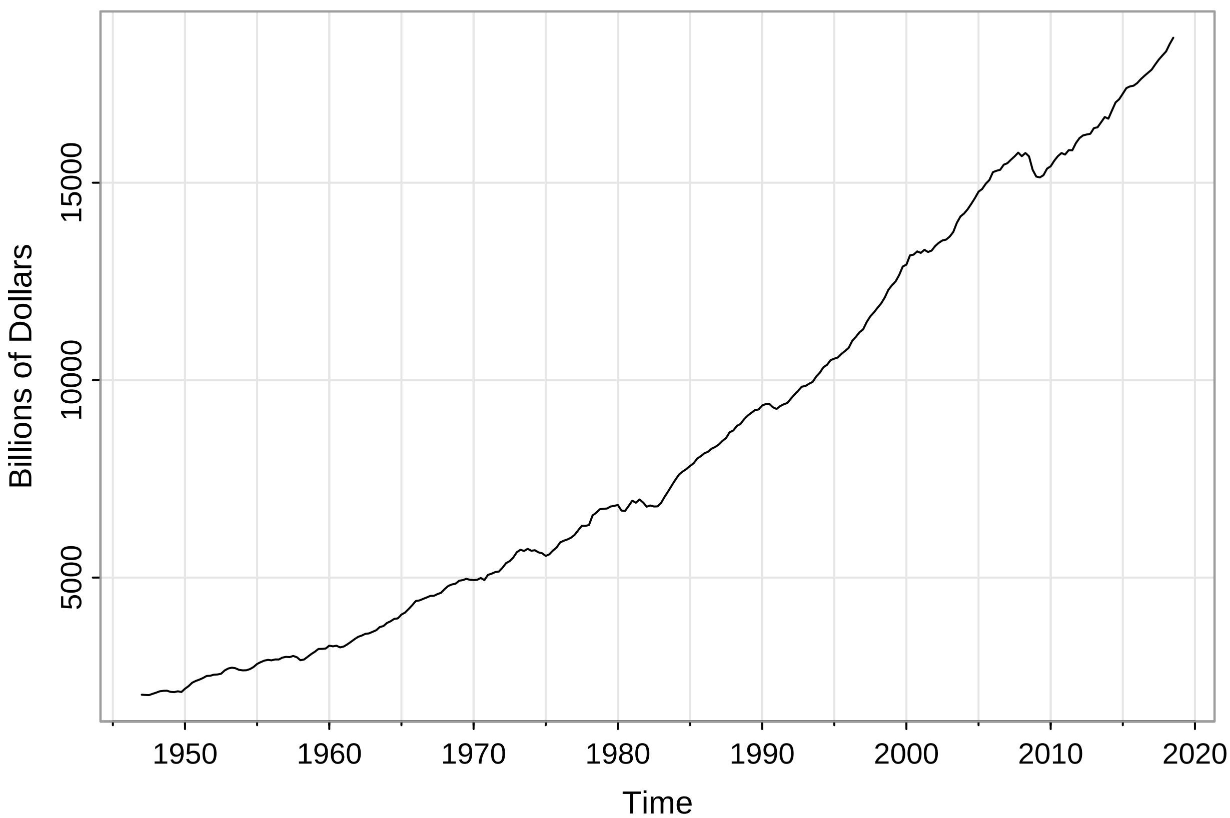
STAT4870 HW5

Anton Yang

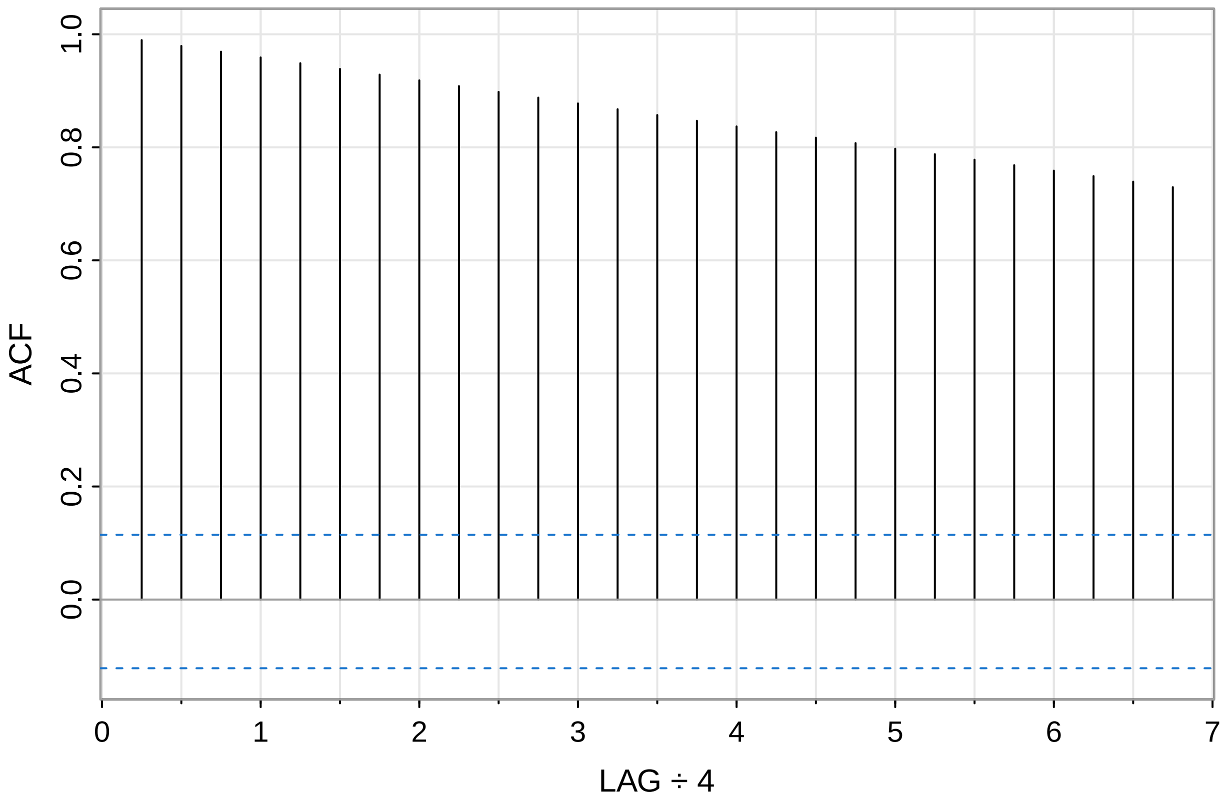
2024-10-28

Question 5.2

```
library(astsa)
data<-gdp
tsplot(data, ylab = "Billions of Dollars")
```



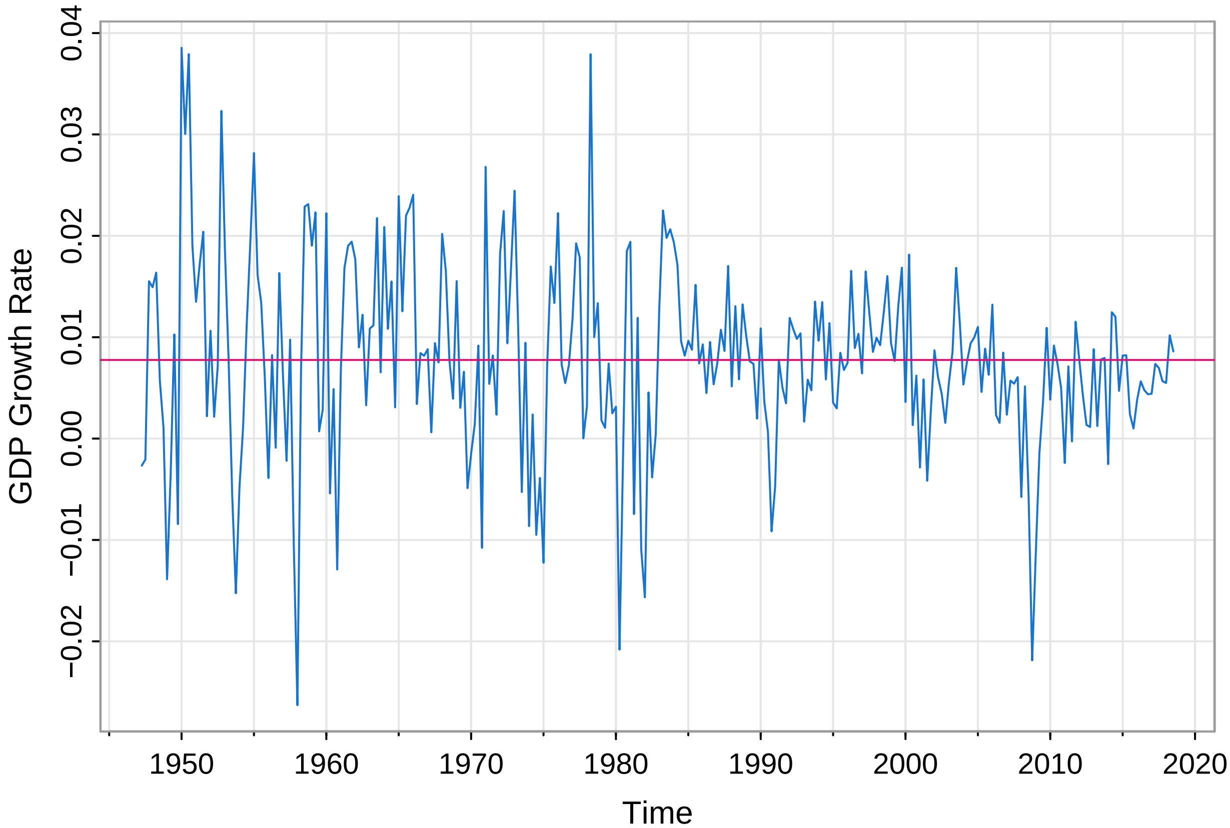

```
acf1(data, main = "")
```



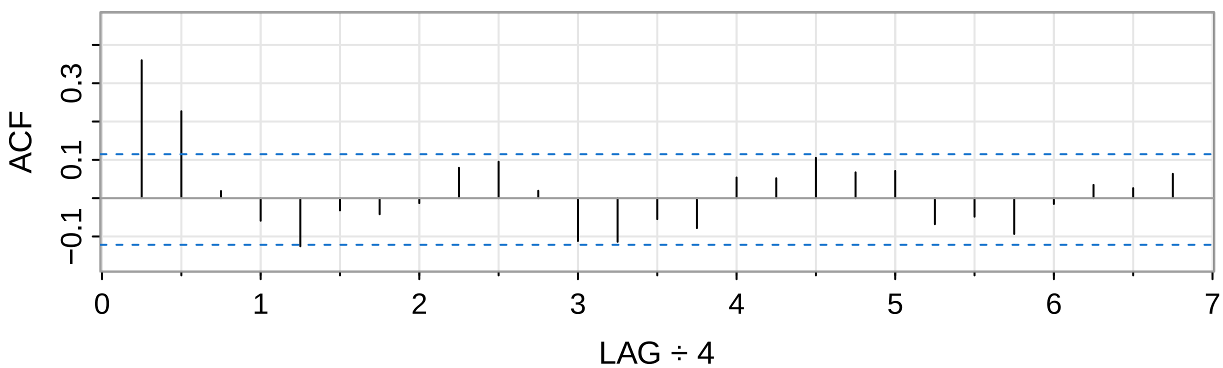

```
##  [1] 0.99 0.98 0.97 0.96 0.95 0.94 0.93 0.92 0.91 0.90 0.89 0.88 0.87 0.86 0.85
## [16] 0.84 0.83 0.82 0.81 0.80 0.79 0.78 0.77 0.76 0.75 0.74 0.73
```

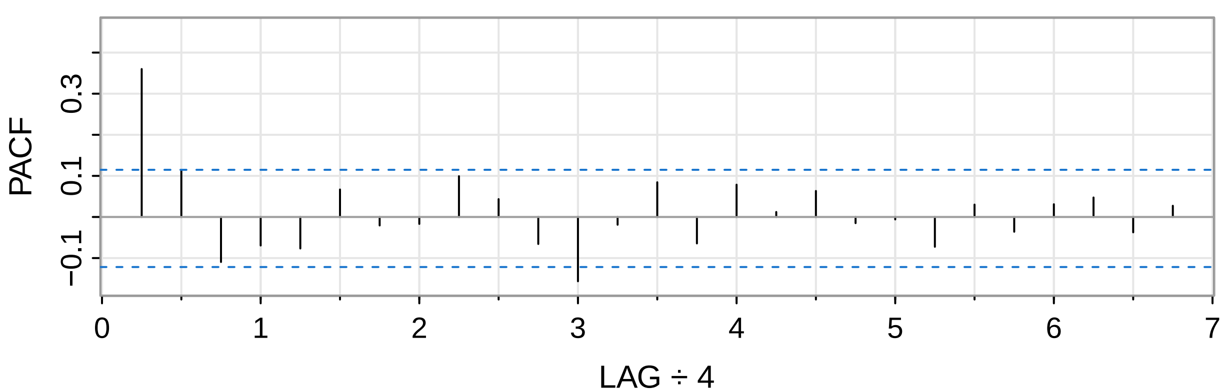
GDP data shows US GDP each quarter from each year from 1947 to 2018. First, we plot the GDP, and we can clearly see that there's a positive trend. This mean that this is not stationary.

```
tsplot(diff(log(data)), ylab = "GDP Growth Rate", col = 4)
mean<-mean(diff(log(data)))
abline(h = mean, col = 6)
```




```
acf2(diff(log(gdp)), main = "")
```



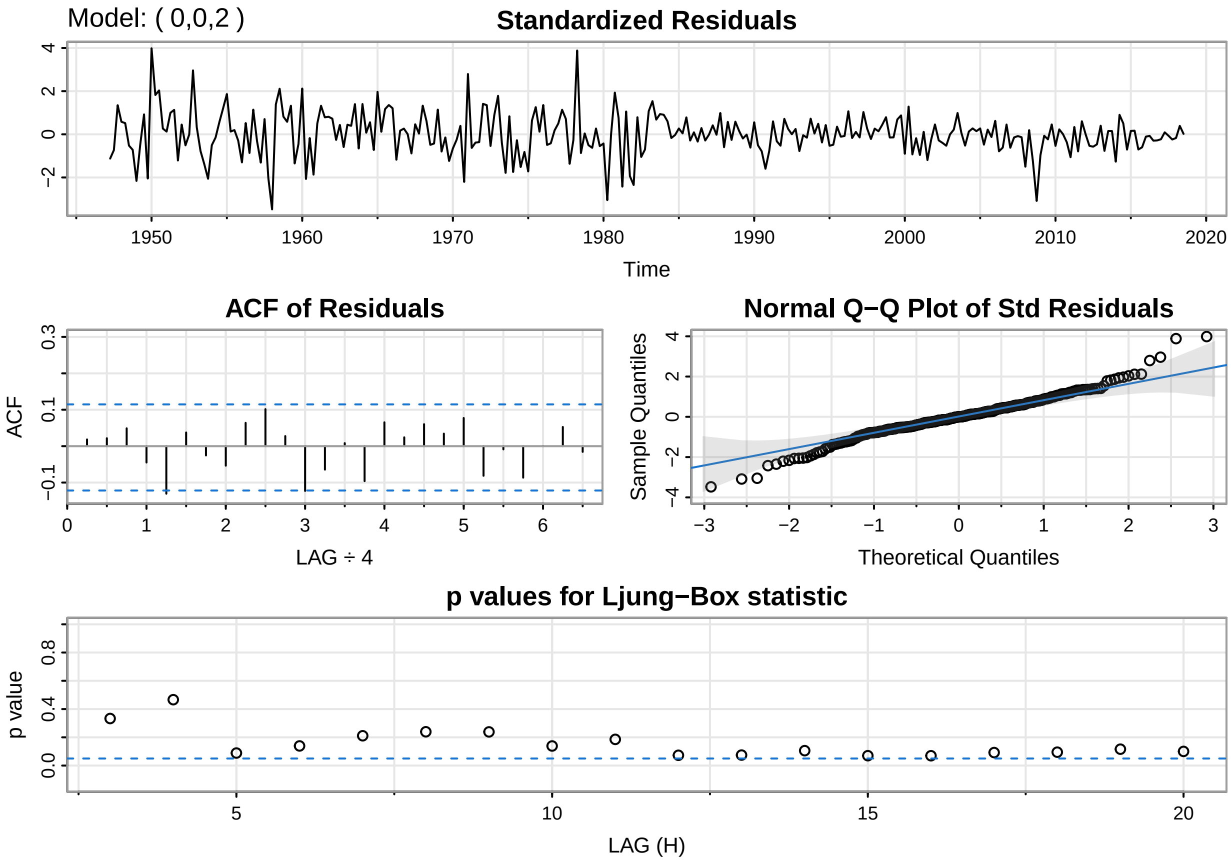


```
##      [,1] [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  0.36 0.23  0.02 -0.06 -0.13 -0.03 -0.04 -0.01 0.08  0.10  0.02 -0.11 -0.11
## PACF 0.36 0.11 -0.11 -0.07 -0.08  0.07 -0.02 -0.02 0.10  0.04 -0.07 -0.16 -0.02
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
## ACF  -0.05 -0.08  0.05  0.05  0.11  0.07  0.07 -0.07 -0.05 -0.09 -0.02  0.03
## PACF  0.08 -0.06  0.08  0.01  0.06 -0.02 -0.01 -0.07  0.03 -0.04  0.03  0.05
##      [,26] [,27]
## ACF   0.03  0.06
## PACF -0.04  0.03
```

From the ACF, we can see that the autocorrelation is getting smaller as the lags getting higher. To make it stationary, we'll plot the difference of the log(GDP), and the plot shows more homogeneous data, and we can see that it looks more stationary. We want to check the autocorrelation and the partial autocorrelation of the difference of the log(GDP). By analyzing the sample ACF and PACF, we might feel that the ACF is cutting off at lag 2 and the PACF is tailing off, which suggests an MA(2) model.

```
sarima(diff(log(gdp)), 0, 0, 2)
```

```
## initial  value -4.672758
## iter   2 value -4.749239
## iter   3 value -4.750696
## iter   4 value -4.750723
## iter   5 value -4.750724
## iter   6 value -4.750725
## iter   7 value -4.750725
## iter   7 value -4.750725
## iter   7 value -4.750725
## final  value -4.750725
## converged
## initial  value -4.751078
## iter   2 value -4.751080
## iter   3 value -4.751080
## iter   4 value -4.751081
## iter   5 value -4.751081
## iter   5 value -4.751081
## iter   5 value -4.751081
## final  value -4.751081
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##       Estimate     SE t.value p.value
## ma1     0.3070 0.0579  5.2988       0
## ma2     0.2258 0.0547  4.1271       0
## xmean   0.0077 0.0008  9.8631       0
##
## sigma^2 estimated as 7.464598e-05 on 283 degrees of freedom
##
## AIC = -6.636312  AICc = -6.636015  BIC = -6.58518
##
```



From the model, we fit:

$$\hat{x}_t = 0.0077 + 0.303\hat{w}_{t-1} + 0.204\hat{w}_{t-2} + \hat{w}_t$$

where $\hat{\sigma}_w = 0.00007464598$ is based on 283 degrees of freedom.

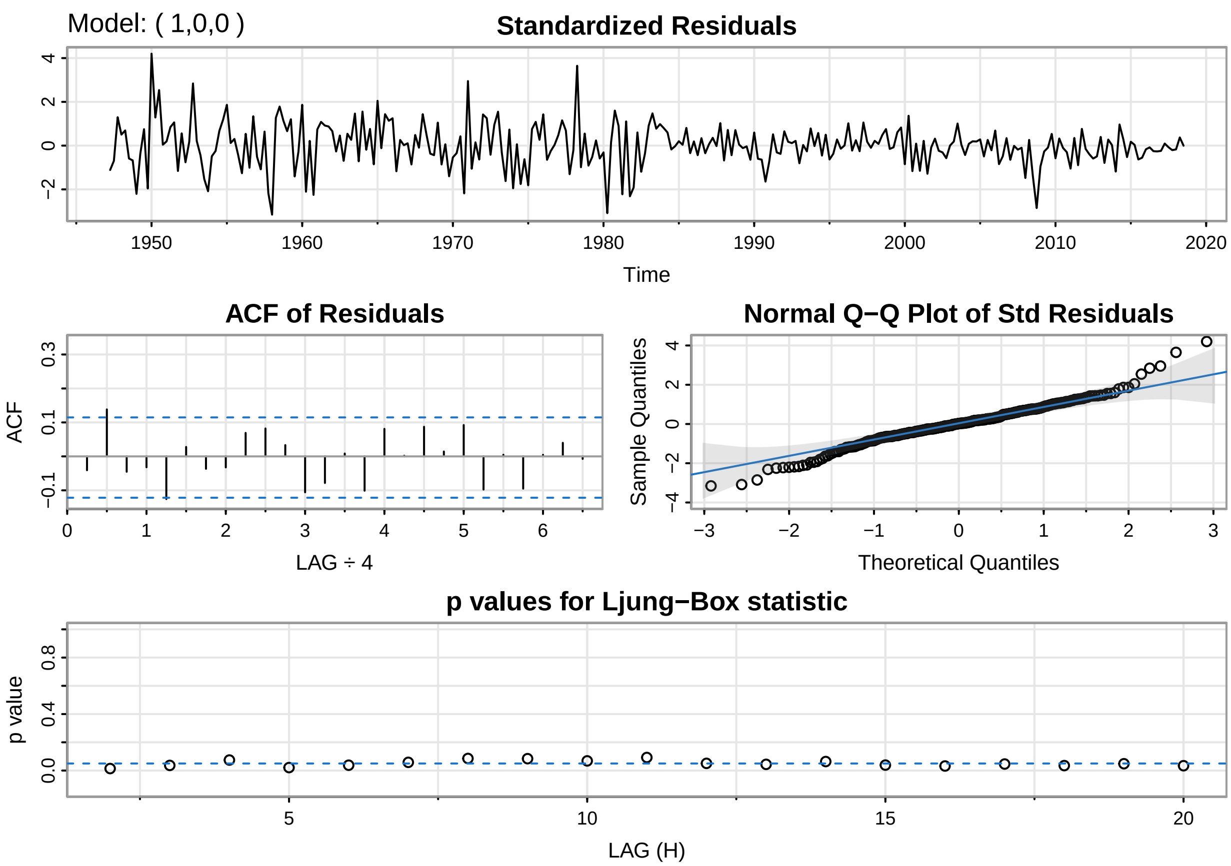
From the model summary, we can see all of the regression coefficients including the constant are very significant to the model.

We can also argue that ACF is tailing off, and PACF cut off at lag 1. Therefore, we want to analyze the AR(1) model.

```
sarima(diff(log(gdp)), 1, 0, 0)
```

```
## initial  value -4.673186
## iter   2 value -4.742918
## iter   3 value -4.742921
## iter   4 value -4.742923
## iter   5 value -4.742925
## iter   6 value -4.742925
## iter   6 value -4.742925
## final  value -4.742925
## converged
## initial  value -4.742229
## iter   2 value -4.742234
## iter   3 value -4.742245
## iter   3 value -4.742245
## iter   3 value -4.742245
## final  value -4.742245
```

```
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##         Estimate     SE t.value p.value
## ar1       0.3603 0.0551  6.5366       0
## xmean     0.0077 0.0008  9.5915       0
##
## sigma^2 estimated as 7.598489e-05 on 284 degrees of freedom
##
## AIC = -6.625634  AICc = -6.625485  BIC = -6.587284
##
```



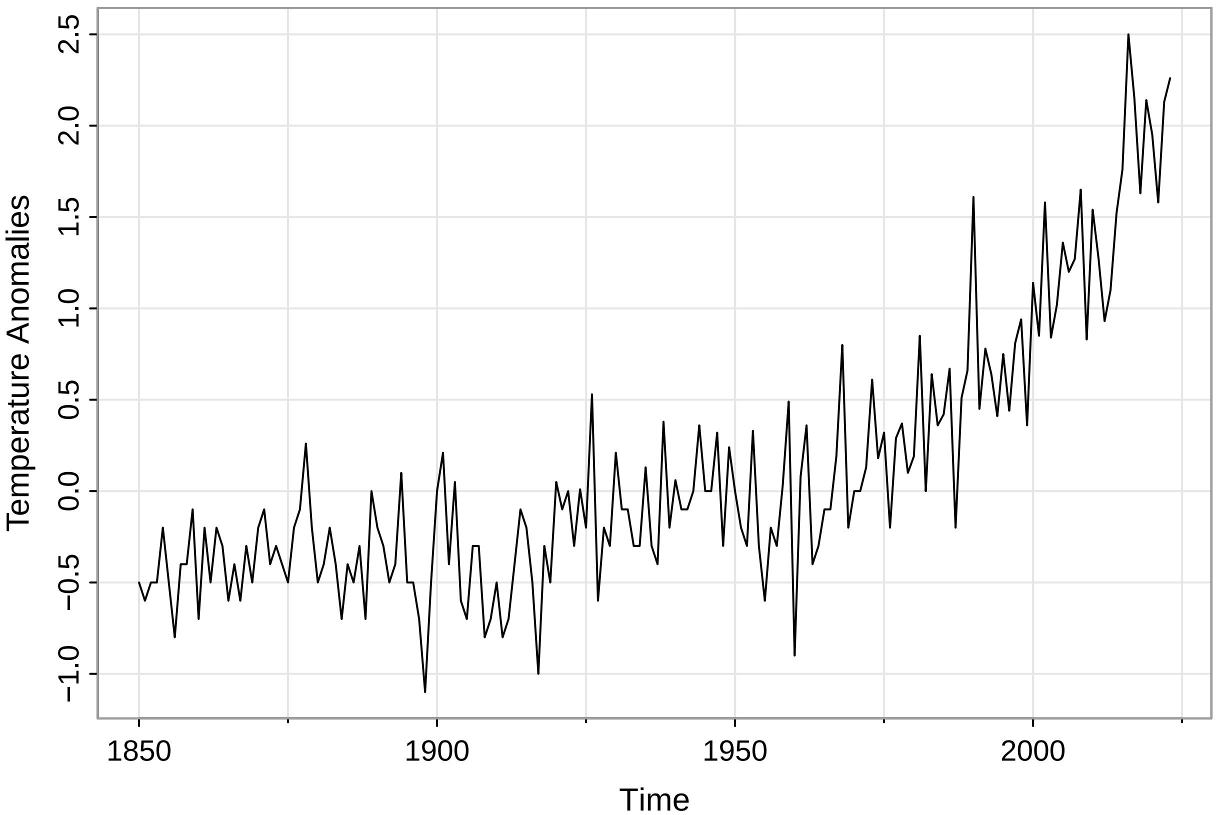
$$\hat{x}_t = 0.0077 + 0.3603\hat{x}_{t-1} + \hat{w}_t$$

We can see that AR(1) model also fits very well with all of the regression coefficients being very significant, and the model has a slightly higher AIC than MA(2), but it has a slightly lower BIC. Since it's better to choose a parsimony model, it might be better to prefer a simpler model AR(1).

However, by investigating the Ljung-Box statistic, we can see that AR(1) clearly looks different than MA(2) model. From the MA(2) model, we can see all the p-values exceed 0.05, so we can feel comfortable not rejecting the null hypothesis that the residuals are white. However, AR(1) have all the p-values of lower than 0.05, so we will reject the null hypothesis that the residuals are white. Therefore, from the diagnostic, we can say that MA(2) could be a better model for GDP data.

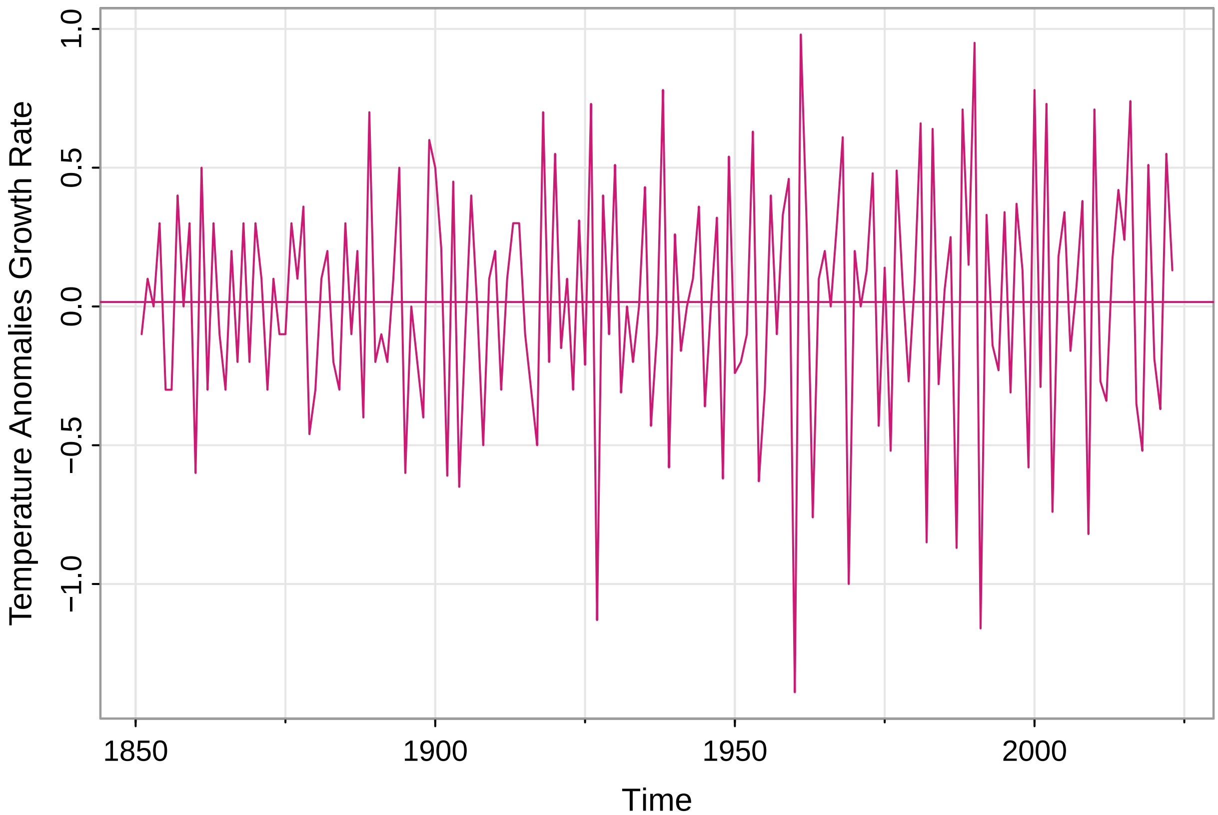
Question 5.4

```r
data<-gtemp_land
tsplot(data, ylab = "Temperature Anomalies")
```

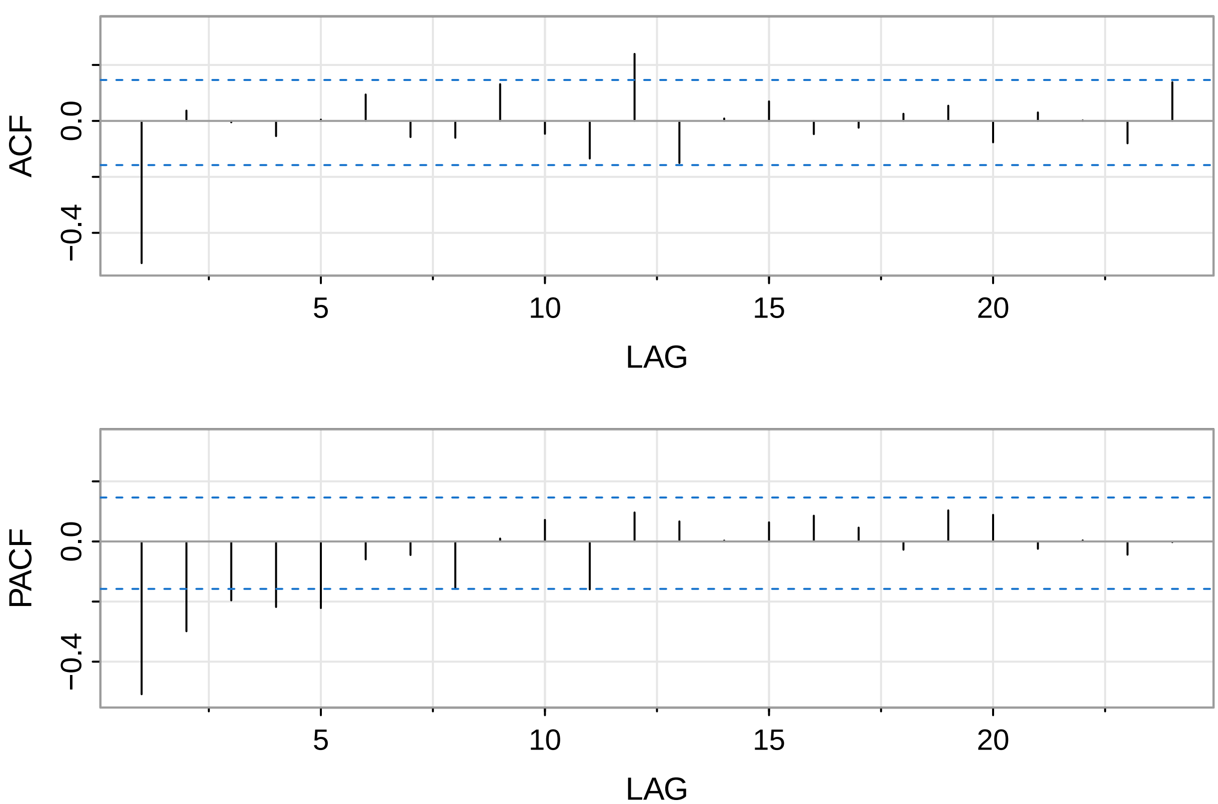


We can clearly see that the temperature anomalies has a positive trend, which means it's not stationary. To make it stationary, we'll take the difference of the gtemp_land.

```r
tsplot(diff(data), col = 6, ylab = "Temperature Anomalies Growth Rate")
mean<-mean(diff(data))
abline(h = mean, col = 6)
```




```
acf2(diff(data), main = "")
```

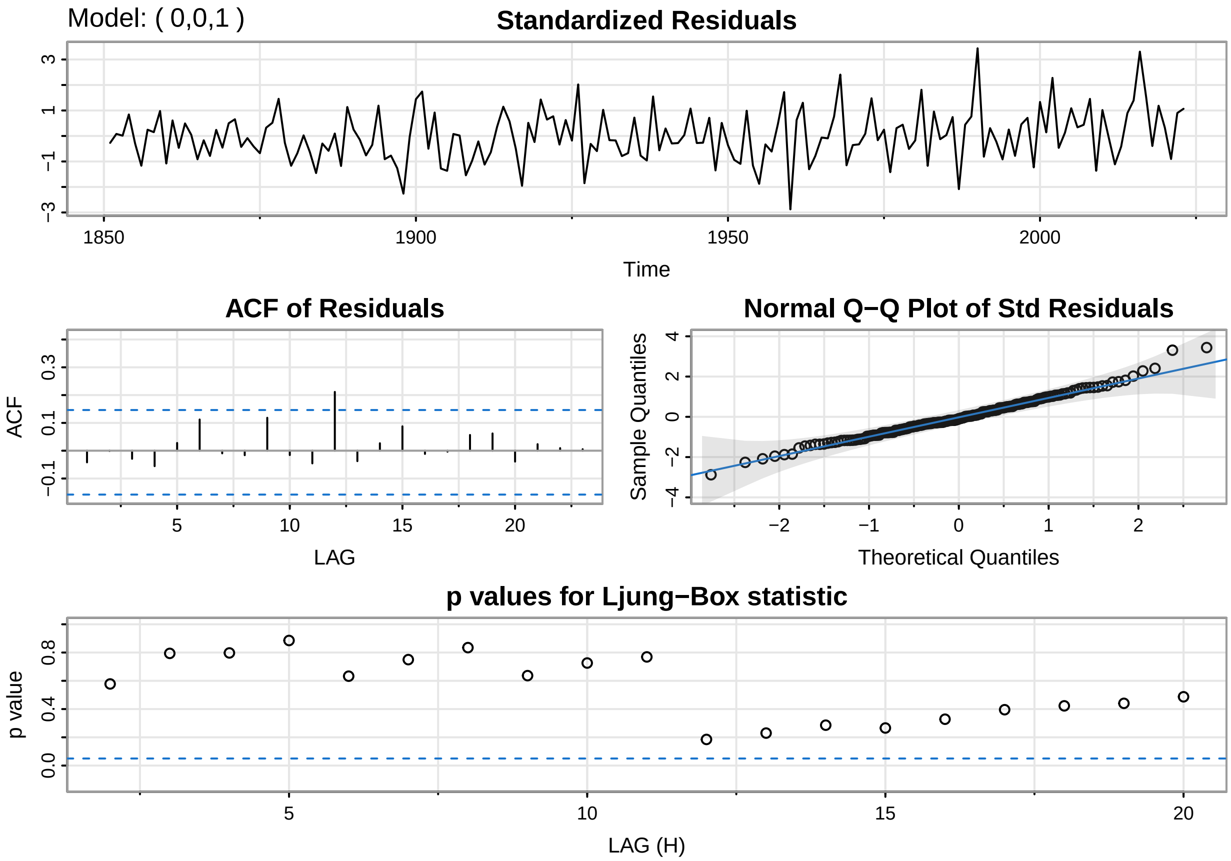


```
##       [,1]  [,2] [,3]  [,4]  [,5]  [,6]  [,7]  [,8] [,9] [,10] [,11] [,12]
## ACF  -0.51  0.04  0.0 -0.05  0.00  0.09 -0.06 -0.06 0.13 -0.05 -0.13  0.24
## PACF -0.51 -0.30 -0.2 -0.22 -0.22 -0.06 -0.04 -0.15 0.01  0.07 -0.16  0.10
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
## ACF  -0.15  0.01  0.07 -0.05 -0.02  0.03  0.05 -0.08  0.03     0 -0.08  0.14
## PACF  0.07  0.00  0.06  0.09  0.05 -0.03  0.10  0.09 -0.02     0 -0.04  0.00
```

By inspecting the sample ACF and PACF plot, we feel that ACF is cutoff at lag 1 and PACF is tailing off, and this suggest an AR(1) model.

```
sarima(diff(data), 0, 0, 1)
```

```
## initial  value -0.842449
## iter   2 value -1.043995
## iter   3 value -1.086437
## iter   4 value -1.105162
## iter   5 value -1.106533
## iter   6 value -1.107112
## iter   7 value -1.107288
## iter   8 value -1.107352
## iter   9 value -1.107368
## iter  10 value -1.107368
## iter  10 value -1.107368
## iter  10 value -1.107368
## final  value -1.107368
## converged
## initial  value -1.104513
## iter   2 value -1.104517
## iter   3 value -1.104520
## iter   4 value -1.104533
## iter   4 value -1.104533
## iter   4 value -1.104533
## final  value -1.104533
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##       Estimate     SE  t.value p.value
## ma1    -0.7974 0.0395 -20.1909  0.0000
## xmean   0.0145 0.0052   2.7905  0.0059
##
## sigma^2 estimated as 0.1091637 on 171 degrees of freedom
##
## AIC = 0.663493  AICc = 0.663901  BIC = 0.7181744
##
```

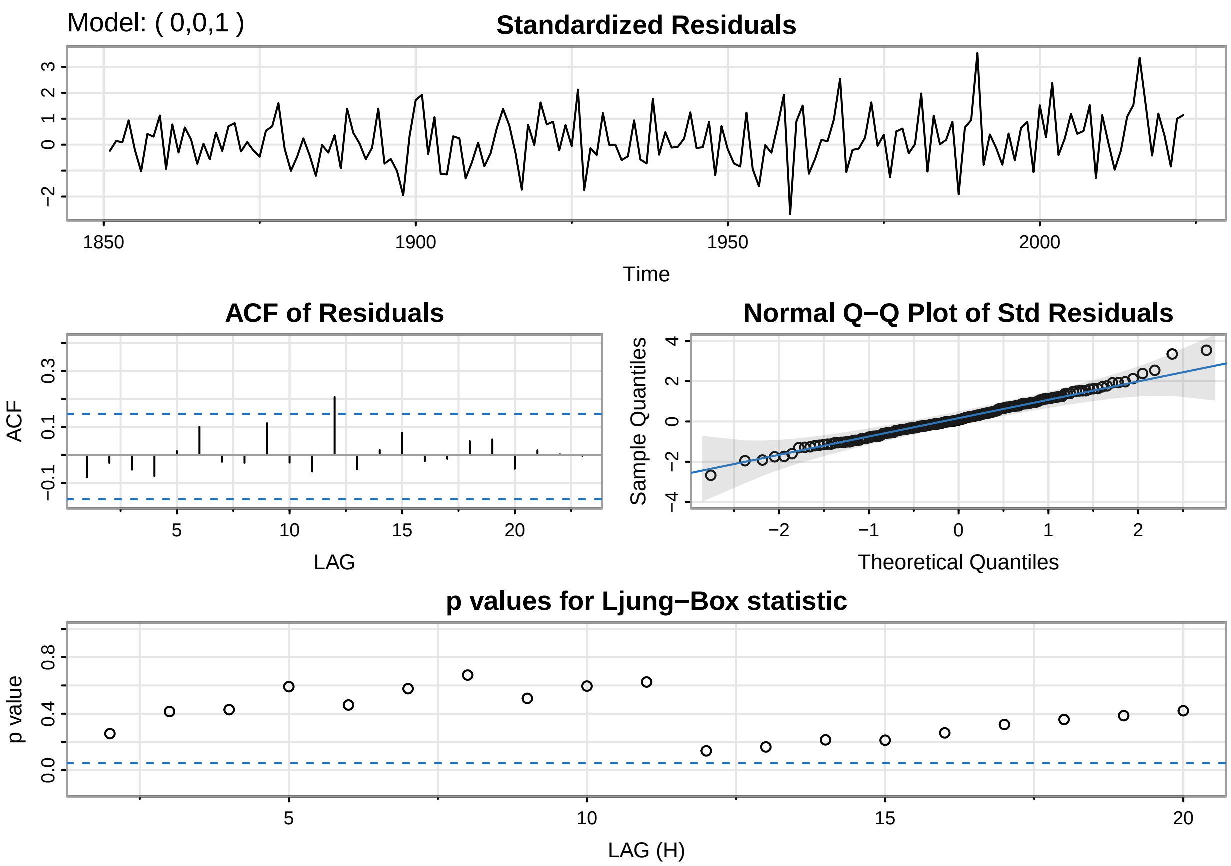


From the summary of the model, we can see that ma1 is signigicant, but the intercept is not as significant. Therefore, we might consider to remove the intercept from the model.

```
sarima(diff(data), 0, 0, 1, no.constant = TRUE)
```

```
## initial  value -0.841763
## iter   2 value -1.039571
## iter   3 value -1.074135
## iter   4 value -1.085539
## iter   5 value -1.087749
## iter   6 value -1.088478
## iter   7 value -1.088589
## iter   8 value -1.088591
## iter   9 value -1.088592
## iter   9 value -1.088592
## iter   9 value -1.088592
## final  value -1.088592
## converged
## initial  value -1.086133
## iter   2 value -1.086146
## iter   3 value -1.086148
## iter   3 value -1.086148
## iter   3 value -1.086148
## final  value -1.086148
## converged
## <><><><><><><><><><><><><><>
```

```
##
## Coefficients:
##     Estimate     SE  t.value p.value
## ma1  -0.7545 0.0423 -17.8414       0
##
## sigma^2 estimated as 0.1133625 on 172 degrees of freedom
##
## AIC = 0.6887024  AICc = 0.6888376  BIC = 0.7251567
##
```



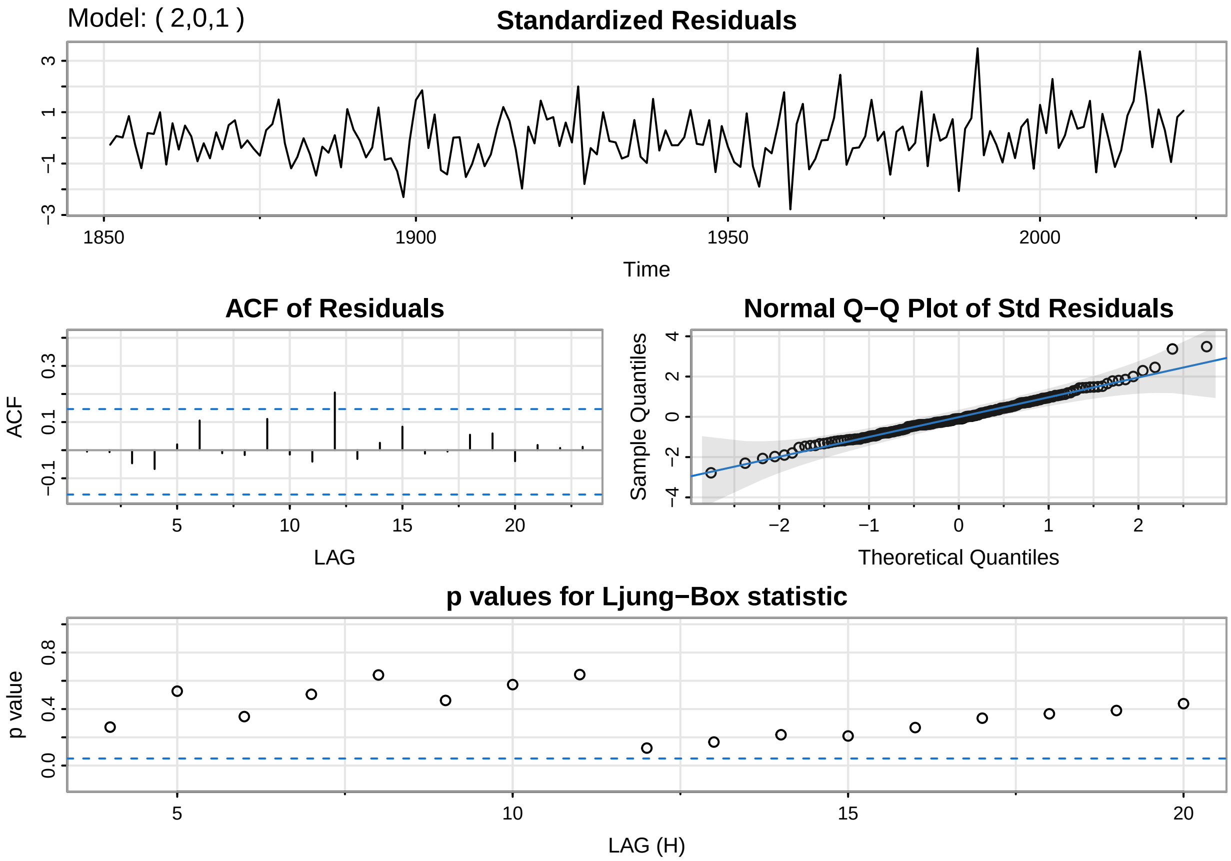
We can see from MA(1) with no intercept, the regression coefficient is still significant, and both AIC and BIC are higher than the original MA(1) model, which suggests that MA(1) with the intercept is a better model.

We can also argue that the model could be ARMA(2,1) where ACF cutsoff at lag 1 and PACF cutsoff at lag 2.

```
sarima(diff(data), 2, 0, 1)
```

```
## initial  value -0.836955
## iter   2 value -1.021219
## iter   3 value -1.076671
## iter   4 value -1.078385
## iter   5 value -1.086027
## iter   6 value -1.098045
## iter   7 value -1.101561
## iter   8 value -1.102978
```

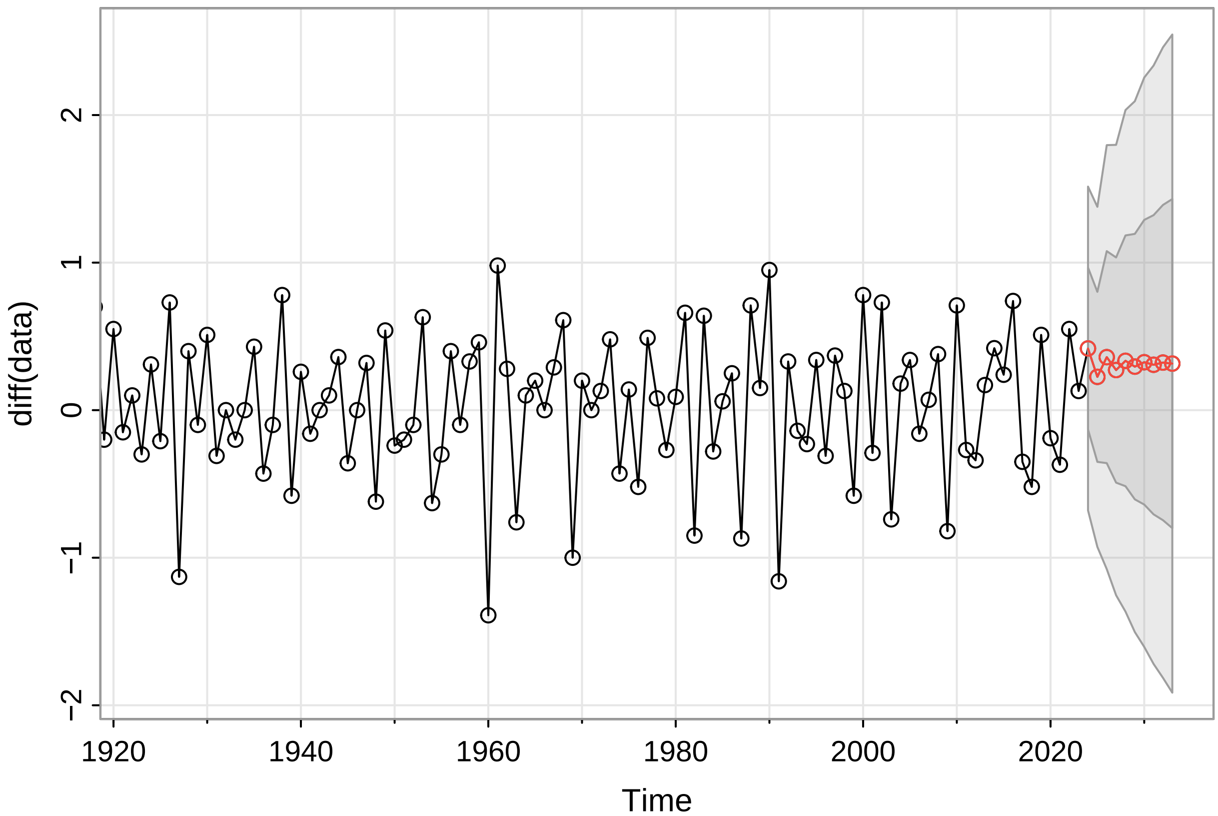
```
## iter   9 value -1.103031
## iter  10 value -1.103044
## iter  11 value -1.103052
## iter  12 value -1.103054
## iter  13 value -1.103055
## iter  14 value -1.103055
## iter  15 value -1.103055
## iter  15 value -1.103055
## iter  15 value -1.103055
## final  value -1.103055
## converged
## initial  value -1.105579
## iter   2 value -1.105599
## iter   3 value -1.105601
## iter   4 value -1.105602
## iter   5 value -1.105602
## iter   6 value -1.105602
## iter   6 value -1.105602
## iter   6 value -1.105602
## final  value -1.105602
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##       Estimate     SE  t.value p.value
## ar1    -0.0579 0.0954  -0.6067  0.5448
## ar2    -0.0156 0.0885  -0.1761  0.8604
## ma1    -0.7776 0.0589 -13.2025  0.0000
## xmean   0.0146 0.0053   2.7502  0.0066
##
## sigma^2 estimated as 0.1089133 on 169 degrees of freedom
##
## AIC = 0.6844762  AICc = 0.6858525  BIC = 0.7756118
##
```



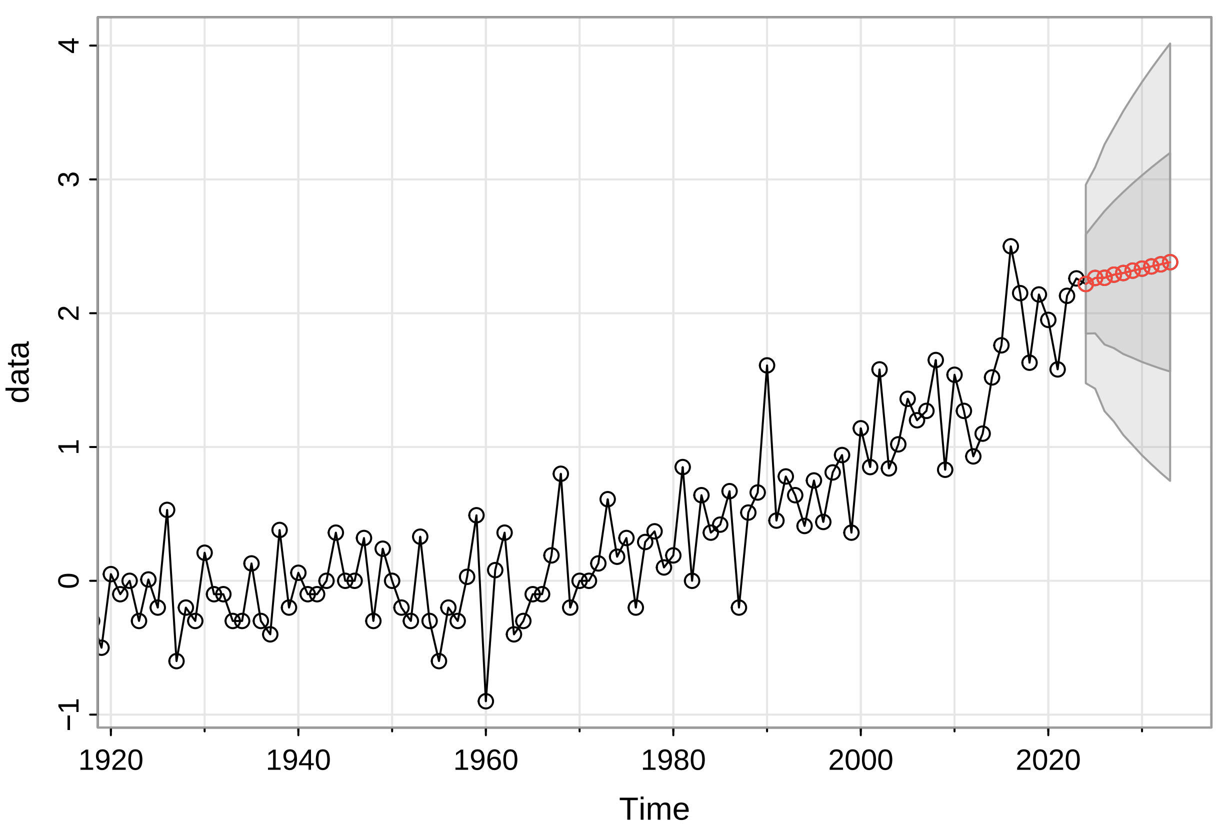
We can see that the some of ARMA(2,1) model's regression coefficients are not significant, which suggests that ARMA(2,1) model is not better than the MA(1) model.

From diagnostic, we can see that from Ljung-Box statistic shows that all lags have a p-value higher than 0.05, which means that we will not reject the null hypothesis of residuals being white noise. From the Q-Q plot, we can see that the observation fits pretty well to theoretical quantile, which means the data is normally distributed. Therefore, the best model for this data is MA(1).

```
predict_diff = sarima.for(diff(data), 10, 1, 1, 0)
```



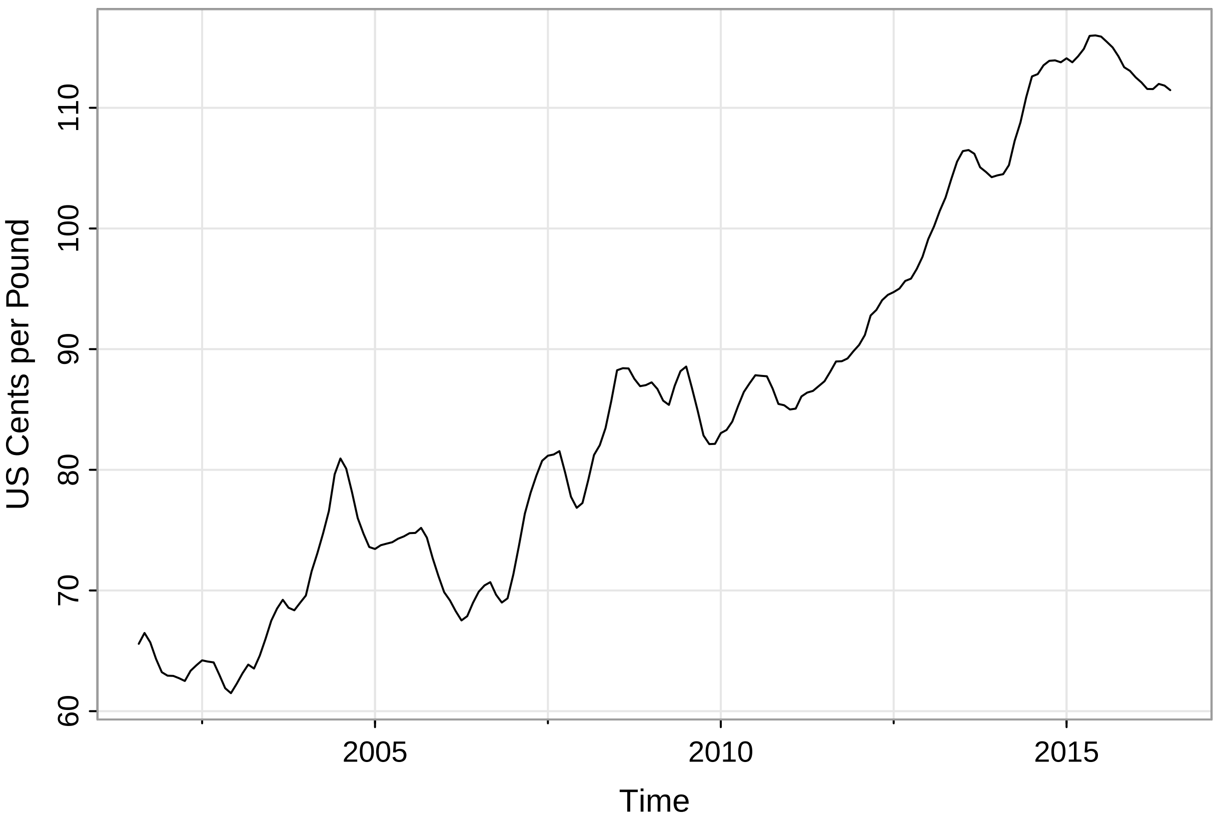

```r
predict_data = sarima.for(data, 10, 1, 1, 0)
```



By forecasting with our best model MA(1), the plot shows the continued upward trend of increasing land temperature deviations. This means that as time increase, the variations also increases so we can expect the more fluctuations.

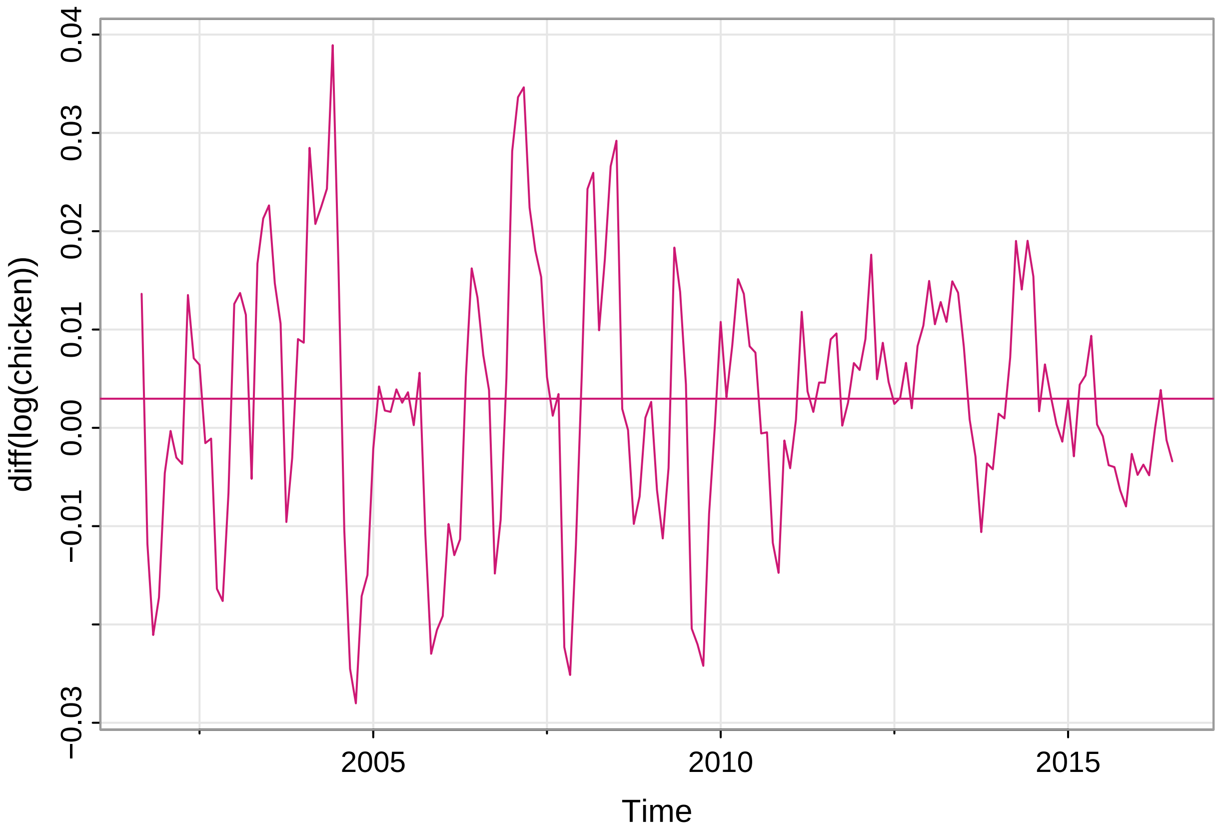
Question 5.9

```
data<-chicken
tsplot(data, ylab = "US Cents per Pound")
```

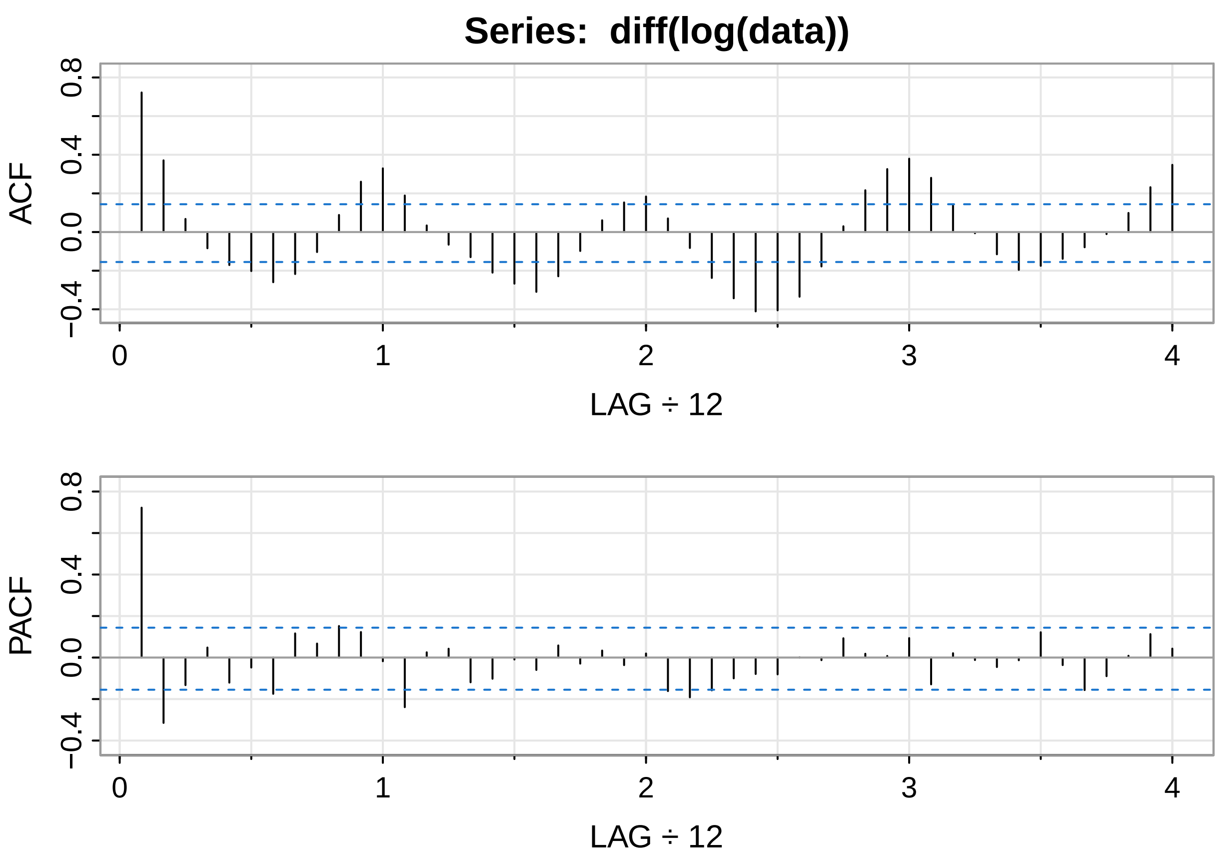


We can see that the Chicken price has an upward trend. This means that this data is not stationary. To make it stationary, we'll take the diff(log(data)).

```
tsplot(diff(log(data)), col = 6, ylab = "diff(log(chicken))")
mean<-mean(diff(log(data)))
abline(h = mean, col = 6)
```




```
acf2(diff(log(data)))
```



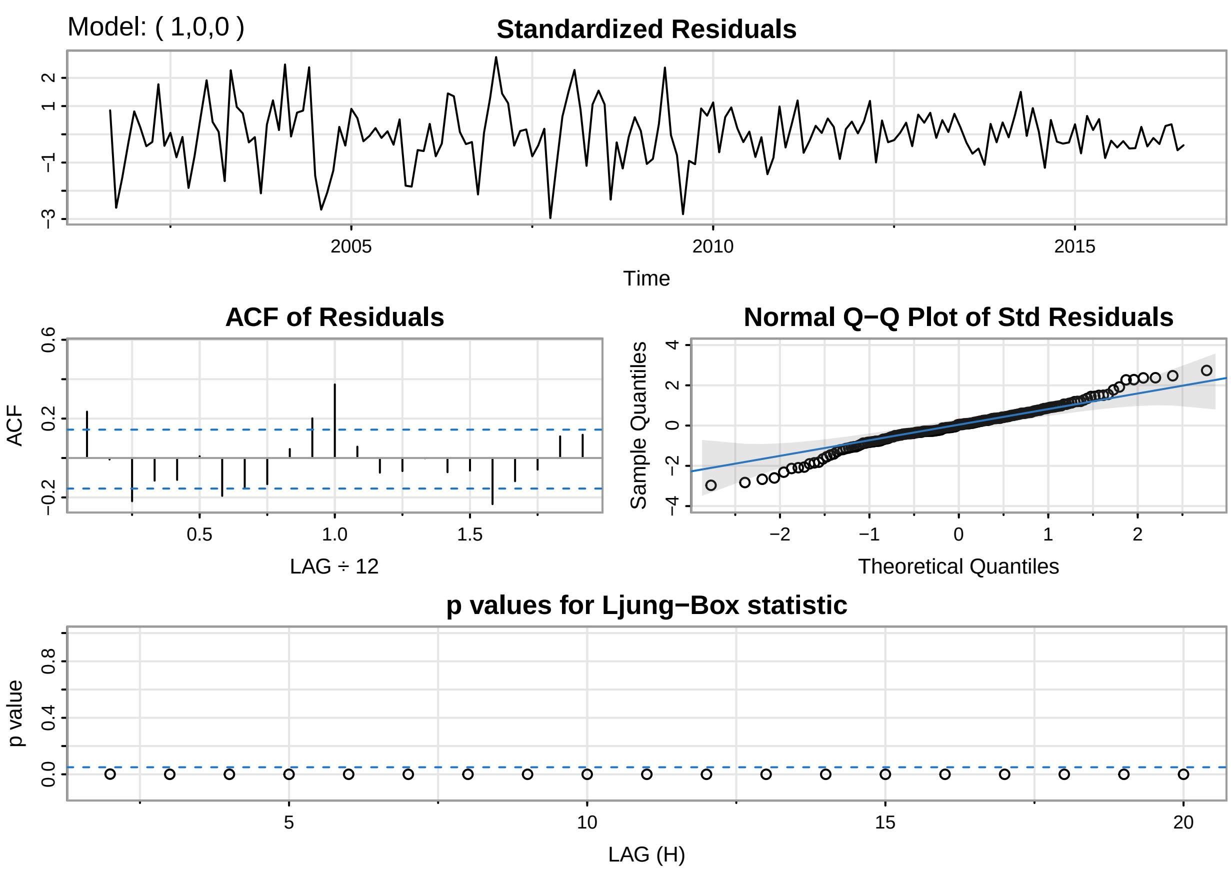

```
##       [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10] [,11] [,12]
## ACF  0.72  0.37  0.07 -0.08 -0.17 -0.20 -0.26 -0.22 -0.10  0.09  0.26  0.33
## PACF 0.72 -0.31 -0.13  0.05 -0.12 -0.05 -0.17  0.12  0.07  0.15  0.12 -0.02
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
## ACF   0.19  0.03 -0.07 -0.13 -0.21 -0.27 -0.31 -0.23 -0.10  0.06  0.15  0.18
## PACF -0.24  0.03  0.04 -0.12 -0.10 -0.01 -0.06  0.06 -0.03  0.03 -0.04  0.02
##      [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36]
## ACF   0.07 -0.08 -0.24 -0.34 -0.41 -0.41 -0.33 -0.18  0.03  0.22  0.33  0.38
## PACF -0.16 -0.19 -0.16 -0.10 -0.08 -0.08  0.00 -0.01  0.09  0.02  0.01  0.09
##      [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48]
## ACF   0.28  0.14 -0.01 -0.12 -0.20 -0.17 -0.14 -0.08 -0.01  0.10  0.23  0.35
## PACF -0.13  0.02 -0.01 -0.05 -0.01  0.12 -0.04 -0.16 -0.09  0.01  0.11  0.04
```

From the sample ACF and PACF plot, we feel that ACF is tailing off and PACF cuts off at lag 1. Therefore, we'll try the model AR(1) model.

```
sarima(diff(log(data)), 1, 0, 0)
```

```
## initial  value -4.378816
## iter   2 value -4.750463
## iter   3 value -4.750469
## iter   4 value -4.750479
## iter   5 value -4.750497
## iter   6 value -4.750505
## iter   7 value -4.750505
## iter   8 value -4.750506
```

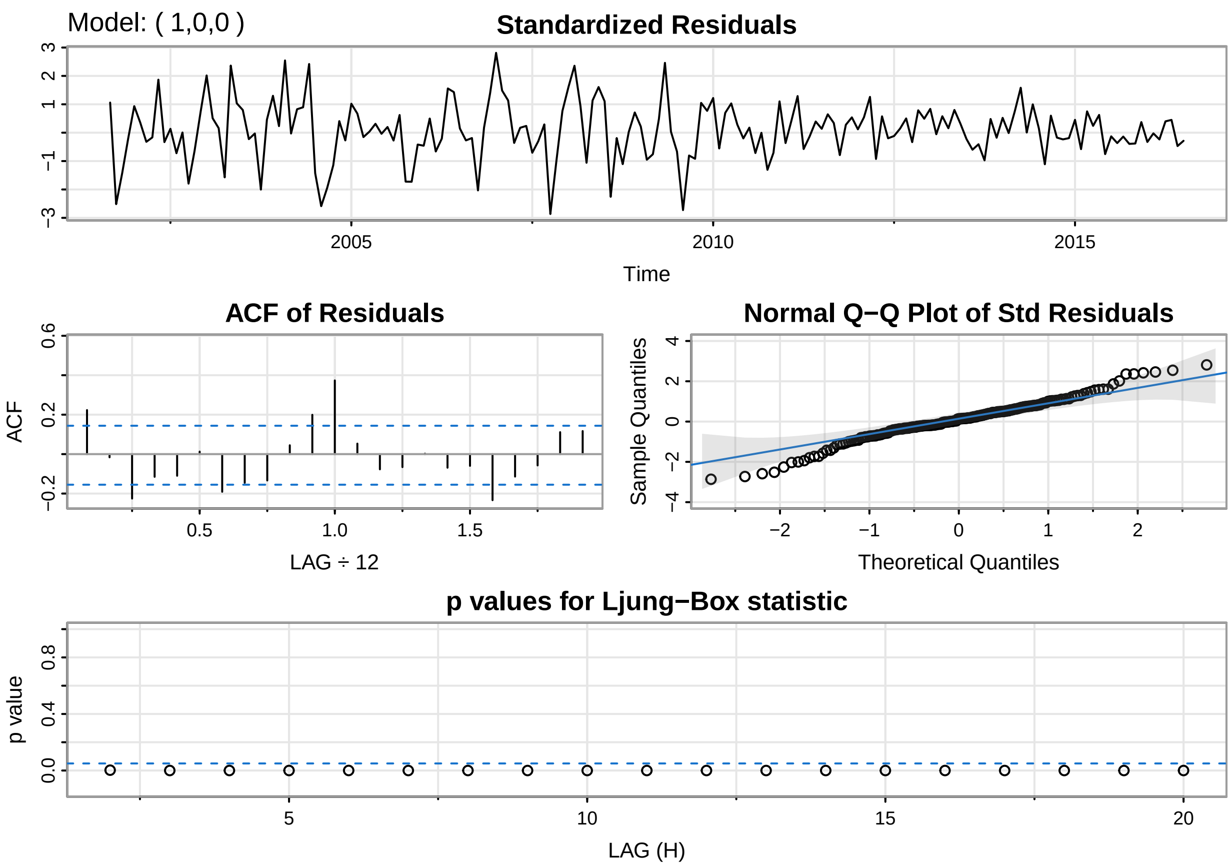
```
## iter   9 value -4.750506
## iter  10 value -4.750506
## iter  10 value -4.750506
## iter  10 value -4.750506
## final  value -4.750506
## converged
## initial  value -4.749085
## iter   2 value -4.749099
## iter   3 value -4.749155
## iter   4 value -4.749157
## iter   5 value -4.749159
## iter   5 value -4.749159
## iter   5 value -4.749159
## final  value -4.749159
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##       Estimate     SE t.value p.value
## ar1     0.7222 0.0512 14.1052  0.0000
## xmean   0.0030 0.0023  1.3188  0.1889
##
## sigma^2 estimated as 7.466975e-05 on 177 degrees of freedom
##
## AIC = -6.626921  AICc = -6.62654  BIC = -6.573501
##
```



We can see that from the model, the intercept is not significant, which suggests the removal of the intercept.

```
sarima(diff(log(data)), 1, 0, 0, no.constant = TRUE)
```

```
## initial  value -4.352722
## iter   2 value -4.747087
## iter   3 value -4.747101
## iter   4 value -4.747108
## iter   4 value -4.747108
## final  value -4.747108
## converged
## initial  value -4.744563
## iter   2 value -4.744563
## iter   3 value -4.744563
## iter   3 value -4.744563
## iter   3 value -4.744563
## final  value -4.744563
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##     Estimate   SE t.value p.value
## ar1    0.737 0.05 14.7259       0
##
## sigma^2 estimated as 7.533969e-05 on 178 degrees of freedom
##
## AIC = -6.628903  AICc = -6.628777  BIC = -6.59329
##
```



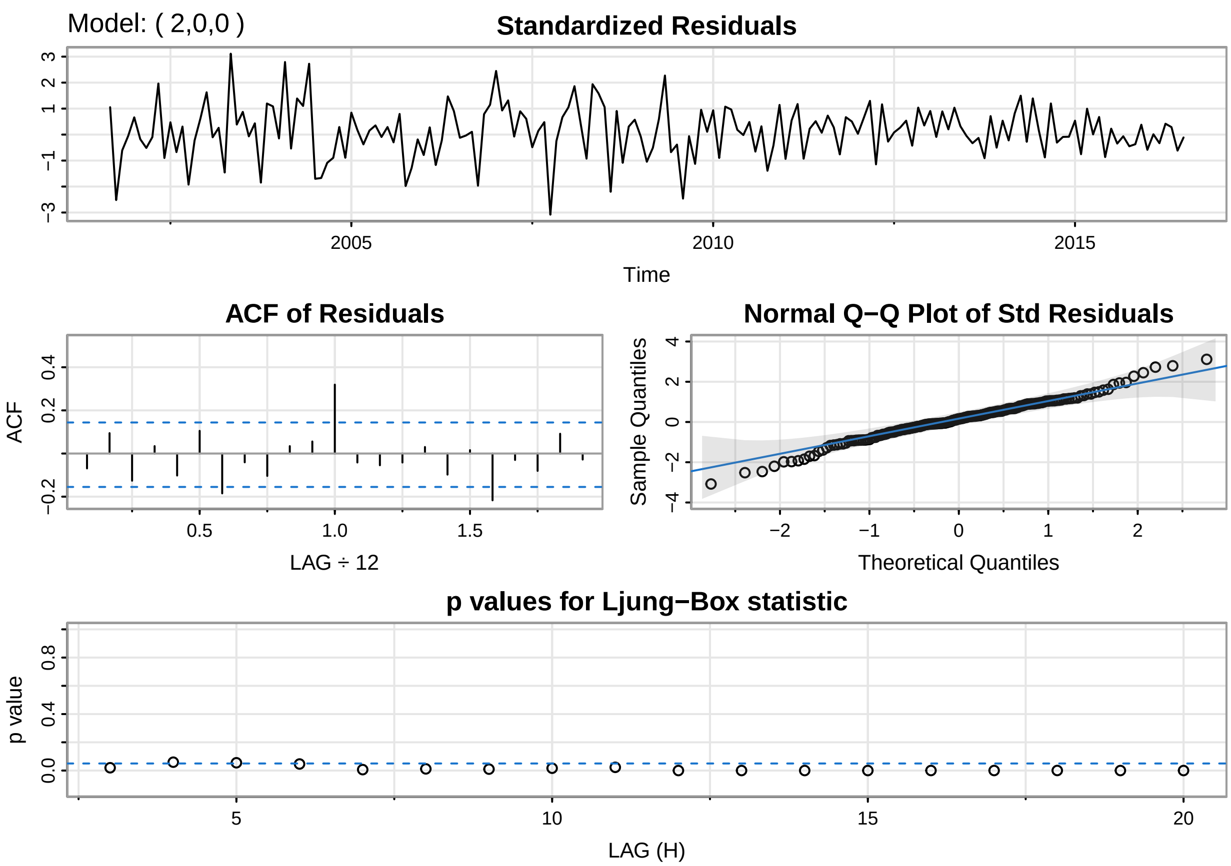
We can see that the regression coefficient remains significant, and both AIC an BIC decreased. Therefore this suggests that AR(1) model without intercept is better.

We can also see that PACF might be cutting off at lag 2 while ACF is still tailing off. Therefore, we'll try AR(2) model without the interept.

```
sarima(diff(log(data)), 2, 0, 0, no.constant = TRUE)
```

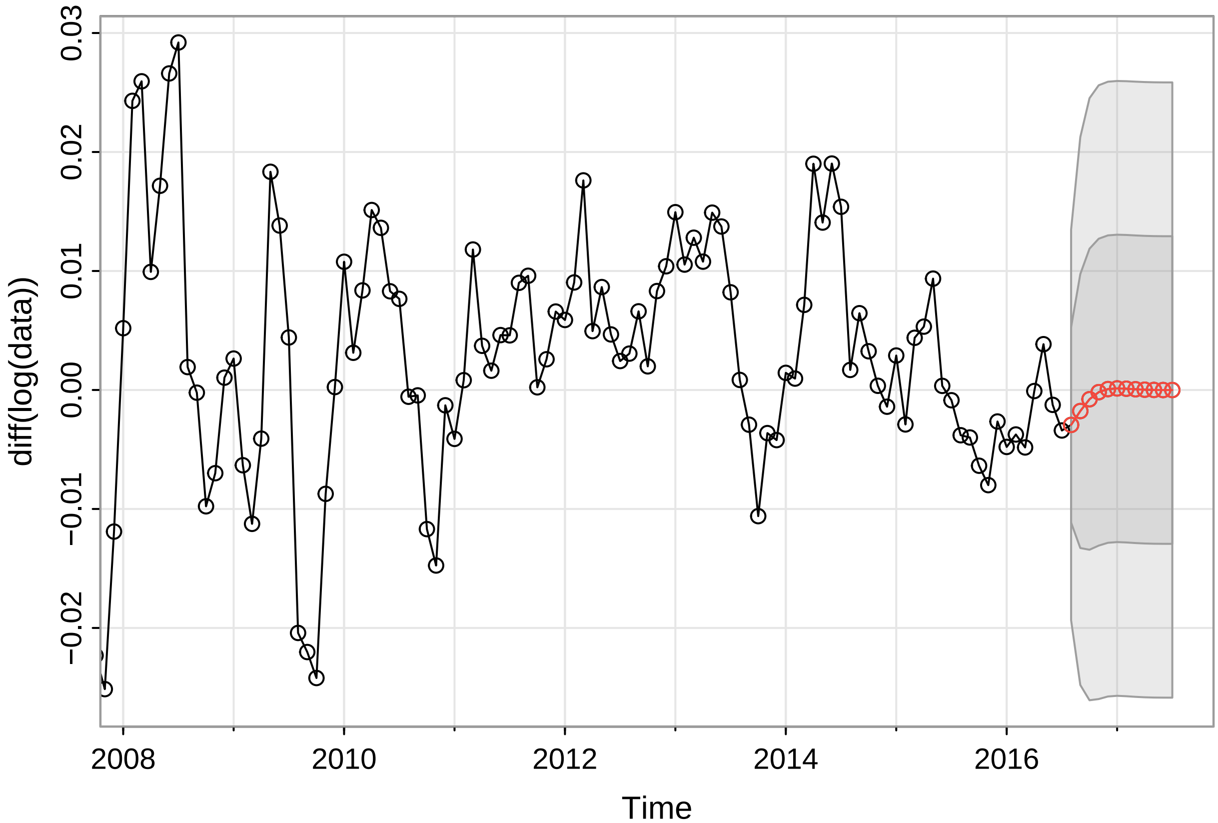
```
## initial  value -4.352272
## iter   2 value -4.508697
## iter   3 value -4.731994
## iter   4 value -4.806593
## iter   5 value -4.813483
## iter   6 value -4.818210
## iter   7 value -4.818213
## iter   8 value -4.818214
## iter   9 value -4.818214
## iter   9 value -4.818214
## iter   9 value -4.818214
## final  value -4.818214
## converged
## initial  value -4.799478
## iter   2 value -4.799584
## iter   3 value -4.799585
## iter   3 value -4.799585
## iter   3 value -4.799585
## final  value -4.799585
```

```
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##      Estimate     SE t.value p.value
## ar1    0.9830 0.0717 13.7048       0
## ar2   -0.3279 0.0718 -4.5696       0
##
## sigma^2 estimated as 6.739915e-05 on 177 degrees of freedom
##
## AIC = -6.727774  AICc = -6.727393  BIC = -6.674354
##
```

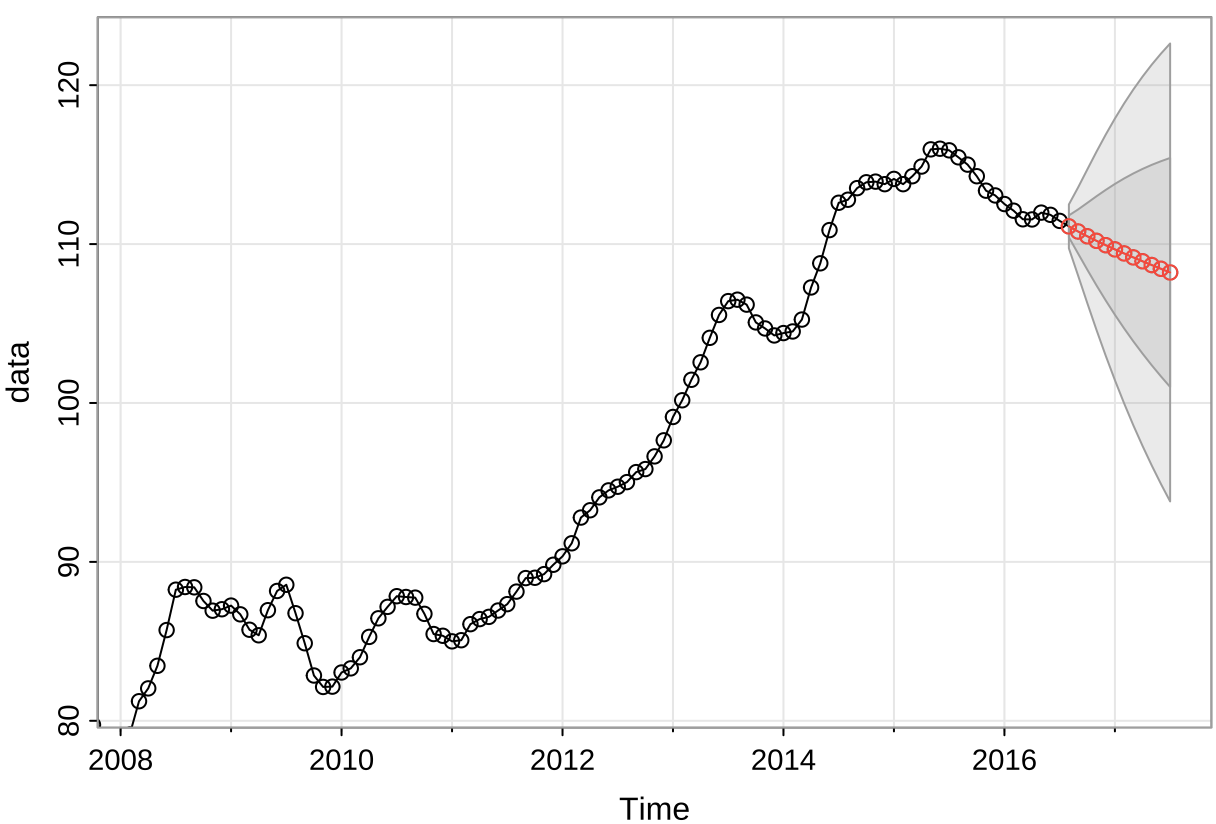


We can see that according to the model summary, both AIC and BIC are lower than the AR(1) model, which means AR(2) are better.Therefore, we'll forecast Chicken data with AR(2) model. However, we can't plot the original data with AR(2) without the intercept, so just for the visualization, we'll plot the chicken data with the AR(2) model with intercept.

```
predict_diff = sarima.for(diff(log(data)), 12, 2, 0, 0, no.constant = TRUE)
```



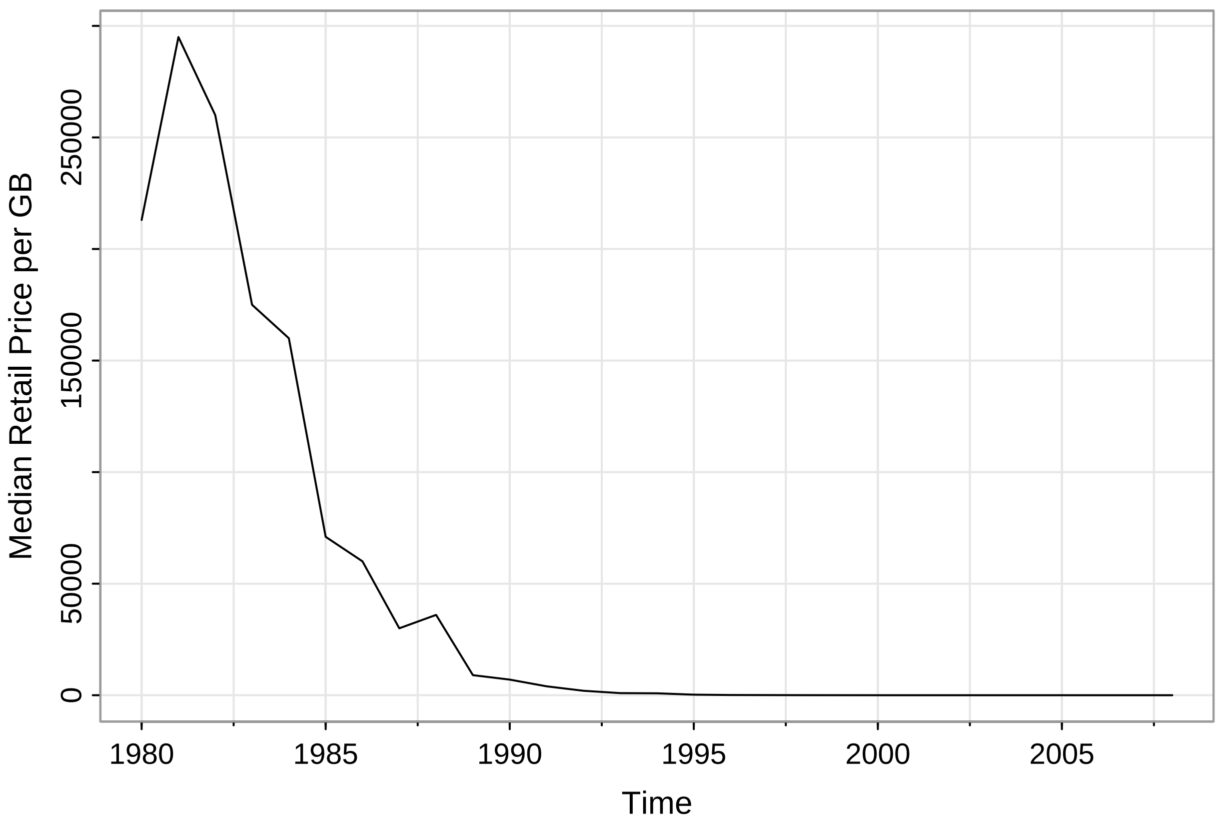

```r
predict_data = sarima.for(data, 12, 2, 0, 0)
```



We can see that from the forecast, we can see that it forecast the chicken price will decrease, which doesn't make sense. This mean we have a perfect fit model, but bad prediction.

Question 5.14a

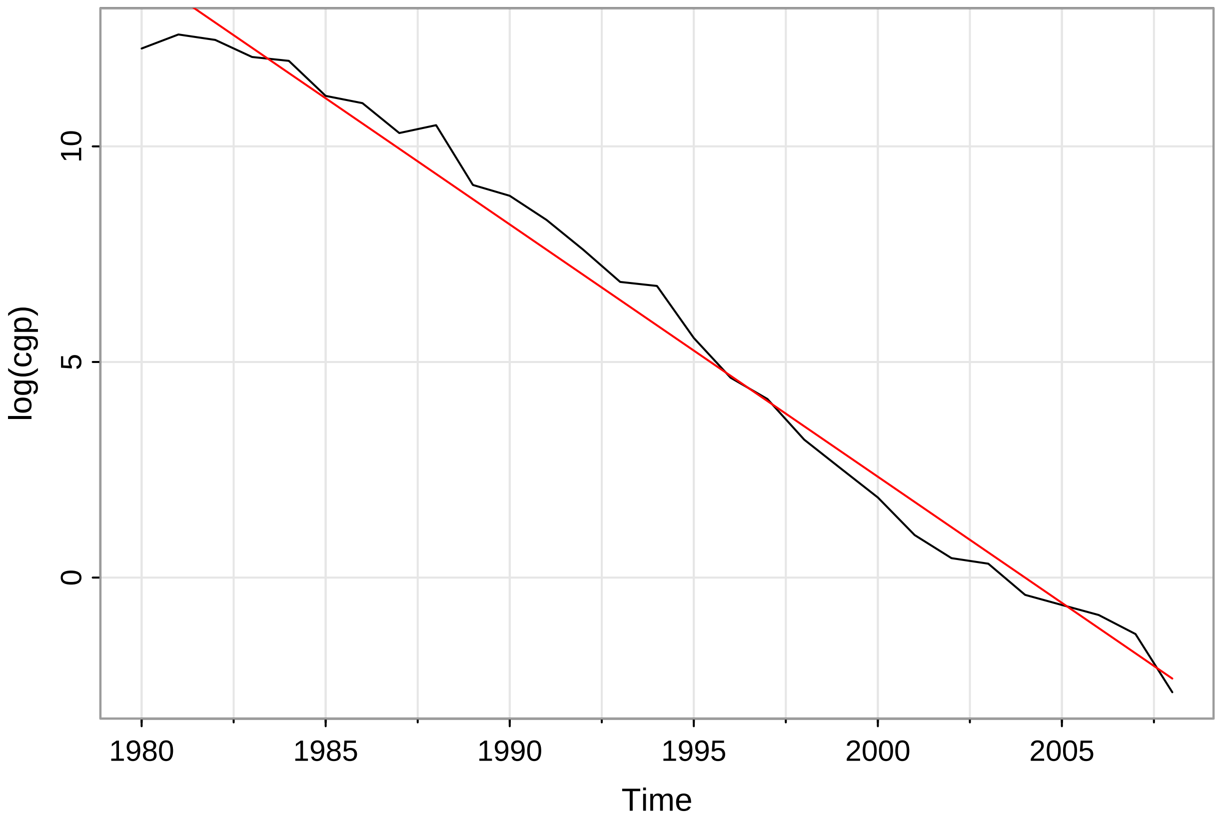
```
data<-cpg
tsplot(data, ylab = "Median Retail Price per GB")
```



From the plot, we see that the median price have a decreasing trend after 1982. We can see that the median price converge to 0 as the technology is more developed. We can see that the median price is near 0 after 1995.

Question 5.14b

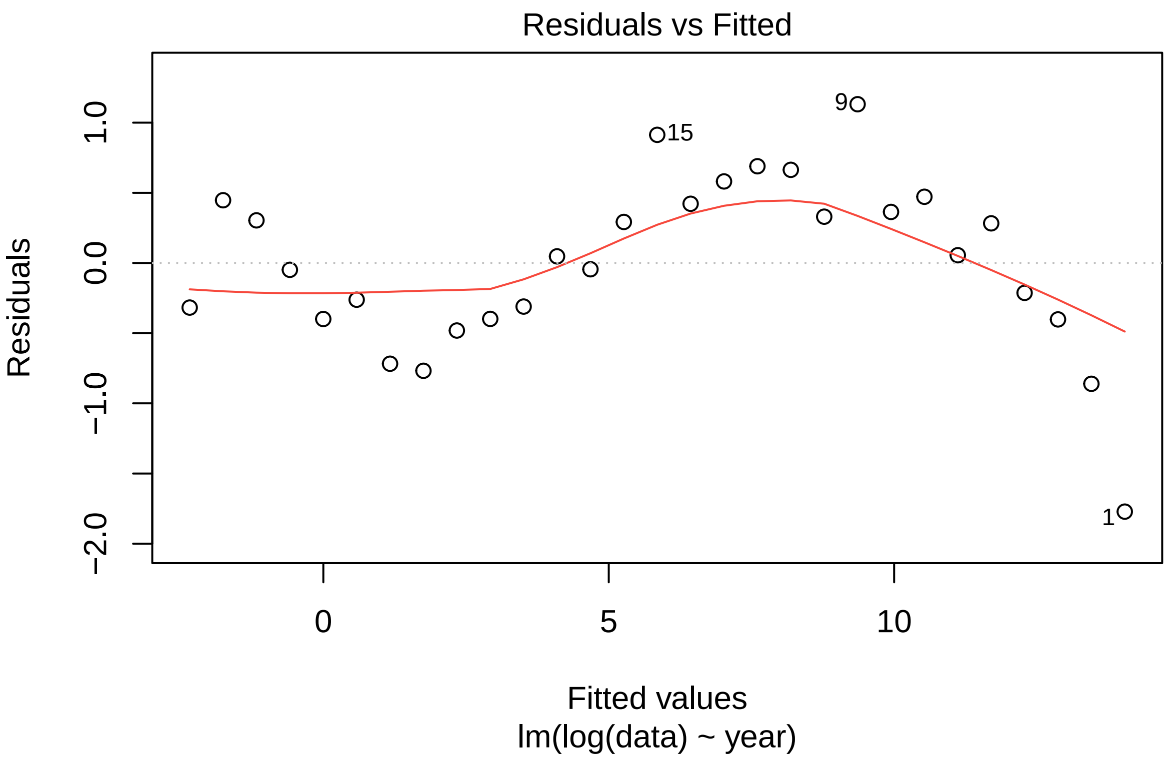
```
year<-time(data)
model<-lm(log(data)~year, na.action = NULL)
tsplot(log(data), ylab = "log(cgp)")
lines(fitted(model), col = "red")
```



We can see that the fitted model to the plot if a straight line. This means that c_t behave like $\alpha e^{1172.4943-0.5851x_{year}}$.

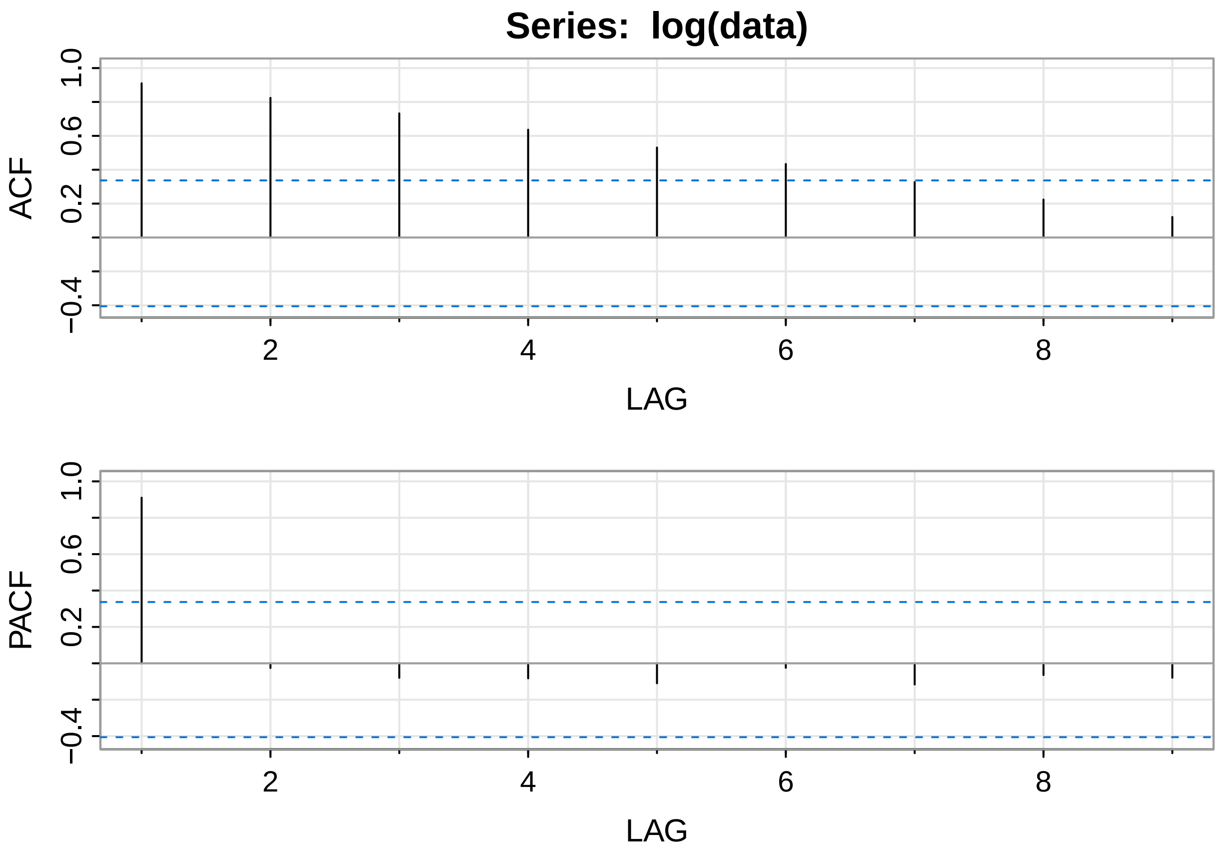
Question 5.14c

```
plot(model,1)
```



From the residual plot, we can clearly see there's a cyclic pattern in a residuals. This means that there are a pattern where the fitted model do not perform well. This suggests that the linear regression might not be a good model for this data. ## Question 5.14d

```
acf2(log(data))
```




```
##      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9]
## ACF  0.91  0.82  0.73  0.64  0.53  0.43  0.33  0.22  0.12
## PACF 0.91 -0.03 -0.08 -0.08 -0.11 -0.02 -0.12 -0.06 -0.08
```

From the ACF and PACF plot, we can see that the ACF is tailing off and the PACF cuts off at lag 1, so we'll use AR(1) model.

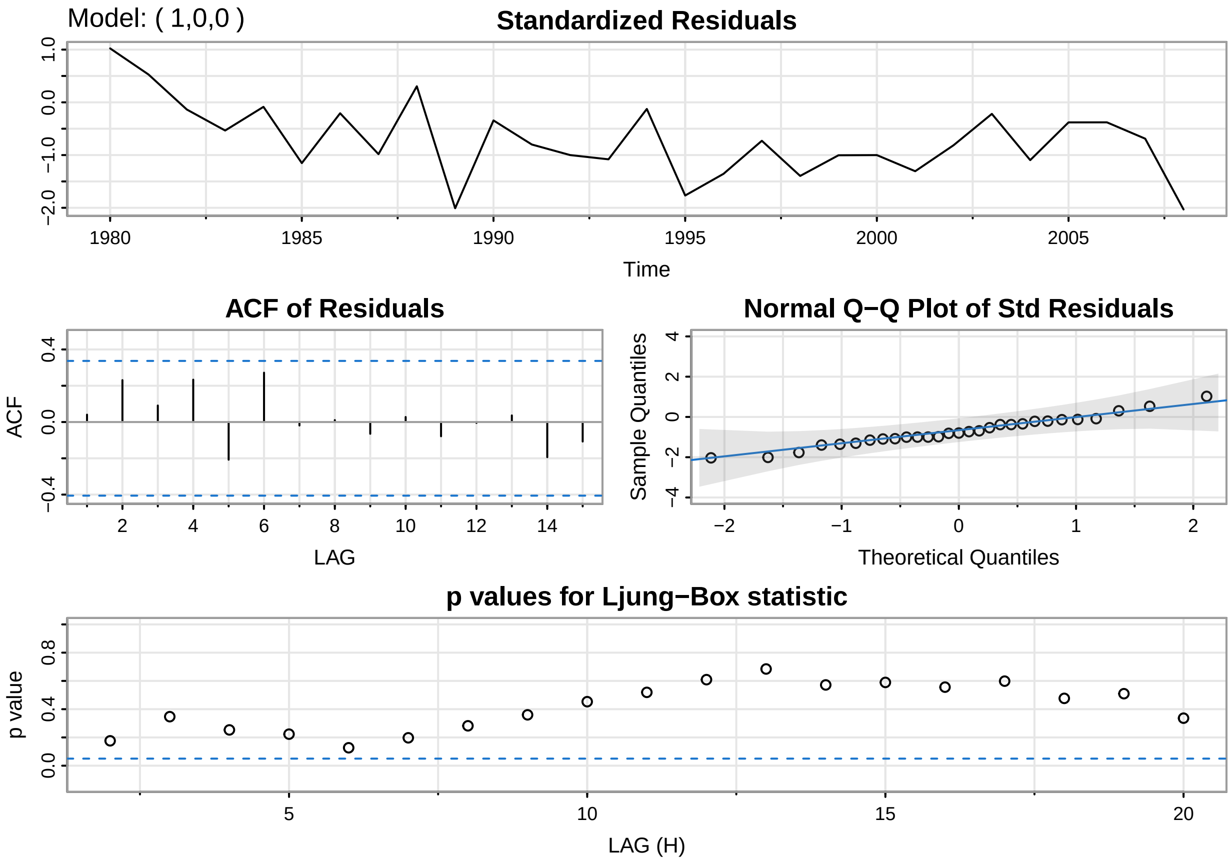
```
sarima(log(data), 1, 0, 0,)
```

```
## initial  value 1.583170
## iter   2 value -0.368933
## iter   3 value -0.382228
## iter   4 value -0.399952
## iter   5 value -0.497976
## iter   6 value -0.519338
## iter   7 value -0.589061
## iter   8 value -0.676984
## iter   9 value -0.702665
## iter  10 value -0.756358
## iter  11 value -0.758825
## iter  12 value -0.763442
## iter  13 value -0.772545
## iter  14 value -0.780862
## iter  15 value -0.805718
## iter  16 value -0.806190
## iter  17 value -0.806834
```

```
## iter  18 value -0.808751
## iter  19 value -0.812937
## iter  20 value -0.829659
## iter  21 value -0.830195
## iter  22 value -0.830338
## iter  23 value -0.830514
## iter  24 value -0.831142
## iter  25 value -0.832646
## iter  26 value -0.833581
## iter  27 value -0.833677
## iter  28 value -0.833810
## iter  29 value -0.834230
## iter  30 value -0.834980
## iter  31 value -0.835778
## iter  32 value -0.835833
## iter  33 value -0.836098
## iter  34 value -0.836649
## iter  35 value -0.838422
## iter  36 value -0.839396
## iter  37 value -0.839437
## iter  38 value -0.839529
## iter  39 value -0.839780
## iter  40 value -0.840428
## iter  41 value -0.840904
## iter  42 value -0.840952
## iter  43 value -0.840997
## iter  44 value -0.841193
## iter  45 value -0.841623
## iter  46 value -0.841931
## iter  47 value -0.841983
## iter  48 value -0.842016
## iter  49 value -0.842206
## iter  50 value -0.842596
## iter  51 value -0.842853
## iter  52 value -0.842916
## iter  53 value -0.842941
## iter  54 value -0.843150
## iter  55 value -0.843552
## iter  56 value -0.843783
## iter  57 value -0.843867
## iter  58 value -0.843886
## iter  59 value -0.844146
## iter  60 value -0.844623
## iter  61 value -0.844834
## iter  62 value -0.844965
## iter  63 value -0.844980
## iter  64 value -0.845361
## iter  65 value -0.846054
## iter  66 value -0.846206
## iter  67 value -0.846469
## iter  68 value -0.846480
## iter  69 value -0.847202
## iter  70 value -0.848701
## iter  71 value -0.849008
```

```
## iter  72 value -0.849151
## iter  73 value -0.849158
## iter  74 value -0.849570
## iter  75 value -0.849971
## iter  76 value -0.850366
## iter  77 value -0.850499
## iter  78 value -0.850505
## iter  79 value -0.850920
## iter  80 value -0.851732
## iter  81 value -0.852000
## iter  82 value -0.852059
## iter  83 value -0.852063
## iter  84 value -0.852233
## iter  85 value -0.852446
## iter  86 value -0.852671
## iter  87 value -0.852710
## iter  88 value -0.852713
## iter  89 value -0.852825
## iter  90 value -0.853024
## iter  91 value -0.853143
## iter  92 value -0.853161
## iter  93 value -0.853165
## iter  94 value -0.853219
## iter  95 value -0.853305
## iter  96 value -0.853388
## iter  97 value -0.853400
## iter  98 value -0.853403
## iter  99 value -0.853441
## iter 100 value -0.853512
## final  value -0.853512
## stopped after 100 iterations
## initial  value 1.595724
## iter   2 value 0.561504
## iter   3 value -0.232372
## iter   4 value -0.276286
## iter   5 value -0.298525
## iter   6 value -0.305311
## iter   7 value -0.307021
## iter   8 value -0.307054
## iter   9 value -0.307239
## iter  10 value -0.307240
## iter  11 value -0.307240
## iter  12 value -0.307240
## iter  12 value -0.307240
## final  value -0.307240
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##       Estimate     SE  t.value p.value
## ar1     0.9956 0.0062 161.3874   0.000
## xmean   4.8760 7.0163   0.6949   0.493
##
## sigma^2 estimated as 0.4594506 on 27 degrees of freedom
```

```
##
## AIC = 2.430294  AICc = 2.446209  BIC = 2.571738
##
```

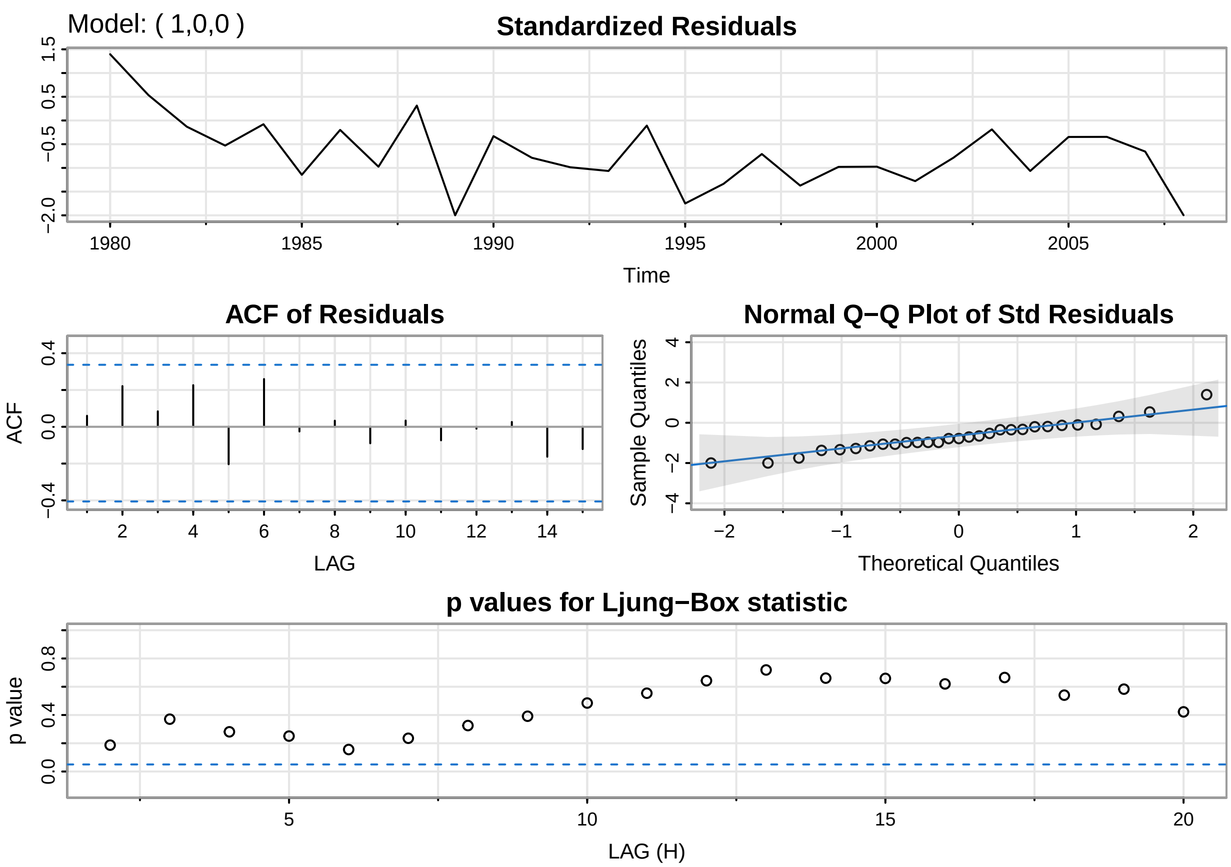


We can see that the intercept is not significant, so we want to remove the intercept to see if the model fits better.

```
sarima(log(data), 1, 0, 0, no.constant = TRUE)
```

```
## initial  value 2.005981
## iter   2 value -0.109653
## iter   3 value -0.349967
## iter   4 value -0.519614
## iter   5 value -0.548459
## iter   6 value -0.553046
## iter   7 value -0.553054
## iter   7 value -0.553054
## final  value -0.553054
## converged
## initial  value -0.077470
## iter   2 value -0.250289
## iter   3 value -0.279653
## iter   4 value -0.295200
## iter   5 value -0.296367
## iter   6 value -0.300260
## iter   7 value -0.300539
```

```
## iter   8 value -0.300542
## iter   9 value -0.300561
## iter  10 value -0.300561
## iter  10 value -0.300561
## final  value -0.300561
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##     Estimate     SE  t.value p.value
## ar1    0.997 0.0042 239.7262       0
##
## sigma^2 estimated as 0.4594107 on 28 degrees of freedom
##
## AIC = 2.374685  AICc = 2.379794  BIC = 2.468982
##
```



We can see that both AIC and BIC decreases, so AR(1) model without the intercept is a better fitting model.

From the ACF of residuals, we can see that the residuals are more random and closer to white noise. Additionally, from the Ljung-Box statistic, we can see that all p-values are higher than 0.05, so we'll not reject the null hypothesis that the residuals are white noise. Thus, this means that the AR(1) model is better than the linear regression model.