

FINAL EXAM

QIAN LOU

Individual Index VaR (Group A)

1. Describe your Index.

1) Show the first four moments mean / variance / skewness / kurtosis for the index.

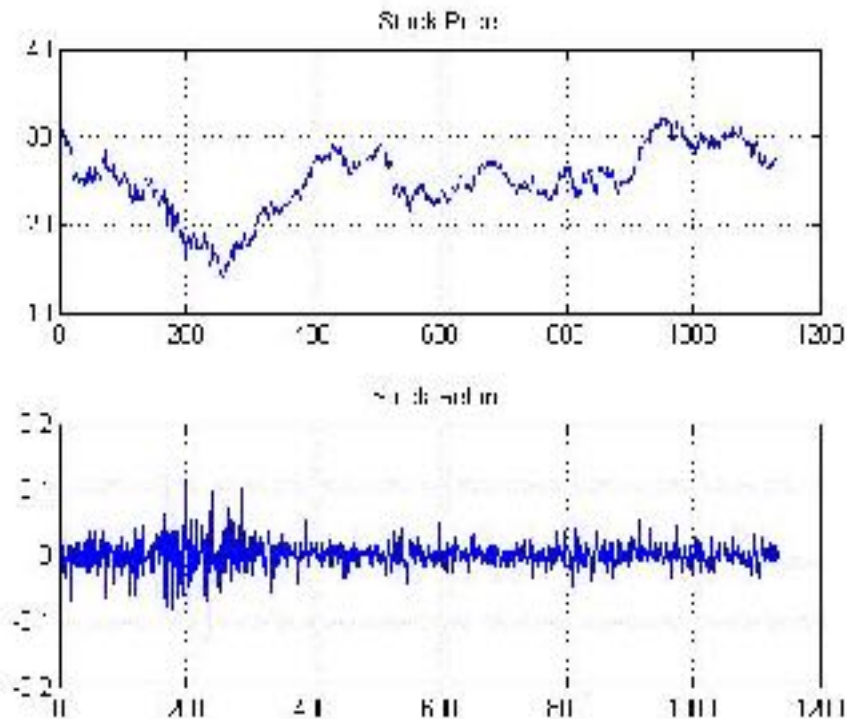
2) What kind of distributions do they have?

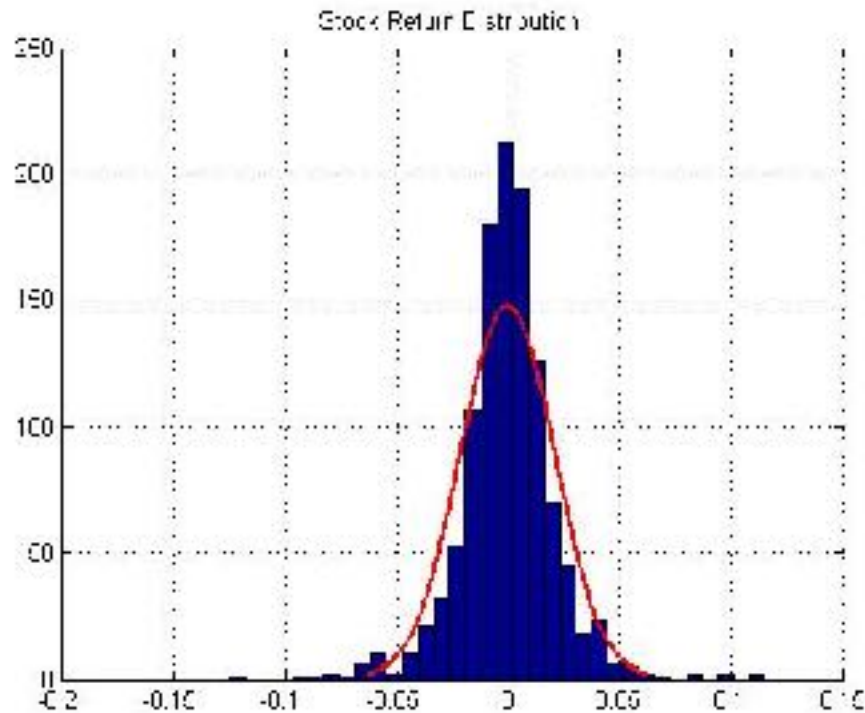
3) Are the returns auto-correlated? Are the returns –squared auto-correlated? Do you need any volatility modeling?

Answer:

Individual Index: MSFT.

The following graph depicts the stock price trend and stock return trend, respectively, in the past five years from 1/1/2008-12/31/2012. The stock price mainly fluctuated between the range in \$15 and \$30. The stock return display a normal distribution like shape, which we will conduct further test later.



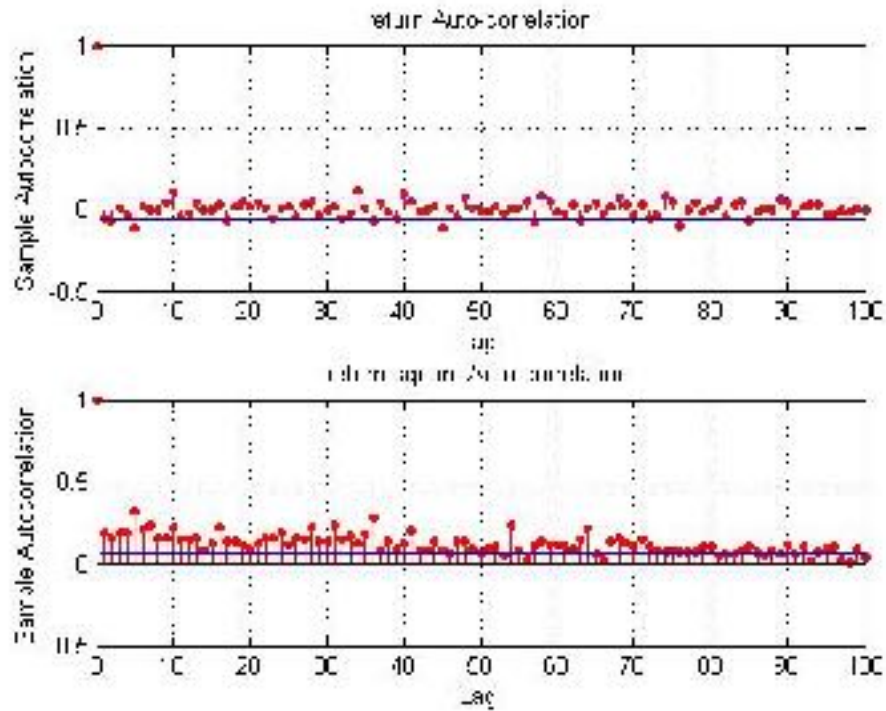


1) First four moments mean / variance / skewness / kurtosis

Four Moments	mean	variance	skewness	kurtosis
MSFT	-1.2018e-004	4.3772e-004	0.0720	7.9213

2) With a mean of -0.0001 and variance of 0.0004, the return has a higher kurtosis than the normal distribution, though the skewness is close to zero. So normal distribution may not exactly capture the feature of return of Microsoft.

3) From the autocorrelation test we can conclude that there is little autocorrelation in the return. However, there is large autocorrelation in the square-return, which indicates GARCH effect, that is, we need to conduct volatility modeling.



2. Try different volatility models and find the best one for your Index.

1) State what different volatility modeling you tried. Show all results.

2) State how do you check which volatility is a better one? Show the test results.

Answer:

1) I chose three volatility models

- EWMA

$$\sigma(t)^2 = 0.06 * R(t-1)^2 + 0.94 * \sigma(t-1)^2$$

- GARCH

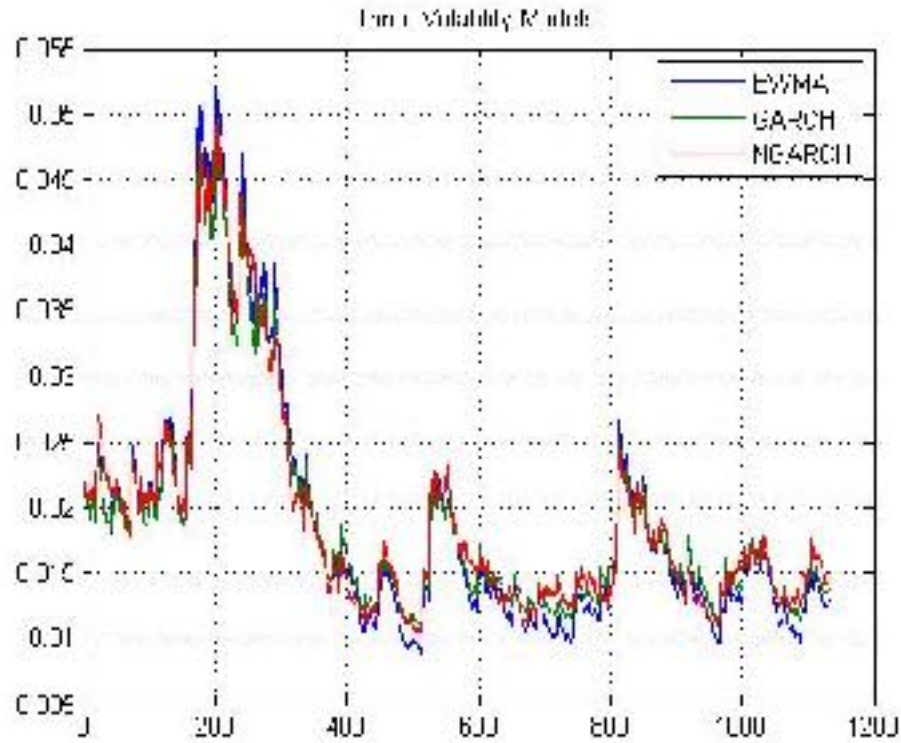
P,Q	K	ARCH	GARCH
GARCH(1,1)	3.0072e-006	0.0499	0.9412

$$\sigma(t)^2 = 3.0072e-006 + 0.0499 * R(t-1)^2 + 0.9412 * \sigma(t-1)^2$$

- NGARCH

omega	alpha	theta	Beta
2.6431e-006	0.0399	0.6521	0.9365

$$\sigma(t)^2 = 2.6431e-006 + 0.0399 * (R(t-1) - 0.6521 * \sigma(t-1))^2 + 0.9365 * \sigma(t-1)^2$$



$$2) R(t+1)^2 = B_0 + B_1 * \sigma(t+1)^2 + e(t+1)$$

	EWMA	GARCH	NGARCH
B0	0.0001	-0.0000	-0.0000
B1	0.8697	1.0279	1.0052

From the regression test, we conclude that NGARCH is the best model of the three, with the intercept B0 equal to 0 and B1 close to 1.

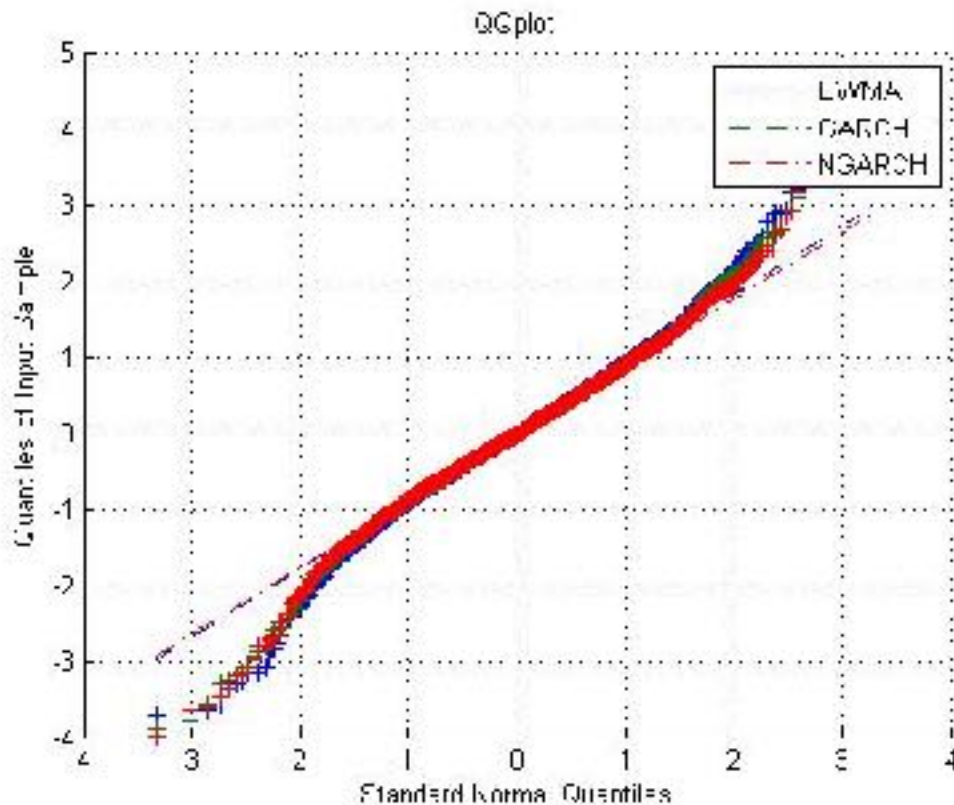
3. Calculate 1% VaR for the security, assuming normal distribution, *t*-distribution, Cornish-Fisher and EVT and estimating Hill parameter. Show graphically the predicted VaR with different distributions for the testing period.

1) Normal Distribution

We chose the NGARCH to be our volatility model, since from the regression test, it shows the best result. “lsigma” stands for leveraged sigma

$$\text{VaR}_{\text{normal}} = -\text{eps} \cdot \text{lsigma}; \quad \text{eps} \sim N(0,1)$$

From the graph we can see that the quantile does not fit very well to the normal distribution. The return's quantile is fatter than normal distribution



2) T-Distribution

From the QMLE method, we get the *d* estimate to be:

$$d = 6.0519$$

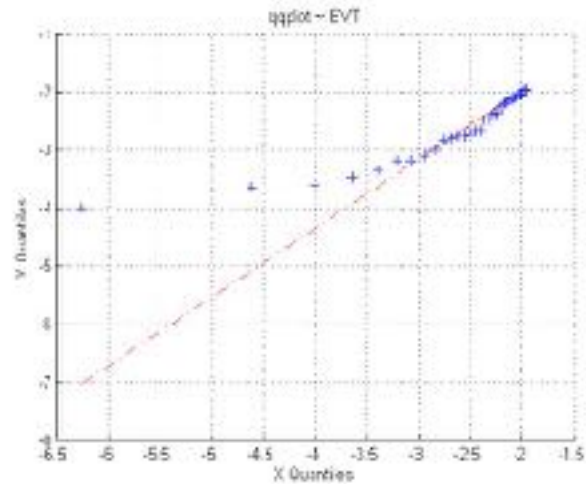
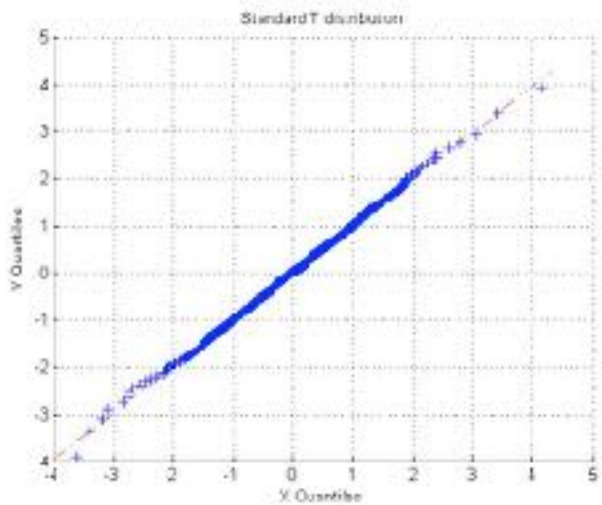
$$\text{VaR}_{\text{td}} = -\text{lsigma} \cdot \text{tinv}(0.01, d) \cdot \sqrt{(d-2)/d};$$

From the graph below, we can see that *t*-distribution actually fits very well, which captures the feature of return standardized by the NGARCH model.

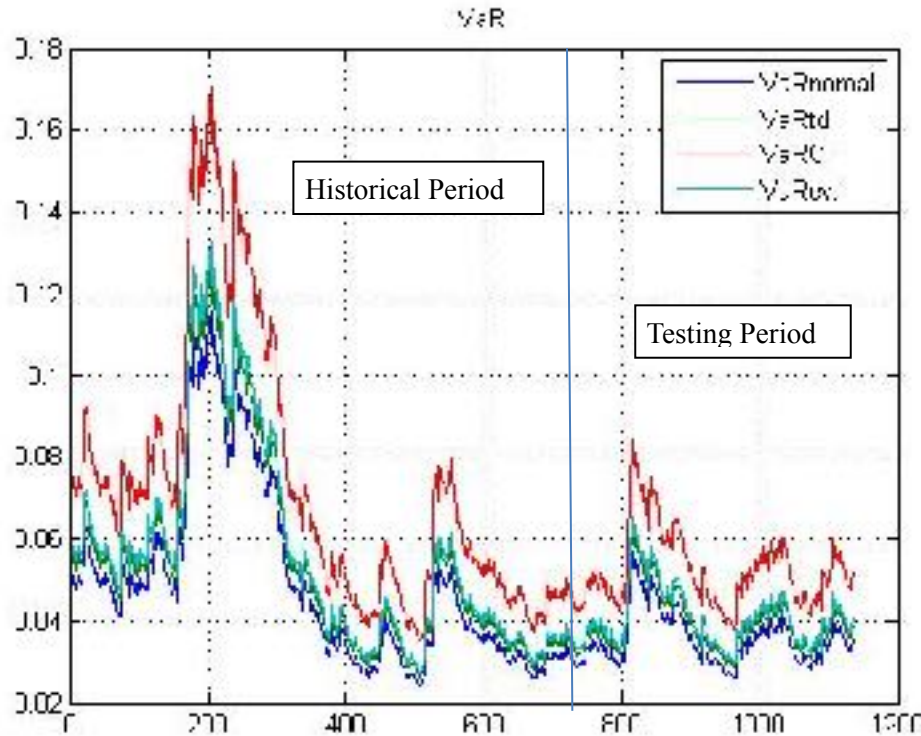
3) Cornish-Fisher

 $CF = -3.4221$; $VaRCF = -\sigma * CF$;

4) EVT

Hill parameter = 0.3325; $C = 0.1898$ 

If we compare the VaR from the distributions analyzed above, we get the following picture. From the graph we can see that the VaR got from Cornish-Fisher is the highest one, while VaR got from normal distribution is the lowest one.



4. Calculate 1%, 1 day VaR using:

1) Monte Carlo Simulation (*t*-distribution / GARCH)

2) FHS (GARCH) for the testing periods.

3) Show graphically how VaR changes for testing period given below.

The design period is 1 Jan, 2008 – 31 December 2010.

Testing period is 1 Jan., 2011 – 31 December 2012.

Answer:

1) Monte Carlo Simulation $\sim t(d)/\text{GARCH}$

Number of design period: 671;

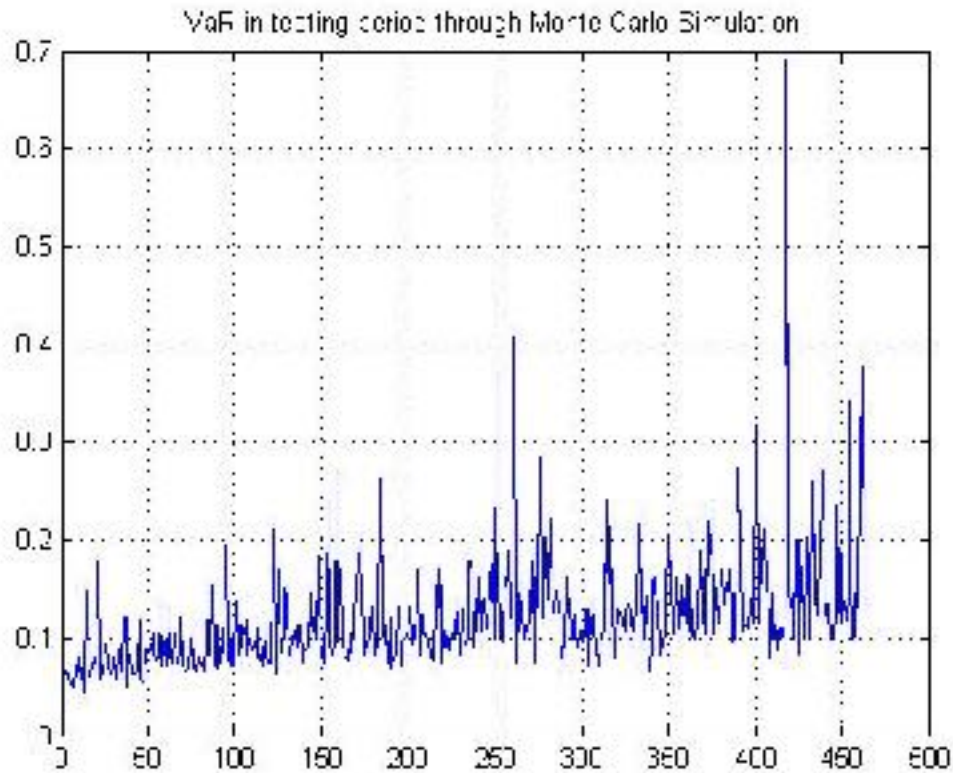
Number of testing period: 463;

GARCH Model for the design period:

P,Q	K	ARCH	GARCH
GARCH(1,1)	2.5571e-006	0.0616	0.9333

The last sigma in the design period, which we use to set the first number in the testing period: $\text{sigmatest} = 0.0118$.

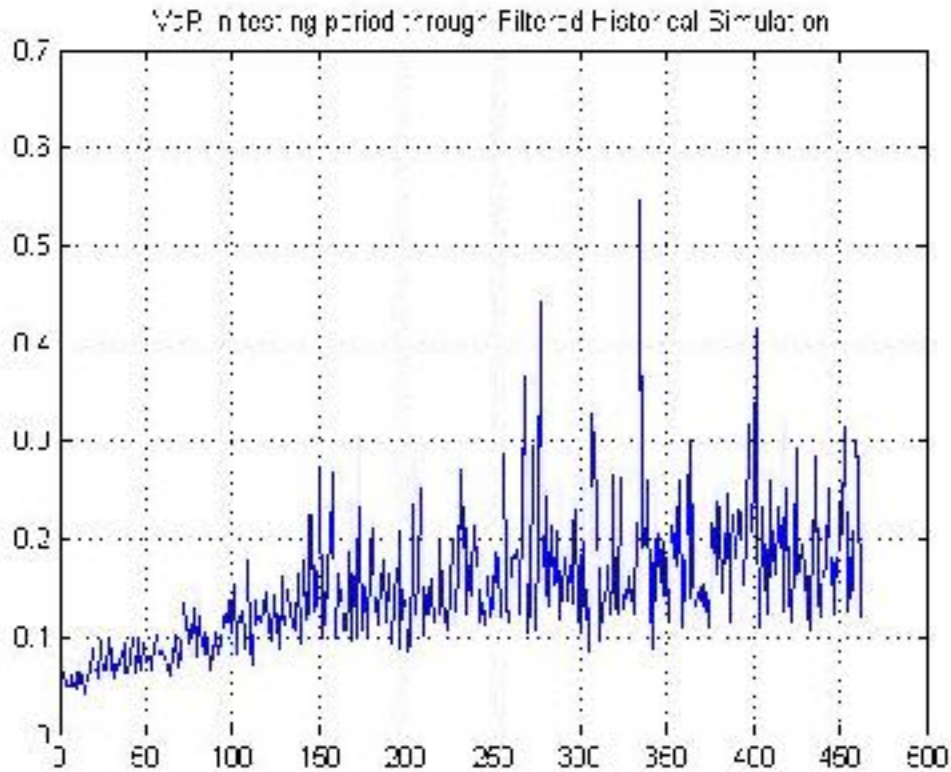
Generating random number from the MATLAB built-in function, we obtain the following graph which depicts the VaR in the testing period.



2) Filtered Historical Simulation

In the filtered historical method, we draw the return from the design period and use it as a sample pool, and then we get the random number z to calculate the volatility evolution.

From the graph we can see there is a lifting trend of VaR in the filtered historical simulation.



5. Back-testing for the unconditional 1% VaR as well as for conditional tests using historical period and testing period as shown above.

5a) Compare VaR with different measure in unconditional test (LR_{uc}), independent test (LR_{ind}) and conditional coverage test (LR_{cc})

5b) Find the best method of calculating VaR for all your indices, and show enough evidence to support your conclusion.

5a) Answer:

Re-estimate the coefficients for the GARCH and NGARCH.

This time, estimate coefficients in the design period only:

P,Q	K	ARCH	GARCH
-----	---	------	-------

GARCH(1,1)	2.5571e-006	0.0616	0.9333
------------	-------------	--------	--------

Omega2	Alpha2	Theta2	Beta2
3.6012e-006	0.0390	0.7040	0.9361

1) LRuc: Unconditional Coverage Testing

Model	EWMA	GARCH	NGARCH
T_0	461	462	462
T_1	2	1	1
$L(\hat{\pi})$	0.0025	0.7954	0.7954
$L(p)$	0.1945*1.0e-003	0.0963*1.0e-003	0.0963*1.0e-003
LR_{uc}	-8.6792	4.2236	4.2236
P-value	1.0000	0.0399	0.0399
Conclusion	Reject	Accept	Accept

From the testing results showing above, we can see that at the 5% significance level, the NGARCH model and GARCH model is in the acceptable range. These two volatility models perform very well, while the EWMA model does not fit very well according to the back testing.

2) LRind: Independent Testing

Model	EWMA	GARCH	NGARCH
T_{00}	458	460	460
T_{01}	2	1	1
T_{10}	2	1	1
T_{11}	0	0	0
$\hat{\pi}_{01}$	0.0043	0.0022	0.0022
$\hat{\pi}_{11}$	0	0	0
$L(\hat{\pi}_1)$	2.5695e-006	7.9887e-004	7.9887e-004

LR _{ind}	0.0261	0.0087	0.0087
P-value	0.8718	0.9258	0.9258
Conclusion	Reject	Reject	Reject

From the test results showing above, we can see that the LR_{ind}, which fits the chi-square distribution, has a high p-value. That is, we should reject the null hypothesis that the volatility (All of the three, EWMA, GARCH, NGARCH) have no clustered effect. Actually, the results show that volatilities are not independent. They are clustered in time.

3) LR_{cc}: Conditional Coverage Testing

Model	EWMA	GARCH	NGARCH
LR _{cc}	-8.6532	4.2322	4.2322
p-value	1	0.1205	0.1205
Conclusion	Reject	Depends	Depends

Finally, we consider the joint testing for independence and correct coverage. From the testing results we can see that in the EWMA method, we should reject the null hypothesis. However, for the other two, the conclusion depends on what level of significance we choose. If we choose the significant level to be less than 12%, then we should reject the two methods. If we choose the significant level to be greater than 12%, we may accept the null hypothesis.

5b) Answer:

Since my role is a capital constraint bank, I will choose a volatility model which gives me a small VaR. Since some of the capital requirements are set as “k times of VaR”, given a small VaR, there’s less capital requirement for me to put on the risk capital, which means I have a lot more to put on other kind of investment. From the results above, I will choose a model which gets the smaller VaR. GARCH or NGARCH may be a better choice for me than EWMA.

EXTRA QUESTION FOR ONE INDEX GROUP A

Calculate 1%, 1 day VaR in HS, WHS, Monte Carlo Simulation (t-distribution / GARCH) and FHS (GARCH) for the testing periods. Show graphically how VaR changes for testing period given above.

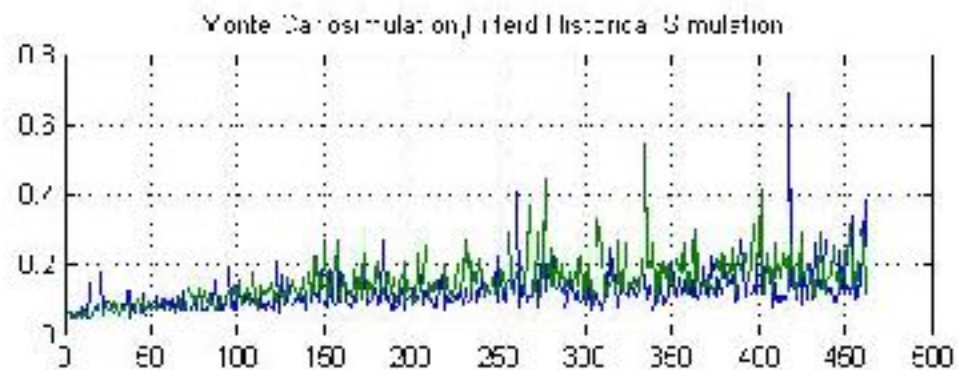
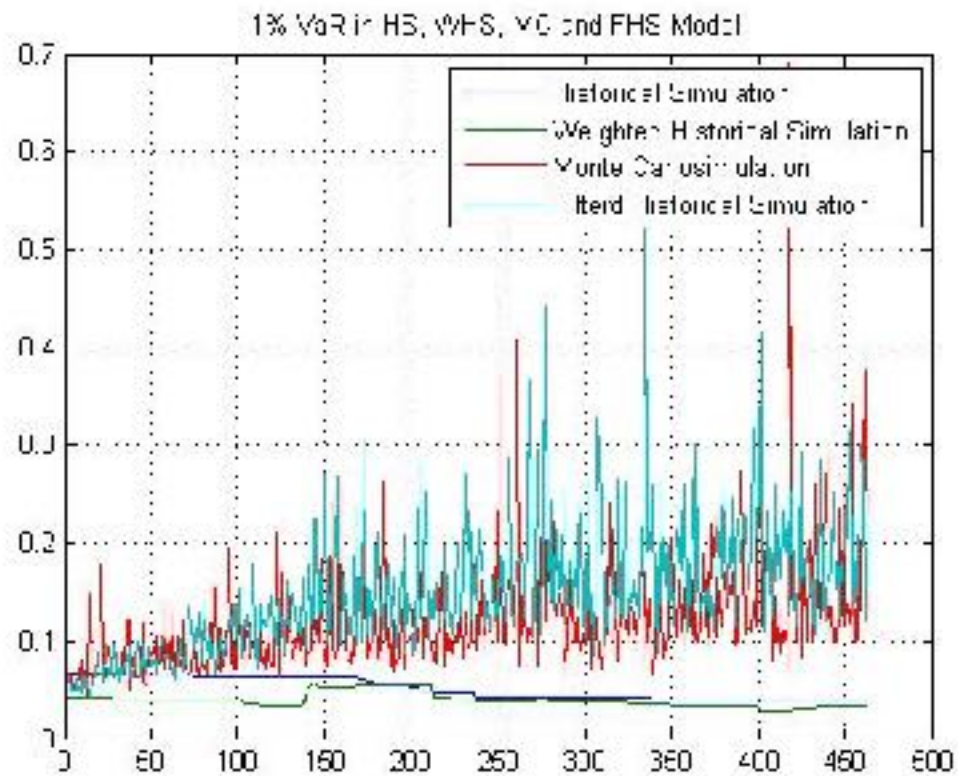
Answer:

For the Monte Carlo Simulation and FHS, we have calculated before.

For the HS and WHS, we used the design period as the observation period and calculate the percentile as time moves by.

Finally, I get the following graph, which depicts the evolution of VaR.

The VaR got from models have more fluctuation compared with the historical simulation.



*Appendix: MATLAB Code***Main Code**

```

% 1
load stockindex % load data

price=stockindex;
logreturn=price2ret(price);

%plot price and return
subplot(211);plot(price);title('Stock Price'),grid on;
subplot(212);plot(logreturn);title('Stock Return'),grid on;

%distribution: Price~lognormal; return~normal
histfit(logreturn);title('Stock Return Distribution'),grid on;

% mean, standard deviation, skewness, kurtosis
meanreturn=mean(logreturn)
stdreturn=std(logreturn);varreturn=stdreturn^2
skewreturn=skewness(logreturn)
kurtreturn=kurtosis(logreturn)

% conducted autocorrelation analysis towards the return
% there is little autocorrelation in the return itself
[ACF1, Lags1, Bounds1] = autocorr(logreturn, 100);
subplot(211),autocorr(logreturn, 100),title('return Auto-correlation');

%The return is not independent, therefore, we need volatility modeling
squarereturn=logreturn.^2;
[ACF2, Lags2, Bounds2] = autocorr(squarereturn, 100);
subplot(212),autocorr(squarereturn, 100);title('return square Auto-
correlation');

%*****
%2
% Several volatility models
% 1) sigma(:,1)~EWMA
% 2) sigma(:,2)~GARCH
% 3) sigma(:,3)~NGARCH
spec=garchset('display','off','P',1,'Q',1);
coef=garchfit(spec,logreturn);

%Estimate coefficient of NGARCH
L1=@(mu) sum(rmgQ(mu,3,logreturn,[]));
A = [0 1 0 1]; %alpha + beta<1
B = 1;
LB = [0 0 0 0];
initials= [0.0001 0.05 0.6 0.94];
options = optimset('Display','off','MaxIter',999999999,'MaxFunEvals',
999999999,'Algorithm','sqp');
[muvalue,fval1] = fmincon(@(mu) (-L1(mu)),initials,A,B,[],[],LB,[],
[],options);
omega=muvalue(1),alpha=muvalue(2),theta=muvalue(3),beta=muvalue(4)

n=length(logreturn);
sigma=zeros(n,3);
sigma(1,:)=stdreturn;

```

```

for i=2:n;
    %EWMA
    sigma(i,1)=sqrt(0.06*(logreturn(i-1)^2)+0.94*(sigma(i-1,1).^2));
    %GARCH
    sigma(i,
2)=sqrt(coef.K+coef.GARCH*(sigma(i-1,2).^2)+coef.ARCH*(logreturn(i-1).^
2));
    %NGARCH
    sigma(i,3)=sqrt(omega+alpha*(logreturn(i-1)-
theta*sigma(i-1,3)).^2+beta*sigma(i-1,3).^2);
end

plot(sigma),grid on;
title('Three Volatility Models');
legend('EWMA','GARCH','NGARCH');

%Test
y=logreturn.^2;% R^2 and sigma^2
x1=zeros(n,2); x1(:,1)=sigma(:,1).^2; x1(:,2)=1; %EWMA
x2=zeros(n,2); x2(:,1)=sigma(:,2).^2; x2(:,2)=1; %GARCH
x3=zeros(n,2); x3(:,1)=sigma(:,3).^2; x3(:,2)=1; %NGARCH
[B1,BINT1,R1,RINT1,STATS1] = regress(y,x1);
[B2,BINT2,R2,RINT2,STATS2] = regress(y,x2);
[B3,BINT3,R3,RINT3,STATS3] = regress(y,x3);

%After testing, we found that the NGARCH is the best model of all.
lsigma=sigma(:,3); %leveraged garch std
standret=logreturn./lsigma; %leveraged garch standard return

%*****
%3
eps=norminv(0.01,0,1);

%Normal
VaRnormal=-eps.*lsigma; %sigma is calculated using the three methods
above
stand=repmat(logreturn,1,3)./sigma;
figure,qqplot(stand);grid on
title('QQplot');legend('EWMA','GARCH','NGARCH');

%t(d) Distribution
L=@(d)n*(gammaaln((d+1)/2)-gammaaln(d/2)-log(pi)/2-log(d-2)/2) ...
-1/2*(1+d)*sum(log(1+standret.^2/(d-2)));
[dvalue,fval] = fminsearch(@(d)(-L(d)),10);

for i = 1: n
    tp=(i-0.5)/n;
    if tp<=0.5
        t(i)= -abs(tinv(2*tp,dvalue))*sqrt((dvalue-2)/dvalue);
    else
        t(i)= abs(tinv(2*(1-tp),dvalue))*sqrt((dvalue-2)/dvalue);
    end
end
figure,qqplot(standret,t);grid on
title('Standard T distribution');
VaRtd=-lsigma*tinv(0.01,dvalue)*sqrt((dvalue-2)/dvalue);

```



```

%Cornish-Fisher
skew=skewness(standret);kurt=kurtosis(standret);F=norminv(0.01);
CF=F+skew/6*(F^2-1)+kurt/24*(F^3-3*F)-skew^2/36*(2*F^3-5*F);
VaRCF=-lsigma*CF;

%EVT
u=0.05;
Tu = fix(n*u);
evtret=sort(standret);evtret=evtret(1:Tu);
uret= evtret(Tu);
tailpara= sum(log(abs(evtret./uret)))/Tu;
C= Tu/n*abs(uret)^(1/tailpara);
VaRevt=lsigma*abs(uret)*(0.01/(Tu/n))^(-tailpara);
for i = 1: Tu
    evt(i)= uret*((i-0.5)/n)/(Tu/n)^(-tailpara);
end
figure,qqplot(evt,evtret),grid on
title('qqplot ~ EVT') % Compare the quatiles only

%plot
VaR=[VaRnormal,VaRtd,VaRCF, VaRevt];
plot(VaR)
title('VaR'),legend('VaRnormal','VaRtd','VaRCF','VaRevt'),grid on

%*****
%4
%Monte Carlo
n=671; %1/1/2008 - 12/31/2010
n2=length(logreturn)-n; %1/1/2011 - 12/31/2012
m=1000;
spec=garchset('display','off','P',1,'Q',1);
coef2=garchfit(spec,logreturn(1:n));

%Choose the last sigma in the design period to be the first number of
sigma
%in testing period.
sigmatest=stdreturn;
for i=2:n;

sigmatest=sqrt(coef2.K+coef2.GARCH*(sigmatest.^2)+coef2.ARCH*(logreturn
(i-1).^2));
end
MonSi=ones(m,n2+1);MonSi(:,1)=sigmatest;
MonR=ones(m,n2);

%Generate random number from t(d), and take the inverse after
converting
%to student t-distribution
z=trnd(dvalue,m,n2)*sqrt((dvalue-2)/dvalue); %used dvalue calculated
before

for i=1:n2
    %R=z*sigma, z generate from normal distrubution by built-in
function
    MonR(:,i)=MonSi(:,i).*z(:,i);
    %GARCH model

```

```

MonSi(:,i+1)=sqrt(coef2.K+coef2.GARCH*(MonSi(:,i).^2)+coef2.ARCH*(MonR(
(:,i).^2));
end
VaRmon=-prctile(MonR,0.01*100);
plot(VaRmon),grid on
title('VaR in testing period through Monte Carlo Simulation');

%FHS
%mx1 column index, using the range from 1~671 historical data as data
pool
index=randi(n,m,n2);
standret2=logreturn./sigma(:,2); %garch standard return
z2=standret2(index); % generate return from the historical data

FSi=ones(m,n2+1);FSi(:,1)=sigmatest;
FR=ones(m,n2);

for i=1:n2
    %R=z*sigma, z generate from historical pool
    FR(:,i)=FSi(:,i).*z2(:,i);

FSi(:,i+1)=sqrt(coef2.K+coef2.GARCH*(FSi(:,i).^2)+coef2.ARCH*(FR(:,i).^
2));
end
VaRfh=-prctile(FR,0.01*100);
plot(VaRfh),grid on
title('VaR in testing period through Filtered Historical Simulation');

%*****
%5
p=0.01;    % 1%VaR
% Three methods of volatility
% coefficients ~ get from design period
% test ~ using data in testing period
%NGARCH coefficients estimated from design period only
L2=@(mu) sum(rmgQ(mu,3,logreturn(1:n2),[]));
A = [0 1 0 1]; %alpha + beta<1
B = 1;
LB = [0 0 0 0];
initials= [0.0001 0.05 0.6 0.94];
options = optimset('Display','off','MaxIter',999999999,'MaxFunEvals',
999999999,'Algorithm','sqp');
[muvalue,fval1] = fmincon(@(mu) (-L2(mu)),initials,A,B,[],[],LB,[],
[],options);
omega2=muvalue(1);alpha2=muvalue(2);theta2=muvalue(3);beta2=muvalue(4);
sigma2=zeros(n2,3);
sigma2(1,:)=sigmatest;
for i=2:n2;
    %EWMA
    sigma2(i,1)=sqrt(0.06*(logreturn(i-1)^2)+0.94*(sigma(i-1,1).^2));
    %GARCH
    sigma2(i,
2)=sqrt(coef2.K+coef2.GARCH*(sigma(i-1,2).^2)+coef2.ARCH*(logreturn(i-1
).^2));
    %NGARCH

```

```

    sigma2(i,3)=sqrt(omega2+alpha2*(logreturn(i-1)-
theta2*sigma(i-1,3)).^2+beta2*sigma(i-1,3).^2);
end

VaR=-eps*sigma2;
vartest=VaR;
test1=logreturn(n+1:end);
test1= repmat(test1,1,3);

%LRuc: Unconditional Coverage Testing
%condition: return<-var ==> return>var

%compare(:,1)~EWMA
%compare(:,2)~GARCH
%compare(:,3)~NGARCH
compare=(test1>vartest);
t=length(compare);
t1=sum(compare)
t0=t-t1

lp=(1-p).^t0*p.*t1
l1=(1-t1./t).^t0.*(t1./t).^t1
lruc=-2*log(lp./l1)
freedom=1;
pvalue=1-chi2cdf(lruc,freedom)

%Independent Testing
%1) EWMA
t00=0;t01=0;
t10=0;t11=0;
for i=2:t
    if compare(i,1)==0 & compare(i-1,1)==0
        t00=t00+1;
    elseif compare(i,1)==1 & compare(i-1,1)==0
        t01=t01+1;
    elseif compare(i,1)==0 & compare(i-1,1)==1
        t10=t10+1;
    elseif compare(i,1)==1 & compare(i-1,1)==1
        t11= t11+1;
    end
end

EWMA_tn=[t00 t01 t10 t11]
pi01=t01./(t01+t00)
pi11=t11./(t10+t11)
lpi1=(1-pi01).^t00.*pi01.^t01.*(1-pi11).^t10.*pi11.^t11
lrind=-2*log(l1(1)/lpi1)
pvalue2=1-chi2cdf(lrind,freedom)
lpiwhole=zeros(1,3);
lpiwhole(1)=lpi1;

%2) GARCH
t00=0;t01=0;
t10=0;t11=0;
for i=2:t
    if compare(i,2)==0 & compare(i-1,2)==0
        t00=t00+1;

```

```

elseif compare(i,2)==1 & compare(i-1,2)==0
    t01=t01+1;
elseif compare(i,2)==0 & compare(i-1,2)==1
    t10=t10+1;
elseif compare(i,2)==1 & compare(i-1,2)==1
    t11= t11+1;
end
end

GARCH_tn=[t00 t01 t10 t11]
pi01=t01./(t01+t00)
pi11=t11./(t10+t11)
lpi1=(1-pi01).^t00.*pi01.^t01.*(1-pi11).^t10.*pi11.^t11
lrind=-2*log(l1(2)/lpi1)
pvalue2=1-chi2cdf(lrind,freedom)
lpiwhole(2)=lpi1;

%3) NGARCH
t00=0;t01=0;
t10=0;t11=0;
for i=2:t
    if compare(i,3)==0 & compare(i-1,3)==0
        t00=t00+1;
    elseif compare(i,3)==1 & compare(i-1,3)==0
        t01=t01+1;
    elseif compare(i,3)==0 & compare(i-1,3)==1
        t10=t10+1;
    elseif compare(i,3)==1 & compare(i-1,3)==1
        t11= t11+1;
    end
end

NGARCH_tn=[t00 t01 t10 t11]
pi01=t01./(t01+t00)
pi11=t11./(t10+t11)
lpi1=(1-pi01).^t00.*pi01.^t01.*(1-pi11).^t10.*pi11.^t11
lrind=-2*log(l1(3)/lpi1)
pvalue2=1-chi2cdf(lrind,freedom)
lpiwhole(3)=lpi1;

%Conditional Coverage Testing
lrcc=-2*log(lp./lpiwhole)
pvalue3=1-chi2cdf(lrcc,freedom+1)

```

Built-in Code

```

function f=rmgQ(mu,casechoice,x,y)
n=length(x);
%First assign the sigma(1)
sigma=zeros(n,1);
stdreturn=std(x);
sigma(1)=stdreturn;
f=zeros(n,1);

switch casechoice
    case 2 %Garch (1-alpha-beta), alpha, beta
        for i=2:n;

```

```

        sigma(i)=sqrt(stdreturn^2*(1-mu(1)-mu(2))
+x(i-1).^2*mu(1)+sigma(i-1).^2*mu(2));
    end
    f=log(normpdf(x,0,sigma));
    case 3 %Leverage
        for i=2:n;
            sigma(i)=sqrt(mu(1)+mu(2)*(x(i-1)-
mu(3)*sigma(i-1)).^2+mu(4)*sigma(i-1).^2);
        end
        f=log(normpdf(x,0,sigma));
    case 9 %t(d)
        for i=2:n;
            sigma(i)=sqrt(mu(1)+mu(2)*(x(i-1)-
mu(3)*sigma(i-1)).^2+mu(4)*sigma(i-1).^2+mu(5)*y(i-1).^2/252);
        end
        f=gammaaln((mu(6)+1)/2)-gammaaln(mu(6)/2)-log(pi)/2-
log(mu(6)-2)/2-log(sigma) ...
        -1/2*(1+mu(6))*log(1+(x./sigma).^2/(mu(6)-2));
end

```