HW2

Name: Philip Situmorang

The data set for this assignment gives used car sales in the U.S. monthly, in millions of dollars, for the period 1992 through 2021(11). The data are in the file UsedCarSales.txt. To do this assignment, you need to construct the Time variable and convert the class for Month to factor class.

Then add these new variables to the data frame [sales<-data.frame(sales, Time, fMonth, if the name of the data frame you read in is sales]. If you create any other new variables, for example, outlier dummies, they also need to be added to the data frame.

```r
sales<-read.csv("UsedCarSales.txt")

Time <- seq(1, nrow(sales))
fMonth<-as.factor(sales$Month)

sales<-data.frame(sales, Time, fMonth)
```

1. During the years 1992 to 2021 the Business Cycle Dating Committee of the National Bureau of Economic Research defined three periods of economic contraction, 2001(4) to 2001(11), 2008(1) to 2009(6), and 2020(3) to 2020(4). Make separate time series plots for (i) Sales and (ii) log Sales, and mark the contraction periods on the plots. Discuss and compare the two plots.

Comment on trend structure, seasonality, and volatility. Do the plots reveal any unusual features? If yes, describe what is notable and discuss the underlying causes. Do the two plots indicate whether an additive decomposition model or a multiplicative decomposition model should be fit to model Sales? Explain your answer.

ANSWER:

Trend: There is an upward trend in used car sales from 1992 to 2021. Overall trend either flattens or drops during economic contraction periods.

Seasonality: Strong seasonality indicated by the plots. Used car sales seem to peak in the spring and summer months and are lowest in winter months. The cycle seems to be annual.

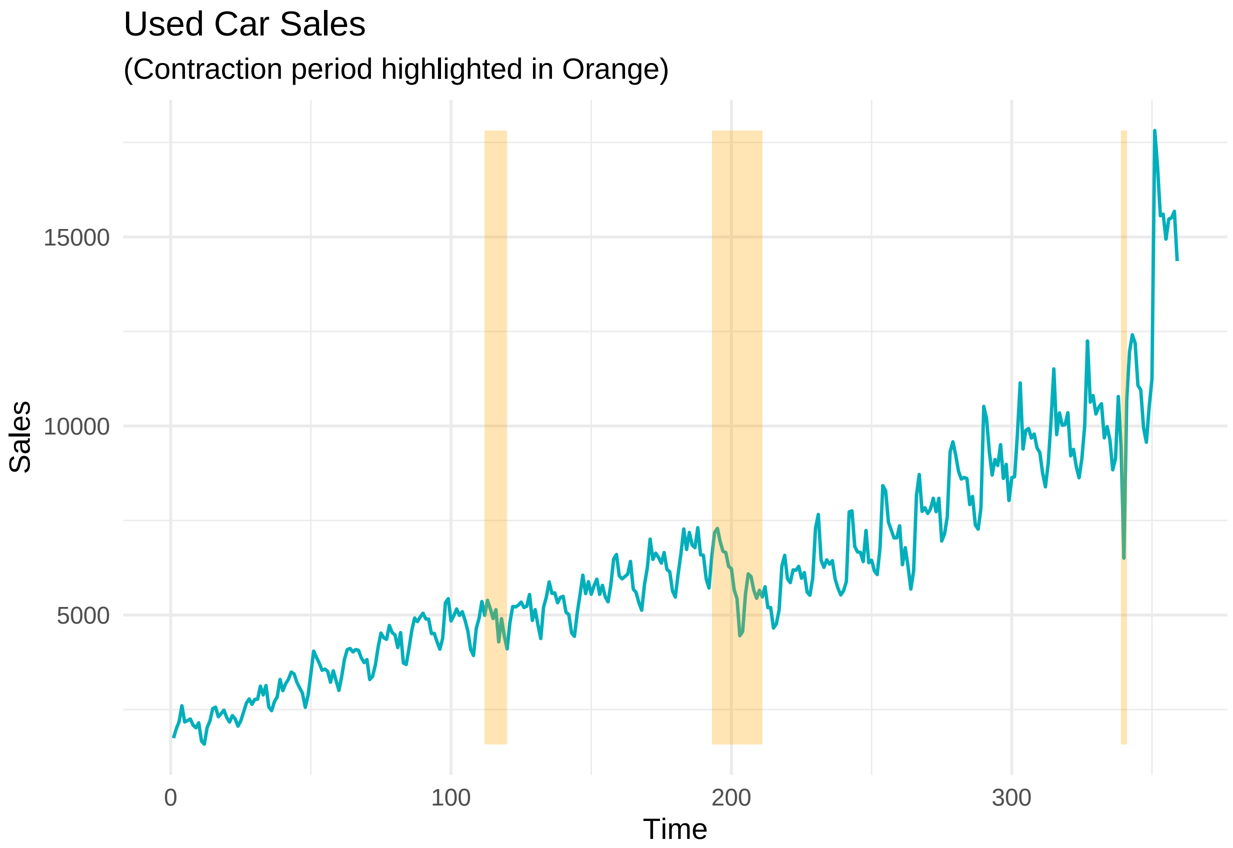
Volatility: Volatility in sales increases as sales increases. This indicates that a multiplicative decomposition model should be fit to model sales.

Unusual features: Overall trend either flattens or drops during economic contraction periods. Between 2001(4) to 2001(11) sales level flattens and in both 2008(1) to 2009(6) and 2020(3) to 2020(4) sales visibly drops. In 2020(3) to 2020(4) in particular, sales sharply drops due to the start of the COVID-19 pandemic and consumers were on lockdown. Sales would peak twice after 2020(3) drop, likely following the two economic impact payments by the government.

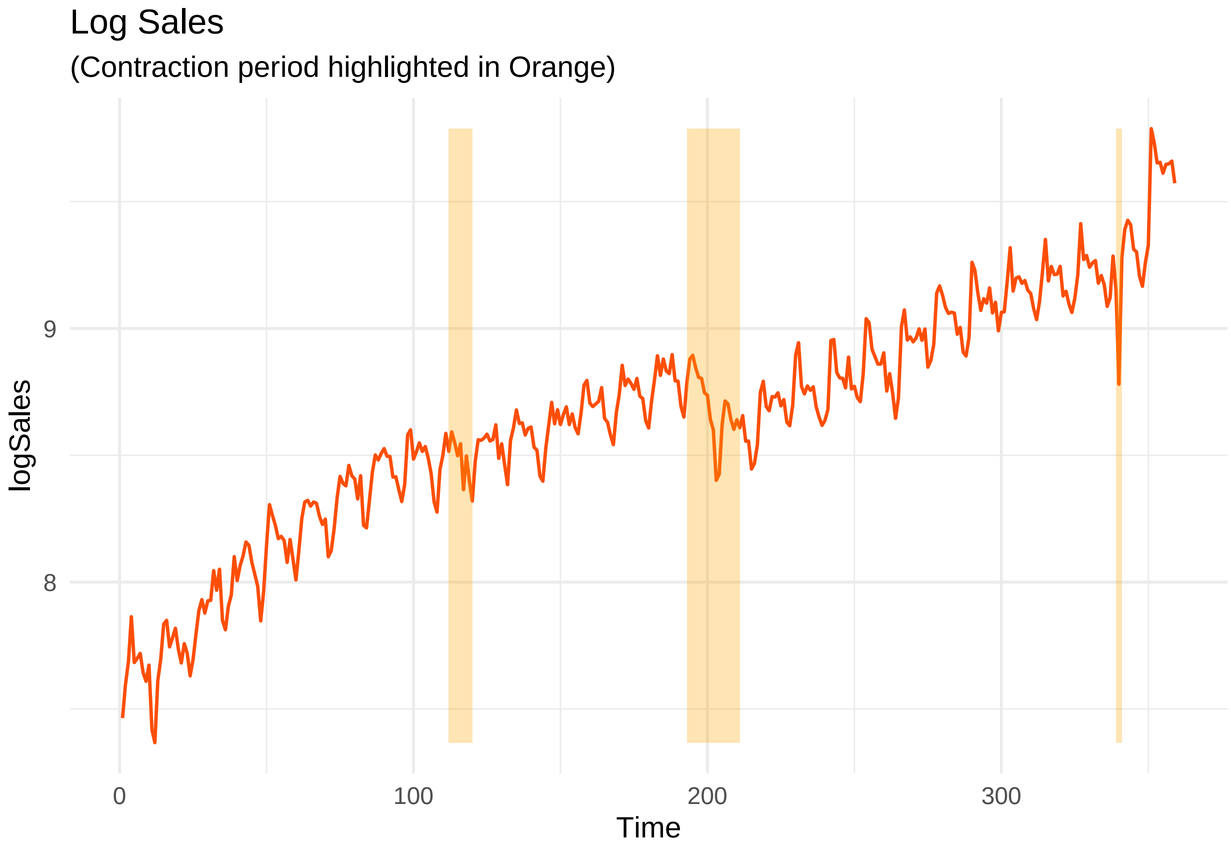
Multiplicative or additive: multiplicative model should be used as volatility increases over time.

```r
## Example data
set.seed(0)
dat <- sales

## Determine highlighted regions
v <- rep(0, max(dat$Time))
```

```
v[c(112:119, 193:210, 339:340)] <- 1

## Get the start and end points for highlighted regions
inds <- diff(c(0, v))
start <- dat$Time[inds == 1]
end <- dat$Time[inds == -1]
if (length(start) > length(end)) end <- c(end, tail(dat$Time, 1))

## highlight region data
rects <- data.frame(start=start, end=end, group=seq_along(start))

ggplot(data=dat, aes(Time, Sales)) +
theme_minimal() +
geom_line(color = "#00AFBB", size = .6) +
geom_rect(data=rects, inherit.aes=FALSE, aes(xmin=start, xmax=end, ymin=min(dat$Sales),
ymax=max(dat$Sales), group=group), color="transparent", fill="orange",
alpha=0.3) +
labs(title = "Used Car Sales", subtitle = "(Contraction period highlighted in Orange)")
```



```
ggplot(data=dat, aes(Time, logSales)) +
theme_minimal() +
geom_line(color = "#FC4E07", size = .6) +
geom_rect(data=rects, inherit.aes=FALSE, aes(xmin=start, xmax=end, ymin=min(dat$logSales),
ymax=max(dat$logSales), group=group), color="transparent", fill="orange", alpha=0.3) +
labs(title = "Log Sales", subtitle = "(Contraction period highlighted in Orange)")
```



2. Form a spectral plot of the log Sales series. The purpose is to gain information about the structure of the time series and give guidance in the formation of a multiplicative decomposition model in the next part. Mark important frequencies on the plot, and explain in detail what it reveals.

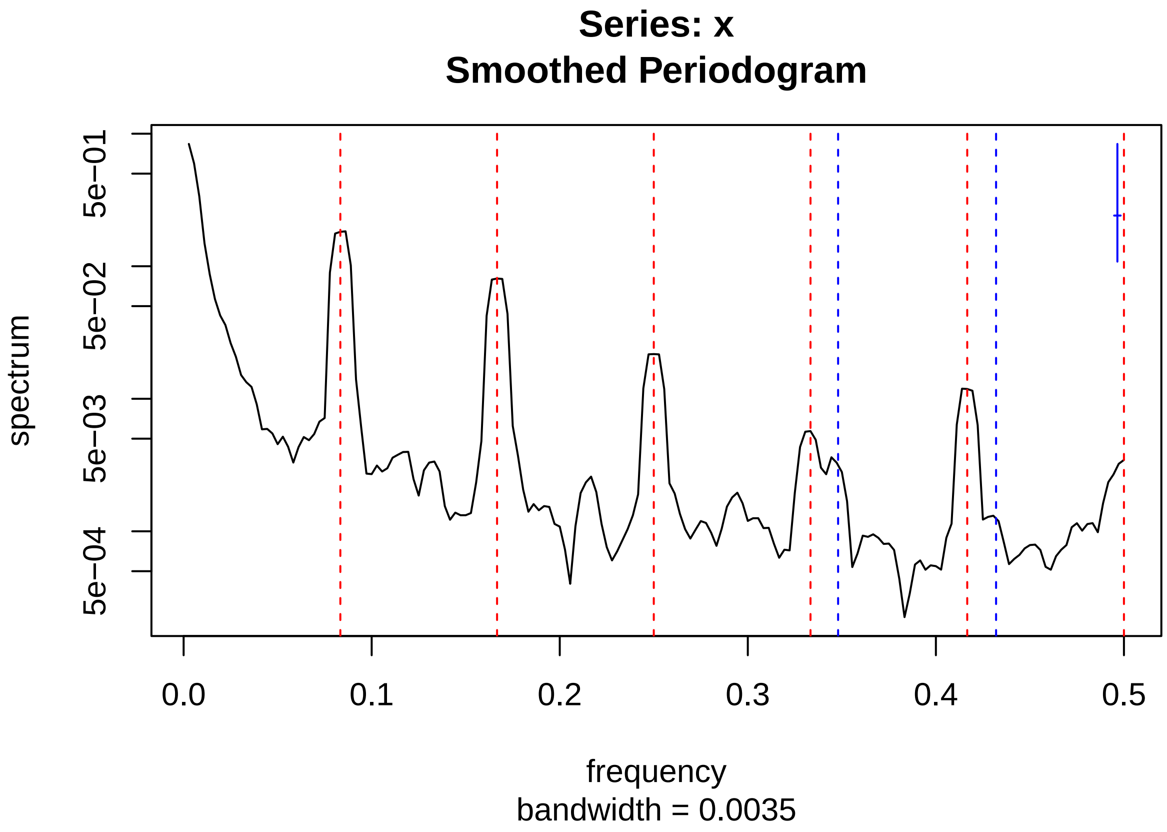
***ANSWER:**

Trend: There is peak at low frequency, indicating presence of trend.

Seasonal frequencies: There is evidence for strong seasonal component as indicated by peaks at the dashed red lines, which mark the seasonal frequencies 1/12, 2/12, 3/12, 4/12, 5/12, and 6/12.

Calendar pair: There is a peak at frequency 348, which indicates that the calendar pair may be significant.

```
logSales.ts <- ts(sales$logSales)
spectrum(logSales.ts, span=5)
abline(v=c(1/12,2/12,3/12,4/12,5/12,6/12),col="red",lty=2)
abline(v=c(0.348,0.432),col="blue",lty=2)
```



3. Using the data for the years 1992 to 2019, fit a multiplicative decomposition model to the Sales data. Include a polynomial trend and a static seasonal component using month dummies. If the spectral plot suggests calendar structure is present, include relevant trigonometric variable pairs. If you find an outlier value which warrants use of a dummy for adjustment, form the dummy and include it in your model (remember to add outlier dummies you create to the data frame). If a calendar pair or an outlier dummy you have included is not significant, remove and refit.

ANSWER: Final fit of the linear model will include time and month variables, as well as observation 203 and c348 and s348 calendar variables. The F-test for observation 203 yields a p-value of 0.005, and the calendar variables c348 and s348 yields the p-value of 0.045, both are small enough to indicate significance. We conclude from F-test below that c432 and S432 are not significant.

```
# filter year <= 2019
sales <- sales %>% filter(Year <= 2019 )
```

```
# add outlier obs203
obs203<-rep(0, max(sales$Time))
obs203[203] <- 1
sales<-data.frame(sales,obs203)
```

```
model1<-lm(logSales~Time+I(Time^2)+I(Time^3)+I(Time^4)+I(Time^5)+I(Time^6)+fMonth+obs203+c348+s348+c432+
```

Partial F-test to determine the significance of Trend variables

```
# excluding the time polynomials
model2<-lm(logSales~fMonth+obs203+c348+s348+c432+s432, data = sales)

anova(model2, model1)
```

```
## Analysis of Variance Table
##
## Model 1: logSales ~ fMonth + obs203 + c348 + s348 + c432 + s432
## Model 2: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + fMonth + obs203 + c348 + s348 + c432 + s432
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    319 58.418
## 2    313  1.243  6    57.175 2399.5 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Partial F-test to determine the significance of fMonth dummy variables

```
# excluding the fMonth dummy variable
model3<-lm(logSales~Time+I(Time^2)+I(Time^3)+I(Time^4)+I(Time^5)+I(Time^6)+obs203+c348+s348+
c432+s432, data = sales)

anova(model3, model1)
```

```
## Analysis of Variance Table
##
## Model 1: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + obs203 + c348 + s348 + c432 + s432
## Model 2: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + fMonth + obs203 + c348 + s348 + c432 + s432
##   Res.Df    RSS Df Sum of Sq     F    Pr(>F)
## 1    324 3.4614
## 2    313 1.2430 11    2.2183 50.78 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Partial F-test to determine the significance of both calendar pair variables

```
# excluding the calendar pairs
model4<-lm(logSales~Time+I(Time^2)+I(Time^3)+I(Time^4)+I(Time^5)+I(Time^6)+fMonth+obs203, data = sales)

anova(model4, model1)
```

```
## Analysis of Variance Table
##
## Model 1: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + fMonth + obs203
## Model 2: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + fMonth + obs203 + c348 + s348 + c432 + s432
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    317 1.2684
## 2    313 1.2430  4  0.025341 1.5952 0.1754
```

Partial F-test to determine the significance of 432 calendar pair variables

```
# excluding the 432 calendar pairs
model5<-lm(logSales~Time+I(Time^2)+I(Time^3)+I(Time^4)+I(Time^5)+I(Time^6)+fMonth+obs203+c348+s348, dat

anova(model5, model1)
```

```
## Analysis of Variance Table
##
## Model 1: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + fMonth + obs203 + c348 + s348
## Model 2: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + fMonth + obs203 + c348 + s348 + c432 + s432
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    315 1.2435
## 2    313 1.2430  2 0.0005009 0.0631 0.9389
```

Partial F-test to determine the significance of 348 calendar pair variables

```
# excluding the 348 calendar pairs
model6<-lm(logSales~Time+I(Time^2)+I(Time^3)+I(Time^4)+I(Time^5)+I(Time^6)+fMonth+obs203+c432+s432, dat

anova(model6, model1)
```

```
## Analysis of Variance Table
##
## Model 1: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + fMonth + obs203 + c432 + s432
## Model 2: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + fMonth + obs203 + c348 + s348 + c432 + s432
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1    315 1.2679
## 2    313 1.2430  2  0.024878 3.1322 0.04499 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Partial F-test to determine the significance of observation 203

```
# excluding the 348 calendar pairs
model7<-lm(logSales~Time+I(Time^2)+I(Time^3)+I(Time^4)+I(Time^5)+I(Time^6)+fMonth+c348+s348+c432+s432, d

anova(model7, model1)
```

```
## Analysis of Variance Table
##
## Model 1: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + fMonth + c348 + s348 + c432 + s432
## Model 2: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + fMonth + obs203 + c348 + s348 + c432 + s432
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1    314 1.2739
## 2    313 1.2430  1  0.030884 7.7766 0.005617 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

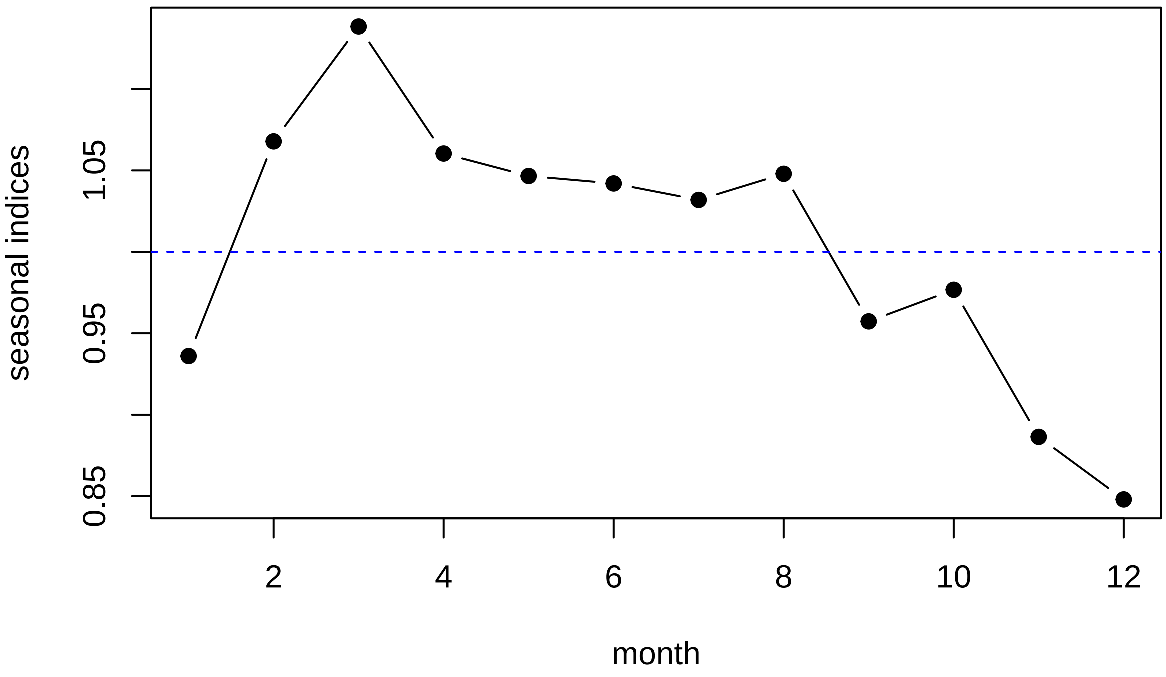
```
# final model
model8<-lm(logSales~Time+I(Time^2)+I(Time^3)+I(Time^4)+I(Time^5)+I(Time^6)+fMonth+obs203+c348+s348, dat
```

(a) Construct estimates of the static seasonal indices, and display them in a table and a plot. Describe the indices. Explain why the seasonal peaks and valleys occur in the months indicated by the estimates.

ANSWER: Sales drop in the winter months and peak in Spring and Summer months. The cold weather could be the cause of the drop - consumers are less likely to go out in the winter months to shop for cars. And tax refunds may influence the jump in the spring and summer sales. A small peak in August may be due to back-to-school sales.

```
b1<-coef(model8)[1]
b2<-coef(model8)[8:18]+b1
```

```
b3<-c(b1,b2)
seas<-b3-mean(b3)

seas.ts<-ts(exp(seas))
plot(seas.ts,ylab="seasonal indices",xlab="month", type="b", pch=19)
abline(h=1, col="blue", lwd=1, lty=2)
```



```
month <- seq(12)
seas_indices <- exp(seas)
seas_df <- data.frame(month, seas, seas_indices)
print.data.frame(tbl_df(seas_df))
```

```
## Warning: `tbl_df()` was deprecated in dplyr 1.0.0.
## Please use `tibble::as_tibble()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
```

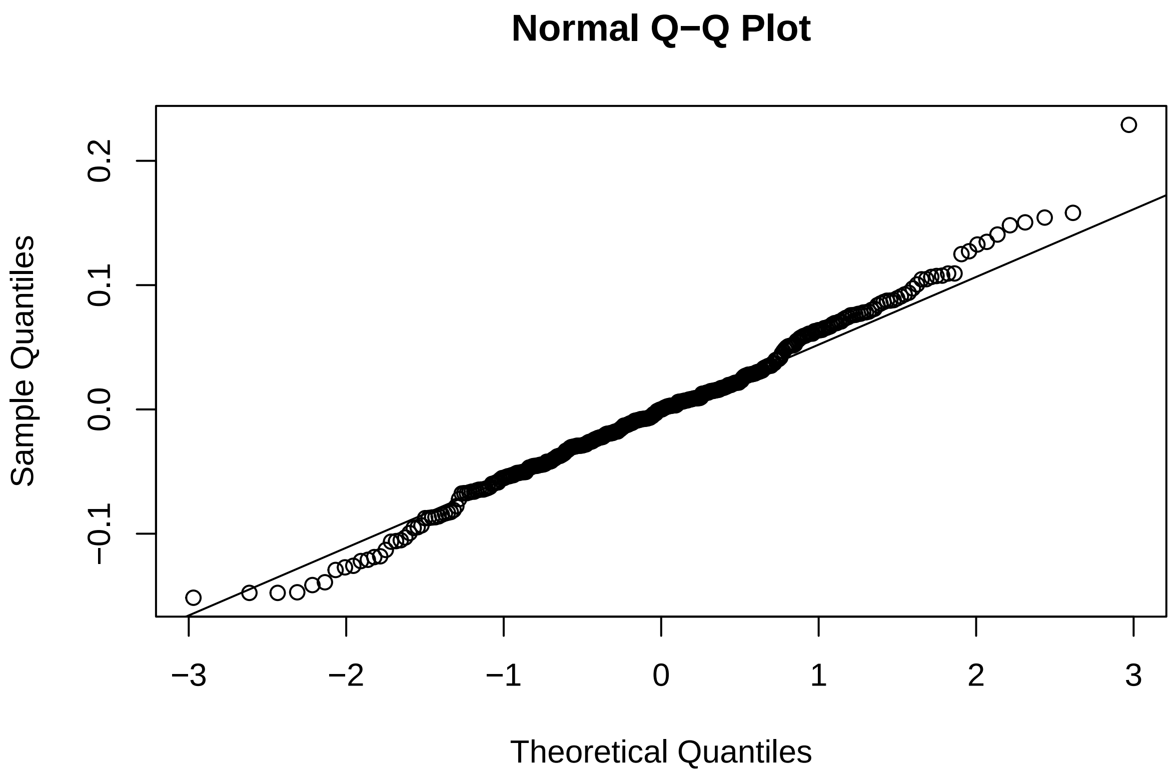
```
##    month        seas seas_indices
## 1      1 -0.06610116    0.9360362
## 2      2  0.06563360    1.0678354
## 3      3  0.12957400    1.1383434
## 4      4  0.05862015    1.0603724
## 5      5  0.04553264    1.0465852
## 6      6  0.04114144    1.0419995
## 7      7  0.03140169    1.0318999
## 8      8  0.04681812    1.0479314
## 9      9 -0.04362525    0.9573126
## 10    10 -0.02355813    0.9767172
## 11    11 -0.12057182    0.8864134
## 12    12 -0.16486528    0.8480079
```

(b) Perform a residual analysis for the fitted model. Include a normal quantile plot of the residuals and the Shapiro–Wilk test, a plot of the residuals vs. time, a residual acf plot, and a residual spectral plot. Interpret the results of each. Is there reduction to white noise by the model?

```
res1 <- resid(model8)
```

(i) QQ Plot

```
qqnorm(res1)
qqline(res1)
```



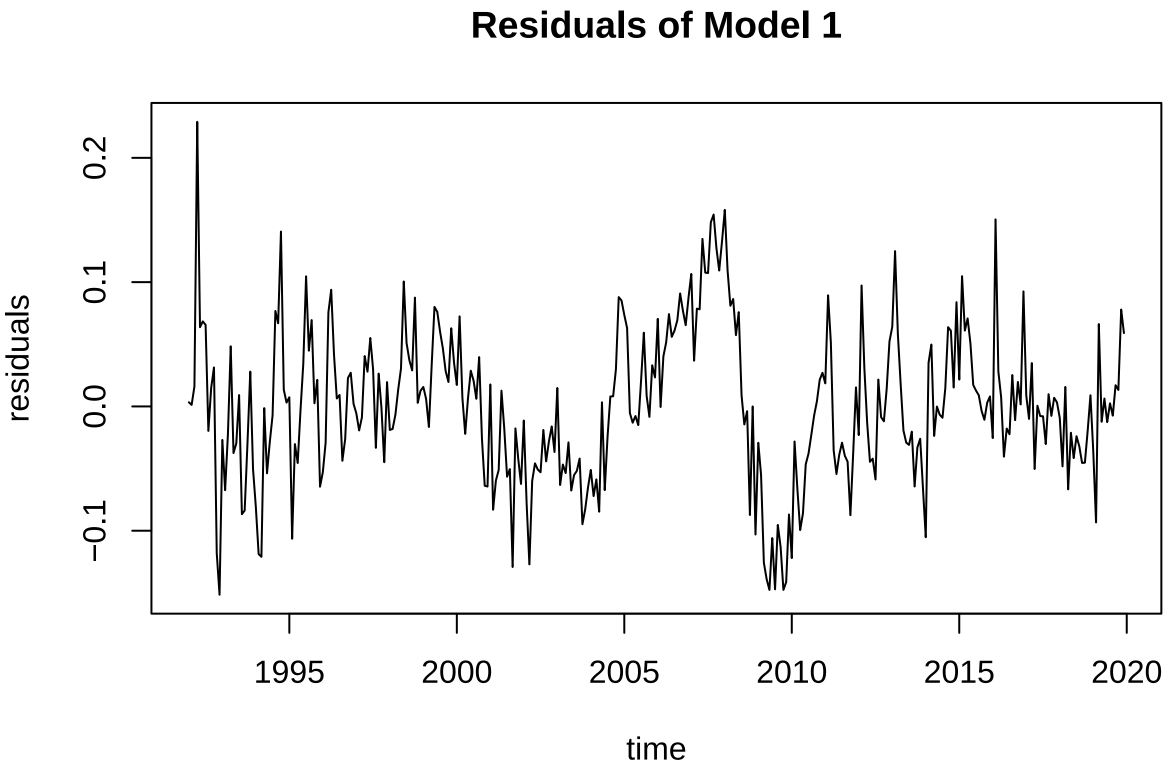
(ii) Shapiro-Wilk Test

```
shapiro.test(res1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res1
## W = 0.99286, p-value = 0.1097
```

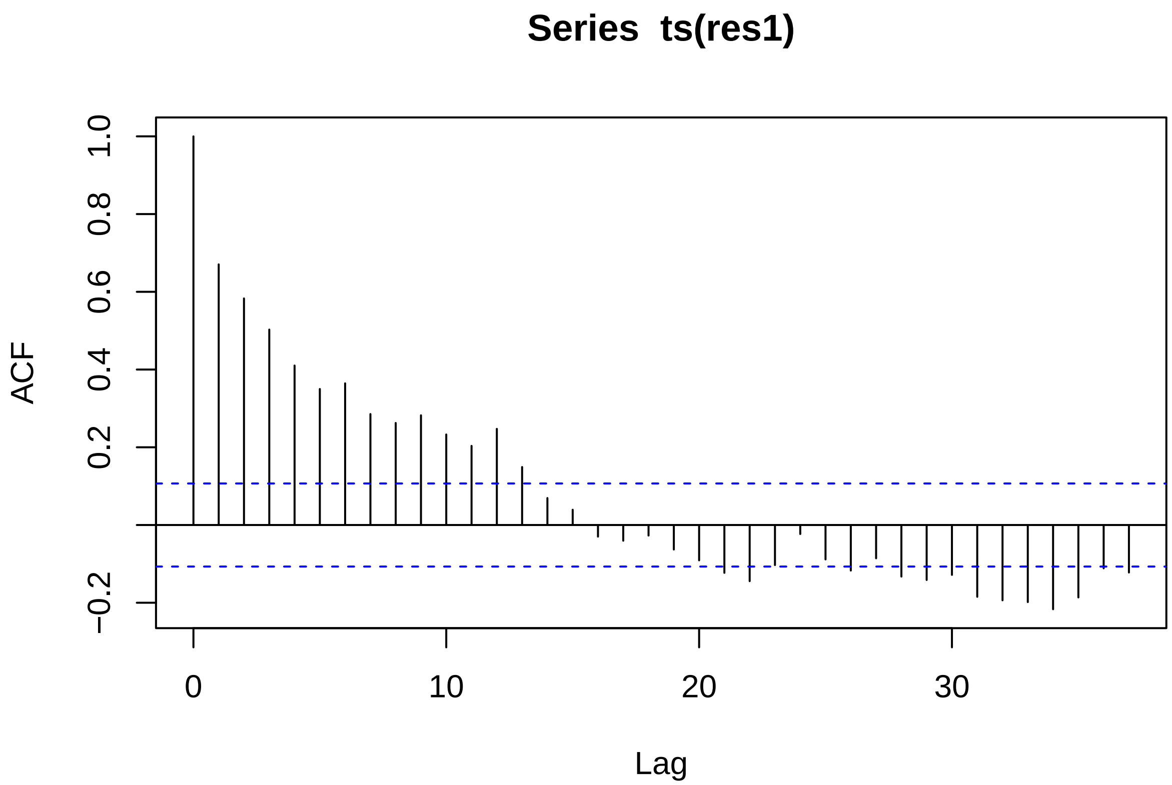
(iii) Residuals Plot

```
resid1 <- ts(res1,start=c(1992,1),freq=12)
plot(resid1, xlab="time",ylab="residuals",main="Residuals of Model 1")
```



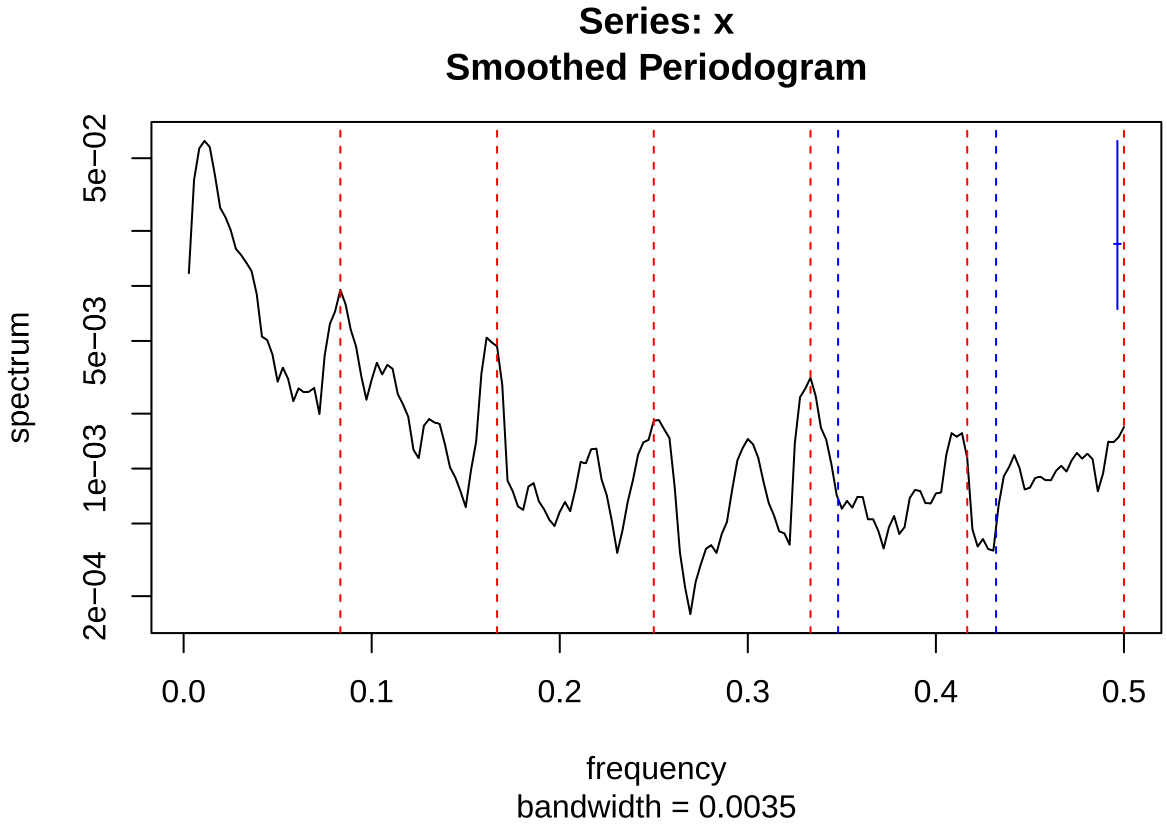
(iv) ACF Plot

```
acf(ts(res1), 37)
```



(v) Residuals Spectral Plot

```
spectrum(res1, span=5)
abline(v=c(1/12,2/12,3/12,4/12,5/12,6/12),col="red",lty=2)
abline(v=c(0.348,0.432),col="blue",lty=2)
```



(vi)Interpretation

- QQPLOT: The QQPLOT doesn't support rejecting the null hypothesis that the residuals are normally distributed. Most of the quantiles (observations) fall on the qqline.
- Shapiro-Wilk Normality Test: The p-value is 0.1097, which is not small enough to reject the null hypothesis that the residuals are normally distributed (given a 95% confidence level, the p-value has to fall below 0.05 to conclude that the residuals are not normal).
- Autocorrelation: The autocorrelation function plot indicates strong correlation between the residuals at lags 1-13 and 31-35, which indicates that the model does not adequately capture trend. We can conclude from the prominence at lags 6, 12, and 30 that there is dynamic seasonality.
- Residual Analysis: The residuals is not flat and rather volatile. show that the model does not track autocorrelation structure and the trend structure adequately. If a model captures trend, the plot of residuals should be relatively flat. The residuals also seem to be volatile, which indicate that there maybe autocorrelation structure which the model does not capture.
- Spectral plot: Spectral plot is not flat - the length of the highest and lowest points is twice the upperhalf of the blue line. The peak at low frequency reveals remaining trend structure. The peaks at the dashed red lines indicate that there remains some uncaptured seasonality in the model. There is little prominence in the 348 and 432 calendar frequencies, which suggests that our model adequately captures changes in sales due to calendar structure.

4. Visual inspection of the plots in part 1 suggests the seasonal structure is dynamic. To explore this, add the following variables to your model in part 3: Dynamic and Dynamic*fMonth. This addition will produce separate static seasonal estimates for the time spans 1992 to 2010, and 2011 to 2019. For consideration of this methodology, review the analysis of the variety stores sales data in the 7 February notes, where wtgrant corresponds to Dynamic, and Time corresponds to fMonth. There the purpose is

to estimate the change in trend, and here the purpose is to estimate dynamic seasonal structure.

(a) Is the addition of the new variables a statistically significant step? Answer with an appropriate test result.

ANSWER: Yes the addition seems to be a statistically significant step. The F-test below shows that the p-Value is 0.017, which is below the 5% threshold.

```
# Time 228 below represent the last month in 2010 (Dec 2010)
Dynamic <- c(rep(0, 228), rep(1, nrow(sales) - 228))
sales <- data.frame(sales, Dynamic)
```

```
sales <-transform(sales,DynamicfMonth=ifelse(Dynamic == 1, fMonth, 0))
sales$DynamicfMonth <- as.factor(sales$DynamicfMonth)
```

```
model9<-lm(logSales~Time+I(Time^2)+I(Time^3)+I(Time^4)+I(Time^5)+I(Time^6)+fMonth+obs203+c348+s348+Dynam

anova(model9, model8)
```

```
## Analysis of Variance Table
##
## Model 1: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + fMonth + obs203 + c348 + s348 + Dynamic + Dynamic *
##     fMonth
## Model 2: logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + I(Time^5) +
##     I(Time^6) + fMonth + obs203 + c348 + s348
##   Res.Df    RSS  Df Sum of Sq     F  Pr(>F)
## 1    303 1.1482
## 2    315 1.2435 -12 -0.095315 2.096 0.01707 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(model9)
```

```
##
## Call:
## lm(formula = logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) +
##     I(Time^5) + I(Time^6) + fMonth + obs203 + c348 + s348 + Dynamic +
##     Dynamic * fMonth, data = sales)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.147002 -0.035361  0.001323  0.031436  0.225152
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         7.434e+00  2.777e-02 267.726  < 2e-16 ***
## Time                1.761e-02  2.079e-03   8.468 1.09e-15 ***
## I(Time^2)          -1.768e-04  5.465e-05  -3.235 0.001351 **
## I(Time^3)           1.812e-06  6.167e-07   2.938 0.003553 **
## I(Time^4)          -1.154e-08  3.346e-09  -3.449 0.000642 ***
## I(Time^5)           3.501e-11  8.600e-12   4.071 5.97e-05 ***
## I(Time^6)          -3.882e-14  8.415e-15  -4.613 5.87e-06 ***
## fMonth2             1.034e-01  1.999e-02   5.174 4.17e-07 ***
## fMonth3             1.702e-01  1.999e-02   8.516 7.84e-16 ***
## fMonth4             1.258e-01  1.998e-02   6.299 1.05e-09 ***
## fMonth5             1.136e-01  2.000e-02   5.680 3.17e-08 ***
## fMonth6           1.122e-01  1.999e-02   5.613 4.49e-08 ***
## fMonth7           1.020e-01  1.999e-02   5.105 5.86e-07 ***
## fMonth8           1.117e-01  2.001e-02   5.582 5.29e-08 ***
## fMonth9           2.281e-02  2.000e-02   1.140 0.255071    
## fMonth10          4.385e-02  2.000e-02   2.192 0.029138 *  
## fMonth11         -6.095e-02  2.030e-02  -3.003 0.002895 ** 
## fMonth12         -1.163e-01  2.002e-02  -5.811 1.58e-08 ***
## obs203           -1.743e-01  6.387e-02  -2.729 0.006728 ** 
## c348             -9.102e-03  4.799e-03  -1.896 0.058849 .  
## s348              6.584e-03  4.808e-03   1.369 0.171876    
## Dynamic           1.302e-02  3.277e-02   0.397 0.691441    
## fMonth2:Dynamic   8.822e-02  3.532e-02   2.498 0.013014 *  
## fMonth3:Dynamic   7.970e-02  3.531e-02   2.257 0.024699 *  
## fMonth4:Dynamic  -2.887e-03  3.524e-02  -0.082 0.934754    
## fMonth5:Dynamic  -5.234e-03  3.534e-02  -0.148 0.882369    
## fMonth6:Dynamic  -1.441e-02  3.531e-02  -0.408 0.683496    
## fMonth7:Dynamic  -1.286e-02  3.527e-02  -0.365 0.715568    
## fMonth8:Dynamic   5.324e-03  3.538e-02   0.151 0.880466    
## fMonth9:Dynamic   7.408e-04  3.533e-02   0.021 0.983283    
## fMonth10:Dynamic -2.131e-03  3.532e-02  -0.060 0.951924    
## fMonth11:Dynamic  2.148e-02  3.559e-02   0.604 0.546578    
## fMonth12:Dynamic  5.706e-02  3.537e-02   1.613 0.107778    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.06156 on 303 degrees of freedom
## Multiple R-squared:  0.9809, Adjusted R-squared:  0.9789 
## F-statistic: 487.1 on 32 and 303 DF,  p-value: < 2.2e-16
```

(b) Construct two sets of static seasonal index estimates, for the two time spans. Code to do this is in the following illustration. Assume the output of the model is such that, line 1 is for the intercept, lines 8 to 18 are for fMonth, line 22 is for Dynamic, and lines 23 to 33 are for Dynamic*fMonth.

```
# fit the model
model<-lm(logSales~Time+I(Time^2)+I(Time^3)+I(Time^4)+I(Time^5)+I(Time^6)+fMonth+obs203+c348+s348+Dynami

#For the first time span:
b1<-coef(model)[1]
b2<-coef(model)[8:18]+b1
b3<-c(b1,b2)
seas1<-exp(b3-mean(b3))

#For the second time span:
b1<-coef(model)[1]+coef(model)[22]
b2<-coef(model)[8:18]+coef(model)[23:33]+b1
b3<-c(b1,b2)
seas2<-exp(b3-mean(b3))

seas1
```

```
## (Intercept)     fMonth2     fMonth3     fMonth4     fMonth5     fMonth6 
##   0.9411077   1.0436511   1.1157220   1.0673107   1.0542906   1.0528758 
##     fMonth7     fMonth8     fMonth9    fMonth10    fMonth11    fMonth12 
##   1.0422042   1.0523024   0.9628198   0.9832919   0.8854602   0.8377821
```

```
seas2
```

```
## (Intercept)     fMonth2     fMonth3     fMonth4     fMonth5     fMonth6 
##   0.9243964   1.1196691   1.1868240   1.0453360   1.0301640   1.0193852 
##     fMonth7     fMonth8     fMonth9    fMonth10    fMonth11    fMonth12 
##   1.0106141   1.0391347   0.9464239   0.9637754   0.8886217   0.8712233
```

```
summary(model)
```

```
## 
## Call:
## lm(formula = logSales ~ Time + I(Time^2) + I(Time^3) + I(Time^4) + 
##     I(Time^5) + I(Time^6) + fMonth + obs203 + c348 + s348 + Dynamic + 
##     Dynamic * fMonth, data = sales)
## 
## Residuals:
##       Min        1Q    Median        3Q       Max 
## -0.147002 -0.035361  0.001323  0.031436  0.225152 
## 
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)    
## (Intercept)       7.434e+00  2.777e-02 267.726  < 2e-16 ***
## Time              1.761e-02  2.079e-03   8.468 1.09e-15 ***
## I(Time^2)        -1.768e-04  5.465e-05  -3.235 0.001351 ** 
## I(Time^3)         1.812e-06  6.167e-07   2.938 0.003553 ** 
## I(Time^4)        -1.154e-08  3.346e-09  -3.449 0.000642 ***
## I(Time^5)         3.501e-11  8.600e-12   4.071 5.97e-05 ***
## I(Time^6)        -3.882e-14  8.415e-15  -4.613 5.87e-06 ***
## fMonth2           1.034e-01  1.999e-02   5.174 4.17e-07 ***
## fMonth3           1.702e-01  1.999e-02   8.516 7.84e-16 ***
## fMonth4           1.258e-01  1.998e-02   6.299 1.05e-09 ***
## fMonth5           1.136e-01  2.000e-02   5.680 3.17e-08 ***
## fMonth6           1.122e-01  1.999e-02   5.613 4.49e-08 ***
## fMonth7           1.020e-01  1.999e-02   5.105 5.86e-07 ***
## fMonth8           1.117e-01  2.001e-02   5.582 5.29e-08 ***
## fMonth9           2.281e-02  2.000e-02   1.140 0.255071    
## fMonth10          4.385e-02  2.000e-02   2.192 0.029138 *  
## fMonth11         -6.095e-02  2.030e-02  -3.003 0.002895 ** 
## fMonth12         -1.163e-01  2.002e-02  -5.811 1.58e-08 ***
## obs203           -1.743e-01  6.387e-02  -2.729 0.006728 ** 
## c348             -9.102e-03  4.799e-03  -1.896 0.058849 .  
## s348              6.584e-03  4.808e-03   1.369 0.171876    
## Dynamic           1.302e-02  3.277e-02   0.397 0.691441    
## fMonth2:Dynamic   8.822e-02  3.532e-02   2.498 0.013014 *  
## fMonth3:Dynamic   7.970e-02  3.531e-02   2.257 0.024699 *  
## fMonth4:Dynamic  -2.887e-03  3.524e-02  -0.082 0.934754    
## fMonth5:Dynamic  -5.234e-03  3.534e-02  -0.148 0.882369    
## fMonth6:Dynamic  -1.441e-02  3.531e-02  -0.408 0.683496    
## fMonth7:Dynamic  -1.286e-02  3.527e-02  -0.365 0.715568    
## fMonth8:Dynamic   5.324e-03  3.538e-02   0.151 0.880466    
## fMonth9:Dynamic   7.408e-04  3.533e-02   0.021 0.983283    
## fMonth10:Dynamic -2.131e-03  3.532e-02  -0.060 0.951924    
## fMonth11:Dynamic  2.148e-02  3.559e-02   0.604 0.546578    
## fMonth12:Dynamic  5.706e-02  3.537e-02   1.613 0.107778    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06156 on 303 degrees of freedom
## Multiple R-squared:  0.9809, Adjusted R-squared:  0.9789
## F-statistic: 487.1 on 32 and 303 DF,  p-value: < 2.2e-16
```

Tabulate and plot these two sets of estimates, and also the estimates in part 3. Describe them and discuss the differences between the two time spans. Why do you think the indicated changes have occurred?

ANSWER: The 1992-2010 indices differ significantly from 2011-2019 indices. Change in sales is more volatile in the latter period. The latter indices are higher in February and March, meaning that from 2011-2019 change in purchases in the two months are higher than in previous years. The indices then drop sharply and falls below the 1992-2010 indices beginning in April and into the summer and fall months.

I think that the seasonality change after the 2009 recession is due to the change in availability of electronic tax-filing, which speeds up tax returns and shifts buying pattern earlier in the year. Which explains why march has higher indices than in previous year.

(i) Table Comparison

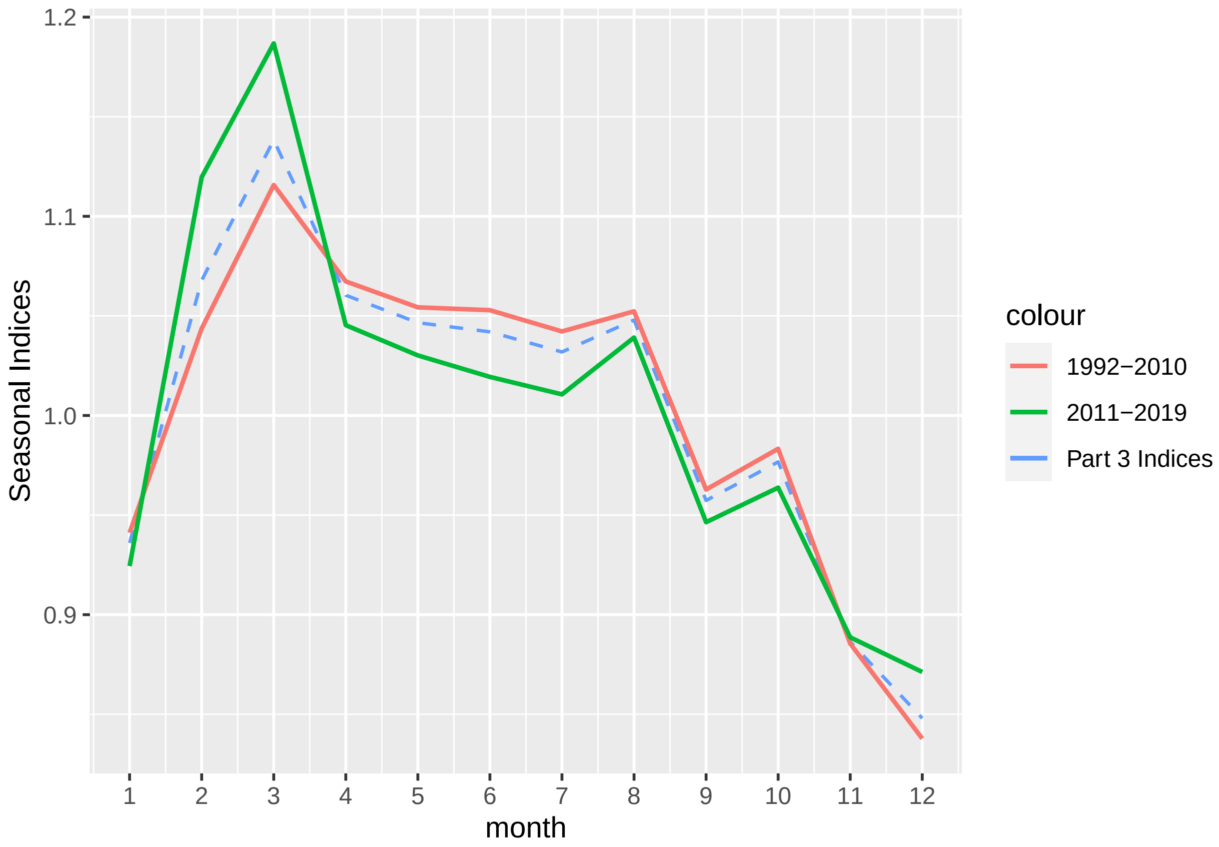
```
month <- seq(12)
seas_ind_part_3 <- seas.ts
seas_ind_1992_2010 <- seas1
seas_ind_2011_2019 <- seas2
seas_df1 <- data.frame(month, seas_ind_part_3, seas_ind_1992_2010, seas_ind_2011_2019)
print.data.frame(seas_df1)
```

```
##             month seas_ind_part_3 seas_ind_1992_2010 seas_ind_2011_2019
## (Intercept)     1       0.9360362          0.9411077          0.9243964
## fMonth2         2       1.0678354          1.0436511          1.1196691
## fMonth3         3       1.1383434          1.1157220          1.1868240
## fMonth4         4       1.0603724          1.0673107          1.0453360
## fMonth5         5       1.0465852          1.0542906          1.0301640
## fMonth6         6       1.0419995          1.0528758          1.0193852
## fMonth7         7       1.0318999          1.0422042          1.0106141
## fMonth8         8       1.0479314          1.0523024          1.0391347
## fMonth9         9       0.9573126          0.9628198          0.9464239
## fMonth10       10       0.9767172          0.9832919          0.9637754
## fMonth11       11       0.8864134          0.8854602          0.8886217
## fMonth12       12       0.8480079          0.8377821          0.8712233
```

(ii) Plot Comparison

```
ggplot(seas_df1, aes(x = month)) +
  geom_line(aes(y = seas_ind_part_3, color = 'Part 3 Indices'), size =.6, , linetype="dashed") +
  geom_line(aes(y = seas_ind_1992_2010, color = '1992-2010'), size = .8) +
  geom_line(aes(y = seas_ind_2011_2019, color = '2011-2019'), size =.8) +
  scale_x_continuous(breaks=1:12) +
  ylab("Seasonal Indices")
```

```
## Don't know how to automatically pick scale for object of type ts. Defaulting to continuous.
```

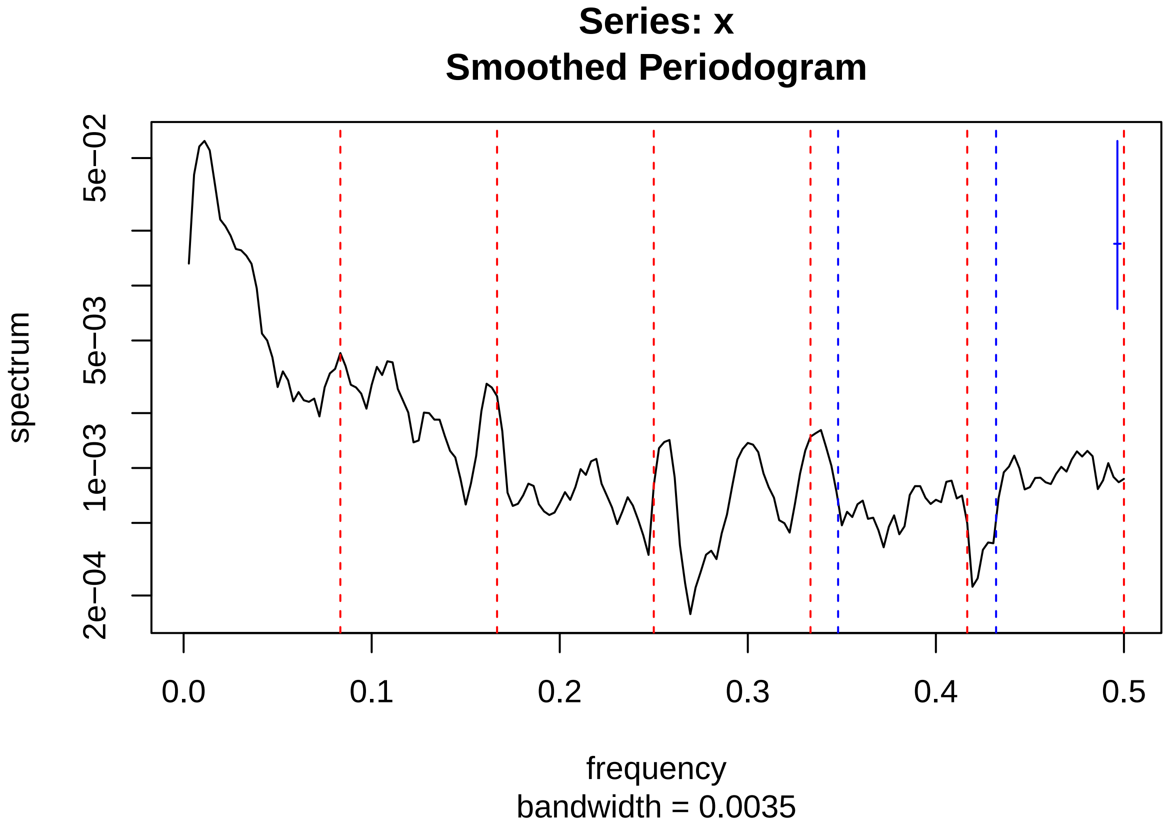


(c) Perform a residual analysis of the model as in part 3(b).

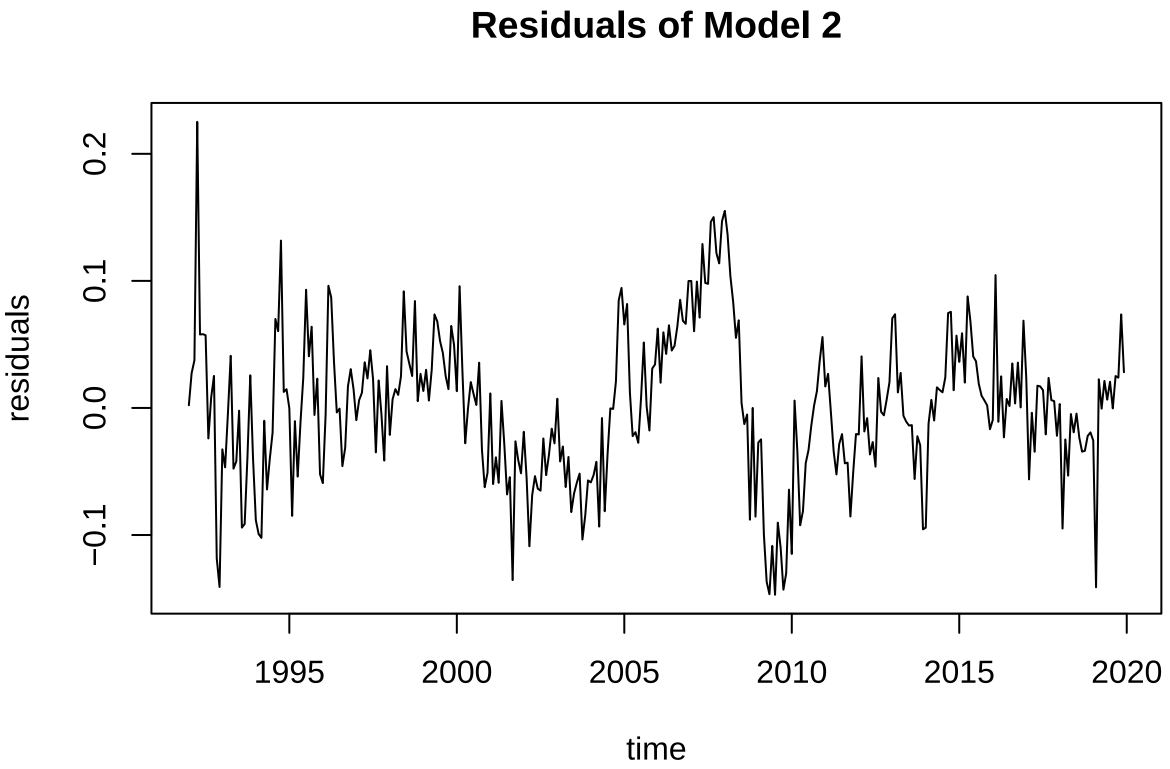
ANSWER: The addition of the dynamic and dynamic fmonth variables have helped in better capturing the seasonality component of our sales data, as indicated by the residual spectral plot below, which shows less prominence in frequencies 1/12, 2/12, 3/12, 4/12, 5/12, and 6/12. The spectral plot is still not flat enough to conclude that the residuals have been reduced to white noise, as indicated by the length of the highest and lowest points which is twice the length of the upper half of the blue line.

There is still a high peak at the beginning of the residual spectral plot, which indicates uncaptured trend. Autocorrelation plot still indicates significant autocorrelation as we have not introduced variables to address autocorrelation.

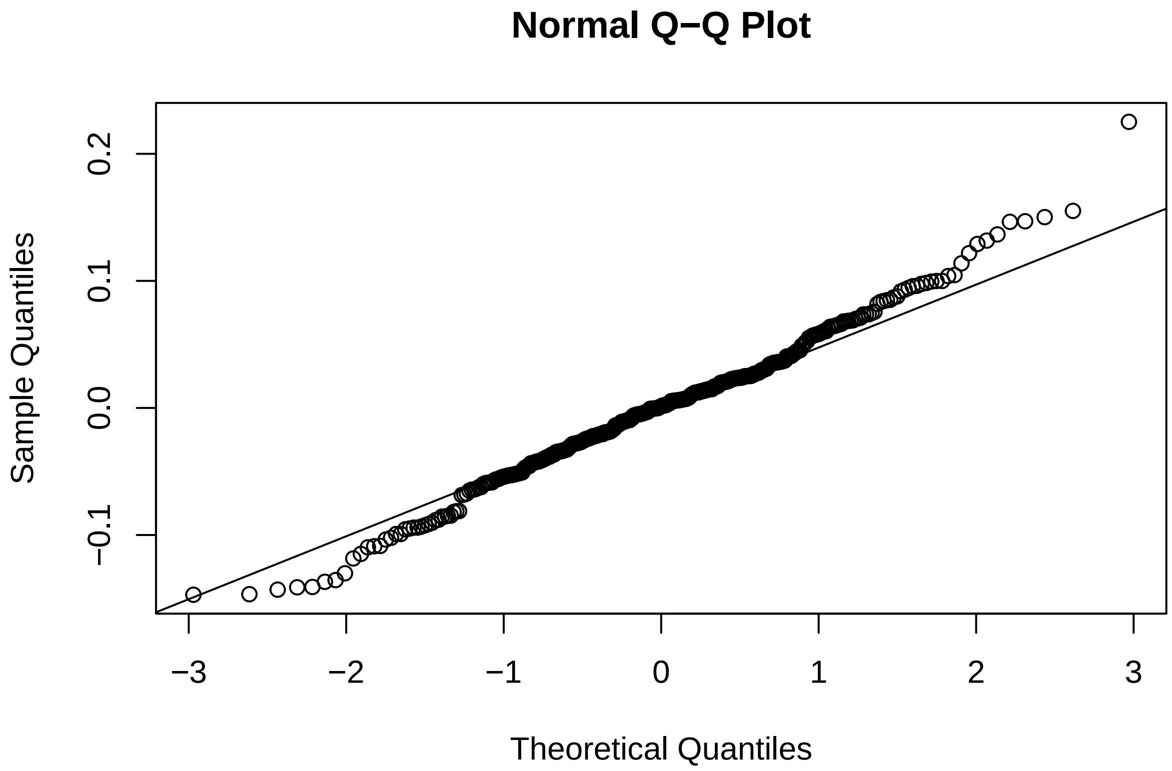
```
res2 <- resid(model)
spectrum(res2, span=5)
abline(v=c(1/12,2/12,3/12,4/12,5/12,6/12),col="red",lty=2)
abline(v=c(0.348,0.432),col="blue",lty=2)
```



```
resid2 <- ts(res2,start=c(1992,1),freq=12)
plot(resid2, xlab="time",ylab="residuals",main="Residuals of Model 2")
```



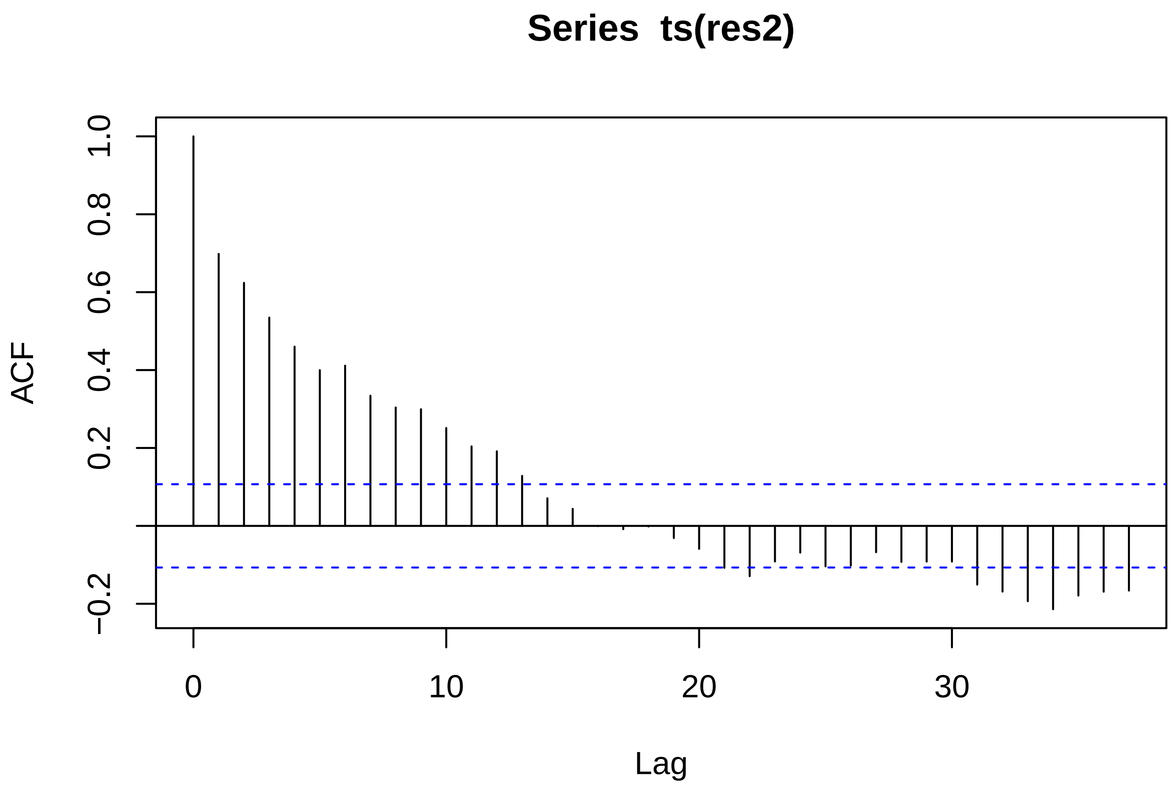
```
qqnorm(res2)
qqline(res2)
```



```
shapiro.test(res2)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res2
## W = 0.99169, p-value = 0.0565
```

```
acf(ts(res2), 37)
```



5. Construct the lag 1 and lag 2 residuals for the model in part 4, and add them to the model. Now perform a residual analysis of this new model and discuss the results. [Be sure to add the two lagged residuals variables to the data frame.]

ANSWER: The residual spectral plot is now flat - the addition of the two lag variables sufficiently captured the remaining autocorrelation structure of the previous model. There is no peak at 0 frequency, which indicates that the trend has also been sufficiently captured. We can conclude that the residuals of this final model has been reduced to white noise.

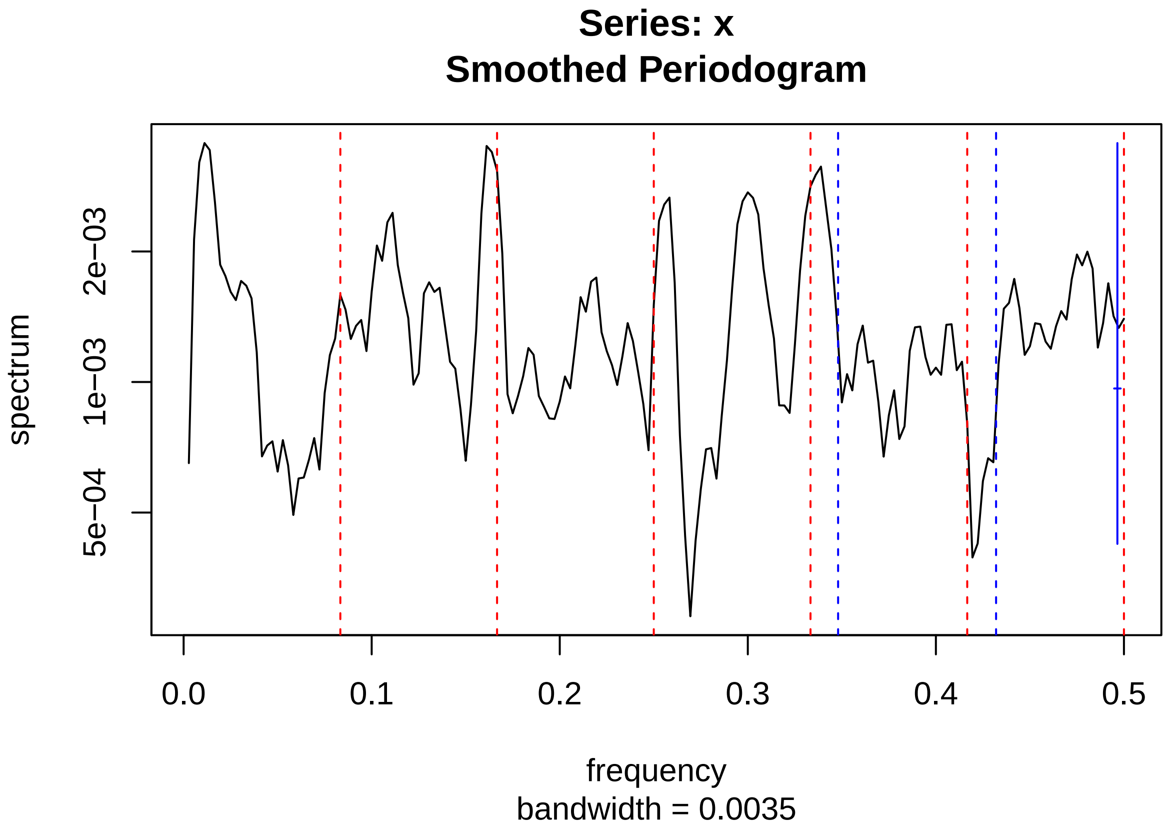
A look into the autocorrelation plots we see prominence at lag 6, which stands above the blue line. This indicates that there remains uncaptured trend yet not to the extent that we cannot conclude that the model has not been mostly reduced to white noise.

```
num_rows <- nrow(sales)
lresid<-c(rep(NA,num_rows))
lag1resid<-lresid
lag2resid<-lresid

lag1resid[2]<-resid(model)[1]

for(i in 3:num_rows){
i1<-i-1
i2<-i-2

lag1resid[i]<-resid(model)[i1]
lag2resid[i]<-resid(model)[i2]
}
```

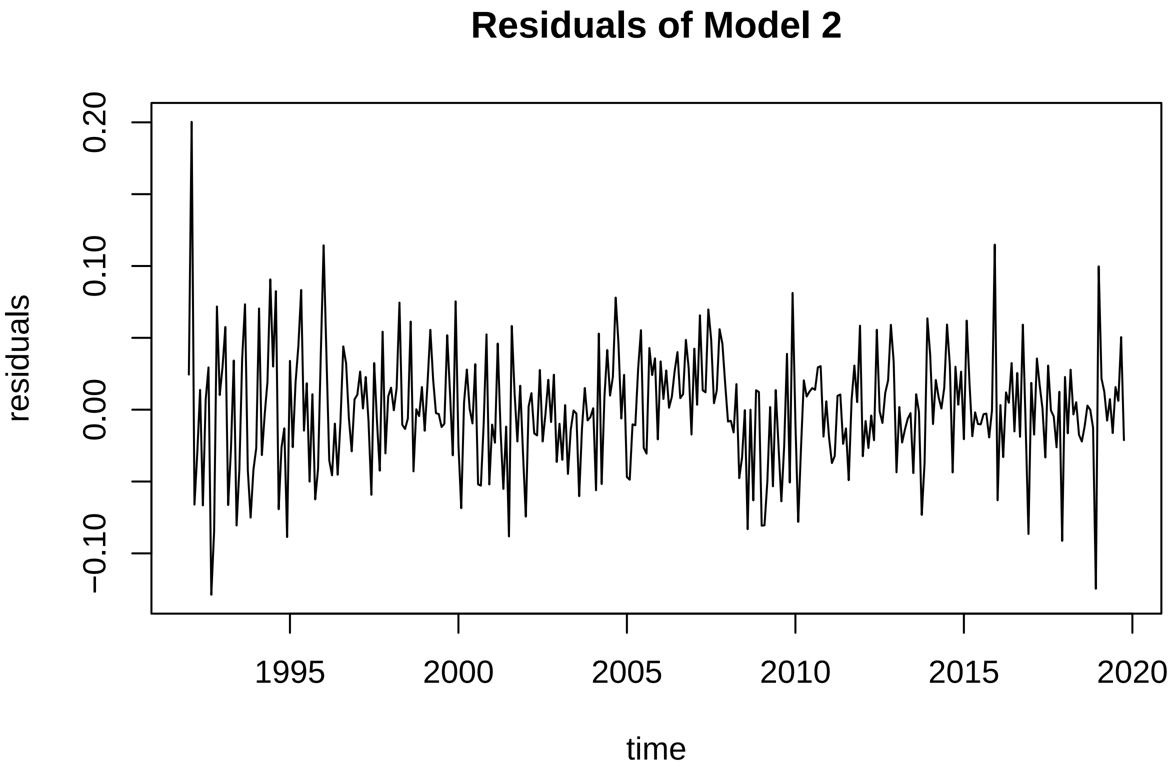
```
sales<-data.frame(sales,lag1resid,lag2resid)
```

```
model10 <-lm(logSales~Time+I(Time^2)+I(Time^3)+I(Time^4)+I(Time^5)+I(Time^6)+fMonth+obs203+c348+s348+Dyr
```

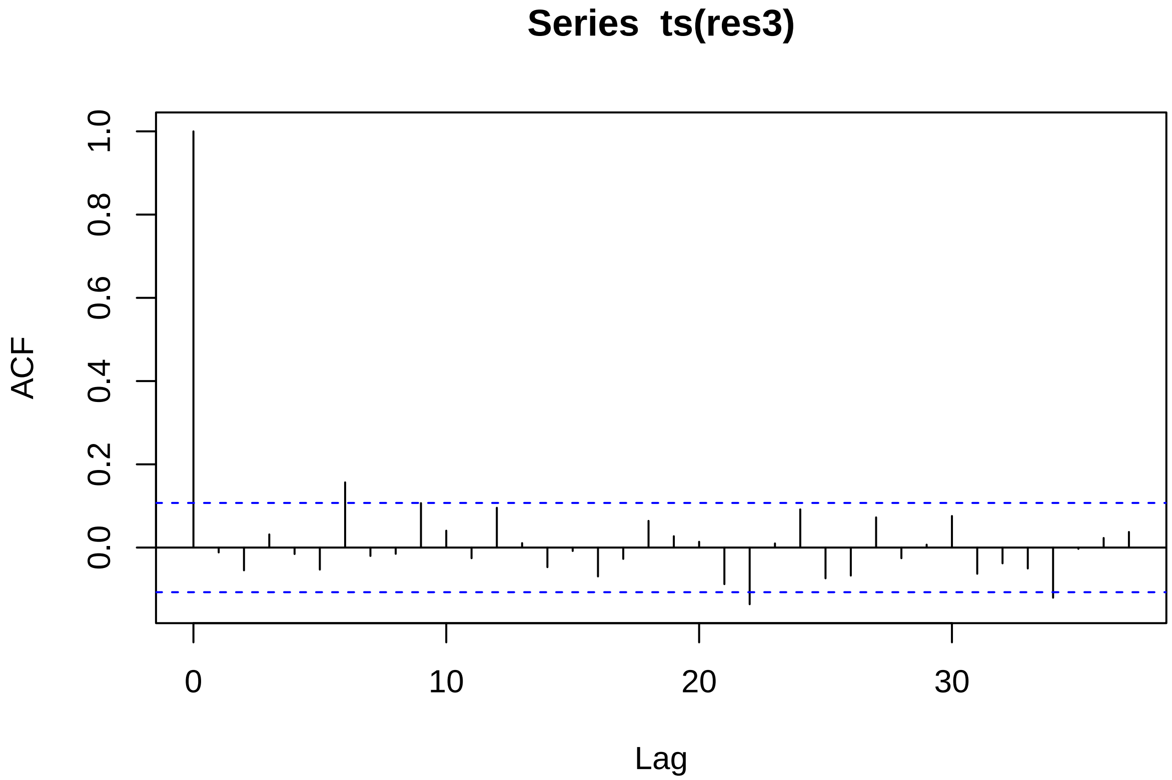
```
res3 <- resid(model10)
spectrum(res3, span=5)
abline(v=c(1/12,2/12,3/12,4/12,5/12,6/12),col="red",lty=2)
abline(v=c(0.348,0.432),col="blue",lty=2)
```



```
resid3 <- ts(res3,start=c(1992,1),freq=12)
plot(resid3, xlab="time",ylab="residuals",main="Residuals of Model 2")
```



```
acf(ts(res3), 37)
```



6. Write a summary of the results obtained from parts 3 through 5. In your discussion discuss what the analysis has revealed about U.S. used car sales.

Part 3 The final model that we selected in part three includes the trend component, fMonth variables, one outlier dummy variable, and calendar pairs from frequency 348. We excluded the 432 calendar pair variables as they are not significant based on our F-test.

Using this model, we observed from the residual spectral plot that there remains uncaptured trend and seasonality components. We identify this by looking at the peak at low frequency (Which indicates uncaptured trend) and the peaks at the seasonal frequencies (which indicates seasonality). We concluded that the model has not reduced the residuals into white noise as the length between highest and lowest peaks of the spectral density plot is twice the upper half of the blue measuring line at the top right corner of the plot.

Part 4 We added the Dynamic and Dynamic fMonth variables and concluded that they are statistically significant to our model. This means that there is dynamic seasonality present in used car sales in between 1992 and 2019.

We compared the indices from years 1992-2010 to the years 2011-2019 and confirmed dynamic seasonality by looking at the indices comparison table and plot. The volatility in the percentage change of sales throughout the year is higher in 2011-2019. Specifically, used car sales increase more sharply in earlier months in the years in 2011-2019 than in 1992-2010.

Residual Analysis of the new model shows that the dynamic variables helped in capturing seasonality, as evidenced by less prominent peaks at the residual spectral plot. The peak at low frequency indicates that there is still uncaptured trend, and the length between the highest and lowest peaks still indicate that the residuals have not been reduced to white noise.

Part 5 We added lag1 and lag2 residuals as variables and saw that the residual spectral plot is now flat, meaning that their addition captured the autocorrelation component and has reduced the residuals to white noise levels. A quick look at the autocorrelation function plot also shows that trend has mostly been captured in the model.