Xiaoxuan Wang
HW4

1.Generate Data using Garch(1,1) model with parameters alpha0=0.01, alpha1=0.3, beta1=0.6

```
function Y = Generate_Data(len)

e=normrnd(0,1,[1,len-1]);
alpha0=0.01;
alpha1=0.3;
beta1=0.6;
Y = zeros(1,len);
Sigma = zeros(1,len);
Y(1)=0.01;
Sigma(1)=0.05;
for n = 1:len-1
    Sigma(n+1) = sqrt(alpha0+alpha1*Y(n)^2+beta1*Sigma(n)^2);
    Y(n+1) = Sigma(n+1)*e(n);
end
```

2.Likelihood Function

```
function like=L(Y,y0,sigma0,Para)
Sigma=zeros(1,length(Y));
Sigma(1)=sqrt(Para(1)+Para(2)*y0^2+Para(3)*sigma0^2);
for n = 2:length(Y)
    Sigma(n) = sqrt(Para(1)+Para(2)*Y(n-1)^2+Para(3)*Sigma(n-
1)^2);
end
like=0;
for n=1:length(Y)
    like=like-Y(n)^2/(2*Sigma(n)^2)-log(sqrt(2*pi)*Sigma(n));
end
```

3.Use data to solve for MLE, CI and P_value; count the number of rejections out of 10000 times

```
count_alpha0=0;
count_alpha1=0;
count_beta1=0;
for i=1:10000
```

```
%generate data using GARCH(1,1) model
Y = Generate_Data(1000);
%to avoid the impact of the initial value of y1 and sigma1
Y=Y(501:length(Y));

ysquare=(Y-mean(Y)).^2;
sum_ysquare=sum(ysquare);
sigma0=sqrt(sum_ysquare/length(Y));
y0=sqrt(sum_ysquare/length(Y));

%to maximize the likelihood function, we minimize -L
lkh=@(Para)-L(Y,y0,sigma0,Para);
[MLE,fval,~,~,~,hessian]=fminunc(lkh,[0.02,0.28,0.61]);
%MLE=[0.0103,0.2994,0.5985]
fval_of_L=-fval;
%fval_of_L=-1.1792e+03
hessian_of_L=-inv((-hessian)/length(Y));

sigma_square_hat_for_alpha0=hessian_of_L(1,1);
lower_bound_alpha0=MLE(1)-
1.96*sqrt(sigma_square_hat_for_alpha0/length(Y));
%lower_bound_alpha0 is 0.0087.
upper_bound_alpha0=MLE(1)+1.96*sqrt(sigma_square_hat_for_alpha0/
length(Y));
%upper_bound_alpha0 is 0.0119.
z_alpha0=(MLE(1)-
0.01)/sqrt(sigma_square_hat_for_alpha0/length(Y));
p_alpha0=2*(1-normcdf(abs(z_alpha0)));
%P-Value for alpha0 is 0.7282.

sigma_square_hat_for_alpha1=hessian_of_L(2,2);
lower_bound_alpha1=MLE(2)-
1.96*sqrt(sigma_square_hat_for_alpha1/length(Y));
%lower_bound_alpha1 is 0.2699.
upper_bound_alpha1=MLE(2)+1.96*sqrt(sigma_square_hat_for_alpha1/
length(Y));
%upper_bound_alpha1 is 0.3290.
z_alpha1=(MLE(2)-
0.3)/sqrt(sigma_square_hat_for_alpha1/length(Y));
p_alpha1=2*(1-normcdf(abs(z_alpha1)));
%P-Value for alpha1 is 0.9693.

sigma_square_hat_for_beta1=hessian_of_L(3,3);
lower_bound_beta1=MLE(3)-
1.96*sqrt(sigma_square_hat_for_beta1/length(Y));
%lower_bound_beta1 is 0.5634.
upper_bound_beta1=MLE(3)+1.96*sqrt(sigma_square_hat_for_beta1/le
ngth(Y));
%upper_bound_beta1 is 0.6336.
```

```
z_beta1=(MLE(3)-0.6)/sqrt(sigma_square_hat_for_beta1/length(Y));
p_beta1=2*(1-normcdf(abs(z_beta1)));
%P-Value for beta1 is 0.9337.
%Thus, we cannot reject Ho: alpha0=0.01, alpha1=0.3, beta1=0.6.
if (p_alpha0<=0.05)
    count_alpha0 = count_alpha0+1;
end
if (p_alpha1<=0.05)
    count_alpha1 = count_alpha1+1;
end
if (p_beta1<=0.05)
    count_beta1 = count_beta1+1;
end
end
```

The percentages of rejecting alpha0=0.01, alpha1=0.3, beta1=0.6 are 4.19%, 5.94% and 5.37% respectively.

count_alpha0	419
count_alpha1	594
count_beta1	537

4.Power Analysis

```
count_alpha0_2=0;
count_alpha1_2=0;
count_beta1_2=0;
for i=1:10000
%generate data using GARCH(1,1) model with alpha1=0.02,
alpha1=0.45, beta1=0.5
Y = Generate_Data_2(1000);
Y=Y(501:length(Y));

ysquare=(Y-mean(Y)).^2;
sum_ysquare=sum(ysquare);
sigma0=sqrt(sum_ysquare/length(Y));
y0=sqrt(sum_ysquare/length(Y));

lkh=@(Para)-L(Y,y0,sigma0,Para);
[MLE,fval,~,~,~,hessian]=fminunc(lkh,[0.03,0.4,0.85]);
fval_of_L=-fval;
hessian_of_L=-inv((-hessian)/length(Y));

sigma_square_hat_for_alpha0=hessian_of_L(1,1);
lower_bound_alpha0=MLE(1)-
1.96*sqrt(sigma_square_hat_for_alpha0/length(Y));
upper_bound_alpha0=MLE(1)+1.96*sqrt(sigma_square_hat_for_alpha0/
length(Y));
```

```
z_alpha0=(MLE(1)-
0.01)/sqrt(sigma_square_hat_for_alpha0/length(Y));
p_alpha0=2*(1-normcdf(abs(z_alpha0)));

sigma_square_hat_for_alpha1=hessian_of_L(2,2);
lower_bound_alpha1=MLE(2)-
1.96*sqrt(sigma_square_hat_for_alpha1/length(Y));
upper_bound_alpha1=MLE(2)+1.96*sqrt(sigma_square_hat_for_alpha1/
length(Y));
z_alpha1=(MLE(2)-
0.3)/sqrt(sigma_square_hat_for_alpha1/length(Y));
p_alpha1=2*(1-normcdf(abs(z_alpha1)));

sigma_square_hat_for_beta1=hessian_of_L(3,3);
lower_bound_beta1=MLE(3)-
1.96*sqrt(sigma_square_hat_for_beta1/length(Y));
upper_bound_beta1=MLE(3)+1.96*sqrt(sigma_square_hat_for_beta1/le
ngth(Y));
z_beta1=(MLE(3)-0.6)/sqrt(sigma_square_hat_for_beta1/length(Y));
p_beta1=2*(1-normcdf(abs(z_beta1)));
if (p_alpha0<=0.05)
    count_alpha0_2 = count_alpha0_2+1;
end
if (p_alpha1<=0.05)
    count_alpha1_2 = count_alpha1_2+1;
end
if (p_beta1<=0.05)
    count_beta1_2 = count_beta1_2+1;
end
end
```

The percentages of rejecting alpha0=0.01, alpha1=0.3, beta1=0.6 are 40.44%, 46.71% and 38.14% respectively. The high percentage of rejection indicates the power of the estimator and testing theory.

count_alpha0_2	4044
count_alpha1_2	4671
count_beta1_2	3814