Diego Saráchaga
NetID: ds1591

HW5

Problem 1

$$r_t = 1 + 0.6 r_{t-1} - 0.4 r_{t-2} + a_t$$

$\{a_t\}$ normal white noise with mean 0 and variance 0.02

This is an AR(2) model $\Rightarrow$ the stationary condition is:

$$|\phi_1| = 0.6 < 1$$
$$|\phi_2| = 0.4 < 1$$
$$|\phi_1 + \phi_2| = 0.2 < 1$$
$$|\phi_1 - \phi_2| = -1 < 1$$

a) under stationary $\Rightarrow E(r_t) = \dfrac{\phi_0}{1-\phi_1-\phi_2} = \dfrac{1}{1-0.6+0.4} = 1.25$

b) for AR(2), the Yule-Walker equations are:

$$\gamma_k = \begin{cases} \phi_1\gamma_1 + \phi_2\gamma_2 + \sigma^2 & \forall k=0 \\ \phi_1\gamma_{k-1} + \phi_2\gamma_{k-2} & \forall k>0 \end{cases}$$

1) $\gamma_0 = \phi_1\gamma_1 + \phi_2\gamma_2 + \sigma^2$

2) $\gamma_1 = \phi_1\gamma_0 + \phi_2\gamma_1 \quad \rightarrow \gamma_1 = \dfrac{0.6\gamma_0}{1+0.4} = 3/7\,\gamma_0$ ④ $\Big|$ ④, 3) $\Rightarrow \gamma_2 = 0.6 \cdot 3/7\,\gamma_0 - 0.4\gamma_0$

3) $\gamma_2 = \phi_1\gamma_1 + \phi_2\gamma_0$

$\Rightarrow \gamma_2 = \gamma_0\left(\dfrac{1.8-2.8}{7}\right) = -1/7\,\gamma_0$ ⑤ $\Big|$ ⑤, ①, ④ $\Rightarrow \gamma_0 = 0.6 \cdot 3/7\,\gamma_0 + 0.4 \cdot 1/7\,\gamma_0 + 0.02$

$$\gamma_0 = \gamma_0\left(\frac{1.8+0.4}{7}\right) + 0{,}02 = \gamma_0\left(\frac{2.2}{7}\right) + 0{,}02$$

$$\gamma_0\left(1 - \frac{2.2}{7}\right) = 0.02$$

$$\gamma_0\left(\frac{4.8}{7}\right) = 0.02 \Rightarrow \gamma_0 = 0{,}029 \Rightarrow$$

$$\Rightarrow \begin{cases} \gamma_1 = 3/7 \cdot 0.029 = 0.0125 \\ \gamma_2 = -1/7 \cdot 0.029 = -0.004 \end{cases}$$

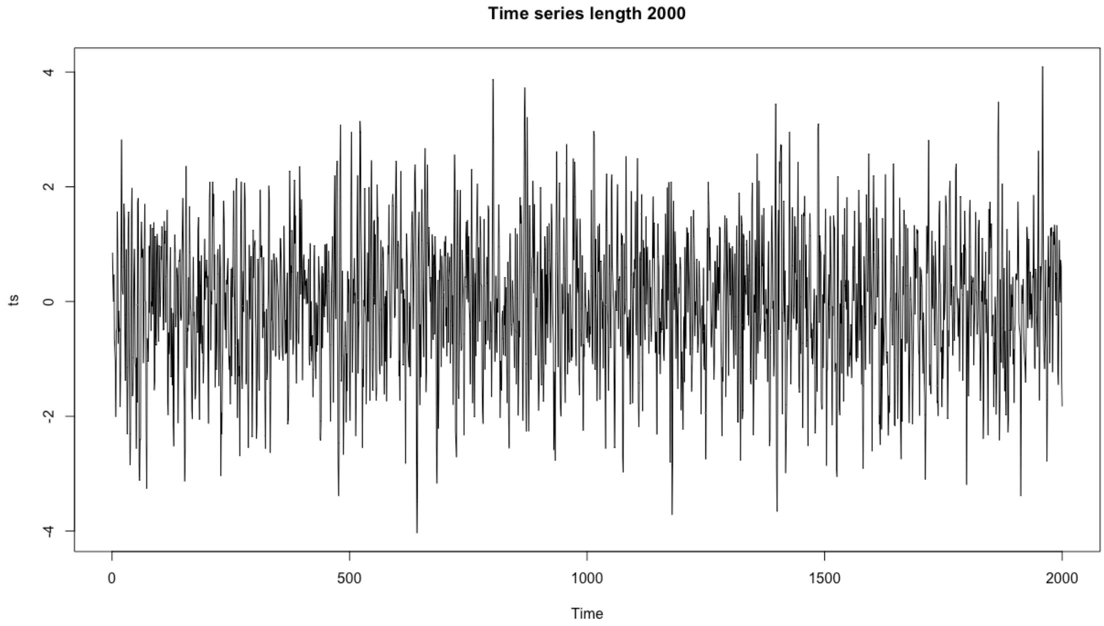
ACF: $\rho_k = \phi \rho_{k-1} + \phi_2 \rho_{k-2} \quad \forall\, k>0$

$$\rho_1 = \gamma_1 / \gamma_0 = 3/7$$

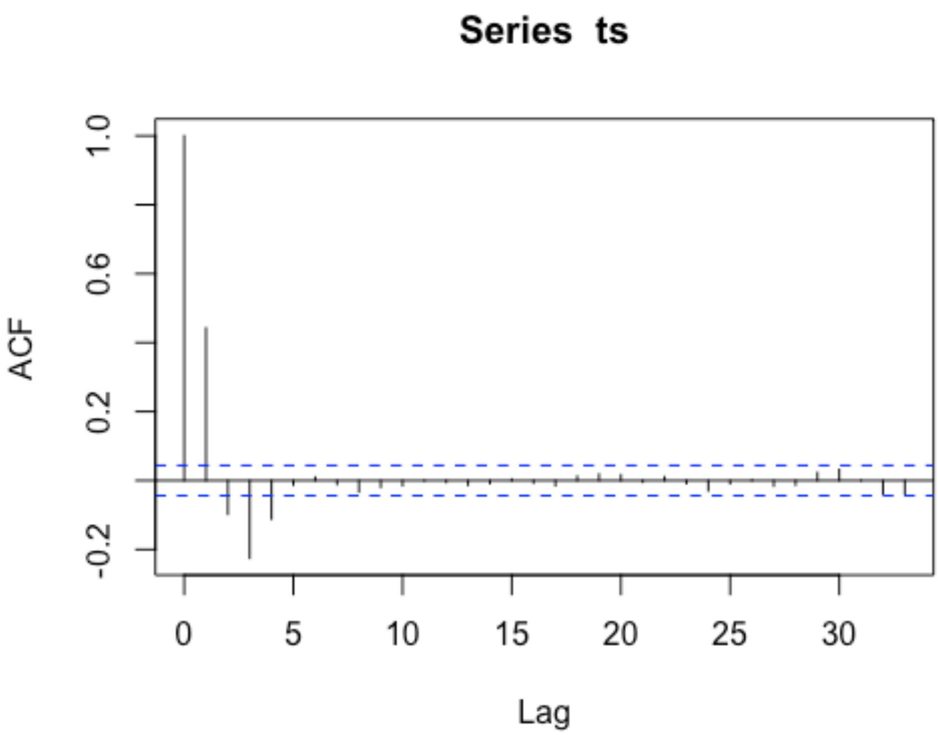
$$\rho_2 = \gamma_2 / \gamma_0 = -1/7$$

$$\rho_3 = \phi_1 \rho_2 + \phi_2 \rho_1 = (0{,}6)(-1/7) - 0.4\,(3/7) = -0.257$$

Problem 1
c)
Time series plot

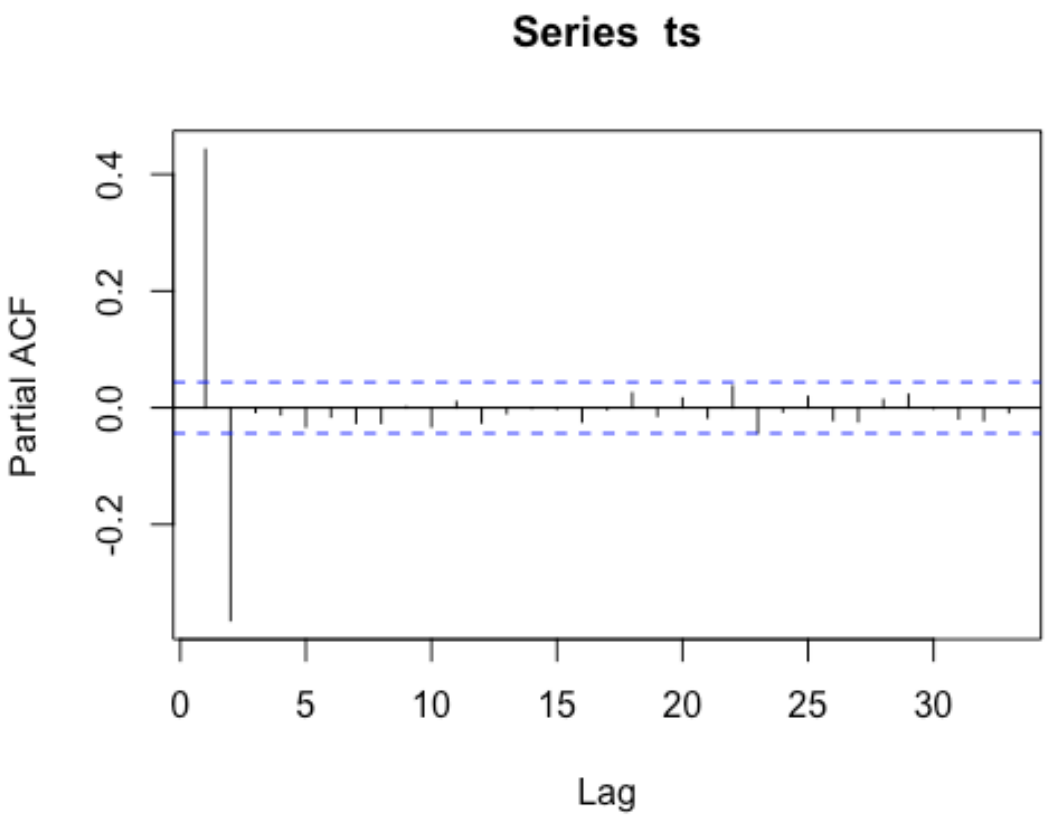


Autocorrelation plot



Lag values:
Lag-1: 0.442
Lag-2: -0.098
Lag-3: -0.225

d)
Partial autocorrelation plot



Lag values:
Lag-1: 0.442
Lag-2: -0.364
Lag-3: -0.008

Problem 2

$$\underbrace{(1-.3B)(1-.5B)}(1-.6B+.4B^2)(r_t-1.3)=a_t$$

$$(1-.5B-.3B+.15B^2)(1-.6B+.4B^2)=$$

$$=1-.6B+.4B^2-.8B+.48B^2-.32B^3+.15B^2-.09B^3+.06B^4=$$

$$=1-1.4B-1.03B^2-.41B^3+.06B^4 \rightarrow ①$$

Substituting the original equation with ① $\rightarrow$

$$\rightarrow (1-1.4B-1.03B^2-.41B^3+.06B^4)(r_t-1.3)=a_t$$

$$r_t-1.4B r_t-1.03B^2 r_t-.41B^3 r_t+.06B^4 r_t-1.3+1.82-1.339+0.533-0.078=a_t$$

$$\Rightarrow r_t = 1.4\, r_{t-1} - 1.03\, r_{t-2} + .41\, r_{t-3} - .06\, r_{t-4} + .364 = a_t$$

This is an AR(4) equation. It's characteristic equation is:

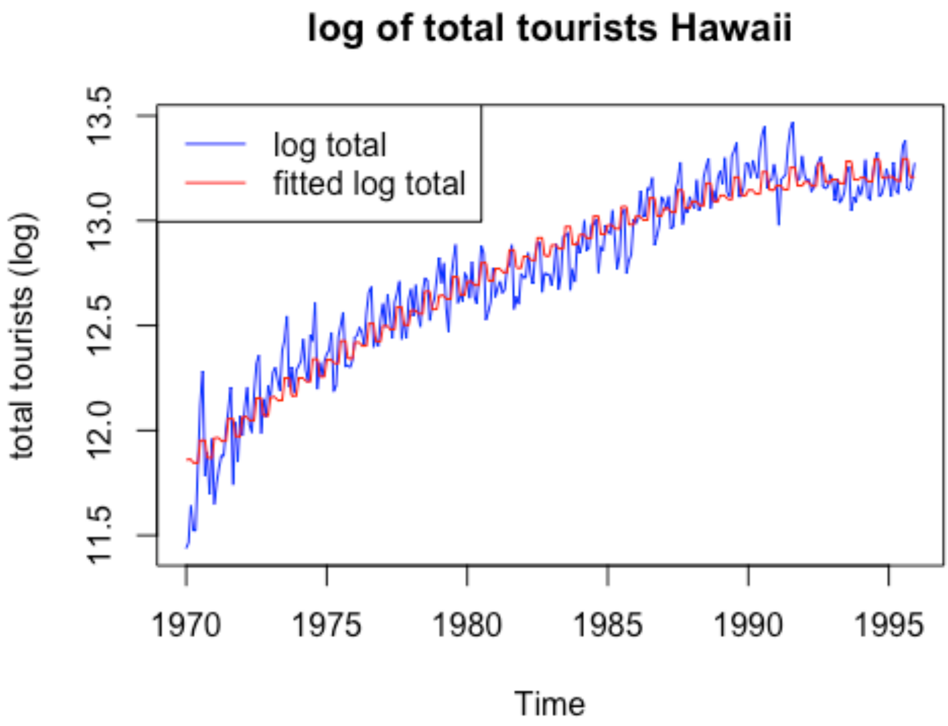
$$1-1.4x+1.03x^2-0.41x^3+0.06x^4=0$$

with solutions $= \left| \begin{array}{l} 2 \\ 10/3 \\ \dfrac{3\pm\sqrt{31}\,i}{4} \end{array} \right.$

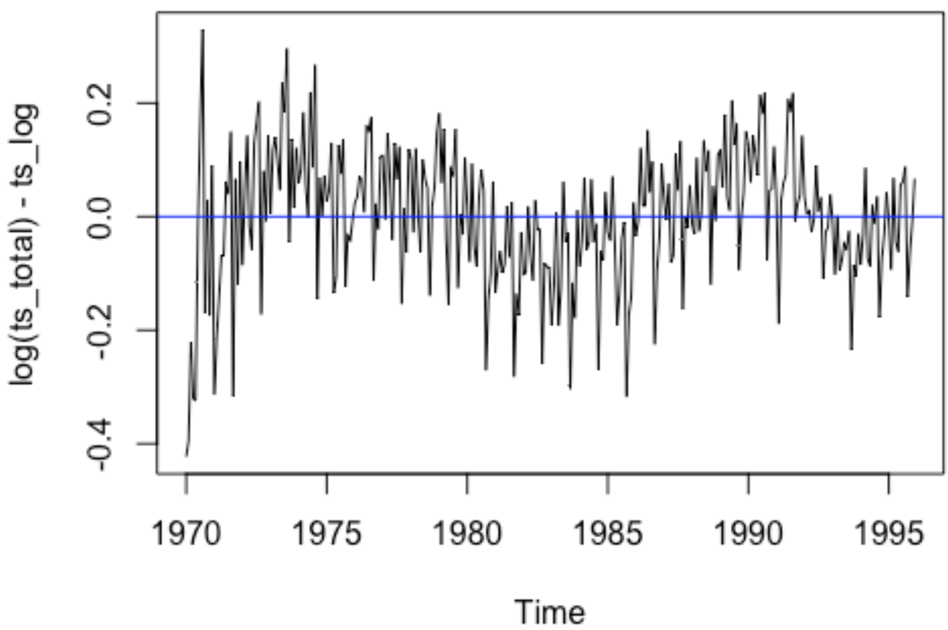
Based on the definition of AR(P), the model is a stationary linear AR model of order 4.

Problem 3

a)

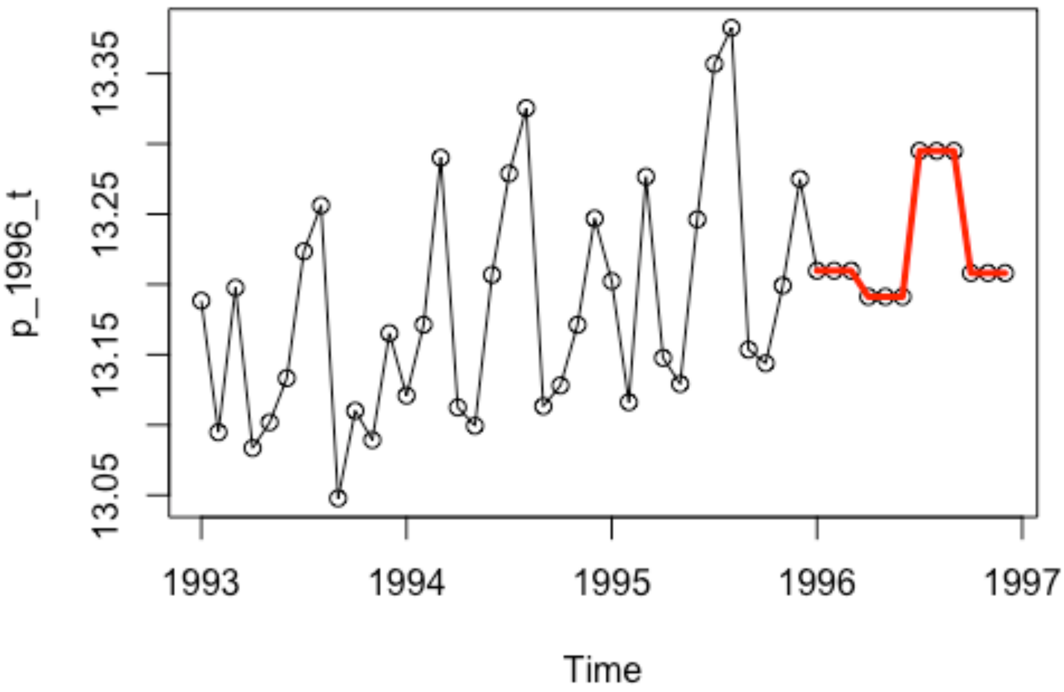


log of total tourists Hawaii
log total
fitted log total
total tourists (log)
13.5
13.0
12.5
12.0
11.5
1970
1975
1980
1985
1990
1995
Time

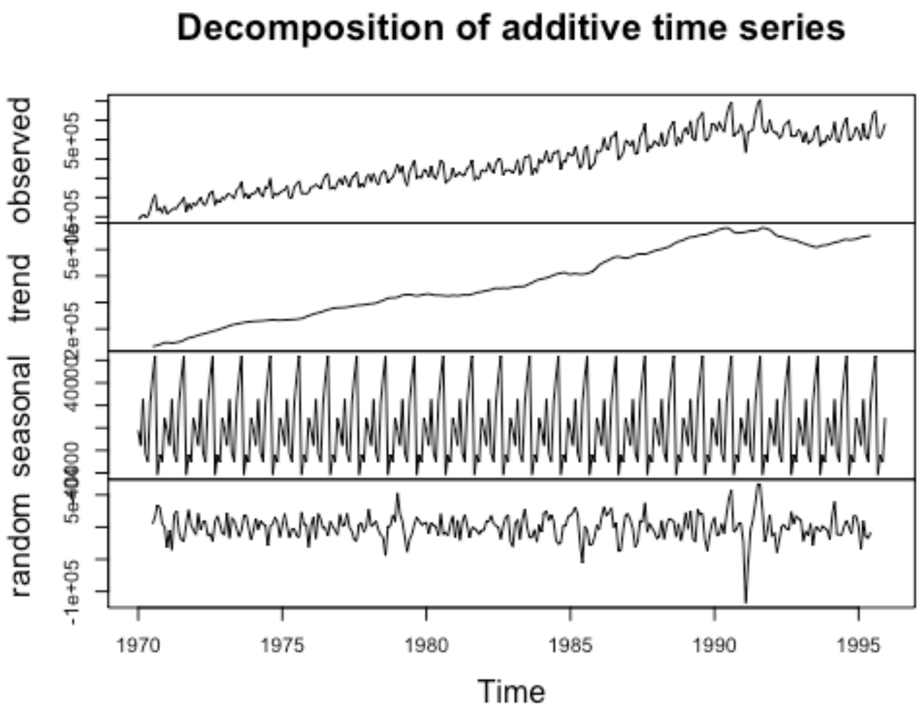


log(ts_total) - ts_log
0.2
0.0
-0.2
-0.4
1970
1975
1980
1985
1990
1995
Time

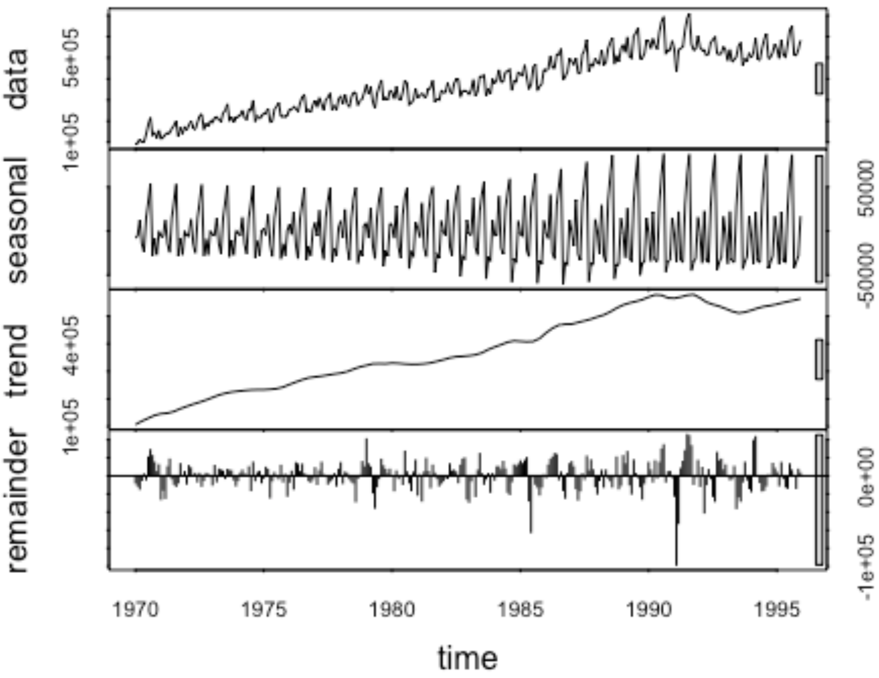
b)



c)
decompose()



stl()


data
seasonal
trend
remainder
1e+05
5e+05
50000
-50000
4e+05
1e+05
0e+00
-1e+05
1970
1975
1980
1985
1990
1995
time

Problem 4

a)

log likelihood = -446.1314

b)

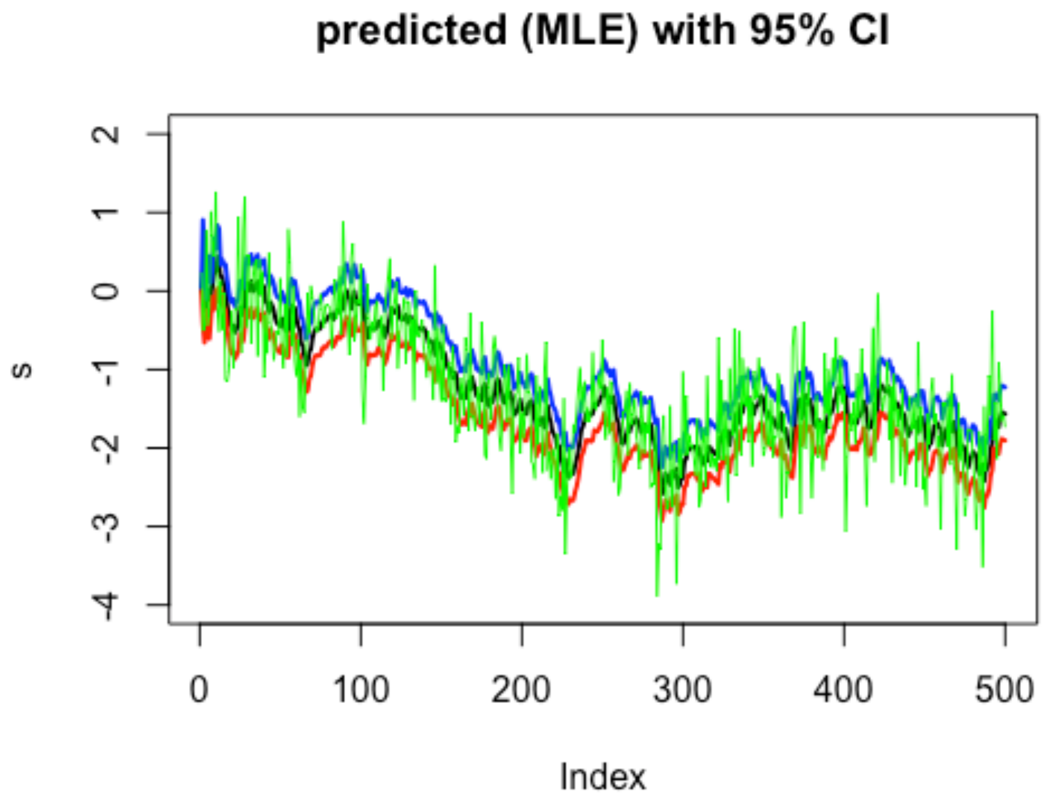
log likelihood = -446.9549

c)

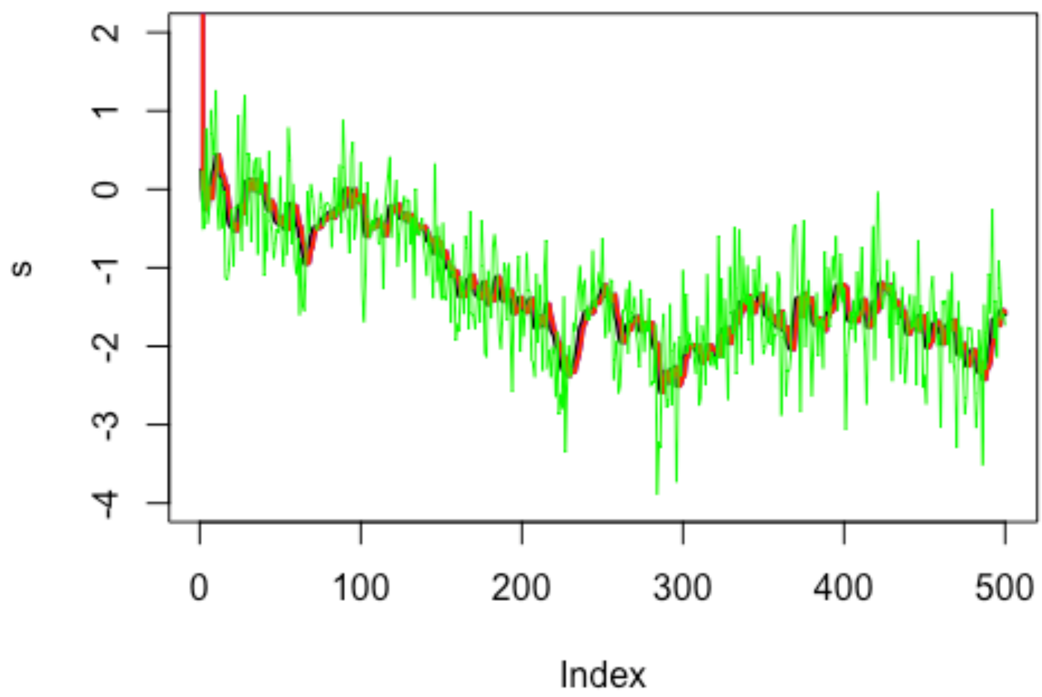
MLE σ_e^2 and σ_η^2.

0.287369728 | 0.009473803

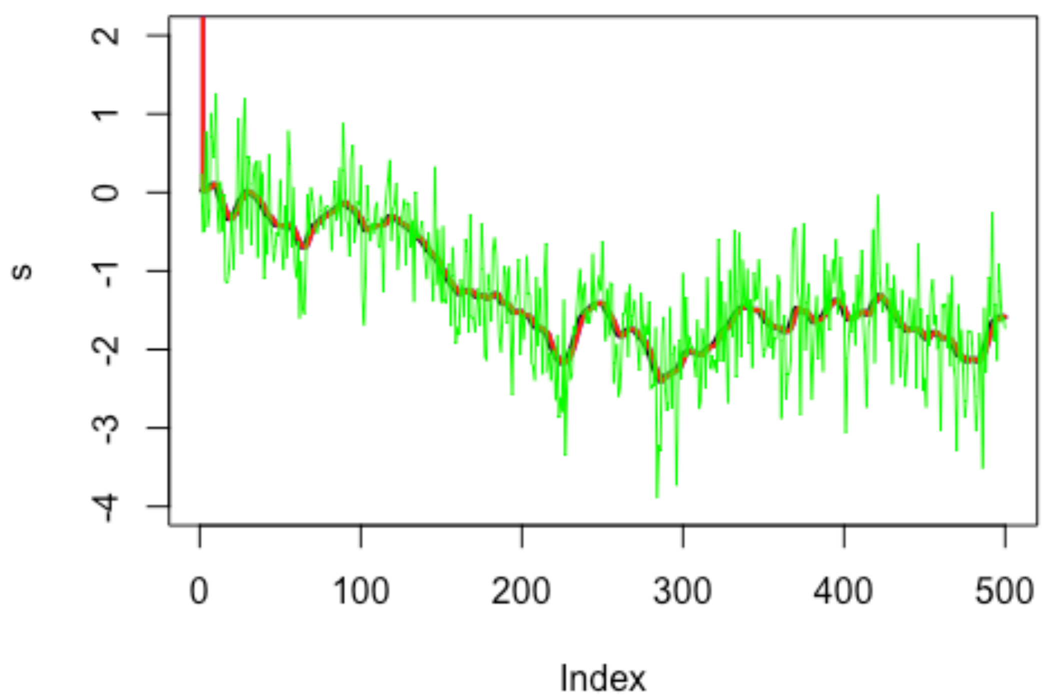
d)



filtered (MLE) with 95% CI



smoothed (MLE) with 95% CI



e)

```
Call:
StructTS(x = yt, type = "level", fixed = c(NA, NA))

Variances:
   level   epsilon
0.009457  0.287413
```

Comparing the variance estimates with the ones from part c), they are practically the same, the difference is minimal.

```
#Problem 1
#c)
ts <- arima.sim(model = list (ar = c(0.6,-0.4)), n = 2000)
plot.ts(ts, main = "Time series length 2000")

ac <- acf(ts, type = "correlation", plot = T)
ac

#d)
pac <- acf(ts, type = "partial", plot = T)
pac


#Problem 3
#a)
#library("forecast")

hawaii<-read.table('hawaii-new.dat',col.names=c('year_month', 'total',
'west', 'east'))
ts_total <- ts(hawaii$total, start = 1970, frequency = 12)
ts_total
n <- 312

ym.ascycle = as.factor(rep(c(1,1,1,2,2,2,3,3,3,4,4,4),n/12))
ym.ascycle

total_lmod <- lm (log(total) ~ year_month + I(year_month^2) + ym.ascycle,
data = hawaii)
total_lmod

ts_log <- ts (total_lmod$fitted.values, start = 1970, freq = 12)

plot.ts(log(ts_total), col = 'blue' , ylab ='total tourists (log)',main ='log
of total tourists Hawaii')
lines(ts_log,col = 'red')
legend('topleft', col = c('blue', 'red'), lty = 1, legend =c('log
total','fitted log total'))

plot.ts(log(ts_total)-ts_log)
abline(h = 0, col ='blue')

#b)
log_total = log(hawaii$total)

p_1996_m <- data.frame(year_month = 9601:9612, ym.ascycle =
as.factor(c(1,1,1,2,2,2,3,3,3,4,4,4)))
p_1996_m

p_1996 <- predict(total_lmod, p_1996_m, interval = "confidence")
p_1996

p_1996.df <- as.data.frame(p_1996)
p_1996.df

p_1996_t <- ts(c(log_total[277:312], p_1996.df$fit), start = 1993, frequency
= 12)
p_1996_t
```

```
plot(p_1996_t, type="o")
lines(seq(1996,1997,by=1/12)[-13], p_1996.df$fit, col='red', lwd=3)

#c)
dec_ts_total<- decompose(ts_total)
plot(dec_ts_total)

stl_ts_total <- stl(ts_total, s.window=12)
plot(stl_ts_total)


#Problem4
#a)
yt <-scan('lt.txt')

st <- array(500)
st[1] <- 0.2
st

sigma <- array(500)
sigma[1] <-  2.26
sigma

Vt <- array(500)
vt <- array(500)
kt <- array(500)

Mu0 = 0.2; S02 = 2.25

#Kalman filter
for (i in 1:500)
{
  vt[i] = yt[i] - st[i]
  Vt[i] = sigma[i] + 0.25
  kt[i] = sigma[i]/Vt[i]
  st[i+1] = st[i] + kt[i]*vt[i]
  sigma[i+1] = (1-kt[i])*sigma[i] + 0.01
}

e_log_likelihood <- (-1/2) * length(vt) * log(2*pi) - sum(log(Vt) +
(vt^2)/Vt)/2
e_log_likelihood


#b)
yt_miss = yt
yt_miss[5]=0
yt_miss[100]=0
yt_miss[165]=0

st <- array(500)
st[1] <- 0.2

sigma <- array(500)
sigma[1] <-  2.26

Vt <- array(500)
vt <- array(500)
kt <- array(500)
```

```
#Kalman filter
for (i in 1:500)
{
  vt[i] = yt_miss[i] - st[i]
  Vt[i] = sigma[i] + 0.25
  kt[i] = sigma[i]/Vt[i]
  st[i+1] = st[i] + kt[i]*vt[i]
  sigma[i+1] = (1-kt[i])*sigma[i] + 0.01
}

e_log_likelihood_miss <- (-1/2) * length(vt-3) * log(2*pi) - sum(log(Vt) +
(vt^2)/Vt)/2
e_log_likelihood_miss


#c)
library(dlm)
#MLE
lt=function(x)
{
  ltm=dlm(FF = 1,
          V = x[1],
          GG = 1,
          W = x[2],
          m0 = 0,
          C0 = 10^7)
  return(ltm)
}
m.lt=dlmMLE(y = yt,
            parm = c(0.2,2.25),
            build = lt,
            lower=c(0,0),
            upper=c(100,100),
            hessian = TRUE,
            control = list(maxit = 500))
m.lt
m.lt$par

#d)
s.filter=dlmFilter(yt,lt(m.lt$par))
s.filter

# prediction
s.predicted=rep(0,500)
#s.predicted

up <- array(500)
low <- array(500)

for (i in 1:500)
{
  s.predicted[i] =
s.filter$U.R[[i]]%%diag(s.filter$D.R[i,]^2,nrow=1)%%t(s.filter$U.R[[i]])
  up[i]=s.filter$a[i]+2*sqrt(s.predicted[i])
  low[i]=s.filter$a[i]-2*sqrt(s.predicted[i])
}

plot(s.filter$a,type="l",ylim=c(-4,2),xlab="Index",ylab="s",main="predicted
(MLE) with 95% CI",lwd=2)
lines(up,col='blue',lwd=2)
```

```
lines(low,col='red',lwd=2)
lines(yt, col = 'green')

# filtering
s.filtered=rep(0, 500)
s.filtered

filt_up <- array(500)
filt_low <- array(500)

for (i in 2:(501))
{
  s.filtered[i-1] =
s.filter$U.C[[i]]%%diag(s.filter$D.C[i,]^2,nrow=1)%%t(s.filter$U.C[[i]])
  filt_up[i]=s.filter$m[i-1]+2*sqrt(s.filtered[i])
  filt_low[i]=s.filter$m[i-1]-2*sqrt(s.filtered[i])
}

plot(s.filter$m[-1],type="l",ylim=c(-4,2),xlab="Index",ylab="s",main="filtered
 (MLE) with 95% CI",lwd=2)
lines(filt_up,col='blue',lwd=2)
lines(filt_low,col='red',lwd=2)
lines(yt, col = 'green')

# smoothing
s.smooth=dlmSmooth(yt,lt(m.lt$par))

s.smoothed=rep(0,500)

smoothed_up <- array(500)
smoothed_low <- array(500)

for (i in 2:(501))
{
  s.smoothed[i-1] =
s.smooth$U.S[[i]]%%diag(s.smooth$D.S[i,]^2,nrow=1)%%t(s.smooth$U.S[[i]])
  smoothed_up[i]=s.smooth$s[i-1]+2*sqrt(s.smoothed[i])
  smoothed_low[i]=s.smooth$s[i-1]-2*sqrt(s.smoothed[i])
}

plot(s.smooth$s[-1],type="l",ylim=c(-4,2),xlab="Index",ylab="s",main="smoothed
 (MLE)  with 95% CI",lwd=2)
lines(smoothed_up,col='blue',lwd=2)
lines(smoothed_low,col='red',lwd=2)
lines(yt, col = 'green')

#e)

StructTS(yt,type="level",fixed=c(NA,NA))
```