Project 2 - xijie zhou

1 Exercise 1

1.1 A

The function code is below:

Listing 1: load_stock.m

```
function [dates, prices] = load_stock(filename)
%load_stock loads rectangular stock data and returns
%serial dates and prices.
%
%Input:
%   filename(string):Name of .csv file with stock data
%   with N proices per day and T total days

%Output:
%   dates(matrix):NxT matrix of special dates
%   prices(matrix):NxT matrix of price
    spy = readmatrix(filename);

    N = sum(spy(1,1) == spy(:,1));
    T = size(spy,1)/N;

    yyyy = floor(spy(:,1)./10^4);
    mm = floor((spy(:,1) - yyyy.*10^4)/10^2);
    dd = spy(:,1) - yyyy.*10^4 - mm.*10^2;

    HH = floor(spy(:,2)./10^2);
    MM = spy(:,2) - HH.*10^2;

    dates = datenum(yyyy,mm,dd,HH,MM,0);
    dates = reshape(dates,N,T);
    prices = log(reshape(spy(:,3),N,T));

end
```

1.2 B

N: (n+1)intervals in a day when we observe
T: the total number of days when we observe N and T are not the same for both of my stocks.Because I have the stocks with different frequencies, like 5-minutes and 5-seconds. Also the total observing days are not same.
N-code: "N = sum(spy(1,1) == spy(:,1));".
T-code: "T = size(spy,1)/N;".

1.3 C

·The stock market hours are usually from 9:30 am to 4:00 pm on weekdays except stock market holidays.
·I expect the value of N is 79.
·It differs from the value of N I computed in the previous question which is 78.
·I think the reason could be we don't have the data at the beginning of the day, so

the real observed data is less than the expected data.

1.4 D

The function code is below:

Listing 2: log_returns.m

```
function [dates_return, deltax] = log_returns(dates,
   prices)

deltax = diff(prices);%return

dates_return = dates(2:end,:);

end
```

1.5 E

I used two stocks which are DIS and PG respectively.

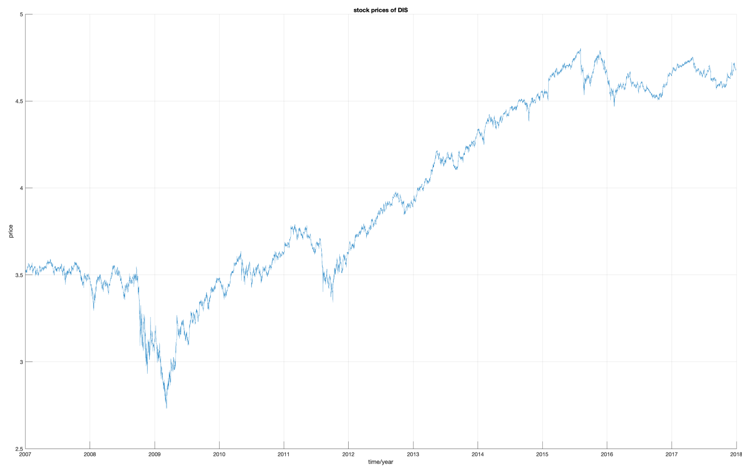

Figure 1: stock prices of DIS.

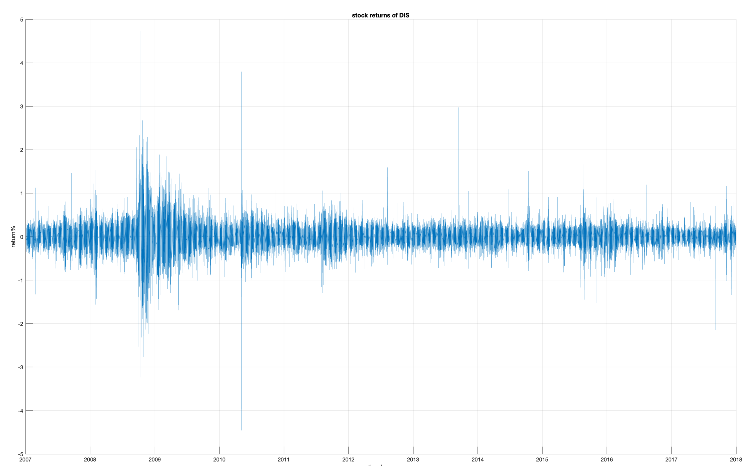

Figure 2: returns of DIS.

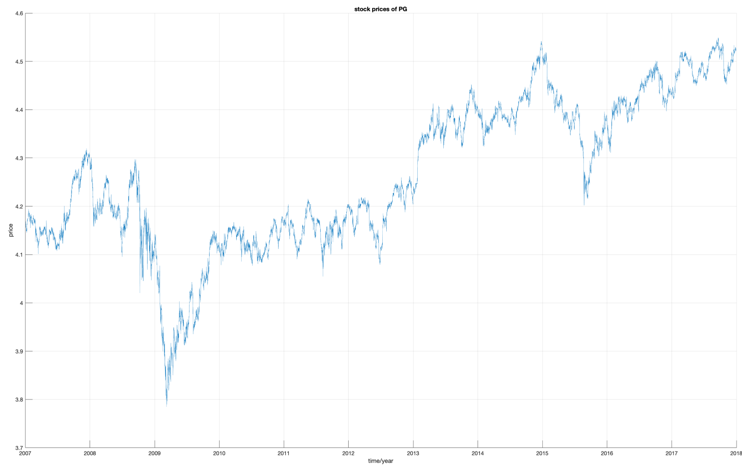

Figure 3: stock prices of PG.

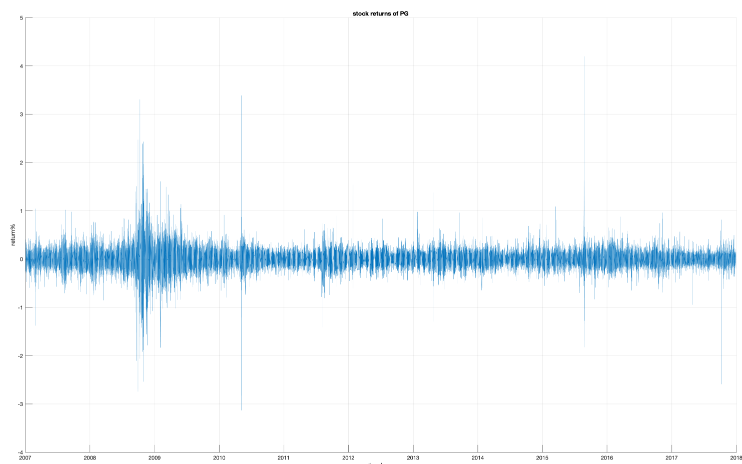

Figure 4: returns of PG.

I can see some outliers in the figures of stock prices and returns when they fluctuate drastically.

The figure code is below:

Listing 3: ex_1.m

```
[dates, prices] = load_stock('DIS.csv');

% deltax = diff(prices); %return
% dates_return = dates(2:end,:);

f = figure(1);
set(f,'units','normalized','outerposition',[0 0 1 1]);
plot(dates(:),prices(:))
datetick('x','yyyy');
title('stock prices');
box off; grid on;
xlabel('time/day');
ylabel('price');
print(f,'-dpng','-r200','figures/1E1A');
close(f);

[dates_return, deltax] = log_returns(dates, prices);

f = figure(2);
set(f,'units','normalized','outerposition',[0 0 1 1]);
plot(dates_return(:),100*deltax(:))
datetick('x','yyyy');
title('stock returns');
box off; grid on;
xlabel('time/day');
ylabel('return%');
print(f,'-dpng','-r200','figures/1E1B');
close(f);

[dates, prices] = load_stock('PG.csv');

f = figure(3);
set(f,'units','normalized','outerposition',[0 0 1 1]);
plot(dates(:),prices(:))
datetick('x','yyyy');
title('stock prices');
box off; grid on;
xlabel('time/day');
ylabel('price');
print(f,'-dpng','-r200','figures/1E2A');
```

```
close(f);

[dates_return, deltax] = log_returns(dates, prices);

f = figure(4);
set(f,'units','normalized','outerposition',[0 0 1 1]);
plot(dates_return(:),100*deltax(:))
datetick('x','yyyy');
title('stock returns');
box off; grid on;
xlabel('time/day');
ylabel('return%');
print(f,'-dpng','-r200','figures/1E2B');
close(f);
```

1.6 F

· A stock split is a decision by a company's board of directors to increase the number of shares that are outstanding by issuing more shares to current shareholders.

· The primary motive is to make shares seem more affordable to small investors even though the underlying value of the company has not changed. This has the practical effect of increasing liquidity in the stock.

· A stock split is usually done by companies that have seen their share price increase to levels that are either too high or are beyond the price levels of similar companies in their sector.

· After a split, the stock price will be reduced since the number of shares outstanding has increased. So if the stock prices decrease suddenly, it could be a stock split.

· The formula to calculate the new price per share is current stock price divided by the split ratio.

· I cannot make sure there are stock splits in my data, although there are some sharp failing prices.

· In various discussions it is often claimed that YahooFinance always processes the splits into their historical data, but GoogleFinance does not.

· It does not affect within day returns. Although the prices of the stock has fallen, the number of share held has increased, so the returns remain unchanged.

2 Exercise 2

2.1 A

The figure is below:

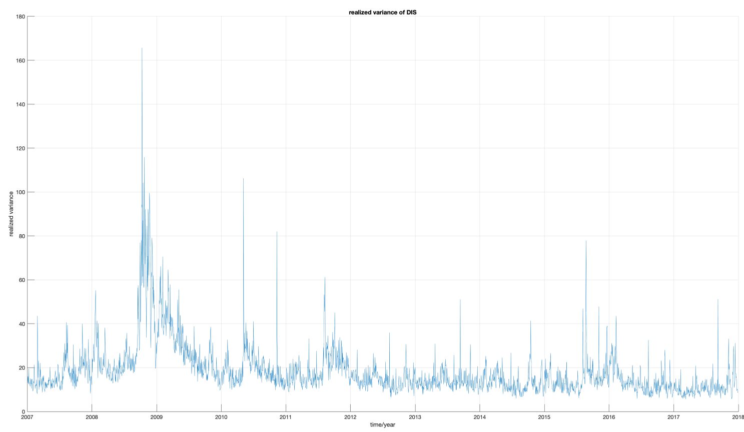

Figure 5: realized variance of DIS.

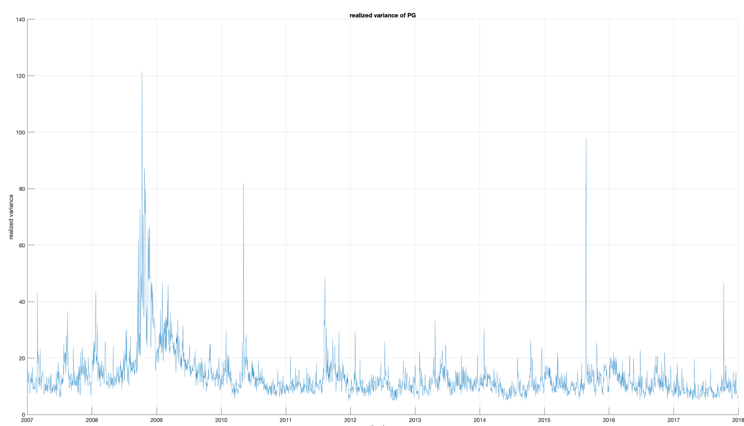

Figure 6: realized variance of PG.

Interpret: The volatilities of the figures about both stocks were very high. They reached their peaks before 2009 following an ups and downs in the rest of time. (It is worth noting that the overall trends of the two stocks in the same year are also very similar, although the scopes are different where the scope of DIS is larger.)

The function code is below:

Listing 4: realized_var.m

```
function RV = realized_var(deltax)

[n,T] = size(deltax);
RV_first = zeros(n,T);

for t = 1:T
    for i = 1:n
    RV_first(i,t) = abs(deltax(i,t)).*abs(deltax(i,t));
    end
end

RV = sum(RV_first);%Sum the entire matrix by column
end
```

2.2 B

The figure is below:

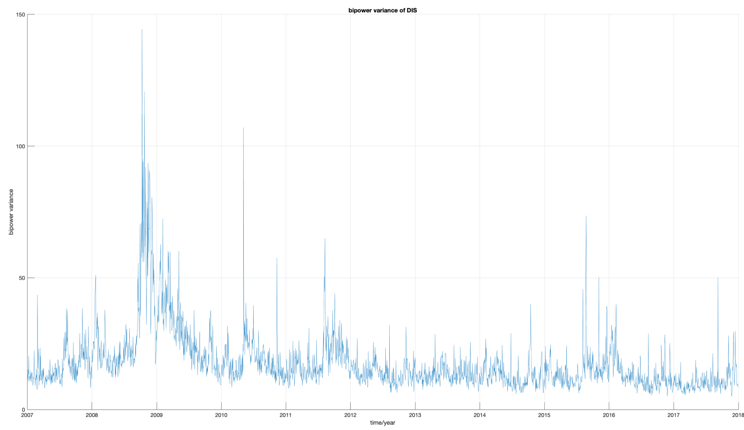

Figure 7: bipower variance of DIS.

The figure is below:

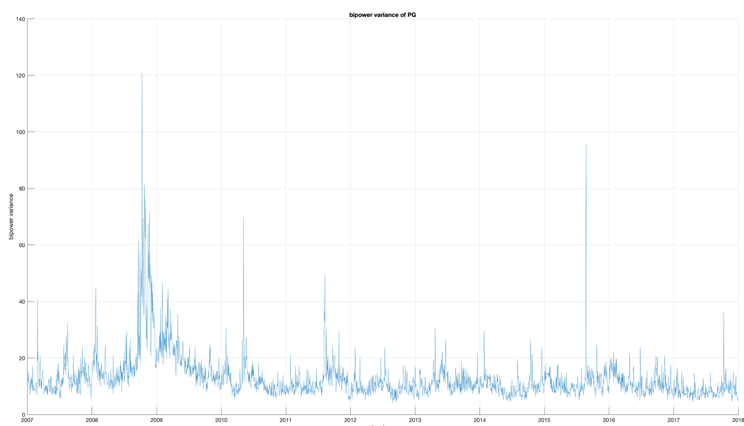

Figure 8: bipower variance of PG.

Interpret: The volatilities of the figures about both stocks were very high and reached their peaks before 2009.And they have the second peak around 2016. So I can see the figures are closely related to the real market. (It is worth noting that the overall trends of the two stocks in the same year are also very similar, although the scopes are different where the scope of DIS is larger.)

The function code is below:

Listing 5: bipower_var.m

```
function BV = bipower_var(deltax)
[n,T] = size(deltax);
BV_first = zeros(n,T);

for t = 1:T
    for i = 2:n
    BV_first(i,t) = abs(deltax(i,t)).*abs(deltax(i-1,t
        ));
    end
end

BV = (pi/2)*sum(BV_first);%Sum the entire matrix by
    column
end
```

2.3 C

The figure is below:

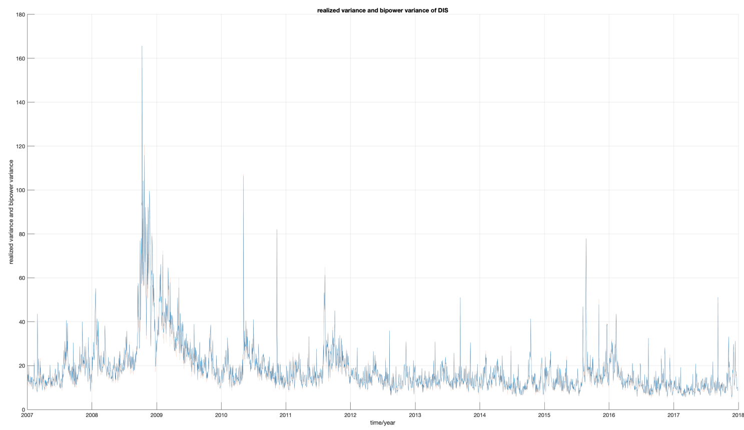

Figure 9: realized variance and bipower variance of DIS.

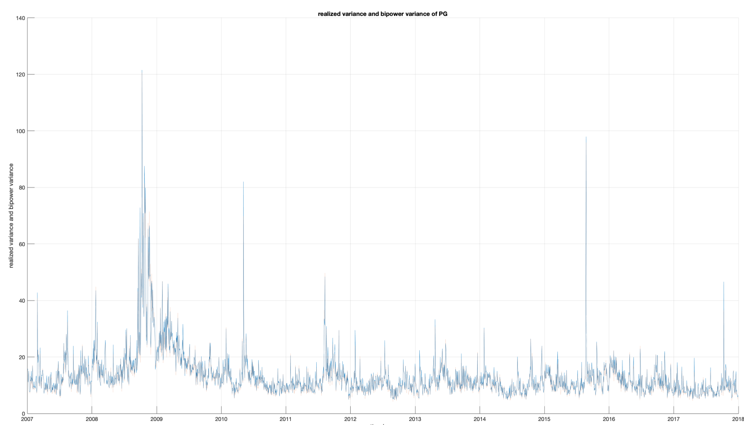

Figure 10: realized variance and bipower variance of PG.

Interpret: The similarities are the overall trend are similar and the scopes of the two variances in the same stock are similar.

The difference is the bipower variance is bigger than the realized variance in many cases, because it can be seen from the figure, dark lines cannot cover light lines..

2.4 D

DIS: the percent is 4.4859%
PG: the percent is 4.6905%
Yes, the value is close to the ones found by Huang and Tauchen which is around 5%.

The whole code for excercise 2 is below:

Listing 6: ex_2.m

```
[dates, prices] = load_stock('DIS.csv');

[dates_return,deltax] = log_returns(dates, prices);

RV1 = realized_var(deltax);

RV_last1 = 100*sqrt(RV1*252);

dates_RV_last1 = dates_return(1,:);%Extract the first
    line of the matrix


f = figure(5);
set(f,'units','normalized','outerposition',[0 0 1 1]);
plot(dates_RV_last1,RV_last1)
datetick('x','yyyy');
title('realized variance of DIS');
box off; grid on;
xlabel('time/year');
```

```
ylabel('realized variance');
print(f,'-dpng','-r200','figures/2A1');
close(f);

BV1 = bipower_var(deltax);
BV_last1 = 100*sqrt(BV1*252);
dates_BV_last1 = dates_RV_last1;

f = figure(6);
set(f,'units','normalized','outerposition',[0 0 1 1]);
plot(dates_BV_last1,BV_last1)
datetick('x','yyyy');
title('bipower variance of DIS');
box off; grid on;
xlabel('time/year');
ylabel('bipower variance');
print(f,'-dpng','-r200','figures/2B1');
close(f);

f = figure(7);
set(f,'units','normalized','outerposition',[0 0 1 1]);
plot(dates_RV_last1,RV_last1);
hold on
p2 = plot(dates_BV_last1,BV_last1);
p2.Color(4) = 0.25;
hold off
datetick('x','yyyy');
title('realized variance and bipower variance of DIS');
box off; grid on;
xlabel('time/year');
ylabel('realized variance and bipower variance');
print(f,'-dpng','-r200','figures/2C1');
close(f);

[n,T] = size(dates_BV_last1);
C1 = zeros(n,T);
for t = 1:T
    C1(t) = (max(RV_last1(t)-BV_last1(t),0))./RV_last1(t);
end

Ca1 = 100*mean(C1);
```

```
[dates, prices] = load_stock('PG.csv');

[dates_return,deltax] = log_returns(dates, prices);

RV2 = realized_var(deltax);

RV_last2 = 100*sqrt(RV2*252);

dates_RV_last2 = dates_return(1,:);%Extract the first
    line of the matrix"dats"

f = figure(8);
set(f,'units','normalized','outerposition',[0 0 1 1]);
plot(dates_RV_last2,RV_last2)
datetick('x','yyyy');
title('realized variance of PG');
box off; grid on;
xlabel('time/year');
ylabel('realized variance');
print(f,'-dpng','-r200','figures/2A2');
close(f);

BV2 = bipower_var(deltax);
BV_last2 = 100*sqrt(BV2*252);
dates_BV_last2 = dates_RV_last2;

f = figure(9);
set(f,'units','normalized','outerposition',[0 0 1 1]);
plot(dates_BV_last2,BV_last2)
datetick('x','yyyy');
title('bipower variance of PG');
box off; grid on;
xlabel('time/year');
ylabel('bipower variance');
print(f,'-dpng','-r200','figures/2B2');
close(f);

f = figure(10);
set(f,'units','normalized','outerposition',[0 0 1 1]);
plot(dates_RV_last2,RV_last2);
hold on
p2 = plot(dates_BV_last2,BV_last2);
p2.Color(4) = 0.25;
hold off
datetick('x','yyyy');
title('realized variance and bipower variance of PG');
```

```
box off; grid on;
xlabel('time/year');
ylabel('realized variance and bipower variance');
print(f,'-dpng','-r200','figures/2C2');
close(f);

[n,T] = size(dates_BV_last2);
C2 = zeros(n,T);
for t = 1:T
    C2(t) = (max(RV_last2(t)-BV_last2(t),0))./RV_last2
        (t);
end

Ca2 = 100*mean(C2);
```

3 Exercise 3

3.1 A

The realized variance is the sum of squared returns. And it is useful because it provides a relatively accurate measure of volatility.
The volatility signature plots are plots of average realized volatility against sampling interval k. The integer k represents multiples of the smallest sampling interval in the data; thus, if we have a series for which the smallest available sampling interval is one minute, for k=1 we construct aver-

age realized volatility using 1-minute returns, for k=2 we construct average realized volatility using 2-minute returns, and so forth.

The volatility signature plot is very useful, because we can research the realized variance in the different time intervals.

3.2 B

The function code is below:

Listing 7: load_stock2.m

```
function [dates, prices] = load_stock2(filename)
    spy = readmatrix(filename);

    N = sum(spy(1,1) == spy(:,1));
    T = size(spy,1)/N;

    yyyy = floor(spy(:,1)./10^4);
    mm = floor((spy(:,1) - yyyy.*10^4)/10^2);
    dd = spy(:,1) - yyyy.*10^4 - mm.*10^2;

    HH = floor(spy(:,2)./10^4);
    MM = floor((spy(:,2)-HH.*10^4)./10^2);
    SS = spy(:,2) - HH.*10^4 - MM.*10^2;

    dates = datenum(yyyy,mm,dd,HH,MM,SS);
    dates = reshape(dates,N,T);
    prices = log(reshape(spy(:,3),N,T));
end
```

N is 4621.

T is 252.

3.3 C

The figure is below:

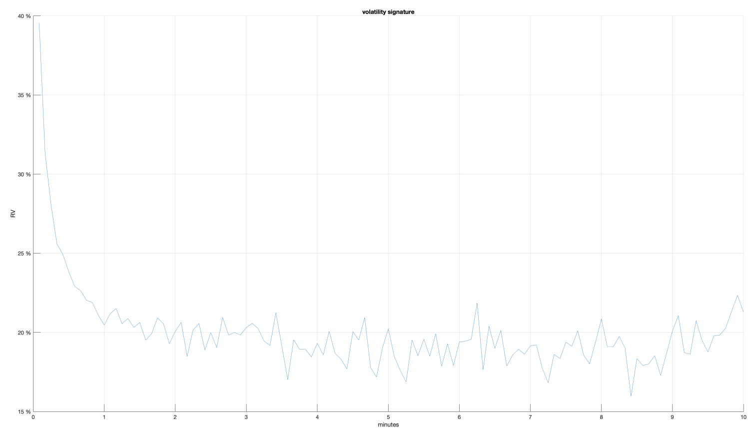

Figure 11: volatility signature of BAC2015 .

Interpret: As J (minutes) increases, RV is decreasing, and the larger J is, the greater the fluctuation of RV.

3.4 D

The figure is below:

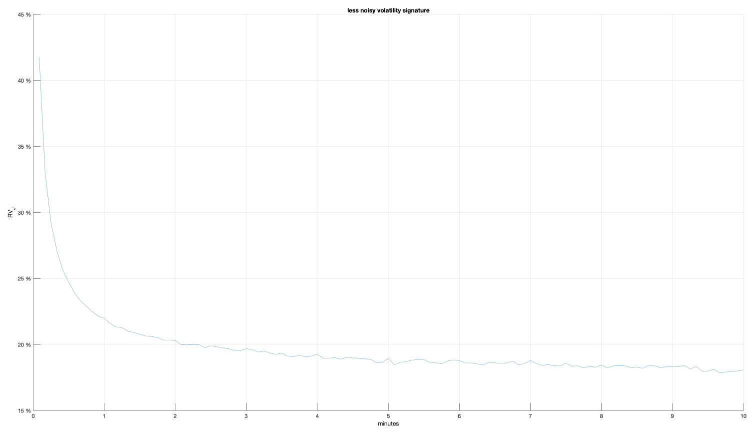

Figure 12: less noisy volatility signature of BAC2015.

Interpret: Compared with the image of the previous question, the similarity is that as J increases, RV decreases. The difference is that the curve is smooth and the fluctuations are small.

3.5 E

I think the above is wrong for the ultra-high sampling frequencies. Because when J is small, RV is rapidly reduced. In a short time, even the RVs represented by the adjacent Js will make a big difference which will not be equal to 0.

3.6 F

As can be seen from the figure "less noisy volatility signature of BAC2015", the RV of 3 to 8 minutes is reasonably flat.
The code is below:

Listing 8: ex_3.m

```
[dates, prices] = load_stock2('BAC-2015.csv');
Jmax = 120;
dates_day = dates(:,1);
prices_day = prices(:,1);
J = 1:1:Jmax;

RV = zeros(Jmax,1);
for i = 1:Jmax
    subprices_day = prices_day(1:i:end);
    deltax = diff(subprices_day);
    RV(i) = sqrt(sum(deltax.*deltax)*252)*100;
end

f = figure(11);
set(f,'units','normalized','outerposition',[0 0 1 1]);
plot(J/12,RV)
title('volatility signature');
box off; grid on;
xlabel('minutes');
ylabel('RV');
ytickformat('%g %%');
print(f,'-dpng','-r200','figures/3C');
close(f);

T =252;
RVt = zeros(Jmax,T);
RV_J = zeros(120,1);

for i = 1:Jmax
   subpricest = prices(1:i:end,:);
   deltax = diff(subpricest);
   RVt = sqrt(sum(deltax.*deltax)*252)*100;
   RV_J(i) = (1/T)*sum(RVt,2);
```

```
end


f = figure(12);
set(f,'units','normalized','outerposition',[0 0 1 1]);
plot(J/12,RV_J)
title('less noisy volatility signature');
box off; grid on;
xlabel('minutes');
ylabel('RV_J');
ytickformat('%g %%');
print(f,'-dpng','-r200','figures/3D');
close(f);
```