# Week 2 – R Data Modeling

# Making distributions

- As always, let's load some data: let's use and open data package called `pdfetch`. This is a portal to finance and government data, including Yahoo Finance.
- Let's go to the Bureau of Labor Statistics (BLS) and load the export-import price index at `http://data.bls.gov/timeseries/EIUIR?output_view=pct_1mth`
- Look up the symbols "EIUIR" and "EIUIR100".

```
require(pdfetch)
```

```
## Warning: package 'pdfetch' was built under R version 3.2.5
```

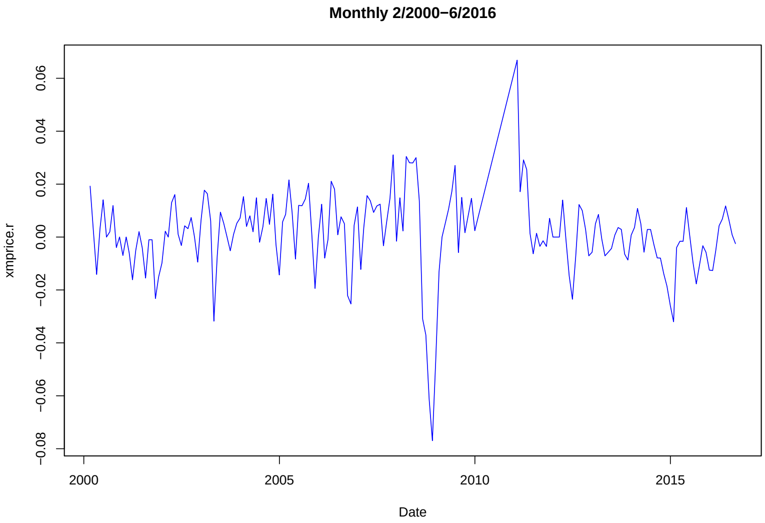
```
require(xts)
```

```
## Warning: package 'zoo' was built under R version 3.2.5
```

```
require(zoo)
EIUIR <- pdfetch_BLS(c("EIUIR", "EIUIR100"),
    2000, 2016)  # start and end years
head(EIUIR)
```

```
##            EIUIR EIUIR100
## 2000-01-31  97.8     87.2
```

```r
plot(xmprice.r, type = "l", col = "blue",
    xlab = "Date", main = "Monthly 2/2000-6/2016")
```



```r
xmprice.r.df <- data.frame(xmprice.r,
    Date = index(xmprice.r), Rate = xmprice.r[,
        1], Rate.abs = abs(xmprice.r[,
        1]))
head(xmprice.r.df)
```

```
##                   EIUIR       Date        Rate    Rate.abs
## 2000-02-29  0.019241100 2000-02-29  0.019241100 0.019241100
## 2000-03-31  0.002004009 2000-03-31  0.002004009 0.002004009
## 2000-04-30 -0.014113137 2000-04-30 -0.014113137 0.014113137
## 2000-05-31  0.003041057 2000-05-31  0.003041057 0.003041057
## 2000-06-30  0.014070584 2000-06-30  0.014070584 0.014070584
## 2000-07-31  0.000000000 2000-07-31  0.000000000 0.000000000
```

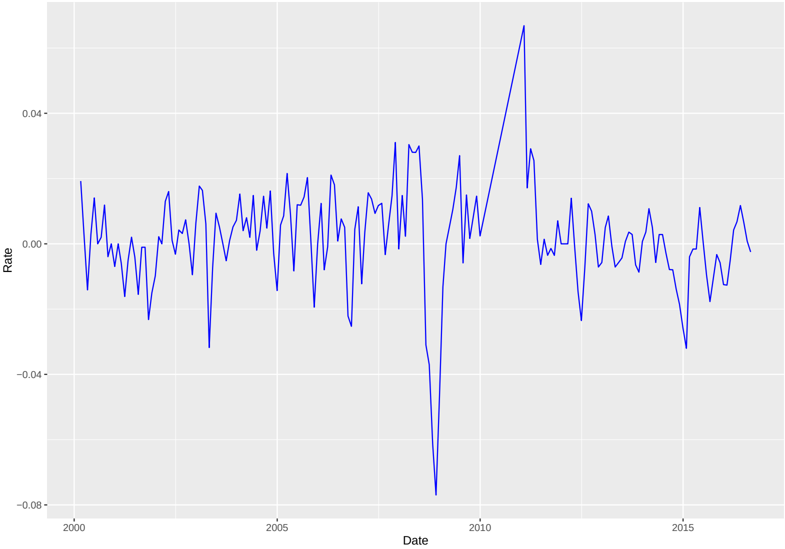
```r
str(xmprice.r.df)
```

```
## 'data.frame':    187 obs. of  4 variables:
##  $ EIUIR   : num  0.01924 0.002 -0.01411 0.00304 0.01407 ...
##  $ Date    : Date, format: "2000-02-29" "2000-03-31" ...
##  $ Rate    : num  0.01924 0.002 -0.01411 0.00304 0.01407 ...
##  $ Rate.abs: num  0.01924 0.002 0.01411 0.00304 0.01407 ...
```

- A “prettier” plot with the ggplot2 package
- Use aes, “aesthetics”, to pick x (horizontal) and y (vertical) axes
- Use geom_line to build the plot

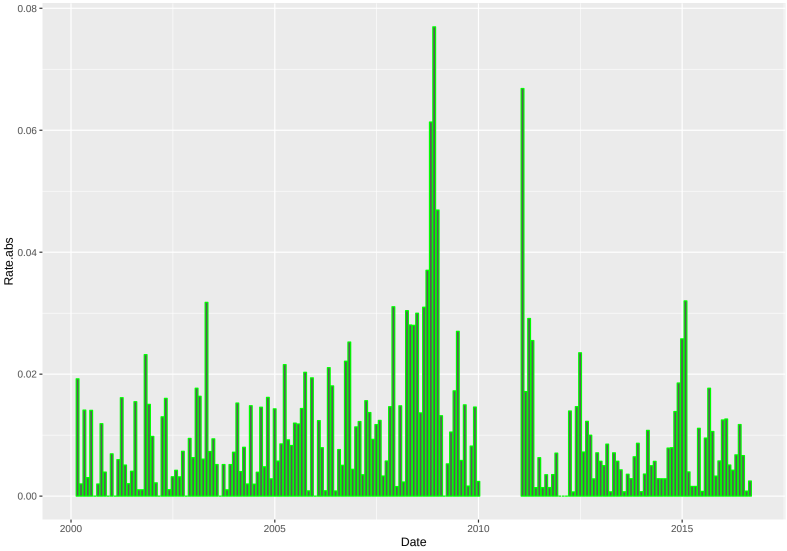
```r
require(ggplot2)
ggplot(xmprice.r.df, aes(x = Date, y = Rate)) +
    geom_line(colour = "blue")
```

```
## Warning: package 'ggplot2' was built under R version 3.2.5
```



- Let's try a bar graph of the absolute value of price rates.
- Use `geom_bar` to build this picture.

```r
require(ggplot2)
ggplot(xmprice.r.df, aes(x = Date, y = Rate.abs)) +
    geom_bar(stat = "identity", colour = "green")
```


0.08
0.06
0.04
0.02
0.00
Rate.abs
2000
2005
2010
2015
Date

# Try this

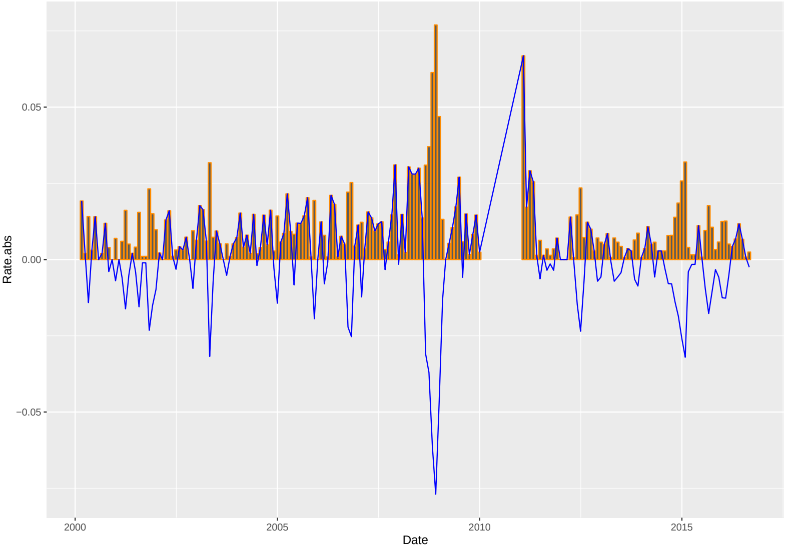
- Overlay returns (geom_line) and their absolute value geom_bar.
- ggplot declares the canvas using the price data frame.
- aes establishes the data series to be used to generate pictures.
- geom_bar builds bar chart.
- geom_line overplots bar chart with a line chart.

```r
require(ggplot2)
ggplot(xmprice.r.df, aes(Date, Rate.abs)) +
    geom_bar(stat = "identity", colour = "darkorange") +
    geom_line(data = xmprice.r.df, aes(Date,
        Rate), colour = "blue")
```

By examining this chart, what business questions about your Univeral Export-Import Ltd supply chain could this help answer? Why is this helpful?

Thinking. . .

# Results



1. Answers the question: When supply and demand tightens, does price volatility cluster?
2. If we are selling, we would experience strong swings in demand and thus in revenue at the customer fulfillment end of the chain.
3. If we are buying, we would experience strong swings in cost and input product utilization at the procurement end of the chain.
4. For the financial implications: we would have a tough time making the earnings we forecast to the market.

Whitman

SCHOOL *of* MANAGEMENT

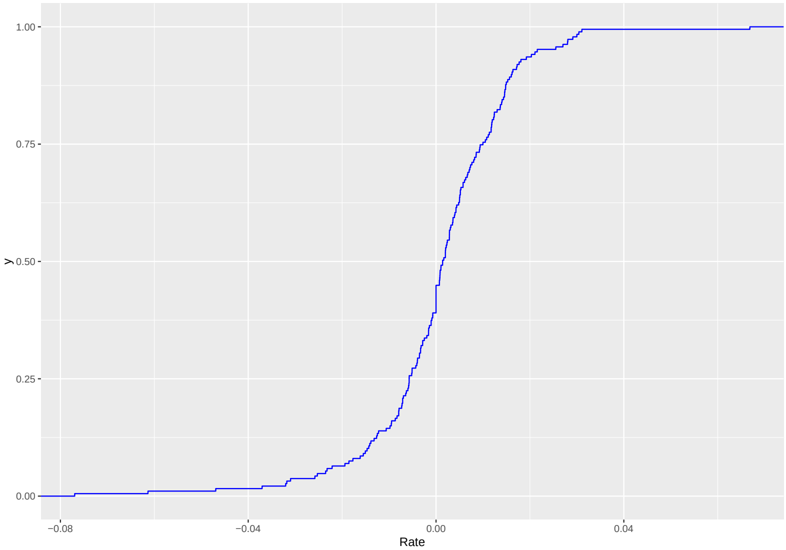
SYRACUSE UNIVERSITY

MBA@SYRACUSE

# Picture this

- We import goods as input to our manufacturing process.
- We might want to know the odds that a very high export-import rate might occur.
- We answer this with a cumulative distribution function (*cdf* or *CDF*) plot. _ we build this plot using the `stat_ecdf()` function in `ggplot2`.

```r
require(ggplot2)
ggplot(xmprice.r.df, aes(Rate)) + stat_ecdf(colour = "blue")
```


1.00
0.75
0.50
0.25
0.00
y
−0.08
−0.04
0.00
0.04
Rate

# Try this

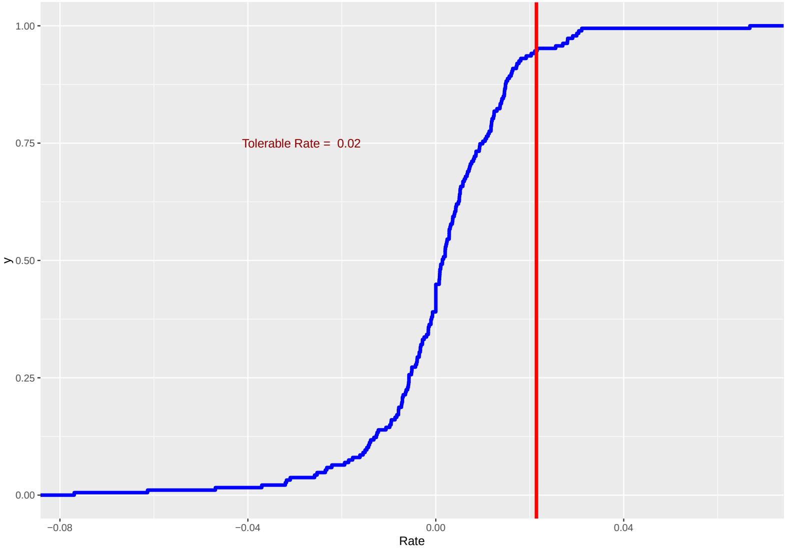
1. Suppose the procurement team's delegation of authority remit states: "Procurement may approve input invoices when there is only a 5% chance that prices will rise any higher than the price rate associated with that tolerance. If input prices do rise higher than the tolerable rate, you must get divisional approval."
2. Plot a vertical line to indicate the maximum tolerable rate for procurement using the BLS EIUR data from 2000 to the present.

- Use `r.tol <- quantile(xmprice.r.df$Rate, 0.95)` to find the tolerable rate.
- Use `+ geom_vline(xintercept = r.tol)` in the CDF plot.

Thinking. . .

# Result

```r
require(ggplot2)
r.tol <- quantile(xmprice.r.df$Rate,
    0.95)
r.tol.label <- paste("Tolerable Rate = ",
    round(r.tol, 2))
ggplot(xmprice.r.df, aes(Rate)) + stat_ecdf(colour = "blue",
    size = 1.5) + geom_vline(xintercept = r.tol,
    colour = "red", size = 1.5) + annotate("text",
    x = r.tol - 0.05, y = 0.75, label = r.tol.label,
    colour = "darkred")
```


Tolerable Rate = 0.02
1.00
0.75
0.50
0.25
0.00
y
−0.08
−0.04
0.00
0.04
Rate

## A little more than you bargained for?

- We used the `paste` and `round` (to two, 2, decimal places) functions to make a label.
- We made much thicker lines (`size = 1.5`).
- 2% is where the line is drawn.

That was intense!

Whitman

SCHOOL *of* MANAGEMENT

SYRACUSE UNIVERSITY

MBA@SYRACUSE

# Next on the agenda

Now that we have *made* some distributions out of live data, let's estimate the parameters of specific distributions that might be fit to that data.

Whitman
SCHOOL of MANAGEMENT
SYRACUSE UNIVERSITY

MBA@SYRACUSE