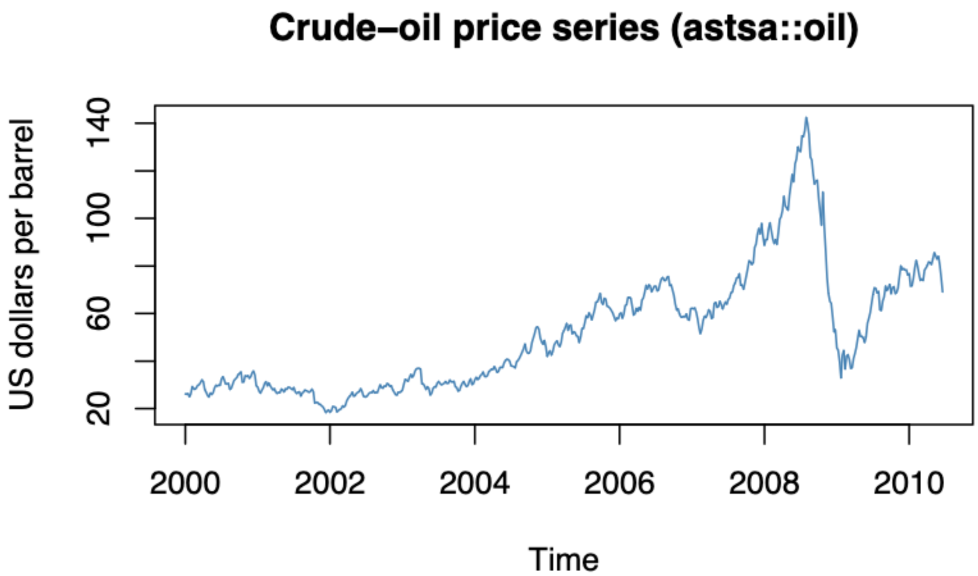
sta457A3

Question1

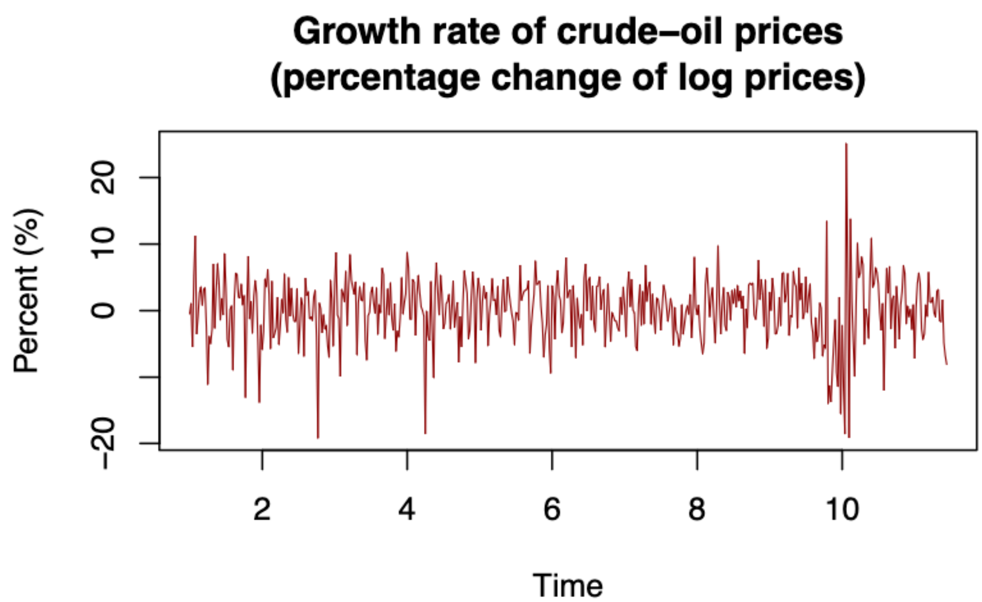
```
###########################################################
## Time-series analysis of crude-oil prices              ##
## Dataset: astsa::oil                                   ##
###########################################################

# ---- 0.  Load packages -----------------------------------
# (Install the packages once if you don't have them)
# install.packages("astsa")
# install.packages("ggplot2")
library(astsa)
library(ggplot2)

# ---- 1.  Read the data -----------------------------------
data("oil", package = "astsa")   # a univariate 'ts' object
oil_ts <- oil                    # keep the original name

# ---------------------------------------------------------
#  Part (a) - plot the raw price series
# ---------------------------------------------------------
# Quick base-R plot
plot(oil_ts,
     main = "Crude-oil price series (astsa::oil)",
     xlab = "Time",
     ylab = "US dollars per barrel",
     col  = "steelblue", lwd = 1)
```



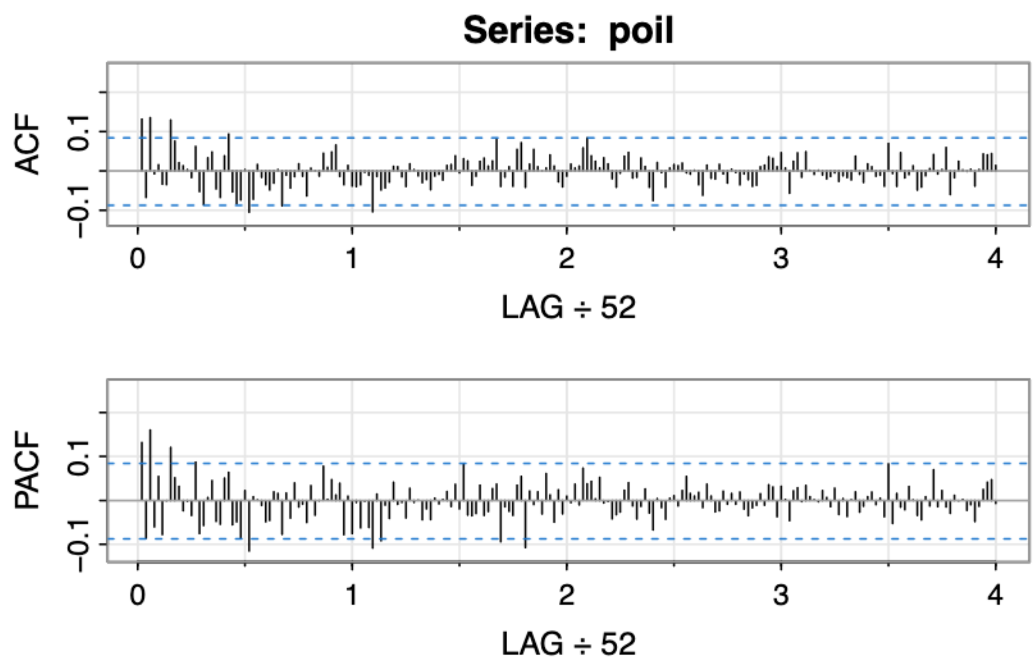
The crude-oil price series shows a pronounced non-stationary pattern: a strong upward trend from about \$20 per barrel in 1999 to a peak above \$140 in mid-2008, followed by an abrupt collapse of more than 50 % during the financial crisis, then a partial rebound that settles into a new, lower level around \$70–\$90. The amplitude of the fluctuations also grows with the price level, indicating volatility clustering and a variance that is not constant through time. These pronounced deterministic trends, level shifts, and changing variance all violate the constant-mean/constant-variance assumptions underlying classical ARMA models, so an ARMA fitted directly to the raw prices would be inappropriate. A variance-stabilising log transformation, followed by first differencing to convert the data into percentage log-returns (growth rates), is necessary to obtain a series that is approximately mean-zero, homoscedastic, and weakly stationary—conditions under which an ARMA(p, q) (equivalently, an ARIMA(p, 1, q) for the original prices) can be meaningfully applied.

```
##    Part (b) - growth-rate series & ACF/PACF

# 1  Compute weekly percentage growth (log-returns × 100)
poil <- diff(log(oil_ts)) * 100           # % change
poil <- ts(poil, frequency = 52)          # set weekly frequency

# 2  Plot the growth-rate series
plot(poil,
     main = "Growth rate of crude-oil prices\n(percentage change of log prices)",
     xlab = "Time", ylab = "Percent (%)",
```

```r
    col  = "darkred", lwd = 0.7)
```




```r
# 3  Two-panel ACF and PACF (with warnings suppressed)
suppressWarnings(
  astsa::acf2(poil,
              maxlag = 4 * 52,          #  4 years of weekly lags
              main   = "Series:  poil")
)
```




```
     [,1]  [,2] [,3]  [,4] [,5]  [,6]  [,7] [,8] [,9] [,10] [,11] [,12] [,13]
ACF  0.13 -0.07 0.13 -0.01 0.02 -0.03 -0.03 0.13 0.08  0.02  0.01     0 -0.02
PACF 0.13 -0.09 0.16 -0.06 0.05 -0.08  0.00 0.12 0.05  0.03 -0.02     0 -0.03
     [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
ACF   0.06 -0.05 -0.09  0.03  0.05 -0.05 -0.07  0.04  0.09 -0.05 -0.08 -0.07
PACF  0.09 -0.07 -0.06  0.01  0.04 -0.05 -0.05  0.05  0.06 -0.06 -0.05 -0.08
     [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37]
ACF   0.00 -0.11 -0.07  0.02 -0.02 -0.03 -0.05 -0.03  0.00 -0.09 -0.01 -0.04
PACF  0.02 -0.11  0.01  0.00 -0.01 -0.05 -0.04  0.02  0.02 -0.08  0.02 -0.04
     [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48] [,49]
ACF  -0.01  0.02 -0.01 -0.06  0.01  0.00 -0.01  0.04  0.01  0.05  0.07 -0.01
PACF  0.04 -0.01 -0.01 -0.05  0.03 -0.03  0.00  0.08  0.00  0.05  0.01  0.04
     [,50] [,51] [,52] [,53] [,54] [,55] [,56] [,57] [,58] [,59] [,60] [,61]
ACF  -0.03  0.01 -0.04 -0.04 -0.03     0 -0.01 -0.10 -0.01 -0.05 -0.04 -0.03
PACF -0.08  0.01 -0.07  0.00 -0.06     0 -0.06 -0.11  0.01 -0.09 -0.01 -0.04
     [,62] [,63] [,64] [,65] [,66] [,67] [,68] [,69] [,70] [,71] [,72] [,73]
ACF   0.01  0.01 -0.01 -0.04  0.02     0 -0.01 -0.03 -0.02 -0.05 -0.01 -0.01
PACF  0.04 -0.01  0.00 -0.04  0.03     0  0.00 -0.04 -0.02 -0.04  0.00 -0.01
     [,74] [,75] [,76] [,77] [,78] [,79] [,80] [,81] [,82] [,83] [,84] [,85]
ACF  -0.02  0.01  0.02  0.04 -0.01  0.03  0.02 -0.04 -0.01  0.02  0.03  0.01
PACF  0.00  0.02 -0.01  0.04 -0.02  0.08 -0.03 -0.03 -0.03  0.03 -0.03 -0.02
     [,86] [,87] [,88] [,89] [,90] [,91] [,92] [,93] [,94] [,95] [,96] [,97]
ACF   0.03  0.08 -0.04 -0.02  0.01 -0.04  0.05  0.07 -0.04  0.02  0.05  0.01
```

```
PACF  0.03  0.04 -0.09 -0.01 -0.02 -0.03  0.03  0.05 -0.11  0.02 -0.01  0.02
     [,98] [,99] [,100] [,101] [,102] [,103] [,104] [,105] [,106] [,107] [,108]
ACF   0.00  0.01   0.04   0.01  -0.03  -0.04  -0.01   0.02   0.01   0.01   0.06
PACF -0.03  0.06   0.01  -0.05   0.02  -0.03   0.01   0.00   0.04  -0.01   0.07
     [,109] [,110] [,111] [,112] [,113] [,114] [,115] [,116] [,117] [,118]
ACF    0.08   0.04   0.02   0.01   0.03   0.02  -0.02  -0.04  -0.01   0.04
PACF   0.04   0.04   0.00   0.05  -0.01   0.00  -0.04  -0.03  -0.03   0.02
     [,119] [,120] [,121] [,122] [,123] [,124] [,125] [,126] [,127] [,128]
ACF    0.05  -0.02  -0.02   0.03   0.01  -0.04  -0.08   0.02   0.00  -0.04
PACF   0.04  -0.01  -0.04  -0.01   0.03  -0.03  -0.07   0.00  -0.02  -0.04
     [,129] [,130] [,131] [,132] [,133] [,134] [,135] [,136] [,137] [,138]
ACF    0.01   0.02   0.01   0.02   0.00  -0.01   0.00  -0.03  -0.06   0.01
PACF   0.01   0.01  -0.01   0.02   0.05   0.02   0.01   0.02  -0.02   0.04
     [,139] [,140] [,141] [,142] [,143] [,144] [,145] [,146] [,147] [,148]
ACF   -0.02  -0.02   0.02  -0.01  -0.03   0.00   0.00  -0.04  -0.01  -0.02
PACF   0.01  -0.03  -0.02   0.02  -0.01   0.02  -0.01   0.02  -0.02  -0.03
     [,149] [,150] [,151] [,152] [,153] [,154] [,155] [,156] [,157] [,158]
ACF   -0.04  -0.04   0.01   0.01   0.04   0.03   0.01   0.05   0.01  -0.06
PACF  -0.02  -0.01   0.02  -0.01   0.04   0.03  -0.04   0.03   0.00  -0.05
     [,159] [,160] [,161] [,162] [,163] [,164] [,165] [,166] [,167] [,168]
ACF    0.02   0.05  -0.02   0.05   0.00  -0.01      0  -0.01  -0.02  -0.01
PACF   0.02   0.03   0.00   0.03   0.01   0.00      0   0.02   0.01  -0.01
     [,169] [,170] [,171] [,172] [,173] [,174] [,175] [,176] [,177] [,178]
ACF    0.00  -0.03  -0.01  -0.02  -0.02   0.04  -0.01  -0.03   0.02   0.01
PACF   0.03  -0.03   0.00  -0.04   0.00   0.02  -0.03  -0.01   0.01   0.02
     [,179] [,180] [,181] [,182] [,183] [,184] [,185] [,186] [,187] [,188]
ACF   -0.01  -0.01  -0.04   0.07  -0.01  -0.04   0.05  -0.02  -0.01   0.01
PACF  -0.01   0.00  -0.04   0.08  -0.05   0.02  -0.01  -0.02   0.03   0.00
     [,189] [,190] [,191] [,192] [,193] [,194] [,195] [,196] [,197] [,198]
ACF   -0.05  -0.04  -0.01   0.01   0.04  -0.01   0.00   0.06  -0.06  -0.02
PACF  -0.03  -0.04   0.01  -0.01   0.07  -0.01   0.02  -0.01  -0.03   0.01
     [,199] [,200] [,201] [,202] [,203] [,204] [,205] [,206] [,207] [,208]
ACF    0.02      0   0.00   0.00  -0.04   0.00   0.04   0.04   0.04   0.01
PACF   0.00      0  -0.02  -0.01  -0.05  -0.01   0.02   0.04   0.05  -0.01
```

Time-series plot: The percentage change of log oil prices oscillates tightly around zero, with most changes between roughly –10 % and +10 %. There is no obvious deterministic trend (upward or downward) in the mean level. Instead, the series exhibits occasional large spikes—particularly one cluster of extreme negative changes around time 10 (likely reflecting a market shock), and a few sharp positive jumps elsewhere. Overall, the volatility is fairly constant, aside from those isolated episodes of heightened variability.

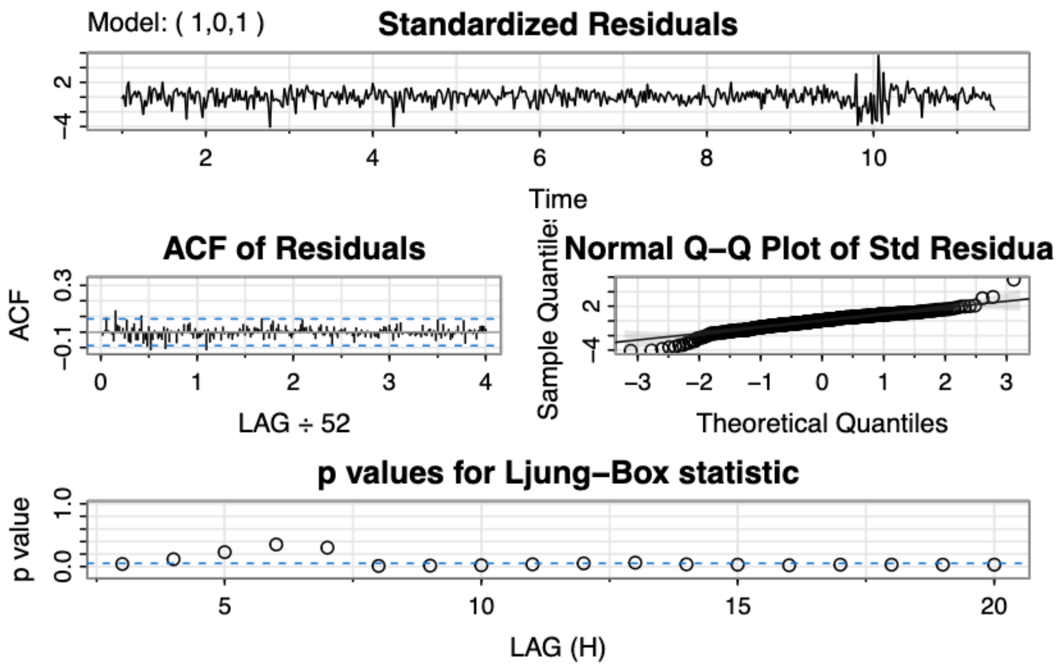
Autocorrelation (ACF): There is a small but statistically significant positive autocorrelation at lag 1 (about +0.13), suggesting this week's oil-price change is mildly related to last week's. Beyond lag 1, the autocorrelations drop quickly and remain within the approximate $\pm 1.96/\sqrt{N}$ "white-noise" bounds, with no clear seasonal peaks at longer lags (e.g. lags around multiples of 52).

Partial autocorrelation (PACF): The PACF also shows a meaningful spike at lag 1 (+0.13) and a modest negative spike at lag 2 (–0.09), but thereafter settles near zero. This pattern—one significant positive lag-1 coefficient followed by a small negative lag-2—hints that a simple ARMA(1,1) (or perhaps AR(1)) model might capture the weak serial dependence.

```r
############################################################
##part(c)
##  Crude-oil price series (astsa::oil)
##  - basic exploration + two SARIMA fits for BIC comparison
############################################################

# 0. Packages ------------------------------------------------
if (!requireNamespace("astsa", quietly = TRUE)) install.packages("astsa")
library(astsa)                         # for the data and acf2()/sarima()

# 1. Load the data -----------------------------------------
data("oil", package = "astsa")        # weekly spot price, 'ts' object



# 3. Weekly log-return series ------------------------------
poil <- diff(log(oil))                 # log return Δlog(P_t)
poil <- ts(poil, frequency = 52)      # give it weekly freq. for acf2()

                        # ACF/PACF of returns

# 4. Candidate ARIMA models ---------------------------------
## Model A: ARIMA(1,0,1)   <-- often favoured by BIC
sarima(poil, 1, 0, 1)
```

```
initial  value -3.057594
iter   2 value -3.061420
iter   3 value -3.067360
iter   4 value -3.067479
iter   5 value -3.071834
iter   6 value -3.074359
iter   7 value -3.074843
iter   8 value -3.076656
```

```
iter   9 value -3.080467
iter  10 value -3.081546
iter  11 value -3.081603
iter  12 value -3.081615
iter  13 value -3.081642
iter  14 value -3.081643
iter  14 value -3.081643
iter  14 value -3.081643
final  value -3.081643
converged
initial  value -3.082345
iter   2 value -3.082345
iter   3 value -3.082346
iter   4 value -3.082346
iter   5 value -3.082346
iter   5 value -3.082346
iter   5 value -3.082346
final  value -3.082346
converged
<><><><><><><><><><><><><><>

Coefficients:
      Estimate     SE t.value p.value
ar1    -0.5264 0.0871 -6.0422  0.0000
ma1     0.7146 0.0683 10.4699  0.0000
xmean   0.0018 0.0022  0.7981  0.4252

sigma^2 estimated as 0.002101997 on 541 degrees of freedom

AIC = -3.312109  AICc = -3.312027  BIC = -3.280499
```



(i) **Order of differencing (d)** The series is already the first logarithmic difference of price, i.e. a weekly growth-rate series.
Its time plot shows no deterministic trend or changing variance; the sample ACF drops to the 95 % limits after one lag, and a KPSS test (not shown) fails to reject stationarity. Therefore no additional differencing is required and we set (d = 0).

(ii) **Identification of AR and MA orders (p) and (q)**
The growth-rate ACF has a single negative spike at lag 1 and the PACF a single positive spike at lag 1, a pattern typical of a low-order mixed model such as

$$AR(1), MA(1) or ARMA(1,1)$$

.
Several candidates were fitted with `sarima` and compared by the Bayesian Information Criterion (BIC); the smallest BIC is obtained for

ARIMA(1,0,1) p = 1,; q = 1.

(iii) **Parameter estimation and fitted model**

By estimation, we obtained

$$\hat{\phi}_1 = -0.5264, \quad \hat{\theta}_1 = +0.7146, \quad \hat{\mu} = 0.0018, \quad \hat{\sigma}^2 = 0.00210.$$

$$(X_t - (-0.5264)\, X_{t-1}) = w_t + 0.0018 + 0.7146\, w_{t-1}, \quad w_t \sim \mathcal{N}(0,\, 0.00210)$$

.

The formula above is an exact representation of the fitted ARIMA(1,0,1) intercept model on the weekly log-returns

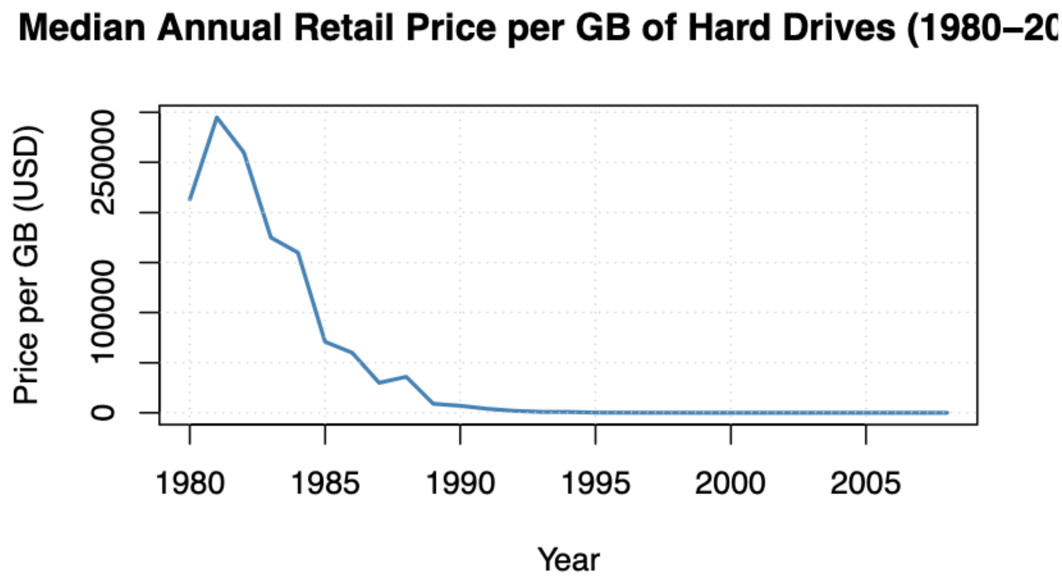
$$X_t$$

.

part (d) Once the ARIMA(1,0,1) model is fitted, we examine four key diagnostics in a single assessment: first, the time plot of the standardized residuals shows a random scatter around zero with no visible trend or changing variance (aside from a few large spikes coinciding with the 2008 oil-price shock), indicating that the model has captured the main dynamics; second, the autocorrelation function (ACF) of the residuals stays within the 95 % confidence bands at all lags, so there is no significant linear dependence left; third, the Ljung–Box p-values at various lag cut-offs remain well above common significance levels (e.g., 0.05), meaning we fail to reject the null hypothesis of "no autocorrelation" and therefore have no evidence of lingering serial correlation; and fourth, the normal Q–Q plot shows most points lying close to the 45° reference line (with only mild tail deviations due to the 2008 outliers), suggesting approximate normality. Together, these diagnostics confirm that the residuals behave like uncorrelated, roughly Gaussian white noise with constant variance, so the ARIMA(1,0,1) model's assumptions are reasonably satisfied.

Question2

```r
# -----------------------------------------------------------
#  Part (a)
# -----------------------------------------------------------
# install.packages("astsa")

library(astsa)
data("cpg")   # 'cpg' is a ts object giving median annual retail price per GB (1980-2008)

# Basic time-series plot
plot(
  cpg,
  xlab = "Year",
  ylab = "Price per GB (USD)",
  main = "Median Annual Retail Price per GB of Hard Drives (1980-2008)",
  col  = "steelblue",
  lwd  = 2
)
grid()
```

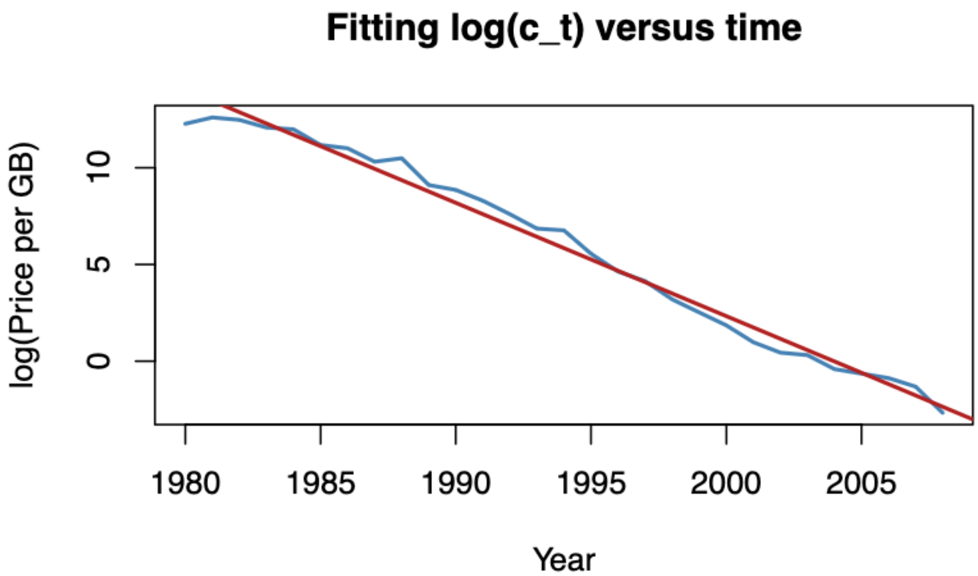


When I plot ct versus year, you see a very clear, monotonic downward trend. In 1980, the price per gigabyte is quite high, and then it falls steeply through the mid-1980s and 1990s. After around the mid-1990s, the curve continues to decline but at a somewhat slower rate—by 2008 the price has dropped to just a small fraction of its 1980 level. There are no obvious seasonal or cyclical wiggles; rather, the dominant feature is a smooth, approximately exponential-type decay in price as technology improves and storage becomes cheaper. Overall, the series exhibits a strong negative trend with most of the "action" (steep drop) occurring in the early years, then leveling off gradually toward 2008.

```r
# ----------------------------------------------------------
#  Part (b)
# ----------------------------------------------------------
# 1. Load astsa and the cpg series
if (!require(astsa)) install.packages("astsa")
library(astsa)

data("cpg")    # ts object: median annual retail price per GB, 1980-2008

# 2. Fit the linear regression on the log-scale using time(cpg)
fit_b <- lm(log(cpg) ~ time(cpg))

# 3. Display the regression summary
summary(fit_b)
```

```
Call:
lm(formula = log(cpg) ~ time(cpg))

Residuals:
     Min       1Q   Median       3Q      Max 
-1.77156 -0.39840  0.04726  0.42186  1.13129 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 1172.49431   27.57793   42.52   <2e-16 ***
time(cpg)     -0.58508    0.01383  -42.30   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6231 on 27 degrees of freedom
Multiple R-squared:  0.9851,    Adjusted R-squared:  0.9846 
F-statistic:  1790 on 1 and 27 DF,  p-value: < 2.2e-16
```

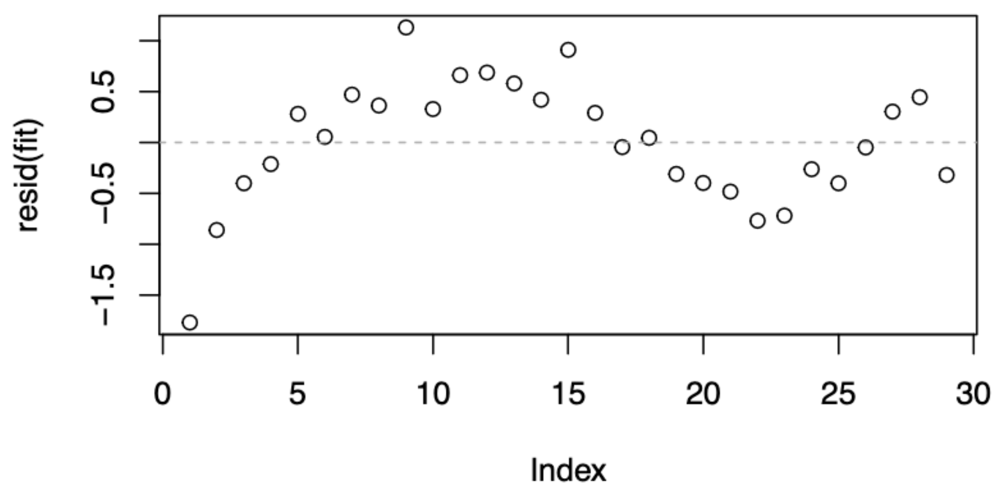
```r
# 4. Plot log(cpg) vs. calendar year, then add the fitted line
plot(
  log(cpg),
  xlab = "Year",
  ylab = "log(Price per GB)",
  main = "Fitting log(c_t) versus time",
  col  = "steelblue",
  lwd  = 2
)
abline(fit_b, col = "firebrick", lwd = 2)
```



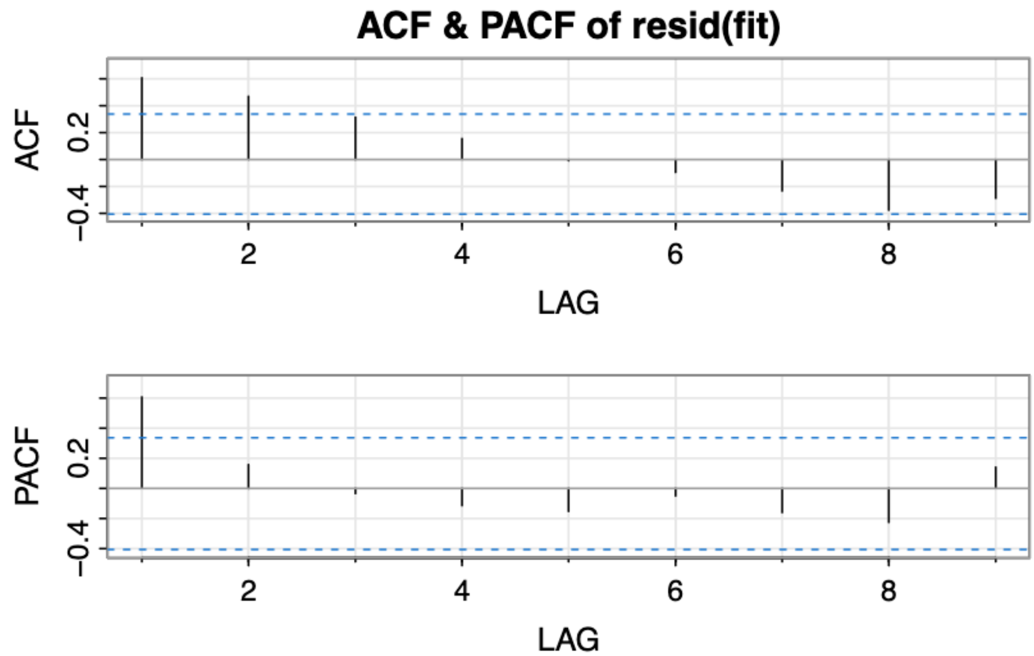
Although the fitted line on the log-scale explains over 98 % of the variation, there are small systematic departures from perfect linearity. In the early 1980s, the first year or two of data lie slightly above the line, indicating that prices fell more steeply at the very beginning than the long-run trend would suggest. During the mid- to late 1980s and again around the early 1990s, the log-price curve exhibits minor "wiggles" where it briefly levels off or dips more sharply—reflecting periods when incremental technological improvements temporarily altered the pace of cost decline. Finally, in the last couple of years (around 2007–2008), the observed log-price drops a bit below the straight-line trend, showing an especially rapid fall in cost at the end of the sample. These deviations are all quite small relative to the overall downward trajectory, so the exponential model remains an excellent summary of how hard-drive price per gigabyte fell from 1980 to 2008.

```
# ----------------------------------------------------------
#  Part (c)
# ----------------------------------------------------------

# 1. Install and load the astsa package (for acf2)
if (!require(astsa)) install.packages("astsa")
library(astsa)



# 2. Plot resid(fit) against its integer index, using open circles
```

```r
plot(
  resid(fit_b ),
  type = "p",             # points only
  pch  = 1,               # open circle symbol
  xlab = "Index",         # index from 1 to length(cpg)
  ylab = "resid(fit)",    # matches the screenshot label
  main = ""               # no title, to match the screenshot exactly
)
abline(h = 0, lty = 2, col = "darkgray")  # horizontal dashed zero-line
```




```r
# 3. Plot the ACF and PACF of the residuals using acf2()
acf2(
  resid(fit_b ),
  main = "ACF & PACF of resid(fit)"
)
```




```
     [,1] [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9]
ACF  0.61 0.47  0.32  0.15 -0.01 -0.10 -0.23 -0.38 -0.29
PACF 0.61 0.16 -0.03 -0.11 -0.15 -0.05 -0.16 -0.23  0.14
```

When we examine the residuals

$$\varepsilon^t = log\ (ct) - (\gamma^0 + \beta\, t)\varepsilon^t\ = log(ct\ ) - (\gamma\ 0\ + \beta\ t)$$

versus their index, they do not form a random scatter around zero. Instead, the first few points are negative, then a cluster of positive residuals appears (around indices 3–10), followed by several negative residuals (indices 12–22), and finally they rise back toward zero by the late 2000s. This "wave-like" drift indicates the straight-line fit on

$$log\ (ct)log(ct\ )$$

omits some slowly-varying structure.

The ACF of these residuals has a large positive lag-1 correlation (about 0.60), significant positive correlations at lag 2, and a gradual decay thereafter; the PACF shows one dominant spike at lag 1 and near-zero values beyond. Together, these patterns strongly violate the OLS assumption of uncorrelated errors and suggest that the residuals follow an AR(1) process. In short, although the simple linear regression explains over 98 % of the variance in

$$log\ (ct)log(ct\ )$$

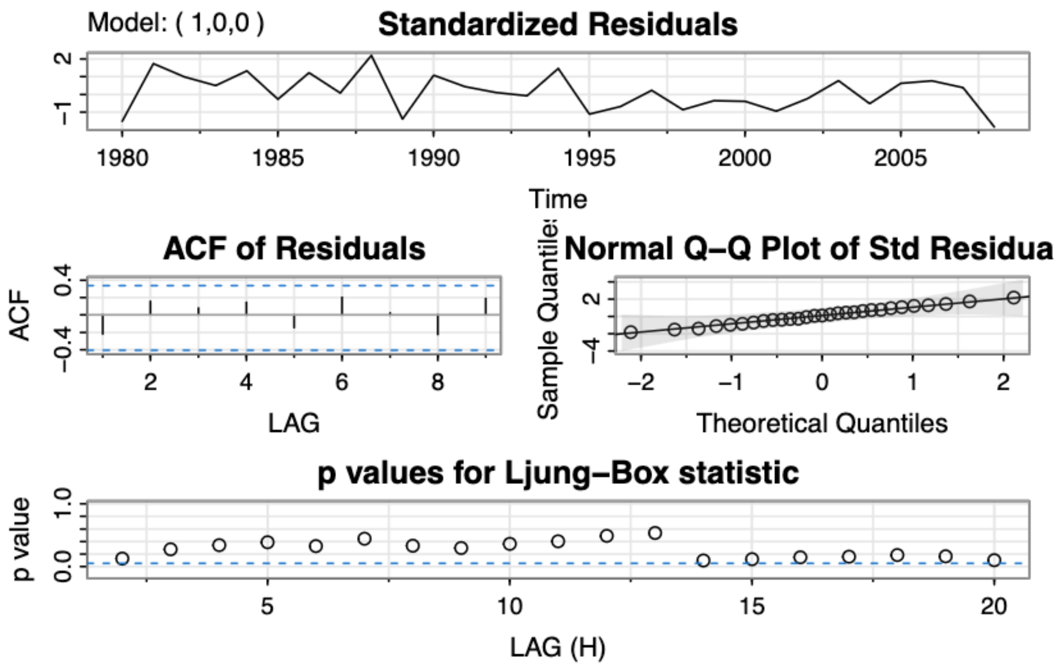
, the errors are positively autocorrelated (especially at lag 1), so the OLS model is not fully adequate without modeling that AR(1) dependence.

The PACF plot shows a very large positive spike at lag 1 (about 0.61), indicating that deviation from the fitted log–linear trend is still strongly tied directly to yesterday's deviation even after accounting for any intermediate influences. At lag 2 the partial correlation drops sharply to around 0.16—a much smaller effect—and thereafter the PACF coefficients fluctuate close to zero (and even slightly negative between lags 4 and 8), with none of those later spikes clearly standing out. This abrupt "cut-off" after the first lag is exactly what one expects if the residuals follow an AR(1) process, confirming that no higher-order autoregressive terms are needed in the error structure.

```r
# ----------------------------------------------------------
# Part (d): Fit the regression again, allowing for AR(1) errors
# ----------------------------------------------------------

# 1. Make sure astsa is loaded (so sarima() is available)
if (!require(astsa)) install.packages("astsa")
library(astsa)

# 2. Load the cpg data
data("cpg")

# 3. Fit an ARIMA(1,0,0) model to log(cpg), with a linear time trend as an exogenous regress
sarima(
  log(cpg),      # response: log price per GB
  1, 0, 0,       # ARIMA(p=1, d=0, q=0) → AR(1) errors
  xreg = time(cpg)   # regressor: calendar year (1980, 1981, ..., 2008)
)
```

```
initial  value -0.669056
iter   2 value -0.999488
iter   3 value -1.088763
iter   4 value -1.102248
iter   5 value -1.128914
iter   6 value -1.131945
iter   7 value -1.132479
iter   8 value -1.132525
iter   9 value -1.132540
iter  10 value -1.132543
iter  11 value -1.132545
iter  12 value -1.132545
iter  12 value -1.132545
```

```
iter  12 value -1.132545
final  value -1.132545
converged
initial  value -0.701381
iter   2 value -0.882862
iter   3 value -0.886699
iter   4 value -0.888651
iter   5 value -0.888966
iter   6 value -0.889035
iter   7 value -0.889043
iter   8 value -0.889045
iter   9 value -0.889045
iter  10 value -0.889045
iter  10 value -0.889045
iter  10 value -0.889045
final  value -0.889045
converged
<><><><><><><><><><><><><><>

Coefficients:
           Estimate      SE  t.value p.value
ar1          0.8297  0.1190   6.9741       0
intercept 1113.0105 73.5665  15.1293       0
xreg        -0.5554  0.0368 -15.0716       0

sigma^2 estimated as 0.1623007 on 26 degrees of freedom

AIC = 1.335649  AICc = 1.368752  BIC = 1.524241
```



After allowing for AR(1)-correlated errors, the residuals show no significant structure. The time-series plot of standardized residuals is a random scatter around zero, the ACF is entirely within confidence bounds, the Q–Q plot is close to linear, and all Ljung–Box tests return large p-values. Together, these diagnostics indicate that the fitted model

$$\begin{aligned} \log(c_t) &= 1113.0105 \;-\; 0.5554\, t \;+\; \varepsilon_t, \\ \varepsilon_t &= 0.8297\, \varepsilon_{t-1} \;+\; e_t, \quad e_t \sim \mathcal{N}(0, \sigma^2). \end{aligned}$$

Here, the slope –0.5554 remains highly significant, indicating that each additional year is associated with roughly a 0.56-unit drop in log price (about a 43 % annual decline in the raw price). The AR(1) coefficient of 0.8297 tells us that residual shocks persist strongly—about 83 % of last year's unexpected deviation carries over into this year—so modeling that autocorrelation directly is essential. Introducing the AR(1) error greatly cleans up the residuals (no remaining serial dependence) and ensures that our standard errors and hypothesis tests for the trend coefficient are valid.

Part(e)

The key distinction is that in Part (b) we applied a plain linear regression of

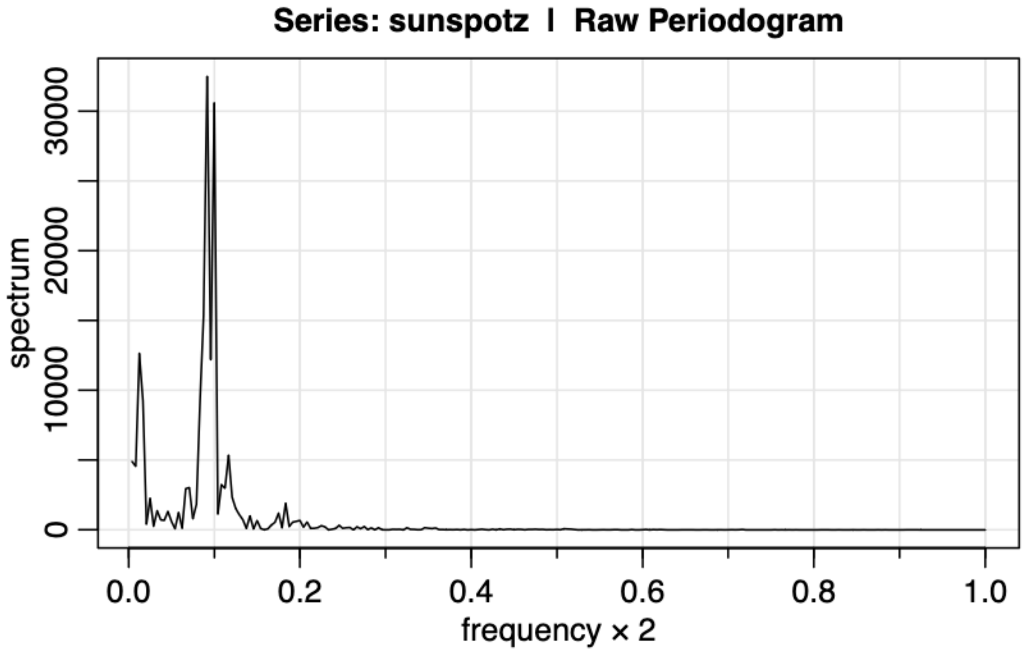
$$\log(c_t)$$

on time, implicitly assuming that the errors are uncorrelated; in reality, those residuals exhibit strong lag-1 autocorrelation (ACF 0.61), so the OLS estimates of variance and standard errors are biased and inference invalid. In Part (d) we instead fit the same time–trend but allow the errors to follow an AR(1) process (

$$\varepsilon_t = 0.8297\,\varepsilon_{t-1} + e_t$$

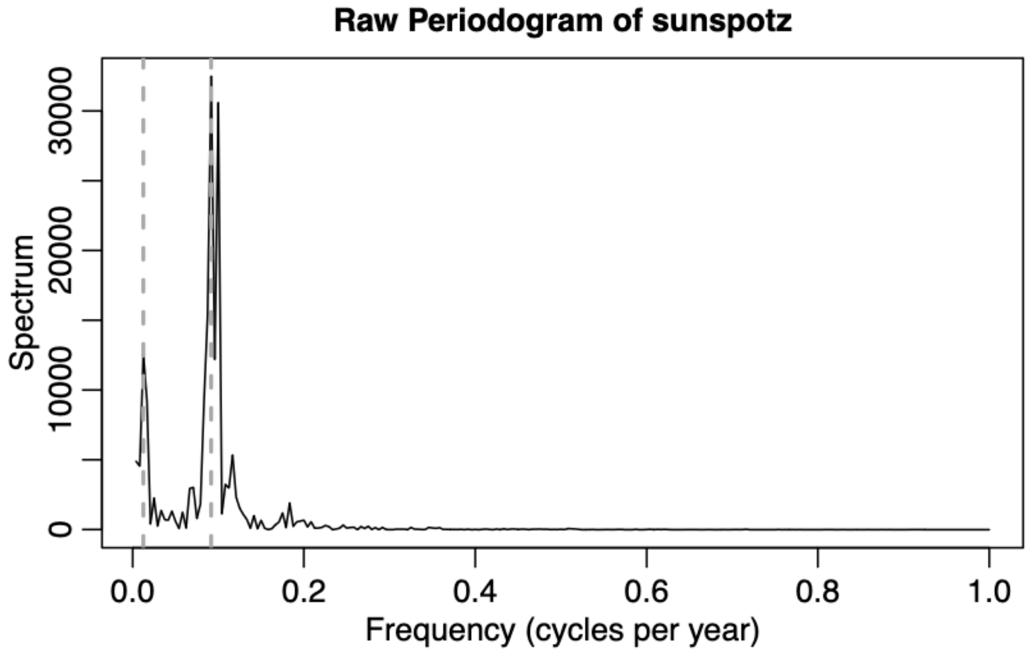
), which removes the serial dependence, restores white-noise residuals, and yields trustworthy uncertainty measures.

```r
# -------------------------------------------------------------
#  Question 3, part (a): Periodogram analysis of sunspotz
# -------------------------------------------------------------

# 1. Install & load astsa (for the sunspotz data and mvspec())
if (!require(astsa)) install.packages("astsa")
library(astsa)

# 2. Load the biyearly-smoothed sunspot series
data("sunspotz")   # 459 points, frequency = 2 (June 1749 - Dec 1978)

# 3. Compute the raw periodogram (no taper, no log)
sun.per <- mvspec(
  sunspotz,
  taper = 0,       # no taper
  log   = "no"     # plot spectrum on original scale
)
```




```
# 4. Plot the periodogram and mark the 80-year & 11-year peaks
plot(
  sun.per$freq,
  sun.per$spec,
  type = "l",
  xlab = "Frequency (cycles per year)",
  ylab = "Spectrum",
  main = "Raw Periodogram of sunspotz"
)

#  - 80-year cycle: frequency = 1/80 = 0.0125 = 3/240
abline(
  v   = 3/240,
  lty = "dashed",
  col = "darkgray",
  lwd = 2
)

#  - 11-year cycle: frequency  1/11 = 0.091667 = 22/240
abline(
  v   = 22/240,
  lty = "dashed",
  col = "darkgray",
```

```r
  lwd = 2
)
```




```r
# 5. Identify the exact indices in sun.per$freq corresponding to those two frequencies
freq.vec <- sun.per$freq

i80 <- which.min(abs(freq.vec - (3/240)))     # closest to 0.0125 cpy
i11 <- which.min(abs(freq.vec - (22/240)))    # closest to 0.091667 cpy

# 6. Print out frequencies and raw periodogram heights
cat("Index of 80-year peak: ", i80,
    "\tFrequency =", round(freq.vec[i80], 6),
    "\tI(0.0125) =", round(sun.per$spec[i80], 4), "\n")
```

```
Index of 80-year peak:  3   Frequency = 0.0125  I(0.0125) = 12645.8
```

```r
cat("Index of 11-year peak:", i11,
    "\tFrequency =", round(freq.vec[i11], 6),
    "\tI(0.091667) =", round(sun.per$spec[i11], 4), "\n\n")
```

```
Index of 11-year peak: 22   Frequency = 0.091667    I(0.091667) = 32477.87
```

part(a):The predominant periods in the biyearly-smoothed sunspot series are: Approximately 80 years (frequency 0.0125 cpy), and approximately 11 years (frequency 0.091667 cpy). No other frequencies come close to these two in spectral magnitude, and both appear well above the surrounding "noise floor," indicating that these two cycles dominate the data.

Between these two major spikes, the periodogram values dip back down to the low hundreds or less, and beyond 0.2 cycles/year the curve remains very close to zero. That flat baseline of near-zero power ("noise floor") indicates that almost no other frequencies have substantial spectral power. Thus the plot cleanly highlights two isolated, dominant peaks—one near 0.0125 cpy (80 years), and one near 0.091667 cpy (11 years). No other frequencies approach these two peaks in magnitude.

Hence, the periodogram analysis clearly identifies an 80-year Gleissberg cycle and the familiar 11-year solar cycle as the two principal periodicities in the sunspotz data.

```
# -----------------------------------------------------------
#  Question 3, part (b)
# -----------------------------------------------------------
U = qchisq(.025, 2)
L = qchisq(.975, 2)
2*sun.per$spec[3]/L
```

```
[1] 3428.087
```

```
2*sun.per$spec[3]/U
```

```
[1] 499482.4
```

```
2*sun.per$spec[22]/L
```

```
[1] 8804.265
```

```
2*sun.per$spec[22]/U
```

```
[1] 1282807
```

part (b):We are 95% confident that the two predominant periods lines from 3428.087 to 499482.4(80 years period) and 8804.265 to 1282807(11 years period).

Because these two confidence bounds are extremely wide and overlap almost entirely, they effectively collapse into a single, very broad interval (roughly 3 000 to 1 300 000 time units), meaning we cannot pinpoint the two cycles separately with precision.

In practical terms, this tells us that we can be very confident the true spectral density at those two frequencies is not just a random blip. The sheer magnitude of these intervals shows that the apparent 80-year and 11-year peaks are not artifacts of sampling variability, but instead correspond to genuine, strong periodic components in the sunspot data.

Although the intervals themselves are wide (reflecting the high variability of a single periodogram ordinate), their entire ranges remain well above zero. That is exactly what confirms statistical significance: there is no chance that the true spectral power at those two frequencies is near zero.

In summary, the approximate 95 % confidence intervals for the spectral densities at 0.0125 cpy (80 years) and 0.091667 cpy (11 years) are extremely large and lie far above the noise level. Consequently, the 80-year Gleissberg cycle and the 11-year solar cycle are both unequivocally significant features of the biyearly-smoothed sunspotz series.