Predict cryptos with ARMA GARCH and r-vine

Monika Monstvilaite

Analysis based on data available at 2018-03-16 01:00:00

Note: data is random, anytrading strategies that

Marginal Log-Returns

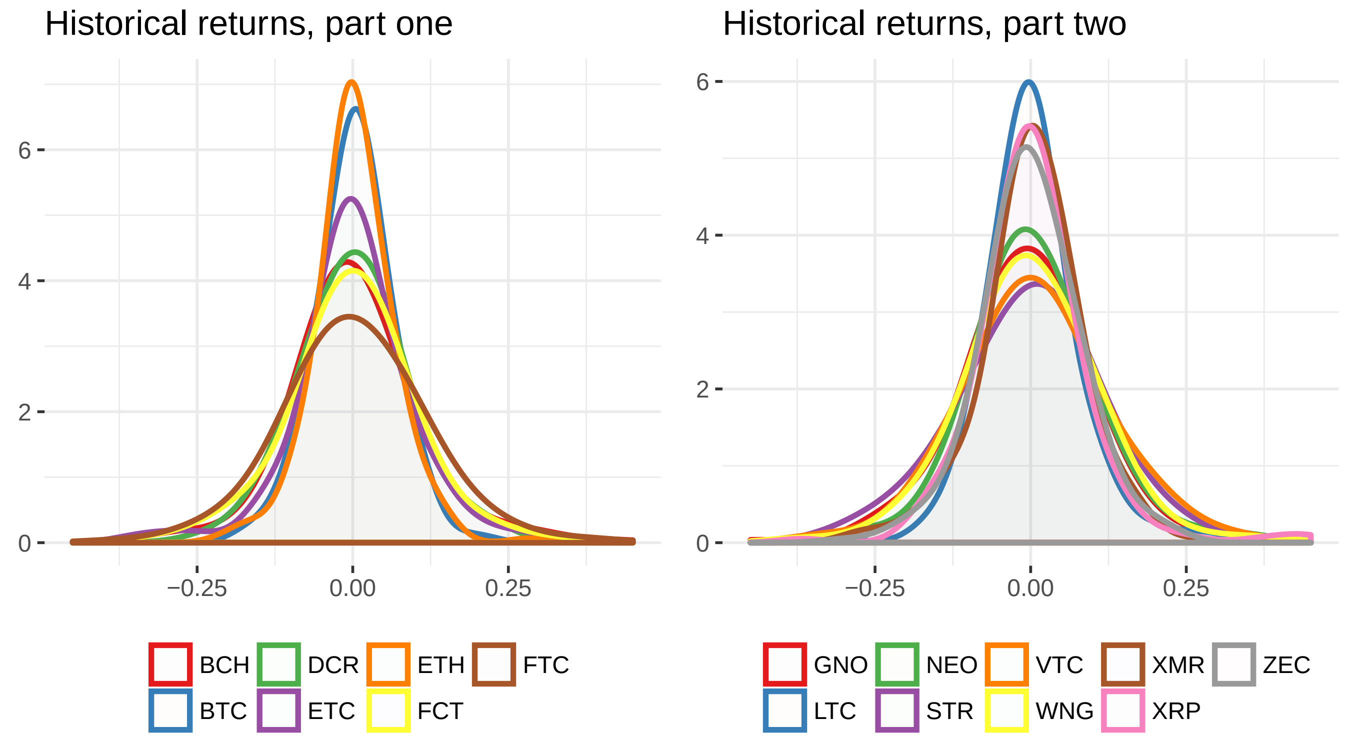
Here is descriptive analysis of historical returns. See mean, median, min and max value for each cryptocurrency from 2017-08-28 02:00:00 to 2018-03-16 01:00:00:

The historical means of the data are the following:

```
##    BCH    BTC    DCR    ETC    ETH    FCT    FTC    GNO    LTC    NEO
##  0.003  0.003  0.002  0.001  0.003 -0.001  0.008 -0.004  0.005  0.003
##    STR    VTC    WNG    XMR    XRP    ZEC
## -0.002  0.005 -0.002  0.002  0.006 -0.001
```

The historical variance of data is the following:

```
##   BCH   BTC   DCR   ETC   ETH   FCT   FTC   GNO   LTC   NEO   STR   VTC
## 0.101 0.060 0.085 0.093 0.062 0.100 0.121 0.110 0.086 0.101 0.114 0.114
##   WNG   XMR   XRP   ZEC
## 0.105 0.080 0.101 0.077
```



ARMA-GARCH models fitted

Standardized error distribution

Should be mean 0, variance 1, no bumps, but cryptos are weird

Transforming to uniform distribution

Nescessary in order to investigate dependence structure

R-Vine copula

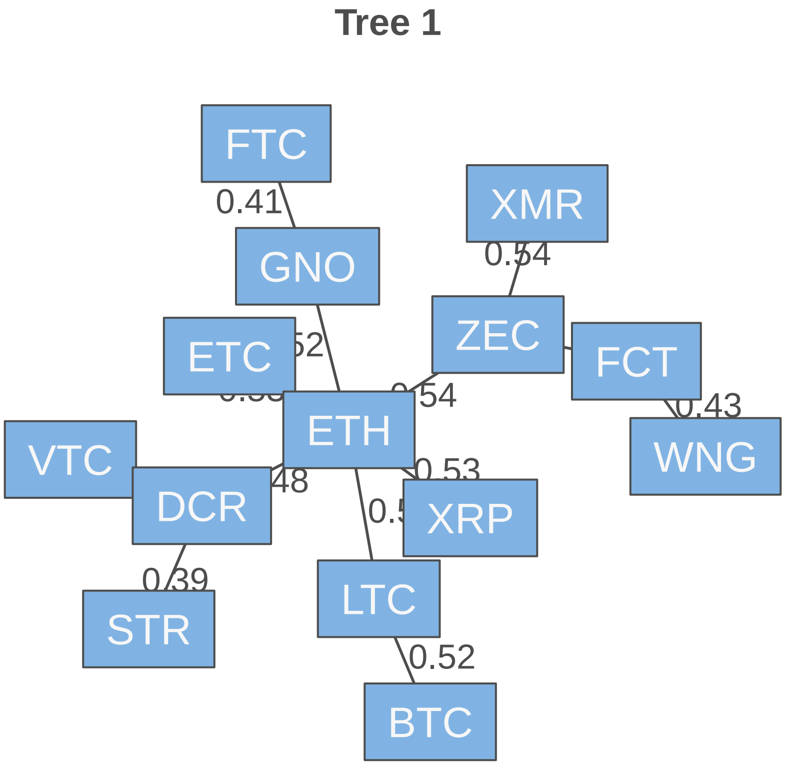
Instead of using correlation between variables, we will use r-vine copula. It is done so because correlation is not able to capture the nuances of assets that are correlated in for e.g. left tail of a distribution, but not a right tail. Intuition: cryptocurrencies might all crash at the same time, but when the market is doing well, they are uncorrelated.

Regular vine copula is a method to construct one solution for a simplified pair copula construction such that it would be as close as possible to real multivariate distribution that we aim to model. This way we are also able capture the dependency structure between the variables in an efficient way.

Kendall?s τ is given by: $\tau = P((X_1 - X_2)(Y_1 - Y_2) > 0) - P((X_1 - X_2)(Y_1 - Y_2) < 0)$

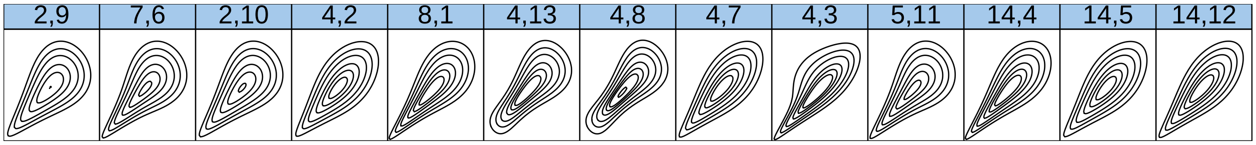
where $(X_1, Y_1) \sim F$ and $(X_2, Y_2) \sim F$ are independent pairs of random variables. Kendall's τ is a rank correlation. It does not depend directly on the values and thus is invariant under strictly monotone transformations (cite(Gruber)). On the contrary, linear correlation parameter Pearson's ρ is it not invariant under non-linear strictly increasing transformations, meaning that value of Pearson's ρ depends on marginal distributions. That is why Kendall?s τ is a more reliable measure in our case.

We plot here the dependence structure. Tree graph below shows the strongest dependencies between cryptocurrencies based on Kendall?s τ.



Please note below how variables are encoded:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
BTC	DCR	ETC	ETH	FCT	FTC	GNO	GNT	LTC	STR	VTC	WNG	XMR	XRP	ZEC



Copulas can help to perceive and visualize the nuances of dependence which is useful when describing the dependence of extreme events. In a figure above we can see that dependence structure can look very different and is not nescessarily symmetrical. Since copulas reveal dependence on a quantile scale, it is especially useful in the context of quantile based risk measures.

Prediction one day ahead

Backtransforming

Calculating variance from GARCH:

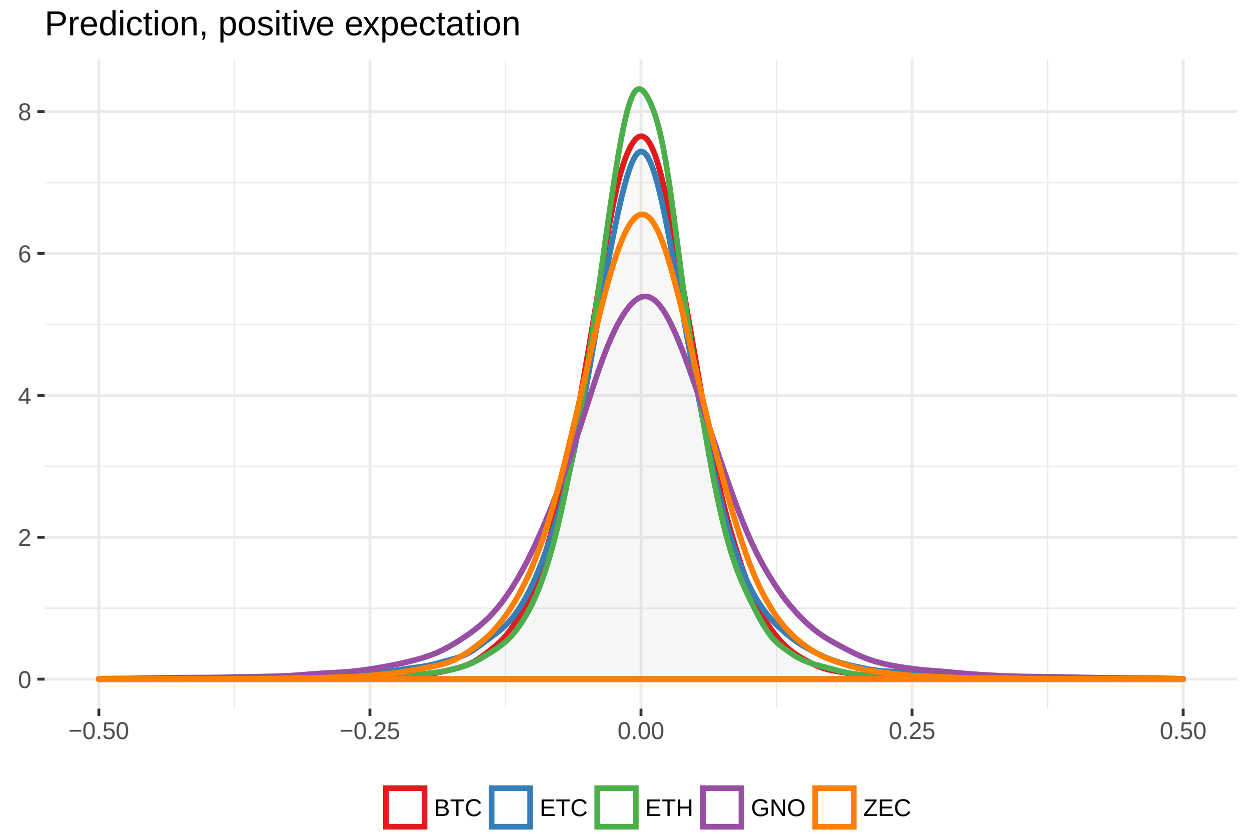
$\sigma^2_{t+1} = \omega + \alpha_1 \epsilon^2_t + \beta_1 \sigma^2_t$

We use the above equation in order to find error distribution on $T+2$ that will be inserted to ARMA equation.

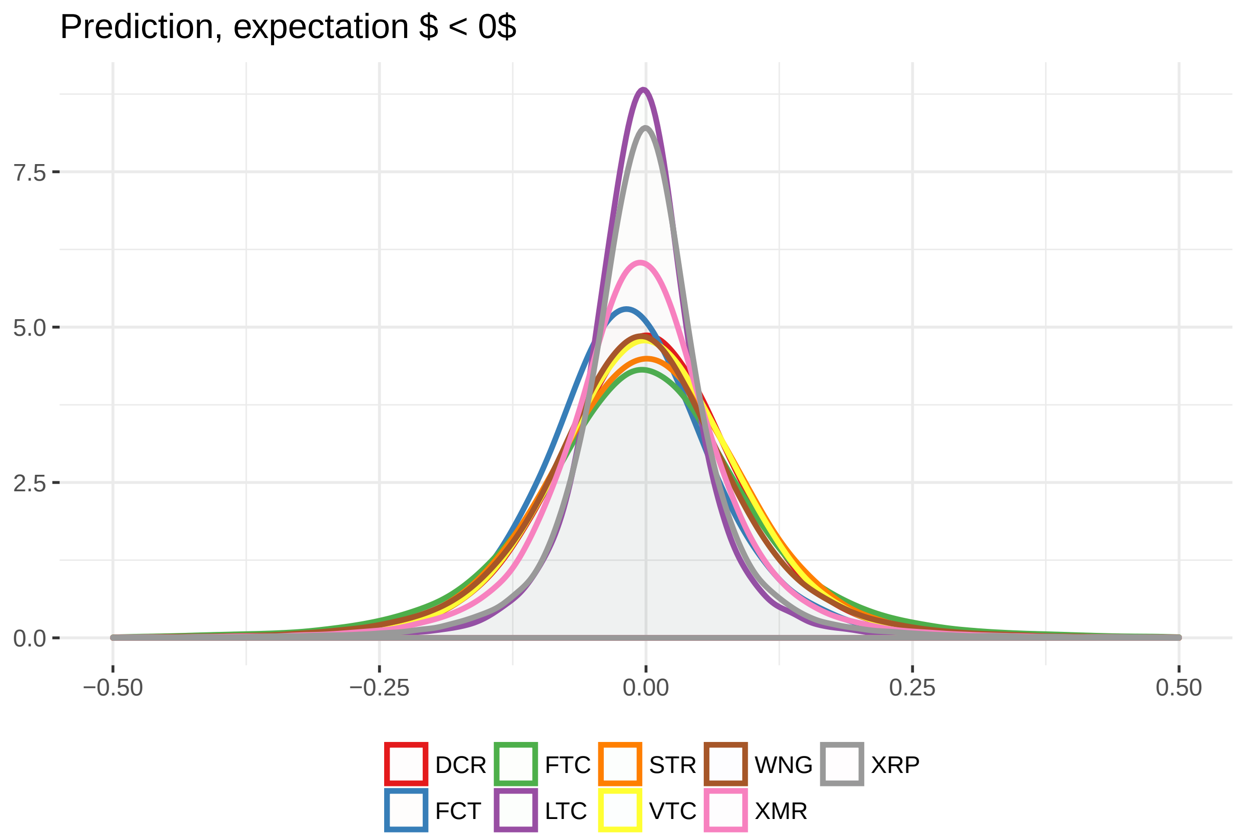
$\epsilon_{t+1} = \sigma_{t+1} z$

Prediction of return distribution one day ahead

$r_{t+1} = \mu + \epsilon_{t+1} + ar_1 r_t + ma_1 \epsilon_t$



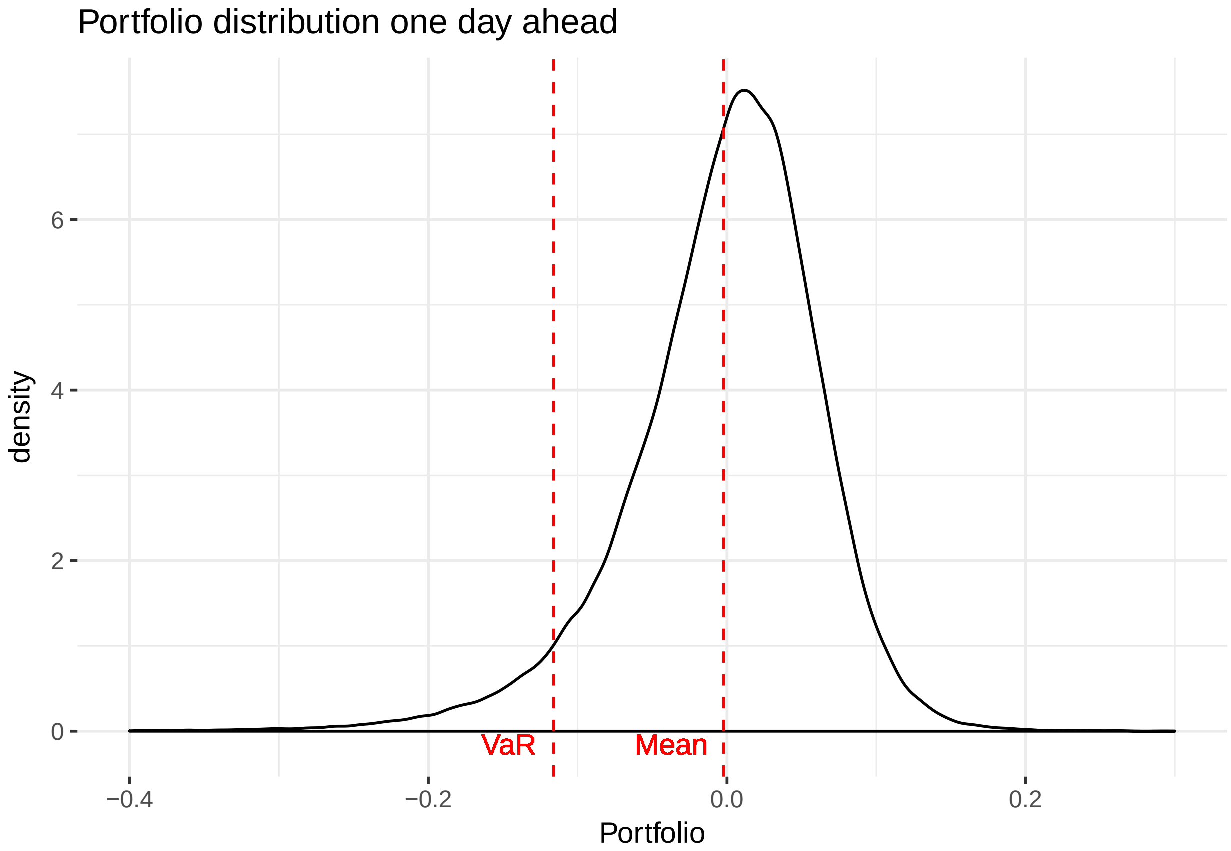
```
##    BTC    ETC    ETH    GNO    ZEC
## 0.0001 0.0002 0.0001 0.0032 0.0004
```



```
##      DCR     FCT     FTC     LTC     STR     VTC     WNG     XMR     XRP
## -0.0002 -0.0168 -0.0033 -0.0038 -0.0004  0.0000 -0.0060 -0.0049 -0.0013
```

Portfolio. FAT LEFT TAIL

```
## [1] 0.01295282 0.03885978 0.03588606 0.79935833 0.11294301
```



```
## [1] -0.1161329
```

```
## [1] 0.0051
```

If investing in all assets equally, expected return is:

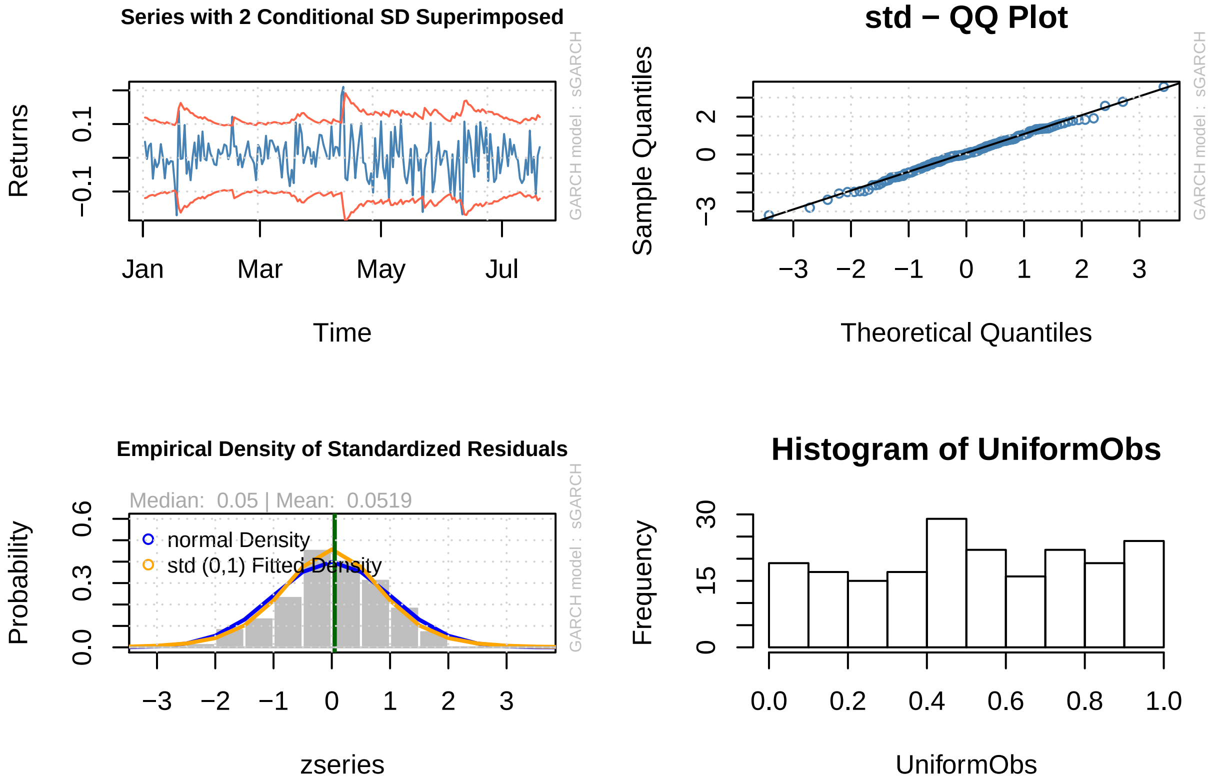
```
##    Portfolio
## -0.002329492
```

Monika’s section

This section is to exaluate how the ARMA-GARCH fitted the marginal time series.

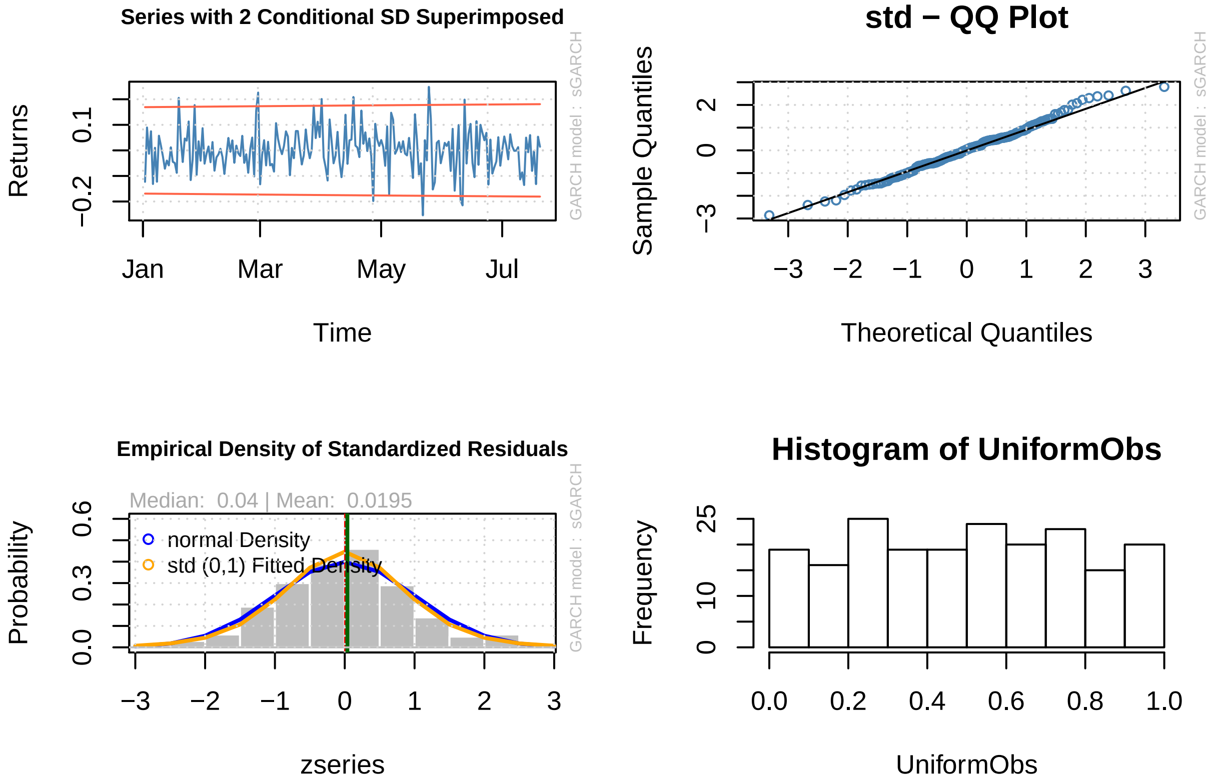
What we pay attention: how arma garch was able to determine Value at Risk: If the red curve is able to determine and adjust to the shocks, model is working well. If the red curve resembles a line (or not too far from it), that means the data is too random. We still include those variables, since they are dependent to other variables in the model.

BTC. VaR, QQ-plot, ACF



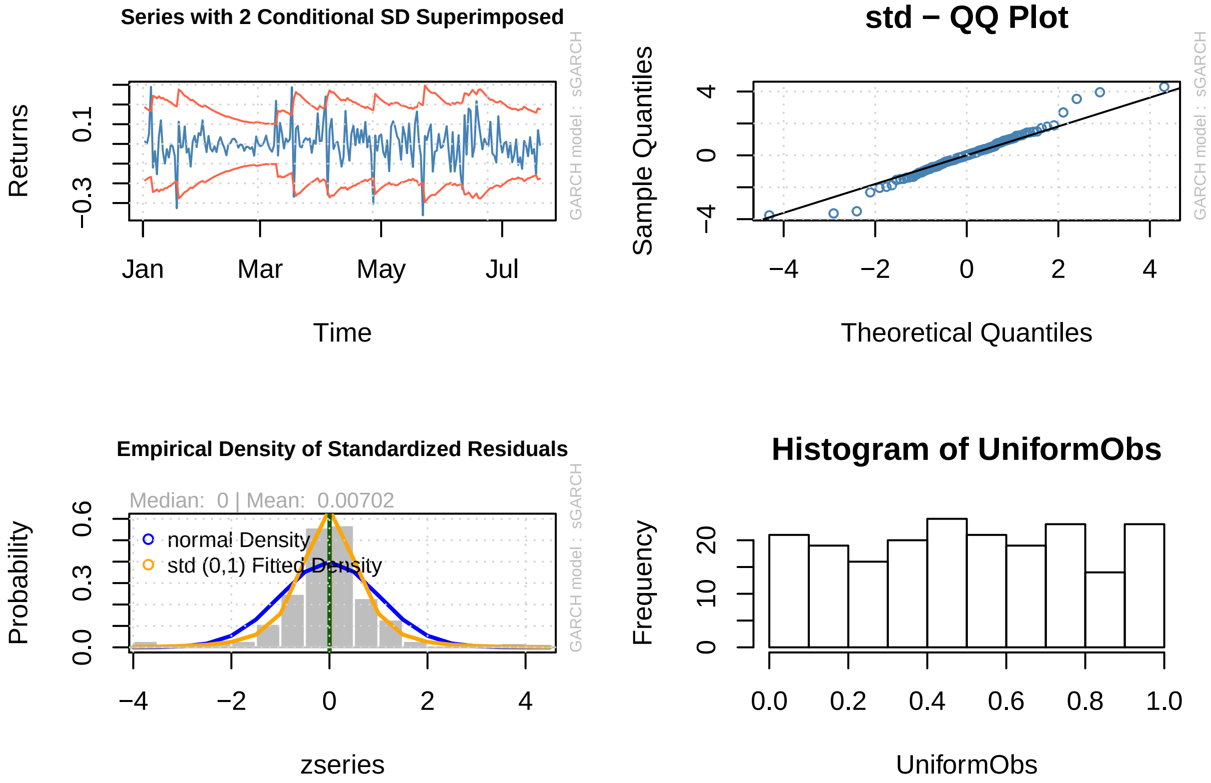
```
##              Estimate   Std. Error   t value     Pr(>|t|)
## omega  0.0003481929 0.0005129137 0.6788529 4.972311e-01
## alpha1 0.0940421829 0.0797333026 1.1794593 2.382153e-01
## beta1  0.8180943076 0.1982861573 4.1258266 3.694055e-05
## shape  6.8413108800 3.8223211719 1.7898315 7.348100e-02
```

DCR. VaR, QQ-plot, ACF



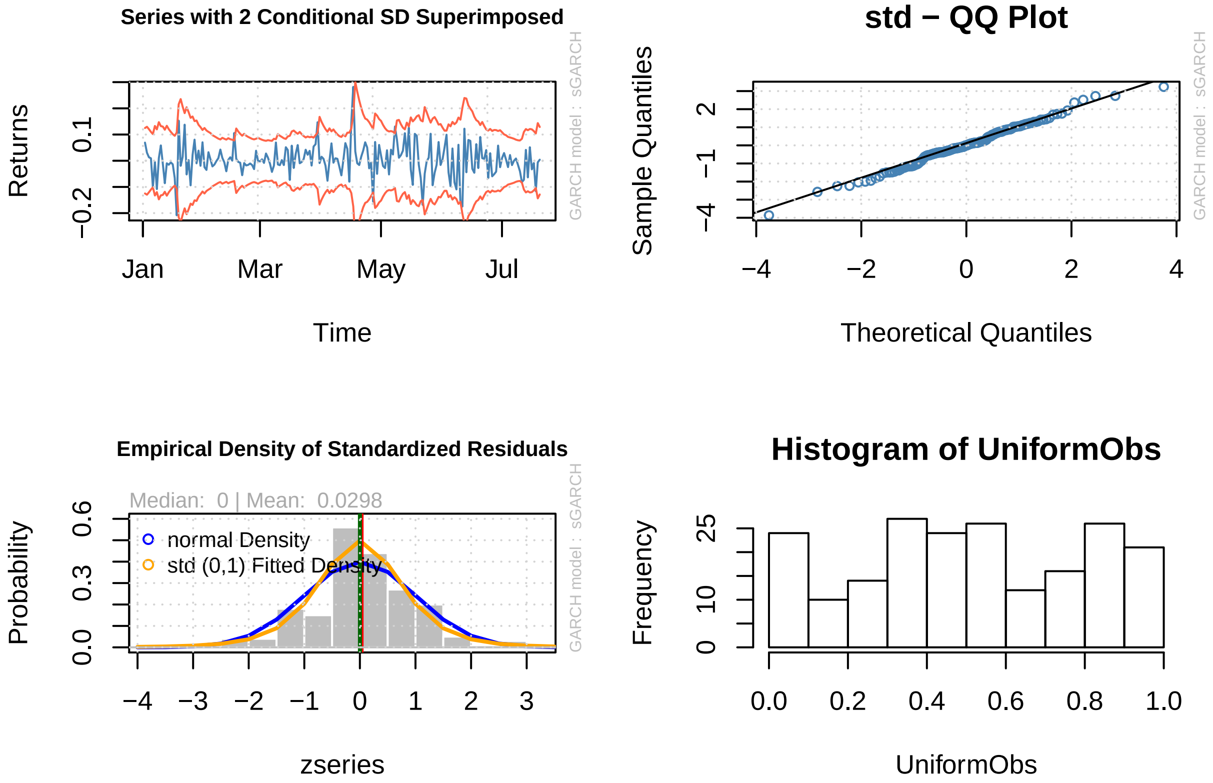
```
##              Estimate   Std. Error      t value   Pr(>|t|)
## omega   1.279690e-05 2.490007e-05 5.139305e-01 0.60730061
## alpha1  8.666266e-10 9.092841e-03 9.530867e-08 0.99999992
## beta1   9.989999e-01 2.775265e-03 3.599655e+02 0.00000000
## shape   8.012069e+00 4.465946e+00 1.794036e+00 0.07280742
```

ETC. VaR, QQ-plot, ACF



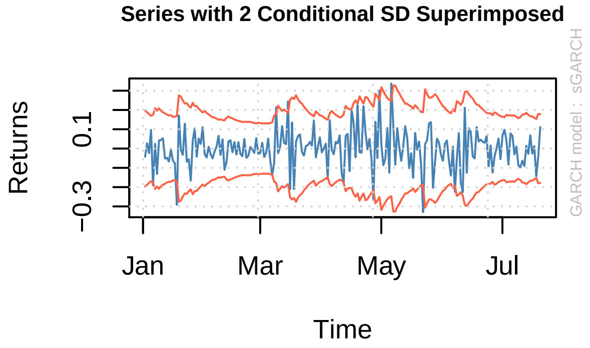
```
##              Estimate   Std. Error    t value     Pr(>|t|)
## ma1    -0.1184906812 0.0646068584 -1.8340264 6.665008e-02
## omega   0.0001621095 0.0002394136  0.6771104 4.983359e-01
## alpha1  0.0963989996 0.0453804275  2.1242418 3.364993e-02
## beta1   0.9026009990 0.0452260918 19.9575281 0.000000e+00
## shape   2.9587027142 0.5154981633  5.7395020 9.495539e-09
```

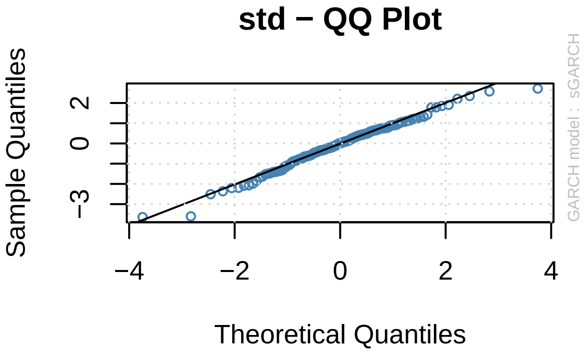
ETH. VaR, QQ-plot, ACF

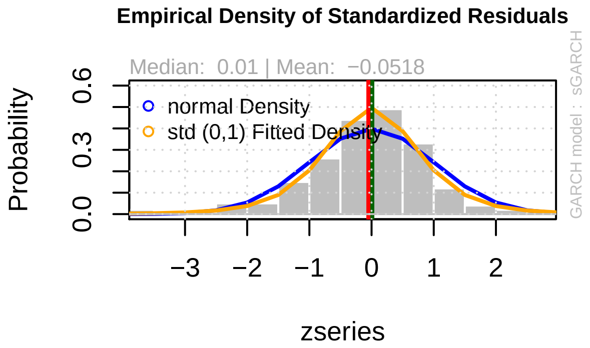


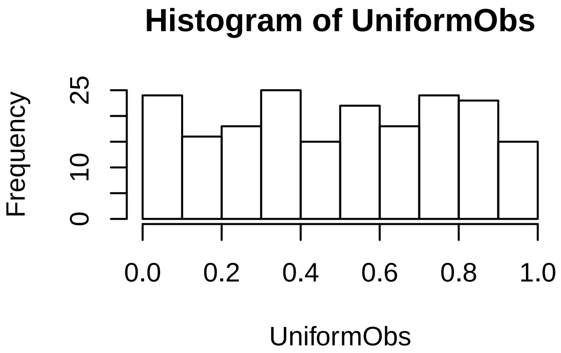
```
##               Estimate   Std. Error  t value     Pr(>|t|)
## omega   0.0003122539 0.0002412107 1.294528 1.954832e-01
## alpha1  0.2106361078 0.1124271054 1.873535 6.099456e-02
## beta1   0.7408829812 0.1164970883 6.359670 2.021883e-10
## shape   4.7299157248 1.8084381693 2.615470 8.910476e-03
```

Factom (FCT). VaR, QQ-plot, ACF


Series with 2 Conditional SD Superimposed

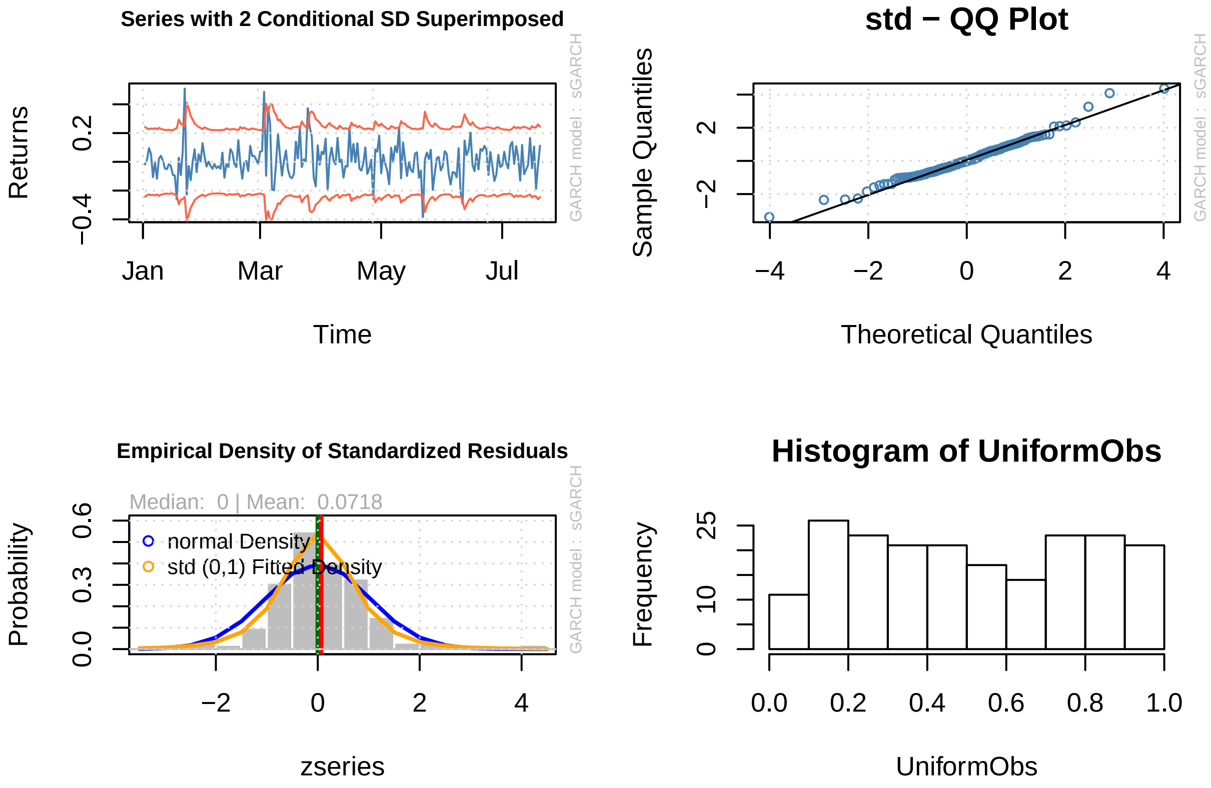

std − QQ Plot


Empirical Density of Standardized Residuals


Histogram of UniformObs

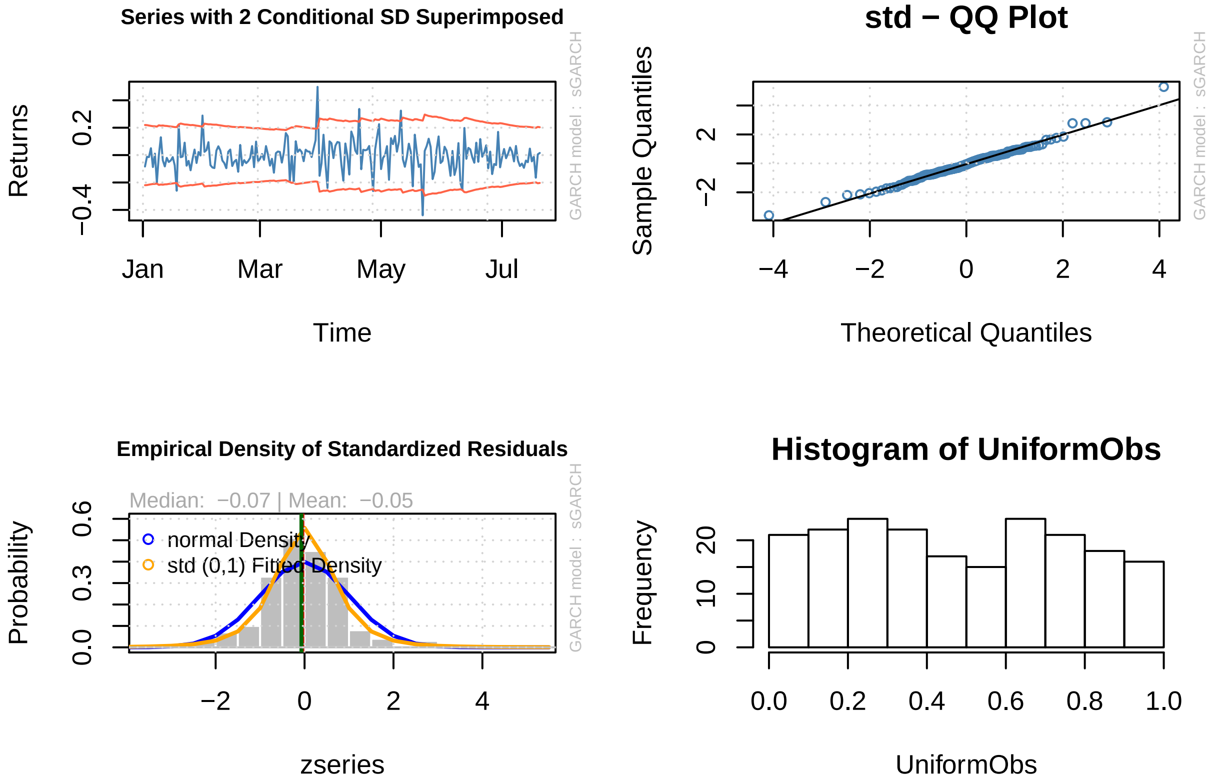
```
##               Estimate   Std. Error    t value     Pr(>|t|)
## ma1    -0.1699014592 0.0739098465 -2.2987662 2.151822e-02
## omega   0.0006142946 0.0006690363  0.9181782 3.585256e-01
## alpha1  0.1331746263 0.0815345950  1.6333512 1.023952e-01
## beta1   0.8225408323 0.1034646126  7.9499726 1.776357e-15
## shape   4.7807360000 1.7398723722  2.7477510 6.000557e-03
```

FeatherCoin (FTC). VaR, QQ-plot, ACF



```
##              Estimate  Std. Error   t value     Pr(>|t|)
## ar1      0.657927206 0.225026238  2.923780 0.0034580910
## ma1     -0.728190844 0.197902099 -3.679551 0.0002336451
## omega    0.004705727 0.003208634  1.466583 0.1424895277
## alpha1   0.114715075 0.083148715  1.379637 0.1676983203
## beta1    0.597220559 0.224333323  2.662202 0.0077631340
## shape    3.834070247 1.127838016  3.399487 0.0006751248
```

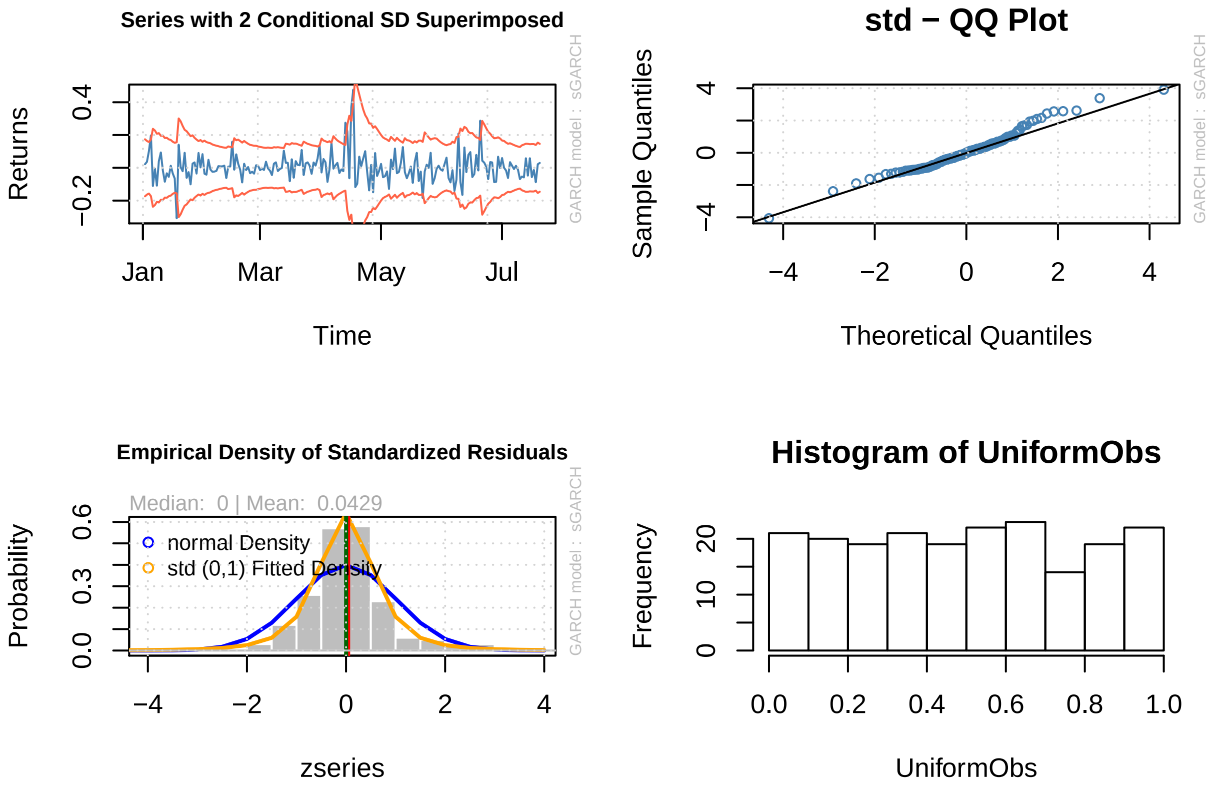
GNO. VaR, QQ-plot, ACF



```
##              Estimate   Std. Error    t value     Pr(>|t|)
## ar1      0.3569023128 0.2383545204  1.4973591 1.342998e-01
## ma1     -0.4850063763 0.2186812154 -2.2178694 2.656374e-02
## omega    0.0002569889 0.0003331118  0.7714793 4.404229e-01
## alpha1   0.0319272021 0.0242343759  1.3174345 1.876930e-01
## beta1    0.9515299959 0.0309810834 30.7132576 0.000000e+00
## shape    3.6086750602 0.9175675925  3.9328711 8.393726e-05
```

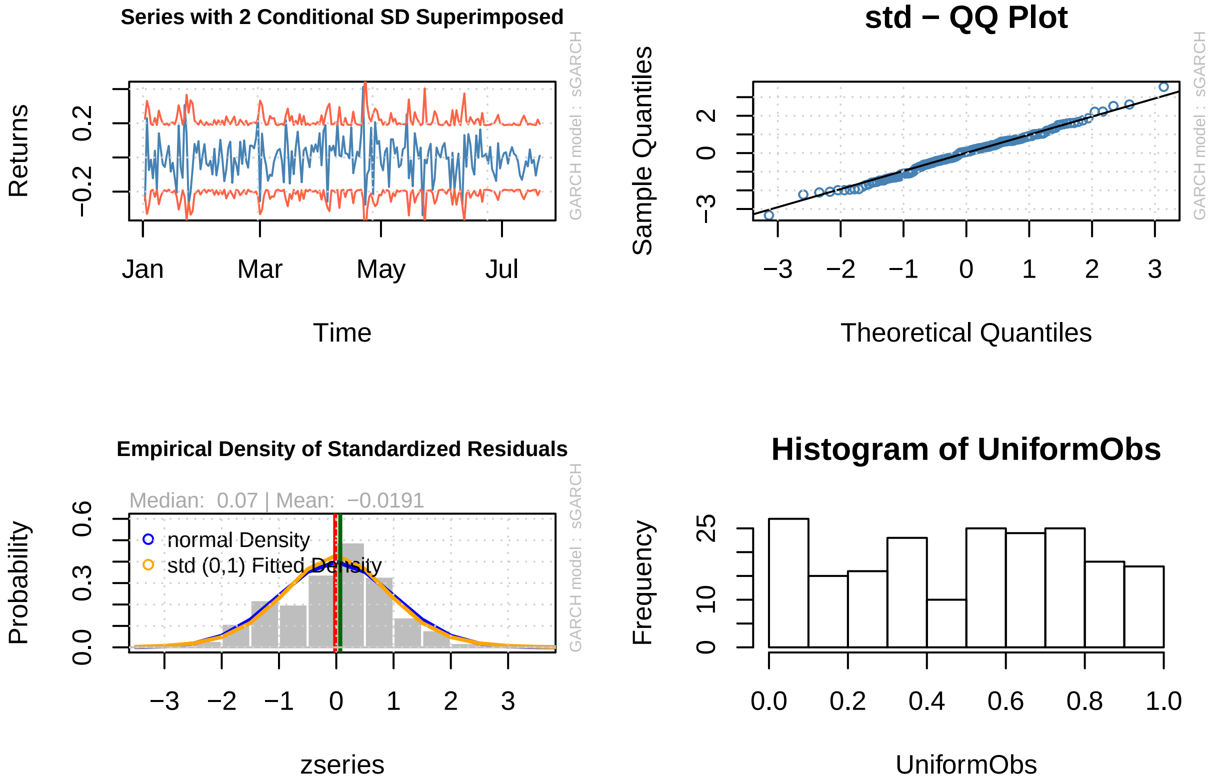
GNT. VaR, QQ-plot, ACF

LTC. VaR, QQ-plot, ACF



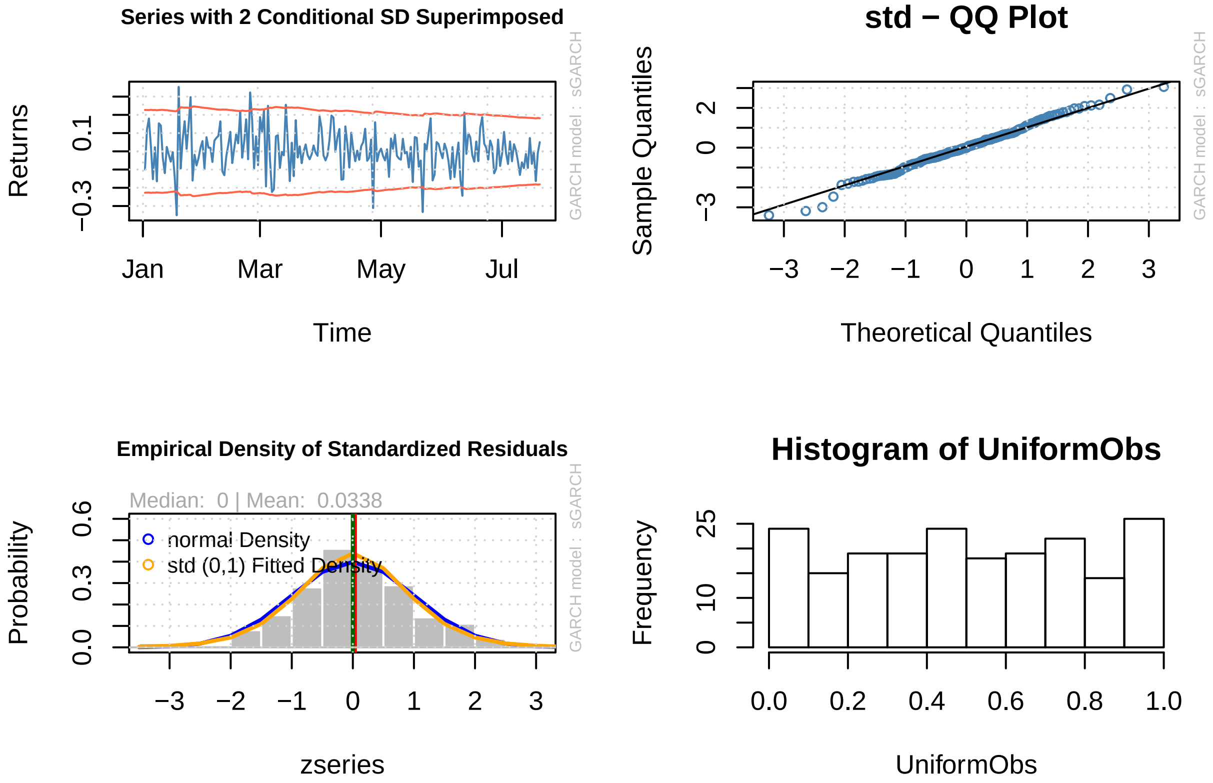
```
##             Estimate   Std. Error   t value     Pr(>|t|)
## ma1    -0.1197902443 0.0695639026 -1.722017 8.506638e-02
## omega   0.0007450262 0.0006180751  1.205397 2.280499e-01
## alpha1  0.1722029393 0.1222236432  1.408917 1.588598e-01
## beta1   0.7725302520 0.1009229145  7.654657 1.931788e-14
## shape   2.9741886919 0.6667161823  4.460952 8.159648e-06
```

STR. VaR, QQ-plot, ACF



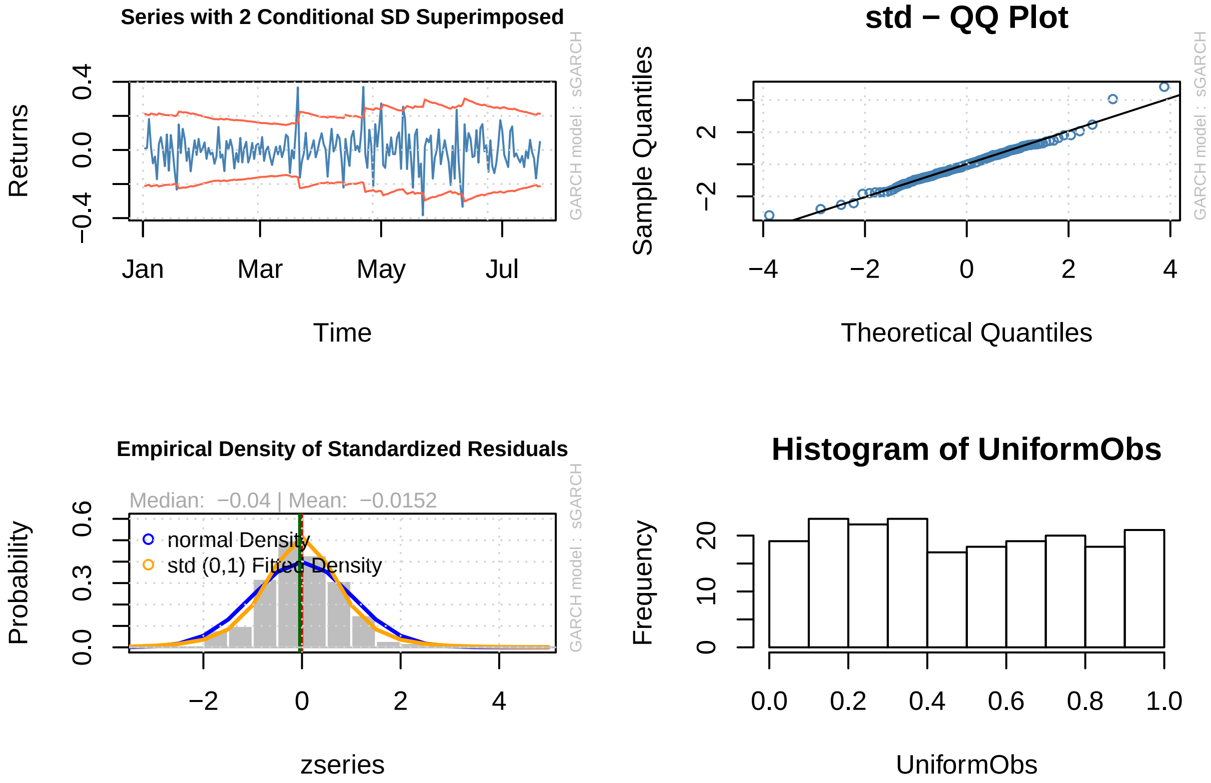
```
##              Estimate  Std. Error       t value     Pr(>|t|)
## ma1    -1.242386e-01  0.04440364 -2.797937e+00 5.143012e-03
## omega   8.859028e-03  0.01127829  7.854937e-01 4.321642e-01
## alpha1  2.920287e-01  0.29794069  9.801571e-01 3.270086e-01
## beta1   1.697252e-08  1.20468976  1.408870e-08 1.000000e+00
## shape   1.164409e+01  2.95889317  3.935285e+00 8.309813e-05
```

VTC. VaR, QQ-plot, ACF



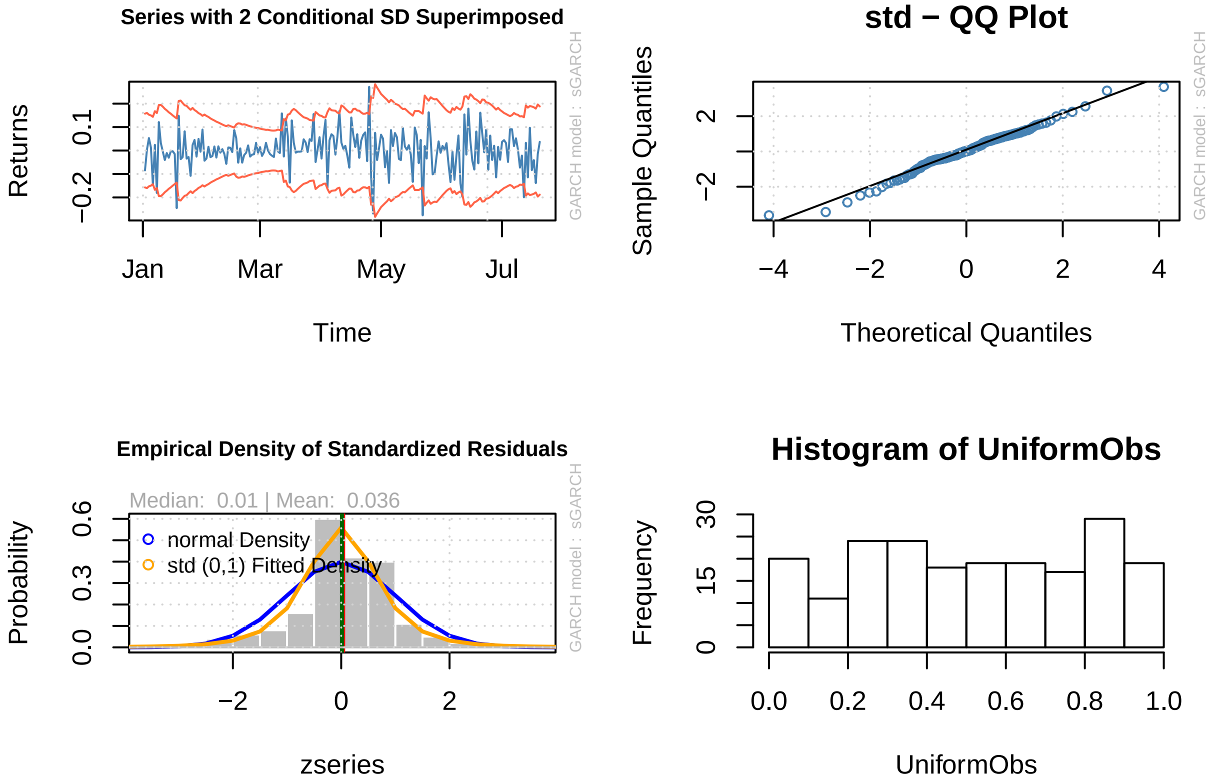
```
##              Estimate   Std. Error      t value   Pr(>|t|)
## omega  1.476034e-11 0.0000485044 3.043094e-07 0.99999976
## alpha1 1.141807e-02 0.0099473258 1.147853e+00 0.25102932
## beta1  9.855193e-01 0.0095264057 1.034513e+02 0.00000000
## shape  9.155458e+00 5.0673249777 1.806764e+00 0.07079915
```

WNG. VaR, QQ-plot, ACF



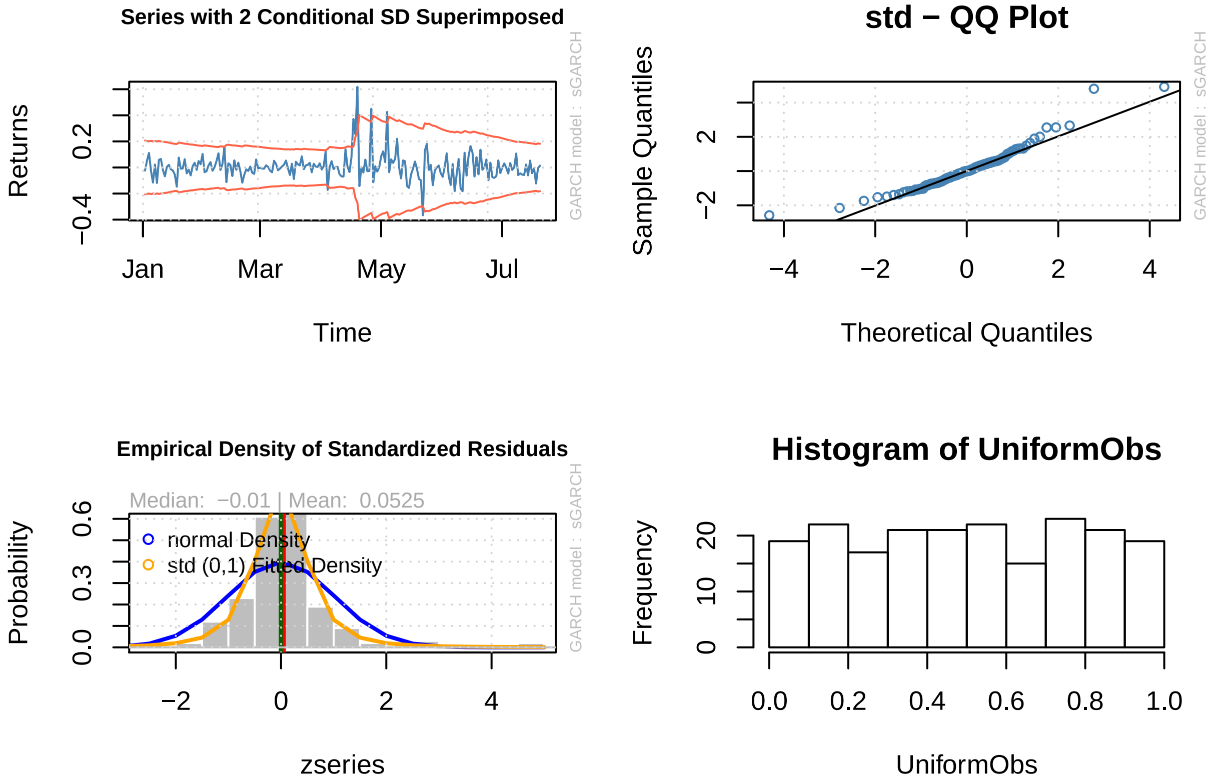
```
##                Estimate   Std. Error    t value    Pr(>|t|)
## ma1    -0.1733479001 0.0868443815 -1.9960750 0.045925758
## omega   0.0001515835 0.0002020066  0.7503889 0.453020504
## alpha1  0.0416747034 0.0240770166  1.7308915 0.083471117
## beta1   0.9485739768 0.0311206404 30.4805417 0.000000000
## shape   4.2579478116 1.3302853527  3.2007778 0.001370572
```

XMR. VaR, QQ-plot, ACF



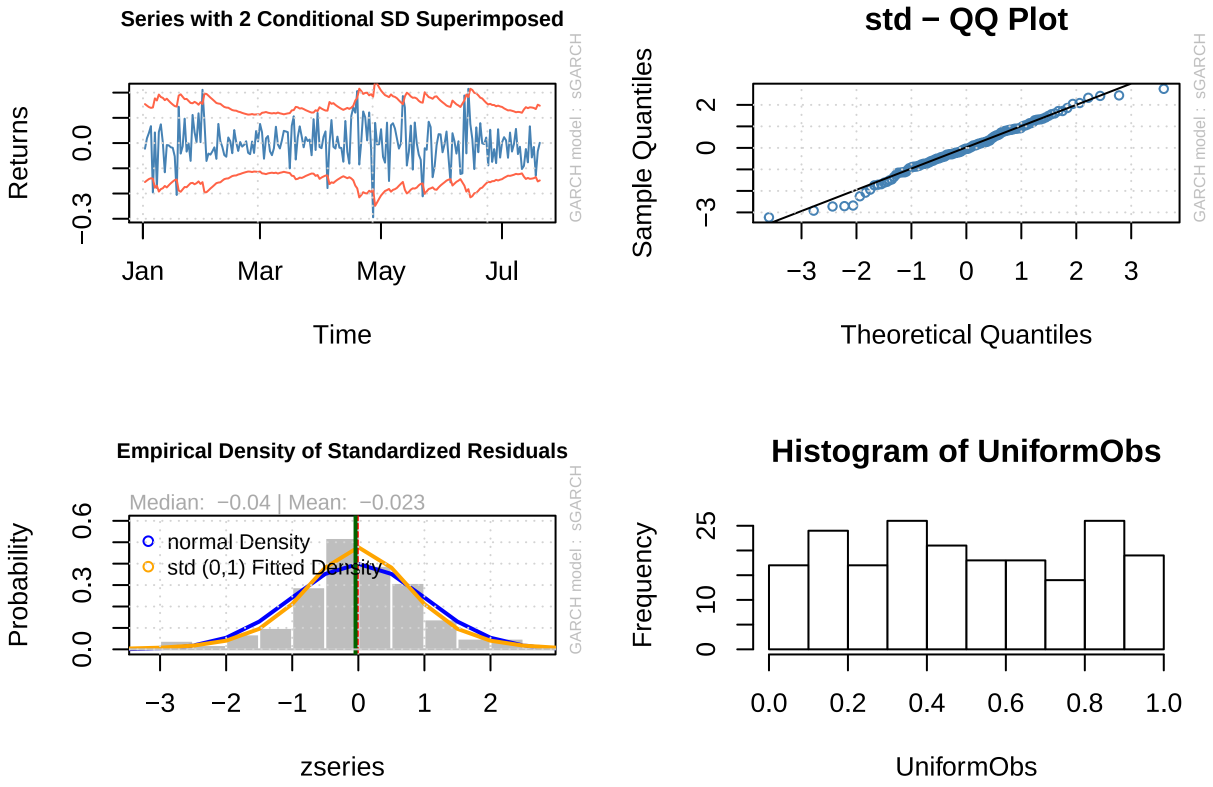
```
##              Estimate   Std. Error    t value     Pr(>|t|)
## ma1    -0.1615643429 0.0655118116 -2.4661865 1.365602e-02
## omega   0.0001292563 0.0001985396  0.6510354 5.150237e-01
## alpha1  0.1115663532 0.0603045878  1.8500475 6.430670e-02
## beta1   0.8874336463 0.0643411880 13.7926214 0.000000e+00
## shape   3.5946544342 0.8535868323  4.2112346 2.539788e-05
```

XRP. VaR, QQ-plot, ACF



```
##              Estimate   Std. Error   t value     Pr(>|t|)
## ma1     -0.151580032 0.0578907197 -2.618382 8.834778e-03
## omega    0.000154541 0.0001388817  1.112753 2.658147e-01
## alpha1   0.052163679 0.0314455797  1.658856 9.714487e-02
## beta1    0.946836186 0.0296650665 31.917548 0.000000e+00
## shape    2.593487220 0.4233782921  6.125697 9.028738e-10
```

ZEC. VaR, QQ-plot, ACF



```
##              Estimate   Std. Error   t value   Pr(>|t|)
## ma1    -0.0946069946 0.0742336869 -1.274448 0.20250459
## omega   0.0003551062 0.0003437574  1.033014 0.30159741
## alpha1  0.0905890616 0.0659086801  1.374463 0.16929790
## beta1   0.8555867551 0.0912721535  9.374017 0.00000000
## shape   5.5608863728 2.4332782509  2.285348 0.02229246
```