Tidyquant

——金融量化分析新工具

小组成员：宋泠萱，张楠、吴国媛、刘宇涵

目录

CONTENT

概要介绍

PART.01

R语言中有大量的用于金融数量分析的工具。在传统上，这些工具大多是以zoo或xts这样的时间序列数据结构为基础的，比如xts包（时间序列数据处理）、quantmod包（金融数据获取和建模）、TTR包（技术分析）和PerformanceAnalystics包（业绩评价）这些在R社区中得到广泛使用的金融量化包。这些工具已经有了几年甚至十几年的历史，它们的功能已经久经考验。

但是这些工具也有一个沿袭自历史的“缺点”，就是zoo/xts这些包的函数不能很好的处理数据框，与目前R语言中更新的tidyverse框架也不兼容。tidyquant包就是出于整合xts与tidyverse的目的而开发的。

tidyquant使用了基于tidyverse的整洁数据架构，将xts、quantmod、TTR和performanceAnalytics工具整合在一起，实现了彼此之间的无缝交互。它拥有少量功能强大的核心函数，同时可以使用tidyverse系列R包中的工具对金融数据进行处理和建模，也可以使用ggplot2进行可视化。

核心函数

PART.02

tq_index( )

返回指数中每个股票的股票代号、公司名称、股指中的权重。来源是www.ssga.com.
（state street global advisor 美国道富金融集团）
tq_index_options() 返回可以从中选择的股票指数列表。

```
> #get the list of stock index options
> tq_index_options()
[1] "DOW"       "DOWGLOBAL"
[3] "SP400"     "SP500"
[5] "SP600"
```

```
> tq_index("SP500")
Getting holdings for SP500
# A tibble: 504 x 8
   symbol company    identifier
   <chr>  <chr>      <chr>
 1 AAPL   Apple Inc. 03783310
 2 MSFT   Microsoft~ 59491810
 3 AMZN   Amazon.co~ 02313510
 4 FB     Facebook ~ 30303M10
 5 GOOGL  Alphabet ~ 02079K30
 6 GOOG   Alphabet ~ 02079K10
 7 TSLA   Tesla Inc  88160R10
 8 BRK.B  Berkshire~ 08467070
 9 JPM    JPMorgan ~ 46625H10
10 JNJ    Johnson &~ 47816010
# ... with 494 more rows, and
```

tq_exchange()

tq_exchange()返回交易所内每只股票的股票代码、公司、收盘价、市值、部门和行业。
tq_exchange_options()返回您可以选择的证券交易所列表。选择是AMEX、NASDAQ和NYSE。
(AMEX美国证券交易所，NASDAQ纳斯达克、 NYSE纽约证券交易所)

```
> tq_exchange_options()
[1] "AMEX"   "NASDAQ" "NYSE"
```

```
> tq_exchange("NASDAQ")
Getting data...

# A tibble: 4,040 x 7
   symbol company
   <chr>  <chr>
 1 AACG   ATA Creativity Globa~
 2 AACQ   Artius Acquisition I~
 3 AACQU  Artius Acquisition I~
 4 AACQW  Artius Acquisition I~
 5 AAL    American Airlines Gr~
 6 AAME   Atlantic American Co~
 7 AAOI   Applied Optoelectron~
 8 AAON   AAON Inc. Common Sto~
 9 AAPL   Apple Inc. Common St~
10 AAWW   Atlas Air Worldwide ~
# ... with 4,030 more rows,
```

tq_get()

获取数据。数据来源：

Yahoo Finance - Daily stock data

美联储经济数据（FRED）

Quandl

Tiingo

Alpha Vantage

Bloomberg

```
> tq_get_options()
 [1] "stock.prices"
 [2] "stock.prices.japan"
 [3] "dividends"
 [4] "splits"
 [5] "economic.data"
 [6] "quandl"
 [7] "quandl.datatable"
 [8] "tiingo"
 [9] "tiingo.iex"
[10] "tiingo.crypto"
[11] "alphavantager"
[12] "alphavantage"
[13] "rblpapi"
```

```
> tq_get("DCOILWTICO",get="economic.data")
# A tibble: 2,681 x 3
   symbol     date       price
   <chr>      <date>     <dbl>
 1 DCOILWTICO 2011-01-03  91.6
 2 DCOILWTICO 2011-01-04  89.4
 3 DCOILWTICO 2011-01-05  90.3
 4 DCOILWTICO 2011-01-06  88.4
 5 DCOILWTICO 2011-01-07  88.1
 6 DCOILWTICO 2011-01-10  89.2
 7 DCOILWTICO 2011-01-11  91.1
 8 DCOILWTICO 2011-01-12  91.8
 9 DCOILWTICO 2011-01-13  91.4
10 DCOILWTICO 2011-01-14  91.5
# ... with 2,671 more rows
```

tq_transmute()
tq_mutate()

对量化数据进行转换（tq_transmute()）和变形（tq_mutate()）：在tidyverse环境内执行并扩展金融计算。

这两个函数是tidyquant整合功能的核心体现，可以对金融数据进行相应的转换和变形。它们的区别如下：

tq_transmute()：返回一个新的tibble数据框，通常与输入的tibble数据框有着不同的周期（即，往往与输入的tibble有着不同的行数）。

tq_mutate()：在原有tibble数据框中添加新的列（即，与输入tibble有相同的行数）。

当数据因为改变周期而改变行数时，tq_transmute() 特别有用。

下面的例子将改变数据的周期，从每日数据变为月度数据。这时，你需要使用 tq_transmute() 来完成这一操作，因为数据的行数改变了。

```
# Change periodicity from daily to monthly using to.period from xts
FANG_data_m <- FANG_data_d %>%
    group_by(symbol) %>%
    tq_transmute(
        select      = adjusted,
        mutate_fun  = to.period,
        period      = "months")

FANG_data_m
```

```
## # A tibble: 144 x 3
## # Groups:   symbol [4]
##    symbol       date adjusted
##     <chr>     <date>    <dbl>
##  1     FB 2014-01-31    62.57
##  2     FB 2014-02-28    68.46
##  3     FB 2014-03-31    60.24
##  4     FB 2014-04-30    59.78
##  5     FB 2014-05-30    63.30
##  6     FB 2014-06-30    67.29
##  7     FB 2014-07-31    72.65
##  8     FB 2014-08-29    74.82
##  9     FB 2014-09-30    79.04
## 10     FB 2014-10-31    74.99
## # ... with 134 more rows
```

tq_performance()
tq_portfolio()

用于构建投资组合，
并对单个资产或投资
组合进行绩效评价

# View options查看其它可用函数
tq_performance_fun_options()

```
#> $table.funs
#>  [1] "table.AnnualizedReturns" "table.Arbitrary"
#>  [3] "table.Autocorrelation"   "table.CAPM"
#>  [5] "table.CaptureRatios"     "table.Correlation"
#>  [7] "table.Distributions"     "table.DownsideRisk"
#>  [9] "table.DownsideRiskRatio" "table.DrawdownsRatio"
#> [11] "table.HigherMoments"     "table.InformationRatio"
#> [13] "table.RollingPeriods"    "table.SFM"
#> [15] "table.SpecificRisk"      "table.Stats"
#> [17] "table.TrailingPeriods"   "table.UpDownRatios"
#> [19] "table.Variability"
#>
#> $CAPM.funs
#>  [1] "CAPM.alpha"       "CAPM.beta"        "CAPM.beta.bear"   "CAPM.beta.bull"
#>  [5] "CAPM.CML"         "CAPM.CML.slope"   "CAPM.dynamic"     "CAPM.epsilon"
#>  [9] "CAPM.jensenAlpha" "CAPM.RiskPremium" "CAPM.SML.slope"   "TimingRatio"
#> [13] "MarketTiming"
#>
#> $SFM.funs
#> [1] "SFM.alpha"       "SFM.beta"        "SFM.CML"         "SFM.CML.slope"
#> [5] "SFM.dynamic"     "SFM.epsilon"     "SFM.jensenAlpha"
```

```r
# Use FANG data set
data(FANG)
# Get returns for individual stock components grouped by symbol
Ra <- FANG %>%
  group_by(symbol) %>%
  tq_transmute(adjusted, periodReturn, period = "monthly", col_rename = "Ra")
# Get returns for SP500 as baseline
Rb <- "^GSPC" %>%
  tq_get(get  = "stock.prices",
         from = "2010-01-01",
         to   = "2015-12-31") %>%
  tq_transmute(adjusted, periodReturn, period = "monthly", col_rename = "Rb")
```

Get performance metrics获得该投资组合的性能指标

RaRb %>%

tq_performance(Ra = Ra, Rb = Rb, performance_fun = table.CAPM)

```
#> # A tibble: 4 x 13
#> # Groups:   symbol [4]
#>   symbol ActivePremium  Alpha AnnualizedAlpha  Beta `Beta-` `Beta+` Correlation
#>   <chr>          <dbl>  <dbl>           <dbl> <dbl>   <dbl>   <dbl>       <dbl>
#> 1 FB             0.431 0.034           0.493 0.846  0.819     3.00       0.234
#> 2 AMZN           0.246 0.0144          0.187 1.46  -0.0442    2.04       0.524
#> 3 NFLX           1.02  0.0632          1.09  1.35  -2.78      1.90       0.234
#> 4 GOOG           0.142 0.0123          0.158 0.901 -0.247     1.56       0.451
#> # … with 5 more variables: Correlationp-value <dbl>, InformationRatio <dbl>,
#> #   R-squared <dbl>, TrackingError <dbl>, TreynorRatio <dbl>
```

运用实例

PART.03


集成R中
的量化
分析包
建模和扩展财务分析
绘制图表
利用tidyquant进行绩效分析

集成R中的量化分析包

使用tq_transmute_fun_options()可查看tq_transmute() 与 tq_mutate()在其他包中的可用函数。

tq_transmute_fun_options() %>% str()

```
> tq_transmute_fun_options() %>% str()
List of 5
 $ zoo                 : chr [1:14] "rollapply" "rollapplyr" "rollmax" "rollmax.default"
 ...
 $ xts                 : chr [1:27] "apply.daily" "apply.monthly" "apply.quarterly" "apply.
weekly" ...
 $ quantmod            : chr [1:25] "allReturns" "annualReturn" "ClCl" "dailyReturn" ...
 $ TTR                 : chr [1:63] "adjRatios" "ADX" "ALMA" "aroon" ...
 $ PerformanceAnalytics: chr [1:7] "Return.annualized" "Return.annualized.excess" "Return.c
lean" "Return.cumulative" ...
```

可在tq_transmute_fun_options()后加上包名称查询特定包的可用函数。

```
> tq_transmute_fun_options()$xts
 [1] "apply.daily"     "apply.monthly"   "apply.quarterly" "apply.weekly"
 [5] "apply.yearly"    "diff.xts"        "lag.xts"         "period.apply"
 [9] "period.max"      "period.min"      "period.prod"     "period.sum"
[13] "periodicity"     "to.daily"        "to.hourly"       "to.minutes"
[17] "to.minutes10"    "to.minutes15"    "to.minutes3"     "to.minutes30"
[21] "to.minutes5"     "to.monthly"      "to.period"       "to.quarterly"
[25] "to.weekly"       "to.yearly"       "to_period"
```

建模和扩展财务分析

tidyquant的最大好处是能够应用数据科学工作流来轻松建模和扩展财务分析。
由于所有tidyquant函数均以tibble格式返回数据，从而允许在tidyverse中进行交互。
这意味着我们可以：

- 无缝扩展数据检索和变异
- 使用（%>%）进行链接操作
- 使 用 dplyr 和 tidyr：select, filter, group_by, nest/unnest, spread/gather等
- 使用purrr：map

建模和扩展财务分析

示例1 使用quantmod的periodReturn将收盘价转化为收益

#使用FANG数据集，其中包括从2013年初到2016年底的FB，AMZN，NFLX和GOOG的股价。

```
> data("FANG")
> FANG
# A tibble: 4,032 x 8
   symbol date        open  high   low close    volume adjusted
   <chr>  <date>     <dbl> <dbl> <dbl> <dbl>     <dbl>    <dbl>
 1 FB     2013-01-02  27.4  28.2  27.4  28    69846400     28
 2 FB     2013-01-03  27.9  28.5  27.6  27.8  63140600     27.8
 3 FB     2013-01-04  28.0  28.9  27.8  28.8  72715400     28.8
 4 FB     2013-01-07  28.7  29.8  28.6  29.4  83781800     29.4
 5 FB     2013-01-08  29.5  29.6  28.9  29.1  45871300     29.1
 6 FB     2013-01-09  29.7  30.6  29.5  30.6 104787700     30.6
 7 FB     2013-01-10  30.6  31.5  30.3  31.3  95316400     31.3
 8 FB     2013-01-11  31.3  32.0  31.1  31.7  89598000     31.7
 9 FB     2013-01-14  32.1  32.2  30.6  31.0  98892800     31.0
10 FB     2013-01-15  30.6  31.7  29.9  30.1 173242600     30.1
# ... with 4,022 more rows
```

建模和扩展财务分析

```
> FANG_annual_returns <- FANG %>%
+   group_by(symbol) %>%
+   tq_transmute(select     = adjusted,
+                mutate_fun = periodReturn,
+                period     = "yearly",
+                type       = "arithmetic")
> FANG_annual_returns
# A tibble: 16 x 3
# Groups:   symbol [4]
   symbol date       yearly.returns
   <chr>  <date>              <dbl>
 1 FB     2013-12-31          0.952
 2 FB     2014-12-31          0.428
 3 FB     2015-12-31          0.341
 4 FB     2016-12-30          0.0993
 5 AMZN   2013-12-31          0.550
 6 AMZN   2014-12-31         -0.222
 7 AMZN   2015-12-31          1.18
 8 AMZN   2016-12-30          0.109
 9 NFLX   2013-12-31          3.00
10 NFLX   2014-12-31         -0.0721
11 NFLX   2015-12-31          1.34
12 NFLX   2016-12-30          0.0824
13 GOOG   2013-12-31          0.550
14 GOOG   2014-12-31         -0.0597
15 GOOG   2015-12-31          0.442
16 GOOG   2016-12-30          0.0171
```

获取年度收益

```
FANG_daily_log_returns <- FANG %>%
    group_by(symbol) %>%
    tq_transmute(select     = adjusted,
                 mutate_fun = periodReturn,
                 period     = "daily",
                 type       = "log",
                 col_rename = "monthly.returns")
```

获取每日对数收益

建模和扩展财务分析

示例2：使用purrr建模财务数据—— 应用回归模型检测正趋势

- 个股建模：

```
> AAPL <- tq_get("AAPL", from = "2007-01-01", to = "2016-12-31")
> AAPL
# A tibble: 2,518 x 8
   symbol date        open  high   low close     volume adjusted
   <chr>  <date>     <dbl> <dbl> <dbl> <dbl>      <dbl>    <dbl>
 1 AAPL   2007-01-03  3.08  3.09  2.92  2.99 1238319600     2.58
 2 AAPL   2007-01-04  3.00  3.07  2.99  3.06  847260400     2.64
 3 AAPL   2007-01-05  3.06  3.08  3.01  3.04  834741600     2.62
 4 AAPL   2007-01-08  3.07  3.09  3.05  3.05  797106800     2.63
 5 AAPL   2007-01-09  3.09  3.32  3.04  3.31 3349298400     2.85
 6 AAPL   2007-01-10  3.38  3.49  3.34  3.46 2952880000     2.98
 7 AAPL   2007-01-11  3.43  3.46  3.40  3.42 1440252800     2.95
 8 AAPL   2007-01-12  3.38  3.40  3.33  3.38 1312690400     2.91
 9 AAPL   2007-01-16  3.42  3.47  3.41  3.47 1244076400     2.99
10 AAPL   2007-01-17  3.48  3.49  3.39  3.39 1646260000     2.92
# ... with 2,508 more rows
```

建模和扩展财务分析

示例2：使用purrr建模财务数据—— 应用回归模型检测正趋势

- 获取年度对数收益

```
get_annual_returns <- function(stock.returns) {
    stock.returns %>%
        tq_transmute(select     = adjusted,
                     mutate_fun = periodReturn,
                     type       = "log",
                     period     = "yearly")
}

AAPL_annual_log_returns <- get_annual_returns(AAPL)
AAPL_annual_log_returns
```

```
# A tibble: 10 x 2
   date       yearly.returns
   <date>              <dbl>
 1 2007-12-31         0.860
 2 2008-12-31        -0.842
 3 2009-12-31         0.904
 4 2010-12-31         0.426
 5 2011-12-30         0.228
 6 2012-12-31         0.282
 7 2013-12-31         0.0776
 8 2014-12-31         0.341
 9 2015-12-31        -0.0306
10 2016-12-30         0.118
```

建模和扩展财务分析

示例2：使用purrr建模财务数据—— 应用回归模型检测正趋势

使用lm()获得线性模型

```
mod <- lm(yearly.returns ~ year(date), data = AAPL_annual_log_returns)
mod
```

```
##
## Call:
## lm(formula = yearly.returns ~ year(date), data = AAPL_annual_log_returns)
##
## Coefficients:
## (Intercept)   year(date)
##    58.86283     -0.02915
```

使用broom包的tidy()对数据进行处理

```
library(broom)
tidy(mod)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)  58.9     113.         0.520   0.617
## 2 year(date)   -0.0291    0.0562    -0.518   0.618
```

建模和扩展财务分析

示例2：使用purrr建模财务数据—— 应用回归模型检测正趋势

```
get_model <- function(stock_data) {
    annual_returns <- get_annual_returns(stock_data)
    mod <- lm(yearly.returns ~ year(date), data = annual_returns)
    tidy(mod)
}
```

```
get_model(AAPL)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)  58.9     113.         0.520   0.617
## 2 year(date)   -0.0291    0.0562    -0.518   0.618
```

自定义函数完成以上操作

建模和扩展财务分析

示例2：使用purrr建模财务数据—— 应用回归模型检测正趋势

- 扩展到多只股票

```
set.seed(10)
stocks_tbl <- tq_index("SP500") %>%
    sample_n(5)
stocks_tbl
```

```
## # A tibble: 5 x 8
##   symbol company    identifier sedol  weight sector   shares_held local_currency
##   <chr>  <chr>      <chr>      <chr>   <dbl> <chr>          <dbl> <chr>
## 1 LEG    Leggett &… 52466010   2510… 1.89e-4 Consume…     1359861 USD
## 2 ECL    Ecolab In… 27886510   2304… 1.60e-3 Materia…     2542874 USD
## 3 HOLX   Hologic I… 43644010   2433… 5.74e-4 Health …     2633340 USD
## 4 NVR    NVR Inc.   62944T10   2637… 4.86e-4 Consume…       35800 USD
## 5 MRO    Marathon … 56584910   2910… 2.85e-4 Energy        8106363 USD
```

从S＆P500中随机抽取十只股票，取其中五只

```
stocks_model_stats <- stocks_tbl %>%
    select(symbol, company) %>%
    tq_get(from = "2007-01-01", to = "2016-12-31") %>%

    # Nest
    group_by(symbol, company) %>%
    nest() %>%

    # Apply the get_model() function to the new "nested" data column
    mutate(model = map(data, get_model)) %>%

    # Unnest and collect slope
    unnest(model) %>%
    filter(term == "year(date)") %>%
    arrange(desc(estimate)) %>%
    select(-term)

stocks_model_stats
```

```
## # A tibble: 5 x 7
## # Groups:   symbol, company [5]
##   symbol company              data          estimate std.error statistic p.value
##   <chr>  <chr>                <list>           <dbl>     <dbl>     <dbl>   <dbl>
## 1 HOLX   Hologic Inc.         <tibble [2,5…   0.0397    0.0425     0.935   0.377
## 2 LEG    Leggett & Platt Inc… <tibble [2,5…   0.0292    0.0199     1.46    0.181
## 3 NVR    NVR Inc.             <tibble [2,5…   0.0229    0.0214     1.07    0.315
## 4 ECL    Ecolab Inc.          <tibble [2,5…   0.0112    0.0228     0.493   0.635
## 5 MRO    Marathon Oil Corpor… <tibble [2,5…  -0.0140    0.0501    -0.280   0.787
```

进行扩展

扩展时的错误处理

无法检索的股票指数，股票代码，FRED数据代码时，tq_get()函数将产生错误并返回NA值。

```
tq_get("XYZ", "stock.prices")
```

```
## [1] NA
```

这种处理方法使长时间运行的脚本不会因为一个错误而中断，但如果用户未阅读警告，可能会将错误延续下去。

利用tidyquant绘制图表

tidyquant包中含有可配合ggplot2使用的可视化方法，主要包括：

1）绘制K线图：geom_barchart，geom_candlestick

2）移动平均：geom_ma

3）显示布林带（Bollinger Band）：geom_bands

4）放大日期范围：coord_x_date，coor_x_datetime

5）绘图风格：theme_tq()，theme_tq_dark()，theme_tq_green()

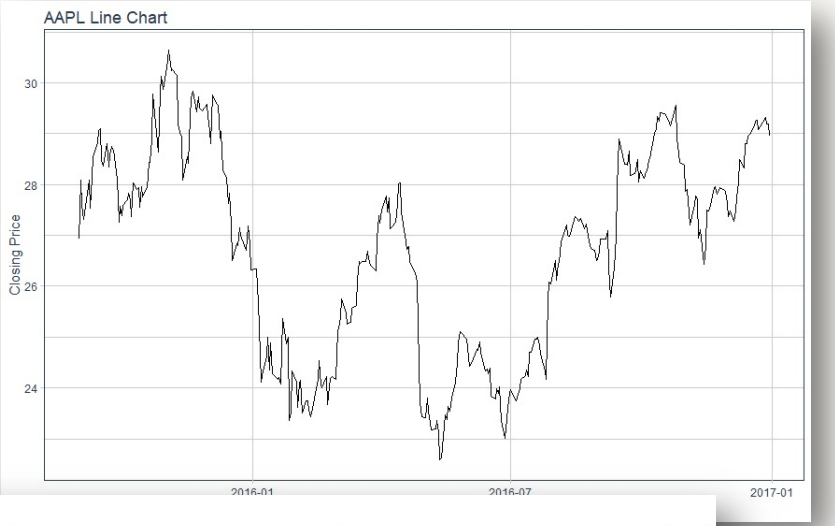
利用tidyquant绘制图表

K线图：基本图形绘制

geom_barchart/geom_candlestick(aes(open,high,low,close),fill_up,

colour_up,fill_down,colour_down)

```
AAPL %>%
    ggplot(aes(x = date, y = close)) +
    geom_line() +
    / geom_barchart(aes(open = open, high = high, low = low, close = close)) +
    / geom_candlestick(aes(open = open, high = high, low = low, close = close)) +
    labs(title = "AAPL Line Chart", y = "Closing Price", x = "") +
    theme_tq()
```

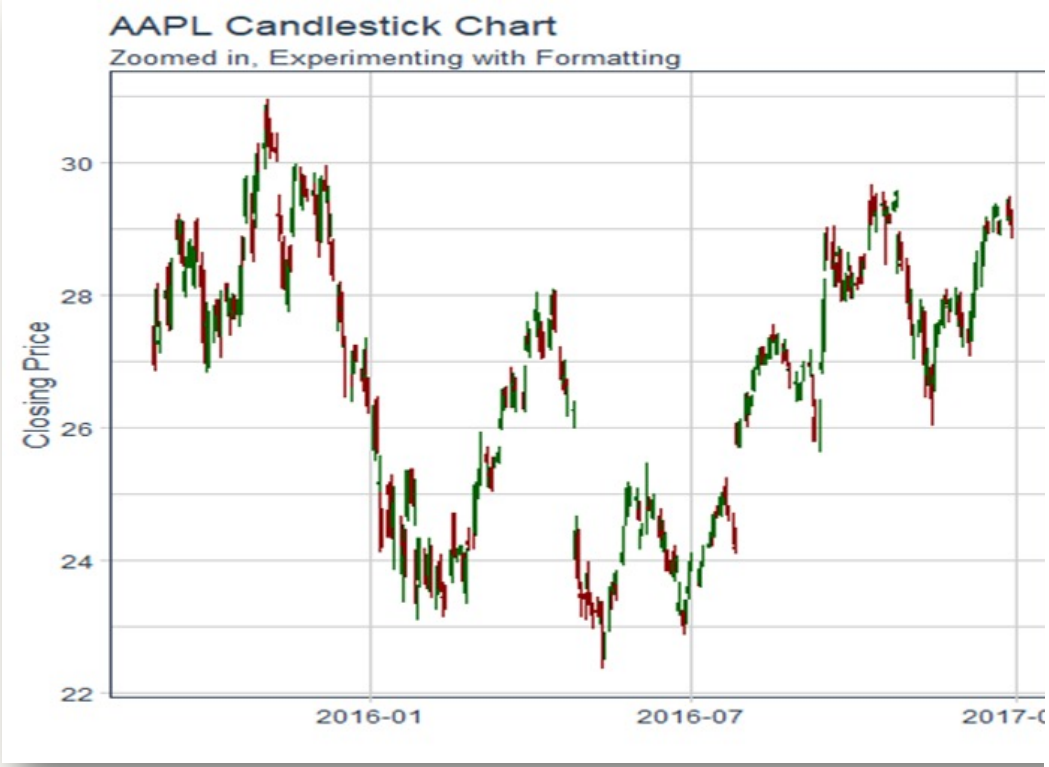


利用tidyquant绘制图表

K线图：边框色及填充色

geom_barchart/geom_candlestick(aes(open,high,low,close),**fill_up,colour_up,fill_down,colour_down**)

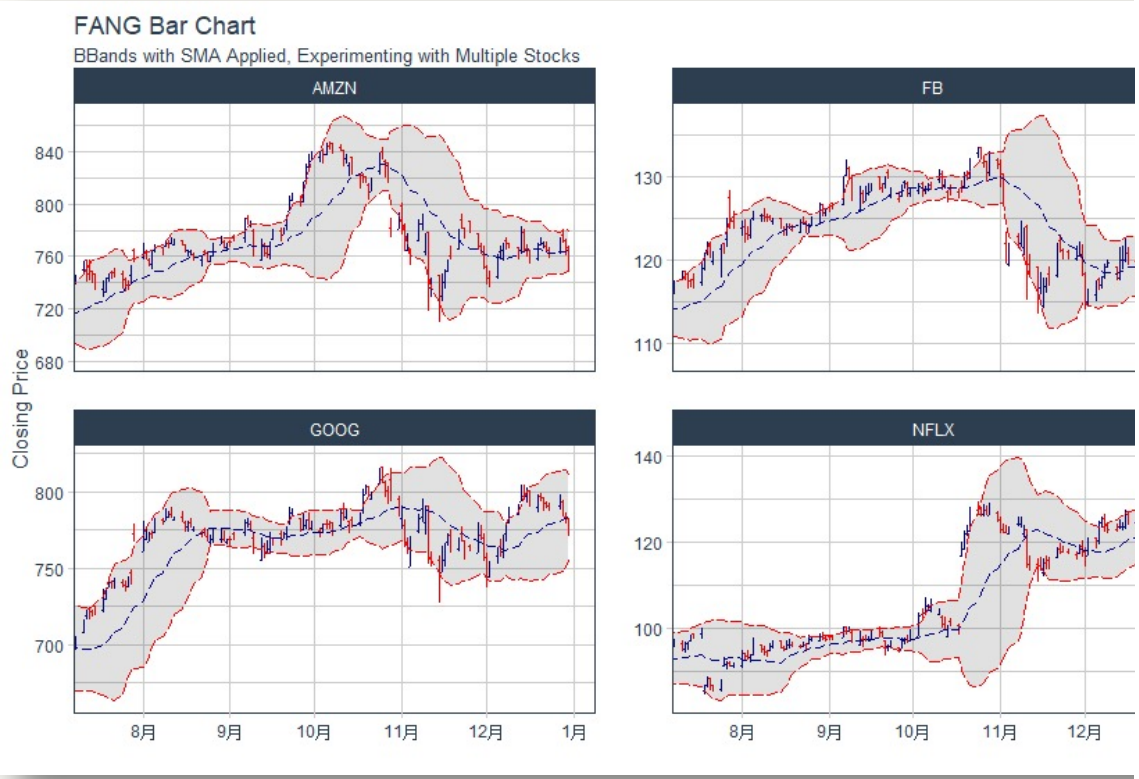
```
AAPL %>%
  ggplot(aes(x = date, y = close)) +
  geom_candlestick(aes(open = open, high = high, low = low, close = close),
                   colour_up = "darkgreen", colour_down = "darkred",
                   fill_up  = "darkgreen", fill_down  = "darkred") +
  labs(title = "AAPL Candlestick Chart",
       subtitle = "Zoomed in, Experimenting with Formatting",
       y = "Closing Price", x = "") +
  theme_tq()
```



利用tidyquant绘制图表

布林带: geom_bband(ma_fun,sd,color,size,n,linetype)

```
start <- end - weeks(24)
FANG %>%
    filter(date >= start - days(2 * 20)) %>%
    ggplot(aes(x = date, y = close,
               open = open, high = high, low = low, close = close,
               group = symbol)) +
    geom_barchart() +
    geom_bbands(ma_fun = SMA, sd = 2, n = 20, linetype = 5) +
    labs(title = "FANG Bar Chart",
         subtitle = "BBands with SMA Applied, Experimenting with Multiple Stocks",
         y = "Closing Price", x = "") +
    coord_x_date(xlim = c(start, end)) +
    facet_wrap(~ symbol, ncol = 2, scales = "free_y") +
    theme_tq()
```



利用tidyquant绘制图表

图表风格: geom_tq() / theme_tq_dark() / theme_tq_dark()

```
AAPL %>%
    ggplot(aes(x = date, y = close)) +
    geom_line() +
    labs(title = "AAPL Line Chart", y = "Closing Price", x = "") +
    theme_tq()
    / theme_tq_dark()
    / theme_tq_green()
```

绩效分析

Tidyquant整合了PerformanceAnalytics包中进行金融资产和投资组合绩效分析的功能，这些功能可以通过使用tq_performance和tq_portfolio来体现。

tq_performance：金融资产的绩效分析，可以使用split、apply、combined等框架进行拓展分析

tq_portfolio：将一组金融资产汇总到一个或多个投资组合中

绩效分析

绩效分析: tq_performance(data, Ra, Rb = NULL, performance_fun, ...)

其中，data表示将要分析资产的数据；
Rb和Ra分别表示基准利率和金融资产的投资回报率；performance_fun表示用于绩效分析的函数，这些函数包含在PerformanceAnalytics包中，可用函数的具体名称可以调用tq_performance_fun_options查看。

绩效分析

数据准备（获取苹果、谷歌、网飞的月度收益率数据）：

```
library(tidyverse)
library(tidyquant)
stock_returns_monthly <- c("AAPL", "GOOG", "NFLX") %>%
    tq_get(get  = "stock.prices",
           from = "2010-01-01",
           to   = "2015-12-31") %>%
    group_by(symbol) %>%
    tq_transmute(select     = adjusted,
                 mutate_fun = periodReturn,
                 period     = "monthly",
                 col_rename = "Ra")
stock_returns_monthly
```

```
# A tibble: 216 x 3
# Groups:   symbol [3]
   symbol date            Ra
   <chr>  <date>       <dbl>
 1 AAPL   2010-01-29 -0.103
 2 AAPL   2010-02-26  0.0654
 3 AAPL   2010-03-31  0.148
 4 AAPL   2010-04-30  0.111
 5 AAPL   2010-05-28 -0.0161
 6 AAPL   2010-06-30 -0.0208
 7 AAPL   2010-07-30  0.0227
 8 AAPL   2010-08-31 -0.0550
 9 AAPL   2010-09-30  0.167
10 AAPL   2010-10-29  0.0607
# ... with 206 more rows
```

绩效分析

夏普比率

```
stock_returns_monthly %>%
    tq_performance(
        Ra = Ra,
        Rb = NULL,
        performance_fun = SharpeRatio
    )
```

```
# Groups:   symbol [3]
  symbol `ESSharpe(Rf=0%,~ `StdDevSharpe(Rf=~ `VaRSharpe(Rf=0~
  <chr>              <dbl>              <dbl>             <dbl>
1 AAPL               0.173              0.292             0.218
2 GOOG               0.129              0.203             0.157
3 NFLX               0.237              0.284             0.272
```

绩效分析

统计指标

```
stock_returns_monthly %>%
    tq_performance(
        Ra = Ra,
        Rb = NULL,
        performance_fun = table.Stats
    )
```

```
# A tibble: 3 x 17
# Groups:   symbol [3]
  symbol ArithmeticMean GeometricMean Kurtosis `LCLMean(0.95)`
  <chr>           <dbl>         <dbl>    <dbl>           <dbl>
1 AAPL           0.0211        0.0186   -0.323          0.0041
2 GOOG           0.0154        0.0126    0.0689        -0.0025
3 NFLX           0.0572        0.0386    3.14           0.0099
```

绩效分析

```
# Groups:   symbol [3]
  symbol AnnualizedReturn `AnnualizedSharpe(~ AnnualizedStdDev
  <chr>             <dbl>                <dbl>             <dbl>
1 AAPL              0.248                0.988             0.251
2 GOOG              0.163                0.618             0.263
3 NFLX              0.575                0.825             0.698
```

年利率计算

```
stock_returns_monthly %>%
    tq_performance(
        Ra = Ra,
        Rb = NULL,
        performance_fun = table.AnnualizedReturns
    )
```

绩效分析

投资组合构建: tq_portfolio(assets.col, returns.col, weights, col_rename,...)

Tq_portfolio函数使用基础资产的加权组合将一组单个资产聚合成一整个投资组合。

投资组合的构建可以通过两种方式：

①提供权向量以构建投资组合；

②提供tibble（需要提前加载dplyr或tibble包）数据框，其中数据框的第一列是股票名，第二列是对应的权重。

绩效分析

投资组合构建: tq_portfolio(assets.col, returns.col, weights, col_rename,...)

① 提供权向量以构建投资组合：

```
wts <- c(0.5, 0.0, 0.5)
portfolio_returns_monthly <- stock_returns_monthly %>%
    tq_portfolio(assets_col  = symbol,
                 returns_col = Ra,
                 weights     = wts,
                 col_rename  = "Ra")
portfolio_returns_monthly
```

```
# A tibble: 72 x 2
   date              Ra
   <date>         <dbl>
 1 2010-01-29  0.0307
 2 2010-02-26  0.0629
 3 2010-03-31  0.130
 4 2010-04-30  0.239
 5 2010-05-28  0.0682
 6 2010-06-30 -0.0219
 7 2010-07-30 -0.0272
 8 2010-08-31  0.116
 9 2010-09-30  0.251
10 2010-10-29  0.0674
# ... with 62 more rows
```

绩效分析

投资组合构建: tq_portfolio(assets.col, returns.col, weights, col_rename,...)

②提供tibble数据框以进行投资组合的构建：

```
wts_map <- tibble(
    symbols = c("AAPL", "NFLX"),
    weights = c(0.5, 0.5)
)
stock_returns_monthly %>%
    tq_portfolio(assets_col  = symbol,
                 returns_col = Ra,
                 weights     = wts_map,
                 col_rename  = "Ra_using_wts_map")
```

绩效分析

投资组合构建: tq_portfolio(assets.col, returns.col, weights, col_rename,...)

同时进行多个投资组合的构建：

股票收益率数据准备：

```
stock_returns_monthly <- c("AAPL", "GOOG", "NFLX") %>%
    tq_get(get  = "stock.prices",
           from = "2010-01-01",
           to   = "2015-12-31") %>%
    group_by(symbol) %>%
    tq_transmute(select     = adjusted,
                 mutate_fun = periodReturn,
                 period     = "monthly",
                 col_rename = "Ra")
stock_returns_monthly_multi <- stock_returns_monthly %>%
    tq_repeat_df(n = 3)
stock_returns_monthly_multi
```

```
# A tibble: 648 x 4
# Groups:   portfolio [3]
   portfolio symbol date           Ra
       <int> <chr>  <date>      <dbl>
 1         1 AAPL   2010-01-29 -0.103
 2         1 AAPL   2010-02-26  0.0654
 3         1 AAPL   2010-03-31  0.148
 4         1 AAPL   2010-04-30  0.111
 5         1 AAPL   2010-05-28 -0.0161
 6         1 AAPL   2010-06-30 -0.0208
 7         1 AAPL   2010-07-30  0.0227
 8         1 AAPL   2010-08-31 -0.0550
 9         1 AAPL   2010-09-30  0.167
10         1 AAPL   2010-10-29  0.0607
# ... with 638 more rows
```

绩效分析

投资组合构建: tq_portfolio(assets.col, returns.col, weights, col_rename,...)

权重数据框构建：

```
weights <- c(
    0.50, 0.25, 0.25,
    0.25, 0.50, 0.25,
    0.25, 0.25, 0.50
)
stocks <- c("AAPL", "GOOG", "NFLX")
weights_table <-  tibble(stocks) %>%
    tq_repeat_df(n = 3) %>%
    bind_cols(tibble(weights)) %>%
    group_by(portfolio)
weights_table
```

```
# A tibble: 9 x 3
# Groups:   portfolio [3]
  portfolio stocks weights
      <int> <chr>     <dbl>
1         1 AAPL       0.5
2         1 GOOG       0.25
3         1 NFLX       0.25
4         2 AAPL       0.25
5         2 GOOG       0.5
6         2 NFLX       0.25
7         3 AAPL       0.25
8         3 GOOG       0.25
9         3 NFLX       0.5
```

绩效分析

投资组合构建：

```
portfolio_returns_monthly_multi <- stock_returns_monthly_multi %>%
    tq_portfolio(assets_col  = symbol,
                 returns_col = Ra,
                 weights     = weights_table,
                 col_rename  = "Ra")
portfolio_returns_monthly_multi
```

```
# A tibble: 216 x 3
# Groups:   portfolio [3]
   portfolio date           Ra
       <int> <date>      <dbl>
 1         1 2010-01-29 -0.0489
 2         1 2010-02-26  0.0482
 3         1 2010-03-31  0.123
 4         1 2010-04-30  0.145
 5         1 2010-05-28  0.0245
 6         1 2010-06-30 -0.0308
 7         1 2010-07-30  0.000600
 8         1 2010-08-31  0.0474
 9         1 2010-09-30  0.222
10         1 2010-10-29  0.0789
# ... with 206 more rows
```

绩效分析

对投资组合进行性能分析：

```
RaRb_single_portfolio <- left_join(portfolio_returns_monthly,
                                   baseline_returns_monthly,
                                   by = "date")
RaRb_single_portfolio %>%
  tq_performance(Ra = Ra, Rb = Rb, performance_fun = table.CAPM)
```

```
# A tibble: 1 x 12
  ActivePremium  Alpha AnnualizedAlpha  Beta `Beta-` `Beta+`
          <dbl>  <dbl>           <dbl> <dbl>   <dbl>   <dbl>
1         0.327 0.0299           0.425 0.754  -0.243   0.503
# ... with 6 more variables: Correlation <dbl>,
#   Correlationp-value <dbl>, InformationRatio <dbl>,
#   R-squared <dbl>, TrackingError <dbl>, TreynorRatio <dbl>
```

THANK YOU

SENYAN