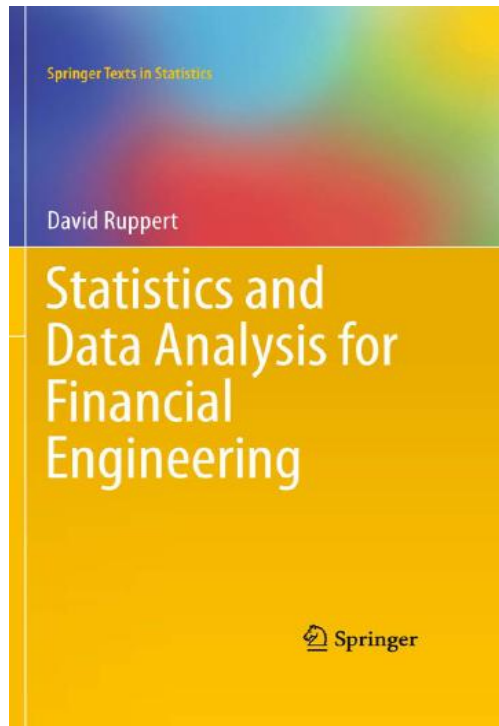
金融模型与R

- 几乎所有的统计模型在金融分析都有涉及
- 线性回归模型与CAPM模型
- 移动平均与MACD指标
- 协整与套利
- GARCH,SV模型与波动率
- 极值理论与涨跌预测
- 突发冲击与传递模型、时间序列聚类
-

相关的R包

- base/stat/MASS
- e1071/kernlab/klar/svmpath
- rugarch/fgarch/gogarch
- rpart/party
- quantmod/blotter/quantstrat/TTR

相关书籍



策略篇

知己知彼，百战不殆！

——《孙子·谋攻》

策略的意义

- 金融建模的目的是为了从数据中找出金融市场的波动规律，继而开始出可以**持续盈利**的策略。
- 好的策略应当可以在一定时期内战胜**市场先生 (Mr. Market)**。

策略的类型

- 价值投资型策略
 - 格雷厄姆、费雪、巴菲特、芒格
- 量化投资型策略
 - 索普、西蒙斯
- 技术分析型策略
 - 江恩



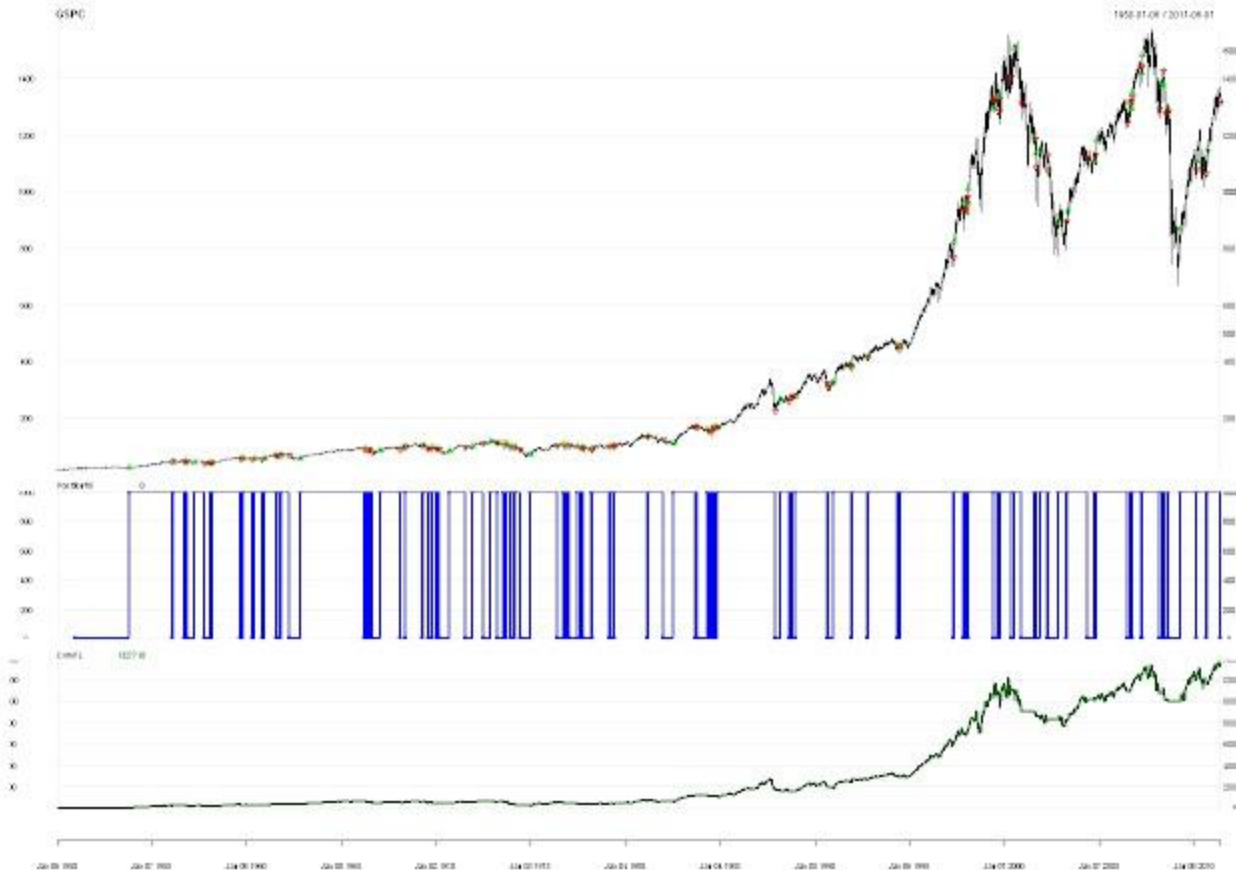
相关R包

- Quantstrat
- Blotter
- quantmod

quantstrat 示例

- ```
#thanks so much to the developers of quantstrat
#99% of this code comes from the demos in the quantstrat package#now let's define our silly countupdown function
CUD <- function(price,n) {
 #CUD takes the n-period sum of 1 (up days) and -1 (down days)
 temp<-runSum(ifelse(ROC(price,1,type="discrete") > 0,1,-1),n)
 colnames(temp) <- "CUD"
 temp
}
try(rm("order_book.CUD",pos=.strategy),silent=TRUE)
try(rm("account.CUD","portfolio.CUD",pos=.blotter),silent=TRUE)
try(rm("account.st","portfolio.st","stock.str","stratCUD","initDate","initEq","start_t","end_t"),silent=TRUE) # Initialize a strategy object
stratCUD <- strategy("CUD") # Add an indicator
stratCUD <- add.indicator(strategy = stratCUD, name = "CUD", arguments = list(price = quote(Cl(mktdata)),n=20), label="CUD") # enter when CUD > 0
stratCUD <- add.signal(strategy = stratCUD, name="sigThreshold",arguments = list(threshold=-0.5, column="CUD",relationship="gt", cross=TRUE),label="CUD.gteq.0")
exit when CUD < 0
stratCUD <- add.signal(strategy = stratCUD, name="sigThreshold",arguments = list(threshold=-0.5,
column="CUD",relationship="lt",cross=TRUE),label="CUD.lt.0") stratCUD <- add.rule(strategy = stratCUD, name='ruleSignal', arguments = list(sigcol="CUD.gteq.0",
sigval=TRUE, orderqty=1000, ordertype='market', orderside='long', pricemethod='market', replace=FALSE), type='enter', path.dep=TRUE)
stratCUD <- add.rule(strategy = stratCUD, name='ruleSignal', arguments = list(sigcol="CUD.lt.0", sigval=TRUE, orderqty='all', ordertype='market', orderside='long',
pricemethod='market', replace=FALSE), type='exit', path.dep=TRUE) currency("USD")
symbol = "GSPC"
stock(symbol, currency="USD",multiplier=1)
#use paste with ^ to get index data
getSymbols(paste("^",symbol,sep=""),adjust=T,from="1900-12-31")
#I use weekly but comment this out if you want to use daily
GSPC<-to.weekly(GSPC) initDate='1950-12-31'
initEq=100000
port.st<-'CUD' #use a string here for easier changing of parameters and re-trying initPortf(port.st, symbols=symbol, initDate=initDate)
initAcct(port.st, portfolios=port.st, initDate=initDate)
initOrders(portfolio=port.st, initDate=initDate) print("setup completed") # Process the indicators and generate trades
start_t<-Sys.time()
out<-try(applyStrategy(strategy=stratCUD , portfolios=port.st))
end_t<-Sys.time()
print("Strategy Loop:")
print(end_t-start_t) start_t<-Sys.time()
updatePortf(Portfolio=port.st,Dates=paste('::',as.Date(Sys.time()),sep=""))
end_t<-Sys.time()
print("trade blotter portfolio update:")
print(end_t-start_t) # hack for new quantmod graphics, remove later
themelist<-chart_theme()
themelistcolup.col<-'lightgreen'
themelistcoldn.col<-'pink' chart.Posn(Portfolio=port.st,Symbol=symbol,theme=themelist,log=TRUE)
```

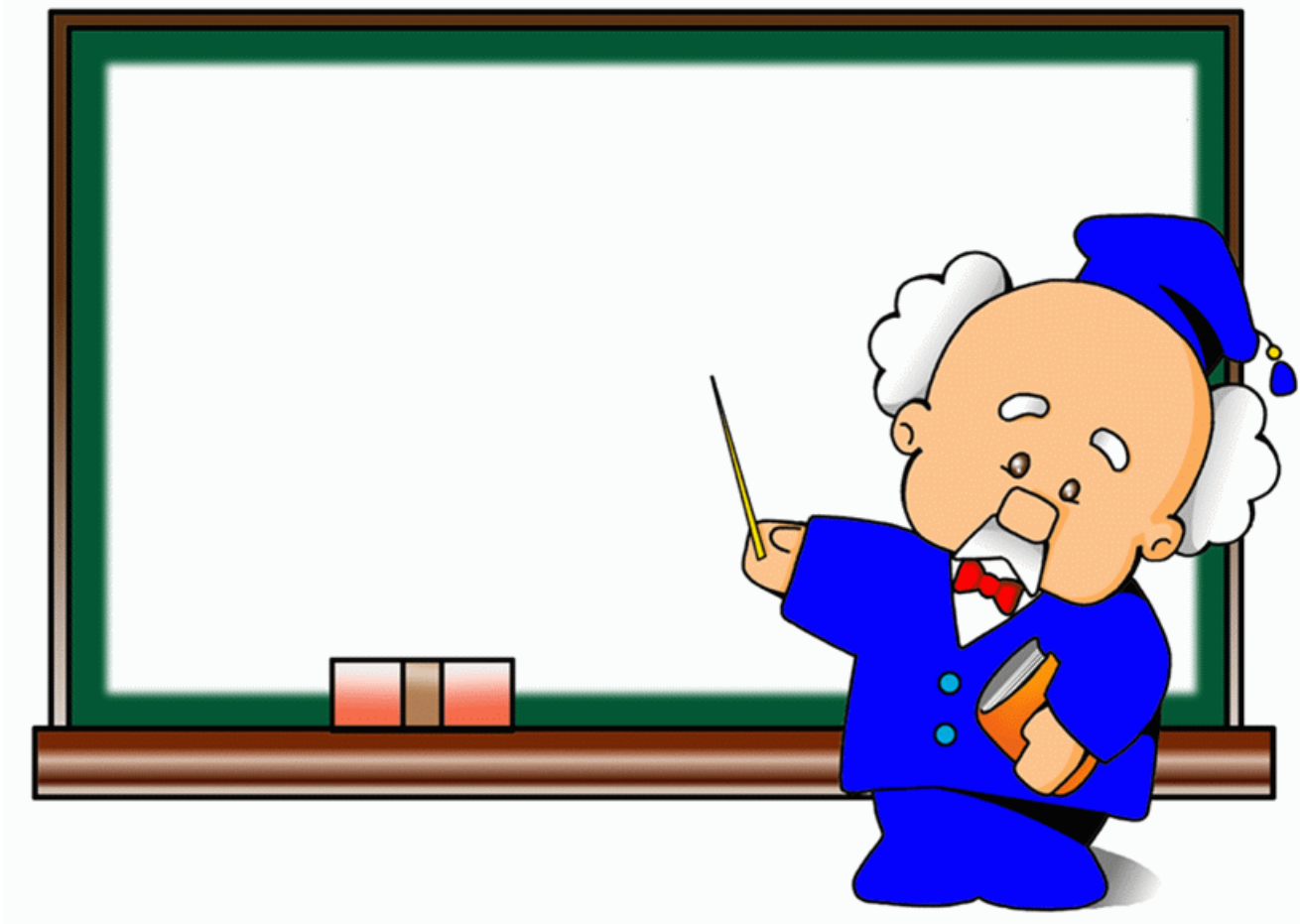
# quantstrat 示例



- 来自 <http://www.r-bloggers.com/a-quantstrat-to-build-on/>

# 报告篇

---



# 报告

---

- 可以简要地叙述模型结果、反映投资状况
- 结合Linux和knitr
- 自动运行模型并生成报告
- 甚至可以短信或者邮件通知你
- .....

**Thank you!**