

Financial Engineering Analytics: A Practice Manual Using R

Alvin Chung

19/02/2019

Contents

1	Introduction to Financial Analytics	1
1.1	Analytics	1
1.2	Chapter Outline	1
1.3	Nomenclature	2
2	Macrofinancial Data Analysis	2
2.1	Imagine This	2
2.2	Building the Stylized Facts	4
2.3	Gettings Caught in the Cross-Current	14

1 Introduction to Financial Analytics

1.1 Analytics

By its very nature the science of data analytics is disruptive. That means, among many other things, that much attention should be paid to the scale and range of invalid, as yet not understood, outlying, and emerging trends. This is as true within the finance domain of knowledge as any other.

Throughout the book, we will learn the core of ideas of programming software development to implement financial analyses (functions, objects, data structures, flow control, input and output, debugging, logical design and abstraction) through writing code. We will learn how to setup stochastic simulations, manage data analyses, and employ numerical optimization algorithms, diagnose their limitations, and work with and filter large data sets. Since code is also an important form of communication among analyst, we will learn how to comment and organize code, as well as document work product.

1.2 Chapter Outline

Here is an outline of topics covered by chapter.

2. R Warm-Ups for Finance. R computations, data structures, financial, probability and statistics calculations, visualization. Documentation with R Markdown

3. More R Warm-Ups. Functions, loops, control bootstrapping, simulations and more visualization.

4. Stylized Facts of Financial Markets. Data from FRED, Yahoo, and other sources. Empirical characteristics of economic and financial time series. Bootstrapping confidence intervals.

5. Term Structure of Interest Rates. Bond pricing, forward and yield curves. Estimating Non-linear regression splines. Applications.

6. Market Risk. Quantile (i.e., Value at Risk) and coherent (i.e., Expected Shortfall) risk measures

7. Credit Risk. Hazard rate models, Markov transition probabilities Risk measures, Laplace simulation with FFT.

- 8. Operational Risk and Extreme Finance.** Generate frequency and severity of operational loss distribution. Estimating operational risk distribution parameters. Simulating loss distributions.
- 9. Measuring Volatility.** Measuring volatility. GARCH estimation. GARCH simulation. Measuring Value at Risk (VaR) and Expected Shortfall (ES).
- 10. Portfolio Optimization.** Combining risk management with portfolio allocations. Optimizing allocations. Simulating the efficient frontier.
- 11. Aggregating Enterprise Risk.** Enterprise risk management analytics and applications. Workflow to build an online application. Introduction to **Shiny** and **ShinyDashboard**. Building a simple app. Using R Markdown and Shiny.

1.3 Nomenclature

Symbol	Name	Usage
α	alpha	significance level
β	beta	regression coefficient, scale parameter
γ	gamma	autocorrelation
Γ	Gamma	Gamma function
δ	delta	delta
Δ	Delta	change
ϵ	epsilon 1	is element of
ε	epsilon 2	error term
ξ	ksi	shape parameter
λ	lambda	hazard rate
μ	mu	mean of a distribution
ν	nu	number of degrees of freedom
π	pi	probability of occurrence
ρ	rho	correlation coefficient
σ	sigma	standard deviation
Σ	sum sigma	arithmetic sum
τ	tau	quantile

2 Macrofinancial Data Analysis

2.1 Imagine This

Your US-based company just landed a contract worth more than 20 percent of your company's current revenue in Spain. Now that everyone has recovered from this coup, your management wants you to

1. Retrieve and begin to analyze data about the Spanish economy

2. Compare the contrast Spanish stock market and government-issued debt value versus the United States and several other countries
3. Begin to generate economic scenarios based on political events that may, or may not, happen in Spain

Up to this point we had reviewed several ways to manipulate data in **R**. We then reviewed some basic finance and statistics concepts in **R**. We also got some idea of the financial analytics workflow.

1. What decision(s) are we making?
2. What are the key business questions we need to support this decision?
3. What data do we need?
4. What tools do we need to analyze the data?
5. How do we communicate answers to inform the decision?

2.1.1 Working an example

Let's use this workflow to motivate our work in this chapter.

1. Let's identify a decision at work (e.g., investment in new machine, financing a building, acquisition of customers, hiring talent, locating manufacturing).
2. For this decision we will list three business questions you need to inform the decision we chose.
3. Now we consider data we might need to answer one of those questions and choose from this set:
 - Macroeconomic data: GDP, inflation, wage, population
 - Financial data: stock market prices, bond prices, exchange rates, commodity prices

Here is the example using the scenario that started this chapter.

1. Our decision is **supply a new market segment**
 - Product: voltage devices with supporting software
 - Geography: Spain
 - Customers: Major buyers at Iberdrola, Repsol, and Endesa
2. We pose three business questions:
 - How would the performance of these companies affect the size and timing of orders?
 - How would the value of their product affect the value of our business with these companies?
 - We are a US functional currency firm, so how would we manage the repatriation of accounts receivable from Spain?
3. Some data and analysis to inform the decision could include
 - Customer stock prices: volatility and correlation
 - Oil prices: volatility
 - USD/EUR exchange rates: volatility
 - All together: correlation among these indicators

2.1.2 How will we proceed

This chapter will develop stylized facts of the market. These continue to be learned the hard way: financial data is not independent, it possesses volatile volatility, and has extremes.

- * Financial stock, bonds, commodity have highly interdependent relationships
- * Volatility is rarely constant and often has a structure (mean reversion) and is dependent on the past
- * Past shocks persist and may or may not dampen (rock in a pool).
- * Extreme events and likely to happen with other extreme events.
- * Negative returns are more likely than positive returns (left skew).

2.2 Building the Stylized Facts

Examples from the 70s, 80s, and 90s have multiple intersecting global events influencing decision makers. We will load some computational help and some data from Brent, format dates, and create a time series object (package zoo will be needed)

```
library(fBasics)
library(evir)
library(qrmdata)
library(zoo)
data("OIL_Brent")
str(OIL_Brent)
```

```
## An 'xts' object on 1987-05-20/2015-12-28 containing:
##   Data: num [1:7258, 1] 18.6 18.4 18.6 18.6 18.6 ...
##   - attr(*, "dimnames")=List of 2
##     ..$ : NULL
##     ..$ : chr "OIL_Brent"
##   Indexed by objects of class: [Date] TZ: UTC
##   xts Attributes:
##   NULL
```

We will compute rates of change for Brent oil prices next.

```
Brent.price <- as.zoo(OIL_Brent)
str(Brent.price)
```

```
## 'zoo' series from 1987-05-20 to 2015-12-28
##   Data: num [1:7258, 1] 18.6 18.4 18.6 18.6 18.6 ...
##   - attr(*, "dimnames")=List of 2
##     ..$ : NULL
##     ..$ : chr "OIL_Brent"
##   Index: Date[1:7258], format: "1987-05-20" "1987-05-21" "1987-05-22" "1987-05-25" "1987-05-26" ...
```

```
Brent.return <- diff(log(Brent.price))[-1] * 100
colnames(Brent.return) <- "Brent.return"
head(Brent.return, n = 5)
```

```
##           Brent.return
## 1987-05-22    0.5405419
## 1987-05-25    0.2691792
## 1987-05-26    0.1611604
## 1987-05-27   -0.1611604
## 1987-05-28    0.0000000
```

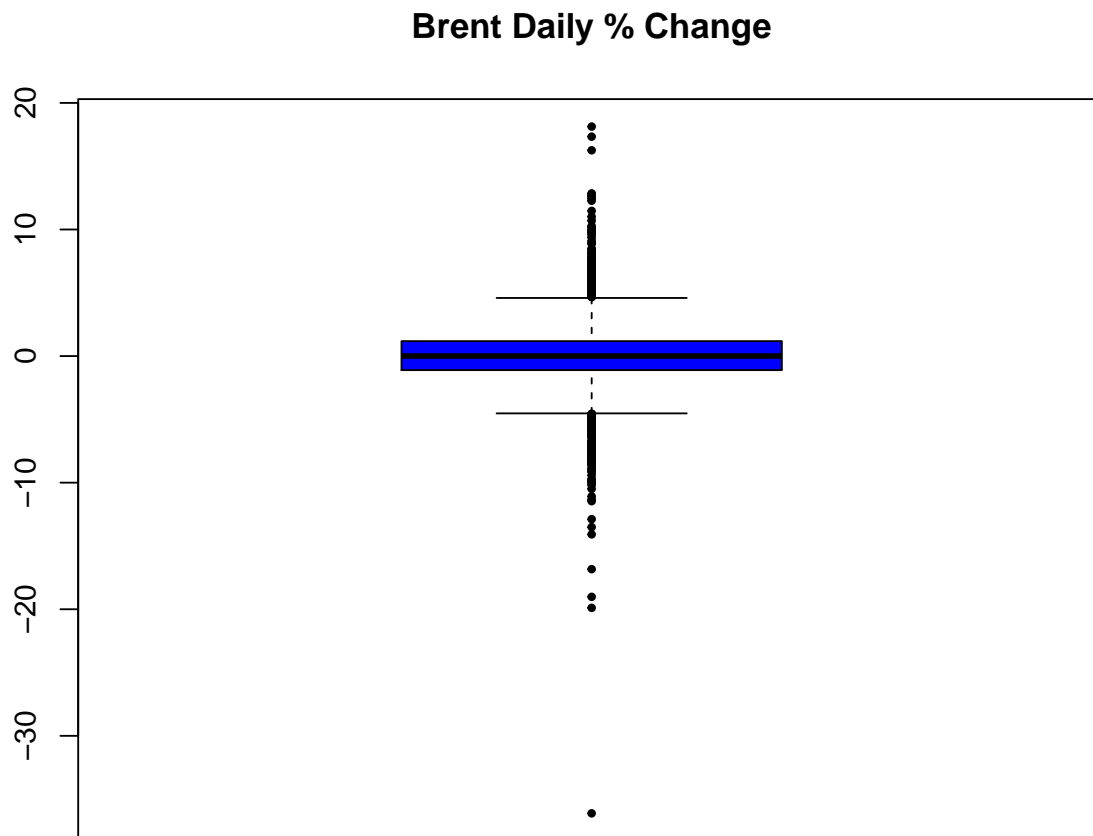
```
tail(Brent.return , n = 5)
```

```
##           Brent.return
## 2015-12-21  -3.9394831
## 2015-12-22  -0.2266290
## 2015-12-23   1.4919348
## 2015-12-24   3.9177726
## 2015-12-28  -0.3768511
```

Let's look at this data with box plots and autocorrelation functions. Box plots will show minimum to maximum with the mean in the middle of the box. Autocorrelation plots will reveal how persistent the returns are over time

We run these statements.

```
boxplot(as.vector(Brent.return), title = FALSE,
        main = "Brent Daily % Change", col = "blue",
        cex = 0.5, pch = 19)
```



```
skewness(Brent.return)
```

```
## [1] -0.6210447
```

```
## attr(,"method")
## [1] "moment"

kurtosis(Brent.return)

## [1] 14.62226
## attr(,"method")
## [1] "excess"
```

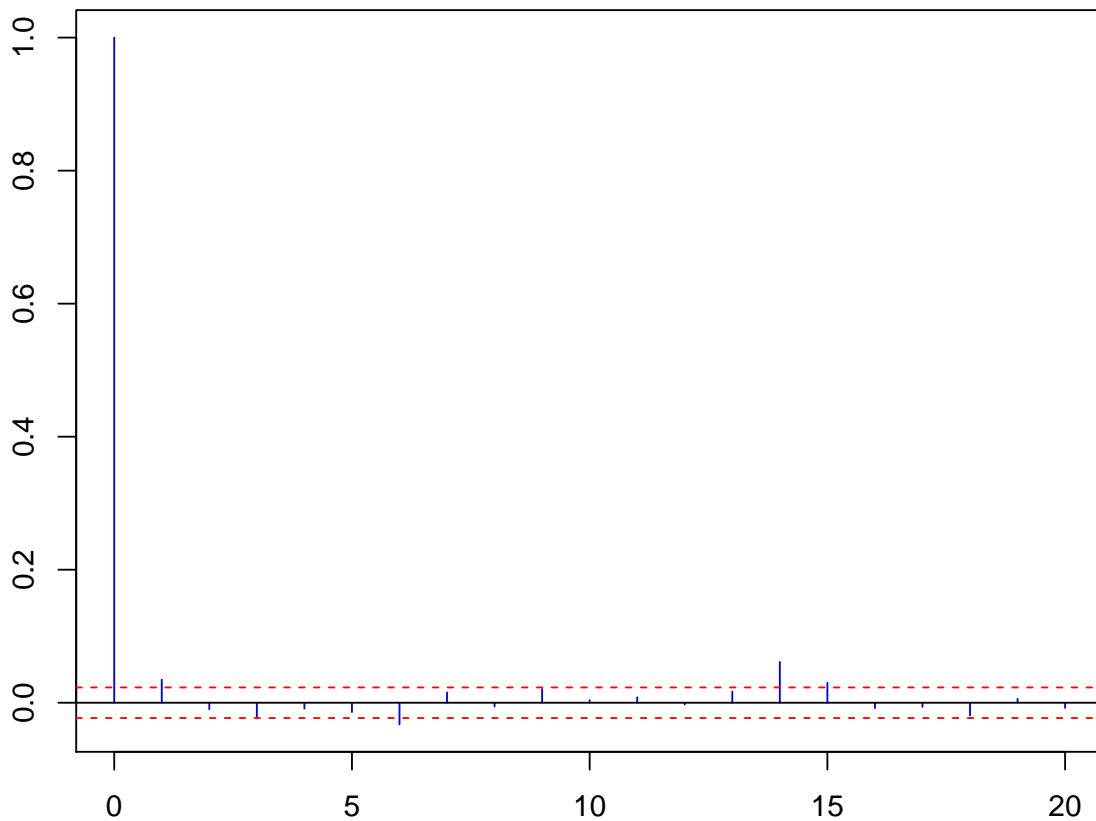
This time series plot shows lots of return clustering and spikes, especially negative ones. Performing some “eyeball econometrics” these clusters seem to occur around - The oil embargo of the ‘70s - The height of the new interest rate regime in Paul Volcker at the Fed - “Black Monday” stock market crash in 1987 - Gulf I - Baring and other derivative business collapses in the 90s’

2. Let’s look at the likelihood of a positive versus negative return. We might want to review the skewness and kurtosis definition and range to help us.

Now to look at persistence:

```
acf(coredata(Brent.return), main = "Brent Daily Autocorrelogram",
    lag.max = 20, ylab = "", xlab = "",
    col = "blue", ci.col = "red")
```

Brent Daily Autocorrelogram

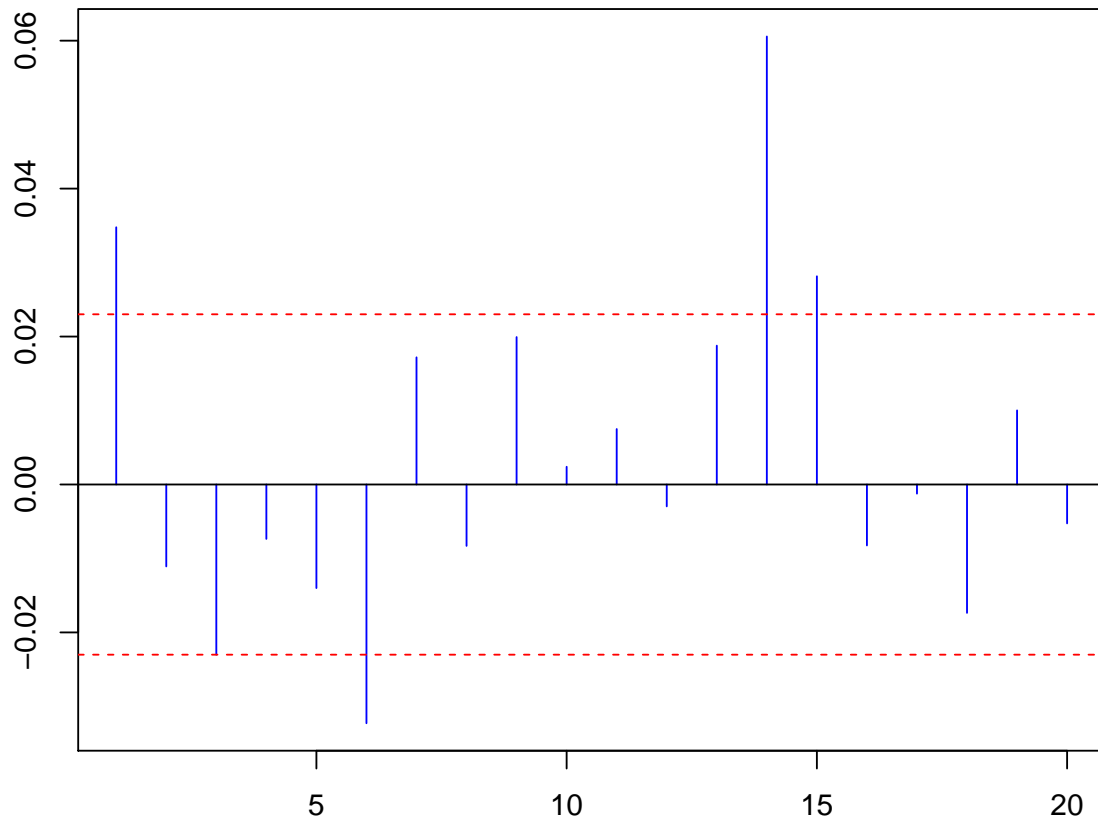


```
class(Brent.return)

## [1] "zoo"

pacf(coredata(Brent.return), main = "Brent Daily Partial Autocorrelogram",
     lag.max = 20, ylab = "", xlab = "",
     col = "blue",
     ci.col = "red")
```

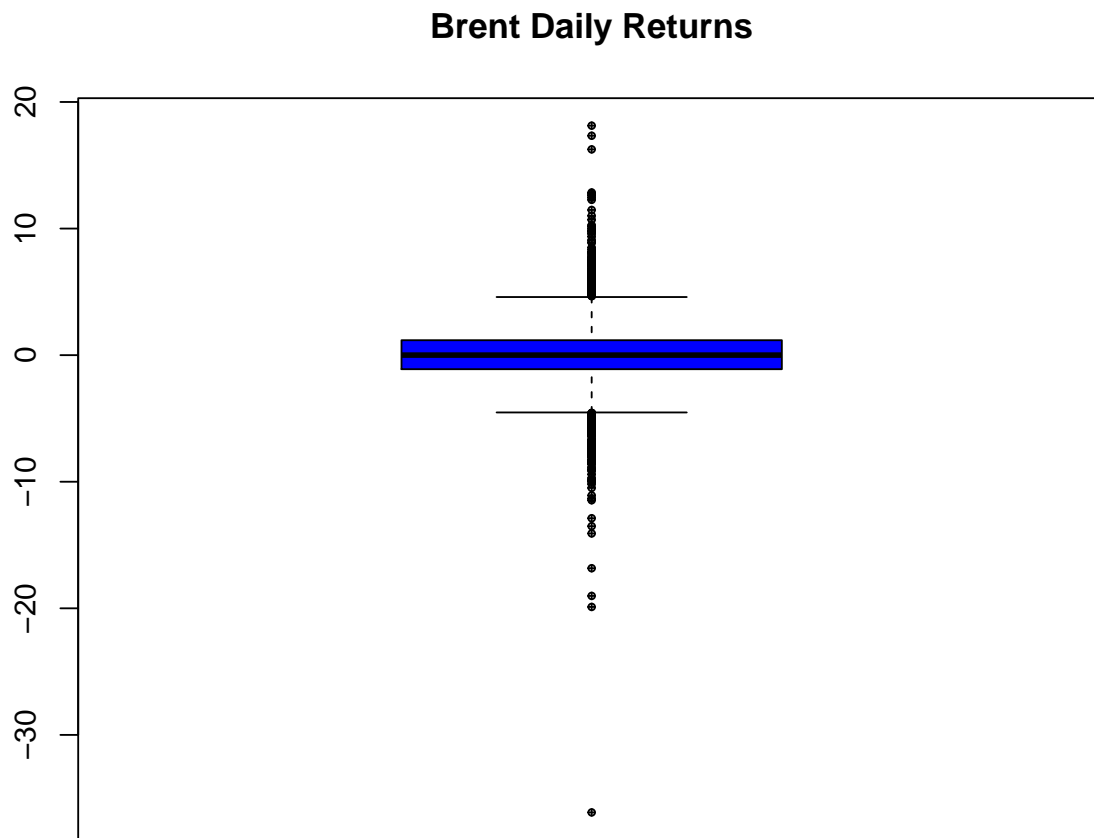
Brent Daily Partial Autocorrelogram



Confidence interval are the red dashed lines. ACF at lag 6 means the correlation of current returns correlate with returns 6 trading days ago, including any correlation from trading day 1 to trading 6. PACF is simpler: it is the raw correlation between day 0 and day 6. ACF starts at lag 0 (today); PACF starts at lag 1 (yesterday).

Here is a first look:

```
boxplot(as.vector(Brent.return), title = FALSE,
       main = "Brent Daily Returns", col = "blue",
       cex = 0.5, pch = 10)
```



with some basic stats to back up the eyeball econometrics in the box plot

```
skewness(Brent.return)
```

```
## [1] -0.6210447  
## attr(,"method")  
## [1] "moment"
```

```
kurtosis(Brent.return)
```

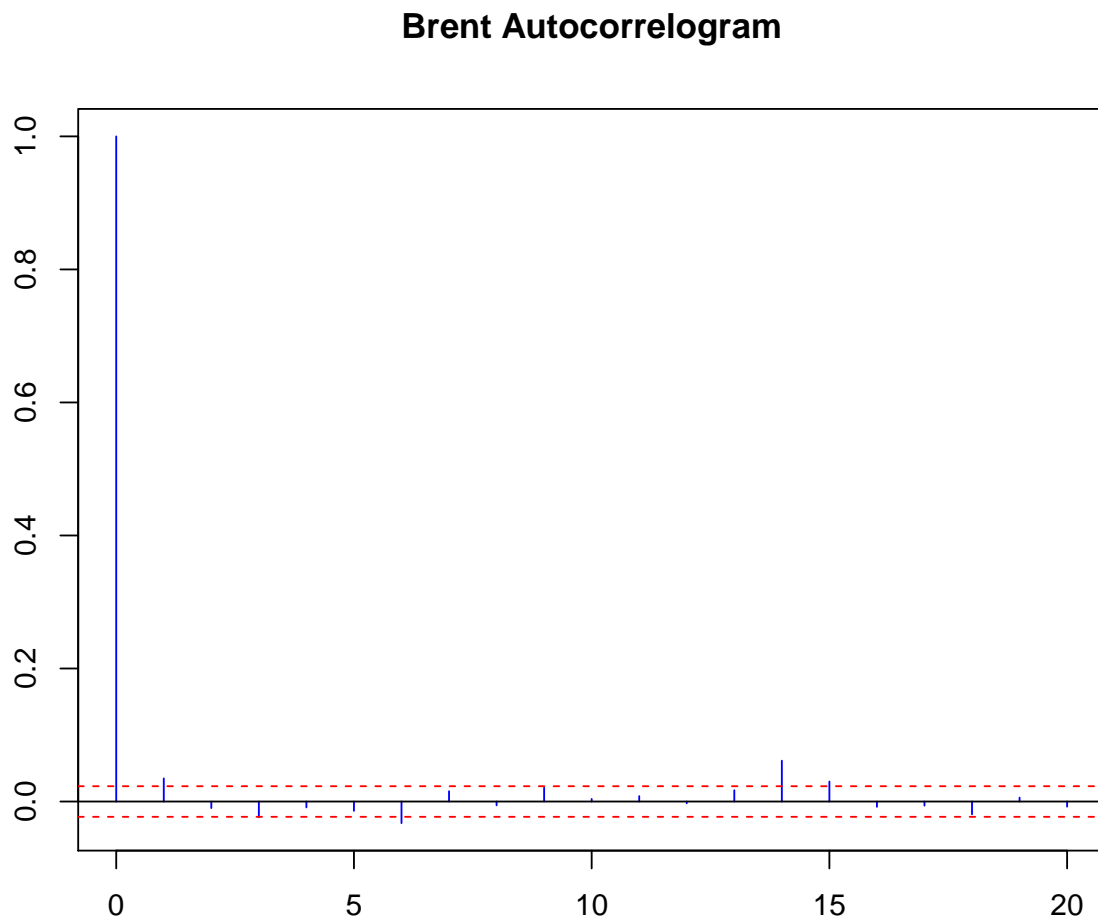
```
## [1] 14.62226  
## attr(,"method")  
## [1] "excess"
```

- A negative skew means there are more observations less than the median than greater.
- This high kurtosis means a pretty heavy tail, especially in negative returns. That means they have happened more often than positive returns.
- A preponderance of negative returns frequently happening spells trouble for anyone owning these assets.

2.2.1 Implications

- We should recommend that management budget for the body of the distribution from the mean and out to positive levels.
 - At the same time management should build a comprehensive playbook for the strong possibility that bad tail events frequently happen and might happen again (and why shouldn't they?)
3. Now for something really interesting

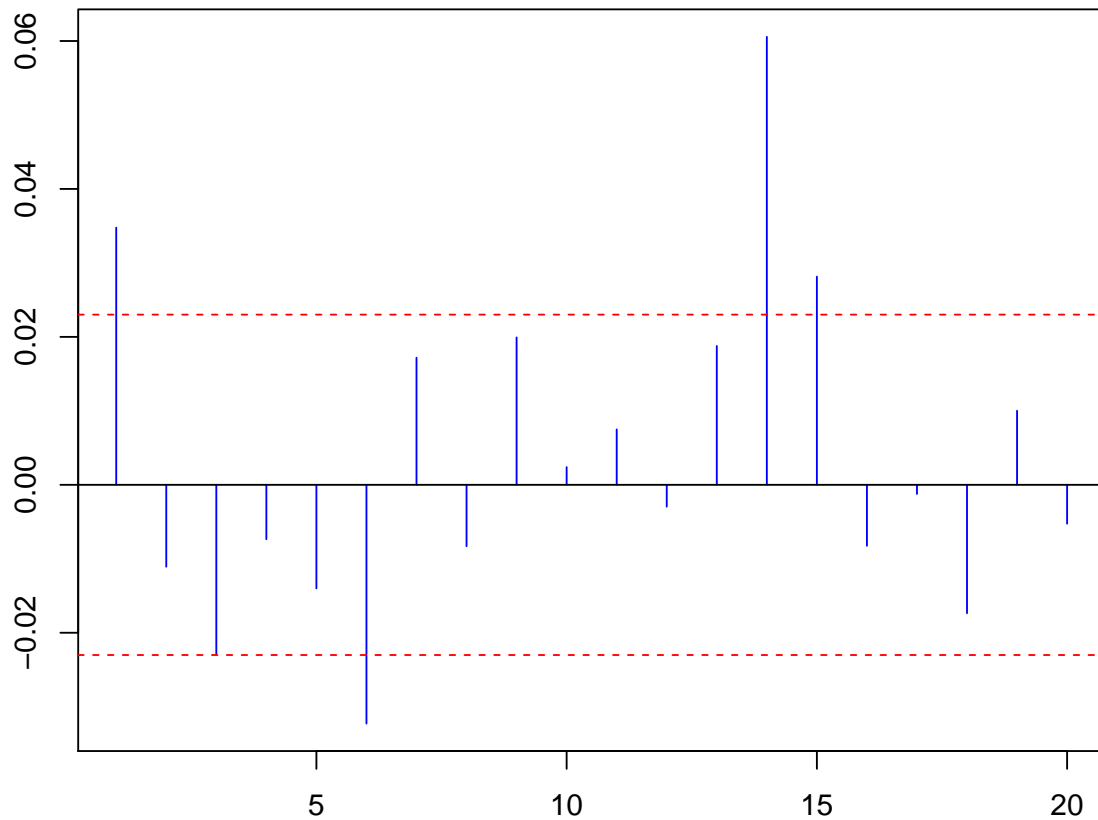
```
acf(coredata(Brent.return), main = "Brent Autocorrelogram",  
    lag.max = 20, ylab = "", xlab = "",  
    col = "blue",  
    ci.col = "red")
```



```
pacf(coredata(Brent.return),  
    main = "Brent Partial Autocorrelogram",  
    lag.max = 20,  
    ylab = "",  
    xlab = "",  
    col = "blue",
```

```
ci.col = "red"  
)
```

Brent Partial Autocorrelogram



On average there are 5 days in the trading week and 20 in the trading month.

Some further thoughts:

- There seems to be positive weekly and negative monthly cycles.
- On a weekly basis negative rates (5 trading days ago) are followed by negative rates (today) and vice-versa with positive rates.
- On a monthly basis negative rates (20 days ago) are followed by positive rates (today).
- There is memory in the markets: positive correlation at least weekly up to a month ago reinforces the strong and frequently occurring negative rates (negative skew and leptokurtotic a.k.a heavy tails)
- Run the PACF for 60 days to see a 40-day negative correlation as well.

2.2.2 Now for something really interesting..again

Let's look just at the size of the Brent returns. The absolute value of the returns (think of oil and countries entering and leaving the EU!) can signal contagion, herd mentality, and simply very large margin calls (and the collateral to back it all up!) Let's run this code:

```
Brent.return.abs <- abs(Brent.return)
## Trading position size matters
Brent.return.tail <- tail(Brent.return.abs[order(Brent.return.abs)], 100)[1]
## Take just the first 100
## observations and pick the first

index <- which(Brent.return.abs > Brent.return.tail,
              arr.ind = TRUE)
## Build an index of those sizes that
## exceed the heavy tail threshold

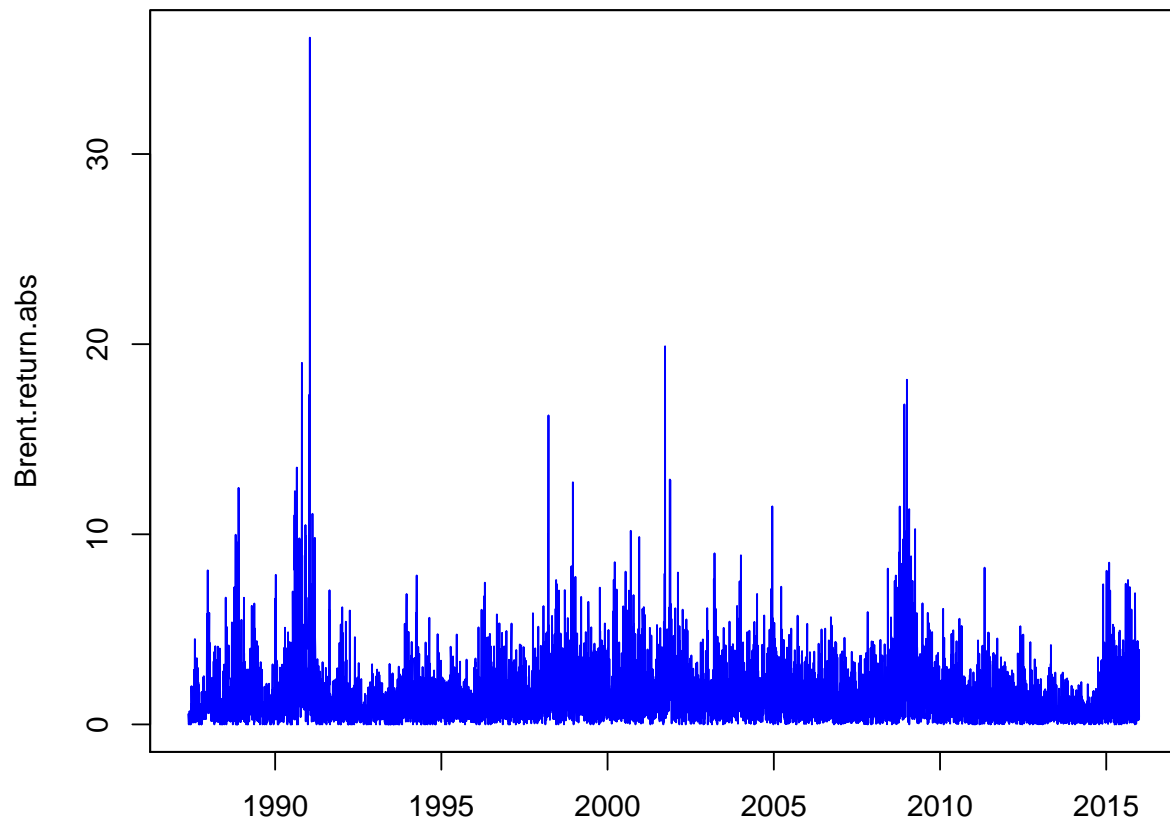
Brent.return.abs.tail <- timeSeries(rep(0, length(Brent.return)),
                                   charvec = time(Brent.return))
# just a lot of zeros we will fill up
# next

Brent.return.abs.tail[index, 1] <- Brent.return.abs[index]
```

What did we do? Let's run some plots next.

```
plot(Brent.return.abs, xlab = "", main = "Brent Daily Return Sizes",
     col = "blue")
```

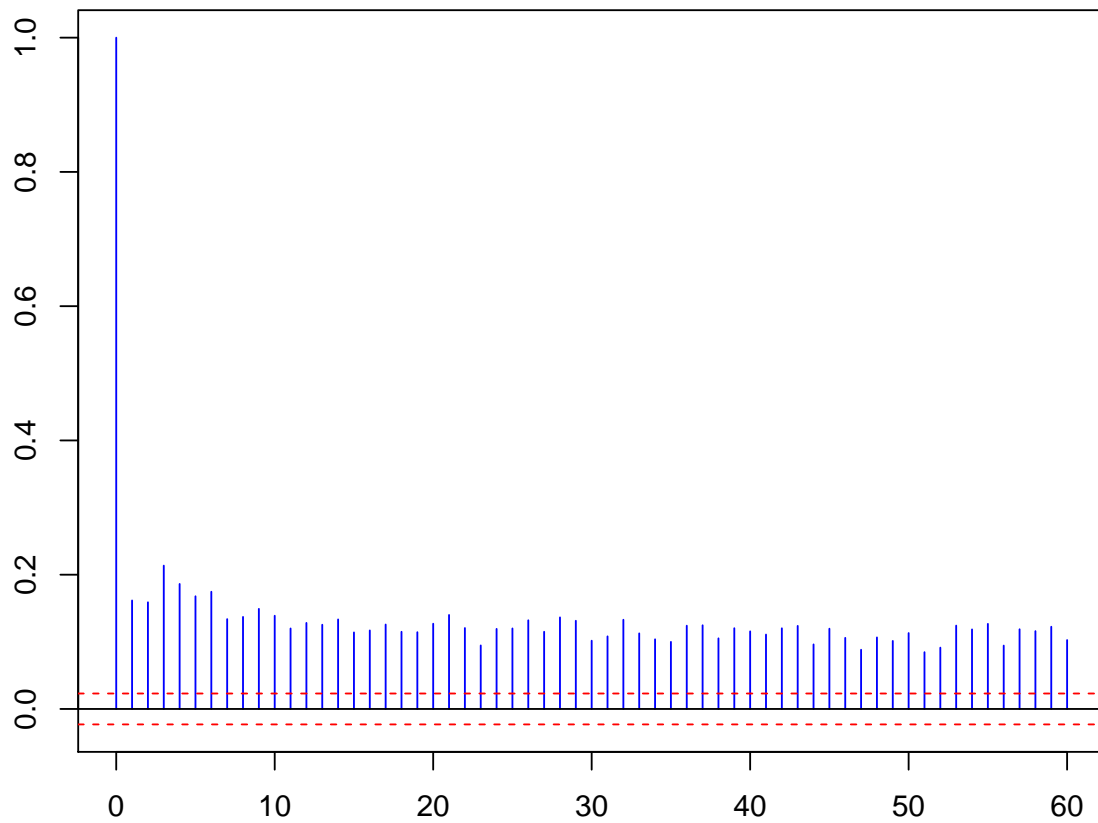
Brent Daily Return Sizes



We see lots of return volatility - just in the pure size along. These are correlated with financial innovations from the 80s and 90s as well as Gulf 1, Gulf 2, Great Recession, and its 9/11 antecedents.

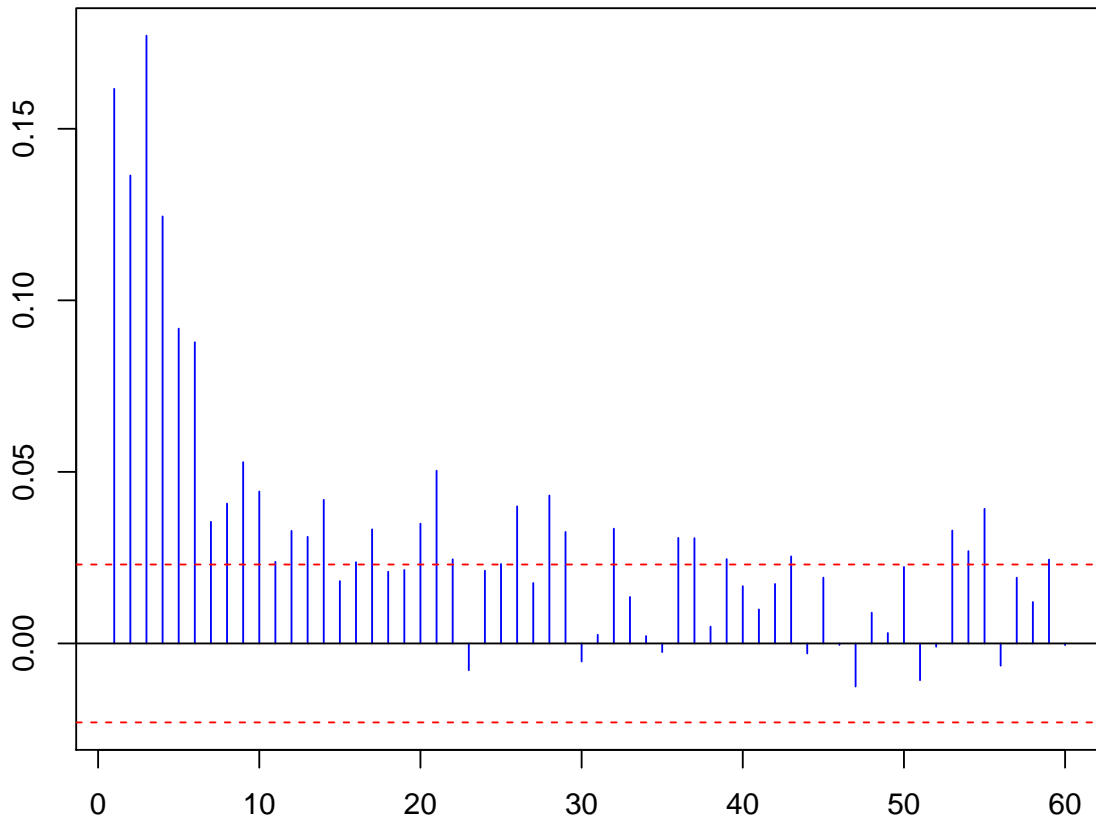
```
acf(coredata(Brent.return.abs), main = "Brent Autocorrelogram",  
    lag.max = 60, ylab = "", xlab = "",  
    col = "blue", ci.col = "red")
```

Brent Autocorrelogram



```
pacf(coredata(Brent.return.abs), main = "Brent Partial Autocorrelogram", lag.max = 60, ylab = "", xlab = "Lag",  
     col = "blue", ci.col = "red")
```

Brent Partial Autocorrelogram



There is *Volatility Clustering galore*. Strong persistent lags of absolute movement in returns evidenced by the ACF plot. There is evidence of dampening with after shocks past trading 10 days ago. Monthly volatility affects today's performance.

Some of this volatility arises from the way Brent is traded. It is lifted through well-heads in the North Sea. It then is scheduled for loading onto ships and loads are then bid, along with routes to destination. It takes about five days to load crude and another five to unload. At each partial loading and unloading, the crude is re-priced. Then there is the voyage lag itself, where paper claims to wet crude create further pricing and volatility.

Next we explore the relationship among financial variables.

2.3 Gettings Caught in the Cross-Current

Now our job is to ask the really important questions around connectivity. Suppose we are banking our investment in certain sectors of an economy, with its GDP, financial capability, employment, exports and imports, and so on.

- How will we decide to contract for goods and services, segment vendors, segment customers, based on these interactions?

- How do we construct our portfolio of business opportunities?
- How do we identify insurgent and relational risks and build a playbook to manage these?
- How will changes in one sector's factors (say, finance, political will) affect factors in another?

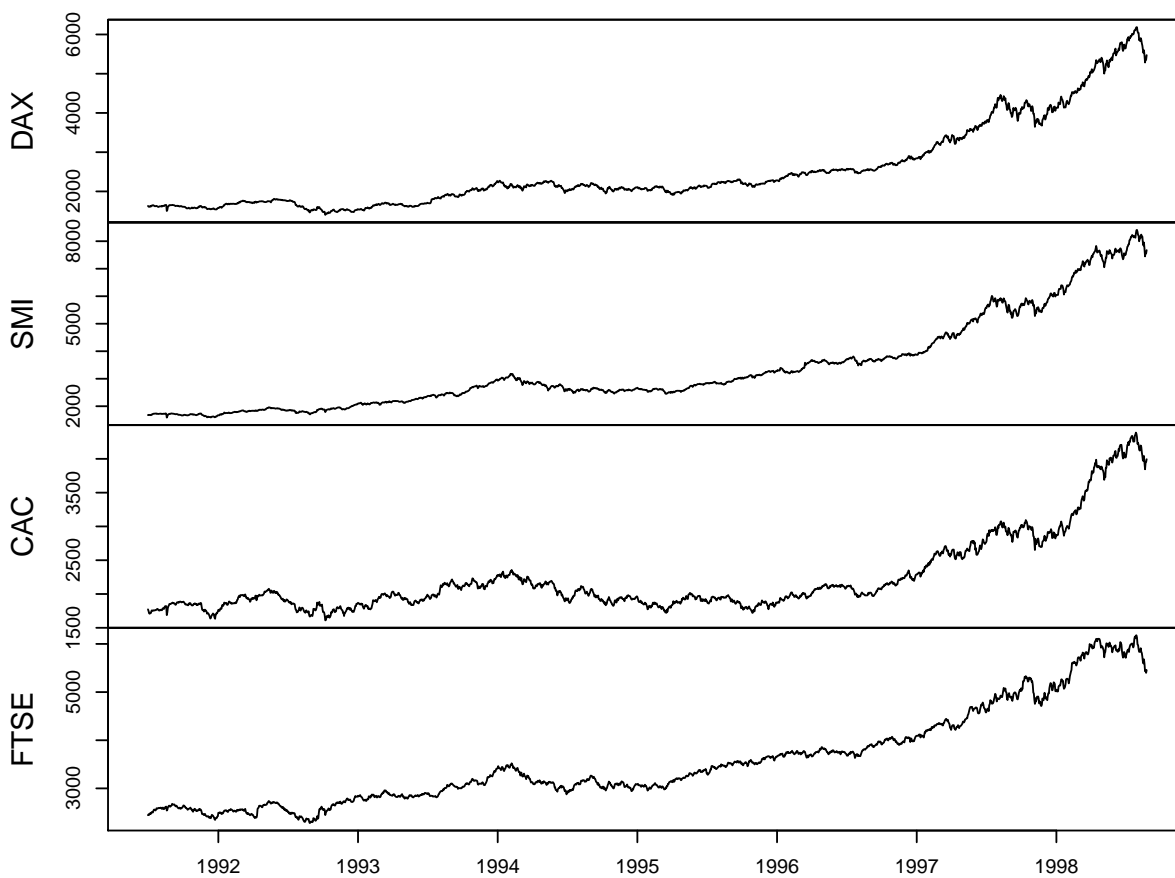
We will not stretch our univariate analysis a bit and look at *cross-correlations* to help us get the ground truth around these relationships, and *begin* to answer some of these business questions in a more specific context.

Let's load the `zoo` and `qrmdata` libraries first and look at the `EuroStoxx50` data set. Here we can imagine we are rebuilding our brand and footprint in the European Union and United Kingdom. Our customers might be the companies based in these countries as our target market.

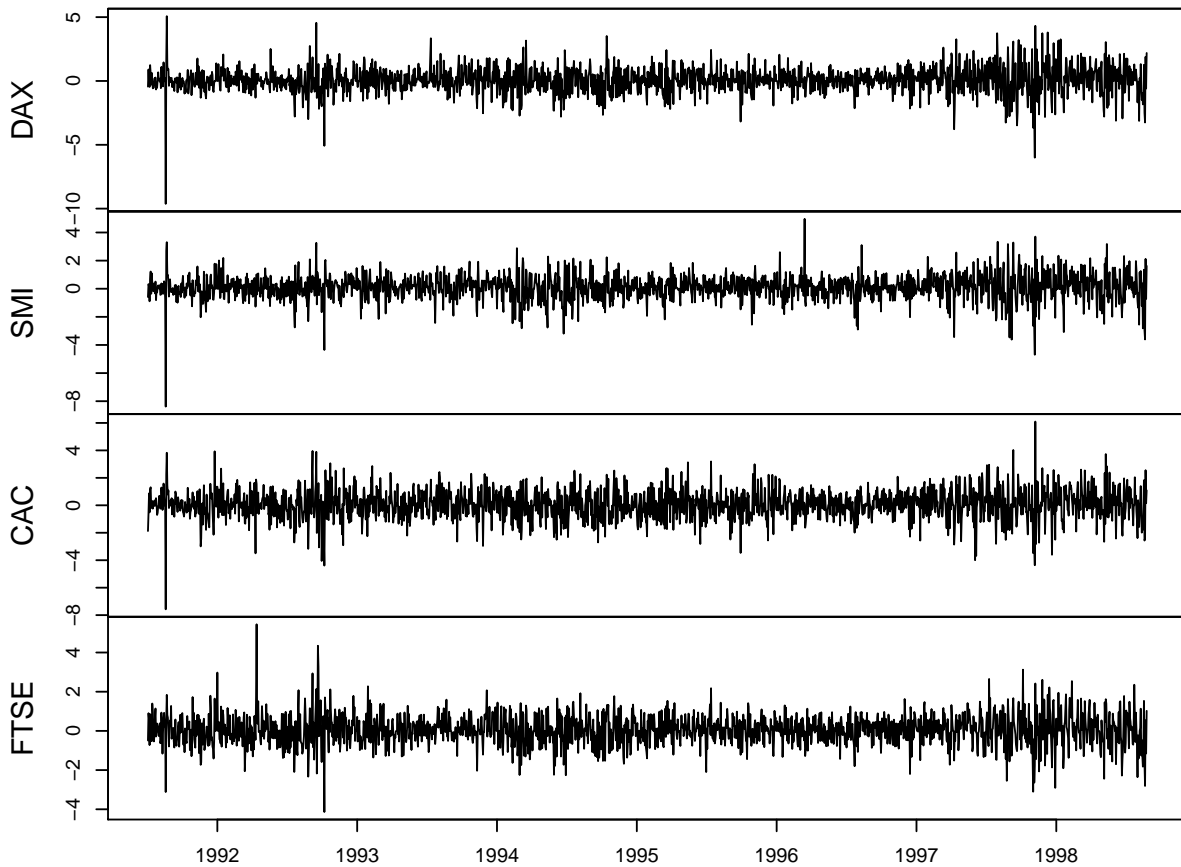
- The data: 4 stock exchange indices across Europe (and the United Kingdom)
- This will allow us to profile the forward capabilities of these companies across their economies.
- Again we will look at returns data using the `diff(log(data))[-1]`

```
library(zoo)
library(qrmdata)
library(xts)
data("EuStockMarkets")
EuStockMarkets.price <- as.zoo(EuStockMarkets)
EuStockMarkets.return <- diff(log(EuStockMarkets.price))[-1] * 100

plot(EuStockMarkets.price, xlab = "", main = "")
```



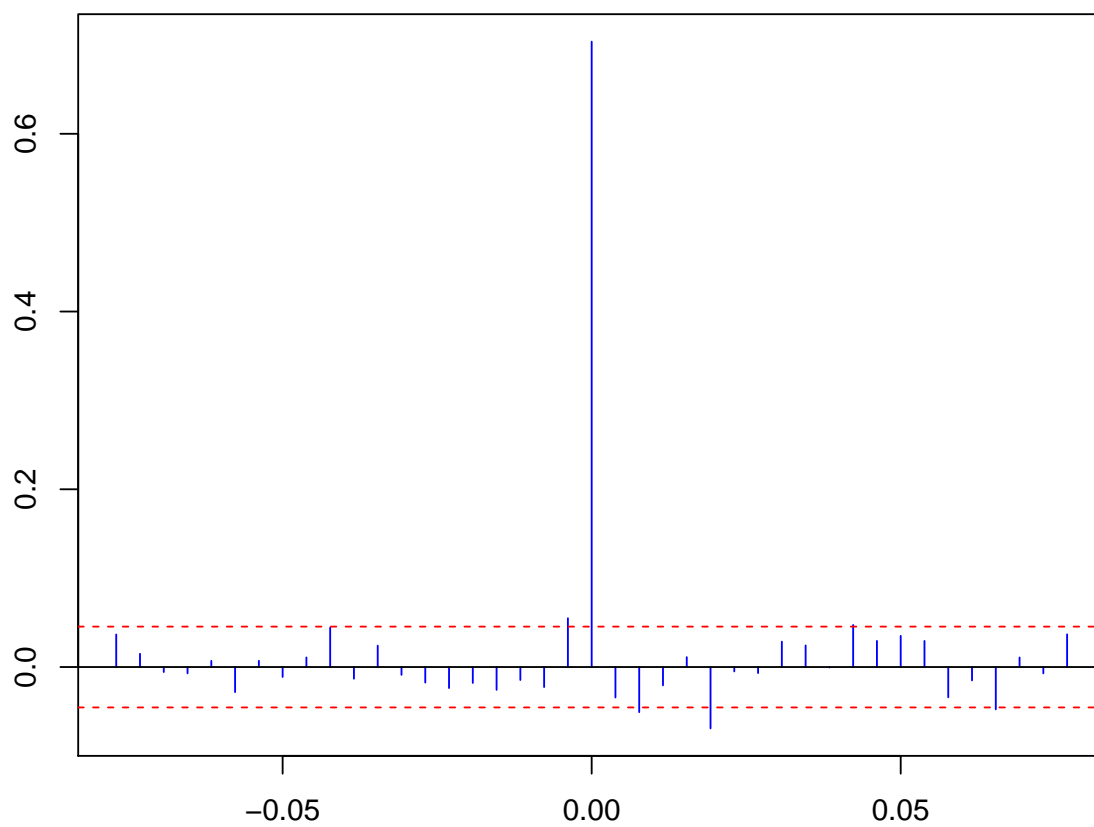
```
plot(EuStockMarkets.return, xlab = " ", main = " ")
```

Let's then look at cross-correlation among one pair of these indices to see how they are related across time (lags) for returns and the absolute value of returns. The function `ccf` will aid us tremendously.

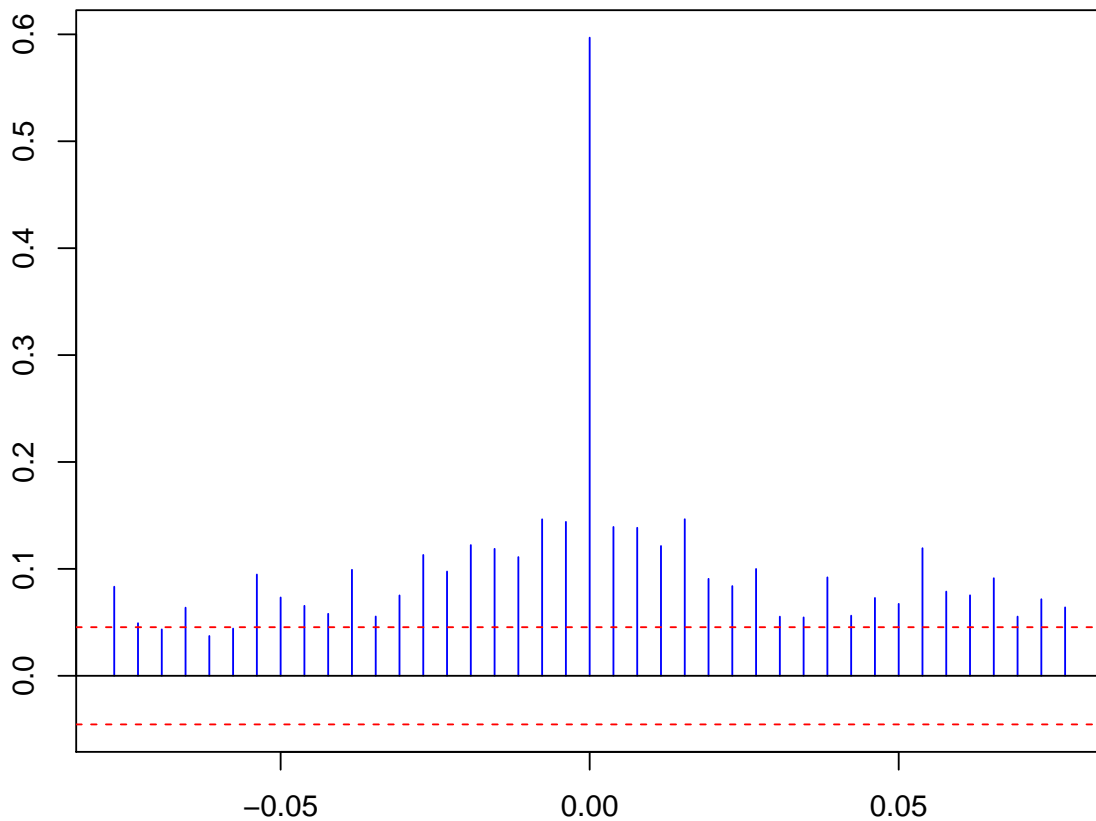
```
ccf(EuStockMarkets.return[,1], EuStockMarkets.return[,2], main = "Returns DAX vs. CAC", lag.max = 20, ylab =
```

Returns DAX vs. CAC



```
ccf(abs(EuStockMarkets.return[,1]),abs(EuStockMarkets.return[,2]),main = "Absolute Returns DAX vs. CAC",)
```

Absolute Returns DAX vs. CAC



We see some small raw correlations across time with raw returns. More revealing, we see volatility of correlation clustering using return sizes. We can conduct one more experiment: a rolling correlation using this function:

```
corr.rolling <-function(x) {  
  dim <-ncol(x)  
  corr.r <-cor(x)[lower.tri(diag(dim),  
                           diag =FALSE)]  
  return(corr.r)  
}
```

We then embed our rolling correlation function, `corr.rolling` into the function `rollapply`. The question we need to answer is: What is the history of correlations, and from the history, the pattern of correlation in the UK and EU stock markets? If there is a “history” with a “pattern”, then we have to manage the risk that conducting business in one country will definitely affect business in another. The implication is that bad things will be followed by more bad things more often than good things. The implication compounds a similar implication across markets.

```
corr.returns <- rollapply(EuStockMarkets.return,  
                          width = 250,  
                          corr.rolling,  
                          align = "right",
```

```

                                by.column = FALSE)
colnames(corr.returns) <- c("DAX & CAC", "DAX & SMI", "DAX & FTSE"
                             , "CAC & SMI", "CAC & FTSE", "SMI & FTSE")

plot(corr.returns, xlab = "", main = "")

```

