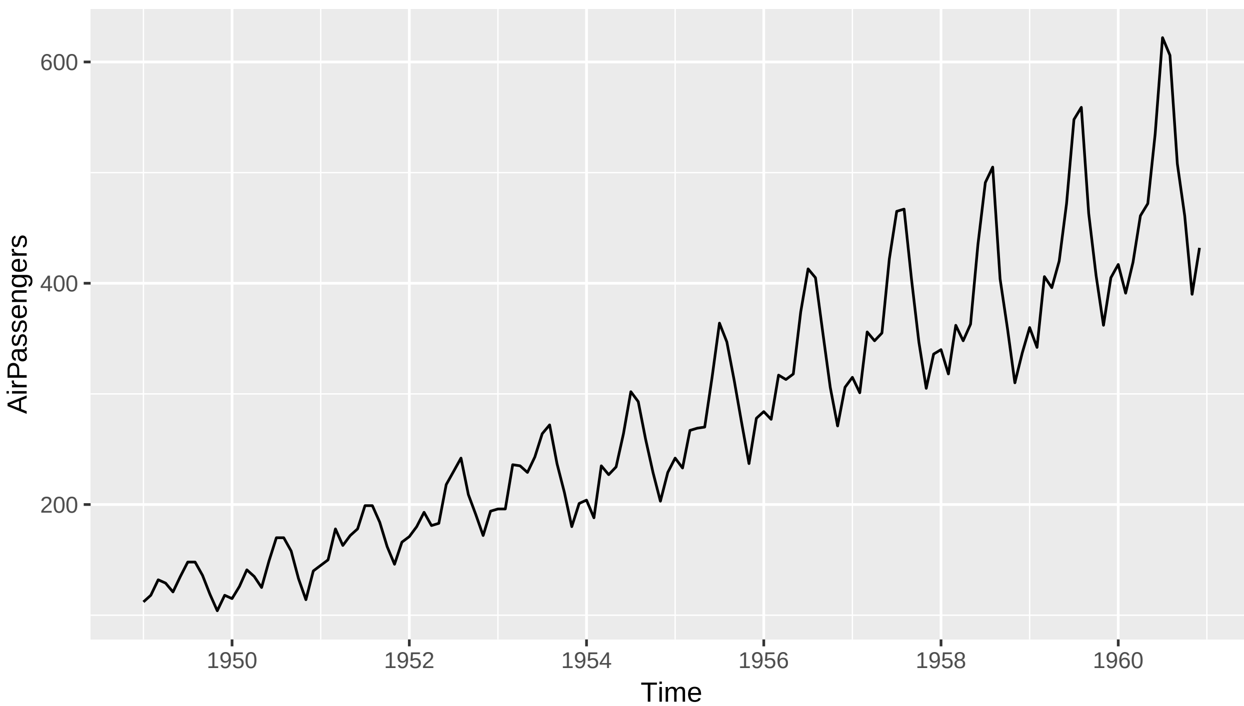
STAT 436 / 536 - Lecture 18: Key

Heteroskedastic Variance Models

- Hetoreskedastic variance models account for non-constant variation.

- Recall the airline passenger's dataset, where the variability in the time series increases in time.

```
data(AirPassengers)
autoplot(AirPassengers)
```

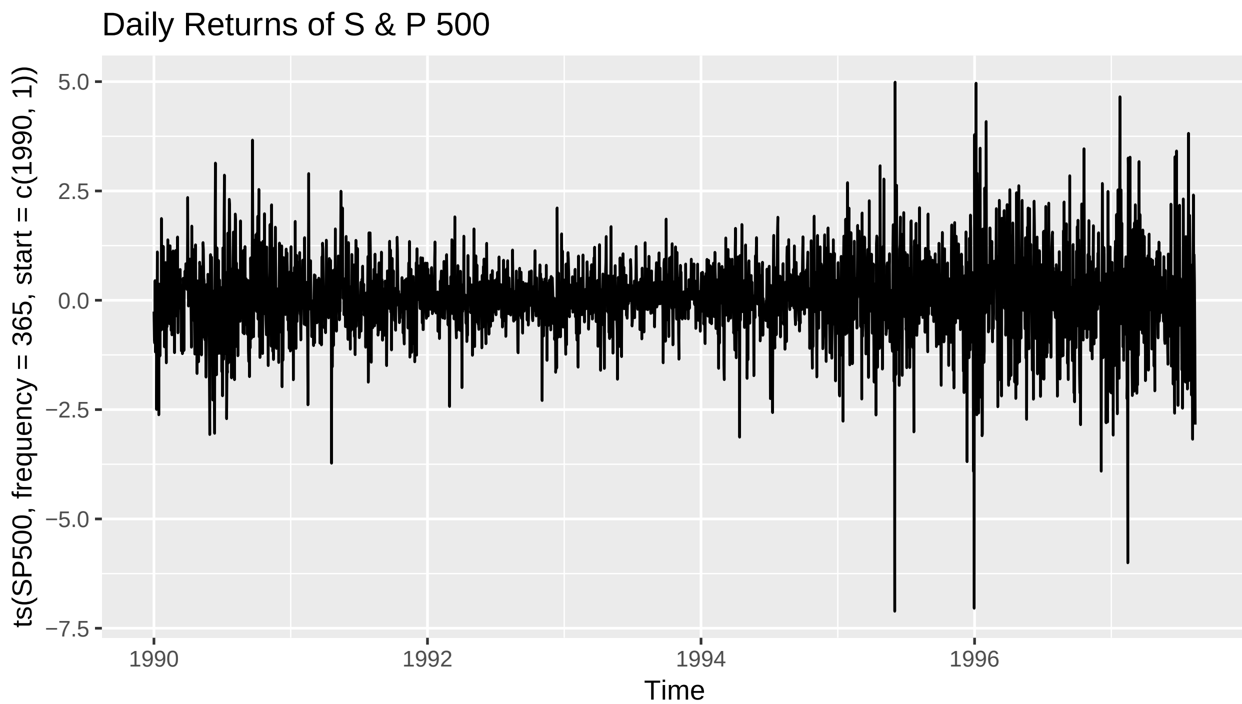


- Another example includes the daily returns from the S & P 500 index. Daily returns are displayed as $100 \times \log(\frac{y_t}{y_{t-1}})$, where y_t is the value of the S&P 500 on day t.

- This series exhibits times of increased variability, which is referred to as volatility.

- Time series with periods of increased volatility are referred to as *conditional heteroskedastic* models.

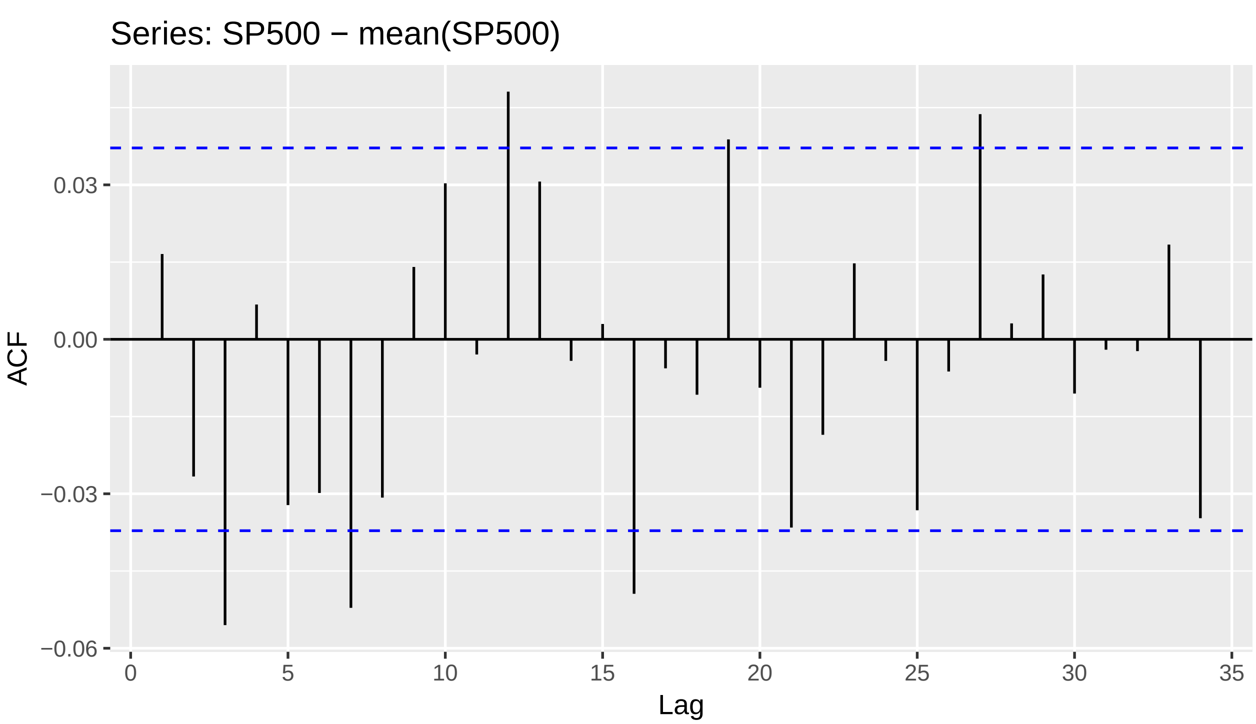
- What are the implications of conditional heteroskedastic model?



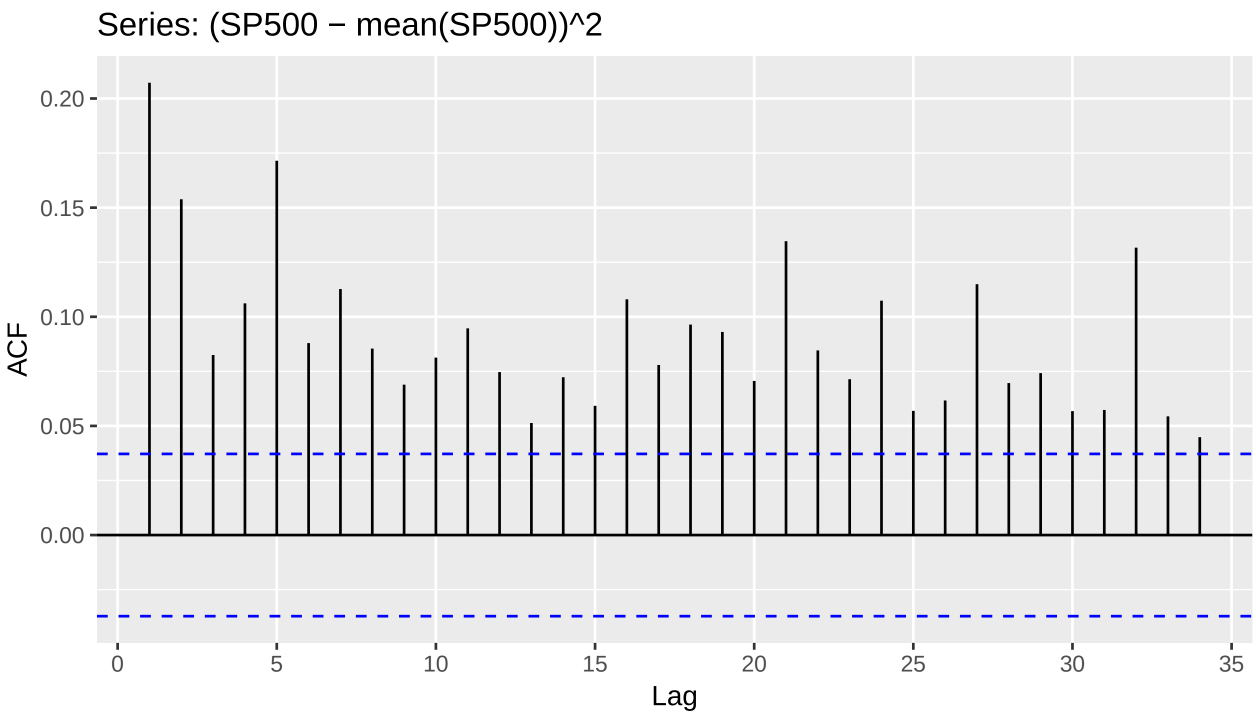
- Models with stochastic variability are not stationary, by definition, as the variance changes in time.

- Stochastic Volatility is typically not obvious from an ACF plot alone, but can be visualized with the squared values of the sequence (that have been de-meaned). why?

```
ggAcf(SP500 - mean(SP500))
```



```r
ggAcf((SP500 - mean(SP500))^2)
```



Modeling Conditional Heteroskedasticity

- A model that accounts for changing variance is necessary for this situation and a common way to do this is to use an autoregressive process.

- Let $\{\epsilon_t\}$ be a first-order autoregressive conditional heteroskedastic (ARCH(1)), where $\{w_t\}$ is white noise with zero mean and variance of 1, then

$$\epsilon_t = w_t\sqrt{\alpha_0 + \alpha_1\epsilon_{t-1}^2}$$

- The variance of ϵ_t can be calculated as:

$$\begin{aligned}
Var(\epsilon_t) &= E(\epsilon_t^2) - E(\epsilon_t)^2 \\
&= E(w_t^2)E(\alpha_0 + \alpha_1\epsilon_{t-1}^2) \\
&= E(\alpha_0 + \alpha_1\epsilon_{t-1}^2) \\
&= \alpha_0 + \alpha_1 Var(\epsilon_{t-1})
\end{aligned}$$

- ARCH models should be applied to an uncorrelated series with no trends or seasonal components. Hence, we need to control for ARIMA type procedures first.

- ARCH models can also include p-th order lags, where

$$\epsilon_t = w_t \sqrt{\alpha_0 + \sum_{i=1}^{p} \alpha_i \epsilon_{t-i}^2}$$

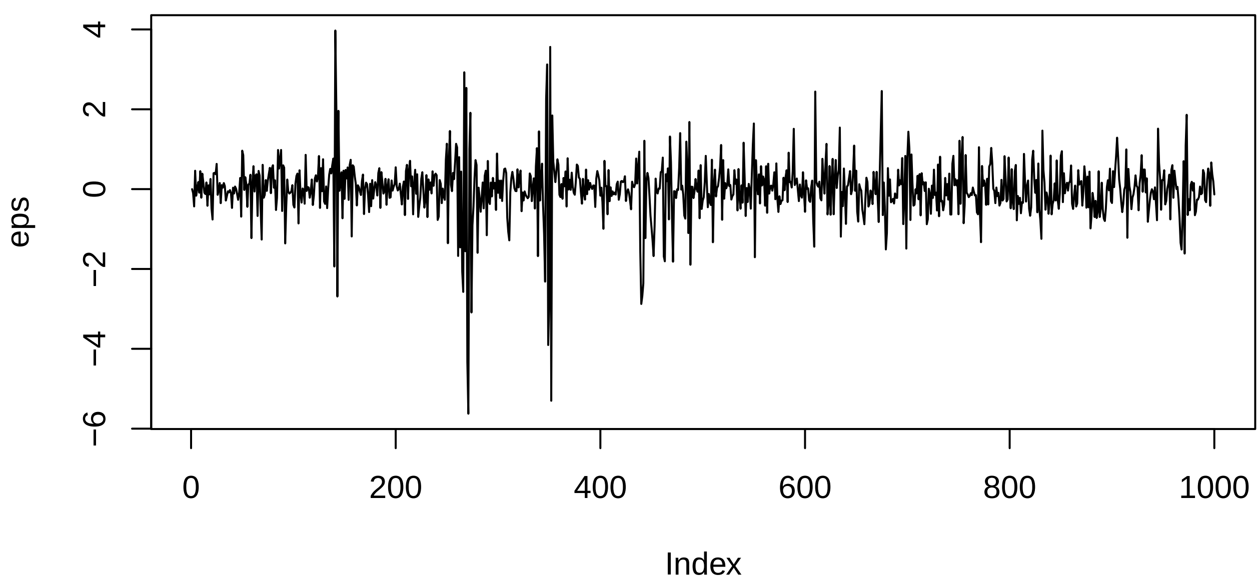
- Furthermore, a more flexible model known as Generalized ARCH or GARCH(p,q) model can be specified. First, define $\epsilon_t = w_t \sqrt{h_t}$, then

$$h_t = \alpha_0 + \sum_{i=1}^{p} \alpha_i \epsilon_{t-i}^2 + \sum_{j=1}^{1} \beta_j h_{t-j}$$

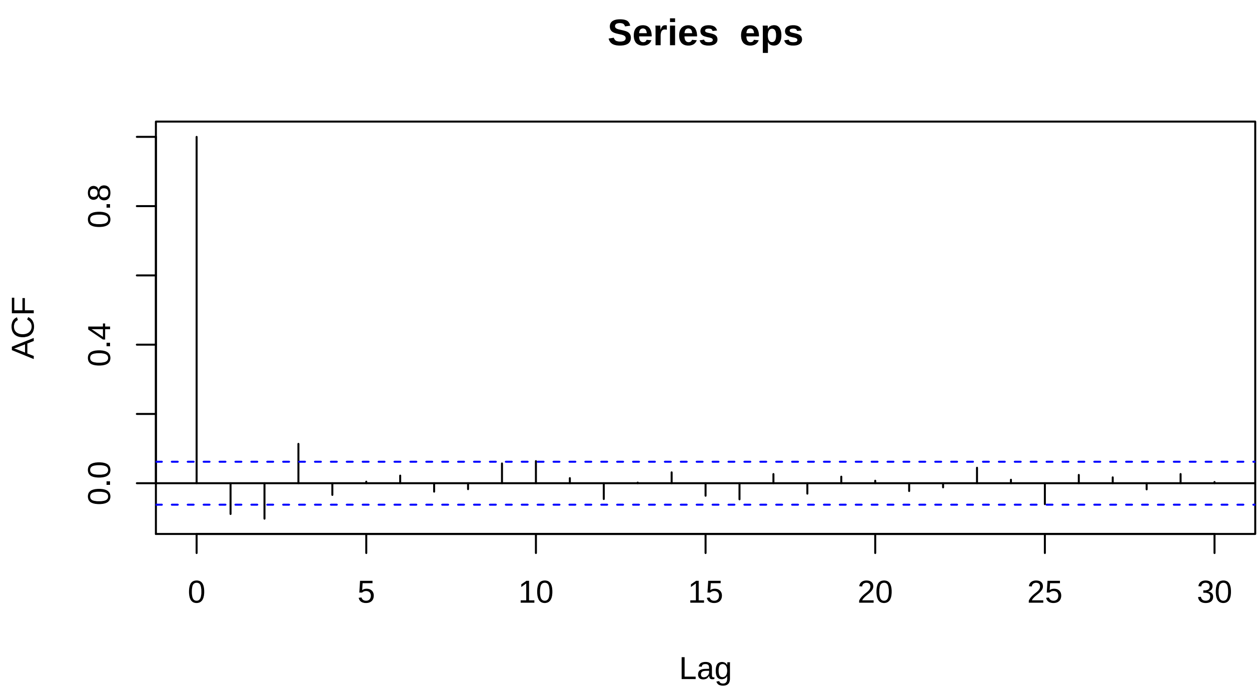
Simulating GARCH Models

- To demonstrate the structure of GARCH models and the impacts of α and β, consider the following simulation.

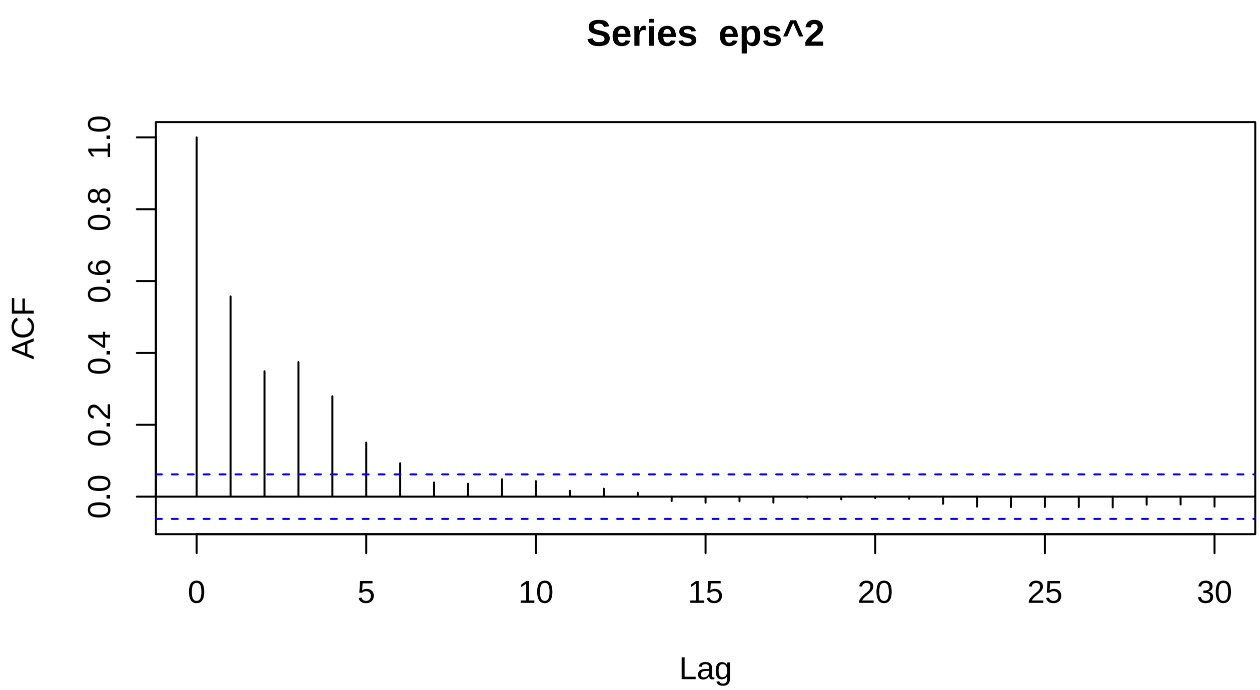
```
set.seed(12042018)
alpha0 <- .1
alpha1 <- .9
beta1 <- 0

num.pts <- 1000

w <- rnorm(num.pts)
eps <- rep(0, num.pts)
h <- rep(0, num.pts)

for (time.pt in 2:num.pts){
  h[time.pt] <- alpha0 + alpha1 * (eps[time.pt - 1]^2) + beta1 * h[time.pt - 1]
  eps[time.pt] <- w[time.pt] * sqrt(h[time.pt])
}
plot(eps, type = 'l')
```



```r
acf(eps)
```




```r
acf(eps^2)
```



- Modify the values for α_0, α_1, and β_1 to assess the impacts of these parameters.

Fitting GARCH Models

- The `garch()` function in the `tseries` package can be used to fit GARCH models.

```
g1 <- garch(SP500, trace = F, grad = 'numerical')
g1
```

```
##
## Call:
## garch(x = SP500, trace = F, grad = "numerical")
##
## Coefficient(s):
##       a0        a1        b1
## 0.004292  0.050041  0.946785
```

```
confint(g1)
```

```
##          2.5 %      97.5 %
## a0 0.002171893 0.006411307
## a1 0.041262969 0.058819194
## b1 0.937377728 0.956191732
```

```
g2 <- garch(eps, trace = F, grad = 'numerical')
g2
```

```
##
## Call:
## garch(x = eps, trace = F, grad = "numerical")
##
## Coefficient(s):
##        a0         a1         b1
## 3.323e-01  2.110e-01  1.432e-12
```

```
confint(g2)
```

```
##          2.5 %     97.5 %
## a0  0.28030034 0.38436748
## a1  0.17021501 0.25180369
## b1 -0.05398036 0.05398036
```

- The `rugarch` package in R contains the functionality to specify the mean structure of the model, with covariates, and fit a GARCH model to the error terms.

- GARCH models do not influence the mean prediction (point estimate) in most situations, but do impact the width of the prediction intervals.

State Space Models

- Recall, the generic state space model formulation.

$$\begin{aligned} y_t &= F_t x_t + v_t \\ x_t &= G_t x_{t-1} + w_t, \end{aligned}$$

where, in particular, $v_t \sim N(0, V_t)$. Hence, the model can naturally handle heteroskedasticity.