Assigment 1 Additional Problem 2

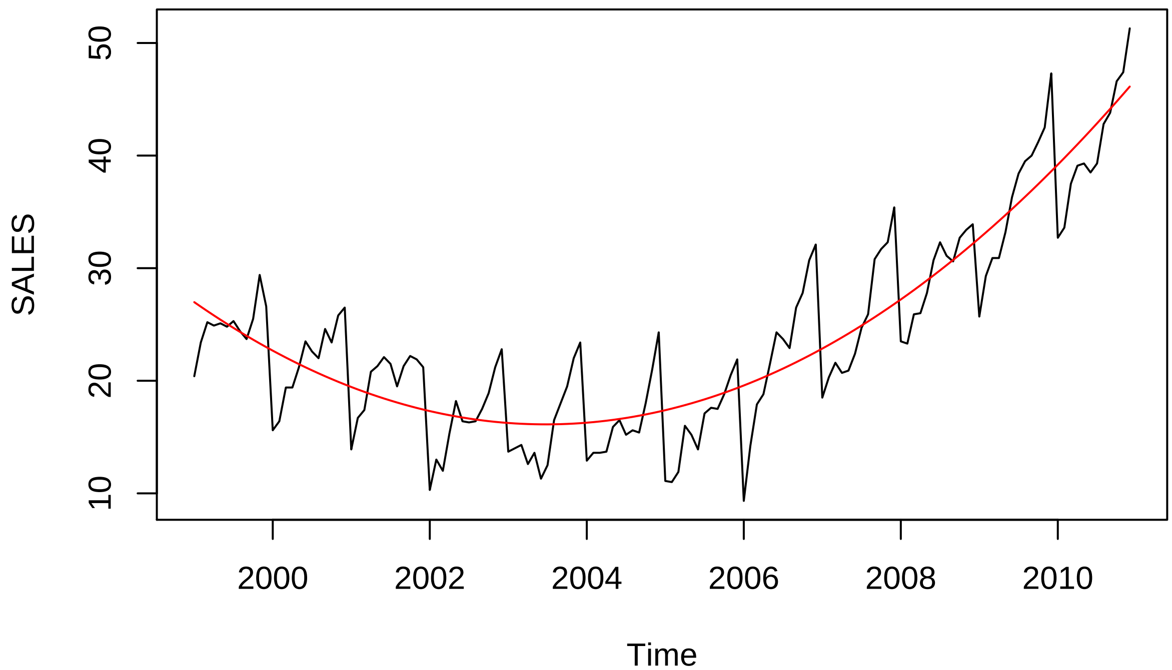
```r
setwd("~/googledrive/Analytics/msan604/AS1/")
rm(list=ls())
```

```r
sales <- read.table("SALES.txt")
sales <- ts(sales, start=1999, frequency=12)
```

```r
t <- time(sales)
t2 <- t^2
month <- as.factor(cycle(sales))
```

A)

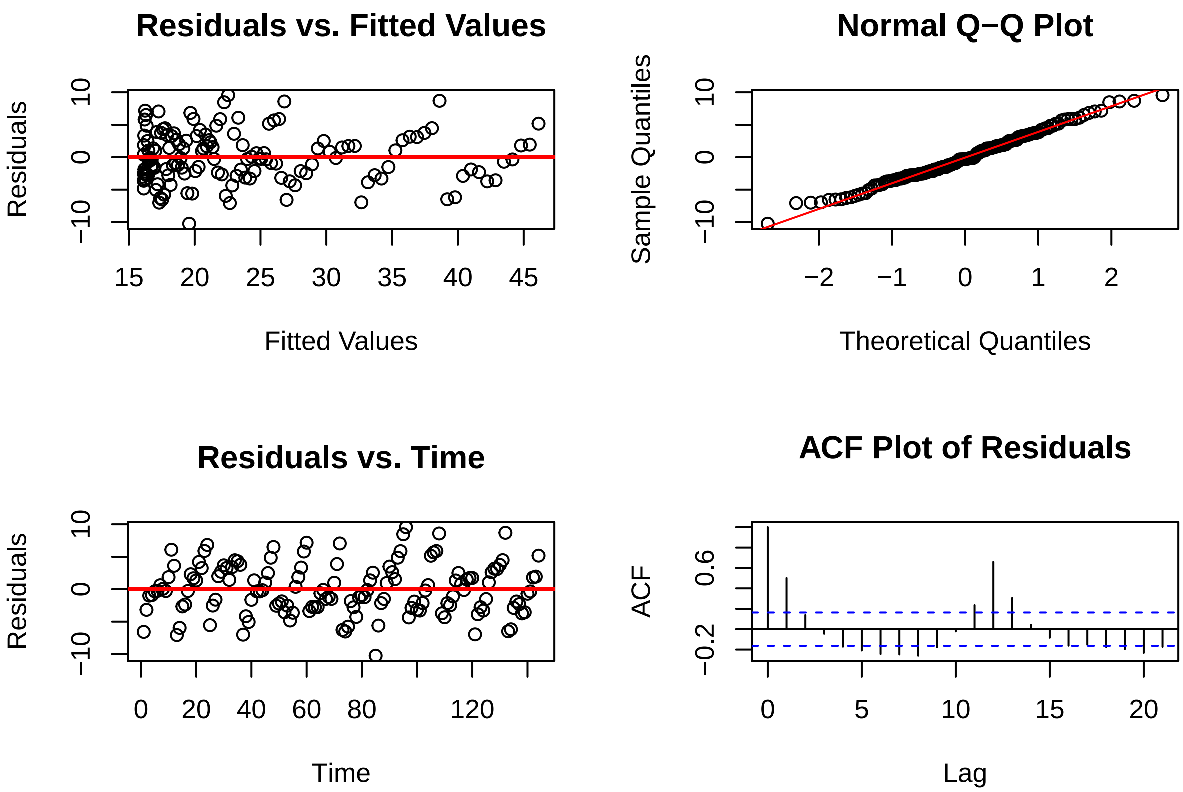
```r
trend_reg <- lm(sales~ t + t2)
plot(sales)
points(t,predict.lm(trend_reg),type='l',col='red')
```



As expected, the shape of the trend is well captured by the model, but the seasonality component is not captured. This is expected because the model was only fitted against the number periods from the beginning of 1999 to the end of 2011.

```r
par(mfrow=c(2,2))
plot(trend_reg$fitted, trend_reg$residuals, main = "Residuals vs. Fitted Values",
     ylab = "Residuals", xlab = "Fitted Values")
abline(h = 0, col = "red", lwd = 2)

#qq-plot of residuals
qqnorm(trend_reg$residuals)
```

```r
# plotting the line, along which the dots in qq-plot should lie
qqline(trend_reg$residuals, col = "red")

 # plotting the residuals vs time
plot(trend_reg$residuals, main = "Residuals vs. Time", ylab = "Residuals", xlab = "Time")

# plotting a horizontal line at 0
abline(h = 0, col = "red", lwd = 2)

 #sample acf plot of residuals
acf(trend_reg$residuals, main = "ACF Plot of Residuals")
```



Homoskedasticity

The residuals do not show any obvious megaphone shape where we would conclude the resiudals are heteroskedastic. So currently, we can assume the residuals have constant variance.

Normality

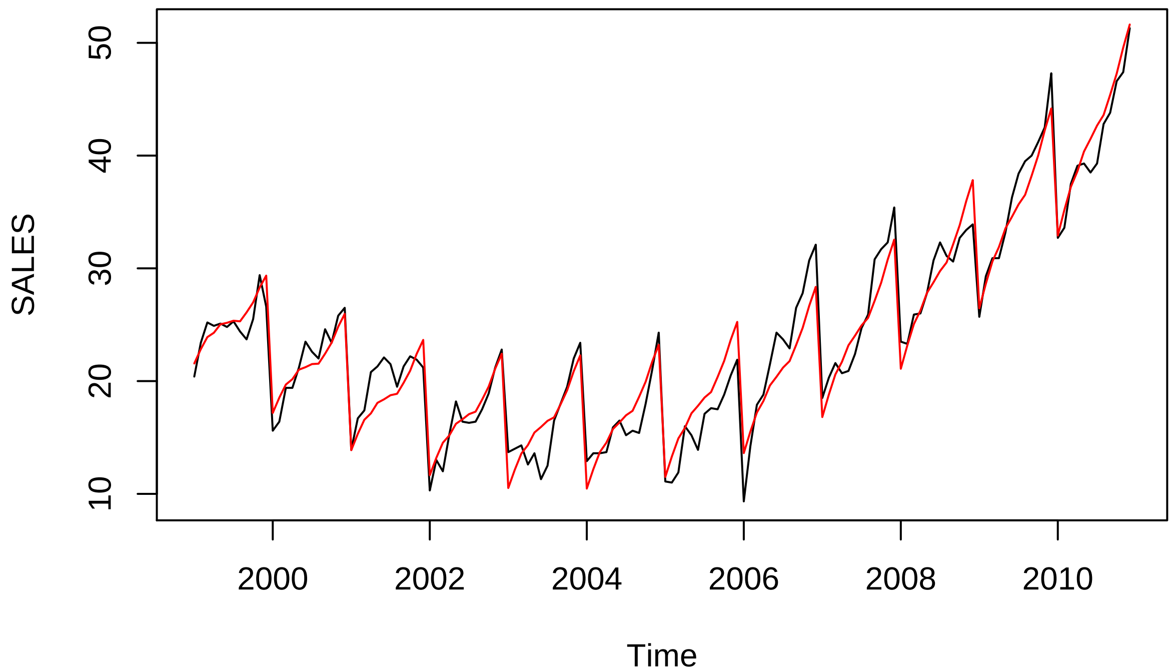
Shown in the Q-Q Plot, most residuals fall on the guideline for normality with a few residuals at the tails straying. But overall, the residuals approximate a normal distribution.

Independence

The sequence plot doesn't show any sequential patterns among the residuals, but the ACF plot shows there are significant spikes at lags > 1. In fact, the ACF plot shows there is a cyclical nature to the correlation between residuals at different lags. This indicates that our regression should now account for that cyclical nature in our residuals.

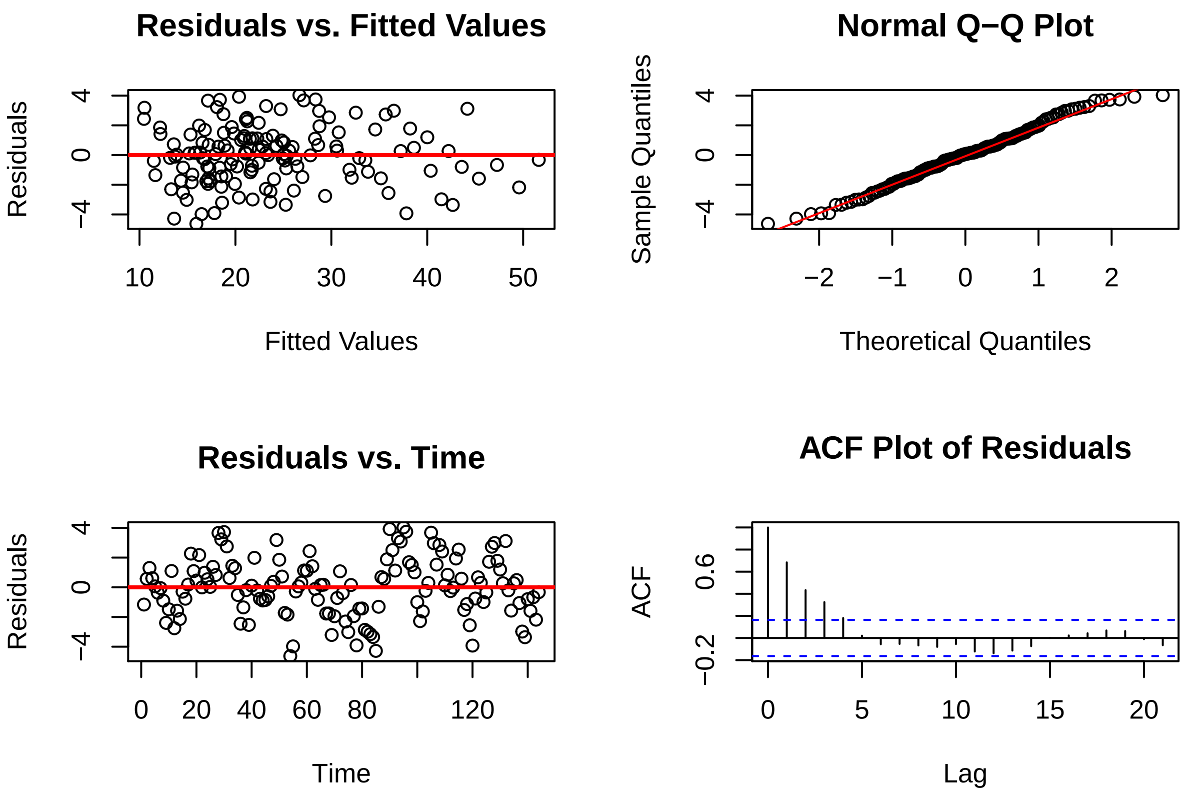
B)

```
par(mfrow=c(1,1))
trend_reg <- lm(sales~ t + t2 + month)
plot(sales)
points(t,predict.lm(trend_reg),type='l',col='red')
```



The model now fits the orignal data much better, now that seasonality is accounted for. In this case, we added a categorical variable that accounts for shifts in sales in different months each year.

```
par(mfrow=c(2,2))
plot(trend_reg$fitted, trend_reg$residuals, main = "Residuals vs. Fitted Values", ylab = "Residuals", xl
abline(h = 0, col = "red", lwd = 2)

#qq-plot of residuals
qqnorm(trend_reg$residuals)

# plotting the line, along which the dots in qq-plot should lie
qqline(trend_reg$residuals, col = "red")

 # plotting the residuals vs time
plot(trend_reg$residuals, main = "Residuals vs. Time", ylab = "Residuals", xlab = "Time")

 # plotting a horizontal line at 0
abline(h = 0, col = "red", lwd = 2)

#sample acf plot of residuals
acf(trend_reg$residuals, main = "ACF Plot of Residuals")
```



Homoskedasticity

The residudals clearly show randomness and no obvious changes in spread, so we can conclude the residuals are homoskedastic.

Normality

Shown in the Q-Q Plot, almost all residuals fall on the normal line. We can safely assume the resiudals are normal.

Independence

The sequence plot doesn't show any sequential patterns among the residuals, and the seasonality nature of the residuals has been removed. However, there are significant spikes at lags greater than 0, so this may affect our ability to forecast with any degree of confidence.

C)

The model in B fits better than the model in A because the model in B accounts for the seasonality in the months of the time series.

B)

```r
t.new <- seq(2011,2012,length=13)[1:12]
t2.new <- t.new^2
month.new <- factor(rep(1:12,1))

new <- data.frame(t=t.new, t2=t2.new, month=month.new)
pred <- predict.lm(trend_reg,new,interval='prediction')

plot(sales,xlim=c(1999,2012),ylim=c(0,80)) #plotting the data

abline(v=2011,col='blue',lty=2)
lines(pred[,1]~t.new,type='l',col='red')
lines(pred[,2]~t.new,col='green')
lines(pred[,3]~t.new,col='green')
```