Time-Series Analysis and Forecasting of the S&P 500 Index

Please download the file "`sp500_full.csv`". It provides the data on daily prices and returns (among other things) of the S&P 500 index from 1957-01-02 to 2021-12-31. The specific function implementations are hidden as the project mainly focuses on the overall methods.

Time Series Modeling and Forecasting

The goal is to fit time series models like ARIMA, ARCH, GARCH, ARMA-GARCH, APARCH to S&P 500 data for inference and forecasting. Use the S&P 500 data from mid-1962 through end of 2021. Load data and useful packages:

```
options(warn = -1)
sp = read.csv("sp500_full.csv", header = T)
idx = which(is.na(sp$sprtrn)==F)     # remove null values
sp = sp[idx, ]
r = sp$sprtrn * 100
t = as.Date(as.character(sp$caldt),"%Y%m%d")
library("forecast");

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

library("tseries")
library("fGarch")

## NOTE: Packages 'fBasics', 'timeDate', and 'timeSeries' are no longer
## attached to the search() path when 'fGarch' is attached.
##
## If needed attach them yourself in your R script by e.g.,
##         require("timeSeries")
```

1. ARMA Model

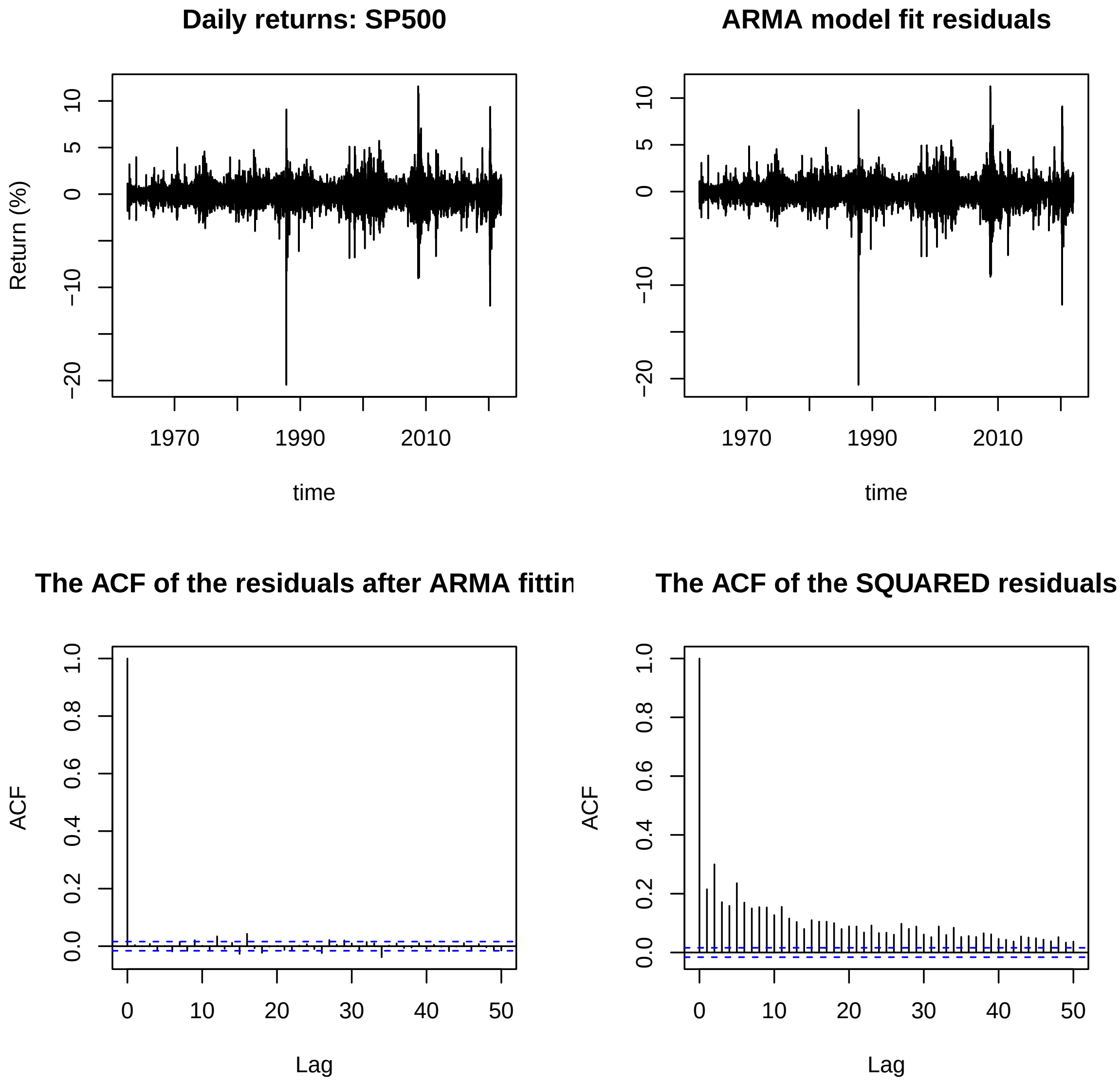
```
fit = auto.arima(r, d=0,max.p=50,max.q=50);
summary(fit)

## Series: r
## ARIMA(1,0,1) with non-zero mean
##
## Coefficients:
##          ar1      ma1    mean
##       0.8269  -0.8436  0.0351
## s.e.  0.0793   0.0759  0.0076
```

```
##
## sigma^2 = 1.06:  log likelihood = -21684.76
## AIC=43377.52   AICc=43377.52   BIC=43407.98
##
## Training set error measures:
##                         ME     RMSE       MAE MPE MAPE      MASE         ACF1
## Training set -0.0001202985 1.029255 0.6864911 NaN  Inf 0.7077847 0.003808404
```

```r
par(mfrow=c(2,2));
plot(t,r,type="l",main="Daily returns: SP500",xlab="time",ylab="Return (%)")
plot(t,fit$residuals,type="l",
    main="ARMA model fit residuals",xlab="time",ylab="")

# ACF of the residuals and their squares
acf(fit$residuals,lag.max = 50,
  main="The ACF of the residuals after ARMA fitting")
acf(fit$residuals^2, lag.max = 50,
  main="The ACF of the SQUARED residuals")
```



From the above two plots, there are several observations:

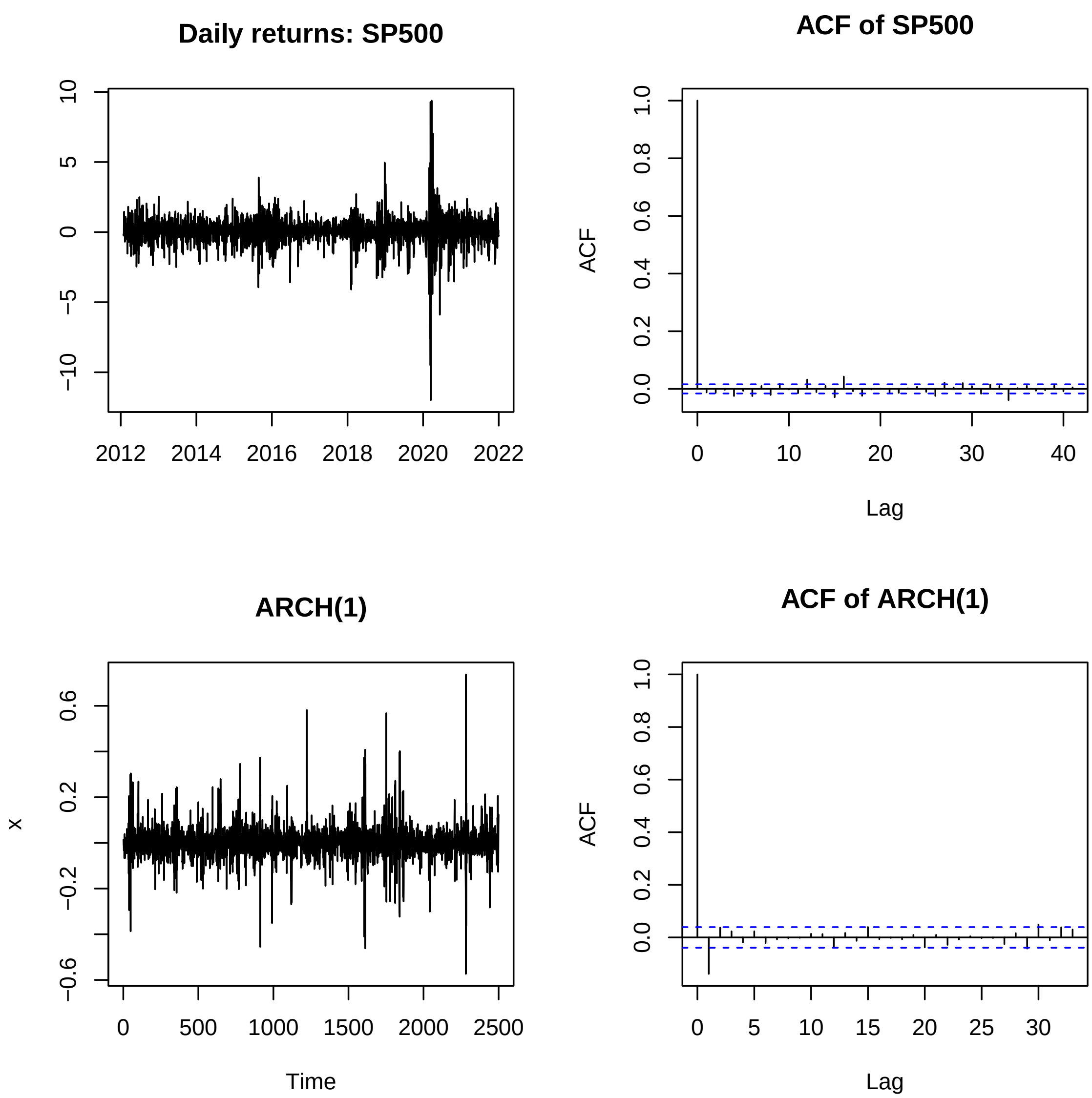
- The empirical autocorrelation function(ACF) of the ARMA residuals for S&P500 seems to indicate a good fit. That is, the residuals are uncorrelated.
- The residuals from fitting the classic linear ARMA time series models appear non-stationary. A period of high-volatility persists for some time.
- The ACF of the squared residuals $\hat{\epsilon}_t^2 = (r_t - \hat{r}_t)^2$ indicates that $\{\hat{\epsilon}_t^2\}$ are correlated with non-linear dependdence. Hence the residuals are dependent.

Thus the ARMA models are not capturing important dynamics in the returns.

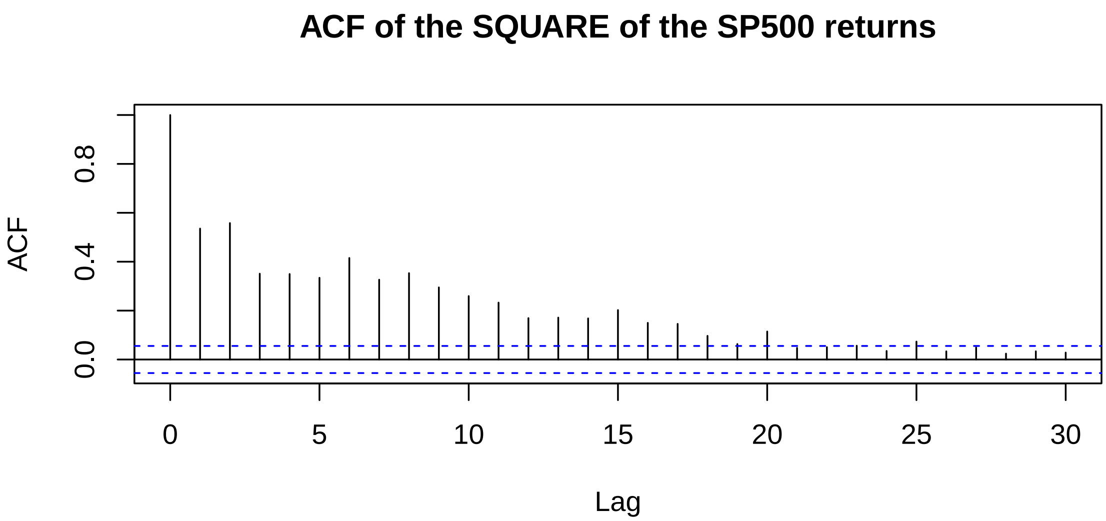
2. ARCH(1) model

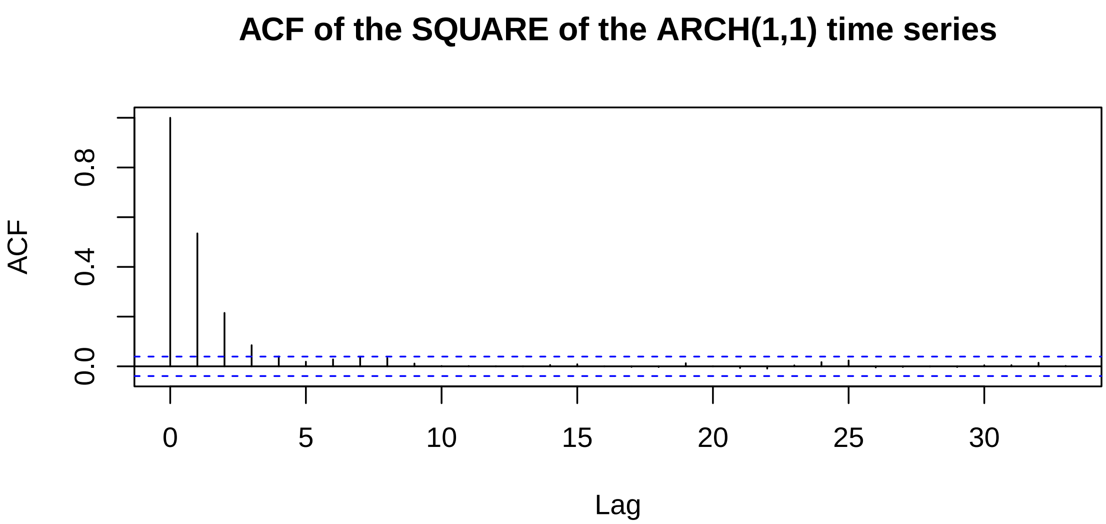
```
alpha0 = 0.001
alpha1 = 0.99
set.seed(123)
spec = garchSpec(model=
      list(omega = alpha0,alpha=alpha1,
           beta=0)); # beta = 0 means ARCH(1) not GARCH(1,1)
x = garchSim(spec,n=10*250); #Simulate ~ 10 years of trading days

par(mfrow=c(2,2));
plot(tail(t,10*250), tail(r,10*250),type="l",
     xlab="",ylab="", main="Daily returns: SP500")
acf(r,main="ACF of SP500");
ts.plot(x, main="ARCH(1)")
acf(x, main="ACF of ARCH(1)")
```




```
par(mfrow=c(2,1));
acf(tail(r,5*250)^2,main="ACF of the SQUARE of the SP500 returns");
acf(x^2,main="ACF of the SQUARE of the ARCH(1,1) time series");
```



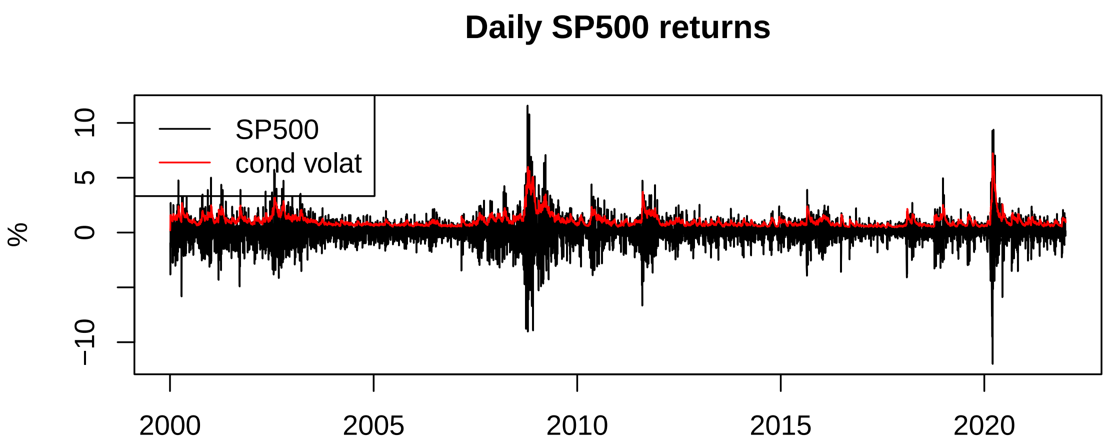


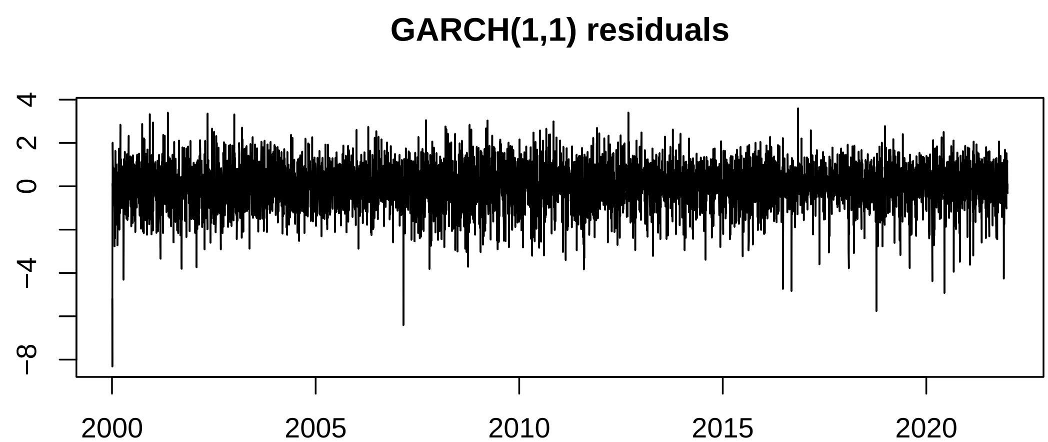
- ARCH(1) has the same phenomenon of uncorrelated returns as ARMA but different phenomenon correlated squared returns.
- The dependence in the market (S&P 500) squared returns is a lot more persistent, which experiences longer time lags.
- ARCH(1) still cannot interpret much information.

3. GARCH(1,1) model

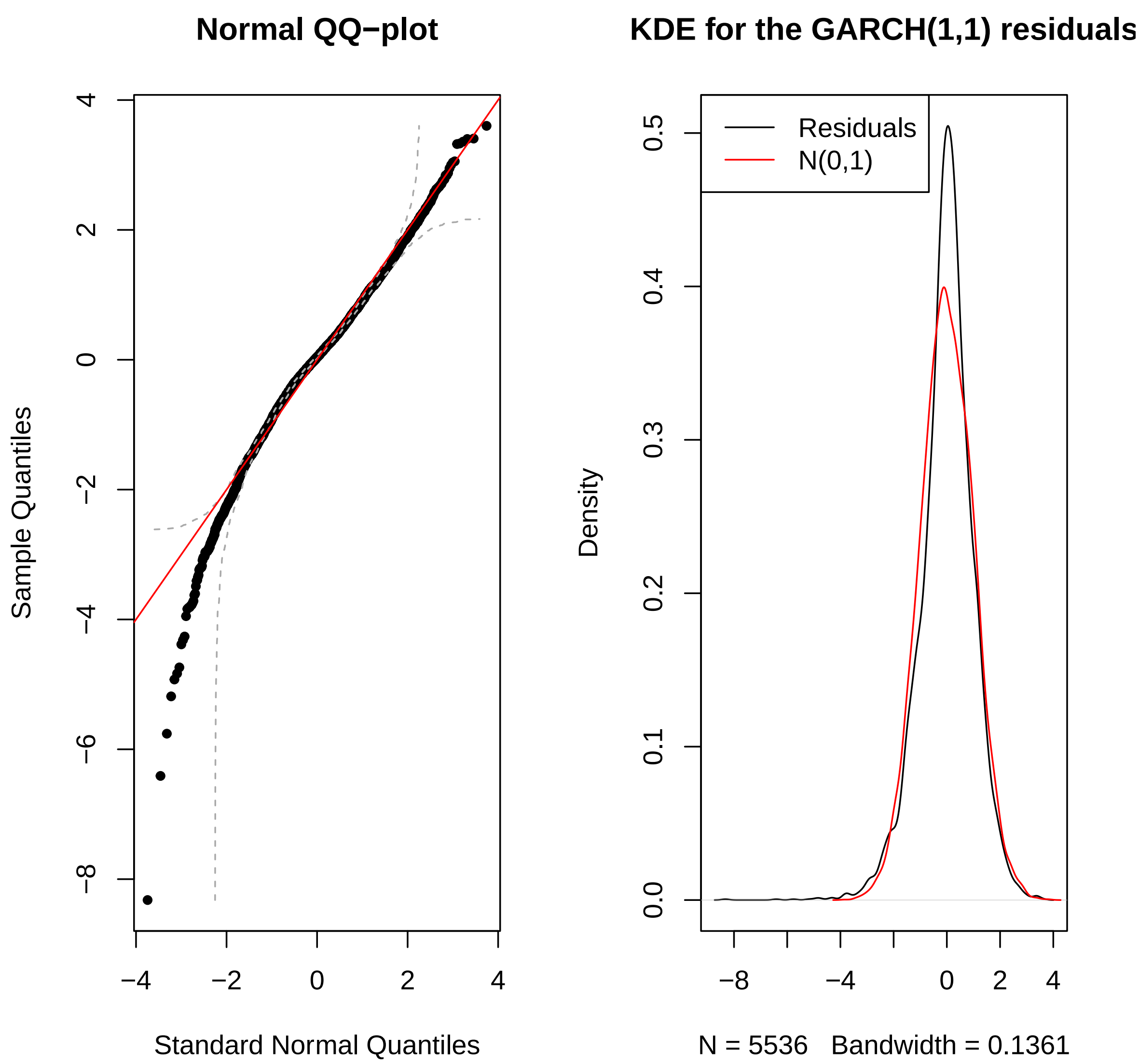
```
idx = which (t>=as.Date("2000-01-01"))
mu.r = mean(r[idx])
fit.garch = fit_simple_garch(data = r[idx]-mu.r)
```

```
par(mfrow=c(2,1));
plot(t[idx],r[idx],main="Daily SP500 returns",
        ylab="%",type="l",xlab="");
lines(t[idx],fit.garch$sigma.t,col="red");
legend("topleft",
        legend = c("SP500","cond volat"),
      col=c("black","red"),lwd=c(1,1))
plot(t[idx],fit.garch$residuals,main="GARCH(1,1) residuals",
        ylab="",type="l",xlab="")
```





```r
par(mfrow=c(1,2));
extRemes::qqnorm(fit.garch$residuals,main="Normal QQ-plot")
abline(a=0,b=1,col="red")
plot(density(fit.garch$residuals),main="KDE for the GARCH(1,1) residuals")
lines(density(rnorm(1e4)),col="red");
legend("topleft",legend=c("Residuals","N(0,1)"),
       col=c("black","red"),lwd=c(1,1))
```



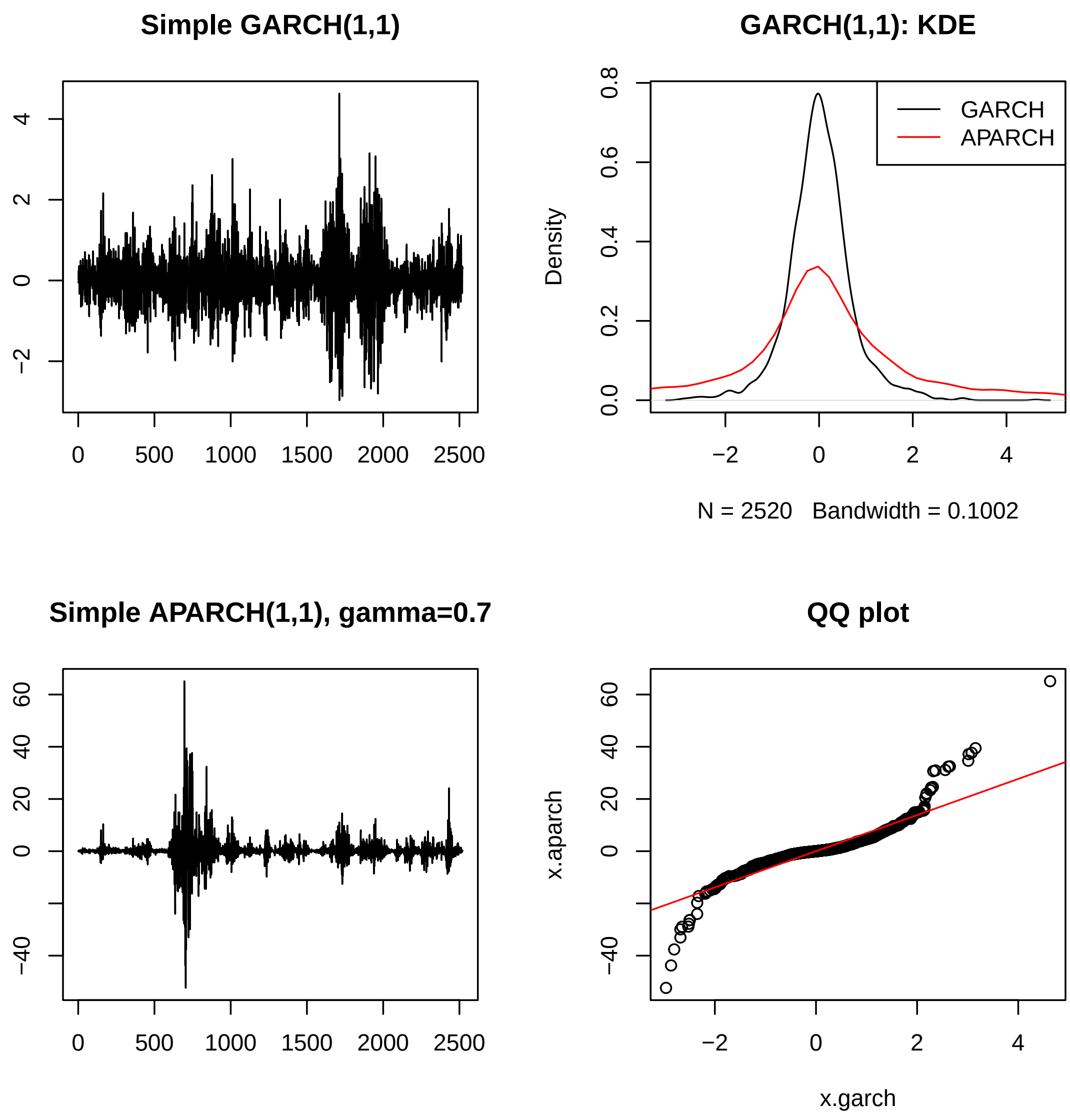
- From the normal QQ-plot, GARCH(1,1) residuals have a heavier left tail than Gaussian. That is, the losses have heavier tails than the gains. Therefore, extreme losses have greater effect on increasing the volatility than extreme gains.

- In practice, extreme losses have greater effect on increasing the volatility than extreme gains.
- This asymmetry phenomenon is not reflected in the simple GARCH(1,1).

4. APARCH model

```r
# the implementation of a simple APARCH(1,1) model
sim_simple_APARCH <- function(n, s0=0.00001, x0=0.00001,
                              a0=0.01, a=0.2, b=0.79,
                              delta=2, gamma=0.7){
s.delta = abs(s0)^delta;
x = x0;
for (t in c(2:(n+1))){
  s.delta = a0+a*abs((abs(x[t-1])-gamma*x[t-1]))^delta + b*s.delta;
  x =c(x,s.delta^(1/delta)*rnorm(1));
}
return(x[-1])}

set.seed(123)
x.garch = sim_simple_APARCH(n=2520, gamma=0);
set.seed(123)
x.aparch = sim_simple_APARCH(n=2520, gamma= 0.7);
par(mfrow=c(2,2));
plot(x.garch,xlab="",ylab="",type="l",
     main="Simple GARCH(1,1)");
plot(density(x.garch),main="GARCH(1,1): KDE")
lines(density(x.aparch),col="red");
legend("topright",legend=c("GARCH","APARCH"),
       col=c("black","red"),lwd=c(1,1))
plot(x.aparch,xlab="",ylab="",type="l",
     main="Simple APARCH(1,1), gamma=0.7");
stats::qqplot(x.garch,
              x.aparch,main="QQ plot");
abline(a=0,b=sd(x.aparch)/sd(x.garch),col="red")
```



Test/validation...

```
fit.aparch = fit_simple_APARCH(data = r[idx]-mu.r)
par(mfrow=c(2,2));
extRemes::qqnorm(fit.aparch$residuals,main="Normal QQ-plot")
abline(a=0,b=1,col="red")
plot(density(fit.aparch$residuals),main="KDE for the APARCH(1,1) residuals")
lines(density(rnorm(1e4)),col="red");
legend("topleft",legend=c("APARCH","N(0,1)"),
       col=c("black","red"),lwd=c(1,1))

extRemes::qqnorm(fit.garch$residuals,main="Normal QQ-plot")
abline(a=0,b=1,col="red")
```

```
plot(density(fit.garch$residuals),main="KDE for the GARCH(1,1) residuals")
lines(density(rnorm(1e4)),col="red");
legend("topleft",legend=c("GARCH","N(0,1)"),
       col=c("black","red"),lwd=c(1,1))
```