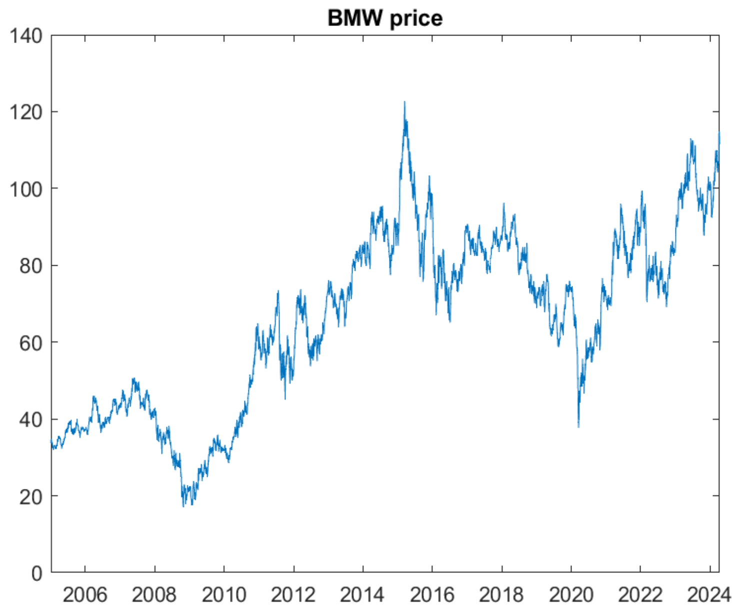
FINANCIAL ECONOMETRICS ASSIGNMENT

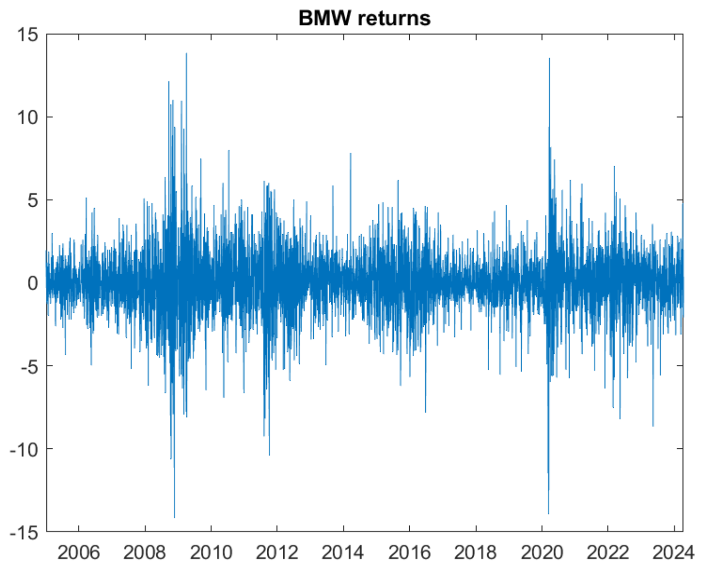
Lorenzo Brontesi (0341313)
Giuseppe De Santis (0349554)
Geanina Adelina Stefania Hosszu (0349063)
Benedetta Marchettini (0341607)

Univariate modelling

Using 0349063 as seed of the random number generator, we get BMW as our stock of interest, in the period 01-01-2005 up to 31-05-2024.

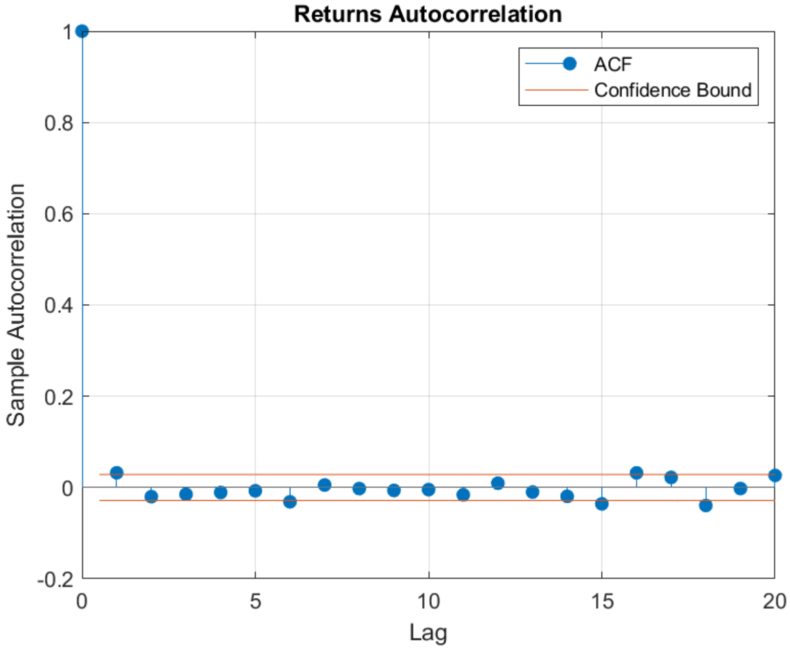


Considering the returns of the stock, computed as the first difference of the log price times 100, we obtain the following graph:

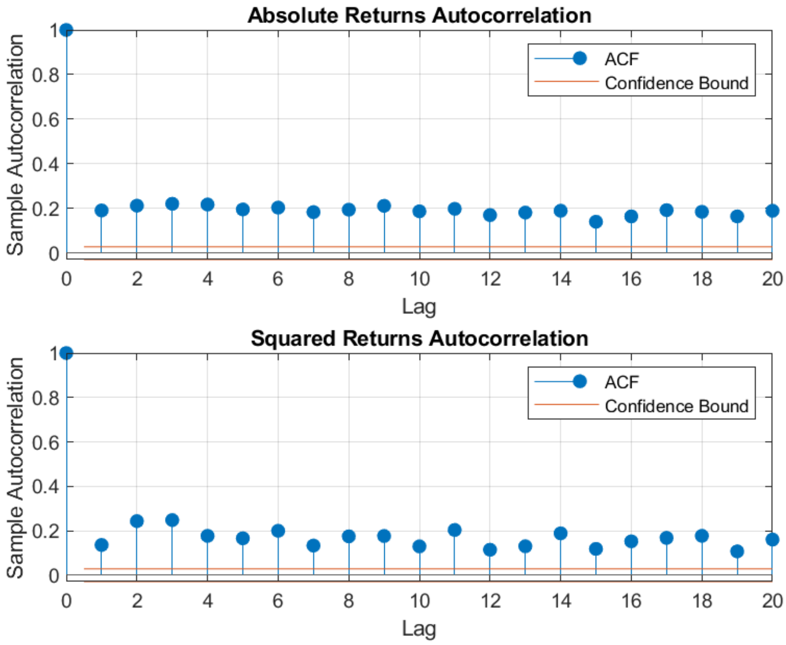


The returns have different stylized facts that characterize their behavior:

- They are weakly dependent over time, as we can see in the autocorrelation, where in most of the lags the value is approximately zero for our stock:

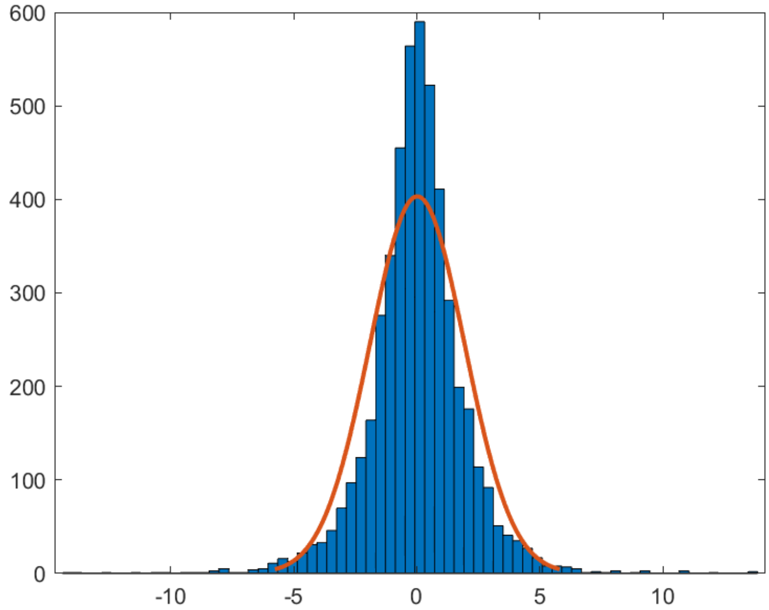


- The autocorrelation of transformations of the returns displays dependence over time, especially for absolute returns, but also for squared returns:



The result shows that the returns are not independent and identically distributed random variables, indeed large absolute returns are more likely to be followed by absolute large returns, this phenomena is known as volatility clustering.
So the conditional variance is time varying and periods of low volatility are alternated by periods of high volatility.

- The distribution of returns is not normal, and it is clear visualizing the histogram of returns of BMW compared with the normal distribution:



We can see that we have a higher peak around mean, but also more extreme events. Indeed the distribution is leptokurtic, namely the tails are thicker, so extreme events are more likely to happen; computing the standardized fourth moment we get a value of 8.6460, which is larger than 3, the value of the kurtosis for a gaussian random variable . It is also noticeable that the mean is approximately 0, and the distribution is not symmetric, with a value for the standardized third moment of -0.0920, which is small but indicates an asymmetry in the left part of the distribution. The sample variance of the asset is 3.7183.
Furthermore, performing a Jarque-Bera test, we get a value in the rejection region as we would expect.

The fact that returns are not IID random variables does not provide any conclusions about the random walk and market efficient hypothesis, and so about the predictability of the returns. In order to test this hypothesis we can use the Cochrane variance ratio test, in which, under the null hypothesis, the log-price is a martingale and so the return is a martingale difference sequence. Being $y_t = ln(P_t)$, the test is based on:

$$V_k = \frac{1}{k}\frac{var(y_t - y_{t-k})}{var(y_t - \ y_{t-1})}$$

that can be rewritten also as:

$$V_k = 1 + 2\sum_{j=1}^{k-1}\left(\frac{k-1}{k}\right)\rho(j)$$

and if the null hypothesis is true, V_k should be 1 because of the zero correlation at any past lags that a MDS should display. The test statistic used is:

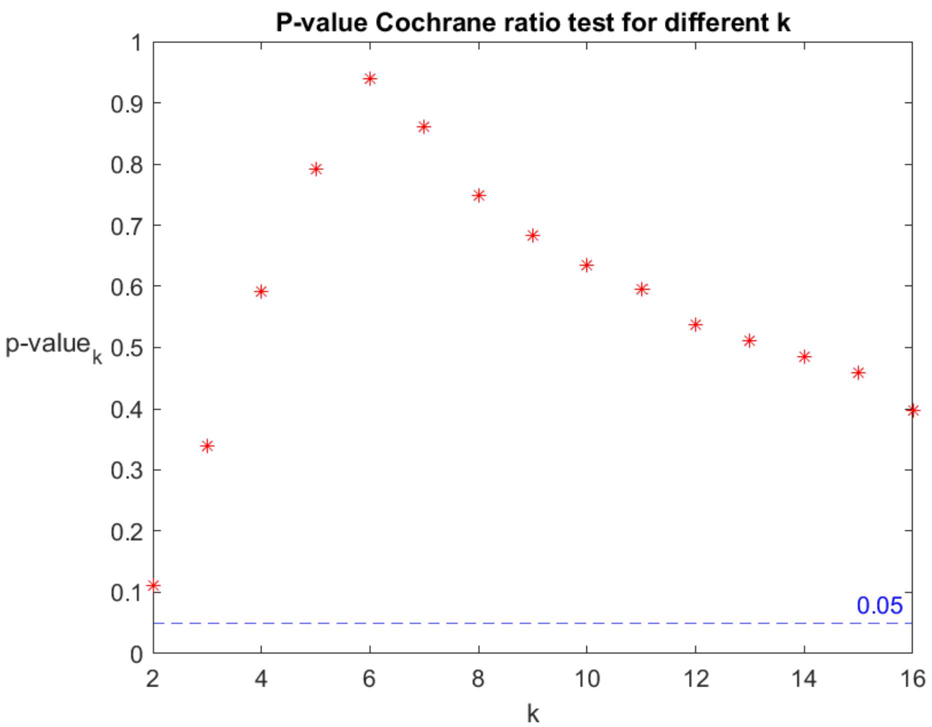
$$\frac{\widehat{V_k} - 1}{SE(\widehat{V_k})}$$

where the asymptotic standard error of the variance ratio test, under the assumption that Δy_t is a martingale difference sequence, is

$$SE(\widehat{V_k}) = [4\sum_{m=1}^{k-1}\frac{1-m}{k})^2\,\widehat{\gamma_2}(m)]^{1/2}$$

where $\widehat{\gamma_2}(m)$ is the ACF at lag m of $(\Delta y_t)^2$.
Under H_0, the statistic is distributed as a standard normal distribution. K should be equal to $n^{1/3}$, where n is the number of observations that in our case is 4937, so k = 17.

In the figure below we can observe how the p-value changes at different k.



The p-values vary with k but no value is lower than the 5% in order to reject the null hypothesis, so our conclusions do not change overall, and we can state that the log price is a martingale.
To account for the phenomena of volatility clustering that returns display, we can use the class of autoregressive conditional heteroskedastic models.
In this kind of model the conditional variance, which we denote by h_t, depends on the last p squared returns, so large past squared returns imply a large conditional variance.
In a $ARCH(p)$ model, the conditional variance h_t is defined as:

$$h_t = \alpha_0 + \alpha_1 y_{t-1}^2 + \ldots + \alpha_p y_{t-p}^2$$

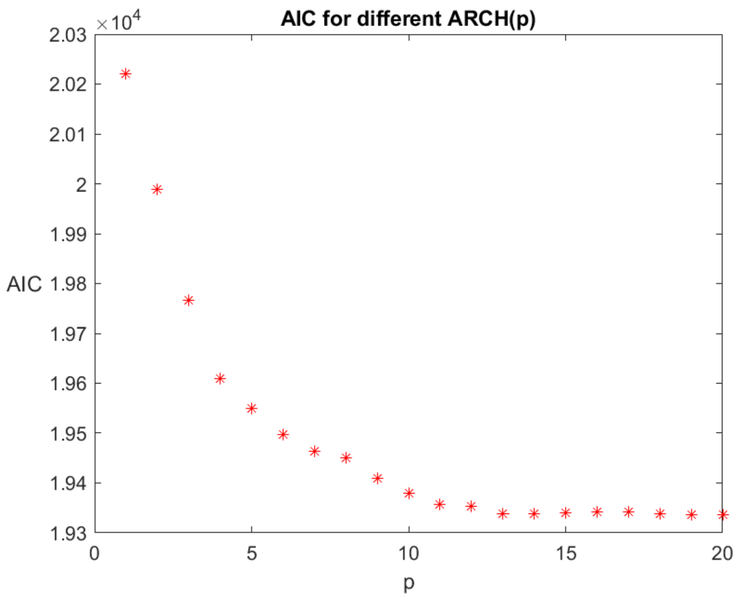
and to ensure that h_t is greater than 0 we need to impose restriction such as $\alpha_0 > 0$ and $\alpha_i \geq 0$ with $i = 1, \ldots, p$.
Assuming a conditional mean equal to zero , we can model the returns as $y_t = \sqrt{h_t}\epsilon_t$, where ϵ_t is an IID process with mean zero and unit variance.
In Gaussian ARCH model the conditional distribution of the returns to all the past information is normal and ϵ_t is a standard normal.

$$\mathrm{y}_t | \mathrm{Y}_{t-1} \sim \mathrm{N}(0, \mathrm{h}_t)$$

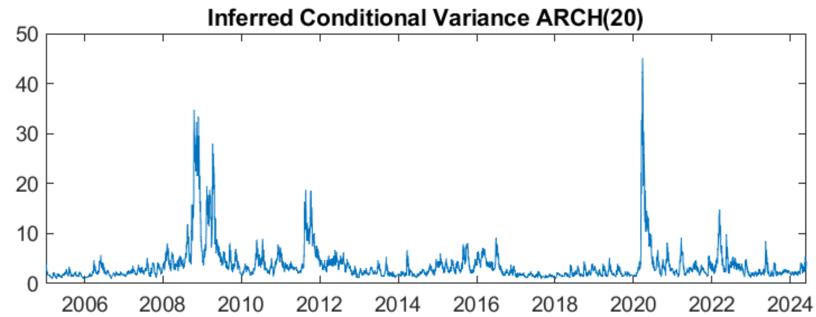
To choose the order p of the model we consider the Akaike Information Criterion (AIC), a function which balances the trade-off between the bias and the variance.
We are going to compute it for each ARCH(p) model and choose the model with the lowest AIC value.
The criterion is defined as:

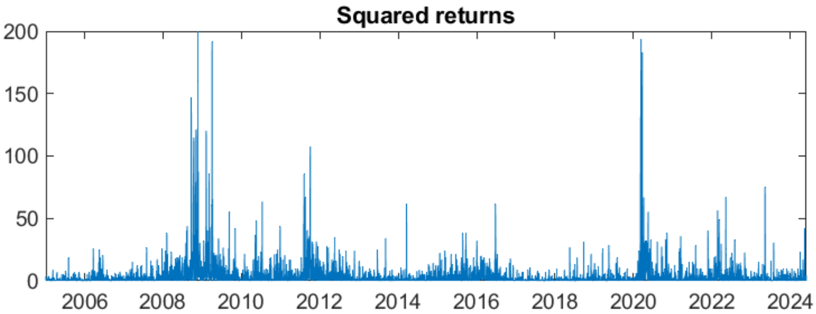
$$\mathrm{AIC} = 2 * (\mathrm{p} + 1) - 2 * \log(\mathrm{L})$$

where L is the likelihood function of the model.
The model which minimizes the AIC is the $ARCH(20)$, as we can see in the graph:



In the following graph we observe the conditional variance estimated by the model compared with the squared returns:



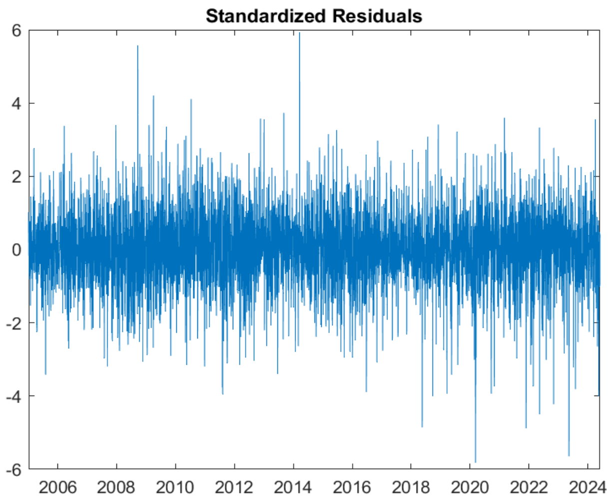


We can see that the model captures the volatility clustering, showing peaks in the period of high volatility occurred during the crisis of the last 20 years which are alternated by periods of low volatility.

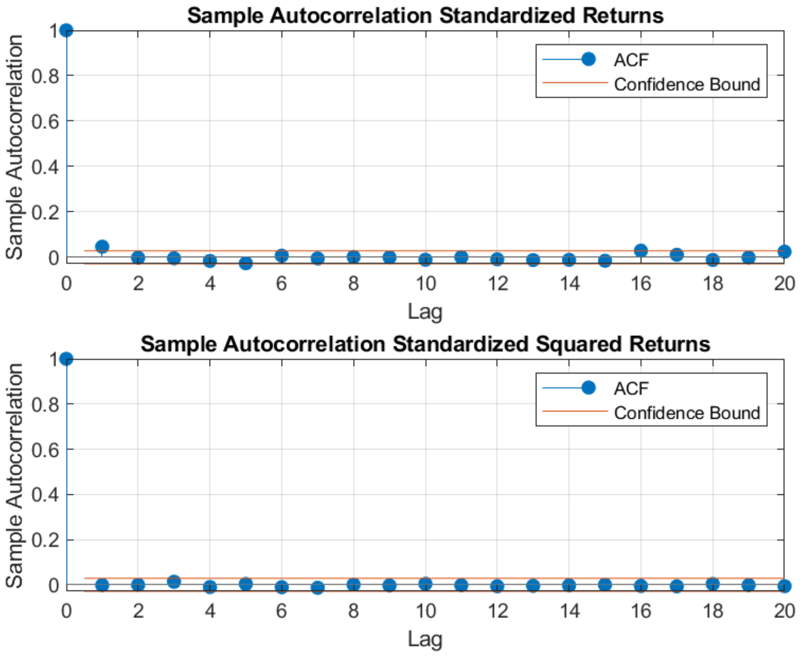
To understand if the model provided a good fit we should consider the standardized residuals:

$$\widehat{\epsilon_t} = \frac{y_t}{\sqrt{\widehat{h_t}}}$$

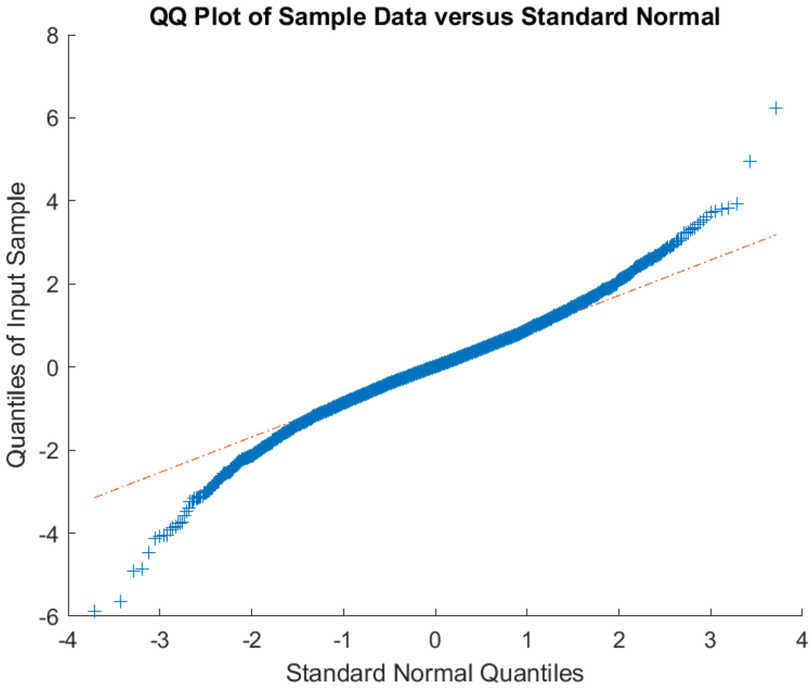
and theoretically they are a realization of an IID process. Observing their behavior they don't show any pattern suggesting that there is no heteroskedasticity left and they are the result of a white noise process.



In order to verify the serial independence of the residuals we check for the autocorrelation:



and we can see that there is no autocorrelation, not even in the squares.
But the residuals should also be generated by a standard normal and so plotting a qq-plot, which compares the quantile of the standard normal distribution with the quantiles of the standardized residuals, we notice that there is a deviation from normality:



The result gives evidence of the fat tails that a gaussian model can not capture .As we would expect, performing a Jarque-Bera test on the standardized residuals we get a value in the rejection region.
The diagnostic checking confirms the validity of the model, capturing the stylized facts of the returns, but there could be improvement, especially to account for fat tail.

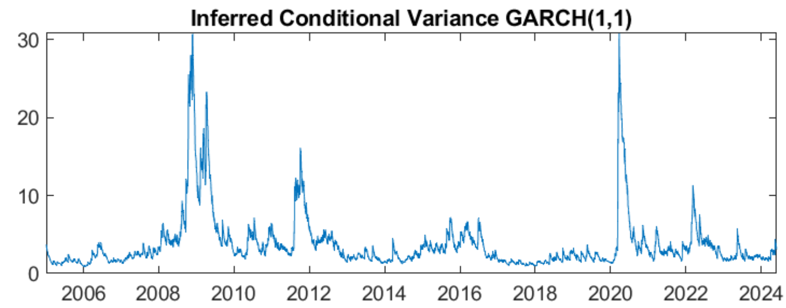
$ARCH(p)$ models where p is a large value are more general but they are difficult to estimate especially for the positivity constraints that we need to ensure for the conditional variance to be positive. A way to simplify this problem is to generalize the class and consider the generalized ARCH models. In a GARCH(p,q) model the conditional variance depends on p past squared returns and on q past values of itself:

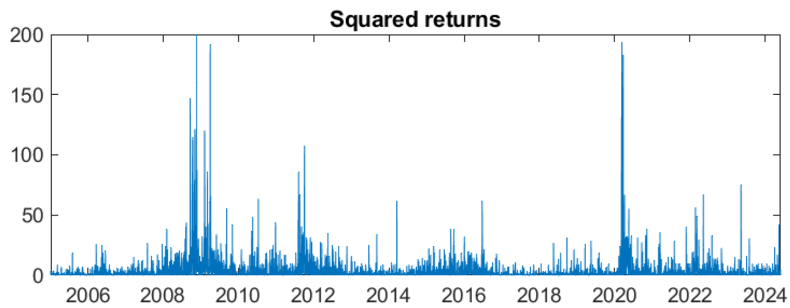
$$h_t = \alpha_0 + \alpha_1 y_{t-1}^2 + \ldots + \alpha_p y_{t-p}^2 + \beta_1 h_{t-1} + \ldots + \beta_q h_{t-q}$$

with constraints $\alpha_0 > 0$, $\alpha_i \geq 0$ with $i = 1, \ldots, p$ and $\beta_j \geq 0$ with $j = 1, \ldots, q$.
Estimating the GARCH(1,1) model, we get the following result :

	Value	Standard Error	TStatistic	Pvalue
$\widehat{\alpha_0}$	0.208656	0.0043889	6.5292	0
$\widehat{\alpha_1}$	0.047917	0.0030765	255.01	0
$\widehat{\beta_1}$	0.94425	0.0037028	15.575	0

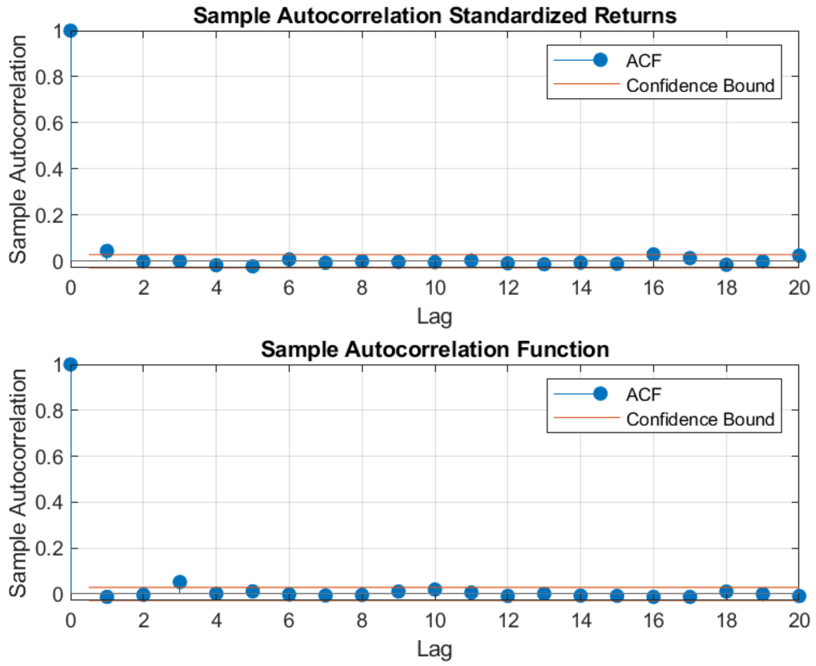
All the parameters are statistically significant and the sum of the coefficients is lower than 1 which indicates that the process is stationary but the value is also near 1, which is the case of the IGARCH model.



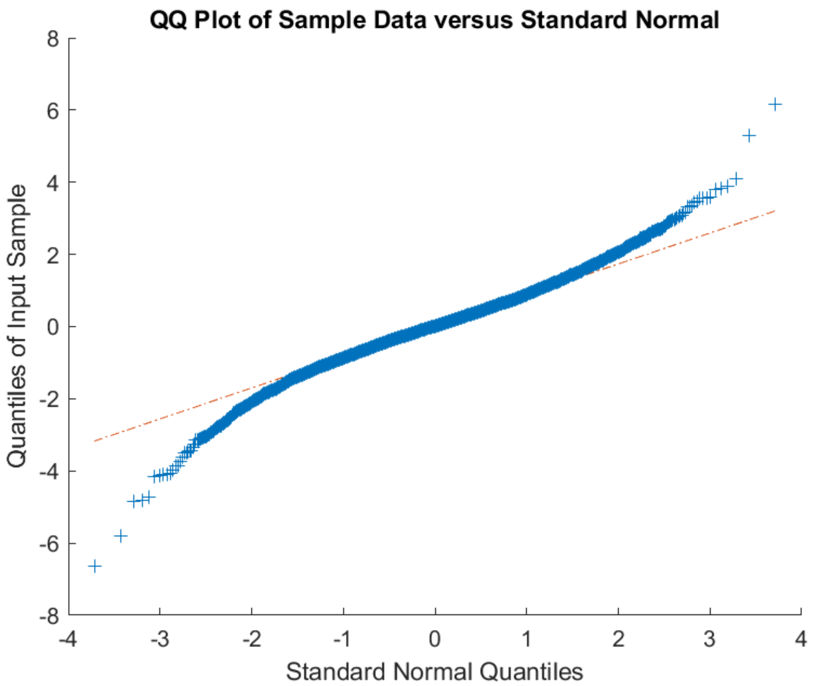


Considering the conditional variance produced by the model, it captures all the periods of high volatility comparing it with the squared returns.
We notice that the conditional variance estimated by the model is smoother and the number of parameters used is lower than the ARCH(20), this simplify the estimation and the

identification of the positivity constraints, so the GARCH seems to be preferable with respect to the ARCH.



Regarding the residuals analysis, the GARCH has a similar outcome of the ARCH model, indeed there is no autocorrelation neither in the standardized residuals nor in their squares, so the process is IID, but analyzing the qq-plot we have the same deviation in the tails because we are still using a gaussian model.



The final decision in order to choose the better specification is taken considering the AIC:

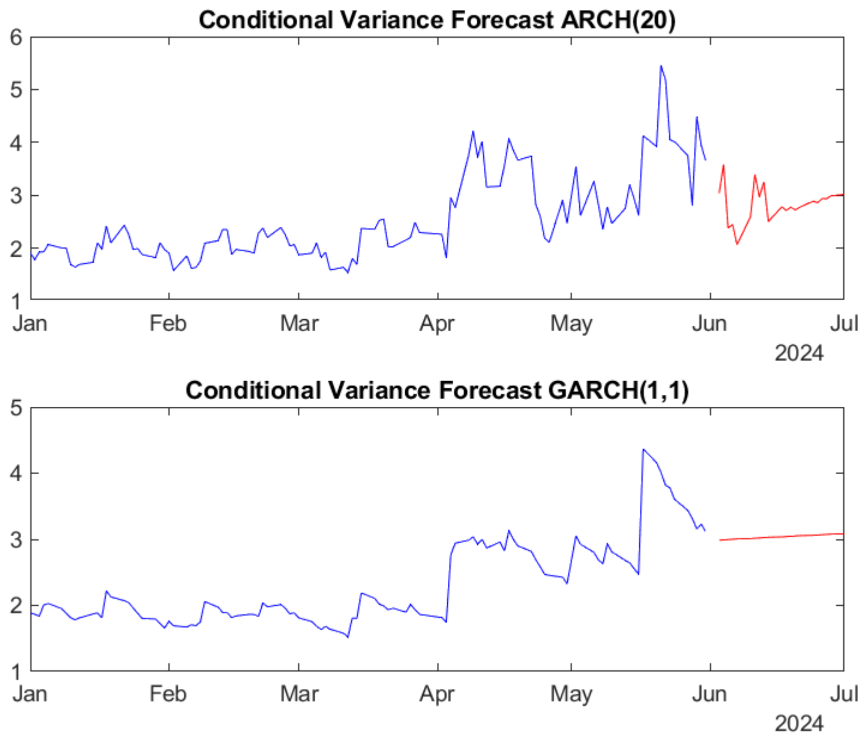
ARCH(20)	19335.8591
GARCH(1,1)	19260.1805

and the GARCH(1,1) minimizes the criterion so it is the preferable model.

The volatility forecasts for the 2 models show different patterns:

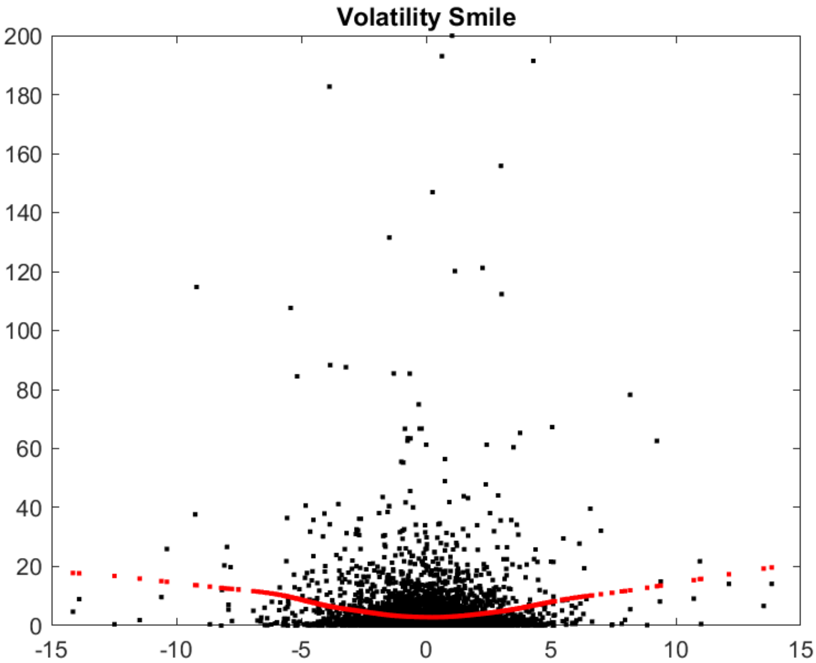
- For the $ARCH(20)$ the forecasts are more volatile in the first part showing ups and downs without a clear trend and then stabilize on an upward trend.

- For the $GARCH(1,1)$ the forecasts are more stable throughout all the period and this is due to the high value of the persistence parameter β_1.

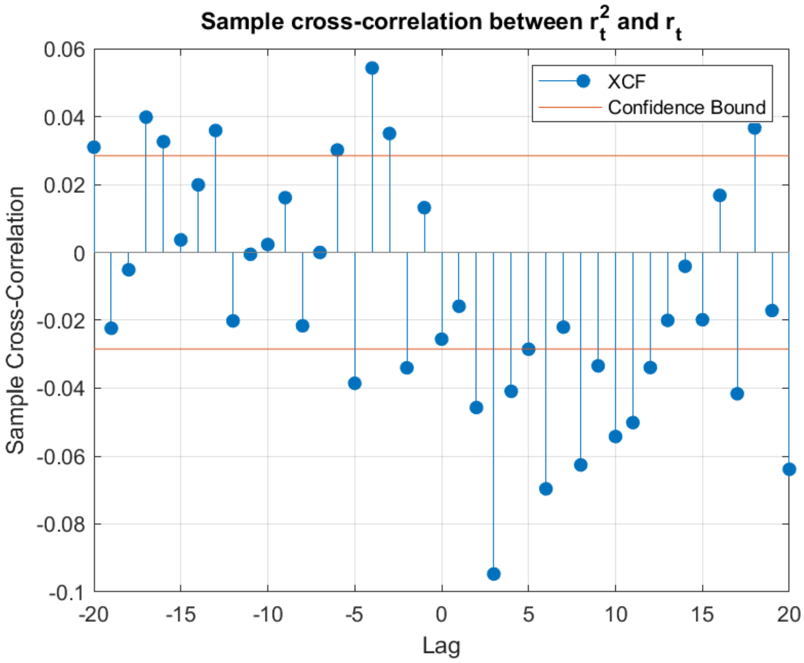


The leverage effect is a phenomena for which the conditional variance reacts asymmetrically to the changes in prices. A negative return triggers a higher volatility in the next day than a positive return.

There are evidence for the leverage such as the asymmetric relationship between y_t^2 and y_{t-1}, called volatility smile:



or the negative cross correlation between past lags and y_t^2:



We can consider two class of models to handle the leverage:

- the GJR-GARCH(1,1) in which we model asymmetric response adding a term in the conditional volatility specification

$$h_t = \alpha_0 + [\alpha_1 + \overline{\alpha_1} I(y_{t-1} < 0)]\, y_{t-1}^2 + \beta_1 h_{t-1}$$

so if the return is negative we have an higher response of the volatility the next day than a positive return, given by $\overline{\alpha_1}$.

- the E-GARCH(1,1) in which we model the log-volatility, so there is no need for the positivity constraints

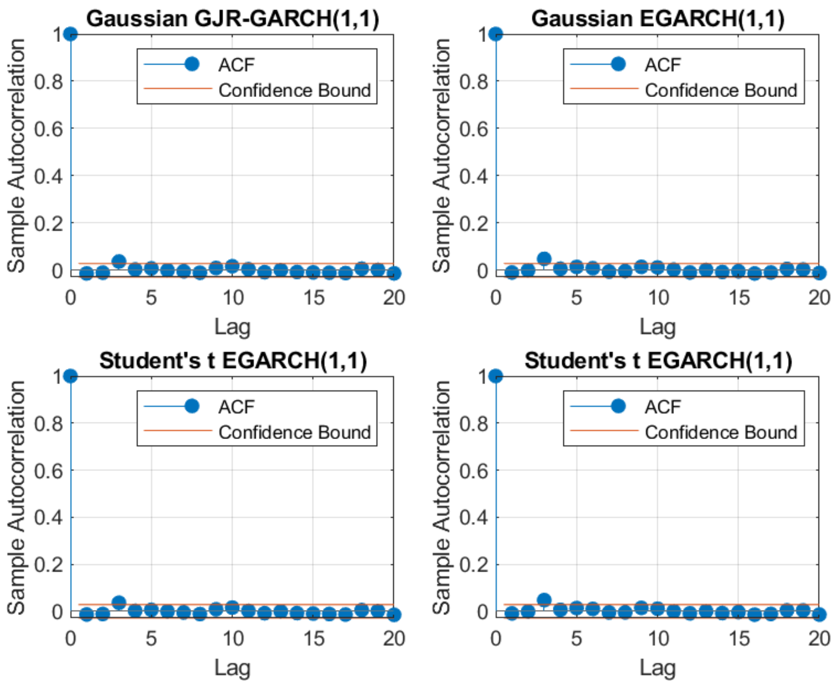
$$ln(h_t) = \alpha_0(1-\alpha_1) + \alpha_1 ln(h_{t-1}) + \theta\epsilon_{t-1} - \gamma\sqrt{\frac{2}{\pi}} + \gamma|\epsilon_{t-1}|$$

the parameter θ accounts for the leverage and we expect it to be negative in order to increase the log-volatility if we have a negative return; γ accounts for the size effect.

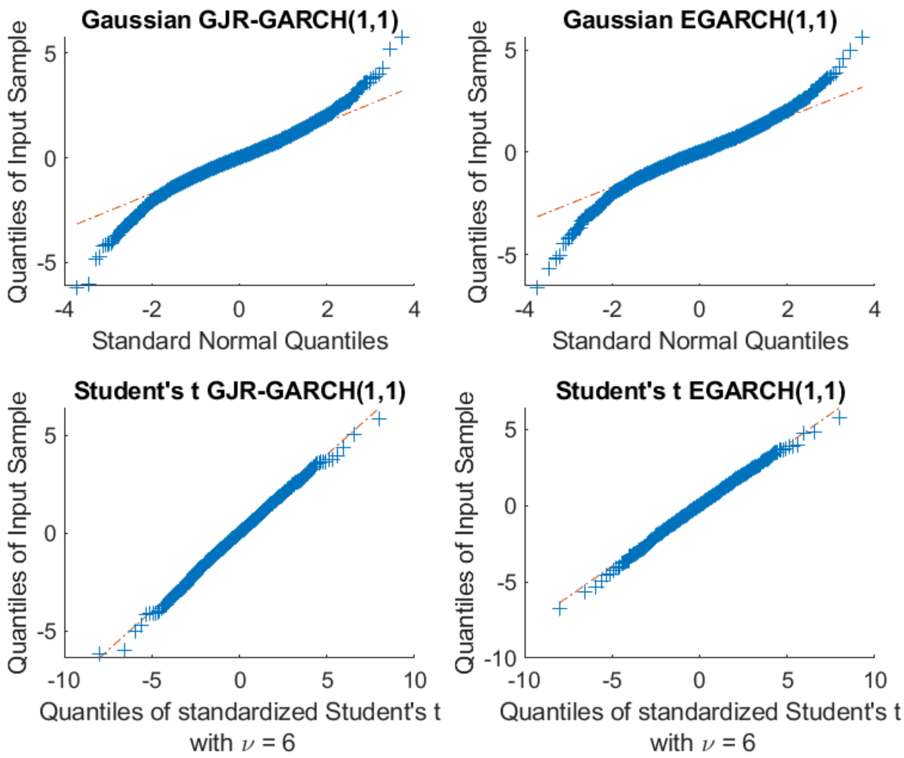
We are going to fit both the models with gaussian and student's t errors in order to account also for the fat tail. When we consider a student's t model we have another parameter to estimate, namely the degrees of freedom of the distribution. For the student's t GJR and EGARCH the number estimated is 6.
The leverage coefficient is statistically significant in all the 4 models.

TStatistic	Gaussian GJR	Gaussian EGARCH	Student's t GJR	Student's t EGARCH
leverage	8.6286	-9.8875	4.5474	-5.3644

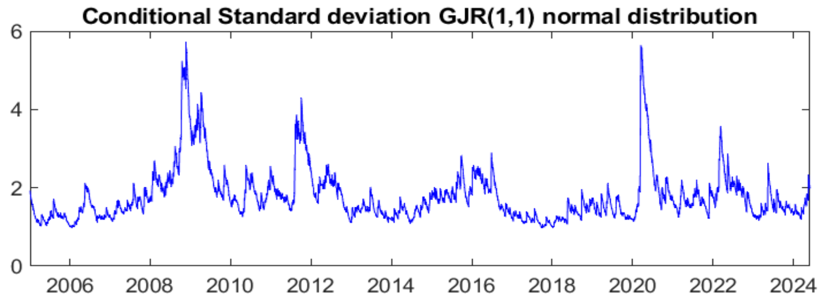
So there is evidence for the leverage.
Considering the diagnostic checking, all the standardized residuals show no autocorrelation in the squares.

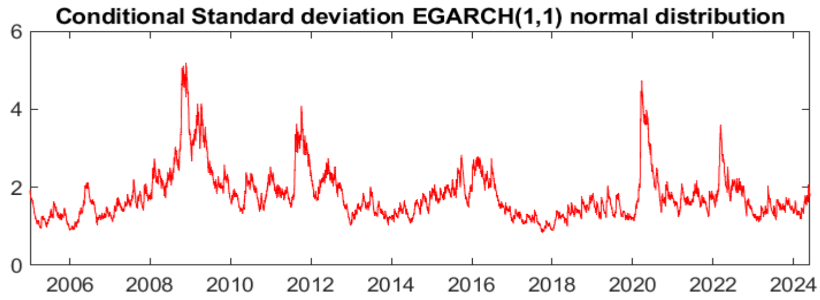


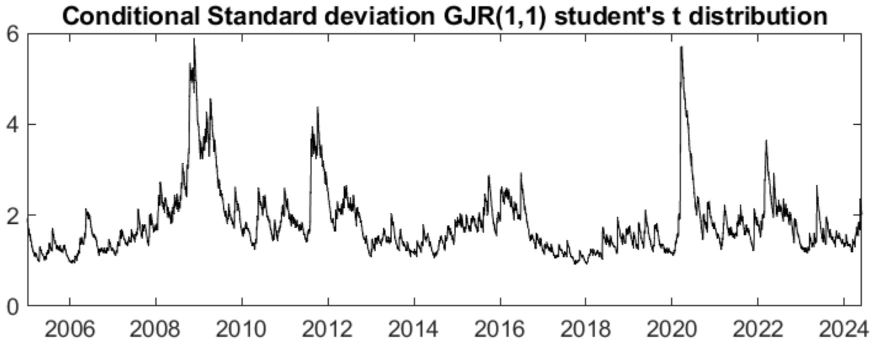
and considering the qq-plot for the 4 models, we notice that for the student's t models the quantiles of the residuals suits the quantiles of a standardized student's t distribution with 6 degrees of freedom (estimated in the student’s t GJR and EGARCH), so considering a fat tailed distribution improve the analysis.

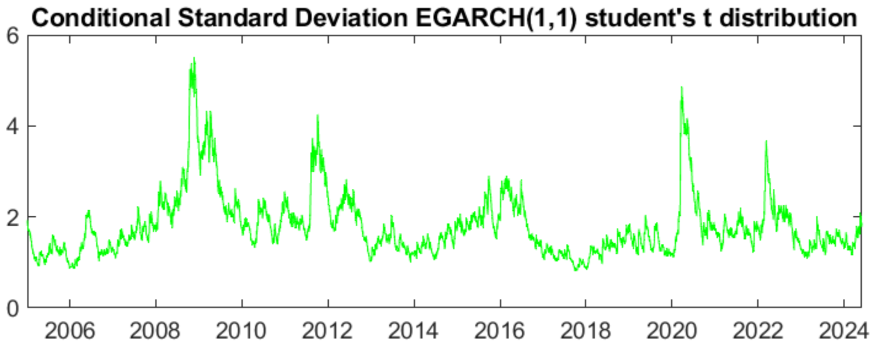


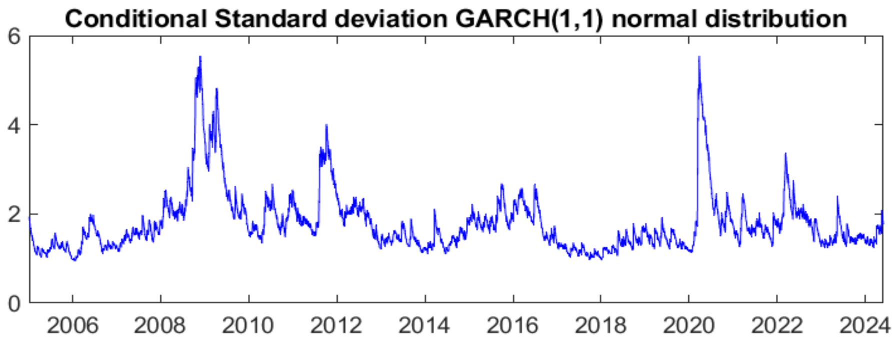
Considering the estimated standard deviations of the 4 models and the one of the GARCH(1,1) specification, the results are very similar, indeed all the models capture the volatility clustering , but considering the leverage improved the representation of the volatility in the crisis period. About the 2 specifications for the asymmetric response of the volatility, the EGARCH estimates display a smoother transition from period of high movement to the low ones.











In our opinion the suitable specification for our time series should account for the leverage effect because we have found evidence for the phenomena and should have a standardized Student's t distribution for the error term because we have seen in the diagnostic checking that the quantiles of the residuals fitted the quantiles of the distribution for the Student's t models. So our final choice is the Student's t EGARCH(1,1) because of the smoother estimation of the conditional variance compared to the Student's t GJR-GARCH(1,1).

```matlab
%% IMPORTING AND CLEANING DATA
clc
clear
close

rng(0349063);
idx=randsample(30,1);
Euro50=readtable("EuroStoxx50.xlsx");
stock=Euro50{idx,1};
asset=readtable(append(stock{1},".csv"),"Range","1307:6244");
asset.Properties.VariableNames([1 5])={'Dates' 'Close'};

p1=asset.Close;
figure(1)
dates=asset.Dates(2:end);
plot(asset.Dates,p1)
title("BMW price")
p1=rmmissing(p1);
r=diff(log(p1))*100;
figure(2),
plot(dates,r)
title("BMW returns")

%% CHARACTERISTICS OF RETURNS
figure(3),
autocorr(r)
title("Returns Autocorrelation")
subplot(2,1,1),autocorr(abs(r))
title("Absolute Returns Autocorrelation")
subplot(2,1,2),autocorr(r.^2)
title("Squared Returns Autocorrelation")

n=length(r);
m1= mean(r);
z=r-m1;
m2= mean(z.^2);
z=normalize(r);
m3= mean(z.^3);
m4= mean(z.^4);
JB = n * (m3^2+(m4-3)^2/4)/6 ;
```

```
figure(4),histfit(r)

%% COCHRANE VARIANCE RATIO TEST
y=log(p1);
n=length(y);
maxK = floor(n^(1/3));
kvalues = 2:maxK;
K = length(kvalues);
[~, pValue, stat, ~, VR] = vratiotest(y,'period',kvalues,'IID', false(1,K));

figure(5),plot(kvalues, pValue, 'r*')
title("P-value Cochrane ratio test for different k")
xlabel("k")
ylabel("p-value_{k}","Rotation",0)
yline(0.05,"b--","0.05")


%% CHOICE OF THE ORDER p
P=20;
logL=zeros(P,1);
AIC=zeros(P,1);
for i=1:P
     Mdl = garch(0,i);
     [~,~,logL(i)] = estimate(Mdl,r);
    AIC(i)=2*(i+1)-2*logL(i);
end
[minAIC,idx]=min(AIC);
numP= idx;
figure(6),plot(1:P,AIC,"r*")
xlabel("p")
ylabel("AIC","Rotation",0)
title("AIC for different ARCH(p)")


%% FIT ARCH(20)

Mdl = garch(0,numP);
EstMdl = estimate(Mdl,r);
cond_var_arch=infer(EstMdl,r);
cond_var_arch_for=forecast(EstMdl,22,r);
```

```
figure(7),
subplot(2,1,1),plot(dates,cond_var_arch)
title("Inferred Conditional Variance ARCH(20)")
hold on
subplot(2,1,2),plot(dates,r.^2)
title("Squared returns")
hold off


%% DIAGNOSTIC CHECKING OF ARCH(20)
standRes=r./sqrt(cond_var_arch);
figure(8),
plot(dates,standRes)
title("Standardized Residuals")
figure(9),
subplot(2,1,1),
autocorr(standRes)
title("Sample Autocorrelation Standardized Returns")
subplot(2,1,2),
autocorr(standRes.^2)
title("Sample Autocorrelation Standardized Squared Returns")
figure(10),
qqplot(standRes)

n_e=length(standRes);
m1_e= mean(standRes);
z_e=standRes-m1_e;
m2_e= mean(z_e.^2);
z_e=normalize(standRes);
m3_e= mean(z_e.^3);
m4_e= mean(z_e.^4);
JB_e = n_e * (m3_e^2+(m4_e-3)^2/4)/6 ;


%% FIT GARCH(1,1)

Mdl_asset1 = garch(1,1);
[EstMdl_2,~,logL_garch] = estimate(Mdl_asset1,r);
cond_var_garch= infer(EstMdl_2,r);
cond_var_garch_for=forecast(EstMdl_2,22,r);
```

```
AIC_garch=2*3-2*logL_garch;

figure(11),
subplot(2,1,1),plot(dates,cond_var_garch)
title("Inferred Conditional Variance GARCH(1,1)")
hold on
subplot(2,1,2),plot(dates,r.^2)
title("Squared returns")
hold off

%% RESIDUALS ANALYSIS GARCH(1,1)
standRes_garch=r./sqrt(cond_var_garch);
figure(12),
subplot(2,1,1),autocorr(standRes_garch)
title("Sample Autocorrelation Standardized Returns")
subplot(2,1,2),autocorr(standRes_garch.^2)
figure(13),
qqplot(standRes_garch)



%% COMPARING CONDITIONAL VARIANCE FORECASTS
dates_for=[datetime(2024,06,3:7) datetime(2024,06,10:14) datetime(2024,06,17:21) datetime(2024,06,24:28) datetime(2024,07,1:2)];
figure(14),
subplot(2,1,1),plot(dates,cond_var_arch,"-b")
hold on
plot(dates_for,cond_var_arch_for,"r")
xlim([datetime(2024,01,01) datetime(2024,07,01)])
title("Conditional Variance Forecast ARCH(20)")
hold off
subplot(2,1,2),plot(dates,cond_var_garch,"-b")
hold on
plot(dates_for,cond_var_garch_for,"r")
xlim([datetime(2024,01,01) datetime(2024,07,01)])
title("Conditional Variance Forecast GARCH(1,1)")
hold off

%% LEVERAGE EVIDENCE
fit= smooth(r(1:end-1), r(2:end).^2, 0.95, 'loess');
```

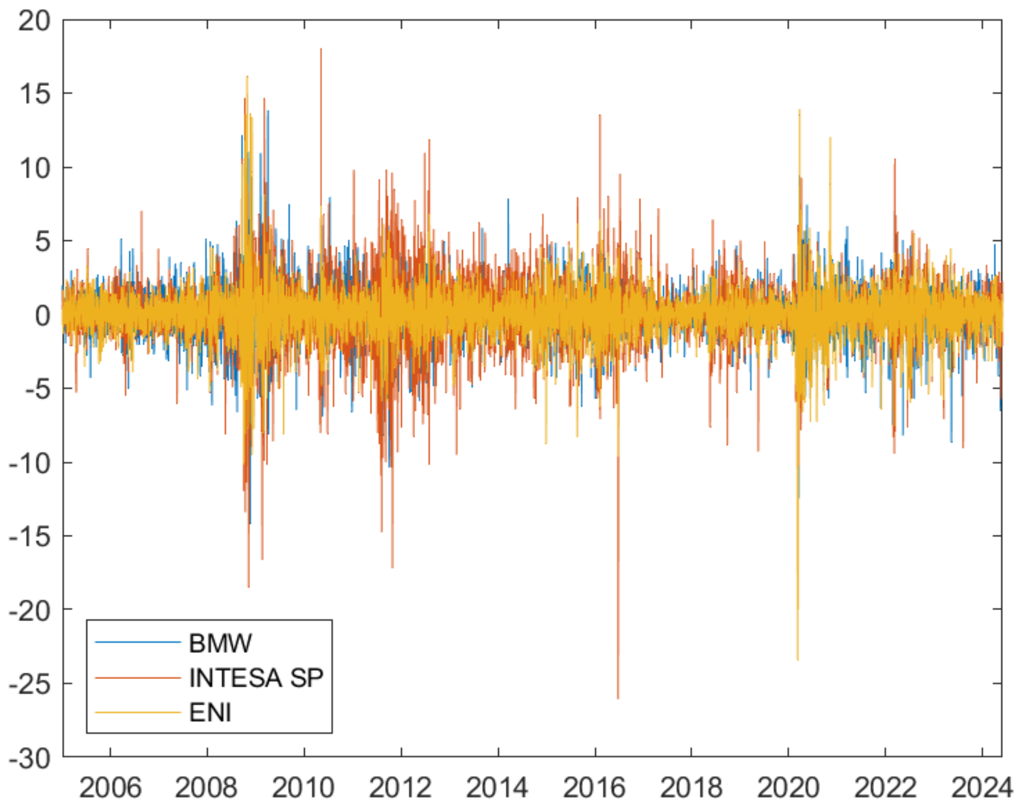
```
figure(15),
crosscorr(r, r.^2)
title("Sample cross-correlation between r_{t}^2 and r_{t}")

figure(16),
plot(r(1:end-1), r(2:end).^2, 'k.', r(1:end-1), fit, 'r.')
title("Volatility Smile")

%% FIT MODELS FOR LEVERAGE
EstMdl_gjr_n = estimate(gjr(1,1), r);
ht1=infer(EstMdl_gjr_n,r);

EstMdl_egarch_n = estimate(egarch(1,1), r);
ht2=infer(EstMdl_egarch_n,r);

Mdl_gjr = gjr(1,1);
Mdl_gjr.Distribution = "t";
EstMdl_gjr = estimate(Mdl_gjr, r);
ht3=infer(EstMdl_gjr,r);

Mdl_Egar = egarch(1,1);
Mdl_Egar.Distribution = "t";
EstMdl_egarch = estimate(Mdl_Egar, r);
ht4=infer(EstMdl_egarch,r);

%% DIAGNOSTIC CHECKING LEVERAGE MODELS
figure(17),
subplot(2,2,1),
autocorr((r./sqrt(ht1)).^2)
title("Gaussian GJR-GARCH(1,1)")
subplot(2,2,2),
autocorr((r./sqrt(ht2)).^2)
title("Gaussian EGARCH(1,1)")
subplot(2,2,3),
autocorr((r./sqrt(ht3)).^2)
title("Student's t EGARCH(1,1)")
subplot(2,2,4),
autocorr((r./sqrt(ht4)).^2)
title("Student's t EGARCH(1,1)")
```

```
figure(18),
subplot(2,2,1),
qqplot(r./sqrt(ht1))
title("Gaussian GJR-GARCH(1,1)")
subplot(2,2,2),
qqplot(r./sqrt(ht2))
title("Gaussian EGARCH(1,1)")
subplot(2,2,3),
pd=makedist("tLocationScale","mu",0,"sigma",1,"nu",6);
qqplot(r./sqrt(ht3),pd)
title("Student's t GJR-GARCH(1,1)")
xlabel(["Quantiles of standardized Student's t" ;" with \nu = 6"])
subplot(2,2,4),
qqplot(r./sqrt(ht4),pd)
title("Student's t EGARCH(1,1)")
xlabel(["Quantiles of standardized Student's t" ;" with \nu = 6"])

%% CONDITIONAL STANDARD DEVIATION COMPARISON
figure(19),
subplot(2,1,1),plot(dates,sqrt(cond_var_garch),"-b")
title("Conditional Standard deviation GARCH(1,1) normal distribution")

figure(20),
subplot(2,1,1),plot(dates,sqrt(ht1),"-b")
title("Conditional Standard deviation GJR(1,1) normal distribution")

subplot(2,1,2),plot(dates,sqrt(ht2),"-r")
title("Conditional Standard deviation EGARCH(1,1) normal distribution")

figure(21),
subplot(2,1,1),plot(dates,sqrt(ht3),"-k")
title("Conditional Standard deviation GJR(1,1) student's t distribution")

subplot(2,1,2),plot(dates,sqrt(ht4),"-g")
title("Conditional Standard Deviation EGARCH(1,1) student's t distribution")
```

Multivariate analysis

The series under investigation in the multivariate analysis, using the matricola number 0349063, are:

- BMW , operating in the car sector
- Intesa SanPaolo, operating in the banking sector
- ENI , operating in the energy sector



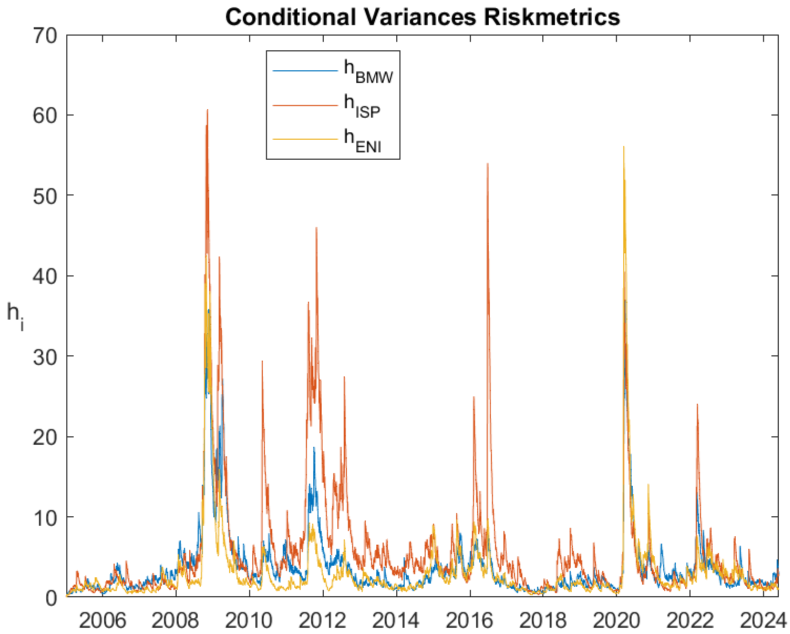
Looking at the returns, we notice that BMW and ENI seem to be more stable than Intesa SanPaolo which tends to oscillate more, but at the same time BMW shows less extreme values than the other 2 stocks and it is all reflected in their values for the sample variance and kurtosis:

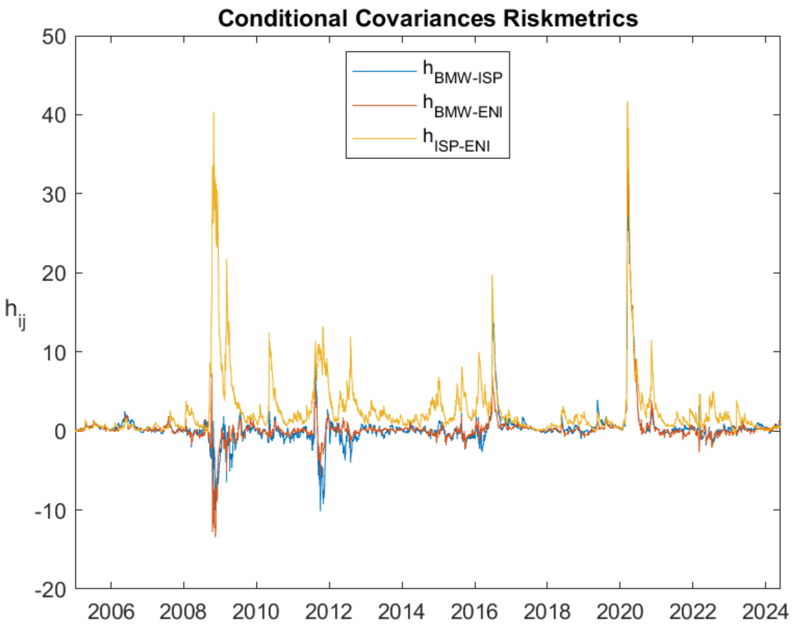
	Sample Variance	**Sample Kurtosis**
BMW	3.7183	8.6460
Intesa SanPaolo	5.8282	12.6799
ENI	3.0879	20.0530

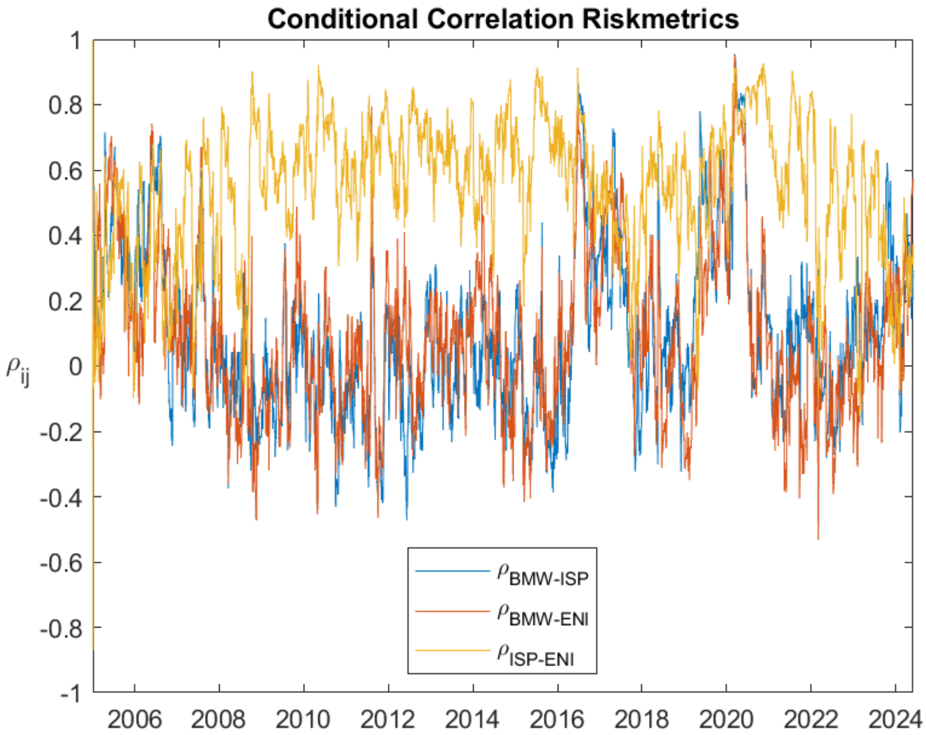
In this section we are going to analyze the co-movements of the 3 stocks, in the first part with multivariate GARCH models and in the second with the copula. Initially we use three different models to estimate the conditional variance-covariance matrix H_t and the conditional correlation matrix P_t of the vector of the observations Y_t .
Each model has its own assumption and estimates the matrices in different ways.
The first is a simple method called the Riskmetrics in which the conditional variance covariance matrix H_t is modeled as an exponential weighted moving average with a fixed constant λ equal to 0.06 :

$$H_t = \lambda Y_{t-1} Y'_{t-1} + (1-\lambda) H_{t-1}$$

This is a simple model which can be used as a benchmark to compare the other 2 models.







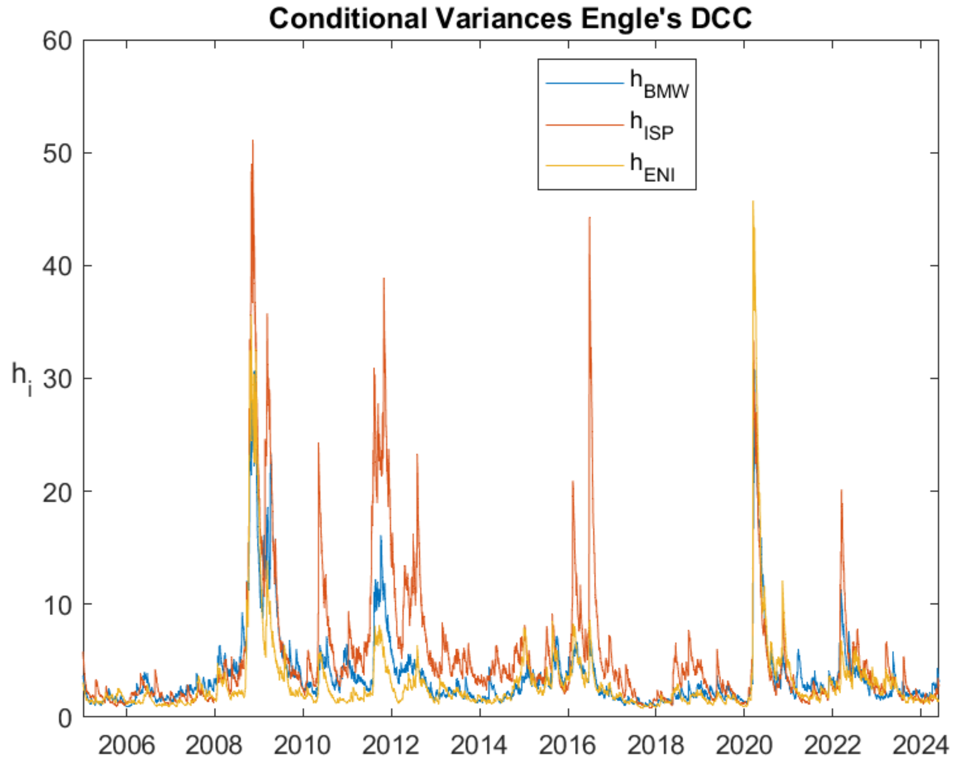
The second model used is Engle's DCC , which is specified in terms of the conditional correlation matrix P_t from which we are going to recover H_t as $H_t = D_t P_t D_t$ where D_t is the diagonal matrix containing the conditional standard deviation of each asset, which follows a GARCH(1,1) specification.
In Engle's model P_t is allowed to vary over time and the specitification does not directly model the conditional correlation matrix, but a covariance conditional matrix Q_t which we rescale to obtain P_t . Q_t is specified as:

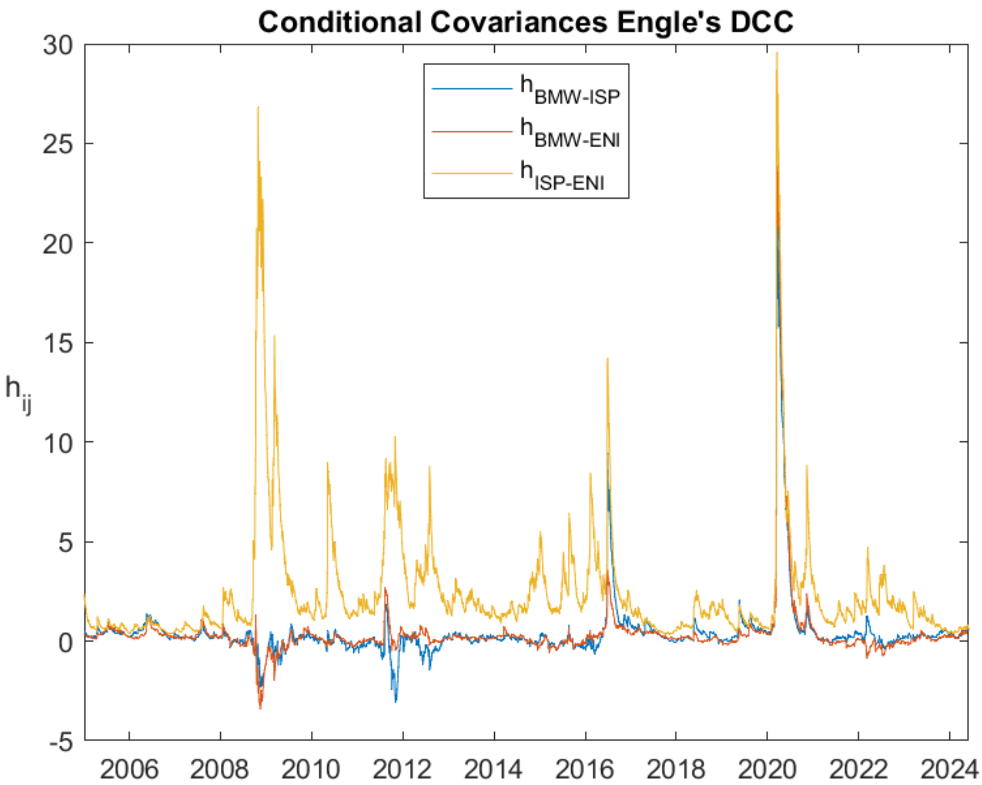
$$Q_t = (1 - a - b)\bar{Q} + aY_{t-1}^* Y_{t-1}^{*\prime} + bQ_{t-1}$$

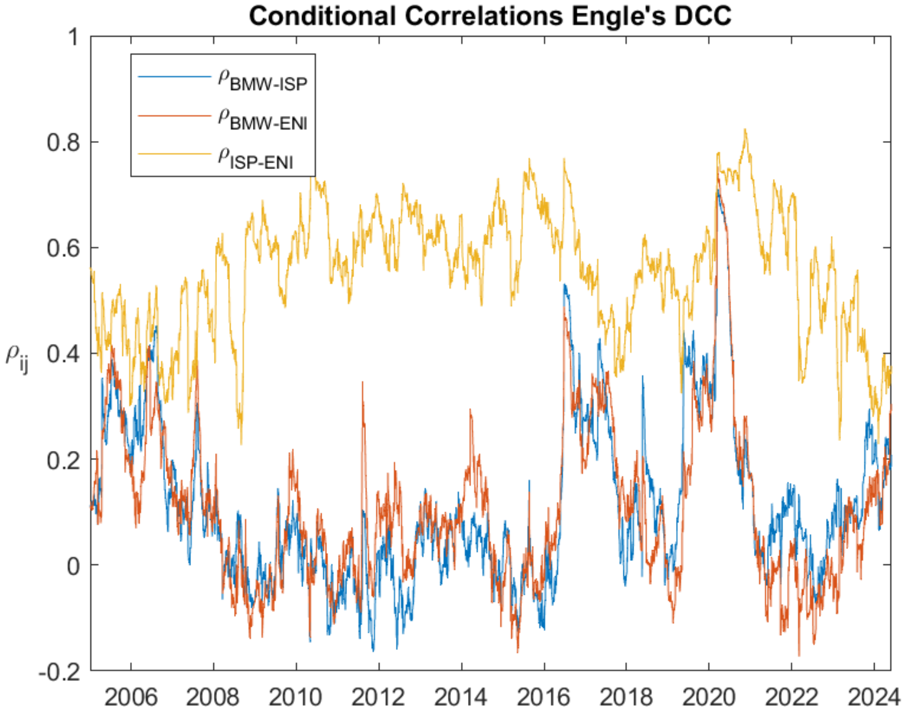
where $\bar{Q}$ is the unconditional covariance matrix of the standardized residuals $Y_{t-1}^* = D_{t-1}^{-1} Y_{t-1}$ and a+b<1. Due to the factorization of H_t the likelihood can be divided in 2 parts and this allows us to estimate the model in 3 simple steps:

- estimate the GARCH(1,1) models for each asset and compute the standardized residuals
- maximize the partial log-likelihood w.r.t. a and b.
- estimate $\bar{Q}$ with the method of moments ,so $\bar{Q} = \frac{1}{n}\sum_t Y_t^* Y_t^{*\prime}$,known also as variance targeting.

In our code in order to estimate a and b we have considered as starting values the ones for the Riskmetrics approach, so a = 0.06 and b = 0.094, and we have got as final result $\hat{a} = 0.0160$ and $\hat{b} = 0.9782$, so the persistence is higher than in Riskmetrics.


Conditional Variances Engle's DCC
h_{BMW}
h_{ISP}
h_{ENI}
h_i
60
50
40
30
20
10
0
2006
2008
2010
2012
2014
2016
2018
2020
2022
2024


Conditional Covariances Engle's DCC
$h_{BMW\text{-}ISP}$
$h_{BMW\text{-}ENI}$
$h_{ISP\text{-}ENI}$
h_{ij}
30
25
20
15
10
5
0
-5
2006
2008
2010
2012
2014
2016
2018
2020
2022
2024



The third model is the Orthogonal GARCH, also called the O-GARCH, because is based on the principal components analysis, a method which performs an orthogonalization of the observation using the eigenvectors of the spectral decomposition of the unconditional covariance matrix of Y_t.
The principal components f_t are obtained as:

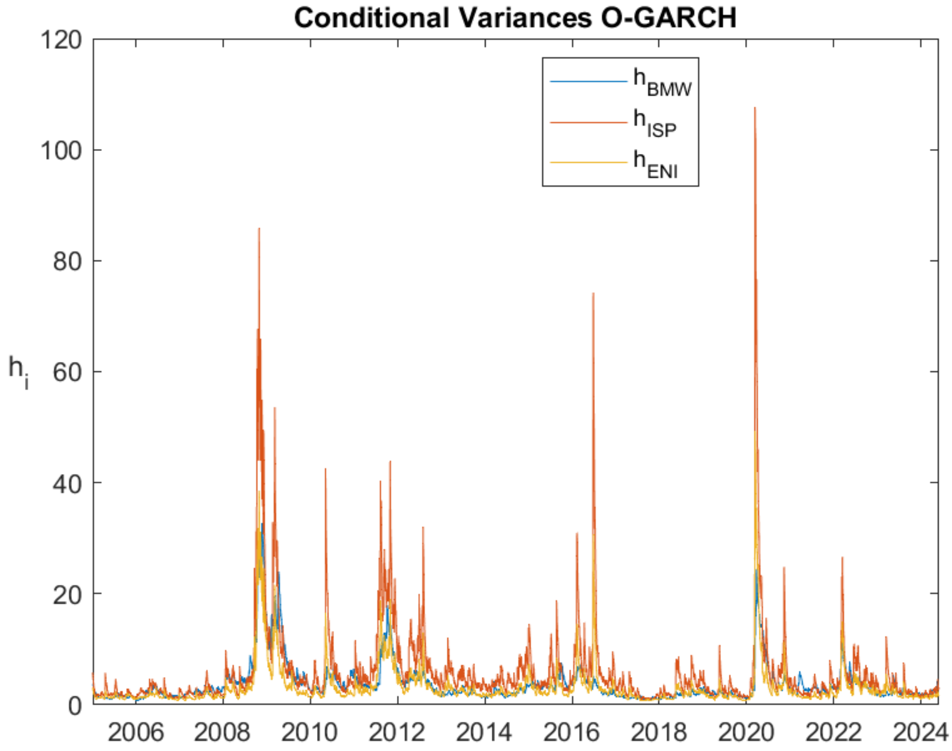
$$f_t = V'Y_t$$

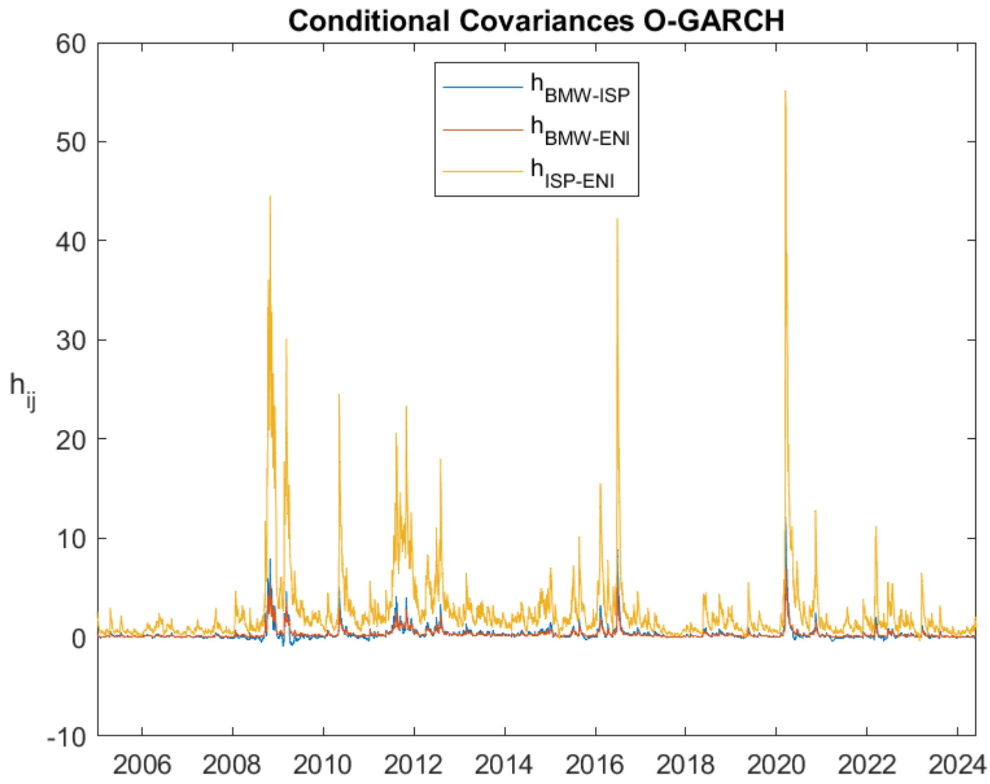
where V is the orthogonal matrix of the spectral decomposition ($S = V\Lambda V'$).
The assumption of the model is that the principal components are uncorrelated not only unconditionally but also conditionally in order to have the conditional variance-covariance matrix of the components diagonal:

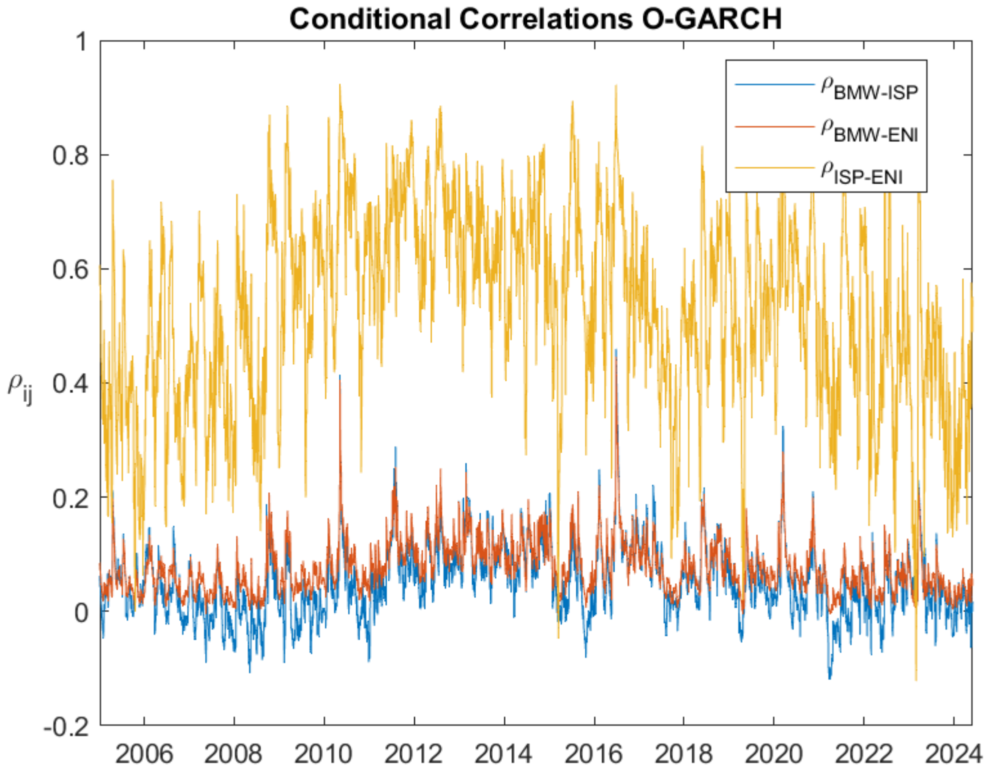
$$H_t^f = diag(\, h_{1,t}^f \,, \ldots, h_{N,t}^f \,)$$

At this point we recover the conditional covariance matrix of the original variables as:

$$H_t \;\; = V H_t^f V'$$


Conditional Variances O-GARCH
h_i
h_BMW
h_ISP
h_ENI
0
20
40
60
80
100
120
2006
2008
2010
2012
2014
2016
2018
2020
2022
2024


Conditional Covariances O-GARCH
h_ij
h_BMW-ISP
h_BMW-ENI
h_ISP-ENI
-10
0
10
20
30
40
50
60
2006
2008
2010
2012
2014
2016
2018
2020
2022
2024



Comparing the graphical results of the three models, it appears clear that Engle's model provides the best estimation because we have smooth and less chaotic paths.
Especially comparing the conditional correlations in the DCC the path is more clear, whereas in the other two we have more swings.
All the models show in the conditional correlations and co-volatilities the strong link between ENI and Intesa SanPaolo which reaches peaks during the financial crisis period, the sovereign debt crisis - which was very relevant in Italy - and the pandemic where, according to the Engle’s model, the conditional correlation reached a value of 80%.
The orthogonal GARCH is the only model which does not show a negative peak in the covariance between BMW and ENI and BMW and Intesa San Paolo during the great financial crisis. For the sovereign debt crisis, the DCC and the Riskmetrics show an initial positive relationship between BMW and the two italian companies because of the first period of the shock which hits all Europe, but Germany recovers faster than Italy and indeed the models show a negative peak in the correlation and covariance for BMW and ISP.
Once again, the O-GARCH seems not to capture the negative relationship between the 2 companies during the period taken into consideration.

The copula represents an alternative way to model the dependence between assets.
From a technical point of view, the copula is the joint distribution of random variables with a marginal distribution which is standard uniform. The main advantage of the copula consists in the fact that it allows us to model the dependence structure - explained in the copula - independently from the marginal behavior.
In practice, as a first step we estimate the marginal behavior of each asset and in our case we use GARCH models.

The distributions of the single assets are considered to be a t-student with degrees of freedom estimated in the GARCH-t model, which produces values:

	BMW	INTESA SP	ENI
Degrees of freedom	5	5	6

Once we have values for the parameters, we consider the standardized-t residuals which we obtain as:

$$\epsilon_t = \frac{y_t}{\sqrt{\widehat{h_t}}} * \sqrt{\frac{v}{v-2}}$$

At this point, we apply the cumulative distribution function of the student-t distribution of each asset to get the probability integral transform which we use to fit the copula.
Using the residuals instead of the initial observations simplifies our job because they are result of an IID process.
Now we proceed by estimating a t-copula which returns the estimation of the correlation coefficient and the degrees of freedom estimated for the copula.
The correlations estimated through the copula are synthesized in the following matrix:

	BMW	INTESA SP	ENI
BMW	1	0.0949	0.0919
INTESA SP	0.0949	1	0.5618
ENI	0.0919	0.5618	1

The highest correlation coefficient is the one between Intesa San Paolo and ENI. The reason could be their importance in the italian economy, being among the biggest five companies by market capitalization, and so a shock in the italian stock market would affect the two simultaneously.
The correlation between BMW and ENI is near 10%, which is an intermediate value, probably due to their dependence on electricity (BMW for the production of their cars and ENI as seller).

The tail dependence is a conditional probability and gives us an idea of what is the probability of observing an extreme event in an asset, given that in another asset an extreme event occured.
This event is divided into 2 categories, the lower tail dependence and the upper tail dependence. Not all the copulas can be used to account for this situation, for example the Gaussian one does not consider the tail dependence, as well as the Clayton or Gumbel copula which can be used to estimate respectively only the lower tail or the upper tail.
In our case, the student copula can be used to estimate this value and the response is symmetric in both the tails and it is equal to:

$$2 * t_{v+1}(w)$$

where $w = -\frac{\sqrt{v+1}\sqrt{1-\rho}}{\sqrt{1+\rho}}$ and ρ is the correlation copula.

The estimates are contained in the following matrix:

	BMW	INTESA SP	ENI
BMW	1	0.0014	0.0104
INTESA SP	0.0014	1	0.1363
ENI	0.0104	0.1363	1

The values for the tail dependence are lower with respect to the correlation: this indicates that the extreme events are mostly unrelated between the assets.
For ENI and Intesa San Paolo, the value is more than 10% and the main reason could be that a large shock in the Italian market would affect both the companies producing extreme events at the same time.

```
%% IMPORTING AND CLEANING THE DATA

rng(0349063);
idx_2=randsample(30,3);
stock_2=Euro50{idx_2,1};
asset_1=readtable(append(stock_2{1},".csv"),"Range","1307:6244");
asset_2=readtable(append(stock_2{2},".csv"),"Range","1307:6244");
asset_3=readtable(append(stock_2{3},".csv"),"Range","1307:6244");

r_1=diff(log(asset_1.Var5))*100;
r_2=diff(log(asset_2.Var5))*100;
r_3=diff(log(asset_3.Var5))*100;

r_t=[r_1 r_2 r_3];
r_t=rmmissing(r_t);
figure(22),
plot(dates,r_t)
legend(["BMW" "INTESA SP" "ENI"],"location","southwest")

%% CHARACTERISTICS OF THE RETURNS
m1= mean(r_t);
z=r_t-m1;
m2= mean(z.^2);
z=normalize(r_t);
m3= mean(z.^3);
m4= mean(z.^4);
```

```matlab
%% RISKMETRICS APPROACH
lambda=0.06;
mh_ii    = NaN(length(r_t),3);
mh_ij    = [];
mrho_ij    = [];
for i = 1:3
    r_i =  r_t(:,i);
    h_ii = filter(1, [1 -(1-lambda)], lambda * r_i.^2);
    mh_ii(:,i) = h_ii;
    for j = i+1:3
         r_j =  r_t(:,j);
         h_jj = filter(1, [1 -(1-lambda)], lambda * r_j.^2);
         h_ij = filter(1, [1 -(1-lambda)], lambda * r_i.*r_j);
         rho_ij =  h_ij ./(sqrt(h_ii.*h_jj));
         mh_ij = [mh_ij, h_ij] ;
         mrho_ij = [mrho_ij, rho_ij];

    end
end
figure(23),
plot(dates,mh_ii)
title("Conditional Variances Riskmetrics")
legend(["h_{BMW}" "h_{ISP}" "h_{ENI}"],"location","best")
ylabel("h_{i}","rotation",0)

figure(24),
plot(dates,mh_ij)
title("Conditional Covariances Riskmetrics")
legend(["h_{BMW-ISP}" "h_{BMW-ENI}" "h_{ISP-ENI}"],"location","best")
ylabel("h_{ij}","rotation",0)

figure(25),
plot(dates,mrho_ij)
title("Conditional Correlation Riskmetrics")
legend(["\rho_{BMW-ISP}" "\rho_{BMW-ENI}" "\rho_{ISP-ENI}"],"location","best")
ylabel("\rho_{ij}","rotation",0)
```

```matlab
%% Engle's DCC APPROACH
cond_var_asset=zeros(length(r_t(:,1)),3);
stand_r_t=zeros(length(r_t(:,1)),3);
for i=1:3
Mdl_asset1 = garch(1,1);
EstMdl_asset(:,i) = estimate(Mdl_asset1,r_t(:,i));
cond_var_asset(:,i)= infer(EstMdl_asset(:,1),r_t(:,i));
stand_r_t(:,i)= r_t(:,i)./sqrt(cond_var_asset(:,i));
end

% MLE of a and b
a = 0.06; b =  0.94;
mQbar = cov(stand_r_t,1);
vPsi0 = [log(a/(1-a)); log(b/(1-b)) ];

f  = @(vPsi)fDCC_LogLikelihood(stand_r_t, mQbar, vPsi);

opts = optimset('Display','iter','TolX',1e-4,'TolFun',1e-4,...
                'Diagnostics','off', 'MaxIter',1000, 'MaxFunEvals', 1000,...
                'LargeScale', 'off', 'PlotFcns', @optimplotfval);

[vPsi, fval, exitflag, output] = fminunc(f, vPsi0, opts);
a = exp(vPsi(1))/(1+exp(vPsi(1)));
b = (1-a)*exp(vPsi(2))/(1+exp(vPsi(2)));

Q=zeros(3,3);
P_engle=zeros(3,3,length(stand_r_t(:,1)));
Ht_engle=zeros(3,3,length(stand_r_t(:,1)));
for i = 1:length(stand_r_t(:,1))
    if i ==1
        Q =  mQbar;
    else
        Q = (1 - a - b) * mQbar + a * ( stand_r_t(i-1,:)' * stand_r_t(i-1,:) ) + b * Q ;
    end
    mQnsqrt = diag(1 ./ sqrt(diag(Q)));
    P_engle(:,:,i) = mQnsqrt * Q * mQnsqrt;
    Ht_engle(:,:,i) = diag(sqrt(cond_var_asset(i,:))) * P_engle(:,:,i) * diag(sqrt(cond_var_asset(i,:)));
end
figure(26),
plot(dates,reshape(Ht_engle(1,1,:),[],1))
hold on
plot(dates,reshape(Ht_engle(2,2,:),[],1))
plot(dates,reshape(Ht_engle(3,3,:),[],1))
title("Conditional Variances Engle's DCC")
legend(["h_{BMW}" "h_{ISP}" "h_{ENI}"],"location","best")
ylabel("h_{i}","rotation",0)
hold off

figure(27),
plot(dates,reshape(Ht_engle(1,2,:),[],1))
hold on
plot(dates,reshape(Ht_engle(1,3,:),[],1))
plot(dates,reshape(Ht_engle(2,3,:),[],1))
title("Conditional Covariances Engle's DCC")
legend(["h_{BMW-ISP}" "h_{BMW-ENI}" "h_{ISP-ENI}"],"location","best")
ylabel("h_{ij}","rotation",0)
hold off

figure(28),
plot(dates,reshape(P_engle(1,2,:),[],1))
hold on
plot(dates,reshape(P_engle(1,3,:),[],1))
plot(dates,reshape(P_engle(2,3,:),[],1))
title("Conditional Correlations Engle's DCC")
legend(["\rho_{BMW-ISP}" "\rho_{BMW-ENI}" "\rho_{ISP-ENI}"],"location","best")
ylabel("\rho_{ij}","rotation",0)
hold off
```

```
%% O-GARCH APPROACH

%pca analysis
V=pca(r_t);
f_t=r_t*V;

%fit GARCH(1,1) for each factor
cond_var_fac=zeros(length(r_t(:,1)),3);
stand_r_t=zeros(length(r_t(:,1)),3);
for i=1:3
    Mdl_ft = garch(1,1);
    EstMdl_fac = estimate(Mdl_ft,f_t(:,i));
    cond_var_fac(:,i)= infer(EstMdl_fac,f_t(:,i));
end
H_ft=zeros(3,3,length(cond_var_fac(:,1)));
%construct Ht for the factor
for i=1:length(cond_var_fac(:,1))
    H_ft(:,:,i)= diag(cond_var_fac(i,:));
end


Ht_ogar=zeros(3,3,length(cond_var_fac(:,1)));
Pt_ogar=zeros(3,3,length(cond_var_fac(:,1)));
% recover Ht for the returns
for i=1:length(cond_var_fac(:,1))
    Ht_ogar(:,:,i)= V*H_ft(:,:,i)*V';
    hinv=diag(1./sqrt(diag(Ht_ogar(:,:,i))));
    Pt_ogar(:,:,i)=hinv*Ht_ogar(:,:,i)*hinv;
end

figure(29),
plot(dates,reshape(Ht_ogar(1,1,:),[],1))
hold on
plot(dates,reshape(Ht_ogar(2,2,:),[],1))
plot(dates,reshape(Ht_ogar(3,3,:),[],1))
title("Conditional Variances O-GARCH")
legend(["h_{BMW}" "h_{ISP}" "h_{ENI}"],"location","best")
ylabel("h_{i}","rotation",0)
hold off
```

```
figure(30),
plot(dates,reshape(Ht_ogar(1,2,:),[],1))
hold on
plot(dates,reshape(Ht_ogar(1,3,:),[],1))
plot(dates,reshape(Ht_ogar(2,3,:),[],1))
title("Conditional Covariances O-GARCH")
legend(["h_{BMW-ISP}" "h_{BMW-ENI}" "h_{ISP-ENI}"],"location","best")
ylabel("h_{ij}","rotation",0)
hold off

figure(31),
plot(dates,reshape(Pt_ogar(1,2,:),[],1))
hold on
plot(dates,reshape(Pt_ogar(1,3,:),[],1))
plot(dates,reshape(Pt_ogar(2,3,:),[],1))
title("Conditional Correlations O-GARCH")
legend(["\rho_{BMW-ISP}" "\rho_{BMW-ENI}" "\rho_{ISP-ENI}"],"location","best")
ylabel("\rho_{ij}","rotation",0)
hold off
%% COPULA
%estimate garch-t model for the asset
stds_residuals=NaN(length(r_t),3);
cond_stdev=NaN(length(r_t),3);
du=NaN(3,1);
mrho=[];
dnu=[];
u=NaN(length(r_t),3);

for i=1:3
[coeff, stds_residuals(:,i),cond_stdev(:,i)] = fgarch11t_fit(r_t(:,i));
du(i) = coeff.Distribution.DoF;
u(:,i)   = tcdf(stds_residuals(:,i)*sqrt(du(i)/(du(i)-2)), du(i));

end
for i=1:3
    for j=i+1:3
     [rho,nu] = copulafit('t',[u(:,i) u(:,j)],'Method','ApproximateML');
     mrho=[mrho rho(2,1)];
     dnu=[dnu nu];
    end

end

w_1 = -sqrt((dnu(1)+1)*sqrt(1-mrho(1))/sqrt(1+mrho(1)));
td_12 = 2*tpdf(w_1,dnu(1)+1); % tail dependence

w_2 = -sqrt((dnu(2)+1)*sqrt(1-mrho(2))/sqrt(1+mrho(2)));
td_13 = 2*tpdf(w_2,dnu(2)+1); % tail dependence

w_3 = -sqrt((dnu(3)+1)*sqrt(1-mrho(3))/sqrt(1+mrho(3)));
td_23 = 2*tpdf(w_3,dnu(3)+1); % tail dependence

rho_matrix=[1 mrho(1) mrho(2); mrho(1) 1 mrho(3); mrho(2) mrho(3) 1];
disp(rho_matrix)

td_matrix=[1 td_12 td_13; td_12 1 td_23; td_13 td_23 1];
disp(td_matrix)
```

Forecasting volatility and value at risk

In this part we are going to consider a well-known measure of risk, the value at risk.
The value at risk is used to measure how much a certain portfolio can lose within a certain period, given a confidence interval.
We have different approach to estimate the VaR, but in our case we consider the econometric approach, based on assumption on the conditional distribution of y_{t+1} given all the information up to time t of the GARCH models.
The conditional VaR one step ahead is defined as:

$$VaR = -(\mu_{t+1|t} + \sqrt{h_{t+1}} F_{\epsilon}^{-1}(p))$$

where $F_{\epsilon}^{-1}(p)$ is the quantile function of the error term applied to the probability level.
In our case we are going to use either z_p, the p-th quantile of the standard normal distribution, or t_ν / $\sqrt{\frac{\nu}{\nu-2}}$ the p-th quantile of the standardized student's t distribution.
Considering the econometric approach to compute the conditional VaR, we are going to use 4 different models to estimate the risk measure one day ahead, and we will consider as benchmark the Riskmetrics approach to make a comparison, as well as for the volatility forecast.
In the Riskmetrics method the volatility is an exponential weighted moving average with smoothing parameter λ fixed at 0.06:

$$h_{t+1} = \lambda y_t^2 + (1-\lambda)h_t$$

and the distribution of the variable is assumed to be $y_{t+1}|Y_t \sim \mathrm{N}(0, \mathrm{h}_{t+1})$
The 4 models used are:

- Gaussian GARCH(1,1)
- Gaussian EGARCH(1,1)
- Student's t-GARCH(1,1)
- Student's t-GJR-GARCH(1,1)

Each volatility forecast and estimation of the VaR one period ahead is based on 3500 observations and the models are re-estimated each 22 days. Given a number of observations of 4902, we are going to estimate 1437 values for the volatility and the VaR.

For the estimation of the VaR we can compare the different models using 3 different assessments.

BACKTESTING

Here we consider the number of violations $r_{t+1} < -VaR_{t+1}$ and we compare it with the expected number $1437 * 0.05$.

	BMW	INTESA SP	ENI
Gaussian GARCH(1,1)	73	62	79
Gaussian EGARCH(1,1)	70	70	77
Student's t-GARCH(1,1)	77	68	86
Student's t-GJR-GARCH(1,1)	75	71	83
Riskmetrics	80	74	84
expected value	71	71	71

As we can observe in the above table, regarding the first and third asset the model with the lowest number of violations is the Gaussian EGARCH(1,1), even though only for BMW the number is lower than the expected value.
For Intesa SanPaolo the basic Gaussian GARCH outperforms the others.
This approach does not consider how big the violations are and so we should consider other measures.

TOTAL LOSS

The second criterion that we can adopt is the one of the total loss incurred, where the loss function considered is the check function:

$$L(r_{t+1}, VaR_{t+1}) = [\alpha - I(r_{t+1} < -VaR_{t+1})]\,(r_{t+1} + VaR_{t+1})$$

It is noticeable that the response is asymmetric and the loss is higher when there is a violation of the VaR than when the VaR is not violated.

	BMW	INTESA SP	ENI
Gaussian GARCH(1,1)	320.7830	335.9951	321.9621
Gaussian EGARCH(1,1)	317.2043	331.6451	316.5726
Student's t-GARCH(1,1)	321.1650	335.4972	320.8960
Student's t-GJR-GARCH(1,1)	320.4253	335.8393	316.6078
Riskmetrics	323.6213	338.0457	323.3675

The result of the total loss reflects the result of the number of violations in the first and third asset with the Gaussian EGARCH(1,1) to have the lowest value, but in ENI also the Student's t-GJR-GARCH(1,1) has value very near to the lowest one, so accounting for the fat tails, combined with the leverage effect, in this asset can be useful.

For Intesa SanPaolo the best model is the Gaussian EGARCH improving the results of the other models; this means that the leverage effect is relevant but it must be modeled with the exponential specification.

DIEBLOD-MARIANO TEST

A third method to assess the VaR consists in using the Diebold-Mariano test in order to establish if the models under analysis have the same level of accuracy or not. The test is based on the sample mean loss differential of two models:

$$\bar{\mathrm{d}} = \frac{1}{m} \sum_{t=T}^{T+m-1} \left[L(r_{t+1}, VaR_{i,t+1}) - L(r_{t+1}, VaR_{j,t+1}) \right]$$

where T is the size of the rolling window and m is the number of forecasts.
In our case the model i is one of the 4 models we used to derive the one day ahead estimation of the Var while the model j is the Riskmetrics, being used as the benchmark. Under the null hypothesis, the two tested models have the same accuracy and the differential loss has expected value equal to 0, whereas under the alternative hypothesis one of the two models has a better accuracy.
The test statistic used is:

$$DM = \frac{\bar{d}}{\sqrt{\frac{\widehat{\sigma_L^2}}{m}}}$$

which under Ho provides an asymptotically standard normal test.

The p-values for each model and each asset are reported in the below table:

	BMW	INTESA SP	ENI
Gaussian GARCH(1,1)	0.1058	0.4550	0.5333
Gaussian EGARCH(1,1)	0.0383	0.2350	0.0596
Student’s t-GARCH(1,1)	0.2062	0.1547	0.3042
Student’s t-GJR-GARCH(1,1)	0.1520	0.6693	0.2028

We reject the null hypothesis at 5% level only for the Gaussian EGARCH(1,1) in the first asset. This indicates that accounting for the leverage improved the estimation of the VaR with respect to a simple Gaussian GARCH or a Student’s t-GJR which, according to the test, have an accuracy equal to the Riskmetrics approach.
For the second asset the null is never rejected, so no model outperforms the Riskmetrics approach.
For the third asset the null hypothesis is never rejected at the 5% level, but for the Gaussian EGARCH the value is near and indeed we reject at the 10% level. This indicates that

accounting for the leverage using the EGARCH specification could give an improvement in the estimation of the VaR.

For the evaluation of the conditional variances forecasts we consider the mean square forecast error, defined as:

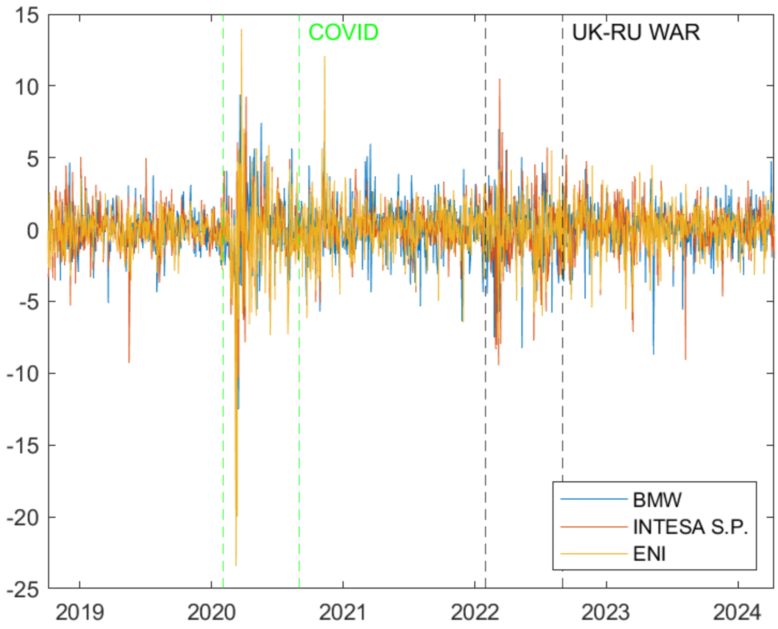
$$MSFE = \sum_{t=T}^{T+m-1} (r_{t+1}^2 - \widehat{h_{t+1|t}})^2$$

Considering the MSFE for the models under evaluation, we will prefer the one with the lowest MSFE.

	BMW	INTESA SP	ENI
Gaussian GARCH(1,1)	113.6880	185.1260	368.8538
Gaussian EGARCH(1,1)	113.1300	180.5834	365.3687
Student’s t-GARCH(1,1)	114.1721	185.7372	368.5128
Student’s t-GJR-GARCH(1,1)	110.5471	191.8431	372.1226
Riskmetrics	114.3819	186.6264	375.3184

The Gaussian EGARCH(1,1) forecasts volatility better than the other models in the second and third asset, instead for BMW the Student’s t-GJR-GARCH has the lowest MSFE. These results indicate that in each stock accounting for the leverage is useful in order to predict the volatility, but considering a fat tail distribution is an advantage only for the first asset , the one with the lowest value for the kurtosis. The result is controversial, but considering a Gaussian GJR, it produces a value for the MSFE similar to the student’s t GJR, so an explanation could be that in order to forecast the volatility in BMW we should consider the leverage effect but with GJR specification.

The 2 periods with the highest uncertainty and volatility in our forecast period are the COVID-19 and the Ukrainian-Russian war.
We can check if in these 2 periods one of the methods outperform the benchmark.



For the pandemic, considering the data from February 2020 to September 2020 we obtain the following results:

BMW	backtesting	total loss	DM test p-values	MSFE
Gaussian GARCH(1,1)	9	24.3045	0.6757	921.3330
Gaussian EGARCH(1,1)	8	24.7778	0.7961	918.8062
Student's t-GARCH(1,1)	9	24.4332	0.5963	921.3633
Student's t-GJR-GARCH(1,1)	8	24.7782	0.9211	917.5645
Riskmetrics	9	24.9523		923.9818

INTESA SP	backtesting	total loss	DM test p-values	MSFE
Gaussian GARCH(1,1)	7	26.4276	0.9123	1347.3594
Gaussian EGARCH(1,1)	8	27.4663	0.6787	1333.2859
Student’s t-GARCH(1,1)	7	26.2375	0.5797	1345.3126
Student’s t-GJR-GARCH(1,1)	8	27.3699	0.6856	1343.4766
Riskmetrics	7	26.6008		1347.3479

ENI	backtesting	total loss	DM test p-values	MSFE
Gaussian GARCH(1,1)	8	18.9817	0.7699	3509.9258
Gaussian EGARCH(1,1)	7	19.1074	0.9004	3498.5249
Student’s t-GARCH(1,1)	9	19.0098	0.7125	3509.3191
Student’s t-GJR-GARCH(1,1)	8	18.9293	0.7752	3506.7689
Riskmetrics	9	19.4078		3514.8384

Regarding the VaR estimation the model seems to show similar values, reflected also in the p-values of the DM test. Instead, considering the MSFE, the Gaussian EGARCH outperforms all the models in the second and third asset, and has a near value to the lowest one produced by the Student’s t-GJR. So in all the assets in order to forecast volatility during the pandemic it was helpful to consider the leverage effect.

Regarding instead the Ukrainian-Russian war, we consider the period from february 2022 to September2022, and we get the following result:

BMW	backtesting	total loss	DM test p-values	MSFE
Gaussian GARCH(1,1)	9	24.5588	0.6725	113.6420
Gaussian EGARCH(1,1)	8	24.7778	0.8051	113.3289
Student’s t-GARCH(1,1)	9	24.6766	0.5865	113.6878
Student’s t-GJR-GARCH(1,1)	8	25.0627	0.9334	112.3042
Riskmetrics	9	25.2064		115.9010

INTESA SP	backtesting	total loss	DM test p-values	MSFE
Gaussian GARCH(1,1)	7	26.7333	0.9392	288.6934
Gaussian EGARCH(1,1)	8	27.8116	0.6331	280.4964
Student’s t-GARCH(1,1)	7	26.5090	0.5876	286.0778
Student’s t-GJR-GARCH(1,1)	8	27.6916	0.6480	285.4648
Riskmetrics	7	26.8491		287.4446

ENI	backtesting	total loss	DM test p-values	MSFE
Gaussian GARCH(1,1)	8	19.3750	0.7843	69.4990
Gaussian EGARCH(1,1)	7	19.5542	0.9277	69.4192
Student’s t-GARCH(1,1)	9	19.3965	0.7272	69.4296
Student’s t-GJR-GARCH(1,1)	8	19.3601	0.8064	69.8818
Riskmetrics	9	19.7710		70.7713

During the analyzed period for the backtesting and the total loss values are very similar for all the model and all the asset, showing that no model was able to outperform the others in estimating the VaR, but considering the volatility forecast , the Gaussian EGARCH in Intesa SanPaolo was able to reduce the MSFE with respect to all the other specifications, whereas in BMW and ENI values are very similar again.

```matlab
%% VAR ESTIMATION

T=3500;
P=0.05;
w=22;
n=length(r_t(:,1))-T;
delta=floor(n/w);

est_var_1=NaN(n,3);
est_var_2=NaN(n,3);
est_var_3=NaN(n,3);
est_var_4=NaN(n,3);

cond_var_1=NaN(n,3);
cond_var_2=NaN(n,3);
cond_var_3=NaN(n,3);
cond_var_4=NaN(n,3);

garch_n = garch(1,1);
garch_n.Offset = NaN;

egarch_n = egarch(1,1);
egarch_n.Offset = NaN;

garch_t = garch(1,1);
garch_t.Distribution = "t";
garch_t.Offset = NaN;

gjr_t = gjr(1,1);
gjr_t.Distribution = "t";
gjr_t.Offset = NaN;
```

```matlab
for i=1:3

    for k=0:w:(delta*w)

        Est_garch_n = estimate(garch_n, r_t(k+1:T+k,i));

        Est_egarch_n = estimate(egarch_n,  r_t(k+1:T+k,i));

        Estgarch_t = estimate(garch_t,  r_t(k+1:T+k,i));
        dof_1=Estgarch_t.Distribution.DoF;

        Est_gjr_t = estimate(gjr_t,  r_t(k+1:T+k,i));
        dof_2=Est_gjr_t.Distribution.DoF;

        for j=0:w-1
            if k==delta*w && j>=(n-delta*w)
                break
            end
            cond_var_1(j+k+1,i)=forecast(Est_garch_n,1,r_t(j+k+1:T+j+k,i));
            est_var_1(j+k+1,i)=-(Est_garch_n.Offset+norminv(P)*sqrt(cond_var_1(j+k+1,i)));

            cond_var_2(j+k+1,i)=forecast(Est_egarch_n,1,r_t(j+k+1:T+j+k,i));
            est_var_2(j+k+1,i)=-(Est_egarch_n.Offset+norminv(P)*sqrt(cond_var_2(j+k+1,i)));

            cond_var_3(j+k+1,i)=forecast(Estgarch_t,1,r_t(j+k+1:T+j+k,i));
            est_var_3(j+k+1,i)=-(Estgarch_t.Offset+(tinv(P,dof_1)/sqrt(dof_1/(dof_1-2)))*sqrt(cond_var_3(j+k+1,i)));

            cond_var_4(j+k+1,i)=forecast(Est_gjr_t,1,r_t(j+k+1:T+j+k,i));
            est_var_4(j+k+1,i)=-(Est_gjr_t.Offset+(tinv(P,dof_2)/sqrt(dof_2/(dof_2-2)))*sqrt(cond_var_4(j+k+1,i)));

        end
    end
end
```

```
%% EVALUATION OF VAR PREDICTION backtesting
m=length(est_var_1);
exp_violations=floor(P*m);

violations1 = r_t(T+1:end,:) < -est_var_1;
violations2 = r_t(T+1:end,:) < -est_var_2;
violations3 = r_t(T+1:end,:) < -est_var_3;
violations4 = r_t(T+1:end,:) < -est_var_4;

backtest1 = sum(violations1);
backtest2 = sum(violations2);
backtest3 = sum(violations3);
backtest4 = sum(violations4);

%% EVALUATION OF VAR PREDICTION total loss incurred (check loss function)

check_function1 = (P- violations1).*(r_t(T+1:end,:) + est_var_1);
check_function2 = (P- violations2).*(r_t(T+1:end,:) + est_var_2);
check_function3 = (P- violations3).*(r_t(T+1:end,:) + est_var_3);
check_function4 = (P- violations4).*(r_t(T+1:end,:) + est_var_4);

total_loss_var1 = sum(check_function1);
total_loss_var2 = sum(check_function2);
total_loss_var3 = sum(check_function3);
total_loss_var4 = sum(check_function4);


%% EVALUATION OF VAR PREDICTION Diebold-Mariano test statistic

lambda=0.06;

h_rm= filter(1, [1 -(1-lambda)], lambda * r_t.^2);
cond_var_rm= lambda*(r_t(T:end-1,:).^2)+(1-lambda)*h_rm(T:end-1,:);
var_rm= -sqrt(cond_var_rm)*norminv(P);
violation_rm= r_t(T+1:end,:) < -var_rm;
backtest_rm= sum(violation_rm);
check_function_rm = (P - violation_rm).*(r_t(T+1:end,:) + var_rm);
total_loss_rm= sum(check_function_rm);


q = floor(m^(1/3));
```

```
d_1 = check_function1-check_function_rm;
d_2 = check_function2-check_function_rm;
d_3 = check_function3-check_function_rm;
d_4 = check_function4-check_function_rm;

gamma_v01=var(d_1);
gamma_v02=var(d_2);
gamma_v03=var(d_3);
gamma_v04=var(d_4);

dbar1 = (1/m)*sum(d_1);
dbar2 = (1/m)*sum(d_2);
dbar3 = (1/m)*sum(d_3);
dbar4 = (1/m)*sum(d_4);

for i=1:3
    [acf_d1(:,i),lags] = autocorr(d_1(:,i));
    acf_d2(:,i) = autocorr(d_2(:,i));
    acf_d3(:,i) = autocorr(d_3(:,i));
    acf_d4(:,i) = autocorr(d_4(:,i));
end

sigma_LRV1d = gamma_v01+2*sum((1-lags(2:q)/q).*(acf_d1(2:q,:).*gamma_v01));
sigma_LRV2d = gamma_v02+2*sum((1-lags(2:q)/q).*(acf_d2(2:q,:).*gamma_v02));
sigma_LRV3d = gamma_v03+2*sum((1-lags(2:q)/q).*(acf_d3(2:q,:).*gamma_v03));
sigma_LRV4d = gamma_v04+2*sum((1-lags(2:q)/q).*(acf_d4(2:q,:).*gamma_v04));

DM_test1d = dbar1./sqrt(sigma_LRV1d/m);
DM_test2d = dbar2./sqrt(sigma_LRV2d/m);
DM_test3d = dbar3./sqrt(sigma_LRV3d/m);
DM_test4d = dbar4./sqrt(sigma_LRV4d/m);

pvalue1d = 2*( 1-normcdf(abs(DM_test1d)));
pvalue2d = 2*( 1-normcdf(abs(DM_test2d)));
pvalue3d = 2*( 1-normcdf(abs(DM_test3d)));
pvalue4d = 2*( 1-normcdf(abs(DM_test4d)));
```

```
MSFE_1  = mean((r_t(T+1:end,:).^2 - cond_var_1).^2);
MSFE_2  = mean((r_t(T+1:end,:).^2 - cond_var_2).^2);
MSFE_3  = mean((r_t(T+1:end,:).^2 - cond_var_3).^2);
MSFE_4  = mean((r_t(T+1:end,:).^2 - cond_var_4).^2);
MSFE_rm = mean((r_t(T+1:end,:).^2 - cond_var_rm).^2);

%% MSFE for asset 1 USING Gaussian GJR-GARCH

cond_var_5=NaN(n,1);
for k=0:w:(delta*w)
    Est_gjr_n = estimate(gjr(1,1),r_t(k+1:T+k,1));
    for j=0:w-1
        if k==delta*w && j>=(n-delta*w)
            break
        end
        cond_var_5(j+k+1,1)=forecast(Est_garch_n,1,r_t(j+k+1:T+j+k,1));
    end
end

MSFE_5  = mean((r_t(T+1:end,1).^2 - cond_var_5).^2);

%% COVID and UK-RU war analysis
covid=dates(dates>= "2020-02-01" & dates<="2020-09-01");
war=dates(dates>= "2022-02-01" & dates<="2022-09-01");
figure(32),
plot(dates(T+1:end),r_t(T+1:end,:))
xline(covid(1),"g--")
xline(covid(end),"g--","COVID","LabelOrientation","horizontal","LabelHorizontalAlignment","RIGHT")
xline(war(1),"K--")
xline(war(end),"K--","UK-RU WAR","LabelOrientation","horizontal","LabelHorizontalAlignment","right")
legend(["BMW " "INTESA S.P." "ENI"],"location","southeast")

m_covid=length(covid);
k= dates>= "2020-02-01" & dates<="2020-09-01";
u=k(dates>= "2020-02-01" & dates<="2020-09-01");
backtest_var1_covid = sum(violations1(u,:));
backtest_var2_covid = sum(violations2(u,:));
backtest_var3_covid = sum(violations3(u,:));
backtest_var4_covid = sum(violations4(u,:));
backtest_rm_covid   = sum(violation_rm(u,:));

total_loss_var1_covid = sum(check_function1(u,:));
total_loss_var2_covid = sum(check_function2(u,:));
total_loss_var3_covid = sum(check_function3(u,:));
total_loss_var4_covid = sum(check_function4(u,:));
total_loss_rm_covid   = sum(check_function_rm(u,:));

q_covid = floor(m_covid^(1/3));

d_1_covid = check_function1(u,:)-check_function_rm(u,:);
d_2_covid = check_function2(u,:)-check_function_rm(u,:);
d_3_covid = check_function3(u,:)-check_function_rm(u,:);
d_4_covid = check_function4(u,:)-check_function_rm(u,:);

gamma_v01_covid=var(d_1_covid);
gamma_v02_covid=var(d_2_covid);
gamma_v03_covid=var(d_3_covid);
gamma_v04_covid=var(d_4_covid);

dbar1_covid = (1/m)*sum(d_1_covid);
dbar2_covid = (1/m)*sum(d_2_covid);
dbar3_covid = (1/m)*sum(d_3_covid);
dbar4_covid = (1/m)*sum(d_4_covid);

for i=1:3
    [acf_d1_covid(:,i),lags] = autocorr(d_1_covid(:,i));
    acf_d2_covid(:,i) = autocorr(d_2_covid(:,i));
    acf_d3_covid(:,i) = autocorr(d_3_covid(:,i));
    acf_d4_covid(:,i) = autocorr(d_4_covid(:,i));
end

sigma_LRV1d_covid = gamma_v01_covid+2*sum((1-lags(2:q_covid)/q_covid).*(acf_d1_covid(2:q_covid,:).*gamma_v01_covid));
sigma_LRV2d_covid = gamma_v02_covid+2*sum((1-lags(2:q_covid)/q_covid).*(acf_d2_covid(2:q_covid,:).*gamma_v02_covid));
sigma_LRV3d_covid = gamma_v03_covid+2*sum((1-lags(2:q_covid)/q_covid).*(acf_d3_covid(2:q_covid,:).*gamma_v03_covid));
sigma_LRV4d_covid = gamma_v04_covid+2*sum((1-lags(2:q_covid)/q_covid).*(acf_d4_covid(2:q_covid,:).*gamma_v04_covid));
```

```
DM_test1d_covid = dbar1_covid./sqrt(sigma_LRV1d_covid/m);
DM_test2d_covid = dbar2_covid./sqrt(sigma_LRV2d_covid/m);
DM_test3d_covid = dbar3_covid./sqrt(sigma_LRV3d_covid/m);
DM_test4d_covid = dbar4_covid./sqrt(sigma_LRV4d_covid/m);

pvalue1d_covid = 2*( 1-normcdf(abs(DM_test1d_covid)));
pvalue2d_covid = 2*( 1-normcdf(abs(DM_test2d_covid)));
pvalue3d_covid = 2*( 1-normcdf(abs(DM_test3d_covid)));
pvalue4d_covid = 2*( 1-normcdf(abs(DM_test4d_covid)));

MSFE_1_covid  = mean((r_t(k,:).^2 - cond_var_1(u,:)).^2);
MSFE_2_covid  = mean((r_t(k,:).^2 - cond_var_2(u,:)).^2);
MSFE_3_covid  = mean((r_t(k,:).^2 - cond_var_3(u,:)).^2);
MSFE_4_covid  = mean((r_t(k,:).^2 - cond_var_4(u,:)).^2);
MSFE_rm_covid  = mean((r_t(k,:).^2 - cond_var_rm(u,:)).^2);

m_war=length(war);
h= dates>= "2022-02-01" & dates<="2022-09-01";
g=h(dates>= "2022-02-01" & dates<="2022-09-01");

backtest_var1_war = sum(violations1(g,:));
backtest_var2_war = sum(violations2(g,:));
backtest_var3_war = sum(violations3(g,:));
backtest_var4_war = sum(violations4(g,:));
backtest_rm_war    = sum(violation_rm(g,:));

total_loss_var1_war = sum(check_function1(g,:));
total_loss_var2_war = sum(check_function2(g,:));
total_loss_var3_war = sum(check_function3(g,:));
total_loss_var4_war = sum(check_function4(g,:));
total_loss_rm_war    = sum(check_function_rm(g,:));

q_war = floor(m_war^(1/3));

d_1_war = check_function1(g,:)-check_function_rm(g,:);
d_2_war = check_function2(g,:)-check_function_rm(g,:);
d_3_war = check_function3(g,:)-check_function_rm(g,:);
d_4_war = check_function4(g,:)-check_function_rm(g,:);
```

```
gamma_v01_war=var(d_1_war);
gamma_v02_war=var(d_2_war);
gamma_v03_war=var(d_3_war);
gamma_v04_war=var(d_4_war);

dbar1_war = (1/m)*sum(d_1_war);
dbar2_war = (1/m)*sum(d_2_war);
dbar3_war = (1/m)*sum(d_3_war);
dbar4_war = (1/m)*sum(d_4_war);

for i=1:3
    [acf_d1_war(:,i),lags] = autocorr(d_1_war(:,i));
    acf_d2_war(:,i) = autocorr(d_2_war(:,i));
    acf_d3_war(:,i) = autocorr(d_3_war(:,i));
    acf_d4_war(:,i) = autocorr(d_4_war(:,i));
end

sigma_LRV1d_war = gamma_v01_war+2*sum((1-lags(2:q_war)/q_war).*(acf_d1_war(2:q_war,:).*gamma_v01_war));
sigma_LRV2d_war = gamma_v02_war+2*sum((1-lags(2:q_war)/q_war).*(acf_d2_war(2:q_war,:).*gamma_v02_war));
sigma_LRV3d_war = gamma_v03_war+2*sum((1-lags(2:q_war)/q_war).*(acf_d3_war(2:q_war,:).*gamma_v03_war));
sigma_LRV4d_war = gamma_v04_war+2*sum((1-lags(2:q_war)/q_war).*(acf_d4_war(2:q_war,:).*gamma_v04_war));

DM_test1d_war = dbar1_war./sqrt(sigma_LRV1d_war/m);
DM_test2d_war = dbar2_war./sqrt(sigma_LRV2d_war/m);
DM_test3d_war = dbar3_war./sqrt(sigma_LRV3d_war/m);
DM_test4d_war = dbar4_war./sqrt(sigma_LRV4d_war/m);

pvalue1d_war = 2*( 1-normcdf(abs(DM_test1d_war)));
pvalue2d_war = 2*( 1-normcdf(abs(DM_test2d_war)));
pvalue3d_war = 2*( 1-normcdf(abs(DM_test3d_war)));
pvalue4d_war = 2*( 1-normcdf(abs(DM_test4d_war)));

MSFE_1_war  = mean((r_t(h,:).^2 - cond_var_1(g,:)).^2);
MSFE_2_war  = mean((r_t(h,:).^2 - cond_var_2(g,:)).^2);
MSFE_3_war  = mean((r_t(h,:).^2 - cond_var_3(g,:)).^2);
MSFE_4_war  = mean((r_t(h,:).^2 - cond_var_4(g,:)).^2);
MSFE_rm_war = mean((r_t(h,:).^2 - cond_var_rm(g,:)).^2);
```