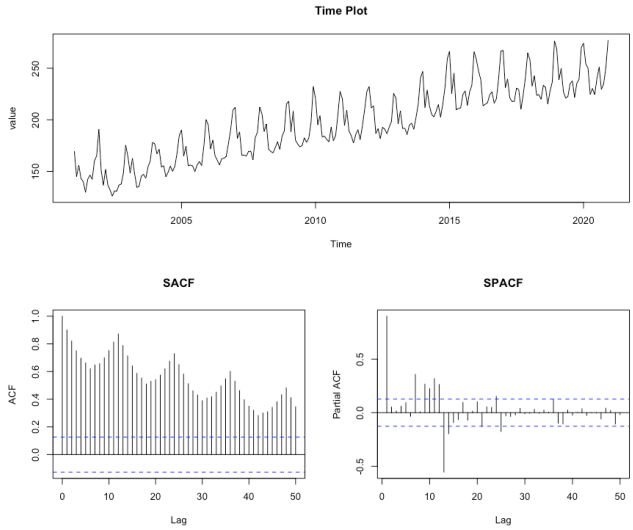
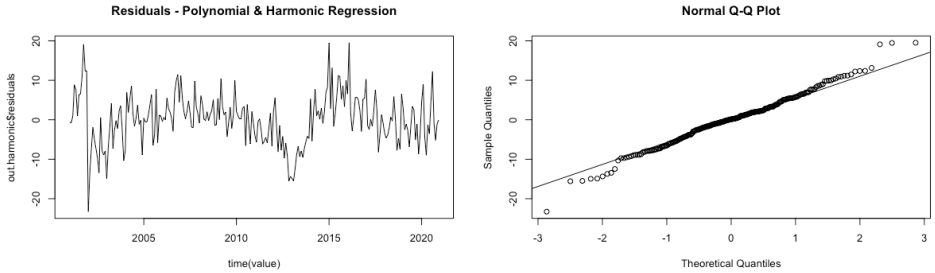
시계열분석입문 – Exam 2(2022)

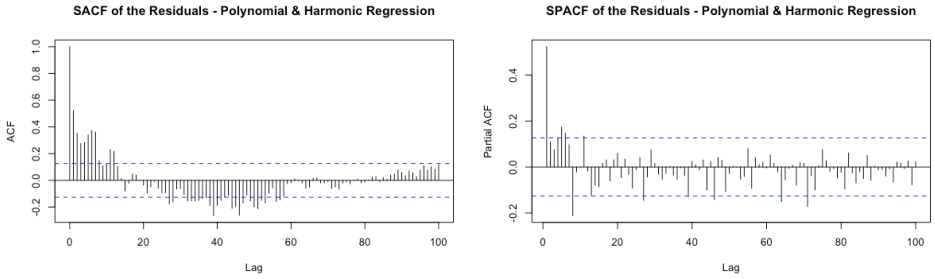
통계학과 2019313106 홍현경



Time plot을 통해, 주어진 시계열 데이터에는 증가하는 추세가 존재함을 확인할 수 있다. Time plot만으로는 형태를 확신할 수는 없지만, 주어진 데이터는 선형적으로 증가하는 추세, 또는 완전히 선형적이지는 않더라도 2, 3차 곡선의 증가하는 형태를 가지고 있다. 더불어, time plot에서는 주기 12의 뚜렷한 계절성이 나타난다. 시간에 따른 데이터의 분산은 크게 달라지지 않으므로 분산안정화 변환은 불필요하며, 데이터의 이상치는 존재하지 않는 것으로 보인다. SACF plot에서 천천히 감소하는 형태와, 12개의 lag마다 감소-증가하는 형태의 패턴이 나타나며, SACF가 모두 파란 선으로 표시된 boundary를 벗어나 있다. 이는 데이터에 추세와 계절성이 모두 존재함을 의미한다. 추가적으로 그린 SPACF plot에서는 lag 12가 boundary를 크게 벗어나 있는 것을 확인할 수 있는데, 이는 데이터에 주기 12의 계절성이 존재한다는 사실을 뒷받침한다. 정리하자면, 주어진 시계열 데이터는 선형적으로, 또는 2, 3차 곡선 형태로 뚜렷하게 증가하는 추세를 가지고 있으며, 주기 12의 뚜렷한 계절성 역시 가지고 있다. 더불어, SACF는 천천히 선형적으로 감소하면서도 lag 12마다 감소-증가하는 뚜렷한 패턴을 가지고 있다. 즉, 주어진 데이터는 시간에 따라 평균이 달라지며 공분산이 시점에 의존하는 비정상시계열이다.

Time plot과 SACF에서 주어진 데이터에 뚜렷한 추세가 존재함을 확인할 수 있었다. 따라서 데이터의 추세를 제거하기 위해 다항회귀의 차수를 1차부터 4차까지 조정하여 회귀식을 적합하였다. 그 결과, 차수를 1차로 설정했을 때의 AIC, BIC, AICC의 값이 각각 2033.004, 2043.446, 2033.105로 가장 낮았지만, 그 이상의 차수로 회귀식을 적합했을 경우에도 AIC, BIC, AICC의 값이 2~4정도만 미세하게 증가할 뿐, 큰 차이가 없었다. 각 차수에 대한 residual plot을 비교한 결과, 차수를 2차로 두었을 때 추세가 가장 잘 제거되었고, 3차, 4차의 경우 큰 차이가 없었다. 따라서 주어진 데이터에는 2차 곡선 형태의 증가하는 추세가 있다고 판단하였고, 차수를 2차로 설정하여 다항회귀를 적합하였다. Time plot과 SACF에서는 추세 뿐만이 아니라 주기 12의 뚜렷한 계절성 역시 관찰 가능했다. 따라서, 다항회귀를 이용하여 데이터에서 추세를 제거한 후 조화회귀를 이용하여 계절성을 제거하였다. Stepwise selection을 이용하여 sin-cos 항의 개수를 조정하였고, 그 결과 costerm1 + sinterm1 + costerm2 + sinterm2 + costerm3 + sinterm3 + costerm4 + sinterm4 + costerm5 + sinterm5 + costerm6이 가장 적합한 것으로 도출되었다.

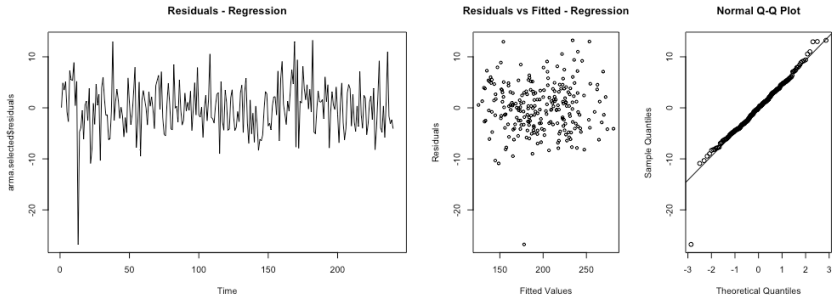


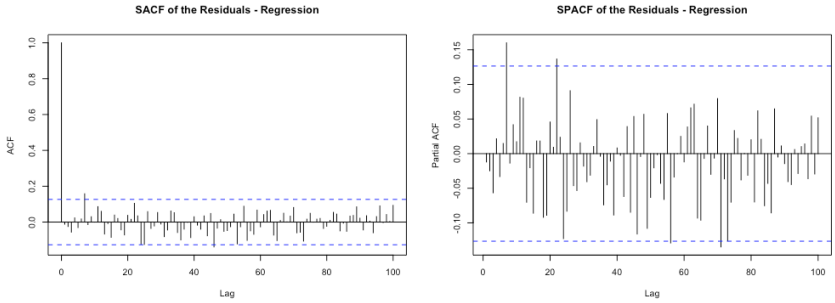


다항회귀와 조화회귀를 이용하여 추세와 계절성을 제거한 후의 잔차에 대한 plot은 다음과 같다. Residual plot을 보면, 추세가 완전히 제거되었다고 보기에는 어려울 수 있으나 기존 데이터의 3차 곡선 형태의 증가하는 추세와 계절성은 제거가 된 것을 확인할 수 있다. SACF에서도 기존의 천천히 감소하는 형태와 12개의 lag마다 증가-감소하는 패턴은 나타나지 않는다. 잔차에 대한 SACF, SPACF plot을 보면 추세와 계절성 제거 이후에도 잔차에서 dependence structure가 나타나는 것을 확인할 수 있다. 따라서, ARMA(p, q) 모델을 이용하여 이 dependence structure를 모델링을 시행해야 한다. 잔차에 대한 SACF는 빠르게 감소하는 형태를 보이고 있다. 더불어, 잔차에 대한 SPACF는 lag 1과 lag 8에서 boundary를 벗어나고 있다. Lag 8에 대한 SPACF가 이 boundary를 크게 벗어나기 때문에, 우선 AR(p)의 차수를 크게 잡고 constraint optimization을 시행하는 것이 적절하다고 판단하였다. 이러한 이유에서 SPACF가 lag 8 이후 절단되었다고 보는 것이 맞다고 판단했으며, 따라서 SACF, SPACF의 형태를 기반으로 AR(8)정도가 적합하다고 판단하였다.

GLS를 이용하여 2차의 추세, 그리고 주기 12의 계절성을 잡아주고, error에 대해 ARMA(p, q) 모델링을 하는 과정에서 가장 적절한 차수를 찾아주기 위해 잔차의 SACF, SPACF에서 도출된 차수를 기반으로 하여 모든 차수의 조합에 대한 grid를 만들어 주었고, 이 중에서 information criteria가 최소화되는 차수 조합을 찾아주었다. ARMA(p, q)에서 p의 범위는 0~8, q의 범위는 0~3으로 설정하여 모든 차수 (p, q) 조합에 대한 AIC, AICC, BIC를 계산하였다. 그 결과, AIC와 AICC, BIC가 가장 작아지는 모델은 ARMA(6, 3)이었다. 그 다음에는 p와 q의 범위를 동일하게 하여 모든 차수 조합에 대한 grid를 만들어 주었고, 이 중에서 out-of-sample forecasting error가 최소화되는 차수 조합을 찾아주었다. 그 결과, ARMA(8, 1) 모델의 out-of-sample forecasting error가 가장 작았다. 이렇게 information criteria를 기준으로 했을 때와 out-of-sample forecasting error를 기준으로 했을 때의 차수가 서로 다르게 도출되었다. 그러나, out-of-sample forecasting error를 기준으로 했을 때, information criteria 기준에서 가장 적합한 모델이었던 ARMA(6, 3)이 3번째로 작은 error 값을 갖고, 두 모델의 error 값의 차이는 1정도로 작은 것으로 나타났다. ARMA(8, 1)과 ARMA(6, 3)의 복잡도 역시 크게 차이가 없기 때문에, 두 기준을 종합하여 ARMA(6, 3)을 error term에 대한 최종 모델로 선정하였다.

최종적으로 데이터에 GLS를 이용하여 다항회귀와 조화회귀를 적합하고, error에 대해 ARMA(6, 3) 모델을 적합하였다. 우선, 각 AR, MA 계수들이 0이라는 것을 귀무가설로 설정하여 계수에 대한 검정을 시행하였고, MA(1)와 MA(2)의 계수의 p-value가 0.05보다 커서 해당 계수가 0이라는 귀무가설이 기각되지 않았다. 따라서, MA(1)와 MA(2)에 대한 계수를 0으로 설정한 후 모델을 다시 적합해주었다.



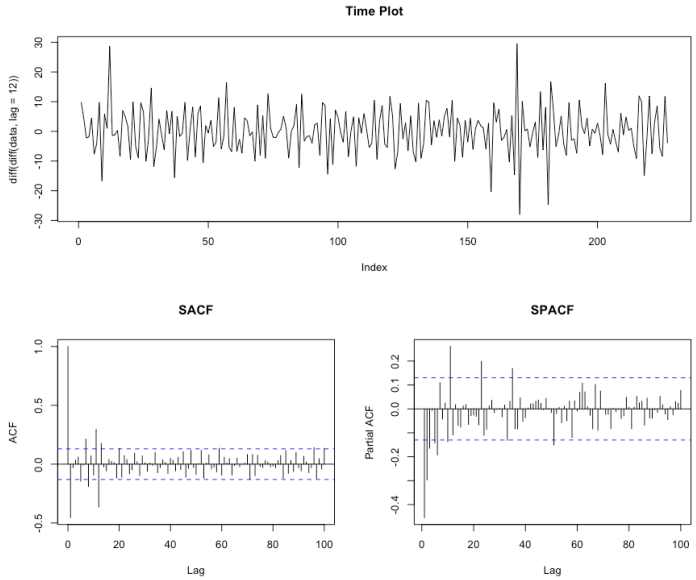


GLS로 다항회귀와 조화회귀를 적합한 후, error term에 대해 MA(1), MA(2)의 계수가 0인 ARMA(6, 3)을 적합한 결과는 다음과 같다. Residual plot을 통해 잔차에 패턴이 존재하지 않음을 확인할 수 있고, 더불어 residual vs fitted plot에서도 어떠한 패턴도 관찰되지 않는다. Normal QQ plot을 통해 잔차가 정규성을 만족함을 확인할 수 있다. 더불어, SACF는 대부분의 lag에 대해 boundary 내부에 존재하며 감소하는 추세나 계절성의 패턴 등을 가지지 않는다. SPACF의 경우에도 대부분이 boundary 내부에 존재하며 특별한 패턴은 관찰되지 않는다. 따라서 residual에 대한 plot들과 SACF, SPACF를 통해 잔차가 iid noise임을 확인할 수 있다.

```
Null hypothesis: Residuals are iid noise.
Test                      Distribution Statistic   p-value
Ljung-Box Q                Q ~ chisq(20)     16.42    0.6901
McLeod-Li Q                Q ~ chisq(20)     10.27    0.9631
Turning points T  (T-158.7)/6.5 ~ N(0,1)       159    0.9591
Diff signs S      (S-119.5)/4.5 ~ N(0,1)       122    0.5769
Rank P          (P-14340)/621.6 ~ N(0,1)     13976    0.5582
```

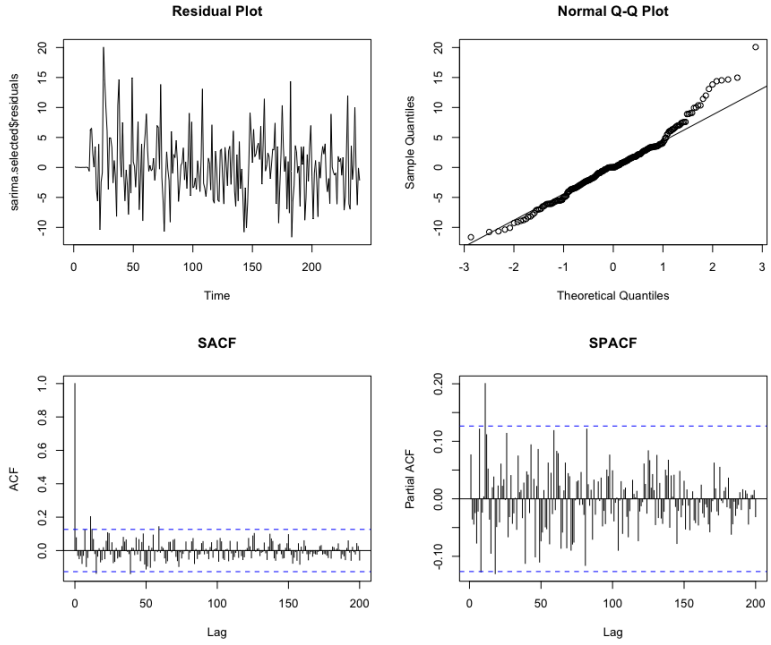
iid noise 여부에 대한 검정 결과 역시 5개의 검정 모두 iid noise라는 귀무가설을 기각하지 못했으므로 Regression + ARMA(p, q)가 잘 적합되었음을 확인할 수 있다. 이어서 정규성 검정을 위해 Shapiro-wilk, Anderson-Darling, Cramer-von Mises, Kolmogorov-Smirnov 그리고 Jarque Bera test 총 5가지 검정을 시행하였는데, Shapiro-wilk, Jarque Bera test를 제외하고는 모두 p-value가 0.05보다 커서 귀무가설이 기각되지 않았다. Normal QQ plot 상으로도 정규성에 큰 문제가 없기 때문에 정규성 가정 역시 만족한다고 판단할 수 있다. 따라서, Regression + ARMA errors에서는 GLS를 이용해 2차의 다항 회귀와, costerm1 + sinterm1 + costerm2 + sinterm2 + costerm3 + sinterm3 + costerm4 + sinterm4 + costerm5 + sinterm5 + costerm6 항을 이용하여 조화 회귀를 적합하며 error는 MA(1), MA(2)의 계수가 0인 ARMA(6, 3)을 따르는 모델이다.

Regression + ARMA 모델을 적합하는 과정에서 확인한 바와 같이 다항 회귀로 추세를 제거한 후에도 약간의 추세로 볼 수 있는 형태가 관찰되었다. 이는 다항회귀를 이용하여 추세를 제거하는 방법이 적합한 방법은 아니라는 것을 의미한다. 따라서, 차분을 이용하여 추세와 계절성을 제거하며, 계절 주기에 대한 ARMA 모델 역시 적합할 수 있는 비정상시계열에 대한 모델인 SARIMA(p, d, q)(P, D, Q)s를 이용해보았다.



계절차분은 추세에 대한 1차 차분을 포함하고 있기 때문에 우선 lag = 12의 계절 차분을 적용하였고, 이후 계절성은 제거되었으나 추세가 완전히 제거되지 않는 문제가 발생해 한번 더 차분을 진행했다. 차분의 경우 계절차분 이후 한 번의 추가 차분만으로도 추세가 완전히 제거되었고, 여기에서 차분을 한번 더 할 경우 과대 차분으로 인해 전체적인 분산이 커지는 문제가 발생했다. 따라서, lag = 12의 계절 차분 1번, 그리고 추세를 제거하기 위한 차분 1번이 적합하다고 판단하였다. 차분 이후 residual plot에서는 추세나 계절성을 비롯한 어떠한 패턴도 나타나지 않는다. SACF에서는 lag 1과 계절 주기에 해당하는 lag 12가 boundary에서 벗어난 것을 확인할 수 있다. 더불어, SPACF는 전체적으로 0에 가까워지는 형태이며, 계절 주기인 12의 배수에 대해서도 SPACF가 감소하는 형태임을 확인할 수 있다. 차분 후의 SACF, SPACF의 형태를 볼 때, SARIMA(0, 1, 1)(0, 1, 1)[s=12] 정도가 적합하다고 판단하였다.

좀 더 정확한 차수 선택을 위해, p, q, P, Q의 범위를 모두 0~2로 설정하여 grid를 만들어 주었고, 모든 조합에 대해 information criteria 값을 계산하여 이 값이 최소화되는 조합을 찾아주었다. 그 결과, SARIMA(0, 1, 2)(2, 1, 2)[s=12]의 AIC, AICC 값이 가장 낮았고, SARIMA(0, 1, 1)(0, 1, 1)[s=12]의 BIC 값이 가장 낮았다. 그 후, p, q, P, Q의 범위를 동일하게 설정하여 grid를 만들어 주었고, 모든 조합에 대해 out-of-sample forecasting error를 계산하여 이 값이 최소화되는 조합을 찾아주었다. 그 결과, SARIMA(0, 1, 2)(2, 1, 2)[s=12]의 error 값이 가장 낮았다. 따라서 information criteria, out-of-sample forecasting error를 모두 최소화 할 수 있는 SARIMA(0, 1, 2)(2, 1, 2)[s=12]를 최종 모델로 선정하였다. 모델을 선정한 후 각 AR, MA 계수들이 0이라는 것을 귀무가설로 설정하여 계수에 대한 검정을 시행하였다. 그 결과, MA(2)의 계수의 p-value가 0.05보다 커서 해당 계수가 0이라는 귀무가설이 기각되지 않았다. 따라서 이 MA(2)의 계수를 0으로 설정해준 후 모델을 다시 적합하였다.



MA(2)의 계수가 0인 SARIMA(0, 1, 2)(2, 1, 2)[s=12], 즉 사실상 SARIMA(0, 1, 1)(2, 1, 2)[s=12] 모델을 적합한 후의 결과는 다음과 같다. 우선 residual plot을 보면 중간에 분산이 작아지는 부분이 존재하기는 하나 큰 패턴은 존재하지 않는 것으로 보인다. 더불어 Normal QQ plot에서 끝부분이 정규분포에서 조금 떨어져 있으나 잔차가 대부분 정규성을 만족함을 확인할 수 있다. 더불어, SACF는 대부분의 lag에 대해 boundary 내부에 존재하며 감소하는 추세나 계절성의 패턴 등을 가지지 않는다. SPACF의 경우에도 대부분이 boundary 내부에 존재하며 특별한 패턴은 관찰되지 않는다. 따라서 residual plot, normal QQ plot, SACF, 그리고 SPACF를 통해 잔차가 iid noise임을 확인할 수 있다.

```
Null hypothesis: Residuals are iid noise.
Test                      Distribution Statistic   p-value
Ljung-Box Q               Q ~ chisq(20)     33.65    0.0286 *
McLeod-Li Q               Q ~ chisq(20)     46.51     7e-04 *
Turning points T  (T-158.7)/6.5 ~ N(0,1)      152    0.3056
Diff signs S       (S-119.5)/4.5 ~ N(0,1)     117    0.5769
Rank P          (P-14340)/621.6 ~ N(0,1)     13550    0.2038
```

iid noise 여부에 대한 검정 결과 역시 5개 중 3개가 iid noise라는 귀무가설을 기각하지 못했으므로 SARIMA 모델의 적합이 잘 되었음을 알 수 있다. Ljung-box test의 경우 p-value가 0.05보다 작기는 하나 0.01보다는 크기 때문에 기각의 기준을 0.01로 잡아준다면 Ljung-box test 역시 기각되지 않는다고 볼 수 있다. 이어서 정규성 검정을 위해 Shapiro-wilk, Anderson-Darling, Cramer-von Mises, Kolmogorov-Smirnov 그리고 Jarque Bera test 총 5가지 검정을 시행하였는데, 각각의 p-value가 모두 0.05 미만으로 모든 test에 대해 정규성을 만족한다는 귀무가설이 기각되었다. 하지만, Normal QQ plot에서도 볼 수 있듯이 끝 부분을 제외한 나머지는 정규분포에 가까우며 정규성 문제의 경우 t분포와 같은 다른 분포를 적용하여 해결할 수 있기 때문에 큰 문제가 되지 않는다고 판단하였다. 따라서 주어진 데이터에 대해 MA(2)의 계수가 0인 SARIMA(0, 1, 2)(2, 1, 2)[s=12], 즉 사실상 SARIMA(0, 1, 1)(2, 1, 2)[s=12]가 가장 적합한 모델이다.

2차 다항회귀 + MA(1), MA(2)의 계수가 0인 ARMA(6, 3)과 MA(2)의 계수가 0인 SARIMA(0, 1, 2)(2, 1, 2)[s=12]의 미래 4개 시점에 대한 예측 결과는 다음과 같다.

	Jan 2021	Feb 2021	Mar 2021	April 2021
Model (c) Point Forecast 95% PI	(269.29, 289.27)	(240.59, 261.89)	(244.59, 266.71)	(220.79, 244.09)
Model (d) Point Forecast 95% PI	(267.77, 288.81)	(239.88, 262.06)	(244.73, 267.98)	(221.83, 246.10)

Model (c): 2차 다항회귀 + MA(1), MA(2)의 계수가 0인 ARMA(6, 3)과 Model (d): MA(2)의 계수가 0인 SARIMA(0, 1, 2)(2, 1, 2)[s=12]중 더 선호하는 모델은 MA(2)의 계수가 0인 SARIMA(0, 1, 2)(2, 1, 2)[s=12]이다. 앞서 언급한 바와 같이, 2차 다항회귀 + MA(1), MA(2)의 계수가 0인 ARMA(6, 3) 모델에서 다항 회귀는 추세를 제거하는데 적절한 방법이 아님을 확인하였으며, 이 모델의 경우 계절성을 모델링하기 위한 조화 회귀의 항 역시 복잡하다. 더불어 모델의 AR, MA의 차수 역시 높으며, constraint optimization을 했음에도 불구하고 제거되는 항이 MA(1), MA(2) 밖에 존재하지 않아 모델이 복잡하다. 그러나, SARIMA(0, 1, 2)(2, 1, 2)[s=12]의 경우 계절 차분 한번과, 또 한번의 추세 차분을 통해서도 데이터가 가진 추세와 계절성을 완전히 제거 가능했고, AR, MA의 차수 역시 높지 않아 모델이 Regression + ARMA errors 모델에 비해 간단하다. 두 모델 모두 적합 후 잔차가 iid noise를 따른다는 점은 동일하기 때문에 추세와 계절성 모델링의 적합성 및 효율성과 모델의 복잡도 측면에서 보았을 때 SARIMA(0, 1, 2)(2, 1, 2)[s=12]가 더 선호된다.

요약하자면, 주어진 데이터는 평균이 시점에 따라 변화하는 비정상시계열이기 때문에 기본적으로 추세와 계절성을 제거해주어야 했다. 주어진 데이터를 이용하여 미래를 예측하기 위해서는 주어진 데이터를 이용하여 모델을 적합해야 하는데, 이 때 Regression + ARMA errors 모형과 SARIMA 모형을 사용하였다. Regression + ARMA errors 모형을 적합하기 위해 우선 AIC, BIC, AICC의 information criteria와 적합 후 제거되는 추세의 비교를 통해 다항회귀의 차수를 2차로 결정하였다. 더불어 계절성을 제거하기 위해 stepwise selection으로 조화회귀를 적용하였다. Error의 dependence structure를 linear process인 ARMA 모형을 이용하여 parametric하게 모델링 해 주기 위해 AIC, AICC, BIC의 information criteria 값, 그리고 out-of-sample forecasting error 두 가지 기준으로 ARMA(p, q)의 차수를 탐색하였고, 최종적으로 ARMA(6, 3)을 선정하였다. 이 때, 검정을 통해 MA(1), MA(2)의 계수가 0임을 확인하여 제약 조건을 걸어준 후 GLS를 이용하여 2차 다항 회귀 + ARMA(6, 3) 모델을 적합하였다. SARIMA 모델의 경우 한번의 계절 차분과 또 한번의 추세 제거를 위한 차분만으로도 추세와 계절성 제거가 충분히 가능했다. SARIMA 모델 역시 차수 확인을 위해 AIC, AICC, BIC의 information criteria 값, 그리고 out-of-sample forecasting error 두 가지 기준으로 차수를 탐색하였고, SARIMA(0, 1, 2)(2, 1, 2)[s=12]를 최종 모델로 선정하였다. 이후 계수 검정을 통해 MA(2)가 0인 것으로 나타나 최종적으로 MA(2)의 계수가 0인 SARIMA(0, 1, 2)(2, 1, 2)[s=12]를 적합했다. 두 모델 모두 적합 후 iid noise를 얻을 수 있어서 모델의 적합이 잘 이루어졌다고 할 수 있지만, 회귀는 추세를 모델링하는데 적합한 방법이 아니라는 점과 모델의 복잡도를 고려했을 때 MA(2)의 계수가 0인 SARIMA(0, 1, 2)(2, 1, 2)[s=12] 모델이 더 선호된다.

```
######################
### Basic settings ###
######################

rm(list=ls(all=TRUE))

setwd("/Users/hyungyeonghong/Desktop/2022_Spring/TSA2022sp/TSA_exam2")

data.original = read.csv("2022practice2.csv", head = FALSE) # change file name
head(data.original)

data.original = data.original$V1 # select a column
head(data.original)
length(data.original)

value = ts(data.original, start = c(2001, 1), end = c(2020, 12), frequency = 12)

################################
### Time Plot, Correlograms ###
################################

layout(matrix(c(1, 1, 2, 3), 2, 2, byrow = TRUE))
plot(value, type = "l"); title(main = "Time Plot");
acf.plot = acf(data.original, lag = 50, plot = FALSE); plot(acf.plot, main = "SACF")
pacf.plot = pacf(data.original, lag = 50, plot = FALSE); plot(pacf.plot, main = "SPACF")


##############################################
### Choosing the data you are going to use ###
##############################################

data = data.original

############################
### Polynomial Regression ###
############################

n = length(data)

x = seq(from = 1, to = n, by = 1)
x2 = x^2
x3 = x^3
x4 = x^4

out.polynomial1 = lm(data ~ 1 + x)
out.polynomial2 = lm(data ~ 1 + x + x2)
```

```
out.polynomial3 = lm(data ~ 1 + x + x2 + x3)
out.polynomial4 = lm(data ~ 1 + x + x2 + x3 + x4)

AIC(out.polynomial1)
AIC(out.polynomial2)
AIC(out.polynomial3)
AIC(out.polynomial4)

BIC(out.polynomial1)
BIC(out.polynomial2)
BIC(out.polynomial3)
BIC(out.polynomial4)

MuMIn::AICc(out.polynomial1)
MuMIn::AICc(out.polynomial2)
MuMIn::AICc(out.polynomial3)
MuMIn::AICc(out.polynomial4)


# 1st Order Polynomial Regression
par(mfrow = c(1, 1))
plot.ts(data)
lines(x = 1:n, y = out.polynomial1$fitted.values, col = "red")
title("Estimated Trend - Polynomial Regression")

par(mfrow = c(2, 2))
plot(x = 1:n, y = out.polynomial1$residuals, type = "l")
title("Residuals - Polynomial Regression")
qqnorm(out.polynomial1$residuals)
qqline(out.polynomial1$residuals)
acf.plot = acf(out.polynomial1$residuals, lag = 50, plot = FALSE)
plot(acf.plot, main = "SACF of the Residuals - Polynomial Regression")
pacf.plot = pacf(out.polynomial1$residuals, lag = 50, plot = FALSE)
plot(pacf.plot, main = "SPACF of the Residuals - Polynomial Regression")


# 2nd Order Polynomial Regression
par(mfrow = c(1, 1))
plot.ts(data)
lines(x = 1:n, y = out.polynomial2$fitted.values, col = "red")
title("Estimated Trend - Polynomial Regression")

par(mfrow = c(2, 2))
plot(x = 1:n, y = out.polynomial2$residuals, type = "l")
title("Residuals - Polynomial Regression")
qqnorm(out.polynomial2$residuals)
```

```
qqline(out.polynomial2$residuals)
acf.plot = acf(out.polynomial2$residuals, lag = 50, plot = FALSE)
plot(acf.plot, main = "SACF of the Residuals - Polynomial Regression")
pacf.plot = pacf(out.polynomial2$residuals, lag = 50, plot = FALSE)
plot(pacf.plot, main = "SPACF of the Residuals - Polynomial Regression")


# 3rd Order Polynomial Regression
par(mfrow = c(1, 1))
plot.ts(data)
lines(x = 1:n, y = out.polynomial3$fitted.values, col = "red")
title("Estimated Trend - Polynomial Regression")

par(mfrow = c(2, 2))
plot(x = 1:n, y = out.polynomial3$residuals, type = "l")
title("Residuals - Polynomial Regression")
qqnorm(out.polynomial3$residuals)
qqline(out.polynomial3$residuals)
acf.plot = acf(out.polynomial3$residuals, lag = 50, plot = FALSE)
plot(acf.plot, main = "SACF of the Residuals - Polynomial Regression")
pacf.plot = pacf(out.polynomial3$residuals, lag = 50, plot = FALSE)
plot(pacf.plot, main = "SPACF of the Residuals - Polynomial Regression")


# 4th Order Polynomial Regression
par(mfrow = c(1, 1))
plot.ts(data)
lines(x = 1:n, y = out.polynomial4$fitted.values, col = "red")
title("Estimated Trend - Polynomial Regression")

par(mfrow = c(2, 2))
plot(x = 1:n, y = out.polynomial4$residuals, type = "l")
title("Residuals - Polynomial Regression")
qqnorm(out.polynomial4$residuals)
qqline(out.polynomial4$residuals)
acf.plot = acf(out.polynomial4$residuals, lag = 50, plot = FALSE)
plot(acf.plot, main = "SACF of the Residuals - Polynomial Regression")
pacf.plot = pacf(out.polynomial4$residuals, lag = 50, plot = FALSE)
plot(pacf.plot, main = "SPACF of the Residuals - Polynomial Regression")

# Final polynomial regression model
out.polynomial = lm(data ~ 1 + x + x2) # choose the final polynomial regression model

par(mfrow = c(2, 2))
plot(x = 1:n, y = out.polynomial$residuals, type = "l") # change the fitted model
title("Residuals - Regression")
```

```
qqnorm(out.polynomial$residuals) # change the fitted model
qqline(out.polynomial$residuals) # change the fitted model
acf.plot = acf(out.polynomial$residuals, lag = 50, plot = FALSE) # change the fitted model
plot(acf.plot, main = "SACF of the Residuals - Regression")
pacf.plot = pacf(out.polynomial$residuals, lag = 50, plot = FALSE) # change the fitted model
plot(pacf.plot, main = "SPACF of the Residuals - Regression")

############################
### Harmonic Regression ###
############################

t = 1:n
d = 12 # change the seasonal period

f1 = n/d
f2 = 2*f1
f3 = 3*f1
f4 = 4*f1
f5 = 5*f1
f6 = 6*f1

costerm1 = cos(f1*2*pi/n*t); sinterm1 = sin(f1*2*pi/n*t)
costerm2 = cos(f2*2*pi/n*t); sinterm2 = sin(f2*2*pi/n*t)
costerm3 = cos(f3*2*pi/n*t); sinterm3 = sin(f3*2*pi/n*t)
costerm4 = cos(f4*2*pi/n*t); sinterm4 = sin(f4*2*pi/n*t)
costerm5 = cos(f5*2*pi/n*t); sinterm5 = sin(f5*2*pi/n*t)
costerm6 = cos(f6*2*pi/n*t); sinterm6 = sin(f6*2*pi/n*t)


# Setting k = 1
out.harmonic1 = lm(out.polynomial$residuals ~ 1 + costerm1 + sinterm1)
summary(out.harmonic1)


# Setting k = 2
out.harmonic2 = lm(out.polynomial$residuals ~ 1 + costerm1 + sinterm1 + costerm2 + sinterm2)
summary(out.harmonic2)


# Setting k = 3
out.harmonic3 = lm(out.polynomial$residuals ~ 1 + costerm1 + sinterm1 + costerm2 + sinterm2 +
costerm3 + sinterm3)
summary(out.harmonic3)


# Setting k = 4
```

```
out.harmonic4 = lm(out.polynomial$residuals ~ 1 + costerm1 + sinterm1 + costerm2 + sinterm2 +
costerm3 + sinterm3 + costerm4 + sinterm4)
summary(out.harmonic4)


AIC(out.harmonic1)
AIC(out.harmonic2)
AIC(out.harmonic3)
AIC(out.harmonic4)


# Stepwise Selection
out.total = lm(out.polynomial$residuals ~ 1 + costerm1 + sinterm1 + costerm2 + sinterm2 + costerm3
+ sinterm3 + costerm4 + sinterm4 + costerm5 + sinterm5 + costerm6 + sinterm6)
step(out.total, direction = "both")


# Final harmonic regression model
out.harmonic = lm(out.polynomial$residuals ~ costerm1 + sinterm1 +
                        costerm2 + sinterm2 + costerm3 + sinterm3 + costerm4 + sinterm4 +
                        costerm5 + sinterm5 + costerm6) # choose the final polynomial regression model


############################################################
### Final Residual Plot(Trend and Seasonality Removed) ###
############################################################
par(mfrow = c(2, 2))
plot(x = time(value), y = out.harmonic$residuals, type = "l")
title("Residuals - Polynomial & Harmonic Regression")
qqnorm(out.harmonic$residuals)
qqline(out.harmonic$residuals)
acf.plot = acf(out.harmonic$residuals, lag = 100, plot = FALSE)
plot(acf.plot, main = "SACF of the Residuals - Polynomial & Harmonic Regression")
pacf.plot = pacf(out.harmonic$residuals, lag = 100, plot = FALSE)
plot(pacf.plot, main = "SPACF of the Residuals - Polynomial & Harmonic Regression")

#############################################
### Regression + Stationary Errors Model ###
#############################################

# auto.arima
library(forecast)
auto.arima(out.harmonic$residuals, d = 0) # change the fitted model


# (Preliminary) Applying GLS with ARMA(p, q) errors
```

```
n = length(data)
col.const = rep(1, n)
col.x1 = x
col.x2 = x2 # remove/add the terms
col.cos1 = costerm1 # remove/add the terms
col.sin1 = sinterm1 # remove/add the terms
col.cos2 = costerm2 # remove/add the terms
col.sin2 = sinterm2 # remove/add the terms
col.cos3 = costerm3 # remove/add the terms
col.sin3 = sinterm3 # remove/add the terms
col.cos4 = costerm4 # remove/add the terms
col.sin4 = sinterm4 # remove/add the terms
col.cos5 = costerm5 # remove/add the terms
col.sin5 = sinterm5 # remove/add the terms
col.cos6 = costerm6 # remove/add the terms


X = cbind(col.const, col.x1, col.x2, col.cos1, col.sin1, col.cos2, col.sin2,
          col.cos3, col.sin3, col.cos4, col.sin4, col.cos5, col.sin5, col.cos6) # change the terms

fit.reg = Arima(data, order = c(6, 0, 3), xreg = X, include.mean = FALSE)   # change the order

layout(matrix(c(1, 1, 2, 3), 2, 2, byrow = TRUE))
plot(fit.reg$residuals, type = "l"); title("Time Plot")
acf.plot = acf(fit.reg$residuals, lag = 50, plot = FALSE); plot(acf.plot, main = "SACF")
pacf.plot = pacf(fit.reg$residuals, lag = 50, plot = FALSE); plot(pacf.plot, main = "SPACF")


# Model Selection by Information Criteria
arima.fit = NULL

pmin = 0; qmin = 0 # change the minimum values
pmax = 8; qmax = 3 # change the maximum values

result.df = expand.grid(val_p = pmin:pmax, val_q = qmin:qmax, AIC = NA, AICC = NA, BIC =
NA); result.df

for(i in 1:nrow(result.df)){
  print(i)

  p = result.df$val_p[i]
  q = result.df$val_q[i]

  arima.fit = Arima(data, order = c(p, 0, q), xreg = X, include.mean = FALSE)

  m = p+q+2 # p: the number of AR parameters, q: the number of MA parameters
```

```
  result.df$AIC[i] = -2*arima.fit$loglik + 2*m
  result.df$AICC[i] = -2*arima.fit$loglik + 2*m*n/(n-m-1)
  result.df$BIC[i] = -2*arima.fit$loglik + m*log(n)
}

result.df[which.min(result.df$AIC), ]
result.df[which.min(result.df$AICC), ]
result.df[which.min(result.df$BIC), ]

result1 = result.df


# Model Selection by Out-of-Sample Forecasting Errors
arima.fit = NULL

m = 30
N = n - m


pmin = 0; qmin = 0 # change the minimum values
pmax = 8; qmax = 3 # change the maximum values

result.df = expand.grid(val_p = pmin:pmax, val_q = qmin:qmax, err = NA); result.df

for(j in 1:nrow(result.df)){
  print(j)

  p = result.df$val_p[j]
  q = result.df$val_q[j]

  err = numeric(m)

  for(i in 1:m){
    train.idx = 1:(N+i-1)
    arima.fit = Arima(data[train.idx], order = c(p, 0, q), xreg = X[train.idx, ], include.mean = FALSE)

    X.hat = forecast(arima.fit, xreg = t(as.matrix(X[N+i, ])), h = 1)
    err[i] = (data[N+i] - X.hat$mean)^2
  }
  result.df$err[j] = mean(err)
}

result.df[which.min(result.df$err), ]
result2 = result.df
```

```
# Final Regression + ARMA errors model
arma.selected =    Arima(data, order = c(6, 0, 3), xreg = X, include.mean = FALSE) # change the order

tseries::adf.test(arma.selected$residuals, k = 100)


# Diagnostics: Coefficients - Are the coefficients significantly away from zero?
2*(1-pnorm(abs(arma.selected$coef[1:9]/(sqrt(diag(arma.selected$var.coef[1:9, 1:9])))))) < 0.05 #
change the slicing index

arma.selected = Arima(data, order = c(6, 0, 3), xreg = X, include.mean = FALSE, fixed = c(NA, NA,
NA, NA, NA, NA, 0, 0, NA, rep(NA, 14))) # change the fixed values

tseries::adf.test(arma.selected$residuals, k = 100)

arma.selected

# Diagnostics: Residual Plot - No patterns in residual plot, normal QQ Plot, SACF, SPACF?
layout(matrix(c(1, 1, 2, 3, 4, 4, 5, 5), 2, 4, byrow = TRUE))

plot(arma.selected$residuals, type = "l")
title("Residuals - Regression")

plot(arma.selected$fitted, arma.selected$residuals ,xlab = "Fitted Values", ylab = "Residuals")
title("Residuals vs Fitted - Regression")

qqnorm(arma.selected$residuals)
qqline(arma.selected$residuals)

acf.plot = acf(arma.selected$residuals, lag = 100, plot = FALSE)
plot(acf.plot, main = "SACF of the Residuals - Regression")

pacf.plot = pacf(arma.selected$residuals, lag = 100, plot = FALSE)
plot(pacf.plot, main = "SPACF of the Residuals - Regression")


# Diagnostics: Formal Tests - Are the Residuals IID?
itsmr::test(arma.selected$residuals)


# Diagnostics: Test for Normality
shapiro.test(arma.selected$residuals)

library(nortest)
ad.test(arma.selected$residuals)
```

```
cvm.test(arma.selected$residuals)
lillie.test(arma.selected$residuals)

library(tseries)
jarque.bera.test(arma.selected$residuals)



####################
### SARIMA Model ###
####################

layout(matrix(c(1, 1, 2, 3), 2, 2, byrow = TRUE))
plot(diff(data, lag = 12), type = "l"); title("Time Plot") # change the lag value for seasonal differencing
acf.plot = acf(diff(data, lag = 12), lag = 50, plot = FALSE); plot(acf.plot, main = "SACF") # change the lag value for seasonal differencing
pacf.plot = pacf(diff(data, lag = 12), lag = 50, plot = FALSE); plot(pacf.plot, main = "SPACF") # change the lag value for seasonal differencing

layout(matrix(c(1, 1, 2, 3), 2, 2, byrow = TRUE))
plot(diff(diff(data, lag = 12)), type = "l"); title("Time Plot")
acf.plot = acf(diff(diff(data, lag = 12)), lag = 100, plot = FALSE); plot(acf.plot, main = "SACF")
pacf.plot = pacf(diff(diff(data, lag = 12)), lag = 100, plot = FALSE); plot(pacf.plot, main = "SPACF")


layout(matrix(c(1, 1, 2, 3), 2, 2, byrow = TRUE))
plot(diff(diff(diff(data, lag = 12))), type = "l"); title("Time Plot")
acf.plot = acf(diff(diff(diff(data, lag = 12))), lag = 50, plot = FALSE); plot(acf.plot, main = "SACF")
pacf.plot = pacf(diff(diff(diff(data, lag = 12))), lag = 50, plot = FALSE); plot(pacf.plot, main = "SPACF")

sarima.fit = arima(data, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1), period = 12)) # change the order and period

layout(matrix(c(1, 1, 2, 3), 2, 2, byrow = TRUE))
plot(sarima.fit$residuals, type = "l"); title("Time Plot")
acf.plot = acf(sarima.fit$residuals, lag = 50, plot = FALSE); plot(acf.plot, main = "SACF")
pacf.plot = pacf(sarima.fit$residuals, lag = 50, plot = FALSE); plot(pacf.plot, main = "SPACF")



# Model Selection by Information Criteria

sarima.fit = NULL

n = length(data)
```

```
pmin = 0; qmin = 0 # change the minimum value
pmax = 2; qmax = 2 # change the minimum value
Pmin = 0; Qmin = 0 # change the minimum value
Pmax = 2; Qmax = 2 # change the minimum value

result.df = expand.grid(val_p = pmin:pmax, val_q = qmin:qmax, val_P = Pmin:Pmax, val_Q =
Qmin:Qmax, AIC = NA, AICC = NA, BIC = NA); result.df

for(i in 1:nrow(result.df)){
  skip_to_next = FALSE
  print(i)

  p = result.df$val_p[i]
  q = result.df$val_q[i]
  P = result.df$val_P[i]
  Q = result.df$val_Q[i]

  tryCatch(expr = {
    sarima.fit = arima(data, order = c(p, 1, q), seasonal = list(order = c(P, 1, Q), period = 12)) #
change the number of differencing and period
    m = p+q+P+Q+2
    result.df$AIC[i] = -2*sarima.fit$loglik + 2*m
    result.df$AICC[i] = -2*sarima.fit$loglik + 2*m*n/(n-m-1)
    result.df$BIC[i] = -2*sarima.fit$loglik + m*log(n)},
    error = function(e){skip_to_next = TRUE})

  if(skip_to_next == TRUE){
    next
  }
}

result.df[which.min(result.df$AIC), ]
result.df[which.min(result.df$AICC), ]
result.df[which.min(result.df$BIC), ]

result3 = result.df



# Model Selection by Out-of-Sample Forecasting Errors

m = 30
N = n - m

pmin = 0; qmin = 0 # change the minimum value
```

```
pmax = 2; qmax = 2 # change the minimum value
Pmin = 0; Qmin = 0 # change the minimum value
Pmax = 2; Qmax = 2 # change the minimum value

result.df = expand.grid(val_p = pmin:pmax, val_q = qmin:qmax, val_P = Pmin:Pmax, val_Q =
Qmin:Qmax, err = NA); result.df

for(j in 1:nrow(result.df)){
  skip_to_next = FALSE
  print(j)

  p = result.df$val_p[j]
  q = result.df$val_q[j]
  P = result.df$val_P[j]
  Q = result.df$val_Q[j]

  err = numeric(m)

  tryCatch(expr = {
    for(i in 1:m){
      train.idx = 1:(N+i-1)
      # change the number of differencing and period
      sarima.fit = arima(data[train.idx], order = c(p, 1, q), seasonal = list(order = c(P, 1, Q), period =
12), method = c("CSS-ML"))
      X.hat = forecast(sarima.fit, h = 1)$mean
      err[i] = (data[N+i] - X.hat)^2
    }
    result.df$err[j] = mean(err)},
    error = function(e){skip_to_next = TRUE})

  if(skip_to_next == TRUE){
    next
  }
}

result.df[which.min(result.df$err), ]
dplyr::arrange(result.df, err)
result4 = result.df


# FINAL MODEL: SARIMA
# change the order, number of differencing and period
sarima.selected = arima(data, order = c(0, 1, 2), seasonal = list(order = c(2, 1, 2), period = 12), method
= c("CSS-ML"))
tseries::adf.test(sarima.selected$residuals)
```

```
# Are the coefficients significantly away from zero?
2*(1-pnorm(abs(sarima.selected$coef/(sqrt(diag(sarima.selected$var.coef))))) < 0.05

# change the order, number of differencing, period and fixed valued
sarima.selected = arima(data, order = c(0, 1, 2), seasonal = list(order = c(2, 1, 2), period = 12),
                        fixed = c(NA, 0, NA, NA, NA, NA), method = c("CSS-ML"))


# No patterns in residual plot, normal QQ Plot, SACF, SPACF?
par(mfrow = c(2, 2))
plot(sarima.selected$residuals, type = "l"); title("Residual Plot")
qqnorm(sarima.selected$residuals); qqline(sarima.selected$residuals)
acf.plot = acf(sarima.selected$residuals, lag = 200, plot = FALSE); plot(acf.plot, main = "SACF")
pacf.plot = pacf(sarima.selected$residuals, lag = 200, plot = FALSE); plot(pacf.plot, main = "SPACF")

# Formal tests to check if the residuals are IID noise?
itsmr::test(sarima.selected$residuals)

# Test for normality?
shapiro.test(sarima.selected$residuals)

library(nortest)
ad.test(sarima.selected$residuals)
cvm.test(sarima.selected$residuals)
lillie.test(sarima.selected$residuals)

library(tseries)
jarque.bera.test(sarima.selected$residuals)


####################
### Forecasting ###
####################

# Regression + ARMA errors model
detach("package:itsmr")
library(forecast)

n = length(data)
new.const = rep(1, 30)
new.x1 = (n+1):(n+30)
new.x2 = new.x1^2
new.t = 1:30
new.d = 12 # change the seasonal period
new.f1 = 30/new.d
new.f2 = 2*new.f1
```

```
new.f3 = 3*new.f1
new.f4 = 4*new.f1
new.f5 = 5*new.f1
new.f6 = 6*new.f1
new.cos1 = cos(new.f1*2*pi/30*new.t)
new.sin1 = sin(new.f1*2*pi/30*new.t)
new.cos2 = cos(new.f2*2*pi/30*new.t)
new.sin2 = sin(new.f2*2*pi/30*new.t)
new.cos3 = cos(new.f3*2*pi/30*new.t)
new.sin3 = sin(new.f3*2*pi/30*new.t)
new.cos4 = cos(new.f4*2*pi/30*new.t)
new.sin4 = sin(new.f4*2*pi/30*new.t)
new.cos5 = cos(new.f5*2*pi/30*new.t)
new.sin5 = sin(new.f5*2*pi/30*new.t)
new.cos6 = cos(new.f6*2*pi/30*new.t)
new.X = cbind(new.const, new.x1, new.x2, new.cos1, new.sin1, new.cos2, new.sin2,
                    new.cos3, new.sin3, new.cos4, new.sin4, new.cos5, new.sin5, new.cos6) # change the
columns
colnames(new.X) =
c("col.const","col.x1","col.x2","col.cos1","col.sin1","col.cos2","col.sin2","col.cos3","col.sin3","col.cos4",
                          "col.sin4","col.cos5","col.sin5","col.cos6")

par(mfrow = c(1, 1))
plot(forecast(arma.selected, xreg = new.X, h = 30))
forecast(arma.selected, xreg = new.X[1:4, ], h = 4) # change the slicing index and the value of h

# SARIMA model
par(mfrow = c(1, 1))
plot(forecast(sarima.selected, h = 30))
forecast(sarima.selected, h = 4) # change the value of h
```