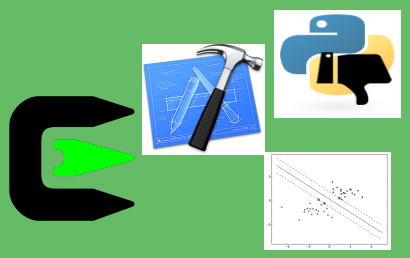
CISC 5352 FINANCIAL PROGRAMMING AND DATA ANALYTICS LECTURE NOTE (9)



Henry Han Ph.D.
Department of Computer and Information Science
Fordham University, New York NY 10023

Last class review

① SVM speedup (to handle scaling problem)
  ① Explicit map to approximate kernel via sampling and stochastic gradient descent (SGD) learning

② Cross-validation
  ① Hold-out, k-fold, LOOCV
  ② Independent test test

③ Gradient-boost and Random forest learning

④ Principal component analysis (PCA)

Q1: what is the major difference between Gradient boost and Random forest?

Q1: what is the major difference between Gradient boost and Random forest?

Both are ensemble learning, but they have different approaches to get final prediction functions

Q1: what is the major difference between Gradient boost and Random forest?

Both are ensemble learning, but they have different approaches to get final prediction functions

GB: optimize the prediction functions of weak learners via gradient learning (along the gradient descent direction of its loss function) to obtain the final prediction function

Q1: what is the major difference between Gradient boost and Random forest?

Both are ensemble learning, but they have different approaches to get final prediction functions

GB: optimize the prediction functions of weak learners via gradient learning (along the gradient descent direction of its loss function) to obtain the final prediction function

RF: the final decision function is just the average of prediction functions of weak learners (forests).

Q1: what is the major difference between Gradient boost and Random forest?

Both are ensemble learning, but they have different approaches to get final prediction functions

GB: optimize the prediction functions of weak learners via gradient learning (along the gradient descent direction of its loss function) to obtain the final prediction function

RF: the final decision function is just the average of prediction functions of weak learners (forests).

GB is more complicate than RF technically

Q2: What makes RF different from decision trees?

Q2: What makes RF different from decision trees?

① RF is a more generalized decision trees (forest)

② It builds many decision trees by using feature bagging techniques to avoid overfitting.

③ feature bagging: randomly selects a subset of input variables under boostrap instead of all variables to build trees in training.

④ Since not full information used in training for each weak learner, it enhances the final prediction funtion's generalization capbability

Q3: Summarize PCA in one sentence?

Q3: Summarize PCA in one sentence?

PCA gives you a reduced-data set of the original data in a new coordinate system by keeping most of data variance information

Download R studio to walk through PCA
https://www.rstudio.com/

R Studio: very nice platform for R programming

```
# Import packages
import numpy as np
from sklearn.decomposition import PCA

X = np.array([[-100, -1], [-200, -1], [-300, -2], [1, 100], [21, -1], [8.3,
9.92]])

print("\n Input data")
print(X)

## set PCA object
pca = PCA(n_components=2)
## conduct PCA
pca.fit(X)

print("\n explained variance in each PC \n")
print(str(pca.explained_variance_)+"\n")

print("explained variance ratios for all PCs\n")
print(str(pca.explained_variance_ratio_) + "\n")

print("PC components\n")
print(pca.components_)
```

High-frequency trading: big data trading

High-frequency real-time trading: a transaction can be done even in a fractions of second (milliseconds)!

High-frequency trading: big data trading

① High-frequency real-time trading: a transaction can be done even in a fractions of second (milliseconds)!

② That is multiple transactions can occur in same second!

High-frequency trading: big data trading

① High-frequency real-time trading: a transaction can be done even in a fractions of second (milliseconds)!

② That is multiple transactions can occur in same second!

③ Different stocks have "different trading frequencies" (trading is actually nonsynchronous)

High-frequency trading: big data trading

① High-frequency real-time trading: a transaction can be done even in a fractions of second (milliseconds)!

② That is multiple transactions can occur in same second!

③ Different stocks have "different trading frequencies" (trading is actually nonsynchronous)

④ Trading happens in an unequally time intervals

High-frequency trading: big data trading

① High-frequency real-time trading: a transaction can be done even in a fractions of second (milliseconds)!

② That is multiple transactions can occur in same second!

③ Different stocks have "different trading frequencies" (trading is actually nonsynchronous)

④ Trading happens in an unequally time intervals

⑤ Demonstrate periodic patterns or diurnal patterns somewhat

diurnal patterns demonstrated by IBM stocks trading volumes in NYSE (25 trading days: Monday 28 February and Friday 31 March 2000) Ito's work 2013

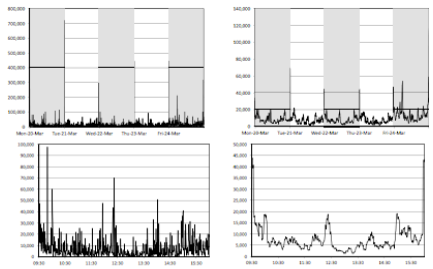


Figure 1 – IBM30s (left column) and the same series smoothed by the simple moving average of nearest 20 observations (right column). Time on the x-axis. Top panel: Monday 20 - Friday 24 March 2000. Each day covers market opening hours between 9.30am-4pm (in the New York local time). Bottom panel: Wednesday 22 March 2000, covering 9.30am-4pm (in the New York local time).

Normal trading hours: 9:30 am -4:00 pm EST
Johnson & Johnson Stock trading data on 2010/10/04 snapshot

date	hour	minute	second	price	volume
20101004	6	25	15	61.7500	100
20101004	8	33	19	61.5600	100
20101004	8	41	09	61.5600	100
20101004	8	48	50	61.6000	100
20101004	8	48	55	61.6000	100
20101004	8	49	04	61.6000	100
20101004	9	00	09	61.6000	200
20101004	9	00	10	61.6000	200
20101004	9	11	13	61.6000	100
20101004	9	11	13	61.6000	500
20101004	9	11	13	61.6000	1000
20101004	9	17	29	61.7000	100
20101004	9	20	40	61.7100	100
20101004	9	26	55	61.7100	200
20101004	9	28	45	61.6000	200
20101004	9	29	30	61.6000	339
20101004	9	29	45	61.5600	300
20101004	9	29	45	61.5500	100
20101004	9	30	00	61.5400	281
20101004	9	30	00	61.5400	281
20101004	9	30	01	61.6200	100
20101004	9	30	01	61.6200	100
20101004	9	30	03	61.5300	100
20101004	9	30	03	61.5300	100
20101004	9	30	03	61.5300	300
20101004	9	30	03	61.5100	155492
20101004	9	30	03	61.6000	2000
20101004	9	30	03	61.5300	100

Johnson & Johnson (JNJ) Stock trading data on 2010/10/15 snapshot

date	hour	minute	second	price	volume
20101015	16	00	00	63.5700	100
20101015	16	00	00	63.5500	100
20101015	16	00	00	63.5500	100
20101015	16	00	03	63.5550	100
20101015	16	00	04	63.5550	100
20101015	16	00	06	63.5500	105
20101015	16	00	16	63.5700	526533
20101015	16	01	29	63.5582	402
20101015	16	06	21	63.5700	200
20101015	16	06	57	63.5700	14674
20101015	16	07	23	63.5800	200
20101015	16	07	23	63.5700	200
20101015	16	07	27	63.5700	929
20101015	16	07	27	63.5900	929
20101015	16	08	09	63.5700	3330
20101015	16	08	21	63.5700	917
20101015	16	10	40	63.5587	300
20101015	16	12	27	63.5587	4500
20101015	16	12	30	63.5700	637
20101015	16	13	56	63.5649	167
20101015	16	18	59	63.5700	21152
20101015	16	25	20	63.5700	266
20101015	17	33	15	63.5800	100
20101015	17	38	34	63.4700	200
20101015	17	38	34	63.4500	200
20101015	18	14	36	63.5700	1100
20101015	18	14	38	63.5700	926
20101015	18	14	41	63.5700	21028

there are total 419,565 transactions collected from Oct 04-Oct 15, 2010 (10 transaction days)

General data analytics mainly focuses on the normal trading window:**9:30 am -4:00 pm EST**

there are total 418,855 transactions in the normal trading window

Q4: How about price change in these transactions (normal trading window)?

Q4: How about price change in these transactions (normal trading window)?

1. About 73% of JNJ transactions were without price change!
2. About 26% of the transactions result in a price change that is <= 1 cent!
3. The empirical distribution (aka histogram) of price changes is symmetric w.r.t. 0
4. This is a special one. Can we generalize it to other data?
5. You are going to verify it in your homework!

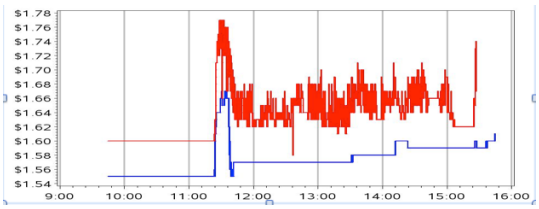
"Most time" prices are fixed or with low standard deviations

However, price peaks (up/down) are essential time for HFT trading

Can we predict these price peaks?

- They can be called jumps, which can also be extended to translation volumes

Can we predict these price peaks?



Aehr Test Systems (AEHR) stock on 4/11/2007 (Credited to J Hasbrouck, NYU)
http://www.aehr.com/: a provider of systems for burn-in and test of memory and logic integrated circuits

Can we predict these price peaks?

It is the state-of-art topic in high-frequency trading yet no good answers. Maybe no good answers forever.

The current research employs wavelet analysis, Fourier analysis, or mining social networks (mining twitter data) to predict possible 'news' that may trigger a coming price peaks!

"Research" leads to tons of money!

In addition to being a major trading approach, it can contribute to 'answering' the following questions:

① Who provides the market liquidity mostly?
  ① Liquidity: ability to trade a security (or other products) quickly.

② The dynamics of bid and ask quotes of a specified stock

③ The dynamics of market microstructure

④ Accurate security pricing and price discovery

⑤ What kinds of big data analytics algorithms for good for high-frequency trading data?

HFT data has different formats

But, they generally consist of two parts:

Trading data

Quote data (BIT and Offer data)

DATE	Hour	Minute	Second	BID	OFR	BIDSIZ	OFRSIZ
20100104	4	07	00	56.97	57.89	5	5
20100104	4	07	00	56.97	57.89	5	5
20100104	4	07	01	56.98	57.89	5	5
20100104	4	07	37	56.97	57.88	5	5
20100104	4	07	48	56.97	0.00	5	0
20100104	4	07	48	56.97	57.88	5	5
20100104	4	08	08	56.96	57.87	5	5
20100104	4	08	44	56.96	0.00	5	0
20100104	4	08	44	56.96	57.88	5	5
20100104	4	08	49	56.97	57.88	5	5
20100104	4	08	49	56.97	57.88	5	5
20100104	4	09	00	56.96	57.88	5	5
20100104	4	09	16	56.96	57.87	5	5
20100104	4	09	26	56.95	57.86	5	5
20100104	4	09	30	56.96	0.00	5	0
20100104	4	09	30	56.96	0.00	5	0
20100104	4	09	30	56.96	57.88	5	5
20100104	4	09	30	56.96	57.87	5	5
20100104	4	10	02	56.95	57.86	5	5
20100104	4	11	48	56.96	57.86	5	5
20100104	4	12	14	56.97	57.86	5	5
20100104	4	12	14	56.97	57.86	5	5
20100104	4	13	00	56.96	57.86	5	5
20100104	4	13	21	56.97	57.86	5	5
20100104	4	13	21	56.97	0.00	5	0
20100104	4	13	21	56.97	57.87	5	5
20100104	4	13	33	56.96	57.86	5	5
20100104	4	13	33	56.97	57.86	5	5

Quote data of caterpillar stock on 01/04 2010

Date	hour	minute	second	price	size
20100104	9	30	0	57.65	3910
20100104	9	30	0	57.65	3910
20100104	9	30	0	57.7	400
20100104	9	30	0	57.68	100
20100104	9	30	0	57.69	300
20100104	9	30	1	57.65	462
20100104	9	30	1	57.65	100
20100104	9	30	1	57.65	100
20100104	9	30	1	57.65	100
20100104	9	30	1	57.7	100
20100104	9	30	1	57.7	100
20100104	9	30	1	57.72	500
20100104	9	30	1	57.72	100
20100104	9	30	2	57.73	100
20100104	9	30	3	57.73	300
20100104	9	30	3	57.72	100
20100104	9	30	4	57.72	300
20100104	9	30	5	57.57	100
20100104	9	30	5	57.57	500
20100104	9	30	5	57.56	300
20100104	9	30	10	57.56	100
20100104	9	30	11	57.85	100
20100104	9	30	11	57.85	200
20100104	9	30	11	57.56	100
20100104	9	30	14	57.8	100
20100104	9	30	14	57.82	100
20100104	9	30	23	57.76	100
20100104	9	30	35	57.55	100
20100104	9	30	35	57.56	100
20100104	9	30	35	57.54	100
20100104	9	30	35	57.75	100
20100104	9	30	35	57.76	300
20100104	9	30	35	57.73	100
20100104	9	30	35	57.76	200
20100104	9	30	35	57.77	200
20100104	9	30	35	57.77	100

Trade data of caterpillar stock on 01/04 2010

HFT data database: TAQ

The trades and quotes (TAQ) of NYSE (new york stock exchange)

It includes transactions of NYSE, AMEX, NASDAQ and regional exchanges

It provides sample and commercial data

http://www.nyxdata.com/Data-Products/Daily-TAQ#13068

ftp://ftp.nyxdata.com/Historical%20Data%20Samples/Daily%20TAQ/

How to compute volatility of these HFT data?

① Old models: ARCH, GARCH and their variants assume data are normal trading data!

② A few models have been proposed. But no a widely accepted one yet. People are looking for better models.

③ Traditional volatility is an annualized volatility based on daily observations.

④ HFT's observation is less-than-second-wise! Annualized volatility must be extended to daily or transaction unit-based volatility

⑤ HFT 's volatility is a short-term volatility

HFT's volatility should be a data-driven model

Traditional volatility is an annualized volatility: we need to multiply daily volatility by $\sqrt{\tau}$ (aka sqrt(252)

① Given n+1 number of observations (e.g., daily observation): 0, 1,2,...,n;

② We use S_i to represent the stock price (close price) at the end of i^{th} interval: [i-1, i]

③ Let τ represent the length of time interval in years (e.g.,1/252: 252 total trading days annually), then the volatility can be estimated as

$$\hat{\sigma} = \frac{s}{\sqrt{\tau}}$$ a normalized standard deviation w.r.t time

s is the estimation of standard deviation of stock log return $u_i = \ln(\frac{S_i}{S_{i-1}})$,

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} u_i^2 - \frac{1}{n(n-1)}(\sum_{i=1}^{n} u_i)^2}$$

The standard error of such estimation is $\hat{\sigma}/\sqrt{2n}$

Realized volatility: a model proposed in 2001

(not a good approach for me because it is based on some wrong assumption!)

- Monthly volatility

  u_t^m be the monthly log return of a stock at month t

  Suppose there are n trading days in month t and daily log returns is u_{ti}

  $u_t^m = \sum_{i=1}^{n} u_{ti}$

  The estimated monthly volatility is $\sigma_m^* = (\frac{n}{n-1}\sum_{i=1}^{n}(u_{ti} - \bar{u}_t)^2)^{1/2}$

  $\bar{u}_t$: sample mean

- Realized volatility

  Let u_t be the daily log return of a stock, suppose that there are n equally spaced log returns available such that $u_t = \sum_{i=1}^{n} r_{ti}$

  $RV_t = \sum_{i=1}^{n} r_{ti}^2$

Other core questions about HFT?

1. Does it increase market liquidity?
2. Does it increase a security's volatility (e.g. yearly volatility?)
3. Does it help price discovery (pricing)
4. How to build a profitable model for HFT data? Which data mining/machine learning and statistical arbitrary algorithms should we use?

Does HFT decrease or increase stock price volatility?

$$VOLT = \beta_0 + \beta_1 HFT + \beta_2 sd\Delta ROE + \beta_3 sdSGR + \beta_4 DISP + \beta_5 LEV + \beta_6 AGE + \beta_7 INST + \beta_8 (1/P) + \beta_9 SIZE + \beta_{10} BM + \beta_{11} RET\_12 + FIRM\_fixed\_effects + Time\_fixed\_effects + e_i$$

VOLT: Volatility

HFT: high-frequency trading

sd Δ ROE: earnings surprise volatility

sdSGR: sales growth volatility

DISP: analyst forecast dispersion

LEV: market leverage

AGE: firm age

INST: institutional holdings

SIZE: firm size

BM: book-to-market ratio

RET_12: past 12-month stock returns

1/P: inverse of stock price

Frank Zhang's first "Model"
(a simple regression model)

His conclusions

HFT increases volatility

HFT hinders price discovery

Does Algorithmic Trading Improve Liquidity?

① A well-cited paper published in journal of Finance.

② Still build a regression model to demonstrate HFT's impact on liquidity→ their conclusion is HFT increases liquidity

③ Data: daily panel of 1082 NYSE common stocks from December 2nd, 2002 through July 31, 2003

④ Method: 2SLS regression with Autoquote as the IV(instrumental variable) for algorism trading

All these conclusions are doubtful!

Why?

All these conclusions are doubtful!

Why?

Their models DID NOT use HFT data. Their conclusions are based very limited data fitted some regression models.

It's their model talk instead of data-talk (data-deriven approach)