

Quantitative Financial Modelling Framework

Aplicaciones y usos en R

Gabriel Cabrera G.

26 de marzo del 2018

- 1 Introducción a Quantmod
- 2 Manos a la Obra: Obtención de Datos
- 3 Gráficos usando Quantmod
- 4 Análisis Técnico
- 5 Múltiples datos
- 6 Aplicación I

Introducción a Quantmod

¿Qué es Quantmod?

- Es un paquete/librería diseñado para desarrollar, testear e implementar modelos estadísticos financieros.
- A través de la función `getSymbols` podemos extraer datos financieros desde varias fuentes: Google Finance, Yahoo Finance, Federal Reserve Bank of St. Louis FRED (más de 11,000 series !!!) y Oanda. Incluso desde fuentes propias: MySQL , R (Rdata) y Comma Separated Value files (csv).
- También se puede usar **BatchGetSymbols** como alternativa, pero no es tan organizado (espíritu *tidy*) con los datos como **Quantmod**.
- Tiene herramientas para realizar análisis técnico.
- Con `chartSeries` se puede graficar, al más puro estilo de los terminales de Bloomberg y/o Reuters, genial no. No obstante... nunca está demás `ggplot2` o `plotly` (Tufte's Principles).

Manos a la Obra: Obtención de Datos

Comencemos: Preamble

Como todo paquete se debe instalar:

```
# Instalación package  
install.packages("quantmod")
```

y posteriormente lo agregamos a nuestro script:

```
# Cargamos "quantmod"  
require("quantmod")
```

Nota: con ctrl + R en windows y cmd + R en MAC OS agregamos más rapido comentarios (sección) en Rstudio.

Función *getSymbols*

La función se compone principalmente de 5 elementos:

- Nombre de la serie (ticker o nemotecnico).
- enviaroment/ambiente (env)
- Fuente/source (src), e.g: src="google", src="yahoo", src="FRED"
- Inicio de la serie (from), e.g: as.Date("1990-01-01").
- Fin de la serie (to), e.g: as.Date("1990-01-01").

Estructura de la función getSymbols

```
getSymbols(" ", env = , src = , from = as.Date(" "), to = as.Date(" "))
```

Nota: El uso de 'env' no es necesario (ver apunte adjunto).

Obtención de Datos

A continuación obtendremos los datos del S&P 500 (**Standard & Poor 500**), aquí necesitamos saber el **ticker** o nemotécnico de la acción (**stock**) que vamos a trabajar, para *Yahoo* es **^GSPC**. Si se desea buscar otra acción basta con ir a <https://finance.yahoo.com> y extraerlo.

```
getSymbols("^GSPC", src = "yahoo", from = as.Date("1960-01-04"),  
           , to = as.Date("2010-01-01"))
```

```
## [1] "GSPC"
```


¿Como son los datos?

```
# Primera 1 observaciones con las 5 primeras columnas
```

```
head(GSPC[,1:5],1)
```

```
##           GSPC.Open GSPC.High GSPC.Low GSPC.Close GSPC.Volume
## 1960-01-04      59.91      59.91      59.91      59.91      3990000
```

```
# Ver la base
```

```
View(GSPC)
```

```
# Clase de los datos
```

```
class(GSPC)
```

```
## [1] "xts" "zoo"
```

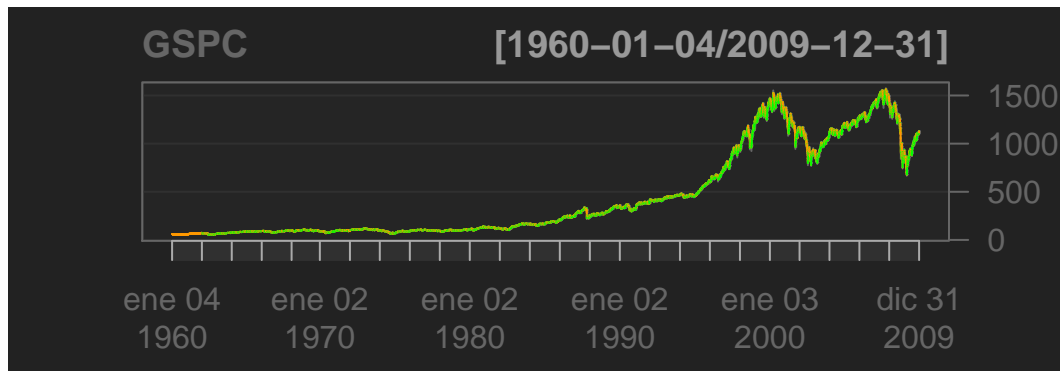
xts & zoo

- Como vamos a estar trabajando con series de tiempo, quantmod las descargará como objeto xts (*eXtensible Time Series*).
- Este es una extensión de zoo.
- La mayoría de los detalles de zoo funcionan con xts.
- No es necesario cargar la librería de zoo y xts, dado que quantmod las carga en el require.

Gráficos usando Quantmod

Función chartSeries

```
# Graficando S&P 500  
chartSeries(GSPC, TA=NULL)
```



Función chartSeries

Como se ve, en el eje de las x muestra el periodo y en el eje de las ordenadas el precio. La opción TA implica que no hay ningún análisis técnico. sin TA aparecen el volumen.

```
# Graficando S&P 500 con Valume  
chartSeries(GSPC)
```

Pero cuando las series son muy largas, podemos ver tendencias pero dificulta ver cambios importantes a nivel de análisis técnico.

```
# Graficando S&P 500 con Valume  
chartSeries(GSPC, subset = "last 3 months")
```

Con el código anterior nos enfocamos solo en los tres meses anteriores.

Análisis Técnico

Bandas de Bollinger

El análisis Técnico se divide en cuatro tipo: Tendencia (*Trend*), Volatilidad (*Volatility*), Momentum (*Momentum*) y Volumen (*Volume*). Uno de los más utilizados son las bandas de Bollinger, las que son formadas por tres líneas.

- La línea central (*Middle Line*, ML) es una media móvil simple.

$$SMA = \frac{P_M + P_{M-1} + \dots + P_{M-(n-1)}}{n} = ML$$

- La línea superior (*Top Line*, TL) es la misma línea central pero desplazada hacia arriba a un número determinado de desviaciones estándares (D).

$$TL = ML + (D * \sigma)$$

- La línea inferior (*Bottom Line*, BL) es la línea central desplazada hacia abajo al mismo número de desviaciones estándares.

$$BL = ML - (D * \sigma)$$

Bandas de Bollinger con quantmod

Realizar las Bandas no es difícil:

```
# Bandas de Bollinger  
chartSeries(GSPC, subset = "last 3 months")  
addBBands()
```

addBBands tiene las siguientes opciones

```
addBBands(n = 20, sd = 2, ma= "SMA", draw = "bands", on = -1)
```

Donde *n* es el numero de periodo de la media movil, *sd* las desviaciones estandar y *ma* el tipo de media movil.

Otros tipos de Análisis en quantmod

Tendencia:

INDICATOR	TTR NAME	QUANTMOD NAME
Welles Wilder's Directional Movement Indicator	ADX	addADX
Double Exponential Moving Average	DEMA	addDEMA
Exponential Moving Average	EMA	addEMA
Simple Moving Average	SMA	addSMA
Parabolic Stop and Reverse	SAR	addSAR
Exponential Volume Weighted Moving Average	EVWMA	addEVWMA
Moving Average Convergence Divergence	MACD	addMACD
Triple Smoothed Exponential Oscillator	TRIX	addTRIX
Weighted Moving Average	WMA	addWMA
ZLEMA	ZLEMA	addZLEMA

Volatility:

INDICATOR	TTR NAME	QUANTMOD NAME
Average True Range	ATR	addATR
Bollinger Bands	BBands	addBBands
Price Envelope	N/A	addEnvelope

Volume:

INDICATOR	TTR NAME	QUANTMOD NAME
Chaiken Money Flow	CMF	addCMF
Volume	N/A	addVo

Momentum:

INDICATOR	TTR NAME	QUANTMOD NAME
Commodity Channel Index	CCI	addCCI
Chande Momentum Oscillator	CMO	addCMO
Detrended Price Oscillator	DPO	addDPO
momentum	addMomentum	
Rate of Change	ROC	addROC
Relative Strength Indicator	RSI	addRSI
Stochastic Momentum Index	SMI	addSMI
Williams %R	WPR	addWPR

Multiples datos

Nvidia, AMD, Oracle e IBM

Ahora descargaremos más de una acción: Nvidia, AMD, Oracle e IBM

```
# Extracción múltiples series
getSymbols(c("ORCL", "AMD", "IBM", "NVDA"), src = "yahoo",
from = as.Date("2000-01-01"), to = as.Date("2018-01-01"))
```

En el Environment se ve que hay cuatro objetos, pero con distintas fechas. Esto es debido que las empresas comenzaron a transar en distintos periodos. Puede existir el caso en que se necesite unir los cuatro objetos considerando como inicio la fecha de la empresa más nueva, para esto:

- Se puede “pasar” a `data.frame`

```
# Ejemplo de xts a data frame
nvda <- as.data.frame(NVDA)
```

posteriormente usar `dplyr` del `tidyverse` y aplicar `merge`

```
# merge dos objetos  
base <- merge(nvda,orcl, by = "date")
```

- La otra forma que es más directa:

```
# Seleccionar una columna para crear nuevo objeto  
ORCL_1 <- ORCL$ORCL.Close
```

y usar:

```
# merge para xts  
merge_base <- merge.xts(ORCL,AMD)
```

Aplicación I

Actividad

- ➊ Usando **quantmod** descargue el índice del **IPSA**, desde el 2000 a la fecha.
- ➋ Calcule las bandas de Bollinger usando `chartSeries` (el intervalo de tiempo lo decide usted).
- ➌ Construya la serie de retorno del índice en un objeto nuevo.
- ➍ Obtenga la media y varianza de la serie construida en el punto anterior.
- ➎ Repita los item del 1 al 4 con el índice **IGPA**
- ➏ Grafique los retornos acumulados de los dos índices usando simplemente `plot`.
- ➐ Ahora con `ggplot2`.
- ➑ Guarde sus cálculos en un `csv`, para poder abrirlo en el futuro en el “programa” que usted desee.