

```
library(dplyr)  
  
r-ladies_global %>%  
  filter(from = 'Taipei', travel_to = 'Madrid')
```



R-Ladies Taipei

R in Finance Industries: Focus on Trading Strategy Development Framework

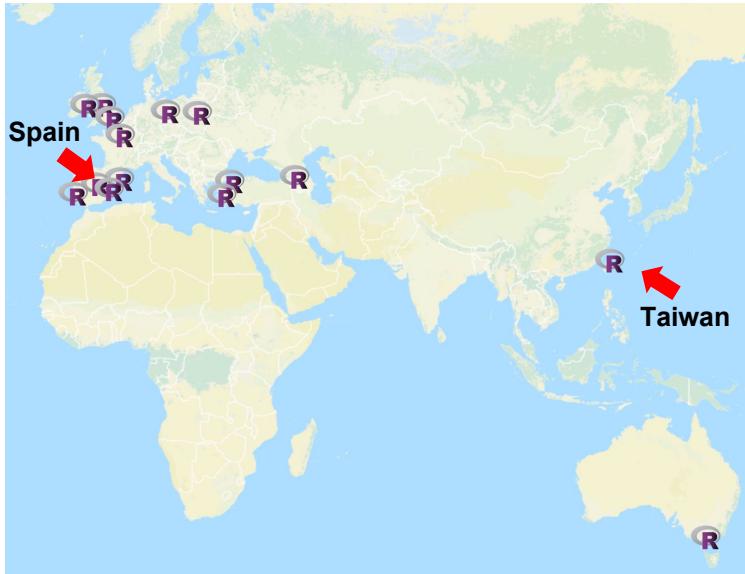


Hello!

I am Chiayi Yen (顏嘉儀).

- Data science consultant @iiNumbers
- P2P lending credit risk modeling consultant @[LendBand](#)
- PhD student @National Taiwan University
- Co-founder of [R-Ladies Taipei](#)
- Microsoft Most Valuable Professionals (MVP) 2017
- Contact me: yen.chiayi@gmail.com

R-Ladies Taipei



- Founded in 2014
- Largest female data science community
- every month in Taipei,
- 30+ meetup, 33+ female speakers
- Our history: goo.gl/HbHNeP

R-Ladies Taipei, together with PyLadies and Girls in Tech Taiwan, teaches Taiwan President Tsai how to code in the event of “Hour of Code 2016” held by Microsoft Taipei.

R-Ladies Madrid (& Barcelona)

Happy Birthday to R-Ladies Taipei 2016



CREATED WITH wevideo FREE VERSION

Barcelona

Hello, I'm Rebeca from R-Ladies Barcelona

0:03 / 1:45

CC HD

This video player displays a video of three women sitting around a table in a restaurant, identified as being from Barcelona. The video is created with the WeVideo free version. A caption at the bottom of the screen reads "Hello, I'm Rebeca from R-Ladies Barcelona". The video has a duration of 0:03 / 1:45 and includes standard video controls like play/pause, volume, and a timestamp.



https://www.youtube.com/watch?v=6d5UTy_Yc34

The Agenda



first

What's possible
that financial
institution
leverage data
technology and
R's power on
their business.



second

Useful R tools
about building a
trading strategy,
such as Quantmod,
Blotter, Quantstrat,
and others.



third

Quantstrat,
a powerful
framework used
to develop
quantitative
trading strategy.



last

Challenges
about data
manipulation and
ML, and its potential
solutions, like MS R
Server, Azure ML
Studio. And others.

1.

What's possible

that financial institution leverage data technology and R's power on their business, according to iiNumbers' experience in Taiwan.



Our Clients



Finance

日盛期貨
Jih Sun Futures

台新銀行
Taishin International Bank

永豐銀行
Sinopac Financial Holdings Co,Ltd

國泰人壽
Cathay Financial Holdings Co,Ltd

E-commerce

台灣大哥大
Taiwan Mobile

udn聯合新聞網
udn.com

TransBiz

Trip+

Schools & Government

工研院
Industrial Technology Research Institute

台北電腦公會
Taipei Computer Association

台大語言所
NTU-Graduate Institute of Linguistics

東吳巨資院
SCU-School of Big Data Management

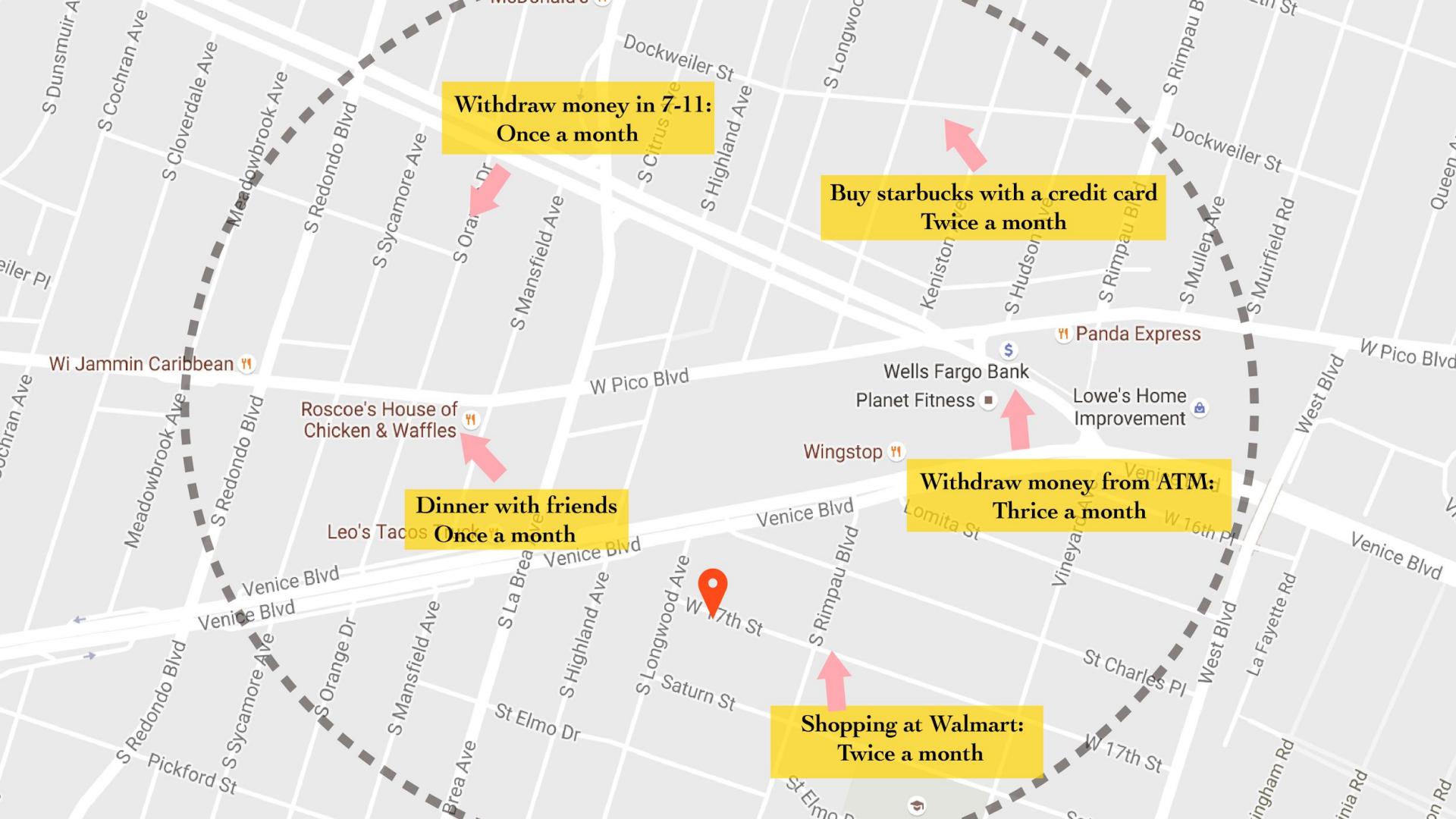
台新銀行

Taishin International Bank

Customer ID	behavior	date	location	ATM ID
u1	credit card	2/10 8:30	Starbuck: TP 101	m1
u2	Withdraw cash	2/10 12:00	7-11: NTU	m2
u1	credit card	2/10 21:00	Roscoe's house of Chicken & Waffles	m3
...

Credit Card User Behavior Analysis

- Q: effective marketing strategy
- Inner Bank data + Public GIS data
- Tools
 - Develop 200+ R crawlers
 - Deploy on Microsoft Azure



Withdraw money in 7-11:
Once a month



Buy starbucks with a credit card
Twice a month



Dinner with friends
Once a month

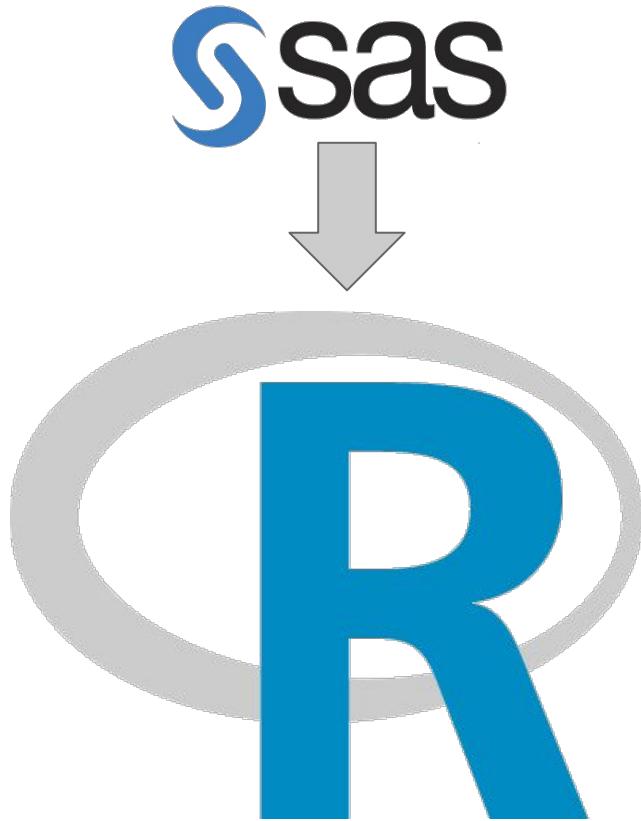


Withdraw money from ATM:
Thrice a month



Shopping at Walmart:
Twice a month





國泰人壽 Cathay Life Insurance

Data Storage Solution

- Q: build environment for big data demand
- Tools
 - Hadoop
 - Microsoft R Server
- Benefit
 - Reusable code
 - Visualization and user interface via R Shiny



日盛期貨 Jih Sun Futures

API development & deploy

- Q: new business opportunity from their unique data (not allowed to re-sell)
- A: API as Service on Azure marketplace

Quantitative trading modeling

- Quantstrat and other analysis tools about backtesting

2. Useful R tools

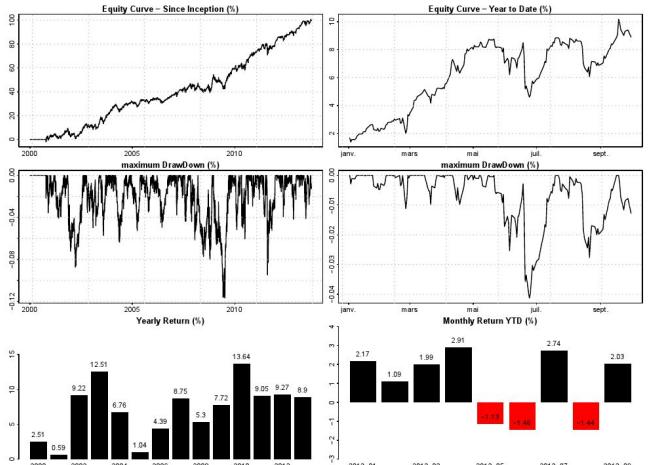
about building a trading strategy, such as Quantmod, Blotter, and others.

Dynamic Equity vs. Bond Allocation
Performance Report



Monthly Percentage Return (gross of fees)

	janv.	avr.	mars	mai	juin	jul.	sept.	oct.	nov.	dc.	YTD
2000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.80	-2.20	1.90	2.50
2001	0.40	1.00	-0.10	-2.30	0.10	-0.60	3.00	1.10	1.50	2.50	-2.90
2002	0.00	-0.10	-4.50	3.10	1.00	-0.50	-3.40	2.50	-2.00	2.50	-2.20
2003	-2.90	2.60	-1.60	3.20	5.10	0.70	-2.50	2.80	1.90	3.40	1.00
2004	1.80	1.80	-0.30	-3.50	0.90	0.60	1.30	2.50	-0.80	0.80	0.20
2005	-1.30	-0.30	-1.90	1.20	3.50	0.10	-0.10	-0.10	1.20	-0.10	1.00
2006	1.50	0.00	0.00	-0.20	-2.00	-0.10	1.30	0.80	1.00	1.00	0.40
2007	0.00	1.80	2.20	1.10	0.60	-0.40	3.00	3.10	0.00	0.00	-0.10
2008	-3.20	2.20	0.30	-2.60	-1.60	0.00	0.80	1.60	-1.70	-1.30	8.40
2009	-2.40	-0.70	2.00	-2.50	-1.90	-0.50	3.80	3.90	1.80	0.30	4.20
2010	-1.10	1.80	1.40	2.50	-2.60	-1.80	6.40	-0.70	4.40	1.50	-0.30
2011	1.20	2.40	0.00	2.90	1.10	-1.60	1.80	-3.80	1.40	-0.80	0.90
2012	0.00	0.10	0.70	2.20	1.10	2.40	1.10	0.20	1.10	0.20	0.70
2013	2.20	1.10	2.00	2.90	-1.10	-1.50	2.70	-1.40	2.00	0.50	9.30



“ Backtest, is a way to test your trading strategy by using the past price data.

Winning ratio

Max drawdown

Sharpe ratio

Turnover

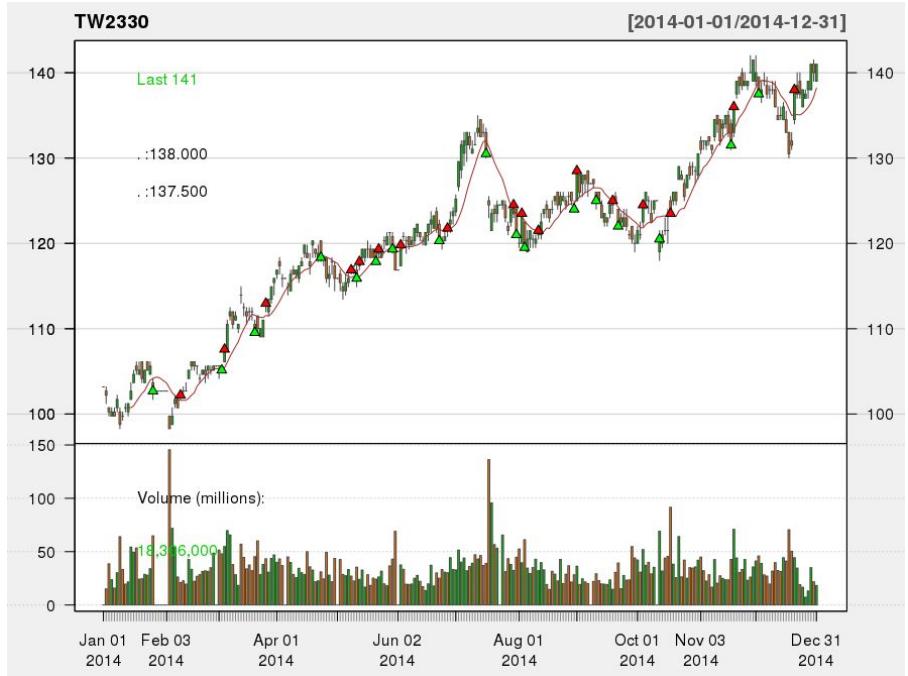
Performance	(%)	Draw Down	(%)	Monthly	(%)	Daily	(%)
Ann.Return	7.17	maxDD	-11.7	HIT Rate	61	HIT Rate	56
Ann.Volatility	8.21	maxDD Date	2009-07-08	Mean Return	0.6	Mean Return	0.03
Sharpe Ratio	0.87	Time to Recover	51 days	Mean > 0	1.77	Mean > 0	0.36
				Mean < 0	-1.39	Mean < 0	-0.39
				Worst	-4.54	Worst	-4.69
				Best	8.38	Best	4.1



Let's Start with a Simple Trading Strategy

‘Hello World’ in Trading Strategy

Simple moving average crossover



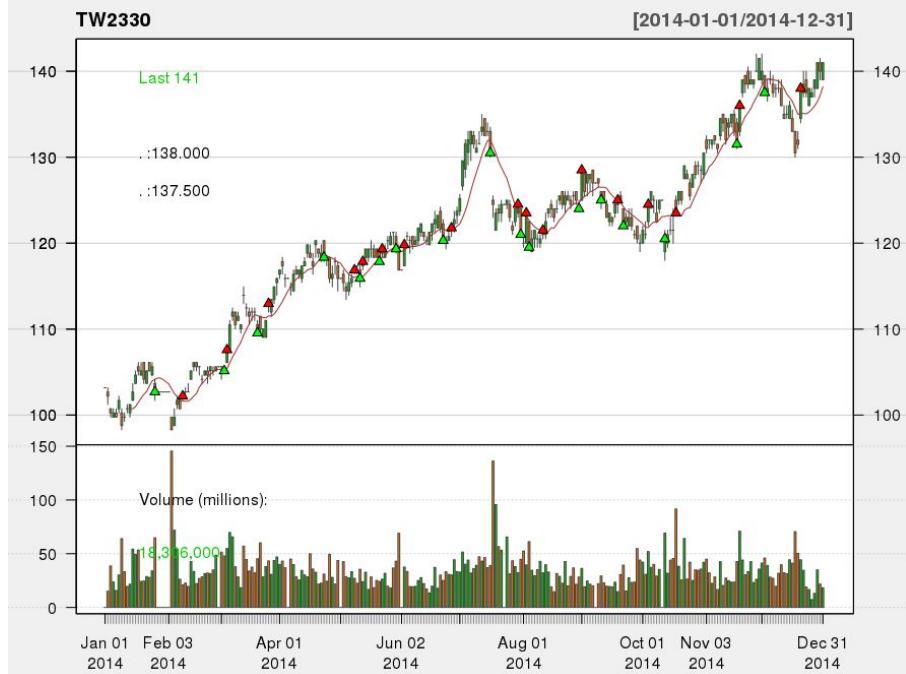
Momentum Strategy

- Taiwan Semiconductor Manufacturing Company (**2330.TW**)
- **BUY** when price > 20-day moving average
- **SELL** when price < 20-day moving average

Useful Tools about Data Access and Technical Analysis

Quantmod

```
library(quantmod)
TW2330 <- getSymbols('2330.TW',
                      src='yahoo',
                      index.class=c("POSIXct"),
                      from=startDate,
                      to=endDate,
                      adjust=TRUE,
                      auto.assign = FALSE)
chartSeries(TW2330, theme = 'white')
addSMA(10)
```



Useful Tools about Bar-by-bar Processing Backtest

Blotter

```
library(blotter)

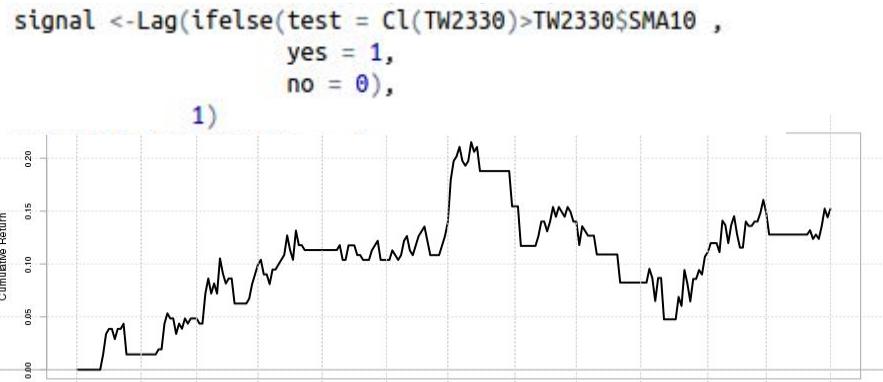
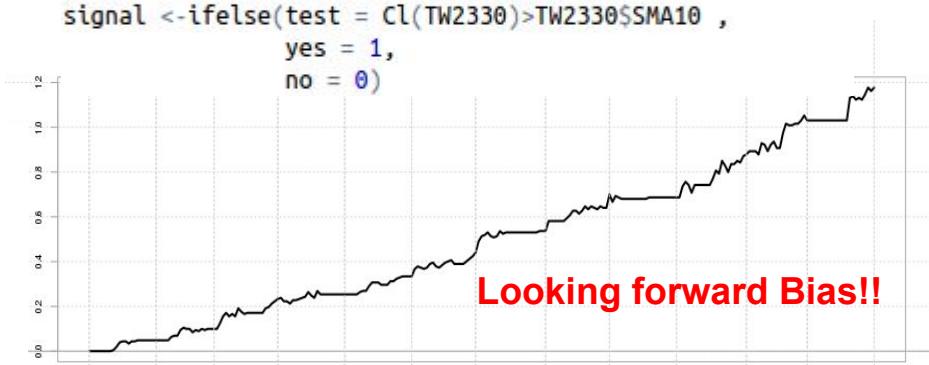
initPortf(name=portfolio.st, symbols = symbols, currency = 'TWD')
initAcct(name=account.st, portfolios=portfolio.st,
  initDate = startDate, initEq = initEq, currency = 'TWD')

# define signal
signal <- Lag(ifelse(test = Cl(TW2330) > TW2330$SMA10 , yes = 1, no = 0), 1)
signal[is.na(signal)] <- 0

# bar-by-bar processing
for(i in 21:length(signal)){
  currentDate <- time(signal)[i]
  equity <- getEndEq(Account = account.st, Date = currentDate)
  closePrice<-coredata(Cl(TW2330[currentDate]))

  position <- getPosQty(Portfolio = portfolio.st, Symbol='TW2330', Date=currentDate)

  if(position==0){
    if(signal[i] == 1){
      addTxn(portfolio.st, Symbol='TW2330', TxnDate=currentDate, TxnPrice=closePrice,
        TxnQty = 500, TxnFees=0, verbose=T)
    }else{
      if (signal[i]==0 ){
        addTxn(portfolio.st, Symbol='TW2330', TxnDate=currentDate, TxnPrice=closePrice,
          TxnQty = -500, TxnFees=0, verbose=T)
      }
    }
  updatePortf(Portfolio = portfolio.st, Dates = currentDate)
  updateAcct(name = account.st, Dates = currentDate)
  updateEndEq(Account = account.st, Dates = currentDate)
}
```



3. Quantstrat,

a powerful framework used to develop quantitative trading strategy.

Backtest tools in R and Python

R: Quantstrat

- US hedge funds use it
 - Peter Carl @ William Blair & Company
 - Brian Peterson @ Braverock Ventures
 - Joshua Ulrich @DV Trading
- Yet, limited document

Python: Quantopian

- Slow speed
- Not flexible syntax
- But community growing

Introduction to a Powerful Backtest Tool: Quantstrat

Indicator

- Quantitative value derived from market data
- Applied in a vectorized or streaming fashion
- Presumed to be able to be calculated in path-independent fashion
- No knowledge of current position or trades
- Examples: 'technical analysis indicators'

Signal

- Describe interaction between market data and Indicators
- Describe the possible desire for an action, but may not be actionable
- Applied in a vectorized or streaming fashion
- Used to inform rule decisions
- Examples: Crossovers, Thresholds, Multiples

Order Rule

- Evaluated in a path-dependent fashion
- Have available all market data prior to current Observation
- Are aware of current position at time of Evaluation
- Generate entry, exit, and risk management Orders
- May enter new orders or modify existing orders

Backtest in Quantstrat

add.indicator

R code

```
add.indicator(strategy.st,
              name = "SMA",
              arguments = list(x = quote(Cl(mktdata)), n=10),
              label="SMA10")
```

Key point

Define an indicator ‘SMA10’ as 10-day simple moving average by using function ‘TTR::SMA’

Backtest in Quantstrat

add.signal

R code

```
add.signal(strategy.st,
           name="sigCrossover",
           arguments = list(columns=c("Close", "SMA10"),
                            relationship=">"),
           label="Cl.gt.SMA")
```

Key point

Generate a signal ‘Cl.gt.SMA’ when close price is greater than 10-day simple moving average

Backtest in Quantstrat

add.rule

R code

```
# 1. Buy Rule: buy when the price crosses above the SMA
add.rule(strategy.st,
          name='ruleSignal',
          arguments = list(sigcol="Cl.gt.SMA",
                           sigval=1,
                           orderqty=500,
                           ordertype='market',
                           orderside='long',
                           pricemethod='market'),
          type='enter',
          path.dep=TRUE)
```

Key point

Define an entry rule ‘enter’ when there’s ‘Cl.gt.SMA’ signal, and put an order that longs 500 quantity in market price.

Backtest in Quantstrat

Let's Demo!

```

strategy.st <- 'SingleMA'
strategy(strategy.st, store=TRUE)

add.indicator(strategy.st, name = "SMA",
              arguments = list(x = quote(Cl(mktdata)),
                               n=10),
              label="SMA10")

# Entry Signal
add.signal(strategy.st, name="sigCrossover",
           arguments = list(columns=c("Close","SMA10"),
                            relationship=">"),
           label="Cl.gt.SMA")

# Exit Signal
add.signal(strategy.st, name="sigCrossover",
           arguments = list(columns=c("Close","SMA10"),
                            relationship="<"),
           label="Cl.lt.SMA")

# Buy Rule
add.rule(strategy.st, name='ruleSignal',
         arguments = list(sigcol="Cl.gt.SMA",
                          sigval=1,
                          orderqty=500,
                          ordertype='market',
                          orderside='long',
                          pricemethod='market'),
         type='enter', path.dep=TRUE)

# Sell Rule
add.rule(strategy.st, name='ruleSignal',
         arguments = list(sigcol="Cl.lt.SMA",
                          sigval=1,
                          orderqty='all',
                          ordertype='market',
                          orderside='long',
                          pricemethod='market'),
         type='exit', path.dep=TRUE)

# import strategy
source('class1/quantstrat/single_SMA_strategy.R')
applyStrategy(strategy = strategy.st, portfolios = portfolio.st)

```

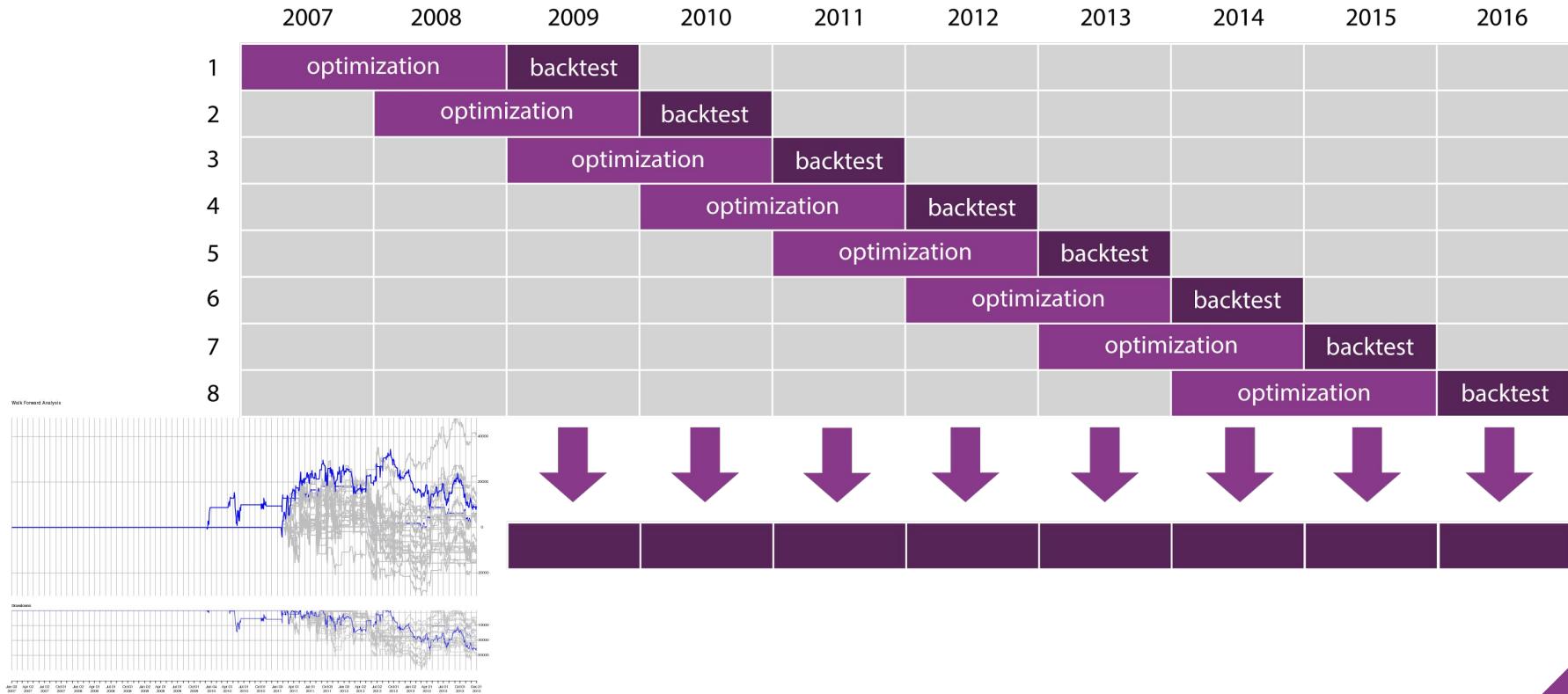
Results: Order Book

Untitled1* backtest_by_quantstrat.R * ob\$SingleMA\$TW2330 *

Filter

	Order.Qty	Order.Price	Order.Type	Order.Side	Order.Threshold	Order.Status	Order.StatusTime	Prefer	Order.Set	Txn.Fees	Rule
!014-02-11 00:00:00	500	102.203365604396	market	long	NA	closed	2014-02-12 00:00:00		NA	0	ruleSignal.rule
!014-03-04 00:00:00	all	105.137433516484	market	long	NA	closed	2014-03-05 00:00:00		NA	0	ruleSignal.rule
!014-03-05 00:00:00	500	107.58249010989	market	long	NA	closed	2014-03-06 00:00:00		NA	0	ruleSignal.rule
!014-03-20 00:00:00	all	109.538535384615	market	long	NA	closed	2014-03-21 00:00:00		NA	0	ruleSignal.rule
!014-03-26 00:00:00	500	112.961614615385	market	long	NA	closed	2014-03-27 00:00:00		NA	0	ruleSignal.rule
!014-04-23 00:00:00	all	118.340739120879	market	long	NA	closed	2014-04-24 00:00:00		NA	0	ruleSignal.rule
!014-05-08 00:00:00	500	116.873705164835	market	long	NA	closed	2014-05-09 00:00:00		NA	0	ruleSignal.rule
!014-05-12 00:00:00	all	115.895682527473	market	long	NA	closed	2014-05-13 00:00:00		NA	0	ruleSignal.rule
!014-05-13 00:00:00	500	117.851727802198	market	long	NA	closed	2014-05-14 00:00:00		NA	0	ruleSignal.rule
!014-05-21 00:00:00	all	117.851727802198	market	long	NA	closed	2014-05-22 00:00:00		NA	0	ruleSignal.rule
!014-05-22 00:00:00	500	119.318761758242	market	long	NA	closed	2014-05-23 00:00:00		NA	0	ruleSignal.rule
!014-05-29 00:00:00	all	119.318761758242	market	long	NA	closed	2014-05-30 00:00:00		NA	0	ruleSignal.rule
!014-06-03 00:00:00	500	119.807773076923	market	long	NA	closed	2014-06-04 00:00:00		NA	0	ruleSignal.rule
!014-06-23 00:00:00	all	120.296784395604	market	long	NA	closed	2014-06-24 00:00:00		NA	0	ruleSignal.rule
!014-06-26 00:00:00	500	121.763818351648	market	long	NA	closed	2014-06-27 00:00:00		NA	0	ruleSignal.rule
!014-07-16 00:00:00	all	130.5	market	long	NA	closed	2014-07-17 00:00:00		NA	0	ruleSignal.rule
!014-07-30 00:00:00	500	124.5	market	long	NA	closed	2014-07-31 00:00:00		NA	0	ruleSignal.rule
!014-07-31 00:00:00	all	121	market	long	NA	closed	2014-08-01 00:00:00		NA	0	ruleSignal.rule
!014-08-04 00:00:00	500	123.5	market	long	NA	closed	2014-08-05 00:00:00		NA	0	ruleSignal.rule
!014-08-05 00:00:00	all	119.5	market	long	NA	closed	2014-08-06 00:00:00		NA	0	ruleSignal.rule

More Features: Walk Forward Analysis



More Features: Advanced Risk Management



Optimization

Order management

Risk Management

Parameter optimization

Fixed Dollar Max Position

Stoploss
Trailing loss

4. Challenges

about data manipulation and ML, and its potential solutions in Microsoft R Server, Azure ML Studio. And others.

100+ GB

That's total size of your tick data

14 GB

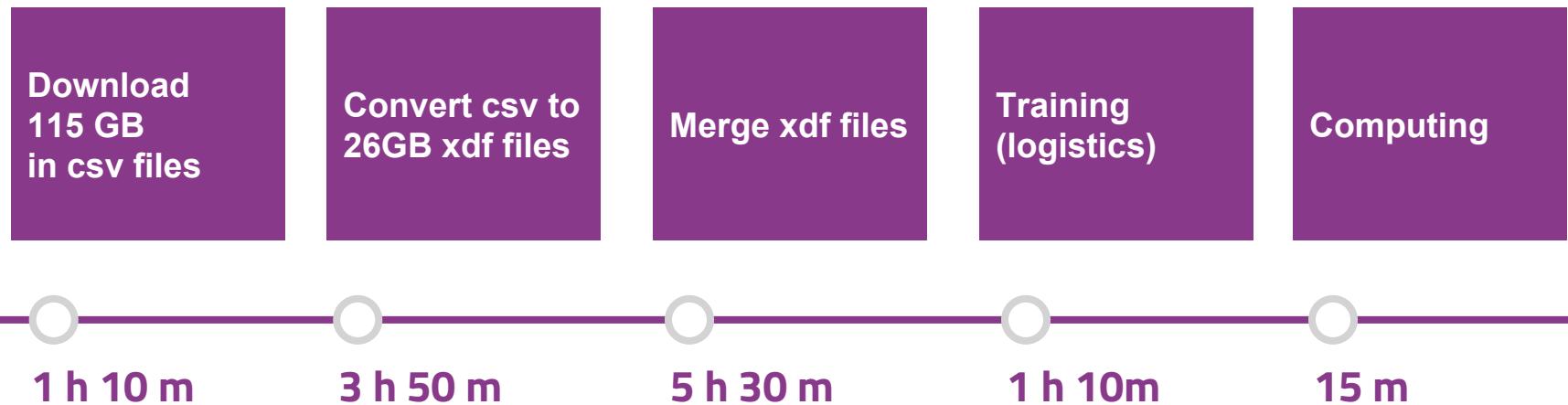
And only limited memory in your machine

You need a IN-DISK SOUTION, like **xdf in MRS**

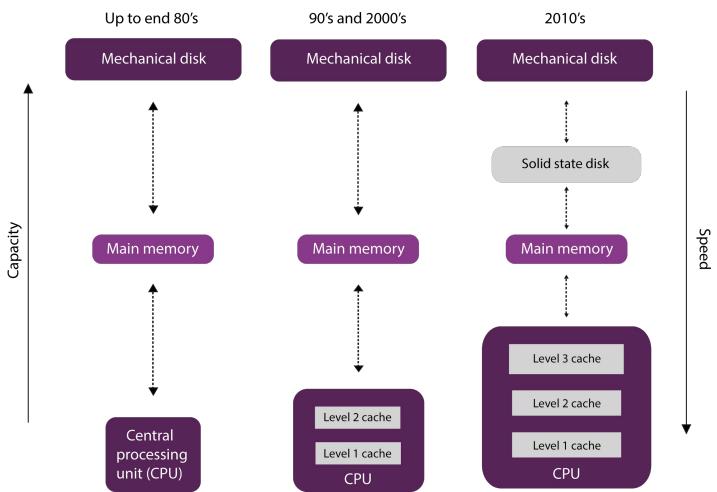
In-disk Solution

xdf in MRS

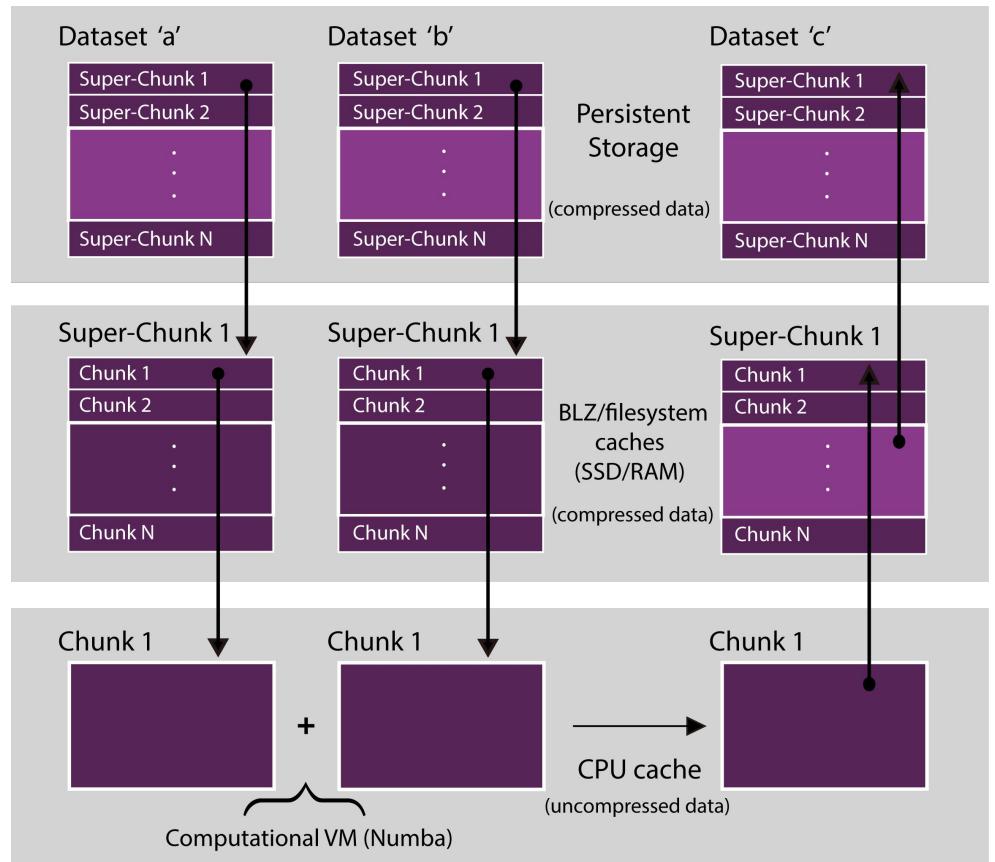
Given 8 cores and 14 GB memory



How does xdf works



Computing ' $C = a + b$ ' with MRS Xdf

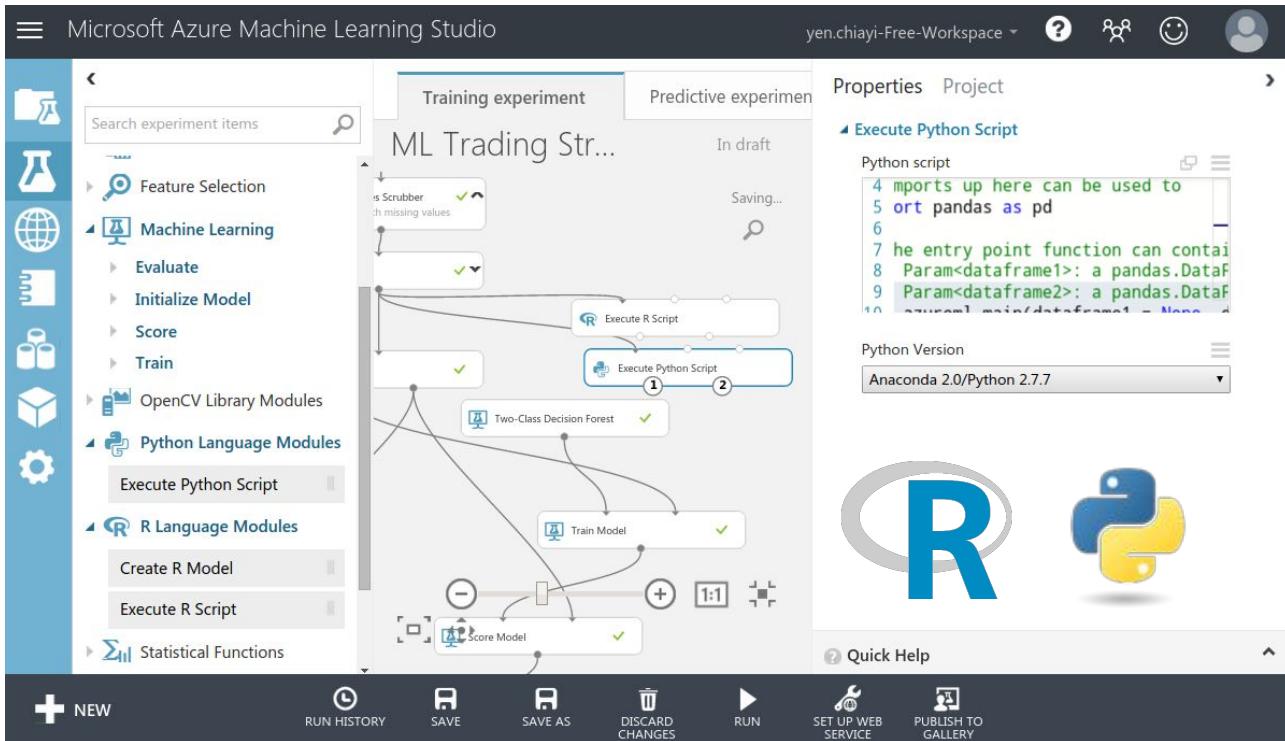


Azure ML Studio

“ Make machine learning accessible to every enterprise, data scientist, developer, information worker, consumer, and device anywhere in the world.

-- *Vision of Azure ML Studio (2014)*

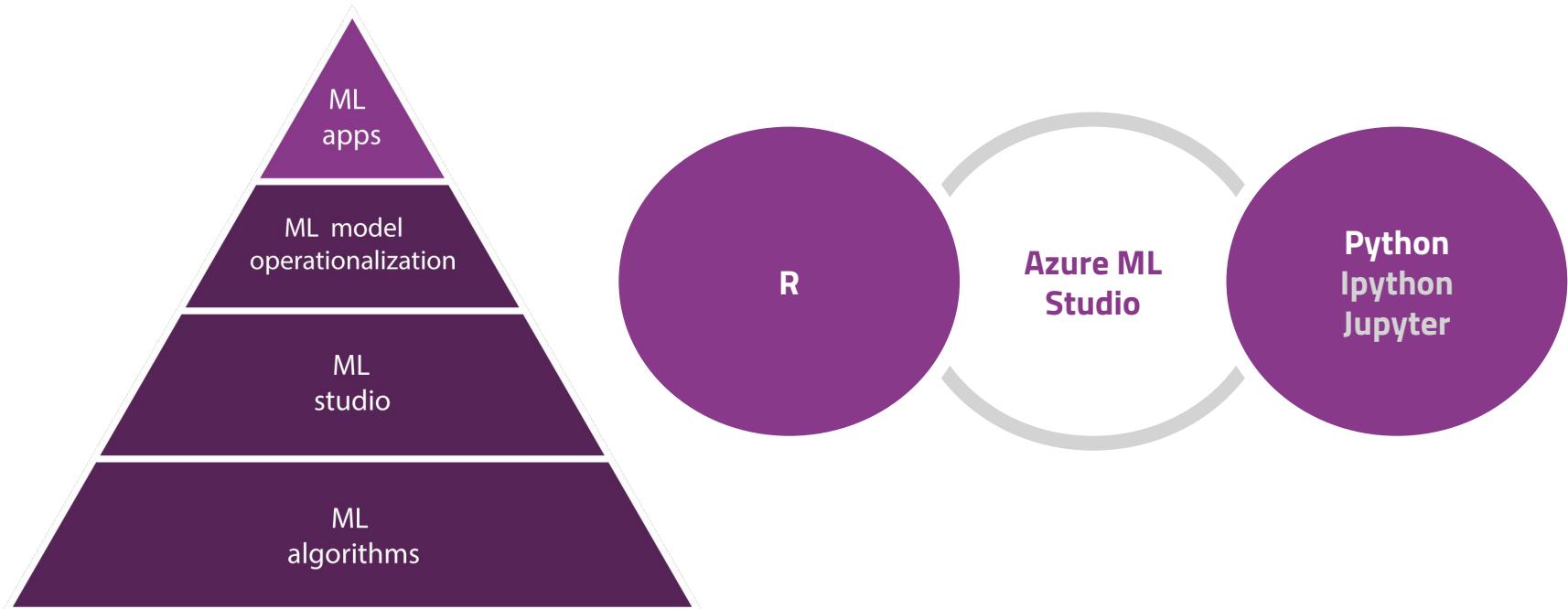
What's special for Azure ML Studio



The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left, there's a sidebar with various icons for data management and machine learning tasks like Feature Selection, Machine Learning, and Python Language Modules. The main area displays a 'Training experiment' titled 'ML Trading Str...'. The flowchart includes steps such as 'Scrubber', 'Execute R Script', 'Execute Python Script' (containing code snippets), 'Two-Class Decision Forest', 'Train Model', and 'Score Model'. A large 'Properties' panel on the right shows the 'Execute Python Script' step with its Python script content and Python Version set to 'Anaconda 2.0/Python 2.7.7'. Below the interface are two large logos: a grey 'R' and a blue and yellow Python logo.

- Web based UI
- Share to any other ML workspace
- Drag & Drop visual design
- Many ML Algorithms
- R/Python scripts
- API Deploy rapidly

Azure ML Studio is able to easily implement R and Python



Integrate with Quanstrat by API from Azure ML Studio

```
get.myIndicator <- function(mktdata){  
  data <- myTTR(mktdata) %>% na.omit()  
  result <- as.xts(api.predict(data), order.by=index(data))  
  colnames(result) <- c('myIndicator')  
  result <- cbind(mktdata, result)$myIndicator  
  result  
}  
  
add.indicator(strategy.st,  
              name = "get.myIndicator",  
              arguments = list(mktdata=quote(mktdata)),  
              label = 'myIndicator')
```

- API from Azure ML Studio as an indicator
- Integrate with quantstrat

Thank you for your listening!



**Look forward to
your visit to
R-Ladies Taipeii!**

- Meetup
[https://www.meetup.com/
R-Ladies-Taipei](https://www.meetup.com/R-Ladies-Taipei)
- FB pages
[https://www.facebook.co
m/groups/twrladies/](https://www.facebook.com/groups/twrladies/)