Week 3 - Macrofinancial Data Analysis

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 and 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

Previously...

- We 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?

Try this. . .

1. Identify a decision at work (e.g., investment in a new machine, financing a building, acquisition of customers, hiring talent, locating manufacturing).
2. For this decision list three business questions you need to inform the decision you chose.
3. Now think about the data you need to answer one of those questions. Choose from this set:

- Macroeconomic data: GDP, inflation, wages, population
- Financial data: stock market prices, bond prices, exchange rates, commodity prices

Thinking...

Results

1. Decision is **supply a new market segment**

- Product: voltage devices with supporting software
- Geography: Spain
- Customers: major buyers at Iberdrola, Repsol, and Endesa

2. Three business questions:

- How would the performance of these companies affect the size and timing of orders?
- How would the value of their products affect the value of our business with these companies?
- We are a US functional currency firm (see FAS 52), so how would we manage the repatriation of accounts receivable from Spain?

3. Data and analysis to inform the decision

- Customer stock prices: volatility and correlation
- Oil prices: volatility
- USD/EUR exchange rates: volatility
- All together: correlations among these indicators

Stylized facts

. . . of the market

Learned the hard Way: not independent, volatile volatility, extreme

- Financial stock, bond, commodity. . . you name it. . . 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 are likely to happen with other extreme events.
- Negative returns are more likely than positive returns (left skew).

Examples from the 70's, 80's, and 90's have lots of global events going on. Load up some computational help and some data from Brent, format dates, and create a time series object (package zoo' will be needed by packagesfBasicsandevir'):

```
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
```

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

```r
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
```

Try this. . .

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.

```
plot(Brent.return, title = FALSE, xlab = "",
    main = "Brent Daily % Change", col = "blue")
```

1. Run the plot and comment.

Now run this:

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

2. Comment on the likelihood of positive versus negative returns. You might want to look up skewness and kurtosis definitions and ranges.

Now to look at persistence:

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

Confidence intervals are the red dashed lines. ACF at lag 6 means the correlation of current Brent returns with returns 6 trading days ago, including any correlations from trading day 1 through 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).

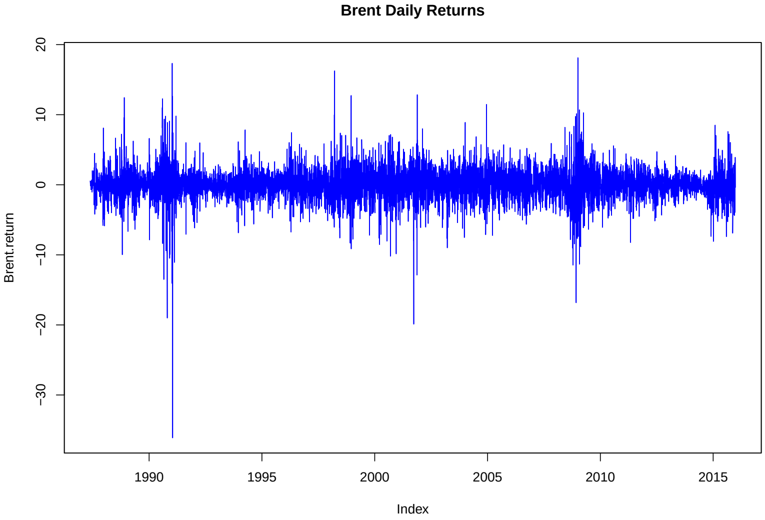
3. How many trading days in a typical week or in a month? Comment on the spikes (blue lines that grow over or under the red dashed lines).

Thinking. . .

We have results

1. The plot

```
plot(Brent.return, main = "Brent Daily Returns",
    col = "blue")
```


Brent Daily Returns
Brent.return
20
10
0
−10
−20
−30
1990
1995
2000
2005
2010
2015
Index

This time series plot shows lots of return clustering and spikes, especially negative ones.

Performing some "eyeball econometrics" these clusters seem to occur around

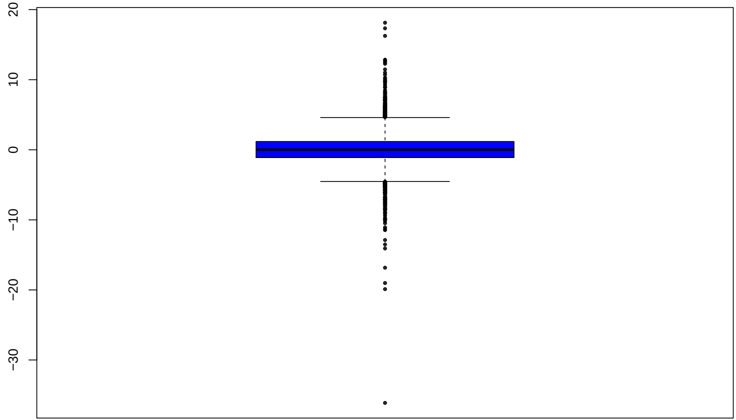
- The oil embargo of the '70s
- The height of the new interest rate regime of Paul Volcker at the Fed
- "Black Monday" stock market crash in 1987
- Gulf I
- Barings and other derivatives business collapses in the '90s

2. How thick is that tail?

Here is a first look:

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

Brent Daily Returns



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

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

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

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

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

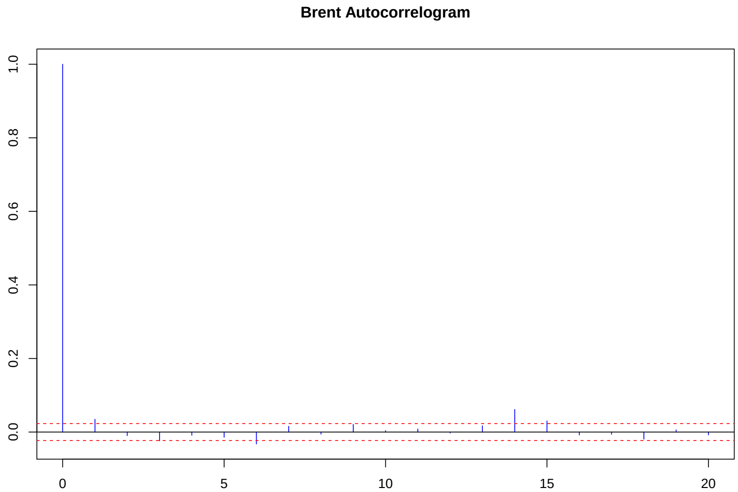
- A negative skew means there are more observations less than the median than greater.
- This high a kurtosis means a pretty heavy tail, especially in negative returns. That means they have happened more often than positive returns.
- A preponderance of negative returns frequently happening spells trouble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

Implications

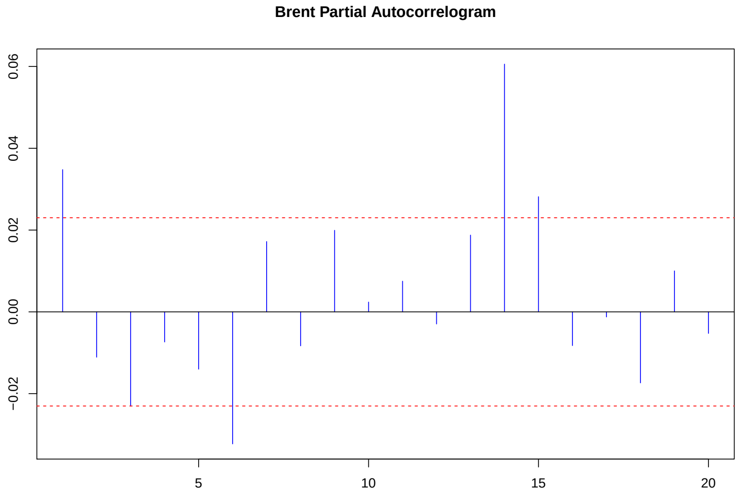
- Budget for the body of the distribution from the mean and out to positive levels.
- Build a comprehensive playbook for the strong possibility that bad tail events frequently happen and might happen again (and why shouldn't they?).

3. Now for something really interesting

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



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



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

Some 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.

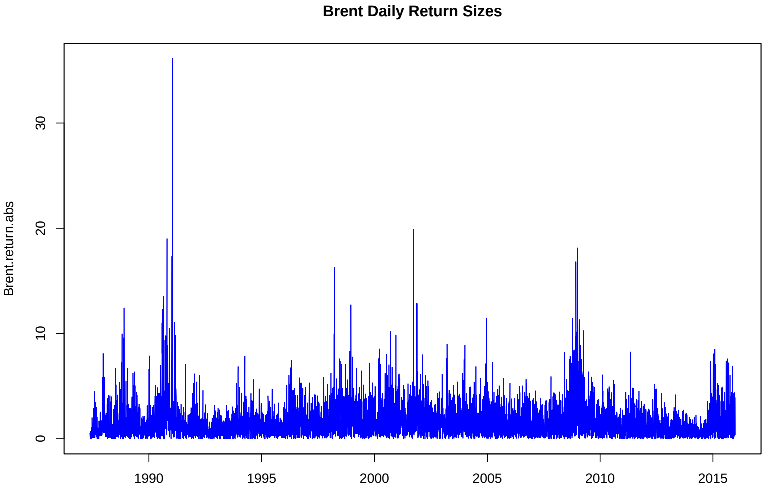
Now for somthing really interesting. . . again

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 of the 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]
# A Phew! is in order
```

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

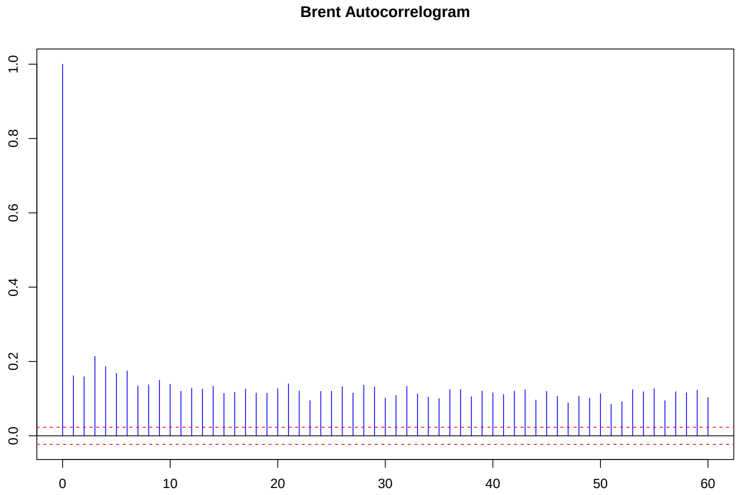
```r
plot(Brent.return.abs, xlab = "", main = "Brent Daily Return Sizes",
    col = "blue")
```



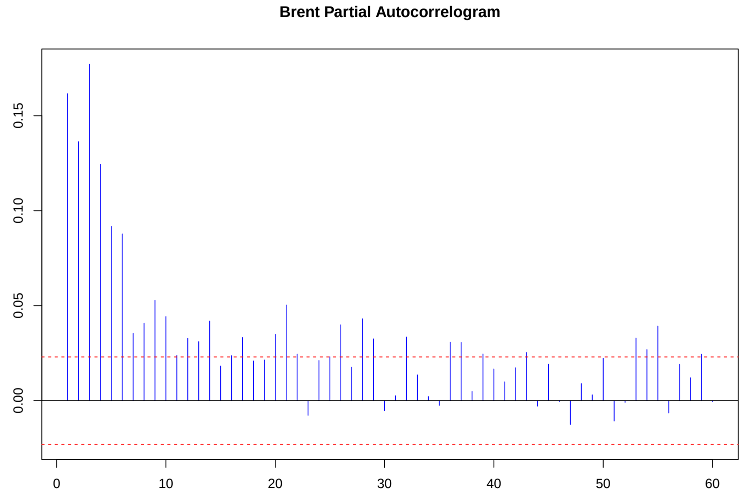
Lots of return volatility – just in pure size

- Same event
- Correlated with financial innovations from the '80s and '90s
- Gulf 1, Gulf 2, Great Recession, and its 9/11 antecedents

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



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



Volatility Clustering galore

- Getting strong persistent lags of absolute movements in returns
- Dampening with after shocks past trading 10 days 10 ago: monthly volatility affects today's performance

Next: What are the relationships among financial variables?

Getting caught in the cross-current

Now our job is to ask the really important questions:

Suppose I am banking my investment in certain sectors of an economy, with its GDP, financial capability, employment, exports and imports, etc.,

then . . .

- How will I decide to contract for goods and services, segment vendors, segment customers, based on these interactions?
- How do I construct my portfolio of business opportunities?
- How do I 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 now stretch out 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.

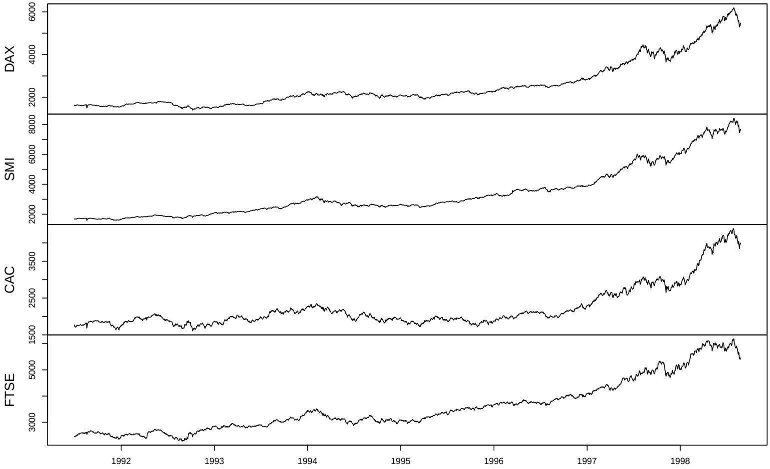
Our customers might be the companies based in these countries as our target market.

- 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.

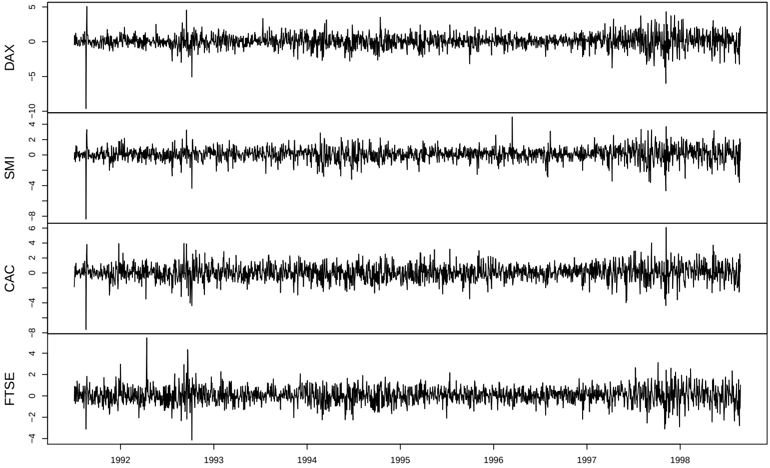
```r
require(zoo)
require(qrmdata)
require(xts)
data("EuStockMarkets")
EuStockMarkets.price <- as.zoo(EuStockMarkets)
EuStockMarkets.return <- diff(log(EuStockMarkets.price))[-1] *
    100
```


Plot the levels and returns.

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


DAX
SMI
CAC
FTSE
6000
4000
2000
8000
6000
4000
2000
3500
2500
1500
5000
3000
1992
1993
1994
1995
1996
1997
1998

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

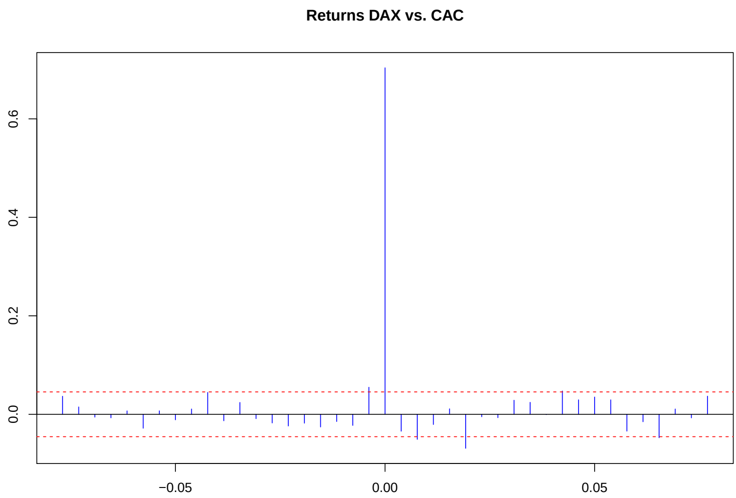

DAX
SMI
CAC
FTSE
1992
1993
1994
1995
1996
1997
1998

We see much the same thing as Brent oil with volatility clustering and heavily weighted tails.

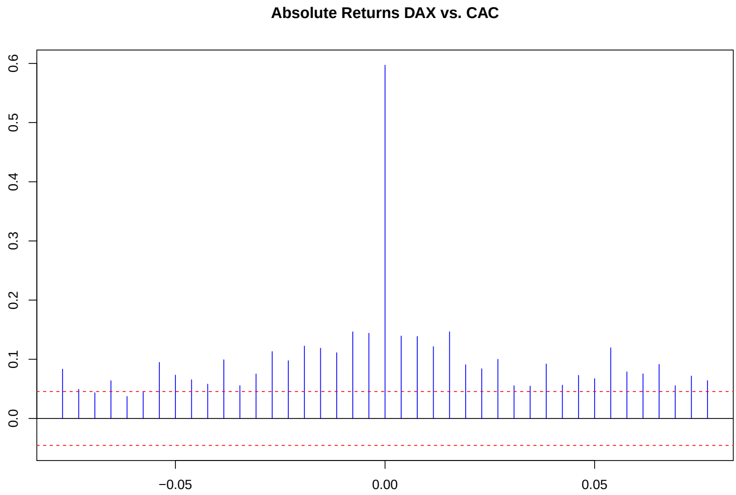
Let's look at cross-correlations among one pair of these indices to see how they are related across time (lags) for returns and the absolute value of returns

```
ccf(EuStockMarkets.return[, 1], EuStockMarkets.return[,
    2], main = "Returns DAX vs. CAC",
    lag.max = 20, ylab = "", xlab = "",
    col = "blue", ci.col = "red")
```


Returns DAX vs. CAC




```r
ccf(abs(EuStockMarkets.return[, 1]),
    abs(EuStockMarkets.return[, 2]),
    main = "Absolute Returns DAX vs. CAC",
    lag.max = 20, ylab = "", xlab = "",
    col = "blue", ci.col = "red")
```

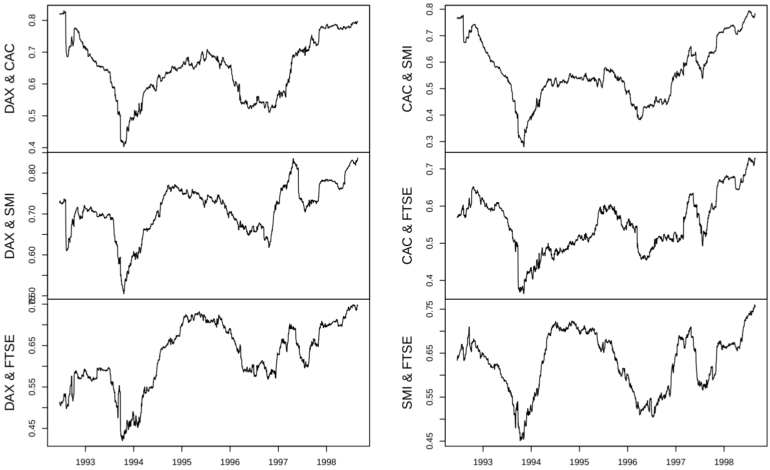

Absolute Returns DAX vs. CAC
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−0.05
0.00
0.05

We see some small raw correlations across time with raw returns. More revealing, we see volatility of correlation clustering using return sizes. 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)
}
```

Embed our rolling correlation function, `corr.rolling`, into the function `rollapply` (look this one up!). The question we need to answer is: What is the history of correlations in the UK and EU stock markets? If this is a "history," then we have to manage the risk that conducting business in one country will definitely affect business in another…and that bad things will be followed by more bad things more often than good things…

```r
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 = "")
```


DAX & CAC
DAX & SMI
DAX & FTSE
CAC & SMI
CAC & FTSE
SMI & FTSE
1993
1994
1995
1996
1997
1998

Again look at the volatility clustering the absolute sizes of returns. Economic performance is certainly subject here to the same dynamics we saw for a single stock.

Try this one now . . .

Let's redo some of the work we just did using another set of techniques. This time we are using the "Fisher" transformation. Look up Fisher in Wikipedia and in your reference texts.

```
fisher <- function(r) {
    0.5 * log((1 + r)/(1 - r))
}
```

1. What is the stated purpose of the Fisher transformation. How can it possibly help us answer our business questions?

2. For three Spanish companies, Iberdrola, Endesa, and Repsol, replicate the Brent and EU stock market experiments above with absolute sizes and tails. Here we already have “series” covered.

Thinking. . .

Results

1. Fisher transformations

- Stabilizes the variance of a variate
- Pulls some of the shockiness (i.e., outliers and aberrant noise) out
- Helps us see the forest for the trees

2. Replicating the Brent and EU stock market experiments.

Load some packages and get some data using quantmod's getSymbols off the Madrid stock exchange. Then make some returns and merge into a master file.

```
require(xts)
require(qrmdata)
require(quantreg)
require(quantmod)
require(matrixStats)

tickers <- c("ELE.MC", "IBE.MC", "REP.MC")
getSymbols(tickers)
```

```
## [1] "ELE.MC" "IBE.MC" "REP.MC"
```

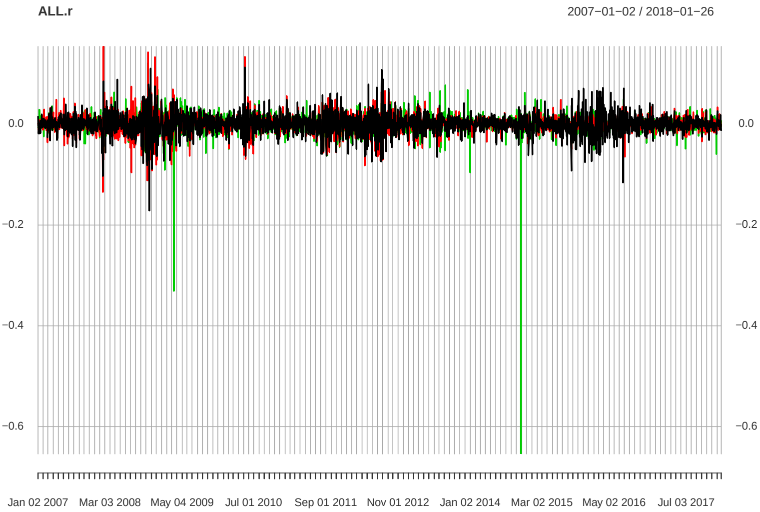
```
REP.r <- na.omit(diff(log(REP.MC[, 4])))[-1])  # clean out missing values
IBE.r <- na.omit(diff(log(IBE.MC[, 4])))[-1])
ELE.r <- na.omit(diff(log(ELE.MC[, 4])))[-1])

ALL.r <- merge(REP = REP.r, IBE = IBE.r,
    ELE = ELE.r, all = FALSE)
```

Next plot the returns and their absolute values, acf and pacf, all like we did in Brent.

Notice

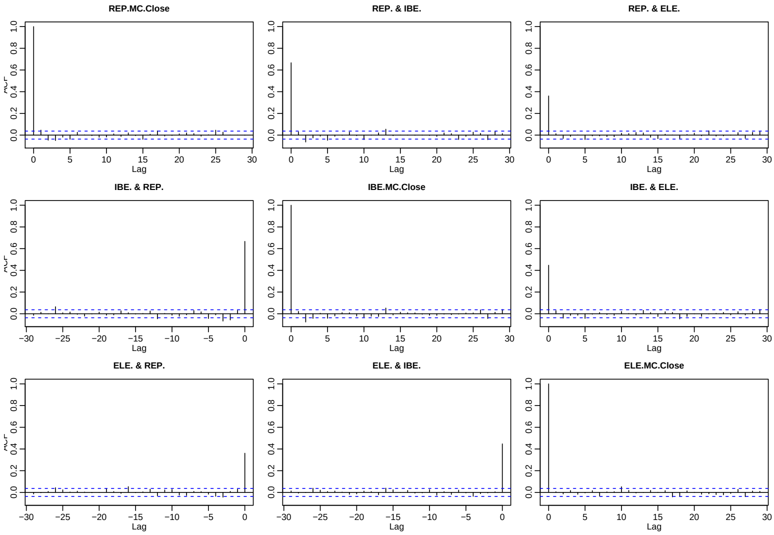
1. The persistence of returns
2. The importance of return size
3. Clustering of volatility


```r
plot(ALL.r)
```

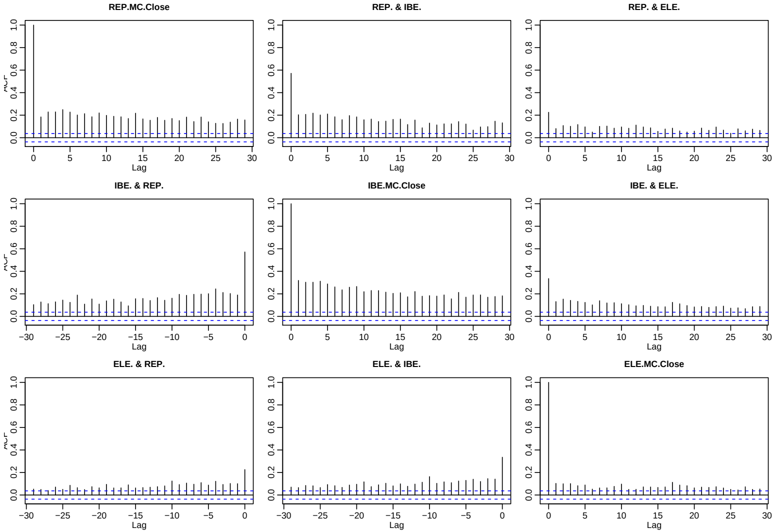


ALL.r
2007−01−02 / 2018−01−26
0.0
−0.2
−0.4
−0.6
0.0
−0.2
−0.4
−0.6
Jan 02 2007
Mar 03 2008
May 04 2009
Jul 01 2010
Sep 01 2011
Nov 01 2012
Jan 02 2014
Mar 02 2015
May 02 2016
Jul 03 2017

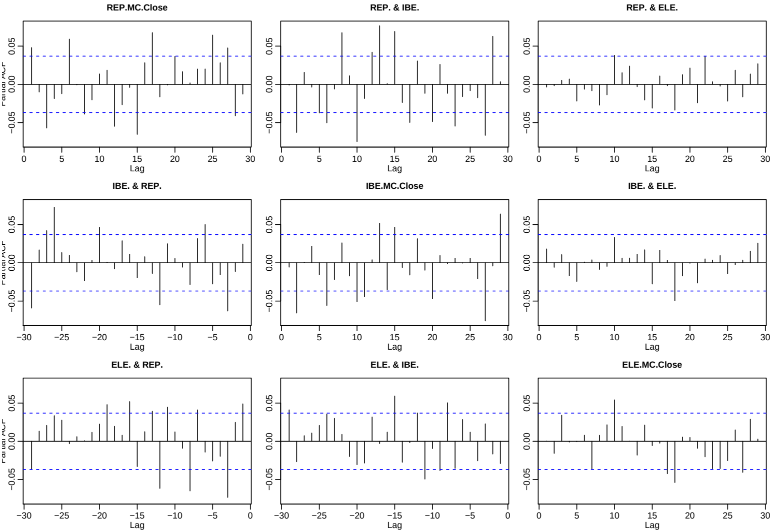
```
par(mfrow = c(2, 1))
acf(ALL.r)
```



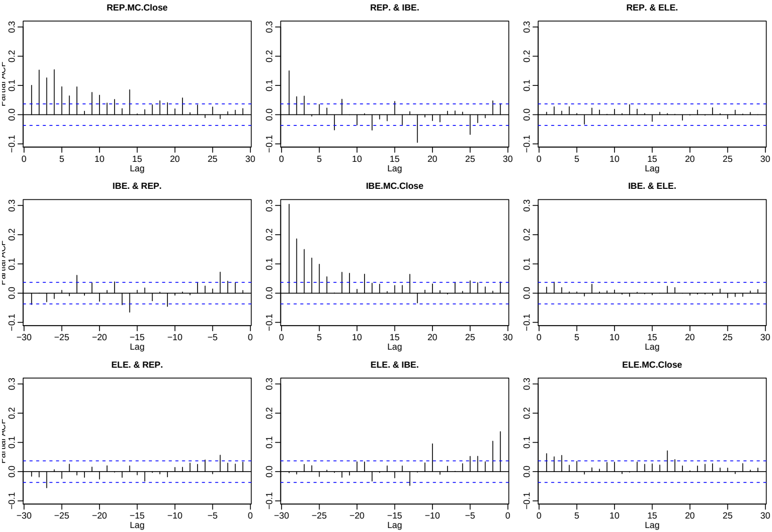
```r
par(mfrow = c(2, 1))
acf(abs(ALL.r))
```



```r
par(mfrow = c(2, 1))
pacf(ALL.r)
```



```r
par(mfrow = c(2, 1))
pacf(abs(ALL.r))
```



Now to examine the correlation structure of markets.

Notice

1. The relationship between correlation and volatility
2. How quantile regression gets us to an understanding of high stress (high and low quantile) episodes


```r
R.corr <- apply.monthly(ALL.r, FUN = cor)
R.vols <- apply.monthly(ALL.r, FUN = colSds)  # from MatrixStats
head(R.corr, 3)
```

```
##            [,1]      [,2]        [,3]      [,4] [,5]       [,6]
## 2007-01-31    1 0.3613554 -0.27540757 0.3613554    1 0.10413800
## 2007-02-28    1 0.5661814 -0.09855544 0.5661814    1 0.10760477
## 2007-03-30    1 0.4500982 -0.08874664 0.4500982    1 0.08538064
##                  [,7]       [,8] [,9]
## 2007-01-31 -0.27540757 0.10413800    1
## 2007-02-28 -0.09855544 0.10760477    1
## 2007-03-30 -0.08874664 0.08538064    1
```

```r
head(R.vols, 3)
```

```
##            REP.MC.Close IBE.MC.Close ELE.MC.Close
## 2007-01-31  0.009787963  0.007892759  0.009777426
## 2007-02-28  0.009181099  0.014571945  0.007674848
## 2007-03-30  0.015317331  0.012719792  0.010919155
```

```r
R.corr.1 <- matrix(R.corr[1, ], nrow = 3,
    ncol = 3, byrow = FALSE)
rownames(R.corr.1) <- tickers
colnames(R.corr.1) <- tickers
head(R.corr.1)
```

```
##            ELE.MC    IBE.MC     REP.MC
## ELE.MC  1.0000000 0.3613554 -0.2754076
## IBE.MC  0.3613554 1.0000000  0.1041380
## REP.MC -0.2754076 0.1041380  1.0000000
```

```r
R.corr <- R.corr[, c(2, 3, 6)]
colnames(R.corr) <- c("ELE.IBE", "ELE.REP",
    "IBE.REP")
colnames(R.vols) <- c("ELE.vols", "IBE.vols",
    "REP.vols")
head(R.corr, 3)
```

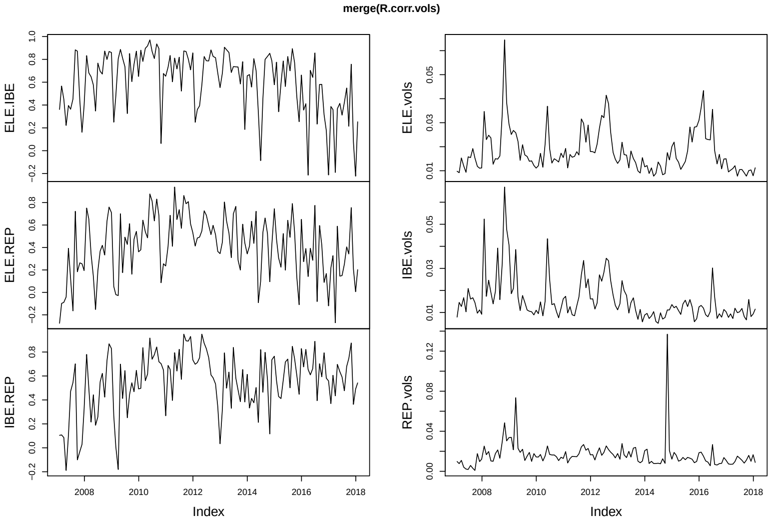
```
##              ELE.IBE     ELE.REP    IBE.REP
## 2007-01-31 0.3613554 -0.27540757 0.10413800
## 2007-02-28 0.5661814 -0.09855544 0.10760477
## 2007-03-30 0.4500982 -0.08874664 0.08538064
```

```r
head(R.vols, 3)
```

```
##               ELE.vols    IBE.vols    REP.vols
## 2007-01-31 0.009787963 0.007892759 0.009777426
## 2007-02-28 0.009181099 0.014571945 0.007674848
## 2007-03-30 0.015317331 0.012719792 0.010919155
```

```r
R.corr.vols <- na.omit(merge(R.corr,
    R.vols))
```

```r
plot.zoo(merge(R.corr.vols))
```



```r
ELE.vols <- as.numeric(R.corr.vols[,
    "ELE.vols"])
IBE.vols <- as.numeric(R.vols[, "IBE.vols"])
REP.vols <- as.numeric(R.vols[, "REP.vols"])
length(ELE.vols)
```

```
## [1] 133
```

```r
# fisher <- function(r) {0.5 * log((1
# + r)/(1 - r))}
rho_vol <- matrix(as.numeric(R.corr.vols[,
    1:3]), nrow = length(ELE.vols), ncol = 3,
    byrow = FALSE)
```

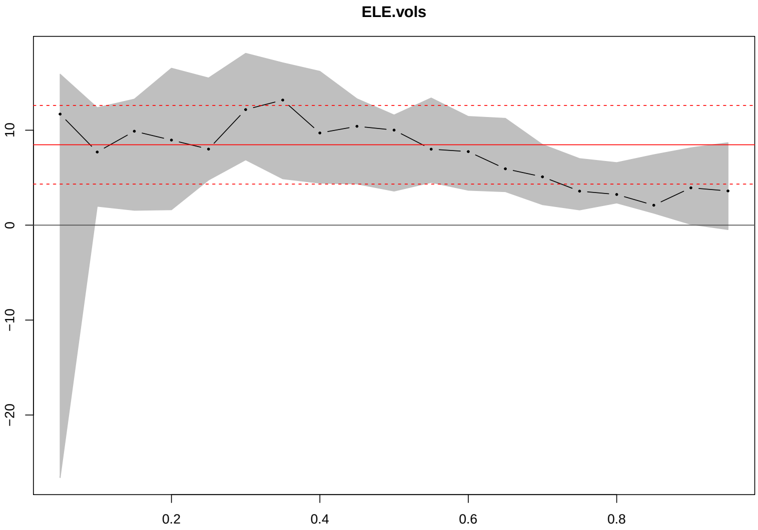
Here is the quantile regression part of the package.

Notice

1. We set `taus` as the quantiles of interest.
2. We run the quantile regression using the `quantreg` package and a call to the `rq` function.
3. We can overlay the quantile regression results onto the standard linear model regression.
4. We can sensitize our analysis with the range of upper and lower bounds on the parameter estimates of the relationship between correlation and volatility.

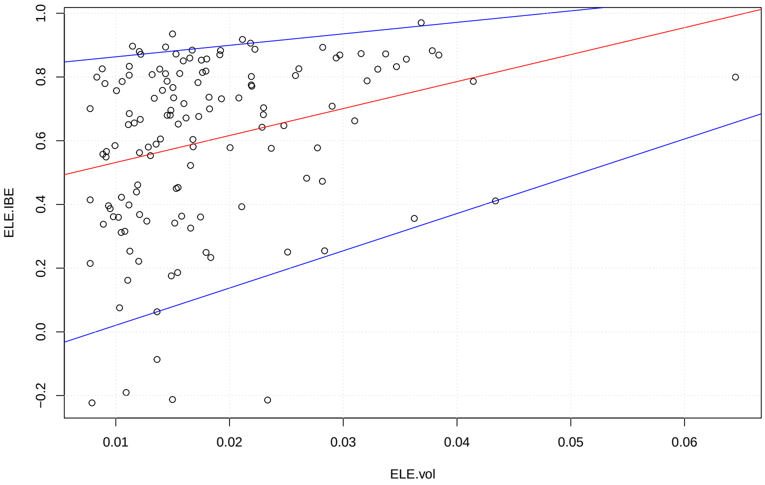
```
taus <- seq(0.05, 0.95, 0.05)
fit.rq.ELE.IBE <- rq(rho_vol[, 1] ~ ELE.vols,
    tau = taus)
fit.lm.ELE.IBE <- lm(rho_vol[, 1] ~ ELE.vols)
```

```r
plot(summary(fit.rq.ELE.IBE), parm = "ELE.vols")
```



Here we build the estimations and plot the upper and lower bounds.

```
taus1 <- c(0.05, 0.95)  # fit the confidence interval (CI)
plot(ELE.vols, rho_vol[, 1], xlab = "ELE.vol",
    ylab = "ELE.IBE")
abline(fit.lm.ELE.IBE, col = "red")
for (i in 1:length(taus1)) {
    # these lines will be the CI
    abline(rq(rho_vol[, 1] ~ ELE.vols,
        tau = taus1[i]), col = "blue")
}
grid()
```


ELE.IBE
ELE.vol
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
0.01
0.02
0.03
0.04
0.05
0.06

Bounding our enthusiasm

1. Quantile regression helps us to see the upper and lower bounds.
2. Relationships between high-stress periods and correlation are abundant.
3. These markets simply reflect normal buying behaviors across many types of exchanges: buying food at Safeway or Whole Foods, buying collateral to insure a project, selling off illiquid assets.

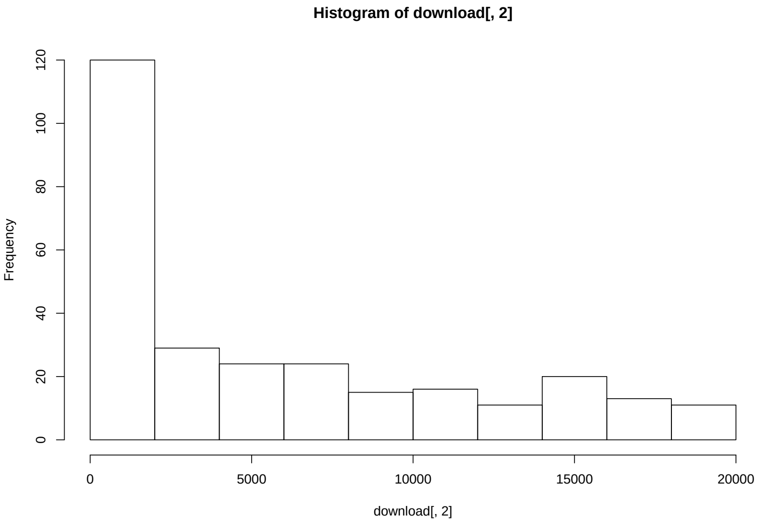
Time is on our side...

Let's start with some US Gross National Product (GNP) data from the St. Louis Fed's open data website ("FRED"). We build a `URL` using the `paste()` function.

```
name <- "GNP"
URL <- paste("http://research.stlouisfed.org/fred2/series/",
    name, "/", "downloaddata/", name,
    ".csv", sep = "")
download <- read.csv(URL)
```

Look at the data:

```r
hist(download[, 2])
```




```r
summary(download[, 2])
```

```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    244.1   700.5  3372.2  5780.2 10011.2 19728.9
```


Create a raw time series object (rownames are dates. . .), select some data, and calculate growth rates

```
GNP <- ts(download[1:84, 2]/1000, start = c(1995,
    1), freq = 4)
GNP.rate <- na.omit(100 * diff(log(GNP)))
```

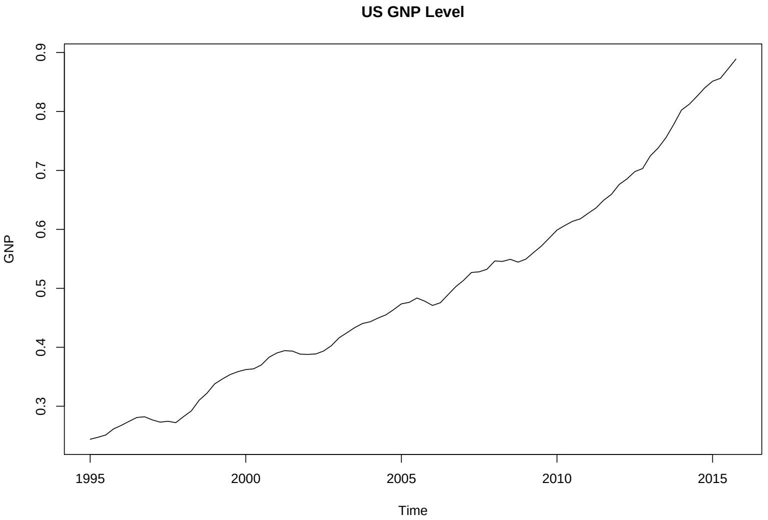
Try this . . .

1. Plot the GNP level and rate.
2. Comment on the patterns.

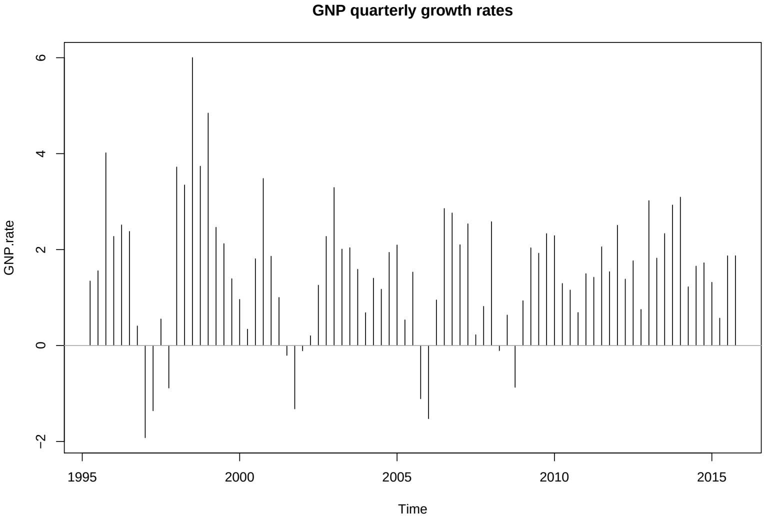
Thinking. . .

Results

```
plot(GNP, type = "l", main = "US GNP Level")
```



```r
plot(GNP.rate, type = "h", main = "GNP quarterly growth rates")
abline(h = 0, col = "darkgray")
```



What we call “nonstationary”

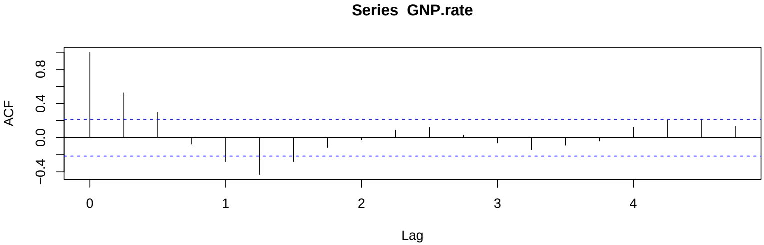
1. The probability distribution (think `hist()`) would seem to change over time.
2. This means that the standard deviation and mean changes as well.
3. Lots of trend in the level and simply dampened sinusoidal in the rate.

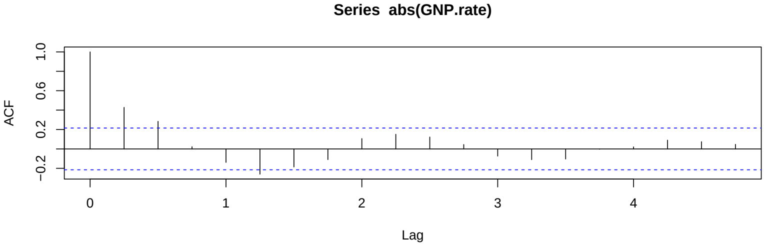
Can we forecast GNP?

Forecasting GNP

As always let's look at ACF and PACF:

```
par(mfrow = c(2, 1))  #stacked up and down
acf(GNP.rate)
acf(abs(GNP.rate))
```


Series GNP.rate
ACF
-0.4
0.0
0.4
0.8
0
1
2
3
4
Lag


Series abs(GNP.rate)
ACF
-0.2
0.2
0.6
1.0
0
1
2
3
4
Lag

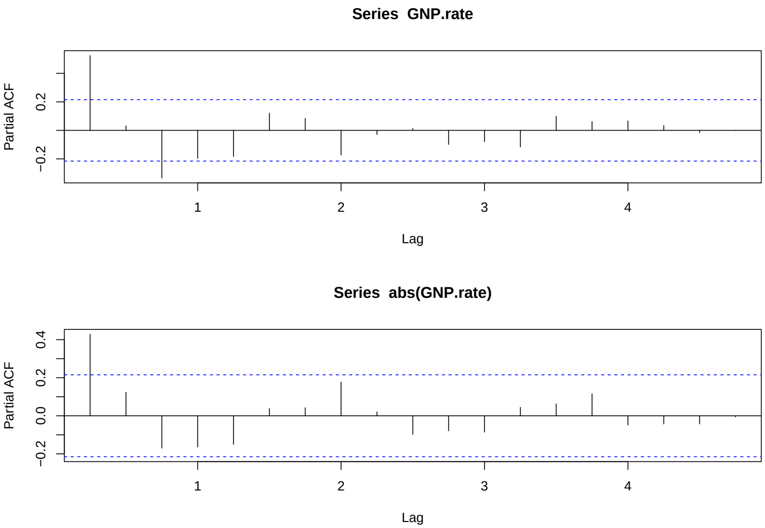
Try this...

```r
par(mfrow = c(2, 1))
pacf(GNP.rate)
pacf(abs(GNP.rate))
par(mfrow = c(1, 1))  #default setting
```

What do you think is going on?

Thinking...

Result


Series GNP.rate
Partial ACF
0.2
−0.2
1
2
3
4
Lag
Series abs(GNP.rate)
Partial ACF
0.4
0.2
0.0
−0.2
1
2
3
4
Lag

What do you think?

- There are several significant autocorrelations within the last 4 quarters.
- Partial autocorrelation also indicates some possible relationship 8 quarters back.

Yet another regression (YAR)...

Let's use `R`'s time series estimation tool `arima`. We think there is a regression that looks like this:

$$x_t = a_0 + a_1 x_{t-1} ... a_p x_{t-p} + b_1 \epsilon_{t-1} + ... + b_q \epsilon_{t-q}$$

where x_t is a first, $d = 1$, differenced level of a variable, here GNP. There are p lags of the rate itself and q lags of residuals. We officially call this an *Autoregressive Integrated Moving Average* process of order (p, d, q), or `ARIMA(p,d,q)` for short.

Estimation is quick and easy.

```
fit.rate <- arima(GNP.rate, order = c(2,
    0, 1))
```

The order is 2 lags of rates, 0 further differencing (already differenced once), and 1 lag of residuals. Let's diagnose the results with `tsdiag()`:

What are the results?

```
fit.rate
```

```
##
## Call:
## arima(x = GNP.rate, order = c(2, 0, 1))
##
## Coefficients:
##           ar1     ar2     ma1  intercept
##       -0.2351  0.4799  0.7113     1.5583
## s.e.   0.2577  0.1318  0.2721     0.2825
##
## sigma^2 estimated as 1.332:  log likelihood = -129.89,  aic = 269.77
```

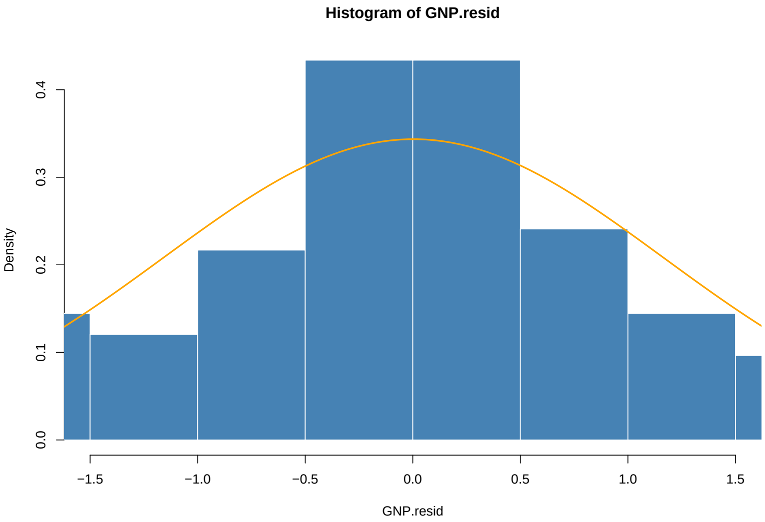
Take out the moving average term and compare:

```
fit.rate.2 <- arima(GNP.rate, order = c(2,
    0, 0))
fit.rate.2
```

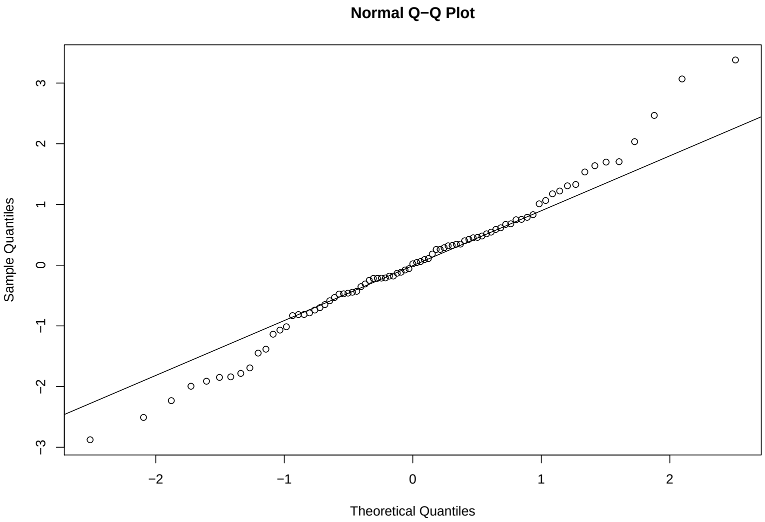
```
##
## Call:
## arima(x = GNP.rate, order = c(2, 0, 0))
##
## Coefficients:
##          ar1     ar2  intercept
##       0.5027  0.0303     1.5588
## s.e.  0.1089  0.1085     0.2715
##
## sigma^2 estimated as 1.373:  log likelihood = -131.08,  aic = 270.17
```



```r
GNP.resid <- resid(fit.rate)
hist(GNP.resid, probability = TRUE, breaks = "FD",
    xlim = c(-1.5, 1.5), col = "steelblue",
    border = "white")
x = seq(-2, 2, length = 100)
lines(x, dnorm(x, mean = mean(GNP.resid),
    sd = sd(GNP.resid)), col = "orange",
    lwd = 2)
```


Histogram of GNP.resid
Density
0.0
0.1
0.2
0.3
0.4
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
GNP.resid

```r
qqnorm(GNP.resid)
qqline(GNP.resid)
```



One way to read the qq-chart is

1. The diagonal line is the normal distribution quantile line.
2. Deviations of actual quantiles from the normal quantile line mean nonnormal.
3. Especially deviations at either (or both) end of the line spell thick tails and lots more "shape" than the normal distribution allows.

Try this out

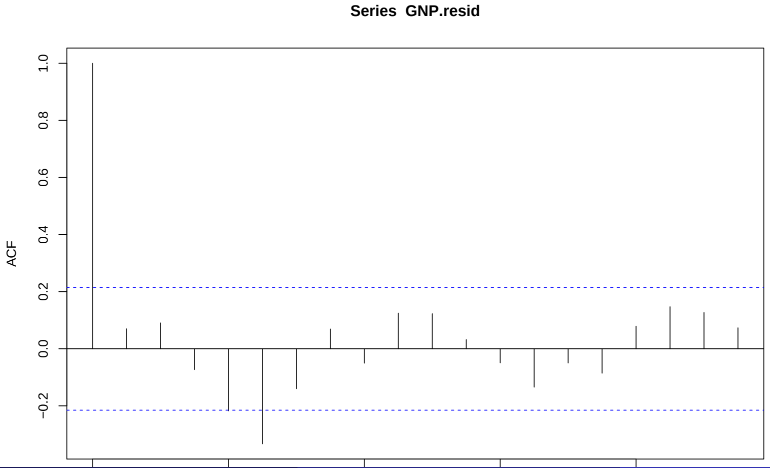
Diagnose the GNP residuals using ACF and the `moments` package to calculate `skewness` and `kurtosis`.

Thinking...

Results

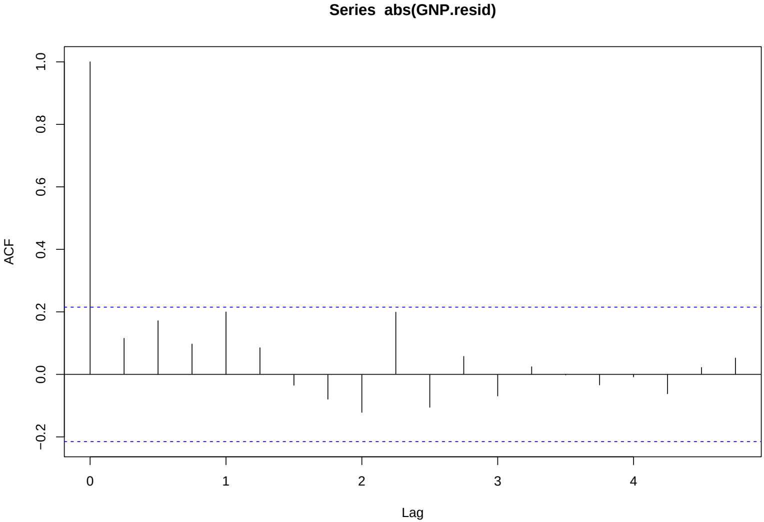
Very thick tailed and serially correlated as evidenced by the usual statistical suspects. But no volatility clustering.

```r
acf(GNP.resid)
```



Nice absolute values (i.e., GNP growth sizes):

```
acf(abs(GNP.resid))
```




```r
require(moments)
skewness(GNP.resid)
```

```
## [1] 0.1547011
```

```r
kurtosis(GNP.resid)
```

```
## [1] 3.639148
```

Positively skewed and thick tailed.

By the by: Where's the forecast?

```
(GNP.pred <- predict(fit.rate, n.ahead = 8))
```

```
## $pred
##          Qtr1     Qtr2     Qtr3     Qtr4
## 2016 1.865057 1.636645 1.687079 1.565599
## 2017 1.618362 1.547655 1.589601 1.545804
##
## $se
##          Qtr1     Qtr2     Qtr3     Qtr4
## 2016 1.154330 1.278538 1.347256 1.357198
## 2017 1.367230 1.367810 1.369599 1.369602
```

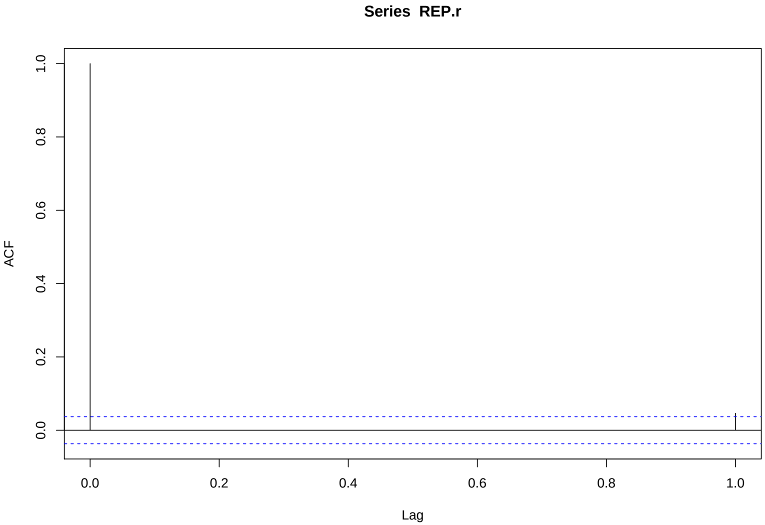
Give it the boot

Goal: An example of simulation-based inference.

- The context is just how dependent is today's stock return on yesterday's?
- We want to use the distribution of real-world returns data, without needing assumptions about normality.
- The null hypothesis is lack of dependence (i.e., an efficient market).
- So repeatedly, the data is changed using the `replicate` function, and the sample ACF is computed.
- This gives us the distribution of the ACF under the null hypotheses, H0: independence while using the empirical distribution of the returns data.

Let's use the Repsol returns. Pull the 1st autocorrelation from the sample:

```
acf(REP.r, 1)
```



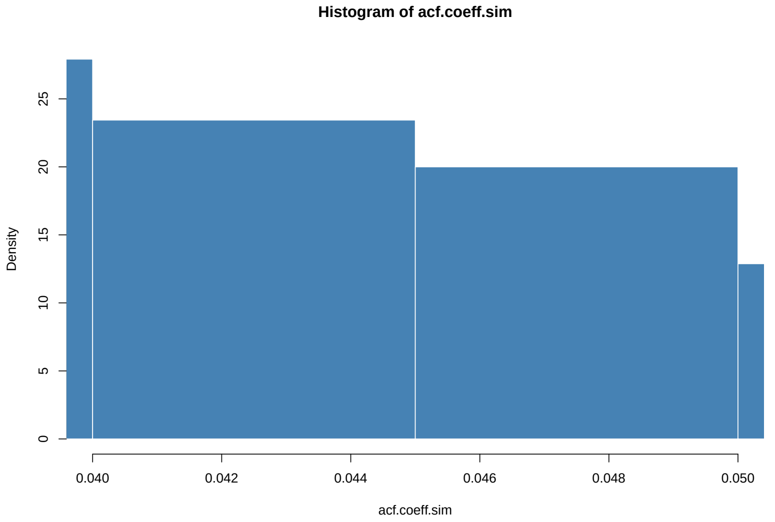
Not much to see – barely a blip – but over the 95% line. Let's further test this idea.

- Obtain 2500 draws from the distribution of the first autocorrelation using the `replicate` function.
- We operate under the null hypothesis of independence, assuming rational markets (i.e, rational markets is a "maintained hypothesis").

```
set.seed(1016)
acf.coeff.sim <- replicate(2500, acf(sample(REP.r,
    size = 2500, replace = FALSE), lag = 2,
    plot = FALSE)$acf[2])
summary(acf.coeff.sim)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.02473  0.02547  0.03557  0.03536  0.04542  0.08887
```

```r
hist(acf.coeff.sim, probability = TRUE,
    breaks = "FD", xlim = c(0.04, 0.05),
    col = "steelblue", border = "white")
```



Try this out

Investigate tolerances of 5% and 1% from both ends of the distribution of the 1-lag acf coefficient using these statements.

```
# At 95% tolerance level
quantile(acf.coeff.sim, probs = c(0.025,
    0.975))
```

```
##        2.5%       97.5%
## 0.004894726 0.065070914
```

```
# At 99% tolerance level
quantile(acf.coeff.sim, probs = c(0.005,
    0.995))
```

```
##         0.5%        99.5%
## -0.005140404  0.076992807
```


Here is the simulated t-stat and p-value.

```
# And the
(t.sim <- mean(acf.coeff.sim)/sd(acf.coeff.sim))
```

```
## [1] 2.321057
```

```
(p_value <- 1 - pt(t.sim, df = 2500 -
    2))
```

```
## [1] 0.01018194
```

- If we set the significance level at 1% that we would be wrong in rejecting the null hypothesis that the estimate is no different from zero $\cdots$
- And if we find that the **p-value** is less than the significance level, then we would be probably right to reject the null hypothesis.

Thinking. . .

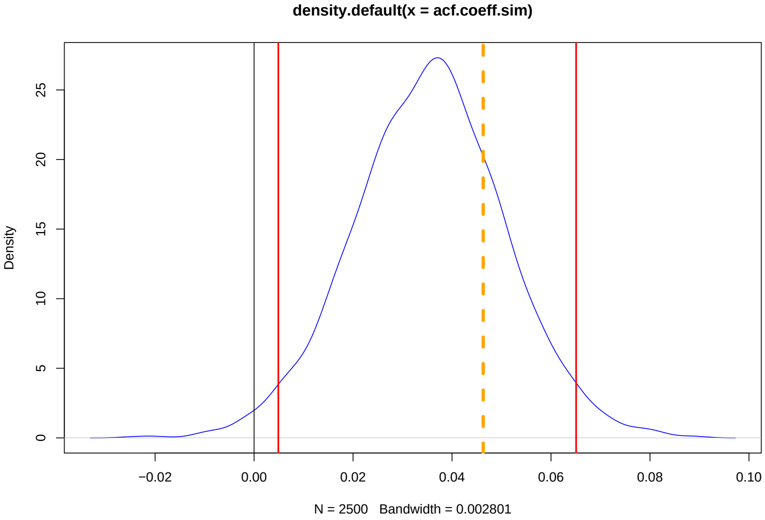
Results

Some (highly preliminary and provisional answers)

1. Quantile values are very narrow...
2. How narrow (feeling like rejecting the null hypothesis)?
3. t-stat is huge, but...
4. ...no buts!, the probability that we would be wrong to reject the null hypothesis is very small.

Plot the simulated density and lower and upper quantiles, along with the estimate of the lag-1 coefficient:

```r
plot(density(acf.coeff.sim), col = "blue")
abline(v = 0)
abline(v = quantile(acf.coeff.sim, probs = c(0.025,
    0.975)), lwd = 2, col = "red")
abline(v = acf(REP.r, 1, plot = FALSE)$acf[2],
    lty = 2, lwd = 4, col = "orange")
```


density.default(x = acf.coeff.sim)
Density
0
5
10
15
20
25
−0.02
0.00
0.02
0.04
0.06
0.08
0.10
N = 2500   Bandwidth = 0.002801

Can we reject the null hypothesis that the coefficient $= 0$? Is the market "efficient"?

1. Reject the null hypothesis since there is a less than 0.02% chance that the coefficient is zero.
2. Read [Fama(2013, p. 365-367)]https://www.nobelprize.org/nobel_prizes/economic-sciences/laureates/2013/fama-lecture.pdf for a diagnosis.
3. If the model is correct (ACF lag-1) then the previous day's return can predict today's return according to our analysis. Thus the market would seem to be inefficient.
4. This means we might be able to create a profitable trading strategy that makes use of the little bit of correlation we found to be significant (net of the costs of trading).

The wrap

- Lots more `R` practice
- ACF and PACF to do EDA on time series
- Stylized facts of financial returns
- Simulated coefficient inference to check efficient markets hypothesis
- Probability distributions
- Risk tolerance from an inference point of view
- Yahoo finance data graps
- Average regression and quantile regression

To prepare for the live session:

List these:

1. What are the top 3 key learnings for you from this segment?
2. What pieces of this segment are still a mystery?
3. What parts would you like more practice on?
4. Review the assignment. What questions do you have about the assignment for the live session?

Thanks! Till next week. . .