About `xlsx`

Peter Carl

"Favorite Packages"
Chicago R User Group, 1 May 2013

xlsx: Read, Write, Format Excel in R

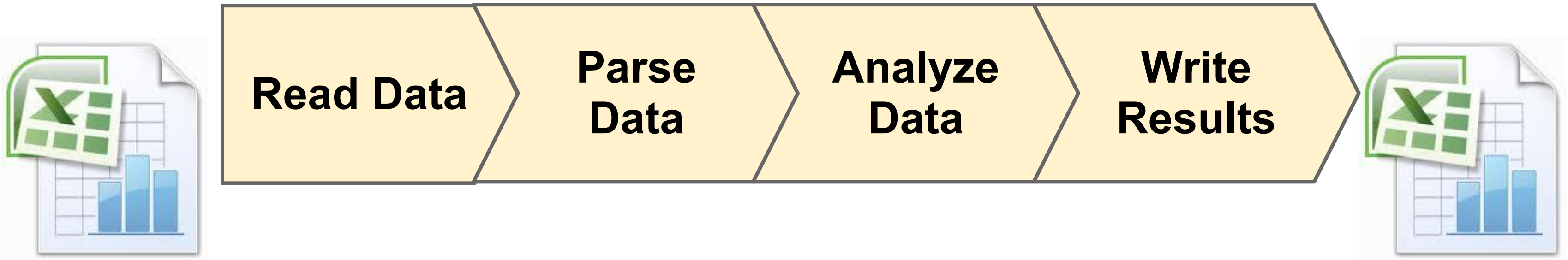
- Control the appearance of the spreadsheet by setting data formats, fonts, colors, borders.
- Available on CRAN
- Uses the Apache POI API, so Excel isn't needed
- Current version: 0.5
- Depends: xlsxjjars, rJava
- Published: 2013-03-18
- Author and Maintainer: Adrian A. Dragulescu <adrian.dragulescu at gmail.com>
- License: GPL-3
- URL: http://code.google.com/p/rexcel/

Apache POI

- Under the hood, xlsx uses a proven, pre-existing and separately developed API between Java and Excel 2007
- Apache POI is a mature project focused on creating and maintaining Java APIs for manipulating file formats based on the Office Open XML standards (OOXML) and Microsoft’s OLE 2 Compound Document format (OLE2)
- More information is at: http://poi.apache.org/

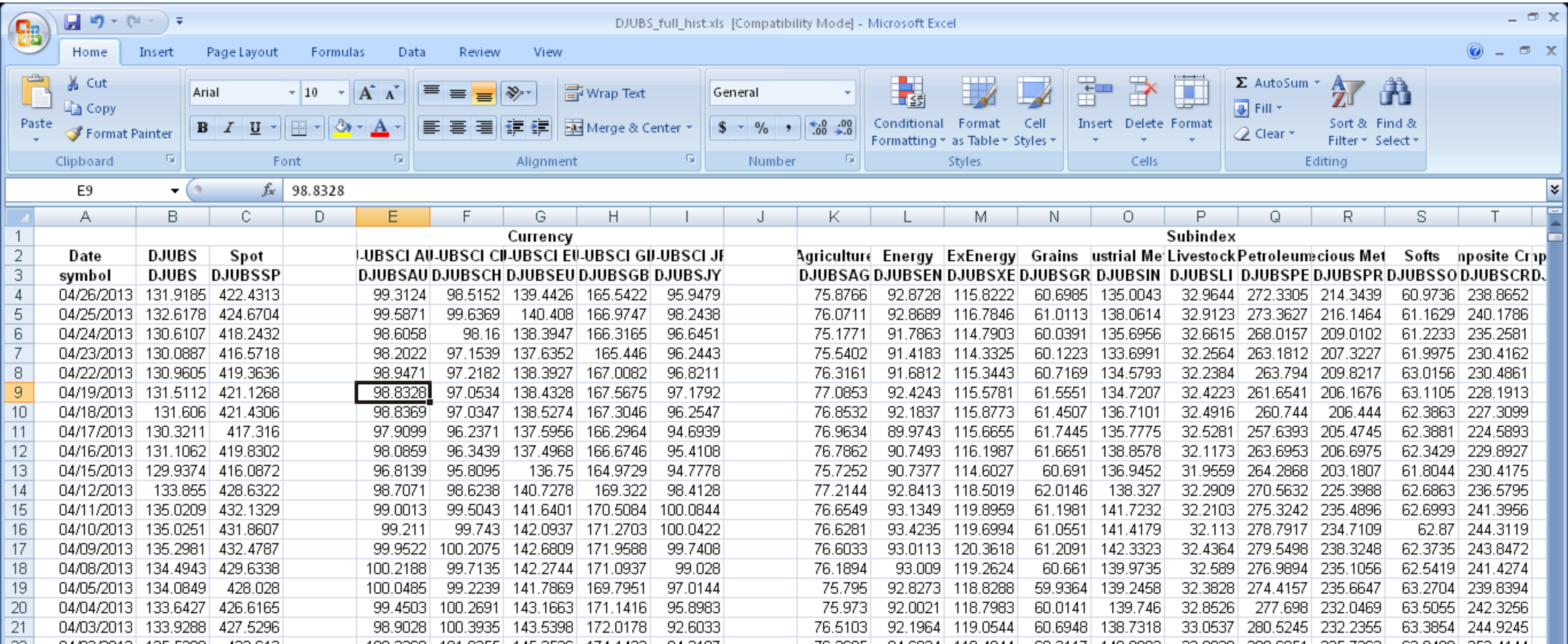
- Uses the rJava package to link Java and R
- All the heavy lifting of parsing XML schemas is being done in Java.

An Example Workflow



input.xls

output.xls



E9 98.8328

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	
1					Currency						Subindex										
2	Date	DJUBS	Spot		-UBSCI AU	-UBSCI CI	-UBSCI EU	-UBSCI GI	-UBSCI JI		Agriculture	Energy	ExEnergy	Grains	ustrial Me	Livestock	Petroleum	cious Met	Softs	nposite Cr	np
3	symbol	DJUBS	DJUBSSP		DJUBSAU	DJUBSCH	DJUBSEU	DJUBSGB	DJUBSJY		DJUBSAG	DJUBSEN	DJUBSXE	DJUBSGR	DJUBSIN	DJUBSLI	DJUBSPE	DJUBSPR	DJUBSSO	DJUBSCR	DJ
4	04/26/2013	131.9185	422.4313		99.3124	98.5152	139.4426	165.5422	95.9479		75.8766	92.8728	115.8222	60.6985	135.0043	32.9644	272.3305	214.3439	60.9736	238.8652	
5	04/25/2013	132.6178	424.6704		99.5871	99.6369	140.408	166.9747	98.2438		76.0711	92.8689	116.7846	61.0113	138.0614	32.9123	273.3627	216.1464	61.1629	240.1786	
6	04/24/2013	130.6107	418.2432		98.6058	98.16	138.3947	166.3165	96.6451		75.1771	91.7863	114.7903	60.0391	135.6956	32.6615	268.0157	209.0102	61.2233	235.2581	
7	04/23/2013	130.0887	416.5718		98.2022	97.1539	137.6352	165.446	96.2443		75.5402	91.4183	114.3325	60.1223	133.6991	32.2564	263.1812	207.3227	61.9975	230.4162	
8	04/22/2013	130.9605	419.3636		98.9471	97.2182	138.3927	167.0082	96.8211		76.3161	91.6812	115.3443	60.7169	134.5793	32.2384	263.794	209.8217	63.0156	230.4861	
9	04/19/2013	131.5112	421.1268		98.8328	97.0534	138.4328	167.5675	97.1792		77.0853	92.4243	115.5781	61.5551	134.7207	32.4223	261.6541	206.1676	63.1105	228.1913	
10	04/18/2013	131.606	421.4306		98.8369	97.0347	138.5274	167.3046	96.2547		76.8532	92.1837	115.8773	61.4507	136.7101	32.4916	260.744	206.444	62.3863	227.3099	
11	04/17/2013	130.3211	417.316		97.9099	96.2371	137.5956	166.2964	94.6939		76.9634	89.9743	115.6655	61.7445	135.7775	32.5281	257.6393	205.4745	62.3881	224.5893	
12	04/16/2013	131.1062	419.8302		98.0859	96.3439	137.4968	166.6746	95.4108		76.7862	90.7493	116.1987	61.6651	138.8578	32.1173	263.6953	206.6975	62.3429	229.8927	
13	04/15/2013	129.9374	416.0872		96.8139	95.8095	136.75	164.9729	94.7778		75.7252	90.7377	114.6027	60.691	136.9452	31.9559	264.2868	203.1807	61.8044	230.4175	
14	04/12/2013	133.855	428.6322		98.7071	98.6238	140.7278	169.322	98.4128		77.2144	92.8413	118.5019	62.0146	138.327	32.2909	270.5632	225.3988	62.6863	236.5795	
15	04/11/2013	135.0209	432.1329		99.0013	99.5043	141.6401	170.5084	100.0844		76.6549	93.1349	119.8959	61.1981	141.7232	32.2103	275.3242	235.4896	62.6993	241.3956	
16	04/10/2013	135.0251	431.8607		99.211	99.743	142.0937	171.2703	100.0422		76.6281	93.4235	119.6994	61.0551	141.4179	32.113	278.7917	234.7109	62.87	244.3119	
17	04/09/2013	135.2981	432.4787		99.9522	100.2075	142.6809	171.9588	99.7408		76.6033	93.0113	120.3618	61.2091	142.3323	32.4364	279.5498	238.3248	62.3735	243.8472	
18	04/08/2013	134.4943	429.6338		100.2188	99.7135	142.2744	171.0937	99.028		76.1894	93.009	119.2624	60.661	139.9735	32.589	276.9894	235.1056	62.5419	241.4274	
19	04/05/2013	134.0849	428.028		100.0485	99.2239	141.7869	169.7951	97.0144		75.795	92.8273	118.8288	59.9364	139.2458	32.3828	274.4157	235.6647	63.2704	239.8394	
20	04/04/2013	133.6427	426.6165		99.4503	100.2691	143.1663	171.1416	95.8983		75.973	92.0021	118.7983	60.0141	139.746	32.8526	277.698	232.0469	63.5055	242.3256	
21	04/03/2013	133.9288	427.5296		98.9028	100.3935	143.5398	172.0178	92.6033		76.5103	92.1964	119.0544	60.6948	138.7318	33.0537	280.5245	232.2355	63.3854	244.9245	

Examine the Data

```
# Look at the resulting data
> head(x.R)
                Aluminum  Brent Crude        Coffee Copper (COMEX)
1991-01-02            NA           NA            NA             NA
1991-01-03  0.0110040000 -0.045238000  0.0138090000   -0.024966000
1991-01-04  0.0004599388 -0.058984333 -0.0037413359   -0.003259374
1991-01-07  0.0060614809  0.150057989  0.0174145756    0.008306786
1991-01-08 -0.0166027909 -0.026213992  0.0007347181   -0.019509577
1991-01-09 -0.0055101154  0.008863234 -0.0031341165   -0.008988240
... snip ...
```

Analyze the Data

```
# Create a table of summary statistics
x.RiskStats = as.data.frame(t(table.RiskStats(x.R)))
```

```
> x.RiskStats
       Annualized Return Annualized Std Dev Annualized Sharpe Ratio

Aluminum          -0.0110             0.2022                 -0.0542
Brent Crude        0.1233             0.3080                  0.4002
Coffee            -0.0403             0.3745                 -0.1075
Copper (COMEX)     0.0909             0.2690                  0.3379
Corn              -0.0387             0.2538                -0.1525
```

Write a Table into a Spreadsheet

```
# Create a new workbook for outputs
outwb <- createWorkbook()

# Define some cell styles within that workbook
csSheetTitle <- CellStyle(outwb) + Font(outwb, heightInPoints=14,
isBold=TRUE)
csSheetSubTitle <- CellStyle(outwb) + Font(outwb,
heightInPoints=12, isItalic=TRUE, isBold=FALSE)
csTableRowNames <- CellStyle(outwb) + Font(outwb, isBold=TRUE)
csTableColNames <- CellStyle(outwb) + Font(outwb, isBold=TRUE) +
Alignment(wrapText=TRUE, h="ALIGN_CENTER") + Border(color="black",
position=c("TOP", "BOTTOM"), pen=c("BORDER_THIN", "BORDER_THICK"))

csRatioColumn <- CellStyle(outwb, dataFormat=DataFormat("0.0")) #
... for ratio results
csPercColumn <- CellStyle(outwb, dataFormat=DataFormat("0.0%")) #
... for percentage results
```

Write a Table into a Spreadsheet

```
# Which columns in the table should be formatted how?
RiskStats.colRatio = list(
  '3'=csRatioColumn,
  '5'=csRatioColumn,
  '8'=csRatioColumn,
  '15'=csRatioColumn)
RiskStats.colPerc =list(
  '1'=csPercColumn,
  '2'=csPercColumn,
  '4'=csPercColumn,
  '6'=csPercColumn,
  '7'=csPercColumn,
  '9'=csPercColumn,
  '10'=csPercColumn,
  '13'=csPercColumn,
  '14'=csPercColumn)
```

Write a Table into a Spreadsheet

```
# Create a sheet in that workbook to contain the table
sheet <- createSheet(outwb, sheetName = "Performance
Table")

# Add the table calculated above to the new sheet
addDataFrame(x.RiskStats, sheet, startRow=3,
startColumn=1, colStyle= c(RiskStats.colPerc, RiskStats.
colRatio), colnamesStyle = csTableColNames, rownamesStyle
= csTableRowNames)
setColumnWidth(sheet, colIndex=c(2:15), colWidth=11)
setColumnWidth(sheet, colIndex=16, colWidth=13)
setColumnWidth(sheet, colIndex=17, colWidth=6)
setColumnWidth(sheet, colIndex=1, colWidth= 0.8*max(length
(rownames(x.RiskStats))))
```

Write a Table into a Spreadsheet

```
# Create the Sheet title ...
rows <- createRow(sheet,rowIndex=1)
sheetTitle <- createCell(rows, colIndex=1)
setCellValue(sheetTitle[[1,1]], "Ex-Post Returns and
Risk")
setCellStyle(sheetTitle[[1,1]], csSheetTitle)

# ... and subtitle
rows <- createRow(sheet,rowIndex=2)
sheetSubTitle <- createCell(rows,colIndex=1)
setCellValue(sheetSubTitle[[1,1]], "Since Inception")
setCellStyle(sheetSubTitle[[1,1]], csSheetSubTitle)
```

Write a Chart into a Spreadsheet

```
# Construct the chart as a  dib, emf, jpeg, pict, png, or
wmf file.
require(gplots)
skewedG2R20 = c(colorpanel(16, "darkgreen","yellow"),
colorpanel(5, "yellow", "darkred")[-1])

png(filename = "corr.jpeg", width = 6, height = 8, units =
"in", pointsize=12, res=120)
require(PApages)
page.CorHeatmap(x.R[,x.commodities], Colv=TRUE, breaks =
seq(-1,1,by=.1), symkey=TRUE, col=skewedG2R20, tracecol="
darkblue", cexRow=0.9, cexCol=0.9)

dev.off()
```

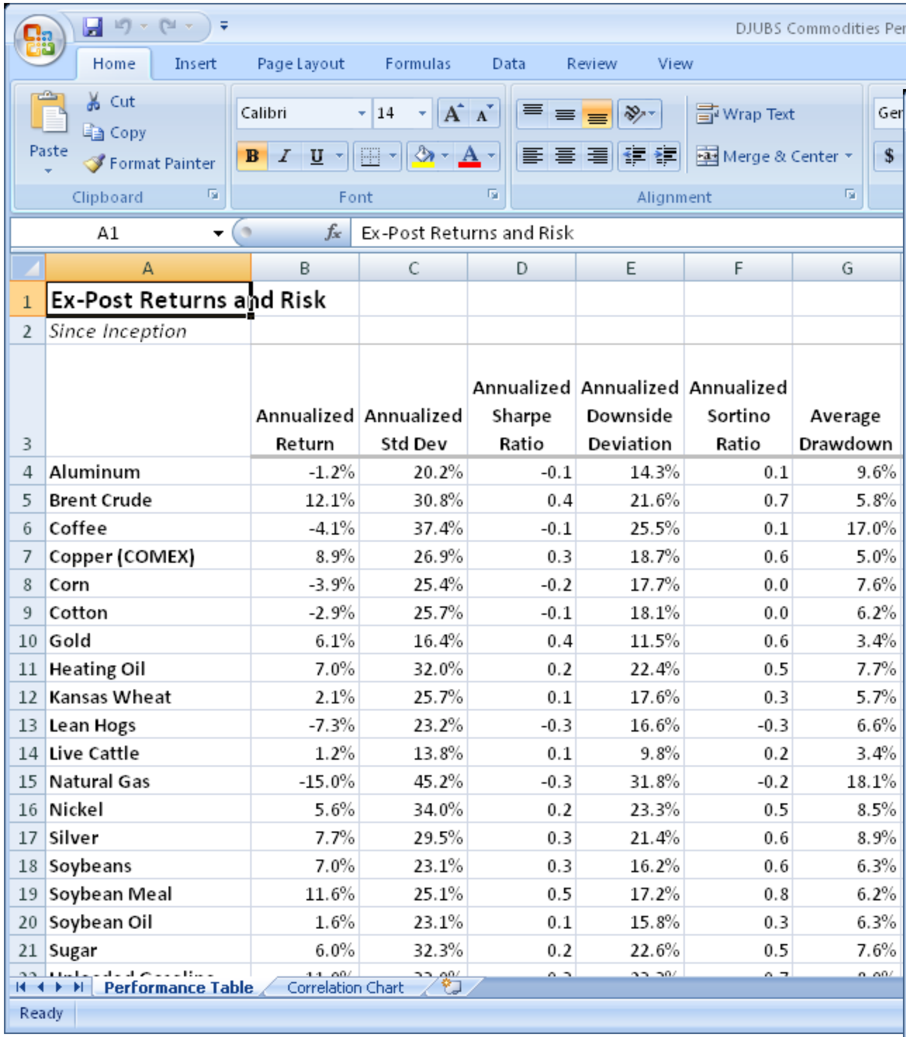
Write a Chart into a Spreadsheet

```
# Create a sheet in that workbook to contain the graph
sheet <- createSheet(outwb, sheetName = "Correlation
Chart")

# Create the Sheet title and subtitle
# ...snip... same as prior

# Add the file created previously
addPicture("corr.jpeg", sheet, scale = 1, startRow = 4,
startColumn = 1)
```

Save the Spreadsheet

```
# Save the workbook to a file...
saveWorkbook(outwb, "DJUBS Commodities Performance
Summary.xlsx")
```

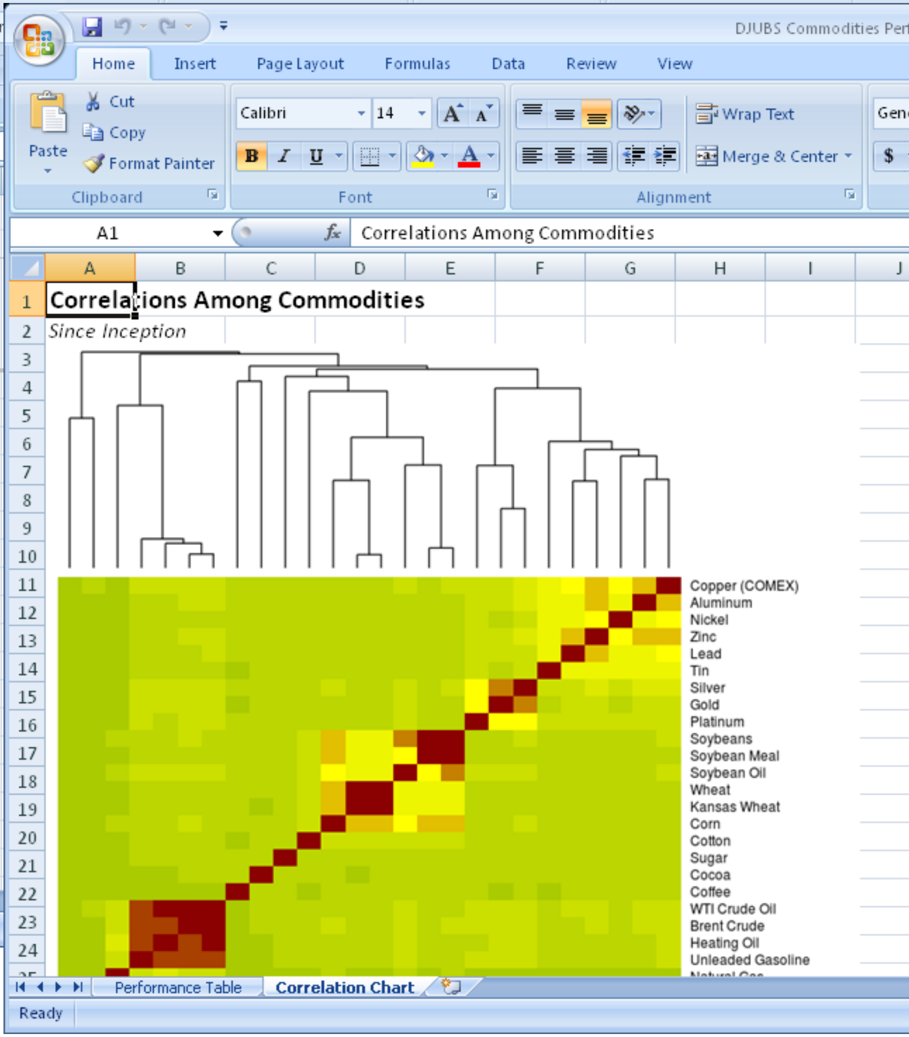
Resulting Spreadsheet



Ex-Post Returns and Risk

Since Inception

	Annualized Return	Annualized Std Dev	Annualized Sharpe Ratio	Annualized Downside Deviation	Annualized Sortino Ratio	Average Drawdown
Aluminum	-1.2%	20.2%	-0.1	14.3%	0.1	9.6%
Brent Crude	12.1%	30.8%	0.4	21.6%	0.7	5.8%
Coffee	-4.1%	37.4%	-0.1	25.5%	0.1	17.0%
Copper (COMEX)	8.9%	26.9%	0.3	18.7%	0.6	5.0%
Corn	-3.9%	25.4%	-0.2	17.7%	0.0	7.6%
Cotton	-2.9%	25.7%	-0.1	18.1%	0.0	6.2%
Gold	6.1%	16.4%	0.4	11.5%	0.6	3.4%
Heating Oil	7.0%	32.0%	0.2	22.4%	0.5	7.7%
Kansas Wheat	2.1%	25.7%	0.1	17.6%	0.3	5.7%
Lean Hogs	-7.3%	23.2%	-0.3	16.6%	-0.3	6.6%
Live Cattle	1.2%	13.8%	0.1	9.8%	0.2	3.4%
Natural Gas	-15.0%	45.2%	-0.3	31.8%	-0.2	18.1%
Nickel	5.6%	34.0%	0.2	23.3%	0.5	8.5%
Silver	7.7%	29.5%	0.3	21.4%	0.6	8.9%
Soybeans	7.0%	23.1%	0.3	16.2%	0.6	6.3%
Soybean Meal	11.6%	25.1%	0.5	17.2%	0.8	6.2%
Soybean Oil	1.6%	23.1%	0.1	15.8%	0.3	6.3%
Sugar	6.0%	32.3%	0.2	22.6%	0.5	7.6%



Correlations Among Commodities

Since Inception

Similar packages

- XLConnect
  - Well-documented package for reading and writing, but *formatting* remains limited

- gdata's `read.xls`
  - Fast and convenient for reading Excel sheets

- excel.link
  - Provides dynamic access to Excel, but requires Excel...

R/Finance 2013 - May 17, 18

Two day conference covers a wide variety of topics in Finance within the context of using R as a primary tool for financial risk management, analysis and trading.

Organized by a local group of R package authors and community contributors, and hosted by the **International Center for Futures and Derivatives** at the **University of Illinois at Chicago**.

Keynote speakers this year:

- **Sanjiv Das** - Santa Clara University; Author of Derivatives: Principles and Practice;
- **Attilio Meucci** - Chief Risk Officer at Kepos Capital, LP; Author of Risk and Asset Allocation
- **Ryan Sheftel** - Managing Director for Electronic Market Making at Credit Suisse
- **Ruey Tsay** - University of Chicago; Author of An Introduction to Analysis of Financial Data with R

Register now: http://rinfinance.com

Contact Information

Peter Carl

pcarl@gsb.uchicago.edu

Slides and complete R script is available at:

http://tradeblotter.wordpress.com/